

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E
INFORMÁTICA INDUSTRIAL

LEONARDO GOMES DE MELO

**SISTEMAS *FUZZY* PROBABILÍSTICOS: GERAÇÃO AUTOMÁTICA
DE REGRAS E DEFUZZIFICAÇÃO BAYESIANA**

DISSERTAÇÃO

CURITIBA

2011

LEONARDO GOMES DE MELO

**SISTEMAS *FUZZY* PROBABILÍSTICOS: GERAÇÃO AUTOMÁTICA
DE REGRAS E DEFUZZIFICAÇÃO BAYESIANA**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de “Mestre em Ciências” – Área de Concentração: Informática Industrial.

Orientadora: Prof^a. Dr^a. Myriam Regattieri Delgado

Co-orientador: Prof. Dr. Luís Alberto Lucas

CURITIBA

2011

Dados Internacionais de Catalogação na Publicação

M528 Melo, Leonardo Gomes de
 Sistemas fuzzy probabilísticos : geração automática de regras e defuzzificação bayesiana /
 Leonardo Gomes de Melo. — 2011.
 91 f. : il. ; 30 cm

Orientadora: Myriam Regattieri Delgado.

Coorientador: Luís Alberto Lucas.

Dissertação (Mestrado) – Universidade Tecnológica Federal do Paraná. Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial. Curitiba, 2011.

Bibliografia: f. 76-79.

1. Sistemas difusos. 2. Probabilidades. 3. Teoria bayesiana de decisão estatística. 4. Algoritmos. 5. Simulação (Computadores). 6. Engenharia elétrica – Dissertações. I. Delgado, Myriam Regattieri, orient. II. Lucas, Luís Alberto, co-orient. III. Universidade Tecnológica Federal do Paraná. Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial. IV. Título.

CDD (22. ed.) 621.3

Biblioteca Central da UTFPR, Campus Curitiba

Título da Dissertação Nº 553:


“Sistemas Fuzzy Probabilísticos: Geração Automática de Regras e Defuzzificação Bayesiana”


por


Leonardo Gomes de Melo

Esta dissertação foi apresentada como requisito parcial à obtenção do grau de MESTRE EM CIÊNCIAS – Área de Concentração: Informática Industrial, pelo Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial – CPGEI – da Universidade Tecnológica Federal do Paraná – UTFPR – Campus Curitiba, às 9h do dia 28 de fevereiro de 2011. O trabalho foi aprovado pela Banca Examinadora, composta pelos professores:


Prof.^a Myriam Regattieri Delgado, Dr.
(Presidente – UTFPR)


Prof. Maurício Fernandes Figueiredo, Dr.
(UFSCAR)


Prof. Cesar Augusto Taconeli, Dr.
(UTFPR)


Prof. Celso Antonio Alves Kaestner, Dr.
(UTFPR)

Visto da coordenação:


Prof. Fábio Kurt Schneider, Dr.
(Coordenador do CPGEI)

RESUMO

MELO, Leonardo. Sistemas *Fuzzy* Probabilísticos: Geração Automática de Regras e Defuzzificação Bayesiana. 92 f. Dissertação – Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2011.

O presente trabalho tem por objetivo propor métodos inovadores para a obtenção automática de sistemas de inferência unindo teorias bem conhecidas para lidar com incerteza, como probabilidade e sistemas *fuzzy*. Assim, as principais contribuições do trabalho podem ser divididas em dois itens: obtenção automática de regras e defuzzificação. No primeiro caso, propõe-se um método automático de geração de regras em Sistemas *Fuzzy* Probabilísticos (SFP) e no segundo, busca-se apresentar um novo método de defuzzificação baseado em inferência bayesiana. A geração automática de regras é inspirada no método Wang-Mendel e prevê a obtenção de regras *fuzzy* com consequentes probabilísticos de forma que todas as informações obtidas do conjunto de dados de treinamento sejam consideradas. A segunda proposta diz respeito à utilização de inferência bayesiana na fase de defuzzificação, com o intuito de incorporar informações a priori no funcionamento do sistema. Neste trabalho, o método de geração automática de regras, o qual dá origem a regras *fuzzy* probabilísticas, é testado tanto em um classificador *fuzzy* probabilístico quanto em um sistema de inferência *fuzzy* probabilístico. O classificador *fuzzy* probabilístico tem seu desempenho comparado com dois classificadores *fuzzy* tradicionais: um obtido pelo método de Wang Mendel original e o outro, aqui denominado de método de Ishibuchi, no qual as regras *fuzzy* apresentam diferentes níveis de confiança. O sistema de inferência *fuzzy* probabilístico é comparado com um sistema *fuzzy* obtido pelo método Wang Mendel tradicional através da aplicação em previsão de séries temporais. Os resultados mostram que o método de geração automática aqui proposto é superior ao método tradicional, em especial quando há um conjunto significativo de treinamento e dados com ruído. No caso da defuzzificação bayesiana, os resultados apresentam-se similares ao do SFP simples, embora a informação a priori possibilite uma forma de suavização da saída obtida.

Palavras-chave: Sistemas *Fuzzy*. Probabilidade. Geração de Regras. Wang Mendel. Inferência Bayesiana.

ABSTRACT

MELO, Leonardo. Probabilistic Fuzzy Systems: Automatic Rule Design and Bayesian Defuzzification. 92 f. Dissertação – Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2011.

The present work aims to propose innovative methods for the automatic design of inference systems by joining well known theories to treat uncertainty, such as probability and *fuzzy* systems. Therefore, the main work contributions can be divided in two items: automatic rule design and defuzzification. In the first case, we propose an automatic method for rules generation in Probabilistic Fuzzy Systems (PFS) and in the second one, we intend to present a new defuzzification method based on bayesian inference. The automatic rule design is inspired in the Wang-Mendel method and provides *fuzzy* rules with probabilistic consequents, in the way that all the obtained information from training data are considered. The second proposal concerns the use of bayesian inference in defuzzification phase, with the aim of incorporating a priori information in the operation of the system. In this work, the method of automatic rule design, which leads to probabilistic *fuzzy* rules, is tested in a probabilistic *fuzzy* classifier as well as in a probabilistic inference system. The probabilistic *fuzzy* classifier has its performance compared with two traditional *fuzzy* classifiers: one obtained by the original Wang-Mendel method and the other, named Ishibuchi method, considers *fuzzy* rules with different trust levels. The probabilistic inference system is compared with a *fuzzy* system obtained by the traditional Wang-Mendel method through the application in time series forecast. The results show that the automatic generation method being proposed outperforms the traditional method, especially when there is a significant training set and noisy data. In the case of the bayesian defuzzification, the results are similar to the simple PFS, although a priori information seems to provide a smoothness of the PFS's output.

Keywords: Fuzzy Systems. Probability. Rules Generation. Wang Mendel. Bayesian Inference.

LISTA DE FIGURAS

FIGURA 1 – EXEMPLO DE FUNÇÃO DE PERTINÊNCIA TRIANGULAR PARA VARIÁVEL ALTURA.	16
FIGURA 2 – EXEMPLO DE FUNÇÃO DE PERTINÊNCIA GAUSSIANA NA PARTIÇÃO DO UNIVERSO DA VARIÁVEL ALTURA.	16
FIGURA 3 – EXEMPLO DE FUNCIONAMENTO DE UM SISTEMA <i>FUZZY</i> ESPECÍFICO.	19
FIGURA 4 – a) FOU (INTERVALAR); b) GEFOU (GERAL).	20
FIGURA 5 – SIF PARA CONJUNTOS: (a) DO TIPO-2; b) NÃO-ESTACIONÁRIOS	21
FIGURA 6 – EXEMPLO DE UM CONJUNTO <i>FUZZY</i> PROBABILÍSTICO	24
FIGURA 7 – SISTEMA DE INFERÊNCIA <i>FUZZY</i> PROBABILÍSTICO	25
FIGURA 8 – (A)WM: PASSO2	46
FIGURA 9 – (B)WM: PASSO3	46
FIGURA 10 – EXEMPLO DE PARTIÇÃO DO ESPAÇO AMOSTRAL $T_{R=1}$	49
FIGURA 11 – EXEMPLO DE PARTIÇÃO PARA A VARIÁVEL DE ENTRADA A	51
FIGURA 12 – EXEMPLO DE PARTIÇÃO PARA A VARIÁVEL DE ENTRADA B	51
FIGURA 13 – EXEMPLO DE PARTIÇÃO DO ESPAÇO AMOSTRAL $T_{R=1}$ PARA O EXEMPLO NUMÉRICO	52
FIGURA 14 – CLASSES DE PONTOS ALEATÓRIOS PARA O EXPERIMENTO 1A	59
FIGURA 15 – CLASSES DE PONTOS ALEATÓRIOS PARA O EXPERIMENTO 1B	59
FIGURA 16 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 1A	60
FIGURA 17 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 1B	60
FIGURA 18 – GRÁFICOS DE COLUNA PARA OS PERCENTUAIS DE ACERTOS DOS EXPERIMENTOS DE CLASSIFICAÇÃO .	64
FIGURA 19 – EXEMPLO DE UMA SÉRIE TEMPORAL PARTICIONADA EM CONJUNTOS <i>FUZZY</i> .	65
FIGURA 20 – SÉRIE TEMPORAL PARA O EXPERIMENTO 2A.	66
FIGURA 21 – SÉRIE TEMPORAL PARA O EXPERIMENTO 2B.	66
FIGURA 22 – ESQUEMA DE PARTICIONAMENTO PARA A SÉRIE CONSUMO.	67
FIGURA 23 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 2A	68
FIGURA 24 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 2B	68
FIGURA 25 – RESULTADOS PARA A SÉRIE CONSUMO, 14 PASSOS À FRENTE.	73
FIGURA 26 – RESULTADOS PARA A SÉRIE BANESPA, 238 PASSOS À FRENTE.	73
FIGURA 27 – CLASSES DE PONTOS ALEATÓRIOS PARA O EXPERIMENTO 1C	90
FIGURA 28 – CLASSES DE PONTOS ALEATÓRIOS PARA O EXPERIMENTO 1D	90
FIGURA 29 – CLASSES DE PONTOS ALEATÓRIOS PARA O EXPERIMENTO 1E	90
FIGURA 30 – CLASSES DE PONTOS ALEATÓRIOS PARA O EXPERIMENTO 1F	90
FIGURA 31 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 1C	91
FIGURA 32 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 1D	91
FIGURA 33 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 1E	92

FIGURA 34 – FUNÇÕES DE PERTINÊNCIA PARA O EXPERIMENTO 1F	92
--	-------	----

LISTA DE TABELAS

TABELA 1	– BASE DE REGRAS DO EXEMPLO	52
TABELA 2	– BASE DE REGRAS DO EXEMPLO - REDUZIDA	52
TABELA 3	– BASE DE REGRAS DO EXEMPLO - FINAL	53
TABELA 4	– EXPERIMENTOS REALIZADOS PARA CLASSIFICAÇÃO	58
TABELA 5	– BASE OBTIDA PELO MÉTODO PROBABILÍSTICO (EXP. 1A E 1B)	61
TABELA 6	– PERCENTUAL DE ACERTOS - EXPERIMENTO 1A	62
TABELA 7	– PERCENTUAL DE ACERTOS - EXPERIMENTO 1B	62
TABELA 8	– MATRIZES DE CONFUSÃO - EXPERIMENTOS 1A E 1B	63
TABELA 9	– MATRIZES DE CONFUSÃO - EXPERIMENTOS 1C A 1F	69
TABELA 10	– BASE OBTIDA MÉTODO PROBABILÍSTICO (SÉRIE CONSUMO)	70
TABELA 11	– BASE OBTIDA MÉTODO PROBABILÍSTICO (SÉRIE BANESPA)	71
TABELA 12	– BASE OBTIDA MÉTODO PROBABILÍSTICO (SÉRIE BANESPA)	72
TABELA 13	– VALORES ESTIMADOS PREVISÃO DAS SÉRIES TEMPORAIS	72
TABELA 14	– DADOS DE TREINAMENTO	81
TABELA 15	– DADOS DE TESTE	81
TABELA 16	– DADOS DE TREINAMENTO	81
TABELA 17	– DADOS DE TESTE	83
TABELA 18	– BASE OBTIDA PELO MÉTODO WM - EXP. 1A	85
TABELA 19	– BASE OBTIDA PELO MÉTODO ISHIBUCHI - EXP. 1A	85
TABELA 23	– BASE OBTIDA PELO MÉTODO WM - EXP. 2B	85
TABELA 20	– BASE OBTIDA PELO MÉTODO WM - EXP. 1B	88
TABELA 21	– BASE OBTIDA PELO MÉTODO ISHIBUCHI - EXP. 1B	88
TABELA 22	– BASE OBTIDA PELO MÉTODO WM - EXP. 2A	89

SUMÁRIO

1	INTRODUÇÃO	11
1.1	CONSIDERAÇÕES INICIAIS	11
1.2	OBJETIVOS	13
1.2.1	Objetivo Geral	13
1.2.2	Objetivos Específicos	13
1.3	ORGANIZAÇÃO DO TEXTO	13
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	SISTEMAS <i>FUZZY</i>	14
2.1.1	Sistemas de Inferência <i>Fuzzy</i> do Tipo-1	15
2.1.2	Sistemas de Inferência <i>Fuzzy</i> do Tipo-2	18
2.1.3	Sistemas de Inferência <i>Fuzzy</i> Não-Estacionários	20
2.1.4	Sistemas de Inferência <i>Fuzzy</i> Probabilísticos	23
2.2	GERAÇÃO DE BASE DE REGRAS	26
2.2.1	Algoritmo de Wang-Mendel	26
2.2.2	Algoritmo de Ishibuchi	27
2.2.3	Outras Abordagens	28
2.3	MÉTODOS DEFUZZIFICADORES EM SISTEMAS DO TIPO-1	29
2.3.1	Médias e Máximos	30
2.3.2	Centro de Gravidade	31
2.3.3	Métodos Generalizadores	33
2.3.4	Conversão em Distribuição de Probabilidades	34
2.3.5	Outros Métodos	36
2.4	MÉTODOS DEFUZZIFICADORES EM SISTEMAS DO TIPO-2	39
2.4.1	Redução de Tipo	39
2.4.2	Defuzzificador Direto	40
2.4.3	Método da Amostragem	40
2.5	PROBABILIDADE E INFERÊNCIA BAYESIANA	41
2.5.1	Teoria das Probabilidades	42
2.5.2	Inferência Bayesiana	43
3	SISTEMAS <i>FUZZY</i> PROBABILÍSTICOS: GERAÇÃO AUTOMÁTICA DE REGRAS E DEFUZZIFICAÇÃO BAYESIANA	45
3.1	MÉTODO PARA GERAÇÃO DE REGRAS <i>FUZZY</i> PROBABILÍSTICAS	45
3.1.1	Algoritmo de Geração de Regras <i>Fuzzy</i> Probabilísticas	47
3.1.2	Consequentes Probabilísticos e o Teorema de Bayes	48
3.1.3	Um Exemplo Simples	50
3.2	PROCESSO DE INFERÊNCIA EM SISTEMAS <i>FUZZY</i> PROBABILÍSTICOS	53
3.3	DEFUZZIFICAÇÃO BAYESIANA EM SISTEMAS <i>FUZZY</i> PROBABILÍSTICOS	54
4	SIMULAÇÕES E RESULTADOS	57
4.1	EXPERIMENTO 1: GERAÇÃO DE REGRAS PARA UM CLASSIFICADOR <i>FUZZY</i>	58
4.1.1	Obtenção da BR para Classificadores	59
4.1.2	Desempenho	61

4.1.3	Variâncias Diferentes	63
4.2	EXPERIMENTO 2: SISTEMA <i>FUZZY</i> PROBABILÍSTICO PARA PREVISÃO DE SÉRIES TEMPORAIS	64
4.2.1	Obtenção da BR para Previsão de ST	65
4.2.2	Desempenho	67
5	CONCLUSÕES	74
5.1	DISCUSSÃO DOS RESULTADOS OBTIDOS	74
5.2	CONTRIBUIÇÕES E PUBLICAÇÕES	75
5.3	SUGESTÕES PARA TRABALHOS FUTUROS	76
	REFERÊNCIAS	77
	ANEXO A – DADOS DAS SÉRIES TEMPORAIS	81
A.1	SÉRIE TEMPORAL CONSUMO	81
A.2	SÉRIE TEMPORAL BANESPA	81
	APÊNDICE A – BASE DE REGRAS	85
	APÊNDICE B – PONTOS GERADOS E FUNÇÕES DE PERTINÊNCIA	90

1 INTRODUÇÃO

1.1 CONSIDERAÇÕES INICIAIS

A incerteza está presente na maioria das aplicações do mundo real. Há incertezas de diversos tipos, mas neste trabalho, dois tipos em especial serão considerados: a incerteza na forma de ambiguidade presente em informações vagas como “a temperatura hoje está alta” e a incerteza associada à probabilidade de ocorrência de um evento como na declaração “a probabilidade de chuva amanhã é 0.8”. O primeiro caso em geral é tratado pelos sistemas *fuzzy* (PEDRYCS, 1998), já o segundo é usualmente considerado na teoria das probabilidades (MEYER, 1983). O objetivo deste trabalho é unir estes dois paradigmas de forma a tratar conjuntamente ambos os tipos de incerteza.

Desde o advento da Teoria dos Conjuntos *Fuzzy*, busca-se conciliá-la com a Teoria das Probabilidades, a fim de se produzir um melhor tratamento da incerteza. Há extensa discussão sobre o tema na literatura, sendo alguns autores favoráveis ao uso de tal combinação, enquanto outros apenas limitam-se a comparar as duas teorias.

Com relação à comparação, pode-se citar, por exemplo, Laviolette e Jr (1994) e Wilson (1994). Os primeiros autores afirmam que a inferência bayesiana produz uma modelagem mais satisfatória para a incerteza do que os conjuntos *fuzzy*, sendo rebatidos segundo que levanta a questão de como se dar uma interpretação “satisfatória” para as funções de pertinência.

Outras abordagens, no entanto, procuram desenvolver sistemas híbridos para o tratamento da incerteza como os trabalhos de Colettia e Scozzafavab (2006) e Meghdadi e Akbarzadeh-T (2001). Todavia, a ideia de unir as duas teorias não é nova e outros autores também tiveram sucesso em desenvolver técnicas *fuzzy* com o auxílio da teoria das probabilidades, como por exemplo as variáveis aleatórias *fuzzy* (KWAKERNAAK, 1978), que dão margem a Distribuições de Probabilidade *Fuzzy* (TAYLOR; BOSWELL, 1987) e consequentemente, ao Teorema do Limite Central para esses tipos de variáveis; e ainda os conjuntos *fuzzy* probabilísticos descritos em Hirota (1981).

Em Colettia e Scozzafavab (2006), os autores rebatem o artigo de Zadeh (1965), o qual afirma que a probabilidade pode ser inadequada para tratar problemas usualmente considerados pela lógica *fuzzy*. Os autores discordam, afirmando que a probabilidade condicional fornece algumas das ferramentas necessárias para tratar os problemas levantados por Zadeh. Como exemplo citam o uso conjunto das duas teorias em casos específicos como o problema de calcular a probabilidade de retirada de uma bola grande dentro de uma caixa contendo bolas de diversos tamanhos. Percebe-se que o evento anterior pode ser classificado como *fuzzy* devido à ambiguidade presente na informação “bola grande”. Existe então a necessidade de se tratar conjuntamente as informações incertas em relação à ocorrência do evento e à ambiguidade na classificação das bolas (grande é um conceito vago). Há outras situações mais complexas. Pode-se imaginar, por exemplo, que se queira calcular a probabilidade de, além de ser escolhida uma bola grande, desta bola ser de determinada cor. Ou ainda, que não seja conhecida a fração de bolas de determinado tamanho. É possível pensar em várias situações neste exemplo com diferentes níveis de incerteza abrindo caminho para integrações (*fuzzy* e probabilidade) com elevados níveis de complexidade.

Assim, neste contexto de combinação ou cooperação, Meghdadi e Akbarzadeh-T (2001) formalizaram a Lógica *Fuzzy* Probabilística e Sistemas *Fuzzy* Probabilísticos (SFP). No sistema proposto por Meghdadi e Akbarzadeh-T (2001) ambos os tipos de incerteza são tratados na forma de uma base de regras fuzzy onde o consequente de cada regra é distribuído em diferentes conjuntos *fuzzy*, cada um associado a uma determinada medida de probabilidade. O que ocorre no sistema proposto por Meghdadi e Akbarzadeh-T (2001) é que algumas regras possuem diversos consequentes, cada um com diferente probabilidade de ocorrência, o que pode ser visto com um conjunto de regras de mesmo antecedente com consequentes diferentes. Essa metodologia cria mecanismos, portanto, de se tratar a inconsistência das regras provenientes de dados de treinamento através de um Sistema de Inferência *Fuzzy* Probabilístico (SFP).

A opção de se utilizar um SFP para o tratamento da incerteza abre inúmeras possibilidades. No entanto surgem questões do tipo:

- Como definir adequadamente as regras e as medidas de probabilidades associadas para os sistemas fuzzy probabilísticos?
- É possível propor um método de defuzzificação específico para tratar a incerteza produzida pela informação amostral proveniente dos consequentes probabilísticos?

Assim, este trabalho busca soluções alternativas para as duas questões levantadas anteriormente, conforme descrito na seção a seguir.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

O objetivo principal dessa dissertação é propor métodos inovadores em sistemas de inferência, unindo teorias bem conhecidas para lidar com incerteza, como probabilidade e sistemas *fuzzy*, de forma a permitir sua aplicação direta em sistemas *fuzzy* probabilísticos (SFP).

1.2.2 Objetivos Específicos

De forma mais específica, busca-se neste trabalho:

- Elaborar um algoritmo de geração automática de regras *fuzzy* inspirado no método de Wang Mendel (WANG; MENDEL, 1992) o qual, a partir de dados de treinamento, deverá gerar um conjunto de regras *fuzzy* probabilísticas, não descartando a informação inerente a dados redundantes ou conflitantes.
- Investigar a aplicação do método de geração de regras em SFPs através de dois tipos de problemas: classificação e previsão de séries temporais.
- Propor um método de defuzzificação bayesiana de forma a tratar informação amostral proveniente na saída de sistemas de inferência.
- Investigar a aplicação do método de defuzzificação bayesiana em SFPs através de um problema de previsão de séries temporais.
- Utilizar estudos de casos que permitam não só testar as novas técnicas propostas mas também compará-las com algumas técnicas usuais de obtenção automática de sistemas *fuzzy*.

1.3 ORGANIZAÇÃO DO TEXTO

Esta dissertação está organizada conforme segue: após este capítulo introdutório, o capítulo 2 estabelece uma revisão sucinta da literatura quanto a sistemas *fuzzy*, geração de base de regras, os métodos defuzzificadores existentes, teoria das probabilidades e inferência bayesiana. A seguir, o capítulo 3 descreve as propostas inerentes ao método para geração de regras *fuzzy* probabilísticas e a defuzzificação bayesiana em sistemas *fuzzy* probabilísticos. O capítulo 4 trata de dois experimentos utilizados para avaliação das novas propostas, enquanto que o capítulo 5 é reservado para discussão de resultados e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 SISTEMAS FUZZY

Os sistemas *fuzzy*, também conhecidos como sistemas nebulosos, sistemas de inferência nebulosa, sistemas *fuzzy* baseados em regras, ou simplesmente modelos *fuzzy*, representam uma das mais importantes ferramentas de modelagem baseada na teoria dos conjuntos *fuzzy*. Sistemas que utilizam este tipo de conjunto têm sido aplicados com sucesso em diversas áreas, como: controle automático, classificação e reconhecimento de padrões, tomada de decisão, sistemas inteligentes, previsão de séries temporais e robótica (JANG; SUN; MIZUTANI, 1997).

Sendo a base dos sistemas *fuzzy*, os conjuntos *fuzzy* vêm sendo amplamente utilizados e pesquisados desde Zadeh (1965), devido à sua flexibilidade em relação à pertinência de seus elementos: ao invés da classificação tradicional (um elemento pertence ou não pertence), os conjuntos *fuzzy* permitem a existência da pertinência parcial. Ao não tratarem de forma rígida as informações disponíveis, possibilitam considerar a ambiguidade presente no mundo real, dando margem à resolução de inúmeros problemas onde se faz necessária uma flexibilização nos limites dos conjuntos trabalhados (PEDRYCZ; GOMIDE, 2007).

Diferentemente de um conjunto clássico (ou conjunto *crisp*), que possui uma função característica (pertinência 0 ou 1), um conjunto *fuzzy* tradicional possui uma função associada, de forma a definir os diferentes graus de pertinência entre 0 e 1. Esta função, μ_A , é chamada de função de pertinência e pode ser definida da seguinte forma:

$$\mu_A : U \rightarrow [0, 1],$$

onde U é o universo do conjunto A . Os formatos mais usuais desta função são, por exemplo, a função triangular, a trapezoidal e a Gaussiana. Estes formatos, bem como alguns mais complexos, serão considerados nas próximas seções.

Ao se trabalhar com os conjuntos *fuzzy*, algumas definições básicas se fazem necessárias:

- Suporte de um conjunto *fuzzy*: conjunto dos elementos do universo para os quais o grau de pertinência é maior do que zero: $S_A = \{x | \mu_A(x) > 0\}$.
- Núcleo de um conjunto *fuzzy*: conjunto dos elementos do universo com grau de pertinência igual a 1: $N_A = \{x | \mu_A(x) = 1\}$.
- Altura de um conjunto *fuzzy*: valor máximo da função de pertinência: $H_A = \sup_x \{\mu_A(x)\}$.

O objetivo principal de conjuntos *fuzzy* é o de tratar a incerteza presente em informações que dizem respeito a conceitos ambíguos, como ‘alto’, ‘baixo’, ‘limpo’, ‘sujo’, etc. Estes conceitos vagos podem ser associados a variáveis linguísticas, ou seja, variáveis que assumem valores linguísticos ao invés de números. Estas variáveis são a base para o raciocínio aproximado (raciocínio *fuzzy*), o qual por sua vez pode ser formalizado por meio dos sistemas de inferência *fuzzy*. Estes sistemas trabalham com um conjunto de regras e podem inferir uma ou mais saídas (*fuzzy* ou não) a partir destas regras e de fatos apresentados como entrada. Assim, conforme discutido por Lin e Lee (1996), um sistema de inferência *fuzzy* é um mapeamento ou função de um espaço de alternativas de entrada para um espaço de saída. Segundo (JANG; SUN; MIZUTANI, 1997), a estrutura básica de um sistema de inferência *fuzzy* possui três componentes conceituais:

- uma **base de regras**, que contém o conjunto de regras *fuzzy*;
- uma **base de dados**, que define a quantidade, formato e localização das funções de pertinência associadas aos conjuntos *fuzzy* que deverão compôr as regras;
- e um **mecanismo de raciocínio**, que realiza, a partir de operadores pré-definidos, um procedimento de inferência (raciocínio *fuzzy*) para obter a saída ou conclusão, baseado nas regras e fatos conhecidos.

