

UNIVERSIDADE FEDERAL DO RIO GRANDE - FURG  
CENTRO DE CIÊNCIAS COMPUTACIONAIS  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO  
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

Dissertação de Mestrado

**Medidas de confiança e suporte sobre funções de *overlap*  
intervalares em sistemas baseados em regras *fuzzy*  
intervalares**

Sidnei da Fonseca Pereira Jr.

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande - FURG, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação

Orientador: Prof. Dr. Graçaliz Pereira Dimuro  
Co-orientador: Prof. Dr. Eduardo Nunes Borges  
Colaborador: Prof. Dr. Humberto Bustince

Rio Grande, 2019

## Ficha catalográfica

P436m Pereira Junior, Sidnei da Fonseca.

Medidas de confiança e suporte sobre funções de *overlap* intervalares em sistemas baseados em regras *fuzzi* intervalares / Sidnei da Fonseca Pereira Junior. – 2019.

77 f.

Dissertação (mestrado) – Universidade Federal do Rio Grande – FURG, Programa de Pós-Graduação em Computação, Rio Grande/RS, 2019.

Orientadora: Dra. Graçaliz Pereira Dimuro.

Coorientador: Dr. Eduardo Nunes Borges.

Colaborador: Dr. Humberto Bustince.

1. Conjuntos *fuzzi* intervalares 2. Raciocínio aproximado 3. Funções de *overlap* 4. Índices de *overlap* 5. Funções de penalidades I. Dimuro, Graçaliz Pereira II. Borges, Eduardo Nunes III. Bustince, Humberto IV. Título.

CDU 004.021



MINISTÉRIO DA EDUCAÇÃO  
UNIVERSIDADE FEDERAL DO RIO GRANDE  
CENTRO DE CIÊNCIAS COMPUTACIONAIS  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO  
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

## DISSERTAÇÃO DE MESTRADO

### Medidas de confiança e suporte sobre funções de *overlap* intervalares em sistemas baseados em regras *fuzzy* intervalares

Sidnei da Fonseca Pereira Júnior

Banca examinadora:

Prof. Dr. Benjamin René Callejas Bedregal

Prof. Dr. Eder Mateus Nunes Gonçalves

Prof. Dr. Héliida Salles Santos

Prof. Dr. Graçaliz Dimuro Pereira  
Orientador(a)

Prof. Dr. Eduardo Nunes Borges  
Coorientador(a)

*Aos meus pais, Sidnei e Lúcia, por me apoiarem em todas as minhas iniciativas, a minha esposa Paula por fazer parte ativa nessa caminhada, a meus filhos Lucas e Nicolle que, com muito carinho e apoio, auxiliaram para que eu chegasse até esta etapa da minha vida, e ao pequeno Otto que nasceu em meio a esse trabalho.*

## RESUMO

PEREIRA JR., Sidnei da Fonseca. **Medidas de confiança e suporte sobre funções de *overlap* intervalares em sistemas baseados em regras *fuzzy* intervalares.** 2019. 77 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

O objetivo deste trabalho é propor o uso de conjuntos *fuzzy* intervalares em funções de *overlap* para construção de índices de *overlap* intervalares e, a partir destes, obter medidas de confiança e suporte para utilização em Sistemas de Classificação baseados em Regras *Fuzzy*, cuja tomada de decisão final é dada por funções de penalidade (ou funções *penalty*). Para o mapeamento de problemas que tratam do raciocínio aproximado a modelagem *fuzzy* é frequentemente realizada em um espaço finito, ou seja, é realizada uma discretização dos valores para que haja um número limitado de possíveis estados. Essa abordagem fornece uma solução computacional rápida e modelagem intuitiva para trabalhar com conjuntos *fuzzy* em várias aplicações. No entanto, ao considerar o uso de quantidades *fuzzy*, estamos interessados em executar operações através de conjuntos definidos pelos números reais. A propagação de erros pode tornar-se importante e afetar o resultado final. É difícil para o especialista determinar se o grau de certeza é 0.5 ou 0.503, por exemplo. Esse problema encontra sua solução dentro do conceito de conjuntos *fuzzy* intervalares. Primeiramente, foi introduzido um método para criar medidas de confiança e suporte baseado em índices de *overlap* intervalares, que geralmente são usados para avaliar o grau de certeza ou interesse de uma determinada regra de associação. Estes índices de *overlap* intervalares são construídos a partir de funções de *overlap* intervalar, que são um tipo especial de funções de agregação, não necessariamente associativas, que servem para aplicações relacionadas aos problemas de sobreposição de conjuntos. Esta dissertação apresenta um novo Mecanismo de Raciocínio *Fuzzy* para ser usado em sistemas de classificação baseados em regras *fuzzy* considerando diferentes índices de *overlap* intervalares, que generaliza os métodos clássicos. Ao considerar vários índices de *overlap* intervalares, funções de penalidade foram utilizadas para a tomada de decisão relacionada à seleção da melhor classe. Por fim, é apresentado o método de consenso para classificação utilizando funções de penalidade e os resultados teóricos relacionados aos métodos desenvolvidos.

**Palavras-chave:** Conjuntos *Fuzzy* Intervalares, Raciocínio Aproximado, Funções de *Overlap*, Índices de *Overlap*, Funções de penalidades.

## ABSTRACT

PEREIRA JR., Sidnei da Fonseca. **Confidence and support measures on interval overlap functions in systems based on interval fuzzy rules.** 2019. 77 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

The aim of this work is to propose the use of interval fuzzy sets in overlap functions to construct interval overlap indices and to obtain confidence and support measures for use in Classification Systems based on Fuzzy Rules, whose final decision-making is given by penalty functions. For the mapping of problems that deal with approximate reasoning, fuzzy modeling is often performed in a finite space, that is, a discretization of the values is performed to have a limited number of possible states. This approach provides a fast computational solution and intuitive modeling to work with fuzzy sets in various applications. However, when considering the use of fuzzy quantities, we are interested in executing operations through sets defined by real numbers. The propagation of errors can become important and affect the end result. Sometimes it is difficult for the specialist to determine whether the degree of certainty is 0.5 or 0.503, for example. This problem finds its solution within the concept of fuzzy intervals. First, a method was introduced to create confidence and support measures based on interval indexes, which are generally used to evaluate the degree of certainty or interest of a given association rule. These interval overlap indices are constructed from interval overlap functions which are a special kind of aggregation functions, not necessarily associative, that serve for applications related to set overlap problems. This dissertation proposal presents a new Fuzzy Reasoning Mechanism to be used in classification systems based on fuzzy rules considering different interval overlap indexes, which generalizes the classical methods. When considering several interval overlap indexes, penalty functions were used for decision-making related to the selection of the best class. Finally, we present the consensus method for classification using penalty functions and the theoretical results related to the methods developed.

**Keywords:** Fuzzy Sets, Intervals Fuzzy Sets, Overlap Functions, Overlap Indices, Penalty Functions.

## LISTA DE FIGURAS

Figura 1	Função de Pertinência, Subconjunto <i>Fuzzy</i> e Subconjunto <i>Crisp</i> . . .	21
Figura 2	União de Conjuntos Fuzzy . . . . .	22
Figura 3	Intersecção de Conjuntos Fuzzy . . . . .	22
Figura 4	Complemento de Conjuntos Fuzzy . . . . .	23
Figura 5	Esquema da Inferência Sobre Regras <i>Fuzzy</i> (Kasabov 1996) . . . . .	27
Figura 6	Sistema de Inferência Fuzzy . . . . .	27
Figura 7	Conjunto <i>Fuzzy</i> Intervalar . . . . .	39
Figura 8	Processo de Aprendizado Supervisionado . . . . .	43
Figura 9	Processo de Classificação . . . . .	44
Figura 10	Conjunto <i>Fuzzy</i> Intervalar Usado Nesta Dissertação . . . . .	49
Figura 11	Termos Linguísticos para Massa Corporal e Idade . . . . .	61
Figura 12	Regras . . . . .	61

## LISTA DE TABELAS

Tabela 1	Exemplos de funções <i>overlap</i> . . . . .	32
Tabela 2	Exemplos de índices de <i>overlap</i> utilizados nesta dissertação . . . . .	33
Tabela 3	Exemplos de índices de <i>overlap</i> construídos usando o Teorema 1 . . . . .	33
Tabela 4	Exemplos de funções <i>overlap</i> intervalares . . . . .	43
Tabela 5	Exemplos de índices de <i>overlap</i> intervalar . . . . .	51
Tabela 6	Exemplos de índices de <i>overlap</i> intervalar construídos pela Definição 25 . . . . .	51
Tabela 7	Exemplos usados para treinamento . . . . .	62
Tabela 8	Valores Fuzzy Intervalares . . . . .	62
Tabela 9	Confiança com diferentes <i>overlaps</i> intervalares . . . . .	65
Tabela 10	$M$ Agregações . . . . .	67
Tabela 11	Resultado Algoritmo 3 para o ponto médio do intervalo . . . . .	67
Tabela 12	Resultados intervalares do Algoritmo 2 . . . . .	68
Tabela 13	Resultados do Algoritmo 2 Limite Inferior do Intervalo . . . . .	68
Tabela 14	Resultados do Algoritmo 2 Limite Superior do Intervalo . . . . .	68



## LISTA DE ABREVIATURAS E SIGLAS

TCF	Teoria dos Conjuntos Fuzzy
TCFI	Teoria dos Conjuntos Fuzzy Intervalar
SF	Sistemas <i>Fuzzy</i>
CF	Conjunto <i>Fuzzy</i> ( <i>Fuzzy Set</i> )
CFI	Conjunto <i>Fuzzy</i> Intervalar ( <i>Interval-valued Fuzzy Set</i> )
MRF	Mecanismo de Raciocínio Fuzzy
MRF-I	Mecanismo de Raciocínio Fuzzy Intervalar
MRF-II	Mecanismo de Raciocínio Fuzzy Intervalar (Limite)Inferior
MRF-IS	Mecanismo de Raciocínio Fuzzy Intervalar (Limite)Superior
SCBRF	Sistema de Classificação Baseado em Regras Fuzzy
SCBRFI	Sistema de Classificação Baseado em Regras Fuzzy Intervalar
FP	Função de Pertinência
BR	Base de Regras
BD	Base de Dados
BC	Base de Conhecimento
SC	Soft Computing (Computação Flexível)
IA	Inteligência Artificial

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	12
1.1	Teoria dos Conjuntos e Lógica <i>Fuzzy</i>	12
1.2	Lógica <i>Fuzzy</i> Intervalar	13
1.3	Problemas de Classificação e Sistemas de Classificação Baseados em Regras <i>Fuzzy</i>	14
1.4	Funções de Agregação e Funções de <i>Overlap</i>	15
1.5	Índice de <i>Overlap</i>	16
1.6	Motivação	16
1.7	Objetivos Geral e Específicos	17
1.8	Organização do Texto	17
<b>2</b>	<b>FUNDAMENTOS TEÓRICOS</b>	19
2.1	Teoria <i>Fuzzy</i>	19
2.1.1	Teoria dos Conjuntos <i>Fuzzy</i>	19
2.1.2	Raciocínio Aproximado	24
2.1.3	Sistema de Inferência <i>Fuzzy</i>	25
2.2	Função de Agregação	28
2.2.1	Função de <i>Overlap</i>	31
2.2.2	Índices de <i>Overlap</i>	32
2.3	Funções de Penalidade ( <i>Penalty</i> )	33
2.4	Teoria Intervalar	35
2.4.1	Aritimética Intervalar	36
2.4.2	Ordem e Ordem Admissível	37
2.5	Teoria <i>Fuzzy</i> Intervalar	38
2.5.1	Teoria dos Conjuntos <i>Fuzzy</i> Intervalar	38
2.5.2	Função de Agregação Intervalar	40
2.5.3	Função de <i>Overlap</i> Intervalar	40
2.6	Sistemas de Classificação	43
2.6.1	Sistemas de Classificação Baseado em Regras <i>Fuzzy</i>	45
<b>3</b>	<b>ÍNDICE DE <i>OVERLAP</i> INTERVALAR E SISTEMAS DE CLASSIFICAÇÃO INTERVALARES</b>	48
3.1	Sistemas de Classificação Baseados em Regras <i>Fuzzy</i> Intervalares Usando Várias Funções <i>Overlap</i> Intervalares e Funções de Agregação	48
3.2	Construção de Grau de Confiança Intervalar Sobre Índices de <i>Overlap</i> Intervalares	50

3.3	Mecanismo de Raciocínio de um Sistema de Classificação Baseado em Regras <i>Fuzzy</i> Intervalar . . . . .	55
3.4	Usando Funções <i>Penalty</i> (penalidade) para Escolher a Melhor Classe em Conjuntos de Regras <i>Fuzzy</i> . . . . .	56
4	<b>AVALIAÇÃO EXPERIMENTAL</b> . . . . .	60
4.1	Calculando o Grau de Confiança Intervalar para Cada Regra . . . . .	62
4.2	Inferência . . . . .	65
4.3	Consenso . . . . .	66
4.4	Resultados . . . . .	67
5	<b>CONCLUSÃO</b> . . . . .	70
	<b>REFERÊNCIAS</b> . . . . .	72

# 1 INTRODUÇÃO

Os estudos sobre lógica realizados por Aristóteles na antiga Grécia definiam que classificar é agrupar metodicamente os seres seguindo um critério de semelhanças e diferenças, o que, de um modo geral, consiste em repartir objetos em grupos distintos, segundo suas características comuns ou diferentes. Ainda hoje, a classificação está presente em todas as ciências, porque todas necessitam organizar os seus fenômenos e resultados dentro de uma ordem que revele conexão ou inteligibilidade entre eles.

Na lógica clássica, consideramos o valor verdade de uma proposição pertencendo somente a duas possibilidades: verdadeiro (1), ou falso (0), e também representamos os predicados por termos exatos como: igual a, menor que, diferente de, etc. Desse modo, os elementos de um conjunto  $A$  em um universo  $X$  pertencem ou não à aquele conjunto.

Em 1965 com o trabalho de ZADEH (1965), surge a teoria dos conjuntos *fuzzy* (TCF) com a proposta de contrapor modelos determinísticos a modelos mais flexíveis, por considerar a possibilidade de infinitos valores ou graus de pertinência de um elemento a um conjunto.

## 1.1 Teoria dos Conjuntos e Lógica *Fuzzy*

Pela definição (ZADEH, 1978), um conjunto  $A$ , do universo  $U$ , é caracterizado pela função de pertinência  $\mu_A : U \rightarrow [0, 1]$ ; então, seja o conjunto *fuzzy*  $A$ , o elemento  $x$  e  $\mu_A(x)$  o grau de pertinência de  $x$  no conjunto  $A$ ; então tem-se que: se  $\mu_A(x) = 1$ , então é porque  $x$  é um elemento do conjunto  $A$ , se  $\mu_A(x) = 0$ , então  $x$  não é um elemento do conjunto  $A$ ; e se  $0 < \mu_A(x) < 1$  então é porque o elemento  $x$  é parcialmente definido como um elemento do conjunto, sendo  $\mu_A(x)$  o grau de pertinência. Desde então, esta teoria tem sido utilizada nas mais diversas áreas, uma vez que, por sua própria natureza *fuzzy*, é apropriada para lidar com a incerteza, ignorância e vagueza presentes em problemas do mundo real.

Como ferramenta, os conjuntos *fuzzy* (CF) permitem modelar a imprecisão e a incerteza que estão muitas vezes presentes em linguagens naturais. Intuitivamente, uma variável linguística é um substantivo, enquanto seus valores são adjetivos representados

por um conjunto *fuzzy*, como por exemplo, altura sendo uma variável linguística que pode assumir os atributos como alto, baixo, ou outros caso necessário.

Os sistemas *fuzzy* (SF) são sistemas de apoio à decisão que mapeiam conhecimento aproximado, impreciso ou incerto. A aquisição deste tipo de conhecimento é realizada através do mapeamento dos conceitos envolvidos. Assim conceitos como: alto e baixo, quente e frio, novo e velho, podem ser representados por meio de conjuntos *fuzzy* para serem manipulados pelos sistemas *fuzzy*.

Nas últimas décadas foram apresentadas várias abordagens para o tratamento da incerteza em sistemas especialistas, em geral um sistema especialista são programas inteligentes para computadores, desenvolvidos para emular o processo de raciocínio de um especialista em alguma área de seu domínio. Segundo KANDEL et al (1996) um dos ramos alternativos da Lógica *Fuzzy* que tem escopo em modelar a imprecisão dos conceitos humanos e inexatos é o Raciocínio Aproximado (RA), para isso a Teoria dos Conjuntos *Fuzzy* forneceram elementos essenciais para o desenvolvimento de Sistemas Baseados em Regras *Fuzzy* (SBRF), apresentando dessa forma um *framework* estruturado para criação e aplicação de um Mecanismo de Raciocínio Aproximado (MRA).

## 1.2 Lógica *Fuzzy* Intervalar

Mas nem toda a incerteza ou ignorância pode ser representada pelas definições de conjuntos *fuzzy*, por ser necessário determinar um valor numérico pertencente ao intervalo  $[0,1]$ , torna-se uma tarefa de difícil modelagem e escolha quando representamos valores muito precisos e o especialista determina valores estimados ou inexatos, ou que representam várias opiniões sobre o mesmo fenômeno, isto é, quando há dificuldade de escolher com determinada certeza o grau de pertinência de um dado elemento pertencer àquele conjunto. Algumas vezes é difícil para o especialista representar seu conhecimento através de números reais, ele pode determinar que um determinado valor possui um grau de pertinência de 0.2, mas é complexo para ele determinar se esse grau de certeza é 0.208 ou 0.220, por exemplo. Além disso, quando a aquisição da informação é realizada entre vários especialistas, é frequente a diferença entre alguns valores de pertinência fornecidos pelos mesmos (BUSTINCE, 2010).

Desta forma, é necessária uma opção para determinar qual o melhor valor que representa a dada situação, e que aceite essas diferenças de avaliações e as ignorâncias presentes. Neste sentido, é razoável que um grau de pertinência seja representado por uma faixa de valores ao invés de um valor pontual, e segundo MOORE (1979) a Matemática Intervalar (MI) tem o objetivo de responder à questão da exatidão e aplicabilidade presentes na Computação Científica. O uso de intervalos faz com que seja possível representar dados inexatos e controlar erros de arredondamento, aproximações e erros de cruzamento de processos.

Então, uma das alternativas para contornar o problema da determinação das funções de pertinência é oferecida pela Lógica *Fuzzy* Intervalar (LFI), como apresentado por BEDREGAL; TAKAHASHI (2006), os Conjuntos *Fuzzy* Intervalares (CFI) foram introduzidos de forma independente por Zadeh (1975) e outros autores nos anos 70, combinando teoria de conjuntos *fuzzy* com matemática intervalar, desenvolvida por uma abordagem da Lógica *Fuzzy* combinada com intervalos, utilizando subintervalos de  $[0, 1]$  para atribuir valores verdade às proposições *fuzzy* e aos graus de pertinência de elementos a subconjuntos *fuzzy*.

