



Adriano Soares Koshiyama

**GPFIS: Um Sistema Fuzzy-Genético Genérico
baseado em Programação Genética**

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Engenharia Elétrica do Departamento de Engenharia Elétrica da PUC-Rio

Orientador : Prof. Marley Maria Bernardes Rebuszi Vellasco
Co-Orientador: Prof. Ricardo Tanscheit

Rio de Janeiro
Abril de 2014



Adriano Soares Koshiyama

GPFIS: Um Sistema Fuzzy-Genético Genérico baseado em Programação Genética

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Engenharia Elétrica do Departamento de Engenharia Elétrica do Centro Técnico Científico da PUC-Rio. Aprovada pela comissão examinadora abaixo assinada.

Prof. Marley Maria Bernardes Rebuzzi Vellasco

Orientador

Departamento de Engenharia Elétrica — PUC-Rio

Prof. Ricardo Tanscheit

Co-Orientador

Departamento de Engenharia Elétrica — PUC-Rio

Prof. Douglas Mota Dias

Departamento de Engenharia Elétrica – PUC-Rio

Prof. Jorge Luís Machado do Amaral

Faculdade de Engenharia – UERJ

Prof. Fernando Antonio Campos Gomide

Departamento de Engenharia de Computação e Automação –
UNICAMP

Prof. José Eugenio Leal

Coordenador do Centro Técnico Científico — PUC-Rio

Rio de Janeiro, 3 de Abril de 2014

Todos os direitos reservados. Proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Adriano Soares Koshiyama

Graduou-se em Ciências Econômicas pela Universidade Federal Rural do Rio de Janeiro. Durante a graduação trabalhou principalmente com Estatística Aplicada, Economia e Apicultura em projetos para revitalização e fortalecimento de produtores no Estado do Rio de Janeiro. No mestrado envolveu-se com a área de Métodos de Apoio à Decisão, em destaque a Inteligência Computacional, trabalhando em projetos de Pesquisa e Desenvolvimento para empresas públicas e privadas. Sua área de interesse é métodos quantitativos de apoio à decisão.

Ficha Catalográfica

Koshiyama, Adriano Soares

GPFIS: Um Sistema Fuzzy-Genético Genérico baseado em Programação Genética / Adriano Soares Koshiyama; orientador: Marley Maria Bernardes Rebuzzi Vellasco; co-orientador: Ricardo Tanscheit. — Rio de Janeiro : PUC-Rio, Departamento de Engenharia Elétrica, 2014.

v., 222 f: il. ; 29,7 cm

1. Dissertação (Mestrado em Engenharia Elétrica) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica.

Inclui referências bibliográficas.

1. Engenharia Elétrica – Tese. 2. Sistemas Fuzzy-Genéticos. 3. Programação Genética. 4. Classificação. 5. Previsão. 6. Regressão. 7. Controle. I. Vellasco, Marley Maria Bernardes Rebuzzi Vellasco. II. Tanscheit, Ricardo. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. IV. Título.

CDD: 621.3

Agradecimentos

Ao povo brasileiro, ao CNPq e a PUC-Rio pelos auxílios e estrutura concedida, sem os quais este trabalho não poderia ter sido realizado.

A minha família, em destaque meus irmãos - Wladimir e Kallyah - e pais - Adriano e Sayonara - por todos os momentos abdicados da minha presença, devido a este manuscrito e demais projetos. Agradeço e saibam que temos uma ligação de afeto para além do infinito.

A minha orientadora, Professora Marley Vellasco, que sempre me estimulou a produzir cientificamente, me incentivando nas inúmeras reuniões e revisões de texto, além da oportunidade de trabalhar em projetos. Também ao meu co-orientador, Professor Ricardo Tanscheit, pelas excelentes conversas e histórias compartilhadas e por ter me permitido absorver um pouco do seu método “minimalista”. A ambos o meu maior respeito e admiração.

Aos amigos que fiz durante a execução das pesquisas no LIRA, como Bernardo Miranda, Camilo Rueda, Leonardo Forero e Karla Figueiredo. A estes agradeço o conhecimento compartilhado e o ambiente de trabalho favorável.

Aos amigos que se formaram durante o mestrado: Cristian Muñoz, Carol Lima, Carol Abreu, Eric Praxedes, Guilherme Stratchan, Manoela Koehler, Carlos Valle, Carol Goloni, Ian Herszterg, Henrique Helfer, Rogério Póvoa, Sandra Canton, Cassio Almeida, Rouhallah Ebrahimi, Marco Silva, Harold Mello, Bianca Amaral, Bianca Gonçalves, Alexandre Figueira, Alexandre Ashade e Waldir Lobão. Agradeço pelos momentos que passamos juntos. Também aos meus amigos de fora da PUC, como Natália Cunha, Max Monteiro, Dyuliano Pantutzo, Ronaldo Pedro, Mayara Rodrigues, Juliana Brito, Carol Almeida, Carol Rosa e Marcelo de Souza. Quero sempre poder estar entre vocês, compartilhando o que há de melhor na vida.

Também a pessoas com quem pude mais aprender do que transmitir, e com as quais formou-se uma amizade sólida. Iury Steiner, João Zanni, Pedro Milet e Douglas Dias, a vocês obrigado por todas as risadas e experiência passada. Ainda, a Tatiana Escovedo e Rubens Nascimento que me adotaram como filho e partilharam comigo trabalhos, pizzas e viagens.

Aos professores Marco Pacheco, Wagner Tassinari, Álvaro Veiga, Carlos Tomei e Maria Lorenzon, dos quais pude absorver, cada um a sua maneira, a forma de lidar e transmitir a ciência, sempre com rigor e crítica, e me permitiram também vislumbrar a aplicação desta fora dos muros da universidade. *In memoriam* a Padre Libanio pela amizade partilhada e exemplo de vida.

Por fim, ao corpo de funcionários do Departamento de Engenharia Elétrica, notavelmente Alcina, Mara, Márcia, Isnarde e Miriam pelo apoio.

Resumo

Koshiyama, Adriano Soares; Vellasco, Marley Maria Bernardes Rebuzzi Vellasco; Tanscheit, Ricardo. **GPFIS: Um Sistema Fuzzy-Genético Genérico baseado em Programação Genética**. Rio de Janeiro, 2014. 222p. Dissertação de Mestrado — Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Sistemas Fuzzy-Genéticos compreendem uma área que une Sistemas de Inferência Fuzzy e Meta-Heurísticas prevalentes nos conceitos de seleção natural e recombinação genética. Esta é de grande interesse para a comunidade científica, pois propicia a descoberta de conhecimento em áreas onde a compreensão do fenômeno em estudo é exíguo, além de servir de apoio à decisão para gestores público-privados. O objetivo desta dissertação é desenvolver um novo Sistema Fuzzy-Genético Genérico, denominado Genetic Programming Fuzzy Inference System (GPFIS). O principal aspecto do modelo GPFIS são as componentes do seu processo de Inferência Fuzzy. Esta estrutura é composta em sua base pela Programação Genética Multigênica e pretende: (i) possibilitar o uso de operadores de agregação, negação e modificadores linguísticos de forma simplificada; (ii) empregar heurísticas de definição do consequente mais apropriado para uma parte antecedente; e (iii) usar um procedimento de defuzzificação, que induzido pela forma de fuzzificação e sobre determinadas condições, pode proporcionar uma estimativa mais acurada. Todas estas são contribuições que podem ser estendidas a outros Sistemas Fuzzy-Genéticos. Para demonstrar o aspecto genérico, o desempenho e a importância de cada componente para o modelo proposto, são formuladas uma série de investigações empíricas. Cada investigação compreende um tipo de problema: Classificação, Previsão, Regressão e Controle. Para cada problema, a melhor configuração obtida durante as investigações é usada no modelo GPFIS e os resultados são comparados com os de outros Sistemas Fuzzy-Genéticos e modelos presentes na literatura. Por fim, para cada problema é apresentada uma aplicação detalhada do modelo GPFIS em um caso real.

Palavras-chave

Sistemas Fuzzy-Genéticos ; Programação Genética ; Classificação ; Previsão ; Regressão ; Controle.

Abstract

Koshiyama, Adriano Soares; Vellasco, Marley Maria Bernardes Re-buzzi Vellasco (advisor); Tanscheit, Ricardo. **GPFIS: A Generic Genetic-Fuzzy System based on Genetic Programming**. Rio de Janeiro, 2014. 222p. Msc. Dissertation — Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Genetic Fuzzy Systems constitute an area that brings together Fuzzy Inference Systems and Meta-Heuristics that are often related to natural selection and genetic recombination. This area attracts great interest from the scientific community, due to the knowledge discovery capability in situations where the comprehension of the phenomenon under analysis is lacking. It can also provides support to decision makers. This dissertation aims at developing a new Generic Genetic Fuzzy System, called Genetic Programming Fuzzy Inference System (GPFIS). The main aspects of GPFIS model are the components which are part of its Fuzzy Inference procedure. This structure is basically composed of Multi-Gene Genetic Programming and intends to: *(i)* apply aggregation operators, negation and linguistic hedges in a simple manner; *(ii)* make use of heuristics to define the consequent term most appropriate to the antecedent part; *(iii)* employ a defuzzification procedure that, driven by the fuzzification step and under some assumptions, can provide a most accurate estimate. All these features are contributions that can be extended to other Genetic Fuzzy Systems. In order to demonstrate the general aspect of GPFIS, its performance and the relevance of each of its components, several investigations have been performed. They deal with Classification, Forecasting, Regression and Control problems. By using the best configuration obtained for each of the four problems, results are compared to other Genetic Fuzzy Systems and models in the literature. Finally, applications of GPFIS actual cases in each category is reported.

Keywords

Genetic-Fuzzy Systems ; Genetic Programming ; Classification ; Forecasting ; Regression ; Control.

Sumário

1	Introdução	16
1.1	Motivação	17
1.2	Objetivos	21
1.3	Contribuições	22
1.4	Descrição e Organização da Dissertação	22
2	Sistemas Fuzzy-Genéticos	24
2.1	Introdução	24
2.2	Linhas Gerais	25
2.3	Sistemas Fuzzy-Genéticos para Base de Regras	32
2.4	Principais Aplicações de Sistemas Fuzzy-Genéticos	40
2.5	Resumo	47
3	Programação Genética	49
3.1	Principais Conceitos	49
3.2	Programação Genética Multigênica	56
3.3	Resumo	59
4	Modelo Genetic Programming Fuzzy Inference System (GPFIS)	61
4.1	Definições e Notações	61
4.2	Arquitetura Básica	63
4.3	Resumo	90
5	Estudos de Casos	91
5.1	Regressão	92
5.2	Classificação	119
5.3	Previsão	142
5.4	Controle	164
5.5	Resumo	174
6	Conclusões	175
	Referências Bibliográficas	178
A	Definições e Operadores Fuzzy	200
A.1	Operadores	201
B	Demonstrações do Capítulo 4	206
B.1	Cálculo dos coeficientes para cada $\mu_{A_{d(k)}}(\mathbf{x}_i)$	206
B.2	Cálculo dos coeficientes da Matriz β	208
B.3	Prova da Desigualdade do DMA	209
C	Testes Estatísticos Não-Paramétricos	211

C.1	Teste do Sinal	213
C.2	Teste de Wilcoxon	215
C.3	Teste de Friedman	216
C.4	Teste de Holm	219

Lista de figuras

1.1	Exemplo de regras fuzzy geradas por Algoritmos Genéticos e Programação Genética, respectivamente.	18
1.2	Exemplo de base de regras fuzzy decodificadas em Programação Genética e Algoritmos Genéticos, respectivamente.	19
2.1	Diagrama genérico de um SFG.	25
2.2	Diagrama das zonas de ação da MHG na estrutura de um SFG.	27
2.3	Diagrama com as possibilidades de atuação dos SFGs para Ajuste Genético de Parâmetros.	28
2.4	Diferentes possibilidades de ajuste das funções de pertinência.	28
2.5	Exemplo de solução usando codificação binária de um AG para seleção de regras pertencentes a uma BR.	29
2.6	Diagrama com as possibilidades de atuação de um SFG para Descoberta de Conhecimento.	30
2.7	Exemplo de aprendizado de granularidade.	31
2.8	Exemplo de atuação conjunta: escolha de granularidade e deslocamento das funções de pertinência.	31
2.9	Visão global das áreas de atuação de um SFG.	32
2.10	Diagrama do processo de sintetização de um SFGBR do tipo Pittsburgh.	33
2.11	Diagrama do processo de sintetização de um SFGBR do tipo Michigan.	36
2.12	Diagrama do processo de sintetização de um SFGBR do tipo IRL.	37
2.13	Diagrama do processo de sintetização do tipo GCCL.	39
3.1	Representação do programa $x^2 + y$: (a) Representação sob a forma de árvore; (b) forma pré-fixada.	51
3.2	Elaboração de uma árvore pelo método Full, considerando a altura máxima como 2 ($t = \text{iteração}$). Reproduzido de Poli et al. [171].	52
3.3	Elaboração de uma árvore pelo método Grow, considerando a altura máxima como 2 ($t = \text{iteração}$). Reproduzido de Poli et al. [171].	53
3.4	Exemplo de função $x^2 + x + 1$. Reproduzido de Poli et al. [171].	53
3.5	Soluções iniciais.	54
3.6	Exemplo de aproximação de cada solução a função alvo. Reproduzido de Poli et al. [171].	54
3.7	Operador genético de Reprodução Direta para PG.	55
3.8	Operador genético de Cruzamento para PG.	55
3.9	Operador genético de Mutação para PG.	56
3.10	Operador genético de Encapsulamento para PG.	56
3.11	Exemplo de um indivíduo multigênico.	57
3.12	Operação de cruzamento de baixo nível.	58
3.13	Operação de mutação.	58
3.14	Operação de cruzamento de alto nível.	59

4.1	Exemplo de função de pertinência triangular.	62
4.2	Diagrama genérico das etapas de elaboração do modelo GPFIS.	64
4.3	Partição uniforme das funções de pertinência normais para as variáveis X_j e Y .	65
4.4	Alternativas de granularidade para as variáveis.	66
4.5	Exemplo de granularidade para Y .	67
4.6	Funções de pertinência relevantes, devido à propriedade da 2-justaposição.	68
4.7	Diagrama do processo de Inferência do modelo GPFIS.	71
4.8	Exemplo de premissas formadas por um indivíduo da PGMG.	72
4.9	Exemplo de Particionamento pelo método da Divisão Uniforme.	73
4.10	Representação exemplificada dos $\mu_{B_k}(Y)$ e $\mu_{A_d}(X_1, X_2)$ no espaço $[0, 1]^n$ dimensional.	77
4.11	Ponto de vista geométrico do método da Máxima Credibilidade.	79
4.12	Diagrama do processo de Inferência: Formulação-Particionamento-Agregação.	80
4.13	Exemplificação dos métodos de Defuzzificação.	85
4.14	Diagrama do Processo: Inferência-Defuzzificação/Decisão-Avaliação.	86
4.15	Exemplo de duas soluções presentes na rotina do modelo GPFIS.	88
4.16	Exemplo de aplicação do operador de mutação.	89
4.17	Exemplo de aplicação do operador de cruzamento de baixo nível.	89
4.18	Exemplo de aplicação do operador de cruzamento de alto nível.	89
5.1	Formato e número das funções de pertinência usadas no GPFIS.	96
5.2	Percentual de casos em que os modelos obtiveram os melhores resultados em acurácia.	104
5.3	Percentual de casos em que os modelos obtiveram os melhores resultados na relação número médio de regras e antecedentes por regra.	107
5.4	Exemplo do formato final das funções de pertinência do modelo FS-MOGFS+TUN para a base AIL.	107
5.5	Exemplo de gráfico a ser analisado pelos sujeitos da pesquisa.	110
5.6	Histograma das observações de tempo antes e após aplicar o logaritmo natural.	112
5.7	Funções de pertinência usadas no modelo GPFIS e no WM-R.	113
5.8	Exemplo de solução proposta pela AR. Os nós transparentes foram suprimidos para facilitar a visualização.	115
5.9	Relação entre os valores preditos e observados dos modelos AR (a) e GPFIS (b).	115
5.10	Grau de pertinência observados para dois e três conjuntos fuzzy.	117
5.11	Exemplo de árvore empregada em Alcalá-Fdez et al. [9].	120
5.12	Exemplo de solução do modelo de Muni e Pal [157].	121
5.13	Funções de pertinência usadas no modelo GPFIS.	125
5.14	Relacionamento entre IP e IS e os tipos de rochas.	137
5.15	Funções de pertinência usadas.	138
5.16	Classes preditas pelo modelo GPFIS.	140
5.17	Funções de pertinência usadas no modelo GPFIS.	146
5.18	Funções de pertinência usadas no modelo GPFIS.	158

5.19	a) Taxa de câmbio R\$/US\$ do período janeiro de 1999 a março de 2011. b) Tendência. c) Sazonalidade. d) Ruído resultante da decomposição estrutural.	160
5.20	Valores observados e preditos pelo modelo GPFIS para a fase in-sample.	161
5.21	Valores observados, preditos e previstos (após a linha vermelha) pelo modelo GPFIS para a fase out-of-sample.	162
5.22	Ilustração do problema Cart-Centering.	165
5.23	Funções de pertinência usadas no modelo GPFIS.	167
5.24	Posição inicial e final do melhor indivíduo nas execuções do GPFIS, usando a configuração Product+RQ:Média:Altura Média dos Máximos.	169
5.25	Superfície de resposta do melhor indivíduo para o Cart-Centering, usando diferentes métodos de defuzzificação: a) Altura, b) Altura Média dos Máximos.	170
5.26	Ilustração do Pêndulo Invertido.	170
5.27	Funções de pertinência usadas no modelo GPFIS.	173
5.28	Solução elaborada pelo melhor indivíduo nas execuções do GPFIS.	173
A.1	Operação de concentração.	204
A.2	Operação de dilatação.	204
C.1	Exemplo de distribuição de probabilidade.	219

Lista de tabelas

2.1	Síntese das principais codificações para SFGBR.	40
3.1	Parâmetros de controle de um algoritmo de PG.	50
4.1	Exemplo da matriz de dados.	62
4.2	Conjuntos e Operadores Fuzzy usados para formar as soluções da Figura 4.8.	72
4.3	Exemplo da matriz de dados.	74
4.4	Exemplo de matriz resultante para Classificação.	74
4.5	Exemplo de matriz resultante para Previsão/Regressão.	76
5.1	Bases de dados consideradas nos experimentos efetuados.	94
5.2	Delineamento experimental sequencial para o modelo GPFIS.	95
5.3	Principais configurações do modelo GPFIS.	96
5.4	Resultados do Experimento 1 para o conjunto de operadores fuzzy. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente.	97
5.5	Resultados do teste de Friedman e Holm para o Experimento 1.	98
5.6	Resultados do Experimento 2 para o conjunto de operadores de agregação. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente.	99
5.7	Regras geradas por cada abordagem operador de agregação.	100
5.8	Resultados do teste de Friedman e Holm para o Experimento 2.	100
5.9	Proporção adicional de tempo tomado pela abordagem MQR com relação as demais operações de agregação.	100
5.10	Resultados do Experimento 3 para o conjunto de métodos de particionamento. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente.	101
5.11	Regras geradas por cada método de particionamento.	101
5.12	Proporção adicional de tempo tomado pelas demais abordagens com respeito ao método de Divisão Uniforme.	102
5.13	Resultados do teste de Friedman e Holm para o Experimento 3.	102
5.14	Resultado final do modelo GPFIS e dos demais SFGs. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente. R/A - Média de Regras e Antecedentes.	105
5.15	Resultados do teste de Friedman e Holm para a comparação entre os SFGs, considerando apenas as bases de dados de baixa dimensionalidade.	106
5.16	Exemplo de frases e dos principais fatores considerados no estudo.	109

5.17 Modelos para regressão e parâmetros usados.	113
5.18 Principais configurações dos modelos baseados em Programação Genética.	114
5.19 Principais resultados dos modelos usados.	114
5.20 Variações no número de funções de pertinência e seu impacto nos Sistemas Fuzzy usados.	116
5.21 Alteração da função de avaliação e seus impactos nos modelos evolutivos.	117
5.22 Base de regras fuzzy.	117
5.23 Bases de dados consideradas nos experimentos efetuados.	123
5.24 Delineamento experimental sequencial para o modelo GPFIS.	124
5.25 Principais configurações do modelo GPFIS.	125
5.26 Resultados em termos de acurácia (%) do Experimento 1 para o conjunto de operadores fuzzy.	126
5.27 Resultados do teste de Friedman e Holm para o Experimento 1.	126
5.28 Resultados em termos de acurácia (%) do Experimento 2 para o conjunto de operadores de agregação.	127
5.29 Resultados do teste de Friedman e Holm para o Experimento 2.	128
5.30 Proporção adicional de tempo tomado pela abordagem MQR com relação as demais operações de agregação.	128
5.31 Regras geradas por cada abordagem no Experimento 2 para as bases de dados classificação binária (BIN) e de múltiplas classes (MULT).	129
5.32 Resultados em termos de acurácia (%) do Experimento 3 para o conjunto de métodos de particionamento.	129
5.33 Resultados do teste de Friedman e Holm para o Experimento 3.	130
5.34 Proporção adicional de tempo tomado pelas demais abordagens com respeito ao método de Div-Unif.	130
5.35 Regras geradas por cada abordagem no Experimento 3 para as bases de dados classificação binária (BIN) e de múltiplas classes (MULT).	131
5.36 Resultados em termos de acurácia (%) na fase de teste para todos os modelos comparados.	133
5.37 Resultados do teste de Friedman e Holm para a comparação entre os SFGs.	133
5.38 Comparação entre número médio de regras e antecedentes.	134
5.39 Tipos de litologias e agrupamentos usados no poço em análise.	136
5.40 Classificadores e parâmetros usados.	138
5.41 Principais configurações dos modelos baseados em Programação Genética.	139
5.42 Base de regras fuzzy do melhor indivíduo do modelo GPFIS.	140
5.43 Acurácia média na fase de teste das 3 execuções por pasta de validação cruzada.	141
5.44 Precisão média na fase de teste das 3 execuções por pasta de validação cruzada.	141
5.45 Principais detalhes da competição NN3.	142

5.46	Dez primeiros melhores colocados na competição NN3, considerando os resultados para as 111 séries temporais.	143
5.47	Delineamento experimental sequencial para o modelo GPFIS.	144
5.48	Principais configurações do modelo GPFIS.	146
5.49	Resultados do Experimento 1 para o conjunto de operadores fuzzy.	149
5.50	Resultados do Experimento 2 para o conjunto de operadores de agregação.	150
5.51	Regras geradas por cada abordagem no Experimento 2.	151
5.52	Proporção adicional de tempo tomado pela abordagem MQR com relação as demais operações de agregação.	151
5.53	Resultados do Experimento 3 para o conjunto de métodos de particionamento.	152
5.54	Proporção adicional de tempo tomado pelas demais abordagens com respeito ao método de Divisão Uniforme.	153
5.55	Regras geradas por cada abordagem no Experimento 3.	153
5.56	Dez primeiros na competição NN3, considerando os resultados para as 111 séries temporais.	154
5.57	Dez primeiros colocados na competição NN3, considerando os resultados referentes as últimas 11 séries temporais.	154
5.58	SMAPE e RMSE obtidos na fase out-of-sample pelo modelo GPFIS em cada série temporal da NN3.	155
5.59	Previsores e parâmetros usados.	158
5.60	Principais configurações dos modelos baseados em Programação Genética.	159
5.61	Principais resultados dos previsores na fase in-sample.	160
5.62	Principais resultados dos previsores na fase out-of-sample.	161
5.63	Base de Regras Fuzzy com representação tradicional.	162
5.64	Base de Regras Fuzzy com representação por ligação de estados linguísticos.	163
5.65	Domínio e valor das variáveis e parâmetros para os testes realizados.	165
5.66	Delineamento experimental fatorial para o modelo GPFIS.	166
5.67	Principais configurações do modelo GPFIS.	167
5.68	Resultados do modelo GPFIS para o Cart-Centering com operação de agregação pelo Máximo.	168
5.69	Resultados do modelo GPFIS para o Cart-Centering com operação de agregação pela Média Aritmética.	169
5.70	Domínio das variáveis e valor dos parâmetros para os testes efetuados.	172
5.71	Principais configurações do modelo GPFIS.	172
A.1	Exemplo de base de regras fuzzy.	200
C.1	Exemplo de resultados para os testes estatísticos em análise.	212
C.2	Exemplo de tabela de resultados para o teste de Friedman.	217
C.3	Resultados do teste de Friedman e Holm para o Experimento.	221

*Assim parecia, digo eu, mas não era. Era meu antagonista – era Wilson que então estava de pé diante de mim na agonia da dissolução. Sua máscara e sua capa jaziam onde ele as havia jogado, no chão. Nenhum fio em todo o seu traje – nenhuma linha em todas as feições marcadas e singulares de seu rosto que não fosse, até mesmo em sua mais absoluta identidade, **o meu próprio!***

Era Wilson; mas ele não falava mais sussurrando, e eu podia ter imaginado que eu mesmo estava falando enquanto ele dizia:

“Venceste, e eu me rendo. Porém, de agora em diante, também está morto – morto para o Mundo, para o Céu e para a Esperança! Em mim é que existias – e, em minha morte, vê, por esta imagem, que é a tua própria quão irremediavelmente assassinaste a ti mesmo .”

Edgar Alan Poe, *Histórias Extraordinárias - William Wilson.*

1

Introdução

A Lógica Fuzzy surgiu na década de 60 com o trabalho pioneiro de Zadeh [214]. Apesar de um início conturbado, conforme descrito por Zadeh [215], com dificuldade de aceitação por parte de um grupo principalmente enraizado na teoria da probabilidade, esta conseguiu crescer e auferir frutos nas mais diversas áreas de aplicação. Um desses foi o Sistema de Inferência Fuzzy proposto por Mamdani [145] no início da década de 70, com a proposta de gerar um controle fuzzy a partir de parâmetros extraídos de operadores humanos da planta.

Este início virtuoso na área de controle possibilitou a propagação do conceito de Sistema de Inferência Fuzzy para novas áreas de aplicação [125, 146, 209], assim como a geração de novos Sistemas de Inferência Fuzzy [194, 204]. A afirmação da metodologia proporcionou que demandas sofisticadas e complexas tomassem conta do espectro das aplicações de tais sistemas. Contudo, ao mesmo tempo que crescia a busca por soluções, especialistas no problema em análise tornavam-se cada vez mais exíguos, seja por questões humanas ou financeiras.

O interessante é que, no mesmo período, com o aprimoramento dos sistemas de gerência e tecnologia da informação, um grande volume de informações foi estocado em bancos de dados nas diferentes esferas públicas e privadas. Os métodos estatísticos foram os primeiros a aproveitarem esse enorme fluxo de informações, servindo estas como fomento para estimação dos modelos e inferências. Entretanto, a natureza restritiva e a dificuldade de interpretação dos resultados de algumas técnicas são ainda o principal empecilho para o uso generalizado dos métodos estatísticos.

Nesse mesmo contexto de disponibilidade de bancos de dados, no início da década de 90 surgiram os primeiros Sistemas de Inferência Fuzzy Híbridos, em especial os Neuro-Fuzzy [121, 158] e os Fuzzy-Genéticos [35, 114] como uma alternativa viável para modelar fenômenos antes concebidos apenas pelo olhar probabilístico. Ambos utilizam a informação sedimentada no banco de dados como a figura do especialista e, a partir disso, ajustam automaticamente os parâmetros do Sistema de Inferência Fuzzy. Por fim, com o avanço dos sistemas

de gerenciamento eletrônico das plantas industriais, também foi possível acoplar Controladores Fuzzy adaptativos, como por exemplo o NEFCON (Neural Fuzzy Controller), proposto por Nauck e Kruse [158].

A partir desses modelos, a presença do especialista não era mais vital na elaboração do Sistema de Inferência Fuzzy. Esse fato viabilizou o uso generalizado desses sistemas para diferentes fins: reconhecimento de padrões, previsão de séries, controle, etc. A próxima seção dispõe a motivação pela concepção de novos modelos na área de Sistemas Fuzzy-Genéticos. Para tanto, inicia-se com a comparação entre Sistemas Neuro-Fuzzy e Fuzzy-Genéticos. Após são efetuadas algumas críticas concernentes à abordagem Fuzzy-Genética tradicional e aponta-se a necessidade de novos modelos na área.

1.1

Motivação

1.1.1

Sistemas Fuzzy-Genéticos

Na área de Sistemas de Inferência Fuzzy (SIFs) Híbridos há duas principais abordagens: Sistemas Fuzzy-Genéticos (SFGs) e Sistemas Neuro-Fuzzy (SNFs). Os SNFs são atrativos devido às características intrínsecas das Redes Neurais: ajuste automático dos parâmetros e aproximação universal [1, 158]. Estes dois elementos são primordiais para a garantia e formação de um SIF com acurácia, adaptatividade e robustez. Contudo, os SFGs são mais flexíveis que os SNFs, pois possibilitam ir além da maximização da acurácia com a inclusão de critérios mais subjetivos como: minimização do número de regras e termos antecedentes, inserção de restrições no formato e sobreposição das funções de pertinência, escolha dos operadores fuzzy, etc. [49, 169]. Além disso, o custo de implementação é baixo, devido ao menor grau de hibridização entre as técnicas. Com isto, basta a existência de um SIF e de uma Meta-Heurística Evolutiva (Algoritmo Genético, por exemplo), que, quando acoplados formam um SFG. Esta maleabilidade e praticidade proporcionada pela meta-heurística é o grande trunfo dos SFGs.

1.1.2

Sistemas Fuzzy-Genéticos baseados em Programação Genética

Na área de SFGs imperam as abordagens que usam Algoritmos Genéticos (AG) [48, 98]. Isto ocorre por motivações históricas (surgiram na década de 70) e práticas (diversas implementações disponíveis, investigação detalhada da sensibilidade do método aos seus parâmetros, farta literatura, etc.). O maior

volume de trabalhos reside na área de Controladores Fuzzy e Formação de Bases de Regras Fuzzy para reconhecimento de padrões [48, 98].

De modo geral, o SFG busca um conjunto de regras compostas por elementos antecedentes e consequentes, que visam a atingir um objetivo fixado pelo usuário. Suponha um problema composto por 10 variáveis de entrada (X_1, X_2, \dots, X_{10}) e uma única saída de (Y). A Figura 1.1 dispõe uma representação típica de uma regra do ponto de vista de AG e Programação Genética (PG).

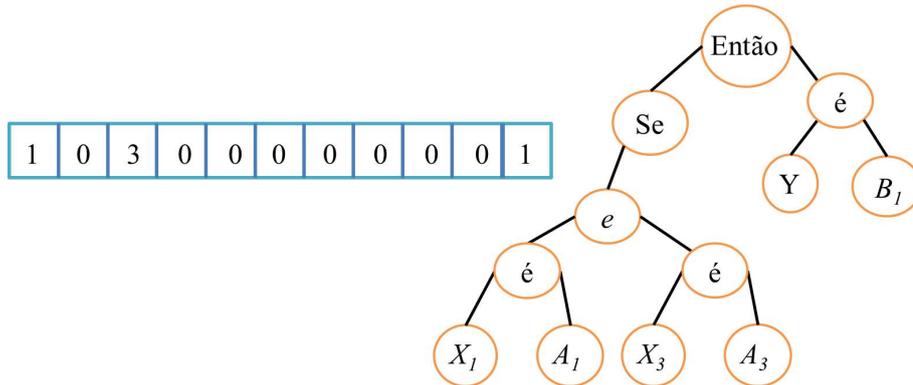


Figura 1.1: Exemplo de regras fuzzy geradas por Algoritmos Genéticos e Programação Genética, respectivamente.

Cada representação expressa uma regra fuzzy lida como: “Se X_1 é A_1 e X_3 é A_3 , então Y é B_1 ”. A diferença entre as abordagens é evidente: os elementos zerados no cromossoma do AG não são necessários na representação induzida pela PG. Agora, suponha uma base de regras conforme Figura 1.2

O primeiro detalhe reside na codificação da solução em AG. Tradicionalmente o cromossoma em AG tem tamanho fixo [151]. Isto é um problema, devido à típica natureza dinâmica de uma base de regras fuzzy. As abordagens que permitem um tamanho variado necessitam da inclusão de novos operadores genéticos e restrições [87, 151]. No caso da PG tal questão já está solucionada pela própria natureza do método. Ainda, apesar do AG revelar valores inteiros em sua codificação, este tipo não é usado com frequência, mas sim a codificação binária [48], devido a maior facilidade de aplicação de operadores, implementações disponíveis, etc., mas que obviamente ampliam a sobrecarga computacional.

Quando o tamanho da base de regras é grande, ambas as abordagens sofrem com a dimensionalidade, porém a complexidade da PG está sempre cotada superiormente pela do AG. Portanto, a vantagem da abordagem por PG é a possibilidade de manipular mais facilmente problemas que requerem uma natureza dinâmica da codificação, como é o caso de uma base de regras

como algumas das principais desvantagens da PG. Este tipo de abordagem de um indivíduo ser uma base de regras fuzzy é conhecido na literatura como abordagem Pittsburgh [49]. Quando um indivíduo incorpora uma regra e a população descreve uma base de regras, este delineamento é denominado de Michigan [49]. Ambos são discutidos com maiores detalhes no Capítulo 2.

A solução encontrada por diversos autores na área foi: incluir novos operadores de recombinação e aderir à abordagem Michigan [28, 89, 112, 142]. O uso de mais operadores visa a ampliar as possibilidades de encontrar uma boa solução, enquanto que a abordagem Michigan tem o intuito de reduzir o tempo de máquina, tornar a aplicação dos operadores de recombinação mais efetiva e ampliar a diversidade das regras. Entretanto, diversos estudos advogam pela maior eficiência da abordagem Pittsburgh, devido a sua relativa facilidade de implementação, convergência a resultados finais superiores e de menor dispersão, tanto que alguns autores costumam usá-la como parte final da rotina de um SFG [111, 117, 118].

Outro fator importante está na própria ideia de formação da regra. A Figura 1.1 exibe uma típica regra formada por termos antecedentes (“Se X_1 é A_1 e X_3 é A_3 ”) e consequente (“Então Y é B_1 ”). A questão é: será que o consequente gerado pela PG é o mais apropriado para o termo antecedente? Esta pergunta é relevante no sentido de que bons antecedentes de regras, com suporte relevante na base de dados, são desperdiçados ao longo do processo de sintetização da base de regras fuzzy.

Cabe ressaltar a exiguidade de outros modelos de PG existentes na literatura, tais como a PG Linear, PG Cartesiana, PG com Inspiração Quântica, etc. [67, 171], [200, 202, 213], como também a baixa utilização de outros tipos de operadores da Lógica Fuzzy (negação, concentração/dilatação, etc.) para a geração de regras fuzzy, devido à dificuldade de decodificar tais operações matemáticas [109, 141].

Todas essas críticas denotam que é premente a busca por novas abordagens em SFGs, principalmente as baseadas em PG. O modelo proposto neste trabalho, denominado de Genetic Programming Fuzzy Inference System (GPFIS) aponta para uma nova concepção de um SFG. O modelo GPFIS tem o intuito de ser genérico, cuja aplicabilidade nos mais diversos problemas (Classificação, Previsão, Regressão e Controle) demande as condições de regularidade de qualquer outro SFG. Esta característica de generalidade cobre outra carência das abordagens atuais em SFGs, que se concentram no problema de classificação. Ainda, o presente trabalho pretende sanar a maioria das deficiências exibidas anteriormente, propiciando em linhas gerais:

- Utilização da base fornecida pela Programação Genética Multigênica (PGMG), uma generalização da PG clássica, com aproveitamento de todo o seu arcabouço, características e vantagens para elaborar um SFG do tipo Pittsburgh.
- Criação de métricas e mecanismos para avaliar o melhor consequente para uma termo antecedente, além da adoção de mecanismos de redução de complexidade da bases de regras.
- Oferta ao usuário da inclusão de operadores linguísticos e diferentes opções de agregação das regras fuzzy geradas, para que ele possa decidir o nível de acurácia/interpretabilidade da solução gerada.

O próximo tópico aborda os objetivos primários e específicos que esta dissertação almeja atingir.

1.2 Objetivos

1.2.1 Objetivo Primário

O objetivo deste trabalho é desenvolver o modelo Genetic Programming Fuzzy Inference System (GPFIS), um SFG genérico, acurado e interpretável. O princípio norteador é proporcionar ao usuário uma maior flexibilidade, como o uso de operadores de negação, dilatação/contração, agregação, entre outras ferramentas, e com isto propiciar um modelo robusto, adequado ao problema em análise.

1.2.2 Objetivos Secundários

A partir de uma Revisão Bibliográfica concisa para verificar o estado da arte na área de SFGs, será possível:

1. Enquadrar o modelo GPFIS no conjunto de SFGs existentes (terminologias, espectro, etc.).
2. Comparar os resultados do modelo GPFIS com os demais SFGs e outros modelos, com respeito à gama de aplicações envolvidas: Classificação, Previsão, Regressão e Controle.

Constituídos esses alvos, a próxima seção descreve as principais contribuições geradas por esta dissertação.

1.3

Contribuições

A principal contribuição desta dissertação é a proposição e desenvolvimento do novo modelo GPFIS. Os resultados apresentados nos estudos de caso também são contribuições, assim como a revisão bibliográfica em SFGs. Com o desenrolar da dissertação, surgiram outros tópicos de interesse:

- Uso de operadores da Lógica Fuzzy pouco visados na literatura de SIF (Dilatação/Concentração e Negação).
- Aplicação de outros operadores de agregação, além do Máximo, para unir as regras com um mesmo conseqüente. Foram elaboradas as soluções matemáticas de alguns, usando métodos de Programação Matemática.
- Elaboração de métricas de associação entre termos antecedentes e conseqüentes das regras.
- Proposição de um modo de defuzzificação ótimo, caso alguns critérios sejam atendidos, que surge de forma espontânea ao se trabalhar com funções de pertinência triangulares.

1.4

Descrição e Organização da Dissertação

Esta dissertação está assim organizada:

- No capítulo 2 realiza-se uma Revisão Bibliográfica com foco principal em SFGs. Como a maioria dos SFGs utilizam AG, grande ênfase é dada à evolução da área a partir dessa técnica. Contudo, também são abordados SFGs baseados em PG, reservando a cada tópico um enfoque baseado na literatura produzida nesse ínterim.
- No capítulo 3 apresenta-se a técnica denominada PG e sua generalização a PGMG. São discutidas ambas as técnicas, além de ser exibido em cada todo o processo característico de um algoritmo evolutivo: inicialização, seleção, recombinação e avaliação.
- No capítulo 4 aborda o arcabouço do Modelo GPFIS e suas etapas de elaboração: Fuzzificação, Inferência, Defuzzificação/Decisão e Avaliação. É destinado um anexo para apresentar noções de operadores usados em Lógica Fuzzy, conjuntos fuzzy e base de regras fuzzy para o leitor leigo.
- No capítulo 5 exibem-se os Estudos de Caso do modelo GPFIS. Para cada problema enfrentado (Classificação, Previsão, Regressão e Controle) o procedimento adotado foi: (i) avaliar as diferentes parametrizações do

modelo GPFIS para um conjunto de benchmarks, de modo a avaliar o desempenho de cada um; (ii) a partir da melhor configuração, efetuar uma comparação com os demais SFGs e assim demonstrar que em aplicações tipo benchmark o modelo é tão eficiente quanto os demais SFGs presentes na literatura; (iii) elaborar uma modelagem para um problema básico, visando à apresentar amplamente os resultados gerados pelo modelo, levando em conta a acurácia e interpretabilidade.

- Por fim, no capítulo 6 apresentam-se as Considerações Finais e algumas novas direções que devem ser seguidas, tanto para o aprimoramento do modelo GPFIS, quanto em termos de potenciais aplicações e comparações inexploradas no trabalho.

2 Sistemas Fuzzy-Genéticos

2.1 Introdução

A área de Sistemas Fuzzy-Genéticos (SFGs) teve início na década de 90 [98], com os trabalhos pioneiros de Karr [124], Valenzuela-Rendón [207] e Thrift [197]. Cada um destes autores abriu caminho para as diferentes abordagens até hoje usadas em SFG: seleção de parâmetros para funções de pertinência [124], busca de base de regras fuzzy, ao codificar em um indivíduo uma regra (abordagem Michigan) [207] ou tratando um indivíduo como uma base de regras (abordagem Pittsburgh) [197]. De forma geral, os autores mencionados procuravam sintetizar um SFG a partir de um Algoritmo Genético (AG), até então a única meta-heurística genética existente.

Destas bases toda uma área se desenvolveu, tanto pela busca de novas aplicações fora da área de controle (reconhecimento de padrões, regressão, entre outras) e de novas meta-heurísticas (por exemplo, Programação Genérica - PG), quanto ao propor outros mecanismos para avaliação, sintetização e formação de um SIF [48, 78]. Toda uma literatura de SFG foi e é constantemente elaborada, tais como livros [49, 86, 113], edições especiais em revistas [7, 8, 162] e diversos exames sobre a área [48, 76, 78, 98]. Isto notabiliza e demonstra o avanço e crescimento desta família de sistemas híbridos.

Este capítulo não tem a pretensão de ser uma revisão extensa de todos os trabalhos presentes na área, mas sim das principais definições e concepções de um SFGs. Para tanto, a próxima seção exhibe as linhas gerais em termos de conceitos, terminologias e formulação de um SFG genérico. Na seção 2.3 disserta-se sobre uma das facetas mais usuais do emprego de um SFG: formação da base de regras fuzzy. Este tópico merece uma atenção especial, devido às diferentes abordagens criadas e ao intenso uso por parte da comunidade de SFG. Na seção 2.4, aborda-se a aplicação de SFGs no contexto de classificação de padrões, previsão, regressão e controle de processos. Por fim, na seção 2.5 efetuam-se as considerações finais do capítulo.

2.2

Linhas Gerais

Esta seção tem por objetivo enunciar as principais definições, conceitos e terminologias usadas na área de SFGs. A apresentação disposta é fruto da união dos exames mais recentes da área [48, 50, 98], porém com maior ênfase à exibida por Herrera [98].

A arquitetura básica e mais frequente de um SFG é mostrada na Figura 2.1. Um SFG é o resultado de uma interação entre um Sistema de Inferência Fuzzy (SIF) e uma Meta-Heurística Genética (MHG). Define-se por MHG o conjunto de Algoritmos Evolucionários usados para a formação de um SIF, como, por exemplo: AG, PG, Algoritmos Evolutivos Multiobjetivos (MOEA), entre outros. O usual na área de SFG é o SIF usufruir da MHG como mecanismo de composição de base de regras, ajuste de funções de pertinência, etc., mas nunca como um ator que possa alterar seu modo de inferência e estrutura, como ocorre nos casos de Sistemas Neuro-Fuzzy [1, 63, 158].

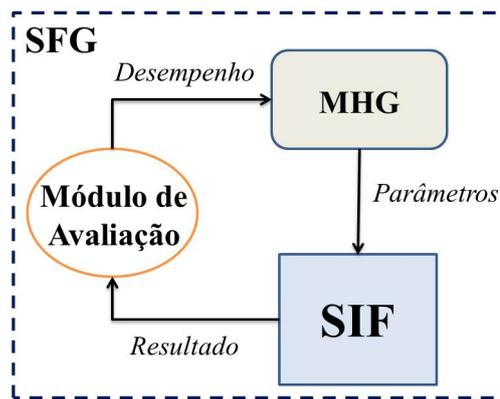


Figura 2.1: Diagrama genérico de um SFG.

Um SIF é composto por um Modo de Raciocínio, ou Mecanismo de Inferência Fuzzy, e uma Base de Conhecimento (BC). A BC é fruto da união da Base de Parâmetros (BP) e da Base de Regras (BR). A definição destas é fornecida abaixo:

- Base de Regras (BR): É composta pelas regras que formam o SIF.
- Base de Parâmetros (BP): É formada por todos os demais parâmetros que compõem o SIF. Logo, a BP compreende os fatores de escala do universo de discurso, tipo dos operadores de inferência e modo de defuzzificação, e, em especial, o número de funções de pertinência, seu formato e disposição.

O Modo de Inferência Fuzzy é o mecanismo que efetua o reconhecimento de padrões, ações de um controlador fuzzy, previsão, etc., a partir das informações previamente contidas em uma BC. Seja $\mathbf{x}_i = (x_{i1}, \dots, x_{iJ})$ ($j = 1, \dots, J$ e $i = 1, \dots, n$) um padrão formado pelas observações da J variáveis de entrada X_j . Esse padrão deve ser caracterizado, a partir de uma BC que possui D regras de um SIF, com um dos K possíveis consequentes (classes, ações, etc. - $k = 1, \dots, K$). Denote-se o l -ésimo conjunto fuzzy associado a j -ésima variável de entrada por $A_{lj} = \{(x_{ij}, \mu_{A_{lj}}(x_{ij}) | x_{ij} \in X_j\}$, munido de uma função de pertinência $\mu_{A_{lj}} : X_j \rightarrow [0, 1]$. Seja B_k o conjunto fuzzy associado a variável de saída $y_i \in Y$. Os passos comumente usados, apresentados de forma genérica para um SIF do tipo Mamdani, são elencados a seguir:

1. Grau de Compatibilidade: a ativação resultante da parte antecedente de todas as regras ($d = 1, \dots, D$) presentes na BR com o padrão \mathbf{x}_i . Para computá-lo usa-se comumente uma t-norma $*$:

$$\mu_{A_d}(\mathbf{x}_i) = \mu_{A_{i1}^d}(x_{i1}) * \dots * \mu_{A_{iJ}^d}(x_{iJ}), \quad (2-1)$$

2. Grau de Associação (prevalente em classificação): associação do padrão \mathbf{x}_i com a k -ésima saída de cada regra na BR, computada a partir do produto do peso da regra nesta saída (RW_d) com o Grau de Compatibilidade:

$$b_d^k = \mu_{A_d}(\mathbf{x}_i) \cdot RW_d \quad (2-2)$$

3. Grau de Ativação Resultante: usa-se um operador de agregação g (bastante comum o máximo) para reunir o Grau de Associação de todas as regras referentes ao j -ésimo consequente:

$$\mu_{B_k^*}(y_i) = g(b_d^k | d = 1, \dots, D; b_d^k > 0) \quad (2-3)$$

4. Defuzzificação/Decisão: a partir de cada $\mu_{B_k^*}(y_i)$, usa-se uma função F que mapeia os Graus de Ativação Resultantes em uma classe, ação de um controlador ou em um valor preciso quaisquer:

$$y_i = F(\mu_{B_1^*}(y_i), \dots, \mu_{B_K^*}(y_i)) \quad (2-4)$$

Fixado então o Modo de Inferência Fuzzy, o papel de um SFG é atuar em alguma área da BC, seja ao alterar a BP ou efetuar o aprendizado de uma BR concisa. A Figura 2.2, adaptada de Córdon et al. [48], ilustra o papel da MHG na atuação sobre o SIF.

Outra definição relevante diz respeito ao propósito do SFG. Em linhas gerais, um SFG possui dois objetivos: Ajuste Genético dos Parâmetros e

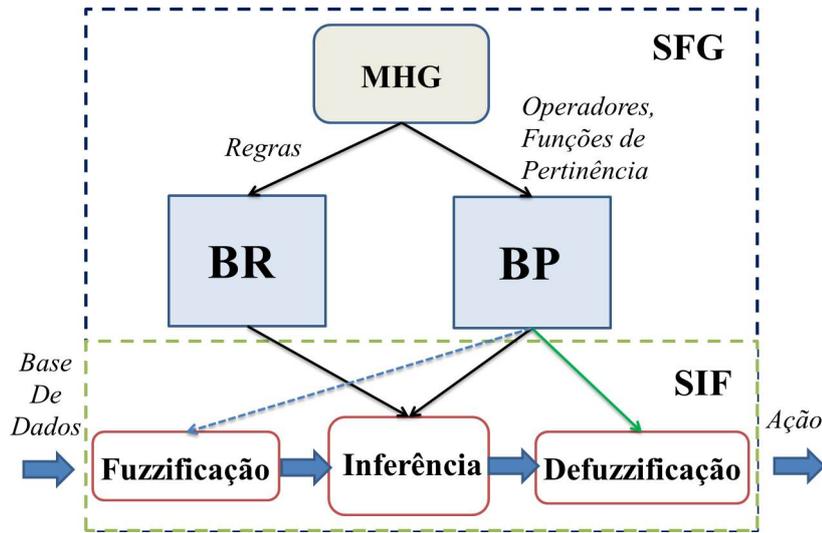


Figura 2.2: Diagrama das zonas de ação da MHG na estrutura de um SFG.

Descoberta de Conhecimento. Estes são descritos a seguir:

- Ajuste Genético dos Parâmetros: Dado a presença a priori de uma BC (elaborada por um especialista ou gerada a partir de um método de extração de regras, tal como de Wang e Mendel [208], a MHG ajusta os parâmetros da BP ou simplifica a BR.
- Descoberta de Conhecimento: utiliza-se o SFG para efetuar o aprendizado completo de um componente da BC, senão grande parte da mesma.

O próximo tópico aborda o primeiro objetivo - Ajuste Genético dos Parâmetros – com maior ênfase, apresentando um esforço na discriminação das principais referências e ideias neste tema.

2.2.1 Ajuste Genético de Parâmetros

Com a ideia de aperfeiçoar um SIF pré-concebido, diversos autores empregam a MHG ora para aprimorar o desempenho em termos de acurácia de classificação, ou regressão, ora para reduzir o número de regras presentes na BR. A Figura 2.3 organiza os quatro principais desdobramentos do Ajuste Genético dos Parâmetros: ajuste das funções de pertinência, adaptação dos operadores do Modo de Inferência Fuzzy, seleção das regras da BR e aperfeiçoamento do modo de defuzzificação.

Cada um destes é descrito a seguir, em conjunto com as principais referências:

- **Ajuste das funções de pertinência:** a partir de uma BC pré-concebida, efetua-se o ajuste das funções de pertinência por meio da

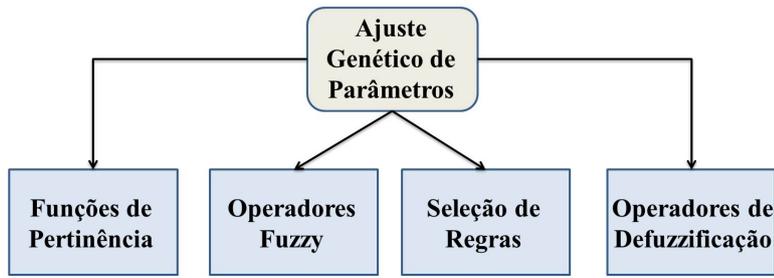


Figura 2.3: Diagrama com as possibilidades de atuação dos SFGs para Ajuste Genético de Parâmetros.

codificação na MHG dos parâmetros que descrevem o suporte da função de pertinência. A Figura 2.4 apresenta graficamente a ideia. Observa-se que o ajuste pode ocorrer tanto no sentido da translação da função de pertinência (Figura 2.4a), no estreitamento de sua amplitude (Figura 2.4b) ou ainda em todos os parâmetros simultaneamente, tornando a função de pertinência assimétrica (Figura 2.4c). Córdon et al. [49] (p.111) apresenta diferentes formas de codificação destas estruturas em uma MHG, no caso o AG. Arslan e Kaya [16] determinam, além dos parâmetros, o tipo de função de pertinência; Esmín et al. [71] ajusta as funções de pertinência a partir de um Enxame de Partículas, obtendo melhores resultados que um AG. Casillas et al. [38] efetua o ajuste das funções de pertinência a partir de modificadores linguísticos. Atualmente tem sido bastante comum o ajuste das funções de pertinência em uma fase pós-construção do SIF [5, 6, 187]. Contudo, poucos são os trabalhos que seguem os critérios apresentados por Oliveira [62] para o ajuste das funções de pertinência, seguindo determinadas restrições para manutenção da semântica (distingibilidade, cobertura do universo de discurso, normalização, etc.).

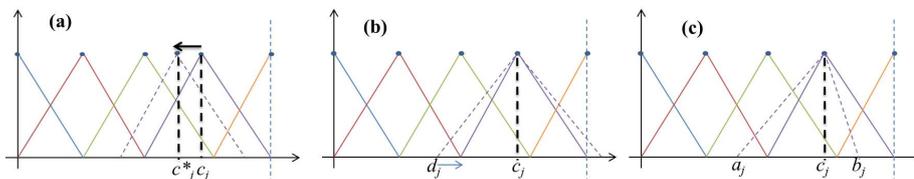


Figura 2.4: Diferentes possibilidades de ajuste das funções de pertinência.

- **Adaptação dos operadores do Modo de Inferência Fuzzy:** O principal intuito dessa abordagem é adaptar os operadores de t-norma, t-conorma e de agregação para melhorar o desempenho do SIF. Assim, as principais abordagens residem no uso de operadores parametrizáveis, tais como de Schweizer-Sklar, Hamacher, Frank, Yager, etc. (Klement et al. [129] faz uma compilação e análise teórica destes). O trabalho de

Alcalá-Fdez et al. [10] é um exemplo desse movimento, assim como os de Crockett et al. [61] e Crockett et al. [60], que ainda levam em conta a modificação das funções de pertinência.

- **Seleção de regras:** A partir de uma BC pré-concebida é possível empreender uma MHG com o objetivo de simplificar a BR presente. Assim, é possível auferir ganhos de interpretabilidade (maior compacidade e menor conflito da BR), além de possibilitar uma melhora na acurácia do SIF. O uso de uma MHG é necessário para essa busca, pois, dada uma BR, o número de outras possíveis BR geradas a partir desta é 2^Q (crescimento a taxa exponencial). A Figura 2.5 apresenta uma possível representação de um AG para esta tarefa.

AG	BR	
1	R ₁	Se X ₁ é P ...
0	R ₂	Se X ₄ é P ...
1	R ₃	Se X ₂ é MG ...
1	R ₄	Se X ₆ é MP ...
1	R ₅	Se X ₇ é M ...
0	R ₆	Se X ₂ é P ...
1	R ₇	Se X ₃ é G ...
0	R ₈	Se X ₁ é M ...

Figura 2.5: Exemplo de solução usando codificação binária de um AG para seleção de regras pertencentes a uma BR.

Das principais abordagens, as mais usuais residem na fixação da BP e geração da BR usando como semente inicial o método proposto por Wang e Mendel [208]. Assim, Casillas et al. [39] aplicam esta abordagem para um conjunto de problemas de regressão, além de considerar a possibilidade de ajustar as funções de pertinência através de deslocamentos e aplicações de modificadores linguísticos. Fernández et al. [77] utiliza esta abordagem com o intuito de classificar padrões em um ambiente cujas classes são desbalanceadas. Em Pulkinnen e Koivisto [173] é utilizada uma abordagem mista de Wang e Mendel [208] com Árvore de Decisão para um SFG voltado à problemas de regressão. Sánz et al. [188] usa esta abordagem em Sistemas Fuzzy Intervalar do tipo 2 para avaliar doenças cardiovasculares.

- **Aperfeiçoamento do modo de Defuzzificação:** uma forma relativamente simples de aprimorar o desempenho de um SFG é alterar o modo

de defuzzificação. Um exemplo simples é, dada a presença de um conjunto de entradas e alvos, fazer: (i) selecionar um conjunto de heurísticas de defuzzificação (centro de gravidade, altura, média dos máximos, etc.); (ii) propagar as entradas no SIF e aplicar cada alternativa de defuzzificação para gerar uma sequência de saídas; e (iii) combinar estas de modo a minimizar o erro entre os valores estimados (a combinação das saídas) e os alvos. Uma abordagem neste sentido é apresentada por Kim et al. [126].

É difícil caracterizar um trabalho como exclusivamente direcionado para ajustar funções de pertinência ou para a seleção de regras, por exemplo. O mais comum é sempre uma abordagem híbrida. O próximo tópico trata do aprendizado da BC quando não se possui a priori uma estrutura a ser aperfeiçoada.

2.2.2 Descoberta de Conhecimento

Em termos de Descoberta de Conhecimento, duas são as principais linhas para a construção de um SFG: Aprendizado da Granularidade (número e formato das funções de pertinência) e Aprendizado da BR. Cada uma destas localiza-se em um dos setores da BC: o Aprendizado da Granularidade na BP e o de regras na BR (Figura 2.6).

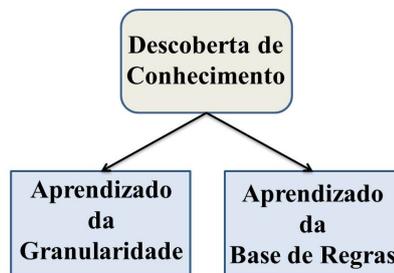


Figura 2.6: Diagrama com as possibilidades de atuação de um SFG para Descoberta de Conhecimento.

A seguir, são descritos o papel e a forma mais prevalente na literatura da concepção dessas abordagens em um SFG.

- **Aprendizado da Granularidade:** é codificado na estrutura da MHG o número de funções de pertinência que cada variável pode assumir. Dada uma estrutura de codificação (cromossoma, árvore, etc.) denominada C , a quantidade de funções de pertinência é delimitada pelo usuário, sendo prevalente o uso de valores no conjunto $C \in \{1, 2, \dots, 7\}$ [5, 54]. Assim,

o valor 1 representa a não inclusão do atributo no processo de geração das regras (don't care) e 2, 3, ...,7 representam o número de funções de pertinência. Estas costumam ser de forma triangular, uniformemente espaçadas e duplamente sobrepostas (nomeadas comumente por strong partition). A Figura 2.7 apresenta um exemplo desta codificação.

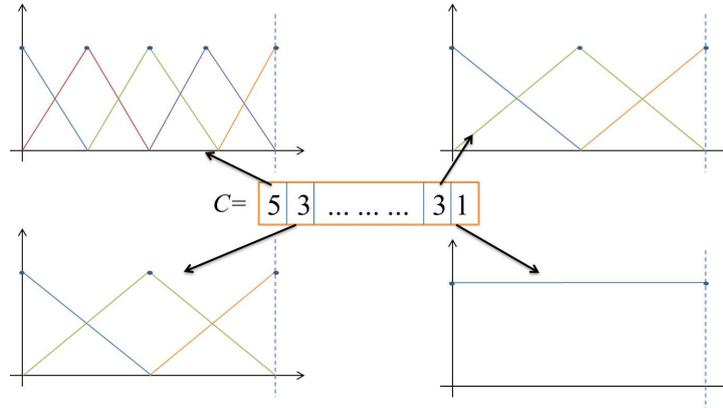


Figura 2.7: Exemplo de aprendizado de granularidade.

A partir das funções de pertinência é usado um método de geração de regras, na maioria dos casos o de Wang e Mendel [208] para a geração da base de regras correspondente à granularidade circunscrita a cada variável. Ainda é bastante comum efetuar um deslocamento das funções de pertinência durante o processo evolutivo. Na codificação $C = [C_1, C_2]$, a parte C_1 é responsável pela definição da granularidade, e C_2 indica o grau de deslocamento (incluindo restrições) de todas as funções de pertinência no universo de discurso correspondente. Desta forma é possível adaptar a granularidade e a disposição das funções de pertinência. Este tipo de abordagem é um exemplo da denominada representação em dupla ou pares (2-tuple) [101], 2000), que tem sido bastante aplicada na literatura recente [5, 81]. A Figura 2.8 apresenta o resultado da atuação do deslocamento nas funções de pertinência.

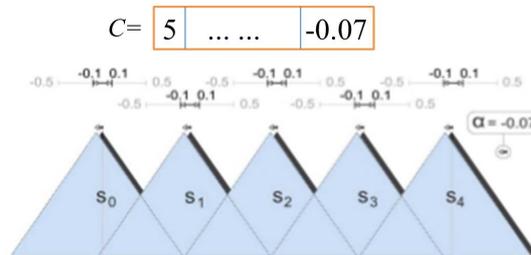


Figura 2.8: Exemplo de atuação conjunta: escolha de granularidade e deslocamento das funções de pertinência.

- **Aprendizado da BR:** A partir de uma definição da BP (formato e número das funções de pertinência, operadores, etc.), a MHG é responsável pela BR que deve ser extraída a partir das informações presentes na base de dados. Há diferentes abordagens para codificar na MHG um conjunto de regras. De forma geral, os trabalhos se concentram no Aprendizado exclusivo da BR [98], tanto que os SFGs desenvolvidos para este propósito são denominados Sistemas Fuzzy-Genéticos para Base de Regras (SFGBR). Este tema em si merece um destaque especial, apresentado na próxima seção.

Por fim, a Figura 2.9 dispõe um panorama geral do escopo do SFG: a união dos dois principais objetivos - Ajuste Genético de Parâmetros e Descoberta de Conhecimento - com seus consequentes desdobramentos para atuar nas diferentes peças que compõem um típico SFG.

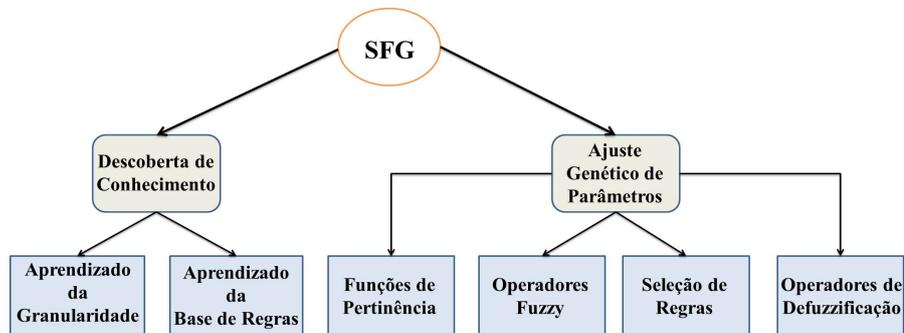


Figura 2.9: Visão global das áreas de atuação de um SFG.

2.3 Sistemas Fuzzy-Genéticos para Base de Regras

Os trabalhos em Sistemas Fuzzy-Genéticos para Base de Regras (SFGBR) dedicam-se ao estudo da forma de codificação de uma BR na MHG, de tal forma que o processo de extração das regras seja eficiente e atenda aos critérios delineados pelo usuário. Na literatura é possível encontrar capítulos de livros voltados para este tema [49], assim como publicações neste tema que vão além do âmbito dos SFGs, mas englobam toda a cadeia de aprendizado de máquina [78]. A seguir são apresentadas as principais abordagens para codificação de uma BR na MHG no âmbito dos SFGs: Abordagem Pittsburgh, Michigan, Iterative Rule Learning e Genetic Cooperative-Competitive Learning. Cada uma destas se diferencia não somente pela forma de codificação da solução, mas também pela forma de avaliação da qualidade da BR.

2.3.1 Abordagem Pittsburgh

Das abordagens possíveis, a do tipo Pittsburgh é a mais intuitiva e simples de ser concebida. Ela considera que cada indivíduo que compõe a população da MHG é uma BR (Figura 2.10). O usuário define a BP, ou seja, os operadores de conjunção, disjunção, a granularidade e o formato das funções de pertinência, e o método de defuzzificação/decisão do SIF. Esta parte do processo permanece fixa e cada indivíduo propõe um conjunto de regras para completar o SIF. O melhor indivíduo é aquele que satisfizer os critérios definidos pelo usuário, tais como acurácia, interpretabilidade e tempo de resposta.

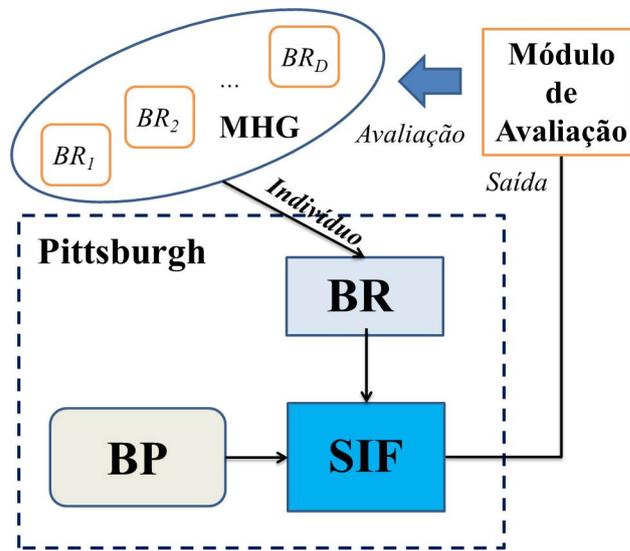


Figura 2.10: Diagrama do processo de síntese de um SFGBR do tipo Pittsburgh.

Em praticamente todas as áreas de aplicação de SFGs há um SFGBR do tipo Pittsburgh. Um dos trabalhos pioneiros na área de SFGs [197] elabora um SFG do tipo Pittsburgh para controle, assim como o trabalho de Alba et al. [2]. Córdon et al. [49] (p.146-147) apresenta uma série de aplicações de codificações do tipo Pittsburgh. Em Sánchez et al. [183] e Tsakonas [201] surgem trabalhos que usam Programação Genética para elaborar um SFG para classificação. Os trabalhos de Gorzalczany e Rudzinski [91] e Brito et al. [33] apresentam um SFGBR do tipo Pittsburgh, a partir do SIF do tipo Mamdani, para modelar séries temporais. Em Casillas et al. [40] é exibido um SFG para aprendizado de BRs compactas, consistentes e completas em problemas de regressão.

Uma das grandes críticas referentes à abordagem Pittsburgh é o tempo computacional que esta demanda para avaliação da população, assim como a

pouca efetividade dos operadores de recombinação para ampliar a diversidade das regras geradas [48]. A alternativa utilizada por diversos autores é a paralelização computacional do SFGBR [177], a hibridização da abordagem Pittsburgh com codificações de BR mais enxutas [117], ou ambas [118]. Apesar desses ganhos, diversos autores recorrem a outros tipos de codificação de forma a tornar o procedimento de geração da BR mais célere. Um meio bastante usado é tomar o indivíduo como um regra, ao invés de uma BR, e assim reduzir drasticamente o tempo de máquina e aumentar o impacto dos operadores. Este tipo de codificação, denominado Michigan, é objeto do próximo tópico.

2.3.2 Abordagem Michigan: Visão Geral

A abordagem Michigan surge com o principal objetivo de gerar uma BR com baixo custo computacional [98]. Nesta representação, um indivíduo codifica somente uma regra, tal que a BR final é resultado da concatenação dos indivíduos da MHG. Logo, a sobrecarga computacional é reduzida drasticamente em termos de memória alocada, custo nos processos de avaliação e recombinação. Por outro lado, o procedimento de avaliação é mais complexo e indireto, já que uma regra pode representar bem o processo quando isolada, mas, quando inserida em uma BR, podem ocorrer conflitos de ativação que corrompem a predição.

A abordagem Michigan inclui, além do módulo de avaliação da BR, um relativo ao aporte de credibilidade para cada regra. Portanto, apesar da BR ser inserida no SIF, cada uma das regras recebe uma avaliação em separado, baseada na contribuição para o resultado final. A partir do crédito concedido é possível elencar as melhores soluções e aplicar os operadores de seleção e recombinação intrínsecos da MHG.