Há inúmeras maneiras de se classificarem os sistemas *fuzzy*. Neste trabalho optou-se por uma classificação fundamentada no tipo de função de pertinência utilizada: conjuntos *fuzzy* do tipo-1, tipo-2, não-estacionários e probabilísticos. Assim, as seções a seguir descrevem de maneira resumida os principais tipos de sistemas *fuzzy* encontrados em aplicações da área, tendo como base esta classificação.

2.1.1 Sistemas de Inferência *Fuzzy* do Tipo-1

A forma mais simples de se trabalhar com sistemas *fuzzy* é utilizando conjuntos do tipo-1, onde cada grau na função de pertinência assume apenas um valor numérico. Uma forma

frequentemente utilizada para representar a função de pertinência deste tipo de conjunto é, por exemplo, a função triangular. A figura 1 ilustra um exemplo que utiliza este tipo de função, juntamente com o tipo trapezoidal, para particionar o universo da variável linguística “altura”, utilizando três termos linguísticos: Baixo, Médio, Alto.

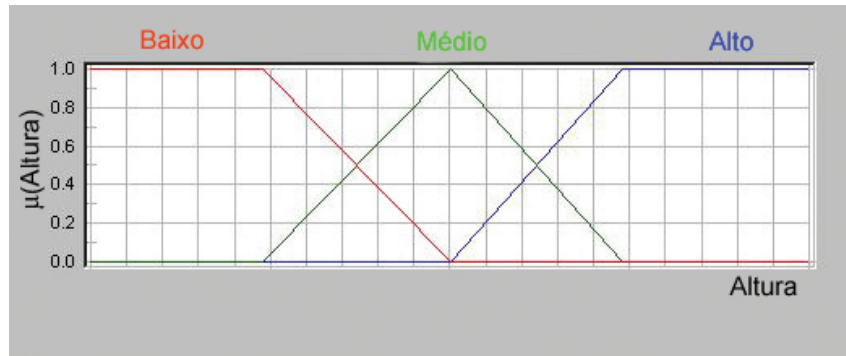


Figura 1: Exemplo de Função de Pertinência Triangular para variável Altura.

Neste trabalho, o formato mais utilizado de função de pertinência será a função Gaussiana:

$$\mu_A(x) = e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

onde μ é a média e σ é o desvio padrão. A figura 2 ilustra como ficaria o exemplo do uso desta função para realizar o mesmo particionamento.

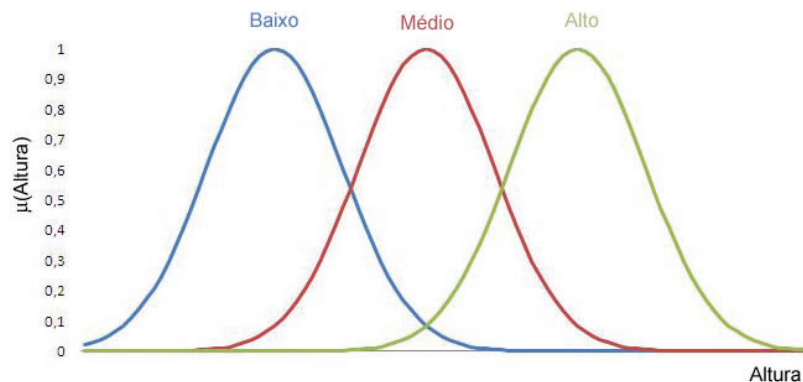


Figura 2: Exemplo de Função de Pertinência Gaussiana na partição do universo da variável Altura.

De forma sucinta, os sistemas do tipo-1 trabalham com a ideia de generalização da regra *Modus Ponens* da Lógica Clássica (DIAS, 2010), onde o processo de se inferir uma conclusão a partir de um fato e de uma regra é dado por:

Se X é A então Y é B . (Regra)

X é A . (Fato)

Logo, Y é B . (Conclusão)

Neste caso, X e Y são as variáveis e A e B são conjuntos clássicos (*crisp*). Assim, no contexto de Raciocínio *Fuzzy*, baseado no *Modus Ponens Generalizado* (MORSI; FAHM, 2002), tem-se:

Se X é A então Y é B .

X é A' .

Logo, Y é B' .

Neste novo exemplo, X e Y são as variáveis linguísticas e A , B , A' e B' são conjuntos *fuzzy*.

Usualmente, os sistemas baseados no raciocínio *fuzzy* (ou sistemas de inferência *fuzzy*) são compostos por regras com múltiplas variáveis de entrada, como por exemplo:

Se X_1 é A_1 E X_2 é A_2 E ... E X_v é A_v então Y é B ,

onde as variáveis X_i são as variáveis linguísticas do antecedente da regra e ' Y é B ' é chamado de consequente.

Valendo-se destes conceitos e da definição de operadores e operações específicas da inferência composicional, os Sistemas *Fuzzy* do tipo-1 trabalham de forma a combinar diferentes variáveis de entrada, cada qual associada às suas respectivas funções de pertinência, através de regras *fuzzy*, as quais produzirão um conjunto *fuzzy* do tipo-1 na saída que pode ser convertido em um valor numérico (*crisp*) através de algum método de defuzzificação.

No processo de inferência tradicionalmente usado em aplicações práticas - inferência escalonada, ou Min-Max (PEDRYCZ; GOMIDE, 2007) - o procedimento é simplificado e cada regra é ativada - com um grau de ativação GA - através da agregação dos valores obtidos no processo de fuzzificação (*matching* de cada entrada com o conjunto *fuzzy* associado) e produz uma saída inferida segundo uma semântica pré-definida. Por fim, as saídas inferidas de cada regra são combinadas por meio de um operador de agregação de regras, dando como resultado um conjunto *fuzzy*.

Antes do sistema entrar em operação, é preciso definir a base de dados, ou seja, as funções de pertinência de cada variável (tanto de entrada quanto de saída), a base de regras (que relaciona os diferentes termos linguísticos de cada variável) e os operadores que serão utilizados

em cada passo (semântica da regra, agregação de antecedentes e das regras, etc...). Isto posto, a operacionalização do sistema pode ser resumida conforme segue:

1. *Matching* (ou fuzzificação) em cada regra. Nesta fase, as entradas (representadas por conjuntos *fuzzy* ou valores *crisp*) são comparadas com os respectivos conjuntos *fuzzy* presentes nas regras. No caso de entradas *crisp*, os valores das variáveis de entrada são aplicados às respectivas funções de pertinência, produzindo um nível de *matching* para cada variável de entrada.
2. Agregação dos antecedentes. Em cada regra, têm-se os diferentes níveis de *matching* (resultado do item anterior), que são combinados segundo um operador de agregação pré-determinado (como o mínimo ou o produto), produzindo o Grau de Ativação (GA) da regra.
3. O resultado GA é combinado com o consequente de cada regra, segundo uma semântica pré-definida, gerando uma saída inferida.
4. As saídas inferidas são agregadas, por meio de um operador de agregação de regras, formando o conjunto *fuzzy* inferido, Y.
5. Se necessário, o conjunto Y é defuzzificado em um valor numérico (*crisp*), y.

O funcionamento de um sistema pode ser melhor visualizado na figura 3, que ilustra um exemplo com duas variáveis de entrada (as primeiras duas colunas), cada uma com o universo particionado em três conjuntos *fuzzy*, uma variável de saída (terceira coluna), com o universo particionado em 5 conjuntos *fuzzy* e 9 regras (as linhas) que definem as relações entre elas. Neste caso, foi utilizado o mínimo para agregação das entradas e semântica das regras, cada uma produzindo uma saída inferida diferente, que na sequência foram agregadas pelo máximo.

Como foi dito, o último passo no processo de inferência *fuzzy* é a defuzzificação. Os métodos mais usuais para se efetuar tal operação estão descritos na seção 2.3. Referências adicionais sobre conjuntos, lógica e sistemas *fuzzy* podem ser encontradas, por exemplo, em Gomide e Gudwin (1997) e em Pedrycz e Gomide (2007).

2.1.2 Sistemas de Inferência *Fuzzy* do Tipo-2

O próprio Zadeh (1975) propôs o conceito de “Conjuntos *Fuzzy* do Tipo-2”, a fim de tratar a incerteza presente nas opiniões dos especialistas de cada área do conhecimento. Ora, se um especialista propõe uma dada função de pertinência de uma forma e um outro especialista,

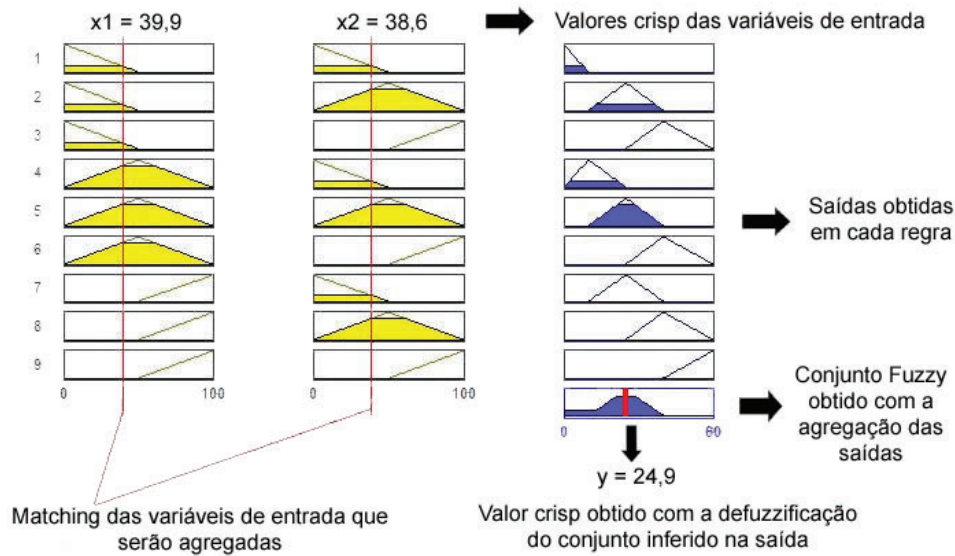


Figura 3: Exemplo de Funcionamento de um Sistema Fuzzy Específico.

ainda que da mesma área, propõe a função de outra forma, ainda que ligeiramente diferente, tem-se como resultado uma incerteza nos graus de pertinência propostos, seja essa incerteza alta ou baixa. Desta forma, conjuntos do tipo-2 são aqueles cujos graus de pertinência são, eles próprios, conjuntos *fuzzy*. Um Conjunto *fuzzy* do tipo-2, \tilde{A} , é formalmente definido como:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x, u) / (x, u),$$

onde $J_x \subseteq [0, 1]$ é o conjunto dos graus de pertinência primária de x , e $\mu_{\tilde{A}}(x, u)$ é a função de pertinência do tipo-2 e a representação em forma de integral denota a união sobre todos os x e u .

Pode-se perceber, portanto, que diferente dos conjuntos tipo-1, onde há apenas um grau de pertinência para cada possível valor que x pode assumir, nos conjuntos de tipo-2 tem-se uma função de pertinência secundária associada a cada valor de x , representando a incerteza presente na forma da função de pertinência associada ao conjunto X .

Os conjuntos do tipo-2 podem ser classificados de duas formas:

- Gerais: seguem a definição formal sem qualquer restrição.
- Intervalares: admite-se que as funções de pertinência secundárias são constantes e iguais a 1.

Os conjuntos intervalares são mais fáceis de se trabalhar devido a uma maior simplicidade dos cálculos e têm sido o maior foco de estudo por conta disso. Não se pode ignorar, porém,

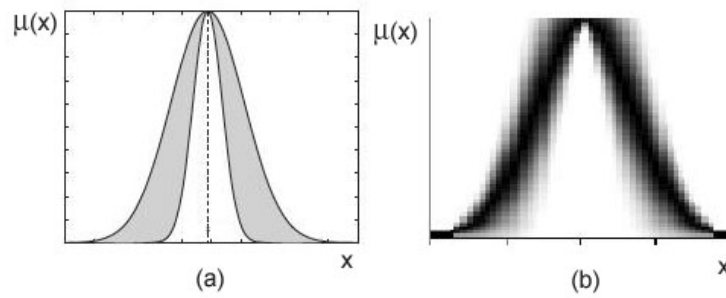


Figura 4: Exemplo de FOU (a) para um conjunto intervalar e um GEFOU (b) para um conjunto geral.

a importância dos conjuntos do tipo geral, que nos podem trazer uma melhor representação da realidade.

Uma vez que os conjuntos do tipo-2 são de difícil visualização, o que usualmente se faz é uma projeção do conjunto no plano, denominada de FOU (Footprint of Uncertainty), bastante útil em conjuntos intervalares, onde há limites mais bem definidos devido à restrição de que os graus secundários são constantes e iguais a 1.

Em conjuntos do tipo geral, entretanto, faz-se necessária uma melhor representação no plano, utilizando tons de cinza para identificar a “força” dos graus de pertinência. Assim surge o GEFOU (*General Footprint of Uncertainty*), proposto por Lucas (2009). A diferença entre a FOU e a GEFOU está melhor explicitada na figura 4, onde percebem-se limites mais bem definidos no conjunto intervalar, enquanto que o conjunto geral apresenta limites mais nebulosos.

Os princípios básicos do SIF (Sistema de Inferência *Fuzzy*) continuam os mesmos. Entretanto, no SIF do tipo-2, as operações serão mais complexas devido à natureza tridimensional dos conjuntos do tipo-2 (KARNIK; MENDEL; LIANG, 1999). Por exemplo, se antes a saída inferida era um conjunto do tipo-1, agora cada regra irá inferir um conjunto do tipo-2.

Assim como no SIF do tipo-1, existem inúmeras maneiras de tratar a saída inferida. Alguns destes métodos serão vistos na seção 2.4. Referências adicionais sobre Sistemas *Fuzzy* do Tipo-2 podem ser encontradas em Lucas (2009).

2.1.3 Sistemas de Inferência *Fuzzy* Não-Estacionários

Conforme já mencionado, os conjuntos do tipo-2 foram introduzidos a fim de modelar a incerteza presente nas opiniões de especialistas na construção das funções de pertinência de um problema envolvendo conjuntos *fuzzy*. Porém, pode ser introduzida ainda variabilidade nas funções de pertinência propostas pelos especialistas (GARIBALDI; OZEN, 2004).

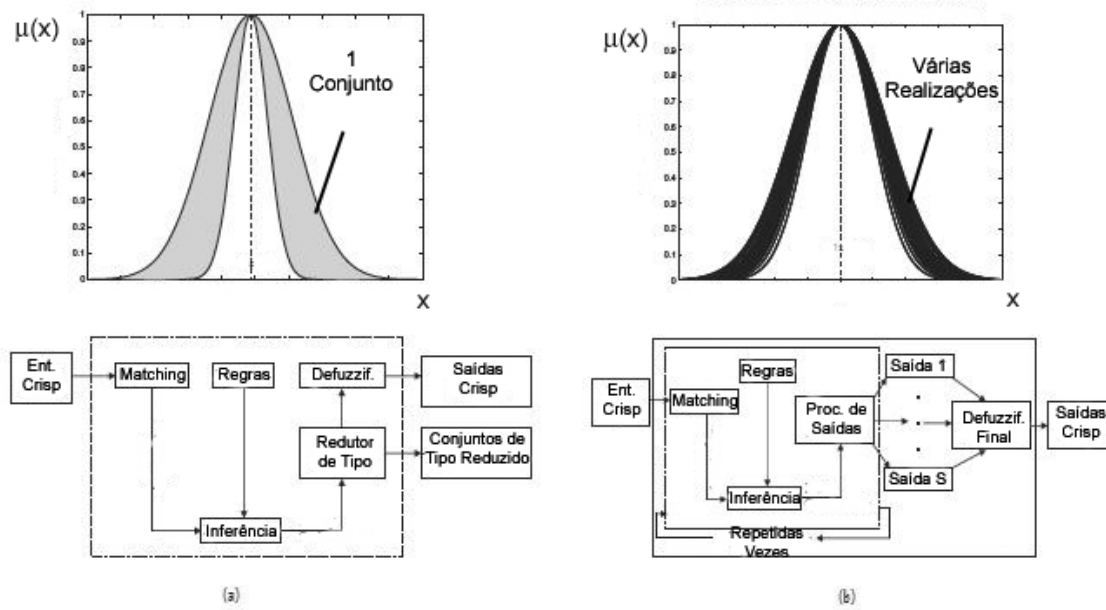


Figura 5: Representação pictórica e esquema do SIF para conjuntos do tipo-2 (a) e conjuntos não-estacionários (b)

Por conta disso, Garibaldi, Musikasuwan e Ozen (2005) propuseram o conceito de Conjuntos *Fuzzy* Não-Estacionários. Neste tipo de conjunto, a função de pertinência é combinada com uma função que representa a variabilidade do sistema. Desta forma, o sistema gera S saídas diferentes, que são combinadas em um defuzzificador final para gerar a saída *crisp*. A figura 5 faz um paralelo entre sistemas do tipo-2 e sistemas não-estacionários. Percebe-se que a principal diferença está justamente nas saídas inferidas para cada realização temporal do sistema não-estacionário, enquanto que no sistema do tipo-2, o processo é semelhante ao sistema do tipo-1, com a inclusão do passo adicional da redução de tipo, operação que consiste basicamente em se converter um conjunto do tipo-1 em um conjunto do tipo-2, que será vista em detalhes na seção 2.4.

Formalmente, um conjunto *fuzzy* não-estacionário pode ser definido como

$$\ddot{A} = \int_{x \in X} \mu_{\ddot{A}}(x, t) . x,$$

onde $x \in X$, a função de pertinência $\mu_{\ddot{A}}(x, t) \in [0, 1]$ e t é uma variável livre indicando tempo, isto é, o tempo em que o conjunto *fuzzy* apresenta uma realização.

Pode-se pensar em três formas principais de variabilidade, a saber:

- Na localização:

$$\mu_{\ddot{A}}(x, t) = e^{-\frac{(x-m(t))^2}{s^2}}$$

- Na escala:

$$\mu_{\tilde{A}}(x, t) = e^{-\frac{(x-m)^2}{s(t)^2}}$$

- Ruído branco:

$$\mu_{\tilde{A}}(x, t) = e^{-\frac{(x-m)^2}{s^2}} \pm \varepsilon(t)$$

Nos três casos, é possível modelar a variabilidade conforme segue:

$$m(t) = m + kf(t),$$

$$s(t) = s + kf(t),$$

$$\varepsilon(t) = kf(t),$$

onde m e s são a média e o desvio padrão do conjunto *fuzzy*, k é uma constante e $f(t)$ é uma função perturbadora. Tal função é o que representa a variabilidade no sistema, gerando pequenas mudanças na função de pertinência do conjunto. Desta forma, o parâmetro da função de pertinência é, verdadeiramente, uma variável aleatória (GARIBALDI; MUSIKASUWAN; OZEN, 2005).

A princípio, qualquer função temporal poderia ser utilizada como uma função perturbadora, mas devido a algumas restrições práticas, uma vez que a definição de tempo pode ser arbitrária e relativa, o número de escolhas acaba se reduzindo a apenas três: função seno, função pseudo-aleatória e série temporal diferenciada. Para detalhes adicionais destas funções, pode-se consultar Garibaldi, Musikasuwan e Ozen (2005).

Estes mesmos autores produziram um experimento para comparar o desempenho de conjuntos não estacionários com conjuntos do tipo-2 intervalares, conseguindo resultados interessantes e chegando a concluir que o sistema não-estacionário está emulando a inferência produzida por um sistema do tipo-2 intervalar.

2.1.4 Sistemas de Inferência *Fuzzy* Probabilísticos

Na natureza, existem vários tipos de incerteza, devido aos mais variados motivos. Dois tipos bastante presentes são: a incerteza estatística (ou por aleatoriedade) e a incerteza não-estatística (ou por imprecisão/ambiguidade). Esta última trata da incerteza presente em variáveis linguísticas e tal imprecisão é tratada através da teoria *fuzzy*. A incerteza estatística trata da aleatoriedade que surge devido à chance de ocorrência de um evento futuro. Este tipo de incerteza é tratada por meio da teoria das probabilidades.

No entanto, em algumas situações, tem-se presente os dois tipos de incerteza: é necessário saber a probabilidade de ocorrência de um evento, mas o próprio evento é um elemento “difuso” (*fuzzy*). Neste tipo de situação, é preciso procurar uma forma conjunta de tratar os dois tipos de incerteza.

Muito já se discutiu a respeito da utilização conjunta de Sistemas *Fuzzy* e Teoria das Probabilidades, conforme já discutido no Capítulo 1. De forma mais detalhada, pode-se citar, por exemplo, Coletta e Scozzafava (2006), que contextualizam um exemplo deste uso conjunto do seguinte modo: suponha que uma caixa tenha bolas de diferentes tamanhos e retira-se uma bola dessa caixa. Deseja-se calcular a probabilidade do evento G = retirar uma bola grande. Logo percebe-se que o evento G é um evento *fuzzy*, mas conhecendo-se as frações f de bolas de cada tamanho na caixa, a probabilidade será dada por

$$P(G) = \sum_i P(G|t_i)P(t_i) \approx \sum_i \mu(t_i)f_i,$$

sendo $\mu(t_i)$ a função *fuzzy* para o evento G , e f_i é a frequência do tamanho da bola i .

Conforme discutido no capítulo 1 é possível se pensar em várias situações neste exemplo com diferentes níveis de incerteza e, ainda assim, o problema poderia ser tratado com uso conjunto de uma função *fuzzy* e pelo Teorema de Bayes.

Percebe-se, porém, que essa abordagem é diferente do bem estabelecido conceito de probabilidade *fuzzy* (PEDRYCS, 1998). Para se entender a diferença, considere as seguintes sentenças:

1. É bastante provável que amanhã choverá pesadamente.
2. Amanhã choverá pesadamente com probabilidade de 90%.

Na sentença 1, tem-se um conceito *fuzzy*, “pesadamente” e a própria probabilidade também

aparece de forma *fuzzy*: “bastante provável”. Já na sentença 2, continua-se com um conceito *fuzzy*, porém, sua probabilidade de ocorrência é conhecida. É neste segundo tipo de situação que Meghdadi e Akbarzadeh-T (2001) estabeleceram o conceito de Lógica *Fuzzy* Probabilística e, conseqüentemente, de Sistemas *Fuzzy* Probabilísticos (SFP).

A lógica *fuzzy* tradicional trabalha com “graus de verdade”, ou seja, uma sentença pode ser pouco verdadeira, muito verdadeira, falsa, muito falsa, pouco falsa, etc. Já na teoria de probabilidades, prevalece a ideia de “probabilidade de verdade”. A lógica *fuzzy* probabilística é, portanto, uma extensão da tradicional, trabalhando não apenas com os graus de verdade, mas também com as probabilidades de cada grau.

Desta forma, tem-se o conceito de conjuntos *fuzzy* probabilísticos, onde o grau de pertinência de um conjunto *fuzzy* é incerto e pode ser representado por meio de uma variável aleatória. Assim, cada grau está associado à respectiva distribuição de probabilidades. Pode-se ter, por exemplo, uma função de pertinência triangular, onde cada grau desta função tem distribuição normal, com médias e variâncias diferentes, conforme ilustrado na figura 6, adaptada de Meghdadi e Akbarzadeh-T (2001).

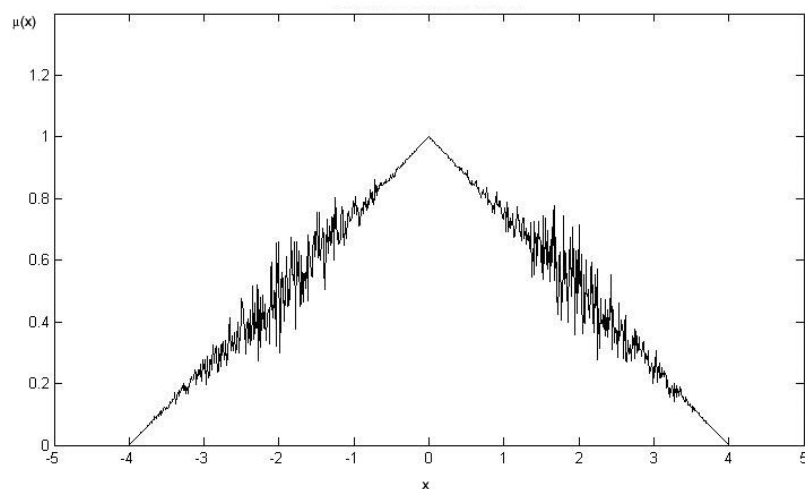


Figura 6: Exemplo de um Conjunto *Fuzzy* Probabilístico

Como uma extensão da lógica clássica, que trabalha com um sistema de inferência baseado na regra *modus ponens*, já foi visto que a lógica *fuzzy* utiliza uma regra *modus ponens* generalizada, sendo que em um sistema de duas entradas e uma saída, uma das regras poderia ter, por exemplo, a seguinte forma:

- Se X_1 é A E X_2 é B então Y é F . (Onde X_1 e X_2 são variáveis linguísticas do antecedentes e Y do conseqüente e A , B e F são conjuntos *fuzzy*.)

Em um SFP, contudo, devido à incerteza por aleatoriedade, este sistema de inferência será alterado para:

- Se X_1 é A E X_2 é B então Y é C com probabilidade P_C , é D com probabilidade P_D , ..., é Z com probabilidade P_Z .

De forma a atender os axiomas, todas as probabilidades do consequente devem somar 1.

Utilizando tal base de regras, pode-se alterar um sistema Mamdani, por exemplo (MEGH-DADI; AKBARZADEH-T, 2001) para se fazer a inferência necessária, conforme visualizado na figura 7. Neste exemplo, há duas variáveis de entrada, A e B , com três conjuntos *fuzzy* na partição de seus universos, e uma variável de saída, F , com cinco conjuntos *fuzzy* na partição.

Assim sendo, em cada regra, tem-se um mecanismo de “roleta”, que seleciona um determinado conjunto de saída, baseado em sua probabilidade de ocorrência. Desta forma, para cada regra, a “roleta gira”, amostrando uma partição do conjunto de saída. A partir daí, o sistema de inferência é o mesmo: as saídas inferidas em cada regra são agregadas por meio de um operador de agregação e o conjunto *fuzzy* final pode ser defuzzificado, no caso de ser necessária uma saída *crisp*.