Seja  $I = [0; 1]$  e  $\mathbb{I} = \{X \mid X \subseteq U\}$ . Seja  $A$  um subconjunto *fuzzy* intervalar, com função de pertinência intervalar  $M_A : U \rightarrow \mathbb{I}$ , com  $M_A(x) = [\mu_{A_i}(x), \mu_{A_s}(x)]$ , onde  $\mu_{A_i}, \mu_{A_s} : U \rightarrow [0; 1]$  são as funções de limite inferior e superior, respectivamente. Nestas aplicações, as funções de pertinências com valores intervalares, são utilizadas para o fim de modelar a incerteza no processo e determinar qual é o grau de pertinência mais preciso para um elemento em relação a um termo linguístico. Os graus de pertinência de intervalos também podem resumir a opinião de vários especialistas, com os parâmetros de intervalo do limite inferior e superior, respectivamente, representando os graus mínimos e máximos fornecidos por eles.

Assim, a distância entre os limites inferiores e superiores do grau de pertinência intervalar de um determinado elemento pode ser entendida como uma medida da falta de certeza do especialista para fornecer um valor exato de pertinência desse elemento.

### 1.3 Problemas de Classificação e Sistemas de Classificação Baseados em Regras Fuzzy

Com os avanços tecnológicos, os problemas de classificação se intensificaram. A sistematização da coleta, armazenamento e disponibilização de dados geraram novos conjuntos de dados implícitos, que se não ganharem uma “forma”, deixam informações sem um significado útil.

A classificação de dados é aplicada em diversos problemas reais, tais como: reconhecimento de imagens, diferenciação de símbolos, classificação de diagnósticos médicos, identificação de padrões, entre outros (AGGARWAL, 2014).

Em particular, os Sistemas de Classificação Baseados em Regras *Fuzzy* (SCBRF) são amplamente utilizados para lidar com problemas de classificação em aplicações do mundo real, por várias razões, entre elas: fornecem um modelo interpretável, geralmente apresentam um bom desempenho e podem combinar informações provenientes de diferentes fontes como: conhecimento especializado, modelos matemáticos, bases de dados ou medidas empíricas (HERRARA et al., 1999).

Um sistema típico de classificação associativa é construído por dois estágios: descobrir as regras de associação inerentes a um banco de dados; selecionar um pequeno

conjunto de regras de associação relevantes para construir um classificador. A tarefa da classificação é encontrar um conjunto de regras para identificar as classes de padrões indeterminados. Uma regra de associação *fuzzy* pode ser considerada uma regra de classificação se o antecedente contiver conjuntos de itens *fuzzy* e a parte resultante contiver apenas um rótulo de classe. Os SCBRF são compostos por, Base de Conhecimento (BC), formada pela base de regras e banco de dados; Mecanismo de Raciocínio Fuzzy (MRF), que classifica os elementos através das informações contidas na base de conhecimento.

Nestes sistemas, um dos conceitos mais importantes é o de função de agregação, cujo o papel é crucial no mecanismo de raciocínio *fuzzy*

#### 1.4 Funções de Agregação e Funções de *Overlap*

O processo de combinar diversos valores numéricos num único valor, é chamado agregação, e as funções matemáticas que fornecem um mecanismo para isso são chamadas de funções de agregação. As funções de agregação são usadas em diversas áreas de conhecimentos tais como: ciência da computação (por exemplo em inteligência artificial, processamento digital de imagens e algoritmos experimentais), em matemática aplicada (por exemplo em probabilidade, estatística e teoria de decisão), em economia (por exemplo nas leis da demanda e em análises de risco), etc (BELIAKOV et al, 2007).

Exemplos de funções de agregação que são encontradas na literatura são: T-norma e T-conorma, que são associativas, generalizando o AND e o OR da lógica clássica para o contexto *fuzzy* (KLEMENT et al, 2000).

Um problema comum em muitos campos é a atribuição de um determinado elemento a uma das várias classes disponíveis. Se a separação entre as classes não for clara, o especialista pode estar inseguro sobre como atribuir objetos a uma classe específica. Talvez essas classes sejam de natureza *fuzzy* e o especialista percebe que os elementos estão simplesmente entre várias classes, ou pode ser que a fronteira entre as classes existe, mas não é claramente perceptível pelo especialista. Em qualquer uma dessas situações, surge o conceito de *overlap* (BUSTINCE et al, 2010).

As funções de *overlap* são um tipo especial de funções de agregação não necessariamente associativas propostas para medir o grau de sobreposição em um sistema de classificação *fuzzy* com duas ou mais classes. Este conceito foi aplicado a alguns problemas de sobreposições interessantes, como no processamento de imagem, tomada de decisão baseada em relações de preferência *fuzzy* e problemas de classificação (BEDREGAL et al, 2013).

As funções de agregação também foram definidas no contexto *fuzzy* intervalar. (outras citações de T-norma intervalar) A introdução dos conceitos e os métodos para construção de funções de *Overlap* Intervalares são discutidos de forma ampla por BEDREGAL et al (2017).

## 1.5 Índice de Overlap

Historicamente, ZADEH (1978) introduziu o índice de consistência entre dois conjuntos *fuzzy*  $A$  e  $B$  sobre o mesmo  $U$  referencial ( $card(U) = n$ ) como extensão natural do índice de sobreposição booleana, da seguinte forma:

$$O_Z(A, B) = \max_{i=1, \dots, n} \min(A(u_i), B(u_i)). \quad (1)$$

Posteriormente, verificou-se que, no campo aplicado, alguns sistemas *fuzzy* interpolativos fornecem melhores resultados quando são utilizadas algumas generalizações do índice de consistência. Tais generalizações são chamadas índices de *overlap* e foram definidas por DUBOIS et al (2000) e estendidas por BUSTINCE et al (2009), entre outros. Em nosso trabalho anterior, ELKANO et al (2017), foi introduzido um novo mecanismo de raciocínio *fuzzy* que leva em consideração as informações fornecidas por várias ou mesmo todas as regras *fuzzy* no sistema, especialmente nos SCBRF. Geralmente são usadas duas medidas para avaliar o grau de certeza ou interesse de uma determinada regra de associação, ou seja, o grau de confiança e o suporte. A confiança de uma associação é classicamente medida pela co-ocorrência de atributos em tuplas de bases de dados.

Até o presente momento, embora se conheça o conceito de função de *overlap* intervalar, não existe o conceito de índice de *overlap* intervalar, de tal forma que não é possível estabelecer o grau de confiança e suporte no contexto de incerteza em que se exige a utilização de lógica *fuzzy* intervalar.

## 1.6 Motivação

Dada a discussão apresentada nas seções anteriores, pergunta-se qual o conceito de índice de *overlap* intervalar para construção dos graus de confiança sobre funções de *overlap* intervalares?

Observa-se que neste contexto, deve-se trabalhar com sistemas de classificação baseados em regras *fuzzy* intervalares (SCBRFI), inspirado no trabalho de SANZ et al. (2013)

Além disso, definindo diferentes funções de *overlap* intervalares, serão obtidos diferentes índices de *overlap* intervalares ou seja, medidas de confiança e suporte, gerando ensembles de SCBRFI. Cada sistema pode resultar em uma classe diferente o que nos leva a outro problema: como tomar a decisão de qual classe realmente deve pertencer o exemplo que está sendo classificado? Isto é, qual é o valor de consenso entre os resultados fornecidos pelo conjunto de SCBRFIs?

O resultado consensual é aquele que apresenta o menor valor de desvio em relação a todos os resultados obtidos. Para calcular esses desvios/dissimilaridades entre os resultados obtidos pelas diferentes tuplas de funções de agregação aplica-se as funções *penalty* (BUSTINCE et al, 2016), que são capazes de fornecer uma medida do desvio dos valores



de saída obtidos através de diferentes funções de agregação, a fim de indicar um valor de consenso relativo as entradas, ou uma penalidade por não ter um consenso.

Finalmente, surge a questão: como deve ser definida a função *penalty* sobre os resultados intervalares obtidos nesta fase?

Esta dissertação propõe-se a responder estas três questões de pesquisa aqui colocadas.

## 1.7 Objetivos Geral e Específicos

O objetivo desta dissertação é definir o conceito de índice de *overlap* intervalar para construção de graus de confiança sobre funções de *overlap* intervalares, que serão aplicados em um mecanismo de raciocínio *fuzzy* intervalar, gerando um conjunto (ensembles) de SCBRFI, com obtenção de consenso de resultados intervalares por meio de funções *penalty*.

Os objetivos específicos são:

1. Definir o conceito de índice de *overlap* intervalar para construção de graus de confiança sobre funções de *overlap* intervalares;
2. Criar o método para medir o grau de confiança e suporte das classes perante as regras utilizando diferentes funções de *overlap* intervalares;
3. Introduzir um novo mecanismo de raciocínio *fuzzy* que utiliza medidas de confiança e suporte construídas por diferentes índices de *overlap* intervalares, a exemplo de ELKANO et al (2017), para ser usado em SCBRFI;
4. Definir métodos para aplicar funções *penalty* sobre dados intervalares para a tomada de decisão relacionada a seleção da melhor classe neste contexto;
5. Desenvolver três algoritmos: para treinamento das regras, inferência e classificação e tomada de decisão com funções *penalty*;
6. Desenvolver um estudo de caso para exemplificar os conceitos e algoritmos desenvolvidos.

## 1.8 Organização do Texto

O restante desta dissertação está organizado conforme mencionado a seguir. No capítulo 2 está descrito a fundamentação teórica da dissertação, constam as definições da teoria *fuzzy*, funções de agregação, funções *penalty*, bem como os conceitos de sistemas de classificação, e por fim a teoria intervalar e teoria *fuzzy* intervalar.

O capítulo 3 descreve a aplicação da dissertação em sistemas *fuzzy* intervalares, com a construção dos graus de confiança, o mecanismo de raciocínio *fuzzy* intervalar

e a aplicação da função  $\text{penaty}$ . Em seguida, no capítulo 4 foi descrito a abordagem proposta com a metodologia utilizada e apresentado o experimento feito com base na fundamentação teórica do capítulo 3 com a discussão dos resultados.

E por fim, no capítulo 6 tem-se a conclusão da dissertação.

## 2 FUNDAMENTOS TEÓRICOS

Neste capítulo são apresentadas os conceitos básicos que serão abordados no decorrer da dissertação. A seção 2.1 apresenta os conceitos da Teoria Fuzzy. E na seção 2.2, está descrito sobre Função de Agregação. Na seção 2.3 são definidas Funções *Penalty*. A seções 2.4 introduzem os conceitos de Sistemas de Classificação e Sistemas de Classificação Baseados em Regras Fuzzy. E nas seções 2.5 e 2.6 constam os conceitos do da Teoria Intervalar e Teoria Fuzzy Intervalar.

### 2.1 Teoria Fuzzy

#### 2.1.1 Teoria dos Conjuntos Fuzzy

A definição de conjuntos *fuzzy* foi introduzida em 1965 por Loft Asker Zadeh, pela necessidade de um tratamento matemático para representar valores linguísticos incertos, aproximados ou difusos, como números “próximo de 5”, “muito quente”, “baixo”. Esses elementos exigem graus de pertinência diferentes dos valores *crisp* (1), quando totalmente pertencente, e (0) quando totalmente não pertencente, da Lógica Clássica. Na teoria dos conjuntos *fuzzy* um elemento pode pertencer parcialmente a um dado conjunto. Um conjunto *fuzzy* contém elementos em variados graus de pertinência no intervalo  $[0, 1]$ .

**Definição 1** (ZADEH, 1978) *Um subconjunto fuzzy A de um universo U é determinado pela função de pertinência  $\mu_A : U \rightarrow [0, 1]$ , onde  $\mu_A(x)$  é o grau de pertinência do elemento  $x \in U$  para o subconjunto fuzzy A.*

Os valores da função de pertinência são números reais no intervalo  $[0, 1]$ , onde  $\mu_A(x) = 1$  significa que o elemento  $x$  pertence totalmente ao conjunto,  $\mu_A(x) = 0$  significa que o elemento  $x$  não pertence ao conjunto, cada valor da função  $\mu_A(x)$  é chamado grau de pertinência, e se  $0 < \mu_A(x) < 1$  então é porque o elemento  $x$  é parcialmente definido como um elemento do conjunto.

Dado um conjunto de universo  $U$ , com  $\text{card}(U) = n$ , denote por  $CF(U)$  o espaço de todos os conjuntos *fuzzy* definidos sobre  $U$ . Dois conjuntos *fuzzy*  $A, B \in CF(U)$  são ditos completamente disjuntos se  $A(u)B(u) = 0$ , para todo  $u \in U$ .

Os conjuntos *fuzzy* possuem várias propriedades básicas, as fundamentais para esta dissertação são:

- Domínio do Conjunto Fuzzy, é o universo total de valores possíveis para os elementos de um conjunto e depende do contexto, como exemplo a “altura” em centímetros, os números “próximos de 5” em reais, e a “temperatura” em graus Celsius são o domínio do conjunto;
- Universo do Discurso, o universo do conjunto *fuzzy* é o espaço *fuzzy* completo de variação de um modelo de variável, sendo o universo do discurso do modelo da variável “próximo de 5” os valores 0 a 10, por exemplo;
- Suporte do Conjunto Fuzzy, é a área efetiva do domínio de um conjunto *fuzzy*, que corresponde ao intervalo no qual a pertinência  $\mu_A(x)$  seja maior que zero. Quando o suporte é um único ponto em  $U$ , com o valor de  $\mu_A(x) = 1$  é chamado *Conjunto Singleton*;
- Normalização, um conjunto *fuzzy* é dito normal quando sua altura é igual a 1, ou seja, um tal conjunto  $A$  é dito normal se existir  $u \in U$  tal que  $A(u) = 1$ ;

**Definição 2** O subconjunto clássico de  $U$  definido por

$$supp_F = \{x \in U \mid \mu_F(x) > 0\}$$

é denominado suporte de  $F$ .

Existem várias formas de representação de Conjuntos Fuzzy, se  $U$  é um universo de discussão e  $x$  um elemento de  $U$ , então o conjunto *fuzzy*  $A$  definido em  $U$  pode ser escrito como uma coleção de pares ordenados:

$$A = \{(x, \mu_A(x)) \mid x \in U\} \quad (2)$$

onde cada par  $(x, \mu(x))$ , tem o valor  $x$  seguido de seu grau de pertinência em  $A$ . Outra notação comumente usada para universos discretos é a seguinte:

$$A = \sum_{x_i \in U} \mu_A(x_i) \mid x_i \quad (3)$$

E, para um universo contínuo de discussão, pode-se escrever a equação 3 como:

$$A = \int_U \mu_A(x) \mid x \quad (4)$$

A Função de Pertinência é uma generalização da função característica da programação funcional, que estabelece uma correspondência entre um elemento no domínio e um valor

verdade que indica o grau de pertinência do elemento no conjunto, está função verdade pode ser representada pela equação abaixo (DUBOIS; PRADE, 2000):

$$\mu : U \rightarrow [0, 1], \text{ onde } A \text{ é subconjunto fuzzy; e } 0 \leq \mu_A(x) \leq 1 \text{ para todo } x \in U. \quad (5)$$

Basicamente um conjunto *fuzzy* consiste em, (figura 1):

- um eixo horizontal, de valores crescentes monotonicamente, que constituem o conjunto que representa o domínio;
- um eixo vertical, entre 0 e 1, que indica o grau de pertinência ao conjunto;
- uma função, que estabelece a relação entre os dois eixos.

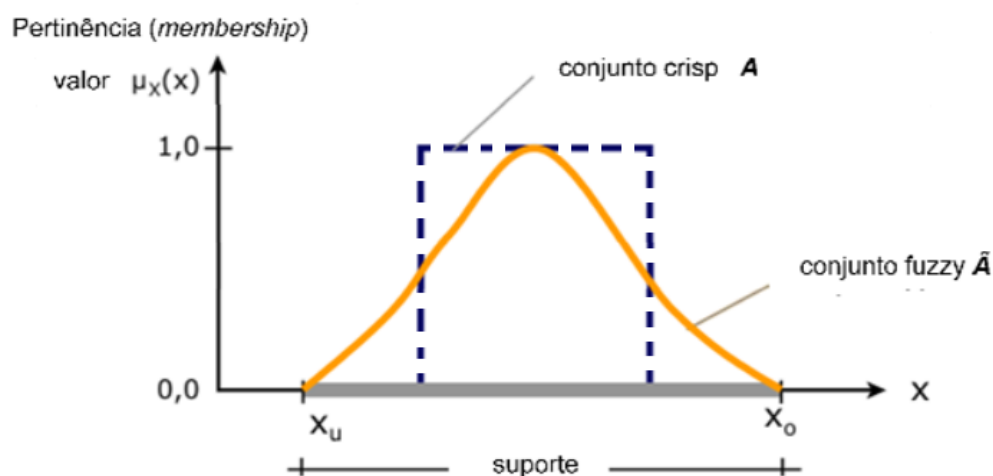


Figura 1: Função de Pertinência, Subconjunto *Fuzzy* e Subconjunto *Crisp*

Os conjuntos *fuzzy* podem ser representados por diferentes tipos de curvas, o especialista deve modelar a curva que mais se aproxima do comportamento da variável em questão. As principais representações de conjuntos *fuzzy* são:

- Linear, é o tipo mais simples, e uma boa escolha para aproximar conceitos mal compreendidos, que não são números *fuzzy*, pode ser linear crescente ou decrescente;
- Curva-S/Z, são o tipo de representação que correspondem a superfícies não lineares crescentes e decrescentes, servem para modelar a dinâmica de variáveis aleatórias;
- Curvas Tipo Sino, é baseada no conceito de aproximações a um valor central;
- Trapezoidal, é uma variação do tipo sino, e possui um processamento rápido e permite uma descontinuidade no conjunto;

- Triangular, outra variação do tipo sino, sendo possível ser simétrico ou não, e serve para representar o conceito "perto de".

As operações sobre conjuntos são importantes para os sistemas baseados em conhecimento, segundo Kandel 1996, uma operação é uma relação entre conjuntos que produz outro conjuntos. O processo de inferência é baseado em conectivos lógicos proposicional (AND, OR). As operações com conjuntos *fuzzy* ( $A$  e  $B$ ) são definidas através da operação de suas funções de pertinência ( $\mu_A(x)$  e  $\mu_B(x)$ ). Elas determinam a compatibilidade entre cada conjunto *fuzzy* e seu valor no domínio [Cox 1994].