Neste instante é que se distinguem as abordagens: Michigan, Iterative Rule Learning e Genetic Cooperative-Competitive Learning. Cada uma destas se diferencia:

- Em menor grau: na forma de concessão da credibilidade individual de cada regra.
- Em maior grau: nos mecanismos aplicados para adaptar e sintetizar a BR.

As métricas mais comuns para avaliar a qualidade de uma regra são o Grau de Confiança (acurácia fuzzy - *CD*) e o Suporte (*Supp*) [49]. Para tanto,

considere-se que a q-ésima regra está associada à h-ésima classe:

$$CD_q = \frac{\sum_{p \in h} \mu_{A_q}(\mathbf{x}_p)}{\sum_{p=1}^n \mu_{A_q}(\mathbf{x}_p)} \quad (2-5)$$

$$Supp_q = \frac{\sum_{p=1}^n \mu_{A_q}(\mathbf{x}_p)}{n} \quad (2-6)$$

a (2-5) mede a intensidade da relação entre a q-ésima regra com a h-ésima classe, enquanto a (2-6) reflete o nível de identificação médio da q-ésima regra com os padrões da base de dados. O cálculo do Suporte independe do problema em análise (classificação, regressão ou previsão), enquanto o Grau de Confiança somente se aplica a problemas de classificação. Por fim, note-se que nesta abordagem torna-se necessária a inclusão de mecanismos para evitar a ocorrência de um viés no processo de geração das regras. O objetivo deste é reduzir a ocorrência de regras semelhantes ou iguais às já elaboradas nas gerações seguintes. Para tanto, pode-se usar penalizações com respeito ao grau de semelhança das regras (ativações semelhantes, mesmo padrões classificados, etc.) ou, ainda, remover da base de dados padrões já classificados (um exemplo é fornecido em Berlanga et al. [28]). A seguir é apresentada a primeira das abordagens: Michigan.

Abordagem Michigan: Aspectos Específicos

Em linhas gerais, a abordagem Michigan emprega um aprendizado por reforço e usa a MHG para adaptar as regras a partir de uma competição ao longo do processo evolutivo [98, 132]. Para tanto, ao considerar aplicações cujo aprendizado é supervisionado (classificação, regressão e previsão, em geral), é necessário substituir a forma de ofertar a base de dados para o SFG: do tradicional batch para o incremental. Assim, um padrão (ou um pequeno subconjunto) é apresentado ao SFG. A BR inicializada (aleatoriamente ou por algum algoritmo) é exposta a esse exemplo, de modo a se verificar qual a regra que promoveu uma classificação ou predição correta. Como as aplicações prevalentes são em classificação [78, 98], métricas de Suporte e Confiança são também computadas. Com estas métricas é possível ordenar os indivíduos, selecionar e aplicar os operadores de recombinação. A Figura 2.11 apresenta um diagrama do processo de sintetização empregada na abordagem Michigan.

Conforme os exemplos são oferecidos, as regras competem entre si de modo que as mais qualificadas permanecem ao longo do processo evolutivo. Na última geração, quando todos os padrões tiverem sido apresentados, a

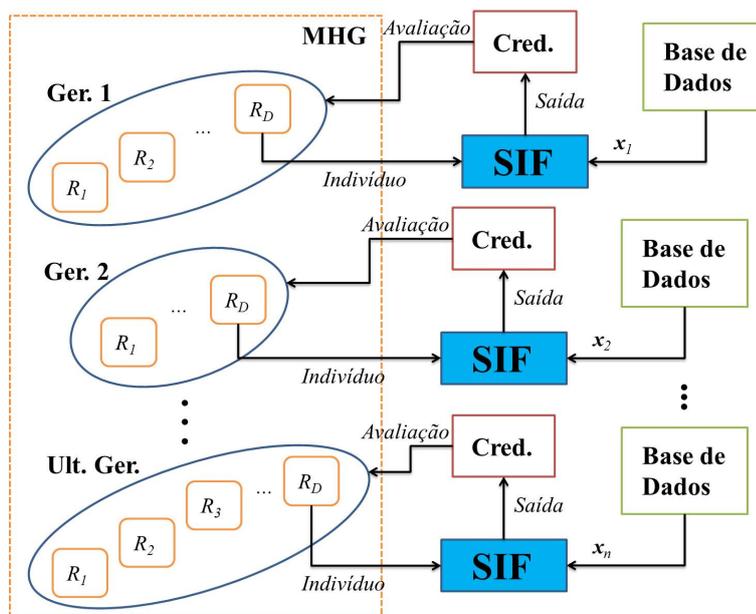


Figura 2.11: Diagrama do processo de sintetização de um SFGBR do tipo Michigan.

população final forma a BR definitiva. Um dos trabalhos pioneiros na área de SFGs ([207]), mostra uma concepção de SFGBR do tipo Michigan para classificação. O primeiro livro em SFGs [86] trata da concepção de um SIF para classificação a partir de uma decodificação do tipo Michigan usando PG. Uma das abordagens mais proeminentes é a de Casillas et al. [37], que apresenta o modelo Fuzzy-XCS, um classificador fuzzy baseado na lógica de construção do tipo Michigan. Neste trabalho, os autores apresentam uma roupagem moderna do método, caracterizando diversas possibilidades de uso, sempre seguindo a lógica do aprendizado por reforço. Marín-Blázquez e Pérez [147] apresentam uma aplicação do modelo Fuzzy-XCS para um sistema de detecção de intrusos.

No exame da literatura realizado por Herrera [98], este verifica a necessidade da concepção de novos trabalhos neste tema. Contudo, em paralelo, surgem duas abordagens derivadas da Michigan: Iterative Rule Learning e Genetic Cooperative-Competitive Learning. Ambas não exploram o conceito de aprendizado por reforço, além de incluírem outros procedimentos para a busca da BR final. Os SFGBR do tipo Iterative Rule Learning são abordados a seguir.

Abordagem Iterative Rule Learning

O Iterative Rule Learning (IRL) é um procedimento de codificação de BR bem consolidado, porém mais recente do que a Michigan [51]. A ideia central do método é executar a MHG um número de vezes, de modo que em cada execução a melhor regra é removida para compor a BR final. A Figura 2.12

apresenta um diagrama do processo de geração da BR.

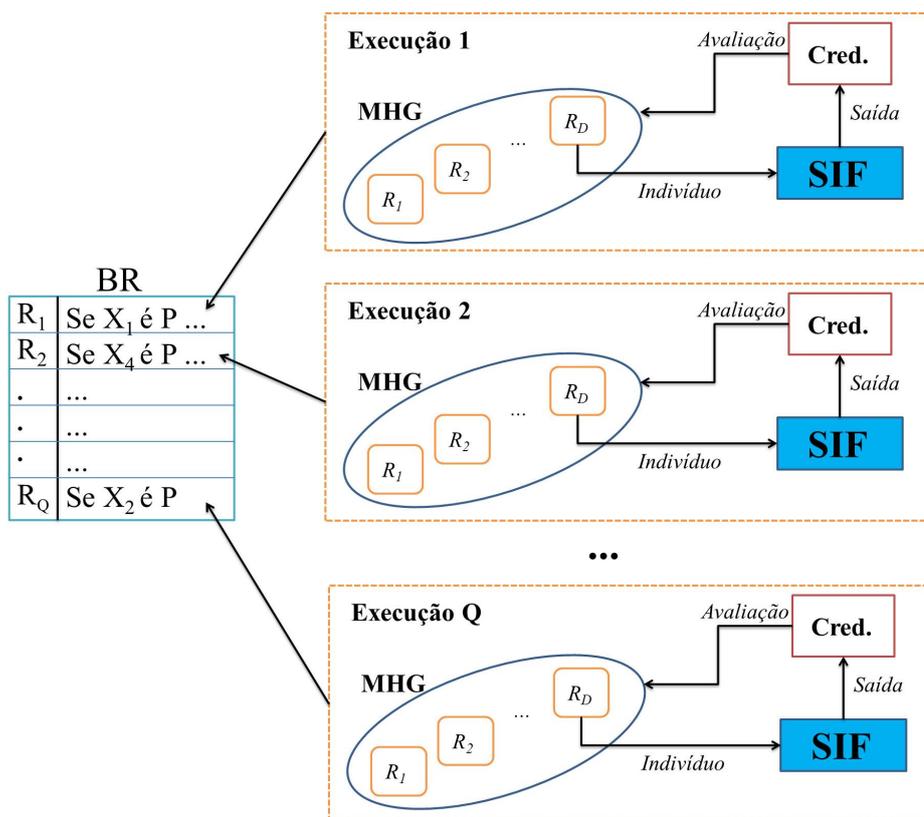


Figura 2.12: Diagrama do processo de síntese de um SFGBR do tipo IRL.

A definição da melhor regra pode ser baseada no Grau de Confiança, Suporte, Cobertura, dentre outras extensamente apresentadas em Córdon et al. [49] (p.226-231). Há diversos esquemas possíveis para definir o critério de parada, tais como: gerar um número de regras tal que possibilitem abranger todos os padrões, ou ao menos gerar uma regra que contemple uma determinada classe, ou consequente, entre outros [49] (p.232).

Na literatura uma das primeiras arquiteturas é a MOGUL (Methodology to Obtain Genetic fuzzy rule-based systems Under the iterative rule Learning approach), proposta em Córdon et al. [57], que forneceu as bases para os demais SFGBR do tipo IRL que surgiram a seguir. Outro exemplo bastante explorado na literatura é o modelo SLAVE (Structural learning algorithm on vague environment), disposto em González e Pérez [88], cujo objetivo é atuar em bases de dados de elevada dimensionalidade. Para tanto, o SLAVE codifica em um AG uma estrutura dividida em duas partes: a primeira define a relevância do atributo e a segunda atribui demais parâmetros associados às características exibidas na primeira parte (regras, granularidade, etc.). Recentemente, em García et al. [85] é apresentado um aperfeiçoamento do trabalho de González e Pérez [88]. O modelo apresentado parte de um avanço proposto anteriormente

por González e Pérez [90], de modo a incluir a possibilidade de geração de regras relacionais fuzzy (modelagem conjunta dos termos antecedentes) e geração de novos atributos a partir de combinações lineares dos iniciais. O autor realiza um estudo com 27 bases de dados para classificação.

Abordagem Genetic Cooperative-Competitive Learning

Os SFGBR do tipo Genetic Cooperative-Competitive Learning (GCCL) são os mais recentes entre as três abordagens do tipo indivíduo-regra [112]. O cerne desses modelos, conforme Herrera [98] e Fernández et al. [78], é efetuar uma abordagem híbrida, em que as regras competem e cooperam entre si. Portanto, deve-se medir a qualidade destas, tanto individualmente, como também em conjunto.

Logo, a abordagem GCCL envolve tipicamente a definição de duas funções de avaliação: uma local e outra global. A função de avaliação local comumente avalia a qualidade da regra com respeito ao Suporte e Grau de Confiança. A função de avaliação global analisa a acurácia e parâmetros da BR, tais como o número de regras, termos antecedentes por regra, etc. A partir da função de avaliação local, os indivíduos são ordenados para efetuar as operações de seleção e recombinação (fator competição). A função de avaliação global avalia a qualidade geral da população ao longo das gerações, de modo que a melhor população é sempre armazenada (fator cooperação). A Figura 2.13 apresenta o processo de sintetização de um SFGBR do tipo GCCL.

De forma similar às outras abordagens, a inclusão de penalização com respeito à semelhança das regras deve ser também empregada, visando à manutenção da diversidade da população. Outra característica comum dessa abordagem é a variação do tamanho da população ao longo do processo evolutivo. Em Berlanga et al. [28] é apresentado um mecanismo para expansão/retração da população, baseado na penalização por similaridade, baixa compatibilidade com os padrões da base de dados e com o auxílio da geração de regras auxiliares para padrões não classificados pelas regras da presente população.

A primeira abordagem neste sentido é apresentada por Ishibuchi et al. [112], que propõem um AG com uma representação de tamanho fixo para reconhecimento de padrões. Neste trabalho o consequente e o peso da regra são determinados a partir do Grau de Confiança da regra. Em Juang et al. [123] é apresentada uma elaboração quase plena de um Controlador Fuzzy, com o ajuste das funções de pertinência, base de regras e seleção do tipo de consequente (TSK ou Mamdani) através de um AG baseado em Evolução Simbiótica. Mucientes et al. [156] exhibe uma aplicação da codificação GCCL

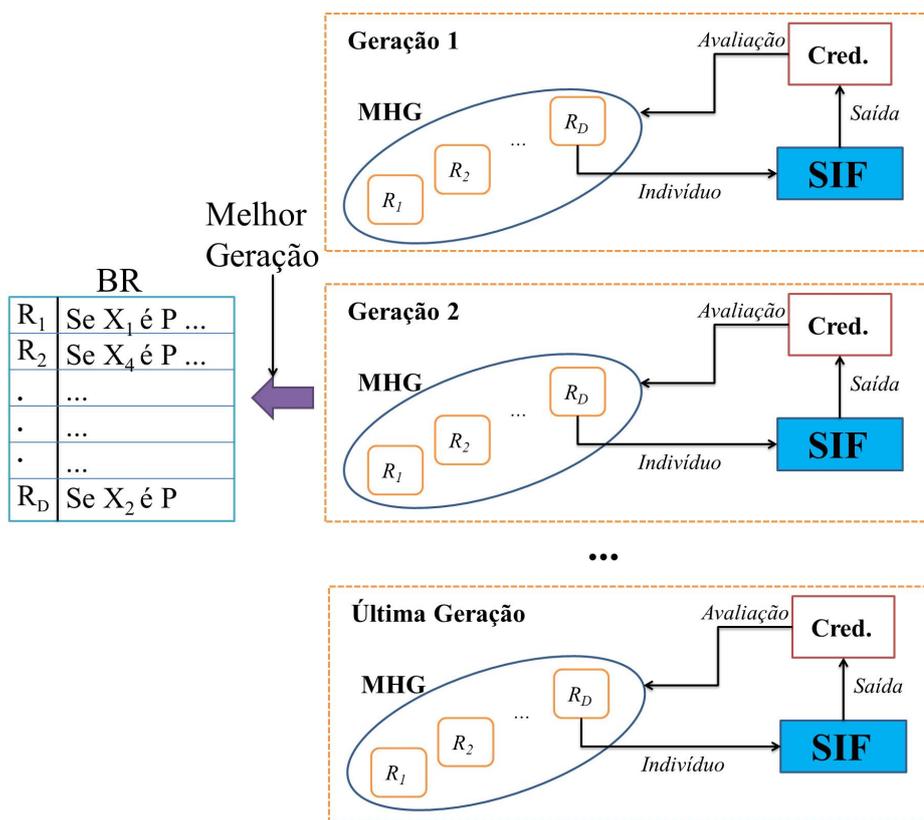


Figura 2.13: Diagrama do processo de sintetização do tipo GCCL.

para aprendizado de regras para um SIF do tipo TSK. A MHG usada é PG com gramática livre de contexto e a aplicação é baseada em uma base de dados real de cinco diferentes máquinas de uma fábrica moveleira.

Em Berlanga et al. [28] é proposto o modelo GP-COACH (Genetic Programming-based learning of COmpact and ACcurate fuzzy rule-based classification systems for High-dimensional problems) aplicado a problemas de classificação com alta dimensionalidade, empregando a PG com a base de regras fuzzy codificada através da formulação GCCL. Diversos elementos expostos nesta seção são baseados no modelo apresentado pelos referidos autores. Este mesmo modelo é estendido em López et al. [142] para reconhecimento de padrões em um ambiente no qual as classes são intensamente desbalanceadas. Ainda, os autores em Palacios et al. [167] propõem um SFGBR do tipo GCCL para extrair regras de bases de dados cujos padrões são imprecisos ou definidos de modo linguístico.

2.3.3

Síntese Final

A Tabela 2.1 apresenta uma síntese geral das características de cada uma das quatro abordagens: Pittsburgh, Michigan, IRL e GCCL. Observa-se

que a única baseada em aprendizado por reforço é a abordagem Michigan. Cumpre destacar que a abordagem Pittsburgh possui um custo de concepção baixo. Em uma situação hipotética na qual o usuário dispõe apenas de um AG simples e de um SIF, a formação de um SFG baseado na lógica Pittsburgh demandaria somente a definição da função de avaliação, enquanto que nas demais abordagens seria necessário definir outros parâmetros (mecanismos de remoção/redução de regras similares, número de execuções, etc.).

Tabela 2.1: Síntese das principais codificações para SFGBR.

Tipo	Aprend.	COD	CIR	CCR	AccR	CBR	PrApl
Pittsburgh	Sup	Ind=BR	Baixo	Alto	Alta	Baixo	Gen
Michigan	Reforço	Ind=R	Alto	Baixo	Média	Alta	Class
IRL	Sup	Ind=R	Médio	Médio	Média	Alta	Class
GCCL	Sup	Ind=R	Médio	Baixo	Média	Alta	Class

Legenda: *Aprend.* – Tipo de Aprendizado, *Cod.* – Codificação da BR, *CIR* – Custo de Implementação Relativo, *CCR* – Custo Computacional Relativo, *AccR* – Acurácia Relativa, *CBR* – Complexidade da BR, *PrApl* – Principais Aplicações, *Gen* - Genérico e *Class* - Classificação.

Por outro lado, a abordagem Pittsburgh é computacionalmente mais custosa do que as demais, devido à necessidade de avaliar uma BR por indivíduo, enquanto que nas outras o objeto de avaliação é uma única regra. Ishibuchi et al. [117] verificam, a partir de um conjunto de benchmarks para classificação, que os resultados da abordagem Michigan proporcionam uma BR mais compacta, porém com acurácia final inferior à da Pittsburgh. Os autores argumentam que isto ocorre por causa da elevada diversidade nas regras elaboradas, a qual prejudicaria na combinação visando à formação de uma BR final. Este fato leva os autores a proporem uma abordagem híbrida Pittsburgh-Michigan, obtendo assim bons resultados.

Por fim, baseado nas referências apresentadas nas seções anteriores, além dos exames da literatura [48, 50, 78, 98], é possível avaliar que o principal uso das abordagens indivíduo-regra são prevalentes em problemas de classificação, porém, como descrito anteriormente, também têm sido usadas na área de controle. Por outro lado, as aplicações de SFGBR do tipo Pittsburgh são voltada para controle, reconhecimento de padrões e previsão. A próxima seção apresenta alguma das principais aplicações de SFG.

2.4

Principais Aplicações de Sistemas Fuzzy-Genéticos

Esta seção apresenta alguns trabalhos disponíveis na literatura de SFGs, relativos à solução de problemas de classificação, previsão, regressão e controle.

Se fosse possível posicionar as aplicações com respeito ao volume de trabalhos, um provável ordenamento seria: 1) classificação, 2) controle, 3) regressão e 4) previsão. De fato, apesar das pesquisas pioneiras na área de SFGs terem se concentrado em problemas de controle, ao longo do tempo observou-se um redirecionamento para o reconhecimento de padrões. Isto se deve, possivelmente, às características de extração e interpretação de conhecimento em nível linguístico que os SIF delineados para classificação possibilitam, quando comparados aos demais modelos disponíveis na literatura (estatísticos e de inteligência computacional). O próximo tópico aborda algumas aplicações de SFGs em previsão de séries temporais.

2.4.1 Previsão

Um dos primeiros trabalhos na área de previsão foi o de Kim e Kim [127], que elaboraram um comitê fuzzy-genético, baseado em AG, para previsão de séries temporais não estacionárias. Cada membro do comitê era elaborado em dois estágios: o primeiro gerava a base de regras fuzzy de modo a contemplar o maior número de padrões possível, enquanto o segundo estágio ajustava as funções de pertinência de modo a ampliar o desempenho da arquitetura. As aplicações do SFG proposto incluem a série Mackey-Glass e taxa de câmbio. O trabalho de Gorzalczany e Rudzinski [91] apresenta uma abordagem relativamente simples para implementação. Estes usam um SFGBR do tipo Pittsburgh, a partir do SIF do tipo Mamdani, para modelar séries temporais genéricas (valores reais, inteiros, etc.). A forma de modelagem é bastante comum e simples. Os autores aplicaram este SFG em um conjunto de séries temporais benchmarks. O trabalho de Brito et al. [33] é semelhante ao descrito anteriormente, de forma que estes apresentam uma ferramenta para previsão usando um SFG baseado em AG.

Em Bergmeir et al. [27] é proposto o uso da MHG denominada Algoritmos Meméticos, para ajustar parâmetros do modelo de transição suave auto-regressivo com coeficientes neurais (NCSTAR) [150], sendo este equivalente a um SIF do tipo TSK aditivo com funções de pertinência de formato logística. O Algoritmo Memético é usado para ajustar os parâmetros de transição entre os modelos, a inclinação da função logística e o peso dado a uma variável exógena ou a defasagem inclusa no modelo. O SFG proposto foi aplicado em competições de séries temporais. Aznarte et al. [19] elaboram um SFG baseado no modelo NCSTAR em dois estágios: o inicial elabora uma BR a partir do uso da Estratégia Evolutiva (1+1)-ES [52] para previsão de séries financeiras,

e no final seleciona-se um subconjunto de regras mais promissoras a partir de AG de codificação binária [53].

O artigo de Hadavandi et al. [93] propõe o uso de um SFG para efetuar previsões para vendas de bens e serviços. O modelo elaborado inicia-se com o particionamento do conjunto de registros da série temporal em k grupos definidos pelo algoritmo k-means. Para cada grupo é gerado um SFG para extração da BR e ajuste da BP. Experimentos demonstram que a abordagem proposta proporciona desempenho superior aos métodos tradicionais de previsão. Por fim, em Chen et al. [46] exibem-se um SFG para evoluir um SIF do tipo TSK hierárquico. O ajuste dos parâmetros que compõem a BP é efetuado pela programação evolutiva incremental probabilística, enquanto que a BR é gerada a partir da programação evolutiva. O modelo proposto é composto por um laço, em que BP e a BR são ajustadas de forma intercalada entre as MHG. O modelo é avaliado por meio da série temporal de Mackey-Glass.

De um modo geral, são exíguos os trabalhos que seguem uma abordagem canônica de um SFG, além de poucos se preocuparem com a interpretabilidade do sistema gerado. Ainda, são muito pouco explorados o uso de modificadores linguísticos e os tipos de codificação da BR, entre outros pontos, afora a ausência de trabalhos que usam PG como MHG do SFG. Há também uma gama de trabalhos na área de previsão que fazem uso da Lógica Fuzzy ([135]), [13, 138],[193]), mas não relacionados ao tema de SFGs.

2.4.2 Regressão

De forma similar, os trabalhos na área de regressão não atraíram inicialmente grande foco da comunidade de SFG [173], mas atualmente têm ganhado mais expressão e espaço na literatura. De forma geral, os trabalhos de SFGs para regressão são semelhantes aos de classificação, devido à grande proximidade dos problemas. Porém estes surgem em um ambiente onde o tema interpretabilidade é um requisito indispensável para a formação de um SFG. Logo, a maioria dos trabalhos em SFGs para regressão usam um MOEA como MHG para sintetização da BR. Um dos poucos a não seguirem esse conceito é o de Alcalá et al. [4], que apresenta uma das primeiras aplicações da representação em dupla. É proposto um SFG, baseado em AG, que aprende a granularidade das funções de pertinência para cada variável, assim como o seu ajuste. O algoritmo de Wang e Mendel [208] é usado para gerar a BR. A aplicação da metodologia proposta envolve duas aplicações reais, visando a estimar o tamanho e custo de linhas de transmissão.

O trabalho de Antonelli et al. [14] propõe um SFG multi-objetivo para gerar um SIF do tipo Mamdani, com elevada acurácia e redução de complexidade. A grande novidade da abordagem reside no aprendizado da granularidade e da base de regras (típica abordagem do tipo Descoberta de Conhecimento). Para tanto, introduz-se o conceito de BR virtual e concreta: a virtual é baseada no número máximo de funções de pertinência para cada variável, enquanto a concreta se baseia na codificação apresentada pelo indivíduo. O algoritmo apresentado foi aplicado a dois benchmarks de regressão. Enquanto isso, em Pulkkinen e Koivisto [173] é apresentado um SFG para aprendizado completo de um SIF. É usada uma MHG baseada em um MOEA, de modo a ajustar os parâmetros das funções de pertinência, granularidade e base de regras. A inicialização da população é beneficiada por uma pré-seleção de variáveis e o uso do método de Wang e Mendel [208] adaptado. O método apresentado provê desempenho superior às demais SFG em um conjunto de 10 benchmarks.

Ainda, trabalhos mais recentes ([5]) fazem uso de um SFG multi-objetivo para maximização da acurácia e interpretabilidade, de modo a definir a granularidade das funções de pertinência de cada variável e, a partir disso gerar a base de regras pelo método de Wang e Mendel [208]. Ainda, durante o processo evolutivo, as funções de pertinência são deslocadas, seguindo a representação em dupla, conforme discutido anteriormente. Por fim, é efetuada uma fase de pós-processamento com o ajuste das funções de pertinência. O modelo apresentado foi usado em um conjunto de dezessete benchmarks.

Benítez e Casillas [26] apresentam um SFG multi-objetivo, visando à solução de problemas de alta dimensionalidade a partir de uma estrutura hierárquica. Este modelo se baseia na ideia de sub-sistemas fuzzy. O MOEA empregado tem uma codificação dupla, que em parte indica quais variáveis vão ingressar em um subsistema e como os centros das funções de pertinência triangulares devem variar. Após as definições dos sub-sistemas, a BR é gerada via Wang e Mendel [208]. A abordagem proposta é usada em cinco benchmarks e comparadas com demais SFGs. Por fim, em Márquez et al. [148] exhibe uma abordagem diferente dos demais trabalhos de SFGs para regressão. A ideia é usar um MOEA que adapta o operador de conjunção (no caso uma t-norma paramétrica) que atua sobre cada regra, de modo a maximizar a acurácia total do sistema e reduzir o número de regras geradas. A BR é inicialmente gerada pelo método de Wang e Mendel [208], seguido de um mecanismo de redução do número de regras. A partir disto, o número de genes usados na codificação têm o mesmo tamanho da BR, de forma que cada gene representa a inclusão ou não da regra, além do tipo de t-norma a ser usada nesta especificamente, dentre

uma família de possibilidades. A abordagem proposta é promissora, contudo o número de regras geradas é elevada.

2.4.3

Controle

Uma das primeiras elaborações de um Controlador Fuzzy – tipicamente do tipo Mamdani - a partir de uma MHG foi a apresentada por Karr [124], que utiliza um AG para ajustar as funções de pertinência das variáveis de entrada e saída. Após este passo inicial, diversos outros trabalhos surgiram, na maioria baseados em AG, para efetuar um ajuste fino da BP e busca de uma BR concisa para compor um Controlador Fuzzy. O conjunto de SFG destinados a elaborar Controladores Fuzzy é denominado de Controladores Fuzzy-Genéticos (CFGs) [44, 76, 98].

Vários trabalhos envolvem o uso de CFGs: em Herrera et al., [100] é exibida uma seleção de base de regras, a partir de uma base inicial fornecida por um especialista, para um SIF do tipo Mamdani usando um AG. Em Juang et al. [123] é apresentada uma elaboração quase plena de um Controlador Fuzzy, com o ajuste das funções de pertinência, base de regras e seleção do tipo de consequente (TSK ou Mamdani) através de AG baseado em Evolução Simbiótica.

Em Liu et al. [141] selecionam-se aplicações de modificadores linguísticos, a partir de um AG, para ajuste das funções de pertinência do controlador. Pal e Pal [165] propõem um Controlador Fuzzy Hierárquico auto organizável, baseado em um AG, para atingir dois objetivos sequencialmente: adquirir controlabilidade e reduzir a complexidade da base de regras.

Recentemente, diversos trabalhos que usam AG para ajustar parâmetros de um Controlador Fuzzy são mais direcionados para aplicações reais, aproveitando o conhecimento previamente estabelecido [76, 95, 184] e para ajustar Controladores Fuzzy tipo 2 [44]. Ainda, há a presença de trabalhos que evoluem Controladores Fuzzy, mas, ao invés de usarem AG, adotam Enxame de Partículas [172] e outros algoritmos bio-inspirados [42, 43].

São exíguos os trabalhos que usam PG para construir Controladores Fuzzy, quando comparados com AG, apesar da sua estrutura dinâmica que beneficia a codificação de bases de regras e demais parâmetros de um SIF [48]. Os primeiros trabalhos neste sentido foram de Alba et al. [2] e Tunstel and Jamshidi [205] que usaram uma PG com gramática restrita para elaborar uma BR de um Controlador Fuzzy. Em Homaifar et al. [104] é apresentado um Controlador Fuzzy para trajetória de robôs móveis. Mais recentemente,

Tsakonas [202] propõe o uso de PG Memética para criar um controlador do tipo TSK. Todas as abordagens citadas adaptam a estrutura da PG para formular um Controlador Fuzzy de forma similar a um procedimento de um AG. Poucas vantagens são efetivamente exploradas da codificação, tais como a possibilidade de combinar diferentes t-normas, t-conormas, usar operadores de agregação, etc. durante o processo de sintetização do controlador.

2.4.4 Classificação

Os SFGs para classificação se distinguem dos demais SIFs, pois ao invés de elaborarem um conceito aproximado (como SIF do tipo Mamdani), discriminam – valor preciso - à qual classe pertence um determinado padrão [48, 108]. Logo, estes SFGs buscam regras do tipo:

Regra: “Se X_1 é A_{l1} , X_2 é A_{l2}, \dots, X_J é A_{lJ} ,
então x_i é classe k com RW igual a γ ”

Um dos primeiros trabalhos neste sentido foi o de Ishibuchi et al. [114], que apresentou um SFGBR do tipo Pittsburgh, que, a partir de um AG, verificava, dentre todo o universo de possíveis regras, quais seriam inclusas na BR. A definição da classe mais compatível com o termo antecedente é empregada nesse trabalho, sendo esta uma das características exploradas no decorrer da presente dissertação. Já no trabalho de Ishibuchi et al. [114], a formulação da maximização da acurácia e redução do número de regras (fator interpretabilidade) já está inserida como um determinante para a qualidade da solução. A grande dificuldade neste momento foi a formulação da função objetivo, devido à utilização de um AG comum. Um pouco depois, em Ishibuchi et al. [110], o trabalho anterior foi aprimorado, com a inclusão da MOEA para atender a ambos critérios.

Desde então, tem surgido diversas implementações de SFGBRs visando ao reconhecimento de padrões, tais como a clássica Michigan [37] e decodificações semelhantes, como: GCCL [112] e a IRL [51]. Uma abordagem interessante neste íterim é a de Delgado et al. [64], que introduz um SFG hierárquico do tipo TSK, utilizado tanto para problemas de ajuste de curva, assim como classificação. Nesta abordagem, o AG realiza o aprendizado de toda a estrutura da BC, ou seja, ajusta os parâmetros das funções de pertinência, define as regras, etc., enquanto a parte consequente é realizada por mínimos quadrados.

Em Ishibuchi et al. [117] é disposta uma hibridização entre um SFGBR do tipo Pittsburgh com o GCCL para problemas de classificação, de modo a aproveitar a maior acurácia da Pittsburgh, com o aumento de diversidade gerada

pela GCCL. Em termos de abordagem GCCL, Palacios et al. [166] apresentam uma extensão do método para bases de dados de baixa qualidade (valores faltantes ou agregados, entre outros). Em Fernández et al. [79] são explorados os modelos de González e Pérez [88] (tipo IRL) e Ishibuchi et al. [117] (híbrido Pittsburgh-GCCL), de modo a investigar diferentes heurísticas de votação em problemas de classificação do tipo um contra um (one-against-one ou pairwise learning). Recentemente, os modelos propostos têm se voltado para redução de tempo computacional, introduzindo conceitos de computação paralela e novas formas de aprendizado [118], seleção de operadores de agregação mais adequados para classificação [21] e SIF do tipo 2 para classificação e com ajuste das funções de pertinência [186, 187].

Todos os SFGs descritos anteriormente baseiam-se ou em um MOEA ou no paradigma prevalente de AG. Ao se modificar a MHG de AG para PG, alguns autores auferiram ganhos em redução da complexidade da BR, assim como em acurácia final. Um dos primeiros trabalhos neste sentido é o apresentado em Tsakonas [201] que usa uma PG com gramática livre de contexto para elaboração de um SFGBR do tipo Pittsburgh, dentre outras aplicações. Nesse trabalho o autor apresenta diversos usos do seu modelo, sendo pouco visado o SFGBR. Logo depois, em Berlanga et al. [28] é apresentado um SFGBR do tipo GCCL, que baseado em PG realiza um estudo mais compreensivo das vantagens do uso da PG em SFGBR. O autor usa 24 bases de dados benchmarks e compara com 4 outros SFGBR presentes na literatura, sendo estes baseados em diferentes codificações (GCCL, IRL e Pittsburgh), assim como MHG distintas. O autor verifica que sua abordagem é mais eficiente, no sentido de melhor acurácia média e menor número de regras geradas. Este mesmo modelo é estendido em López et al. [142] para reconhecimento de padrões em ambiente onde as classes são intensamente desbalanceadas.

Por fim, em Muni e Pal [157] é exibida uma abordagem bastante similar ao proposto nesta dissertação. Os autores usam a PG Multigênica (PGMG) de modo a evoluir classificadores, onde cada árvore do indivíduo representa todo um conjunto de regras associadas a h -ésima classe, ou seja, para um problema de H classes é necessária a decodificação de H árvores no indivíduo. Este trabalho também se confunde com o de Koshiyama et al. [131], porém este último usa uma decodificação de solução mais direta (usa operações matemáticas que se confundem com os operadores fuzzy), enquanto que em Muni e Pal [157] as regras são decodificadas literalmente. Contudo, ambas as abordagens devem ser vistas como um caso especial do modelo GPFIS para

classificação, pois este permite que um indivíduo decodifique mais árvores do que classes necessárias. Além disso, a definição da classe que melhor se ajusta ao antecedente da regra é realizada por uma heurística, não sendo necessário a sua busca pela PGMG.

Um exame da literatura de PG para classificação é apresentado em Espejo et al. [72], que aborda alguns outros trabalhos concernentes ao uso de SFG baseados em PG.

2.5

Resumo

Este capítulo apresentou os principais conceitos referentes aos SFGs, tais como: terminologias, áreas de atuação de um SFG (Ajuste de Parâmetros ou Descoberta de Conhecimento), descrição das codificações mais comuns usadas nos SFGBR (Pittsburgh, Michigan, IRL e GCCL), aplicações e desenvolvimento teórico nas áreas de Classificação, Regressão, Previsão e Controle. Apesar de não cobertos ao longo da revisão bibliográfica realizada, observa-se que os assuntos de interesse atual na literatura são:

1. Métricas para avaliar interpretabilidade;
2. Duelo entre acurácia-interpretabilidade.

Referente ao primeiro objetivo, algumas edições especiais [11] debatem o tema, com o objetivo de definir a própria noção de interpretabilidade, tangenciando temas como complexidade, semântica, etc., além de propor novas métricas para avaliar o quanto um SIF é interpretável. Com relação ao segundo tema, apesar de não ser tão recente esta problemática [114], a forma que esta questão tem sido tratada com o uso de Algoritmos Evolutivos Multi-Objetivos (MOEA) tem despertado a atenção da comunidade de SFG [76].

Por fim, outro tema tem sido pouco explorado: previsão de séries temporais no contexto de SFGs, principalmente no de SFGBR. Na área de SIF esta aplicação é clássica, com o uso de SIFs do tipo Mamdani [44] e TSK [45], além de outras técnicas como Mapas Cognitivos Fuzzy [193], Sistemas Fuzzy Evolucionários [138] e Sistemas Neuro-Fuzzy [63]. Ainda, verifica-se certo excesso de SFGs elaborados para um tema em particular (classificação, por exemplo), sendo exíguas as modelagens genéricas que costumeiramente se consolidam na literatura e no uso generalizado pela comunidade, como por exemplo, o modelo ANFIS [121]. Por fim, os trabalhos na área de SFGs tem dado maior ênfase no desenvolvimento e aprimoramentos na parte MHG do SFG, delegando pouco espaço a avanços na parte do SIF do SFG.

O modelo GPFIS surge como uma arquitetura genérica, passível de atuar nas mais diferentes aplicações. A partir das terminologias apresentadas, é possível caracterizar o modelo GPFIS como do tipo Pittsburgh, cuja MHG embarcada é a Programação Genética Multigênica. Assim, procura-se uma arquitetura acurada que possibilite o rápido aprendizado de uma BR, com fácil inclusão de vários tipos de t-norma, t-conorma, operadores de agregação, negação e modificadores linguísticos, de modo a maximizar o número de flexibilidades, acurácia e a aceção linguística da solução ao usuário final. Para tanto, antes de apresentar formalmente o modelo GPFIS, torna-se necessário descrever o algoritmo evolutivo denominado Programação Genética Multigênica, uma generalização da Programação Genética clássica.

3 Programação Genética

3.1 Principais Conceitos

A Programação Genética (PG) [133, 136, 137, 171] é uma técnica da Computação Evolutiva inspirada nos conceitos de seleção natural e recombinação genética. Permite sintetizar programas de computador que solucionem problemas sem a necessidade de serem programados para tal. A partir da elaboração inicial de Koza [133], em pouco mais de 20 anos a área se expandiu consideravelmente, abrangendo diferentes campos de aplicação, tais como Biotecnologia [105, 198], Engenharia Elétrica [69, 192] e Finanças [160, 203], além de apresentar novas representações [31, 152], entre as quais as mais conhecidas:

- Programação Genética Linear [30, 31].
- Programação Genética Cartesiana [152, 153].

Existem ainda outras representações híbridas, como a Programação Genética Linear com Inspiração Quântica [67]. Esta dissertação aborda com maior ênfase a Programação Genética Multigênica ou de Múltiplas Árvores [102], que deve ser entendida como uma generalização da PG Clássica proposta por Koza [133]. A questão da generalização é apresentada com maior clareza na sessão seguinte, dedicada a este tipo de PG.

A PG, como conceito geral, é uma meta-heurística baseada no conceito de população de soluções. O espaço de busca da PG é formado a partir das possíveis combinações entre os chamados Terminais e as Funções ou Operações Matemáticas especificados abaixo:

- **Terminais:** variáveis de entrada (atributos da classificação, defasagens em séries temporais, etc.) e constantes (escalares).
- **Funções ou Operações Matemáticas:** operações unárias (logaritmo, exponencial, etc.), binárias (soma, subtração, etc.), ternárias (iterações como "for" e condicionais como o "if"), além de outras definidas pelo usuário.

Os Terminais expressam as quantidades que são combinadas e transformadas para a formação de uma solução. O meio de combinação, ou seja, a forma funcional da solução, é expressa a partir do conjunto de Operações Matemáticas. Para iniciar uma rotina de PG Clássica, o usuário, além de definir os Terminais e as Operações Matemáticas, deve configurar um conjunto de parâmetros básicos, de acordo com a Tabela 3.1.

Tabela 3.1: Parâmetros de controle de um algoritmo de PG.

Tamanho da População	Número de Gerações
Método de Seleção	Taxa de Elitismo
Taxa de Mutação	Taxa de Cruzamento
Altura da Árvore	Método de Geração da População Inicial
Taxa de Reprodução Direta	Taxa de Destruição
Taxa de Edição	Taxa de Encapsulamento

Observa-se que as três primeiras linhas apresentam parâmetros similares aos dos Algoritmos Genéticos (AGs). Contudo, ao analisar as três últimas, surgem seis novos parâmetros que são específicos para o método de PG. Os operadores de recombinação (edição, reprodução direta, cruzamento, mutação, encapsulamento e destruição) são apresentadas no tópico seguinte com maior ênfase. Os diferentes métodos de seleção, assim como os de geração inicial da população também são apresentados a seguir.

Por fim, a partir do estabelecimento dos conjuntos de Terminais e Operações Matemáticas e das configurações dos parâmetros que compõem a PG, o último ingrediente para iniciar a rotina da PG é definir a Função de Avaliação. Esta diz respeito ao objetivo que o usuário deseja alcançar. Por exemplo, pode refletir o erro de classificação em um problema de reconhecimento de padrões, ou o lucro obtido a partir de uma alocação de recursos em um problema de planejamento. A correta definição da Função de Avaliação é imprescindível para o pleno funcionamento do método.

O Algoritmo 1 ilustra o processo de execução da PG. Gera-se uma população inicial aleatória, tal que cada ente é formado por Terminais e Operações Matemáticas escolhidos de forma aleatória, segundo um método de geração da população inicial. Contudo o tamanho do indivíduo (altura da árvore) é limitado. Os indivíduos são avaliados e os de maior aptidão são selecionados por alguma heurística para a aplicação dos operadores de recombinação. Posteriormente, aplica-se um critério de parada, como, por exemplo, o número de gerações. Enquanto este critério não for atendido, incrementa-se o número de gerações e repetem-se as etapas.

Algorithm 1 Programação Genética

- 1: Passos Iniciais:
 - Definição dos Terminais e Operações Matemáticas
 - Configurar Demais Parâmetros
 - Estabelecer a Função de Avaliação
 - 2: Inicialização da População
 - 3: Avaliação
 - 4: **if** Critério de Parada foi Atingido **then** Retornar a População
 - 5: **else**
 - 6: **while** Critério de Parada não for Atingido **do**
 - 7: Seleção
 - 8: Recombinação
 - 9: Avaliação
 - 10: **end while**
 - 11: **end if**
 - 12: Retornar a Última População
-

3.1.1
Representação e Formação da Solução

Na PG clássica, cada indivíduo é denotado por uma expressão matemática, codificada por meio de uma estrutura em árvore (Figura 3.1).

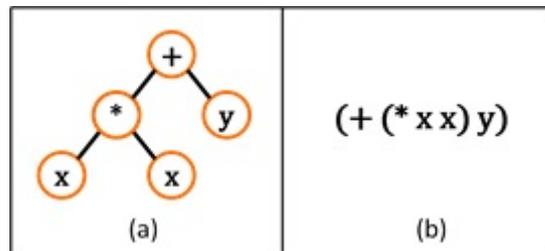


Figura 3.1: Representação do programa $x^2 + y$: (a) Representação sob a forma de árvore; (b) forma pré-fixada.

As Operações Matemáticas aparecem nos vértices internos da árvore. Enquanto isto, os Terminais aparecem nos vértices extremos (folhas). Para que os programas criados pela PG sejam válidos, é necessário que os conjuntos de Operações Matemáticas e de Terminais atendam à propriedade de clausura. Esta estabelece que as Operações Matemáticas devem ser escolhidas de modo a aceitar como argumento qualquer valor e tipo (presente no conjunto de Terminais), e o valor retornado pela operação deve ser assumido por qualquer outra Operação Matemática.

Dentre os diversos processos de geração inicial da população, destacam-se três:

- Full (Figura 3.2).

- Grow (Figura 3.3).
- Ramped Half-Half.

O método Full (Figura 3.2) de geração inicial de população estabelece que todos os ramos acima das folhas devem conter as Operações Matemáticas, enquanto que os Terminais se localizam nos nós compreendidos no nível mais baixo da árvore. Cada nó acima da extremidade inferior é composto por uma operação selecionada aleatoriamente do conjunto de Operações Matemáticas. O mesmo procedimento é realizado para a escolha dos Terminais. Caso só haja operações binárias, as árvores possuem a mesma largura, a qual pode variar consideravelmente no caso de de operações unárias e ternárias.

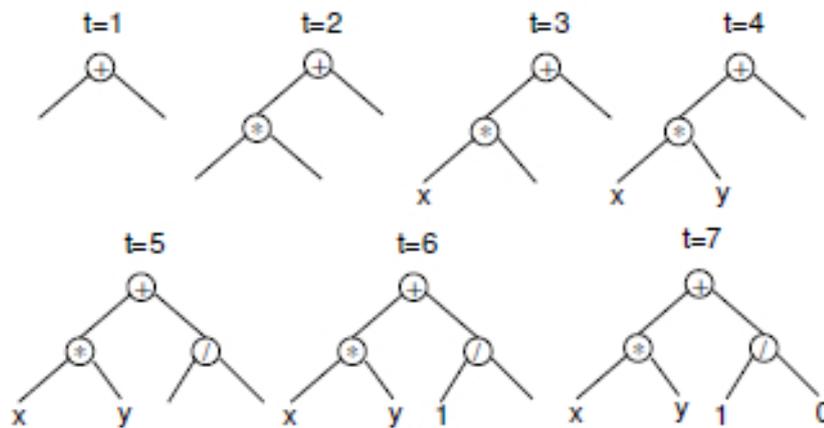


Figura 3.2: Elaboração de uma árvore pelo método Full, considerando a altura máxima como 2 ($t = \text{iteração}$). Reproduzido de Poli et al. [171].

O método Grow (Figura 3.3) apresenta uma abordagem que possibilita a geração de árvores com alturas diferentes, porém atingindo um máximo pré-especificado pelo usuário. A ideia é que para cada nó criado seja escolhida, de forma aleatória, uma componente do conjunto de Terminais ou Operações Matemáticas. Caso seja um Terminal, o processo de ramificação cessa; caso seja uma operação matemática, a ramificação prossegue.

O método Ramped Half-Half é uma mescla dos anteriores. Para cada método é estabelecida uma probabilidade e , dada a necessidade de se construir uma solução, é verificado qual método será usado. Em geral, e no caso desta dissertação, utiliza-se esta abordagem durante a execução das rotinas, pois ela possibilita a geração de uma gama de soluções com tamanhos e perfis diversificados.

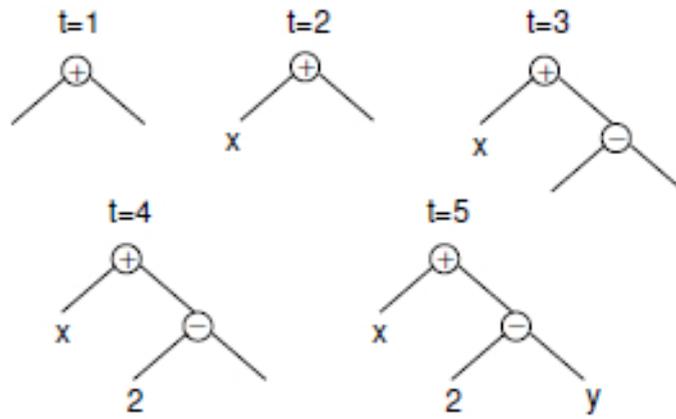


Figura 3.3: Elaboração de uma árvore pelo método Grow, considerando a altura máxima como 2 ($t = \text{iteração}$). Reproduzido de Poli et al. [171].

3.1.2 Avaliação e Seleção

O processo de avaliação se resume em avaliar a qualidade da solução decodificada em cada indivíduo. Para tanto, a medida de desempenho, ou, mais especificamente, a Função de Avaliação, deve estabelecer uma relação íntima com o problema específico. Por exemplo, se a questão tratada é aproximar a função $x^2 + x + 1$ no domínio $[a, b]$, como de acordo com a Figura 3.4, uma possível métrica é a diferença entre a função $x^2 + x + 1$ e a equação expressa no indivíduo.

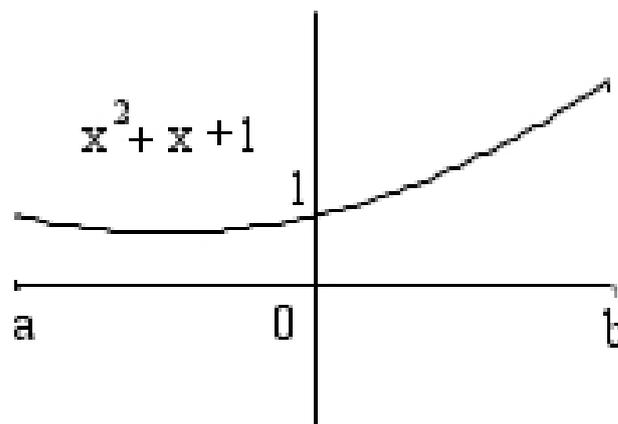


Figura 3.4: Exemplo de função $x^2 + x + 1$. Reproduzido de Poli et al. [171].

Considerem-se quatro soluções, conforme a Figura 3.5. Para avaliar cada função expressa na árvore, pode-se fazer uma comparação gráfica com a função alvo e verificar qual a mais próxima (Figura 3.6). No caso, as expressões (a)

e (b) são as mais próximas, enquanto que (c) e (d) não apresentaram bom desempenho.

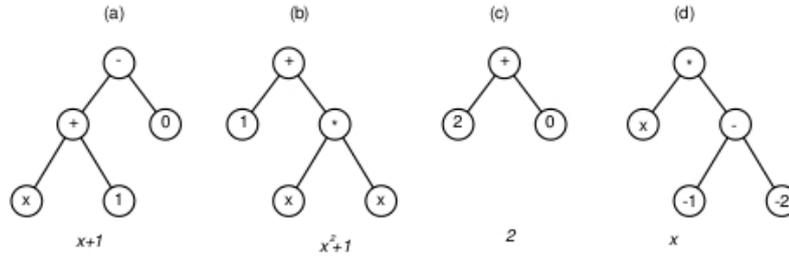


Figura 3.5: Soluções iniciais.

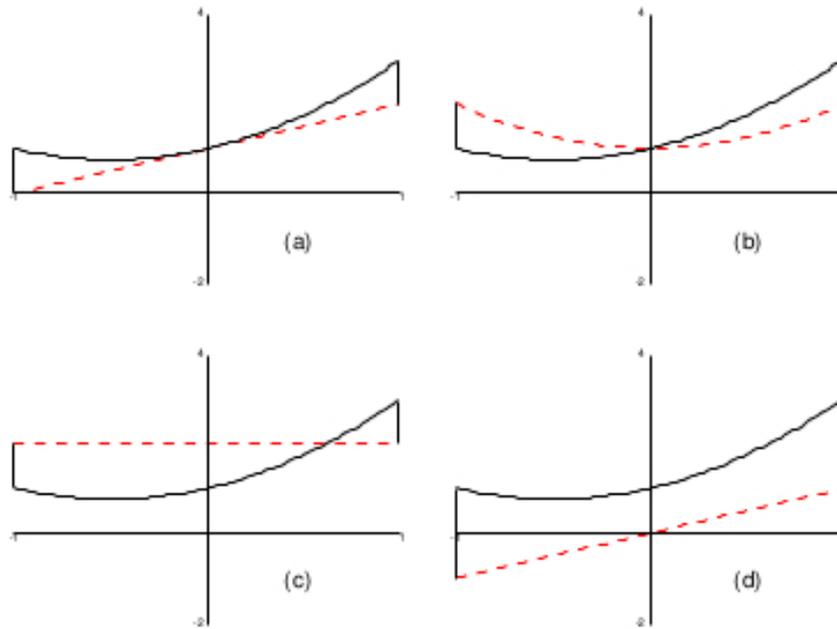


Figura 3.6: Exemplo de aproximação de cada solução a função alvo. Reproduzido de Poli et al. [171].

A partir da finalização do processo de avaliação dos indivíduos, estes são submetidos à seleção. Uma proporção dos melhores indivíduos, definida pela taxa de elitismo, fará parte da próxima população. O segundo processo de seleção dá-se a partir da definição de um método. O mais comum em PG é o do torneio [171], em que se seleciona um número T (definido a priori pelo usuário) de indivíduos da população e se observam os seus desempenhos. O melhor indivíduo é então selecionado para a formação da próxima população. Este procedimento é repetido um número de vezes para dar início ao processo de recombinação.

3.1.3 Recombinação

Neste instante, os indivíduos selecionados para recombinação sofrem a ação dos operadores. Em suma os operadores de recombinação são [133]:

- **Reprodução Direta:** o indivíduo da população de recombinação é selecionado e copiado, sem qualquer alteração, para a próxima geração (Figura 3.7).

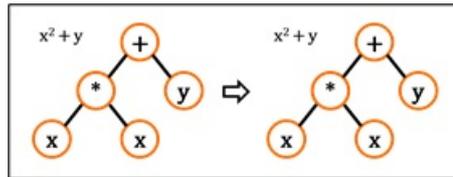


Figura 3.7: Operador genético de Reprodução Direta para PG.

- **Cruzamento:** dois indivíduos são selecionados e um ponto de corte é escolhido aleatoriamente em cada um dos genitores. As subárvores resultantes são permutadas, formando novos indivíduos para a próxima geração (Figura 3.8).

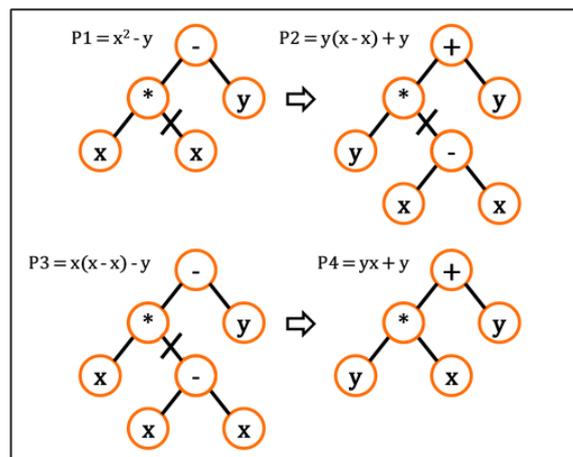


Figura 3.8: Operador genético de Cruzamento para PG.

- **Mutação:** seleciona, aleatoriamente, um ponto interno ou externo da árvore. Em seguida, esta operação remove a subárvore que está neste ponto e insere uma nova subárvore, gerada aleatoriamente (Figura 3.9).
- **Encapsulamento:** identifica subárvores potencialmente úteis e atribui a elas um nome para que sejam referenciadas e utilizadas posteriormente. Para esta atribuição, seleciona-se um ponto interno de um indivíduo, remove-se a subárvore no ponto selecionado e uma nova função é definida

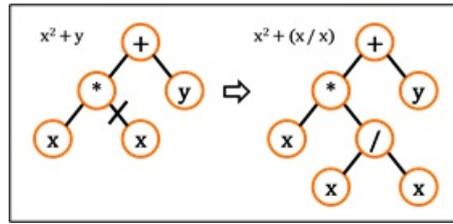


Figura 3.9: Operador genético de Mutação para PG.

para referenciar esta subárvore. A esta nova função é atribuído um nome e o conjunto de Operações Matemáticas é acrescido desta nova função. No ponto selecionado do indivíduo, uma chamada da função encapsulada é criada. Este tem papel essencial para interromper o efeito do cruzamento, pois a função torna-se um ponto indivisível (Figura 3.10);

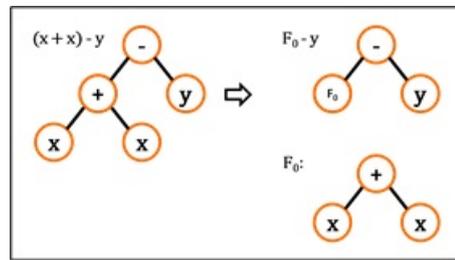


Figura 3.10: Operador genético de Encapsulamento para PG.

- **Edição:** proporciona um meio para editar e simplificar expressões (por exemplo, no domínio booleano, um programa “(and x x)” é simplificado a “x”, “(not(not(x)))” é simplificado a “x”).
- **Destruição:** é uma forma de reduzir o número de indivíduos com baixa aptidão nas primeiras gerações. É controlado por dois parâmetros, a porcentagem de indivíduos mantidos e a condição que especifica quando este operador será chamado.

De fato, nem todos os operadores de recombinação apresentados são sempre usados. Os principais são o cruzamento, mutação e reprodução direta. Tais operadores são também empregados na Programação Genética Multigênica, tema da próxima seção.

3.2 Programação Genética Multigênica

A Programação Genética Multigênica (PGMG) [102, 191] pode ser encarada como uma generalização da PG tradicional, pois denota um indivíduo como um complexo de estrutura em árvores, que, da mesma forma que

na PG, recebe um conjunto de Terminais X_{ij} , buscando prever a saída Y_i . Recentemente, a PGMG tem sido aplicada com sucesso em problemas benchmark como classificação de padrões [131, 157] e regressão (conhecida na área como regressão simbólica) [107], e também em aplicações reais: melhora do desempenho na recuperação de óleo e gás [73] e em aplicações de engenharia civil [175].

3.2.1 Representação e Formação da Solução

A representação da PGMG é similar ao da PG no tocante à estrutura em árvore. Contudo, um indivíduo para a PGMG é um complexo de estruturas em árvore, conforme Figura 3.11.

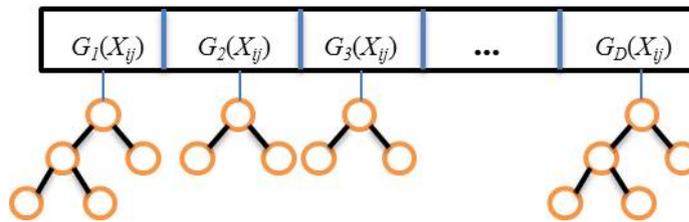


Figura 3.11: Exemplo de um indivíduo multigênico.

Cada árvore desta estrutura é uma solução parcial para o problema, sendo que a saída final pode ser resultado da combinação linear, dada por:

$$\hat{Y}_i = \beta_0 + \sum_{d=1}^D \beta_d G_d \quad (3-1)$$

no qual \hat{Y}_i é a saída resultante de um indivíduo qualquer da população, β_d são parâmetros lineares e G_d ($d = 1, \dots, D$) são as saídas de cada equação decodificada no indivíduo. Os β_d desse método podem ser estimados por mínimos quadrados, ou valores quaisquer podem ser atribuído para cada equação β_d (média aritmética, por exemplo). Dado \hat{Y}_i é possível avaliar a qualidade global do indivíduo. Também, é factível usar cada árvore de forma separada, por exemplo, em um problema de classificação onde uma função discriminante é elaborada por cada árvore representante de uma classe. Assim, o erro computado por cada árvore é agregado utilizando uma média aritmética ou outro operador de agregação. É fácil ver que quando $D = 1$, a PGMG reduz-se à solução obtida por uma PG clássica. O processo de avaliação e seleção são efetuados de forma similar a PG Clássica.

3.2.2 Recombinação

Em relação aos operadores de recombinação, a operação de mutação na PGMG é similar à efetuada na PG clássica. No caso da operação de cruzamento, é necessária uma distinção no nível em que a operação é realizada, sendo possível aplicar o cruzamento no baixo e no alto níveis. A Figura 3.12 apresenta um indivíduo multigênico com cinco equações ($D = 5$) efetuando uma operação de cruzamento de baixo nível, enquanto a Figura 3.13 apresenta a operação de mutação.

O baixo nível é o espaço onde é possível manipular as estruturas (Terminais e Operações Matemáticas) das equações presentes em um indivíduo. No caso, tanto a mutação quanto o cruzamento de baixo nível na PGMG são semelhantes ao modo efetuado na PG clássica.

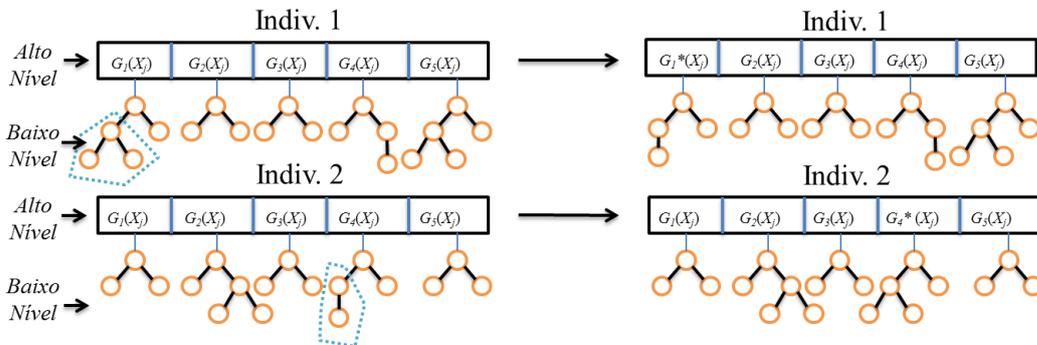


Figura 3.12: Operação de cruzamento de baixo nível.

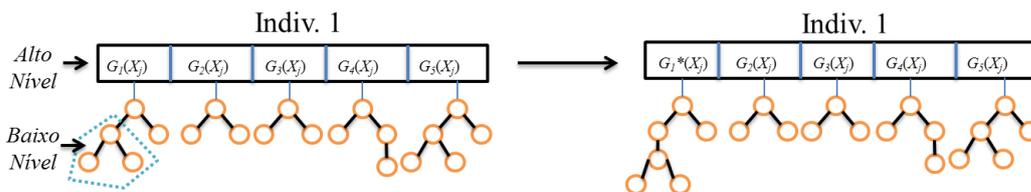


Figura 3.13: Operação de mutação.

Um exemplo de cruzamento de alto nível para um indivíduo com cinco equações é apresentado na Figura 3.14. O alto nível é o espaço em que se manipulam de forma macro as equações presente no indivíduo. Logo, verifica-se que, a partir de dois pontos aleatórios, são permutadas equações de um indivíduo para o outro. Este ponto de corte aleatório pode ser simétrico, quando se troca o mesmo número de equações entre indivíduos, ou assimétrico,

caso contrário. Os efeitos do cruzamento de alto nível tendem a afetar mais substancialmente a saída resultante do que a operação de cruzamento de baixo nível e a mutação.

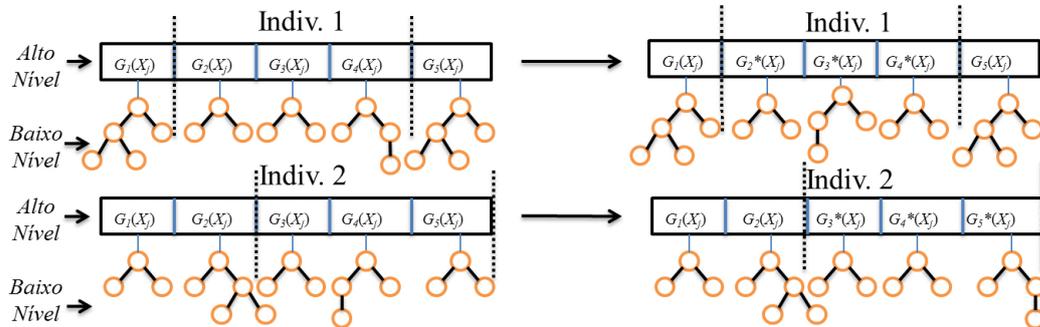


Figura 3.14: Operação de cruzamento de alto nível.

Ainda, há a presença do operador de reprodução direta, sendo seu uso idêntico ao da PG Clássica. Em linhas gerais, o procedimento evolutivo da PGMG se diferencia da PG pela adição de dois parâmetros:

- Número máximo de árvores por indivíduo.
- Taxa de Cruzamento de Alto Nível.

No caso do número máximo de árvores por indivíduo, sempre se utiliza um valor elevado para que não haja empecilhos no processo de sintetização da solução. Com respeito à taxa de cruzamento de alto nível, esta é um parâmetro que deve ser ajustado, sendo sempre apresentada na tabela de configurações do algoritmo durante as execuções efetuadas.

Através dos elementos exibidos sobre a PGMG, o próximo capítulo trata do modelo GPFIS. Neste a PGMG é usada como meta-heurística para formação da base de regras fuzzy, de modo que cada árvore presente nos indivíduo decodifica uma regra (mais especificamente somente os termos antecedentes da regra). Cada indivíduo fornece o seu conjunto de regras, de maneira que o modelo GPFIS explora métodos e operadores para refinar a solução expressa em cada codificação.

3.3 Resumo

Este capítulo apresentou os principais conceitos referentes a PG e a sua generalização PGMG. Com a base da PGMG, o modelo GPFIS usa as árvore (função) codificada em um indivíduo da PGMG como um conjunto de

premissas de uma base de regras fuzzy. A partir dessas premissas, o modelo GPFIS executa uma sequência de procedimentos para tornar essas em regras fuzzy.

4

Modelo Genetic Programming Fuzzy Inference System (GPFIS)

Este capítulo aborda o modelo Genetic Programming Fuzzy Inference System (GPFIS). São descritas suas etapas de construção, desde o mapeamento de valores precisos em graus de pertinência a conjuntos fuzzy, até ao procedimento de Inferência, Defuzzificação/Decisão e Avaliação. A seção 4.1 apresenta as definições e notações utilizadas nas demais seções. A seção 4.2 descreve a arquitetura do modelo GPFIS, enquanto que na seção 4.3 é apresentado um resumo do capítulo. O Anexo A, destinado a leitores não familiarizados com Sistemas Fuzzy, resume um conjunto de definições e descrição de operadores empregados ao longo da construção do modelo.

4.1

Definições e Notações

Seja $\mathbf{x}_i = [x_{i1}, \dots, x_{iJ}]$ o i -ésimo padrão, objeto ou amostra da base de dados ($i = 1, \dots, n$), formada por cada observação x_{ij} obtida em cada j -ésima variável X_j ($j = 1, \dots, J$). Suponha que X_j está cotada inferiormente por a e superiormente b , isto é, se $x_{ij} \in X_j$, então $a \leq x_{ij} \leq b$. Denote por A_{lj} o l -ésimo conjunto fuzzy associado a j -ésima variável, ou seja, $A_{lj} = \{(x_{ij}, \mu_{A_{lj}}(x_{ij})) | x_{ij} \in X_j\}$ para todo $l = 1, \dots, L$, munido da função de pertinência $\mu_{A_{lj}} : X_j \rightarrow [0, 1]$. Logo, um conjunto fuzzy é um par que associa cada x_{ij} a um grau de pertinência (compatibilidade) $\mu_{A_{lj}}(x_{ij})$ ao conjunto fuzzy A_{lj} [22]. Um dos formatos tradicionais para as funções de pertinência é a triangular. A função de pertinência triangular é dada por:

$$\mu_{A_{lj}}(x_{ij}, \alpha, \beta, \gamma) = \max\left(\min\left(\frac{x_{ij} - \alpha}{\beta - \alpha}, \frac{\gamma - x_{ij}}{\gamma - \beta}\right)\right) \quad (4-1)$$

que, para o caso de A_{1j}, \dots, A_{5j} , pode ser caracterizada pela Figura 4.1.

Considere: $a = \min(X_j)$, $b = \max(X_j)$, $c = (a + b)/2$, $d = (a + c)/2$ e $e = (c + b)/2$. Neste caso (Figura 4.1), as funções de pertinência associadas a

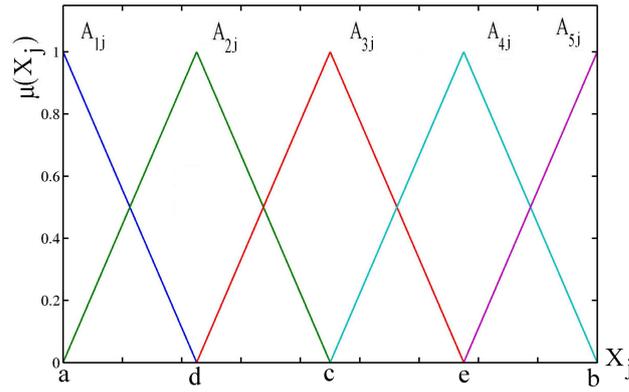


Figura 4.1: Exemplo de função de pertinência triangular.

cada conjunto fuzzy são descritas por:

$$A_{1j} = \{(x_{ij}, \mu_{A_{1j}}(x_{ij}, a - d, a, d) | x_{ij} \in X_j)\} \quad (4-2)$$

$$A_{2j} = \{(x_{ij}, \mu_{A_{2j}}(x_{ij}, a, d, c) | x_{ij} \in X_j)\} \quad (4-3)$$

$$A_{3j} = \{(x_{ij}, \mu_{A_{3j}}(x_{ij}, d, c, e) | x_{ij} \in X_j)\} \quad (4-4)$$

$$A_{4j} = \{(x_{ij}, \mu_{A_{4j}}(x_{ij}, c, e, b) | x_{ij} \in X_j)\} \quad (4-5)$$

$$A_{5j} = \{(x_{ij}, \mu_{A_{5j}}(x_{ij}, e, b, b + e) | x_{ij} \in X_j)\} \quad (4-6)$$

a partir destas informações, a Tabela 4.1 mostra uma típica matriz de dados que contém em suas linhas informações de cada padrão da base de dados e, em suas colunas, informações relativas as variáveis e aos graus de pertinência aos respectivos conjuntos fuzzy.