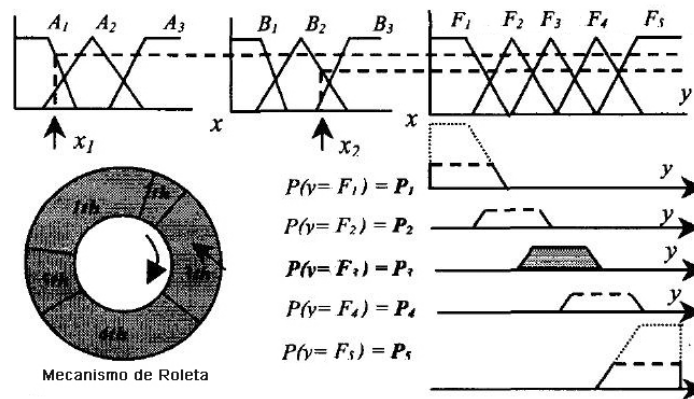


Figura 7: Sistema de Inferência Fuzzy Probabilístico

Pode-se perceber que, neste processo, a mesma entrada do sistema pode produzir várias saídas diferentes. Assim, há uma incerteza estatística na saída do sistema. A fim de se lidar com essa incerteza, Liu e Li (2005) sugerem um processo iterativo que gere várias saídas diferentes e que depois serão agregadas, calculando-se a esperança matemática destas saídas defuzzificadas. Tal operação é relativamente simples, bastando calcular uma média dos centróides das saídas, ponderando-os por suas respectivas probabilidades de ocorrência.

Outra opção seria a de gerar apenas uma iteração e utilizar a saída gerada apenas uma vez, correndo-se o risco da saída gerada ser aquela com a menor probabilidade de ocorrência. Dependendo da aplicação e da necessidade do usuário, pode-se, ainda, descartar as demais regras e manter apenas aquela com maior probabilidade. Nesta dissertação, contudo, deseja-se apresentar a opção de forma mais generalizada e o SFP proposto trabalhará com várias iterações, de forma a tratar conjuntamente informações quantitativas e qualitativas contidas em um conjunto de dados de teste, conforme descrito no Capítulo 3.

Para referências adicionais sobre SFP, sugere-se Meghdadi e Akbarzadeh-T (2001) e Liu e Li (2005).

2.2 GERAÇÃO DE BASE DE REGRAS

Em um Sistema de Inferência *Fuzzy* (SIF), foi visto que é necessário que haja uma base de regras, que pode ser obtida através do conhecimento de um especialista, ou de algoritmos de geração automática a partir de um conjunto de dados de treinamento.

Existem diversas abordagens para geração automática, que de maneira geral, podem ser classificadas como abordagens evolutivas ou não-evolutivas. As primeiras não serão o foco desta dissertação, porém, um aprofundamento sobre estas técnicas é dado, por exemplo, por Herrera (2008).

Dentre os algoritmos não-evolutivos mais conhecidos para geração de base de regras estão o algoritmo de Wang-Mendel (WANG; MENDEL, 1992) - ou simplesmente WM - e o algoritmo de Ishibuchi (ISHIBUCHI; NAKASHIMA, 2001), mais específico para o caso de classificação. Há várias semelhanças entre eles, sendo que uma das principais diferenças é que o método Ishibuchi utiliza graus de confiança para cada regra medidos pela discrepância entre os dados utilizados para gerar a regra. Como são os métodos utilizados nas comparações apresentadas no Capítulo 4, ambos são vistos em detalhes nas próximas seções.

2.2.1 Algoritmo de Wang-Mendel

O método de Wang Mendel (WANG; MENDEL, 1992) é utilizado tanto para a geração de regras de SIFs de uso geral como em classificadores *fuzzy*, que são sistemas *fuzzy* com objetivo de classificar pontos em classes pré-selecionadas, onde a saída do sistema é meramente uma representação numérica desta classe. Neste último caso, diversos pontos estão associados a diferentes classes e deseja-se construir uma base de regras através destes pontos. O algoritmo WM pode ser resumido nos seguintes passos:

1. Defina o número de termos linguísticos e particione o universo de todas as variáveis de entrada.
2. Crie uma regra *fuzzy* para cada elemento do conjunto de pontos de treinamento. Para cada variável de entrada, selecione a função de pertinência de maior grau.
3. Calcule o grau de ativação de todas as regras, utilizando um operador apropriado.
4. Elimine as redundâncias e inconsistências: apenas as regras com maior grau de ativação permanecem.

Considerando-se 3 pontos de treinamento, (figuras 8 e 9, na seção 3.1), p , k e m , do tipo x_1, x_2, classe , tem-se como exemplo de possíveis regras resultantes:

- Regra p : Se x_1 é A_1 e x_2 é A_2 , então classe é 1.
- Regra k : Se x_1 é A_1 e x_2 é A_2 , então classe é 1.
- Regra m : Se x_1 é A_1 e x_2 é A_2 , então classe é 2.

Neste exemplo, as regras geradas pelos pontos p e k são redundantes e ambas são inconsistentes com a regra gerada pelo ponto m . Neste caso, segundo o algoritmo de WM, ficaria na base apenas a regra que possuir o maior grau de ativação.

2.2.2 Algoritmo de Ishibuchi

A grande diferença entre este método e o anterior é que ele atribui graus de confiança para cada regra, baseados no maior grau de ativação das mesmas. Assim, Ishibuchi e Nakashima (2001) propuseram o seguinte algoritmo:

1. Defina o número de termos linguísticos e particione o universo de todas as variáveis de entrada.
2. Crie uma regra *fuzzy* para cada elemento do conjunto de pontos de treinamento. Para cada variável de entrada, selecione a função de pertinência de maior grau.
3. Calcule o grau de ativação de todas as regras, utilizando um operador apropriado.
4. Calcule $\beta_{\text{Class}} = \text{soma dos graus de ativação para cada classe com regras de mesmo antecedente}$.

5. Elimine as regras inconsistentes, selecionado aquela que possuir o maior β_{Class} .
6. Calcule $\bar{\beta} = (\text{Soma dos graus de ativação que não são da classe escolhida}) / (\text{N}^\circ \text{ de Classes} - 1)$
7. Para as regras remanescentes, calcule o nível de confiança para cada uma delas, dado por $(\beta_{Class} \text{ da classe} - \bar{\beta}) / (\text{Soma de todos os } \beta_{Class} \text{ com mesmos antecedentes})$

No exemplo dos três pontos, mostrado na seção anterior, o resultado por este método seria o seguinte:

- Passo 4: $\beta_{Class=1} = GA_p + GA_k$ e $\beta_{Class=2} = GA_m$.
- Passo 5: Selecione a regra m supondo $\beta_{Class=2} > \beta_{Class=1}$.
- Passo 6: $\bar{\beta} = \beta_{Class=1} / (2 - 1) = (GA_p + GA_k) / 1$.
- Passo 7 - Nível de confiança da regra m: $(\beta_{Class=2} - \beta_{Class=1}) / (\beta_{Class=2} + \beta_{Class=1}) = (GA_m - GA_p - GA_k) / (GA_p + GA_k + GA_m)$.

Como se tem, nesse caso, um sistema de inferência que possui um grau de confiança para cada regra em sua base de regras, a inferência se dará simplesmente multiplicando o grau de ativação da regra ativada pelo seu respectivo nível de confiança e procedendo-se normalmente na sequência do processo.

Percebe-se que tanto o algoritmo WM quanto o de Ishibuchi podem descartar uma grande quantidade de informação, se houver um número elevado de regras inconsistentes. Assim, a proposta deste trabalho é apresentar uma abordagem alternativa que considere as inconsistências.

2.2.3 Outras Abordagens

Na literatura, há ainda diversos outros métodos propostos para geração automática de regras, sendo que esse assunto, bem como a fase de defuzzificação, tornou-se uma abrangente área de pesquisa.

Como exemplo de métodos para geração automática de regras pode-se citar, por exemplo, Tuma (2009), que propõe uma adaptação do método WM, específica para classificadores *fuzzy*. O que se faz é particionar o espaço dos dados de treinamento, construindo-se hiperplanos através do centróide do conjunto de exemplos. Desta forma, a cada hiperplano será associada uma regra para o classificador.

Há ainda Rojas et al. (2000), cuja proposta é a de um Sistema *Fuzzy* auto-organizado, ou seja, um algoritmo que começa com um sistema simples, mas que constrói automaticamente as funções de pertinência, as regras *fuzzy* e ajusta as funções de acordo com o que se pressupõe ser um sistema adequado.

Já Johanyák e Kovács (2007) introduzem dois métodos de geração de regras, o RBE-SI e o RBE-DSS (sendo que RBE significa *Rule Base Extension*- Extensão da Base de Regras). Ambos utilizam praticamente o mesmo algoritmo, diferindo apenas na maneira em que cada um define as formas dos conjuntos *fuzzy* (SI - *Set Interpolation* e DSS - *Default Set Shapes*). Basicamente, o que se faz é utilizar pontos de treinamento para gerar as duas regras iniciais e calcula-se o Erro Quadrático Médio para o sistema, utilizando apenas estas duas regras. Caso o erro seja alto, novas regras são geradas e assim sucessivamente, até que seja encontrado um erro aceitável.

Ivancic, Malaviya e Peters (1999), por outro lado, sugerem a utilização do método de agrupamento das C-Médias para aglomerar as funções de pertinência e, desta forma, tentar reduzi-las, tornando o sistema mais simples.

Outra ideia nessa mesma linha é a de utilizar medidas de similaridade, também de forma a reduzir o número de funções de pertinência, como proposto por Setnes et al. (1998). Chen e Linkens (2004) vão ainda além, utilizando esta ideia não apenas para redução, mas desenvolvendo ainda um algoritmo para simplificação do sistema como um todo.

Esses são apenas alguns exemplos dos métodos mais recentes que se valem da tentativa de procurar uma maneira simples e eficiente de geração de regras, de forma a considerar toda a informação constante nos dados de treinamento. Percebe-se, portanto, uma preocupação crescente com o desenvolvimento de um método simples que, utilizando toda a informação possível, consiga ser representativo o suficiente para uma boa eficiência do sistema *fuzzy*.

2.3 MÉTODOS DEFUZZIFICADORES EM SISTEMAS DO TIPO-1

A seguir serão detalhados alguns dos métodos mais usuais de defuzzificação em sistemas *fuzzy* do Tipo-1, bem como outros propostos na literatura. Detalhes sobre os métodos mais comuns, como MeOM, FOM, MOM, LOM, FM, RCOM, COA, COS e HM podem ser encontrados, por exemplo, em Leekwijck e Kerre (1999). Referências adicionais são mencionadas para cada método específico. As seguintes notações são utilizadas nesta seção, especificamente:

- D: valor defuzzificado.

- Y : conjunto de saída inferido.
- N : total de pontos considerados para cálculo (específico para cada método).
- y_i : valores do suporte do conjunto Y .
- $\mu(y_i)$: graus de ativação dos valores y_i .
- c_i : valores do suporte de Y para os quais o grau de ativação é máximo.

2.3.1 Médias e Máximos

Entre os métodos mais intuitivos existentes, existem vários que envolvem cálculo de média ou outras medidas de posição e tendência central. A seguir, serão descritas as principais.

- **MeOM** (*Mean of Maximum*) e **MeOS** (*Mean of Support*) : um dos métodos mais elementares existentes, consiste em simplesmente calcular a média dos valores máximos do conjunto de saída, ou seja,

$$D = \frac{\sum_{i=1}^N c_i}{N}$$

N , neste caso, refere-se ao número de observações no núcleo de Y e em funções contínuas, o somatório será substituído por uma integral. Ao invés de considerar o núcleo de Y , pode-se considerar o suporte do mesmo, ou seja, todos os valores com grau de pertinência maiores do que zero. Desta forma, tem-se o método conhecido como MeOS.

- **FOM** (*First of Maximum*), **MOM** (*Middle of Maximum*) e **LOM** (*Last of Maximum*): ao invés de ser calculada a média dos máximos, estes métodos calculam simplesmente algumas medidas de posição. O valor inferido através do FOM nada mais é do que o mínimo, ou seja:

$$D = \inf(y_i \in Y | \mu(y_i) = h(Y)) = \inf(c_i)$$

onde $h(Y)$ é a altura (maior valor alcançado) do conjunto. O MOM calcula o ponto médio:

$$D = \frac{LI + LS}{2}$$

onde LI e LS são os limites inferior e superior dos valores com pertinência máxima de Y , respectivamente. E o LOM retorna o máximo destes mesmos valores:

$$D = \max(y_i \in Y | \mu(y_i) = h(Y)) = \max(c_i)$$

Métodos bastante utilizados e facilmente encontrados na literatura.

- FM ou COM (*Fuzzy Mean* ou *Center of Mean*) e WFM (*Weighted Fuzzy Mean*): o FM é uma otimização do método COA, que é visto mais adiante. O que ele faz é calcular a média dos conjuntos inferidos na saída, sendo que o método WFM é uma generalização do mesmo. Detalhes adicionais sobre estes dois métodos, frequentemente utilizados em controladores *fuzzy* e filtros de imagem, podem ser encontrados em Jager, Verbruggen e Brujin (1992).
- QM (*Quality Method*) e EQM (*Extended Quality Method*): um caso particular do método WFM ocorre quando o peso escolhido é

$$w_i = 1/d_i$$

onde d_i é a cardinalidade do suporte de Y, ou seja, total de elementos de Y para os quais o grau de pertinência é maior do que zero (assumindo que o conjunto é discreto ou foi discretizado).

Uma extensão do QM é obtida quando se usa $w_i = 1/d_i^\xi$, sendo $\xi > 0$. Ambos os métodos foram propostos por Hellendoorn e Thomas (1993) .

- HWSM (*Height Weighted Second Maximum*): o segundo máximo de cada função de pertinência de saída é avaliado, e o valor final é calculado como uma média ponderada destes valores, onde os pesos são os graus das funções de pertinência aos quais eles pertencem. Para detalhes, veja Shi e Sen (2000).
- RCOM (*Random Choice of Maximum*): como o próprio nome sugere, este método consiste em uma escolha aleatória do valor de saída, novamente considerando apenas o centro de Y.

2.3.2 Centro de Gravidade

Outros métodos muito comuns em Sistemas Difusos são aqueles baseados no centróide (ou centro de gravidade) do(s) conjunto(s) de saída. A seguir são descritos os principais:

- COA (*Center of Area*): o valor inferido é o centróide do conjunto Y. Este cálculo é feito da seguinte forma:

$$D = \frac{\sum_{i=1}^N y_i \cdot \mu(y_i)}{\sum_{i=1}^N \mu(y_i)},$$

onde N é o total de pontos do suporte de Y. Em casos contínuos, o somatório torna-se uma integral. O ponto inferido, neste caso, divide a área do conjunto em duas partes iguais.

- COS (*Center of Sums*): COA é um método preferido a este, devido ao seu custo computacional (MADAU; FELDKAMP, 1996). Este método considera a área de cada conjunto individualmente, ao invés de calcular a união entre todos eles. Assim sendo,

$$D = \frac{\sum_{i=1}^N y_i \cdot \sum_{j=1}^M \mu(y_j)}{\sum_{i=1}^N \sum_{j=1}^M \mu(y_j)},$$

onde N é o total de pontos do conjunto Y e M é o total de conjuntos inferidos pelas regras, antes da agregação das mesmas. É comum, na literatura, encontrar o método COG (*Center of Gravity*), referindo-se tanto ao COA quanto ao COS.

- ECOA (*Extended Center of Area*): o método COA também pode ser generalizado atribuindo-se um expoente $\gamma \in]0, +\infty[$ aos valores $\mu(y_j)$. Este expoente é interpretado por um grau de confiança e com isso, tem-se o método ECOA, proposto por Runkler e Glesner (1993b).
- COLA (*Center of Largest Area*): utilizado quando o conjunto *fuzzy* inferido é não-convexo. O procedimento consiste em determinar o subconjunto com a maior área e então, calcular o centro de gravidade do mesmo. Para detalhes, veja Kovacic e Bogdan (2006).
- ICOG (*Indexed Center of Gravity*) e MICOG (*Modified Indexed Center of Gravity*): quando os valores da função de pertinência do conjunto inferido são muito baixos, ou iguais em todo Y (parte não-informativa da função de pertinência), utiliza-se um α -corte no conjunto, e a fórmula do COA é calculada considerando-se apenas o conjunto que está acima deste α -corte estipulado. Uma versão modificada deste método (MICOG) e detalhes adicionais sobre o mesmo podem ser encontrados em Czogala e Leski (2000).

- HM (*Height Method*): neste método, o centróide de cada saída inferida é primeiramente avaliado, e a saída final é calculada como a média das saídas individuais. Novamente, pode-se utilizar os graus da função de pertinência como pesos para o cálculo desta média.

Evidentemente, existem outras variações destes métodos, implementados de acordo com a necessidade que o problema exige.

2.3.3 Métodos Generalizadores

Generalizações dos métodos anteriores foram propostas e logo ganharam destaque devido às tentativas de procurar melhorar o processo de defuzzificação. São eles: BADD (YAGER; FILEV, 1991), GLSD (YAGER; FILEV, 1993a) e SLIDE (YAGER; FILEV, 1993b). A seguir são fornecidos alguns detalhes sobre cada um deles e também de outros dois.

- BADD (*Basic Defuzzification Distributions*): generalização do COA, onde o fator de confiança γ é introduzido. Desta forma, tem-se:

$$D = \frac{\sum_{i=1}^N y_i \cdot \mu(y_i)^\gamma}{\sum_{i=1}^N \mu(y_i)^\gamma}$$

A diferença entre este método e o ECOA é que, aqui, $\gamma \in [0, +\infty[$. Interessante notar que, quando $\gamma = 0$, BADD é reduzido ao MeOS, e quando $\gamma = 1$, é reduzido ao COA.

- GLSD (*Generalized Level Set Defuzzification*): outro método baseado nos α -cortes de Y. Neste caso, tem-se:

$$D = \frac{\sum_{i=1}^N a_i \cdot m_i \cdot \gamma^i}{\sum_{i=1}^N a_i \cdot \gamma^i}$$

onde N é o número de α -cortes (determinado a partir de um passo de amostragem dos valores da função de pertinência de Y), a_i é o número de elementos no i-ésimo α -corte e m_i é a média do i-ésimo α -corte.

Os métodos MeOS, MeOM e COA podem ser entendidos como casos particulares deste método.

- **SLIDE** (*Semi Linear Defuzzification*): uma extensão dos outros dois métodos é obtida quando se tem, além do fator de confiança γ , também um fator de rejeição β , na fórmula:

$$D = \frac{(1 - \beta) \cdot \sum_{i=1}^N y_i \cdot \mu(y_i) + \sum_{i=1}^M y_i \cdot \mu(y_i)}{(1 - \beta) \cdot \sum_{i=1}^N \mu(y_i) + \sum_{i=1}^M \mu(y_i)}$$

onde N é o número de elementos menores do que γ , M é o número de elementos maiores ou iguais a γ . Os métodos COG, ICOG e MeOM podem ser obtidos como casos particulares deste método.

- **DFR** (*Direct Fuzzy Reasoning*) e **RDFR** (*Restricted Direct Fuzzy Reasoning*): estes métodos utilizam a variável agregada na entrada do sistema, da seguinte forma:

$$D = \frac{\sum_{i=1}^N x_i \cdot s_i \cdot w_i}{\sum_{i=1}^N c_i \cdot s_i}$$

onde x_i é a variável agregada na i-ésima regra, s_i é o centróide do conjunto inferido nesta regra e w_i é o peso do mesmo. Quando os pesos são iguais, tem-se RDFR, que se dá como:

$$D = \frac{\sum_{i=1}^N x_i \cdot w_i}{\sum_{i=1}^N c_i}$$

Formas genéricas destes defuzzificadores também podem ser consideradas adicionando-se o fator de confiança como um expoente de x_i .

Alguns destes métodos, como o BADD, são deduzidos a partir da suposição de que o conjunto Y pode ser convertido em uma distribuição de probabilidade e então, o valor defuzzificado D é obtido calculando-se a esperança matemática desta distribuição. Esse método estendido é detalhado na próxima seção.

2.3.4 Conversão em Distribuição de Probabilidades

Esse método consiste em duas etapas. No primeiro passo, é aplicada uma transformação no conjunto Y, baseada na confiança que se tem no mesmo. Yager (1996) sugere algumas:

- TBADD (*Transformation for Basic Defuzzification Distributions*): $z_i = y_i^\alpha$. Dependendo da escolha de α , essa transformação pode levar à concentração ou dilatação do conjunto.

- TSLIDE (*Transformation for Semi-Linear Defuzzification*):

$$z_i = y_i, \text{ se } y_i \geq \alpha.$$

$$z_i = (1 - \beta) \cdot y_i, \text{ se } y_i < \alpha.$$

Pode-se trocar β por $e^{-\beta}$, no caso de ser desejável uma transformação não-linear, chamada então de TNSLIDE.

- Transformações Drásticas: não preservam a ordem original dos valores do conjunto Y.

O segundo passo consiste em uma nova transformação, na qual o conjunto Z obtido no passo anterior é então convertido em uma distribuição de probabilidades P. Dois métodos são sugeridos:

- Normalização: consiste em simplesmente dividir cada valor pelo somatório de valores possíveis, ou seja:

$$p_i = \frac{z_i}{\sum_{i=1}^N z_i}$$

- Método de Dubois e Prade: um método alternativo que toma a média das diferenças dos valores, ou seja:

$$p_i = \sum_{i=1}^N \frac{z_i - z_{i+1}}{i}$$

A esperança da distribuição, então, é calculada da forma usual:

$$D = \sum_{i=1}^N p_i \cdot y_i$$

Um passo complementar é também sugerido por Yager (1996), que consiste em verificar a compatibilidade entre o valor inferido D e o seu conjunto original, Y. Uma vez que a esperança retorna o valor mais provável do conjunto Y, mas isso não necessariamente significa que D contenha a extensão máxima de informação contida em Y, então pode-se calcular uma medida que caracteriza o conhecimento contido neste valor de D:

$$K(D, Y) = Poss(D, \Gamma) = \max(y_i, \Gamma(y_i)),$$

onde $Poss(D, \Gamma)$ é uma medida de possibilidade, ou seja, o Supremo entre a norma-t (geralmente, o mínimo) entre D e Γ e

$$\Gamma(y_i) = \frac{\exp(-(y_i - D))^2}{\sum_{i=1}^N \exp(-(y_i - D))^2}$$

que pode ser entendida como uma distribuição de probabilidades que caracteriza a distância entre os elementos de Y e o valor inferido D. Pode-se perceber que K é máximo quando D é igual ao maior valor do conjunto Y.

2.3.5 Outros Métodos

Além dos métodos mais usuais apresentados, existe ainda extensa literatura disponível, visto que a defuzzificação como um todo tem sido o foco de muita pesquisa. A seguir, serão citados mais alguns métodos propostos por pesquisadores do assunto.

- RTI (*Run-Time Integration*) e AI (*Adaptative Integration*): em vista de incrementar os métodos de centro de gravidade, Watanabe, Dettloff e Yount (1990) propuseram um método que considera as áreas de todos os conjuntos de saída. Para tanto, integrações são necessárias e o custo computacional, por vezes, é alto.

Pensando nisso, Eisele, Hentschel e Kunemud (1994) procuraram melhorar este método integrativo, levando em conta a forma do conjunto de saída Y. Mais detalhes podem ser encontrados nas referências mencionadas.

- IV (*Influence Value*) e DECADE (*Decreased Effort Centroid Defuzzification Algorithm*): estes métodos são outras alternativas propostas devido ao alto custo computacional nas integrações necessárias do método COS. O que o IV faz é calcular um centróide aproximado trabalhando com a base de regras e criando pseudo-regras nas áreas de sobreposição entre elas.

Já o DECADE utiliza um algoritmo com algumas operações básicas, mas também chega a um centróide aproximado. Para detalhes sobre ambos os métodos, veja ??) e Runkler (1996), respectivamente.

- **CDD** (*Constraint Decision Defuzzification*): considera que a defuzzificação é um processo decisório no qual um valor *crisp* é escolhido entre limites difusos (*fuzzy constraints*). Desta forma, um outro componente *fuzzy* é inserido no sistema e coopera com um defuzzificador básico, como o MOM ou COA, etc.

Em outras palavras, este novo componente descreve a saída do sistema como “o valor D deveria ser grande” ou “o valor D deveria ser próximo ao COA”, por exemplo. Detalhes adicionais podem ser obtidos em Runkler e Glesner (1993a).

- **FCD** (*Fuzzy Clustering Defuzzification*) e **KBMD** (*Knowledge-Based Mountain-Like Defuzzification*): proposto por Runkler, Glesner e Genther (1994), o FCD consiste em dividir o conjunto de saída Y em aglomerados (*clusters*), nos quais cada elemento recebe um grau de pertinência do aglomerado. O algoritmo então calcula o centro de cada aglomerado e lhe atribui também um grau de pertinência. O valor D então é computado como aquele com o maior grau.

O KBMD, proposto por Yager (1995), também faz uso de aglomerados no processo, mas incrementa o algoritmo com uma função que torna possível uma combinação entre diferentes soluções.

- **SB** (*Similarity-Based*): este método oferece uma interpretação semântica para o problema da defuzzificação, com o auxílio de operadores que visam auxiliar o processo na presença de ambiguidade linguística. Uma descrição detalhada do mesmo é fornecida por Boixader, Jacas e Recasens (1997).
- **WS** (*Weighted Singleton*) e **LI** (*Linear Interpolation*): ambos os métodos são propostos em Brubaker (1993). No primeiro, o sistema trabalha com *singletons* e retorna saídas simples, que são agrupadas segundo um peso pré-estabelecido. No segundo método, é feita uma interpolação entre os “degraus” da superfície de resposta do sistema.

Como o próprio autor afirma, tais métodos tendem mais para uma “combinação” do que para a “defuzzificação” propriamente dita, embora os resultados possam ser similares aos de conhecidos defuzzificadores.

- **WI** (*Weighted Intervals*): na defuzzificação, todo o conjunto Y é convertido em um valor *crisp*, podendo haver considerável perda de informação, dependendo do problema a ser abordado. Pensando nisso, Poleshuk e Komarov (2008) propuseram um método no qual o valor defuzzificado é um “intervalo de importância”, que conserva parte da informação contida no conjunto inferido.

- *MTV (Most Typical Values)*: valendo-se da Teoria da Tipicalidade, este método calcula os valores mais típicos do conjunto inferido e avalia as decisões às quais eles levam. Através de um fator chamado “multiplicidade”, é escolhida a solução mais popular ou, se diferentes soluções apresentam uma mesma multiplicidade, então é calculada a média das soluções.

Um detalhamento melhor desta técnica é encontrado em Kandel e Friedman (1998).

- *SDM (Saade’s Defuzzification Method)*: procurando uma alternativa a métodos probabilísticos, Saade e Diab (2000) propuseram um método que faz um “ranqueamento” de saídas *crisp* e escolhe o valor para a saída do sistema. Esta técnica, segundo o autor, pode ser utilizada em Controladores *Fuzzy* que utilizem saídas *crisp*, como o min-max WAF.
- *FMA (Fuzzy Multi-Criteria Analysis)*: alguns métodos de defuzzificação são utilizados no problema de avaliação multi-critérios. Dentro deste contexto, pode-se citar os mais usuais: o método do Valor à Direita (*Right Value*) baseia-se na porção direita do conjunto inferido para avaliar as alternativas de decisão; o método do Grau de Otimalidade (*Degree of Optimality*) avalia o desempenho de cada alternativa com respeito a cada critério a ser analisado, para se chegar a uma decisão ótima; o método do α -corte utiliza os α -cortes no desempenho de pesos dos critérios; o método do Grau de Dominância (*Degree of Dominance*) compara conjuntos difusos para determinar qual deles exerce dominância sobre os demais.