As operações básicas e as propriedades para conjuntos *fuzzy* foram definidas originalmente por ZADEH (1978). Por exemplo, as operações básicas definidas para conjuntos *fuzzy*:

$$\text{União: } A \cup B = \{(x, \max(\mu_A(x); \mu_B(x))) \mid x \in U\} \quad (6)$$

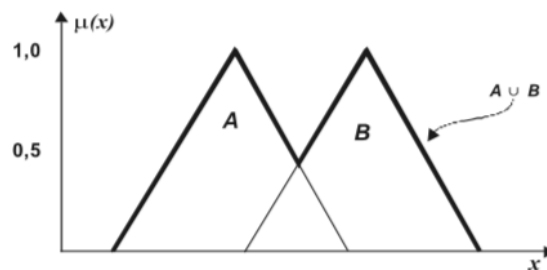


Figura 2: União de Conjuntos Fuzzy

$$\text{Intersecção: } A \cap B = \{(x, \min(\mu_A(x); \mu_B(x))) \mid x \in U\} \quad (7)$$

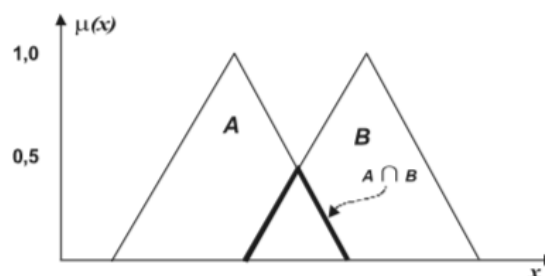


Figura 3: Intersecção de Conjuntos Fuzzy

$$\text{Complemento: } \neg A = \{(x, \mu_{\neg A}(x)) \mid \mu_{\neg A}(x) = 1 - \mu_A(x)\} \quad (8)$$

onde  $x \in U$ ,  $\mu_A$  é o grau de pertinência para o conjunto *fuzzy*  $A$ .

Nesta seção foram apresentadas as noções básicas sobre a teoria dos conjuntos *fuzzy*. Na seção 2.5 serão extendidas para *fuzzy* intervalar e a seguir será descrita a Lógica Fuzzy,

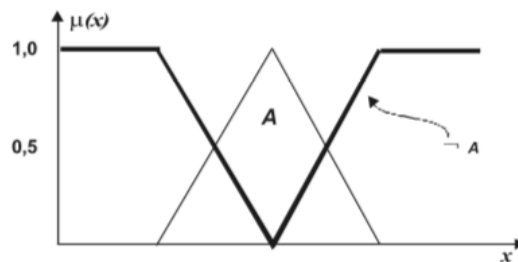


Figura 4: Complemento de Conjuntos Fuzzy

apresentando as variáveis linguísticas, o raciocínio aproximado e o sistema de inferência *fuzzy*.

A Lógica Fuzzy, é baseada na Teoria dos Conjuntos Fuzzy, ela é a lógica que serve como base para a modelos de raciocínio aproximados, e o seu conhecimento é fundamental para entendimento dos mecanismos de inferência que serão propostos no decorrer desta dissertação. Com uma abordagem diferente da lógica clássica que requer uma especialização do sistema, fórmulas e equações exatas e valores numéricos precisos, a lógica *fuzzy* permite uma forma alternativa de pensar, que auxilia a modelar sistemas complexos usando altos níveis de abstração, estando desta forma, mais próxima da experiência e conhecimento do especialista.

Com a lógica *fuzzy* é possível expressar conceitos subjetivos, como "próximo de", "muito distante", que ao mesmo tempo serão mapeados dentro de uma classificação numérica exata.

Na lógica *fuzzy* também temos os conectivos lógicos AND, OR e NOT, são generalizados para o intervalo  $[0,1]$  pelas normas triangulares (t-normas), conormas triangulares (t-conormas) e negações fuzzy, respectivamente (KLEMENT et al, 2000).

Uma das partes da lógica *fuzzy* é o tratamento de variáveis linguísticas, cujos valores são palavras ou sentenças em linguagem natural, ou seja, são variáveis cujos valores são nomes de conjuntos *fuzzy*. Para de exemplificar o conceito de variável linguística considere a palavra *altura*, um substantivo, em linguagem natural; é uma aferição subjetiva dependente da observação quem afere, da média regional ou da precisão de um instrumento; ela não pode ser caracterizada precisamente; se trabalharmos com conjuntos *fuzzy*, pode-se descrever aproximadamente *altura* no universo  $U = [100, 200]$  (cm). *Altura* é uma variável linguística constituída de conjuntos *fuzzy* como: *muito baixa*, *baixa*, *média*, *alta*, *muita alta*, adjetivos. Essas expressões são chamadas *termos* da variável linguística *altura*. A definição de cada termo (conjunto) é feita por uma função de pertinência.

**Definição 3** *Uma variável linguística X no universo U é uma variável cujos valores assumidos são subconjuntos fuzzy de U.*

As variáveis linguísticas são de grande importância em aplicações. Como exemplos, parâmetros de sistemas especializados como velocidade, temperatura, largura, também

podem ser entendidos como variáveis linguísticas. Sendo assim, as variáveis linguísticas são como o universo e os termos o subconjunto *fuzzy* deste universo (KLEMENT et al, 2000).

### 2.1.2 Raciocínio Aproximado

Os sistemas baseados em regras possuem um mecanismo de inferência para gerar resultados a partir das entradas fornecidas e das regras armazenadas em sua base de conhecimento. Os sistemas *fuzzy* possuem procedimentos de inferência também chamados de raciocínio aproximado ou raciocínio *fuzzy* [Jang 1997]. O raciocínio aproximado relaciona-se ao processo de modelagem que utiliza os conjuntos *fuzzy* e a lógica *fuzzy*, onde se pode obter conclusões a partir de premissas incertas (KANDEL et al, 1996).

A inferência *fuzzy* utiliza uma generalização da inferência modus ponens (GMP) e uma regra de inferência composicional, assim a inferência *fuzzy* é definida como

$$\begin{aligned} R : x \text{ é } A &\rightarrow y \text{ é } B \\ x \text{ é } A' \\ y \text{ é } B' \end{aligned}$$

Logo,

$$B' = A' \circ R(x, y) = A' \circ (A \rightarrow B) \quad (9)$$

onde  $A$ ,  $A'$ ,  $B$ , e  $B'$  são conjuntos *fuzzy*,  $x$  e  $y$  são variáveis *fuzzy*,  $R(x, y)$  é a relação binária *fuzzy* de implicação e  $\circ$  é o operador da relação de composição. O conjunto *fuzzy*  $B'$  também chamado de região *fuzzy* solução  $B'$ .

O conceito matemático de relação *fuzzy* é formalizado a partir do produto cartesiano usual entre conjuntos, estendendo a função característica de uma relação por uma função de pertinência.

**Definição 4** Uma relação *fuzzy*  $R$  sobre  $U_1 \times U_2 \times \dots \times U_n$  é qualquer subconjunto *fuzzy* de  $U_1 \times U_2 \times \dots \times U_n$ . assim, uma relação *fuzzy*  $R$  é definida por uma função de pertinência  $\mu_R : U_1 \times U_2 \times \dots \times U_n \rightarrow [0, 1]$ .

Com essa definição na relação *fuzzy*  $R$  indicada por  $\mu_R$ , o número  $\mu_R(x_1 \times x_2 \times \dots \times x_n)$  determina o grau com que os elementos  $x_i$ , que compõem a  $n$ -upla  $(x_1, x_2, \dots, x_n)$ , estão relacionados segundo a relação  $R$ .

**Definição 5** O produto cartesiano *fuzzy* dos subconjuntos  $A_1, A_2, \dots, A_n$  de  $U_1, U_2, \dots, U_n$ , respectivamente, é a relação *fuzzy*  $A_1 \times A_2 \times \dots \times A_n$ , cuja função de pertinência é dada por

$$\mu_{A_1 \times A_2 \times \dots \times A_n}(x_1, x_2, \dots, x_n) = \mu_{A_1}(x_1) \wedge \mu_{A_2}(x_2) \wedge \dots \wedge \mu_{A_n}(x_n), \quad (10)$$

em que  $\wedge$  representa o mínimo.



Existem várias formas de representar relações, geralmente são utilizadas três formas, por listagem de todos os pares *fuzzy*, de forma tabular e através de matrizes (TSOUKALAS, 1997) (SILVEIRA, 2002).

Suponha a relação *fuzzy* binária  $R$  definida em  $X$  e  $Y$ . Os pares da relação podem ser listados pela equação:

$$R = \{((x, y), \mu_R(x, y)) \mid x \in X \text{ e } y \in Y\} \quad (11)$$

Para um produto cartesiano discreto tem-se a equação:

$$R = \sum_{(x_i, y_j) \in X \times Y} \mu_R(x_i, y_j) / (x_i, y_j) \quad (12)$$

Já para produtos cartesianos contínuos tem-se:

$$R = \int_{X \times Y} \mu_R(x, y) / (x, y) \quad (13)$$

A combinação de relações *fuzzy* definidas nos produtos cartesianos são realizadas através de composição. Dado duas relações *fuzzy*, uma em  $X \times Y$  e outra em  $Y \times Z$ , quer-se relacionar diretamente elementos de  $X$  com  $Z$ . O conjunto  $Y$  serve como relação entre os dois conjuntos. A principal função na composição é computar os graus de pertinência dos pares  $(x, z)$  na relação composta, indicado por  $\mu(x, z)$ . As regras *fuzzy* (RF) são matematicamente equivalentes às relações *fuzzy* e um mecanismo de inferência é matematicamente equivalente a uma composição. São vários os tipos de composição, a mais comumente usada para computação e engenharia é a composição *max-min*, em SILVEIRA (2002) e DE BARROS; BASSANEZI (2010) são apresentadas outros tipos.

Sejam as relações  $R_1$  e  $R_2$  definidas no produto cartesiano  $X \times Y$  e  $Y \times Z$  respectivamente. A composição Max-Min de  $R_1$  e  $R_2$  é uma nova relação  $R_1 \otimes R_2$ , com  $\otimes \in \{\circ\}$ , definida em  $X \times Z$  como apresentado

$$R_1 \otimes R_2 = \int_{X \times Z} \vee [\mu_{R_1}(x, y) \wedge \mu_{R_2}(y, z)] / (x, z)$$

e a partir desse operador pode ser gerada a função de pertinência

$$\mu_{R_1 \otimes R_2}(x, z) = \vee [\mu_{R_1}(x, y) \wedge \mu_{R_2}(y, z)]$$

### 2.1.3 Sistema de Inferência Fuzzy

Existem diversas formas de relações de implicação *fuzzy* na literatura (TSOUKALAS, 1997) (SILVEIRA, 2002) (DE BARROS; BASSANEZI, 2010). As relações de implicação são obtidas através de diferentes operadores de implicação  $\phi$ . As informações

dos antecedentes e consequentes das regras são entradas para cada implicação  $\phi$ , e sua saída é uma relação de implicação.

Seja a regra genérica abaixo

**Se  $x$  é  $A$  então  $y$  é  $B$ .**

A forma básica de representar a regra acima é através da relação de implicação

$$R(x, y) = \int_{(x,y)} \mu_R(x, y)/x, y \quad (14)$$

ou em um universo discreto

$$R(x_i, y_j) = \sum_{(x_i, y_j)} \mu_R(x_i, y_j)/x_i, y_j \quad (15)$$

onde  $\mu(x, y)$  é a função de pertinência da relação de implicação.

Na regra acima um operador de implicação  $\phi$  usa como entrada partes dos antecedentes e consequentes das funções de pertinência, chamados  $\mu_A(x)$  e  $\mu_B(y)$ , restando como saída  $\mu(x, y)$ , dado por:

$$\mu(x, y) = f[\mu_A(x), \mu_B(y)] \quad (16)$$

Quando se possui uma coleção de regras, elas são ligadas por conectivos SENÃO (ELSE), que podem interpretar uma intersecção ou uma união, dependendo do operador de implicação utilizado (DUBOIS et al, 2000).

Um método de inferência *fuzzy* pode ser descrito na regra genérica abaixo:

$$R_i: \text{Se } x_1 \text{ é } A_{1i} \text{ e } x_2 \text{ é } A_{2i} \dots \text{ e } x_{ki} \text{ é } A_{ki} \text{ então } y \text{ é } B_i$$

É baseada na suposição que uma regra que possui  $k$  elementos de condições é decomposta em  $k$  implicações  $A_{ji} \rightarrow B_i$  para  $(j = 1, 2, \dots, k)$ . Cada implicação é usada separadamente para inferir o valor de  $B'_i$  através da aplicação da regra composicional:

$$B'_i = A'_{ji} \circ (A_{ji} \rightarrow B_i) \quad (17)$$

para  $(j = 1, 2, \dots, k)$ . Os valores  $B'_i$  são agregados por um dos operadores de agregação (normalmente AND, OR). O método de inferência *fuzzy* decomposicional é ilustrado na figura 5 sobre as duas regras abaixo:

$$R_1: \text{Se } x_1 \text{ é } A_{11} \text{ e } x_2 \text{ é } A_{21} \text{ então } y \text{ é } B_1$$

$$R_2: \text{Se } x_1 \text{ é } A_{12} \text{ e } x_2 \text{ é } A_{22} \text{ então } y \text{ é } B_2$$

Com base nessas informações, é possível definir sistemas de inferência *fuzzy* como a arquitetura apresentada na figura 6.

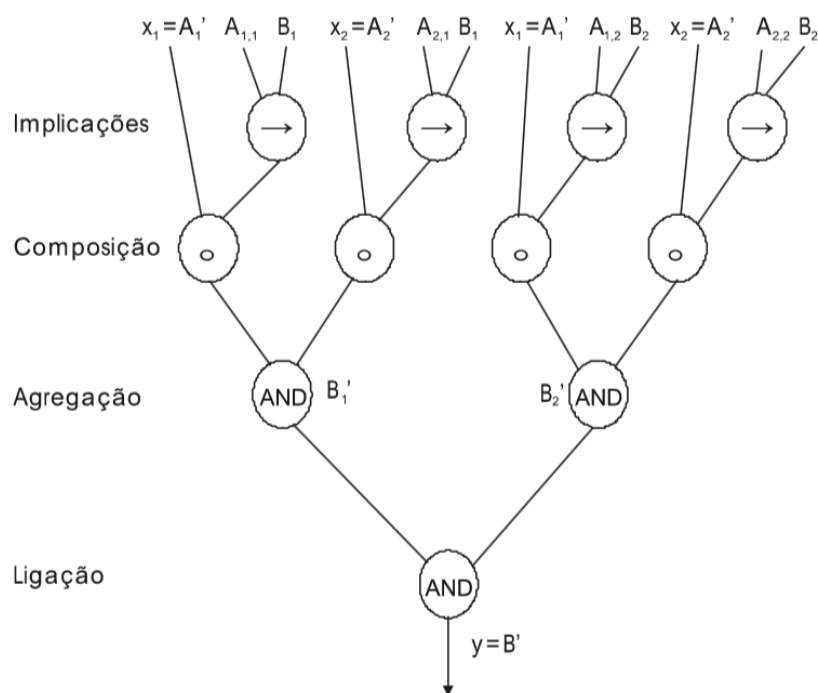


Figura 5: Esquema da Inferência Sobre Regras *Fuzzy* (Kasabov 1996)

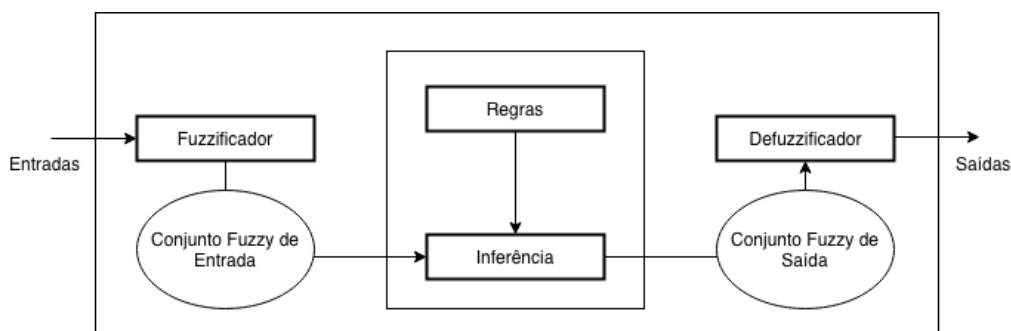


Figura 6: Sistema de Inferência Fuzzy

Os dados de entrada do sistema podem ser valores precisos, resultados de observações ou medições, por isso é necessário realizar um mapeamento destes dados para os conjuntos *fuzzy* de entrada, este estágio é chamado *fuzzificação*, é neste estágio que são ativadas as regras relevantes para uma dada situação. No próximo estágio, através do processamento de *inferência* é obtido o conjunto *fuzzy* de saída, e no estágio de *defuzzificação* é efetuada a interpretação dessa informação.

Para a etapa de *fuzzificação* são construídos os CF apropriados, na modelagem das funções de pertinência, através das quais o significado dos termos linguísticos relevantes são capturados, dependentes do contexto no qual os termos são usados. E somente as regras relevantes para uma dada situação são ativadas durante a etapa de *fuzzificação*. Os valores de entrada do sistema são mapeados pelos conjuntos *fuzzy* relevantes ao problema em questão, atribuindo a estes valores graus de pertinência para cada conjunto *fuzzy* ou

variável linguística.

Com os conjuntos *fuzzy* e as variáveis de entrada, as regras serão tratadas pelo mecanismo de inferência, resultando conjuntos *fuzzy* solução que serão tratados pelos mecanismos de defuzzificação apresentados a seguir.

A inferência é uma forma de simular o raciocínio dos especialistas através de regras. Como visto anteriormente, isto é feito através da composição de uma relação  $R$  representando uma regra de implicação e uma proposição  $A$  qualquer, e ligação das regras *fuzzy* através das operações AND e OR. Ou seja, o processo de inferência resulta em inferir novos fatos baseados nas regras *fuzzy* e nas informações de entradas adicionais.

Depois que todas as regras de inferência forem executadas, o resultado, um conjunto *fuzzy* solução de saída  $\mu_{OUT}(x)$ , deve ser defuzzificado para obtermos uma saída *crisp*. A escolha do método de defuzzificação pode ter um impacto significativo na velocidade e acuracidade do sistema *fuzzy*. Os métodos mais frequentes são, centroid, centro de soma e a média de máximo (DE BARROS; BASSANEZI, 2010).

$$x^* = \frac{\sum x_i \mu_{OUT}(x_i)}{\sum \mu_{OUT}(x_i)} \quad (18)$$

$$x^* = \frac{\sum x_i \sum \mu_{B'_K}(x_i)}{\sum \sum \mu_{B'_K}(x_i)} \quad (19)$$

onde  $\mu_{B'_K}$  é a função de pertinência, no ponto  $x_j$ , resultando da varredura da  $k$ -ésima regra.

$$x^* = \frac{\sum x_m}{M}, \quad (20)$$

onde  $x$  é o  $m$ -ésimo elemento e a função  $\mu_{OUT}(x)$  está no valor máximo e  $M$  é o total de elementos.