Tabela 4.1: Exemplo da matriz de dados.

Padrão	Variável			Conjuntos Fuzzy				
	X_1	...	X_J	$\mu_{A_{11}}(X_1)$...	$\mu_{A_{L1}}(X_1)$...	$\mu_{A_{LJ}}(X_J)$
1	x_{11}	...	x_{1J}	$\mu_{A_{11}}(x_{11})$...	$\mu_{A_{L1}}(x_{11})$...	$\mu_{A_{LJ}}(x_{1J})$
2	x_{21}	...	x_{2J}	$\mu_{A_{11}}(x_{21})$...	$\mu_{A_{L1}}(x_{21})$...	$\mu_{A_{LJ}}(x_{2J})$
.
.
.
i	x_{i1}	...	x_{iJ}	$\mu_{A_{11}}(x_{i1})$...	$\mu_{A_{L1}}(x_{i1})$...	$\mu_{A_{LJ}}(x_{iJ})$
.
.
.
n	x_{n1}	...	x_{nJ}	$\mu_{A_{11}}(x_{n1})$...	$\mu_{A_{L1}}(x_{n1})$...	$\mu_{A_{LJ}}(x_{nJ})$

Assim, as primeiras J colunas representam cada uma das variáveis,

enquanto que as linhas os n padrões. Ao lado das J variáveis são dispostas as imagens de cada x_{ij} nas funções de pertinência $\mu_{A_{1j}}(x_{ij}), \dots, \mu_{A_{Lj}}(x_{ij})$. Observa-se a presença de dois espaços: o dos objetos $\mathbf{x}_i = [x_{i1} \ x_{i2} \ \dots \ x_{iJ}] \in \mathbb{R}^J$ e o das variáveis $X_j = [x_{1j} \ x_{2j} \ \dots \ x_{nj}]^T \in \mathbb{R}^n$ [122]. O interessante nesse ponto é que cada $\mu_{A_{lj}}$ induz também um espaço, tanto de objetos $\mu_{A_j}(x_{ij}) = [\mu_{A_{1j}}(x_{ij}) \ \mu_{A_{2j}}(x_{ij}) \ \dots \ \mu_{A_{Lj}}(x_{ij})] \in [0, 1]^L$ como de variáveis $\mu_{A_j}(X_j) = [\mu_{A_{1j}}(x_{1j}) \ \mu_{A_{1j}}(x_{2j}) \ \dots \ \mu_{A_{1j}}(x_{nj})]^T \in [0, 1]^n$. Este fato é explorado na concepção e formulação de métricas para a associação entre antecedentes e consequentes de regras fuzzy.

Por fim, a notação y_i representa a i -ésima observação da variável de saída Y (variável dependente, resposta, etc.), ao passo que $B_k, k = 1, \dots, K$ são os conjuntos fuzzy em Y . Como o modelo GPFIS é genérico, isto é, voltado para solucionar problemas de Classificação, Previsão, Regressão e Controle, as terminologias relativas a \mathbf{x}_i e y_i devem ser ajustadas para cada caso. Por exemplo, em Previsão $y_i = y_t$, isto é, o valor da série temporal no instante t , enquanto $\mathbf{x}_i = \mathbf{y}_{t,P} = [y_{t-1}, y_{t-2}, \dots, y_{t-P}]$ são as defasagens, ou atrasos da série temporal. Em Controle $y_i = y_t$ representa a ação do controlador no tempo t , enquanto que as variáveis de entrada são os estados da planta (note que em Controle não se possui uma base de dados prévia com o valor de y_t para cada \mathbf{x}_t - aprendizado por reforço). Por fim, em Classificação $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{iJ}]$ é o i -ésimo padrão formado pelas observações dos J atributos X_j , necessários para discriminar a classe $y_i = C_i \in \mathbb{Z}_+$.

4.2 Arquitetura Básica

Este tópico apresenta o modelo GPFIS. Cada uma das seções subsequentes tratará de uma etapa da modelagem: Fuzzificação, Inferência, Defuzzificação/Decisão e Avaliação. A Figura 4.2 expõe de forma macro os passos para o ajuste do modelo GPFIS, enquanto que a seguir apresenta-se o pseudo-código do modelo GPFIS.

Como o GPFIS é um SFG do tipo Pittsburgh [98], o seu principal objetivo é elaborar uma base de regras concisa. A seguir são descritas suas etapas de construção, iniciando pelo mapeamento de valores precisos em graus de pertinência a conjuntos fuzzy. O procedimento de Inferência Fuzzy é subdividido na elaboração das premissas de regras fuzzy (Formulação), na definição do consequente mais apropriado (Particionamento) e na união das ativações de cada regra fuzzy (Agregação). Por fim, são realizadas as etapas de Defuzzificação/Decisão e a Avaliação. De forma geral, o modelo GPFIS

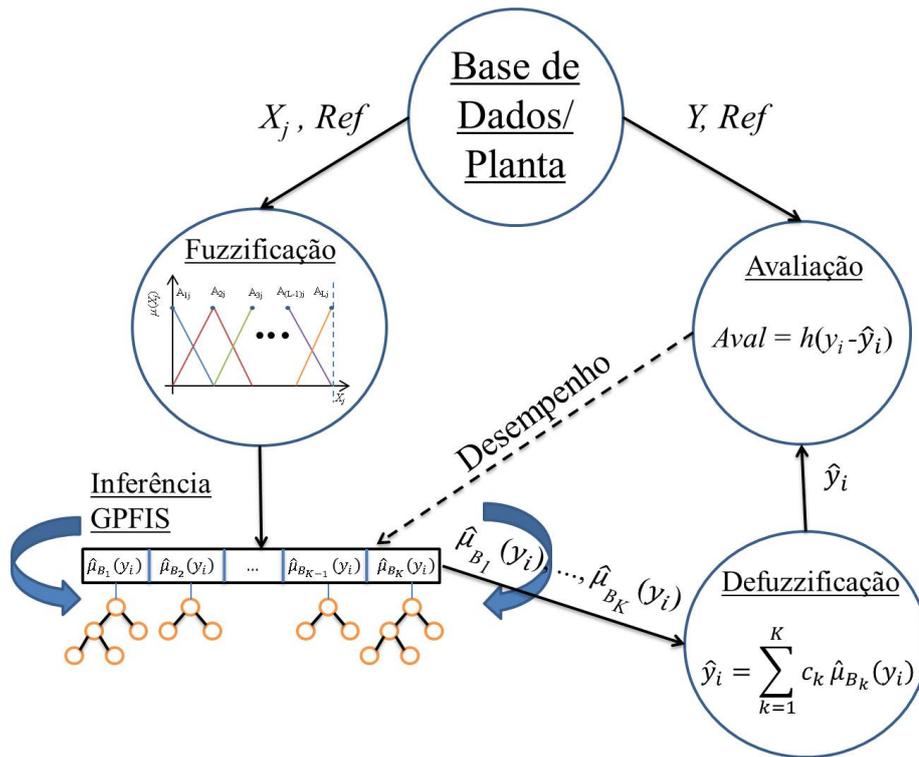


Figura 4.2: Diagrama genérico das etapas de elaboração do modelo GPFIS.

Algorithm 2 Modelo GPFIS

- 1: Passos Iniciais:
 - Adequação da Base de Dados ou Modelo da Planta
 - Definição dos Conjuntos Fuzzy
 - 2: Inicialização da População
 - 3: **for** i **do** 1 Número Total de Gerações
 - 4: Fuzzificação das Variáveis de Entrada
 - 5: Inferência:
 - Formulação
 - Particionamento
 - Agregação
 - 6: Defuzzificação/Decisão
 - 7: Avaliação
 - 8: Seleção e Recombinação
 - 9: **end for**
 - 10: Retornar a Última População
-

assemelha-se aos seus pares Fuzzy-Genéticos com maior intensidade na etapa de Fuzzificação, Defuzzificação/Decisão e Avaliação. A grande distinção ocorre na etapa de Inferência, que inicialmente exibe um resultado que relaciona intimamente y_i com suas funções de pertinência $\mu_{B_k}(y_i)$. Com este resultado, o modelo GPFIS busca aproximações razoáveis para $\mu_{B_k}(y_i)$, usando informações

provenientes de $\mu_{A_{ij}}(x_{ij})$ que são processadas nas subetapas da Inferência: Formulação-Particionamento-Agregação.

4.2.1 Fuzzificação

Esta etapa lida com o mapeamento dos valores crisp das variáveis de entrada/saída a graus de pertinência a conjuntos fuzzy. Definidos os pares de entrada e saída¹, x_{ij} e y_i respectivamente, torna-se necessário a definição dos respectivos conjuntos fuzzy de cada variável, denotados respectivamente por A_{ij} e B_k . Esta etapa leva em conta dois fatores:

- Forma funcional e definição do suporte de cada função de pertinência $\mu_{A_{ij}}(x_{ij})$ e $\mu_{B_k}(y_i)$.
- Rótulo linguístico apropriado, qualificando o subespaço compreendido pela função de pertinência com um adjetivo correspondente ao contexto.

Em teoria, ambas as tarefas devem ser desempenhadas por um especialista, cuja capacidade possibilitaria auferir ganhos de interpretabilidade e explicabilidade nas regras fuzzy geradas. Na prática, devido à dificuldades de encontrar um profissional capaz de qualificar o problema em análise (em alguns casos este não existe), é bastante comum autores [18, 28, 57, 112, 117] optarem por uma partição uniforme conforme a Figura 4.3.

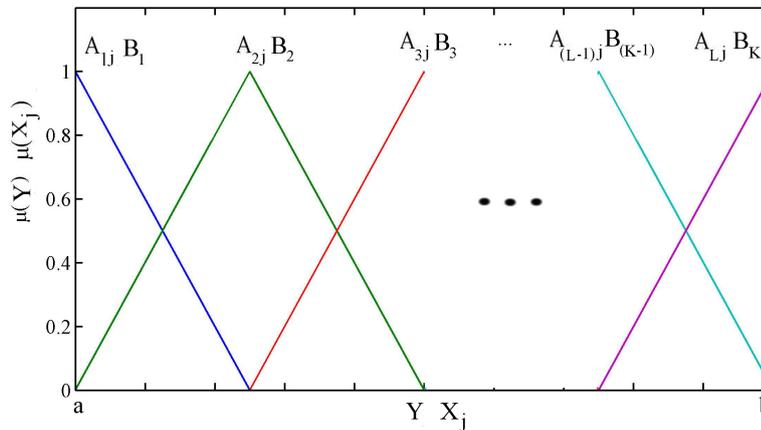


Figura 4.3: Partição uniforme das funções de pertinência normais para as variáveis X_j e Y .

Em grande parte dos problemas de Classificação, Regressão e Previsão são considerados 5 conjuntos fuzzy por variável ($L = 1, \dots, 5$). No caso de

¹Para facilitar tanto a notação e exposição, foi somente considerado a existência de uma única variável de saída para o modelo (sistema do tipo MISO). Contudo, para os casos de mais de uma (modelo multivariado), basta a inclusão de um subscrito em y_i , tornando-o y_{iq} , onde q representa a q -ésima saída, por exemplo.

Controle, este número geralmente cresce para 7 conjuntos fuzzy por variável ($L = 1, \dots, 7$). É predominante o formato triangular de forma geral [4, 5, 28, 50, 165, 195], entretanto empregam-se também funções de pertinência gaussianas [141, 172, 202].

É também comum [5, 116] não se fixar o número de funções de pertinência por variável, mas sim prover um conjunto de possibilidades. Isto, segundo Figura 4.4, proporciona ao algoritmo de busca um portfólio de opções para escolha dos termos antecedentes para as regras. O viés negativo desta abordagem é a interferência na acepção das regras fuzzy geradas.

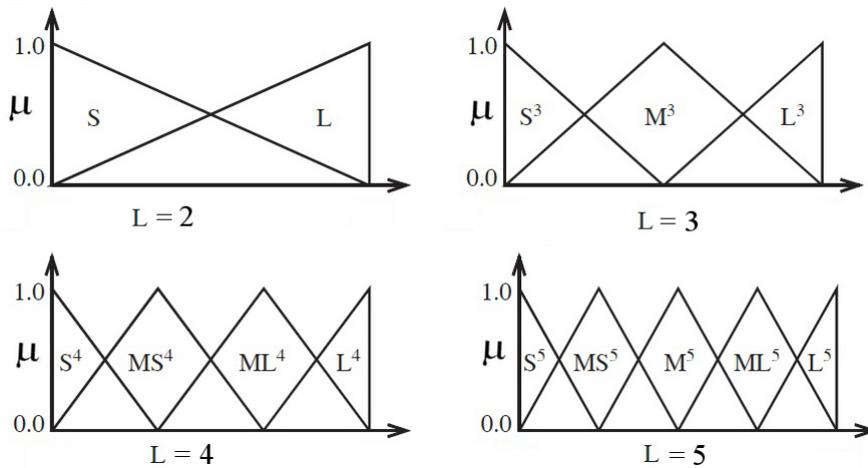


Figura 4.4: Alternativas de granularidade para as variáveis.

A descrição mais usada nos resultados do modelo GPFIS é equivalente à Figura 4.3. Esta escolha deve-se tanto à proposição apresentada a seguir, pois fornece imediatamente um método de defuzzificação, quanto ao nível elevado de compreensão linguística das soluções geradas. Contudo, o uso de formatos Trapezoidais, Gaussianas, etc., com suportes de tamanho diferentes não invalida o restante da metodologia, pois esta continua sendo válida.

4.2.2 Inferência

Este tópico aborda o mecanismo de Inferência do modelo GPFIS. De modo a embasar o procedimento elaborado, inicia-se com uma proposição que une $y_i \in Y$ a seus $\mu_{B_k}(y_i)$. Esta proposição abre a possibilidade de buscar aproximações $\hat{\mu}_{B_k}(y_i)$ para $\mu_{B_k}(y_i)$. O modelo GPFIS investe nesse sentido ao propor um processo de Inferência em três etapas: Formulação-Particionamento-Agregação. Assim, ao usar informações provenientes dos $\mu_{A_{ij}}(x_{ij})$ deseja-se buscar boas aproximações $\hat{\mu}_{B_k}(y_i)$ que sejam acuradas e interpretáveis.

Proposição

Dado $y_i \in Y$, considere-se que têm como conjuntos fuzzy B_1, B_2, \dots, B_K , fazendo-se uso de uma função de pertinência $\mu_{B_k} : Y \rightarrow [0, 1]$ triangular conforme a equação (4-1). Graficamente, cada B_k ($k = 1, \dots, K$) pode ser descrito por:

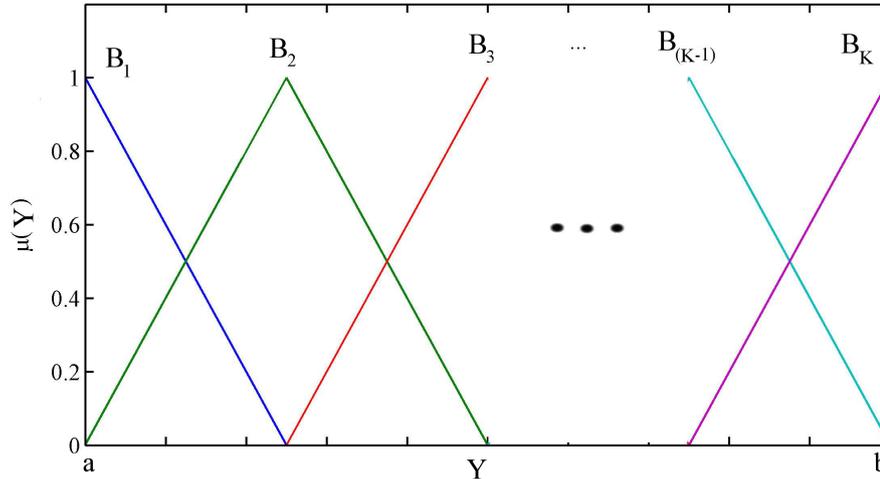


Figura 4.5: Exemplo de granularidade para Y .

Os parâmetros de cada função de pertinência exibida na Figura 4.5 podem ser facilmente obtidos a partir de três fatos:

- As funções de pertinência são 2-justapostas, isto é:
 $\min(\mu_{B_s}(y_i), \mu_{B_f}(y_i), \mu_{B_k}(y_i)) = 0$, para todo $s \neq f \neq k$ e $s, f, k \in \{1, 2, \dots, K\}$.
- As funções de pertinência são normais, isto é, $\exists c_k \in Y$ tal que $\max_{y_i}(\mu_{B_k}(y_i)) = \mu_{B_k}(c_k) = 1$. Obviamente um conjunto fuzzy é normal se a sua função de pertinência é também normal.
- As funções de pertinência estão uniformemente particionadas no universo de discurso de Y (a pré-imagem do vértice superior de cada j -ésima função de pertinência (c_j) é computado recursivamente por:
 $c_1 = \min(Y)$ e $c_{j+1} = c_j + \frac{(\min(Y) - \max(Y))}{j-1}$ para todo $j > 1$).

Isto é visível na Figura 4.1 e equações (4-2-4-6). A partir disso, considere-se a seguinte proposição:

Proposição 1. *Considere $y_i \in Y$ com K conjuntos fuzzy $B_k, k = 1, \dots, K$ normais, triangulares e 2-justapostos. Então y_i pode ser reescrito por:*

$$y_i = c_1\mu_{B_1}(y_i) + c_2\mu_{B_2}(y_i) + \dots + c_K\mu_{B_K}(y_i) \tag{4-7}$$

onde c_k define o valor modal da k -ésima função de pertinência.

Prova. Pela 2-justaposição ocorre que: $\min(\mu_{B_d}(y_i), \mu_{B_f}(y_i), \mu_{B_s}(y_i)) = 0$, $\forall d \neq f \neq s$ e $d, f, s \in \{1, 2, \dots, K\}$, mas vale $\min(\mu_{B_d}(y_p), \mu_{B_l}(y_p)) \geq 0$.

Seja $R = \{\mu_{B_1}(y_i), \dots, \mu_{B_K}(y_i)\}$ e se considerem duas possibilidades:

- Caso 1: Somente um elemento em R é maior do que 0.
- Caso 2: Há dois elementos em R maiores do que 0.

Caso 1: a solução é trivial, pois, se há somente um elemento em R maior do que 0, isto significa que $\mu_{B_k}(y_i) = 1$. O único valor que satisfaz a $\mu_{B_k}(y_i) = 1$ é y_i é igual a c_k (devido à normalização). A equação (4-7) afirma que: $y_i = c_1\mu_{B_1}(y_i) + c_2\mu_{B_2}(y_i) + \dots + c_K\mu_{B_K}(y_i)$ e, como: $R = \{0, 0, \dots, 1, \dots, 0\}$, tem-se: $y_i = c_k\mu_{B_k}(y_i)$ (o restante dos termos é zero).

Como visto, para que $\mu_{B_k}(y_i) = 1$, é necessário que $y_i = c_k$, e, finalmente:

$$y_i = c_k\mu_{B_k}(y_i) \Rightarrow c_k = c_k\mu_{B_k}(c_k) \Rightarrow c_k = c_k \quad (4-8)$$

Caso 2: Sem perda de generalidade, considere-se que y_i é tal que $\min(\mu_{B_1}(y_i), \mu_{B_2}(y_i)) > 0$. Como ilustração, veja a Figura 4.6.

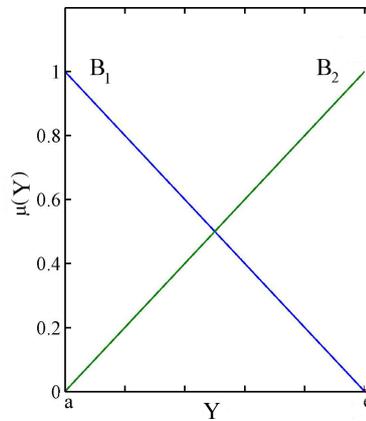


Figura 4.6: Funções de pertinência relevantes, devido à propriedade da 2-justaposição.

Como houve duas ativações, o restante dos $\mu_{B_k}(y_i) = 0, \forall k > 2$, logo:

$$y_i = c_1\mu_{B_1}(y_i) + c_2\mu_{B_2}(y_i) + \dots + c_K\mu_{B_K}(y_i) \quad (4-9)$$

$$y_i = c_1\mu_{B_1}(y_i) + c_2\mu_{B_2}(y_i) \quad (4-10)$$

é possível reescrever: $\mu_{B_1}(y_i) = \frac{e-y_i}{e-a}$ e $\mu_{B_2}(y_i) = \frac{y_i-a}{e-a}$

$$y_i = c_1 \left(\frac{e - y_i}{e - a} \right) + c_2 \left(\frac{y_i - a}{e - a} \right) \quad (4-11)$$

finalmente, observando que $c_1 = a$ e $c_2 = e$:

$$\begin{aligned} y_i &= a \left(\frac{e - y_i}{e - a} \right) + e \left(\frac{y_i - a}{e - a} \right) \\ y_i &= y_i \left(\frac{e - a}{e - a} \right) \Rightarrow y_i = y_i \quad \square \end{aligned} \quad (4-12)$$

A partir da proposição enunciada, tem-se as seguintes conclusões:

1. Se são conhecidos os $\mu_{B_k}(y_i)$, então também é conhecido y_i .
2. Se não se conhecem os $\mu_{B_k}(y_i)$, mas uma estimativa $\hat{\mu}_{B_k}(y_i)$ destes, tal que $\sup_{y_i} |\mu_{B_k}(y_i) - \hat{\mu}_{B_k}(y_i)| \leq \epsilon$, quando $\epsilon \rightarrow 0$ tem-se que a defuzzificação ótima, ou seja, o valor \hat{y}_i que aproxima y_i , é dado por:

$$\hat{y}_i = c_1 \hat{\mu}_{B_1}(y_i) + c_2 \hat{\mu}_{B_2}(y_i) + \dots + c_K \hat{\mu}_{B_K}(y_i) \quad (4-13)$$

ambas as conclusões são de extrema importância, tanto prática quanto teoricamente. Porém, cabe ressaltar que a expressão (4-13) é relevante quando é possível mensurar a distância entre $\mu_{B_k}(y_i)$ e $\hat{\mu}_{B_k}(y_i)$ (em problemas de controle isto não é sempre possível), além de o método de defuzzificação funcionar plenamente somente quando $\hat{\mu}_{B_k}(y_i) \approx \mu_{B_k}(y_i)$. Portanto, fora desse regime, um método clássico como Média dos Máximos ou Centro de Gravidade [130, 179] pode ocasionalmente proporcionar um desempenho melhor.

A questão é como gerar boas estimativas de $\hat{\mu}_{B_k}(y_i)$, tendo em conta as restrições induzidas por $\mu_{B_k}(y_i)$ (por exemplo, de domínio pois $\mu_{B_k}(y_i) \in [0, 1]$), além de procurar uma solução acurada e linguisticamente interpretável. Nas aplicações, y_i é a variável de interesse (preço, tensão, etc.) e o mais comum é conhecer a priori informações sobre os x_{ij} (idade, temperatura, etc.). A abordagem clássica usa os x_{ij} e elabora um modelo $f : \mathbb{R}^J \rightarrow \mathbb{R}$ com o desejo de minimizar: $\sum_{i=1}^n |y_i - f(x_{i1}, \dots, x_{iJ})|$.

Para o caso de $\mu_{B_k}(y_i)$, visando à geração de uma solução interpretável, não se deve usar x_{ij} , mas sim as informações advindas de $\mu_{A_{ij}}(x_{ij})$. Portanto, deve-se construir uma $f : [0, 1]^J \rightarrow [0, 1]$, que satisfaça as restrições e permita gerar uma solução interpretável. Por exemplo, considerem-se as seguintes

soluções:

$$\hat{\mu}_{B_1}(y_i) = \mu_{A_{11}}(x_{i1})\mu_{A_{32}}(x_{i2}) \quad (4-14)$$

$$\hat{\mu}_{B_1}(y_i) = \mu_{A_{11}}(x_{i1}) + \mu_{A_{32}}(x_{i2}) \quad (4-15)$$

$$\hat{\mu}_{B_1}(y_i) = \frac{\mu_{A_{11}}(x_{i1}) + \mu_{A_{32}}(x_{i2})}{2} \quad (4-16)$$

a primeira expressão (4-14) é viável, pois o produto de dois números entre $[0, 1]$ gera um número entre $[0, 1]$. Ainda, a solução é interpretável, pois se pode ler: “Se X_1 é A_1 e X_2 é A_3 , então Y é B_1 ”, conforme observado no tópico Base de Regras Fuzzy (Anexo A).

A segunda expressão (4-15) não é viável, pois a soma de dois números entre $[0, 1]$ pode fornecer um valor fora deste intervalo. Já a expressão (4-16) é perfeitamente viável, pois a soma de dois valores entre $[0, 0,5]$ é no máximo igual a 1. Além disso, é uma solução plausível de ser analisada de forma verbal: “Se X_1 é A_1 , então Y é B_1 ”, acrescentado de “Se X_2 é A_3 , então Y é B_1 ”. Neste caso, não houve o conectivo lógico “ou” devido à ausência de uma t-conorma, e o meio que possibilitou esta solução ser viável foi o uso do operador de agregação média aritmética.

Como há um número grande de combinações dos termos antecedentes, a necessidade de se gerarem soluções acuradas, viáveis e interpretáveis torna-se mais complexa, conforme o aumento da dimensionalidade do problema. Como existe uma base de dados ou a dinâmica de uma planta, é possível empregar uma meta-heurística para gerar uma base de regras, de maneira a unir as informações presentes em $\mu_{A_{11}}(x_{i1}), \mu_{A_{12}}(x_{i2}), \dots, \mu_{A_{1J}}(x_{iJ})$, com o intuito de produzir uma solução viável e com desempenho apropriado ao problema sobre análise.

Dentro deste propósito o modelo GPFIS combina, a partir de operadores da Lógica Fuzzy, cada $\mu_{A_{1j}}(x_{ij})$ de modo que este seja uma descrição mais aproximada possível dos $\mu_{B_k}(y_i)$. Com este intuito, a etapa de Inferência é subdivida em 3 passos: Formulação, Particionamento e Agregação, cada uma com um determinado propósito no resultado final da modelagem. A Figura 4.7 apresenta o encadeamento do processo desejado.

A partir dos $\mu_{A_{1j}}(x_{ij})$ advindos do processo de Fuzzificação, na etapa de Formulação eles são combinados conforme os operadores fuzzy (t-normas, modificadores linguísticos, etc.) materializados em um indivíduo da população do modelo GPFIS, formando assim uma premissa ou termo antecedente de regra (denotada por $\mu_{A_d}(\mathbf{x}_i)$). O papel da etapa de Particionamento é definir

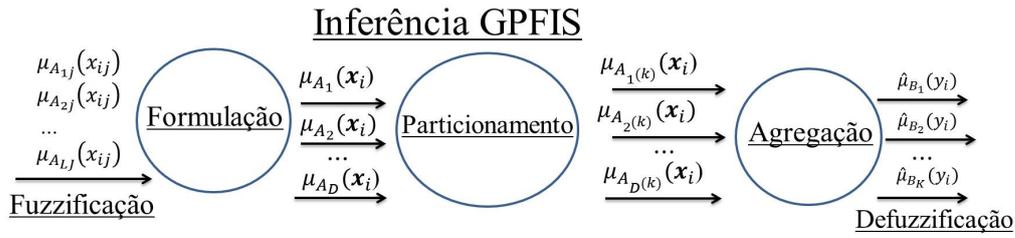


Figura 4.7: Diagrama do processo de Inferência do modelo GPFIS.

o conseqüente que melhor se identifica com o termo antecedente, formando a partir dessa união uma regra linguística (no caso, denotada por $\mu_{A_d(k)}(\mathbf{x}_i)$). Como devem haver um conjunto de diferentes regras destinadas à um mesmo conseqüente, a etapa de Agregação fornece subsídios para a concepção de um grau de ativação final. De forma sequencial, as próximas seções elaboram o procedimento de Inferência do modelo GPFIS.

Formulação

A primeira etapa do processo de Inferência Fuzzy é a formulação das premissas das regras fuzzy. Uma premissa é comumente definida por:

$$“Se X_1 \text{ é } A_{11} \text{ e } \dots \text{ e } X_j \text{ é } A_{1j} \text{ e } \dots \text{ e } X_J \text{ é } A_{1J}”$$

ou, em termos matemáticos:

$$\mu_{A_d}(\mathbf{x}_i) = \mu_{A_{11}}(x_{i1}) * \dots * \mu_{A_{1j}}(x_{ij}) * \dots * \mu_{A_{1J}}(x_{iJ}) \quad (4-17)$$

onde $\mu_{A_d}(\mathbf{x}_i)$ é o grau de pertinência conjunto da i -ésima amostra na d -ésima premissa ($d = 1, \dots, D$), computado a partir de uma t -norma $*$. Então, uma premissa pode ser construída a partir de uma combinação das $\mu_{A_{1j}}(x_{ij})$ com o uso de t -normas, t -conormas, operadores de negação e modificadores linguísticos. Como o espaço das possíveis combinações é grande, uma forma de se buscar as mais promissoras é fazer uso da PGMG, sendo exemplificada na Figura 4.8 uma possível solução fornecida por esta MHG.

Por exemplo, a premissa 1 representa: $\mu_{A_1}(\mathbf{x}_i) = \mu_{A_{21}}(x_{i1}) * \mu_{A_{32}}(x_{i2})$ e, em termos linguísticos, “Se X_1 é A_{21} e X_2 é A_{32} ”. A Tabela 4.2 apresenta as componentes usadas para elaborar as soluções exemplificadas na Figura 4.8.

No âmbito do modelo GPFIS os conjuntos fuzzy representam o conjunto de Terminais, enquanto que o conjunto de Operações Matemáticas é formado pelo de Operadores Fuzzy (Terminais e Operações Matemáticas são terminologias usuais na área de PG). Portanto, no modelo GPFIS, a PGMG é usada

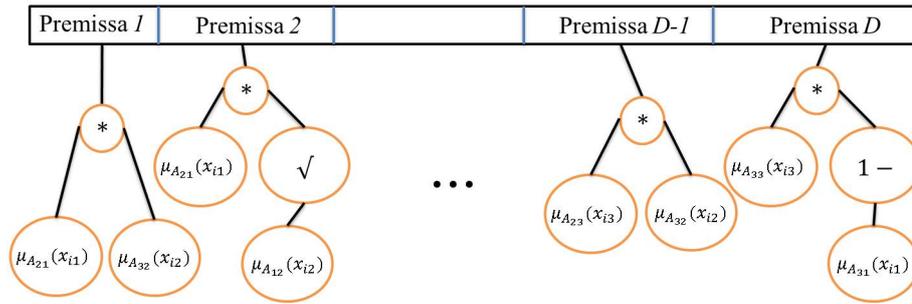


Figura 4.8: Exemplo de premissas formadas por um indivíduo da PGMG.

Tabela 4.2: Conjuntos e Operadores Fuzzy usados para formar as soluções da Figura 4.8.

Conjuntos Fuzzy de Entrada (Terminais)	Operadores Fuzzy (Operações Matemáticas)
$\mu_{A_{11}}(x_{i1}), \dots, \mu_{A_{L1}}(x_{i1}), \dots, \mu_{A_{1j}}(x_{ij}), \dots, \mu_{A_{Lj}}(x_{ij})$	t-norma (*), modificador linguístico ($\sqrt{\quad}$) e negação clássica (1 -)

somente para obter as premissas $\mu_{A_d}(\mathbf{x}_i)$ para a formação da base de regras fuzzy. Para que haja a composição plena de uma base de regras fuzzy, torna-se necessário definir o termo consequente B_k mais adequado para uma dada $\mu_{A_d}(\mathbf{x}_i)$. O próximo tópico - Particionamento - exhibe os métodos elaborados para a definição do termo consequente mais apropriado para um dado termo antecedente.

Particionamento

Esta seção trata das diferentes metodologias, tanto as presentes na literatura, quanto as elaboradas pelo autor, para a escolha do consequente de cada $\mu_{A_d}(\mathbf{x}_i)$. Nem sempre será possível utilizar todos os métodos, pois alguns demandam à existência de uma base de dados. Em todos os descritos são exibidos os recursos necessários para a aplicação, além da formulação e motivação matemática do método.

Divisão Uniforme

Este método de particionamento aplica uma heurística relativamente ingênua e pode ser usado em Previsão/Regressão, Classificação e Controle. O algoritmo de particionamento pode ser construído como:

1. Calcule: $U = \lfloor D/K \rfloor$ ($\lfloor \cdot \rfloor$ é a operação que retorna o menor inteiro). Note que D é o número total de premissas codificadas em um indivíduo e K o número de consequentes possíveis.

2. Particione:

- (a) Consequente 1: $\mu_{A_{1(1)}}(\mathbf{x}_i), \dots, \mu_{A_{U(1)}}(\mathbf{x}_i)$;
- (b) Consequente 2: $\mu_{A_{U+1(2)}}(\mathbf{x}_i), \dots, \mu_{A_{2 \times U(2)}}(\mathbf{x}_i)$;
- (c) ...
- (d) Consequente K: $\mu_{A_{(K-1) \times U+1(K)}}(\mathbf{x}_i), \dots, \mu_{A_{K \times U(K)}}(\mathbf{x}_i)$

A Figura 4.9 exemplifica este método, que em certo grau se assemelha à forma clássica de se buscar a base de regras pela MHG (tanto premissas quanto termos consequentes). Por fim, note-se que se $U = \lfloor D/K \rfloor = 0$, tem-se que o indivíduo é tido como inviável.

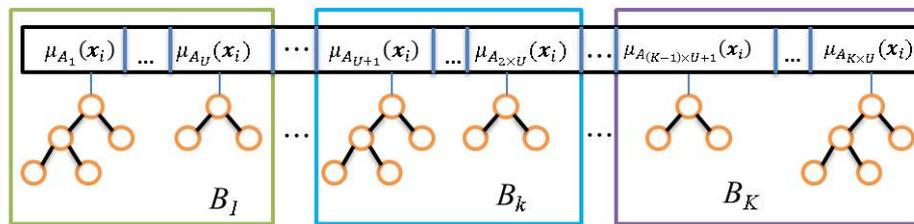


Figura 4.9: Exemplo de Particionamento pelo método da Divisão Uniforme.

Entretanto, uma $\mu_{A_d}(\mathbf{x}_i)$ com elevado suporte na base de dados, ou especializada para um determinado termo consequente pode acabar associada a um outro consequente pouco ou nada compatível. Este fato é uma das consequências negativas de se permitir que a MHG busque todos os elementos para a formação de uma regra. Um caminho possível para se superar essa desvantagem, além de se promover a redução no espaço de busca do problema, é fazer uso de métricas para avaliar a compatibilidade entre $\mu_{A_d}(\mathbf{x}_i)$ e os termos consequentes B_k , utilizando para tanto as informações presentes na base de dados. Contudo, como em Controle comumente não se possui uma base de dados à priori, o procedimento de Divisão Uniforme é o mais usado.

Máxima Confiança (Acurácia Fuzzy)

O método de Divisão Uniforme não explora as informações dispostas na base de dados. Para tanto, considere-se um problema de Classificação ou de Previsão/Regressão, em que a existência de um banco de dados é típica.

Classificação: Suponha-se a existência de $i = 1, \dots, n$ padrões para os $j = 1, \dots, J$ atributos, tal que cada padrão está associado a uma única classe $C_i = k$ ($k = 1, \dots, K$). Considere-se que em cada atributo há $l = 1, \dots, L$

conjuntos fuzzy A_{lj} . Cada um destes elementos são dispostos na matriz de dados, conforme Tabela 4.3.

Tabela 4.3: Exemplo da matriz de dados.

Padrão	X_1	...	X_J	$\mu_{A_{11}}(X_1)$...	$\mu_{A_{L1}}(X_1)$...	$\mu_{A_{LJ}}(X_J)$	C_i
1	x_{11}	...	x_{1J}	$\mu_{A_{11}}(x_{11})$...	$\mu_{A_{J1}}(x_{11})$...	$\mu_{A_{LJ}}(x_{1J})$	1
2	x_{21}	...	x_{2J}	$\mu_{A_{11}}(x_{21})$...	$\mu_{A_{J1}}(x_{21})$...	$\mu_{A_{LJ}}(x_{2J})$	1
.
.
.
i	x_{i1}	...	x_{iJ}	$\mu_{A_{11}}(x_{i1})$...	$\mu_{A_{J1}}(x_{i1})$...	$\mu_{A_{LJ}}(x_{iJ})$	k
.
.
.
n	x_{n1}	...	x_{nJ}	$\mu_{A_{11}}(x_{n1})$...	$\mu_{A_{L1}}(x_{n1})$...	$\mu_{A_{LJ}}(x_{nJ})$	k

Sem perda de generalidade, considere-se uma premissa:

$$\mu_{A_d}(X_1, X_2) = \mu_{A_{21}}(x_{i1})\mu_{A_{22}}(x_{i2})^2 \quad (4-18)$$

além disso, considere-se $C_{i \in k} = 1$ se o i -ésimo padrão pertence à classe k , e 0 caso contrário. Em razão dos resultados e das definições, tem-se a configuração da matriz resultante mostrada na Tabela 4.4.

Tabela 4.4: Exemplo de matriz resultante para Classificação.

Padrão	$\mu_{A_d}(X_1, X_2)$	C_i	$C_{i \in 1}$...	$C_{i \in k}$...	$C_{i \in K}$
1	$\mu_{A_{21}}(x_{11})\mu_{A_{22}}(x_{12})^2$	1	1	...	0	...	0
2	$\mu_{A_{21}}(x_{21})\mu_{A_{22}}(x_{22})^2$	1	1	...	0	...	0
.
.
.
i	$\mu_{A_{21}}(x_{i1})\mu_{A_{22}}(x_{i2})^2$	k	0	...	1	...	0
.
.
.
n	$\mu_{A_{21}}(x_{n1})\mu_{A_{22}}(x_{n2})^2$	K	0	...	0	...	1

O método da Máxima Confiança avalia o grau de compatibilidade da premissa da regra com respeito a todas as K classes. Assim, para cada uma das K classes é computado um Grau de Confiança à classe k (CD_k), dado por:

$$CD_k = \frac{\sum_{i \in k} \mu_{A_d}(\mathbf{x}_i)}{\sum_{i=1}^n \mu_{A_d}(\mathbf{x}_i)} \in [0, 1] \quad (4-19)$$

o CD_k pode ser avaliado como a identificação da parte antecedente aos padrões da classe k (numerador), com relação ao total de compatibilidade da parte

anterior à classe k e às demais (denominador). Logo, $0 \leq CD_k \leq 1$, onde $CD_k = 1$ significa compatibilidade total, e $CD_k = 0$, o contrário.

De fato, o CD_k é amplamente utilizado por diversos autores de SIFs em geral [21, 75, 115, 187], tanto com o intuito de fornecer pesos às regras geradas ou ainda definir o consequente. Há também o Grau de Confiança Penalizado [116] - PCD_k -, dado por:

$$PCD_k = \max\left\{\frac{\sum_{i \in k} \mu_{A_d}(\mathbf{x}_i) - \sum_{i \notin k} \mu_{A_d}(\mathbf{x}_i)}{\sum_{i=1}^n \mu_{A_d}(\mathbf{x}_i)}, 0\right\} \quad (4-20)$$

o objetivo do PCD_k é contabilizar o grau de ativação das demais classes, e descontá-las do CD_k . Assim, estar-se-ia buscando regras que são mais específicas a uma determinada classe e não as mais generalistas.

A definição da classe consequente é dada pelos passos abaixo, tal que, para todo o $\mu_{A_d}(\mathbf{x}_i)$:

1. Calcula-se o CD_k ou PCD_k para todas as K classes.
2. A classe consequente do $\mu_{A_d}(\mathbf{x}_i)$ é determinada pela que obteve maior CD_k ou PCD_k .
3. Caso algum $\mu_{A_d}(\mathbf{x}_i)$ tenha $CD_k = 0$ ou $PCD_k = 0$ para todo k , esta é excluída.

É possível observar que nem toda $\mu_{A_d}(\mathbf{x}_i)$ deverá ter um consequente, e ainda, não é garantido que toda a classe pode ter uma $\mu_{A_d}(\mathbf{x}_i)$ associada. Por fim, há outras implementações disponíveis na literatura sobre o CD_k , principalmente as que levam em conta o desbalanceamento das classes [75, 163].

Regressão e Previsão: De modo similar², considere-se a existência de $i = 1, \dots, n$ amostras (instantes de tempo) compostos por observações de $j = 1, \dots, J$ variáveis de entrada (defasagens de ordem J). A variável de saída é denotada por $y_i \in Y$, possuindo - por simplicidade de notação - $k = 1, \dots, K$ conjuntos fuzzy B_k .

Problemas de Previsão e Regressão são menos abordados na literatura de SFGs do que os de Classificação. Dos trabalhos de SFGs, as principais abordagens são a partir do uso de SIFs do tipo Mamdani [5, 15, 173], e, em menor número, do tipo Takagi-Sugeno-Kang [83]. Entretanto, fornecer pesos para as regras geradas, uma abordagem comum nos SIFs para Classificação,

²Para facilitar a notação, suponha novamente o problema de Previsão semelhante ao de Regressão em termos de notação.

não ocorre para as aplicações em Previsão/Regressão. Credita-se a isso à dificuldade de adaptar uma expressão semelhante a (4-19) para o cálculo do Grau de Confiança. Esta dificuldade surge, principalmente, pelo fato de que em problemas de Regressão/Previsão não se tem algo do tipo $\{0, 1\}$ (pertence ou não pertence), porém um valor entre $[0, 1]$ (grau de pertinência) para o termo consequente. Um detalhe não atentado na bibliografia estudada está na expressão do CD_k . Considere-se a matriz resultante na Tabela 4.4, com a $\mu_{A_d}(X_1, X_2) = \mu_{A_d}(\mathbf{X})$ e os $C_{i \in 1}, \dots, C_{i \in K}$. A expressão (4-19) pode ser reescrita como:

$$CD_k = \frac{\sum_{i \in k} \mu_{A_d}(\mathbf{X})}{\sum_{i=1}^n \mu_{A_d}(\mathbf{X})} = \frac{\langle \mu_{A_d}(\mathbf{X}), C_{i \in k} \rangle}{\|\mu_{A_d}(\mathbf{X})\|_1} \quad (4-21)$$

onde $\langle \cdot, \cdot \rangle$ e $\|\cdot\|_1$ denotam o produto interno usual e a norma do absoluto (manhattan), respectivamente. O resultado apresentado indica que é possível elaborar métricas a partir de $\mu_{A_d}(\mathbf{X})$ e $\mu_{B_k}(Y)$, por meio dos operadores $\|\cdot\|$ e $\langle \cdot, \cdot \rangle$. Portanto, considere-se a mesma $\mu_{A_d}(\mathbf{x}) = \mu_{A_d}(X_1, X_2)$ apresentada na matriz resultante adequada ao problema de Previsão/Regressão, conforme Tabela 4.5.

Tabela 4.5: Exemplo de matriz resultante para Previsão/Regressão.

Amostra	$\mu_{A_d}(X_1, X_2)$	y_i	$\mu_{B_1}(Y)$	$\mu_{B_2}(Y)$...	$\mu_{B_K}(Y)$
1	$\mu_{A_{21}}(x_{11})\mu_{A_{22}}(x_{12})^2$	y_1	$\mu_{B_1}(y_1)$	$\mu_{B_2}(y_1)$..	$\mu_{B_K}(y_1)$
2	$\mu_{A_{21}}(x_{21})\mu_{A_{22}}(x_{22})^2$	y_2	$\mu_{B_1}(y_2)$	$\mu_{B_2}(y_2)$..	$\mu_{B_K}(y_2)$
.
.
.
i	$\mu_{A_{21}}(x_{i1})\mu_{A_{22}}(x_{i2})^2$	y_i	$\mu_{B_1}(y_i)$	$\mu_{B_2}(y_i)$..	$\mu_{B_K}(y_i)$
.
.
.
n	$\mu_{A_{21}}(x_{n1})\mu_{A_{22}}(x_{n2})^2$	y_n	$\mu_{B_1}(y_n)$	$\mu_{B_2}(y_n)$..	$\mu_{B_K}(y_n)$

Tanto a $\mu_{A_d}(X_1, X_2)$, quanto os $\mu_{B_k}(Y)$ podem ser representados como vetores no hiper-cubo $[0, 1]^n$ dimensional. Esta representação é exemplificado na Figura 4.10.

Surgem, a partir dessa representação, algumas possíveis métricas para avaliar o grau de compatibilidade de cada $\mu_{A_d}(\mathbf{x}_i)$ com os respectivos $\mu_{B_j}(y_i)$. A primeira é denominada Grau de Similaridade com o k-ésimo termo consequente (SD_k), dada por:

$$SD_k = (1 - \frac{\|\mu_{A_d}(\mathbf{X}) - \mu_{B_k}(Y)\|_1}{n}) I_{\{0,1\}} \quad (4-22)$$

onde $I_{\{0,1\}} = 0$ caso $\langle \mu_{A_d}(\mathbf{X}), \mu_{B_k}(Y) \rangle = 0$, e 1 caso contrário. A SD_k

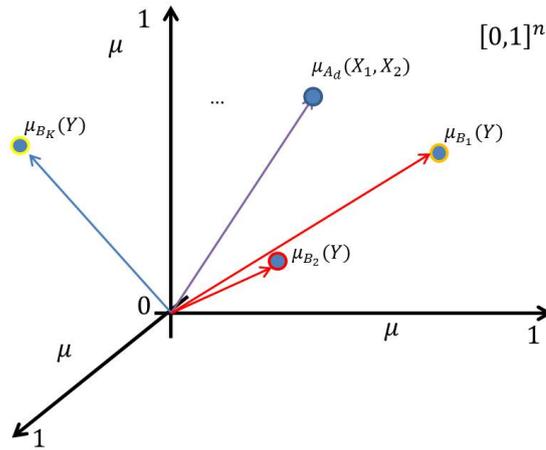


Figura 4.10: Representação exemplificada dos $\mu_{B_k}(Y)$ e $\mu_{A_d}(X_1, X_2)$ no espaço $[0, 1]^n$ dimensional.

mensura a similaridade entre $\mu_{A_d}(\mathbf{X})$ e $\mu_{B_k}(Y)$, a partir da distância absoluta entre os dois vetores. O termo $I_{\{0,1\}}$ leva em conta a possibilidade do grau de ativação de $\mu_{A_d}(\mathbf{x}_i)$ para todo i ser igual a zero, ou seja, uma regra inativa. A $SD_k \in [0, 1]$, tal que $SD_k = 1$ quando os graus de ativação, tanto na parte antecedente, quanto na conseqüente são perfeitamente semelhantes, e $SD_k \rightarrow 0$ quando estes são mais dessincronizados. De um ponto de vista geométrico, um maior SD_k está relacionado ao inverso da distância dos vetores no hipercubo. Logo, apesar de dois vetores estarem com o mesmo sentido, por uma questão de valores estes possuem pouca similaridade (ver $\mu_{A_d}(\mathbf{X})$ e $\mu_{B_k}(Y)$, na Figura 4.10).

Uma solução é avaliar o ângulo formado pelos vetores $\mu_{A_d}(\mathbf{X})$ e o $\mu_{B_k}(Y)$. Tem-se o resultado de Álgebra Linear que $\langle v_1, v_2 \rangle = \cos(\theta) \|v_1\|_2 \|v_2\|_2$, onde θ é o ângulo formado pelos vetores v_1 e v_2 (conforme representado pela Figura 4.10) e $\|\cdot\|_2$ é a norma euclidiana. A partir disso, o Grau de Confiança Fuzzy ao k -ésimo conseqüente (FCD_k) é dado por:

$$FCD_k = \frac{\langle \mu_{A_d}(\mathbf{X}), \mu_{B_k}(Y) \rangle}{\|\mu_{A_d}(\mathbf{X})\|_2 \|\mu_{B_k}(Y)\|_2} \quad (4-23)$$

onde $FCD_k = 1$ caso $\mu_{A_d}(\mathbf{X})$ e $\mu_{B_k}(Y)$ sejam linearmente dependentes, e $FCD_k = 0$ caso sejam ortogonais.

Por fim, caso uma $\mu_{A_d}(\mathbf{X})$ também possua um grau elevado de associação com as demais $\mu_{B_k}(Y)$, torna-se necessário elaborar uma métrica que desconte esse fato, possibilitando a geração de regras mais associadas a um único conseqüente e ortogonais aos demais. O Grau de Confiança Fuzzy Penalizado

para o conseqüente k ($PFCD_k$) é dado por:

$$PFCD_k = \max \left\{ \frac{\langle \mu_{A_d}(\mathbf{X}), \mu_{B_k}(Y) \rangle}{\|\mu_{A_d}(\mathbf{X})\|_2 \|\mu_{B_k}(Y)\|_2} - \sum_{p \neq k} \frac{\langle \mu_{A_d}(\mathbf{X}), \mu_{B_p}(Y) \rangle}{\|\mu_{A_d}(\mathbf{X})\|_2 \|\mu_{B_p}(Y)\|_2}, 0 \right\}, p=1, \dots, K \quad (4-24)$$

o $PFCD_k$ desconta o grau de dependência linear entre $\mu_{A_d}(\mathbf{X})$ e o k -ésimo conseqüente com relação aos demais. Quando $PFCD_k = 1$, tem-se que a regra é perfeitamente associada ao conseqüente e ortogonal aos demais, e $PFCD_k \rightarrow 0$ caso contrário.

A definição do conseqüente B_k das $\mu_{A_d}(\mathbf{x}_i)$ é dado pelos passos, para todo $d = 1, \dots, D$:

1. Seleciona-se uma das métricas: SD_k , FCD_k ou $PFCD_k$.
2. Calcula-se a métrica escolhida para todos os K conseqüentes.
3. O k -ésimo conseqüente para $\mu_{A_d}(\mathbf{x}_i)$ é definido pelo que maximizar a métrica selecionada.
4. Caso $\mu_{A_d}(\mathbf{x}_i)$ tenha $SD_k = 0$, $FCD_k = 0$ ou $PFCD_k = 0$ para todo k , essa é descartada.

Como no caso anterior, nem toda $\mu_{A_d}(\mathbf{x}_i)$ estará associada a um determinado conseqüente, como também pode haver algum conseqüente que esteja inativo.

Máxima Credibilidade

O objetivo deste método é estimar a “credibilidade” de cada $\mu_{A_d}(\mathbf{x}_i)$ para cada conseqüente, induzida a partir dos pesos estimados por Mínimos Quadrados Restritos. Para tanto, deve-se resolver o seguinte problema de minimização restrita:

$$\min : \|\mathcal{U}(\mu_{A_d})\beta - \Omega(\mu_B)\|_2^2 \quad (4-25)$$

$$s.j : \beta \mathbf{1} = \mathbf{1} \quad (4-26)$$

$$\beta \geq 0 \quad (4-27)$$

onde $\mathcal{U}(\mu_{A_d}) = [\mu_{A_1}(\mathbf{X}) \ \mu_{A_2}(\mathbf{X}) \ \dots \ \mu_{A_D}(\mathbf{X})]$ é a matriz cujas colunas são as $\mu_{A_d}(\mathbf{X})$. O $\Omega(\mu_B) = [\mu_{B_1}(Y) \ \mu_{B_2}(Y) \ \dots \ \mu_{B_K}(Y)]$ é a matriz, com colunas

que perfazem o grau de pertinência de todo y_i a cada k -ésimo conjunto fuzzy. Também, $\mathbf{1} = [1 \ 1 \ \dots \ 1]^T$ e $\beta = [b_{d1} \ b_{d2} \ \dots \ b_{dK}]$ é a matriz de pesos com D linhas e K colunas, positivos, tal que a soma dos pesos de cada $\mu_{A_d}(\mathbf{X})$ associado a um mesmo conseqüente é igual a 1. Por fim, $\|\cdot\|_2$ é a norma de Frobenius.

Para calcular a matriz β é necessário resolver um problema de minimização com restrições. Este problema é convexo, existe ao menos uma solução viável, além de possuir restrições lineares que denota a existência de solução única [119]. Porém, como o problema possui restrições de desigualdade, uma expressão analítica é inviável, sendo necessário recorrer a métodos iterativos. O Anexo B, seção B.2, trata da solução por Multiplicadores de Lagrange e o método das Variáveis Ativas.

A interpretação de cada $b_{dk} \in \beta$ é o peso (grau de credibilidade) de cada $\mu_{A_d}(\mathbf{X})$ em explicar o comportamento do k -ésimo conseqüente. De um ponto de vista geométrico, o b_{dk} representa o valor necessário para a melhor projeção de $\mu_{A_d}(\mathbf{X})$ no subespaço gerado por cada $\mu_{B_k}(Y)$, dada as restrições. Para valores próximo de zero, isto denota que não é bem estabelecida uma combinação linear restrita que aproxime $\mu_{A_d}(\mathbf{X})$ do k -ésimo conseqüente (note que todos os vetores residem na mesma escala $[0,1]$). A Figura 4.11 apresenta este detalhe de um ponto de vista geométrico.

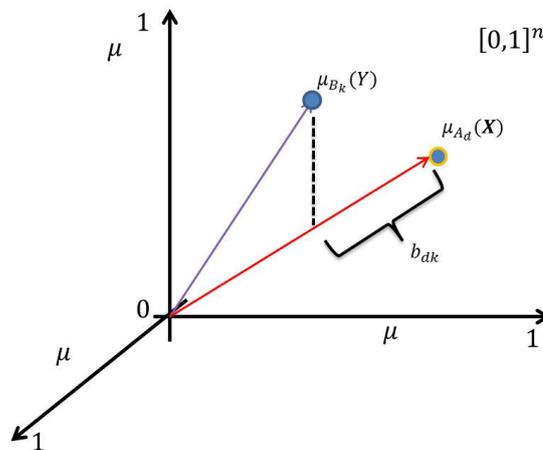


Figura 4.11: Ponto de vista geométrico do método da Máxima Credibilidade.

A partir disso, o método da Máxima Credibilidade particiona toda $\mu_{A_d}(\mathbf{X})$ para cada k -ésimo conseqüente:

1. Computar o método da Máxima Credibilidade.
2. Buscar o b_{dk} associado à $\mu_{A_d}(\mathbf{X})$ com maior valor entre os K possíveis conseqüentes.

3. Definir $\mu_{A_d}(\mathbf{X})$ ao conseqüente B_k . Remover a $\mu_{A_d}(\mathbf{X})$ do conjunto de possibilidades.

No caso de empates, uma das funções pode ser escolhida de modo arbitrário. Há também a possibilidade de não se remover a $\mu_{A_d}(\mathbf{X})$ após a seleção do máximo b_{dk} . Assim, é possível ter-se regras com mais de um conseqüente. Esta possibilidade é mais bem descrita na próxima seção.

Agregação

Este tópico trata da etapa final do processo de Inferência: a operação de Agregação das ativações de cada regra. A Figura 4.12 ilustra a cadeia de processos até então desenvolvida.

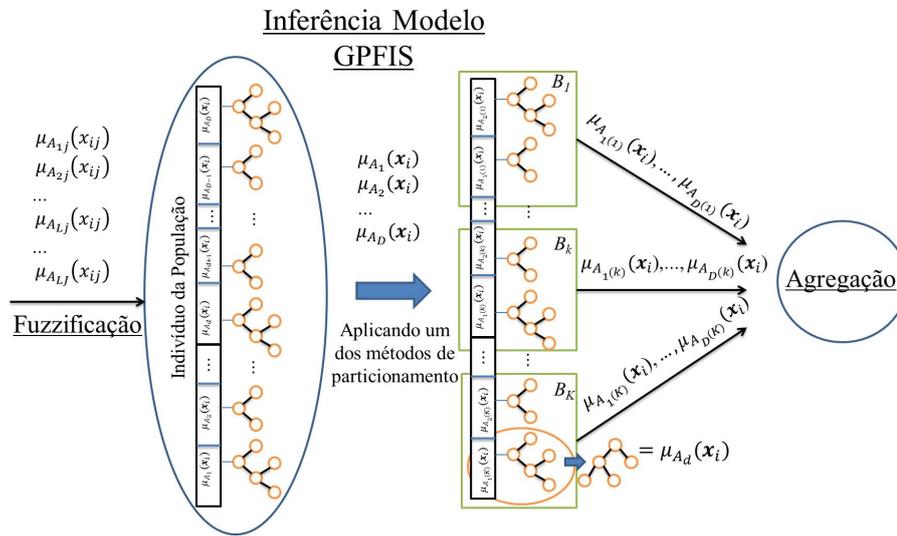


Figura 4.12: Diagrama do processo de Inferência: Formulação-Particionamento-Agregação.

Com a definição da base de regras - $\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i), \forall k = 1, \dots, K$ - dado uma nova amostra \mathbf{x}_i^* , ocasionalmente haverá a compatibilidade dessa com diversas regras. A etapa de Agregação visa a reunir os graus de pertinência de \mathbf{x}_i^* às regras de um mesmo termo conseqüente, de modo a gerar um valor consensual.

A partir dessa diretriz, considere-se $D^{(k)}$ o número de regras associadas ao k-ésimo termo conseqüente ($d^{(k)} = 1^{(k)}, 2^{(k)}, \dots, D^{(k)}$). Dado um operador de agregação $g : [0, 1]^{D^{(k)}} \rightarrow [0, 1]$ [25], o grau de pertinência estimado de y_i ao

k-ésimo termo consequente ($\hat{\mu}_{B_k}(y_i)$) é:

$$\hat{\mu}_{B_1}(y_i) = g[\mu_{A_{1(1)}}(\mathbf{x}_i), \dots, \mu_{A_{D(1)}}(\mathbf{x}_i)] \quad (4-28)$$

$$\hat{\mu}_{B_2}(y_i) = g[\mu_{A_{1(2)}}(\mathbf{x}_i), \dots, \mu_{A_{D(2)}}(\mathbf{x}_i)] \quad (4-29)$$

...

$$\hat{\mu}_{B_K}(y_i) = g[\mu_{A_{1(K)}}(\mathbf{x}_i), \dots, \mu_{A_{D(K)}}(\mathbf{x}_i)] \quad (4-30)$$

decidir a forma funcional da $g[\cdot]$ (operador de Agregação), encontrada nas expressões do modelo GPFIS depende da aplicação em questão. Alguns operadores de agregação são generalistas, enquanto outros somente funcionam caso haja uma base de dados. Além disso, cada um destes operadores altera a forma de interpretação e a qualidade dos resultados. A seguir são apresentados alguns possíveis operadores de agregação geralmente usados no modelo GPFIS. Considere-se como exemplo geral as duas regras em formato simbólico:

$$\mu_{A_1^{(1)}}(\mathbf{x}_i) = \mu_{A_{11}}(x_{i1})\mu_{A_{12}}(x_{i2}) \quad (4-31)$$

$$\mu_{A_2^{(1)}}(\mathbf{x}_i) = \mu_{A_{31}}(x_{i1})\mu_{A_{12}}(x_{i2}) \quad (4-32)$$

que expressam em termos linguísticos:

“Se X_1 é A_1 e X_2 é A_1 , então Y é B_1 .”

“Se X_1 é A_3 e X_2 é A_1 , então Y é B_1 .”

- $g[\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i)] \rightarrow \max\{\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i)\}$: o operador de agregação máximo é um dos mais empregados em Sistemas Fuzzy do tipo Mamdani. Este operador é bastante genérico, tendo sua aplicação em diferentes ramos (Classificação, Controle e Previsão/Regressão). Como cada uma das regras está atrelada a um k-ésimo consequente, tem-se que $\hat{\mu}_{B_k}(y_i) = \max\{\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i)\}$. Como o resultado final é agregado por uma t-conorma, o conectivo lógico “ou” se manifesta para compor o resultado final. A partir do exemplo acima, tem-se:

$$\hat{\mu}_{B_1}(y_i) = \max\{\mu_{A_{11}}(x_{i1})\mu_{A_{12}}(x_{i2}), \mu_{A_{31}}(x_{i1})\mu_{A_{12}}(x_{i2})\} \quad (4-33)$$

em termos linguísticos:

“Se X_1 é A_1 e X_2 é A_1 , então Y é B_1 .”,

ou

“Se X_1 é A_3 e X_2 é A_1 , então Y é B_1 .”

- $g[\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i)] \rightarrow \frac{1}{D^{(k)}} \sum_{d^{(k)}=1}^{D^{(k)}} \mu_{A_{d^{(k)}}}(\mathbf{x}_i)$: a operação de média aritmética visa a dar pesos iguais a todas as regras que compõe o

consequente, logo, não discrimina qual das regras é mais compatível com o padrão apresentado (operador máximo). Também, esta pode ser usada em todas as aplicações do modelo GPFIS. Esta forma de agregação induz uma outra interpretação, como por exemplo:

$$\hat{\mu}_{B_1}(y_i) = \frac{1}{2}\mu_{A_{11}}(x_{i1})\mu_{A_{12}}(x_{i2}) + \frac{1}{2}\mu_{A_{31}}(x_{i1})\mu_{A_{12}}(x_{i2}) \quad (4-34)$$

em termos linguísticos:

“Se X_1 é A_1 e X_2 é A_1 , então Y é B_1 com peso igual a 0.5”,

acrescentado de

“Se X_1 é A_3 e X_2 é A_1 , então Y é B_1 com peso igual a 0.5.”

– $g[\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i)] \rightarrow \prod_{d(k)=1}^{D(k)} \mu_{A_{d(k)}}(\mathbf{x}_i)$: o operador de produto é uma t-norma e atende às propriedades de um operador de agregação. Como os demais, seu uso é irrestrito para o modelo GPFIS. Esta forma de agregação induz uma interpretação semelhante a estrutura do $\max\{\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i)\}$, porém com a substituição do conectivo “ou” pelo “e”.

– $g[\mu_{A_{1(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D(k)}}(\mathbf{x}_i)] \rightarrow \sum_{d(k)=1}^{D(k)} w_{d(k)} \mu_{A_{d(k)}}(\mathbf{x}_i)$, $\sum_{d(k)=1}^{D(k)} w_{d(k)} = 1$, $w_{d(k)} \geq 0$: a operação de combinação convexa generaliza a média aritmética na medida que os pesos $w_{d(k)}$ podem ser quaisquer valores entre [0,1], com a restrição que somem 1. Semelhantemente, a interpretação se altera, tal que $w_{d(k)}$ indica o peso, ou o grau de influência dessa regra no resultado final. Supondo $w_{1(1)} = 0,25$ e $w_{2(1)} = 0,75$, a expressão de agregação é dada por:

$$\hat{\mu}_{B_1}(y_i) = \frac{1}{4}\mu_{A_{11}}(x_{i1})\mu_{A_{12}}(x_{i2}) + \frac{3}{4}\mu_{A_{31}}(x_{i1})\mu_{A_{12}}(x_{i2}) \quad (4-35)$$

em termos linguísticos:

“Se X_1 é A_1 e X_2 é A_1 , então Y é B_1 com peso igual a 0,25”,

acrescentado de

“Se X_1 é A_3 e X_2 é A_1 , então Y é B_1 com peso igual a 0,75.”

Nesse caso, a segunda regra influência três vezes mais do que a primeira no resultado final. Apesar de haver diferentes premissas $\mu_{A_d}(\mathbf{x}_i)$ que sejam bem associadas ao consequente alvo, este método selecionaria as mais preponderantes; para as menos associadas o $w_{d(k)}$ seria próximo de 0, ou seja, regras inativas que poderiam ser excluídas da base de regras.

Cabe ressaltar que este método se aplica a problemas com $i = 1, \dots, n$ amostras. Logo, problemas de Controle tipicamente não se adequam a

este tipo de agregação. O cômputo de cada $w_{d^{(k)}}$ é baseado na minimização da função custo: $\sum_{i=1}^n (\mu_{B_k}(y_i) - \sum_{d^{(k)}=1}^{D^{(k)}} w_{d^{(k)}} \mu_{A_{d^{(k)}}}(\mathbf{x}_i))^2$, sujeita às restrições supra citadas. A solução encontrada para esse problema é semelhante à apresentada no tópico Particionamento, subtópico Máxima Credibilidade, cuja solução é mais simples. A busca pelos $w_{d^{(k)}}$ deve ser realizada para os J consequentes, portanto, devem ser resolvidos K problemas de minimização com restrições. O anexo, B seção B.1, trata da solução deste problema.

Por fim, note-se que é possível usar a matriz de pesos computadas no método de Particionamento Máxima Credibilidade como operador de agregação. Considere-se o exemplo anterior, mas com $\mu_{A_1}(\mathbf{x}_i) = \mu_{A_{11}}(x_{i1})\mu_{A_{12}}(x_{i2})$ e $\mu_{A_2}(\mathbf{x}_i) = \mu_{A_{31}}(x_{i1})\mu_{A_{12}}(x_{i2})$ não particionadas, e supondo que o problema tenha no total dois possíveis consequentes B_1 e B_2 . A matriz de pesos β será dada por:

$$\beta = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} 0,25 & 0,75 \\ 0,90 & 0,10 \end{bmatrix}$$

Usando essa informação como agregação, tem-se:

$$\hat{\mu}_{B_1}(y_i) = \frac{1}{4}\mu_{A_{11}}(x_{i1})\mu_{A_{12}}(x_{i2}) + \frac{3}{4}\mu_{A_{31}}(x_{i1})\mu_{A_{12}}(x_{i2}) \quad (4-36)$$

$$\hat{\mu}_{B_2}(y_i) = \frac{9}{10}\mu_{A_{11}}(x_{i1})\mu_{A_{12}}(x_{i2}) + \frac{1}{10}\mu_{A_{31}}(x_{i1})\mu_{A_{12}}(x_{i2}) \quad (4-37)$$

o que resulta em termos linguísticos:

“Se X_1 é A_1 e X_2 é A_1 ,
então Y é B_1 com peso igual a 0,25 e B_2 com peso igual a 0,90”,
acrescentado de
“Se X_1 é A_3 e X_2 é A_1 ,
então Y é B_1 com peso igual a 0,75 e B_2 com peso igual a 0,10.”

Neste caso, uma mesma regra possui diferentes consequentes. Em geral, esta abordagem é pouco usada, devido à dificuldade de interpretação dos resultados, contudo constitui-se em mais um grau de liberdade para o usuário do modelo GPFIS.