Tais métodos podem ser vistos mais detalhadamente em Deng, Lau e Millar (2000) e Deng e Yeh (2006).

Evidentemente que estes são apenas alguns dos métodos atualmente pesquisados. Citar todos seria uma impossibilidade, já que cada vez mais técnicas são propostas para procurar contornar o problema da defuzzificação.

Existem vários estudos sobre a escolha de um método ideal de defuzzificação, comparação de métodos, ranqueamento, propriedades procuradas dos defuzzificadores e assuntos correlatos, mas ainda hoje não se chegou a uma conclusão ótima sobre o tema. Ao leitor interessado no assunto, algumas referências podem ser consultadas, como, por exemplo, em Shi e Sen (2000), Leekwijck e Kerre (1999) e Saade (1996), além das já mencionadas nas seções anteriores.

Uma visão mais aprofundada dos métodos usuais pode ser encontrada, por exemplo, em Kovacic e Bogdan (2006) e sobre os fundamentos e axiomas da defuzzificação, sugere-se Thiele (1998).

2.4 MÉTODOS DEFUZZIFICADORES EM SISTEMAS DO TIPO-2

Assim como em conjuntos do tipo-1, há uma série de propostas para se proceder com a defuzzificação em conjuntos do tipo-2. Os métodos mais usuais consistem em duas etapas, ou seja, na redução de tipo e numa posterior aplicação de algum método de defuzzificação de conjuntos do tipo-1, conforme explicitado na seção anterior.

Existem duas metodologias principais que podem ser utilizadas para se reduzir um conjunto do tipo-2 para um conjunto do tipo-1. A primeira delas é uma redução exata, denominada CTR - *Centroid Type Reduction* (KARNIK; MENDEL; LIANG, 1999) - baseada nos centróides dos conjuntos embutidos. A segunda é uma redução aproximada, porém bastante eficiente, e que reduz sobremaneira o custo computacional. Esta segunda abordagem, proposta por Lucas (2009) chama-se VSCTR (*Vertical Slice Centroid Type Reduction*), por ser baseada nas fatias verticais do conjunto tipo-2.

Contudo, pode-se considerar outras metodologias, que podem ou não passar pela redução de tipo. Será visto, nesta seção, as principais metodologias de redução de tipo e de defuzzificação.

2.4.1 Redução de Tipo

A redução de tipo através do método CTR consiste em três etapas, a saber:

- 1º Passo: É preciso identificar todos os conjuntos embutidos (*embedded sets*) no conjunto de interesse, ou seja, todas as combinações possíveis de pontos presentes no conjunto do tipo-2 que, separadamente, podem ser entendidos como um conjunto do tipo-1.

- 2º Passo: Para cada um dos conjuntos embutidos, é calculado o centróide do mesmo, ou seja,

$$C = \frac{\sum_{j=1}^N x_j u_j}{\sum_{j=1}^N u_j}.$$

- 3º Passo: Por fim, é feita a combinação dos graus de pertinência com seus respectivos centróides calculados, que serão os pontos do conjunto reduzido, através da seguinte fórmula:

$$Cr = \phi_k / C_k,$$

onde

$$\phi_k = \mu_1^k \mu_2^k \dots \mu_n^k.$$

Quando é produzido mais de um ponto nesta operação, mantém-se aquele cujo grau de pertinência é o maior. A norma-t é usualmente dada pelo mínimo.

A fim de se reduzir o custo computacional, a técnica VSCTR consiste em uma redução de tipo através dos centróides das fatias verticais, ou seja, calcula-se o centróide para cada fatia e é feita a combinação simples destes centróides com seus respectivos pontos. Lucas (2009) demonstra que o valor defuzzificado aproximado fica bastante próximo do valor real, a um custo computacional significativamente menor.

2.4.2 Defuzzificador Direto

Proposto por Copland e John (2006), este método pode ser utilizado em conjuntos do tipo-2 intervalares. O defuzzificador direto secciona o conjunto de interesse em uma função de pertinência superior e outra inferior, calculando um centro de área diretamente do conjunto tipo-2, ponderando-o através destas funções seccionadas, ou seja,

$$C_{\tilde{A}} = \frac{\sum_{i=1}^N \overline{\mu}_{\tilde{A}}(x_i)x_i + \sum_{i=1}^N \underline{\mu}_{\tilde{A}}(x_i)x_i}{\sum_{i=1}^N \overline{\mu}_{\tilde{A}}(x_i) + \sum_{i=1}^N \underline{\mu}_{\tilde{A}}(x_i)}$$

onde $\overline{\mu}_{\tilde{A}}(x_i)$ e $\underline{\mu}_{\tilde{A}}(x_i)$ são as funções de pertinência superior e inferior, respectivamente.

Copland e John (2006) compararam o seu defuzzificador com outras metodologias e os resultados apresentaram-se bastante satisfatórios, principalmente em relação ao método da amostragem que será visto a seguir.

2.4.3 Método da Amostragem

Também proposto a fim de diminuir o custo computacional na redução do tipo, esse método simplesmente amostra os conjuntos embutidos de um conjunto do tipo-2, procedendo-se o cálculo da defuzzificação da mesma maneira. A diferença é que, notavelmente, um número muito menor de cálculos é requerido, já que não se consideram todas as combinações possíveis de conjuntos embutidos (GREENFIELD; JOHN; COUPLAND, 2005).

A grande vantagem deste método é que ele não requer uma amostra de tamanho muito grande para se conseguir uma boa aproximação do valor final defuzzificado e que, diferente do Método Direto, ele pode ser utilizado em conjuntos gerais. Contudo, comparando-o com outras técnicas (COPLAND; JOHN, 2006), sua superfície de resposta apresenta-se bastante “ruidosa” e muito pouco suave.

A fim de se contornar este problema, Linda e Manic (2010) propuseram que fosse utilizada, na amostragem, uma distribuição de probabilidades, por exemplo, a gaussiana. Os parâmetros desta distribuição são calculados através da FOU, sendo a média dada por

$$m(x) = \frac{\overline{\mu}_A^\alpha(x) + \underline{\mu}_A^\alpha(x)}{2}$$

e o desvio padrão dado por

$$s(x) = \frac{|\overline{\mu}_A^\alpha(x) - \underline{\mu}_A^\alpha(x)|}{2^\beta}$$

onde $\overline{\mu}_A^\alpha(x)$ e $\underline{\mu}_A^\alpha(x)$ são as funções de pertinência superior e inferior do conjunto projetado na FOU, α é um alfa-plano qualquer e β controla a propagação da distribuição gaussiana.

A utilização desta função como função densidade de probabilidade no processo de amostragem melhorou razoavelmente a superfície de resposta obtida através deste método, porém, tal abordagem é válida apenas em conjuntos simétricos, onde a ponderação feita irá amostrar valores no centro do conjunto. Em situações de outro tipo, a amostragem poderá se tornar viciada.

2.5 PROBABILIDADE E INFERÊNCIA BAYESIANA

Alguns autores, como Tibiriçá e Nassar (2003) sugerem uma abordagem híbrida para o tratamento da incerteza, a qual eles denominam de “Abordagem *Fuzzy*-Probabilística”. Trata-se de uma forma de tratar a incerteza por aleatoriedade (probabilidade) e a incerteza por imprecisão (possibilidade) conjuntamente. Viu-se que a última é tratada através de Sistemas *Fuzzy*, dando origem aos tipos de sistema previamente explicados.

Já a primeira, como dito anteriormente, é tratada através da Teoria das Probabilidades e, dentro deste contexto, existe a Inferência Bayesiana. A diferença básica entre esse tipo de inferência e a inferência clássica é que, ao se construir um modelo para tratar a incerteza, todos os parâmetros deste modelo são encarados como variáveis aleatórias, ao passo que na inferência clássica, isso não acontece.

Para descrever de forma mais detalhada a lógica da Inferência Bayesiana, faz-se necessária uma breve revisão da literatura sobre a Teoria das Probabilidades. A seguir são revistos alguns conceitos que serão necessários para entendimento das propostas que serão feitas nesta dissertação. Cabe salientar que esta é apenas uma introdução sobre probabilidade e inferên-

cia bayesiana e, para um estudo mais aprofundado do assunto, sugere-se, por exemplo, Meyer (1983) e Paulino, Turkman e Murteira (2005).

2.5.1 Teoria das Probabilidades

Considerando um experimento E qualquer, o conjunto de todos os resultados possíveis deste experimento é chamado de espaço amostral, ω . Um determinado evento, A é definido como sendo um subconjunto deste espaço amostral e a probabilidade de ocorrência deste evento A , em um universo de variáveis discretas, pode ser calculada através da definição clássica de probabilidade (MEYER, 1983), a saber:

$$P(A) = \frac{\#A}{\#\omega},$$

onde $\#A$ é o número de casos favoráveis ao evento A e $\#\omega$ é o número todos os casos possíveis em um determinado experimento.

A função de probabilidade obedece aos seguintes axiomas:

- $0 < P(A) < 1$
- $P(\omega) = 1$ e $P(\emptyset) = 0$.
- Para eventos A_i que sejam mutuamente exclusivos, $P(\bigcup A_i) = \sum P(A_i)$.

Um dos teoremas mais importante de probabilidade diz respeito à Probabilidade Condicional, ou seja, a probabilidade de um evento A , condicionada à ocorrência do evento B será dada por:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}. \quad (1)$$

Daí decorre que imediatamente que, para dois eventos independentes (ou seja, não relacionados entre si), A e B , $P(A \cap B) = P(A) \cdot P(B)$. Ainda, se há a ocorrência de partições do espaço amostral ω , a probabilidade de um evento B , relacionada às partições de um evento C , será calculada através do Teorema da Probabilidade Total:

$$P(B) = P(B|C_1) \cdot P(C_1) + P(B|C_2) \cdot P(C_2) + \dots + P(B|C_p) \cdot P(C_p) \quad (2)$$

No caso em que se deseja calcular uma probabilidade condicionada a um evento cujo espaço amostral esteja particionado, utiliza-se o Teorema de Bayes:

$$P(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)}, \quad (3)$$

onde, nesse caso, o termo $P(B)$ será calculado pelo Teorema da Probabilidade Total.

Estes conceitos, em especial as equações 2 e 3, serão utilizadas para a obtenção das probabilidades de ocorrências dos consequentes do SFP proposto.

2.5.2 Inferência Bayesiana

Este tipo de inferência faz uso do Teorema de Bayes para que a informação *a priori* que se tem sobre um certo parâmetro seja atualizada através da função de verossimilhança. Na Inferência Bayesiana, em contraposição ao que se faz na Inferência Clássica, todos os parâmetros são considerados como sendo variáveis aleatórias (PAULINO; TURKMAN; MURTEIRA, 2005). Desta forma, faz sentido falar em distribuições de probabilidade para cada um dos parâmetros e a eles é atribuída uma distribuição *a priori*, que reflete o conhecimento prévio que existe acerca dos dados que serão observados. Após a amostragem das observações, a distribuição *a priori* é combinada com a função de verossimilhança através do Teorema de Bayes, originando a distribuição *a posteriori*

$$p(\theta|x) = \frac{p(x|\theta) \cdot p(\theta)}{p(x)}$$

que pode ser utilizada, então, para se obter estimativas dos parâmetros θ de interesse.

A função de verossimilhança, $p(x|\theta)$, associa, para um valor fixo de x , a probabilidade de ser observado este x , a cada valor que θ assume. Assim, observa-se a plausibilidade (ou verossimilhança) de cada um dos valores de θ quando os dados (amostra) são observados. Importante salientar, contudo, que os conceitos de probabilidade e plausibilidade são diferentes: neste último, fixa-se a amostra (x) e varia-se o parâmetro (θ), ao passo que no primeiro, ocorre o inverso.

A função de verossimilhança é obtida através do produtório da distribuição de probabilidades dos dados amostrais, variando-se o parâmetro de interesse. Por exemplo, suponha uma série de variáveis aleatórias X_1, \dots, X_n , independentes e identicamente distribuídas, com função de probabilidades $f(x_i|\theta)$. Sua função de verossimilhança será dada pela distribuição conjunta

$$L(\theta|x) = f(x|\theta) = \prod_i f(x_i|\theta).$$

Existem famílias de distribuições conjugadas (EHLERS, 2003), o que torna mais simples a obtenção da *posteriori*, dispensando o cálculo deste produtório, sendo que o caso mais usual é o da distribuição gaussiana. Supondo que $x|\theta \sim N(\theta, \sigma^2)$ e *a priori* $\theta \sim N(\lambda_0, \tau_0^2)$, sendo que θ é o parâmetro de interesse, τ^2 é chamado de “precisão” e calculado como o inverso da variância, então *a posteriori*, considerando um tamanho de amostra n , será $\theta|x \sim N(\lambda_1, \tau_1^2)$, onde:

$$\lambda_1 = \frac{\tau_0^{-2} * \lambda_0 + n * \sigma^2 * \bar{x}}{\tau_0^{-2} + n * \sigma^{-2}} e \quad (4)$$

$$\tau_1^{-2} = \tau_0^{-2} + n * \sigma^{-2}, \quad (5)$$

Na maioria das vezes, os parâmetros *a priori* são desconhecidos e é preciso especificar *prioris* também para estes parâmetros. Nesta dissertação, contudo, a título de simplificação das operações, assume-se que os valores são conhecidos e iguais às estimativas obtidas dos valores observados na amostra dos dados de treinamento. Também será assumida a distribuição gaussiana para os dados utilizados com fins de experimento.

3 SISTEMAS *FUZZY* PROBABILÍSTICOS: GERAÇÃO AUTOMÁTICA DE REGRAS E DEFUZZIFICAÇÃO BAYESIANA

Considerando as vantagens dos sistemas híbridos (*fuzzy* + probabilidade) frente às discussões de superioridade de cada um deles, Meghdadi e Akbarzadeh-T (2001) propuseram a Lógica *Fuzzy* Probabilística e Sistemas *Fuzzy* Probabilísticos (SFP). Conforme discutido no Capítulo 2, neste tipo de sistema, onde ambos os tipos de incerteza são considerados, há uma base de regras cujo consequente é seccionado em diferentes conjuntos, cada qual com uma determinada probabilidade de ocorrência.

Esta mudança de estrutura no sistema possibilita o tratamento de regras conflitantes em um sistema de inferência *fuzzy* que utiliza um algoritmo pré-estabelecido para geração da base de regras a partir de dados de treinamento. Neste caso, algumas regras terão uma probabilidade maior de ocorrência do que outras e isso pode ser tratado através de um SFP. Assim, este trabalho busca apresentar um método automático de geração de regras *fuzzy* para um SFP.

No entanto, apesar de ser um dos pontos de maior relevância deste trabalho, a geração automática de regras não é a única contribuição da pesquisa realizada. Outro ponto importante a ser considerado é o seguinte: dado que um SFP produzirá diferentes saídas a cada amostragem, a fase de defuzzificação pode depender de uma abordagem mais sofisticada. Portanto, propõe-se a utilização de Inferência Bayesiana para tratar as saídas do sistema, de forma a combinar informações *a priori* disponíveis sobre essas saídas, com o objetivo de tratar a informação amostral produzida pelo sistema. Assim, as seções a seguir detalham essas duas propostas.

3.1 MÉTODO PARA GERAÇÃO DE REGRAS *FUZZY* PROBABILÍSTICAS

Antes da apresentação do método proposto, julga-se necessária uma análise mais detalhada das afirmações de que o método de Wang Mendel (WM) desconsidera informações de redundância e inconsistência possivelmente presentes na base de treinamento.

Suponha, por exemplo, que se queira aplicar o algoritmo de WM a um problema de classificação. Com base no algoritmo Wang Mendel original descrito na seção 2.2.1, e definindo o

termo “Região de Interesse” como sendo o produto cartesiano entre os suportes dos antecedentes de uma mesma regra, os passos 2 e 3 do algoritmo poderiam ser reescritos como a seguir e ilustrados nas figuras 8 e 9.

- Passo2: Defina a região de interesse ao qual o ponto pertence, ou seja, para cada variável de entrada, selecione a função de pertinência de maior grau, criando uma regra *fuzzy* associada a cada ponto.
- Passo 3: Calcule o quanto cada ponto pertence à região de interesse, ou seja, calcule o grau de ativação de cada regra, utilizando uma norma-t apropriada.

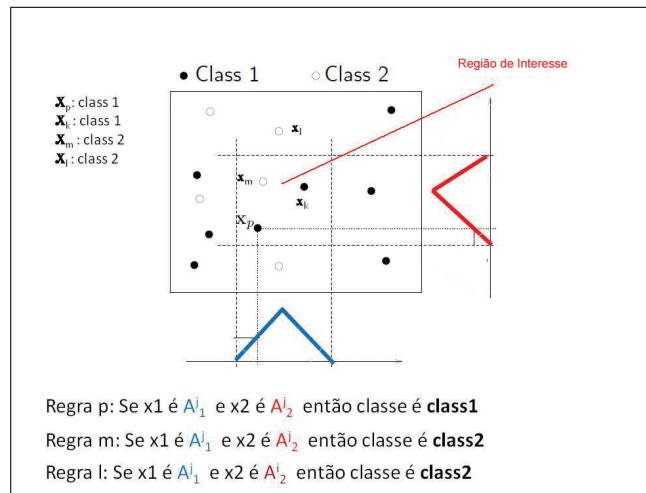


Figura 8: (a)WM: Passo2

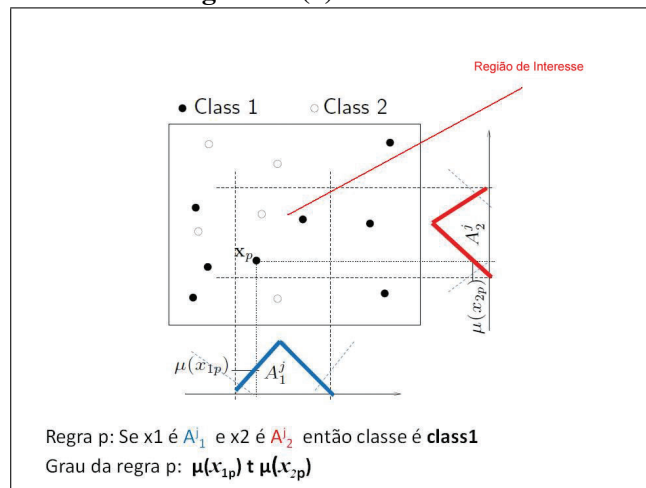


Figura 9: (b)WM: Passo3

Agora, imagine que o antecedente das regras p e k apareça também em 10 outras regras de uma base de regras. Ou seja, 10 pontos associados à classe 1 e utilizados para treinamento estão numa mesma região de interesse e por isso ativam os mesmos conjuntos *fuzzy* do antecedente.

Suponha agora que o antecedente da regra m apareça em apenas 1 regra desta mesma base. Ou seja, esse ponto, associado à classe 2 e utilizado para treinamento, está na mesma região anterior, mas com grau de ativação ligeiramente superior.

Considerando o passo 4 do algoritmo WM reescrito como

- Passo 4: Ignore todas as informações de redundância e de inconsistência e preserve apenas a regra com maior grau de ativação na região de interesse;

Portanto, a regra m permanecerá na base, mas talvez não esteja representando corretamente a informação contida no conjunto de dados.

Assim, o que se propõe é uma alteração no algoritmo WM clássico conforme apresentado na próxima seção.

3.1.1 Algoritmo de Geração de Regras *Fuzzy* Probabilísticas

Com o objetivo de contornar os problemas levantados na seção anterior, assim como apresentar um método de geração automática de regras para sistemas *fuzzy* probabilísticos, esta seção apresenta uma alternativa ao método WM, resumida nos seguintes passos:

1. Defina o número de termos linguísticos e particione o universo de todas as variáveis.
2. Defina a região de interesse T_r ao qual o ponto pertence, ou seja, para cada variável, selecione a função de pertinência de maior grau, criando uma regra *fuzzy* associada a cada ponto.
3. Calcule o quanto cada ponto associado ao consequente K_i pertence à região de interesse T_r , ou seja, calcule o grau de ativação GA_{T_r, K_i} da regra “Se x_1 é A_j e x_2 é A_k , então y é K_i ”, utilizando uma norma-t apropriada.
4. Calcule N_i = quantidade de pontos associados ao consequente K_i aparece na região de interesse, ou seja, quantas vezes a regra “Se x_1 é A_j e x_2 é A_k , então y é K_i ” aparece na base de regras (total de regras redundantes).
5. Calcule N = quantidade de pontos na região de interesse, ou seja quantas vezes o antecedente da regra aparece na base de regras.
6. Calcule $P(K_i) = N_i/N$, a probabilidade inicial de ocorrência do consequente K_i considerando toda a região de interesse.

7. Calcule para o consequente K_i , o valor de GA_{T_r, K_i}^{max} , como o grau máximo de ativação das regras redundantes (mesmos antecedentes e consequentes) na região de interesse.
8. Seja $GA_{T_r}^{max}$ o conjunto de todos os pontos dessa região que possuem o maior grau de ativação (foco na região de interesse). Combine a probabilidade inicial $P(K_i)$ com o grau máximo de ativação das regras de consequente K_i , GA_{T_r, K_i}^{max} , através do Teorema de Bayes, obtendo a probabilidade final para cada consequente possível, $P(K_i|GA_{T_r}^{max})$. Ou seja, obtenha a probabilidade de ocorrência do consequente K_i , dado que se está no foco $GA_{T_r}^{max}$ da região T_r . Desta forma, combina-se uma informação de quantidade dos pontos associados a K_i ($P(K_i)$) com uma informação de qualidade dos pontos associados a K_i (GA_{T_r, K_i}^{max}).

Utilizando essas mudanças no algoritmo, todas as regras geradas permanecerão na base, inclusive as inconsistentes. Porém, com diferentes probabilidades de ocorrência.

Por exemplo, supondo duas regras inconsistentes,

- Se x_1 é A_j e x_2 é A_k , então y é K_1
- Se x_1 é A_j e x_2 é A_k , então y é K_2

ambas permanecerão da seguinte forma:

- Se x_1 é A_j e x_2 é A_k , então y é K_1 com $P(K_1|GA_{T_r}^{max})$ e y é K_2 com $P(K_2|GA_{T_r}^{max})$.

O passo-a-passo completo do algoritmo será detalhado através de um exemplo na seção 3.1.3. O passo final (passo 8), que faz uso do Teorema de Bayes para formalizar a regra resultante descrita anteriormente, será detalhado a seguir.

3.1.2 Consequentes Probabilísticos e o Teorema de Bayes

O passo 8 do algoritmo proposto permite que, ao invés de se utilizar somente a análise frequentista (probabilidade $P(K_i)$) dada pela quantidade de pontos de um consequente na região de interesse, esta informação passe a ser combinada com a informação de qualidade dos pontos, medida pelo grau de ativação desses pontos aos conjuntos que definem a região de interesse.

A grande questão que surge é:

- como calcular a probabilidade $P(K_i|GA_{T_r}^{max})$ de forma a combinar estas informações garantindo que os axiomas e teoremas da probabilidade sejam preservados?

Conforme visto na seção 2.5.2, se há uma partição no espaço amostral e deseja-se calcular uma probabilidade condicional, utiliza-se o Teorema de Bayes, formalizado na equação 3.

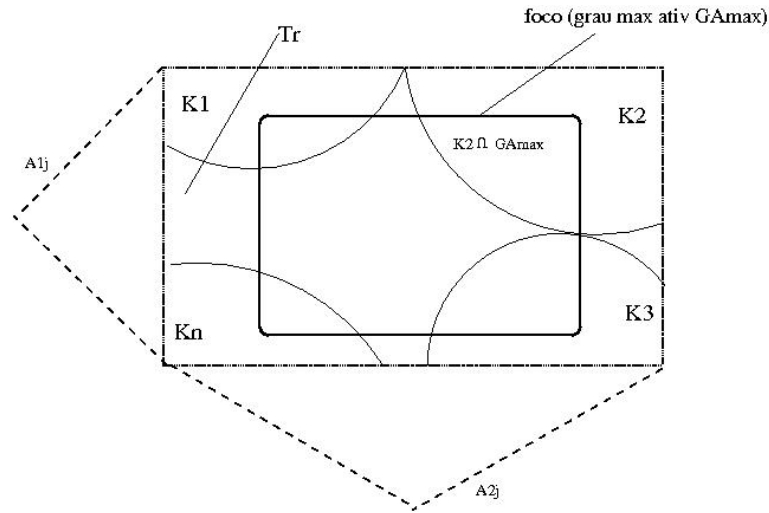


Figura 10: Exemplo de Partição do Espaço Amostral $T_{r=1}$

No caso geral de uma base de regras de um sistema *fuzzy* obtida por um algoritmo automático de geração, cada região de interesse T_r pode ser entendida como um espaço amostral (ou seja, $T_r = \omega$), que está particionado em diferentes consequentes, K_i . A figura 10 ilustra esta ideia. Deste modo, deseja-se obter uma regra representativa desta região de interesse (T_r), de forma que as informações de redundância e inconsistência possam ser preservadas. Para isso, é preciso associar uma medida de probabilidade a cada consequente (K_i) que ocorre nesta região, utilizando a maior parte da informação (pontos) disponível.

Não é interessante, entretanto, que sejam considerados pontos com graus baixos de ativação neste cálculo. Isto porque pontos que têm consequente K_i mas que possuem grau baixo de ativação não são representativos daquela determinada região de interesse. Em outras palavras, pontos desta região com um alto grau de ativação possuem maior qualidade e devem ser utilizados, em detrimento dos pontos de menor qualidade. Tem-se, portanto, uma “sub-região” de interesse, onde está o foco, ou seja, o conjunto formado por todos os pontos com grau de ativação máxima. Como pode ser que exista um único ponto de cada classe onde o valor é máximo, este conjunto pode ter cardinalidade igual ao total de classes.

Então, o que se deseja obter no passo 8 do algoritmo proposto é, na região T_r , a probabilidade do consequente ser K_i , dado que os pontos considerados são de qualidade, ou seja, que se está na sub-região de maior interesse, onde a pertinência é máxima. Deste modo, pode-se reescrever o Teorema de Bayes da seguinte maneira:

$$P(K_i|GA_{Tr}^{max}) = \frac{P(K_i \cap GA_{Tr}^{max})}{P(GA_{Tr}^{max})}, \quad (6)$$

onde:

$P(K_i|GA_{Tr}^{max})$: Probabilidade Final do consequente ser K_i , dado que está na região de maior nível de ativação.