Com tudo que foi exposto nesta seção, pode-se dizer que os sistemas *fuzzy* são normalmente utilizados em várias áreas, principalmente para desenvolvimento de controladores *fuzzy*. Os Sistemas Fuzzy são um bom método para resolver problemas complexos e não-lineares, porém, existem algumas limitações. Os SF não são capazes de aprender novas regras, e a definição de boas regras e funções de pertinência não são tarefas fáceis. Para auxiliar na solução destas limitações serão apresentadas nas próximas seções, a combinação de sistemas *fuzzy* com outras técnicas como, sistemas de classificação e aritmética intervalar.

## 2.2 Função de Agregação

Um conceito crucial para a presente dissertação é o da função de agregação. O processo de combinar diversos valores numéricos em um único valor, que de alguma forma represente todos eles é chamado de agregação e a função numérica que realiza este pro-

cesso é denominada de função de agregação. e elas desempenham um papel importante no contexto dos SCBRF(I), uma vez que são usadas para obter um único valor de saída a partir de vários valores de entrada (BELIAKOV et al, 2007). As funções de agregação também são usadas em qualquer outra aplicação, como reconhecimento de padrões, processamento de imagem e tomada de decisão. Mas quando esses valores são graus de pertinência ou valores verdades num contexto *fuzzy* (ou de alguma de suas extensões) essas funções ou operadores de agregação têm que satisfazer certas propriedades (MAYOR; TRILLAS, 1986).

**Definição 6** (BELIAKOV et al, 2007; MAYOR; TRILLAS, 1986) *Uma função de agregação  $n$ -ária é um mapeamento  $A : [0, 1]^n \rightarrow [0, 1]$  satisfazendo as propriedades que seguem:*

(A1) *A é crescente<sup>1</sup> em cada argumento: para cada  $i \in \{1, \dots, n\}$ , se  $x_i \leq y$ , então*

$$A(x_1, \dots, x_n) \leq A(x_1, \dots, x_{i-1}, y, x_{i+1}, \dots, x_n);$$

(A2) *As condições de fronteira:  $A(0, \dots, 0) = 0$  e  $A(1, \dots, 1) = 1$ .*

**Definição 7** (BELIAKOV et al, 2007; MAYOR; TRILLAS, 1986) *Uma função  $F : [0, 1]^n \rightarrow \mathbb{R}$  é idempotente se, para cada  $x \in [0, 1]$  tem-se que*

$$F(x, \dots, x) = x.$$

Se considerarmos que cada um dos graus de pertinência a ser agregado, são resultados das opiniões de diversos especialistas ou da aplicação de diversos métodos, então quando esses graus são todos zero ou todos um, o resultado da agregação também deve ser zero ou um, respectivamente.

**Definição 8** (BELIAKOV et al, 2007) *Uma função de agregação  $n$ -ária  $A : [0, 1]^n \rightarrow [0, 1]$  é dita 0-postiva se também satisfizer a propriedade que segue:*

(A3) *As condições de fronteira para 0:  $A(x_1, \dots, x_n) = 0$  se e somente se  $x_i = 0$ , para todo  $i \in \{1, \dots, n\}$ .*

*Uma função de agregação  $n$ -ária  $A : [0, 1]^n \rightarrow [0, 1]$  é dita 1-positiva se também satisfizer a propriedade que segue:*

(A4) *A condição de fronteira para 1:  $A(x_1, \dots, x_n) = 1$  se e somente se  $x_i = 1$ , para todo  $i \in \{1, \dots, n\}$ .*

---

<sup>1</sup>Foi considerado que uma função crescente pode não ser estritamente crescente (e, analogamente, para as funções decrescentes).

As funções de agregações são classificadas em conjuntivas (o resultado sempre é menor ou igual ao menor dos graus de pertinências), disjuntiva (o resultado sempre é maior ou igual ao menor dos graus de pertinências), média (o resultado sempre é um valor que está entre o menor e o maior grau de pertinência de entrada), e híbrida (quando não é nem conjuntiva, nem disjuntiva e nem uma média).

**Definição 9** (BELIAKOV et al, 2007) *Uma função de agregação  $f : [0, 1]^n \rightarrow [0, 1]$  é dito que é uma média se for limitado pelo mínimo e máximo de seus argumentos, isto é, para todos  $(x_1, \dots, x_n) \in [0, 1]^n$ , isso sustenta que:*

$$\min\{x_1, \dots, x_n\} \leq f(x_1, \dots, x_n) \leq \max\{x_1, \dots, x_n\}.$$

Devido à monotonicidade das funções de agregação  $f$ , o comportamento de uma média é equivalente à propriedade de idempotência. Ou seja,  $f$  é uma média se e somente se  $f$  é idempotente.

**Definição 10** (LOWEN, 1996) *Uma t-norma é uma função de agregação bivariada  $T : [0, 1]^2 \rightarrow [0, 1]$  satisfazendo as propriedades que seguem, para todo  $x, y, z \in [0, 1]$ :*

(T1) *Comutatividade:*  $T(x, y) = T(y, x)$ ;

(T2) *Associatividade:*  $T(x, T(y, z)) = T(T(x, y), z)$ ;

(T3) *Condições Limites:*  $T(x, 1) = x$ .

Um elemento  $x \in ]0, 1]$  é um divisor zero não-trivial de  $T$  se existe  $y \in ]0, 1]$  tal que  $T(x, y) = 0$ . Uma t-norma é positiva se e somente se não tem divisores zero não triviais, isto é, se  $T(x, y) = 0$  então  $x = 0$  ou  $y = 0$ . Alguns exemplos de t-normas positivas e contínuas são

$$\begin{aligned} T_M(x, y) &= \min\{x, y\}, \\ T_P(x, y) &= xy. \end{aligned}$$

que são as t-normas do mínimo e do produto, respectivamente. Devido à propriedade associativa, trivialmente, pode-se definir t-normas n-ária. (KLEMENT et al, 2000)

Entre as classes de funções de agregação, os chamados operadores de Média Ponderada Ordenada (OWA) são um caso muito relevante. Esses operadores foram definidos por Yager (YAGER, 1993) da seguinte maneira:

**Definição 11** (BEDREGAL et al, 2017) *Seja  $w = (w_1, \dots, w_n) \in [0, 1]^n$  um vetor de pesos (ex.,  $w_i \in [0, 1]$  e  $\sum_{i=1}^n w_i = 1$ ). Um operador OWA de dimensão  $n$  associado ao*

vetor de peso  $w$  é uma função OWA :  $[0, 1]^n \rightarrow [0, 1]$  definida por

$$OWA(x_1, \dots, x_n) = \sum_{i=1}^n w_i x_{(i)}$$

onde  $(.)$  denota a permutação of  $\{1, \dots, n\}$  de modo que  $x_{(1)} \geq x_{(2)} \geq \dots \geq x_{(n)}$ .

### 2.2.1 Função de Overlap

As funções de *overlap* generalizam operadores de interseção, como o mínimo ou, em geral, t-normas. Por outro lado, observe que uma função de *overlap* (sobreposição) é uma instância particular da função de agregação sem divisores de zero ou divisores de um. Vamos denotar por  $O$  o conjunto de funções de overlap no sentido da Definição 12.

**Definição 12** (BUSTINCE et al, 2010) Uma função overlap é uma função bivariada  $O: [0, 1]^2 \rightarrow [0, 1]$  satisfazendo as propriedades que seguem, para todo  $x, y \in [0, 1]$ :

- (O1)  $O$  é comutativa:  $O(x, y) = O(y, x)$ ;
- (O2)  $O(x, y) = 0$  se e somente se  $x = 0$  ou  $y = 0$ ;
- (O3)  $O(x, y) = 1$  se e somente se  $x = y = 1$ ;
- (O4)  $O$  é crescente;
- (O5)  $O$  é contínuo.

Uma função *overlap*  $O$  é associativa se e somente se  $O$  for uma t-norma contínua e positiva (veja BUSTINCE et al (2010)).

Exemplos de funções *overlap* estão presentes na Tabela 1 (conforme BUSTINCE et al (2010, 2012); JURIO et al (2013); BEDREGAL et al (2013); DIMURO; BEDREGAL (2014); DIMURO et al (2016); DIMURO; BEDREGAL (2015); DIMURO et al (2014,?); DIMURO; BEDREGAL (2015)). Observe que  $T_M$  e  $T_P$  também são t-normas.

**Proposição 1** GARCIA-JIMENEZ et al (2015, Proposição 5) Seja  $O: [0, 1]^2 \rightarrow [0, 1]$  uma função overlap e  $T: [0, 1]^n \rightarrow [0, 1]$  uma t-norma n-ária. Então, tem-se que

$$O(x, T(y_1, \dots, y_n)) = T(O(x, y_1), \dots, O(x, y_n))$$

se e somente se  $T = \min$ .

Tabela 1: Exemplos de funções *overlap*

Funções <i>Overlap</i>
$T_M(x, y) = \min\{x, y\}$
$T_P(x, y) = xy$
$O_p(x, y) = x^p y^p, p > 0$ (em particular, $O_{\sqrt{\cdot}}(x, y) = \sqrt{xy}$ )
$O_{DB}(x, y) = \begin{cases} \frac{2xy}{x+y} & \text{se } x + y \neq 0 \\ 0 & \text{se } x + y = 0 \end{cases}$
$O_2^V(x, y) = \begin{cases} \frac{1+(2x-1)^2(2y-1)^2}{2} & \text{se } x, y \in ]0.5, 1]; \\ \min\{x, y\} & \text{caso contrário.} \end{cases}$
$O_{m\frac{1}{2}}(x, y) = \min\{\sqrt{x}, \sqrt{y}\}$
$O_{mM}(x, y) = \min\{x, y\} \max\{x^2, y^2\}$
$O_k(x, y) = \min\{x^k y, x y^k\}$ com $k \in ]0, 1[$
$O_{rat}(x, y) = \frac{\sqrt{xy}}{\sqrt{xy}+1-xy}$

### 2.2.2 Índices de *Overlap*

Os índices de *overlap* são usados para medir o grau de sobreposição entre dois conjuntos *fuzzy* e consiste na generalização do índice de consistência de Zadeh entre dois conjuntos *fuzzy* sobre o mesmo universo referencial.

Em 1982, Dubois e Prade (DUBOIS et al, 2000) apresentaram a seguinte axiomatização para o índice *overlap* (ou sobreposição).

**Definição 13** (DUBOIS; PRADE, 2000) *Um índice de overlap é uma função  $\mathcal{O} : CF(U) \times CF(U) \rightarrow [0, 1]$  de modo que, para todo  $A, B, C \in CF(U)$ , as seguintes condições:*

- (O1)  $\mathcal{O}(A, B) = 0$  se e somente se  $A$  e  $B$  têm suporte disjunto, isto é, para todo  $u \in U$ , tem-se que  $A(u)B(u) = 0$ ;
- (O2)  $\mathcal{O}(A, B) = \mathcal{O}(B, A)$ ;
- (O3) Se  $B \leq C$ , então  $\mathcal{O}(A, B) \leq \mathcal{O}(A, C)$ .

Um índice de *overlap*  $\mathcal{O}$  é dito normal sempre que a seguinte condição vale:

- (O4) Se existe  $u \in U$  tal que  $A(u) = B(u) = 1$ , então  $\mathcal{O}(A, B) = 1$ .

Várias definições de índices de *overlap* podem ser encontradas na literatura (ver, por exemplo: GARCIA-JIMENEZ et al (2015); BUSTINCE et al (2009); DUBOIS et al (2000); ZADEH (1978)). Abordagem proposta por BUSTINCE et al (2009), que também foi formalizada por GARCIA-JIMENEZ et al (2015), juntamente com um método para construir índices de *overlap* por meio de funções de *overlap* foi utilizada em ELKANO



Tabela 2: Exemplos de índices de *overlap* utilizados nesta dissertação

Índices de <i>Overlap</i>	
$\mathcal{O}_Z(A, B)$	$= \max_{u \in U} \min\{A(u), B(u)\}$
$\mathcal{O}_x(A, B)$	$= \begin{cases} 0 & \text{se } \forall u \in U : A(u)B(u) = 0 \\ x & \text{caso contrário,} \end{cases}$ para todo $x \in ]0, 1]$
$\mathcal{O}_\pi(A, B)$	$= \frac{1}{n} \sum_{u \in U} A(u) \cdot B(u)$ , onde $n = \text{card}(U)$

Tabela 3: Exemplos de índices de *overlap* construídos usando o Teorema 1

Função de <i>Overlap</i> $\mathcal{O}$	Função de Agregação $M$	Índice de <i>Overlap</i> $\mathcal{O}$
$\mathcal{O}_\sqrt{\phantom{x}}$	média aritmética	$\mathcal{O}_\sqrt{\phantom{x}}(A, U) = \frac{1}{n} \sum_{u \in U} \sqrt{A(u) \cdot 1}$
$\mathcal{O}_M$	máximo	$\mathcal{O}_Z(A, U) = \max_{u \in U} \min\{A(u), 1\}$ (ZADEH, 1978)
$\mathcal{O}_{rat}$	média aritmética	$\mathcal{O}_{rat}(A, U) = \frac{1}{n} \sum_{u \in U} \frac{\sqrt{A(u) \cdot 1}}{\sqrt{A(u) \cdot 1 + 1 - A(u)}}$

et al (2017) com o objetivo de definir a confiança e apoiar as medidas das regras de associação usando índices de *overlap*.

Alguns exemplos de índices de *overlap* estão presentes na Tabela 2. Observe que  $\mathcal{O}_Z$  é o índice de consistência de ZADEH (1978).  $\mathcal{O}_Z$  e  $\mathcal{O}_\pi$  são normais e  $\mathcal{O}_x$  não é normal para  $x \neq 1$ . (GARCIA-JIMENEZ et al, 2015)

**Teorema 1** (GARCIA-JIMENEZ et al, 2015) *Seja  $M : [0, 1]^n \rightarrow [0, 1]$  uma função de agregação 0-positiva e  $O : [0, 1]^2 \rightarrow [0, 1]$  uma função de overlap. Então, o mapeamento  $\mathcal{O} : CF(U) \times CF(U) \rightarrow [0, 1]$  definido, para todo  $A, B \in CF(U)$  e  $u_i \in U$ , com  $i = 1, \dots, n$ , como*

$$\mathcal{O}(A, B) = M(O(A(u_1), B(u_1)), \dots, O(A(u_n), B(u_n))) \quad (21)$$

*é um índice de overlap. Em contra partida, se  $O$  é uma função de overlap e  $M : [0, 1]^n \rightarrow [0, 1]$  é uma função de agregação tal que  $\mathcal{O}$ , definida pela Equação (21), é um índice de overlap, então  $M$  é 0-positiva.*

A Tabela 3 mostra alguns exemplos de índices de *overlap* construídos baseados no Teorema 1, usando algumas funções de sobreposição mostradas na Tabela 1.

### 2.3 Funções de Penalidade (*Penalty*)

Pode-se encontrar diferentes definições de funções de penalidade na literatura (ver, por exemplo YAGER (1993); YAGER; RYBALOV (1997); CALVO et al (2004); CALVO; BELIAKOV (2010); BUSTINCE et al (2014, 2011); BELIAKOV (2014); BELIAKOV; JAMES (2014); WILKIN; BELIAKOV (2015)).

**Definição 14** Uma função  $f : [0, 1] \rightarrow \mathbb{R}$  é convexa se para todo  $x, y \in [0, 1]$  e para todo  $\lambda \in [0, 1]$ , tem-se a desigualdade  $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$ .

**Definição 15** Uma função  $f : [0, 1] \rightarrow \mathbb{R}$  é quasi-convexa se para todo  $x, y \in [0, 1]$  e para todo  $\lambda \in [0, 1]$  tem-se a desigualdade  $f(\lambda x + (1 - \lambda)y) \leq \max\{f(x), f(y)\}$ .

**Proposição 2** BUSTINCE et al (2016, Proposição 2.2) Considere a função  $g : [0, 1] \rightarrow \mathbb{R}$ , (ou funções penalty):

(i) Se  $g$  é monotônica então  $g$  é quasi-convexa;

(ii) Se  $g$  é convexa então  $g$  é quasi-convexa;

(iii) Se  $g$  é convexa então  $g$  tem um minimizador em  $[0, 1]$ .

**Definição 16** (KURDILA; ZABARANKIN, 2005) Uma função  $f : [0, 1] \rightarrow \mathbb{R}$  é pouco semicontínua em  $x_0 \in [0, 1]$  se

$$\liminf_{x \rightarrow x_0} f(x) \geq f(x_0).$$

**Corolário 1** (KURDILA; ZABARANKIN, 2005, Corolário 2.1) Seja  $f : [0, 1] \rightarrow \mathbb{R}$  uma função semi-convexa e pouco semicontínua. Então, o conjunto de minimizadores de  $f$  é um conjunto não vazio conexo.

Nesta dissertação, considerando a discussão de Bustince et al. (BUSTINCE et al, 2016), decidimos adotar a seguinte definição:

**Definição 17** (BUSTINCE et al, 2016, Definição 4.1) Para qualquer intervalo fechado  $\mathbb{I} \subseteq \mathbb{R}$ , a função  $P : [0, 1]^{n+1} \rightarrow \mathbb{R}^+$  é uma função penalty se e somente se existir  $c \in \mathbb{R}^+$  tal que:

(P1)  $P(\vec{x}, y) \geq c$ , para todo  $\vec{x} \in [0, 1]^n, y \in [0, 1]$ ;

(P2)  $P(\vec{x}, y) = c$  se e somente se  $x_i = y$ , para todo  $i = 1 \dots n$ , e

(P3)  $P$  é quasi-convexo e semicontínuo em  $y$  para cada  $\vec{x} \in [0, 1]^n$ .

**Definição 18** (BUSTINCE et al, 2016, Definição 4.2) Seja  $P$  uma função penalty no sentido da Definição 17. A função  $f_P : [0, 1]^n \rightarrow [0, 1]$  é dita uma  $P$ -função, se, para cada  $\vec{x} \in [0, 1]^n$ , tem-se que

$$f_P(\vec{x}) = \frac{a + b}{2}, \quad (22)$$

onde  $[a, b] = cl(Minz(P(\vec{x}, \cdot)))$ , e  $Minz(P(\vec{x}, \cdot))$  é o conjunto minimizador de  $P(\vec{x}, \cdot)$ , isso é,

$$Minz(P(\vec{x}, \cdot)) = \{y \in [0, 1] \mid P(\vec{x}, y) \leq P(\vec{x}, z), \text{ para cada } z \in [0, 1]\},$$

e  $cl(S)$  é o fecho de  $S \subseteq [0, 1]$ .

**Teorema 2** (BUSTINCE et al, 2016, Teorema 4.1) *Uma função  $f : [0, 1]^n \rightarrow [0, 1]$  é uma  $P$ -função se e somente se  $f$  é idempotente.*

Então, qualquer função de agregação média pode ser representada por uma  $P$ -função.