4.2.3

Defuzzificação e Decisão

As fases até agora abordadas tratam do mapeamento dos valores precisos a conjuntos fuzzy de uma variável (Fuzzificação) e as etapas conjuntas de Formulação-Particionamento-Agregação (Inferência). Após ambos os procedimentos tem-se a necessidade de tornar os valores novamente precisos, empregando para isso técnicas de Defuzzificação/Decisão. Nesse ponto, deve-se realizar uma distinção entre o tipo de problema em questão. Métodos de Defuzzificação são abordados para os problemas de Previsão/Regressão e Controle, enquanto os de Decisão são para Classificação. As próximas seções abordam as técnicas. Em todos os casos, considera-se que $\hat{\mu}_{B_k}(y_i)$ e $\hat{\mu}_{C_i \in k}(\mathbf{x}_i)$ (grau de pertinência estimado de \mathbf{x}_i a classe k) já esteja calculado para todo k -ésimo termo consequente ou k -ésima classe.

Métodos de Defuzzificação

A literatura de SIF apresentam diversas heurísticas para Defuzzificação [130, 179], sendo todas estas passíveis de implementação no modelo GPFIS. Contudo, devido ao uso generalizado dos conjuntos fuzzy normalizados, uniformemente distribuídos e 2-justapostos, a forma mais empregada no modelo GPFIS é dada pela normalização da expressão (4-13):

$$\hat{y}_i = \frac{c_1 \hat{\mu}_{B_1}(y_i) + \dots + c_K \hat{\mu}_{B_K}(y_i)}{\hat{\mu}_{B_1}(y_i) + \dots + \hat{\mu}_{B_K}(y_i)} \quad (4-38)$$

onde cada c_k é o valor tal que $\mu_{B_k}(c_k) = 1, \forall k$, que em outros termos, é a pré-imagem do vértice superior de cada triângulo. Esta combinação linear é conhecida na literatura de SIF como Método da Altura. A sua grande vantagem é a celeridade computacional com relação ao método do Centro de Gravidade, que demanda o cômputo de integrações numéricas. Este método é bastante usado em problemas do tipo Previsão/Regressão e também em Controle. Como, em alguns problemas de Controle, a atuação se realiza dentro de um conjunto restrito de valores, nem sempre é possível empregar a equação (4-13). Para tanto, sugere-se o método de Altura Média dos Máximos, dado por:

$$\hat{y}_i = \frac{\phi_1 c_1 \hat{\mu}_{B_1}(y_i) + \dots + \phi_K c_K \hat{\mu}_{B_K}(y_i)}{\phi_1 \hat{\mu}_{B_1}(y_i) + \dots + \phi_K \hat{\mu}_{B_K}(y_i)} \quad (4-39)$$

onde ϕ_k é uma função indicadora, tal que $\phi_k = 1$, caso $\hat{\mu}_{B_k}(y_i) \geq \hat{\mu}_{B_z}(y_i), \forall j \neq z$ e $\phi_k = 0$ caso contrário. Para exemplificar a diferença entre os métodos, seja a Figura 4.13.

Foram obtidos como resultado: $\mu_{NM}(y_i) = 0,8$, $\mu_{PZ}(y_i) = 0,6$, enquanto

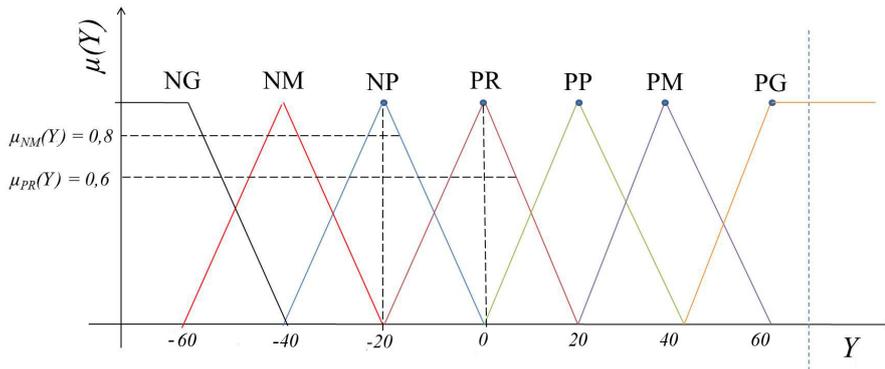


Figura 4.13: Exemplificação dos métodos de Defuzzificação.

que $\mu_{NG}(y_i) = \mu_{NP}(y_i) = \mu_{PP}(y_i) = \mu_{PM}(y_i) = \mu_{PG}(y_i) = 0$. Então, de acordo com 4-13:

$$\hat{y}_i = (-20 \cdot 0,8 + 0 \cdot 0,6) / (0,8 + 0,6) = -11,43 \quad (4-40)$$

pelo método da Altura Média dos Máximos, a resposta seria:

$$\hat{y}_i = (-20 \cdot 0,81 + 0 \cdot 0,60) / (0,81 + 0,60) = -20 \quad (4-41)$$

neste caso, a atuação seria mais severa com o método de Altura Média dos Máximos do que com o da Altura.

Métodos de Decisão

Em linhas gerais, há dois tipos de formulação para problemas de Classificação: one-against-all (uma classe em relação a todas, um contra todos, etc.) e one-against-one (uma classe em relação a outra, pairwise learning, etc.) [84]. Cada uma destas metodologias invoca formas de modelagem distintas, principalmente na parte de Formulação do modelo, e, portanto, afetam o processo de Decisão da respectiva classe do p-ésimo padrão.

O modelo GPFIS foi formulado no tópico Inferência a partir do princípio do one-against-all. Isto se deve ao menor custo computacional (necessário buscar K mapeamentos, onde K é o número de classes) e com o intuito de demonstrar que o problema de Classificação possui similaridades com de Previsão/Regressão, e por isso pode partilhar alguns dos seus princípios. Para a formulação explorada, a Decisão pelo pertencimento do i-ésimo padrão à classe k é dada por:

$$\hat{C}_i = \arg_k \max \{ \hat{\mu}_{C_i \in 1}(\mathbf{x}_i), \dots, \hat{\mu}_{C_i \in K}(\mathbf{x}_i) \} \quad (4-42)$$

onde \hat{C}_i é a classe estimada, resultado do k-ésimo argumento que assume o valor

máximo na expressão (4-42). A ideia é indicar que \mathbf{x}_i pertence à classe com a qual é mais compatível, segundo as regras disponíveis. Quando há empate, uma heurística decisória pode ser aplicada. Quando há empate, uma heurística decisória pode ser aplicada (a classe que possui maior proporção), ou nenhuma classe específica é atribuída a \mathbf{x}_i . Para os problemas do tipo one-against-one, pode-se aplicar o critério do máximo, conforme a expressão (4-42), ou os diferentes métodos de votação apresentados em [79].

4.2.4 Avaliação e Recombinação

Esta seção finaliza as etapas de construção do modelo GPFIS. Em uma visão macro, a Figura 4.14 apresenta o trajeto percorrido nas duas últimas seções e o papel da seção atual.

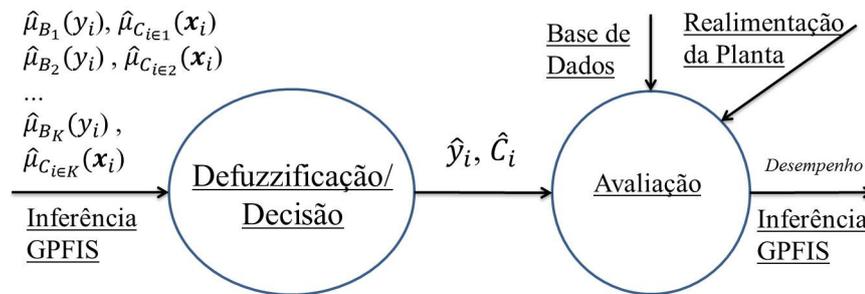


Figura 4.14: Diagrama do Processo: Inferência-Defuzzificação/Decisão-Avaliação.

Como observado na Figura 4.14, a etapa de Avaliação também deve ser desdobrada para os diferentes segmentos de aplicação do modelo GPFIS. Isto é discutido com maior ênfase no tópico Função Custo. Elaborada a função objetivo, há então a possibilidade de ranquear os indivíduos da população e aplicar os operadores de recombinação similares aos da PGMG.

Função Custo

De forma sucinta, a avaliação no modelo GPFIS é definida por um objetivo primário:

- Minimização do Erro - Classificação e Previsão/Regressão.
- Redução do Erro de Trajetória ou Busca do Tempo Mínimo de Parada - Controle.

E um objetivo secundário:

- Redução da Complexidade do Indivíduo - Classificação, Previsão/Regressão e Controle.

O objetivo primário domina a forma de posicionamento dos indivíduos da população, enquanto que o segundo se manifesta como critério de desempate.

Como nos problemas de Classificação e Previsão/Regressão o valor da variável de saída (y_i ou C_i) é conhecido, torna-se fácil avaliar o indivíduo do modelo GPFIS. A função de avaliação para problemas de classificação, sejam binários ou de múltiplas classes, é dada pelo Erro Médio de Classificação (EMC) como:

$$EMC = \frac{\sum_{i=1}^n |C_i(\mathbf{x}_i) - \hat{C}_i(\mathbf{x}_i)|}{n} \quad (4-43)$$

onde para dado um padrão \mathbf{x}_i , $|C_i(\mathbf{x}_i) - \hat{C}_i(\mathbf{x}_i)| = 0$, se $C_i(\mathbf{x}_i) = \hat{C}_i(\mathbf{x}_i)$ e 1, caso contrário. O indivíduo que minimizar o EMC é considerado o melhor na população.

Para os problemas de Previsão/Regressão, um tipo de função objetivo usada é o Desvio Médio Absoluto (DMA):

$$DMA = \frac{\|Y - \hat{Y}\|_1}{n} = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} \quad (4-44)$$

o melhor indivíduo é o que minimizar o DMA. Um outro detalhe que surge, usando a expressão do DMA (4-44), adicionado a (4-7) e (4-13), considerando cada $\hat{\mu}_{B_k}(y_i)$ já normalizado é:

$$DMA = \frac{\|Y - \hat{Y}\|_1}{n} \leq \frac{\sum_{k=1}^K |c_k| \|\varepsilon_k\|_1}{n} \quad (4-45)$$

onde $\varepsilon_j = \mu_{B_k}(Y) - \hat{\mu}_{B_k}(Y)$, utilizado na etapa de Formulação do modelo GPFIS (seção 4.2.2). Os c_k são os pontos tais que $\mu_{B_k}(c_k) = 1$. A demonstração completa da desigualdade é encontrada no Anexo II seção B.3.

Como $|c_k|$ é fixo, caso $\sum_{k=1}^K \|\varepsilon_k\|_1 \rightarrow 0$, então $\hat{\mu}_{B_k}(y_i) \rightarrow \mu_{B_k}(y_i)$ e $\hat{y}_i \rightarrow y_i$. Portanto, é válida a busca por uma base de regras representativa e que forme bons $\hat{\mu}_{B_k}(y_i)$, pois assim as estimativas geradas estarão cada vez mais próximas de y_i .

Para o problema de Controle, a definição da função objetivo fica sujeita ao problema em questão. Em rastreamento de trajetórias, uma possível função de avaliação é dada por:

$$EQM = \frac{\|\mathbf{x}_t\|_2}{J} \quad (4-46)$$

onde EQM é o Erro Quadrático Médio, \mathbf{x}_t é o vetor de estados \mathbf{z}_t subtraído do vetor de referência \mathbf{ref}_t no instante t . Ao minimizar o EQM , o controlador

GPFIS obtém sucesso em acertar a trajetória. Em problemas de tempo mínimo, onde o controlador deve ser elaborado visando a atingir a referência no menor tempo possível, a função objetivo do controlador GPFIS pode ser dado pelo valor de t necessário para atingir a $\|\mathbf{x}_t\|_2 \leq \epsilon$, onde ϵ é uma tolerância fixada pelo operador. Quanto menor o t , melhor o indivíduo na população de controladores GPFIS.

Por fim, o segundo objetivo principal de todas as abordagens supracitadas é a redução da complexidade. Esta é baseada no método de Pressão Lexicográfica Parcimoniosa [143]. A ideia básica do método de desempate é: dado dois indivíduos com desempenhos idênticos, o melhor entre eles é o que possui menor número de nós na árvore. Menor quantidade de nós indica regras com menos antecedentes, com menos operadores de concentração/dilatação, negação e indivíduos com menos $\mu_{A_d}(\mathbf{x}_i)$ e, portanto, com uma menor base de regras fuzzy.

A minimização da função custo é o objetivo mais preponderante. Porém, a redução do tamanho do indivíduo (medido em número de nós da árvore) é essencial para a busca de bases de regras mais compactas, definidas por um menor número de antecedentes e de regras.

Recombinação

Após toda a população ter sido avaliada, independentemente do problema, o passo seguinte é escolher, via um procedimento de seleção (roleta, torneio, etc.), os indivíduos mais aptos da população. A partir desse subconjunto da população, aplicam-se os operadores de recombinação comuns da PGMG. Conforme descrito no Capítulo 3, há três operadores básicos: mutação e cruzamento de baixo e de alto nível. De forma sucinta, considere-se os dois indivíduos presentes na Figura 4.15.

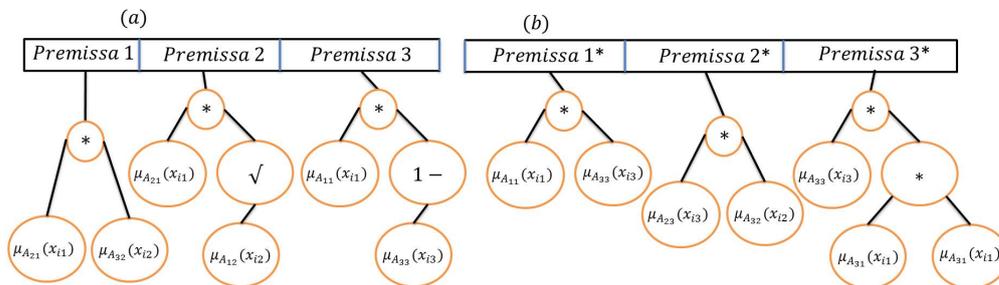


Figura 4.15: Exemplo de duas soluções presentes na rotina do modelo GPFIS.

Suponha-se que estes dois indivíduos foram selecionados (método do torneio, por exemplo) e deverão gerar soluções adaptadas para a próxima

população. Considere-se a aplicação dos operadores de cruzamento de baixo nível e mutação presentes nas Figuras 4.16 e 4.17, respectivamente.

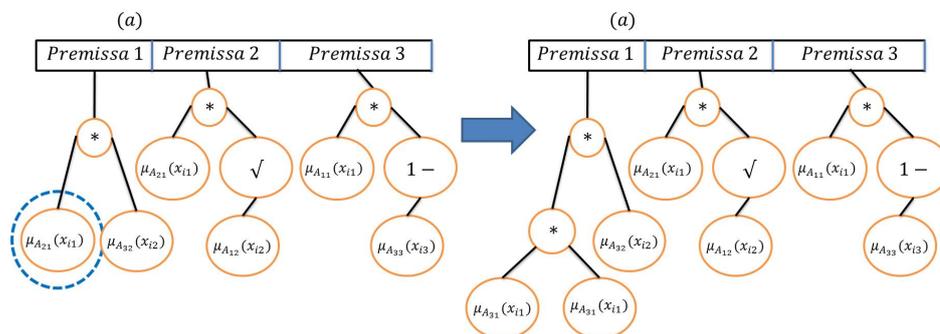


Figura 4.16: Exemplo de aplicação do operador de mutação.

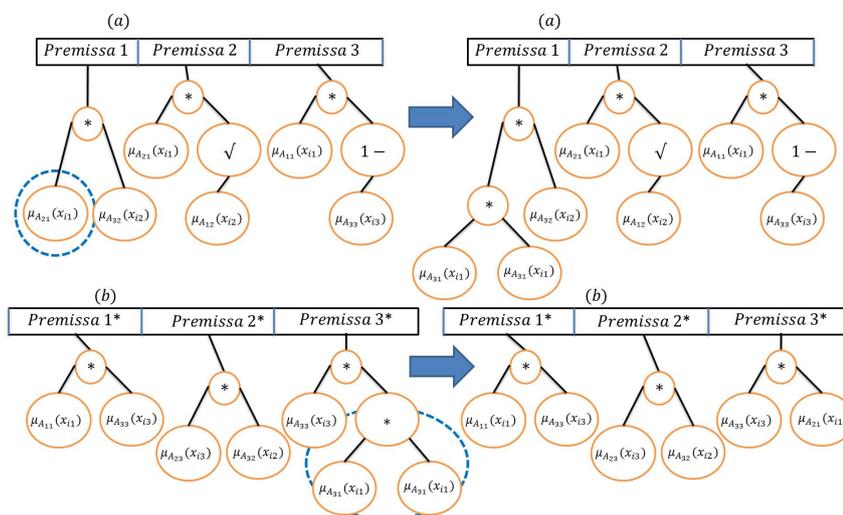


Figura 4.17: Exemplo de aplicação do operador de cruzamento de baixo nível.

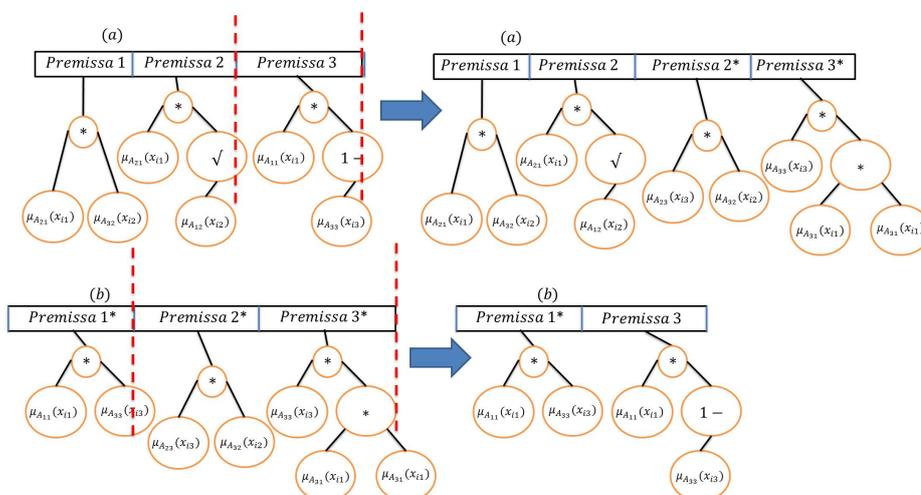


Figura 4.18: Exemplo de aplicação do operador de cruzamento de alto nível.

Ainda, suponha-se uma aplicação do operador de cruzamento de alto nível, conforme a Figura 4.18. É notável que as operações de mutação e cruzamento de baixo nível são aplicadas no nível das regras, ou seja, estas alteram a estrutura lexical dos antecedentes, seja ao ampliar/reduzir seu número ou adicionar/remover operadores de dilatação/contração e negação. O operador de cruzamento de alto nível atinge ao nível da base de regras fuzzy, com o intuito de expandir/retrair a base de regras decodificada no indivíduo. Após essas operações, todo o processo é repetido até atender o critério de parada definido pelo usuário.

4.3 Resumo

Neste capítulo apresentou-se o GPFIS, um modelo Fuzzy-Genético genérico e aplicável a distintos problemas. Na sua elaboração foram usados conceitos presentes na literatura de Programação Genética, Sistemas Fuzzy, Operadores e Otimização. A união deles proporcionou diversos ganhos, tanto teóricos, quanto práticos. O próximo capítulo apresenta diferentes testes do modelo GPFIS, efetuados a partir de experimentos com benchmarks presentes na literatura, assim como algumas aplicações reais do modelo. Os resultados são comparados com os obtidos por outros Sistemas Fuzzy-Genéticos presentes na literatura e também por outros modelos.

5

Estudos de Casos

Este Capítulo apresenta os estudos de casos realizados para o modelo GPFIS. O intuito principal reside em exibir a amplitude e qualidade das soluções fornecidas nas diferentes esferas de aplicação: Regressão, Previsão, Classificação e Controle. A seguir cada seção se refere a um destes problemas, tal que em todas é adotada o seguinte formato:

- **Primeiro tópico - Investigação empírica:** objetiva identificar a configuração mais competitiva para o modelo GPFIS, visando à futura comparação com demais modelos (SFGs ou outros) no tema tratado. Para tanto, apresenta-se o benchmark, a sua justificativa (número de bases de dados, preferência pela presença de SFGs, etc.) e o procedimento experimental realizado. Após, exibe-se a sucessão de análises que constroem a melhor configuração do modelo GPFIS, dado o benchmark escolhido.
- **Segundo tópico - Comparação com outros modelos:** em geral, os benchmarks escolhidos são atrelados a resultados de outros SFGs ou modelos presentes na literatura. Então, a partir da melhor configuração do modelo GPFIS elaborada no primeiro tópico, comparam-se estes resultados com os dos demais métodos, tanto do ponto de vista de acurácia/eficácia, quanto de complexidade do sistema (base de regras, etc.). Com isto é possível avaliar o modelo GPFIS frente ao estado da arte e modelos concorrentes nas outras áreas.
- **Terceiro tópico - Aplicação detalhada:** Faz uso de um exemplo ilustrativo com o propósito de apresentar, de forma detalhada, os resultados obtidos pelo modelo. Assim, espera-se que o leitor ganhe intuição sobre a elaboração e uso do método, além de apresentar as suas potencialidades.

Cabe ressaltar que o número de avaliações e a escolha da função de avaliação foram os critérios usados para tornar as comparações comensuráveis. Além disso, análises estatísticas são efetuadas sempre que possível, com o objetivo de verificar qual método ou configuração obteve resultados significativamente superiores, para um dado conjunto de teste. Esta abordagem tem

se consolidado na literatura de SFGs [65, 66, 92]. O anexo C fornece ao leitor os principais conceitos e métodos estatísticos usados ao longo do texto e relevantes para o pleno entendimento e discussão dos resultados.

Todas as rotinas executadas foram implementadas em MATLAB R2010a [149], executadas em um PC Windows 7 com processador Intel i5, 8GB de RAM. Foi usado como parte da implementação do modelo GPFIS a biblioteca GPTIPS 1.0 de PGMG [190]. Além disso, utilizam-se as rotinas estatísticas foram usados o KEEL [3] e o pacote estatístico R [174]. Por fim, as bases de dados usados nas aplicações benchmarks foram em sua maioria obtidos do University of California Irvine (UCI) [20] e do repositório KEEL [3].

5.1

Regressão

5.1.1

Investigação Empírica da Arquitetura do Modelo GPFIS

Descrição do Problema

Em termos de aplicação benchmark disponíveis em SFGs desenhados para Regressão, encontram-se três principais trabalhos recentemente: Pulkkinen e Koivisto [173], Alcalá et al. [5] e Gacto et al. [83]. A seguir são descritos os pontos principais destes trabalhos.

- **Pulkkinen e Koivisto [173]**: propõe um SFG multiobjetivo, de modo a definir o número de parâmetros das funções de pertinência de cada variável, além do aprendizado do conjunto de regras fuzzy. O modelo é baseado em um SIF do tipo Mamdani, com t-norma produto e defuzzificação do tipo centro de gravidade. Como o SFG gerado busca definir também os parâmetros das funções de pertinência, restrições e codificações especiais são impostas para a viabilidade do sistema. Além disso, o sistema é inicializado a partir do método de Wang e Mendel [208], combinado a algoritmos de árvore de decisão. Por fim, o método é aplicado em nove bases de dados benchmarks, sendo três de alta dimensionalidade, e é efetuado uma comparação com nove diferentes SFGs.
- **Gacto et al. [83]**: trata de um SFG que emprega um SIF do tipo TSK em consórcio com um algoritmo evolutivo multi-objetivo. O modelo denomina-se Multi-Objective Evolutionary Algorithm-TSK (MOGA-TSK). O MOGA-TSK efetua o processo de parametrização das funções de pertinência e seleção das regras em dois estágios:

- O primeiro efetua o aprendizado da base de conhecimento a partir do método de Wang e Mendel [208] adaptado para TSK.
- O segundo estágio é um refinamento e ajuste dos parâmetros e regras definidos no primeiro estágio.
- Foram utilizados oito benchmarks, sendo três de alta dimensionalidade, e o método comparou os resultados com os do ANFIS [121] e do LEL-TSK [4].

Apesar de estes estudos conterem os ingredientes necessários para uma posterior comparação, optou-se pelo trabalho elaborado por Alcalá et al. [5], que propõe o modelo denominado Fast and Scalable Multiobjective Genetic Fuzzy System (FS-MOGFS). Na sua versão básica, é constituído por:

- Cada cromossoma (C) é constituído de duas partes ($C = C_1 \cup C_2$):
 - C_1 : esta parte decodifica o número de funções de pertinência triangulares e uniformemente dispostas, tal que: $C_1 = [L_1, L_2, \dots, L_k]$, onde L_k é o número de funções de pertinência que particionam a k -ésima variável, com $L_k \in \{1, 2, \dots, 7\}$. Se $L_k = 1$, a variável está inativa (don't care).
 - C_2 : o FS-MOGFS usa o conceito de representação em dupla. Para um $\alpha_k \in [-0.1, 0.1]$, $C_2 = [\alpha_1, \alpha_2, \dots, \alpha_k]$, com o intuito de que cada α_k seja um grau de deslocamento relativo ao conjunto de funções de pertinência da k -ésima variável.
- Para buscar os melhores valores para C , este incorpora um Multiobjective Genetic Algorithm (MOGA) baseado no SPEA2 [82]. Há dois objetivos: minimizar o erro (definido em termos de Erro Quadrático Médio - EQM) e o número de regras dos sistema. Para tanto, o MOGA usa operadores de cruzamento e mutação especiais [70, 99].
- Além das técnicas para ampliar a diversidade durante a rotina do FSMOGFS, este emprega uma técnica de reinicialização da população com elitismo, de modo a ampliar a diversidade das soluções no espaço de busca. Tal reinicialização é baseada em um métrica computada a partir da distância de Hamming e de outras informações advindas da população.
- De modo a gerar uma base de conhecimento completa (regras e funções de pertinência), para cada cromossoma é realizado uma rotina de extração de regras via Wang e Mendel [208]. Para evitar que a mesma base de regras seja criada, a cada geração uma parte dos padrões é omitida e

são geradas no máximo 50 regras por indivíduo. Caso algum indivíduo ultrapasse este valor, o mesmo é penalizado.

- Por fim, há uma inicialização especial da população, cujos detalhes são melhor descritos em Alcalá et al. [5]p.670. O SIF usado é do tipo Mamdani, com t-norma e implicação do tipo mínimo e defuzzificação por centro de gravidade.

Os autores estendem o FS-MOGFS para dois casos:

- **FS-MOGFS^e**: idêntico ao FS-MOGFS, entretanto com rápido cômputo do EQM ao desconsiderar uma porção do banco de dados. Esta abordagem não é tratada na comparação com o GPFIS.
- **FS-MOGFS+TUN**: similar ao anterior, contudo, dado o critério de parada (após duas reinicializações da população), efetua um ajuste fino nos parâmetros das funções de pertinência a partir do método descrito em [80]. Este modelo obteve os melhores resultados, perfazendo um total de 100000 avaliações. Assim, é o objeto de comparação com o GPFIS.

Alcalá et al. [5] considera as seguintes bases de dados para experimentação, conforme a Tabela 5.1.

Tabela 5.1: Bases de dados consideradas nos experimentos efetuados.

Base de dados	Abrv.	# Variáveis	# Padrões
Electrical Maintenance	ELE	4	1056
Auto MPG6	MPG6	5	398
Auto MPG8	MPG8	7	398
Analcat	ANA	7	4052
Abalone	ABA	8	4177
Stock	STP	9	950
Weather Izmir	WIZ	9	1461
Weather Ankara	WAN	9	1609
Forest Fires	FOR	12	517
Mortgage	MOR	15	1049
Treasury	TRE	15	1049
Baseball	BAS	16	337
MV Artificial Domain	MV	10	40768
Elevators	ELV	18	16559
Computer-Activity	CA	21	8192
Ailerons	AIL	40	13750
The Insurance Company	TIC	85	9822

Das 17 bases de dados, 5 são de elevada dimensionalidade (ELV, AIL, MV, CA e TIC). Para cada base de dados, o procedimento foi executar 6 vezes cada uma das 5 partições de validação cruzada (5-fold-cv). O número

de avaliações fixado foi de 100000 no total. Os autores efetuaram comparação com outros três SFGs. Em termos comparativos, a abordagem empreendida por Alcalá et al. [5] está mais próxima da de Gacto et al. [83].

A partir do conjunto de dados, número de avaliações e os algoritmos a serem comparados, o próximo tópico exibe o delineamento do experimento de modo a avaliar a melhor configuração para o modelo GPFIS.

Formulação dos Experimentos

Dado o número elevado de benchmarks, aproveitou-se para demonstrar de modo empírico o efeito da inclusão de operadores fuzzy, de agregação e dos métodos de particionamento. Para tanto, foi realizado o experimento sequencial disposto na Tabela 5.2.

Tabela 5.2: Delineamento experimental sequencial para o modelo GPFIS.

Descrição	Experimento 1	Experimento 2	Experimento 3
Hipótese Avaliada	Operadores Fuzzy	Operadores de Agregação	Métodos de Particionamento
Configuração Básica	Agregação por Máximo + Divisão Uniforme	Melhor configuração anterior + Divisão Uniforme	Melhor configuração anterior
Configurações Avaliadas	Operadores Fuzzy testados: - Produto (Prod) - Produto + Raiz Quadrada (Prod+RQ) - Produto + Negação (Prod+Neg) - Produto + Raiz Quadrada + Negação (Prod+RQ+Neg)	Operadores de Agregação testados: - Máximo - Média Aritmética - Produto - Mínimos Quadrados Restrito (MQR)	Métodos de Particionamento: - Divisão Uniforme (Div-Unif) - Grau de Similaridade (SD) - Grau de Confiança Fuzzy (FCD) - Máxima Credibilidade (Max-Cred)

O Experimento 1 visa a verificar o desempenho isolado das diferentes configurações do conjunto de operadores fuzzy. A hipótese central é avaliar o quanto de acurácia é proporcionado pela adição do operador de negação, de dilatação (raiz quadrada) e se há interação positiva entre estes. A t-norma escolhida é o produto, devido a sua capacidade de realizar operação de contração quando há o encontro de duas funções de pertinência idênticas.

O Experimento 2 tem como intuito investigar qual das diferentes operações de agregação produz o melhor resultado. Como em todos os experimentos precedentes, o método de particionamento é o da Divisão Uniforme. O

Experimento 3 proporciona avaliar outros métodos: Grau de Similaridade (SD) e de Confiança (Acurácia) Fuzzy (FCD) e Máxima Credibilidade (Max-Cred).

A Tabela 5.3 apresenta as demais configurações do modelo GPFIS. A Figura 5.1 exhibe o formato e número das funções de pertinência usadas, mantidas intactas ao longo de todo o processo de sintetização do modelo.

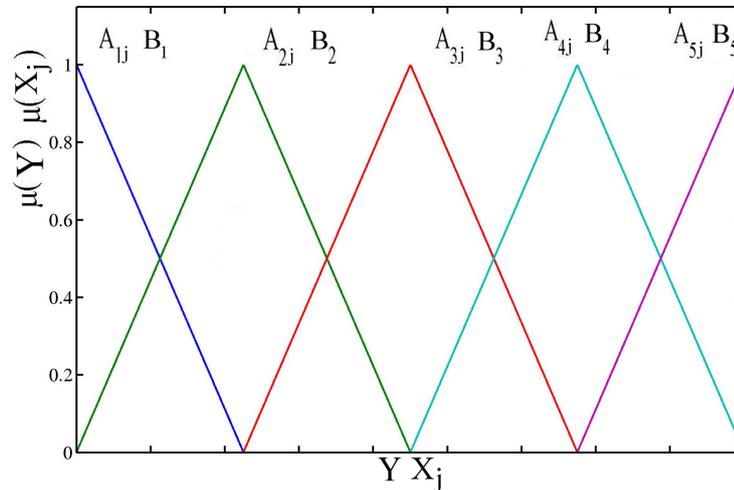


Figura 5.1: Formato e número das funções de pertinência usadas no GPFIS.

Tabela 5.3: Principais configurações do modelo GPFIS.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	1000
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.1
Operadores Fuzzy	Tabela 5.2

De forma similar à Alcalá et al. [5], são efetuadas 100000 avaliações para cada execução, tal que os resultados reportados são frutos de uma validação cruzada de 5 pastas (5-fold-cv). Para cada pasta o modelo é executado 6 vezes, e, portanto, as métricas relatadas para cada base de dados são uma média dos 30 modelos treinados (6 para cada uma das 5 pastas). Por fim, o nível de significância adotado para a investigação empírica foi de 5%.

O método de defuzzificação escolhido foi o da Altura, enquanto que a função de avaliação é dada pelo EQM:

$$EQM = \frac{1}{2} \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \quad (5-1)$$

O próximo tópico exhibe os resultados encontrados nos experimentos.

Resultados e Discussões

A Tabela 5.4 apresenta a média e o desvio padrão (σ) do EQM no Experimento 1, para as fases de treinamento (Trt.) e de teste (Tst.).

Tabela 5.4: Resultados do Experimento 1 para o conjunto de operadores fuzzy. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente.

Base de Dados	Prod		Prod +Neg		Prod+RQ		Prod +RQ+Neg	
	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.
ELE	123209	128386	34681	37529	112309	124277	35194	38327
σ	14185	32717	9203	12696	27999	44035	10390	12473
MPG6	7,6726	10,4497	3,9521	5,5366	7,044	10,0048	4,0947	5,6752
σ	0,6279	2,9335	0,4538	1,6573	0,7846	2,8157	0,4196	1,8377
MPG8	7,8005	9,2336	4,0490	6,0403	7,0262	8,6524	3,8644	5,4395
σ	0,2710	1,4889	0,7324	2,0113	0,5206	1,9052	0,5141	1,8071
ANA	0,0218	0,0224	0,0029	0,0034	0,0212	0,0223	0,0027	0,0032
σ	0,0019	0,0066	0,0002	0,0008	0,0016	0,0066	0,0003	0,0006
ABA	2,7818	2,8033	2,6017	2,6814	2,7682	2,8146	2,6032	2,6589
σ	0,1685	0,2815	0,1296	0,2596	0,1447	0,2711	0,1025	0,231
STP	3,8958	4,3784	1,7878	2,2178	3,5842	4,9118	1,9293	2,3689
σ	0,8074	1,5295	0,3760	0,8858	0,4199	2,6225	0,4093	0,7797
WIZ	2,9875	3,2784	2,1747	3,2947	3,4255	3,8828	2,412	3,0587
σ	0,0862	0,6132	0,6702	2,4577	1,5495	2,0059	0,6471	1,5041
WAN	5,6959	5,7972	3,4162	4,4576	5,4307	5,9654	3,4266	3,8391
σ	0,8467	1,1033	1,0713	2,4481	1,2537	2,4287	1,1565	1,7509
FOR	794	4067	424	3792	873	3375	614	3319
σ	639	2559	220	2311	635	2481	474	2197
MOR	0,0594	0,0709	0,0593	0,1119	0,0558	0,0866	0,0583	0,0755
σ	0,0087	0,0355	0,0152	0,1127	0,0211	0,0879	0,0278	0,0428
TRE	0,1405	0,4968	0,0755	0,1280	0,1455	0,4327	0,0877	0,1491
σ	0,0733	0,4644	0,0281	0,1336	0,0763	0,4563	0,0403	0,1096
BAS	2,6839	3,8977	1,8927	3,3928	2,5396	3,4664	1,9383	3,3694
σ	0,2654	0,9950	0,2597	0,8232	0,1876	0,6617	0,2135	0,8579
MV	4,2795	4,2757	3,1793	3,1764	3,1905	3,1858	2,4522	2,4532
σ	0,9823	0,9733	1,6218	1,5850	1,0229	0,9866	1,172	1,1698
ELV	1,2679	1,2761	0,9775	0,9916	1,1028	1,1129	0,9208	0,9311
σ	0,0424	0,0617	0,0777	0,0855	0,0152	0,0329	0,0498	0,0509
CA	11,0836	11,0417	6,1679	6,9241	10,8571	11,3852	6,0687	6,6985
σ	1,8889	1,9603	0,7995	0,9307	2,0077	2,8267	0,6366	1,2994
AIL	3,1753	3,2253	2,2273	2,2817	3,1016	3,1514	2,2934	2,3419
σ	0,1939	0,1995	0,2195	0,2450	0,1655	0,1949	0,0219	0,0245
TIC	0,0266	0,0273	0,0262	0,0272	0,0265	0,0271	0,0263	0,0271
σ	0,0006	0,0017	0,0004	0,0016	0,0004	0,0015	0,0004	0,0016

De uma forma geral, a abordagem Prod+RQ+Neg obteve os melhores resultados em 64,71% das bases de dados, enquanto que a abordagem simples do Produto produziu os piores resultados. A Tabela 5.5 exibe os resultados do teste de Friedman e Holm para comparar as 4 abordagens. De fato, como a combinação Prod+RQ+Neg obteve o menor posto (1,3824), este é selecionado para ser o objeto de comparação com as demais abordagens. Neste sentido, este auferiu os melhores resultados (p-valor < 0,05) com respeito às abordagens Prod e Prod+RQ. Não se observa diferença substancial entre as configurações Prod+Neg e Prod+RQ+Neg (p-valor = 0,1631).

Tabela 5.5: Resultados do teste de Friedman e Holm para o Experimento 1.

i	Método	Posto	
3	Prod	3,4706	
2	Prod+RQ	3,1471	
1	Prod+Neg	2,0000	
0	Prod+RQ+Neg	1,3824	
Teste	p-valor		
Friedman	< 0,0001		
Iman e Daveport	< 0,0001		
Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
Prod	4,7158	<0,0001	0,0167
Prod+RQ	3,9852	<0,0001	0,025
Prod+Neg	1,3948	0,1631	0,050

Infere-se que a inclusão da Negação Clássica tem maior impacto nos resultados do que a inclusão da Raiz-Quadrada. Porém, a união destes operadores amplia a acurácia do modelo. Tais resultados são esperados, devido à amplitude de combinações disponíveis que são geradas pela introdução destas técnicas no espaço de busca do modelo GPFIS.

A Tabela 5.6 apresenta as métricas de média e desvio padrão (σ) do EQM para o Experimento 2 na fase de treinamento e teste. Além disso, a Tabela 5.7 exibe as informações referentes ao volume de regras geradas por cada abordagem nas bases de dados de Baixa Dimensionalidade (BD) e Alta Dimensionalidade (AD).

A abordagem de agregação por Produto não foi efetiva, devido à dificuldade de se encontrarem regras que sejam ativadas ao mesmo tempo para um mesmo conseqüente (observe-se que, ao agregar regras pelo produto, troca-se o conectivo “ou” pelo “e”). Esta maior dificuldade se reflete nos resultados em termos de acurácia, como também no baixo número médio de regras geradas.

O uso do operador de agregação por Mínimos Quadrados Restritos (MQR) mostra ser uma abordagem positiva, tanto pela redução do número de

Tabela 5.6: Resultados do Experimento 2 para o conjunto de operadores de agregação. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente.

Base de Dados	Máximo		Média Arit.		Produto		MQR	
	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.
ELE	35194	38327	22649	25268	54809	61966	14208	16336
σ	10390	12473	2850	4119	13591	19122	1435	1599
MPG6	4,0947	5,6752	3,2882	4,3539	4,7187	6,1615	2,7035	4,1471
σ	0,4196	1,8377	0,1396	0,119	0,3331	0,7369	0,0577	0,1551
MPG8	3,8644	5,4395	3,2643	4,1509	4,7612	6,3376	2,6546	4,3211
σ	0,5141	1,8071	0,0987	0,2175	0,2263	0,7522	0,0920	0,3269
ANA	0,0027	0,0032	0,0026	0,0031	0,0029	0,0033	0,0022	0,0029
σ	0,0003	0,0006	0,0001	0,0001	0,0001	0,0002	0,0000	0,0001
ABA	2,6032	2,6589	2,4772	2,5085	2,7599	2,7965	2,3054	2,3756
σ	0,1025	0,231	0,0351	0,0402	0,0523	0,0508	0,0197	0,0390
STP	1,9293	2,3689	1,1649	1,2489	2,6984	2,9485	0,9615	1,1011
σ	0,4093	0,7797	0,0709	0,0643	0,3052	0,4901	0,0951	0,0940
WIZ	2,412	3,0587	1,5273	1,7241	3,2679	4,1842	0,8898	0,9927
σ	0,6471	1,5041	0,1117	0,2082	0,6477	1,3585	0,0515	0,0682
WAN	3,4266	3,8391	2,1292	2,3902	4,9217	6,281	1,4968	1,5623
σ	1,1565	1,7509	0,1851	0,2214	0,2932	1,904	0,1467	0,1610
FOR	614	3319	628	4644	1377	2964	509	3464
σ	474	2197	208	1884	234	764	121	633
MOR	0,0583	0,0755	0,0472	0,0512	0,0787	0,1082	0,0159	0,0196
σ	0,0278	0,0428	0,0024	0,0024	0,0165	0,0818	0,0013	0,0042
TRE	0,0877	0,1491	0,072	0,0799	0,1463	0,2319	0,0337	0,0423
σ	0,0403	0,1096	0,0081	0,0124	0,0235	0,0778	0,0037	0,0049
BAS	1,9383	3,3694	1,6911	2,9248	2,2933	3,5132	1,3464	3,1178
σ	0,2135	0,8579	0,07459	0,1605	0,1296	0,19639	0,1004	0,3696
MV	2,4522	2,4532	2,3595	2,3626	3,3694	3,3822	1,1567	1,1587
σ	1,172	1,1698	0,3797	0,3738	0,7484	0,7821	0,2565	0,2618
ELV	0,9208	0,9311	0,8487	0,8572	1,0342	1,0453	0,6788	0,6927
σ	0,0498	0,0509	0,0161	0,0186	0,0271	0,0314	0,0330	0,0410
CA	6,0687	6,6985	5,9776	6,1632	7,2542	7,717	5,4492	5,6553
σ	0,6366	1,2994	0,6077	0,5553	0,3651	0,2695	0,6357	0,7019
AIL	2,2934	2,3419	1,9719	1,9902	2,8677	2,9268	1,7267	1,7546
σ	0,0219	0,0245	0,08658	0,0931	0,1223	0,1003	0,0054	0,0065
TIC	0,0263	0,0271	0,0262	0,0270	0,0267	0,0274	0,0257	0,0267
σ	0,0004	0,0016	0,0001	0,0002	0,0001	0,0002	0,0007	0,0015

regras¹ quanto por garantir EQM menores em média. Este operador produziu os melhores resultados em 82,35% dos casos, seguido da agregação pela Média Aritmética (11,76% dos casos), que, por sua vez gerou um número grande de regras. O operador clássico do Máximo não foi tão efetivo, seja pela acurácia ou pela na geração de uma base de regras mais compacta.

A Tabela 5.8 mostra os principais resultados do teste de Friedman e Holm, que comprovam a significância da abordagem MQR (p-valor < 0,05).

De fato, tais resultados eram esperados por causa do maior investimento

¹Uma regar foi considerada como ativa quando o peso atrelado a ela é maior do que 0,01.

Tabela 5.7: Regras geradas por cada abordagem operador de agregação.

Métrica \ Abordagem	Máximo	Média	Produto	MQR
Média (BD)	26,32	40,12	13,33	15,65
Desvio Padrão (BD)	8,05	6,64	6,80	2,88
Média (AD)	30,71	42,96	16,17	14,54
Desvio Padrão (AD)	7,84	2,16	7,08	2,00

Tabela 5.8: Resultados do teste de Friedman e Holm para o Experimento 2.

i	Método	Posto	
3	Produto	3,6765	
2	Máximo	3,0882	
1	Média	2,0588	
0	MQR	1,1765	
Teste	p-valor		
Friedman	< 0,0001		
Iman e Daveport	< 0,0001		
Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
Produto	5,6457	<0,0001	0,0167
Máximo	4,3173	<0,0001	0,025
Média	1,9926	0,0463	0,050

computacional demandado pelo MQR. A Tabela 5.9 apresenta a média do tempo computacional adicional (em termos percentuais) com relação aos demais operadores de agregação. De fato, o investimento computacional do MQR é intensivo, ao demandar de 5 a 8 vezes mais tempo do que os operadores Produto e Máximo. Com respeito a Média Aritmética, esta distância reduz para 4 a 6 vezes mais tempo, devido ao elevado número de regras geradas por esse operador. Esta característica deve ser observada pelo usuário no momento da escolha do operador de agregação, embora em termos absolutos resultem em minutos a mais.

Tabela 5.9: Proporção adicional de tempo tomado pela abordagem MQR com relação as demais operações de agregação.

Métrica \ Abordagem	Máximo	Média	Produto
Média (BD)	502,1%	424,7%	735,2%
Desvio Padrão (BD)	38,4%	46,1%	56,7%
Média (AD)	590,1%	542,4%	826,6%
Desvio Padrão (AD)	47,8%	42,6%	77,8%

A Tabela 5.10 apresenta os principais resultados no tocante aos diferentes métodos de particionamento (Experimento 3), enquanto que a Tabela 5.11

apresenta o número médio de regras e o desvio padrão das abordagens nas bases de dados de BD e AD.

Tabela 5.10: Resultados do Experimento 3 para o conjunto de métodos de particionamento. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente.

Base de Dados	Div-Unif,		SD		FCD		Max-Cred	
	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.
ELE	14208	16336	14695	16818	11922	14059	26974	30018
σ	1435	1599	815	493	1213	2666	8140	10296
MPG6	2,7035	4,1471	2,8896	4,0029	2,7227	4,3333	3,3155	4,2491
σ	0,0577	0,1551	0,0664	0,3359	0,2181	1,3297	0,2792	1,1153
MPG8	2,6546	4,3211	2,7800	4,0874	2,5591	4,1275	3,2877	4,3592
σ	0,0920	0,3269	0,0910	0,1161	0,3207	1,1697	0,3822	1,4173
ANA	0,0022	0,0029	0,0025	0,0029	0,0024	0,0030	0,0036	0,0046
σ	0,0000	0,0001	0,0001	0,0001	0,0002	0,0007	0,0022	0,0030
ABA	2,3054	2,3756	2,3698	2,4253	2,4853	2,5552	2,5495	2,5723
σ	0,0197	0,0390	0,0678	0,1540	0,0556	0,1733	0,1292	0,2350
STP	0,9615	1,1011	1,1431	1,2829	1,0675	1,2240	1,4281	1,5980
σ	0,0951	0,0940	0,1730	0,2562	0,1796	0,2553	0,2290	0,3037
WIZ	0,8898	0,9927	0,8110	0,8775	0,7192	0,8047	1,2522	1,3570
σ	0,0515	0,0682	0,0455	0,0400	0,0453	0,0913	0,3913	0,4691
WAN	1,4968	1,5623	1,3068	1,3673	1,1108	1,1902	1,8025	1,9905
σ	0,1467	0,1610	0,0791	0,0918	0,0830	0,1374	0,5771	0,8050
FOR	509	3464	1476	2446	1049	2899	733	3266
σ	121	633	868	2456	682	2157	401	2066
MOR	0,0159	0,0196	0,0129	0,0153	0,0092	0,0109	0,0242	0,0276
σ	0,0013	0,0042	0,0013	0,0024	0,0008	0,0018	0,0066	0,0116
TRE	0,0337	0,0423	0,0306	0,0367	0,0245	0,0320	0,3940	0,4650
σ	0,0037	0,0049	0,0015	0,0018	0,0021	0,0120	0,1620	0,2100
BAS	1,3464	3,1178	1,4685	3,0374	1,3132	3,1067	1,7769	3,2320
σ	0,1004	0,3696	0,0674	0,2461	0,1251	0,6253	0,1756	0,8145
MV	1,1567	1,1587	2,6066	2,5992	0,5029	0,5062	2,8241	2,8023
σ	0,2565	0,2618	1,4990	1,4899	0,2324	0,2225	1,3898	1,3311
ELV	0,6788	0,6927	0,8751	0,8863	0,8622	0,8784	0,8908	0,9043
σ	0,0330	0,0410	0,1064	0,1080	0,0391	0,0398	0,0702	0,0766
CA	5,4492	5,6553	4,8850	5,0601	4,0623	4,2043	7,9300	8,2099
σ	0,6357	0,7019	0,7288	0,7388	0,7252	0,8595	1,2671	1,2895
AIL	1,7267	1,7546	1,8288	1,8577	1,7798	1,8102	1,8860	1,9170
σ	0,0054	0,0065	0,0035	0,0033	0,1273	0,1280	0,1673	0,1917
TIC	0,0257	0,0267	0,0258	0,0267	0,0275	0,0282	0,0258	0,0268
σ	0,0007	0,0015	0,0003	0,0014	0,0004	0,0016	0,0004	0,0014

Tabela 5.11: Regras geradas por cada método de particionamento.

Métrica \ Abordagem	Div-Unif.	SD	FCD	Max-Cred
Média (BD)	15,65	15,14	13,67	15,18
Desvio Padrão (BD)	8,05	3,39	6,80	2,88
Média (AD)	14,54	13,01	16,89	14,25
Desvio Padrão (AD)	7,84	2,91	7,08	2,06

Os métodos de particionamento baseados no conceito de Máxima Confiança (FCD e SD) obtiveram os melhores resultados em 76,47 % dos casos, demonstrando ser uma alternativa viável em termos de acurácia quando comparados ao método ingênuo de Divisão Uniforme. Há uma semelhança de resultados entre o FCD e o SD, sendo o FCD um pouco superior (58,82%) nos casos analisados. Com relação ao tempo computacional (Tabela 5.12), os métodos são semelhantes, entretanto com tempo computacional mais elevado do que o da Divisão Uniforme. Em geral, o método de Máxima Credibilidade não apresentou bons resultados.

Tabela 5.12: Proporção adicional de tempo tomado pelas demais abordagens com respeito ao método de Divisão Uniforme.

Métrica \ Abordagem	FCD	SD	Max Cred.
Média (BD)	92,1%	84,7%	75,2%
Desvio Padrão (BD)	18,4%	11,1%	9,7%
Média (AD)	100,1%	92,4%	86,6%
Desvio Padrão (AD)	37,8%	22,6%	17,8%

A semelhança de resultados entre o FCD e o SD são manifestados no teste de Friedman e Holm (Tabela 5.13), tornando inviável qualquer conclusão quanto ao particionamento de melhor desempenho. De fato, ambos não são substancialmente superiores à Divisão Uniforme. Contudo é possível observar que o FCD obteve resultados significativamente melhores do que a Máxima Credibilidade.

Tabela 5.13: Resultados do teste de Friedman e Holm para o Experimento 3.

i	Método	Posto	
3	Max-Cred	3,8235	
2	Div-Unif	2,2353	
1	SD	2,000	
0	FCD	1,9412	
Teste	p-valor		
Friedman	< 0,0001		
Iman e Daveport	< 0,0001		
Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
Max. Cred.	4,2509	<0,0001	0,0167
Div. Unif.	0,6642	0,5065	0,025
SD	0,1328	0,8943	0,050

Como os métodos baseados na Máxima Confiança (SD e FCD) não obtiveram resultados estatisticamente melhores do que o da Divisão Uniforme, a escolha final reside naquele que apresentou melhores resultados em termos

numéricos. Como o SD e o FCD foram superiores na maioria das bases de dados, e ambos obtiveram resultados bastante semelhantes, opta-se pelo uso dos dois na comparação estrita com demais SFGs. Assim, na confrontação com outras abordagens, é possível efetuar um desempate. Então, tem-se a configuração final do modelo GPFIS:

- **Operadores Fuzzy:** Produto, Negação e Raiz Quadrada.
- **Operador de Agregação:** MQR.
- **Método de Particionamento:** FCD ou SD.

O próximo tópico aborda a descrição e os resultados da comparação com os outros SFGs.

5.1.2

Comparação com Demais Sistemas Fuzzy-Genéticos

Considere-se que o modelo GPFIS com Método de Particionamento por SD é denotado por GPFIS-SD, enquanto que o baseado em FCD é o denominado GPFIS-FCD. A comparação final é resultado da sequência de experimentos:

1. Modelo GPFIS-SD e GPFIS-FCD com relação a todos os outros SFGs.
2. Comparação estrita entre o melhor GPFIS e o FS-MOGFS+TUN.

Do primeiro experimento, além do modelo FS-MOGFS+TUN proposto por Alcalá et al. [5], como descrito na introdução, outros três SFGs são usados para comparação:

- GR-MF [56]: utiliza o algoritmo evolutivo para definir a granularidade e os parâmetros das funções de pertinência triangulares de um SIF do tipo Mamdani. Dada uma disposição das funções de pertinência, emprega o método de Wang e Mendel [208] para gerar as regras do sistema.
- GA-WM [55]: o modelo busca, através do AG, sintetizar a granularidade, a amplitude das funções de pertinência triangulares e adaptar o universo de discurso da variável, ora o contraindo, ora o ampliando. De forma similar, dada uma solução, a base de regras é criada a partir do procedimento de Wang e Mendel [208].
- GLD-WM [4]: esta abordagem é semelhante ao FS-MOGFS no tocante ao aprendizado da granularidade de cada variável, além da inclusão do deslocamento das funções de pertinência triangulares e uniformemente

distribuídas, através da representação em dupla. Por fim, a base de regras é gerada via Wang e Mendel [208]. Entretanto, o ajuste final das funções de pertinência, como no caso do FS-MOGFS+TUN, não é realizado.

Todos estes possuem um único objetivo: maximizar a acurácia do SIF. O segundo experimento visa a dois objetivos: efetuar uma comparação direta do modelo GPFIS mais acurado com o FS-MOGFS+TUN e sanar o fato de que os demais SFGs não conseguiram produzir resultados para as cinco bases de dados de alta dimensionalidade [5]. Em todas as bases de dados e experimentos foram seguidos os procedimentos descritos anteriormente: 100000 avaliações e 6 execuções para cada pasta da validação cruzada (5-fold-cv).

A partir dessa configuração, a Tabela 5.14 exhibe os resultados do modelo GPFIS-SD, GPFIS-FCD e de seus pares SFGs presentes em Alcalá et al. [5].

Em geral, o modelo GPFIS-FCD obteve melhores resultados em 41% do total das bases de dados, acompanhado do FS-MOGFS+TUN com 23% e do GPFIS-SD em 18% dos casos (Figura 5.2). Ocorreram empates em duas bases de dados, enquanto que o modelo GLD-WM obteve o melhor resultado em uma única base. Os demais SFGs não obtiveram bons resultados. Em problemas de alta dimensionalidade, o modelo GPFIS-FCD obteve melhores resultados em três das cinco bases de dados.

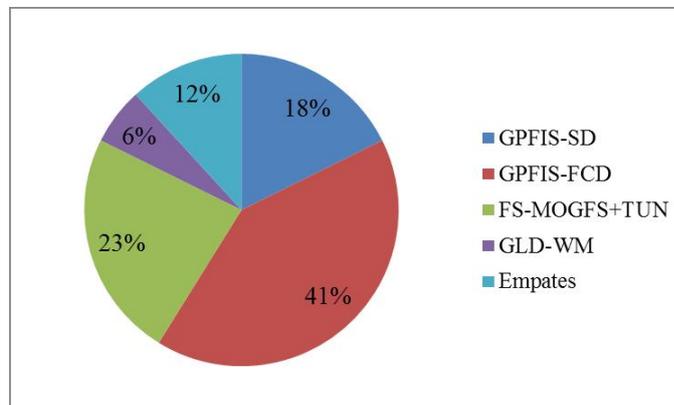


Figura 5.2: Percentual de casos em que os modelos obtiveram os melhores resultados em acurácia.

Tabela 5.14: Resultado final do modelo GPFIS e dos demais SFGs. Valores na linha do σ representam o desvio padrão do EQM. Resultado nesta tabela (Tra./Tst.) devem ser multiplicados por 10^5 , 10^{-5} e 10^{-8} no caso de BAS, ELV e AIL respectivamente. R/A - Média de Regras e Antecedentes.

Base de Dados	GR-MF			GA-WM			GLD-WM			FS-MOGFS+TUN			GPFIS -SD			GPFIS -FCD		
	R/A	Tra.	Tst.	R/A	Tra.	Tst.	R/A	Tra.	Tst.	R/A	Tra.	Tst.	R/A	Tra.	Tst.	R/A	Tra.	Tst.
ELE	97/4	16645	18637	47/4	17230	18977	33/4	11483	13384	9/2	8803	9842	16/3	14695	16818	19/3	11922	14059
σ		2319	3386		2501	3195		1085	1978		739	1391		815	493		1213	2666
MPG6	243/5	1,423	28,933	186/5	1,879	8,824	82/5	2,294	4,387	22/3	2,778	4,548	18/3	2,890	4,003	21/3	2,723	4,333
σ		0,073	8,633		0,235	6,079		0,249	0,899		0,220	1,047		0,066	0,336		0,218	1,330
MPG8	262/7	1,356	49,36	214/7	1,563	15,216	135/7	1,709	4,782	24/3	2,725	4,381	18/3	2,780	4,087	23/3	2,559	4,128
σ		0,104	16,2		0,183	9,13		0,170	1,445		0,294	0,909		0,091	0,116		0,321	1,170
ANA	148/7	0,005	0,017	150/7	0,003	0,008	92/7	0,006	0,008	17/3	0,003	0,003	8/3	0,002	0,003	14/4	0,002	0,003
σ		0,001	0,008		0,001	0,005		0,001	0,004		0,000	0,001		0,000	0,000		0,000	0,001
ABA	498/8	2,358	2,885	143/8	2,433	2,549	31/8	2,487	2,545	10/3	2,393	2,454	16/3	2,370	2,425	19/3	2,485	2,555
σ		0,052	0,263		0,052	0,163		0,078	0,170		0,092	0,163		0,068	0,154		0,056	0,173
STP	343/9	0,4	1,543	344/9	0,389	2,192	217/9	0,299	0,435	25/3	0,724	0,892	17/3	1,143	1,283	18/3	1,068	1,224
σ		0,019	2,484		0,017	3,168		0,025	0,067		0,112	0,154		0,173	0,256		0,180	0,255
WIZ	331/9	1,176	9,602	218/9	1,233	3,529	107/9	0,926	1,150	15/3	0,867	1,011	17/3	0,811	0,878	21/3	0,719	0,805
σ		0,077	8,879		0,065	4,023		0,041	0,123		0,040	0,177		0,046	0,040		0,045	0,091
WAN	397/9	1,406	7,381	279/9	1,522	2,82	133/9	1,111	2,075	11/2	1,313	1,581	15/2	1,307	1,367	19/3	1,111	1,190
σ		0,067	5,404		0,065	2,825		0,077	1,407		0,174	0,580		0,079	0,092		0,083	0,137
FOR	396/12	113	3300	395/12	47	3693	377/12	49	3847	33/3	1593	2406	8/3	1476	2446	8/5	1049	2899
σ		17	2207		24	2787		18	2714		570	2161		868	2456		682	2157
MOR	209/15	0,03	0,176	160/15	0,02	0,093	78/15	0,016	0,022	9/3	0,015	0,018	15/3	0,013	0,015	16/2	0,009	0,011
σ		0,002	0,28		0,003	0,147		0,002	0,005		0,004	0,012		0,001	0,002		0,001	0,002
TRE	189/15	0,066	0,144	136/15	0,045	0,064	70/15	0,033	0,045	11/3	0,030	0,040	15/3	0,031	0,037	18/3	0,025	0,032
σ		0,011	0,191		0,007	0,046		0,005	0,015		0,004	0,012		0,002	0,002		0,002	0,012
BAS	262/16	0,255	12,439	262/16	0,202	11,706	244/16	0,138	3,610	21/6	1,305	2,699	16/3	1,469	3,037	20/3	1,313	3,107
σ		0,02	2,177		0,031	2,562		0,014	0,621		0,172	0,620		0,067	0,246		0,125	0,625
MV	-	-	-	-	-	-	-	-	-	16/3	0,159	0,160	10/3	2,607	2,599	20/3	0,503	0,506
σ											0,031	0,032		1,499	1,490		0,232	0,223
ELV	-	-	-	-	-	-	-	-	-	8/3	0,900	0,900	12/3	0,875	0,886	18/3	0,862	0,878
σ											0,200	0,200		0,106	0,108		0,039	0,040
CA	-	-	-	-	-	-	-	-	-	15/5	4,763	5,063	15/4	4,885	5,060	18/4	4,062	4,204
σ											0,404	0,760		0,729	0,739		0,725	0,860
AIL	-	-	-	-	-	-	-	-	-	20/4	1,864	1,905	17/3	1,829	1,858	21/3	1,780	1,810
σ											0,221	0,233		0,003	0,003		0,127	0,128
TIC	-	-	-	-	-	-	-	-	-	25/7	0,026	0,027	11/4	0,026	0,027	7/4	0,028	0,028
σ											0,000	0,002		0,000	0,001		0,000	0,002

A Tabela 5.15 apresenta o resultado do teste de Friedman e Holm para a comparação entre os SFGs, avaliando apenas as bases de baixa dimensionalidade e o nível de significância de 10%, conforme Alcalá et al. [5].

Tabela 5.15: Resultados do teste de Friedman e Holm para a comparação entre os SFGs, considerando apenas as bases de dados de baixa dimensionalidade.

i	Modelo	Posto
5	GR-MF	5,6667
4	GA-WM	5,0417
3	GLD-WM	3,6250
2	FS-MOGFS+TUN	2,3750
1	GPFIS-FCD	2,2917
0	GPFIS-SD	2,0000

Teste	p-valor
Friedman	< 0,0001
Iman e Daveport	< 0,0001

Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
GR-MF	4,8007	< 0,0001	0,020
GA-WM	3,9824	< 0,0001	0,025
GLD-WM	2,1276	0,0333	0,0333
FS-MOGFS+TUN	0,4909	0,6234	0,050
GPFIS-FCD	0,3818	0,7025	0,100

Dentre todos os modelos, o GPFIS-SD obteve o menor posto (2,0000), sendo este escolhido para ser o método de controle para comparação no teste de Holm. É possível verificar que o modelo GPFIS-SD obteve significativamente maior acurácia com relação a GR-MF, GA-WM e GLD-WM (p-valor < 0,033). Tal conclusão não se verifica com relação aos modelos GPFIS-FCD e FS-MOGFS+TUN (p-valor > 0,10). Este demonstra que, apesar de o modelo GPFIS-SD não ter obtido com maior prevalência os melhores resultados nas bases de dados, esse esteve sempre próximo do melhor, o que se reflete no seu posto menor. Este resultado viabiliza o segundo experimento, que enseja a comparação direta entre o GPFIS-SD (menor posto) e o FS-MOGFS+TUN.

Ao considerar estritamente os dois modelos, é possível verificar que em 10 das 17 bases de dados (58,88%) o modelo GPFIS-SD obteve melhores resultados do que o FS-MOGFS+TUN, ocorrendo 2 empates no total. Ao efetuar o teste do sinal, esta diferença não foi tida como significativa (S=10, p-valor=0,3018). Isto pode ter ocorrido devido aos empates e ao baixo número de bases de dados usadas. Em termos de complexidade da base de regras fuzzy, é possível observar que o GPFIS-SD obteve em 53% dos casos a base de regras

mais compacta², seguido do FS-MOGFS+TUN em 41% do total (Figura 5.3).

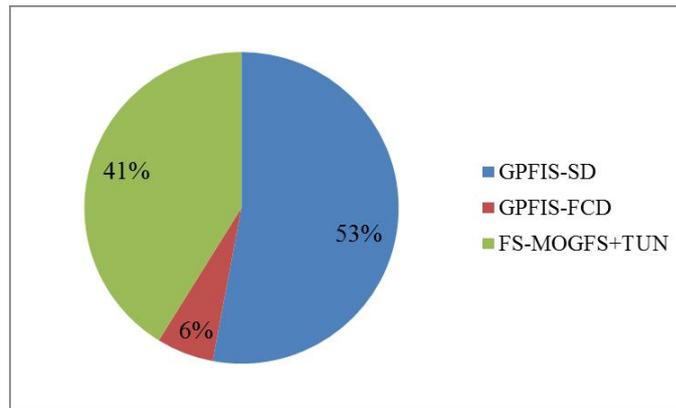


Figura 5.3: Percentual de casos em que os modelos obtiveram os melhores resultados na relação número médio de regras e antecedentes por regra.

Por fim, considerando interpretabilidade e implementação, o modelo GPFIS supera o FS-MOGFS+TUN em diversos fatores, tais como:

- Não promove alteração na parametrização das funções de pertinência, como no caso do FS-MOGFS+TUN. Este promove o deslocamento das funções de pertinência e foram afetadas conforme mostrado na Figura 5.4.

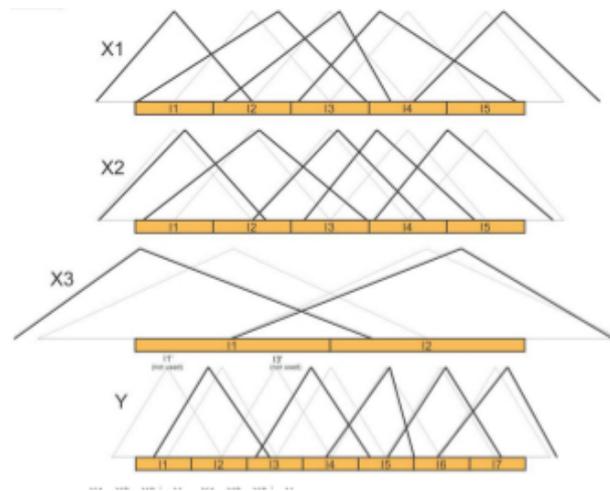


Figura 5.4: Exemplo do formato final das funções de pertinência do modelo FS-MOGFS+TUN para a base AIL.

- Usa uma meta-heurística de um único objetivo, enquanto que o FS-MOGFS+TUN realiza uma busca multi-objetivo.
- Em geral, o coeficiente de variação dos resultados do GPFIS é menor do que o do FS-MOGFS+TUN.

²Cabe ressaltar que o modelo GPFIS auferiu em 18% dos casos a intersecção entre menor complexidade e maior acurácia nas bases de dados em análise.

Tais fatores advogam pela qualidade do modelo GPFIS em produzir bons resultados sem grandes esforços computacionais ou complexidade de implementação.

A próxima seção exibe uma aplicação detalhada do modelo GPFIS no ramo da Psicolinguística.

5.1.3

Aplicação Detalhada

Descrição do Problema

A aplicação proposta para o uso do modelo GPFIS reside na área de Psicolinguística³. O objetivo principal é demonstrar a aplicabilidade da modelagem proposta na descoberta de associações e inferências com respeito ao comportamento de indivíduos. Teve-se acesso a uma tese de doutorado [182], aplicando um SIF do tipo Mamdani para descrever graus de bilinguismo de indivíduos, um fenômeno que, segundo a autora, é inviável de ser modelado pela incerteza probabilística. É possível encontrar trabalhos na área de processamento e descrição de emoções a partir de textos ou frases em uma abordagem computacional [36, 199], mas estes não tangenciam a aplicação sob análise.