$P(K_i \cap GA_{Tr}^{max})$: Probabilidade de se obter o grau máximo de ativação na região de interesse, para o consequente K_i .

$P(GA_{Tr}^{max})$: Probabilidade Total da região de maior interesse Tr , calculada conforme equação 2.

Heuristicamente, pode-se considerar que a probabilidade $P(GA_{Tr}^{max}|K_i)$ é entendida como sendo o próprio grau máximo de ativação para a classe K_i , ou seja, $P(GA_{Tr}^{max}|K_i) = GA_{Tr,K_i}^{max}$. Embora a prova desta passagem esteja fora do escopo deste trabalho, pode-se utilizar este conceito e o Teorema da Probabilidade Condicional, reescrevendo a interseção, sendo que a equação final será da forma:

$$P(K_i|GA_{Tr}^{max}) = \frac{P(K_i).P(GA_{Tr}^{max}|K_i)}{P(GA_{Tr}^{max})} = \frac{P(K_i).GA_{Tr,K_i}^{max}}{P(GA_{Tr}^{max})}, \quad (7)$$

onde:

$P(K_i)$: Probabilidade inicial de ocorrência do consequente K_i , calculada conforme descrito no passo 6 do algoritmo.

GA_{Tr,K_i}^{max} : Grau máximo de ativação para o consequente K_i , na região de interesse Tr , calculado conforme o passo 7 do algoritmo.

Obtém-se, portanto, uma probabilidade normalizada, que combina tanto a probabilidade inicial (calculada de acordo com o número de pontos na região de interesse que apontam para um determinado consequente) quanto o grau de ativação que é utilizado no método tradicional de Wang-Mendel, ou seja, as informações na mesma região de interesse são agregadas tanto probabilística quanto possibilisticamente.

3.1.3 Um Exemplo Simples

Como ilustração do método proposto, imagine um problema de classificação com as partições das variáveis de entrada obtidas pelo especialista, conforme o passo 1 do algoritmo descrito na seção 3.1.1, e mostradas nas figuras 11 e 12.

Considere ainda a base de regras e seus respectivos Graus de Ativação, conforme tabela 1,

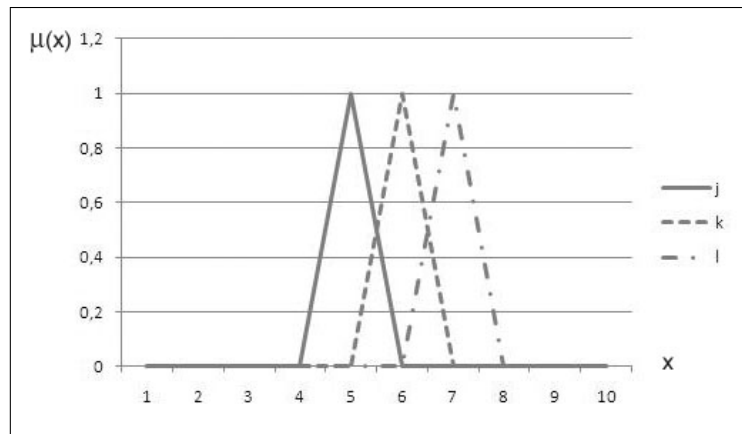


Figura 11: Exemplo de Partição para a variável de entrada A

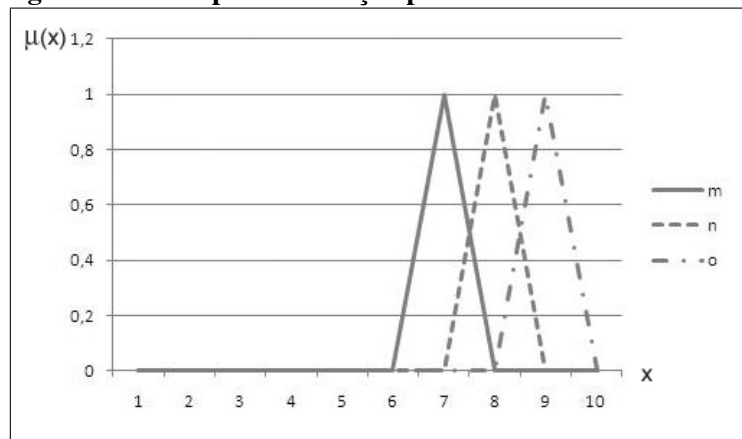


Figura 12: Exemplo de Partição para a variável de entrada B

obtidos pelos passos 2 e 3 do algoritmo.

Utilizando os passos 4, 5 e 6 para se computar as probabilidades iniciais e, agregando-se as regras redundantes pelo máximo, de acordo com o passo 7, a base ficaria reduzida na forma da tabela 2.

Agora, as probabilidades totais em cada região de interesse serão calculadas através da equação 2, resultando em:

$$P(GA_{Tr=1}^{max}) = 2/3 * 0,9 + 1/3 * 0,7 = 0,83,$$

$$P(GA_{Tr=2}^{max}) = 2/3 * 0,9 + 1/3 * 0,3 = 0,70 \quad e$$

$$P(GA_{Tr=3}^{max}) = 1 * 0,5 = 0,50.$$

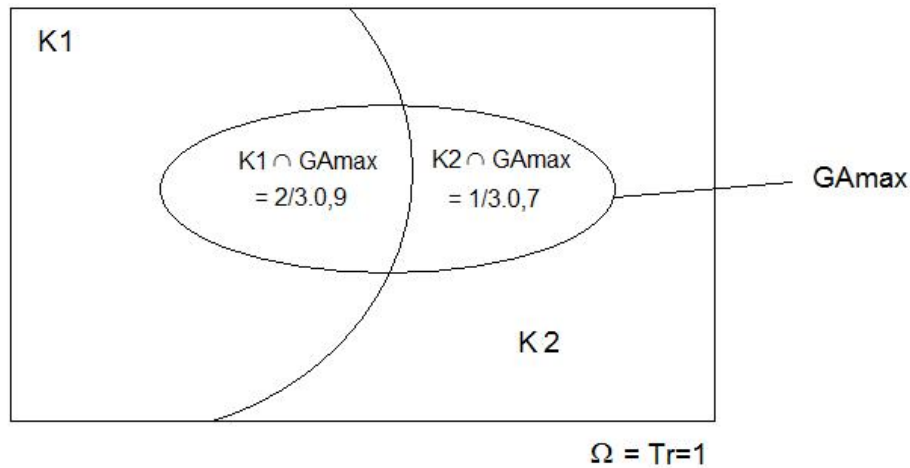
Tabela 1: Base de Regras do Exemplo

Tr	Se x_1 é	e x_2 é	então Y é	GA_{Tr,K_i}
1	A_j	B_m	1	0.8
1	A_j	B_m	1	0.9
1	A_j	B_m	2	0.7
2	A_k	B_n	2	0.9
2	A_k	B_n	2	0.6
2	A_k	B_n	1	0.3
3	A_l	B_o	3	0.5
3	A_l	B_o	3	0.4

Tabela 2: Base de Regras do Exemplo - Reduzida

Tr	Se x_1 é	e x_2 é	então Y é	GA_{Tr,K_i}^{max}	Prob. Inicial ($P(K_i)$)
1	A_j	B_m	1	0,9	2/3
1	A_j	B_m	2	0,7	1/3
2	A_k	B_n	2	0,9	2/3
2	A_k	B_n	1	0,3	1/3
3	A_l	B_o	3	0,5	1

Para melhor entendimento, a visualização da partição do espaço amostral $T_{r=1}$ é fornecida na figura 13.

**Figura 13: Exemplo de Partição do Espaço Amostral $T_{r=1}$ para o exemplo numérico**

Finalmente, no passo 8, as probabilidades finais $P(K_i|GA_{Tr}^{max})$ serão calculadas pelo Teorema de Bayes, conforme a equação 7. Assim, multiplicando a probabilidade inicial pelo grau de ativação máximo e dividindo-o pela respectiva probabilidade total, a base final será dada pela tabela 3.

Tal base de regras poderá ser tratada, portanto, através do já citado Sistema de Inferência Fuzzy Probabilístico (SFP).

Tabela 3: Base de Regras do Exemplo - Final

Se x_1 é	e x_2 é	então Y é	Prob. Final
A_j	B_m	1	0.7189
A_j	B_m	2	0.2811
A_k	B_n	2	0.8571
A_k	B_n	1	0.1429
A_l	B_o	3	1

3.2 PROCESSO DE INFERÊNCIA EM SISTEMAS FUZZY PROBABILÍSTICOS

Na seção 2.1.4, foi descrito o funcionamento de um Sistema *Fuzzy* Probabilístico, sendo que os passos iniciais, como *matching* e agregação dos antecedentes, pouco diferem de um Sistema *Fuzzy* convencional. O passo de obtenção da saída inferida, contudo, é um pouco mais complexo, já que para cada regra, pode-se ter diferentes saídas, cada qual com sua respectiva probabilidade de ocorrência.

O que se faz, portanto, é utilizar um mecanismo de roleta para selecionar um entre os possíveis consequentes de cada regra. Depois deste passo complementar, a base de regras acaba reduzindo-se à forma do Sistema *Fuzzy* tradicional e, a partir daí, pode-se proceder com qualquer operador de agregação de regras e aplicar ao conjunto fuzzy obtido algum dentre os vários métodos defuzzificadores mencionados na seção 2.3, como, por exemplo, o COA (centróide).

Entretanto, devido à natureza estocástica do sistema, cada rodada da roleta pode “escolher” um consequente distinto, produzindo uma saída diferente para um mesmo padrão de entrada. Assim, de forma a proporcionar uma amostra significativa de resultados, o que usualmente se faz é rodar várias iterações do processo (gerar várias amostras), obtendo várias saídas defuzzificadas, que podem ser combinadas através de uma estatística pré-definida, como a média, moda, ou mediana, dependendo da aplicação.

Assim, assume-se que após a obtenção da partição dos universos (definição das funções de pertinência de cada variável pelo especialista), da definição dos operadores que serão utilizados no processo de inferência (assim como no sistema *fuzzy* tradicional) e da obtenção da base de regras pelo método proposto, conforme descrito nas seções anteriores, a operacionalização do SFP pode ser resumida nos seguintes passos, para um determinado padrão p de entrada:

1. *Matching* em cada regra. Processo idêntico ao descrito na seção 2.1.1.
2. Agregação dos antecedentes. Processo idêntico ao descrito na seção 2.1.1 produzindo o nível de disparo ou grau de ativação de cada regra.

3. Utilização do mecanismo de roleta para selecionar qual consequente será utilizado no próximo passo.
4. O grau de ativação resultante do passo 2 é combinado com o consequente sorteado em cada regra, segundo uma semântica pré-definida, gerando as saídas inferidas.
5. As saídas inferidas são agregadas por meio de um agregador de regras (por exemplo, o máximo), formando o conjunto *fuzzy* resultante, Y .
6. Se necessário, o conjunto Y é defuzzificado para gerar a saída *crisp*, y_m .
7. Retorna-se ao passo 3 até que seja gerada uma amostra significativa para o padrão de entrada p considerado. O tamanho n da amostra de saídas pode ser previamente definido da forma que o operador julgar mais conveniente para seu uso, conforme já discutido no penúltimo parágrafo da seção 2.1.4.
8. Calcula-se uma estatística (por exemplo, a moda) sobre as amostras de saídas y_m , de forma a se obter a saída final y para o padrão p -ésimo de entrada.

Em sistemas classificadores, a saída será uma classe pré-definida, sendo o passo 6 suprimido e o passo 8 considerado um mero tomador de decisão. Contudo, em sistemas cuja saída é *fuzzy*, faz-se necessário o passo de defuzzificação e, na próxima seção, sugere-se como ela pode ser feita em Sistemas Probabilísticos.

3.3 DEFUZZIFICAÇÃO BAYESIANA EM SISTEMAS FUZZY PROBABILÍSTICOS

Conforme descrito na seção anterior, é usual que sistemas *fuzzy* probabilísticos trabalhem com algumas iterações a fim de se produzir uma amostra razoável das diferentes saídas do sistema. Utilizando um método defuzzificador como o do centróide, por exemplo, o que se faz é simplesmente calcular uma estatística (como a média) destes diferentes valores defuzzificados.

É razoável assumir, no entanto, que se pode ter alguma ideia *a priori* do comportamento da saída do sistema. Considerando que a amostra obtida pelas iterações pode ser entendida como a verossimilhança no teorema de Bayes, pode-se combinar essa informação de forma a obter uma estimativa “melhorada” para a saída do sistema, através da média da *posteriori*, ao invés de apenas calcular a média da amostra.

Esta operação é relativamente simples, conforme já foi descrito na seção 2.5.2. Nesse caso, o parâmetro θ de interesse é a saída do sistema e deseja-se encontrar a distribuição *a posteriori*

para este parâmetro, que será dada, segundo o Teorema de Bayes, por

$$p(\theta|x) = \frac{p(x|\theta) \cdot p(\theta)}{p(x)},$$

sendo que $p(x)$, chamada de distribuição preditiva, é apenas uma constante normalizadora para garantir que a distribuição seja de probabilidades, usualmente pode-se retirá-la da equação e substituir o sinal de igualdade por proporcionalidade:

$$p(\theta|x) \propto p(x|\theta) \cdot p(\theta)$$

A distribuição *a priori*, $p(\theta)$, pode ser definida de inúmeras maneiras. Pode-se, por exemplo, utilizar a informação de um especialista de forma a definir probabilidades subjetivas para o sistema. Pode-se trabalhar com *prioris* conjugadas, ou ainda, utilizar métodos de definição de *prioris* não-informativas. Para o estudo em questão, adotou-se o método de *prioris* conjugadas gaussianas, cujos parâmetros serão definidos a partir dos dados de treinamento do sistema. Para detalhes adicionais sobre o assunto, pode-se consultar, por exemplo, Paulino, Turkman e Murteira (2005).

Como visto na seção 2.5.2, o conceito da função de verossimilhança, $p(x|\theta)$, diz respeito à plausibilidade que é atribuída ao parâmetro, quando os dados (amostra) são observados. No caso das saídas de um sistema *fuzzy* que se deseja estimar, a amostra observada será dada pelas saídas realizadas quando o sistema efetivamente operar. Assim, é possível escrever o Teorema de Bayes da seguinte forma:

$$p(\theta|y) \propto p(y|\theta) \cdot p(\theta)$$

Relembrando, então, que $p(\theta)$ é a *priori* atribuída através dos dados de treinamento e $p(y|\theta)$ é a verossimilhança construída com os diferentes valores defuzzificados, obtidos na saída do sistema *fuzzy*. Atribuindo uma *priori* Normal, pode-se utilizar a propriedade descrita na seção 2.5.2, de forma que a média e variância da *posteriori* são dadas por

$$\lambda_1 = \frac{\tau_0^{-2} \cdot \lambda_0 + n \cdot \sigma^2 \cdot \bar{x}}{\tau_0^{-2} + n \cdot \sigma^{-2}} \quad \text{e} \quad (8)$$

$$\tau_1^{-2} = \tau_0^{-2} + n \cdot \sigma^{-2}, \quad (9)$$

onde:

τ_0^2 : Variância da *priori*.

λ_0 : Média da *priori*.

n : Tamanho da amostra (número de iterações pré-definido).

σ^2 : Variância dos dados da amostra.

\bar{x} : Média dos dados da amostra.

Como exemplo, suponha que um determinado SFP com 10 iterações tenha produzido as seguintes saídas para um certo padrão de entrada:

7,44 ; 6,42 ; 5,75 ; 6,09 ; 4,87 ; 4,50 ; 3,79 ; 3,79 ; 4,80 ; 5,27

Considerando uma *priori* $\sim N(4,75; 0,09)$ e, que estes dados resultam em média igual a 5,27 e desvio padrão de 1,17, utilizam-se as equações 8 e 9, obtendo-se a média

$$\lambda_1 = \frac{0,09^{-1} \cdot 4,75 + 10 \cdot 1,17^2 \cdot 5,27}{0,09^{-1} + 10 \cdot 1,17^{-2}} = 6,78$$

e variância como o inverso da precisão

$$\tau_1^{-2} = 0,09^{-1} + 10 \cdot 1,17^{-2} = 18,41.$$

Como a distribuição Normal é simétrica, utilizando qualquer um dos estimadores bayesianos usuais - média, moda ou mediana - a saída do sistema, ou seja, a estimativa para o parâmetro obtida, seria 6,78.

A grande desvantagem desse método talvez seja seu custo computacional, devido à necessidade de se trabalhar com um número de iterações que, por vezes, pode ser elevado. Contudo, pode-se perceber que ele pode ser utilizado em conjunto com algum dos métodos previamente apresentados na seção 2.3.

4 SIMULAÇÕES E RESULTADOS

No capítulo anterior, foram elaboradas propostas para geração de base de regras (BR) e defuzzificação em Sistemas *Fuzzy* Probabilísticos. Para que tais métodos sejam testados, o que se propõe são dois experimentos distintos, em diferentes áreas de aplicação de sistemas automáticos.

Sistemas automáticos são sistemas com dispositivos de auto-regulamentação, formados por leis e estratégias com as quais controlam seu comportamento. Para ser autônomo, um sistema precisa primeiramente ser automático, mas a autonomia depende ainda da capacidade de desenvolver suas próprias estratégias de controle (STEELS, 1995). Uma abordagem automática pressupõe um grau de intervenção baixo por parte do usuário na definição dos parâmetros críticos do modelo a ser produzido. Neste trabalho a definição automática ficará restrita à base de regras probabilísticas (combinação de termos linguísticos, total de regras e medidas de probabilidade associadas a cada regra) ficando a base de dados (partição do universo definida pelo número total, formato e localização das funções de pertinência) e os parâmetros do processo de inferência (mecanismo de raciocínio *fuzzy*) definidos pelo usuário.

Nesta proposta, a base de regras probabilísticas será definida automaticamente. Entretanto, os parâmetros relacionados à base de dados (formato e localização ou suporte das funções de pertinência) e os operadores associados ao mecanismo de raciocínio ou inferência (operadores de agregação do antecedente, semântica das regras, agregação de regras etc.), deverão ser definidos pelo usuário ou projetista.

O primeiro dos experimentos produzidos diz respeito à aplicação do método a um problema de classificação considerando-se duas instâncias distintas: uma onde as classes são bem distintas entre si e outra onde o nível de ruído na classificação aumenta e as classes são mais misturadas. São comparadas as eficiências do algoritmo de WM, do algoritmo Ishibuchi e do método probabilístico proposto. Como critério de comparação optou-se pelo percentual de acertos,

$$P_{Ac} = \frac{PC}{PT} \cdot 100,$$

onde PC = total de pontos de teste classificados corretamente e PT = total de pontos de teste. Também será utilizada para avaliação a matriz de confusão, que evidencia a quantidade de verdadeiros positivos e negativos.

Com o intuito de verificar como os diferentes métodos se comportam ao se aumentar a “confusão” presente nos dados, vários experimentos foram produzidos, com médias e desvios diferentes, conforme descrito na tabela 4.

Tabela 4: Experimentos Realizados para Classificação		
Experimento	Média Classes (1,2,3)	Desvio Padrão
1a	(10,30,50)	9
1b	(10,20,30)	9
1c	(10,30,50)	12
1d	(10,20,30)	12
1e	(10,30,50)	15
1f	(10,20,30)	15

O segundo experimento diz respeito à aplicação dos métodos (geração de regras e defuzzificação bayesiana) na previsão em séries temporais, sendo que duas séries são utilizadas a fim de comparar o algoritmo de WM, o método probabilístico simples e o método probabilístico com defuzzificação bayesiana. Para comparação das três técnicas, calculou-se o Erro Quadrático Médio de cada uma, que é dado pela expressão

$$\sum_{i=1}^N \frac{(O - R)^2}{N}, \quad (10)$$

onde O é o valor observado no experimento, R é o valor real e N é o total de iterações.

4.1 EXPERIMENTO 1: GERAÇÃO DE REGRAS PARA UM CLASSIFICADOR *FUZZY*

Com o intuito de se avaliar o desempenho do método probabilístico, comparando-o com dois métodos tradicionais de geração de classificadores *fuzzy*, implementou-se no software MATLAB um classificador *fuzzy* com duas entradas e uma saída.

Para fins de experimento, foi suposta a existência de três classes de pontos, razoavelmente separadas conforme a figura 14 (Experimento 1a), ou um pouco mais próximas entre si, conforme a figura 15 (Experimento 1b). Os classificadores probabilísticos obtidos a partir dos experimentos 1a e 1b são detalhados nas próximas sessões.

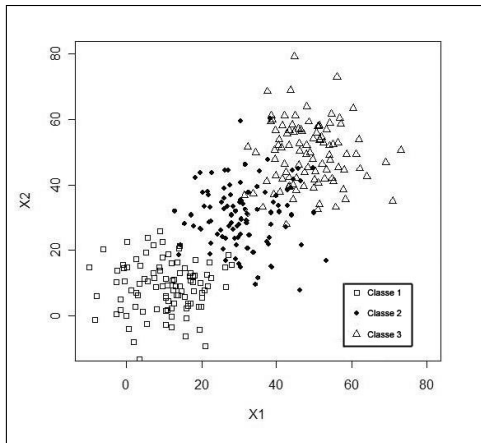


Figura 14: Classes de Pontos Aleatórios Utilizados no Experimento 1a

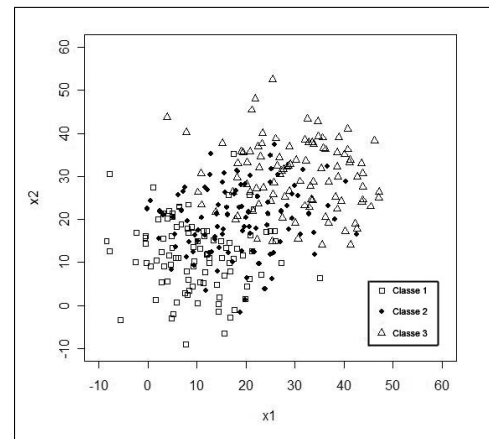


Figura 15: Classes de Pontos Aleatórios Utilizados no Experimento 1b

4.1.1 Obtenção da BR para Classificadores

Para ambos os experimentos, foram utilizadas funções gaussianas (com médias diferentes e $\sigma = 9$). Estas funções foram utilizadas tanto para geração aleatória dos pontos (função “normrnd” do MATLAB) quanto para a partição do universo das variáveis de entrada. As médias para a geração dos pontos de cada classes foram definidas como (10, 30 e 50) para o experimento 1a e (10, 20 e 30) para o experimento 1b. As funções podem ser visualizadas nas figuras 16 e 17.

Como o objetivo deste trabalho não engloba a partição automática do universo, as funções utilizadas para gerar os pontos foram as mesmas usadas nas partições (isto para todos os métodos comparados).

Foram utilizados 1000 pontos artificiais para cada classe, dividindo-os em 5 grupos de 200 pontos cada. Desta forma, procedeu-se com a validação cruzada - *5-fold cross validation* (WEISS; KULIKOWSKI, 1991)- ou seja, os 4 últimos grupos, totalizando 800 pontos, foram utilizados para treinamento dos algoritmos, enquanto que o grupo inicial de 200 pontos, foi utilizado para teste. Em seguida, outro grupo foi utilizado para teste e os demais para treinamento e assim sucessivamente, até que todos os grupos tivessem tido a chance de participar da fase de teste. Os resultados, em termos de desempenho, são apresentados considerando-se as médias obtidas a partir das 5 etapas da validação.

Na fase de treinamento, a base de regras foi obtida por cada método específico (sendo as bases obtidas pelo Probabilístico apresentadas na tabela 5 e as bases dos outros dois métodos, no Apêndice A). Na fase de teste, os valores reais foram comparados com os estimados pelos três sistemas. Como critério de comparação, verificou-se o percentual de acertos em cada método.

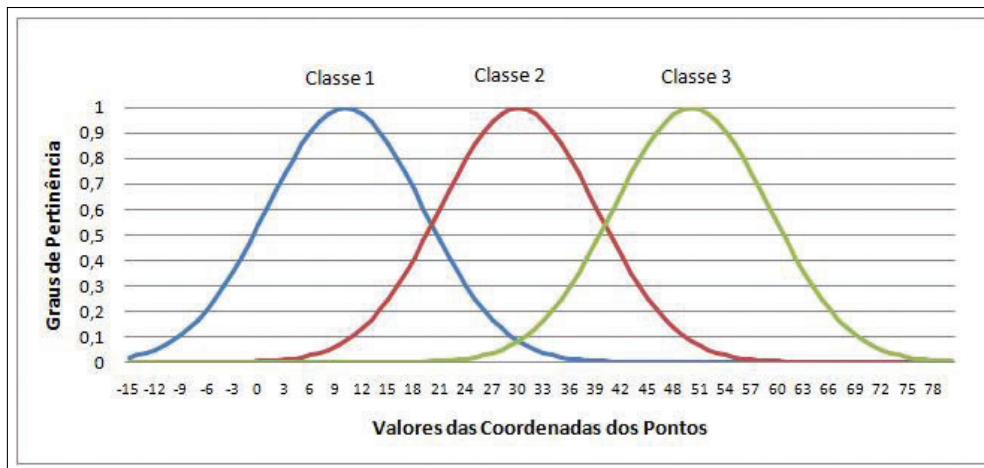


Figura 16: Funções de Pertinência para o Experimento 1a

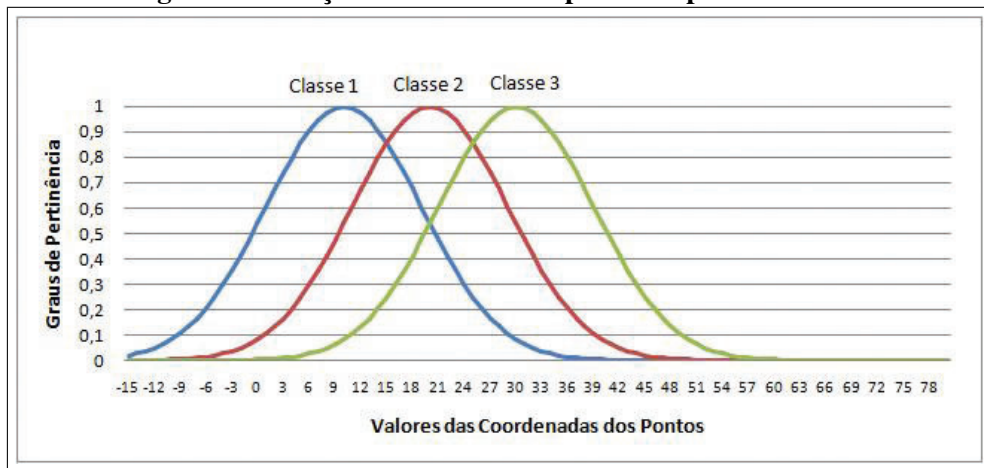


Figura 17: Funções de Pertinência para o Experimento 1b

Após a obtenção da base de regras de cada um dos métodos, tais regras foram implementadas como um classificador *fuzzy* comum, ou seja, utiliza-se o sistema de inferência convencional, sendo que cada regra dirá a que classe os valores de entrada pertencem. No fim, tais saídas são agregadas pelo máximo, ou seja, a classe indicada na saída no sistema será aquela cuja regra obteve o maior nível de ativação. Como não há defuzzificação em classificadores *fuzzy*, a saída obtida é a saída sugerida pelo sistema.