A função *penalty*  $P$  descreve a dissimilaridade ou desacordo entre uma entrada arbitrária  $\vec{x}$  e um valor  $y$ . Então, a  $P$ -função  $f$  é uma função que minimiza a dissimilaridade escolhida.

Nesta dissertação, uma função *penalty* definida sobre um produto cartesiano de reticulados, introduzido por BUSTINCE et al (2014).

**Teorema 3** (BUSTINCE et al, 2016) *Seja  $K_i : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ , com  $i = 1, \dots, n$ , funções semi-contínuas pouca quasi-convexa com um mínimo exclusivo em  $K_i(0) = 0$ , e  $\mathcal{D} : CF(U) \times CF(U) \rightarrow \mathbb{R}^+$  seja a distância entre conjuntos fuzzy, definidos, para todos  $X, Y \in CF(U)$ , por*

$$\mathcal{D}(A, B) = \sum_{i=1}^n | A(u_i) - B(u_i) |, \quad (23)$$

onde  $n = \text{card}(U)$ . Então o mapeamento  $P_{\nabla} : CF(U)^m \times \mathcal{L}_m^* \rightarrow \mathbb{R}^+$ , dado, para todo  $\vec{A} \in CF(U)^m, Y \in \mathcal{L}_m^*$ , por :

$$P_{\nabla}(\vec{A}, Y) = \sum_{q=1}^m K_q(\mathcal{D}(A_q, B_{y_q})) = \sum_{q=1}^m K_q \left( \sum_{p=1}^n | A_q(u_p) - y_q | \right) \quad (24)$$

é uma função de *penalty* definida sobre o produto cartesiano de reticulados (lattices)  $\mathcal{L}_m^{*(n+1)}$ .

## 2.4 Teoria Intervalar

Na computação científica a qualidade do resultado na computação dos dados é dependente do conhecimento e do controle dos erros. Para resolver essa questão surgiu a matemática intervalar baseada na aritmética de MOORE (1979). Existem três tipos de fontes de erros em computação numérica clássica (a qual representa números reais como ponto flutuante): a propagação do erro nos dados iniciais, os arredondamentos e o erro de truncamento. A matemática intervalar busca resolver esse problema que se concentra fundamentalmente em dois aspectos:

- Na criação de um modelo computacional que exprima o controle e a análise dos erros que ocorrem no processo computacional;
- Na escolha de técnicas de programação adequadas para desenvolvimento de softwares científicos buscando minimizar os erros nos resultados.

Para lidar com problemas comuns em computação numérica, Moore 1966, elaborou uma análise intervalar em que cada valor real é tratado como um intervalo real fechado, equação (25), obtendo a partir daí, topologias, uma aritmética, relações, etc..

$$[a, b] = \{x \in \mathbb{R} \mid a \leq x \leq b\} \quad (25)$$

Para construir uma Teoria de Representação Fuzzy Intervalar, é também necessário um mecanismo de transformação, ou melhor, uma forma de representar entidades reais (cujos elementos pertencem aos  $\mathbb{R}$ ) em entidades intervalares. A classe de representações intervalares canônicas está associada aos algoritmos intervalares que satisfazem as propriedades apresentadas em (T. Hickey et al 2001), entre elas a optimalidade, que retornam intervalos corretos e de menor abrangência possível. Então seja  $f : \mathbb{R} \rightarrow \mathbb{R}$ , se  $f$  é uma função real total não-assintótica, se para algum intervalo  $[a, b]$ , o conjunto  $\{f(x') \mid a \leq x' \leq b\}$  não tem um supremo nem um ínfimo, logo a função que melhor define é

$$RIC(f)([a, b]) = [\min f([a, b]), \max f([a, b])]; \quad (26)$$

e é chamada de representação intervalar canônica (RIC) de  $f$ . Desta forma,  $RIC(f)$  é uma função que mapeia cada intervalo  $[a, b]$  no menor intervalo contendo  $f([a, b])$  (BE-DREGAL; TAKAHASHI, 2006).

### 2.4.1 Aritmética Intervalar

Assim como a teoria dos conjuntos *fuzzy*, a teoria dos intervalos possui uma aritmética bem definida matematicamente onde são definidas as principais operações aritméticas para intervalos, baseadas nas respectivas operações aritméticas reais sobre os extremos dos intervalos (MOORE, 1979).

Seja  $\mathbb{R}$  o conjunto dos números reais, e sejam  $a, b \in \mathbb{R}$ , tais que  $a \leq b$ . Então o conjunto  $\{x \in \mathbb{R} \mid a \leq x \leq b\}$  é um intervalo de reais ou simplesmente um intervalo, que será denotado por:  $X = [a, b] = \{x \in \mathbb{R} \mid a \leq x \leq b\}$ . Os pontos do conjunto dos intervalos de reais serão denotados por letras latinas maiúsculas, tais como X, Y, Z, ...

Sejam  $[\underline{x}, \bar{x}]$  e  $[\underline{y}, \bar{y}]$  dois intervalos em  $\mathbb{R}^+$  de modo que  $x \leq_L y$ , as regras da aritmética intervalar são as seguintes (SANZ et al., 2013):

- Adição:  $[\underline{x}, \bar{x}] + [\underline{y}, \bar{y}] = [\underline{x} + \underline{y}, \bar{x} + \bar{y}]$
- Subtração:  $[\underline{x}, \bar{x}] - [\underline{y}, \bar{y}] = [\underline{y} - \bar{x}, \bar{y} - \underline{x}]$
- Multiplicação:  $[\underline{x}, \bar{x}] * [\underline{y}, \bar{y}] = [\underline{x} * \underline{y}, \bar{x} * \bar{y}]$
- Divisão:  $[\underline{x}, \bar{x}] / [\underline{y}, \bar{y}] = [\min(\min(\underline{x} + \underline{y}), 1), \bar{x} + \bar{y}]$

**Definição 19** Seja  $[\underline{x}, \bar{x}]$  um intervalo. O ponto médio de  $[\underline{x}, \bar{x}]$  é dado pelo número real  $m([\underline{x}, \bar{x}]) = \frac{\underline{x} + \bar{x}}{2}$ .

### 2.4.2 Ordem e Ordem Admissível

Seja  $A$  um conjunto não-vazio e  $\leq$  uma relação sobre  $A$ . Uma ordem parcial é um par  $(A, \leq)$ , onde a relação  $\leq$  satisfaz as seguintes propriedades, para todo  $x, y, z \in A$ .

- $x \leq y$  (reflexividade)
- $x \leq y$  e  $y \leq z \Rightarrow x \leq z$  (transitividade)
- $x \leq y$  e  $y \leq x \Rightarrow x = y$  (anti-simetria)

Escreve-se  $x < y$  se  $x \leq y$  e  $x \neq y$ . Diz-se, também que  $A$  é um conjunto parcialmente ordenado.

Seja  $(L, \preceq)$  seja um conjunto limitado ordenado parcialmente (*poset*) com menor elemento  $0_L$  e maior elemento  $1_L$ . Um mapeamento  $M : L([0, 1])^n \rightarrow L([0, 1])$  é uma função de agregação  $n$ -ária ( $n \in \mathbb{N}, n \geq 2$ ) em  $(L([0, 1]), \preceq)$  se é  $\preceq$ -decrecente, ou seja, para todo  $\mathbf{x}, \mathbf{y} \in L([0, 1])^n$ ,  $M(\mathbf{x}) \preceq M(\mathbf{y})$  sempre que  $x_1 \preceq y_1, \dots, x_n \preceq y_n$  e satisfaz as condições de fronteira  $M(0_L, \dots, 0_L) = 0_L$ ,  $M(1_L, \dots, 1_L) = 1_L$  (MESIAR; NOVÁK, 1997).

Uma ordem parcial  $(A, \leq)$  diz-se total (ou  $A$  é totalmente ordenado) se todos os elementos de  $A$  são comparáveis, isto é, para todo  $x, y \in A$ ,  $x \leq y$  ou  $y \leq x$ .

Quando precisarmos usar uma relação de ordem total para intervalos, ou seja, quando a maior associação de intervalo precisar ser determinada, usaremos a definida por Xu e Yager em (XU; YAGER, 2006): Seja  $[\underline{x}, \bar{x}], [\underline{y}, \bar{y}] \in L([0, 1])$ , e seja  $s([\underline{x}, \bar{x}]) = \underline{x} + \bar{x} - 1$  seja o conjunto  $[\underline{x}, \bar{x}]$ , e  $h([\underline{x}, \bar{x}]) = 1 - (\bar{x} - \underline{x})$  seja o grau de acurácia de  $[\underline{x}, \bar{x}]$ . Então

- 1) se  $s([\underline{x}, \bar{x}])s([\underline{y}, \bar{y}])$ , então  $[\underline{x}, \bar{x}] < [\underline{y}, \bar{y}]$ ;
- 2) se  $s([\underline{x}, \bar{x}]) = s([\underline{y}, \bar{y}])$ , então
  - a) se  $h([\underline{x}, \bar{x}]) = h([\underline{y}, \bar{y}])$ , então  $[\underline{x}, \bar{x}] = [\underline{y}, \bar{y}]$ ;
  - b) se  $h([\underline{x}, \bar{x}]) < h([\underline{y}, \bar{y}])$ , então  $[\underline{x}, \bar{x}] < [\underline{y}, \bar{y}]$ .

Em BUSTINCE et al (2012), a noção de ordens admissíveis em  $\mathbb{I}$  foi introduzida e investigada. Os autores estabeleceram que uma relação binária  $\preceq$  em  $L([0, 1])$  é uma ordem admissível se for uma ordem linear em  $L([0, 1])$  refinando  $\leq_L$ , ou seja, se para todos  $[a, b], [c, d] \in L([0, 1])$  de modo que  $[a, b] \leq_L [c, d]$  então também é  $[a, b] \preceq [c, d]$ . O uso de ordens admissíveis nos permite comparar intervalos usando ordens totais entre eles. Em seguida, são apresentados alguns exemplos de ordens admissíveis.

Exemplo: Seja  $[a, b], [c, d] \in L([0, 1])$ :

- $[a, b] \preceq_{L1} [c, d] \Leftrightarrow a < c$  ou  $(a = c$  e  $b \leq d)$ ;
- $[a, b] \preceq_{L2} [c, d] \Leftrightarrow b < d$  ou  $(b = d$  e  $a \leq c)$ ;

- $[a, b] \preceq_{XY} [c, d] \Leftrightarrow a + b < c + d$  ou  $(a + b = c + d \text{ e } b - a \leq d - c)$  (definido por Xu e Yager em XU; YAGER (2006)).

## 2.5 Teoria Fuzzy Intervalar

Os conjuntos *fuzzy* intervalares nos permitem lidar não apenas com informações vagas (falta de limites entre classes afins), mas também com incerteza (falta de informação). O objetivo desta seção é apresentar alguns conceitos básicos sobre conjuntos *fuzzy* intervalares.

### 2.5.1 Teoria dos Conjuntos Fuzzy Intervalar

A teoria dos conjuntos fuzzy já se consolidou como uma ferramenta útil para modelar incerteza. Porém, muitas vezes existe dificuldade para se estipular os graus de pertinência a serem aplicados para determinados problemas. Com intuito de contornar essa situação, vários autores, como (MOORE, 1979), (SILVEIRA, 2002), (BEDREGAL; TAKAHASHI, 2006), representam dos graus de pertinência através de intervalos reais contidos em  $[0, 1]$ , expandindo assim conjuntos fuzzy para conjuntos fuzzy intervalares. Nas configurações fuzzy, há situações em que os especialistas têm problemas para construir os graus de pertinência dos elementos em relação aos conjuntos *fuzzy*. Quando este é o caso, é aconselhável usar extensões de conjuntos *fuzzy*. Uma das extensões mais utilizadas é a dos conjuntos fuzzy intervalares.

Os conjuntos *fuzzy* intervalares foram introduzidos por Zadeh (ZADEH, 1978), nos anos 70, e nos permite lidar tanto com a imprecisão (classes com limites ocultos) como com a incerteza (falta de informação). Os conjuntos *fuzzy* intervalares são um caso particular de n-tipos de conjuntos *fuzzy*, cujas propriedades estruturais estão relacionadas a Matemática Intervalar. Então, os graus de pertinência de intervalo podem ser usados para representar ou a incerteza numérica relacionada a um grau de associação ou uma série de valores relacionados com diferentes opiniões de especialistas sobre o grau de pertinência a ser adotado. Nesta dissertação vamos utilizar as definições sobre conjuntos *fuzzy* intervalar descrita por SANZ et al (2010).

O conjunto fuzzy intervalar será definido a partir de: um grau de pertinência, que será um sub-intervalo do intervalo  $[0, 1]$ ; e de uma entrada, que é um elemento de um conjunto cujos elementos são intervalos de um universo de intervalos. Esta definição estende toda a teoria de Conjuntos Fuzzy para uma teoria de Conjuntos Fuzzy Intervalar suficientemente capaz de expressar qualquer conjunto Fuzzy (seja este intervalar ou não) em que os valores inteiros dos conjuntos fuzzy não intervalares são representados por intervalos degenerados. Tal generalização pode ser ainda estendida para qualquer entidade fuzzy, pois todas elas são definidas a partir de uma entrada e um grau de pertinência.

Dessa forma, um subconjunto fuzzy intervalar  $A$  de um conjunto  $\mathbb{IR}$  é definido como

o conjunto de pares ordenados representados na equação (27).

**Definição 20** (SANZ et al., 2013) - Um Conjunto Fuzzy Intervalar (CFI)  $A$  sobre o universo  $U \neq \emptyset$  é dado por:

$$A = \{(u, A(u)) | u \in U\} \quad (27)$$

onde

$$A(u) = [\underline{A}(u), \overline{A}(u)] \in L([0, 1]) \quad (28)$$

seja

$$L([0, 1]) = \{X = [\underline{x}, \overline{x}] | (\underline{x}, \overline{x}) \in [0, 1]^2 \text{ e } \underline{x} \leq \overline{x}\} \quad (29)$$

Observe que, dado um CFI  $A$ , a associação de cada elemento  $u_i \in U$  é representada por um intervalo  $A(u_i) = [\underline{A}(u_i), \overline{A}(u_i)]$  com tamanho  $W(A(u_i)) = \overline{A}(u_i) - \underline{A}(u_i)$  conforme figura 7

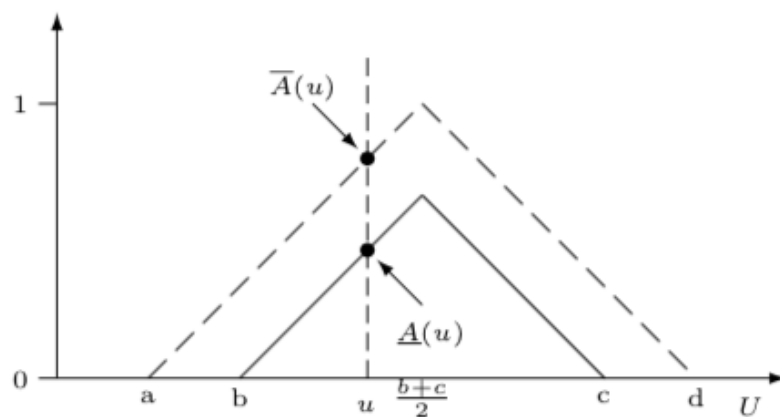


Figura 7: Conjunto Fuzzy Intervalar

Com essa definição é possível perceber que  $A(u) = [\underline{A}(u), \overline{A}(u)]$ , é o grau de pertinência intervalar de  $u \in U$ . A figura 10 mostra que o intervalo  $[\underline{A}_q(u), \overline{A}_q(u)]$  e não um valor de  $[0, 1]$  é atribuído como membro para cada elemento  $u \in U$ .

Um estudo sobre a evolução do CFI e vários resultados em sua representação, tipos e operações conjuntivas podem ser encontrados em (BUSTINCE et al, 2008).

A lógica fuzzy (intervalar) é baseada na teoria dos conjuntos fuzzy intervalares exposta anteriormente. Ela é composta por variáveis e modificadores linguísticos, proposições envolvendo tais variáveis e conectivos. As variáveis linguísticas são variáveis cujos valores são nomes de conjuntos fuzzy intervalares. E os modificadores linguísticos fuzzy intervalares proporcionam uma mudança na função de pertinência de forma que ela relacione o valor semântico do modificador à uma equação matemática, fazendo com que o valor original da função de pertinência seja alterado.

### 2.5.2 Função de Agregação Intervalar

**Definição 21** (SANZ et al., 2013) A função  $T : L([0, 1])^2 \rightarrow L([0, 1])$  é dito ser uma *t-norma intervalar* (*t-norma IV*) se for comutativa, associativa, crescente em ambos os argumentos (em relação à ordem  $\leq_L$  e somente se  $\underline{x} \leq \underline{y}$  e  $\bar{x} \leq \bar{y}$ ), e tem o elemento neutro  $1_L$ . Da mesma forma, uma função  $S : L([0, 1])^2 \rightarrow L([0, 1])$  é dito ser uma *t-conorma intervalar* (*t-conorma IV*) se for comutativa, associativa, crescente com respeito a  $\leq_L$ , e tiver o elemento neutro  $0_L$ .

Uma *T-norma intervalar* é dita *t-representável* se existirem duas *T-normas*  $T_a$  e  $T_b$  em  $[0, 1]$ , sendo  $T_a \leq T_b$  de modo que  $T(A, B) = [T_a(\underline{x}, \underline{y}), T_b(\bar{x}, \bar{y})]$  para todo  $A, B \in L([0, 1])$ .

### 2.5.3 Função de Overlap Intervalar

**Definição 22** (BEDREGAL et al, 2017) Uma função  $IO : L([0, 1])^2 \rightarrow L([0, 1])$  é uma *função overlap intervalar* se satisfizer as condições que seguem:

(IO1)  $IO$  é comutativa;

(IO2)  $IO(A, B) = [0, 0]$  se e somente se  $AB = [0, 0]$ ;

(IO3)  $IO(A, B) = [1, 1]$  se e somente se  $AB = [1, 1]$ ;

(IO4)  $IO$  é monotônica no segundo componente, ou seja  $IO(A, B) \leq_L IO(A, Z)$  quando  $B \leq_L Z$ .

(IO5)  $IO$  é Moore contínuo<sup>2</sup>.

Note que, por (IO1) e (IO4), funções *overlap* intervalares também são monotônicas no primeiro componente. Observe também que os primeiros quatro itens em nossa definição são análogos aos primeiros quatro itens da Definição 12. No último item, no entanto, e para ter uma noção de continuidade, tomamos a continuidade de Moore.

**Observação 1** (BEDREGAL et al, 2017) A partir de agora, sempre consideramos a relação de ordem  $\leq_L$  em  $L([0, 1])$ , exceto nos casos em que outra relação de ordem diferente é especificada explicitamente.