O estudo em questão insere-se no âmbito de uma investigação mais ampla sobre como se dá a integração entre a informação linguística e visual na leitura de gráficos. Portanto, o experimento tem por objetivo avaliar como o conhecimento da informação representada graficamente (se a informação apresentada é conhecida/desconhecida) influencia na análise do gráfico. Logo, a tarefa experimental consiste em avaliar se um gráfico apresentado, após uma frase, representa ou não corretamente o conteúdo da frase. Para responder, os participantes devem usar teclas do computador em que foram afixadas as letras S (sim) e N (não), sendo os tempos de tomada de decisão registrados em um programa usado em pesquisa psicolinguística.

De forma sucinta, a Tabela 5.16 expõe as principais variáveis e objetos do estudo.

³Agradecimentos à Profa. Erica Rodrigues e a Luane Fragoso pela disponibilização da base de dados, auxílio e discussão na escrita dos experimentos e resultados.

Tabela 5.16: Exemplo de frases e dos principais fatores considerados no estudo.

Lista	Frase	Tipo de Informação	Correção	# Itens	Comparativo
Lista 1	1 - A bicicleta é o meio de transporte menos poluente.	CC	CG	1	Menos
	2 - TV Câmara e TV Senado são os programas televisivos menos assistidos pela população brasileira.	CC	CG	2	Menos

	15 - San Pedro Sula é a cidade das Américas com o maior número de homicídios por habitante.	DC	IG	1	Mais
Lista 2	1- A bicicleta é o meio de transporte menos poluente.	CC	IG	1	Menos
	2 - TV Câmara e TV Senado são os programas televisivos menos assistidos pela população brasileira.	CC	IG	2	Menos

	15 - San Pedro Sula é a cidade das Américas com o maior número de homicídios por habitante.	DC	CG	1	Mais

Legenda: CC - Conhecida, DC - Desconhecida, CG - Congruente e IG - Incongruente.

Foram construídas duas listas, tal que cada uma contém o total de 15 frases. Cada frase descreve uma situação, como, por exemplo: “San Pedro Sula é a cidade das Américas com o maior número de homicídios por habitante”. A partir da situação descrita, o leitor é convidado a verificar, a partir de um gráfico (Figura 5.5) se este é Congruente ou Incongruente com a frase.

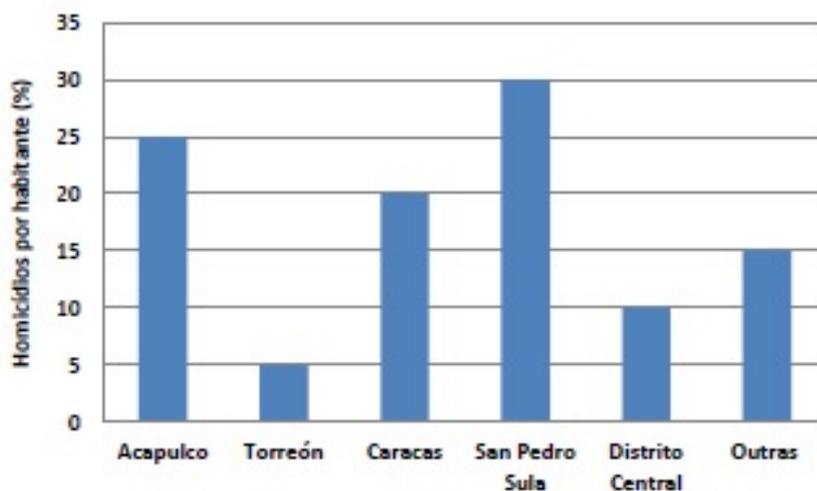


Figura 5.5: Exemplo de gráfico a ser analisado pelos sujeitos da pesquisa.

Cada situação apresentada ao leitor é composta por quatro variáveis manipuladas:

- **Tipo de Informação:** se a frase é de conhecimento geral⁴ (2 níveis: Conhecido ou Desconhecido).
- **Compatibilidade:** se o gráfico expressa adequadamente o conteúdo da frase (2 níveis: Congruente ou Incongruente).
- **# Itens:** quantos itens (TV Globo, Bicicleta, etc.) estão contidos nesse fato (2 níveis: 1 ou 2 itens).
- **Comparativo:** se o advérbio de comparação expressa uma comparação de superioridade ou inferioridade (2 níveis: mais ou menos, maior ou menor, etc.).

Cada frase pertencente a uma lista é exposta para um grupo e são mensuradas duas quantidades:

- **Sim/Não:** considera-se um Sim quando o indivíduo afirma que a frase e o gráfico são congruentes, e Não para o caso contrário.
- **Tempo de Resposta (ms):** registra-se o tempo em milissegundos que o sujeito leva para efetuar a resposta.

⁴Considerando um grupo com ensino médio completo.

Como a aplicação envolve um problema de regressão, não se levou em conta a variável Sim/Não, embora em novos estudos ela possa ser incorporada à modelagem. Cabe ressaltar que a diferença entre as listas reside em comutar as frases e gráficos congruentes em incongruentes, e vice-versa. Portanto, o objetivo do estudo é avaliar em que medida o conhecimento prévio acerca de um conteúdo expresso para um gráfico pode vir a influenciar os tempos de resposta dos sujeitos, bem como a correção das respostas. A priori os especialistas formulam algumas previsões em função das hipóteses em estudo:

- Se a frase é conhecida, então tempo de resposta é menor.
- Se frase é desconhecida, então tempo de resposta é maior.
- A presença de 2 itens demanda um tempo de processamento maior pelos indivíduos.
- Itens negativos (menor, menos) pedem mais tempo de resposta dos sujeitos.
- A interação entre Tipo de Informação/Correção deve ser considerada no estudo. Um possível ordenamento com respeito ao tempo de resposta seria:
 1. Conhecida e Congruente.
 2. Desconhecida e Congruente.
 3. Conhecida e Incongruente.
 4. Desconhecida e Incongruente.

Na abordagem tradicional da Psicolinguística, a partir dessas hipóteses de estudo e dada uma coleta de dados, a investigação seguiria para o uso de métodos estatísticos, tais como Análise de Variância [58, 59] e Modelos de Regressão [210]. Como é de conhecimento, estes modelos são restritos a pressupostos que nem sempre são verificados na prática, como, por exemplo, Normalidade, Homocedasticidade, entre outros. Contudo, não se tem a intenção de utilizar o modelo GPFIS por causa da restritividade de tais métodos, mas sim pelas características próprias do GPFIS, tais como descoberta de conhecimento, maior interpretabilidade dos resultados e possibilidade de incorporar o conhecimento a priori dos especialistas. Ainda, o GPFIS pode atuar de forma conjunta, pois a sua inferência não se assemelha em grande maneira à dos métodos estatísticos. Logo, ao considerar os resultados apresentados por cada método pode possibilitar ao pesquisador uma análise mais refinada das hipóteses preliminares.

O próximo tópico aborda a descrição dos experimentos.

Descrição dos Experimentos

Com o intuito de verificar se há influência no tempo de resposta dos sujeitos dadas variações na estrutura das frases, uma amostra aleatória contendo 60 indivíduos foi considerada. Metade destes (30 indivíduos) foram apresentados às frases presentes na Lista 1, enquanto os demais, à Lista 2. De forma simplificada, a seguir são descritas as variáveis independentes e a dependente do experimento:

- Variáveis independentes, ou entradas do modelo GPFIS (Todas estas variáveis binárias):
 - Tipo de Informação (TI)
 - Correção (COR)
 - Número de Itens (NI)
 - Comparativo (COMP)
- Variável dependente, ou saída do modelo GPFIS:
 - Tempo de Resposta em milissegundos (ms)

Como a dispersão com relação à média dos tempos de resposta é elevada (coeficiente de variação = 59,34%), é justificável a aplicação de uma transformação. Para tanto, ao invés de se trabalhar com o tempo de resposta, opta-se pelo logaritmo neperiano do tempo de resposta. Esta transformação, conforme demonstrado pela Figura 5.6, reduziu bastante a dispersão em torno da média (coeficiente de variação da transformação = 6,14%).

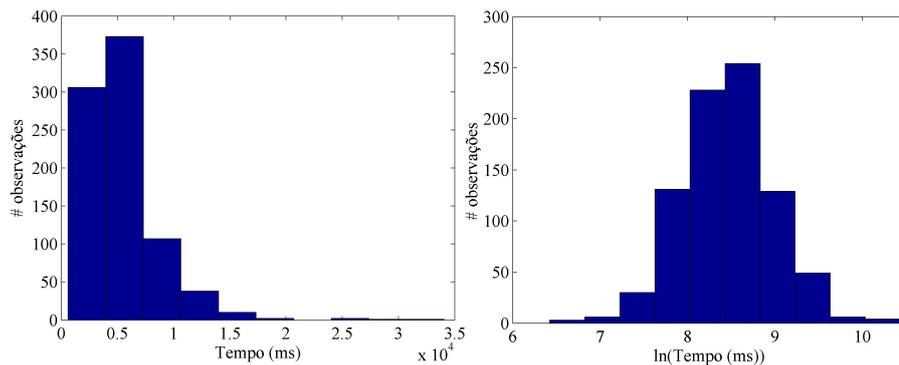


Figura 5.6: Histograma das observações de tempo antes e após aplicar o logaritmo natural.

Além do modelo GPFIS, outros modelos foram empregados: Regressão Log-Linear, Árvore de Regressão, PGMG e o método de Wang e Mendel para extração de regras. A Tabela 5.17 apresenta cada um destes, além dos parâmetros que os compõem.

Tabela 5.17: Modelos para regressão e parâmetros usados.

Modelo	Parâmetro
GPFIS	Tabela 5.18
PGMG	Tabela 5.18
Árvore de Regressão (AR) [96]	-
Regressão Log-Linear (REG+SW)	Stepwise [96]
Wang e Mendel para Regressão (WM-R) [208]	-

O procedimento experimental usado foi a validação cruzada em 5 pastas (5-fold-cv). Neste caso específico, 6 indivíduos de cada lista foram selecionados para a fase de teste (12 no total); os demais foram usadas para treinamento dos modelos. Para cada pasta, os algoritmos são executados 6 vezes e a média das métricas obtidas dos 30 resultados são reportadas.

A partir de informações dos especialistas, a Figura 5.7 apresenta as funções de pertinência delineadas para modelar o tempo de resposta.

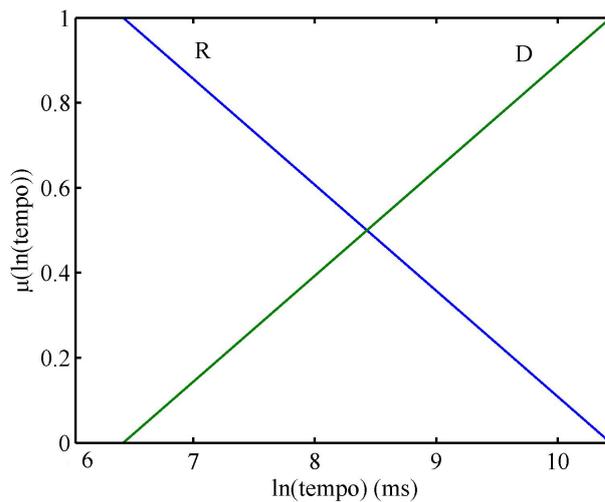


Figura 5.7: Funções de pertinência usadas no modelo GPFIS e no WM-R.

Cada rótulo linguístico representa: Rápido (R) e Devagar (D). Logo, o objetivo do modelo GPFIS (e do WM-R) é verificar a associação de cada variável independente e suas interações com respeito a estes dois consequentes. A função de avaliação dos modelos evolutivos é o EQM (erro quadrático médio). Também foi computado o quadrado da correlação entre o valor predito e o observado (Cor^2), para medir de forma adimensional o grau de similaridade entre os valores exibidos pelo modelo e o observado entre os sujeitos. Esta métrica também exerce uma ligação com o clássico coeficiente de determinação (R^2), que neste caso tem seu cálculo inviabilizado (no sentido de interpretação),

Tabela 5.18: Principais configurações dos modelos baseados em Programação Genética.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	50
Altura máxima da árvore	7
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.7
Operadores Fuzzy	Produto e Negação
Operador de Agregação	MQR
Método de Particionamento	SD
Defuzzificação	Altura

devido à ausência de intercepto (nível) em alguns modelos, como o GPFIS e WM-R.

Resultados e Discussões

A Tabela 5.19 exhibe os principais resultados dos modelos, tanto no sentido da acurácia (EQM, Cor^2) quando no da complexidade da abordagem (número de regras, nós ou coeficientes). De forma geral, o modelo de AR apresentou os melhores resultados, tanto em termos de EQM, tanto como de Cor^2 . O modelo GPFIS figura em penúltimo lugar, sendo somente superior ao WM-R. Em termos de complexidade, a AR só fica atrás da PGMG, enquanto o modelo de Regressão Log-Linear com Step-Wise (REG+SW) apresenta um bom custo-benefício. O modelo GPFIS é o segundo menos complexo. A Figura 5.8 apresenta uma solução típica da AR.

Tabela 5.19: Principais resultados dos modelos usados.

Modelos	EQM		Cor^2		Regras\Nós\ Coeficientes
	Trt.	Tst.	Trt.	Tst.	
GPFIS	0,3250	0,4058	0,0603	0,0495	14
WM-R	1,9872	1,9986	0,0215	0,0359	28
PGMG	0,1187	0,1283	0,1167	0,0837	173
AR	0,1152	0,1259	0,1429	0,1025	55
REG+SW	0,1210	0,1283	0,0996	0,0797	5

Portanto, apesar dos bons resultados exibidos pela AR, em termos de aceção final da solução esta se torna mais complexa. Ainda é possível observar

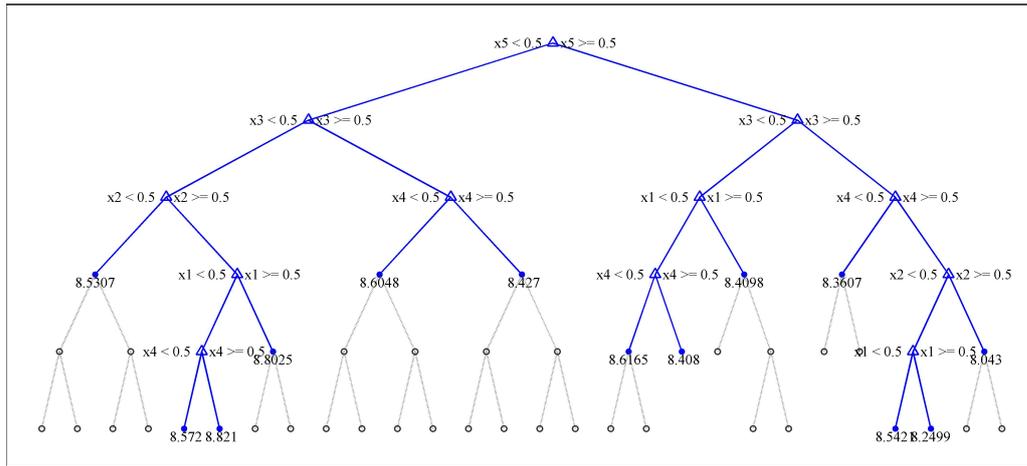


Figura 5.8: Exemplo de solução proposta pela AR. Os nós transparentes foram suprimidos para facilitar a visualização.

que as folhas da árvore exibem valores muito próximos da média do logaritmo do tempo de resposta (8,4582). Esse comportamento também é visualizado na comparação entre os valores preditos e observados, tanto pelo modelo GPFIS quanto pela AR denotados na Figura 5.9.

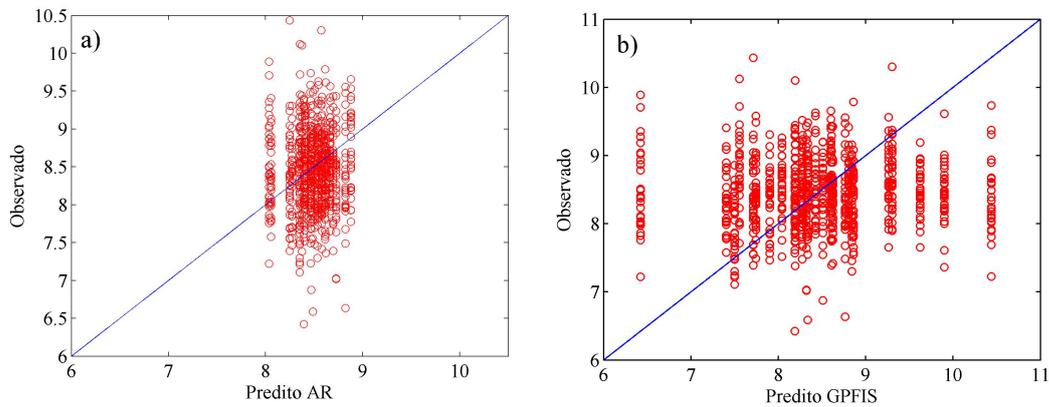


Figura 5.9: Relação entre os valores preditos e observados dos modelos AR (a) e GPFIS (b).

Verifica-se que a maior parte dos valores preditos pela AR situa-se na faixa entre 8 e 9, faixa esta que concentra a maior densidade de observações. Um dos motivos da AR ter obtido melhores métricas deve-se à sua característica de se ajustar a valores próximos à média. Contudo, o modelo GPFIS conseguiu captar valores além dessa faixa (em torno de 6-11), o que justifica o Cor^2 alto (metade do AR) e do EQM mais elevado ainda (3 vezes maior do que o AR). Uma possível forma de aprimorar os resultados é contradizer o saber do especialista e usar, ao invés de somente duas funções de pertinência, algumas

outras. A Tabela 5.20 apresenta os resultados do do WM-R e do GPFIS para 2, 3 e 5 funções de pertinência distribuídas de maneira uniforme e normalizada no universo de discurso do tempo de resposta. Observa-se a melhora significativa em termos de EQM, porém o Cor^2 piora significativamente.

Tabela 5.20: Variações no número de funções de pertinência e seu impacto nos Sistemas Fuzzy usados.

Modelos	EQM		Cor ²		Regras
	Trt.	Tst.	Trt.	Tst.	
GPFIS(2)	0,3250	0,4058	0,0603	0,0495	14
GPFIS(3)	0,1349	0,1349	0,0075	0,0195	5
GPFIS(5)	0,1349	0,1349	0,0072	0,0200	6
WM-R(2)	1,9872	1,9986	0,0215	0,0359	28
WM-R(3)	0,3319	0,3603	0,0062	0,0010	28
WM-R(5)	0,6764	0,5558	0,0297	0,0363	28

Uma das possíveis explicações para esta piora com o aumento do número de funções de pertinência é apresentada na Figura 5.10. Esta revela o grau de pertinência observado de cada tempo de resposta e sua distribuição com respeito às funções de pertinência. Como grande parte das observações está próximo da média, é possível observar que a parte preponderante dos graus de pertinência situa-se na faixa de valores de tempo de resposta entre 8 e 9. No caso de duas funções de pertinência, estas ficam quase identicamente distribuídas entre os conjuntos fuzzy R e D. Entretanto, ao se considerarem 3 funções de pertinência, a concentração situa-se no conjunto fuzzy M, o que, portanto, torna intensiva a busca por soluções para esse consequente específico. Porém, o que se observa de forma geral é a busca do modelo GPFIS por uma solução similar à AR: valores próximos à média (senão à média em si em alguns momentos) para minimizar o EQM⁵.

Existem, em princípio, duas possibilidades para melhorar os resultados: (i) buscar uma disposição que mais bem distribua os graus de pertinência entre os conjuntos fuzzy, (ii) ou alterar a função de avaliação, escolhendo por exemplo o Cor^2 ao invés do EQM. Neste intuito foi avaliada a alteração da função de avaliação de EQM para Cor^2 no contexto dos modelos evolutivos (GPFIS e PGMG); os resultados são apresentados na Tabela 5.21.

De uma forma geral os resultados apresentaram uma melhora substancial, não em termos de EQM, mas em Cor^2 . O modelo GPFIS com 5 funções de pertinência exhibe os melhores resultados entre as abordagens evolutivas.

⁵De fato, a escolha pela média faz certo sentido, apesar de ingênua, pois caso se tivesse que escolher um valor que melhor represente a variável tempo de resposta (ou seu logaritmo), a opção pela média é a que minimizaria o EQM.

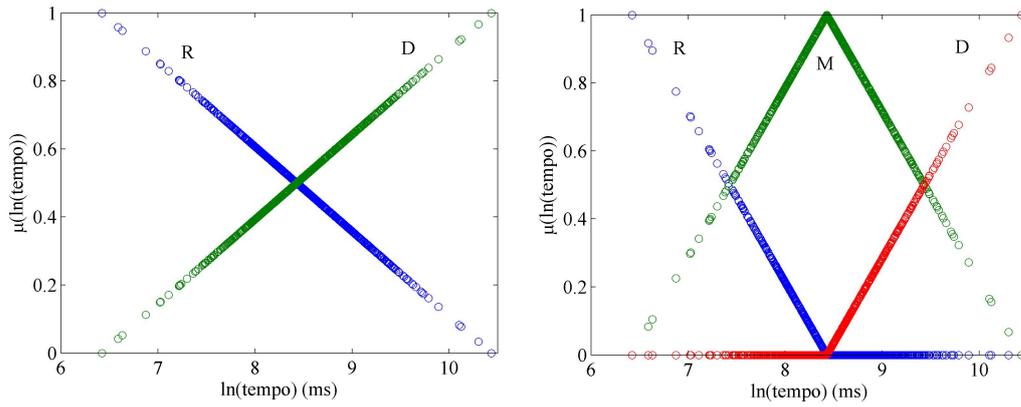


Figura 5.10: Grau de pertinência observados para dois e três conjuntos fuzzy.

Tabela 5.21: Alteração da função de avaliação e seus impactos nos modelos evolutivos.

Modelos	EQM		Cor ²		Regras\Nós
	Trt.	Tst.	Trt.	Tst.	
GPFIS(2)	0,7004	0,7466	0,1062	0,0807	14
GPFIS(3)	1,8372	1,8045	0,1054	0,0717	12
GPFIS(5)	0,3274	0,3178	0,1180	0,0935	15
PGMG	0,1187	0,1273	0,1171	0,0894	184

Porém, essa melhora ainda não é substancial ao ponto de prevalecer sobre a AR. Possivelmente, com a ampliação do número de avaliações (5000 efetuadas atualmente) seja possível obter melhores resultados. Com respeito à PGMG, a melhora não foi relativamente substancial em relação à abordagem anterior.

Por fim, a partir da concepção inicial das funções de pertinência estabelecidas pelos especialistas (duas, conforme Figura 5.10), a Tabela 5.22 apresenta a base de regras fuzzy do modelo GPFIS.

Tabela 5.22: Base de regras fuzzy.

Regra	Antecedente	Cons.	Peso	Tempo Médio
R1	TI é CC e NI é 1 e COMP é Mais	R	0,45	3662ms
R2	NI é 2 e COMP é Mais	R	0,28	4568ms
R3	COMP é Mais e COR é IC	R	0,27	4656ms
R1	COR é IG e COMP é Menos	D	0,34	5580ms
R2	TI é DC	D	0,33	4915ms
R3	NI é 2 e COMP é Menos	D	0,33	5326ms

Legenda: Cons. - Consequente, TI - Tipo de Informação, NI - Número de Itens, COMP - Comparativo e COR - Correção.

Em geral, a base de regras fuzzy elaborada cumpre com as previsões ensejadas pelo corpo de especialista no início do estudo. As regras possuem duas características que para os especialistas são imprescindíveis: (i) análise

dos fatores principais (isolados) e (ii) interação entre os níveis dos fatores. Um exemplo da (i) é a regra: Se Tipo de Informação é Desconhecida, então o Tempo é Devagar. Quando se avalia o tempo médio dos indivíduos nessa condição, verifica-se que eles levam em média 4915ms para responder à questão. Com relação a (ii) há diversas; por exemplo: Se Tipo de Informação é Conhecido e Número de Itens é 1 e Comparativo é Mais, então o sujeito responde de forma rápida (em média 3662ms). A descoberta destas interações responde a algumas das previsões e essas podem ser incorporadas em um outro método, como Análise de Variância para comparar o tempo médio desta condição com as demais interações de modo a avaliar se esta é significativa.

5.2

Classificação

Em problemas de Classificação, o número de SFGs presentes na literatura mostra-se relativamente superior do que para as áreas de Regressão, Previsão e Controle. Os trabalhos mais recentes são:

- **Berlanga et al. [28]**: os autores propõe o modelo GP-COACH (Genetic Programming Based Learning of Compact and Accurate Fuzzy Rule-Based Classification System for High Dimensional Problems). Este SFG é orientado para a geração de um base de regras a partir da PG clássica, em um formato do tipo GCCL (Genetic Competitive-Cooperative Learning - a base de regras fuzzy é a melhor população ao longo do processo de sintetização da PG). Ademais, as suas principais características são:
 - Gera regras do tipo DNF (Disjunctive Normal Form)⁶, com termos linguísticos representados por funções de pertinência triangulares.
 - Por seguir a abordagem GCCL, são usadas duas funções de avaliação:
 - * Local: avalia a qualidade no nível da regra, usando métricas de Suporte e Confiança (Acurácia Fuzzy).
 - * Global: avalia de forma global a qualidade da população (base de regras). Leva em conta a acurácia e complexidade do conjunto de regras.
 - Emprega novos mecanismos para a avaliação, recombinação e seleção: promoção de diversidade para geração de regras (avaliação), geração de regras secundárias para cobrir padrões não classificados (recombinação) e competição entre os indivíduos da população anterior e da atual para permanecer na nova população (seleção). No total são usadas 24 bases de dados para classificação. Os resultados, em termos de acurácia e complexidade, são comparados com os de outros quatro SFGs presentes na literatura. Este mesmo modelo é usado por López et al. [142], porém com a inclusão de representação em pares e do método de sobreamostragem (oversampling) para solucionar problemas de alto desbalanceamento das classes.
- **Alcalá-Fdez et al. [9]**: O trabalho proposto em [9] apresenta uma nova abordagem para mineração de regras fuzzy em bases de dados de elevada dimensionalidade, denominado FARC-HD. Esta abordagem é definida por três etapas:

⁶Um exemplo deste tipo é: “Se X_1 é A_1 ou A_2 e X_2 é A_3 **ou** A_4 e ...”.

1. *Extração de regras fuzzy para classificação*: este processo emprega um algoritmo de busca em árvore (Figura 5.11), de modo a listar as regras mais frequentes a partir da métrica do Suporte, modificada para contemplar o desbalanceamento entre as classes. A partir desse processo são geradas regras candidatas para discriminação das classes.

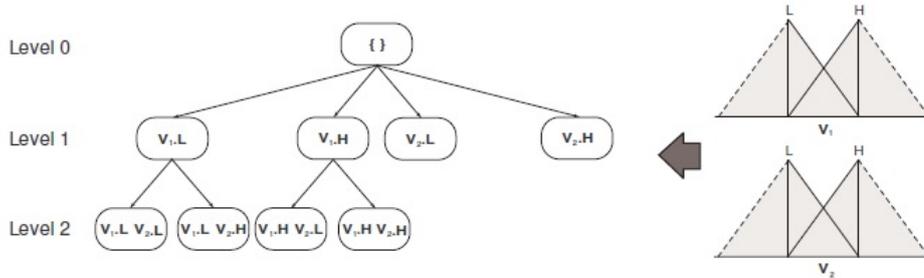


Figura 5.11: Exemplo de árvore empregada em Alcalá-Fdez et al. [9].

2. *Seleção de regras por qualidade de ajuste*: esta etapa se baseia na extração das regras mais pertinentes ao processo de classificação. Os autores usam uma métrica denominada $wWRAcc$ [9] de modo a avaliar a qualidade de cada regra candidata. A partir desta métrica as regras são elencadas com respeito ao nível em que estão na árvore e as melhores são incluídas na base de regras.
3. *Seleção por AG de regras e ajuste das funções de pertinência*: os autores aplicam o conceito de representação em pares para ajustar as funções de pertinência, além de efetuar a escolha de um subconjunto das regras previamente geradas, de maneira a maximizar um compromisso com respeito à acurácia de classificação e redução da complexidade da base de regras fuzzy.

A abordagem apresentada por Alcalá-Fdez et al. [9] descreve uma abordagem tipicamente de refinamento da base de regras fuzzy e das funções de pertinência. Os autores utilizam 26 bases de dados, comparando os resultados com os de 4 SFGs baseados em AGs e outros classificadores.

- **Muni e Pal [157]**: Os autores propõem um SFGBR do tipo Pittsburgh baseado em PGMG (no caso denominado de PG de Múltiplas Árvores). Cada indivíduo é composto por H árvores, onde H é o número de classes do problema. Cada árvore expressa um conjunto de regras específicas para a classe atrelada (um exemplo de solução é disposta na Figura 5.12). Os autores avaliam a geração da base de regras iniciais, mesclando

o processo aleatório com o de protótipos elaborados via algoritmo C-Means. As funções de pertinência usadas são gaussianas, além da inclusão de novos operadores de recombinação para sintetização da solução. É uma abordagem relativamente clássica, direcionada para a maximização da acurácia.

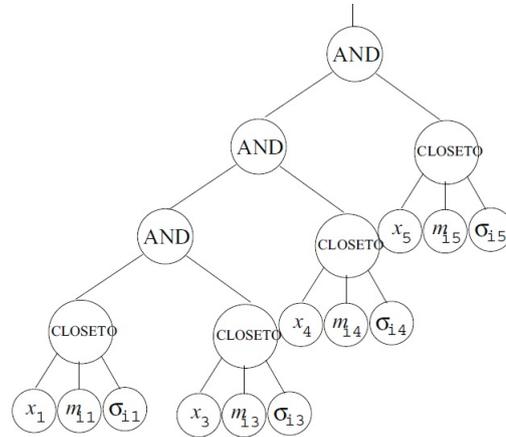


Figura 5.12: Exemplo de solução do modelo de Muni e Pal [157].

Para avaliar a proposta, os autores empregam cinco bases de dados e a comparação é feita com os resultados de Lim et al. [140]. Nenhuma informação é disponibilizada com relação ao número de regras geradas. Este trabalho também se confunde com o de Koshiyama et al. [131], porém este último usa uma decodificação de solução mais direta (usa operações matemáticas que se confundem com os operadores fuzzy), enquanto que em Muni e Pal [157] as regras são decodificadas literalmente. Contudo, ambas as abordagens devem ser vistas como um caso especial do modelo GPFIS para classificação, pois este permite que um indivíduo decodifique mais árvores do que as classes necessárias, fora o amplo uso de operadores fuzzy e agregação por diferentes métodos. Além disso, no caso do modelo GPFIS, a definição da classe que mais bem se ajusta ao antecedente da regra é realizada por heurísticas, enquanto que em Muni e Pal [157] e Koshiyama et al. [131] a sua busca é realizada pela PGMG (ou de Múltiplas Árvores).

- **Aydogan et al. [18]:** Propõem um algoritmo híbrido denominado hGA, baseado em AG e Programação Inteira. A codificação do AG é binária e a abordagem usada é a GCCL, em que a escolha da base de regras de cada geração é determinada a partir da solução de um problema de otimização combinatória. Os autores ainda usam um operador de mutação que exerce uma busca local, alterando diversas vezes os valores

binários, de modo a encontrar uma regra que maximize a acurácia produzida pela base de regras (coopere melhor com a população). Os autores selecionam 17 bases de dados presentes em Berlanga et al. [28] e seguem fielmente a formulação das funções de avaliação local e global, procedimento experimental e análises estatísticas. Apesar dos resultados superiores com relação aos de Berlanga et al. [28], os autores não relatam se o número de avaliações empreendidas pela otimização combinatória e o mecanismo de busca local é abatido do máximo a ser empreendido (20000 avaliações por base de dados). Ainda, questões relativas a complexidade da base de regras não são apresentadas no texto.

- **Sanz et al. [187]**: Os autores propõem o IVTURS (Interval-Valued fuzzy reasoning method with TUning and Rule Selection). A ideia central do IVTURS reside na alteração do processo de inferência fuzzy para classificação, de modo que este admita um procedimento de raciocínio fuzzy para funções de pertinência intervalares. Além disto, neste modelo é incluso a possibilidade de modificadores linguísticos para alterar a parametrização das funções de pertinência triangulares. Com este fim, o cromossoma do AG possui duas partes: uma representa os atributos em consideração para a formação da base de regras, enquanto que a outra indica quais regras pertencem à base de regras. Para a seleção das regras é necessário que uma base seja pré-concebida. Para tanto, o método emprega o algoritmo FARC-HD [9] para a geração da base de regras. O IVTURS é voltado somente para maximizar a acurácia de classificação. São usadas 27 bases de dados para avaliar a qualidade da abordagem e os resultados são comparados aos de dois outros métodos de classificação. Nenhuma informação é fornecida quanto à complexidade da base de regras ou quanto ao formato das funções de pertinência no final do processo.

Considera-se que o trabalho de Berlanga et al. [28] é o mais adequado para fins de comparação devido ao grande número de bases de dados e SFGs empregados, além de ser baseado em PG. Ainda, este trabalho torna possível a comparação tanto em termos de acurácia, quanto de compacidade da base de regras gerada. A partir de 22⁷ das 24 bases de dados usadas em Berlanga et al. [28], o próximo tópico efetua uma investigação empírica da melhor arquitetura para o modelo GPFIS.

⁷Não foi possível usar duas das bases devido à dificuldade de encontrá-las nos sítios indicados.

5.2.1

Investigação Empírica da Arquitetura do Modelo GPFIS

Descrição dos Experimentos

As principais informações das 22 bases de dados consideradas, conforme Berlanga et al. [28], são dispostas na Tabela 5.23.

Tabela 5.23: Bases de dados consideradas nos experimentos efetuados.

Base	# Padrões	# Atributos	# Classes
Bupa	345	6	2
Cleveland	297	13	5
Ecoli	336	7	8
Glass	214	9	6
HillValley1	1212	100	2
HillValley2	1212	100	2
Iris	150	4	3
Libras	360	90	15
Magic	19020	10	2
Page-blocks	5472	10	5
Parkinsons	195	22	2
Pen-based	10992	16	10
Pima	768	8	2
Ringnorm	7400	20	2
Satimage	6435	36	6
Segment	2310	19	7
Sonar	208	60	2
Spambase	4597	57	2
Twonorm	7400	20	2
Wdbc	569	30	2
Wine	178	13	3
Yeast	1484	8	10

Para todas as bases de dados, as rotinas do modelo GPFIS seguiram as linhas de Berlanga et al. [28]:

- Validação cruzada de 10 pastas (10-fold-cv).
- 3 execuções para cada pasta, totalizando 30 execuções.
- Cada execução com um total de 20000 avaliações.

A partir desta descrição, a Tabela 5.24 exhibe o delineamento de um experimento sequencial, de modo a identificar a configuração mais apropriada em termos de acurácia para o modelo GPFIS.

De forma similar à seção do modelo GPFIS para Regressão, o Experimento 1 é voltado para investigar a influência das diferentes configurações do conjunto de Operadores Fuzzy. A hipótese central é avaliar o quanto de

Tabela 5.24: Delineamento experimental sequencial para o modelo GPFIS.

Descrição	Experimento 1	Experimento 2	Experimento 3
Hipótese Avaliada	Operadores Fuzzy	Operadores de Agregação	Métodos de Particionamento
Configuração Básica	Agregação por Máximo + Divisão Uniforme	Melhor configuração anterior + Divisão Uniforme	Melhor configuração anterior
Configurações Avaliadas	Operadores Fuzzy testados: - Produto (Prod) - Produto + Raiz Quadrada (Prod+RQ) - Produto + Negação (Prod+Neg) - Produto + Negação + Raiz Quadrada (Prod+RQ+Neg)	Operadores de Agregação testados: - Máximo - Média Aritmética - Produto - Mínimos Quadrados Restrito (MQR)	Métodos de Particionamento: - Divisão Uniforme (Div-Unif) - Grau de Confiança (CD) - Máxima Credibilidade (Max-Cred)

acurácia é proporcionada pela adição do operador de negação, de dilatação (raiz quadrada) e se há interação relevante entre estes. A t-norma escolhida é o produto, devido a sua capacidade de realizar a operação de contração quando há o encontro de duas funções de pertinência idênticas.

A partir da melhor configuração anterior, o Experimento 2 examina o tipo de Operador de Agregação mais eficiente. Por fim, o Experimento 3 proporciona a possibilidade do exame de outros Métodos de Particionamento, analisando dois tipos: Grau de Confiança (CD) e Máxima Credibilidade (Max-Cred), em comparação á abordagem ingênua da Divisão Uniforme (Div-Unif).

A Tabela 5.25 apresenta as demais configurações do modelo GPFIS, e a Figura 5.13 mostra o formato e número das funções de pertinência usadas, sendo estas mantidas intactas ao longo de todo o processo de sintetização do modelo. As métricas relatadas (acurácia, número de regras, etc.) para cada base de dados são frutos de uma média dos 30 modelos (3 para cada uma das 10 pastas). Por fim, o nível de significância adotado para a investigação empírica foi de 5%.

O método de Decisão escolhido foi o da classe mais pertinente (descrito no Capítulo 4, seção Defuzzificação/Decisão). A função de avaliação para problemas de classificação, sejam binários ou de múltiplas classes, é dada pelo Erro Médio de Classificação (*EMC*) como:

$$EMC = \frac{\sum_{i=1}^n |C_{i \in k}(\mathbf{x}_i) - \hat{C}_{i \in k}(\mathbf{x}_i)|}{n} \quad (5-2)$$

Tabela 5.25: Principais configurações do modelo GPFIS.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	200
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.13
Operadores Fuzzy	Tabela 5.24

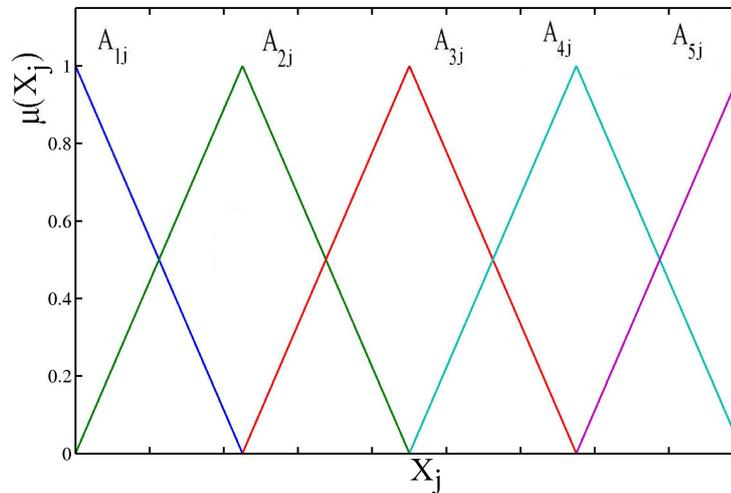


Figura 5.13: Funções de pertinência usadas no modelo GPFIS.

onde para dado um padrão \mathbf{x}_i , $|C_{i \in k}(\mathbf{x}_i) - \hat{C}_{i \in k}(\mathbf{x}_i)| = 0$, se $C_{i \in k}(\mathbf{x}_i) = \hat{C}_{i \in k}(\mathbf{x}_i)$ e 1, caso contrário. O indivíduo que minimizar o EMC é considerado o melhor na população. O próximo tópico exibe os resultados e análises efetuadas para cada experimento.

Resultados e Discussões

A Tabela 5.26 apresenta a acurácia (%) obtida para cada configuração descrita no Experimento 1. É possível verificar que em torno de 43% das bases de dados a configuração Produto+Raiz-Quadrada+Negação (Prod+RQ+Neg) obteve os melhores resultados, seguido de Produto+Negação em 30% dos casos. Este resultado indica que o uso de outros operadores fuzzy fora do âmbito das t-normas possibilitam ganhos de acurácia em problemas de classificação.

Tabela 5.26: Resultados em termos de acurácia (%) do Experimento 1 para o conjunto de operadores fuzzy.

Base	Prod		Prod+RQ		Prod +Neg		Prod+RQ+Neg	
	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.
Bupa	36,04	21,74	47,65	34,30	56,52	50,72	51,47	41,06
Cleveland	57,79	51,67	58,91	52,22	61,16	55,00	61,72	52,78
Ecoli	68,74	65,49	72,01	66,76	71,59	67,75	70,19	65,00
Glass	58,58	48,52	61,36	51,48	63,77	54,04	64,69	60,16
HillValley1	52,17	51,82	53,13	52,89	52,90	51,07	54,11	53,22
HillValley2	52,17	51,87	53,80	51,57	52,88	52,15	54,42	52,29
Iris	96,47	96,22	96,30	96,44	96,64	94,67	96,72	96,00
Libras	36,12	27,89	36,42	24,00	32,67	24,33	30,30	24,67
Magic	31,10	31,13	34,73	34,73	35,15	35,15	35,16	35,14
Page-blocks	92,88	92,16	94,07	93,56	93,48	92,77	94,60	94,21
Parkinsons	64,15	70,89	79,89	81,11	86,67	86,89	86,74	83,78
Pen-based	49,39	49,16	53,11	52,92	50,21	50,49	53,14	53,02
Pima	70,00	69,74	74,63	74,15	75,46	72,74	75,22	74,15
Ringnorm	70,18	69,35	87,93	87,50	87,05	86,86	90,60	90,20
Satimage	74,46	73,87	75,97	75,56	77,78	77,38	77,36	77,05
Segment	74,16	73,51	75,87	75,02	74,25	73,87	77,05	75,95
Sonar	64,95	60,50	73,03	70,33	77,59	64,83	77,78	67,83
Spambase	71,95	71,67	80,59	80,53	73,14	73,03	81,82	81,65
Twonorm	65,38	65,45	72,39	73,15	79,45	78,72	86,84	86,17
Wdbc	83,05	82,38	91,43	91,01	93,70	92,92	93,43	92,08
Wine	87,31	81,30	90,80	90,00	91,15	83,33	92,06	87,59
Yeast	47,69	47,67	50,51	49,17	50,08	49,13	49,24	48,17
Média	63,85	61,54	68,84	66,29	69,70	66,72	70,67	67,82

A Tabela 5.27 apresenta os resultados no tocante ao teste de Friedman e Holm a partir das acurácias na fase de teste de cada configuração.

Tabela 5.27: Resultados do teste de Friedman e Holm para o Experimento 1.

i	Método	Posto	
3	Prod	3,4706	
2	Prod+RQ	3,1471	
1	Prod+Neg	2,0000	
0	Prod+RQ+Neg	1,3824	
Teste	p-valor		
Friedman	< 0,0001		
Iman e Daveport	< 0,0001		
Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
Prod	4,7158	<0,0001	0,0167
Prod+RQ	3,9852	<0,0001	0,025
Prod+Neg	1,3948	0,1631	0,05

Como apontado anteriormente e atestado pelo teste de Friedman, a configuração Prod+RQ+Neg auferiu o menor posto (1,3824), seguida da abordagem Prod+Neg com posto igual a 2,0000. Como a Prod+RQ+Neg tem menor posto e o resultado do teste de Friedman aponta que há diferença

significativa com respeito à acurácia obtida por cada configuração (p-valor < 0,05), torna-se necessário aplicar o teste de Holm. Este denota que a configuração Prod+RQ+Neg é substancialmente superior a Prod e Prod+RQ, além de atestar a ausência de diferença significativa entre a Prod+Neg. Portanto, é possível inferir que grande parte da melhora é atribuída ao operador de negação, sendo o papel do modificador linguístico incrementar de forma marginal a acurácia final do modelo. A partir da configuração Prod+RQ+Neg, a Tabela 5.28 apresenta os resultados de acurácia (%) com respeito ao Experimento 2.

Tabela 5.28: Resultados em termos de acurácia (%) do Experimento 2 para o conjunto de operadores de agregação.

Base	Máximo		Média Aritmética		Produto		MQR	
	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.
Bupa	76,57	56,52	77,79	71,98	74,74	62,32	76,19	70,12
Cleveland	62,16	53,22	64,88	56,89	61,90	55,67	67,74	57,43
Ecoli	65,98	62,84	74,14	68,43	77,30	71,67	82,33	75,00
Glass	61,24	53,65	67,31	55,82	62,39	54,04	71,71	62,23
HillValley1	54,06	51,90	71,46	69,78	53,30	51,32	76,31	74,27
HillValley2	55,40	52,15	56,26	52,95	54,38	51,87	62,57	59,53
Iris	96,35	94,44	98,42	95,11	97,33	94,89	98,69	94,89
Libras	24,39	19,00	29,56	20,00	27,79	20,67	53,48	35,37
Magic	81,87	81,62	81,64	81,55	77,02	76,64	82,28	82,33
Page-blocks	92,25	91,19	95,21	94,79	81,24	79,96	94,96	94,51
Parkinsons	89,30	84,44	91,30	84,67	86,59	81,56	92,25	79,15
Pen-based	40,24	40,05	47,75	47,58	43,23	42,98	77,62	76,94
Pima	79,68	73,29	79,00	74,36	75,61	73,80	79,71	74,60
Ringnorm	85,33	84,84	91,77	91,05	91,05	90,82	88,47	88,01
Satimage	69,11	68,74	75,23	74,91	71,28	71,00	82,48	81,55
Segment	61,62	61,56	68,86	68,44	61,53	61,31	85,05	84,60
Sonar	84,79	64,50	84,75	65,50	78,39	60,50	86,45	72,86
Spambase	88,46	87,91	87,30	86,88	82,41	81,92	88,37	87,75
Twonorm	86,18	84,81	90,41	89,75	86,15	85,65	93,89	93,43
Wdbc	96,78	93,51	97,49	95,30	93,65	92,32	97,58	95,26
Wine	92,15	87,04	96,13	88,15	90,67	86,67	99,19	92,47
Yeast	46,21	45,53	48,35	46,76	45,65	45,46	56,56	53,42
Média	72,28	67,85	76,14	71,85	71,53	67,86	81,54	76,62

Observa-se que em 69,56% dos casos o operador de agregação MQR auferiu os melhores resultados, seguido da média aritmética (26,08% do total). Ainda, a acurácia média geral da agregação por MQR foi em torno de 5% maior do que a por Média Aritmética e 10% superior do que pelo procedimento do Máximo (abordagem padrão). A Tabela 5.29 apresenta os resultados do teste de Friedman e Holm para o Experimento 2. É possível verificar que o uso do MQR proporciona resultados significativamente superiores aos dos demais operadores de agregação (p-valor < 0,05) e que o uso da média aritmética resulta em um posto menor do que o obtido com o uso do máximo. Novamente,

a agregação pelo método do Produto não proporcionou bons resultados.

Tabela 5.29: Resultados do teste de Friedman e Holm para o Experimento 2.

i	Método	Posto	
3	Produto	3,6765	
2	Máximo	3,0882	
1	Média Aritmética	2,0588	
0	MQR	1,1765	
Teste	p-valor		
Friedman	< 0,0001		
Iman e Daveport	< 0,0001		
Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
Produto	5.6457	<0,0001	0,0167
Máximo	4.3173	<0,0001	0,025
Média Aritmética	1.9926	0,0463	0,05

Os resultados ensejam duas observações: (i) a definição de operadores de agregação diferentes do tradicional Máximo, como a Média Aritmética e o MQR, possibilitam auferir bons resultados; (ii) fornecer pesos ótimos a cada regra e assim selecionar as que devem compor a base de regras final (média ponderada), ao invés de responsabilizar todas igualmente (média aritmética), é relevante para obtenção de bons resultados. Contudo, a abordagem por MQR demanda um custo computacional superior em 2 a 3 vezes em problemas de classificação binária (BIN), e de 6 a 7 nos de múltiplas classes (MULT), quando comparada às abordagens que fazem uso dos demais operadores de agregação (Tabela 5.30). Tal fato deve ser atentado pelo usuário no momento da escolha deste tipo de operador.

Tabela 5.30: Proporção adicional de tempo tomado pela abordagem MQR com relação as demais operações de agregação.

Métrica \ Abordagem	Máximo	Média	Produto
Média (BIN)	102,1 %	84,7 %	135,2 %
Desvio Padrão (BIN)	8,4 %	6,1 %	6,7 %
Média (MULT)	590,1 %	472,4 %	726,6 %
Desvio Padrão (MULT)	87,8 %	92,6 %	77,8 %

A Tabela 5.31 apresenta o número médio de regras obtido pelo uso de cada operador de agregação durante o Experimento 2. Verifica-se que as abordagens que geraram o menor número de regras foram Produto e por MQR⁸. Um dos motivos do Produto ter gerado poucas regras deve-se

⁸Uma regra foi considerado como ativa quando o peso associado a ela foi maior do que 0,01.

à característica semântica das regras geradas (conectivo “e”), o que torna um conjunto de regras atrelados a um mesmo conseqüente, teoricamente, em uma única regra. Por fim, a agregação pelo MQR garante bons resultados de acurácia e redução de complexidade da base de regras fuzzy.

Tabela 5.31: Regras geradas por cada abordagem no Experimento 2 para as bases de dados classificação binária (BIN) e de múltiplas classes (MULT).

Métrica \ Abordagem	Máximo	Média Aritmética	Produto	MQR
Média (BIN)	22,17	28,11	8,22	11,32
Desvio Padrão (BIN)	7,43	9,10	6,78	3,45
Média (MULT)	32,65	45,82	18,24	26,66
Desvio Padrão (MULT)	26,66	32,17	25,64	23,33

De modo a avaliar os diferentes Métodos de Particionamento (Experimento 3), a Tabela 5.32 apresenta os resultados de acurácia (%) para cada base de dados.

Tabela 5.32: Resultados em termos de acurácia (%) do Experimento 3 para o conjunto de métodos de particionamento.

Base	Div-Unif		CD		Max-Cred	
	Tra.	Tst.	Tra.	Tst.	Tra.	Tst.
Bupa	76,19	70,12	74,09	68,39	75,14	69,25
Cleveland	67,74	57,43	70,95	58,22	69,43	57,54
Ecoli	82,33	75,00	84,69	78,53	83,71	76,57
Glass	71,71	62,23	74,65	67,58	72,01	61,05
HillValley1	76,31	74,27	67,03	64,85	71,67	69,56
HillValley2	62,57	59,53	58,08	55,51	60,33	57,52
Iris	98,69	94,89	98,59	95,56	99,06	95,11
Libras	53,48	35,37	69,04	50,00	57,71	45,27
Magic	82,28	82,33	81,30	81,28	81,79	81,81
Page-blocks	94,96	94,51	94,67	94,55	95,12	94,77
Parkinsons	92,25	79,15	92,24	84,34	92,25	81,75
Pen-based	77,62	76,94	83,38	82,65	76,94	76,42
Pima	79,71	74,60	78,57	75,24	79,14	74,92
Ringnorm	88,47	88,01	90,74	90,19	89,61	89,10
Satimage	82,48	81,55	83,52	82,65	82,78	81,96
Segment	85,05	84,60	89,34	88,36	87,21	87,19
Sonar	86,45	72,86	84,91	74,29	85,68	73,57
Spambase	88,37	87,75	89,07	88,57	88,72	88,16
Twonorm	93,89	93,43	94,99	94,49	94,44	93,96
Wdbc	97,58	95,26	97,26	95,67	97,42	95,47
Wine	99,19	92,47	99,69	95,63	99,67	93,42
Yeast	56,56	53,42	59,02	55,35	57,09	54,92
Média	81,54	76,62	82,54	78,27	81,68	77,24

Observa-se que as média gerais das acurácias de cada abordagem são próximas, diferindo no máximo em torno de 2%. Porém, é possível avaliar que: (i) o método da Máxima Credibilidade provê resultados frequentemente

melhores do que aqueles proporcionados pela Divisão Uniforme (73,91% dos casos); (ii) o método da Máxima Confiança proporciona resultados de acurácia melhores do que aqueles proporcionados pela Divisão Uniforme e pela Máxima Credibilidade (82,60% e 78,26% dos casos, respectivamente).

A Tabela 5.33 exhibe os resultados do teste de Friedman e Holm para o Experimento 3. É possível verificar que o método CD (Máxima Confiança) obteve o menor posto (1,4091). Logo, este é selecionado para ser o método de controle, na comparação aos pares. Observa-se que nas comparações, tanto o método da Máxima Credibilidade quanto o da Divisão Uniforme são significativamente inferiores em termos de acurácia ao CD (p -valor $< 0,05$).

Tabela 5.33: Resultados do teste de Friedman e Holm para o Experimento 3.

i	Método	Posto	
2	Div-Unif	2,5455	
1	Max-Cred.	2,0455	
0	Max-Conf.	1,4091	
Teste	p-valor		
Friedman	$< 0,0001$		
Iman e Daveport	$< 0,0001$		
Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
Div-Unif	3,7689	$< 0,0001$	0,025
Max-Cred.	2,1106	0,0348	0,05

Novamente, esta maior acurácia deve-se a um maior investimento computacional deste método. A Tabela 5.34 apresenta a proporção adicional de tempo tomada pelo método do CD e Máxima Credibilidade com respeito ao da Divisão Uniforme. Verifica-se que nos problemas de classificação binária o tempo computacional dobra, enquanto que nos de múltiplas classes triplica proporcionalmente.

Tabela 5.34: Proporção adicional de tempo tomado pelas demais abordagens com respeito ao método de Div-Unif.

Métrica \ Abordagem	CD	Max-Cred.
Média (BIN)	92,1 %	84,7 %
σ (BIN)	8,4 %	6,1 %
Média (MULT)	190,1 %	152,4 %
σ (MULT)	87,8 %	87,6 %

Este maior investimento computacional também se reflete no número de regras geradas. A Tabela 5.35 apresenta o número médio de regras geradas para cada método de particionamento. Na comparação entre Divisão Uniforme e CD, verifica-se uma redução de 30% em média no tamanho da base de regras da

última. Um dos motivos para esta redução deve-se ao refinamento do processo de definição do melhor consequente para uma parte antecedente, o que auxilia o modelo na geração de regras relevantes para o processo de classificação, aproveitando ao máximo o esforço da meta-heurística.

Tabela 5.35: Regras geradas por cada abordagem no Experimento 3 para as bases de dados classificação binária (BIN) e de múltiplas classes (MULT).

Métrica \ Abordagem	Div-Unif	CD	Max-Cred
Média (BIN)	11,32	7,79	13,37
σ (BIN)	3,45	4,58	2,35
Média (MULT)	26,66	23,14	32,42
σ (MULT)	23,33	18,31	22,12

Portanto, a melhor configuração obtida para o modelo GPFIS é:

- **Operadores Fuzzy:** Produto, Raiz Quadrada e Negação.
- **Operador de Agregação:** Mínimos Quadrados Restrito (MQR).
- **Método de Particionamento:** CD.

A partir dessa configuração, o próximo tópico efetua a comparação do modelo GPFIS com o SFG proposto por Berlanga et al. [28] e os demais presentes neste trabalho.

5.2.2

Comparação com Demais Sistemas Fuzzy-Genéticos

Descrição dos Experimentos e Proposta de Solução

Para efetuar a comparação do modelo GPFIS com os demais SFGs, utilizou-se a melhor configuração obtida na investigação anterior, além do procedimento experimental (10-fold-cv e 3 execuções para cada pasta), parâmetros da Tabela 5.25 e funções de pertinência conforme Figura 5.13. A descrição do modelo GP-COACH [28] foi efetuada na introdução do Capítulo. Como este trabalho contempla uma comparação com outros 4 SFGs, estes são brevemente descritos a seguir:

- **2SLAVE:** os autores González e Pérez [89] propuseram o 2SLAVE, um modelo baseado em um AG para aprender uma base de regras fuzzy. O 2SLAVE decodifica em cada indivíduo do AG uma regra, e promove o processo de sintetização conforme os conceitos do tipo IRL. Ainda, esse embarca no processo evolutivo um método de seleção de entradas.

- GP-PITT: em Tsakonas [201] desenvolve um SFGBR que utiliza uma PG orientada à gramática, que usa a concepção Pittsburgh para decodificar em um indivíduo toda a base de regras fuzzy do sistema.
- GCCL-Ishi: em Ishibuchi et al. [112] é proposto um SFGBR que segue uma formulação GCCL, que, a partir de um AG de tamanho fixo, busca somente a parte antecedente da regra. A classe é determinada pela seleção da mais compatível com o termo antecedente, enquanto que o peso da regra é dada por esse grau de compatibilidade auferido a partir dos padrões na base de dados (usando a equação do CD - confiança ou acurácia fuzzy).
- FRBCS: os autores Berlanga et al. [29] apresentam um sistema bastante similar ao GP-COACH, mas, por ser anterior, não possui algumas características em termos de operadores genéticos e de dinâmica populacional.

A partir dos resultados de cada modelo [28] e do procedimento experimental exposto, o próximo tópico apresenta os resultados e discussões da comparação efetuada destes modelos com o GPFIS.

Resultados e Discussões

A Tabela 5.36 exibe a acurácia obtida por cada modelo na fase de teste, considerando as 22 bases de dados em análise.

Em 86,36% das bases de dados (19 em 22) o modelo GPFIS obteve os melhores resultados dentre todos os SFGs, sendo somente inferior nas bases Ringnorm e Parkinsons (classes binárias) e Iris (múltiplas classes). A acurácia média geral do GPFIS foi em torno de 4% superior à do GP-COACH, com um desvio padrão por volta de 1% inferior.

A Tabela 5.37 expõe os resultados do teste de Friedman e Holm baseado na acurácia de cada SFG. É possível verificar que o posto médio do GPFIS foi o menor de todos (1,3636), enquanto que o do GP-COACH foi o segundo menor (2,5455) – embora quase o dobro. Como o posto médio do modelo GPFIS é menor, este foi selecionado para ser o método de controle para comparação com demais SFGs. O teste de Holm para múltiplas comparações evidencia que o modelo GPFIS obteve, dentre as bases de dados analisadas, acurácia significativamente superior às do demais SFGs (p -valor $< 0,05$).

Dado que o modelo GPFIS obteve bons resultados em termos de acurácia, uma questão que a considerar é se isto foi fruto da ampliação da base de regras. A Tabela 5.38 exibe o número médio de regras geradas para cada base de dados, assim como o número médio de antecedentes contidos em cada. Em 18 das 22

Tabela 5.36: Resultados em termos de acurácia (%) na fase de teste para todos os modelos comparados.

Base	2SLAVE	GP-PITT	GCCL-Ishi	FRBCS	GP-COACH	GPFIS
Bupa	58,58	56,45	58,27	62,20	63,63	68,39
Cleveland	46,19	56,46	54,15	56,69	55,23	58,22
Ecoli	57,49	43,94	71,17	76,75	77,72	78,53
Glass	44,39	45,12	60,69	56,61	65,33	67,58
HillValley1	51,76	49,97	20,02	49,78	52,89	64,85
HillValley2	51,21	49,2	28,00	50,69	53,99	54,29
Iris	94,67	48,44	94,67	97,11	97,56	95,56
Libras	25,83	5,28	20,74	47,69	45,56	50,00
Magic	74,29	64,79	76,02	74,51	79,82	81,28
Page-blocks	91,42	92,92	90,34	91,09	91,23	94,55
Parkinsons	81,75	74,53	83,27	85,75	86,48	83,19
Pen-based	81,16	44,67	82,18	75,53	82,20	82,65
Pima	66,45	64,28	69,11	73,16	74,37	75,24
Ringnorm	79,64	50,51	91,70	93,84	91,13	90,19
Satimage	33,45	23,82	63,12	68,06	72,50	82,65
Segment	72,81	21,62	84,07	80,38	85,96	88,36
Sonar	70,72	52,42	72,40	71,15	67,48	74,29
Spambase	70,14	81,89	69,87	74,55	82,80	88,57
Twonorm	84,35	48,80	90,12	91,97	84,83	94,49
Wdbc	91,80	63,09	91,09	95,02	93,90	95,44
Wine	91,53	38,19	91,21	91,13	95,10	95,63
Yeast	14,51	31,76	49,01	52,16	48,56	55,35
Média	65,19	50,37	68,69	73,45	74,92	78,27
σ	22,53	19,76	22,86	16,25	16,12	14,68

Tabela 5.37: Resultados do teste de Friedman e Holm para a comparação entre os SFGs.

i	Método	Posto
5	GP-PITT	5,3182
4	2SLAVE	4,5227
3	GCCL-Ishi	4,0682
2	FRBCS	3,1818
1	GP-COACH	2,5455
0	GPFIS	1,3636

Teste	p-valor
Friedman	< 0,0001
Iman e Daveport	< 0,0001

Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
GP-PITT	7,0107	<0,0001	0,01
2SLAVE	5,6005	<0,0001	0,0125
GCCL-Ishi	4,7946	<0,0001	0,0167
FRBCS	3,2232	0,0013	0,025
GP-COACH	2,0951	0,0362	0,050

bases de dados o GPFIS gerou menos regras do que o modelo GP-COACH, citado no estudo de Berlanga et al. [28] como elaborado para gerar bases de

regras compactas. Ainda, em 21 das 22 bases o número médio de antecedentes por regra também foi menor, demonstrando que o GP-COACH, apesar de usar regras do tipo DNF, não conseguiu gerar bases de regras mais compactas do que o GPFIS.

A maior compacidade do modelo GPFIS deve-se a uma conjunção de fatores: o método da pressão lexicográfica (Capítulo 4 - Avaliação) e uso do MQR e CD, que possibilitam excluir regras inativas, com baixo poder preditivo e pouco identificadas com a classe consequente. Com respeito aos poucos antecedentes por regra, isso talvez se deva ao poder de sumarização que o operador de negação exerce sobre o processo de formação das regras, conforme discutido previamente.

Tabela 5.38: Comparação entre número médio de regras e antecedentes.

Base	GPFIS		GP-COACH	
	# Regras	# Antecedentes	# Regras	# Antecedentes
Bupa	4,27	2,15	10,07	2,49
Cleveland	19,73	2,16	23,83	7,44
Ecoli	17,60	2,19	25,57	6,21
Glass	14,10	2,27	17,43	5,56
HillValley1	3,77	3,40	7,27	7,65
HillValley2	7,93	2,05	6,90	8,16
Iris	5,60	1,86	3,23	1,75
Libras	64,00	3,04	113,93	76,03
Magic	5,10	3,01	9,33	4,33
Page-blocks	8,67	2,65	14,97	3,51
Parkinsons	4,77	1,87	6,40	3,77
Pen-based	53,50	2,48	89,70	9,35
Pima	5,80	2,09	17,23	5,15
Ringnorm	13,47	2,61	17,50	9,90
Satimage	21,17	2,62	27,53	13,29
Segment	18,10	2,50	23,30	6,85
Sonar	5,87	2,21	14,03	6,35
Spambase	8,40	1,78	10,27	7,48
Twonorm	18,07	1,77	51,67	9,15
Wdbc	6,33	1,76	4,90	3,03
Wine	10,03	1,50	7,57	4,65
Yeast	18,97	2,48	32,20	6,44
Média	15,24	2,29	24,31	9,48

Conclui-se que toda a elaboração do modelo, com a inclusão de outros operadores (negação, agregação, etc.) e a definição da melhor classe consequente, dado as métricas estabelecidas, foram relevantes para atingir este nível de acurácia e baixa complexidade na base de regras. Basta ter em conta que alguns SFGs comparados partilham de algumas características, como ser do tipo Pittsburgh, baseado em PG, efetuar seleção de consequentes, entre outros.

Contudo, apesar destas características exercerem sua parcela de contribuição, exclusivamente estes fatores não são sinônimos de elevada acurácia e baixa complexidade.

A próxima seção efetua uma análise detalhada do modelo GPFIS, apresentando uma aplicação real, análise da base de regras fuzzy e comparação com demais classificadores.

5.2.3

Aplicação Detalhada

Descrição do Problema

Uma das tarefas mais importantes na indústria de exploração e produção de óleo e gás é a identificação litológica. Litologia é a descrição das características físicas macroscópicas de uma rocha tais como cor, textura, tamanho do grão e conteúdo mineral [189, 206]. Com base nessa descrição e conhecendo-se a localização de cada tipo de rocha no poço, é possível inferir onde se encontram as formações geradoras de contenção do hidrocarboneto e do reservatório, elementos necessários para a ocorrência de um sistema petrolífero.

Tradicionalmente, técnicas estatísticas são utilizadas na identificação litológica através do estudo dos perfis. Uma das mais utilizadas é a Análise Discriminante [34]. Mais recentemente, técnicas de Inteligência Computacional vêm sendo utilizadas com relativo sucesso. Destacam-se o uso de Redes Neurais, Máquina de Vetores de Suporte e Sistemas de Inferência Fuzzy [176, 185].

Para que tais modelos sejam aceitos pelo corpo de especialistas, devem ir além em termos de acurácia da abordagem padrão (Análise Discriminante Linear), assim como oferecer aos especialistas interpretações dos resultados. Uma alternativa viável é partir do uso de Sistemas Fuzzy-Genéticos [49, 98], que conferem ao usuário relativa acurácia e boa compreensão linguística das classificações efetuadas pelo modelo.

Contudo, na grande maioria dos trabalhos, os perfis utilizados na identificação litológica são aqueles apenas disponíveis nos poços, tais como os de raios gama, porosidade neutrônica e resistividade. Estas informações são somente conhecidas após a abertura dos poços. Na fase anterior, ou seja, fora do poço, os únicos registros disponíveis da subsuperfície são volumes com atributos derivados dos levantamentos sísmicos.

Dentre os atributos elásticos sísmicos existentes, um dos mais utilizados é a impedância acústica, tanto a compressional (IP) como a cisalhante (IS). Ela é definida como o produto entre a densidade e a velocidade de propagação

da onda compressional e cisalhante em um meio [196]. Utilizando-se as impedâncias compressional e cisalhante, é possível criar um modelo com dados de poços para classificar litologias e extrapolá-lo para os dados de sísmica.

Para este trabalho foram utilizados dados de um poço da costa brasileira. O poço selecionado para este estudo localiza-se em uma região com um sistema misto para a formação de rochas sedimentares. A litologia encontrada neste poço foi interpretada por um geólogo especialista que, para tal, utilizou-se dos perfis convencionais de poços de raios gama, densidade, fator fotoelétrico e sônico compressional, além da descrição petrográfica de amostras laterais e de fácies dos perfis de imagem (resistivo e acústico). Nesta interpretação litológica foram encontradas sete diferentes tipos de rochas: arenito, arenito carbonático, calcarenito, mudstone, packstone, argilito arenoso e folhelho. Estes foram agrupados em quatro diferentes classes, pois as características que diferenciam as rochas de uma mesma classe não são detectadas na escala sísmica. Na Tabela 5.39 é apresentado o agrupamento realizado para criar as quatro classes.

Tabela 5.39: Tipos de litologias e agrupamentos usados no poço em análise.

Litologia	Classe	Padrões	Freq.
Arenito	Arenito	1991	33,21%
Arenito Carbonático	Misto	1803	30,08%
Calcarenito	Carbonato	1480	24,69%
Mudstone			
Packstone			
Argilito Arenoso	Background	721	12,03%
Folhelho			

Para realizar a classificação serão usados, além dos valores diretos das impedâncias (IP e IS), mais dois atributos calculados a partir delas. O primeiro é a diferença entre IP e IS ($IP - IS$) enquanto o segundo é a Razão de Poisson (RP): $RP = \frac{IP^2 - 2IS^2}{2(IP^2 - IS^2)}$. A escolha desses outros atributos foi baseada no estudo de física de rochas, que aponta ambos como bons discriminantes litológicos [41, 94].