Devido à natureza estocástica do método probabilístico, foi utilizado o processo iterativo proposto por Liu e Li (2005), de forma que fossem computadas diferentes saídas defuzzificadas (num total de n) para uma mesma entrada, com o intuito de produzir uma amostra estatisticamente significativa. O resultado final foi agrupado calculando-se a moda dos resultados obtidos em cada iteração, que também utilizava o máximo como agregador. Utilizou-se a moda por se acreditar que esta seja uma estatística mais representativa dos resultados, uma vez que os valores representam atributos e não valores numéricos. No caso da média por exemplo, se o classificador produzisse na saída amostras do tipo 1, 1, 1, 3, 3 o sistema poderia concluir que a

Tabela 5: Base Obtida pelo Método Probabilístico p/ Experimentos 1a e 1b

Se X1 é	e X2 é	então Classe é	p/ Exp. 1a: com probabilidade	p/ Exp. 1b: com probabilidade
1	1	1	0,9868	0,8579
1	1	2	0,0132	0,1344
1	1	3	0	0,0077
1	2	1	0,5797	0,5427
1	2	2	0,4203	0,4158
1	2	3	0	0,0415
1	3	1	0	0,1675
1	3	2	0,92	0,6056
1	3	3	0,08	0,2269
2	1	1	0,5182	0,5152
2	1	2	0,4818	0,4559
2	1	3	0	0,0289
2	2	1	0,0195	0,2112
2	2	2	0,9468	0,6268
2	2	3	0,0337	0,162
2	3	1	0	0,0397
2	3	2	0,4968	0,3797
2	3	3	0,5032	0,5806
3	1	1	0	0,2999
3	1	2	0,9452	0,5322
3	1	3	0,0548	0,1679
3	2	1	0	0,0303
3	2	2	0,4691	0,4074
3	2	3	0,5309	0,5623
3	3	1	0	0,0083
3	3	2	0,0164	0,1243
3	3	3	0,9836	0,8674

saída final seria 2 sendo que em nenhum momento esta saída foi produzida. Utilizando a moda, o resultado seria apresentado corretamente, ou seja, a classe indicada seria 1.

Com isso, o tempo de processamento desse método, utilizando 5000 iterações, foi de cerca de 5 minutos para cada etapa da validação cruzada, em um PC com processador Core 2 Duo de 1,66 GHz. Também efetuou-se o experimento com 100, 500, 1000, 2500 e 10000 iterações e chegou-se à conclusão de que, para o aumento de cada 100 iterações, obtinha-se um ganho de 0,2% na precisão da estimativa, em média. Porém, o número de 5000 foi escolhido por ser considerado um número com o qual obtém-se uma boa precisão sem onerar de forma demasiada o custo computacional.

4.1.2 Desempenho

Os percentuais de acertos obtidos em cada experimento estão resumidos nas tabelas 6 e 7.

Pode-se utilizar o Erro Padrão apontado nas tabelas para construção dos limites de intervalo de confiança (95%), de forma que, para o experimento 1a, tem-se (81,27% ; 81,60%) para o método WM, (66,10% ; 66,50%) para o método Ishibuchi e (81,77% ; 82,10%) para o método

Tabela 6: Percentual de Acertos - Experimento 1a

Método	Cruz. 1	Cruz. 2	Cruz. 3	Cruz. 4	Cruz. 5	Média	Erro Padrão
Algoritmo WM	81,67%	80,50%	81,17%	81,33%	82,50%	81,43%	0,1672%
Alg. Ishibuchi	70,00%	63,67%	64,33%	67,33%	66,17%	66,30%	0,2030%
Método Prob.	80,67%	81,50%	83,00%	82,50%	82,00%	81,93%	0,1654%

Tabela 7: Percentual de Acertos - Experimento 1b

Método	Cruz. 1	Cruz. 2	Cruz. 3	Cruz. 4	Cruz. 5	Média	Erro Padrão
Algoritmo WM	49,33%	44,83%	47,83%	51,00%	43,00%	47,20%	0,2143%
Alg. Ishibuchi	49,83%	48,17%	48,83%	49,00%	49,00%	48,97%	0,2147%
Método Prob.	56,83%	58,33%	58,50%	56,33%	55,67%	57,13%	0,2127%

WM. Conclui-se, com isso, que a diferença entre o método de WM e o Probabilístico não é significativa, ainda que ambos sejam significativamente superiores ao método Ishibuchi.

Para o segundo experimento, há pouca diferença entre os Erros Padrões dos três métodos, de forma que se fossem construídos intervalos de confiança neste caso, os limites dos mesmos seriam iguais aos valores das médias de cada método. Neste caso, portanto, conclui-se que os três métodos diferem significativamente.

Percebe-se, desta forma, que no experimento 1a, com as classes mais distintas entre si, o método de Ishibuchi apresentou desempenho levemente inferior aos outros dois. Já no experimento 1b, embora todos os resultados tenham piorado, o método probabilístico obteve um desempenho superior aos dos outros dois, sendo que o método de Ishibuchi foi ligeiramente superior ao de WM.

Pelas matrizes de confusão apresentadas na tabela 8, nota-se uma variabilidade maior de precisões (medida pelo Desvio Padrão dos percentuais de acerto por classe) no Método Ishibuchi, ora favorecendo uma das classes, ora outra. A esse respeito, o método probabilístico apresentou menor variabilidade nos dois experimentos, ou seja, sua precisão parece não variar em favorecimento a uma ou outra classe, o que é uma propriedade desejada quando se trabalha com classificadores de maneira geral.

O que se conclui é que o método aqui proposto pouco parece contribuir quando as classes são mais separadas e de fácil definição e, neste caso, o método usual de WM seria o mais indicado. Por outro lado, com a presença de ruído na classificação, o probabilístico parece ser capaz de modelar melhor a incerteza presente e, por conta disso, apresentou resultados melhores.

Tabela 8: Matrizes de Confusão - Experimentos 1a e 1b

Método (1a)	Classe. 1	Classe. 2	Classe 3	Precisão Classe	Precisão Total	Desvio Padrão
WM - Classe 1	710	290	0	71%	81,43%	11,22%
WM - Classe 2	11	933	56	93,3%	-	-
WM - Classe 3	0	200	800	80%	-	-
Ishib. - Classe 1	873	118	9	87,3%	66,3%	21,68%
Ishib. - Classe 2	353	440	207	44%	-	-
Ishib. - Classe 3	35	289	676	67,6%	-	-
Prob. - Classe 1	862	138	0	86,2%	81,93%	7,05%
Prob. - Classe 2	113	738	149	73,8%	-	-
Prob. - Classe 3	0	142	858	85,8%	-	-
Método (1b)	Classe. 1	Classe. 2	Classe 3	Precisão Classe	Precisão Total	Desvio Padrão
WM - Classe 1	623	363	14	62,3%	47,2%	28,35%
WM - Classe 2	236	648	116	64,8%	-	-
WM - Classe 3	20	835	145	14,5%	-	-
Ishib. - Classe 1	966	34	0	96,6%	48,97%	42,32%
Ishib. - Classe 2	646	346	8	34,6%	-	-
Ishib. - Classe 3	172	671	157	15,7%	-	-
Prob. - Classe 1	575	363	62	57,5%	57,13%	12,15%
Prob. - Classe 2	280	448	272	44,8%	-	-
Prob. - Classe 3	41	268	691	69,1%	-	-

4.1.3 Variâncias Diferentes

A fim de avaliar de forma mais ampla o desempenho dos diferentes métodos considerados na aplicação de classificadores *fuzzy*, foram produzidos outros 4 experimentos, nos mesmos moldes dos dois anteriores, alterando apenas as variâncias utilizadas para geração dos dados artificiais. Como, nos primeiros dois experimentos, foi utilizada variância de $\sigma^2 = 9^2$, agora utilizaram-se, para outros dois pares de experimentos, variâncias de $\sigma^2 = 12^2$ e $\sigma^2 = 15^2$. Os pontos gerados para os experimentos 1c a 1f, bem como suas respectivas funções de pertinência, são apresentados no Apêndice B. A metodologia de realização dos experimentos permanece a mesma e suas respectivas matrizes de confusão são apresentadas na tabela 9.

O que se percebe é que, com o aumento da variabilidade, a precisão dos métodos diminui e a diferença entre eles parece se estreitar. Contudo, a variabilidade do método probabilístico continuou mantendo-se baixa, sem onerar de forma demasiada sua precisão. O método de WM apresentou desempenho por vezes próximo do probabilístico, mas sua variabilidade é alta ao se analisarem os seis experimentos conjuntamente. E o método de Ishibuchi, por fim, apresenta baixo desempenho quando as classes estão mais distintas e aparenta uma melhora quando as classes estão mais próximas entre si, se comparando com o desempenho dos outros dois métodos. Todavia, este bom desempenho é viciado pelo favorecimento que o método dá a uma determinada classe, em detrimento das demais, sendo, portanto, o método que apresenta maior variabilidade em sua precisão. Um resumo gráfico desta variabilidade de desempenho pode ser vista na figura 18.

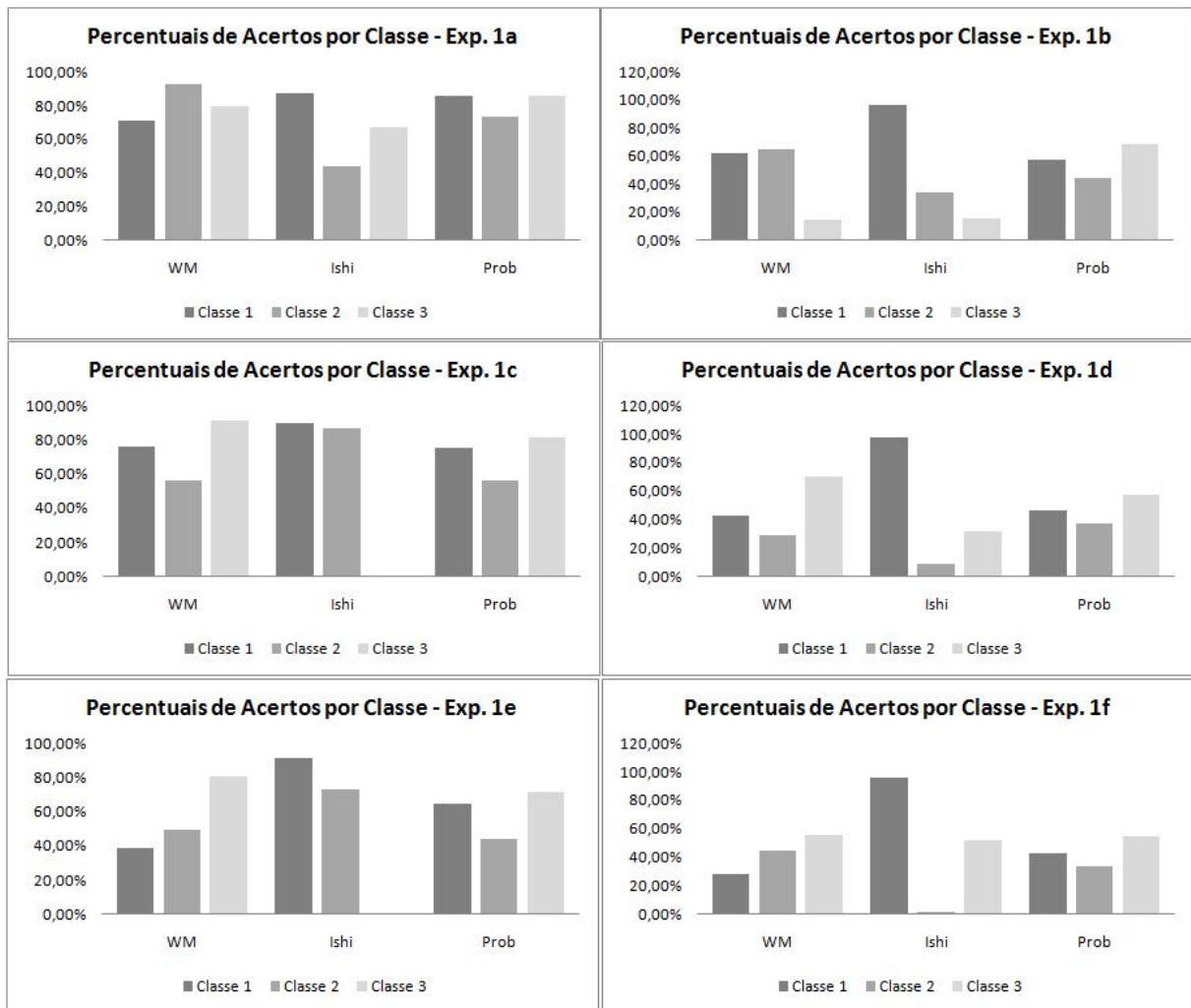


Figura 18: Gráficos de Coluna para os Percentuais de Acertos dos Experimentos de Classificação .

4.2 EXPERIMENTO 2: SISTEMA *FUZZY* PROBABILÍSTICO PARA PREVISÃO DE SÉRIES TEMPORAIS

Outra aplicação considerada, a qual permite testar o método bayesiano de defuzzificação proposto, foi a previsão de duas séries temporais¹. A primeira, Consumo, trata de vendas físicas na região metropolitana de São Paulo, comportando observações mensais de janeiro de 1970 a junho de 1980. A segunda, Banespa, apresenta observações de valores diários de ações do Banespa PN, de 01/01/1995 a 27/12/2000, ainda que apenas parte da série tenha sido considerada para o experimento. Os dados de ambas as séries podem ser vistos no Anexo A.

Neste caso, o Método WM foi aplicado para construção das regras, conforme sugerido por Ribeiro, Goldschmidt e Choren (2009), bem como o método probabilístico. Como o Método de Ishibuchi foi desenvolvido mais especificamente para classificadores, ele não foi considerado nesta aplicação.

A metodologia consiste em particionar a série em conjuntos *fuzzy*, conforme exemplifica a figura 19, extraída de Ribeiro, Goldschmidt e Choren (2009), que utiliza um esquema de partições utilizando 7 conjuntos *fuzzy*. Após a partição do universo, deve-se escolher uma janela para a previsão, sendo que as variáveis de entrada do sistema serão os valores passados que são vistos nessa janela, enquanto que a variável de saída será o valor que se deseja prever.

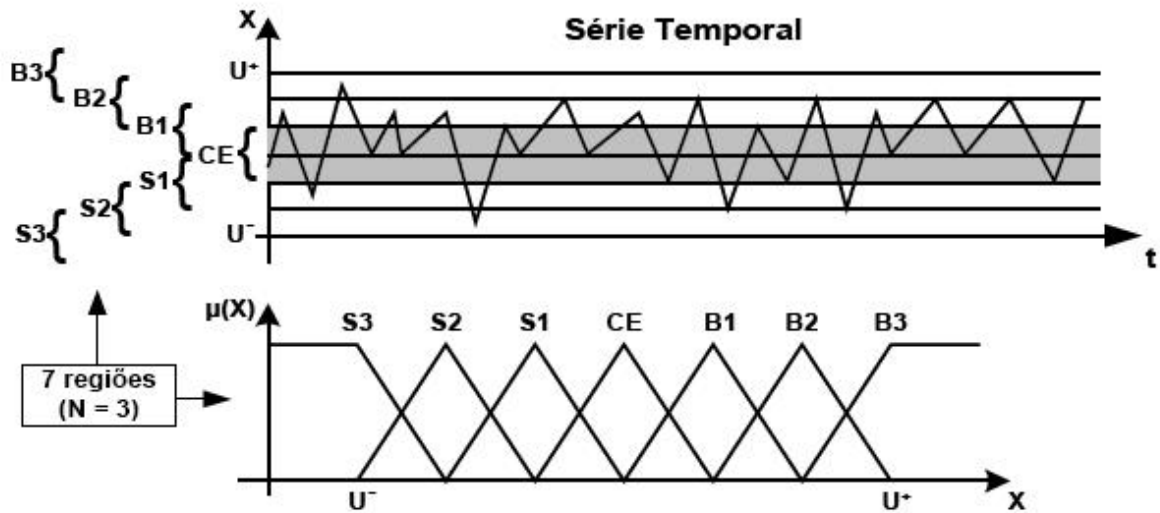


Figura 19: Exemplo de uma série temporal particionada em conjuntos *fuzzy*.

Além do processo de obtenção de regras, os experimentos descritos nesta seção procuram comparar a defuzzificação de um sistema *fuzzy* convencional construído com base no método (WM), aplicando-se o centróide; a defuzzificação pela moda dos centróides no método probabilístico (Prob. Simples); e a defuzzificação bayesiana no método probabilístico (Prob. Bayes). Estes experimentos são descritos em detalhes nas próximas sessões.

4.2.1 Obtenção da BR para Previsão de ST

A primeira série considerada conta com 154 observações, das quais separaram-se 136 para treinamento e 18 para teste. A segunda série trata de valores do Banespa, com 746 observações, das quais utilizaram-se 504 para treinamento e 242 para teste. Em previsões de séries temporais, como os dados são utilizados em sequência, não é usual a utilização da validação cruzada. Assim, os resultados serão mostrados considerando-se apenas a divisão treinamento/teste citada. As séries podem ser vistas nas figuras 20 e 21.

Para a primeira série, o universo de cada variável foi particionado em 4 conjuntos (não-uniformes), definidos pelo especialista através do histograma dos dados (conforme figura 22)

¹Extraídas de <http://www.ime.usp.br/pam/ST.html>.

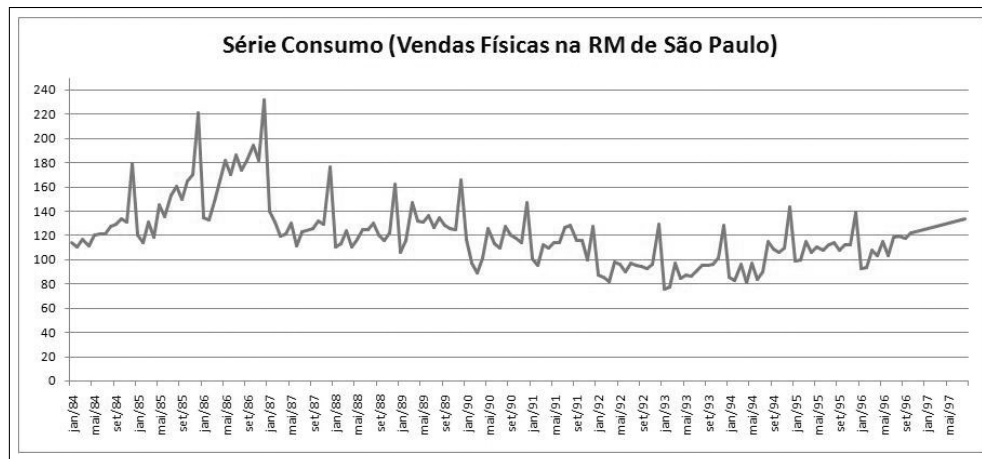


Figura 20: Série Temporal para o Experimento 2a.



Figura 21: Série Temporal para o Experimento 2b.

e após algumas tentativas de se encontrar uma quantidade de conjuntos que fizesse com que o sistema apresentasse resultados razoáveis.

Considerou-se uma janela de 4 observações para se prever a quinta (horizonte de previsão igual a 1), resultando num sistema *fuzzy* com 4 variáveis de entrada e 1 de saída, cada qual particionada em 4 conjuntos *fuzzy*, onde as partições não uniformes são as mesmas para todas as variáveis, sendo médias de 90, 130, 170 e 210 e desvios de 15, 10, 10 e 15, respectivamente.

Na segunda série, utilizaram-se 7 conjuntos (partição uniforme) para cada variável do sistema e também uma janela de 4 observações para previsão. Foram utilizadas funções de pertinência gaussianas com média e desvio padrão definidos pelo especialista, sendo as médias iguais a 25, 25, 45, 55, 65, 75 e 85 e todos os desvios iguais a 3.

As partições do universo das variáveis são mostradas nas figuras 23 e 24.

As metodologias para obtenção da base de regras através dos métodos WM e probabilísticos permanecem as mesmas do caso de classificação, diferindo apenas na saída do sistema que, ao

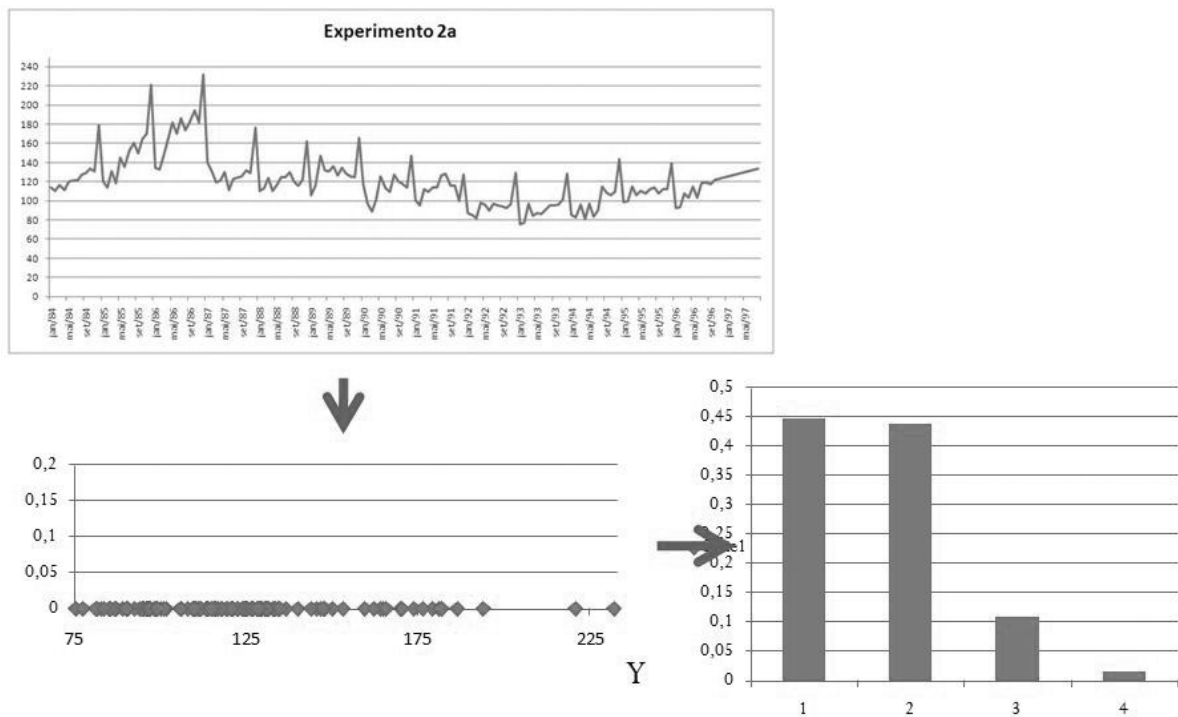


Figura 22: Esquema de Particionamento para a Série Consumo.

invés de apresentar uma classe, infere um conjunto *fuzzy*.

As bases de regras para ambos os experimentos estão nas tabelas 10 e 11.

4.2.2 Desempenho

Os resultados obtidos nos experimentos podem ser vistos na tabela 13.

As séries previstas e seus comparativos com a série real estão apresentados nas figuras 25 e 26.

Para ambas as séries, o que parece ocorrer é que o método probabilístico saiu-se melhor do que o WM, mas o bayesiano conseguiu melhorar ainda mais os resultados do primeiro, no que diz respeito às previsões pontuais.

Contudo, devido à baixa cardinalidade dos dados de teste, é difícil dizer qual dos três métodos melhor se ajustou aos dados reais, no que diz respeito à primeira série. Na segunda, entretanto, os três parecem ter se ajustado bem, de forma que, aparentemente, o bayesiano apresentou menor erro.