Seja  $\mathfrak{D}$  o conjunto de todas as funções *overlap* intervalares. Nós definimos  $\mathfrak{D}$  na relação binária:

$$IO_1 \leq_{\mathfrak{D}} IO_2 \text{ se e somente se } IO_1(A, B) \leq_L IO_2(A, B), \text{ para todo } A, B \in L([0, 1]) \quad (30)$$

<sup>2</sup> $IO$  é Moore contínuo se é contínua em relação à métrica de Moore adaptada para  $L([0, 1])^2$  da seguinte maneira  $D_M((A_1, A_2), (B_1, B_2)) = \sqrt{\max(|\underline{A}_1 - \underline{B}_1|, |\bar{A}_1 - \bar{B}_1|)^2 + \max(|\underline{A}_2 - \underline{B}_2|, |\bar{A}_2 - \bar{B}_2|)^2}$ .



Claramente,  $\leq_{\mathfrak{D}}$  é uma ordem parcial em  $\mathfrak{D}$ . Além disso, e analogamente ao caso de funções de *overlap* com valor real, temos o seguinte resultado.

**Proposição 3** (BEDREGAL et al, 2017)  $(\mathfrak{D}, \leq_{\mathfrak{D}})$  é um lattice ilimitado.

**Lema 1** (BEDREGAL et al, 2017) Seja  $IO$  uma função *overlap* intervalar e  $A, B \in L([0, 1])$ . Se  $A \neq B$  então  $IO(A, [1, 1]) \neq IO(B, [1, 1])$

**Proposição 4** (BEDREGAL et al, 2017) Seja  $IO$  uma função *overlap* intervalar. Se  $IO$  é associativa então  $IO$  é Moore contínuo e uma  $t$ -norma intervalar positiva<sup>3</sup>

**Teorema 4** (BEDREGAL et al, 2017) Seja  $O_1$  e  $O_2$  uma função *overlap* tal que  $O_1 \leq O_2$ . Então a função  $\widetilde{O_1 O_2} : L([0, 1])^2 \rightarrow L([0, 1])$  definida por

$$\widetilde{O_1 O_2}(A, B) = [O_1(\underline{A}, \underline{B}), O_2(\overline{A}, \overline{B})] \quad (31)$$

é uma função *overlap* intervalar.

Analogamente à noção de  $t$ -representatividade em (CORNELIS et al, 2006), uma função de *overlap* intervalar  $IO$  é dito ser  $o$ -representável se existem funções de *overlap*  $O_1$  e  $O_2$  tal que  $IO = \widetilde{O_1 O_2}$ .  $O_1$  e  $O_2$  são chamadas representativas de  $IO$ .

**Definição 23** (BEDREGAL et al, 2017) Seja  $F : L([0, 1])^n \rightarrow L([0, 1])$  uma função monotônica. As projeções esquerda e direita  $F$  são as funções  $\underline{F}, \overline{F} : [0, 1]^n \rightarrow [0, 1]$  definidas por

$$\begin{aligned} \underline{F}(x_1, \dots, x_n) &= \underline{F}([x_1, x_1], \dots, [x_n, x_n]) \\ \overline{F}(x_1, \dots, x_n) &= \overline{F}([x_1, x_1], \dots, [x_n, x_n]), \end{aligned} \quad (32)$$

respectivamente.

**Proposição 5** (BEDREGAL et al, 2017) Seja  $IO : L([0, 1])^2 \rightarrow L([0, 1])$  uma função *overlap* intervalar. Se  $IO$  é fortemente positivo (FP), isto é, afirma que  $\underline{A} = 0$  ou  $\underline{B} = 0$  enquanto que  $IO(A, B) = [0, z]$ , para algum  $z \in (0, 1]$ , então  $\underline{IO}$  assim como  $\overline{IO}$  são funções *overlap*.

Agora podemos caracterizar  $o$ -representável funções de *overlap* intervalares.

**Teorema 5** (BEDREGAL et al, 2017) Uma função *overlap* intervalar  $IO$  é  $o$ -representável se e somente se  $IO = \widetilde{IO \overline{IO}}$ .

<sup>3</sup>Uma  $t$ -norma intervalar  $T$  é chamada positiva se  $T(A, B) = [0, 0]$  se e somente se  $A = [0, 0]$  ou  $B = [0, 0]$ .

**Lema 2** (BEDREGAL et al, 2017) *Seja  $F : L([0, 1])^n \rightarrow L([0, 1])$  uma função monotônica. Então, para qualquer  $A_i \in L([0, 1])$ , com  $i = 1, \dots, n$ , segue que*

$$F(A_1, \dots, A_n) = [F(\underline{A}_1, \dots, \underline{A}_n), \overline{F}(\overline{A}_1, \dots, \overline{A}_n)]$$

*se e somente se  $F$  for inclusão monotônica<sup>4</sup>.*

**Teorema 6** (BEDREGAL et al, 2017) *Uma função overlap intervalar  $IO$  é o-representável se e somente se for inclusão monotônica.*

**Definição 24** (BEDREGAL et al, 2017) *Uma função overlap intervalar  $IO$  é semi o-representável se existe funções overlap  $O_i : [0, 1]^2 \rightarrow [0, 1]$ , com  $i = 1, \dots, 8$ , e funções de agregação  $M_1, M_2 : [0, 1]^4 \rightarrow [0, 1]$ , tal que para cada  $A, B \in L([0, 1])$ , segue que:*

$$IO(A, B) = [M_1(O_1(\underline{A}, \overline{B}), O_2(\overline{A}, \underline{B}), O_3(\underline{A}, \underline{B}), O_4(\overline{A}, \overline{B})), M_2(O_5(\underline{A}, \overline{B}), O_6(\overline{A}, \underline{B}), O_7(\underline{A}, \underline{B}), O_8(\overline{A}, \overline{B}))] \quad (33)$$

**Teorema 7** (BEDREGAL et al, 2017) *Seja  $O_i$ , com  $i = 1, \dots, 8$ , uma função overlap e  $M_1, M_2 : [0, 1]^4 \rightarrow [0, 1]$  uma função de agregação tal que  $O_i \leq_{\Delta} O_{i+4}$ , for  $i = 1, \dots, 4$ , segue que  $M_1 \leq M_2$ . Então, para a função overlap intervalar  $IO : L([0, 1])^2 \rightarrow L([0, 1])$ , definido como na Equação (33), segue que:*

1. *Se  $O_1 = O_2$ ,  $O_5 = O_6$  e  $M_1$  e  $M_2$  são comutativas nos dois primeiros componentes então  $IO$  satisfaz (IO1);*
2. *Se alguma das  $M_1$  ou  $M_2$  satisfaz a propriedade (M3) com respeito ao quarto item, ou seja,  $M_1(x_1, x_2, x_3, x_4) = 0$  então  $x_4 = 0$ , e o mesmo para  $M_2$ . Então  $IO$  satisfaz (IO2);*
3. *Se  $M_1$  ou  $M_2$  satisfaz a propriedade (M4) referente ao terceiro item, então  $IO$  satisfaz (IO3);*
4. *Se  $O_i \leq_{\Delta} O_{i+4}$ , para  $i = 1, \dots, 4$ , então  $IO$  satisfaz (IO4);*
5.  *$IO$  satisfaz (IO5) se e somente se  $M_1$  e  $M_2$  forem contínuos.*

**Corolário 2** (BEDREGAL et al, 2017) *Se existem funções  $M_1, M_2$  e  $O_i$ , com  $i = 1, \dots, 8$ , satisfazendo a condição do Teorema 7, então a função overlap intervalar  $IO$  definida na Equação (33) é semi o-representável.*

Semi o-representabilidade e representabilidade estão relacionados, como discutiremos a seguir.

---

<sup>4</sup> $A \subseteq B \rightarrow F(A) \subseteq F(B)$

Tabela 4: Exemplos de funções *overlap* intervalares

Funções <i>Overlap</i> Intervalares
$T_P(A, B) = AB$
$IO_p(A, B) = A^p B^p, p > 0$ (em particular, $IO_{\sqrt{}}(A, B) = \sqrt{AB}$ )
$O_{rat}(A, B) = \frac{\sqrt{AB}}{\sqrt{AB+1}-AB}$

**Proposição 6** (BEDREGAL et al, 2017) *Seja IO um overlap intervalar. Se IO é o-representável ou existe uma t-norma T de tal forma que  $IO = \mathcal{T}_T$  ou  $IO = \mathcal{T}_{T,b}$ , para algum  $t \in [0, 1]$ , ou  $IO = \mathcal{T}'_T$ , então IO é semi o-representável<sup>5</sup>.*

Na Tabela 4 estão apresentados alguns exemplos de funções *overlap* intervalares.

## 2.6 Sistemas de Classificação

Considere o problema padrão de aprendizado supervisionado, determinar a classe de um novo exemplo em uma das classes existentes e conhecidas,  $C_j \in C = C_1, \dots, C_M$ , onde  $M$  é o número de classes do problema. Cada um dos exemplos  $e \in E$  é descrito por um conjunto de  $N$  observações  $X(E) = (e_1, \dots, e_N)$  e cada uma dessas observações é chamada de variável, atributo ou característica.

Portanto, o projeto de um Classificador ou Sistema de Classificação é para encontrar uma função de decisão

$$D : X(E) \rightarrow C$$

de forma haver um critério que avalia a qualidade do Sistema de Classificação. Normalmente, o objetivo é obter um classificador capaz de determinar a classe dos exemplos em qualquer ponto do espaço de atributos com o mínimo erro possível.

Para realizar o processo de aprendizado supervisionado, figura 8, o sistema parte de um conjunto de exemplos classificados corretamente, chamado de *conjunto de treinamento*. O processo de aprendizado indutivo extrai as informações necessárias que permitem classificar novos exemplos posteriormente. A informação extraída é descrita pelo modelo de representação do conhecimento escolhido.

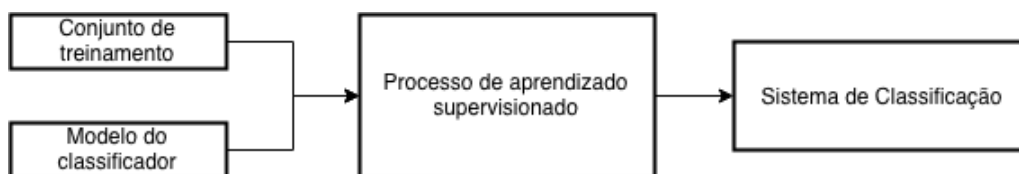


Figura 8: Processo de Aprendizado Supervisionado

<sup>5</sup>A noção de t-representabilidade foi introduzida por DESCHRIJVER (2008)

O processo de classificação em si, figura 9, é realizado quando o Sistema de Classificação recebe um padrão de dados válido do qual sua classe não é conhecida e tem que tomar a decisão sobre a classe à qual pertence.

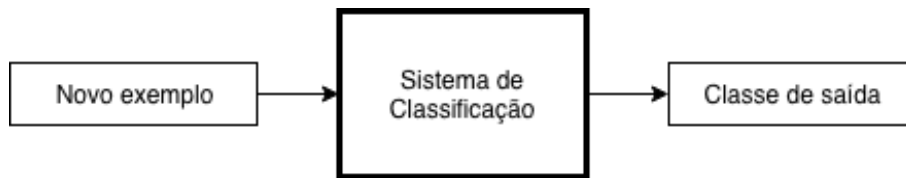


Figura 9: Processo de Classificação

Para avaliar a qualidade do Sistema de Classificação, são utilizadas técnicas de estimativa de erros. Essas técnicas são baseadas na porcentagem de sucesso do classificador em um conjunto de testes, que é composto por um conjunto de exemplos que não foram usados no processo de aprendizado.

Na literatura especializada, inúmeras estratégias têm sido propostas para abordar o problema da classificação; desde estratégias puramente estatísticas, discriminantes, até estratégias incluídas na Computação Flexível (BUSTINCE, 2010), como redes neurais, lógica *fuzzy*, lógica *fuzzy* intervalar ou técnicas bayesianas.

Existem muitos problemas de classificação envolvendo várias classes. A classificação multi-categoria em Aprendizado de Máquina tem sido amplamente estudada, alguns algoritmos de aprendizado são projetados para lidar com problemas binários e de múltiplas categorias, mas existem outras técnicas de aprendizado, cuja extensão para problemas de multi-classificação não é fácil.

De um modo ou de outro, o caso binário onde apenas duas classes são consideradas é o problema de classificação mais simples (do ponto de vista do número de classes), assim como mais classes são consideradas, a dificuldade do problema é aumentada, é por isso estratégias de decomposição surgiram.

Uma maneira fácil de realizar um problema de várias classes é usar técnicas de binarização, em que o problema original é decomposto em vários problemas binários mais fáceis. Técnicas de decomposição binária (*ensembles*) consistem em duas etapas diferentes.

A primeira é a estratégia de decomposição, as estratégias mais comuns são as decomposições One-vs-One (OVO) e One-vs-All (OVA). O segundo consiste em fazer a predição final da classe a partir das saídas dos classificadores binários, uma combinação correta de saídas dos classificadores é crucial para fazer a predição correta (GALAR et al, 2011). Desta forma, um *ensemble* é um conjunto de classificadores cujas decisões individuais são combinadas de alguma forma.

De acordo com as características acima, nesta dissertação nos concentramos em *ensembles* de sistemas de classificação resultantes da combinação da lógica *fuzzy* com técnicas baseadas em regras, ou seja, Sistemas de Classificação Baseados em Regras

Fuzzy (SCBRF) (ISHIBUCHI et al, 2005), e sua extensão para o uso de lógica *fuzzy* intervalar (SCBRFI) (SANZ et al, 2010).

A principal razão para a escolha destes classificadores é obter modelos facilmente interpretáveis. Esses modelos usam variáveis linguísticas, que são simples de entender pelo usuário final ou pelo especialista. Além disso, como mencionado na motivação, usamos os conceitos dos Conjuntos Fuzzy Intervalar para lidar com o problema da escolha das funções de pertinências, com o objetivo de minimizar a incerteza na coleta ou por necessitarmos de uma solução relativamente precisa e muito condicionada pela função de associação que usamos (BUSTINCE, 2010).

### 2.6.1 Sistemas de Classificação Baseado em Regras *Fuzzy*

As técnicas de classificação decidem quanto um registro é ou não parte de alguma classe, tipo ou categoria. Classificação é constantemente utilizado para agrupamento, criando conjuntos baseados em regras. Muitas vezes, esses sistemas aprendem revisando muitas instâncias de itens nas categorias (classes ou atributos) para deduzir regras de classificação. A classificação ajuda a decidir se uma nova entrada corresponde a um padrão previamente observado ou não, e é frequentemente usada para classificar o comportamento ou os padrões como incomuns. A quantidade de aplicações para reconhecimento de padrões são ilimitadas, ele pode ser usado para detectar atividade de rede suspeita ou fraude, caracterizar interesses de usuários semelhantes, descobrir quando a mensagem de um usuário indica frustração ou satisfação. Cada uma destas técnicas funciona melhor quando utilizado com uma grande quantidade de dados de entrada. Em alguns casos, essas técnicas devem não só trabalhar em grandes quantidades de entrada, mas devem produzir resultados rapidamente, e esses fatores tornam a escalabilidade uma questão importante.

Geralmente, os sistemas de classificação baseados em regras *fuzzy* são divididos em dois tipos (SANZ et al, 2010) sistemas interpolativos *fuzzy* e classificadores baseados em regras *Fuzzy*. Os sistemas interpolativos *fuzzy* (GARCIA-JIMENEZ et al, 2015) são comumente usados quando são necessárias saídas contínuas. Nesses sistemas, o algoritmo de inferência é baseado na regra de interpolação, que pode ser descrita com duas etapas. No primeiro, o índice de consistência entre os antecedentes das regras e o fato é calculado; e, no segundo, a conclusão é obtida relacionando o valor resultante e o conseqüente das regras, por meio do índice de consistência/confiança.

Os dois principais componentes dos SCBRF são os seguintes.

1. Base de conhecimento: É composta tanto pela Base de Regras (BR) quanto pelo banco de dados, onde as regras e as funções de associação são armazenadas, respectivamente.
2. Mecanismo de raciocínio *fuzzy*: Este é o mecanismo usado para classificar objetos

usando as informações armazenadas na base de conhecimento.

Para gerar a base de conhecimento, é aplicado um algoritmo de aprendizado de regra *fuzzy* que usa um conjunto  $P$  de padrões rotulados  $x_p = (x_{p1} \dots, x_{pn})$ ,  $p = \{1, 2, \dots, P\}$ , onde  $x_{pi}$  é o  $i$ -ésimo valor do atributo ( $i = \{1, 2, \dots, n\}$ ). Cada um dos  $n$  atributos é descrito por um conjunto de termos linguísticos junto com suas funções de associação correspondentes. Em ELKANO et al (2017) é definido o uso de regras *fuzzy* da seguinte forma:

Regra  $R_q$ : Se  $\chi_1$  é  $A_{q1}$  e  $\dots$  e  $\chi_n$  é  $A_{qn}$  então Classe  $C'_q \in \{C_1, \dots, C_M\}$  com  $Cnf_q$

onde  $R_q$  é o rótulo da regra  $q$ ,  $\chi = (x_1, \dots, x_n)$  é um vetor de padrão  $n$ -dimensional,  $A_{qi}$  é um conjunto *fuzzy* antecedente representando um termo linguístico,  $C_q$  é o rótulo de classe e  $Cnf_q \in [0, 1]$  é a confiança (peso) da regra.

### 2.6.1.1 Mecanismo de Raciocínio Fuzzy

Seja  $x_p = (x_{p1}, \dots, x_{pn})$  um novo padrão a ser classificado,  $L$  denota o número de regras na BR e  $M$  é o número de classes do problema; então, os passos do mecanismo de raciocínio *fuzzy* (SANZ et al, 2010) são os seguintes.