A Figura 5.14 apresenta o gráfico entre IP e IS com os dados do poço utilizado. As quatro classes estão identificadas pelas cores amarelo=arenito, vermelho=misto, azul=carbonato e verde=background. Quando as duas impedâncias são utilizadas em conjunto, observa-se que todos os dados, de acordo com os trabalhos [41, 94], localizam-se ao longo de uma tendência extremamente simples e estreita.

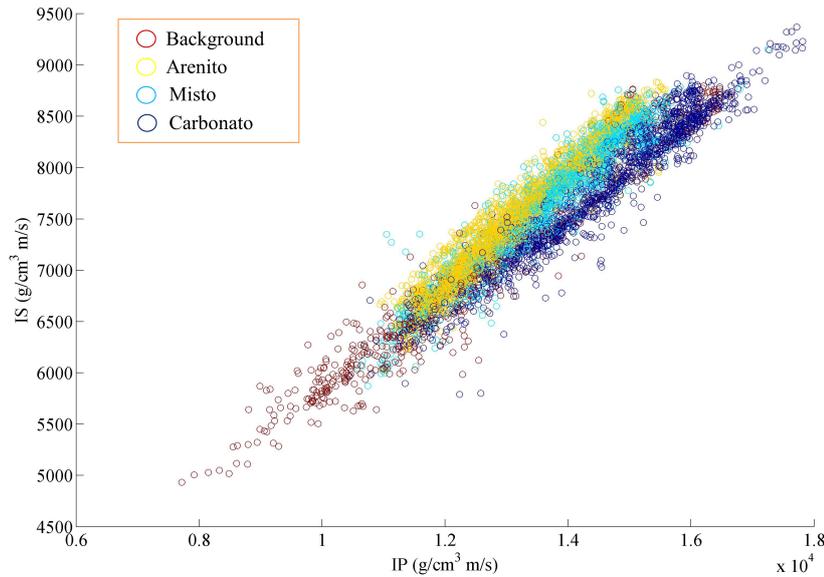


Figura 5.14: Relacionamento entre IP e IS e os tipos de rochas.

A partir dos atributos IP, IS, IP-IS e RP amostrados busca-se um modelo que possa inferir, a partir destes comportamentos, o tipo de rocha em questão: Arenito, Misto, Carbonato ou Background. A grande vantagem do GPFIS é a possibilidade de oferecer tal interpretação em nível linguístico e ainda possibilitar que o especialista decida qual o tipo de rocha, a partir do grau de pertinência de uma amostra sísmica às diferentes classes.

Descrição dos Experimentos

Além do modelo GPFIS, outros modelos para classificação foram também usados. A Tabela 5.40 apresenta cada um deles, com os parâmetros que os compõem. Observe-se a presença de abordagens estatísticas, como a Análise Discriminante Linear usada pelos especialistas, além de técnicas de inteligência computacional. Cabe ressaltar que a escolha do valor para cada parâmetro deveu-se a testes preliminares efetuados, visando à seleção da melhor configuração. Também, utilizou-se um SFG para base de regras do tipo Pittsburgh a partir da PG [23], denominado Pitt-GFS. Para este foi considerado o mesmo número de funções de pertinência e perfil usados no GPFIS (Figura 5.15) para cada atributo, de forma a tornar as abordagens mais próximas possíveis. Cabe ressaltar que a definição do formato e número das funções de pertinência foi estabelecida a priori pelos especialistas.

O poço analisado possui 5995 padrões no total. A forma de avaliar a acurácia de cada método foi a validação cruzada em 10 pastas (10-fold-cv), perfazendo em cada pasta o total de 5394 padrões de treino e 601 de teste.

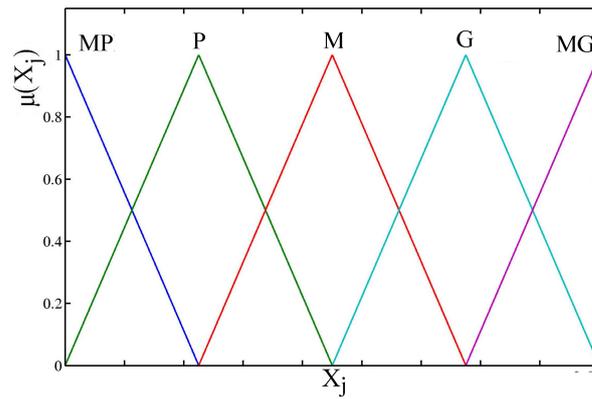


Figura 5.15: Funções de pertinência usadas.

Os resultados relatados são frutos da média de 3 execuções de 10-fold-cv para cada método. As métricas calculadas foram a acurácia total, que não discrimina o desbalanceamento entre as classes e a precisão média [74]. A grande vantagem da precisão média para a aplicação compreende o fato de esta levar em conta o desbalanceamento entre as classes (Arenito possui mais padrões do que Background, por exemplo) e, portanto, penalizar classificadores que privilegiam mais a classe dominante em detrimento das demais.

A Tabela 5.41 apresenta os parâmetros usados no modelo GPFIS.

No caso da PGMG, o processo utilizado foi similar ao de [128], concernente ao tratamento de um problema de múltiplas classes como de classes binárias. Para tanto, a PGMG é executada quatro vezes, dividindo de forma equivalente o número de avaliações factíveis (5000 para cada execução), tal que para cada execução é elaborada uma função discriminadora para uma determinada classe. No final das quatro execuções, as funções discriminadoras são

Tabela 5.40: Classificadores e parâmetros usados.

Modelo	Parâmetro
GPFIS	Tabela 5.41
SFGBR do tipo Pittsburgh (Pitt-GFS) [23]	Tabela 5.41
Bayes Ingênuo (NB) [154]	-
KNN [154]	3-nearest-neighbour, distância euclidiana
Árvore de Classificação (CART) [154]	-
Análise Discriminante Linear (DISC) [122]	-
PGMG	Tabela 5.41
Perceptron de Múltiplas Camadas (MLP) [97]	Uma camada escondida, função de ativação logística (escondida e saída) e 10 neurônios

Tabela 5.41: Principais configurações dos modelos baseados em Programação Genética.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	200
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.15
Operadores Fuzzy	Produto e Negação
Operador de Agregação	MQR
Método de Particionamento	CD
Decisão	Máxima Pertinência

reunidas e os padrões separados para a fase de teste são classificados. Para as abordagens evolutivas, a função de avaliação foi o Erro Médio de Classificação.

Resultados e Discussões

A Tabela 5.43 exhibe os resultados referentes à acurácia dos classificadores em análise. Verifica-se que, em geral, o modelo GPFIS obteve a maior acurácia, em média 4% maior do que o segundo modelo de melhor acurácia (MLP). Em comparação, Pitt-GFS obteve resultados piores do que o GPFIS e o NB. A abordagem padrão - DISC - proporcionou resultados com desempenho pior do que o do GPFIS. A Tabela 5.44 apresenta a precisão atingida por cada modelo. Ainda, o modelo GPFIS obteve os melhores resultados, proporcionando 6,98% maior precisão de classificação do que o NB, o segundo melhor neste quesito.

De uma forma geral, os resultados do GPFIS são superiores à abordagem do especialista (DISC) em dois quesitos: melhores resultados quando se avalia pelo volume de padrões corretamente classificados (acurácia) e equilíbrio dos esforços para atingir o máximo de padrões de diferentes classes (precisão). Cabe ressaltar que a abordagem NB, que demanda pouco esforço computacional, também auferiu bons resultados e pode ser útil em situações que requeiram aprendizado contínuo e decisões em curtíssimo prazo.

Os resultados indicam que o modelo GPFIS comporta-se relativamente bem em situações com baixa intensidade de desbalanceamento entre as classes, quando comparado aos demais classificadores. Uma explicação para isto vem da análise da base de regras fuzzy (Tabela 5.42) em conjunção com a Figura

5.16. Para este exemplo, considere-se a litologia Background. Verifica-se que a primeira regra, R1, estabelece que se IS é pequeno (P) então o padrão sísmico pertence ao tipo Background. Ainda, R2 e R3, quando conjuntamente avaliadas, descrevem que se IP é pequeno e IS é pequeno, com IP-IS não sendo nem médio e muito grande, então o padrão sísmico pertence ao tipo de rocha Background. Em suma, pode-se verificar que R1 e R2 delimitam a região em que os padrões da classe Background estão localizados: valores pequenos de IP e IS (Figura 5.14); R3 coopera com a R2 de modo a se concentrar mais nos padrões que se situam nesta região e relativamente perto de IP e IS médio. A partir dessa construção da região de discriminação, o modelo GPFIS consegue auferir boa classificação para as classes com menos padrões para treinamento.

Tabela 5.42: Base de regras fuzzy do melhor indivíduo do modelo GPFIS.

Regra	Antecedente	Consequente	Peso
R1	Se IP não é P ou M e IP-IS é G e RP é G	Arenito	0,40
R2	Se RP não é G	Arenito	0,19
R3	Se IP-IS é MG	Arenito	0,41
R4	Se IP é M e RP é G e IP-IS não é P ou M	Misto	0,20
R5	Se IS é G e IP-IS é M ou G ou MG	Misto	0,80
R6	Se IS é MG e IP-IS é M	Carbonato	0,50
R7	Se IP é G e RP é M	Carbonato	0,23
R8	Se IS é G e RP é M	Carbonato	0,52
R9	Se IP-IS não é M e RP não é G	Carbonato	0,18
R10	Se IS é P	Background	0,25
R11	Se IP é P e IP-IS não é M	Background	0,49
R12	Se IP não é MG e IS é P e IP-IS não é MG	Background	0,26

A Figura 5.16 apresenta a região de discriminação do modelo GPFIS com

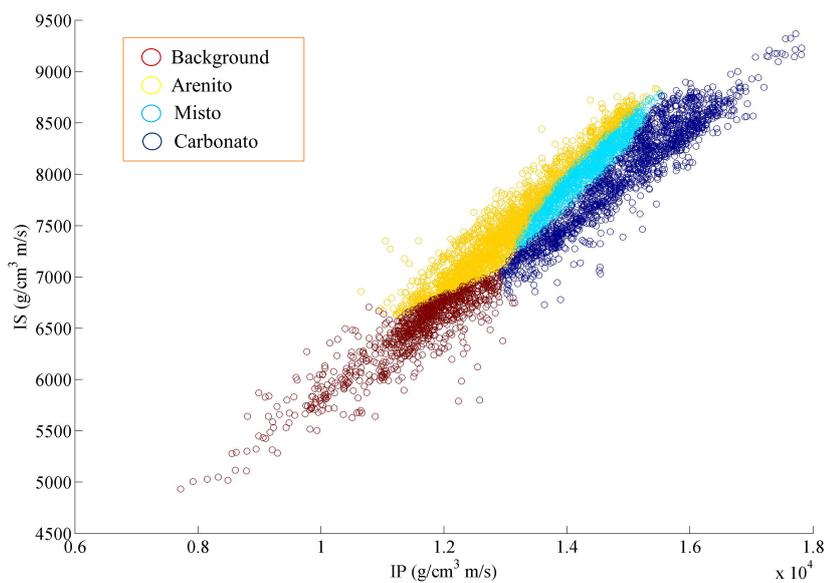


Figura 5.16: Classes previstas pelo modelo GPFIS.

a base de regras conforme Tabela 5.42. Observa-se que os padrões da litologia Background se localizam na faixa de valores pequenos e médios de IP e IS. Quando um padrão sísmico possui valores elevados de IP e IS, segundo o modelo GPFIS ele é classificado como Carbonato. Por fim, suponha um padrão sísmico com IP por volta de $14000 \text{ g/cm}^3\text{m/s}$ e IS de $8200 \text{ g/cm}^3\text{m/s}$. Após computar IP-IS e RP é possível avaliar que o grau de pertinência desse padrão para cada litologia é: Background=0,00, Carbonato=0,02, Arenito = 0,60 e Misto = 0,398. Um especialista pode interpretar este resultado de duas maneiras: (i) definir como rocha do tipo Arenito, a partir do critério de decisão da classe mais compatível; (ii) estabelecer que este padrão possui em torno de 60,00% de Arenito, 39,80% de Misto e traços de Carbonato (talvez por medições ruidosas). Esta última é devida à escassez de homogeneidade do padrão sísmico (por exemplo, grandes amostras, ou áreas com diferentes topologias). Ambos os tipos de interpretabilidades podem ser úteis ao especialistas e são viáveis a partir de um Sistema Fuzzy para Classificação como o GPFIS.

Tabela 5.43: Acurácia média na fase de teste das 3 execuções por pasta de validação cruzada.

Pasta	MLP	Pitt-GFS	GPFIS	PGMG	DISC	NB	CART	KNN
I	69,11%	57,22%	69,17%	47,17%	61,00%	69,50%	46,83%	48,33%
II	66,33%	57,15%	68,73%	57,48%	57,26%	57,76%	52,25%	53,92%
III	55,87%	54,09%	55,98%	44,69%	44,07%	60,43%	44,74%	45,74%
IV	60,60%	59,04%	63,44%	53,20%	46,74%	69,95%	45,58%	41,90%
V	51,53%	54,42%	68,78%	54,65%	50,25%	68,61%	56,09%	53,26%
VI	62,99%	55,04%	61,83%	56,82%	61,10%	51,92%	44,91%	46,91%
VII	46,52%	34,17%	65,55%	50,81%	40,57%	35,06%	43,57%	45,58%
VIII	51,78%	52,06%	49,83%	47,56%	46,67%	46,50%	41,17%	47,00%
IX	54,44%	54,22%	58,50%	58,22%	45,67%	51,33%	41,67%	45,67%
X	57,79%	49,81%	55,57%	52,63%	39,43%	46,26%	44,43%	48,25%
Média	57,70%	52,72%	61,74%	52,32%	49,28%	55,73%	46,12%	47,66%

Tabela 5.44: Precisão média na fase de teste das 3 execuções por pasta de validação cruzada.

Pasta	MLP	Pitt-GFS	GPFIS	PGMG	DISC	NB	CART	KNN
I	60,16%	47,95%	64,10%	44,66%	61,30%	61,38%	41,35%	44,23%
II	60,05%	54,64%	66,66%	49,73%	60,84%	53,63%	47,00%	49,62%
III	56,95%	53,65%	60,05%	38,49%	51,95%	61,47%	46,57%	47,90%
IV	58,20%	58,39%	65,39%	46,33%	54,11%	70,26%	45,97%	44,07%
V	43,67%	56,44%	69,66%	50,23%	55,69%	67,89%	57,84%	56,08%
VI	58,22%	51,65%	61,64%	50,08%	63,16%	51,00%	42,31%	45,85%
VII	43,47%	34,39%	64,47%	42,79%	46,62%	36,11%	40,45%	41,95%
VIII	49,23%	52,99%	52,21%	43,10%	52,50%	46,70%	38,71%	46,11%
IX	48,43%	49,69%	54,49%	50,16%	44,82%	47,49%	38,37%	41,22%
X	50,43%	44,19%	48,87%	46,94%	36,66%	41,73%	39,20%	43,41%
Média	52,88%	50,40%	60,75%	46,25%	52,77%	53,77%	43,78%	46,04%

5.3 Previsão

Diferentemente do realizado até agora, no tema Previsão o GPFIS não é comparado a outros SFGs. A avaliação do GPFIS em problemas de previsão faz uso da competição NN3 (Neural Network Competition) realizada em 2007, como uma sessão especial no IJCNN⁹. O objetivo da competição foi efetuar uma avaliação empírica de diferentes modelos de previsão, com maior ênfase nos de Redes Neurais, de modo a verificar quais deles possuíam maior acurácia fora da amostra de treinamento (out-of-sample). Enfatiza-se que, nos resultados do período da competição¹⁰ e na literatura disponível após a competição, não foi observado modelo algum do tipo SFG.

O que torna tal competição relevante é a gama de modelos de previsão envolvidos, como Redes Neurais, Métodos de Decomposição e Estatísticos. A Tabela 5.45 mostra de forma resumida os principais detalhes da competição.

Tabela 5.45: Principais detalhes da competição NN3.

Número de Competidores	25
Número de Benchmarks de Previsão	8
Quantidade de Séries Temporais	111
Periodicidade	Mensal
Tamanho Mínimo\Máximo das Séries	69\114
Métrica de Avaliação	SMAPE*
Horizonte de Previsão	18 meses à frente

* Symmetric Mean Absolute Percentage Error

Além da diversidade de modelos, o elevado número de séries temporais é também outro grande atrativo da NN3. A Tabela 5.46 apresenta os 10 melhores colocados na competição. Verifica-se, dentre as 10 melhores colocações, a presença de sete benchmarks de previsão (quatro estatísticos e três de inteligência computacional). Um destes - *Stat. Contender Wildi* - obteve o melhor resultado dentre todos os modelos. Em linhas gerais, ele se baseia na decomposição estrutural da série temporal mensal em 4 componentes:

$$y_t = T_t + C_t + S_t + I_t \quad (5-3)$$

onde T_t é a tendência, C_t é o ciclo, S_t é a sazonalidade e I_t é o ruído. A ideia é formular um modelo em espaço de estados para estimar as componentes T_t , C_t e S_t de modo a minimizar I_t . Este modelo inclui uma função objetivo

⁹<http://www.ijcnn2007.org/competition.htm>

¹⁰<http://www.neural-forecasting-competition.com/NN3/results.htm>

menos sensível à presença de valores discrepantes e um AG em conjunto com abordagens de programação matemática. A estimação de cada componente é realizada a partir da recursão proporcionada pelo Filtro de Kalman. Outras informações deste método, como combinação de previsões e resultados, podem ser encontradas no sítio da competição.

Tabela 5.46: Dez primeiros melhores colocados na competição NN3, considerando os resultados para as 111 séries temporais.

Colocação	Autor	SMAPE
-	Stat. Contender - Wildi	14,84%
-	Stat. Benchmark - Theta Method (Nikolopoulos)	14,89%
1	Illies, Jäger, Kosuchinas, Rincon, Sakenas e Vaskevcius	15,18%
-	Stat. Benchmark - ForecastPro (Stellwagen)	15,44%
-	CI Benchmark - Theta AI (Nikolopoulos)	15,66%
-	Stat. Benchmark - Autobox (Reilly)	15,95%
2	Adeodato, Vasconcelos, Arnaud, Chunha e Monteiro	16,17%
3	Flores, Anaya, Ramirez e Morales	16,31%
4	Chen e Yao	16,55%
5	D'yakonov	16,57%
6	Kamel, Atiya, Gayar e El-Shishiny	16,92%
7	Abou-Nasr	17,54%
8	Theodosiou e Swamy	17,55%
-	CI Benchmark - Naive MLP (Crone)	17,84%
9	de Vos	18,24%
10	Yan	18,58%
-	CI Benchmark - Naive SVR (Crone e Pietsch)	18,60%

O competidor campeão (excluindo os benchmarks de previsão) é a de Illies, Jäger, Kosuchinas, Rincon, Sakenas e Vaskevcius. Estes apresentam o uso de uma Echo State Network [120], apoiada por métodos de decomposição estrutural da série temporal, como o X-12 Arima¹¹ para ajuste sazonal e o uso de um filtro de médias móveis para remoção da tendência. Através da série filtrada pelos dois métodos, a Echo State Network é treinada e a previsão 18-passos-a-frente são efetuadas. Ainda, observa-se a presença de modelos de previsão reconhecidos pela literatura científica, como o método Theta [17], assim como pela indústria (Autobox e ForecastPro) nas melhores colocações da competição.

A crítica que pode ser estabelecida a tal competição reside no longo horizonte de previsão (18 passos-a-frente) e por somente conter séries cuja periodicidade é mensal. Na literatura são também encontradas outras competições de previsão, como as da família M (M1, M2 e M3) [144, 164]. A mais recente - M3

¹¹www.census.gov/srd/www/x12a

- comporta 3003 séries temporais com diferentes periodicidades (diária, mensal, etc.), subtraídas de áreas como indústria, governo, entre outras, contando com 24 métodos, na maioria de cunho estatístico.

Como o intuito é investigar de forma empírica o comportamento do modelo e, a partir disto, efetuar a comparação com os demais modelos, optou-se por uma competição menor de modo que seja factível analisar as diferentes configurações em tempo hábil. O próximo tópico apresenta a solução proposta para a competição NN3, a partir do modelo GPFIS para previsão.

5.3.1

Investigação Empírica da Arquitetura do Modelo GPFIS

Descrição dos Experimentos

De forma similar ao realizado em Classificação e Regressão, diversos experimentos foram conduzidos de modo a verificar a melhor configuração e a caracterizar a importância de cada etapa no resultado final. O delineamento dos experimentos foi sequencial, ou seja, os melhores parâmetros encontrados no experimento anterior são reproduzidos no próximo. A Tabela 5.47 exibe o delineamento.

Tabela 5.47: Delineamento experimental sequencial para o modelo GPFIS.

Descrição	Experimento 1	Experimento 2	Experimento 3
Hipótese Avaliada	Operadores Fuzzy	Operadores de Agregação	Métodos de Particionamento
Configuração Básica	Agregação por Máximo + Divisão Uniforme	Melhor configuração anterior + Divisão Uniforme	Melhor configuração anterior
Configurações Avaliadas	Operadores Fuzzy testados: - Produto (Prod) - Produto + Raiz Quadrada (Prod+RQ) - Produto + Negação (Prod+Neg) - Produto + Raiz Quadrada + Negação (Prod+RQ+Neg)	Operadores de Agregação testados: - Máximo - Média Aritmética - Produto - Mínimos Quadrados Restrito (MQR)	Métodos de Particionamento: - Divisão Uniforme (Div-Unif) - Grau de Similaridade (SD) - Grau de Confiança Fuzzy (FCD) - Máxima Credibilidade (Max-Cred)

O Experimento 1 tem o propósito de investigar o melhor conjunto de operadores fuzzy para o modelo GPFIS. São testadas 4 combinações: Produto (Prod), Produto+Negação (Prod+Neg), Produto+Raiz Quadrada (Prod+RQ)

e Produto+Raiz Quadrada+Negação (Prod+RQ+Neg). É tido como melhor conjunto a configuração que minimizar o SMAPE médio out-of-sample nas 111 séries da NN3. O Experimento 2 parte da melhor combinação entre os operadores fuzzy avaliados, e visa a verificar o efeito dos diferentes operadores de agregação usados. São testados: Máximo, Produto, Média Aritmética e Mínimos Quadrados Restritos (MQR).

Por fim, o Experimento 3 avalia a eficiência de outros métodos de particionamento além da padrão Divisão Uniforme (Div-Unif). Investigam-se o Grau de Similaridade (SD), Grau de Confiança Fuzzy (FCD) e Máxima Credibilidade (Max-Cred). A partir da melhor configuração obtida no Experimento 3, o tópico seguinte trata da comparação com os demais modelos de previsão participantes.

Cabe ressaltar que nesta investigação não se fez uso do procedimento estatístico (Teste de Friedman e Holm). Isto se deve a forma com que a comparação final entre o modelo GPFIS e os demais é feita: avalia-se qual dos modelos obteve o menor SMAPE médio out-of-sample nas 111 séries (única informação disponível no sítio da competição, não mencionando a sua significância).

Para todos os experimentos e séries foi usado o seguinte procedimento para a construção do modelo GPFIS:

- Remoção da tendência da série temporal.
- Treinamento do modelo GPFIS com a série sem tendência.
- Previsão multi-step adicionada da tendência estimada para o período.

O cômputo da tendência presente na série foi efetuada a partir de um modelo linear, dado por:

$$y_t = \beta_0 + \beta_1 t + z_t \Rightarrow y_t = r_t + z_t, \quad t = 1, \dots, T$$

no qual β_0 é o nível da série, β_1 a inclinação e r_t é a tendência linear da série y_t no instante de tempo t . Os parâmetros β_0 e β_1 são estimados por Mínimos Quadrados Ordinários. Com r_t calcula-se $z_t = y_t - r_t$ tal que z_t representa a série y_t subtraída da sua inclinação. Através de z_t é elaborado o modelo GPFIS, aproveitando as informações presentes em $z_{t-1}, z_{t-2}, \dots, z_{t-P}$, onde P é o número de defasagens aplicadas sobre z_t . Foi fixado $P = 12$, com o número e formato dos conjuntos fuzzy de acordo com a Figura 5.17. Os conjuntos fuzzy associados a z_t são B_1, \dots, B_5 , enquanto que aos z_{t-p} são associados A_{1p}, \dots, A_{5p} , para todo $p = 1, \dots, P$.

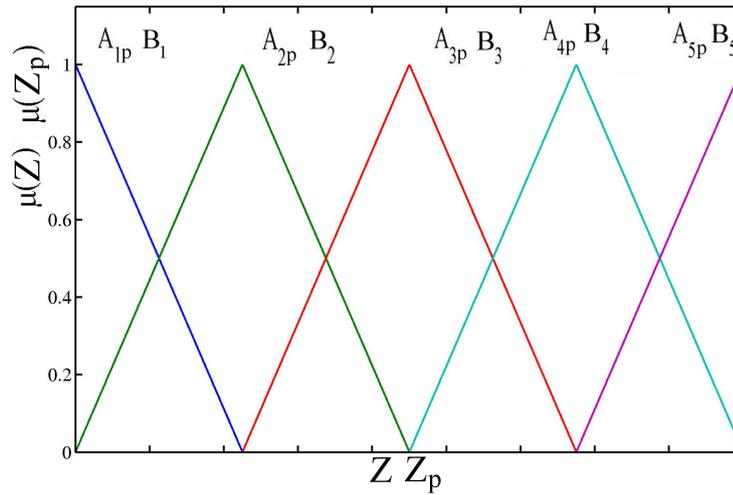


Figura 5.17: Funções de pertinência usadas no modelo GPFIS.

Demais configurações usadas em todos os experimentos e séries são dispostas na Tabela 5.48.

Tabela 5.48: Principais configurações do modelo GPFIS.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	100
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.17
Operadores Fuzzy	Tabela 5.47

Cada indivíduo é avaliado a partir de $\hat{y}_t = \hat{z}_t + r_t$, onde \hat{z}_t é o valor predito (defuzzificado) pelo modelo GPFIS através das informações in-sample (dentro da amostra). Com \hat{y}_t é calculado o SMAPE in-sample:

$$SMAPE = \frac{\sum_{t=1}^T \frac{|y_t - \hat{y}_t|}{|(y_t + \hat{y}_t)/2|}}{T} \quad (5-4)$$

a partir do melhor indivíduo obtido durante as 10000 avaliações efetuadas, a previsão multistep $\hat{z}_{t+1}, \dots, \hat{z}_{t+h}$ é realizada, em que $h = 18$ passos-a-frente. Com \hat{z}_{t+h} calcula-se $\hat{y}_{t+h} = \hat{z}_{t+h} + \hat{r}_{t+h}$, onde \hat{r}_{t+h} reflete a tendência estimada para o horizonte de previsão desejado. Este procedimento é executado 10 vezes para cada série temporal em cada experimento. Com os 10 resultados de

SMAPE in-sample e out-of-sample, computa-se uma média, que se constitui no resultado final do modelo GPFIS para a série em análise. O propósito desse processo é reduzir a dispersão dos resultados, devido a diferentes sementes iniciais usadas nos geradores de números aleatórios. Ainda, a escolha de 10000 avaliações no máximo deve-se em parte a este fato (10 execuções), mas visando à demonstrar que é possível obter boas soluções de forma rápida.

Além do SMAPE, outras métricas de qualidade de ajuste e avaliação também foram computadas, sendo apresentadas a seguir, com uma breve descrição da informação contida em cada uma. Ao leitor interessado em métricas de qualidade de ajuste em previsão, sugere-se a leitura de Hyndman e Koehler [106], que oferece uma descrição de cada métrica, além de outras não usadas nesta dissertação.

- **RMSE:** O Root Mean Squared Error (RMSE) é a versão obtida usando a raiz quadrada no EQM, de modo a obter um valor na escala do y_t original (o EQM está na escala quadrática do valor). A sua expressão é obtida por:

$$RMSE = \sqrt{\frac{\sum_{t=1}^T (y_t - \hat{y}_t)^2}{T}} \quad (5-5)$$

- **MAPE:** O Mean Absolute Percentage Error (MAPE) é uma das métricas mais reconhecidas na literatura. O MAPE é dado por:

$$MAPE = \frac{\sum_{t=1}^T \frac{|y_t - \hat{y}_t|}{|y_t|}}{T} \quad (5-6)$$

a vantagem do MAPE é a sua fácil interpretação, devido a adimensionalidade proporcionada pela razão de medidas. As principais desvantagens do MAPE são: (i) indefinição ou geração de resultados espúrios quando y_t assume valores próximo de 0; (ii) assimetria com respeito ao valor observado e predito, ou seja, fornece valores mais elevados quando o valor predito é maior do que o observado, e vice-versa. Portanto, o uso do SMAPE tem se generalizado com o objetivo de corrigir esta distorção.

- **U-Theil:** O U-Theil auferido quanto a previsão gerada pelo modelo em questão é relativamente superior à do modelo ingênuo:

$$U - Theil = \frac{\sqrt{\sum_{t=1}^T (y_t - \hat{y}_t)^2}}{\sqrt{\sum_{t=\gamma+1}^T (y_t - y_{t-1})^2}} \quad (5-7)$$

o numerador expressa o valor de RMSE obtido pelo modelo e o denominador, o RMSE obtido pelo método de previsão Ingênuo. A grande

vantagem desta métrica é verificar se o modelo em questão possui desempenho superior a um mais simples (neste caso U-Theil < 1), além de seu uso independer da série temporal sob análise.

- **MASE**: O Mean Absolute Scaled Error é uma métrica proposta recentemente por [106] e computada como:

$$MASE = \frac{\sum_{t=1}^T |y_t - \hat{y}_t|}{\sum_{t=\gamma+1}^T |y_t - y_{t-1}|} \quad (5-8)$$

a diferença do MASE para o U-Theil é a troca da noção de distância entre elementos. Enquanto que o U-Theil a expressa de forma euclidiana (penalizando mais as grandes distorções), o MASE segue a concepção de Manhattan, que dá menor peso para valores preditos distantes do observado.

Por fim, optou-se por reportar estatísticas dos resultados obtidos nas 111 séries, ao invés de apresentar a métrica computada para cada série. Isto visa à redução do espaço necessário para exibir as métricas auferidas nas 111 séries, além de possibilitar apresentar os resultados ao leitor de forma objetiva. Para tanto, calcularam-se a média, o desvio padrão, a mediana e o desvio médio absoluto (DMA) para cada métrica computada nas 111 séries. Apresentam-se os resultados obtidos nas 11 últimas. Frisa-se que todas as métricas reportadas são da fase out-of-sample, a partir da previsão multi-step 18 passos-a-frente.

Resultados e Discussões

A Tabela 5.49 apresenta as estatísticas das métricas de qualidade de ajuste obtidas no Experimento 1¹², para diferentes combinações dos operadores fuzzy na fase out-of-sample.

Ao considerar a análise pela média geral das 111 séries, observa-se que a abordagem Produto (Prod) obteve o melhor desempenho, tanto no quesito da média como no da mediana dos resultados. Ainda, ao considerar o subconjunto das 11 últimas séries, a média e mediana dos resultados auferidos reforçam este ponto. Observa-se que a combinação Prod+Neg, que proporcionara resultados superiores em Classificação e Regressão, não apresentou o mesmo desempenho agora.

Um dos possíveis motivos para a abordagem simples do Produto ter auferido os melhores resultados é o menor espaço de busca induzido pelo

¹²Durante a apresentação dos resultados, o identificador (111) significa a estatística da métrica considerando as 111 séries temporais, enquanto que para o subconjunto de 11 últimas são identificadas por (11).

Tabela 5.49: Resultados do Experimento 1 para o conjunto de operadores fuzzy.

Estatística	Métrica	Prod	Prod+RQ	Prod+Neg	Prod+RQ+Neg
Média (111)	RMSE	1370,36	1442,36	1425,36	1407,36
	MASE	0,8883	0,9278	0,9082	0,9170
	SMAPE	0,1842	0,2044	0,1994	0,2035
	MAPE	0,2468	0,2620	0,2506	0,2599
	U-Theil	0,8952	0,9646	0,9078	0,9149
Desvio Padrão (111)	RMSE	1592,19	1672,19	1646,19	1651,19
	MASE	0,4383	0,4644	0,4421	0,4528
	SMAPE	0,1407	0,1622	0,1538	0,1449
	MAPE	0,3437	0,3918	0,3612	0,3560
	U-Theil	0,9621	1,0112	0,9716	0,9850
Mediana (111)	RMSE	1095,94	1201,94	1126,94	1164,94
	MASE	0,8583	0,8986	0,8645	0,8807
	SMAPE	0,1563	0,2245	0,1709	0,1853
	MAPE	0,1547	0,2011	0,1759	0,1820
	U-Theil	0,7167	0,7761	0,7370	0,7427
DMA (111)	RMSE	674,96	759,96	729,96	733,96
	MASE	0,2920	0,3282	0,3017	0,3212
	SMAPE	0,1009	0,1157	0,1249	0,1150
	MAPE	0,1107	0,1391	0,1272	0,1155
	U-Theil	0,4174	0,4904	0,4295	0,4468
Média (11)	RMSE	1664,23	1751,23	1694,23	1718,23
	MASE	0,8486	0,9056	0,8714	0,8595
	SMAPE	0,1767	0,2090	0,1888	0,2024
	MAPE	0,1731	0,2126	0,1795	0,1986
	U-Theil	0,7589	0,8192	0,7612	0,7788
Desvio Padrão (11)	RMSE	2455,80	2555,80	2504,80	2519,80
	MASE	0,3355	0,3923	0,3392	0,3526
	SMAPE	0,1472	0,1622	0,1685	0,1530
	MAPE	0,1891	0,2308	0,1893	0,2107
	U-Theil	0,5018	0,5194	0,5226	0,5180
Mediana (11)	RMSE	1072,37	1171,37	1099,37	1136,37
	MASE	0,9084	0,9640	0,9241	0,9165
	SMAPE	0,1441	0,1788	0,1454	0,1696
	MAPE	0,1470	0,1736	0,1589	0,1501
	U-Theil	0,6788	0,7122	0,6993	0,6845
DMA (11)	RMSE	1522,73	1592,73	1569,73	1562,73
	MASE	0,2855	0,3280	0,3089	0,2939
	SMAPE	0,1559	0,2025	0,1717	0,1676
	MAPE	0,1551	0,1663	0,1607	0,1653
	U-Theil	0,4259	0,5008	0,4461	0,4512

uso exclusivo deste operador, combinado ao baixo número de avaliações disponíveis. Assim, ao mesmo tempo em que se restringem áreas onde se localizariam boas soluções (com o uso de negação, ou raiz quadrada), também se reduz a possibilidade do modelo tráfegar por regiões com péssimas soluções, com o risco de exaurir o número de avaliações sem obter um bom candidato.

A partir da melhor combinação dos operadores fuzzy determinados anteriormente, a Tabela 5.50 apresenta as estatísticas e métricas obtidas para o Experimento 2.

Tabela 5.50: Resultados do Experimento 2 para o conjunto de operadores de agregação.

Estatística	Métrica	Máximo	Média	Produto	MQR
Média (111)	RMSE	1370,36	1189,36	1446,36	990,36
	MASE	0,8883	0,8751	0,8928	0,8606
	SMAPE	0,1842	0,1758	0,1993	0,1625
	MAPE	0,2468	0,2320	0,2558	0,2078
	U-Theil	0,8952	0,8846	0,9117	0,8843
Desvio Padrão (111)	RMSE	1592,19	1407,19	1664,19	1215,19
	MASE	0,4383	0,4286	0,4386	0,4174
	SMAPE	0,1407	0,1335	0,1515	0,1262
	MAPE	0,3437	0,3380	0,3582	0,3185
	U-Theil	0,9621	0,9407	0,9651	0,9391
Mediana (111)	RMSE	1095,94	923,94	1174,94	749,94
	MASE	0,8583	0,8388	0,8594	0,8380
	SMAPE	0,1563	0,1409	0,1598	0,1261
	MAPE	0,1547	0,1380	0,1572	0,1368
	U-Theil	0,7167	0,7034	0,7265	0,6859
DMA (111)	RMSE	674,96	500,96	761,96	329,96
	MASE	0,2920	0,2684	0,3036	0,2453
	SMAPE	0,1009	0,0849	0,1015	0,0677
	MAPE	0,1107	0,0903	0,1173	0,0681
	U-Theil	0,4174	0,3959	0,4296	0,3740
Média (11)	RMSE	1664,23	1488,23	1743,23	1294,23
	MASE	0,8486	0,8423	0,8510	0,8202
	SMAPE	0,1767	0,1729	0,1825	0,1584
	MAPE	0,1731	0,1727	0,1770	0,1657
	U-Theil	0,7589	0,7443	0,7746	0,7384
Desvio Padrão (11)	RMSE	2455,80	2255,80	2522,80	2070,80
	MASE	0,3355	0,3348	0,3431	0,3176
	SMAPE	0,1472	0,1406	0,1594	0,1275
	MAPE	0,1891	0,1647	0,1924	0,1398
	U-Theil	0,5018	0,4830	0,5020	0,4783
Mediana (11)	RMSE	1072,37	902,37	1151,37	722,37
	MASE	0,9084	0,8913	0,9209	0,8751
	SMAPE	0,1441	0,1360	0,1537	0,1218
	MAPE	0,1470	0,1369	0,1576	0,1184
	U-Theil	0,6788	0,6690	0,6873	0,6656
DMA (11)	RMSE	1522,73	1361,73	1608,73	1182,73
	MASE	0,2855	0,2802	0,2860	0,2607
	SMAPE	0,1559	0,1350	0,1705	0,1161
	MAPE	0,1551	0,1497	0,1580	0,1328
	U-Theil	0,4259	0,4092	0,4259	0,4085

Similarmente ao observado nos estudos de casos anteriores, a agregação por Produto não foi efetiva. A maior dificuldade no processo de geração das regras reflete-se no baixo desempenho dessa abordagem (Tabela 5.51). O operador de agregação por Mínimos Quadrados Restritos (MQR) é indicado em ambos os sentidos, tanto pela redução do número de regras¹³ com respeito a abordagem por Máximo, quanto por garantir métricas de acurácia melhores

¹³Uma regra foi considerado como ativa quando o peso atrelado a ela foi maior do que 0,01.

na competição e no subconjunto das 11 últimas séries.

Tabela 5.51: Regras geradas por cada abordagem no Experimento 2.

Métrica \ Abordagem	Máximo	Média	Produto	MQR
Média (111)	16,57	18,24	9,48	14,56
Desvio Padrão (111)	7,76	10,12	2,57	4,94
Média (11)	19,64	20,35	10,51	17,55
Desvio Padrão (11)	9,04	7,38	3,95	4,19

De fato, tais resultados eram esperados por causa do maior investimento computacional demandado pelo MQR. A Tabela 5.52 apresenta a média do tempo computacional adicional (em termos percentuais) com relação aos demais operadores de agregação. Como relatado nas seções anteriores, o investimento computacional demandado pelo operador MQR é intensivo, com ordens de magnitude 5 vezes maior aos demais métodos. Ou seja, para cada avaliação efetuada por essa abordagem é possível, em média, efetuar em torno 5 avaliações por outros métodos de agregação. Logo, o usuário, tendo em mente este custo adicional, deve buscar mecanismos de atenuação, tais como a redução no número de avaliações (reduzir em torno de 5 vezes, ou próximo a isto) de modo a tornar o procedimento viável para aplicação.

Tabela 5.52: Proporção adicional de tempo tomado pela abordagem MQR com relação as demais operações de agregação.

Métrica \ Abordagem	Máximo	Média	Produto
Média (111)	457,71%	408,32%	562,19%
Desvio Padrão (111)	29,30%	22,54%	34,97%
Média (11)	484,91%	414,53%	510,22%
Desvio Padrão (11)	19,22%	18,69%	22,47%

A Tabela 5.53 apresenta os principais resultados no tocante aos diferentes métodos de particionamento (Experimento 3), revelando as estatísticas e métricas computadas para cada método considerado.

Em princípio, ao considerar a média e mediana obtida nas 111 séries e no subconjunto de 11, a abordagem por FCD auferiu os melhores resultados dentre os métodos de particionamento. Ao analisar os contrastes, observa-se uma diferença de 0,36% e de 0,83% no contraste entre: Divisão Uniforme-FCD e Max-Cred-FCD, respectivamente. Contrariamente ao problema de Regressão, o método de particionamento FCD obteve o SMAPE claramente menor do que o SD (cerca de 1,51%).

Tabela 5.53: Resultados do Experimento 3 para o conjunto de métodos de particionamento.

Estatística	Métrica	Div-Unif	SD	Max-Cred	FCD
Média (111)	RMSE	990,36	1034,36	996,36	953,73
	MASE	0,8606	0,8728	0,8597	0,8411
	SMAPE	0,1625	0,1740	0,1672	0,1589
	MAPE	0,2078	0,2182	0,2051	0,1969
	U-Theil	0,8843	0,8934	0,8988	0,8441
Desvio Padrão (111)	RMSE	1215,19	1253,19	1194,19	1106,39
	MASE	0,4174	0,4189	0,4246	0,3583
	SMAPE	0,1262	0,1410	0,1235	0,1175
	MAPE	0,3185	0,3257	0,3197	0,3102
	U-Theil	0,9391	0,9431	0,9522	0,6900
Mediana (111)	RMSE	749,94	762,94	741,94	697,81
	MASE	0,8380	0,8389	0,8276	0,8301
	SMAPE	0,1261	0,1416	0,1377	0,1209
	MAPE	0,1368	0,1524	0,1488	0,1276
	U-Theil	0,6859	0,6992	0,6825	0,7242
DMA (111)	RMSE	329,96	374,96	351,96	328,52
	MASE	0,2453	0,2515	0,2590	0,2440
	SMAPE	0,0677	0,0798	0,0659	0,0541
	MAPE	0,0681	0,0870	0,0751	0,0665
	U-Theil	0,3740	0,3784	0,3770	0,3531
Média (11)	RMSE	1294,23	1312,23	1274,23	1023,46
	MASE	0,8202	0,8327	0,8086	0,7859
	SMAPE	0,1584	0,1618	0,1655	0,1421
	MAPE	0,1657	0,1755	0,1774	0,1501
	U-Theil	0,7384	0,7492	0,7434	0,7167
Desvio Padrão (11)	RMSE	2070,80	2118,80	2048,80	1360,06
	MASE	0,3176	0,3350	0,3230	0,3405
	SMAPE	0,1275	0,1358	0,1318	0,1082
	MAPE	0,1398	0,1533	0,1394	0,1155
	U-Theil	0,4783	0,4799	0,4798	0,5206
Mediana (11)	RMSE	722,37	756,37	718,37	690,62
	MASE	0,8751	0,8917	0,8848	0,8282
	SMAPE	0,1218	0,1407	0,1079	0,1021
	MAPE	0,1184	0,1289	0,1244	0,1120
	U-Theil	0,6656	0,6738	0,6543	0,4870
DMA (11)	RMSE	1182,73	1227,73	1196,73	853,44
	MASE	0,2607	0,2638	0,2720	0,2746
	SMAPE	0,1161	0,1294	0,1143	0,0927
	MAPE	0,1328	0,1467	0,1261	0,0989
	U-Theil	0,4085	0,4133	0,4198	0,4239

Com relação ao tempo computacional (Tabela 5.54) os métodos são semelhantes, entretanto com o tempo computacional mais elevado do que a Divisão Uniforme (praticamente o dobro de tempo de máquina). Com respeito ao número de regras (Tabela 5.55), a média obtida pelo FCD foi menor do que nos demais métodos, porém pouco substancial.

Tabela 5.54: Proporção adicional de tempo tomado pelas demais abordagens com respeito ao método de Divisão Uniforme.

Métrica \ Abordagem	FCD	SD	Max-Cred
Média (111)	84,77%	88,99%	77,51%
Desvio Padrão (111)	7,96%	8,15%	5,43%
Média (11)	92,23%	95,63%	84,74%
Desvio Padrão (11)	5,36%	6,12%	4,33%

Tabela 5.55: Regras geradas por cada abordagem no Experimento 3.

Métrica \ Abordagem	Div-Unif.	SD	FCD	Max-Cred
Média (111)	14,56	14,02	13,41	18,94
Desvio Padrão (111)	4,94	2,92	2,14	7,11
Média (11)	17,55	16,71	15,48	22,55
Desvio Padrão (11)	4,19	3,84	4,47	5,14

Por fim, a partir dos resultados da Tabela 5.53 tem-se a configuração final do GPFIS:

- Operadores Fuzzy: Produto.
- Operador de Agregação: MQR.
- Método de Particionamento: FCD.

O próximo tópico aborda a comparação com os outros modelos de previsão.

5.3.2 Comparação com Demais Modelos

A partir da melhor configuração anteriormente definida, captam-se os resultados referentes ao SMAPE médio obtido nas 111 séries temporais, assim como no subconjunto das 11 últimas, para efeito de comparação com os outros modelos. A Tabela 5.56 apresenta a colocação do modelo GPFIS com respeito aos resultados da competição¹⁴.

Nota-se a presença do modelo GPFIS na segunda colocação, com um SMAPE de aproximadamente 0,71% maior do que o do primeiro colocado, a Echo State Network treinada por Illies, Jäger, Kosuchinas, Rincon, Sakenas e Vaskevcius. Apesar disto, o modelo GPFIS possui a vantagem da interpretação linguística das previsões ensejadas, ao contrário da Echo State Network, ou do benchmark estatístico do Wildi.

¹⁴<http://www.neural-forecasting-competition.com/NN3/results.htm>

Tabela 5.56: Dez primeiros na competição NN3, considerando os resultados para as 111 séries temporais.

Colocação	Autor	SMAPE
-	<i>Stat. Contender - Wildi</i>	14,84%
-	<i>Stat. Benchmark - Theta Method (Nikolopoulos)</i>	14,89%
1	Ilies, Jäger, Kosuchinas, Rincon, Sakenas e Vaskevcius	15,18%
-	<i>Stat. Benchmark - ForecastPro (Stellwagen)</i>	15,44%
-	<i>CI Benchmark - Theta AI (Nikolopoulos)</i>	15,66%
2	GPFIS (Prod-MQR-FCD)	15,89%
-	<i>Stat. Benchmark - Autobox (Reilly)</i>	15,95%
3	Adeodato, Vasconcelos, Arnaud, Chunha e Monteiro	16,17%
4	Flores, Anaya, Ramirez e Morales	16,31%
5	Chen e Yao	16,55%
6	D'yakonov	16,57%
7	Kamel, Atiya, Gayar e El-Shishiny	16,92%
8	Abou-Nasr	17,54%
9	Theodosiou e Swamy	17,55%
-	<i>CI Benchmark - Naive MLP (Crone)</i>	17,84%
10	de Vos	18,24%

Para o subconjunto das 11 últimas séries, a Tabela 5.57 apresenta a colocação do modelo GPFIS: está novamente na segunda colocação, com um resultado 0,53% maior ao do modelo proposto por Yan, uma Rede Neural baseada em Modelos Lineares Generalizados.

Tabela 5.57: Dez primeiros colocados na competição NN3, considerando os resultados referentes as últimas 11 séries temporais.

Colocação	Autor	SMAPE
-	<i>CI Benchmark - Theta AI (Nikolopoulos)</i>	13,07%
-	<i>Stat. Benchmark - Autobox (Reily)</i>	13,49%
-	<i>Stat. Benchmark - ForecastPro (Stellwagen)</i>	13,52%
1	Yan	13,68%
-	<i>Stat. Benchmark - Theta (Nikolopoulos)</i>	13,70%
2	GPFIS (Prod-MQR-FCD)	14,21%
3	Ilies, Jager, Kosuchinas, Rincon, Sakenas e Vaskevcius	14,26%
4	Chen e Yao	14,46%
5	Yousefi, Miromeni e Lucas	14,49%
6	Ahmed, Atiya, Gayar e El-Shishiny	14,52%
7	Flores, Anaya, Ramirez e Morales	15,00%
8	Adeodato, Vasconcelos, Arnaud, Chunha e Monteiro	15,10%
	<i>Stat. Contender - Wildi</i>	15,32%
9	Luna, Soares e Ballini	15,35%
10	Theodosiou e Swamy	16,19%

Por fim, a Tabela 5.58 exhibe os resultados do modelo GPFIS de melhor configuração, com respeito ao SMAPE e RMSE médios das 10 execuções efetuadas para cada série temporal da NN3.

Tabela 5.58: SMAPE e RMSE obtidos na fase out-of-sample pelo modelo GPFIS em cada série temporal da NN3.

Série	RMSE	SMAPE (%)	Série	RMSE	SMAPE (%)	Série	RMSE	SMAPE (%)	Série	RMSE	SMAPE (%)
NN3-001	624,62	8,3883	NN3-029	1498,60	34,3862	NN3-057	368,18	4,5006	NN3-085	711,99	14,1195
NN3-002	274,13	4,9849	NN3-030	740,57	8,6958	NN3-058	329,07	4,2562	NN3-086	1814,21	21,6748
NN3-003	498,24	6,5898	NN3-031	587,41	37,7177	NN3-059	286,98	5,6019	NN3-087	429,99	17,3483
NN3-004	479,95	7,6890	NN3-032	1280,88	20,3049	NN3-060	184,24	2,8473	NN3-088	299,40	5,8678
NN3-005	422,24	9,3356	NN3-033	813,01	14,7995	NN3-061	526,73	10,0929	NN3-089	199,17	8,0365
NN3-006	399,21	7,8263	NN3-034	588,07	11,2620	NN3-062	1537,14	39,5468	NN3-090	372,27	7,1834
NN3-007	1034,11	18,2013	NN3-035	401,35	11,6480	NN3-063	1031,36	16,4278	NN3-091	259,88	3,7906
NN3-008	1325,96	22,6936	NN3-036	530,69	8,7602	NN3-064	575,98	7,0071	NN3-092	101,83	3,1441
NN3-009	670,58	14,4699	NN3-037	1157,94	13,3420	NN3-065	1245,17	17,5041	NN3-093	10407,42	67,8101
NN3-010	558,28	13,8953	NN3-038	901,69	17,0456	NN3-066	338,69	3,8362	NN3-094	528,53	9,7980
NN3-011	1004,31	12,0895	NN3-039	574,90	11,9332	NN3-067	1026,33	18,2030	NN3-095	1427,99	26,9009
NN3-012	658,29	10,4403	NN3-040	457,50	12,4275	NN3-068	152,64	4,5212	NN3-096	2316,56	46,0566
NN3-013	933,41	10,7438	NN3-041	1158,38	20,7794	NN3-069	516,09	22,9550	NN3-097	850,48	8,9116
NN3-014	745,35	13,9672	NN3-042	899,10	12,5695	NN3-070	1191,61	12,5847	NN3-098	1617,87	28,7101
NN3-015	615,84	8,4854	NN3-043	986,08	34,7662	NN3-071	1750,82	25,5908	NN3-099	1057,08	19,6824
NN3-016	629,38	6,6981	NN3-044	1660,30	32,9163	NN3-072	418,77	7,3929	NN3-100	155,23	3,8004
NN3-017	697,81	8,7245	NN3-045	1059,09	29,2668	NN3-073	550,32	10,1030	NN3-101	167,09	2,5700
NN3-018	405,26	8,8887	NN3-046	1315,25	35,2586	NN3-074	1044,80	22,2701	NN3-102	1540,97	25,0294
NN3-019	1384,53	36,9246	NN3-047	1586,43	26,8622	NN3-075	396,03	6,7880	NN3-103	4880,03	18,1570
NN3-020	743,47	15,2757	NN3-048	1439,43	20,6981	NN3-076	1477,97	14,3424	NN3-104	690,62	8,7113
NN3-021	1763,15	32,2113	NN3-049	1308,86	29,7341	NN3-077	150,43	2,2861	NN3-105	230,12	3,8457
NN3-022	996,11	18,9458	NN3-050	866,51	10,3544	NN3-078	842,88	10,6909	NN3-106	273,19	4,7937
NN3-023	546,74	13,8698	NN3-051	682,81	14,1087	NN3-079	493,62	8,1549	NN3-107	181,48	4,1587
NN3-024	1780,15	15,9010	NN3-052	1030,48	9,8147	NN3-080	673,63	9,8572	NN3-108	1343,28	30,7407
NN3-025	1587,43	54,4663	NN3-053	436,67	5,1081	NN3-081	2002,41	40,1683	NN3-109	414,41	10,2083
NN3-026	1315,34	27,6934	NN3-054	625,93	10,1498	NN3-082	2858,24	31,2485	NN3-110	832,04	30,8678
NN3-027	850,02	15,9945	NN3-055	893,42	8,9783	NN3-083	503,27	10,3086	NN3-111	704,85	17,2718
NN3-028	655,80	8,8001	NN3-056	229,28	2,5619	NN3-084	272,24	4,9049			

A partir da Tabela 5.58, outros trabalhos que venham a usar a competição NN3 podem usufruir dos resultados do melhor modelo GPFIS e elaborar uma comparação além da simples média do SMAPE obtido nas 111 (ou 11) séries temporais. A próxima seção aborda a aplicação detalhada do modelo GPFIS para um problema de Previsão, a partir das configurações estabelecidas no tópico da investigação empírica.

5.3.3

Aplicação Detalhada

Descrição do Problema

Em uma economia moderna, as relações de trocas são de grande importância, em níveis micro e macroeconômico. As relações microeconômicas envolvem indivíduos na compra e venda de bens e serviços, na maior parte com uma mesma moeda sendo o meio de troca. As relações macroeconômicas envolvem agentes amplos da economia, como países, que, na maioria, possuem moedas distintas. Ao realizarem uma troca, é necessário que haja um balizador do valor de uma moeda em unidades monetárias de outra moeda. Isto é definido como taxa de câmbio [134].

A taxa de câmbio é dada pela demanda e oferta das moedas dos países que interagem comercialmente. A taxa de câmbio pode apresentar três situações: valorizada, quando são necessárias mais unidades monetárias externas para adquirir a moeda interna; em paridade, quando a troca é igual; e desvalorizada, quando é necessária uma maior quantidade de moeda interna para adquirir moeda externa [178]. Economistas têm chamado a atenção para a importância da competitividade da taxa de câmbio no processo de desenvolvimento econômico. Ao exercer impactos de aumento de produtividade e investimento, o nível da taxa de câmbio torna-se uma variável-chave para a promoção do crescimento.

A política cambial, na qual a competitividade da taxa de câmbio é elemento fundamental, representa um ferramental estratégico a ser utilizado por governos visando ao crescimento e desenvolvimento pleno de uma sociedade. Além de evitar crises na balança comercial, um câmbio relativamente desvalorizado pode fornecer estímulos para aumentos de investimento e poupança, impactando o processo de acumulação de capital [134]. Quanto às questões produtivas, um câmbio competitivo gera impactos positivos para o resto da economia a partir do estímulo à produção de manufaturas para exportação, além de evitar processos de desindustrialização ou agropecuária predatória

[32].

Portanto, analisar como os estados anteriores da taxa de câmbio (desvalorizada, valorizada, etc.) impactam no seu comportamento futuro é importante para a formulação de políticas externas, estabilização e expansão econômica. Ainda, fornecer previsões acuradas do comportamento futuro da série pode auxiliar tanto instituições públicas e privadas para formulação de hedges e níveis de reserva cambial. O uso do modelo GPFIS é uma alternativa viável para atingir ambos os objetivos: proporcionar a previsão e obter uma descrição linguística da estrutura de correlação da série. A próxima seção detalha a base de dados usada e demais modelos comparados.

Descrição dos Experimentos

Esta aplicação trata da taxa de câmbio Real (R\$) por Dólar americano (US\$) mensal do período de janeiro de 1999 a janeiro de 2014 [68] (antes de 1999 o câmbio não era flutuante). O horizonte de previsão definido foi de 6 passos-a-frente. As métricas out-of-sample retratam o período de agosto de 2013 a janeiro de 2014, enquanto que o espaço de tempo anterior compreende as métricas in-sample.

Além do modelo GPFIS, outros previsores foram usados. A Tabela 5.59 apresenta cada um deles e os parâmetros que os compõem. Observe-se a presença de abordagens estatísticas, tais como os da Família Box & Jenkins (ARIMA e SARIMA) [155] e os denotados por Amortecimento Exponencial: Holt-Winters Aditivo (HWA) e Multiplicativo (HWM) [155]. Para estes seguiu-se a metodologia padrão: (i) identificação (determinação da estrutura do modelo); (ii) estimação (cômputo das quantidades previamente desconhecidas do modelo); (iii) diagnóstico (verificação da adequabilidade da estrutura em termos de acurácia e pressupostos); (iv) previsão. Também foi construído um modelo Ingênuo de previsão.

Foram também usadas técnicas advindas da inteligência computacional: PGMG e um Perceptron de Múltiplas Camadas (MLP). Para a PGMG segue-se a mesma parametrização do modelo GPFIS de modo a prover meios de comparação entre as abordagens. O treinamento da MLP foi realizado segundo o procedimento padrão: separação da base de dados no conjunto de treinamento (80%) e validação (20%); estimação dos parâmetros da MLP e previsão multi-step. Cada abordagem de inteligência computacional foi executada 15 vezes, sendo os resultados reportados fruto da média destas 15 execuções.

Como conjunto de entradas para o modelo GPFIS, MLP e PGMG foram fornecidas 12 defasagens (y_{t-1}, \dots, y_{t-12}) da série temporal da taxa de

Tabela 5.59: Previsores e parâmetros usados.

Modelo	Parâmetro
GPFIS	Tabela 5.60
PGMG	Tabela 5.60
Ingênuo [155]	-
Box e Jenkins ARIMA [155]	$p = 7, d = 1$ e $q = 0$
Holt-Winters Aditivo (HWA) [155]	-
Holt-Winters Multiplicativo (HWM) [155]	-
MLP [97]	Uma camada escondida, função de ativação logística (escondida) e linear (saída) e 10 neurônios

câmbio (y_t). A Figura 5.18 apresenta as funções de pertinência delineadas tanto para a taxa de câmbio no instante t (y_t), quanto para as suas respectivas defasagens (y_{t-p} , $p = 1, \dots, 12$). Cada rótulo linguístico representa: Muito Pouco Desvalorizada (MPD), Pouco Desvalorizada (PD), Desvalorizada (D), Muito Desvalorizada (MD) e Extremamente Desvalorizada (ED).

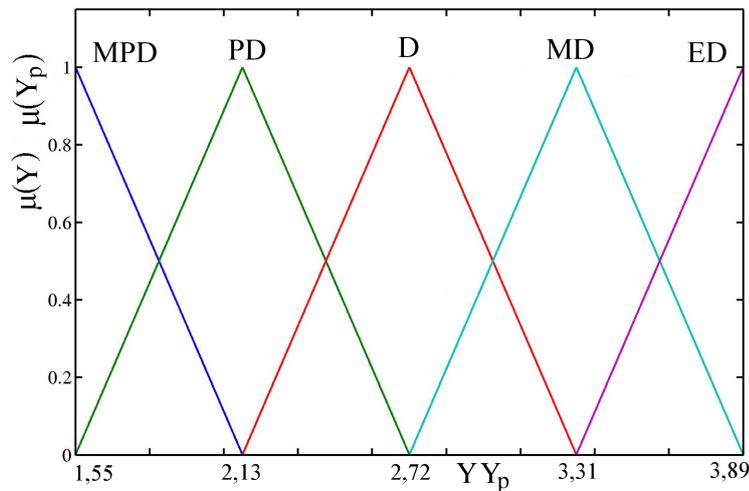


Figura 5.18: Funções de pertinência usadas no modelo GPFIS.

Nos modelos MLP e PGMG foram incluído 12 entradas binárias representando cada mês no ano, codificadas no formato 1 de n, que toma o valor 1 quando o mês a ser predito corresponde a esta entrada e 0 em caso contrário. Com isto, busca-se captar o efeito sazonal da série presente nas variações médias nos meses. Por fim, cabe ressaltar que a escolha do valor para cada parâmetro levou em conta a testes preliminares efetuados. A Tabela 5.60 apresenta os parâmetros usados no modelo GPFIS e na PGMG.

Tabela 5.60: Principais configurações dos modelos baseados em Programação Genética.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	100
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.18
Operadores Fuzzy	Produto
Operador de Agregação	MQR
Método de Particionamento	FCD
Defuzzificação	Altura

Resultados e Discussões

A Figura 5.19 apresenta a série temporal dos valores observados da taxa de câmbio R\$/US\$ (a) e a sua decomposição estrutural em três partes: tendência (b), sazonalidade (c) e ruído (d). Observa-se o crescimento da tendência (Figura 5.19b) da taxa de câmbio, até o pico em 2003, ultrapassando os R\$/US\$ 4,00 devido a intensa fuga de capitais, o aumento do risco-país e o processo eleitoral de 2002-2003 [168].

Ao analisar dezembro de 2002, é possível perceber que a tendência da série se inflexiona, devido ao aumento do nível de reservas monetárias no período, o que indicava que a balança comercial tendia a um processo de equilíbrio [24]. De meados de 2003 a 2008, reduzidas as incertezas com relação à política econômica do novo governo Lula, a taxa de câmbio apresenta uma apreciação em sua trajetória, muito favorecida pela redução da taxa básica de juros norte-americana e a expansão econômica vivida pelo país [139].

Em momentos de crise, os investidores tendem a aumentar suas demandas por títulos do governo americano, considerados como tendo risco zero. Dessa forma o preço da moeda norte-americana aumenta, ou seja, o real é novamente desvalorizado. Esse foi um dos fatores mais significativos para a desvalorização cambial durante a crise de 2008 [181]. Outro ponto na Figura (Figura 5.19c) é a sazonalidade do câmbio, de pequena escala, variando de -0,04 a 0,04 centavos. No primeiro semestre do ano, a sazonalidade não exerce efeito no câmbio no mês de janeiro, enquanto que, de fevereiro a maio, há efeito negativo valorizando o Real frente ao Dólar.

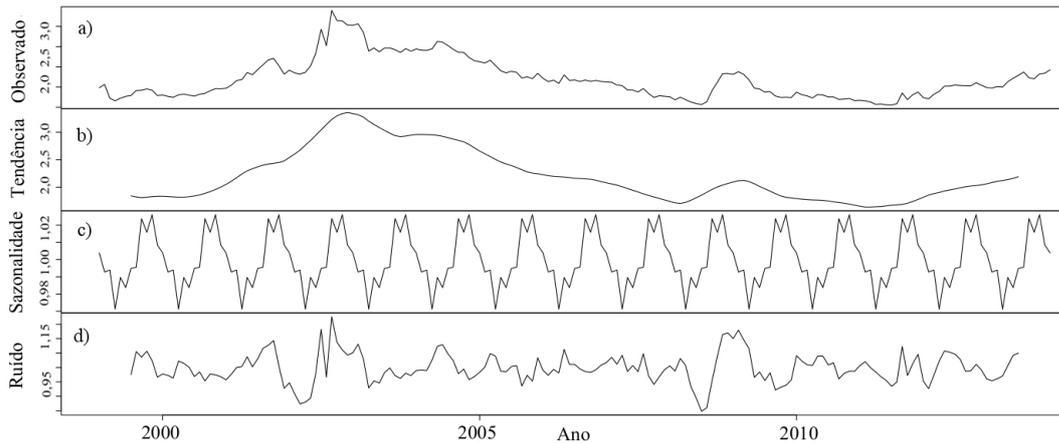


Figura 5.19: a) Taxa de câmbio R\$/US\$ do período janeiro de 1999 a março de 2011. b) Tendência. c) Sazonalidade. d) Ruído resultante da decomposição estrutural.

No segundo semestre, o câmbio se desvaloriza, atingindo o pico em agosto, devido à intensificação das compras de insumos e matérias-primas para a produção do fim de ano. Após, o câmbio retorna vagarosamente à estabilização. A parte aleatória representa 20% do observado, ou seja, é possível inferir 80% do comportamento da série da taxa de câmbio através da tendência e da sazonalidade.

Com respeito ao desempenho dos modelos de previsão usados, a Tabela 5.61 apresenta as métricas de qualidade de ajuste na fase in-sample. Verifica-se que a PGMG obteve os melhores resultados em termos de RMSE, MAPE e SMAPE. Notam-se bons resultados do modelo Ingênuo, pois verifica-se ao longo do processo de identificação do modelo ARIMA(7,1,0) que a série temporal em análise, após sofrer diferenciação, segue um comportamento quase de um passeio aleatório, estrutura esta idêntica à do método Ingênuo.

Tabela 5.61: Principais resultados dos previsores na fase in-sample.

Modelo\Métrica	RMSE	MAPE	SMAPE
Ingênuo	0,13209	3,4688%	3,4528%
HWA	0,14305	3,7279%	3,7345%
HWM	0,14495	3,6844%	3,6945%
ARIMA (7,1,0)	0,11899	3,5176%	3,5036%
MLP (1,10)	0,11547	3,5107%	3,5115%
PGMG	0,11508	2,9677%	2,9758%
GPFIS	0,11746	3,2339%	3,2736%

A Tabela 5.62 apresenta o resultado dos previsores na fase out-of-sample. Observa-se que, em termos de RMSE e MAPE, o modelo HWM obteve

os melhores resultados dentre os modelos. Quando considerado o SMAPE agregado em 6 passos de previsão, o HWM também proporcionou os melhores resultados. Entretanto, quando se considera previsões de curto e médio prazo, 2 a 4 passos-a-frente, o modelo GPFIS se apresenta como uma abordagem acurada para a série em análise, com SMAPE abaixo de 3%.

Tabela 5.62: Principais resultados dos previsores na fase out-of-sample.

Modelo\Métrica	RMSE	MAPE	SMAPE (2-passos)	SMAPE (4-passos)	SMAPE (6-passos)
Ingênuo	0,13797	4,8090%	4,3329%	3,5174%	4,9833%
HWA	0,08532	3,2867%	3,9137%	3,7861%	3,2701%
HWM	0,08305	3,2209%	3,8042%	3,6742%	3,2091%
ARIMA (7,1,0)	0,09973	3,6368%	3,5545%	4,1089%	3,5834%
MLP (1,10)	0,11661	4,6115%	4,0069%	4,7724%	4,4838%
PGMG	0,09242	3,6012%	3,7957%	4,4135%	3,5310%
GPFIS	0,11045	4,3013%	2,9791%	2,9242%	4,4281%

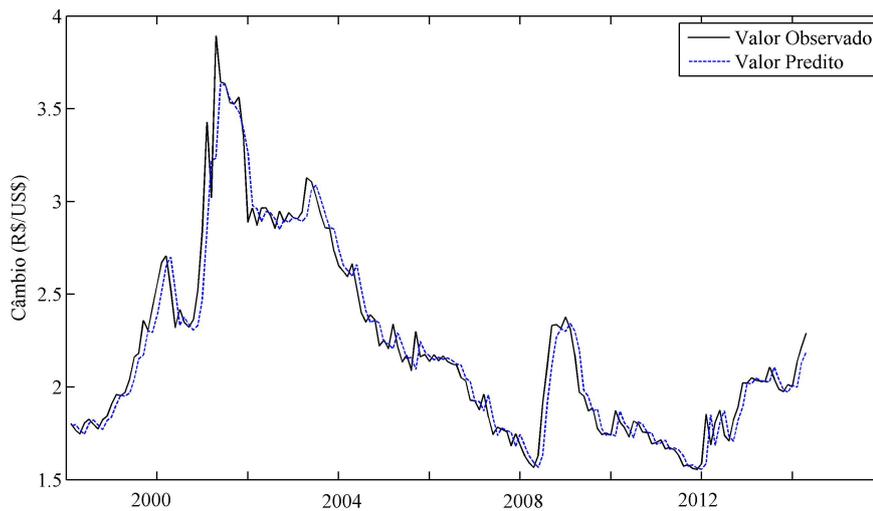


Figura 5.20: Valores observados e preditos pelo modelo GPFIS para a fase in-sample.