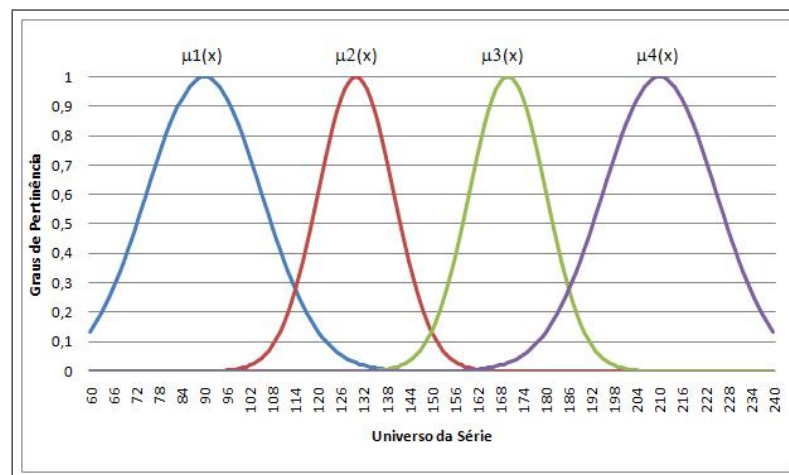


Figura 23: Funções de Pertinência para o Experimento 2a

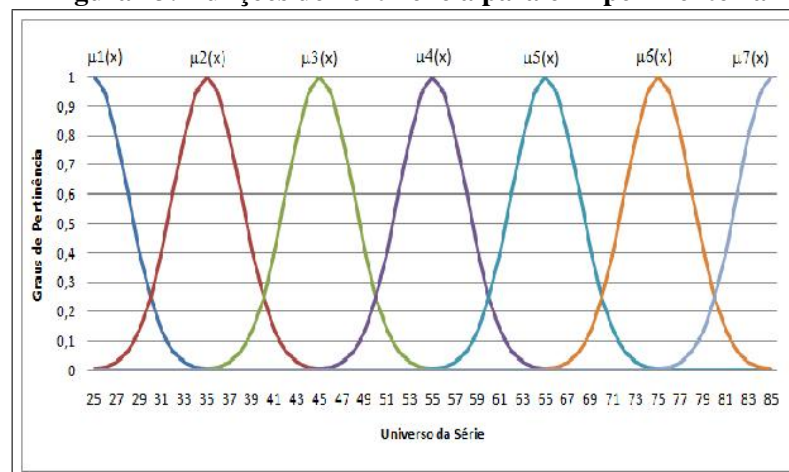


Figura 24: Funções de Pertinência para o Experimento 2b

Tabela 9: Matrizes de Confusão - Experimentos 1c a 1f

Método (1c)	Classe. 1	Classe. 2	Classe 3	Precisão Classe	Precisão Total	Desvio Padrão
WM - Classe 1	760	239	1	76%	74,53%	17,65%
WM - Classe 2	161	562	277	56,2%	-	-
WM - Classe 3	3	83	914	91,4%	-	-
Ishib. - Classe 1	899	101	0	89,9%	59,07%	51,17%
Ishib. - Classe 2	127	873	0	87,3%	-	-
Ishib. - Classe 3	0	1000	0	0%	-	-
Prob. - Classe 1	756	237	7	75,6%	71,4%	13,02%
Prob. - Classe 2	229	568	203	56,8%	-	-
Prob. - Classe 3	6	176	818	81,8%	-	-
Método (1d)	Classe. 1	Classe. 2	Classe 3	Precisão Classe	Precisão Total	Desvio Padrão
WM - Classe 1	430	366	204	43%	47,7%	21,15%
WM - Classe 2	303	293	404	29,3%	-	-
WM - Classe 3	99	193	708	70,8%	-	-
Ishib. - Classe 1	981	15	4	98,1%	46,5%	46,09%
Ishib. - Classe 2	830	94	76	9,4%	-	-
Ishib. - Classe 3	466	214	320	32%	-	-
Prob. - Classe 1	471	354	175	47,1%	47,63%	10,01%
Prob. - Classe 2	267	379	354	37,9%	-	-
Prob. - Classe 3	102	319	579	57,9%	-	-
Método (1e)	Classe. 1	Classe. 2	Classe 3	Precisão Classe	Precisão Total	Desvio Padrão
WM - Classe 1	391	561	48	39,1%	56,53%	21,96%
WM - Classe 2	183	493	324	49,3%	-	-
WM - Classe 3	10	178	812	81,2%	-	-
Ishib. - Classe 1	913	87	0	91,3%	54,8%	48,32%
Ishib. - Classe 2	269	731	0	73,1%	-	-
Ishib. - Classe 3	7	993	0	0%	-	-
Prob. - Classe 1	649	321	30	64,9%	60,37%	14,16%
Prob. - Classe 2	258	445	297	44,5%	-	-
Prob. - Classe 3	26	257	717	71,7%	-	-
Método (1f)	Classe. 1	Classe. 2	Classe 3	Precisão Classe	Precisão Total	Desvio Padrão
WM - Classe 1	285	577	138	28,5%	43%	13,73%
WM - Classe 2	253	447	300	44,7%	-	-
WM - Classe 3	181	261	558	55,8%	-	-
Ishib. - Classe 1	960	14	26	96%	50,17%	47,1%
Ishib. - Classe 2	792	19	189	1,9%	-	-
Ishib. - Classe 3	436	38	526	52,6%	-	-
Prob. - Classe 1	431	358	211	43,1%	44,13%	10,39%
Prob. - Classe 2	295	343	362	34,3%	-	-
Prob. - Classe 3	149	301	550	55%	-	-

Tabela 10: Base Obtida pelo Método Probabilístico para a Série Consumo

Se X1 é	e X2 é	e X3 é	e X4 é	então Y é	com probabilidade
1	1	1	1	1	0,8135
1	1	1	1	2	0,1865
1	1	1	2	1	1
1	1	2	1	1	0,8626
1	1	2	1	2	0,1374
1	1	2	2	2	1
1	2	1	1	1	0,8422
1	2	1	1	2	0,1578
1	2	1	2	2	1
1	2	2	2	1	0,1826
1	2	2	2	2	0,8174
2	1	1	1	1	0,8209
2	1	1	1	2	0,1791
2	1	1	2	1	0,5112
2	1	1	2	2	0,4888
2	1	2	1	1	0,7281
2	1	2	1	2	0,2719
2	1	2	2	2	1
2	2	1	2	1	0,6583
2	2	1	2	2	0,3417
2	2	2	1	2	1
2	2	2	2	1	0,0547
2	2	2	2	2	0,7089
2	2	2	2	3	0,2363
2	2	2	3	1	0,4227
2	2	2	3	2	0,4478
2	2	2	3	3	0,1295
2	2	3	1	1	0,5296
2	2	3	1	2	0,4704
2	2	3	2	1	0,3765
2	2	3	2	2	0,6235
2	2	3	3	3	1
2	3	1	1	2	1
2	3	1	2	2	1
2	3	2	1	1	1
2	3	2	2	2	1
2	3	3	3	3	0,3997
2	3	3	3	4	0,6003
3	1	1	2	1	1
3	1	2	2	2	1
3	2	1	1	1	1
3	2	2	2	2	1
3	3	3	3	3	0,5
3	3	3	3	4	0,5
3	3	3	4	2	0,3132
3	3	3	4	3	0,6868
3	3	4	2	2	1
3	3	4	3	3	0,3765
3	3	4	3	4	0,6235
3	4	2	2	2	1
3	4	3	3	4	1
3	4	3	4	2	1
4	2	2	2	2	0,6259
4	2	2	2	3	0,3741
4	3	3	4	3	1
4	3	4	2	2	1

Tabela 11: Base Obtida pelo Método Probabilístico para a Série Banespa

Se X1 é	e X2 é	e X3 é	e X4 é	então Y é	com probabilidade
1	1	1	1	1	0,5
1	1	1	1	2	0,5
1	1	1	2	3	1
1	1	2	3	2	1
1	2	3	2	2	1
1	2	3	3	3	1
2	1	1	1	1	1
2	1	2	3	3	1
2	2	1	1	1	1
2	2	2	1	1	1
2	2	2	2	1	0,0167
2	2	2	2	2	0,9253
2	2	2	2	3	0,058
2	2	2	3	3	1
2	2	3	2	2	1
2	2	3	3	3	1
2	3	2	2	2	0,273
2	3	2	2	3	0,727
2	3	3	3	3	1
3	2	1	2	3	1
3	2	2	2	2	1
3	2	2	3	2	0,5814
3	2	2	3	3	0,4186
3	2	3	2	2	1
3	2	3	3	3	1
3	3	2	1	2	1
3	3	2	2	2	1
3	3	2	3	2	0,5091
3	3	2	3	3	0,4909
3	3	3	2	2	0,5525
3	3	3	2	3	0,4475
3	3	3	3	2	0,0227
3	3	3	3	3	0,8952
3	3	3	3	4	0,0821
3	3	3	4	3	0,0672
3	3	3	4	4	0,9328
3	3	4	3	3	1
3	3	4	4	3	0,0707
3	3	4	4	4	0,9293
3	4	3	3	4	1
3	4	4	3	3	1
3	4	4	4	4	1
4	3	3	2	1	1
4	3	3	3	3	1
4	3	3	4	4	1
4	3	4	4	4	1
4	4	3	3	2	0,1381
4	4	3	3	3	0,8619
4	4	3	4	4	1
4	4	4	3	3	0,7702
4	4	4	3	4	0,2298
4	4	4	4	3	0,0666
4	4	4	4	4	0,858
4	4	4	4	5	0,0754
4	4	4	5	4	0,1134
4	4	4	5	5	0,8866
4	4	5	4	4	1

Tabela 12: Base Obtida pelo Método Probabilístico para a Série Banespa (Cont.)

Se X1 é	e X2 é	e X3 é	e X4 é	então Y é	com probabilidade
4	4	5	5	4	0,1479
4	4	5	5	5	0,7939
4	4	5	5	6	0,0582
4	5	4	4	4	0,7948
4	5	4	4	5	0,2052
4	5	5	4	4	0,4147
4	5	5	4	5	0,5853
4	5	5	5	4	0,0577
4	5	5	5	5	0,9423
4	5	5	6	6	1
5	4	4	4	4	0,819
5	4	4	4	5	0,181
5	4	4	5	4	0,5622
5	4	4	5	5	0,4378
5	4	5	4	4	1
5	4	5	5	5	1
5	5	4	4	4	0,8696
5	5	4	4	5	0,1304
5	5	4	5	4	0,3177
5	5	4	5	5	0,6823
5	5	5	4	4	0,8101
5	5	5	4	5	0,1899
5	5	5	5	4	0,1085
5	5	5	5	5	0,8849
5	5	5	5	6	0,0066
5	5	5	6	6	1
5	5	6	6	6	1
5	6	6	6	5	0,0696
5	6	6	6	6	0,9304
6	5	5	5	5	1
6	5	5	6	6	1
6	5	6	6	6	1
6	6	5	5	5	0,4444
6	6	5	5	6	0,5556
6	6	5	6	6	1
6	6	6	5	5	0,4638
6	6	6	5	6	0,5362
6	6	6	6	5	0,057
6	6	6	6	6	0,9147
6	6	6	6	7	0,0284
6	6	6	7	6	0,4057
6	6	6	7	7	0,5943
6	6	7	6	6	1
6	6	7	7	6	1
6	7	6	6	6	1
6	7	7	6	6	1
7	6	6	6	6	1
7	7	6	6	6	1

Tabela 13: Valores Estimados para Previsão das Séries Temporais

Série	EQM WM	EQM Prob. Simples	EQM Prob. Bayes.
Consumo	205,45	184,39	170,37
Banespa	5,24	4,79	4,71

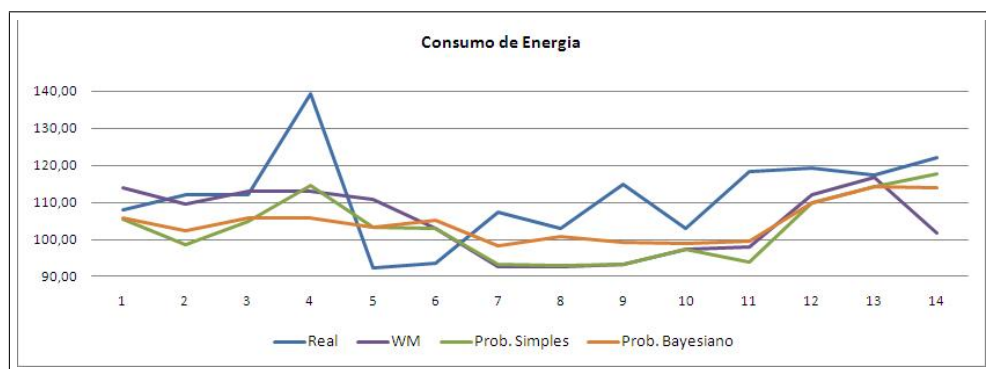


Figura 25: Resultados para a Série Consumo, 14 passos à frente.

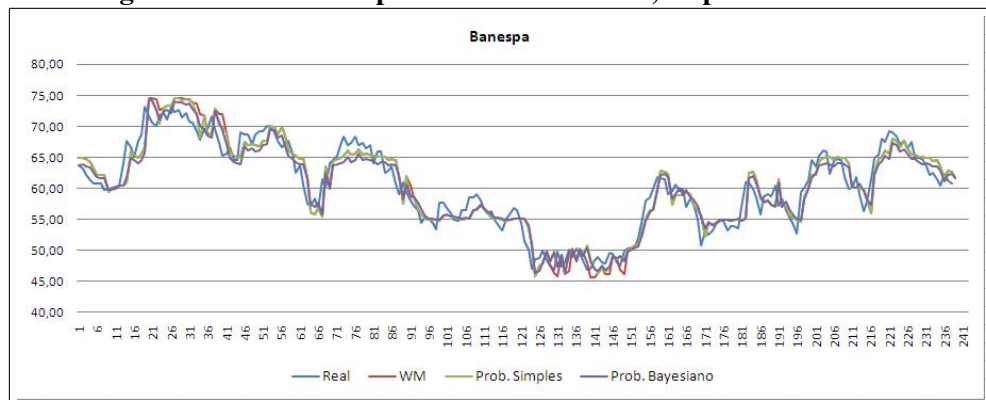


Figura 26: Resultados para a Série Banespa, 238 passos à frente.

5 CONCLUSÕES

5.1 DISCUSSÃO DOS RESULTADOS OBTIDOS

Esta dissertação teve como objetivo principal propor a união de duas técnicas tradicionais para o tratamento da incerteza, de forma a prover Sistemas *Fuzzy* Probabilísticos com geração automática de regras e defuzzificação através de inferência bayesiana. Para isso, providenciou-se uma breve revisão da literatura no que tange aos diferentes tipos de sistemas *fuzzy*, os métodos de geração de regras, métodos de defuzzificação existentes e inferência bayesiana.

No que diz respeito aos métodos mais clássicos para geração de base de regras *fuzzy*, consideraram-se dois para fins de comparação. O primeiro é o Método de Wang Mendel que, por descartar informações redundantes e inconsistentes pode causar certa perda de informação, originando algum tipo de viés no sistema de inferência, dependendo do problema a ser tratado. O segundo método é o algoritmo de Ishibuchi que, apesar de utilizar graus de confiança nas regras, apenas reduz a perda de informações redundantes e inconsistentes.

Com o intuito de contornar esse problema, foi proposto um novo método para geração de regras, que ao invés de descartar regras conflitantes, calcula a probabilidade de ocorrência de cada uma. Tal base de regras é então considerada em um Sistema de Inferência *Fuzzy* Probabilístico.

A fim de se testar a eficácia do novo método, esse foi comparado com o algoritmo de WM clássico e com o método de Ishibuchi em um problema de classificação gerado artificialmente, considerando-se as validações cruzadas. O que se pôde concluir foi que, para as instâncias consideradas, o método probabilístico não apresentou diferença significativa quando comparado ao método WM, nos casos onde as classes estavam mais separadas (menor grau de confusão), mas ambos foram superiores ao método Ishibuchi. Entretanto, o método proposto apresentou-se superior aos outros dois quando as classes estavam mais misturadas (maior grau de confusão).

Também se pôde observar que, em experimentos com maior variabilidade nos dados, todos os métodos apresentaram uma queda no desempenho. Contudo, diferente dos demais, a precisão

do método probabilístico não apresentou grandes distorções, ou seja, o método manteve-se equilibrado, não favorecendo uma ou outra classe em detrimento das demais.

Com isso, conclui-se que, no que diz respeito à geração de uma base de regras quando há maior nível de ruído na informação, o não descarte da informação obtida no conjunto de dados pode representar um ganho.

Em seguida, consideraram-se sistemas *fuzzy* probabilísticos de forma mais generalizada do que em classificadores *fuzzy*. Neste tipo de sistema, existe a necessidade da fase de defuzzificação e propôs-se que fosse utilizada inferência bayesiana, com o intuito de aprimorar os resultados obtidos na saída do sistema.

Para se testar a proposta, considerou-se um sistema de previsão de séries temporais baseado em sistemas de inferência *fuzzy*. Neste caso, a defuzzificação bayesiana utilizada em um SFP foi comparada com a defuzzificação obtida, também em um SFP, calculando-se as modas dos valores dos centróides resultantes de n realizações, bem como com a defuzzificação pelo método do centróide em um SIF tradicional, cuja base de regras foi obtida pelo método WM. Os resultados apontam um desempenho razoável do SIF tradicional e do SFP simples utilizando a moda, enquanto que o SFP com defuzzificação bayesiana apresentou o menor erro.

Interessante notar que a informação *a priori* dos dados de treinamento influencia os resultados na saída do sistema de teste, produzindo saídas “suavizadas” pela média e variância da priori.

Pode-se concluir, desta forma, que as propostas elaboradas apresentaram um bom desempenho para as instâncias analisadas, ainda que com um custo computacional elevado. Além disso, pode-se perceber que, no que diz respeito à metodologia bayesiana, a *priori* pode exercer forte influência nos resultados finais.

5.2 CONTRIBUIÇÕES E PUBLICAÇÕES

As maiores contribuições do trabalho são as duas propostas inovadoras: A primeira, de desenvolver um método simples para geração de regras *fuzzy* em sistemas probabilísticos, e a segunda, a de procurar uma melhor sofisticação na defuzzificação neste tipo de sistema, através do uso da inferência bayesiana.

No decorrer do trabalho, dois artigos foram publicados, a saber:

- Defuzzificação Probabilística em Conjuntos *Fuzzy* do Tipo-2. Simpósio Nacional de Probabilidade e Estatística. São Pedro/SP. 26/07 a 30/07/2010, pp, 1-6.

- Método Probabilístico para Geração de Regras *Fuzzy*. Congresso Brasileiro de Sistemas *Fuzzy* - CBSF-2010. Sorocaba, 09/11 a 12/11/2010, pp, 425 - 432.

5.3 SUGESTÕES PARA TRABALHOS FUTUROS

No que tange às propostas de geração de base de regras, espera-se em pesquisas futuras que

- o método probabilístico seja confrontado com outras modificações dos algoritmos de WM e Ishibuchi;
- outros métodos de agregação das saídas sejam avaliados;
- os estudos de caso para outros problemas de classificação sejam ampliados;

Quanto à defuzzificação bayesiana, considera-se que são necessários outros testes de aplicação desta técnica, com diferentes *prioris*, diferentes funções de pertinência, diferentes números de iterações, etc.

Espera-se, ainda, que tais técnicas sejam adaptadas para serem aplicadas em outros tipos de sistemas *fuzzy*, como por exemplo,

- sistemas do tipo-2;
- sistemas não-estacionários.

REFERÊNCIAS

- BOIXADER, D.; JACAS, J.; RECASENS, J. Similarity-based approach to defuzzification. **6th IEEE International Conference on Fuzzy Systems**, v. 2, p. 761–765, 1997.
- BRUBAKER, D. I. Two new defuzzification techniques. **WESCON Conference Record**, p. 455–459, 1993.
- CHEN, M.-Y.; LINKENS, D. Rule-base self-generation and simplification for data-driven fuzzy models. **Fuzzy Sets and Systems**, v. 142, p. 243–265, 2004.
- COLETTIA, G.; SCOZZAFAVAB, R. Conditional probability and fuzzy information. **Computational Statistics & Data Analysis** **51**, p. 115–132, 2006.
- COPLAND, S.; JOHN, R. An investigation into alternative methods for the defuzzification of an interval type-2 fuzzy set. **IEEE Intl. Conf. on Fuzzy Systems**, Vancouver, Canada, 2006.
- CZOGALA, E.; LESKI, J. An equivalence of approximate reasoning under defuzzification. **Technical University of Silesia Akademicka**, v. 16, p. 44–101, 2000.
- DENG, H.; LAU, M.; MILLAR, K. Defuzzification in fuzzy multicriteria analysis. **19th International Conference of the NA Fuzzy Information Processing Society**, p. 222–226, 2000.
- DENG, H.; YEH, C.-H. Simulation-based evaluation of defuzzification-based approaches to fuzzy multiattribute decision making. **IEEE Transactions on Fuzzy Systems**, v. 36, p. 968–977, 2006.
- DIAS, C. M. C. **Lógica matemática: introdução ao cálculo proposicional**. [S.l.]: 2a Ed, 2010.
- EHLERS, R. S. **Introdução à Inferência Bayesiana**. [S.l.]: Universidade Federal do Paraná, 2003.
- EISELE, M.; HENTSCHEL, K.; KUNEMUD, T. Hardware realization of fast defuzzification by adaptive integration. **Microelectronics for Neural Networks and Fuzzy Systems**, p. 318–323, 1994.
- GARIBALDI, J.; OZEN, T. Nondeterministic fuzzy reasoning . **In prep. for IEEE Transactions on Fuzzy Systems**, 2004.
- GARIBALDI, J. M.; MUSIKASUWAN, S.; OZEN, T. The association between non-stationary and interval type-2 fuzzy sets: A case study. **FUZZ-IEEE**, p. 224–229, 2005.
- GOMIDE, F. A. C.; GUDWIN, R. R. Modelagem, controle, sistemas e lógica fuzzy. **Departamento de Engenharia de Computação e Automação Industrial (DCA)**, 1997.
- GREENFIELD, S.; JOHN, R.; COUPLAND, S. A novel sampling method for type-2 defuzzification. **UKCI** **06**, p. 120–127, 2005.

- HELLENDORRN, H.; THOMAS, C. On defuzzification in fuzzy controllers. **J. Intell. and Fuzzy Systems** 2, p. 109–123, 1993.
- HERRERA, F. Genetic fuzzy systems: Taxonomy, current research trends and prospects. **Evolutionary Intelligence**, v. 1, p. 27–46, 2008.
- HIROTA, K. Concepts of probabilistic set. **Fuzzy sets and systems**, v. 5, p. 31–46, 1981.
- ISHIBUCHI, H.; NAKASHIMA, T. Effect of rule weights in fuzzy rule-based classification systems. **IEEE Trans. on Fuzzy Systems**, v. 9, p. 506–515, 2001.
- IVANCIC, F.; MALAVIYA, A.; PETERS, L. An automatic rule base generation method for fuzzy pattern recognition with multi-phased clustering. **Second International Conference on Intelligent Electronic Systems**, v. 3, p. 66–75, 1999.
- JAGER, R.; VERBRUGGEN, H.; BRUJIN, P. The role of defuzzification methods in the application of fuzzy control. **Preprints IFAC Symp. on Intelligent Components and Instrumentation for Control Applications**, p. 111–116, 1992.
- JANG, J.-S.; SUN, C.-T.; MIZUTANI, E. **Neuro-Fuzzy and Soft Computing**. [S.l.]: Prentice Hall, 1997.
- JOHANYÁK, Z. C.; KOVÁCS, S. Sparse fuzzy system generation by rule base extension. **IEEE International Conference on Fuzzy Systems**, p. 99–104, 2007.
- KANDEL, A.; FRIEDMAN, M. Defuzzification using most typical values. **IEEE Transactions on Fuzzy Systems**, v. 28, p. 901–906, 1998.
- KARNIK, N. K.; MENDEL, J. M.; LIANG, Q. Type-2 fuzzy logic systems. **International Journal on Fuzzy Systems**, v. 7, p. 643–658, 1999.
- KOVACIC, Z.; BOGDAN, S. **Fuzzy Controller Design: Theory and Applications**. [S.l.]: CRC Press, 2006.
- KWAKERNAAK, H. Fuzzy random variables. **Inf. Sci.** 17, v. 3, p. 253–278, 1978.
- LAVIOLETTE, M.; JR, J. W. S. The efficacy of fuzzy representations of uncertainty. **IEEE Transactions on Fuzzy Systems**, v. 2, 1994.
- LEEKWIJCK, W. V.; KERRE, E. E. Defuzzification: criteria and classification. **Fuzzy Sets and Systems**, v. 108, p. 159–178, 1999.
- LIN, C.; LEE, C. S. G. **Neural Fuzzy Systems**. [S.l.]: Prentice Hall, 1996.
- LINDA, O.; MANIC, M. Importance sampling based defuzzification for general type-2 fuzzy sets. **Não Publicado**, 2010.
- LIU, Z.; LI, H.-X. A probabilistic fuzzy logic system for modeling and control. **IEEE Transactions on Fuzzy Systems**, v. 13, p. 848–859, 2005.
- LUCAS, L. A. **Sistema de Inferência Fuzzy Geral do Tipo-2 Aplicado à Classificação**. Tese (Doutorado) — Universidade Tecnológica Federal do Paraná, 2009.

- MADAU, D.; FELDKAMP, L. Influence value defuzzification method. **5th IEEE International Conference on Fuzzy Systems**, v. 3, p. 1819–1824, 1996.
- MEGHADADI, A. H.; AKBARZADEH-T, M.-R. Probabilistic fuzzy logic and probabilistic fuzzy systems. **10th IEEE International Conference on Fuzzy Systems**, v. 3, p. 1127–1130, 2001.
- MEYER, P. **Probabilidade: Aplicações à Estatística**. [S.l.]: LTC, 1983.
- MORSI, N. N.; FAHM, A. A. On generalized modus ponens with multiple rules and a residuated implication. **Fuzzy Sets and Systems**, v. 129, p. 267–274, 2002.
- PAULINO, C. D.; TURKMAN, M. A. A.; MURTEIRA, B. **Estatística Bayesiana**. [S.l.]: Serviço de Educação e Bolsas - Fundação Calouste Gulbenkian, 2005.
- PEDRYCS, W. **An Introduction to Fuzzy Sets: Analysis and Design**. [S.l.]: MIT Press, 1998.
- PEDRYCZ, W.; GOMIDE, F. **Fuzzy Systems Engineering : Toward Human-Centric Computing**. [S.l.]: IEEE Press, 2007.
- POLESHUK, O. M.; KOMAROV, E. G. New defuzzification method based on weighted intervals. **NAFIPS**, p. 1–3, 2008.
- RIBEIRO, C. V.; GOLDSCHMIDT, R. R.; CHOREN, R. Métodos para previsão de séries temporais e suas tendências de desenvolvimento. **Monografias em Sistemas e Computação**, v. 03, p. 01–29, 2009.
- ROJAS, I. et al. Self-organized fuzzy system generation from training examples. **IEEE Transaction on Fuzzy Systems**, v. 8, p. 23–36, 2000.
- RUNKLER, T. Extended defuzzification methods and their proprieties. **5th IEEE International Conference on Fuzzy Systems**, v. 1, p. 694–700, 1996.
- RUNKLER, T. A.; GLESNER, M. Defuzzification as crisp decision making under fuzzy constraints. **Workshop Fuzzy Systems, Braunschweig**, 1993.
- RUNKLER, T. A.; GLESNER, M. Defuzzification with improved static and dynamic behavior: extended center of area. **Proc. European Cong. on Fuzzy and Intelligent Technologies**, p. 845–851, 1993.
- RUNKLER, T. A.; GLESNER, M.; GENTHER, H. Defuzzification based on fuzzy clustering. **IEEE International Conference on Fuzzy Systems**, p. 1645–1648, 1994.
- SAADE, J. A unifying approach to defuzzification and comparison of the outputs of fuzzy controllers. **IEEE Transactions on Fuzzy Systems**, v. 4, p. 227–237, 1996.
- SAADE, J.; DIAB, H. Defuzzification techniques for fuzzy controllers. **IEEE Transactions on Fuzzy Systems**, v. 30, p. 223–229, 2000.
- SETNES, M. et al. Similarity measures in fuzzy rule base simplification. **IEEE Transaction on Systems, Man and Cybernetics**, v. 28, p. 376–386, 1998.
- SHI, Y.; SEN, P. A new defuzzification method for fuzzy control of power converters. **Industry Applications Conference**, v. 2, p. 1202–1209, 2000.