1. *Matching degree*, ou seja, a força de ativação de-para todas as regras na BR com o padrão  $x_p$ . Para calcular usa-se uma função de agregação, em geral, a T-norma do produto ou do mínimo, aqui denotada por  $T$ :

$$\mu_{A_q}(x_p) = T(\mu_{A_{q1}}(x_{p1}), \dots, \mu_{A_{qn}}(x_{pn})) \quad q = 1, \dots, L$$

2. *Association degree*, para calcular o grau de associação do padrão  $x_p$  com as classes  $M$  de acordo com cada regra no BR. Para este objetivo, um operador de combinação  $h$  é aplicado para combinar o grau de correspondência com o peso da regra. Ao usar regras na forma apresentado em ELKANO et al (2017), esse grau de associação refere-se apenas à classe resultante da regra (ou seja,  $k = Classe(R_q)$ ):

$$b_q^k = \mu_{A_q}(x_p) \times Cnf_q^k \quad k = Classe(R_q), \quad q = 1, \dots, L$$

3. *Pattern Classification*, grau de consistência de classificação dos padrões para todas as classes. Usa-se uma função de agregação  $\mathbb{M}$ , que combina os graus positivos de associação calculados na etapa anterior, em geral a função  $\max$  (máximo), mas em trabalhos recentes a integral de choquet e suas generalizações (LUCCA) :

$$Y_k = \mathbb{M}(b_q^k, q = 1, \dots, L \text{ e } b_q^k > 0), \quad k = 1, \dots, M.$$

4. Classificação. Aplica-se uma função de decisão  $F$  sobre o grau de consistência do sistema para a classificação de padrões para todas as classes. Esta função determinará a classe de rótulo  $l$  correspondente ao valor máximo:

$$F(Y_1, \dots, Y_M) = \arg \max(Y_k), \quad k = 1, \dots, M$$

### 3 ÍNDICE DE *OVERLAP* INTERVALAR E SISTEMAS DE CLASSIFICAÇÃO INTERVALARES

Neste capítulo será apresentado um exemplo de sistema de classificação baseado em regras *fuzzy* intervalares usando vários índices de *overlap* intervalares e funções de agregação. Assim, a Seção 3.2 define a construção de graus de confiança de índices de *overlap* intervalares. Na Seção 3.3 é descrito o mecanismo de raciocínio de um sistema de classificação baseado em regras *fuzzy* intervalares e na seção 3.4 consta a escolha da melhor classe em conjuntos de regras *fuzzy* usando funções *penalty*.

#### 3.1 Sistemas de Classificação Baseados em Regras *Fuzzy* Intervalares Usando Várias Funções *Overlap* Intervalares e Funções de Agregação

São muitos os problemas de classificação envolvendo várias classes, o projeto de um sistema de classificação consiste em encontrar uma regra de decisão que permita determinar a classe de um novo objeto (também chamado de exemplo  $\chi \in E$ ) entre as classes já existentes e conhecidas,  $C \in \{C_1, \dots, C_M\}$ . Um exemplo é descrito através de um conjunto de observações  $\chi = (\chi_1, \dots, \chi_n)$ . Cada uma dessas observações é chamada de variável, atributo ou característica. Como já apresentado anteriormente, o projeto de um classificador ou um sistema de classificação pode ser visto como a busca de um mapeamento:

$$D : E \rightarrow \{C_1, \dots, C_M\}$$

sendo otimizado por algum critério que determine a qualidade do classificador.

Um classificador *fuzzy* intervalar é um SCBRF que utiliza a incerteza no mecanismo de raciocínio (RIID; RUSTERN, 2014) e que consiste em regras no seguinte formato:

$$\text{Regra } R_q: \text{ Se } \chi_1 \text{ é } A_{q1} \text{ e } \dots \text{ e } \chi_n \text{ é } A_{qn} \text{ então Classe } C'_q \text{ com } C_n f_q$$

onde  $R_q$  é o rótulo da regra,  $\chi = (\chi_1, \dots, \chi_n)$  é um vetor n-dimensional de exemplo,  $A_{qi}$  denota o rótulo linguístico intervalar do i-ésimo recurso associado à regra  $R_q$ ,  $C_q$  é um



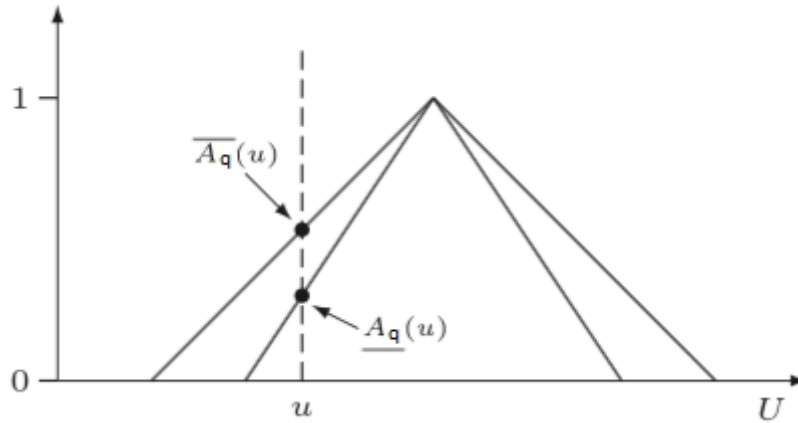


Figura 10: Conjunto *Fuzzy* Intervalar Usado Nesta Dissertação

consequente da classe e  $Cn.f_q \in L([0, 1])$  é o grau de confiança intervalar da regra  $R_q$  (ou seja, o peso da regra), onde cada um desses conceitos serão definidos a seguir.

Retomamos o conceito apresentado na equação (28) onde  $A_{qi}$ , é o grau de pertinência intervalar de  $u \in U$ . Por exemplo, a figura 10 mostra que o intervalo  $A_q = [A_q(u), \overline{A_q}(u)]$  e não um valor de  $[0, 1]$  é atribuído como membro para cada elemento  $u \in U$ .

Esta seção tem como objetivo definir o novo Mecanismo de Raciocínio *Fuzzy* Intervalar (MRF-I). Para tanto, modificamos todas as etapas do método de raciocínio *fuzzy* abordado em nosso trabalho anterior (ELKANO et al, 2017). Quando precisarmos usar uma relação de ordem total para intervalos, ou seja, quando a maior associação de intervalo precisar ser determinada na última etapa da MRF-I, usaremos a definida por Xu e Yager em (XU; YAGER, 2006). Ressalta-se que, embora a definição do MRF seja no contexto fuzzy intervalar, na prática pode-se calcular os resultados intervalares obtidos por agregações intervalares decompondo o conjunto de dados intervalares em dois subconjuntos de dados, um consistindo dos valores dos extremos inferiores e o outro dos valores dos extremos superiores dos graus de pertinência intervalares. Assim o MRF-I se decompõe em dois mecanismos, o MRF-II (inferior) e o MRF-IS (superior).

Seja  $L$  o número de regras na BR e  $M$  o número de classes do problema. Se  $x_p = (x_{p1}, \dots, x_{pn})$  for um novo padrão a ser classificado, as etapas do MRF-I serão as seguintes.

1. *Interval matching degree*: Vamos considerar o *matching degree* calculado pelo *overlap* intervalar  $IO$ , porque estamos tratando do intervalo:

$$[A_q(x_p), \overline{A_q}(x_p)] = IO([A_{q1}(x_{p1}), \overline{A_{q1}}(x_{p1})], \dots, [A_{qn}(x_{pn}), \overline{A_{qn}}(x_{pn})]) \quad q = 1, \dots, L; \quad (34)$$

2. *Interval association degree*: Aplica-se um operador de agregação intervalar  $h$  ao

grau de correspondência dos intervalos calculados anteriormente e o peso da regra:

$$\left[ \underline{b}_q^k, \overline{b}_q^k \right] = h(\left[ \underline{A}_q(x_p), \overline{A}_q(x_p) \right], \left[ \underline{Cn.f}_q^k, \overline{Cn.f}_q^k \right]) \quad k = 1, \dots, M, q = 1, \dots, L.; \quad (35)$$

Devemos salientar que o peso da regra é um elemento de  $L([0, 1])$ . Para calculá-lo, utiliza-se uma das funções de agregação conforme Subseção 2.5.3 fazendo uso da aritmética intervalar introduzida na Seção 2.4. A equação resultante é mostrada a seguir:

$$\left[ \underline{Cn.f}_q, \overline{Cn.f}_q \right] = \frac{\sum_{x_p \in \text{Classe } C_q} \left[ \underline{A}_q(x_p), \overline{A}_q(x_p) \right]}{\sum_{p=1}^P \left[ \underline{A}_q(x_p), \overline{A}_q(x_p) \right]} \quad (36)$$

onde para toda a regra  $q$  existe  $p$  tal que  $\underline{A}_q(x_p) > 0$

3. *Interval pattern classification soundness degree for all classes*: É o grau de consistência de todas as classes, agrega-se os graus de associação de cada classe aplicando uma função de agregação intervalar:

$$\left[ \underline{Y}_k, \overline{Y}_k \right] = f\left(\left[ \underline{b}_q^k, \overline{b}_q^k \right]\right), q = 1, \dots, L \text{ e } \underline{b}_q^k > 0_L \quad k = 1, \dots, M.; \quad (37)$$

4. *Classificação*. Aplica-se uma função de decisão  $F$  sobre a saída do grau de consistência do sistema para a classificação dos padrões para todas as classes:

$$F\left(\left[ \underline{Y}_1, \overline{Y}_1 \right], \dots, \left[ \underline{Y}_M, \overline{Y}_M \right]\right) = \arg \max\left(\left[ \underline{Y}_k, \overline{Y}_k \right]\right), \quad k = 1, \dots, M. \quad (38)$$

onde adota-se  $\max\left(\left[ \underline{Y}_k, \overline{Y}_k \right]\right) = \left[ \max_{k=1, \dots, M} \underline{Y}_k, \max_{k=1, \dots, M} \overline{Y}_k \right]$ .

Neste último passo é selecionado o maior valor da saída da equação (37) (grau de consistência).

### 3.2 Construção de Grau de Confiança Intervalar Sobre Índices de *Overlap* Intervalares

A seguir introduz-se a definição de índice de *overlap* intervalar para que se possa contruir graus de confiança intervalares.

**Definição 25** *Seja  $\leq$  uma ordem admissível sobre  $L([0, 1])$ , um índice de overlap intervalar é uma função  $\mathcal{IO} : CFI(U) \times CFI(U) \rightarrow L([0, 1])$  de modo que, para todo  $A, B, C \in CFI(U)$ , as seguintes condições:*

Tabela 5: Exemplos de índices de *overlap* intervalar

Índices de <i>Overlap</i> Intervalar	
$\mathcal{IO}_Z(A, B)$	$= \max_{u \in U} \min\{A(u), B(u)\}$
$\mathcal{IO}_X(A, B)$	$= \begin{cases} 0_L & \text{se } \forall u \in U : A(u)B(u) = 0 \\ X & \text{caso contrário,} \end{cases}$ para todo $X \in L([0, 1])$
$\mathcal{IO}_\pi(A, B)$	$= \frac{1}{n} \sum_{u \in U} A(u) \cdot B(u)$ , onde $n = \text{card}(U)$

Tabela 6: Exemplos de índices de *overlap* intervalar construídos pela Definição 25

Função de <i>Overlap</i> Intervalar $IO$	Função de Agregação $M$	Índice de <i>Overlap</i> Intervalar $\mathcal{O}$
$IO_\surd$	média aritmética	$\mathcal{IO}_\surd(A, B) = \frac{1}{n} \sum_{u \in U} \sqrt{A(u) \cdot B(u)}$
$IO_M$	máximo	$\mathcal{IO}_Z(A, B) = \max_{u \in U} \min\{A(u), B(u)\}$
$IO_{rat}$	média aritmética	$\mathcal{IO}_{rat}(A, B) = \frac{1}{n} \sum_{u \in U} \frac{\sqrt{A(u) \cdot B(u)}}{\sqrt{A(u) \cdot B(u) + 1 - A(u)}}$

( $\mathcal{IO}1$ )  $\mathcal{IO}(A, B) = [0, 0]$  se e somente se  $A$  e  $B$  tem suporte disjuntivo, isto é, para todo  $u \in U$ , tem-se que  $A(u)B(u) = [0, 0]$ ;

( $\mathcal{IO}2$ )  $\mathcal{IO}(A, B) = \mathcal{IO}(B, A)$ ;

( $\mathcal{IO}3$ ) Se  $B \leq C$ , então  $\mathcal{IO}(A, B) \leq \mathcal{IO}(A, C)$ .

Um índice de *overlap* intervalar  $\mathcal{IO}$  é dito normal sempre que a seguinte condição contenha:

( $\mathcal{IO}4$ ) Se existe  $u \in U$  tal que  $A(u) = B(u) = [1, 1]$ , então  $\mathcal{IO}(A, B) = [1, 1]$ .

Alguns exemplos de índices de *overlap* estão presentes na Tabela 5. Observe que  $\mathcal{IO}_Z$  é a versão intervalar do índice de consistência de ZADEH (1978).  $\mathcal{IO}_Z$  e  $\mathcal{IO}_\pi$  são normais e  $\mathcal{IO}_X$  não é normal para  $x \neq 1$ .

A Tabela 6 mostra alguns exemplos de índices de *overlap* intervalares construídos como extensões dos índices de *overlap* introduzidos em ELKANO et al (2017), usando algumas funções de *overlap* mostradas na Tabela 4.

Para calcular o grau de confiança da regra  $R_q$ , ou seja, para calcular  $Cnf_q$ , pode-se usar o grau de confiança e o suporte como apresentado na Equação (36). A confiança de uma associação é classicamente medida pela co-ocorrência de atributos em tuplas no banco de dados.

Considere um conjunto de  $p$  regras  $\{R_1, \dots, R_p\}$  conforme segue:

Regra  $R_1$ : Se  $\chi_1$  é  $[A_{11}, \overline{A_{11}}]$  e  $\dots$  e  $\chi_n$  é  $[A_{1n}, \overline{A_{1n}}]$  então Classe  $C'_1$  com  $[Cnf_1, \overline{Cnf_1}] = ?$

...

$R_p$ : Se  $\chi_1$  é  $[A_{p1}, \overline{A_{p1}}]$  e  $\dots$  e  $\chi_n$  é  $[A_{pn}, \overline{A_{pn}}]$  então Classe  $C'_p$  com  $[Cnf_p, \overline{Cnf_p}] = ?$

e pegue um conjunto de  $m$  exemplos  $\chi_l$  com  $l = 1, \dots, m$ .

A seguir, apresentamos o Algoritmo (1), que foi desenvolvido para criar medidas de confiança e suporte intervalares, usando índices de *overlap* intervalar. Este algoritmo é uma generalização do proposto em ELKANO et al (2017).

**Algoritmo 1**

**Entrada:** Um conjunto de regras  $R_j$ , com  $j \in \{1, \dots, p\}$ , e um conjunto de exemplos  $\chi_l$ , com  $l \in \{1, \dots, m\}$ .

**Saída:** As medidas de Confiança  $([Cnf(R_j), \overline{Cnf}(R_j)])$  e Suporte  $([Supp(R_j), \overline{Supp}(R_j)])$ , para cada regra  $R_j$ , com  $j \in \{1, \dots, p\}$ .

- 1: Selecionar uma função *overlap* intervalar  $IO$  e um índice de *overlap* intervalar  $\mathcal{IO}$ ;
- 2: Buscar os exemplos  $\chi_l = (\chi_{l1}, \dots, \chi_{ln})$ , e as respectivas  $C_l$  classes que pertencem, com  $l = \{1, \dots, m\}$ ;
- 3: **for**  $q = 1$  to  $p$  **do**
- 4: Selecionar o  $s \leq m$  exemplos que nos dizem que o objeto considerado pertence à classe  $C_q$  associada à regra  $R_q$ ;
- 5: **for**  $j = 1$  to  $s$  **do**
- 6: Calcular (Grau de pertinência intervalar)

$$c_j([\underline{\chi}_j, \overline{\chi}_j]) = IO([\underline{A}_{q1}(\chi_{j1}), \overline{A}_{q1}(\chi_{j1})], \dots, [\underline{A}_{qn}(\chi_{jn}), \overline{A}_{qn}(\chi_{jn})]); \quad (39)$$

- 7: **end for**
- 8: Construir o conjunto *fuzzy* intervalar em  $U$

$$\mathbb{C}_{qs} = \{(u_1, [\underline{c}_1(\chi_1), \overline{c}_1(\chi_1)]), \dots, (u_s, [\underline{c}_s(\chi_s), \overline{c}_s(\chi_s)]), (u_{s+1}, 0), \dots, (u_m, 0)\}; \quad (40)$$

- 9: **for**  $l = 1$  to  $m$  **do**
- 10: Calcular

$$c_l([\underline{\chi}_l, \overline{\chi}_l]) = IO([\underline{A}_{q1}(\chi_{l1}), \overline{A}_{q1}(\chi_{l1})], \dots, [\underline{A}_{qn}(\chi_{ln}), \overline{A}_{qn}(\chi_{ln})]); \quad (41)$$

- 11: **end for**
- 12: Construir o conjunto *fuzzy* em  $U$

$$\mathbb{C}_{qm} = \{(u_1, [\underline{c}_1(\chi_1), \overline{c}_1(\chi_1)]), \dots, (u_m, [\underline{c}_m(\chi_m), \overline{c}_m(\chi_m)])\};$$

- 13: Calcular

$$[Cnf(R_q), \overline{Cnf}(R_q)] = \frac{\mathcal{IO}(\mathbb{C}_{qs}, U)}{\mathcal{IO}(\mathbb{C}_{qm}, U)}; \quad (42)$$

- 14: Calcular

$$[Supp(R_q), \overline{Supp}(R_q)] = \mathcal{IO}(\mathbb{C}_{qs}, U); \quad (43)$$

- 15: **end for**

**Observação 2** Observe que:

1. Se, na Equação (42) e (43), tomamos a expressão para o índice de overlap intervalar  $\mathcal{IO}_\pi$ , definido na Tabela 5, alterarmos os conjuntos de grau de confiança e o suporte apresentado por Ishibuchi et al. (ISHIBUCHI et al, 2005), pela equação (36) que são dadas, respectivamente, por:

$$\begin{aligned} [\underline{Cnf}(R_q), \overline{Cnf}(R_q)] &= \frac{\mathcal{IO}_\pi(\mathbb{C}_{qs}, U)}{\mathcal{IO}_\pi(\mathbb{C}_{qm}, U)} \\ &= \frac{\frac{1}{m} \sum_{i=1}^s c_i([\underline{\chi}_i, \bar{\chi}_i])}{\frac{1}{m} \sum_{i=1}^m c_i([\underline{\chi}_i, \bar{\chi}_i])} \\ &= \frac{\sum_{i=1}^s c_i([\underline{\chi}_i, \bar{\chi}_i])}{\sum_{i=1}^m c_i([\underline{\chi}_i, \bar{\chi}_i])} \end{aligned} \quad (44)$$

e

$$[\underline{Supp}(R_q), \overline{Supp}(R_q)] = \mathcal{IO}_\pi(\mathbb{C}_{qs}, U) = \frac{1}{m} \sum_{i=1}^s c_i([\underline{\chi}_i, \bar{\chi}_i]). \quad (45)$$

onde sabe-se que  $0_L \notin \sum_{i=1}^m c_i([\underline{\chi}_i, \bar{\chi}_i])$

2. Na construção do grau de confiança intervalar  $Cnf$  da regra, usando a Equação (42), só levamos em consideração os conjuntos  $\mathbb{C}_{qs}$  e  $\mathbb{C}_{qm}$ . Existem situações em que é necessário considerar o índice de overlap intervalar do conjunto composto pelas regras que não classificam classe considerada nesse momento (SANZ et al, 2010). No Algoritmo (1), isso significa que devemos construir o conjunto

$$\mathbb{C}_{q(m-s)} = \{(u_1, 0), \dots, (u_s, 0), (u_{s+1}, 0), (u_{s+1}, c_{s+1}([\underline{\chi}_{s+1}, \bar{\chi}_{s+1}]), \dots, (u_m, c_m([\underline{\chi}_m, \bar{\chi}_m]))\}$$

e construir o grau de confiança da seguinte forma:

$$[\underline{Cnf}(R_q), \overline{Cnf}(R_q)] = \frac{\max\{0, \mathcal{IO}(\mathbb{C}_{qs}, U) - \mathcal{IO}(\mathbb{C}_{q(m-s)}, U)\}}{\mathcal{IO}(\mathbb{C}_{qm}, U)}. \quad (46)$$

Usando a notação do Passo 6 do Algoritmo (1), criamos os seguintes conjuntos sobre o referencial  $U$ :

$$\binom{s}{A_{ql} \gamma} = \{(u_l, A_{ql}(\chi_{l\gamma})) | u_l \in U\}, \quad (47)$$

onde  $\gamma = 1, \dots, n$ ,  $q \in \{1, \dots, p\}$  é o rótulo da regra  $R_q$ ,  $s \leq m$  é o número de exemplos que informa que o objeto considerado pertence à classe  $C_q$  associada à regra  $R_q$ ,  $m$  é o número total de exemplos e  $\chi_l = (\chi_{l1}, \dots, \chi_{ln})$ , com  $l = \{1, \dots, m\}$ , é o conjunto ordenado de exemplos, levando em consideração a classe que classificam, utilizando o grau de pertinência do conjunto fuzzy intervalar  $A_{qi}$  segundo a Definição 20.