Este fato é relevante, pois em séries financeiras, além de avaliar o SMAPE é necessário compará-lo com a variação média da série na fase out-of-sample. Se a variação média da série for menor do que o SMAPE, o modelo está propondo valores mais elevados ou menores do que a dispersão real da série. Neste caso, a série em análise teve uma variação média em torno de 3,43% nestes 6 meses. Logo, o modelo GPFIS possibilitou previsões dentro do intervalo médio de variação da série. Por fim, a Figura 5.20 mostra os valores preditos pelo modelo GPFIS na fase in-sample, enquanto a Figura 5.21 apresenta a previsão efetuada pelo modelo (localizados após a linha vermelha). Verifica-se que em

três dos quatro primeiros meses o modelo acompanhou relativamente bem o comportamento da série, enquanto que nos dois últimos o erro incorrido pelo modelo foi elevado, não logrando acompanhar a trajetória crescente da série.

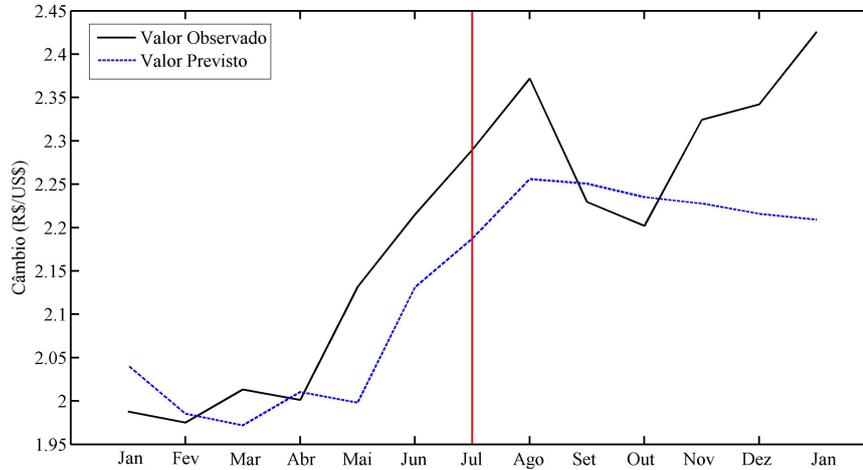


Figura 5.21: Valores observados, preditos e previstos (após a linha vermelha) pelo modelo GPFIS para a fase out-of-sample.

Por fim, a Tabela 5.63 apresenta a base de regras obtida pelo modelo GPFIS na forma padrão. A Tabela 5.64 apresenta a mesma ideia, porém de forma sucinta, com o objetivo de revelar diretamente quais estados linguísticos das defasagens (y_{t-p} , $p = 1, 2, \dots, 12$) estão relacionados aos comportamentos expressados pela taxa de câmbio no instante atual (y_t): Muito Pouco Desvalorizado (MPD), Pouco Desvalorizado (PD), Desvalorizado (D), Muito desvalorizado (MD) e Extremamente Desvalorizado (ED).

Tabela 5.63: Base de Regras Fuzzy com representação tradicional.

Regra	Antecedente	Consequente	Peso
R1	Se Y_{t-1} é MPD	MPD	1,00
R2	Se Y_{t-1} é PD	PD	0,96
R3	Se Y_{t-6} é D e Y_{t-12} é PD	PD	0,04
R4	Se Y_{t-1} é D	D	0,62
R5	Se Y_{t-7} é MD	D	0,18
R6	Se Y_{t-2} é D	D	0,20
R7	Se Y_{t-7} é MD	MD	0,76
R8	Se Y_{t-3} é ED	MD	0,24
R9	Se Y_{t-2} é MD e Y_{t-7} é PD	ED	0,67
R10	Se Y_{t-2} é ED	ED	0,18
R11	Se Y_{t-1} é ED	ED	0,15

Com respeito à Tabela 5.63, é possível avaliar o total de 11 regras, que em sua maioria são descritas por um elemento como termo antecedente. Em

Tabela 5.64: Base de Regras Fuzzy com representação por ligação de estados linguísticos.

Consequente	Antecedente				
	MPD	PD	D	MD	ED
MPD	Y_{t-1}				
PD		Y_{t-1}, Y_{t-12}	Y_{t-6}		
D			Y_{t-1}, Y_{t-2}	Y_{t-7}	
MD				Y_{t-7}	Y_{t-3}
ED		Y_{t-7}		Y_{t-2}	Y_{t-1}, Y_{t-2}

geral, o modelo revela uma maior predominância do passado mais recente (Y_{t-1} e Y_{t-2}) como fator de maior peso na determinação do comportamento atual da série (característica inercial). De forma geral a natureza autorregressiva dos estados linguísticos é predominante, ou seja, a parte antecedente e a consequente compartilham o mesmo rótulo linguístico em 8 das 11 regras. Isto é verificado facilmente pela Tabela 5.64 que apresenta a ligação dos estados anteriores dos termos linguísticos com os instantes atuais.

Verifica-se uma predominância dos elementos na diagonal principal e nas subdiagonais, como em um formato semelhante a matrizes esparsas. Esse comportamento denota que não há ligações efetivas entre os rótulos linguísticos distantes, por exemplo, MPD com ED. Isto pode ser explicado por dois fatores: (i) natureza do processo fracamente sazonal (como verificado no início do tópico) e (ii) comportamento comum de agregados econômicos relevantes para a conjuntura econômica. Como dentro do período de um ano é comum séries temporais, por exemplo físicas, atingir valores baixos, médios e elevados seguidamente. Contudo como a série em si possui pouco efeito sazonal, além de ser parcialmente administrada pelo governo, as suas flutuações são pequenas e desejavelmente com transições lentas. Estes dois fatos implicam em uma matriz de ligação entre os estados linguísticos mais esparsa, diferentemente, por exemplo, de uma série de velocidade de vento ou vazão hidráulica, que apresentariam uma matriz de ligação com um formato mais cheio.

5.4

Controle

Em problemas de controle, o número de SFGs presentes na literatura é relativamente superior ao observado nas áreas de Regressão e Previsão. Contudo, grande parte da literatura não fornece uma boa base de comparação entre modelos. Conjectura-se que um dos motivos esteja associado a necessidade de em problemas de controle efetuarem-se simulações que, para tanto, demandam um tempo de máquina razoável, tornando-se mais dispendiosa no âmbito dos SFGs. Como existe uma população de soluções e esta deve ser sintetizada ao longo de várias iterações, o processo como um todo para geração de uma solução torna-se oneroso, pois para atribuir a cada indivíduo sua avaliação requer-se uma simulação.

Apesar desta dificuldade, uma investigação empírica foi dirigida de modo a apresentar os diferentes comportamentos do modelo GPFIS quando submetido a certas parametrizações em problemas de controle. Para tanto, é adotado o benchmark Cart-Centering [2, 205], também denominado de Parada Mínima ou de Centralização do Carro. Este problema é relativamente simples e torna factível efetuar um conjunto de experimentos com o propósito de elucidar as configurações mais eficientes para este problema. Entretanto, o procedimento estatístico empregado nas seções anteriores não o é agora.

O próximo tópico exhibe o delineamento experimental efetuado para a investigação empírica. A melhor configuração obtida é usada no benchmark do Pêndulo Invertido [133, 202].

5.4.1

Investigação Empírica da Arquitetura do Modelo GPFIS

Descrição dos Experimentos

O benchmark Cart-Centering consiste de um carro com massa m que, em um instante t qualquer, está na posição x_t , com uma velocidade v_t sobre um trilho de atrito desprezível. Institui-se um ponto de referência ($x_t = 0$, por exemplo) no qual o carro deve ser parado, isto é, o carro deve chegar em $x_t = 0$ com $v_t = 0$. A Figura 5.22 ilustra a ideia disposta. Costuma-se admitir uma tolerância ϵ tanto para a posição quanto para a velocidade, de modo que se considera que o carrinho está estacionado se $|x_t - ref_x| < \epsilon$ e $|v_t - ref_v| < \epsilon$, onde ref_x e ref_v representam a referência estabelecida pelo operador da planta.

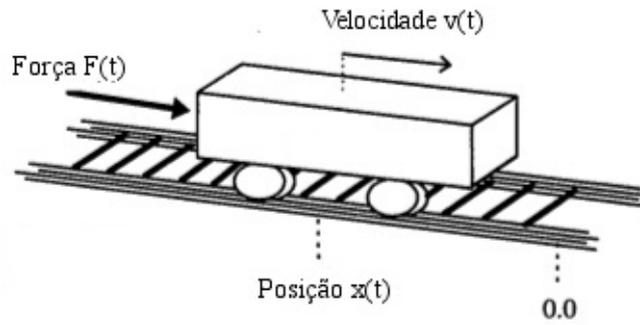


Figura 5.22: Ilustração do problema Cart-Centering.

A dinâmica do sistema é dada por:

$$v_{t+\tau} = v_t + \tau \frac{F_t}{m} \quad (5-9)$$

$$x_{t+\tau} = x_t + \tau v_t \quad (5-10)$$

onde τ é o passo de amostragem e F_t é a força aplicada no carro em t . O intuito é aplicar consecutivos empurrões F_t no carro de maneira a centralizá-lo na referência o mais rapidamente possível. Alba et al. [2], seguindo os parâmetros executados por Thrift [197], propõe o uso da PG para encontrar uma base de regras fuzzy para um controlador fuzzy neste problema.

De modo a comparar os resultados obtidos com o estudo de Alba et al. [2] e empreender uma investigação empírica, a Tabela 5.65 mostra os domínios das variáveis e dos parâmetros ali utilizados.

Tabela 5.65: Domínio e valor das variáveis e parâmetros para os testes realizados.

Variável	Domínio
F_t	$[-2,5 \ 2,5]$ N
v_t	$[-2,5 \ 2,5]$ m/s
x_t	$[-2,5 \ 2,5]$ m
Parâmetro	Valor
τ	0,02s
ϵ	0,5
m	2,0 kg
ref_x	$x_t = 0$
ref_v	$v_t = 0$

Portanto, o sistema deve conduzir o carro até $|x_t - 0| < 0,5$ e $|v_t - 0| < 0,5$. São dadas 16 condições iniciais com valores uniformemente espaçados no domínio de x_t , considerando $v_t = 0$. A função de avaliação da PGMG que deve ser minimizada é dada por:

$$Aval_{GPFIS} = t_{\epsilon} + \frac{\sum_{t=1}^T |x_t|}{T} \quad (5-11)$$

onde t_{ϵ} é o tempo necessário para satisfazer o critério de parada ($|x_t-0| < 0,5$ e $|vt-0| < 0,5$). Um indivíduo do modelo GPFIS é considerado não apto se, em algum caso de treinamento, não conseguir estacionar o carro em menos de 10 segundos. Após o ajuste do controlador, submetem-se ao melhor indivíduo 1000 posições aleatórias, visando à avaliar qual o tempo médio que o controlador GPFIS tomou para conseguir centralizar os carros nas 1000 situações. Estes resultados são reportados ao longo das discussões na próxima seção.

A Tabela 5.66 exibe o delineamento do experimento realizado.

Tabela 5.66: Delineamento experimental fatorial para o modelo GPFIS.

Descrição	Experimento
Hipótese Avaliada	Operadores Fuzzy:Operadores de Agregação:Defuzzificação
Configurações	1- Prod:Máximo:Altura
	2- Prod:Máximo:Altura Média dos Máximos
	3- Prod+RQ:Máximo:Altura
	4- Prod+RQ:Máximo:Altura Média dos Máximos
	5- Prod+RQ+Neg:Máximo:Altura
	6- Prod+RQ+Neg:Máximo:Altura Média dos Máximos
	7- Prod:Média:Altura
	8- Prod:Média:Altura Média dos Máximos
	9- Prod+RQ:Média:Altura
	10- Prod+RQ:Média:Altura Média dos Máximos
	11- Prod+RQ+Neg:Média:Altura
	12- Prod+RQ+Neg:Média:Altura Média dos Máximos

Legenda: o símbolo “:” representa interação.

O desenho é diferente dos anteriores, pois neste caso é empregado o conceito de experimento fatorial, isto é, efetua-se a combinação de todas as configurações sob análise. Como há três tipos de operadores fuzzy (Prod, Prod+RQ e Prod+RQ+Neg), dois operadores de agregação (Máximo e Média) e dois métodos de defuzzificação (Altura e Média dos Máximos), há então doze configurações possíveis.

A Tabela 5.67 apresenta as demais configurações do modelo GPFIS e a Figura 5.23 mostra o formato e número das funções de pertinência, mantidas intactas ao longo de todo o processo de sintetização do modelo. Seguindo as recomendações de Alba et al. [2], são efetuadas 25000 avaliações no total para cada uma das 10 execuções realizadas em cada configuração investigada.

Tabela 5.67: Principais configurações do modelo GPFIS.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	250
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.23
Operadores Fuzzy	Tabela 5.66

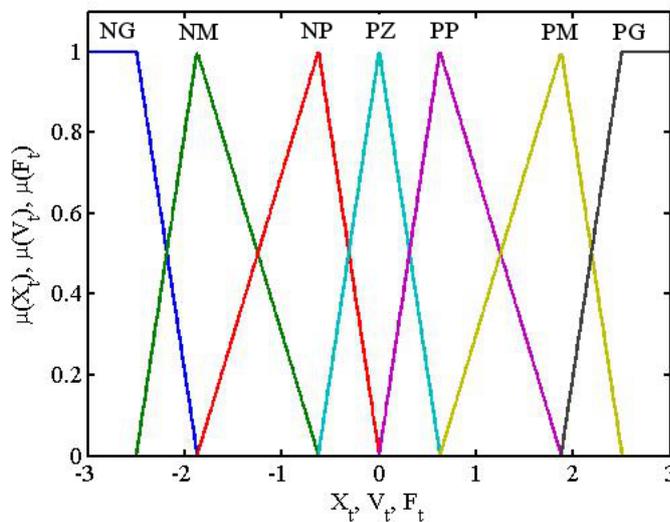


Figura 5.23: Funções de pertinência usadas no modelo GPFIS.

Resultados e Discussões

A Tabela 5.68 apresenta os resultados obtidos durante as simulações mediante a utilização do operador de agregação do Máximo, enquanto que a Tabela 5.69 exhibe aqueles referentes à agregação pela Média Aritmética.

Observa-se que em praticamente todas as configurações o uso da Média Aritmética como operador de agregação reduziu em média 39% o tempo médio necessário para o controlador centralizar o carrinho na posição $|x_t - 0| < 0,5$ e $|v_t - 0| < 0,5$. Cabe ressaltar que o método de defuzzificação Altura Média dos Máximos reduz em média o tempo em 14%. Contudo, de forma contrária ao observado nos problemas anteriores, o uso do operador de negação não apresenta um efeito tão positivo, apesar de, nestas configurações, poucas regras terem sido geradas.

Tabela 5.68: Resultados do modelo GPFIS para o Cart-Centering com operação de agregação pelo Máximo.

Atributo	Máximo		
	Prod	Prod+RQ	Prod+RQ+Neg
Média de Passos	230,4	215,9	224,6
Desvio Padrão de Passos	22,3	25,7	37,8
Tempo médio (0,02s/amostragem)	4,608	4,318	4,492
Número Médio de Regras	15	21	14
Atributo	Altura	Média dos	Máximos
	Prod	Prod+RQ	Prod+RQ+Neg
Média de Passos	213,5	243,6	203,5
Desvio Padrão de Passos	54,6	94,1	60,8
Tempo médio (0,02s/amostragem)	4,27	4,872	4,070
Número Médio de Regras	18	24	15

A melhor configuração obtida é:

- Operadores Fuzzy: Produto e Raiz Quadrada.
- Operador de Agregação: Média Aritmética.
- Defuzzificação: Altura Média dos Máximos.

A Figura 5.24 exibe 16 posições iniciais e as finais do melhor indivíduo. Observa-se que todas estão no intervalo $|x_t-0| < 0.5$ e, apesar de adentrarem a mesma por volta do passo 100, é necessário que a segunda tolerância $|v_t-0| < 0.5$ seja estabelecida.

Por fim, a Figura 5.25 apresenta a superfície de resposta da melhor configuração do modelo GPFIS, considerando o método de defuzzificação (a): Altura e (b): Altura Média dos Máximos. Verifica-se que a superfície de (a) é mais suave do que a de (b), devido a maior gama de possíveis valores que F_t pode assumir.

5.4.2

Comparação com Demais Sistemas Fuzzy-Genéticos

Cart-Centering

A comparação dos resultados obtidos pela melhor configuração do modelo GPFIS reside na confrontação com outros dois SFGs e com a solução ótima do problema Cart-Centering. Note que o número de avaliações foi escolhido de modo a tornar esta comparação comensurável. Apresentam-se a seguir os dois métodos:

Tabela 5.69: Resultados do modelo GPFIS para o Cart-Centering com operação de agregação pela Média Aritmética.

Atributo	Média Aritmética		
	Prod	Prod+RQ	Prod+RQ+Neg
Média de Passos	165,7	160,2	205,5
Desvio Padrão de Passos	20,4	18,9	38,9
Tempo médio (0,02s/amostragem)	3,314	3,204	4,110
Número Médio de Regras	30	27	26
Atributo	Altura	Média dos	Máximos
	Prod	Prod+RQ	Prod+RQ+Neg
Média de Passos	139,5	135,8	144,9
Desvio Padrão de Passos	15,6	18,9	11,4
Tempo médio (0,02s/amostragem)	2,790	2,716	2,890
Número Médio de Regras	26	28	24

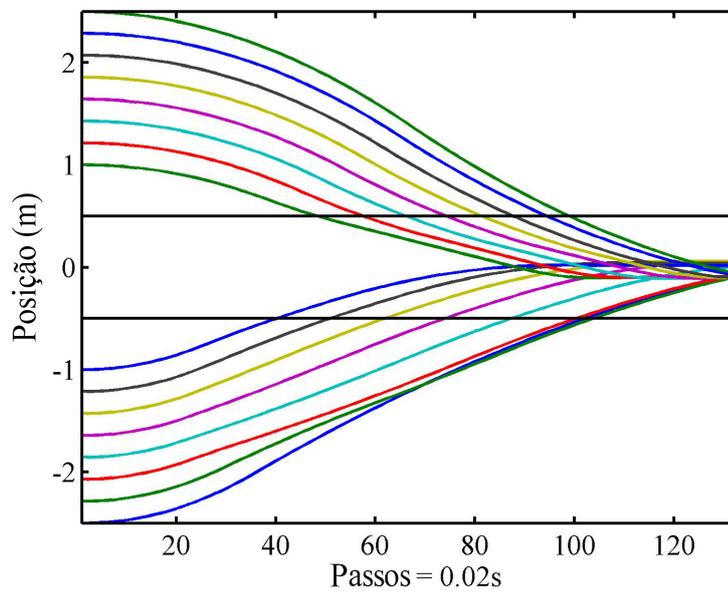


Figura 5.24: Posição inicial e final do melhor indivíduo nas execuções do GP-FIS, usando a configuração Product+RQ:Média:Altura Média dos Máximos.

- Thrift [197]: uma dos pioneiros em SFGs. O autor propõe um SFGBR do tipo Pittsburgh para problemas de controle, usando para tal um AG. A formulação da proposta é a mais básica e clássica.
- Alba et al. [2]: uma das primeiras implementações de SFGs baseados em PG. Os autores propõe um SFGBR do tipo Pittsburgh, e de forma similar a Thrift [197], não empregam novos operadores de recombinação ou ajuste de funções de pertinência.
- Solução Ótima: elaborada por métodos de Controle Ótimo.

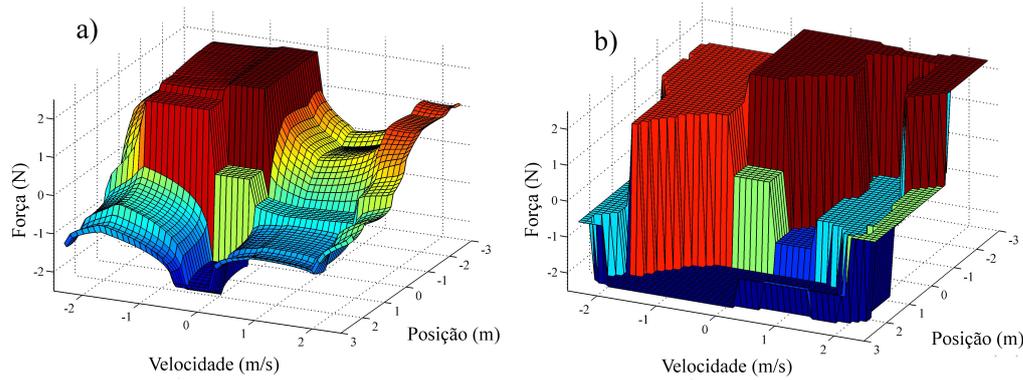


Figura 5.25: Superfície de resposta do melhor indivíduo para o Cart-Centering, usando diferentes métodos de defuzzificação: a) Altura, b) Altura Média dos Máximos.

A média de passos efetuadas pelo modelo GPFIS (135,8 passos) é inferior à de Alba et al. [2] (158 passos) e de Thrift [197] (149 passos). A solução por Controle Ótimo é de 129 passos.

Pêndulo Invertido

Descrição dos Experimentos

O segundo experimento se baseia no benchmark denominado Pêndulo Invertido. Este problema é denotado por um carro com massa M e um pêndulo de massa m e altura λ preso ao centro da superfície do carro. O carro se move em duas possíveis direções (para a frente e para trás) em um trilho de atrito desprezível. O controlador deve aplicar consecutivas F_t de modo a aumentar ou reduzir v_t e, conseqüentemente, a velocidade angular ω_t e o ângulo do pêndulo θ_t . A Figura 5.26 apresenta uma ilustração do problema do Pêndulo Invertido.

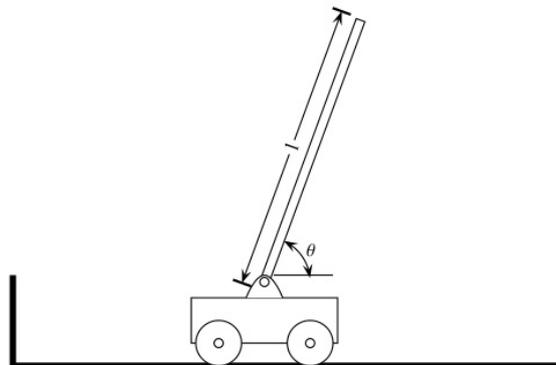


Figura 5.26: Ilustração do Pêndulo Invertido.

Koza [133] apresenta a dinâmica do pêndulo invertido:

$$\phi_t = \frac{g \sin \theta_t + \cos \theta_t \frac{-F_t - m\lambda\omega_t\theta_t^2 \sin \theta}{M+m}}{\lambda \left[\frac{4}{3} - \frac{m \cos^2 \theta}{M+m} \right]} \quad (5-12)$$

onde g e ϕ_t são a aceleração da gravidade e a angular, respectivamente. A partir de ϕ_t , e considerando o passo de tempo como τ , tem-se a equação recursiva de ω_t , usando o método de Euler:

$$\omega_{t+1} = \omega_t + \tau\phi_t \quad (5-13)$$

$$\theta_{t+1} = \theta_t + \tau\omega_t \quad (5-14)$$

a aceleração a_t do carrinho sobre o trilho é dada por:

$$a_t = \frac{F_t + m\lambda[\theta^2 \sin \theta - \omega_t \cos \theta]}{M + m} \quad (5-15)$$

por conseguinte, a partir da aceleração é possível atualizar a velocidade e a posição do carro:

$$v_{t+1} = v_t + \tau a_t \quad (5-16)$$

$$x_{t+1} = x_t + \tau v_t \quad (5-17)$$

a partir desta formulação é possível simular a trajetória do pêndulo, como também a posição do carro ao longo do trilho. Diversos autores utilizam o Pêndulo Invertido como benchmark para apresentar a qualidade de diferentes SFGs para controle [100, 165, 172]. Contudo, este trabalho utiliza o estudo de Tsakonas [202], que propõe um SFG baseado em PG Memética, a partir de uma formulação do SIF tipo TSK. Os motivos que levam a essa escolha deve-se ao trabalho ser bastante recente e baseado em PG.

De modo a efetuar uma comparação comensurável com Tsakonas [202], utilizam-se os domínios de cada variável e os valores dos parâmetros como mostrados na Tabela 5.70. Em Tsakonas [202] são consideradas duas condições iniciais: $\theta_0 = \{-0,18, 0,18\}rad$, com $\omega_0 = \{0,0\}rad/s$. O tempo permitido para que o pêndulo seja centrado na origem na posição $|\theta_t-0| < 0.01$, com $|\omega_t-0| < 0.01$ é de no máximo 1 segundo (100 passos de tempo).

Como em Tsakonas [202], 100000 avaliações foram realizadas; todo este procedimento foi executado 10 vezes. A Tabela 5.71 exhibe os demais

parâmetros, enquanto que a Figura 5.27 mostra as funções de pertinência para cada variável. Por fim, a equação (5-18) exhibe a função de avaliação (escolhida de forma similar a Tsakonas [202]):

$$Aval_{GPFIS} = \sum_{t=1}^T (\theta_t - ref_{\theta}) = \sum_{t=1}^T \theta^2 \quad (5-18)$$

Tabela 5.70: Domínio das variáveis e valor dos parâmetros para os testes efetuados.

Variável	Domínio
F_t	[-25 25] N
θ_t	[-0,34 0,34]rad
ω_t	[-0,87 0,87]rad/s
v_t	sem restrições
x_t	sem restrições
Parâmetro	Valor
τ	0,01s
ϵ	0,01
m	0,1 kg
M	1,0 kg
λ	0,5m
ref_{θ}	$\theta_t = 0$
ref_{ω}	$\omega_t = 0$

Tabela 5.71: Principais configurações do modelo GPFIS.

Parâmetro	Valor
Tamanho da população	100
Número de gerações	1000
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim
Conjuntos Fuzzy de Entrada	Figura 5.27
Operadores Fuzzy	Tabela 5.66
Operador de Agregação	Média Aritmética
Método de Particionamento	Divisão Uniforme
Defuzzificação	Altura Média dos Máximos

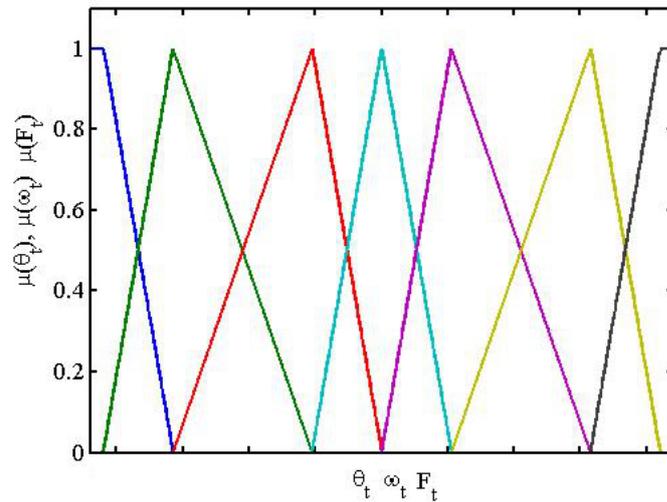


Figura 5.27: Funções de pertinência usadas no modelo GPFIS.

Resultados e Discussões

A Figura 5.28 mostra o comportamento do melhor indivíduo nas 100000 avaliações disponíveis, dadas duas condições iniciais: $\theta_0 = \{-0,18, 0,18\}rad$, com $\omega_0 = \{0,0\}rad/s$. A média dos melhores indivíduos nas 10 execuções foi de 0,27 segundos para se estabelecer em $|\theta_t - 0| < 0,01$ durante o período de 1 segundo, gerando um total de 14 regras em média. Em Tsakonas [202] o SFG proposto para a mesma tarefa levou 0,61 segundos para satisfazer os requerimentos, entretanto produzindo somente 7 regras.

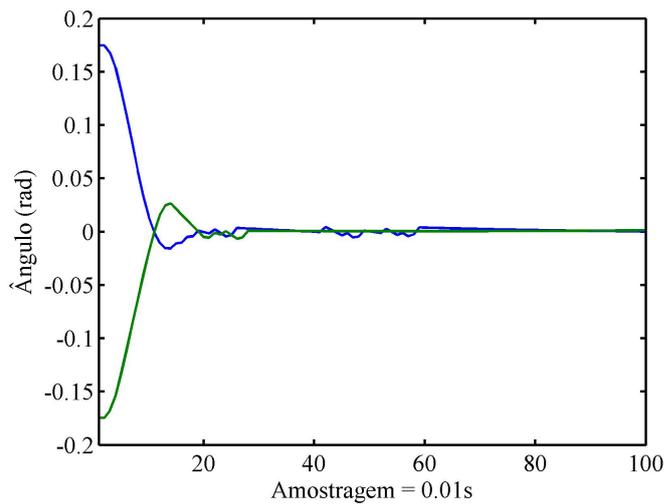


Figura 5.28: Solução elaborada pelo melhor indivíduo nas execuções do GPFIS.

5.5

Resumo

Este capítulo exibiu as principais análises empíricas do modelo GPFIS, concernente aos problemas que sua arquitetura genérica possibilita alcançar: Classificação, Previsão, Regressão e Controle. As investigações empíricas proporcionaram a avaliação dos diferentes componentes do mecanismo de Inferência do modelo GPFIS (Formulação-Particionamento-Agregação). Na maioria dos casos, o uso da negação clássica possibilitou o aprimoramento dos resultados, em termos de acurácia e redução da complexidade da base de regras fuzzy. O operador de agregação MQR, apesar do aumento do tempo computacional, possibilitou intensos ganhos em acurácia e interpretabilidade, assim como o método de particionamento CD ou FCD. Em controle foi possível verificar que a troca do método de defuzzificação (Altura para Altura Média dos Máximos) viabilizou aumentar o desempenho do controlador, ao restringir o universo de valores disponíveis para a defuzzificação.

Por fim, além das investigações empíricas desenvolvidas para cada problema, comparações com outros SFGs e modelos e aplicações reais foram realizadas. Estas exibiram um comparativo do modelo GPFIS frente ao estado da arte na literatura, assim como reforçaram a sua posição de modelo que busca atingir altos níveis de acurácia, mas com manutenção da interpretabilidade da solução final. A partir disso, o próximo capítulo finaliza a dissertação, ao apresentar as Conclusões e Trabalhos Futuros.

6 Conclusões

Esta dissertação abordou o desenvolvimento do modelo Genetic Programming Fuzzy Inference System (GPFIS). Foram exibidas suas principais características, tais como (i) ser baseado em Programação Genética Multigênica; (ii) possibilitar o uso de operadores de agregação, negação e modificadores linguísticos de forma simplificada; (iii) empregar heurísticas de definição do consequente mais apropriado para uma parte antecedente; e (iv) induzido pela disposição das funções de pertinência, aproveitar o procedimento de defuzzificação pelo método da altura.

Durante as investigações empíricas fica claro que cada componente apontado anteriormente é parte essencial para a acurácia e compacidade do modelo proposto. A partir dos ganhos proporcionados por cada componente foi possível auferir resultados em benchmarks superiores aos de outros Sistemas Fuzzy-Genéticos e modelos presentes na literatura. As aplicações detalhadas tiveram como objetivo ilustrar o funcionamento do modelo e evidenciar a sua qualidade.

O modelo GPFIS abre diversas frentes para trabalhos futuros. A seguir são sugeridas algumas linhas, ordenando-as de forma análoga à arquitetura do modelo GPFIS: Fuzzificação, Inferência, Defuzzificação e Avaliação.

- **Funções de Pertinência:** em todos os experimentos foi usada a formulação 2-justaposta e uniformemente distribuída para as funções de pertinência triangulares. Contudo, é sempre possível alterar esta formulação, visando a ganhos em acurácia ao usar mecanismos como a representação em pares [4], ou, em uma fase pós-processamento, usar uma meta-heurística auxiliar para ajustar as funções de pertinência [5, 187]. Este caminho onera a acepção linguística do sistema, mas, caso a aplicação não demande descoberta de conhecimento, mas sim desempenho - por exemplo, na área de previsão - esta opção pode ser facilmente incorporada à modelagem proposta.
- **Operadores Fuzzy:** ao longo das aplicações apresentadas o conjunto de operadores fuzzy ficou restrito a: produto (t-norma e concentrador

linguístico), raiz-quadrada (dilatador linguístico) e negação clássica. Entretanto é sempre possível investigar o efeito de outras t-normas, modificadores linguísticos e negações. Embora haja algum prejuízo em termos de interpretabilidade, t-conormas também podem ser empregadas. Apesar de não relatado, em alguns testes na área de controle, a inclusão do operador soma probabilística possibilitou um melhora dos resultados.

- **Operadores de Agregação:** dos operadores investigados, o baseado em Mínimos Quadrados Restritos (MQR) proveu o melhor desempenho, em termos de acurácia e compacidade da base de regras fuzzy. Porém, houve reflexo na elevação do tempo de máquina. Há a possibilidade de se investigarem outros operadores que sejam mais efetivos em determinadas situações, como em classificação. Abrem-se, assim, duas frentes:
 - *Substituir o MQR:* como o MQR é uma combinação linear de pesos no intervalo $[0, 1]$, talvez seja possível substituir a forma de estimar estes pesos, sem a necessidade de se resolver uma sequência de sistemas lineares. Para tanto, pode-se ponderar as regras pelo seu grau de identidade com o respectivo consequente. Definir uma função que mensura esse grau de identidade (por exemplo, o produto interno) pode ser fruto de uma investigação mais detalhada.
 - *Investigar outros Operadores:* como descrito, o MQR é uma combinação linear. Poder-se-ia empregar uma combinação não-linear, como, por exemplo, usar um modelo de regressão logística (semelhantemente ao perceptron com função de ativação logística). O modelo de regressão logística é uma escolha comum em Estatística quando se pretende aliar acurácia e interpretabilidade em problemas de classificação. Ainda, é possível buscar uma expressão similar ao MQR substituindo a métrica do erro quadrático médio pela Entropia Cruzada [180]. Outro caminho seria, em vez de buscar uma média ponderada (caso do MQR), selecionar os pesos por meio de uma média geométrica ponderada. Por fim, outros modelos não-lineares podem ser usados, embora o custo computacional deva ser sempre levado em conta.
- **Métodos de Particionamento:** existem dois caminhos passíveis nesse sentido: (i) investir em novas heurísticas que definam o consequente mais apropriado; (ii) propor um método de particionamento (e também de agregação) adequado para problemas de controle. Com respeito ao primeiro, como os métodos propostos se baseiam na noção euclidiana de

produto interno e norma, é sempre viável mudar a forma de perceber distâncias e ângulos formados entre as ativações. Uma possibilidade é a inclusão de uma matriz de ponderação que reflita, por exemplo, a importância do reconhecimento de um determinado subconjunto de padrões pelo modelo. Sobre o segundo item, torna-se necessário o estudo de métodos adaptativos de estimação, para adequadamente fornecer um conjunto de ferramentas para o modelo GPFIS neste sentido.

- **Programação Genética:** durante o trabalho não foi efetuado um ajuste fino nos parâmetros da Programação Genética. Tanto que é fácil observar a repetição das taxas de recombinação, tamanho da população, etc. nos diferentes experimentos. Atuou-se desta forma para demonstrar que é possível obter bons resultados a partir de uma configuração padrão. Um ajuste fino poderia aprimorar ainda mais os resultados.
- **Defuzzificação:** a maior parte dos resultados do modelo GPFIS foram baseados no método de defuzzificação da Altura. Esta foi motivada inicialmente pela proposição apresentada no início do Capítulo 4, que obtém a decomposição de um valor em uma combinação linear de conjuntos fuzzy, triangulares, uniformemente distribuídos e 2-justapostos. Cabe a pergunta: seria esta proposição um corolário de um teorema? Isto é, pode ser possível remover a questão da triangularidade das funções de pertinência, ou a 2-justaposição pode passar para uma 3-justaposição ou J-justaposição, entre outros. Este tópico merece um exame mais aprofundado e pode revelar outros métodos de fuzzificação/defuzzificação que possibilitam auferir as mesmas propriedades do resultado obtido pela proposição enunciada.

Todos os itens acima podem contribuir para o aperfeiçoamento do modelo GPFIS e possibilitar ganhos em outros modelos na área de Sistemas Fuzzy-Genéticos. Existe sempre a possibilidade de expandir os experimentos realizados fazendo uso de outros benchmarks e comparações com outros Sistemas Fuzzy-Genéticos presentes na literatura. É possível buscar uma formulação do modelo GPFIS para atividades não-supervisionadas (análise de agrupamentos, por exemplo) e ampliar as aplicações reais para a área médica, séries financeiras e robótica, por exemplo.

Referências Bibliográficas

- [1] ABRAHAM, A. **Neuro fuzzy systems: State-of-the-art modeling techniques**. In: Mira, J.; Prieto, A., editors, CONNECTIONIST MODELS OF NEURONS, LEARNING PROCESSES, AND ARTIFICIAL INTELLIGENCE, volume 2084 de **Lecture Notes in Computer Science**, p. 269–276. Springer Berlin Heidelberg, 2001.
- [2] ALBA, E.; COTTA, C. ; TROYA, J. M. **Type-constrained genetic programming for rule-base definition in fuzzy logic controllers**. In: PROCEEDINGS OF THE FIRST ANNUAL CONFERENCE ON GENETIC PROGRAMMING, GECCO '96, p. 255–260, Cambridge, MA, USA, 1996. MIT Press.
- [3] ALCALÁ-FDEZ, J.; FERNANDEZ, A.; LUENGO, J.; DERRAC, J.; GARCÍA, S.; SÁNCHEZ, L. ; HERRERA, F. **Journal of Multiple-Valued Logic and Soft Computing**. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework, journal, v.17, n.2, p. 255–287, 2011.
- [4] ALCALÁ, R.; ALCALÁ-FDEZ, J.; HERRERA, F. ; OTERO, J. **International Journal of Approximate Reasoning**. Genetic learning of accurate and compact fuzzy rule based systems based on the 2-tuples linguistic representation, journal, v.44, n.1, p. 45–64, 2007.
- [5] ALCALÁ, R.; GACTO, M. J. ; HERRERA, F. **Fuzzy Systems, IEEE Transactions on**. A fast and scalable multiobjective genetic fuzzy system for linguistic fuzzy modeling in high-dimensional regression problems, journal, v.19, n.4, p. 666–681, 2011.
- [6] ALCALÁ, R.; NOJIMA, Y.; HERRERA, F. ; ISHIBUCHI, H. **Soft Computing**. Multiobjective genetic fuzzy rule selection of single granularity-based fuzzy classification rules and its interaction with the lateral tuning of membership functions, journal, v.15, n.12, p. 2303–2318, 2011.

- [7] ALCALÁ, R.; NOJIMA, Y.; ISHIBUCHI, H. ; HERRERA, F. **International Journal of Computational Intelligence Systems**. Special issue on evolutionary fuzzy systems, journal, v.5, n.2, p. 209–211, 2012.
- [8] ALCALÁ, R.; NOJIMA, Y.; ISHIBUCHI, H. ; HERRERA, F. **Knowledge-Based Systems**. Special issue on “evolutionary fuzzy systems” {EFSs}, journal, v.54, n.0, p. 1–2, 2013.
- [9] ALCALÁ-FDEZ, J.; ALCALÁ, R. ; HERRERA, F. **Fuzzy Systems, IEEE Transactions on**. A fuzzy association rule-based classification model for high-dimensional problems with genetic rule selection and lateral tuning, journal, v.19, n.5, p. 857–872, 2011.
- [10] ALCALÁ-FDEZ, J.; HERRERA, F.; MÁRQUEZ, F. ; PEREGRÍN, A. **International Journal of Intelligent Systems**. Increasing fuzzy rules cooperation based on evolutionary adaptive inference systems, journal, v.22, n.9, p. 1035–1064, 2007.
- [11] ALONSO, J. M.; MAGDALENA, L. **Information Sciences**. Special issue on interpretable fuzzy systems, journal, v.181, n.20, p. 4331–4339, 2011.
- [12] ALSINA, C.; FRANK, M. J. ; SCHWEIZER, B. **Associative Functions: Triangular Norms and Copulas**. Singapore: World Scientific, 2006.
- [13] ANGELOV, P.; BUSWELL, R. **Fuzzy Systems, IEEE Transactions on**. Identification of evolving fuzzy rule-based models, journal, v.10, n.5, p. 667–677, 2002.
- [14] ANTONELLI, M.; DUCANGE, P.; LAZZERINI, B. ; MARCELLONI, F. **International Journal of Approximate Reasoning**. Learning concurrently partition granularities and rule bases of mamdani fuzzy systems in a multi-objective evolutionary framework, journal, v.50, n.7, p. 1066–1080, 2009.
- [15] ANTONELLI, M.; DUCANGE, P. ; MARCELLONI, F. **International Journal of Approximate Reasoning**. An efficient multi-objective evolutionary fuzzy system for regression problems, journal, v.54, n.9, p. 1434–1451, 2013.

- [16] ARSLAN, A.; KAYA, M. **Fuzzy Sets and Systems**. Determination of fuzzy logic membership functions using genetic algorithms, journal, v.118, n.2, p. 297–306, 2001.
- [17] ASSIMAKOPOULOS, V.; NIKOLOPOULOS, K. **International Journal of Forecasting**. The theta model: a decomposition approach to forecasting, journal, v.16, n.4, p. 521–530, 2000.
- [18] AYDOGAN, E. K.; KARAOGLAN, I. ; PARDALOS, P. M. **Applied Soft Computing**. hga: Hybrid genetic algorithm in fuzzy rule-based classification systems for high-dimensional problems, journal, v.12, n.2, p. 800–806, 2012.
- [19] AZNARTE, J. L.; ALCALÁ-FDEZ, J.; ARAUZO-AZOFRA, A. ; BENÍTEZ, J. M. **Expert Systems with Applications**. Financial time series forecasting with a bio-inspired fuzzy model, journal, v.39, n.16, p. 12302–12309, 2012.
- [20] BACHE, K.; LICHMAN, M. **UCI machine learning repository**, 2013.
- [21] BARRENECHEA, E.; BUSTINCE, H.; FERNANDEZ, J.; PATERNAIN, D. ; SANZ, J. A. **Axioms**. Using the choquet integral in the fuzzy reasoning method of fuzzy rule-based classification systems, journal, v.2, n.2, p. 208–223, 2013.
- [22] BARROS, L. C.; BASSANEZI, R. C. **Tópicos de lógica fuzzy e biomatemática**. Campinas: Unicamp-Imecc, 2006.
- [23] BASTIAN, A. **Fuzzy Sets and Systems**. Identifying fuzzy models utilizing genetic programming, journal, v.113, n.3, p. 333–350, 2000.
- [24] BATISTA-JUNIOR, P. N. **Estudos Avançados**. Vulnerabilidade externa da economia brasileira, journal, v.16, n.45, p. 173–185, 2002.
- [25] BELIAKOV, G.; PRADERA, A. ; CALVO, T. **Aggregation functions: A guide for practitioners**. Heidelberg: Springer Publishing Company, 2008.
- [26] BENÍTEZ, A. D.; CASILLAS, J. **Soft Computing**. Multi-objective genetic learning of serial hierarchical fuzzy systems for large-scale problems, journal, v.17, n.1, p. 165–194, 2013.

- [27] BERGMEIR, C.; TRIGUERO, I.; MOLINA, D.; AZNARTE, J. L. ; BENITEZ, J. M. **IEEE Trans on Neural Networks and Learning Systems**. Time series modeling and forecasting using memetic algorithms for regime-switching models, journal, v.23, n.11, p. 1841–1847, 2012.
- [28] BERLANGA, F.; RIVERA, A.; DEL JESUS, M. ; HERRERA, F. **Inf. Sci.** Gp-coach: Genetic programming-based learning of compact and accurate fuzzy rule-based classification systems for high-dimensional problems, journal, v.180, n.8, p. 1183–1200, 2010.
- [29] BERLANGA, F. J.; JESUS, M. J.; GACTO, M. J. ; HERRERA, F. **A genetic-programming-based approach for the learning of compact fuzzy rule-based classification systems**. In: Rutkowski, L.; Tadeusiewicz, R.; Zadeh, L. A. ; Zurada, J. M., editors, **ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING – ICAISC 2006**, volume 4029 de **Lecture Notes in Computer Science**, p. 182–191. Springer Berlin Heidelberg, 2006.
- [30] BRAMEIER, M.; BANZHAF, W. **Genetic Programming and Evolvable Machines**. Evolving teams of predictors with linear genetic programming, journal, v.2, n.4, p. 381–407, 2001.
- [31] BRAMEIER, M.; BANZHAF, W. **Linear Genetic Programming**. Número XVI em Genetic and Evolutionary Computation. Springer, 2007.
- [32] BRESSER-PEREIRA, L. C. **Exchange rate, fix, float or manage it?** In: Vernengo, M., editor, **FINANCIAL INTEGRATION OR DOLLARIZATION: NO PANACEA**, p. prefacio. Edward Elgar, 2006.
- [33] BRITO, V. B.; VELLASCO, M. M. B. R. ; TANSCHKEIT, R. **Learning and Nonlinear Models**. Fuzzyfuture: ferramenta de previsão de séries temporais baseada em sistema híbrido fuzzy-genético, journal, v.10, p. 99–108, 2012.
- [34] BUSCH, J. M.; FORTNEY, W. G. ; BERRY, L. N. **Society of Petroleum Engineers Formation Evaluation**. Determination of lithology from well logs by statistical analysis, journal, v.2, p. 412–418, 1987.
- [35] CARSE, B.; FOGARTY, T. C. ; MUNRO, A. **Fuzzy Sets and Systems**. Evolving fuzzy rule based controllers using genetic algorithms, journal, v.80, n.3, p. 273–293, 1996.

- [36] CARVALHO, J. P.; BATISTA, F. ; COHEUR, L. **A critical survey on the use of fuzzy sets in speech and natural language processing.** In: FUZZY SYSTEMS (FUZZ-IEEE), 2012 IEEE INTERNATIONAL CONFERENCE ON, p. 1–8, 2012.
- [37] CASILLAS, J.; CARSE, B. ; BULL, L. **Fuzzy Systems, IEEE Transactions on.** Fuzzy-xcs: A michigan genetic fuzzy system, journal, v.15, n.4, p. 536–550, 2007.
- [38] CASILLAS, J.; CORDON, O.; DEL-JESUS, M. J. ; HERRERA, F. **IEEE Transactions on Fuzzy Systems.** Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction, journal, v.13, n.1, p. 13–29, 2005.
- [39] CASILLAS, J.; CORDON, O.; DEL-JESUS, M. J. ; HERRERA, F. **IEEE Transactions on Fuzzy Systems.** Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction, journal, v.13, n.1, p. 13–29, 2005.
- [40] CASILLAS, J.; MARTÍNEZ, P. ; BENÍTEZ, A. D. **Soft Computing.** Learning consistent, complete and compact sets of fuzzy rules in conjunctive normal form for regression problems, journal, v.13, n.5, p. 451–465, 2009.
- [41] CASTAGNA, J. P.; BATZLE, M. L. ; KAN, T. K. **Investigations in Geophysics.** Rock physics: The link between rock properties and avo response, journal, v.8, p. 135–171, 1993.
- [42] CASTILLO, O.; MARTÍNEZ-MARROQUÍN, R.; MELIN, P.; VALDEZ, F. ; SORIA, J. **Information Sciences.** Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot, journal, v.192, n.0, p. 19–38, 2012. *Swarm Intelligence and Its Applications.*
- [43] CASTILLO, O.; MELIN, P. **Information Sciences.** Optimization of type-2 fuzzy systems based on bio-inspired methods: A concise review, journal, v.205, n.0, p. 1–19, 2012.
- [44] CASTILLO, O.; MELIN, P. **Applied Soft Computing.** A review on the design and optimization of interval type-2 fuzzy controllers, journal, v.12, n.4, p. 1267–1278, 2012.

- [45] CHANG, P.-C.; LIU, C.-H. **Expert Systems with Applications**. A {TSK} type fuzzy rule based system for stock price prediction, journal, v.34, n.1, p. 135–144, 2008.
- [46] CHEN, Y.; YANG, B.; ABRAHAM, A. ; PENG, L. **IEEE Transactions on Fuzzy Systems**. Automatic design of hierarchical takagi ndash;sugeno type fuzzy systems using evolutionary algorithms, journal, v.15, n.3, p. 385–397, 2007.
- [47] CONOVER, W. J. **Practical Nonparametric Statistics**. 2. ed., New York: John Wiley, 1980.
- [48] CORDÓN, O.; GOMIDE, F.; HERRERA, F.; HOFFMANN, F. ; MAGDALENA, L. **Fuzzy Sets and Systems**. Ten years of genetic fuzzy systems: Current framework and new trends, journal, v.41, p. 5–31, 2004.
- [49] CORDÓN, O.; HERRERA, F.; HOFFMANN, F. ; MAGDALENA, L. **Genetic fuzzy systems: Evolutionary tuning and learning of fuzzy knowledge bases**. Singapore: World Scientific, 2001.
- [50] CORDÓN, O. **International Journal of Approximate Reasoning**. A historical review of evolutionary learning methods for mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems, journal, v.52, n.6, p. 894–913, 2011.
- [51] CORDÓN, O.; HERRERA, F. **International Journal of Approximate Reasoning**. A three-stage evolutionary process for learning descriptive and approximate fuzzy-logic-controller knowledge bases from examples, journal, v.17, n.4, p. 369–407, 1997.
- [52] CORDÓN, O.; HERRERA, F. **IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics**. A two-stage evolutionary process for designing tsk fuzzy rule-based systems, journal, v.29, n.6, p. 703–715, 1999.
- [53] CORDÓN, O.; HERRERA, F. **Fuzzy Sets and Systems**. Hybridizing genetic algorithms with sharing scheme and evolution strategies for designing approximate fuzzy rule-based systems, journal, v.118, n.2, p. 235–255, 2001.
- [54] CORDÓN, O.; HERRERA, F.; MAGDALENA, L. ; VILLAR, P. **Information Sciences**. A genetic learning process for the scaling factors, granularity and contexts of the fuzzy rule-based system data base, journal, v.136, n.1–4, p. 85–107, 2001.

- [55] CORDÓN, O.; HERRERA, F.; MAGDALENA, L. ; VILLAR, P. **Inf Sci**. A genetic learning process for the scaling factors, granularity and contexts of the fuzzy rule-based system data base, journal, v.136, n.1-4, p. 85-107, 2001.
- [56] CORDÓN, O.; HERRERA, F. ; VILLAR, P. **IEEE Transactions on Fuzzy Systems**. Generating the knowledge base of a fuzzy rule-based system by the genetic learning of the data base, journal, v.9, n.4, p. 667-674, 2001.
- [57] CORDÓN, O.; JESUS, M. J.; HERRERA, F. ; LOZANO, M. **International Journal of Intelligent Systems**. Mogul: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach, journal, v.14, n.11, p. 1123-1153, 1999.
- [58] CORREA, L. M. S.; RODRIGUES, E. S. **Erros de atração no processamento da concordância sujeito-verbo e a questão da autonomia do formulador sintático**. In: Maia, M.; Finger, I., editors, PROCESSAMENTO DA LINGUAGEM, p. 303-335. Editora da EDUCAT, Pelotas, 2006.
- [59] CREMER, M.; SCHOONEN, R. **Applied Psycholinguistics**. The role of accessibility of semantic word knowledge in monolingual and bilingual fifth-grade reading, journal, v.34, n.6, p. 1195-1217, 2013.
- [60] CROCKETT, K.; BANDAR, Z. ; MCLEAN, D. **On the optimization of t-norm parameters within fuzzy decision trees**. In: IEEE INTERNATIONAL ON FUZZY SYSTEMS CONFERENCE, 2007. FUZZ-IEEE 2007, p. 1-6, 2007.
- [61] CROCKETT, K. A.; BANDAR, Z.; FOWDAR, J. ; O'SHEA, J. **Expert Systems**. Genetic tuning of fuzzy inference within fuzzy classifier systems, journal, v.23, n.2, p. 63-82, 2006.
- [62] DE OLIVEIRA, J. V. **IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans**. Semantic constraints for membership function optimization, journal, v.29, n.1, p. 128-138, 1999.
- [63] DE SOUZA, F. J.; VELLASCO, M. M. ; PACHECO, M. A. **Fuzzy Sets and Systems**. Hierarchical neuro-fuzzy quadtree models, journal, v.130, n.2, p. 189-205, 2002.

- [64] DELGADO, M. R.; ZUBEN, F. V. ; GOMIDE, F. **Information Sciences**. Hierarchical genetic fuzzy systems, journal, v.136, n.1–4, p. 29–52, 2001.
- [65] DEMŠAR, J. **Journal of Machine Learning Research**. Statistical comparisons of classifiers over multiple data sets, journal, v.7, p. 1–30, 2006.
- [66] DERRAC, J.; GARCÍA, S.; MOLINA, D. ; HERRERA, F. **Swarm and Evolutionary Computation**. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, journal, v.1, n.1, p. 3–18, 2011.
- [67] DIAS, D. M.; PACHECO, M. A. C. **The Computer Journal**. Quantum-inspired linear genetic programming as a knowledge management system, journal, v.56, n.9, p. 1043–1062, 2013.
- [68] DO BRASIL, B. C. **Taxa de câmbio r\$/us\$ mensal: período 1999-2011**, 2014.
- [69] DONG, G. L.; BYONG, W. L. ; SOON, H. C. **Electric Power Systems Research**. Genetic programming model for long-term forecasting of electric power demand, journal, v.40, n.1, p. 17–22, 1997.
- [70] ESHELMAN, J. **Foundations of Genetic Algorithms**. The CHC Adaptive Search Algorithm : How to Have Safe Search When Engaging in Nontraditional Genetic Recombination, journal, p. 265–283, 1991.
- [71] ESMIN, A.; AOKI, A. R. ; LAMBERT-TORRES, G. **Particle swarm optimization for fuzzy membership functions optimization**. In: IEEE INTERNATIONAL CONFERENCE ON SYSTEMS, MAN AND CYBERNETICS, volume 3, 2002.
- [72] ESPEJO, P. G.; VENTURA, S. ; HERRERA, F. **IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews**. A survey on the application of genetic programming to classification, journal, v.40, n.2, p. 121–144, 2010.
- [73] FATTAH, K. A. **Oil and Gas Business**. A new approach calculate oil-gas ratio for gas condensate and volatile oil reservoirs using genetic programming, journal, v.1, p. 311–323, 2012.
- [74] FAWCETT, T. **Pattern recognition letters**. An introduction to roc analysis, journal, v.27, n.8, p. 861–874, 2006.

- [75] FAZZOLARI, M.; ALCALÁ, R.; NOJIMA, Y.; ISHIBUCHI, H. ; HERRERA, F. **Improving a fuzzy association rule-based classification model by granularity learning based on heuristic measures over multiple granularities**. In: 2013 IEEE INTERNATIONAL WORKSHOP ON GENETIC AND EVOLUTIONARY FUZZY SYSTEMS (GEFS), p. 44–51, 2013.
- [76] FAZZOLARI, M.; ALCALA, R.; NOJIMA, Y.; ISHIBUCHI, H. ; HERRERA, F. **Fuzzy Systems, IEEE Transactions on**. A review of the application of multiobjective evolutionary fuzzy systems: Current status and further directions, journal, v.21, n.1, p. 45–65, 2013.
- [77] FERNÁNDEZ, A.; DEL-JESUS, M. J. ; HERRERA, F. **Information Sciences**. On the 2-tuples based genetic tuning performance for fuzzy rule based classification systems in imbalanced data-sets, journal, v.180, n.8, p. 1268–1291, 2010.
- [78] FERNÁNDEZ, A.; GARCIA, S.; LUENGO, J.; BERNADÓ-MANSILLA, E. ; HERRERA, F. **IEEE Transactions on Evolutionary Computation**. Genetics-based machine learning for rule induction: State of the art, taxonomy, and comparative study, journal, v.14, n.6, p. 913–941, 2010.
- [79] FERNÁNDEZ, A.; CALDERÓN, M.; BARRENECHEA, E.; BUSTINCE, H. ; HERRERA, F. **Fuzzy Sets and Systems**. Solving multi-class problems with linguistic fuzzy rule based classification systems based on pairwise learning and preference relations, journal, v.161, n.23, p. 3064–3080, 2010.
- [80] GACTO, M. J.; ALCALÁ, R. ; HERRERA, F. **Soft Computing**. Adaptation and application of MOEA for rule reduction and parameter tuning of fuzzy rule-based systems, journal, v.13, n.5, p. 419–436, 2008.
- [81] GACTO, M. J.; ALCALÁ, R. ; HERRERA, F. **Soft Computing**. Adaptation and application of multi-objective evolutionary algorithms for rule reduction and parameter tuning of fuzzy rule-based systems, journal, v.13, n.5, p. 419–436, 2009.
- [82] GACTO, M. J.; ALCALÁ, R. ; HERRERA, F. **Applied Intelligence**. A MOEA for an effective tuning of fuzzy logic controllers in heating, ventilating and air conditioning systems, journal, v.36, n.2, p. 330–347, 2012.

- [83] GACTO, M. J.; GALENDE, M.; ALCALÁ, R. ; HERRERA, F. **Obtaining accurate tsk fuzzy rule-based systems by multi-objective evolutionary learning in high-dimensional regression problems.** In: 2013 IEEE INTERNATIONAL CONFERENCE ON FUZZY SYSTEMS (FUZZ), p. 1–7, 2013.
- [84] GALAR, M.; FERNÁNDEZ, A.; BARRENECHEA, E.; BUSTINCE, H. ; HERRERA, F. **Pattern Recognition.** An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes, journal, v.44, n.8, p. 1761–1776, 2011.
- [85] GARCÍA, D.; GONZÁLEZ, A. ; PÉREZ, R. **Journal of Computer and System Sciences.** A feature construction approach for genetic iterative rule learning algorithm, journal, v.80, n.1, p. 101–117, 2014.
- [86] GEYER-SCHULZ, A. **Fuzzy rule-based expert systems and genetic machine learning.** Heidelberg: Physica-Verlag, 1995.
- [87] GOLDBERG, D. E.; KORB, B. ; DEB, K. **Complex systems.** Messy genetic algorithms: Motivation, analysis, and first results, journal, v.3, n.5, p. 493–530, 1989.
- [88] GONZÁLEZ, A.; PÉREZ, R. **IEEE Transactions on Fuzzy Systems.** Slave: a genetic learning system based on an iterative approach, journal, v.7, n.2, p. 176–191, 1999.
- [89] GONZÁLEZ, A.; PÉREZ., R. **IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics.** Selection of relevant features in a fuzzy genetic learning algorithm, journal, v.31, n.3, p. 417–425, 2001.
- [90] GONZÁLEZ, A.; PÉREZ, R. **Mathware & Soft Computing.** Improving the genetic algorithm of slave, journal, v.16, n.1, p. 59–70, 2009.
- [91] GORZALCZANY, M. B.; RUDZINSKI, F. **Genetic fuzzy rule-based modelling of dynamic systems using time series.** In: Rutkowski, L.; Korytkowski, M.; Scherer, R.; Tadeusiewicz, R.; Zadeh, L. A. ; Zurada, J. M., editors, SWARM AND EVOLUTIONARY COMPUTATION, volume 7269 de **Lecture Notes in Computer Science**, p. 231–239. Springer Berlin Heidelberg, 2012.
- [92] GRACZYK, M.; LASOTA, T.; TELEC, Z. ; TRAWINSKI, B. **Non-parametric statistical analysis of machine learning algorithms for regression problems.** In: Setchi, R.; Jordanov, I.; Howlett, R. J. ;

- Jain, L. C., editors, KNOWLEDGE-BASED AND INTELLIGENT INFORMATION AND ENGINEERING SYSTEMS, volume 6276 de **Lecture Notes in Computer Science**, p. 111–120. Springer Berlin Heidelberg, 2010.
- [93] HADAVANDI, E.; SHAVANDI, H. ; GHANBARI, A. **Expert Systems with Applications**. An improved sales forecasting approach by the integration of genetic fuzzy systems and data clustering: Case study of printed circuit board, journal, v.38, n.8, p. 9392–9399, 2011.
- [94] HAN, D. **Effects of porosity and clay content on acoustic properties of sandstone and unconsolidated sediments**. Stanford: PhD Thesis, Stanford University.
- [95] HASSAN, L. H.; MOGHAVVEMI, M.; ALMURIB, H. A. ; STEINMAYER, O. **International Journal of Electrical Power Energy Systems**. Application of genetic algorithm in optimization of unified power flow controller parameters and its location in the power system network, journal, v.46, n.0, p. 89–97, 2013.
- [96] HASTIE, T.; TIBSHIRANI, R. ; FRIEDMAN, J. H. **The elements of statistical learning : Data mining, inference, and prediction**. 2. ed., New York: Springer Verlag, 2009.
- [97] HAYKIN, S. **Neural networks and learning machines**. New York: Prentice-Hall, 2009.
- [98] HERRERA, F. **Evolutionary Intelligence**. Genetic fuzzy systems: taxonomy, current research trends and prospects, journal, v.1, n.1, p. 27–46, 2008.
- [99] HERRERA, F.; LOZANO, M. ; SÁNCHEZ, A. M. **International Journal of Intelligent Systems**. A taxonomy for the crossover operator for real-coded genetic algorithms: An experimental study, journal, v.18, n.3, p. 309–338, 2003.
- [100] HERRERA, F.; LOZANO, M. ; VERDEGAY, J. L. **International Journal of Approximate Reasoning**. Tuning fuzzy logic controllers by genetic algorithms, journal, v.12, n.3–4, p. 299–315, 1995.
- [101] HERRERA, F.; MARTINEZ, L. **IEEE Transactions on Fuzzy Systems**. A 2-tuple fuzzy linguistic representation model for computing with words, journal, v.8, n.6, p. 746–752, 2000.

- [102] HINCHLIFFE, M.; HIDDEN, H.; MCKAY, B.; WILLIS, M.; THAM, M. ; BARTON, G. **Modelling chemical process systems using a multi-gene genetic programming algorithm**. In: Koza, J. R., editor, LATE BREAKING PAPERS AT THE GENETIC PROGRAMMING 1996 CONFERENCE STANFORD UNIVERSITY JULY 28-31, 1996, p. 56–65, Stanford University, CA, USA, 1996. Stanford University.
- [103] HOFFMANN, F.; NELLES, O. **Information Sciences**. Genetic programming for model selection of tsf-fuzzy systems, journal, v.136, n.1–4, p. 7–28, 2001.
- [104] HOMAIFAR, A.; BATTLE, D.; TUNSTEL, E. ; DOZIER, G. **International Journal of Knowledge Based Intelligent Engineering Systems**. Genetic programming design of fuzzy logic controllers for mobile robot path tracking, journal, v.4, n.1, p. 33–52, 2000.
- [105] HUTTERER, S.; ZAUNER, G.; HUML, M.; SILYE, R. ; SCHILCHER, K. **Data mining techniques for AFM- based tumor classification**. In: IEEE SYMPOSIUM ON COMPUTATIONAL INTELLIGENCE IN BIOINFORMATICS AND COMPUTATIONAL BIOLOGY (CIBCB 2012), p. 105–111, 2012.
- [106] HYNDMAN, R. J.; KOEHLER, A. B. **International Journal of Forecasting**. Another look at measures of forecast accuracy, journal, v.22, n.4, p. 679–688, 2006.
- [107] ICKE, I.; BONGARD, J. **Improving genetic programming based symbolic regression using deterministic machine learning**. In: de la Fraga, L. G., editor, 2013 IEEE CONFERENCE ON EVOLUTIONARY COMPUTATION, volume 1, p. 1763–1770, Cancun, Mexico, 2013.
- [108] ISHIBUCHI, H. **Multiobjective genetic fuzzy systems: Review and future research directions**. In: IEEE INTERNATIONAL FUZZY SYSTEMS CONFERENCE, 2007. FUZZ-IEEE 2007, p. 1–6, 2007.
- [109] ISHIBUCHI, H.; KAISHO, Y. ; NOJIMA, Y. **Complexity, interpretability and explanation capability of fuzzy rule-based classifiers**. In: IN IEEE INTERNATIONAL CONFERENCE ON FUZZY SYSTEMS, 2009. FUZZ-IEEE 2009, p. 1730–1735. IEEE, 2009.
- [110] ISHIBUCHI, H.; MURATA, T. ; TURKŞEN, I. **Fuzzy Sets and Systems**. Single-objective and two-objective genetic algorithms for selecting

- linguistic rules for pattern classification problems, journal, v.89, n.2, p. 135–150, 1997.
- [111] ISHIBUCHI, H.; NAKASHIMA, T. ; KURODA, T. **A hybrid fuzzy gbml algorithm for designing compact fuzzy rule-based classification systems.** In: IN IEEE INTERNATIONAL CONFERENCE ON FUZZY SYSTEMS, 2000. FUZZ-IEEE 2000, p. 706–711. IEEE, 2000.
- [112] ISHIBUCHI, H.; NAKASHIMA, T. ; MURATA, T. **IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics.** Design of accurate classifiers with a compact fuzzy-rule base using an evolutionary scatter partition of feature space, journal, v.29, n.5, p. 601–618, 1999.
- [113] ISHIBUCHI, H.; NAKASHIMA, T. ; NII, M. **Classification and modeling with linguistic information granules: advanced approaches advanced approaches to linguistic data mining.** Heidelberg: Springer, 2005.
- [114] ISHIBUCHI, H.; NOZAKI, K.; YAMAMOTO, N. ; TANAKA, H. **IEEE Transactions on Fuzzy Systems.** Selecting fuzzy if-then rules for classification problems using genetic algorithms, journal, v.3, n.3, p. 260–270, 1995.
- [115] ISHIBUCHI, H.; YAMAMOTO, T. **Fuzzy Sets and Systems.** Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining, journal, v.141, n.1, p. 59–88, 2004.
- [116] ISHIBUCHI, H.; YAMAMOTO, T. **IEEE Transactions on Fuzzy Systems.** Rule weight specification in fuzzy rule-based classification systems, journal, v.13, n.4, p. 428–435, 2005.
- [117] ISHIBUCHI, H.; YAMAMOTO, T. ; NAKASHIMA, T. **IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics.** Hybridization of fuzzy gbml approaches for pattern classification problems, journal, v.35, n.2, p. 359–365, 2005.
- [118] ISHIBUCHI, H.; YAMANE, M. ; NOJIMA, Y. **Rule weight update in parallel distributed fuzzy genetics-based machine learning with data rotation.** In: IN IEEE INTERNATIONAL CONFERENCE ON FUZZY SYSTEMS, 2013. FUZZ-IEEE 2013, p. 1–8. IEEE, 2013.