- STEELS, L. When are robots intelligent autonomous agents? **Journal of Robotics and Autonomous Systems**, v. 15, p. 3–9, 1995.
- TAYLOR, M. S.; BOSWELL, S. B. A central limit theorem for fuzzy random variables. **Fuzzy Sets and Systems**, v. 24, p. 331–344, 1987.
- THIELE, H. **Towards Axiomatic Foundations for Defuzzification Theory**. [S.l.], 1998.
- TIBIRIÇÁ, C. A. G.; NASSAR, S. M. Desenvolvimento de uma abordagem híbrida difuso-probabilística para a modelagem de incerteza. **Universidade Federal de Santa Catarina**, 2003.
- TUMA, C. C. M. **Aprendizado de Máquina Baseado em Separabilidade Linear em Sistema de Classificação Híbrido-Nebuloso Aplicado a Problemas Multiclasse**. Dissertação (Mestrado) — UFSCAR, 2009.
- WANG, L. X.; MENDEL, J. M. Generating fuzzy rules by learning from examples. **IEEE Trans. Systems Man and Cybernetics**, v. 22, p. 1414–1427, 1992.
- WATANABE, H.; DETTLOFF, W. D.; YOUNT, K. A vlsi fuzzy logic con troller with reconfigurable, cascable architecture. **IEEE Journal of Solid-state Circuits**, v. 2, p. 376–382, 1990.
- WEISS, S. M.; KULIKOWSKI, C. **Computer Systems That Learn**. [S.l.]: Morgam Kaufmann, 1991.
- WILSON, N. Vagueness and bayesian probability. **IEEE Transactions on Fuzzy Systems**, v. 2, p. 34–36, 1994.
- YAGER, R. On the use of combinability functions for intelligent defuzzification. **4th IEEE International Conference on Fuzzy Systems**, v. 1, p. 193–198, 1995.
- YAGER, R. R. Fuzzy models induced by alternative defuzzification methods. **IEEE Transactions on Fuzzy Systems**, v. 1, p. 457–462, 1996.
- YAGER, R. R.; FILEV, D. P. A generalized defuzzification method via badd distributions. **International Journal of Intelligent Systems**, v. 6, p. 687–697, 1991.
- YAGER, R. R.; FILEV, D. P. An adaptive approach to defuzzification based on level sets. **Fuzzy Sets and Systems**, v. 53, p. 355–360, 1993.
- YAGER, R. R.; FILEV, D. P. Slide: A simple adaptive defuzzification method. **IEEE Transactions on Fuzzy Systems**, v. 1, p. 69–78, 1993.
- ZADEH, L. The concept of a linguistic variable and its application to approximate reasoning. **Information Sciences**, v. 8, p. 199–249, 1975.
- ZADEH, L. A. Fuzzy sets. **Inform. and Control** 8, p. 338–353, 1965.

ANEXO A – DADOS DAS SÉRIES TEMPORAIS

A.1 SÉRIE TEMPORAL CONSUMO

Tabela 14: Dados de Treinamento

114,13	110,79	116,46	111,57	120,66	121,15	121,27	127,02	129,04	133,3
130,6	179,39	120,64	114,05	130,6	118,26	145,54	135,13	153,35	159,95
150,01	164,93	170,37	220,96	134,26	133,11	147,84	164,46	181,86	170,44
186,64	174,21	181,62	194,16	181,9	232,01	140,16	130,78	119,04	120,73
129,81	111,04	122,75	133,95	125,41	132,05	129,54	176,37	110,09	113,25
124,03	110,63	116,72	124,63	124,38	130,27	119,87	115,75	122,44	162,43
105,89	115,59	147	131,7	131,32	136,66	126,43	134,88	128,26	125,32
124,61	166,11	116,25	96,93	89,27	101,87	125,57	113,31	109,39	127,33
120,56	117,73	113,81	147,25	100,15	95,11	112,26	109,39	114,2	113,8
126,47	128,36	115,71	116,09	99,53	127,27	87,08	85,67	82,02	98,2
96,44	90,23	97,15	95,08	94	93	96,09	129,21	75,39	77,7
97,34	84,97	87,55	86,64	90,52	95,4	95,2	95,8	101,23	128,49
85,63	82,77	96,55	81,33	96,91	83,76	90,19	114,84	108,4	106,05
109,71	143,86	99,12	99,28	114,75	106,13				

Tabela 15: Dados de Teste

110,02	108,07	112,52	113,87	107,84	112,12	112,03	139,37	92,24	93,56
107,37	102,89	114,78	102,88	118,41	119,23	117,36	122,06		

A.2 SÉRIE TEMPORAL BANESPA

Tabela 16: Dados de Treinamento

42,26	44,36	46,66	44,28	43,28	42,61	43,82	43,28	41,25	41,92
43,61	41,92	41,25	41,99	40,23	41,24	39,22	40,58	39,89	38,88
40,56	38,27	38,54	38,54	38,54	39,22	38,54	38,75	38,54	38,88
38,61	38,81	38,47	38,2	39,96	42,53	42,26	42,6	43,28	43,61
43,28	44,62	45,91	45,97	45,95	45,03	47,32	47,67	47,94	47,94
57,1	57,01	57,1	57,68	57,78	60,2	67,26	71,62	71,52	73,56
72,59	72,59	70,17	72,58	72,59	74,49	75,49	77,91	82,27	79,85

77,9	75,97	75,01	74,14	71,62	67,75	71,62	74,04	73,56	74,52
74,52	72,1	69,68	73,56	71,62	70,17	70,55	69,68	68,72	64,84
64,75	63,88	64,84	62,91	60,01	54,68	59,52	59,04	58,07	56,13
60,01	58,07	58,54	60,01	60,01	57,59	55,17	55,65	51,78	51,3
55,17	55,16	54,39	52,46	53,23	53,42	50,81	49,36	50,33	51,79
53,24	54,68	57,88	57,11	62,91	62,43	62,43	64,84	67,22	67,75
67,26	68,81	67,75	66,78	65,81	63,88	62,62	62,13	60,98	59,52
58,56	60,49	59,52	59,52	56,62	55,65	56,13	56,62	58,07	55,17
53,23	53,71	53,21	52,75	52,25	52,26	49,36	45,49	45,28	46,46
44,52	41,62	40,65	38,71	39,87	35,81	30,97	28,07	29,23	29,51
25,16	29,03	34,36	40,17	39,68	39,67	42,57	39,1	38,23	41,33
42,28	40,65	40,65	40,62	38,71	34,41	33,87	30,58	32,91	32,9
32,89	35,32	35,81	35,81	37,73	37,46	38,71	40,65	42,57	45,34
42,58	41,13	40,65	40,17	38,23	42,58	44,42	45	47,42	50,33
53,71	53,71	53,23	50,13	51,3	55,07	58,06	60,49	57,59	57,6
59,72	58,06	59,52	60	59,03	56,23	58,07	57,1	53,04	52,75
56,14	55,46	55,17	53,71	51,3	46,26	49,84	48,39	48,39	50,32
52,71	52,02	52,71	52,71	51,22	49,73	49,73	49,73	49,73	50,23
52,72	51,72	50,72	47,74	40,28	36,8	28,85	39,29	40,78	43,26
46,7	43,76	41,18	41,18	44,26	44,76	42,87	40,28	47,24	51,72
50,23	48,74	47,74	49,73	49,22	49,73	50,53	51,72	51,42	50,72
51,72	50,23	49,22	53,21	51,72	54,7	57,98	58,39	59,58	63,36
66,04	69,12	66,77	69,42	65,64	65,64	69,62	70,62	70,62	71,61
69,12	67,63	70,62	72,62	77,09	78,32	77,98	77,19	77,58	78,88
76,59	77,47	76,1	76,58	75,59	74,1	75,09	74,89	74,99	74,59
70,62	70,62	70,02	69,42	69,12	70,63	72,61	73,6	72,8	71,61
72,61	72,1	76,58	78,75	78,68	78,47	81,66	82,05	79,57	79,07
78,47	77,48	74,58	70,22	67,63	74,89	75,48	73,5	74,59	74,89
74,78	76,68	76,58	77,58	75,09	76,59	78,54	77,58	77,57	79,57
78,57	79,56	81,56	80,55	78,52	76,78	75,89	74,59	75	75,09
75,59	76,57	77,28	74,2	73,81	72,49	72,49	70,41	70,72	69,82
67,98	66,44	65,15	63,19	62,96	63,85	67,09	65,05	65,05	65,05
63,64	61,96	59,68	61,17	60,87	60,67	59,78	58,68	58,18	62,56
60,67	62,16	62,15	59,58	57,45	58,18	62,16	60,67	59,13	60,67
59,69	59,68	59,3	59,19	59,68	61,47	62,07	61,91	61,66	64,45

66,14	65,94	63,9	61,66	53,71	50,69	50,23	51,83	52,71	52,7
53,71	51,42	50,72	48,04	45,25	43,26	42,27	44,26	48,47	47,25
47,24	47,24	45,55	45,24	45,24	44,56	42,77	43,71	42,77	44,39
46,25	46,29	47,74	48,74	49,83	50,13	49,93	47,84	51,33	53,21
51,42	51,72	52,61	51,52	53,91	55,72	57,68	58,69	57,49	58,59
58,48	64,05	65,63	62,66	62,16	60,17	61,27	59,04	56	56,69
57,09	57,4	59,48	59,87	58,18	56,49	60,67	61,66	62,66	62,96
62,96	67,04	67,04	66,15	65,15	62,56	63,65	63,65	63,46	59,18
61,57	63,64	64,64	69,12						

Tabela 17: Dados de Teste

66,15	64,55	62,66	64,55	63,65	63,16	62,14	61,27	60,78	60,77
60,77	59,78	59,97	60,18	60,27	60,47	64,05	67,61	66,64	65,14
67,73	68,68	73,1	71,61	70,62	70,13	71,81	72,11	71,11	73,1
72,26	72,61	71,43	72,22	70,82	70,62	69,22	67,93	69,62	69,52
71,61	69,62	67,33	65,34	65,64	66,64	64,63	64,65	69,11	68,63
68,79	67,14	68,63	69,14	69,17	70,02	69,92	69,12	67,75	66,64
67,14	67,63	65,55	62,44	63,65	60,57	57,59	57,19	58,38	56,69
61,5	60,1	64	64,5	65,2	66,99	68,3	67	67,4	68,45
67	67,35	66,46	67	64	66	66	62,6	62,8	63,5
60,75	59	60,9	59,5	58,1	57,2	56,49	54,5	55,4	55,19
54,4	53,45	57,7	57,65	56,8	56	55	54,7	56,5	56,5
58,5	58,5	59	58	56,66	56,01	56,3	55	53,98	53,31
55,2	55,85	56,8	56,5	54,49	51,5	50,2	47	48,6	48,8
50	48,5	47,5	46,9	49,78	47,4	48	50,1	49	50,3
49,5	48,5	46,89	46,99	48,35	48,9	48	47,91	49,7	49,4
48,1	47,5	49,99	50,29	50,5	50,7	52,1	55,3	58,04	58,5
60	61,89	61,7	61,45	59	59,65	60,59	59,55	60	56,97
58,36	57,6	55	50,78	53	52,48	53,19	54,19	54,81	54,82
53,2	53,91	54	53,5	57,1	60,91	61,02	59,92	58,59	55,8
58,8	59	58,7	60,5	57,22	58,4	56,48	55,5	54,1	52,65
59,5	60,1	61,3	64,66	63,5	65,2	66,2	65,9	62,39	64,49
64,75	64,51	61,9	60,01	60,1	61,9	59	56,31	57,81	61,42
64,99	65,52	68,1	67,5	69,3	69	68,3	67,2	67,75	66,4
67,52	65	65	64,84	63,59	62,2	62,6	61,45	60,5	62,4

61,4 60,8

APÊNDICE A – BASE DE REGRAS

Tabela 18: Base Obtida pelo Método WM - Exp. 1a

Se X1 é	e X2 é	então Classe é
1	1	1
1	2	2
1	3	2
2	1	2
2	2	2
2	3	2
3	1	2
3	2	2
3	3	3

Tabela 19: Base Obtida pelo Método Ishibuchi - Exp. 1a

Se X1 é	e X2 é	então Classe é	com confiança
1	1	1	0,6454
1	2	1	0,2619
1	3	2	0,2699
2	1	1	0,3066
2	2	2	0,3662
2	3	3	0,2801
3	1	2	0,3225
3	2	3	0,3178
3	3	3	0,6832

Tabela 23: Base Obtida pelo Método WM - Exp. 2b

Se X1 é	e X2 é	e X3 é	e X4 é	então Y é
1	1	1	2	3
1	2	3	3	3
2	2	1	1	1
2	2	2	1	1
2	2	2	2	2
2	2	2	3	3
2	2	3	2	2

Se X1 é	e X2 é	e X3 é	e X4 é	então Y é
2	2	3	3	3
2	3	2	2	2
2	3	3	3	3
3	2	1	2	3
3	2	2	2	2
3	2	2	3	3
3	3	2	1	2
3	3	2	2	2
3	3	2	3	2
3	3	3	2	2
3	3	3	3	3
3	3	3	4	4
3	3	4	4	4
3	4	4	4	4
4	3	3	2	1
4	3	3	3	3
4	3	4	4	4
4	4	3	3	3
4	4	3	4	4
4	4	4	3	3
4	4	4	4	4
4	4	4	5	5
4	4	5	4	4
4	4	5	5	5
4	5	4	4	4
4	5	5	4	5
4	5	5	5	5
4	5	5	6	6
5	4	4	4	4
5	4	4	5	4
5	4	5	5	5
5	5	4	4	4
5	5	4	5	5
5	5	5	4	4

Se X1 é	e X2 é	e X3 é	e X4 é	então Y é
5	5	5	5	5
5	5	5	6	6
5	5	6	6	6
5	6	6	6	6
6	5	5	5	5
6	5	5	6	6
6	5	6	6	6
6	6	5	5	6
6	6	5	6	6
6	6	6	5	6
6	6	6	6	6
6	6	6	7	6
6	6	7	7	6
6	7	7	6	6
7	6	6	6	6

Tabela 20: Base Obtida pelo Método WM - Exp. 1b

Se X1 é	e X2 é	então Classe é
1	1	1
1	2	2
1	3	2
2	1	1
2	2	2
2	3	3
3	1	2
3	2	2
3	3	2

Tabela 21: Base Obtida pelo Método Ishibuchi - Exp. 1b

Se X1 é	e X2 é	então Classe é	com confiança
1	1	1	0,7294
1	2	1	0,2633
1	3	2	0,4198
2	1	2	0,2265
2	2	2	0,4398
2	3	3	0,324
3	1	2	0,3371
3	2	3	0,2793
3	3	3	0,7382

Tabela 22: Base Obtida pelo Método WM - Exp. 2a

Se X1 é	e X2 é	e X3 é	e X4 é	então Y é
1	1	1	1	1
1	1	1	2	1
1	1	2	1	1
1	1	2	2	2
1	2	1	1	1
1	2	1	2	2
1	2	2	2	1
2	1	1	1	1
2	1	1	2	1
2	1	2	1	1
2	1	2	2	2
2	2	1	2	2
2	2	2	1	2
2	2	2	2	3
2	2	2	3	2
2	2	3	1	1
2	2	3	2	2
2	3	2	1	1
2	3	2	2	2
3	1	1	2	1
3	2	1	1	1
3	2	2	2	2
3	3	4	2	2
3	3	4	3	4
3	4	2	2	2
3	4	3	4	2
4	2	2	2	2

APÊNDICE B – PONTOS GERADOS E FUNÇÕES DE PERTINÊNCIA

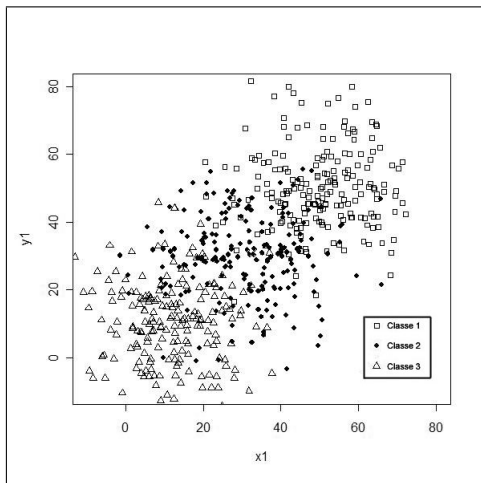


Figura 27: Classes de Pontos Aleatórios Utilizados no Experimento 1c

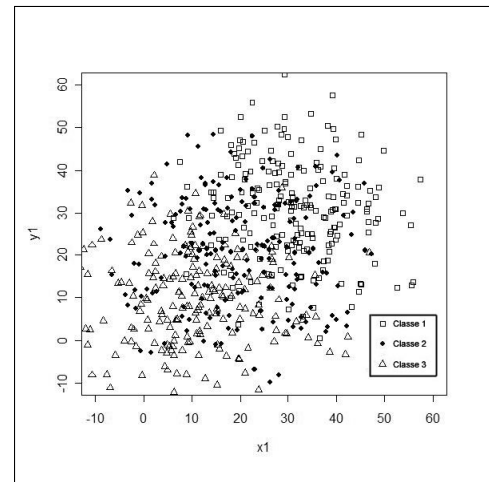


Figura 28: Classes de Pontos Aleatórios Utilizados no Experimento 1d

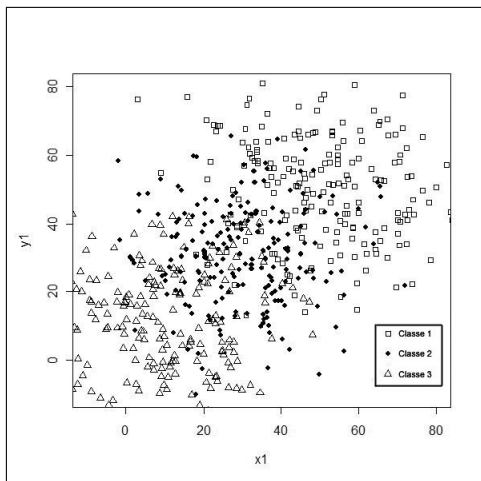


Figura 29: Classes de Pontos Aleatórios Utilizados no Experimento 1e

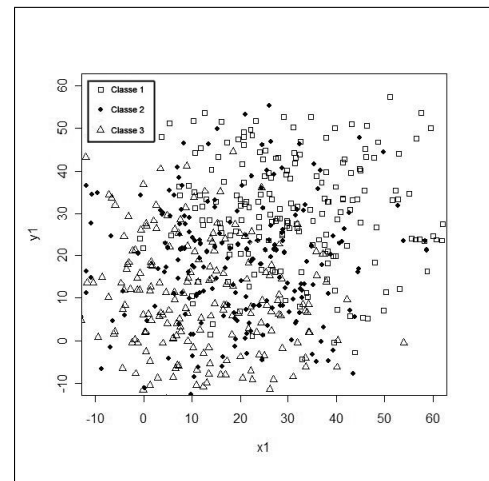


Figura 30: Classes de Pontos Aleatórios Utilizados no Experimento 1f

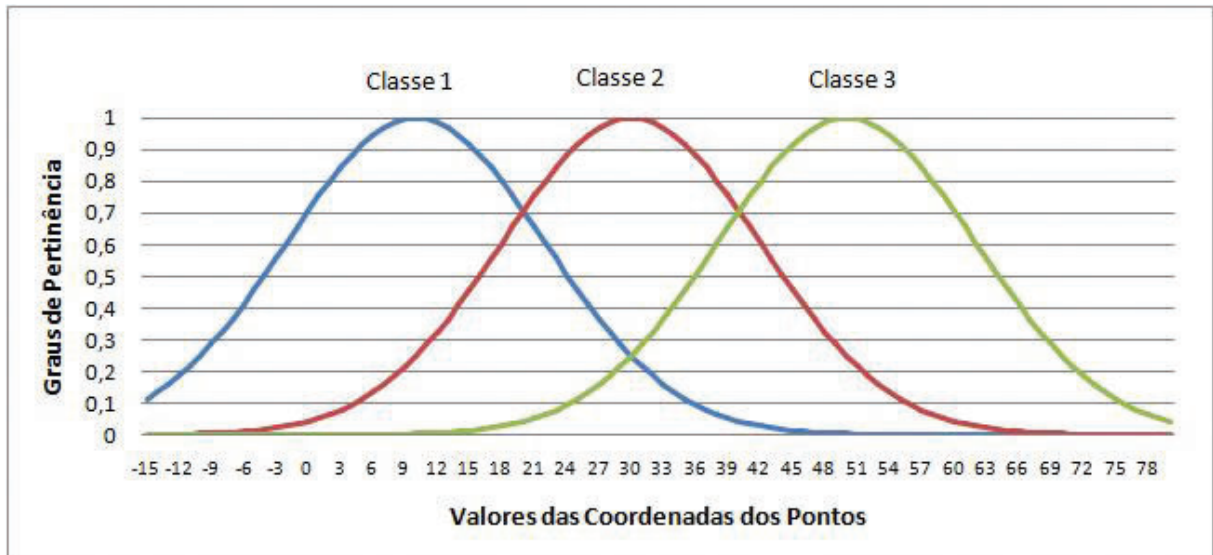


Figura 31: Funções de Pertinência para o Experimento 1c

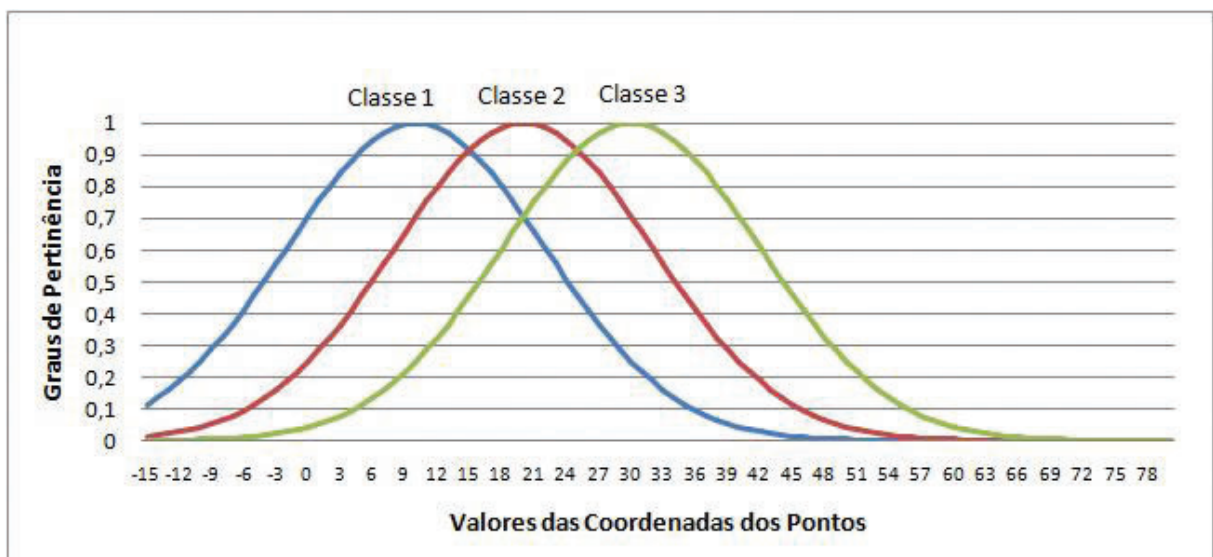


Figura 32: Funções de Pertinência para o Experimento 1d

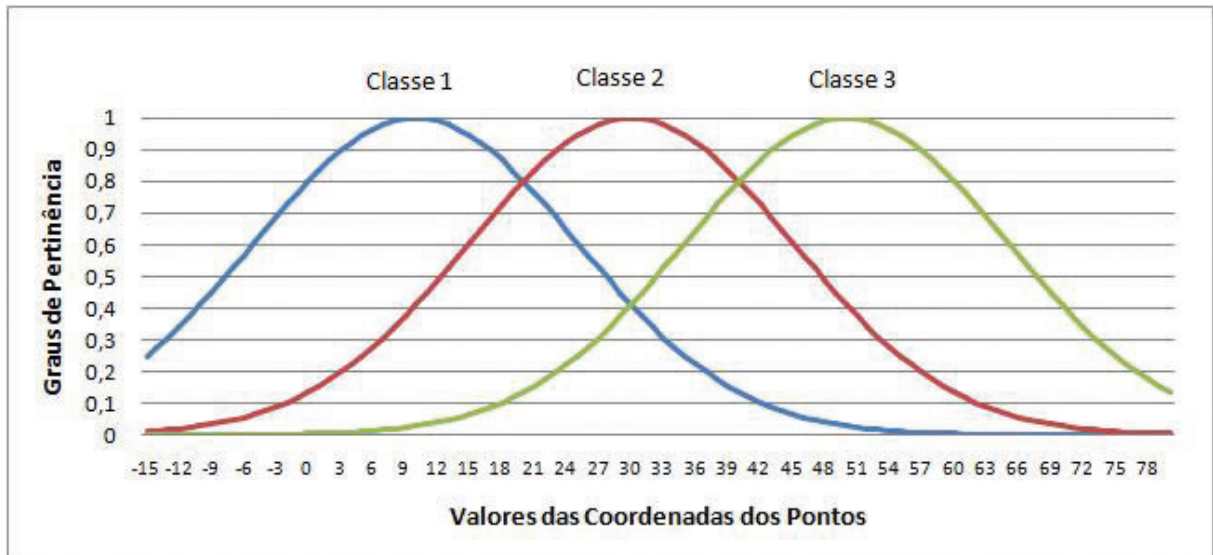


Figura 33: Funções de Pertinência para o Experimento 1e

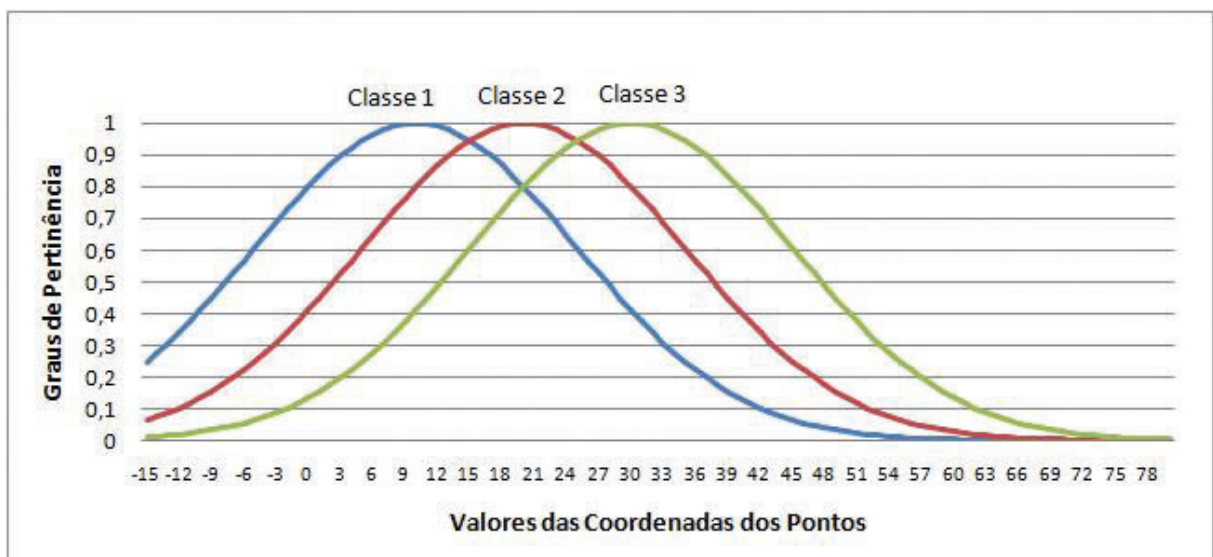


Figura 34: Funções de Pertinência para o Experimento 1f