- [119] IZMAILOV, A.; SOLODOV, M. **Otimização volume 1: Condições de Otimalidade, Elementos de Análise Convexa e Dualidade**. Rio de Janeiro: IMPA, 2009.
- [120] JAEGER, H.; MAASS, W. ; PRINCIPE, J. **Neural Networks**. Special issue on esn and liquid state machines, journal, v.20, n.3, p. 287–289, 2007.
- [121] JANG, J. S. R.; SUN, C. ; MIZUTANI, E. **Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence**. New Jersey: Prentice-Hall, 1997.
- [122] JOHNSON, R. A.; WICHERN, D. W. **Applied multivariate statistical analysis**. New York: Prentice Hall, 2002.
- [123] JUANG, C.-F.; LIN, J.-Y. ; LIN, C.-T. **IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics**. Genetic reinforcement learning through symbiotic evolution for fuzzy controller design, journal, v.30, n.2, p. 290–302, 2000.
- [124] KARR, C. **AI Expert**. Genetic algorithms for fuzzy controllers, journal, v.6, n.2, p. 26–33, 1991.
- [125] KICKERT, W. J. M.; KOPPELAAR, H. **IEEE Transactions on Systems, Man and Cybernetics**. Application of fuzzy set theory to syntactic pattern recognition of handwritten capitals, journal, v.SMC-6, n.2, p. 148–151, Feb 1976.
- [126] KIM, D.; CHOI, Y.-S. ; LEE, S.-Y. **Fuzzy Sets and Systems**. An accurate {COG} defuzzifier design using lamarckian co-adaptation of learning and evolution, journal, v.130, n.2, p. 207–225, 2002.
- [127] KIM, D.; KIM, C. **Fuzzy Systems, IEEE Transactions on**. Forecasting time series with genetic fuzzy predictor ensemble, journal, v.5, n.4, p. 523–535, 1997.
- [128] KISHORE, J. K.; PATNAIK, L. M.; MANI, V. ; AGRAWAL, V. K. **IEEE Trans. Evol. Comput.** Application of genetic programming for multicategory pattern classification, journal, v.4, n.3, p. 242—258, 2000.
- [129] KLEMENT, E. P.; MESIAR, R. ; PAP, E. **Triangular norms**. Dordrecht: Kluwer Academic Publishers, 2000.
- [130] KLIR, G. J.; YUAN, B. **Fuzzy sets and fuzzy logic: theory and applications**. New Jersey: Prentic Hall PTR, 1995.

- [131] KOSHIYAMA, A.; ESCOVEDO, T.; DIAS, D.; VELLASCO, M. ; TANSCHKEIT, R. **GPF-CLASS: A genetic fuzzy model for classification**. In: de la Fraga, L. G., editor, 2013 IEEE CONFERENCE ON EVOLUTIONARY COMPUTATION, volume 1, p. 3275–3282, Cancun, Mexico, 2013.
- [132] KOVACS, T. **Strength or accuracy: credit assignment in learning classifier systems**. Berlin: Springer, 2004.
- [133] KOZA, J. R. **Genetic Programming: On the Programming of Computers by Means of Natural Selection**. Massachusetts: MIT Press, 1992.
- [134] KRUGMAN, P. R.; OBSTFELD, M. **Economia Internacional: Teoria e Política**. Makron Books: São Paulo, 2001.
- [135] KUO, I.-H.; HORNG, S.-J.; KAO, T.-W.; LIN, T.-L.; LEE, C.-L. ; PAN, Y. **Expert Systems with Applications**. An improved method for forecasting enrollments based on fuzzy time series and particle swarm optimization, journal, v.36, n.3, Part 2, p. 6108–6117, 2009.
- [136] LANGDON, W. B.; GUSTAFSON, S. M. **Genetic Programming and Evolvable Machines**. Genetic programming and evolvable machines: ten years of reviews, journal, v.11, n.3, p. 321–338, 2010.
- [137] LANGDON, W. B.; POLI, R. **Foundations of Genetic Programming**. Springer-Verlag, 2002.
- [138] LEITE, D.; BALLINI, R.; COSTA, P. ; GOMIDE, F. **Evolving Systems**. Evolving fuzzy granular modeling from nonstationary fuzzy data streams, journal, v.3, n.2, p. 65–79, 2012.
- [139] LESSA, A. C.; COUTO, L. F. ; FARIAS, R. S. **Revista Brasileira de Política Internacional**. Política externa planejada: os planos plurianuais e a ação internacional do brasil, de cardoso a lula (1995-2008), journal, v.52, n.1, p. 89–109, 2009.
- [140] LIM, T.-S.; LOH, W.-Y. ; SHIH, Y.-S. **Machine Learning**. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms, journal, v.40, n.3, p. 203–228, 2000.

- [141] LIU, B. D.; CHEN, C. Y. ; TSAO, J. Y. **IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics**. Design of adaptive fuzzy logic controller based on linguistic-hedge concepts and genetic algorithms, journal, v.31, n.1, p. 32–53, 2001.
- [142] LÓPEZ, V.; FERNÁNDEZ, A.; DEL JESUS, M. J. ; HERRERA, F. **Knowledge-Based Systems**. A hierarchical genetic fuzzy system based on genetic programming for addressing classification with highly imbalanced and borderline data-sets, journal, v.38, n.0, p. 85 – 104, 2013.
- [143] LUKE, S.; PANAIT, L. **Lexicographic parsimony pressure**. In: et al., W. B. L., editor, **GECCO 2002: PROCEEDINGS OF THE GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE**, p. 829–836, New York, 2002. Morgan Kaufmann Publishers.
- [144] MAKRIDAKIS, S.; HIBON, M. **International Journal of Forecasting**. The m3-competition: results, conclusions and implications, journal, v.16, n.4, p. 451–476, 2000.
- [145] MAMDANI, E. H. **Proceedings of the Institution of Electrical Engineers**. Application of fuzzy algorithms for control of simple dynamic plant, journal, v.121, n.3, p. 1585–1588, 1974.
- [146] MAMDANI, E. H.; ASSILIAN, S. **International Journal of Man-Machine Studies**. An experiment in linguistic synthesis with a fuzzy logic controller, journal, v.7, n.1, p. 1–13, 1975.
- [147] MARÍN-BLÁZQUEZ, J. G.; PÉREZ, G. M. **Soft Computing**. Intrusion detection using a linguistic hedged fuzzy-xcs classifier system, journal, v.13, n.3, p. 273–290, 2009.
- [148] MÁRQUEZ, A. A.; MÁRQUEZ, F. A.; ROLDÁN, A. M. ; PEREGRÍN, A. **Knowledge-Based Systems**. An efficient adaptive fuzzy inference system for complex and high dimensional regression problems in linguistic fuzzy modelling, journal, v.54, n.0, p. 42–52, 2013.
- [149] MATLAB. **MATLAB R2010a**. Mathworks, Natick, MA, 2010.
- [150] MEDEIROS, M. C.; VEIGA, A. **IEEE Transactions on Neural Networks**. A flexible coefficient smooth transition time series model, journal, v.16, n.1, p. 97–113, 2005.
- [151] MICHALEWICZ, Z. **Genetic Algorithms + Data Structure = Evolutionary Programs**. New York: Springer-Verlag, 1992.

- [152] MILLER, J. F. **Cartesian genetic programming**. In: Miller, J. F., editor, **CARTESIAN GENETIC PROGRAMMING**, Natural Computing Series, chapter 2, p. 17–34. Springer, 2011.
- [153] MILLER, J. F.; THOMSON, P. **Cartesian genetic programming**. In: Poli, R.; Banzhaf, W.; Langdon, W. B.; Miller, J.; Nordin, P. ; Fogarty, T. C., editors, **GENETIC PROGRAMMING**, volume 1802 de **Lecture Notes in Computer Science**, p. 121–132. Springer Berlin Heidelberg, 2000.
- [154] MITCHELL, T. M. **Machine learning**. Burr Ridge: McGraw Hill, 1997.
- [155] MORETTIN, P. A.; TOLÓI, C. M. C. **Análise de Séries Temporais**. 2. ed., Edgard Blucher: São Paulo, 2006.
- [156] MUCIENTES, M.; VIDAL, J. C.; BUGARÍN, A. ; LAMA, M. **Soft Computing**. Processing time estimations by variable structure tsk rules learned through genetic programming, journal, v.13, n.5, p. 497–509, 2009.
- [157] MUNI, D. P.; PAL, N. R. **Fuzzy Information and Engineering**. Evolution of fuzzy classifiers using genetic programming, journal, v.4, n.1, p. 29–49, 2012.
- [158] NAUCK, D.; KLAWONN, F. ; KRUSE, R. **Foundations of neuro-fuzzy systems**. New York: John Wiley & Sons, 1997.
- [159] NGUYEN, H. T.; WALKER, E. A. **A First Course in Fuzzy Logic**. 2. ed., London: Chapman & Hall/CRC, 2000.
- [160] NIKOLAEV, N. Y.; IBA, H. **Genetic programming of polynomial models for financial forecasting**. In: Chen, S.-H., editor, **GENETIC ALGORITHMS AND GENETIC PROGRAMMING IN COMPUTATIONAL FINANCE**, chapter 5, p. 103–123. Kluwer Academic Press, 2002.
- [161] NOCEDAL, J.; WRIGHT, S. J. **Numerical Optimization**. 2. ed., New York: Springer-Verlag, 2006.
- [162] NOJIMA, Y.; ALCALÁ, R.; ISHIBUCHI, H. ; HERRERA, F. **Soft Computing**. Special issue on evolutionary fuzzy systems, journal, v.15, n.12, p. 2299–2301, 2011.
- [163] NUNES, W.; VELLASCO, M. ; TANSCHKEIT, R. **A quantum-inspired evolutionary algorithm for fuzzy classification**. In: 2013 JOINT IFSA WORLD CONGRESS AND NAFIPS ANNUAL MEETING (IFSA/NAFIPS), p. 29–34, 2013.

- [164] ORD, K.; HIBON, M. ; MAKRIDAKIS, S. **International Journal of Forecasting**. The m3-competition, journal, v.16, n.4, p. 433 – 436, 2000.
- [165] PAL, T.; PAL, N. R. **IEEE Transactions on Evolutionary Computation**. Sogarg: A self-organized genetic algorithm-based rule generation scheme for fuzzy controllers, journal, v.7, n.4, p. 397–415, 2003.
- [166] PALACIOS, A. M.; SÁNCHEZ, L. ; COUSO, I. **Evolutionary Intelligence**. Extending a simple genetic cooperative-competitive learning fuzzy classifier to low quality datasets, journal, v.2, n.1-2, p. 73–84, 2009.
- [167] PALACIOS, A. M.; SÁNCHEZ, L. ; COUSO, I. **International Journal of Approximate Reasoning**. Linguistic cost-sensitive learning of genetic fuzzy classifiers for imprecise data, journal, v.52, n.6, p. 841–862, 2011.
- [168] PAULANI, L. M. **Revista de Economia Política**. Brasil delivery: A política econômica do governo lula, journal, v.23, n.4, p. 58–73, 2003.
- [169] PEDRYCZ, W. **Fuzzy Evolutionary Computation**. Norwell: Kluwer Academic Publishers, 1997.
- [170] PEDRYCZ, W.; GOMIDE, F. **An introduction to fuzzy sets: analysis and design**. Massachusetts: MIT Press, 1998.
- [171] POLI, R.; LANGDON, W. B. ; MCPHEE, N. F. **A Field Guide to Genetic Programming**. Raleigh: Lulu.com, 2008.
- [172] PRADO, R. P.; GARCÍA-GALÁN, S.; EXPOSITO, J. ; YUSTE, A. J. **IEEE Transactions on Fuzzy Systems**. Knowledge acquisition in fuzzy-rule-based systems with particle-swarm optimization, journal, v.18, n.6, p. 1083–1097, 2010.
- [173] PULKKINEN, P.; KOIVISTO, H. **IEEE Transactions on Fuzzy Systems**. A dynamically constrained multiobjective genetic fuzzy system for regression problems, journal, v.18, n.1, p. 161–177, 2010.
- [174] R CORE TEAM. **R: A Language and Environment for Statistical Computing**. R Foundation for Statistical Computing, Vienna, Austria, 2013.
- [175] REZANIA, M.; JAVADIA, A. A. **Canadian Geotechnical Journal**. A new genetic programming model for predicting settlement of shallow foundations, journal, v.44, n.12, p. 1462–1473, 2007.

- [176] ROGERS, S. J.; FANG, J. H.; KARR, C. L. ; STANLEY, D. A. **AAPG Bulletin**. Determination of lithology from well logs using a neural network, journal, v.76, n.5, p. 731–739, 1992.
- [177] ROJAS, I.; GONZALEZ, J.; POMARES, H.; ROJAS, F.; FERNÁNDEZ, F. ; PRIETO, A. **International Journal of Approximate Reasoning**. Multidimensional and multideme genetic algorithms for the construction of fuzzy systems, journal, v.26, n.3, p. 179–210, 2001.
- [178] ROSSI, J. W. **Pesquisa e Planejamento Econômico**. Determinação da taxa de câmbio: testes empíricos para o brasil, journal, v.21, n.2, p. 397–412, 1991.
- [179] ROYCHOWDHURY, S.; PEDRYCZ, W. **Int Journal of Intelligent Systems**. A survey of defuzzification strategies, journal, v.16, n.6, p. 679–695, 2001.
- [180] RUBINSTEIN, R. Y.; KROESE, D. P. **The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning**. New York: Springer-Verlag, 2004.
- [181] SALAMA, P. **Estudos Avançados**. Argentina, brasil e méxico diante da crise internacional, journal, v.23, n.65, p. 27–48, 2009.
- [182] SALGADO, A. C. P.; HEYE, J. W. B. **Medidas de bilingualidade: uma proposta**.
- [183] SÁNCHEZ, L.; COUSO, I. ; CORRALES, J. A. **Information Sciences**. Combining {GP} operators with {SA} search to evolve fuzzy rule based classifiers, journal, v.136, n.1–4, p. 175–191, 2001.
- [184] SANTIS, E. D.; RIZZI, A.; SADEGHIAN, A. ; MASCIOLI, F. M. F. **Genetic optimization of a fuzzy control system for energy flow management in micro-grids**. In: 2013 JOINT IFSA WORLD CONGRESS AND NAFIPS ANNUAL MEETING (IFSA/NAFIPS), p. 418–423, 2013.
- [185] SANTOS, R.; VELLASCO, M.; ARTOLA, F. ; FONTOURA, S. D. **Neural net ensembles for lithology recognition**. In: Windeatt, T.; Roli, F., editors, MULTIPLE CLASSIFIER SYSTEMS. 2003.

- [186] SANZ, J. A.; FERNÁNDEZ, A.; BUSTINCE, H. ; HERRERA, F. **Information Sciences**. Improving the performance of fuzzy rule-based classification systems with interval-valued fuzzy sets and genetic amplitude tuning, journal, v.180, n.19, p. 3674–3685, 2010.
- [187] SANZ, J. A.; FERNANDEZ, A.; BUSTINCE, H. ; HERRERA, F. **IEEE Transactions on Fuzzy Systems**. IVTURS: A linguistic fuzzy rule-based classification system based on a new interval-valued fuzzy reasoning method with tuning and rule selection, journal, v.21, n.3, p. 399–411, 2013.
- [188] SANZ, J. A.; GALAR, M.; JURIO, A.; BRUGOS, A.; PAGOLA, M. ; BUSTINCE, H. **Applied Soft Computing**. Medical diagnosis of cardiovascular diseases using an interval-valued fuzzy rule-based classification system, journal, n.0, p. A publicar, 2013.
- [189] SCHLUMBERGER. <http://www.glossary.oilfield.slb.com/Display.cfm?Term=lithology>: . 2013.
- [190] SEARSON, D. **GPTIPS User Guide (genetic programming & symbolic regression for MATLAB)**. Newcastle University, 1. ed., 2009.
- [191] SEARSON, D.; WILLIS, M. ; MONTAGUE, G. **Journal of Chemometrics**. Co-evolution of non-linear PLS model components, journal, v.21, n.12, p. 592–603, 2007.
- [192] SHINTEMIROV, A.; TANG, W. ; WU, Q. H. **IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews**. Power transformer fault classification based on dissolved gas analysis by implementing bootstrap and genetic programming, journal, v.39, n.1, p. 69–79, Jan. 2009.
- [193] STACH, W.; KURGAN, L. A. ; PEDRYCZ, W. **Fuzzy Systems, IEEE Transactions on**. Numerical and linguistic prediction of time series with the use of fuzzy cognitive maps, journal, v.16, n.1, p. 61–72, 2008.
- [194] TAKAGI, T.; SUGENO, M. **IEEE Transactions on Systems, Man and Cybernetics**. Fuzzy identification of systems and its applications to modeling and control, journal, v.SMC-15, n.1, p. 116–132, 1985.
- [195] TAO, C. W.; TAUR, J. S.; HSIEH, T. W. ; TSAI, C. L. **IEEE Transactions on Control Systems Technology**. Design of a fuzzy controller with fuzzy swing-up and parallel distributed pole assignment

- schemes for an inverted pendulum and cart system, journal, v.16, n.6, p. 1277–1288, 2008.
- [196] TELFORD, W. M.; GELDART, L. P. ; SHERIFF, R. E. **Applied Geophysics**. 2. ed., Cambridge: Cambridge University Press, 2005.
- [197] THRIFT, P. **Fuzzy logic synthesis with genetic algorithms**. In: PROCEEDINGS OF THE 4TH INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, ICGA '91, p. 514–518, San Francisco, 1991. Morgan Kaufmann.
- [198] TOMITA, Y.; KATO, R.; OKOCHI, M. ; HONDA, H. **Journal of Bioscience and Bioengineering**. A motif detection and classification method for peptide sequences using genetic programming, journal, v.106, n.2, p. 154–161, 2008.
- [199] TONG, R. M.; YAGER, R. R. **Characterizing buzz and sentiment in internet sources: Linguistic summaries and predictive behaviors**. In: Shanahan, J. G.; Qu, Y. ; Wiebe, J., editors, COMPUTING ATTITUDE AND AFFECT IN TEXT: THEORY AND APPLICATIONS, volume 20 de **The Information Retrieval Series**, p. 281–296. Springer Netherlands, 2006.
- [200] TORRES, E. A.; COTTA, C. ; LINERO, J. M. T. **Mathware & Soft Computing**. Evolutionary design of fuzzy logic controllers using strongly-typed gp, journal, v.6, n.1, p. 109–124, 1999.
- [201] TSAKONAS, A. **Information Sciences**. A comparison of classification accuracy of four genetic programming-evolved intelligent structures, journal, v.176, n.6, p. 691–724, 2006.
- [202] TSAKONAS, A. **Expert Systems with Applications**. Local and global optimization for takagi–sugeno fuzzy system by memetic genetic programming, journal, v.40, n.8, p. 3282–3298, 2013.
- [203] TSANG, E. P. K.; LI, J. **EDDIE for financial forecasting**. In: Chen, S.-H., editor, GENETIC ALGORITHMS AND GENETIC PROGRAMMING IN COMPUTATIONAL FINANCE, chapter 7, p. 161–174. Kluwer Academic Press, 2002.
- [204] TSUKAMOTO, Y. **Advances in fuzzy set theory and applications**. An approach to fuzzy reasoning method, journal, v.137, p. 149, 1979.

- [205] TUNSTEL, E.; JAMSHIDI, M. **Intl. Journal of Intelligent Automation and Soft Computing**. On genetic programming of fuzzy rule-based systems for intelligent control, journal, v.2, p. 273–284, 1996.
- [206] U.S. GEOLOGICAL SURVEY. <http://earthquake.usgs.gov/learn/glossary/?term=lithology>: . 2013.
- [207] VALENZUELA-RENDÓN, M. **The fuzzy classier system: A classier system for continuously varying variables**. In: PROCEEDINGS OF THE 4TH INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, ICGA '91, p. 223–230, San Francisco, 1991. Morgan Kaufmann.
- [208] WANG, L. X.; MENDEL, J. M. **IEEE Transactions on Systems, Man and Cybernetics**. Generating fuzzy rules by learning from examples, journal, v.22, n.6, p. 1414–1427, 1992.
- [209] WECHSLER, H. **International Journal of Bio-Medical Computing**. A fuzzy approach to medical diagnosis, journal, v.7, n.3, p. 191–203, 1976.
- [210] WURM, L. H.; FISICARO, S. A. **Journal of Memory and Language**. What residualizing predictors in regression analyses does (and what it does not do), journal, v.72, n.0, p. 37–48, 2014.
- [211] YAGER, R. R. **IEEE Transactions on Systems, Man and Cybernetics**. On ordered weighted averaging aggregation operators in multicriteria decisionmaking, journal, v.18, n.1, p. 183–190, Jan 1988.
- [212] YAGER, R. R.; KACPRZYK, J. **The Ordered Weighted Averaging Operators: Theory and Applications**. Norwell: Kluwer, 1997.
- [213] YAZDANI, S.; SHOOREHDELI, M. A. ; TESHNEHLAB, M. **Identification of fuzzy models using cartesian genetic programming**. In: IN INTERNATIONAL CONFERENCE ON COMPUTATIONAL INTELLIGENCE AND SECURITY, p. 76–81. IEEE, 2008.
- [214] ZADEH, L. A. **Information and Control**. Fuzzy sets, journal, v.8, n.3, p. 338–353, 1965.
- [215] ZADEH, L. A. **Information Sciences**. Is there a need for fuzzy logic?, journal, v.178, n.13, p. 2751–2779, 2008.

A Definições e Operadores Fuzzy

Os conceitos dispostos nesse anexo podem ser encontrados com maior clareza e profundidade em Klir e Yuan [130], Pedrycz e Gomide [170] ou Barros e Bassanezi [22].

A.0.1 Base de Regras Fuzzy

Uma base de regras fuzzy é um conjunto de proposições compostas por elementos antecedentes e consequentes [22]. Esta pode ser disposta em formato de uma tabela, conforme a Tabela A.1.

Tabela A.1: Exemplo de base de regras fuzzy.

Regra	Conteúdo
R_1	“ Se Antecedente 1 então Consequente 1”
R_2	“ Se Antecedente 2 então Consequente 2”
⋮	⋮
R_D	“ Se Antecedente D então Consequente D ”

Seja $\mathbf{x}_i = [x_{i1}x_{i2}...x_{iJ}]$ a i -ésima amostra ou padrão, composto por valores das J variáveis de entrada X_j ($j = 1, \dots, J$ e $i = 1, \dots, n$). O l -ésimo conjunto fuzzy ($l = 1, \dots, L$) associado a j -ésima variável é denotado por A_{lj} , onde $A_{lj} = \{(x_{ij}, \mu_{A_{lj}}(x_{ij})) | x_{ij} \in X_j\}$, onde $\mu_{A_{lj}}$ é uma função de pertinência associada a A_{lj} , tal que $\mu : X_j \rightarrow [0, 1]$. Logo, um conjunto fuzzy é um par que associa cada x_{ij} a um grau de pertinência (compatibilidade) $\mu_{A_{lj}}(x_{ij})$ ao conjunto fuzzy A_{lj} .

Seja a parte antecedente da d -ésima regra dada por:

Antecedente é igual a “ Se x_{i1} é A_{l1} e x_{i2} é A_{l2} e ... x_{iJ} é A_{lJ} ”

De forma semelhante, porém descrita em termos matemáticos, tem-se a mesma d -ésima premissa é dada por:

$$\mu_{A_d}(\mathbf{x}_i) = \mu_{A_{l1}}(x_{i1}) * \mu_{A_{l2}}(x_{i2}) * \dots * \mu_{A_{lJ}}(x_{iJ}) \quad (\text{A-1})$$

onde cada $\mu_{A_{lj}}(x_{ij})$ é o grau de pertinência da i -ésima observação da j -ésima variável, ao conjunto fuzzy A_{lj} . O $\mu_{A_d}(\mathbf{x}_i)$ é o grau de pertinência conjunto calculado através de cada ativação marginal $\mu_{A_{lj}}(x_{ij})$. O operador $*$ é um mecanismo que reproduz matematicamente a conjunção “e”, presente na d -ésima regra.

Considere y_i a i -ésima observação da variável de saída Y (variável dependente, resposta, etc.). De forma similar, tome $B_k, k = 1, \dots, K$ conjuntos fuzzy que denotam estados linguísticos da variável Y . Seja a relação de consequência, que torna uma premissa (termo antecedente) em uma regra fuzzy dada por:

Se “ d -ésimo antecedente” **então** y_i é B_k

Comumente, diferentes termos antecedentes se relacionam com um mesmo termo consequente. Portanto, é comum que a ativação de regras com antecedentes diferentes, porém com consequentes iguais aconteçam simultaneamente. A descrição deste tipo de evento é dado por:

“Antecedente Resultante: Antecedente da Regra $1^{(k)}$ **ou/adicionado do** Antecedente da Regra $2^{(k)}$ **ou /adicionado do ... ou/adicionado do** Ativação ao Antecedente da Regra $D^{(k)}$ ”

Em outros termos:

$$\mu_{B_k}(y_i) = \mu_{A_{1^{(k)}} \cup \dots \cup D^{(k)}}(\mathbf{x}_i) = M[\mu_{A_{1^{(k)}}}(\mathbf{x}_i), \mu_{A_{2^{(k)}}}(\mathbf{x}_i), \dots, \mu_{A_{D^{(k)}}}(\mathbf{x}_i)] \quad (\text{A-2})$$

Para tanto, é necessário escolher um operador que replique de forma matemática o conectivo lógico “adicionado do” ou o “ou”, representado no caso pelo operador M . Em alguns momentos, M pode tomar a forma de uma t-conorma (a ser descrita em maior ênfase a seguir), e em outros casos pode ser somente um operador de agregação. Tais operadores são alvo de descrição no tópico seguinte.

A.1 Operadores

Este tópico descreve os cinco operadores mais comuns em SIF: conjunção, negação, concentração/dilatação e agregação. São descritas as propriedades de cada com alguns exemplos em seguida. Para maiores detalhes sobre teoria de operadores em Lógica Fuzzy ver Klir e Yuan [130] e Klement et al. [129].

A.1.1

Conjunção e Disjunção

Os operadores de conjunção e disjunção na Lógica Fuzzy são usados para modelar os conectivos “e”/“ou”, respectivamente. Estes são usados comumente na formação de regras linguísticas. Na família de operadores de conjunção encontram-se os denominados t-norma e de disjunção os t-conorma. Por exemplo, se houver uma regra do tipo: “*Se Pressão é Grande e Vazão é Elevada ou Temperatura é Baixa ...*”, os conectivos lógicos clássicos - \cup e \cap - são generalizados pelo uso de operadores t-norma e t-conorma.

Cada uma das propriedades, presentes em cada operador, foram escolhidas de modo a fornecer subsídios matemáticos e filosóficos para o papel que desempenham em unir graus de pertinência (verdades parciais) a diferentes conceitos linguísticos. A seguir são apresentadas as respectivas definições de operadores t-norma e t-conorma, suas propriedades e exemplos.

Definição 1. Uma função $*$: $[0, 1]^2 \rightarrow [0, 1]$ é uma t-norma se, para todo $a, b, c \in [0, 1]$, tem-se:

$$a * b = b * a \text{ (comutativa)} \quad (\text{A-3})$$

$$a * (b * c) = (a * b) * c \text{ (associativa)} \quad (\text{A-4})$$

$$1 * a = a \text{ e } a * 0 = 0 \text{ (elementos de contorno)} \quad (\text{A-5})$$

$$a * c \leq b * c, \forall a \leq b \text{ (monotônica crescente)} \quad (\text{A-6})$$

As operações de mínimo e produto são os operadores t-norma mais reconhecidos e usados, tanto na Lógica Fuzzy, quanto em Probabilidade. Mas, há toda uma gama de operadores t-norma na literatura [129] pouco explorados em SFG, porém com grande relevância teórica.

Definição 2. Uma função \oplus : $[0, 1]^2 \rightarrow [0, 1]$ é uma t-conorma se, para todo $a, b, c \in [0, 1]$, tem-se:

$$a \oplus b = b \oplus a \text{ (comutativa)} \quad (\text{A-7})$$

$$a \oplus (b \oplus c) = (a \oplus b) \oplus c \text{ (associativa)} \quad (\text{A-8})$$

$$1 \oplus a = 1 \text{ e } a \oplus 0 = a \text{ (elementos de contorno)} \quad (\text{A-9})$$

$$a \oplus c \leq b \oplus c, \forall a \leq b \text{ (monotônica crescente)} \quad (\text{A-10})$$

Como exemplo, as operações de máximo e soma probabilística satisfazem as propriedades requeridas para uma t-conorma. Como no caso das t-normas, há um grande número de outras t-conormas disponíveis. Para as operadores

t-norma e t-conorma, alguns autores impõe a propriedade de continuidade [12, 159]. Isto acarretaria no fato de que mudanças pequenas nos graus de pertinência não deveriam causar grandes variações no resultado final da composição.

A.1.2 Negação

Os operadores de negação ampliam o leque de possibilidades para formação de composições linguísticas, além da capacidade de reduzir a quantidade de regras, criando um conjunto mais compacto de composição. Por exemplo, se um especialista do processo reconhece que uma determinada reação química ocorre quando a temperatura “Não é Baixa”, este pode excluir todas as demais possibilidades (temperatura é média, alta, etc.), através de uma simples operação lógica. Apesar de à priori o operador de negação tornar as regras relativamente menos interpretáveis, este pode viabilizar uma base de regras fuzzy mais compacta.

A seguir são fornecidas a definição e propriedades desejáveis para um operador de negação, tendo em consideração $a, b \in [0, 1]$.

Definição 3. Uma aplicação $\eta : [0, 1] \rightarrow [0, 1]$ é uma negação se satisfazer:

$$\eta(1) = 0 \text{ e } \eta(0) = 1 \text{ (fronteiras)} \quad (\text{A-11})$$

$$\eta(\eta(a)) = a \text{ (involução)} \quad (\text{A-12})$$

$$\eta(b) \leq \eta(a), \forall a \leq b \text{ (monotônica decrescente)} \quad (\text{A-13})$$

Principais exemplos são as aplicações: $\eta(x) = 1 - x$ e $\eta(x) = \frac{1-x}{1+x}$, onde a primeira remete à mesma aplicação usada na lógica clássica [22].

A.1.3 Concentração e Dilatação

As operações de concentração e dilatação são modificadores linguísticos que basicamente funcionam como advérbios nas regras. Por exemplo, “Se Temperatura é **Muito** Alta” o termo “Muito” é o advérbio de intensidade na regra. Dos modificadores linguísticos, há dois tipos de operações principais - concentração e dilatação - que são definidos a partir de $\mu : X \rightarrow [0, 1]$.

Definição 4. Uma função $Conc : [0, 1] \rightarrow [0, 1]$ é uma operação de concentração se:

$$Conc(\mu(x)) \leq \mu(x), \forall x \in X \quad (\text{A-14})$$

Definição 5. Uma função $Dil : [0, 1] \rightarrow [0, 1]$ é uma operação de dilatação se:

$$Dil(\mu(x)) \geq \mu(x), \forall x \in X \quad (A-15)$$

Operações clássicas de dilatação e concentração são do tipo potência, $Pot : [0, 1] \times \mathbb{R}^+ \rightarrow [0, 1]$, dadas por:

$$Pot(\mu(x), s) = \mu(x)^s, \forall s \in [0, \infty) \quad (A-16)$$

quando $s \in (0, 1)$ tem-se uma operação de dilatação, enquanto que $s \in (1, \infty)$ uma operação de concentração. Para os casos de $s = 1$ não há modificação linguística, enquanto que $s \rightarrow 0$ o conjunto fuzzy torna-se cada vez mais clássico. Tais operações são ilustradas nas Figuras A.1 e A.2 [170].

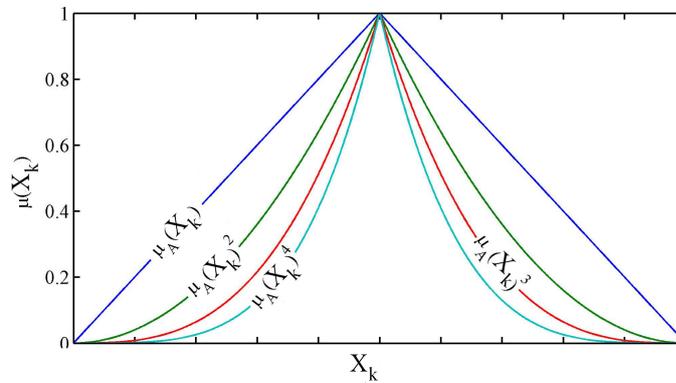


Figura A.1: Operação de concentração.

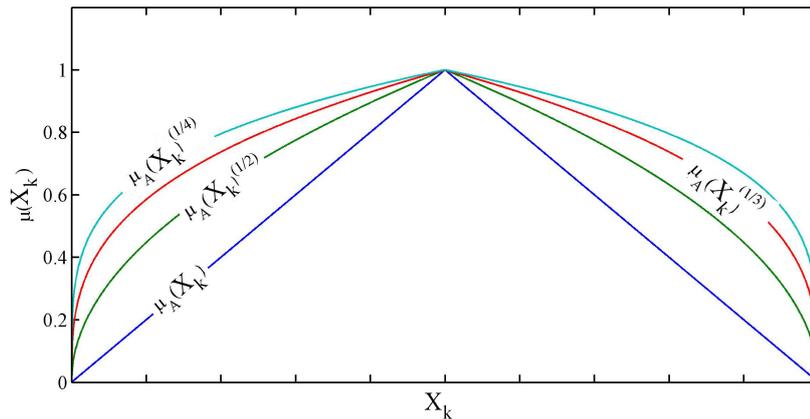


Figura A.2: Operação de dilatação.

A.1.4 Agregação

Operadores de agregação são usados diariamente nos mais diferentes ramos de atividade. Um exemplo é a média aritmética, bastante usada para agregar informações sobre uma quantidade de forma a fornecer a mesma relevância a cada elemento do conjunto de informações.

De forma sucinta, para $a_1, \dots, a_J \in [0, 1]$, tem-se a definição de um operador de agregação.

Definição 6. Um operador $M : [0, 1]^J \rightarrow [0, 1]$ de dimensão J , é um operador de agregação se:

$$M(0, 0, \dots, 0) = 0 \text{ e } M(1, 1, \dots, 1) = 1, \quad (\text{A-17})$$

$$M(a_1, a_2, v, \dots, a_{J-1}) \leq M(a_1, a_2, w, \dots, a_{J-1}), \forall v \leq w \in [0, 1] \quad (\text{A-18})$$

A condição (A-17) garante que o máximo de M é obtido quando $a_1 = a_2 = \dots = a_J = 1$ e o mínimo se $a_1 = a_2 = \dots = a_J = 0$ (condições de contorno). A condição (A-18) remete à monotonicidade não decrescente da operação de agregação. Há uma enorme gama de operadores de agregação presentes na literatura, tais como t-normas, t-conormas, estatísticas de ordem (quartis e mediana), médias (aritmética, harmônica, geométrica, ponderada), família dos *Ordered Weighted Average*, entre outros [211, 212].

B Demonstrações do Capítulo 4

B.1

Cálculo dos coeficientes para cada $\mu_{A_{d^{(k)}}}(\mathbf{x}_i)$

O problema original é encontrar um conjunto de $w_{d^{(k)}}$, de modo que:

$$\min : \sum_{i=1}^n (\mu_{B_k}(y_i) - \sum_{d^{(k)}=1}^{D^{(k)}} w_{d^{(k)}} \mu_{A_{d^{(k)}}}(\mathbf{x}_i))^2 \quad (\text{B-1})$$

$$s.j : \sum_{d^{(k)}=1}^{D^{(k)}} w_{d^{(k)}} = 1 \quad (\text{B-2})$$

$$w_{d^{(k)}} \geq 0 \quad (\text{B-3})$$

o que torna este problema mais custoso do que o irrestrito (cuja solução é a de mínimos quadrados) são as restrições de desigualdade ($w_{d^{(k)}} \geq 0$). Uma forma de facilitar este processo é implementar o método das variáveis ativas (ou do conjunto ativo) [161]. Em linhas gerais, o método procura soluções de mínimo que atendam as restrições de igualdade, formuladas a partir do Lagrangeano do problema. Caso todos os elementos desta solução satisfaçam as restrições de desigualdade, esta é retornada; caso contrário os elementos que não satisfazem a desigualdade são anulados (valores tornados em zero) e, portanto, excluídos do processo de otimização. Sem a presença destes, novamente busca-se uma solução de mínimo que atendam as restrições de igualdade e repete-se o processo.

Para facilitar a notação, denote por $[\mu_{A_{1^{(k)}}}(\mathbf{x}_i), \dots, \mu_{A_{D^{(k)}}}(\mathbf{x}_i)] = X$, $\mu_{B_k}(y_i) = z$ e $w_{d^{(k)}} = v$, onde X é uma matriz de dimensões $n \times D^{(k)}$, z é um vetor $n \times 1$ e v é um vetor $D^{(k)} \times 1$. A seguir é apresentado o pseudo-código do método.

Algorithm 3 Método das Variáveis Ativas

- 1: Passos Iniciais:
 - Definição de X e z e remoção de regras inativas (colunas de X totalmente zeradas)
 - Defina w^* como a solução final
 - 2: Ache v^* que solucione o sistema linear (B-7)
 - 3: $w^* = v^*$
 - 4: **while** dw^* não satisfazer (B-3)
 - 5: Encontre os elementos em w^* que não satisfazem (B-3) e atribua o valor 0
 - 6: Exclua as colunas de X relativas aos elementos de w^* que foram zerados
 - 7: Ache v^* que solucione o sistema linear (B-7)
 - 8: Substitua em w^* os novos v^* computados
 - 9: **end while**
 - 10: Retorne w^*
-

O sistema linear (B-7) a ser resolvido é dado pela minimização do Lagrangeano:

$$L(v, \lambda) = (Xv - z)^T(Xv - z) + \lambda(\mathbf{1}^T v - 1) \quad (\text{B-4})$$

no qual λ representa o multiplicador de Lagrange associado a restrição de igualdade ($\mathbf{1}^T v = 1$) e $\mathbf{1} = [1, 1, \dots, 1]_{D^{(k)} \times 1}^T$. A condição de 1º ordem é dado pelo ponto em que o vetor gradiente ∇ se anula em todas as v direções:

$$\begin{aligned} \nabla[L(v, \lambda)] &= 0 \\ X^T Xv - X^T z + \lambda \mathbf{1} &= 0 \\ X^T Xv + \lambda \mathbf{1} &= X^T z \end{aligned} \quad (\text{B-5})$$

além de:

$$\begin{aligned} \frac{dL(v, \lambda)}{d\lambda} &= 0 \\ \mathbf{1}^T v - 1 &= 0 \\ \mathbf{1}^T v &= 1 \end{aligned} \quad (\text{B-6})$$

ao unir (B-5) e (B-6) verifica-se a necessidade de se encontrar v e λ que satisfaçam ambas as equações. Então, forma-se um sistema linear com ambos os resultados:

$$\begin{bmatrix} X^T X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} v \\ \lambda \end{bmatrix} = \begin{bmatrix} X^T z \\ 1 \end{bmatrix} \quad (\text{B-7})$$

Existe diversas formas de encontrar v e λ : eliminação gaussiana, invertendo a matriz $\begin{bmatrix} X^T X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix}$, usando métodos iterativos como gradientes conjugados, entre outros. É fácil ver que a condição de 2º ordem (mínimo local) é satisfeita, bastando mostrar que a matriz Hessiana H do Lagrangeano com respeito a v é positiva definida (ou semi-definida):

$$H[L(v, \lambda)] = X^T X = w^T X^T X w = (Xw)^T Xw = \|Xw\|_2^2 \geq 0 \quad (\text{B-8})$$

como a Hessiana de $L(v, \lambda)$ é positiva semi-definida e independe de v , a solução v^* proveniente do sistema linear (B-7) é mínimo global sujeito à restrição de igualdade.

B.2 Cálculo dos coeficientes da Matriz β

O problema original é encontrar uma solução que minimize:

$$\min : \|X\beta - Z\|_2^2 \quad (\text{B-9})$$

$$sj : \beta \mathbf{1} = \mathbf{1} \quad (\text{B-10})$$

$$\beta_{d^{(k)}_k} \geq 0 \quad (\text{B-11})$$

onde $X = [\mu_{A_1^{(k)}}(\mathbf{x}_i), \dots, \mu_{A_{D^{(k)}}}(\mathbf{x}_i)]$ é a matriz cujas colunas são as $\mu_{A_{d^{(k)}}}(\mathbf{x}_i)$. O $Z = [\mu_{B_k}(y_i) \ \mu_{B_2}(y_i) \ \dots \ \mu_{B_J}(y_i)]$ é a matriz, com colunas que perfazem o grau de pertinência de todo y_i a cada k -ésimo conjunto fuzzy. Também, $\mathbf{1} = [1 \ 1 \ \dots \ 1]^T$ com $D^{(k)}$ linhas e uma coluna, e: $\beta =$

$$\begin{bmatrix} b_{11} & \dots & b_{1K} \\ b_{21} & \dots & b_{2K} \\ \dots & \dots & \dots \\ b_{D^{(k)}1} & \dots & b_{D^{(k)}K} \end{bmatrix}$$

é a matriz de pesos com $D^{(k)}$ linhas e K colunas, positivos, tal que a soma dos pesos de cada $\mu_{A_{d^{(k)}}}(\mathbf{x}_i)$ por k -ésimo conseqüente é igual a 1. Por fim, $\|\cdot\|_2$ é a norma de Frobenius. A solução do problema (B-10) é praticamente idêntico ao apresentado na seção anterior, sendo necessário cumprir os seguintes passos:

1. Formula-se o problema sem as restrições de desigualdade.
2. Soluciona-se o sistema linear (B-12).
3. Dada a matriz de soluções β , aplica-se o algoritmo das variáveis ativas, isto é, verifica-se se existe algum elemento β negativo:
 - Caso não haja, retorna-se a matriz β para o transcorrimto do modelo GPFIS;
 - Caso contrário, remove-se as $\mu_{A_{d(k)}}(\mathbf{x}_i)$ que estão com coeficiente negativo, atribuindo um valor 0 para as linhas de coeficientes negativos. Repete-se o procedimento a partir do 2º item.

O sistema linear a ser solucionado é dado por:

$$\begin{bmatrix} X^T X & R \\ R^T & N \end{bmatrix} \begin{bmatrix} \beta \\ \lambda \end{bmatrix} = \begin{bmatrix} X^T Z \\ K \end{bmatrix} \quad (\text{B-12})$$

onde:

$$R = \begin{bmatrix} 1 & \dots & 1 \\ 1 & \dots & 1 \\ \dots & \dots & \dots \\ 1 & \dots & 1 \end{bmatrix} \text{ e } N = \begin{bmatrix} 0 & \dots & 0 \\ 0 & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & 0 \end{bmatrix}$$

tal que as dimensões da matriz R são D linhas e D colunas, a N é uma matriz quadrada com D linhas e D colunas e K é uma matriz com D linhas e J colunas. Por fim, λ são os multiplicadores de Lagrange associados a cada restrição de igualdade.

B.3

Prova da Desigualdade do DMA

A partir da expressão do DMA (4-44), mais a (4-7) e (4-13), e considerando cada $\hat{\mu}_{B_k}(y_i)$ já normalizado fornece:

$$\frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} = \frac{\sum_{i=1}^n |c_1 \mu_{B_1}(y_i) + \dots + c_K \mu_{B_K}(y_i) - (c_1 \hat{\mu}_{B_1}(y_i) + \dots + c_K \hat{\mu}_{B_K}(y_i))|}{n} \quad (\text{B-13})$$

rearrumando a (B-13), tem-se:

$$\frac{\sum_{i=1}^n |c_1 (\mu_{B_1}(y_i) - \hat{\mu}_{B_1}(y_i)) + \dots + c_J (\mu_{B_K}(y_i) - \hat{\mu}_{B_K}(y_i))|}{n} \quad (\text{B-14})$$

usando a Formulação do modelo GPFIS para Previsão/Regressão, sabe-se que $\mu_{B_k}(y_i) - \hat{\mu}_{B_k}(y_i) = \varepsilon_{ik}$. Então:

$$\frac{\sum_{i=1}^n |c_1 \varepsilon_{i1} + \dots + c_K \varepsilon_{iK}|}{n} \leq \frac{\sum_{i=1}^n [|c_1 \varepsilon_{i1}| + \dots + |c_K \varepsilon_{iK}|]}{n} \quad (\text{B-15})$$

a segunda parte implicada pela desigualdade triangular. Observa-se que:

$$\begin{aligned} \sum_{i=1}^n |c_k \varepsilon_{ik}| &= |c_k \varepsilon_{1k} + c_k \varepsilon_{2k} + \dots + c_k \varepsilon_{nk}| \leq \\ &|c_k \varepsilon_{1k}| + |c_k \varepsilon_{2k}| + \dots + |c_k \varepsilon_{nk}| = |c_k| \sum_{i=1}^n |\varepsilon_{ik}| = \\ &= |c_k| \|\varepsilon_k\|_1 \end{aligned} \quad (\text{B-16})$$

usando esse fato para todo k , a expressão (B-15) exhibe:

$$\begin{aligned} \frac{\sum_{i=1}^n [|c_1 \varepsilon_{i1}| + \dots + |c_K \varepsilon_{iK}|]}{n} &= \frac{\sum_{i=1}^n |c_1 \varepsilon_{i1}| + \dots + \sum_{i=1}^n |c_K \varepsilon_{iK}|}{n} \\ &= \frac{|c_1| \|\varepsilon_1\|_1 + \dots + |c_K| \|\varepsilon_K\|_1}{n} = \frac{\sum_{k=1}^K |c_k| \|\varepsilon_k\|_1}{n} \end{aligned} \quad (\text{B-17})$$

C

Testes Estatísticos Não-Paramétricos

Este Anexo apresenta alguns testes estatísticos ditos não-paramétricos. O termo não-paramétrico nesta situação específica exprime o fato de que estes não dependem da distribuição marginal ou conjunta de probabilidade das variáveis aleatórias sobre consideração. Ao leitor interessado na discussão sobre testes ou métodos paramétricos e não-paramétricos ver em Conover [47]. Em suma, este apêndice trata de quatro testes não-paramétricos usados ao longo do texto:

1. Teste do Sinal
2. Teste de Wilcoxon
3. Teste de Friedman (Iman e Davenport)
4. Teste (Critério) de Holm

De fato, os 2 primeiros testes tratam de comparação entre pares de realizações de duas variáveis aleatórias (resultados de dois algoritmos como: número de regras, acurácia, etc.), buscando identificar qual possui valores maiores ou menores. O terceiro generaliza o teste do Sinal, ao possibilitar efetuar a comparação entre n-uplas de realização de variáveis aleatórias. O teste de Iman e Davenport é um refinamento do de Friedman, ao maximizar o poder do teste para poucas amostras. Contudo, ambos examinam, por exemplo, se um algoritmo ou um subconjunto possibilitam melhores resultados do que os demais, não tendo em conta qual especificamente. Logo, é premente efetuar múltiplas comparações, visando à comparar uma a uma das realizações das variáveis aleatórias. Neste instante, surge o Teste (Critério) de Holm para corrigir o nível de significância e possibilitar uma inferência mais segura.

Os detalhes de cada teste são descritos em 5 etapas:

1. Base de dados
2. Pressupostos
3. Hipóteses

4. Estatística de Teste

5. Regra de Decisão

Cabe ressaltar a terminologia usada ao longo da descrição dos testes e sua relação com o tema SFGs e inteligência computacional:

- variável aleatória - acurácia, número de regras, tempo de máquina, entre outros de um algoritmo.
- amostra aleatória - realizações da variável aleatória, que por definição são independentes e identicamente distribuídos.
- tratamento - algoritmo, configuração ou método em análise.
- bloco - base de dados.

Por fim, todos os testes aqui apresentados são para tratamentos relacionados ou blocados (related), isto é, os resultados expressos por cada algoritmo estão sujeitos as bases de dados (ver Tabela C.1 como exemplo). Para testes que não é tido em consideração o bloco (unrelated) ver Conover [47], que apresenta por exemplo o Mann-Whitney e o Kruskal-Wallis, ambos similares aos testes de Wilcoxon e Friedman, respectivamente. Em todos os casos somente é focado o formato bicaudal para os testes, ou seja, é examinado o caso que um algoritmo produz resultados diferentes (maiores ou menores) do que outros algoritmos. Isto deve-se ao maior poder deste testes (menor a probabilidade de aceitar uma hipótese quando esta é falsa). Caso haja um resultado significativo, a verificação de qual obteve melhor desempenho é realizada por inspeção. Outras referências sobre testes não-paramétricos na área de inteligência computacional são de Demsar [65], Graczyk et al. [92] e Derrac et al. [66]. Na dissertação os programas utilizados para efetuar cada teste foram o R [174] e o KEEL [3].

Tabela C.1: Exemplo de resultados para os testes estatísticos em análise.

Base de Dados	Algoritmo 1	Algoritmo 2	...	Algoritmo K
Iris	100,00	89,56	...	98,45
Wine	96,54	89,54	...	98,16
.
.
.
Bupa	50,45	56,43	...	70,18

C.1

Teste do Sinal

A ideia por trás do Teste do Sinal é verificar se uma variável aleatória X em um par (X, Y) tende a possuir valores maiores do que a outra variável aleatória Y . Uma característica do teste do sinal é que este não leva em conta a magnitude da diferença. Isto por um lado é positivo com respeito a situações onde a medida de acurácia é afetada pela dimensionalidade do problema (erro quadrático médio, por exemplo) e, portanto, ter em conta a magnitude da diferença dos resultados entre as bases de dados não faz sentido (ver Capítulo 5, na seção Comparação entre SFGs uma situação típica). Por outro lado, em momentos onde a magnitude é plausível, este pode levar a um erro de conclusão do pesquisador.

C.1.1

Base de Dados

Os dados consistem de uma amostra aleatória bivariada $\{(X_1, Y_1), (X_1, Y_1), \dots, (X_n, Y_n)\}$ de tamanho n . Dado cada i -ésimo par (X_i, Y_i) , uma comparação é feita de modo a contar os sinais positivos: $X_i > Y_i$; negativos: $X_i < Y_i$; e empates: $X_i = Y_i$ para todo i .

C.1.2

Pressupostos

1. As variáveis aleatórias X_i e Y_i , $i = 1, \dots, n$ são mutualmente independentes (o resultado em um base de dados não afeta o obtido em outra).
2. A escala de medida é no mínimo ordinal em cada par.

C.1.3

Hipóteses

Denote por $P(+)$ a probabilidade de obter um sinal positivo e $P(-)$ o negativo. Então:

- $H_0: P(+) = P(-)$
- $H_1: P(+) \neq P(-)$

Há diversas interpretações possíveis a partir desta formulação. Uma destas é X_i e Y_i não possuem o mesmo parâmetro de locação. Isto é, ou X_i e Y_i não possuem a mesma média ($E(X_i) \neq E(Y_i)$) ou mediana.

C.1.4 Estatística de Teste

Seja T a estatística de teste. O seu valor é dado por:

$$T = \text{número total de '+'} \quad (\text{C-1})$$

pela estatística de teste ser uma função de variáveis aleatórias, esta também é uma variável aleatória com uma distribuição específica. Sob hipótese H_0 pode se demonstrar que T tem uma distribuição binomial.

C.1.5 Regra de Decisão

A regra de decisão depende de dois parâmetros:

- n = número de amostras.
- α = nível de significância ($\alpha \in [0, 1]$).

o primeiro destes é facilmente computado, enquanto que o segundo é estabelecido à priori pelo pesquisador. Normalmente $\alpha = 0,05$, ou seja, um resultado só é significativo se a probabilidade de se obter um valor igual a T for menor do que 0,05. O α deve ser escolhido de forma a minimizar o erro tipo I (falso positivo, isto é, desacreditar na hipótese padrão quando esta era verdadeira).

Em linhas gerais, para $n \leq 20$ rejeite a hipótese H_0 :

- Se $T \leq t$
- Se $T \geq n - t$

para achar o valor de t é necessário recorrer a tabela Binomial (ver em Conover [47]), buscando o valor de t correspondente a n e $\alpha/2$ (bicaudal).

Para $n > 20$, é efetuada uma aproximação de t por uma distribuição Normal, dada por:

$$t = 1/2(n + z_{\alpha/2}\sqrt{n}) \quad (\text{C-2})$$

onde $z_{\alpha/2}$ é obtido pela tabela de distribuição Normal. Para $\alpha = 0,05$ a equação (C-2) é aproximada por:

$$t \approx n/2 - \sqrt{n} \quad (\text{C-3})$$

C.2

Teste de Wilcoxon

O teste de Wilcoxon surge com a ideia de ter em conta a magnitude da diferença entre os pares de variáveis aleatórias. As hipóteses são praticamente semelhantes, com a inserção de maior ênfase na comparação entre as medianas das distribuições. Em termos de pressupostos é adicionado um referente a simetria da distribuição.

C.2.1

Base de Dados

Os dados consistem de uma amostra aleatória bivariada $\{(X_1, Y_1), (X_1, Y_1), \dots, (X_n, Y_n)\}$ de tamanho n . Dado cada i -ésimo par (X_i, Y_i) , é computada a diferença absoluta dada por: $D_i = |X_i - Y_i|$ e o sinal da diferença: $S_i = 1$ se $X_i > Y_i$ e $S_i = -1$ caso contrário.

Após computar cada $\{D_1, \dots, D_n\}$ ranqueie ou posicione as diferenças no sentido crescente (posto 1 para o menor e n para a maior). Caso haja empates entre diversos pares, forneça a média das posições empatas, isto é, suponha que haja empate na 3^o, 4^o, 5^o e 6^o. Para este conjunto de posições forneça o posto: $(3 + 4 + 5 + 6)/4 = 4,5$.

C.2.2

Pressupostos

1. Os D_i são mutualmente independentes.
2. Os D_i possuem a mesma mediana.
3. A escala de medida de cada D_i deve ser no mínimo intervalar.
4. A distribuição de cada D_i é simétrica.

A definição de simetria na distribuição é dada por:

Definição 7. *A distribuição de uma variável aleatória X é simétrica em torno de uma constante c , se $P(X \leq c - x) = P(X \geq c + x)$*

Se uma distribuição é simétrica e a média existe, então a média e a mediana coincidem (distribuição Normal, por exemplo), por estarem posicionadas no centro da distribuição (no caso em c).

C.2.3

Hipóteses

Seja $d_{.50}$ a mediana comum dos D_i . Então:

- $H_0: d_{.50} = 0$
- $H_1: d_{.50} \neq 0$

C.2.4

Estatística de Teste

Seja T a estatística de teste. Denote por R_i o posto com sinal, dado por: $R_i = R(D_i)S_i$, onde $R(D_i)$ representa o posto de D_i computado anteriormente. A partir disso, tem-se a estatística de teste, sob H_0 , dada por:

$$T = \frac{\sum_{i=1}^n R_i}{\sqrt{\sum_{i=1}^n R_i^2}} \quad (\text{C-4})$$

C.2.5

Regra de Decisão

A regra de decisão depende de dois parâmetros:

- n = número de amostras.
- α = nível de significância ($\alpha \in [0, 1]$).

Em linhas gerais, para $n \leq 50$ rejeite a hipótese H_0 :

- Se $T \leq w_{\alpha/2}$
- Se $T \geq w_{1-\alpha/2}$

para achar o valor de $w_{\alpha/2}$ e $w_{1-\alpha/2}$ deve-se utilizar a tabela específica para o teste [47]. Quando $n > 50$ pode-se usar a aproximação Normal dada por:

$$w_{\alpha/2} = [n(n+1)/4] + z_{\alpha/2} \sqrt{n(n+1)(2n+1)/24} \quad (\text{C-5})$$

C.3

Teste de Friedman

O teste de Friedman é a generalização do teste do Sinal. Este se propõe avaliar, dada uma medida de desempenho, qual o algoritmo, ou algoritmos obtiveram o posto significativamente diferente dos demais métodos em comparação. Portanto, ao comparar 3 ou mais modelos de forma simultânea, o teste de Friedman examina qual obteve a maior acurácia, menor erro quadrático médio, entre outros.

C.3.1

Base de Dados

Considere a matriz de resultados (Tabela C.2).

Tabela C.2: Exemplo de tabela de resultados para o teste de Friedman.

Base de Dados	Algoritmo 1	Algoritmo 2	Algoritmo 3	R_1	R_2	R_3
Iris	100,00	89,56	98,45	1	3	2
Wine	96,54	89,54	98,16	2	3	1
.
.
.
Bupa	50,45	56,43	70,18	3	2	1

Considerando que a medida de desempenho é a acurácia, logo, quanto maior esta, melhor posicionado o j -ésimo algoritmo deve estar. Então, cada $R(X_{ij})$ é o posto do j -ésimo método na i -ésima base de dados. Por exemplo, o algoritmo 1 obteve posto 1 na Base Iris, 2 na Wine e 3 na Bupa. Caso ocorresse um empate, a média das posições é colocada no lugar, semelhantemente ao caso do teste de Wilcoxon. Obviamente, os dados consistem de uma amostra aleatória K -variada $\{(X_{11}, X_{12}, \dots, X_{1K}), \dots, (X_{n1}, X_{n2}, \dots, X_{nK})\}$ de tamanho n .

C.3.2

Pressupostos

1. Os $(X_{i1}, X_{i2}, \dots, X_{iK})$ são mutualmente independente, isto é, o resultado obtido em uma base de dados não afeta o auferido em outra.
2. Em cada bloco as observações podem ser posicionadas por algum critério.

C.3.3

Hipóteses

- H_0 : Os posicionamentos obtidos por cada algoritmo são igualmente prováveis (ou seja, as diferentes configurações, ou modelos possuem desempenho semelhante).
- H_1 : No mínimo um dos algoritmos tende a obter resultados superiores/inferiores com respeito a algum outro método.

C.3.4

Estatística de Teste

1. Compute $R_j = \sum_{i=1}^n R(X_{ij})$.
2. Calcule o termo $A = \sum_{i=1}^n \sum_{j=1}^K [R(X_{ij})]^2$.
3. Então, o termo $B = 1/n \sum_{j=1}^K (R_j)^2$.

O termo A refere-se a variação entre os blocos e tratamentos (chamado de soma dos quadrados totais), enquanto que o B capta a variação entre os tratamentos (soma dos quadrados dos tratamentos).

A forma usual do teste de Friedman é dada por:

$$T_F = \frac{(K - 1)[nB - n^2K(K + 1)^2/4]}{A - nK(K + 1)^2/4} \quad (C-6)$$

com T_F distribuído por uma Qui-Quadrado com $K - 1$ graus de liberdade. Se $A = B$, considere o ponto na região crítica e calcule a probabilidade da hipótese H_0 ser aceita, como: $\hat{\alpha} = (1/K)^{n-1}$

Iman e Davenport propõe uma outra estatística de teste:

$$T_{ID} = \frac{(n - 1)[B - nK(K + 1)^2/4]}{A - B} \quad (C-7)$$

com T_{ID} distribuído por uma F com $K - 1$ e $(n - 1)(K - 1)$ graus de liberdade.

C.3.5

Regra de Decisão

A regra de decisão depende de três parâmetros:

- n = número de blocos (bases de dados).
- K = número de tratamentos (algoritmos).
- α = nível de significância ($\alpha \in [0, 1]$).

Rejeite a hipótese H_0 se T_{ID} exceder o quantil $1 - \alpha$ da distribuição F com $K - 1$ e $(n - 1)(K - 1)$ graus de liberdade.

C.4 Teste de Holm

De fato, o teste de Holm não é necessariamente um teste, mas sim um critério de correção no nível de significância α estabelecido previamente pelo usuário. Mas, por conveniência será usado este rótulo. Contudo, o próximo tópico apresenta o leitor brevemente o conceito de p-valor, relevante para a compreensão plena dos resultados.

C.4.1 P-valor

Como descrito em tópicos anteriores, toda estatística de teste é uma função de variável aleatória, e, portanto, esta também se torna uma variável aleatória. Na maioria dos testes, trabalha-se com variáveis aleatórias contínuas (exceto o caso da Binomial, mas que em uma amostra de tamanho razoável pode ser aproximada por uma Normal). Então, suponha que a estatística de teste T possua uma distribuição de probabilidade conforme Figura C.1.

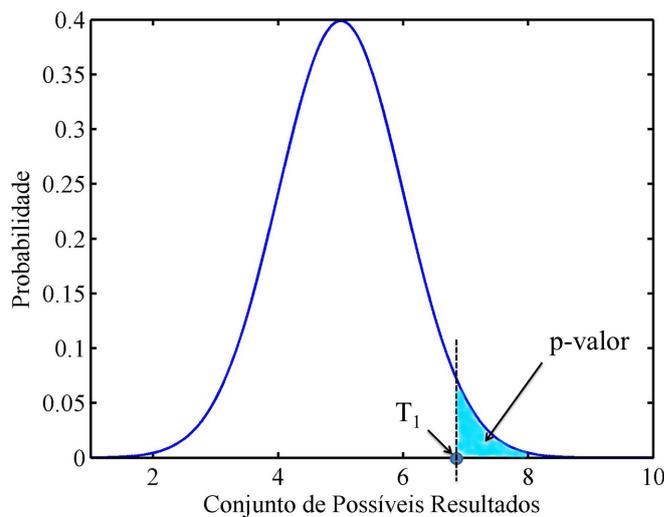


Figura C.1: Exemplo de distribuição de probabilidade.

Suponha uma realização de T , denominada de T_1 . Neste caso específico, o p-valor é a área embaixo da curva na região a direita de T_1 , ou seja, $P(T \geq T_1)$. Este representa a probabilidade da variável aleatória atingir um valor maior ou igual T_1 . Caso o valor de T_1 represente uma estatística que diz respeito a desvio entre médias, então quanto menor $P(T \geq T_1)$ maior é T_1 . Como a maior parte, senão todos os testes estatísticos são construídos sob hipótese H_0 , o p-valor representa a probabilidade de não rejeitar a H_0 . Quando o p-valor é menor do que α diz se que a hipótese H_0 é rejeitada, isto é, a probabilidade

da hipótese nula estar correta (p-valor) é menor do que o nível de proteção estabelecido pelo pesquisador com respeito a um falso positivo (α).

C.4.2 Múltiplas Comparações

Caso o resultado do teste de Friedman resulte em rejeição da hipótese H_0 , torna-se necessário efetuar múltiplas comparações entre os algoritmos, de maneira que se descubra quais diferem significativamente entre si. A ideia por trás do teste de Holm é utilizar duas técnicas:

- Corrigir o nível de significância por: $\frac{\alpha}{(h-1+k)}$, onde h é o número de comparações até então efetuadas e k é a posição do algoritmo na média dos postos computados.
- Começar os confrontamentos entre os algoritmos pelos que possuem menor posto médio.

A justificativa do primeiro item diz respeito ao conceito de reduzir a probabilidade de se aceitar uma hipótese, quando na verdade esta era falsa. Considere por exemplo um conjunto de K hipóteses, por exemplo, dadas por:

- H_0 : Algoritmo k não tem posto menor do que o 2.
- H_1 : Algoritmo k tem posto menor do que o 2.

Para computar o verdadeiro nível de significância α , devido as múltiplas comparações, é necessário observar que o evento a ser testado é: $P(RejH_0^1|VerdH_0^1 \cap RejH_0^2|VerdH_0^2 \cap \dots \cap RejH_0^K|VerdH_0^K)$ e não mais o isolado $P(RejH_0|VerdH_0)$ onde bastava o α . Como o resultado de uma não interfere na outra (são independentes), tem-se: $P(RejH_0^1|VerdH_0^1)P(RejH_0^2|VerdH_0^2)\dots P(RejH_0^K|VerdH_0^K)$.

Então sabe-se que a probabilidade de cometer um erro do tipo I é dada:

$$\hat{\alpha} = P(RejH_0^1|VerdH_0^1 \cap \dots \cap RejH_0^K|VerdH_0^K) \quad (C-8)$$

$$= 1 - P(AceitoH_0^1|VerdH_0^1 \cap \dots \cap AceitoH_0^K|VerdH_0^K) \quad (C-9)$$

$$= 1 - P(AceitoH_0^1|VerdH_0^1)\dots P(AceitoH_0^K|VerdH_0^K) \quad (C-10)$$

$$= 1 - \prod_{k=1}^K P(AceitoH_0^k|VerdH_0^k) \quad (C-11)$$

o termo $\prod_{k=1}^K P(AceitoH_0^k|VerdH_0^k)$ vai para zero quando K é grande. Na

prática, o K dificilmente é grande, e, portanto, a forma de reduzir a probabilidade de se cometer um erro do tipo I é corrigir o nível de significância, fator este que governa a aceitação/rejeição da hipótese nula.

Nesta dissertação será usado o conceito de algoritmo de controle. Este se refere ao algoritmo que obteve o menor posto dentro os demais. Em confronto a este todas as demais comparações são efetuadas. Logo, Holm propõe o seguinte critério para a proteção contra o erro tipo I:

1. Considere R_0 o algoritmo com menor posto médio. Seja $h = 1, \dots, K - 1$ o número de comparações.
2. Calcule o p-valor da comparação. através do γ que satisfaz a igualdade:

$$|R_0 - R_h| = t_{1-\gamma/2} \left[\frac{2n(A - B)}{(n - 1)(K - 1)} \right] \quad (C-12)$$

onde $t_{1-\gamma/2}$ com quantis $1 - \gamma/2$ e $(n - 1)(K - 1)$ graus de liberdade.

3. Ordene o p-valor obtido em cada h-ésima comparação, do maior para o menor. Denote por $p_{(1)}, \dots, p_{(K-1)}$ esta sequência enumerada.
4. Rejeite a hipótese H_0 se: $p_{(h)} < \frac{\alpha}{h}$, para todo h .

Nos resultados apresentados, este processo é apresentado em uma Tabela, conforme a seguir:

Tabela C.3: Resultados do teste de Friedman e Holm para o Experimento.

i	Método	Posto	
3	Prod	3,4706	
2	Prod+RQ	3,1471	
1	Prod+Neg	2,0000	
0	Prod+RQ+Neg	1,3824	
Teste	p-valor		
Friedman	< 0,0001		
Iman e Daveport	< 0,0001		
Método	$z = (R_0 - R_i)/SE$	p-valor	Holm
Prod	4,7158	<0,0001	0,0167
Prod+RQ	3,9852	<0,0001	0,025
Prod+Neg	1,3948	0,1631	0,050

As primeiras linhas exibem o posto médio obtido por cada configuração. Abaixo é apresentado o p-valor obtido no teste de Friedman e Iman e Davenport. A partir do algoritmo que obteve o menor posto (no caso Prod+RQ+Neg), este é usado como método de controle para efetuar

as múltiplas comparações. Logo abaixo é apresentado as confrontações de cada h-ésimo algoritmo (Prod, Prod+RQ e Prod+Neg) com respeito ao Prod+RQ+Neg. É apresentado o p-valor obtido nas comparações e tanto o nível de significância (coluna Holm) quanto o ordenamento são corrigidos pelo teste de Holm. Neste caso, rejeita-se H_0 na comparação entre Prod+RQ+Neg:Prod e Prod+RQ+Neg:Prod+RQ, logo, o posto médio do Prod+RQ+Neg foi menor significativamente do que destes outros algoritmos. No caso do paralelo entre Prod+RQ+Neg:Prod+Neg não se observa diferença significativa.