### 3.3 Mecanismo de Raciocínio de um Sistema de Classificação Baseado em Regras *Fuzzy* Intervalar

O MRF-I de um SCBRFI geralmente é implementado pela única abordagem vencedora que seleciona o rótulo da classe associado à regra que fornece o maior grau de ativação da regra.

Considere um conjunto de  $M$  classes  $\{C_1, \dots, C_M\}$  e um conjunto de  $p$  regras  $\{R_1, \dots, R_p\}$ , com  $M \leq p$ , dado como:

Regra  $R_q$ : Se  $\chi_1$  é  $A_{q1}$  e  $\dots$  e  $\chi_n$  é  $A_{qn}$  então Classe  $C_q$  com  $Cn.f_q$ ,

onde  $q \in \{1, \dots, p\}$ ,  $C_q \in \{C_1, \dots, C_M\}$  representa a classe da regra  $R_q$ , e  $Cn.f_q$  é o grau de confiança intervalar da regra  $R_q$ , que é calculado pela aplicação do Algoritmo 1.

Então, dado o esquema:

Regra  $R_1$ : Se  $\chi_1$  é  $A_{11}$  e  $\dots$  e  $\chi_n$  é  $A_{1n}$  então Classe  $C_1$  com  $Cn.f_1$

⋮

Regra  $R_p$ : Se  $\chi_1$  é  $A_{p1}$  e  $\dots$  e  $\chi_n$  é  $A_{pn}$  então Classe  $C_p$  com  $Cn.f_p$

Fato:  $\chi_1$  é  $A'_1$  e  $\dots$  e  $\chi_n$  é  $A'_n$

no Algoritmo 2 calculamos a classe  $C \in \{C_1, \dots, C_M\}$  para o qual o fato (consequente) pertence, levando em consideração diferentes valores de *overlaps* intervalares entre os fatos e os antecedentes das regras. Representamos esses *overlaps* intervalares por meio de diferentes índices de *overlap* intervalares. Esse mecanismo de raciocínio (usando diferentes índices de *overlap* intervalares) é o que distingue o Algoritmo 2 dos clássicos (veja ISHIBUCHI et al (2005)) e do proposto em ELKANO et al (2017).

---

**Algoritmo 2**


---

**Entrada:** Um conjunto de regras  $R_j$ , com  $j \in \{1, \dots, p\}$ , um conjunto de classes  $\{C_1, \dots, C_M\}$ , com  $M \leq p$  e um Fato.

**Saída:** Uma classe  $C \in \{C_1, \dots, C_M\}$ .

- 1: Selecionar uma função *overlap* intervalar  $IO$  e uma função de agregação  $M$ ;
  - 2: Selecionar um conjunto de  $S$  índices de *overlap* intervalares  $\{IO_1, \dots, IO_S\}$ ;
  - 3: Executar  $r$  vezes o Algoritmo 1. Cada regra  $R_j$  atribuiu um  $r$ -tupla de graus de confiança (e suporte):  $(\left[ \underline{Cnf}_1^j, \overline{Cnf}_1^j \right], \dots, \left[ \underline{Cnf}_r^j, \overline{Cnf}_r^j \right])$ ;
  - 4: **for**  $C = C_1$  to  $C_M$  **do**
  - 5:   Selecionar o conjunto de regras  $R_j$  que atribuiu a classe  $C$ ;
  - 6:    $\alpha =$  número de regras que atribuíram a classe  $C$ ;
  - 7:   **for**  $t = 1$  to  $r$  **do**
  - 8:     **for**  $L = 1$  to  $S$  **do**
  - 9:      **for**  $j = 1$  to  $\alpha$  **do**
  - 10:       Calcular  $\left[ \underline{k}_{jCnfIO_L}^C, \overline{k}_{jCnfIO_L}^C \right] = IO(IO_L(A'_1, A_{j1}), \dots, IO_L(A'_n, A_{jn})) \cdot Cnf_t^j$ ;
  - 11:      **end for**
  - 12:      Calcular  $\underline{K}_{CnfIO_L}^C = \max_{j=1}^{\alpha} k_{jIO_L}^C$ ;
  - 13:      Calcular  $\overline{K}_{CnfIO_L}^C = \max_{j=1}^{\alpha} k_{jIO_L}^C$ ;
  - 14:     **end for**
  - 15:     Calcular  $\mathbb{K}^C = \frac{S \times r}{M} \left[ \underline{K}_{CnfIO_L}^C, \overline{K}_{CnfIO_L}^C \right]$ ;
  - 16:   **end for**
  - 17: **end for**
  - 18: Seja  $C = \arg \max \mathbb{K}^C$ .
- 

### 3.4 Usando Funções *Penalty* (penalidade) para Escolher a Melhor Classe em Conjuntos de Regras *Fuzzy*

Um fator chave no Algoritmo 2 é a função de agregação  $M$  selecionada para o Passo 1. Se, para um determinado sistema de classificação, executamos esse algoritmo várias vezes, cada um deles com uma função de agregação diferente, podemos ter resultados diferentes, gerando *ensembles* (conjuntos de classificadores *fuzzy*) intervalares. No algoritmo 3 proposto a seguir adotaremos a saída das  $M$  agregações pelo ponto médio do intervalo.

Esse fato nos obriga a fornecer um sistema de consenso para selecionar a classe final resultante. Para esse sistema, adaptamos o método de tomada de decisão apresentado em (BUSTINCE et al, 2014), que usa para as funções *penalty* na fase de exploração  $P_{\nabla}$  definidas sobre um produto cartesiano de reticulados.



Denotamos por  $C^*$  uma cadeia de elementos que pertencem a  $[0, 1]$  e consideramos um produto Cartesiano  $\mathcal{L}_m^* = \underbrace{C^* \times \dots \times C^*}_m$ . Denotamos por  $B_{y_q}$  o conjunto *fuzzy* em  $U$  de modo que todos os valores de associação sejam iguais para  $y_q \in [0, 1]$ , que é,  $B_{y_q}(u) = y_q \in [0, 1]$ , para todo  $u \in U$ . Considere  $\vec{Y} = (y_1, \dots, y_m)$ ,  $\vec{B}_Y = (B_{y_1}, \dots, B_{y_m}) \in CF(U)^m$ .

**Teorema 8** (ELKANO et al, 2017) *Seja  $K_i : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ , com  $i = 1, \dots, n$ , funções semi-contínuas quasi-convexas com um mínimo exclusivo em  $K_i(0) = 0$ , e  $\mathcal{D} : CF(U) \times CF(U) \rightarrow \mathbb{R}^+$  seja a distância entre conjuntos fuzzy, definidos, para todos  $X, Y \in CF(U)$ , por*

$$\mathcal{D}(A, B) = \sum_{i=1}^n |A(u_i) - B(u_i)|, \quad (48)$$

onde  $n = \text{card}(U)$ . Então o mapeamento  $P_{\nabla} : CF(U)^m \times \mathcal{L}_m^* \rightarrow \mathbb{R}^+$ , dado, para todo  $\vec{A} \in CF(U)^m, Y \in \mathcal{L}_m^*$ , por :

$$P_{\nabla}(\vec{A}, Y) = \sum_{q=1}^m K_q(\mathcal{D}(A_q, B_{y_q})) = \sum_{q=1}^m K_q \left( \sum_{p=1}^n |A_q(u_p) - y_q| \right) \quad (49)$$

é uma função *penalty* definida sobre o produto cartesiano de reticulados (lattices)  $\mathcal{L}_m^{*(n+1)}$ .

Observe que **(P1)** e **(P2)** da Definição 17 segue da BUSTINCE et al (2014, Teorema 6). A prova de **(P3)** Definição 17 é devido à soma das funções semi-convexas e semi-contínuas inferiores  $K_i$ .

No Algoritmo 2, nós deixamos o usuário escolher  $S$  graus de confiança intervalares e  $r$  índices de *overlap* intervalares. Então, para cada classe  $C_i \in \mathcal{C} = \{C_1, \dots, C_M\}$  temos um conjunto de  $S \times r$  valores numéricos. Consideramos um produto cartesiano de vários reticulados como classes em nosso problema. Além disso, cada reticulado possui  $S \times r$  elementos:

$$C_1 \times \dots \times C_M = (K_{Cfn1_{\mathcal{O}_1}}^{C_1}, \dots, K_{Cfnr_{\mathcal{O}_S}}^{C_1}) \times \dots \times (K_{Cfn1_{\mathcal{O}_1}}^{C_M}, \dots, K_{Cfnr_{\mathcal{O}_S}}^{C_M}).$$

Considere um conjunto de  $M$  classes e outro conjunto de  $M$  funções de agregação. Em (BUSTINCE et al, 2014) é proposto um método de otimização, de modo que, a partir de todas as regras de agregação de  $M$ , obtidas ao calcular as permutações com repetições das funções de agregação  $M$ , podemos selecionar a tupla de tal forma que, quando aplicamos os componentes da tupla (na sua ordem) aos componentes  $S \times r$  da classe correspondente, obtemos a tupla  $M$  - que minimiza a dissimilaridade entre as classes de entrada  $M$  e a tupla de números considerada.

Modificamos (BUSTINCE et al, 2014, Algoritmo 1) para o nosso problema (Algoritmo 3), aplicando a função *penalty* adotada  $P_{\nabla}$  (Equação (49)) para calcular o desvio

entre as saídas fornecidas pelas diferentes tuplas de funções de agregação e, em seguida, selecione aquela que minimiza essa dissimilaridade.

---

**Algoritmo 3**


---

**Entrada:** Usando o Algoritmo 2, calculamos:

$$\begin{aligned} & (K_{Cfn1\mathcal{O}_1}^{C_1}, \dots, K_{Cfnr\mathcal{O}_S}^{C_1}) \\ & \quad \vdots \\ & (K_{Cfn1\mathcal{O}_1}^{C_M}, \dots, K_{Cfnr\mathcal{O}_S}^{C_M}) \end{aligned}$$

**Saída:** Uma classe  $C \in \{C_1, \dots, C_M\}$ .

- 1: Selecionar uma função de *penalty*  $P_{\nabla}$  definida sobre o produto cartesiano de  $\mathbb{M}$  reticulados (Teorema 8);
- 2: Selecionar uma  $\mathbb{M}$ -tupla  $(M_1, \dots, M_{\mathbb{M}})$  de funções de agregação idempotentes;
- 3: Calcular todas as permutações com repetições  $(M_{\sigma(1)}, \dots, M_{\sigma(\mathbb{M})})$  da tupla  $(M_1, \dots, M_{\mathbb{M}})$ .
- 4: **for** cada permutação do passo 3 **do**

$$m_{\sigma(1)}^{C_1} = M_{\sigma(1)}(K_{Cfn1\mathcal{O}_1}^{C_1}, \dots, K_{Cfnr\mathcal{O}_S}^{C_1})$$
- 5:
$$\dots$$

$$m_{\sigma(\mathbb{M})}^{C_M} = M_{\sigma(\mathbb{M})}(K_{Cfn1\mathcal{O}_1}^{C_M}, \dots, K_{Cfnr\mathcal{O}_S}^{C_M})$$
- 6: Calcular a função de *penalty*

$$P_{\nabla}(C_1, \dots, C_M, (m_{\sigma(1)}^{C_1}, \dots, m_{\sigma(\mathbb{M})}^{C_M})); \quad (50)$$

7: **end for**

- 8: Pegar a tupla de agregações  $(M_{m(\sigma(1))}^{C_1}, \dots, M_{m(\sigma(\mathbb{M}))}^{C_M})$ , que, com os valores obtidos no Passo 5, minimiza a Eq. (50);

9: Calcular

$$\begin{aligned} \mathbb{K}^{C_1} &= M_{m(\sigma(1))}^{C_1}(K_{Cfn1\mathcal{O}_1}^{C_1}, \dots, K_{Cfnr\mathcal{O}_S}^{C_1}) \\ & \quad \vdots \\ \mathbb{K}^{C_M} &= M_{m(\sigma(\mathbb{M}))}^{C_M}(K_{Cfn1\mathcal{O}_1}^{C_M}, \dots, K_{Cfnr\mathcal{O}_S}^{C_M}) \end{aligned}$$

- 10: Tem-se  $C = \arg \max_{\{C_1, \dots, C_M\}} \mathbb{K}^C$
- 

É imediato que:

**Proposição 7** (ELKANO et al, 2017) *Seja  $\mathcal{M} = (M_1, \dots, M_{\mathbb{M}})$  uma tupla de funções de agregação idempotente. Se executar o Algoritmo 2 várias vezes com diferentes funções de agregação idempotentes  $M \in \mathcal{M}$ , quando executar o algoritmo de consenso (Algoritmo 3), tem-se que:*

- (i) *Se  $M_1 = \dots = M_{\mathbb{M}}$ , então o resultado é dado pelo Algoritmo 2.*
- (ii) *Se, para todo  $i = 1, \dots, \mathbb{M}$ ,  $M_i$  é a média aritmética, então o Algoritmo 3 é o algoritmo de votação ponderada na tomada de decisão (BUSTINCE et al, 2012).*

**Observação 3** *Note-se que um raciocínio análogo sobre o funcionamento do Algoritmo 2 com diferentes funções de agregação  $M$  e a execução do Algoritmo 3 também podem ser feitas usando diferentes  $t$ -normas na Etapa 1 do Algoritmo 2. Não desenvolvemos este caso, pois é semelhante ao de diferentes funções de agregação  $M$ .*

## 4 AVALIAÇÃO EXPERIMENTAL

O objetivo desta seção é ilustrar os algoritmos desenvolvidos anteriormente por meio de um classificador baseado em regras *fuzzy* intervalares.

O exemplo consiste em:

1. Três classes  $C_1, C_2$  e  $C_3$ ;
2. Duas variáveis *fuzzy* (atributos):  $\chi_1 =$  índice de massa corporal (IMC) e  $\chi_2 =$ , idade. Os valores que eles podem receber,  $\chi_1$  e  $\chi_2$ , são dados pelos rótulos linguísticos baixo, médio e alto representados na Figura 11;
3. Vinte e cinco exemplos ou padrões, mostrados na Tabela 7. Além disso, a Figura 12 mostra os exemplos usados para aprender as regras *fuzzy* e o novo exemplo a ser classificado assim que as regras forem aprendidas. Para o processo de aprendizagem pode-se observar que temos quatro exemplos de classe 1 (símbolo de adição), dez exemplos de classe 2 (círculos) e onze exemplos de classe 3 (triângulos);
4. Cinco regras aprendidas (R1-R5 representadas em Fig. 12) a partir dos dados da Tabela 7:

Regra  $R_1$ : Se  $\chi_1$  é Médio e  $\chi_2$  é Baixo então Classe  $C_1$  com  $Cnf_1 = [?, ?]$

Regra  $R_2$ : Se  $\chi_1$  é Médio e  $\chi_2$  é Médio então Classe  $C_2$  com  $Cnf_2 = [?, ?]$

Regra  $R_3$ : Se  $\chi_1$  é Médio e  $\chi_2$  é Alto então Classe  $C_2$  com  $Cnf_3 = [?, ?]$

Regra  $R_4$ : Se  $\chi_1$  é Alto e  $\chi_2$  é Médio então Classe  $C_3$  com  $Cnf_4 = [?, ?]$

Regra  $R_5$ : Se  $\chi_1$  é Alto e  $\chi_2$  é Alto então Classe  $C_3$  com  $Cnf_5 = [?, ?]$

A partir da análise da Tabela 7 deduzimos que os 25 padrões (exemplos) são da forma:

$$\chi_l = (\chi_{l1}, \chi_{l2})$$

com  $l = 1, \dots, 25$  em que  $\chi_{l1}$  representa o valor da variável  $\chi_1$  (Massa Corporal) no exemplo  $l$  e  $\chi_{l2}$  (Idade) é o valor da variável  $\chi_2$  no exemplo  $l$ .

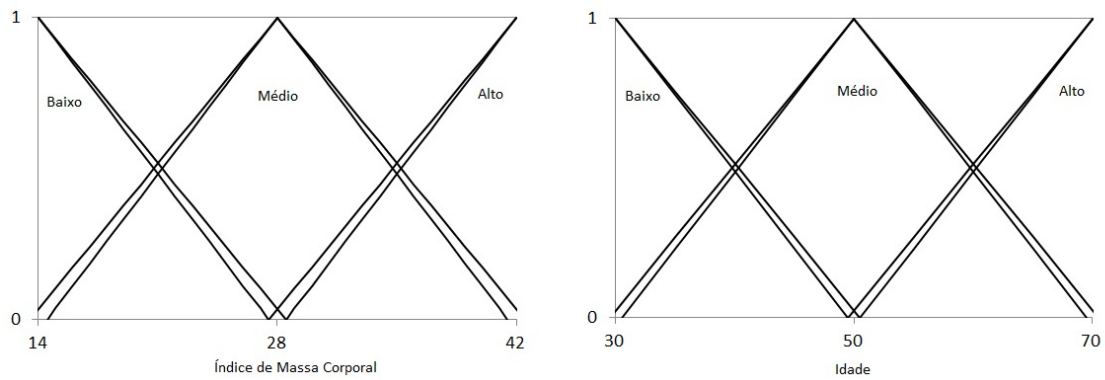


Figura 11: Termos Linguísticos para Massa Corporal e Idade

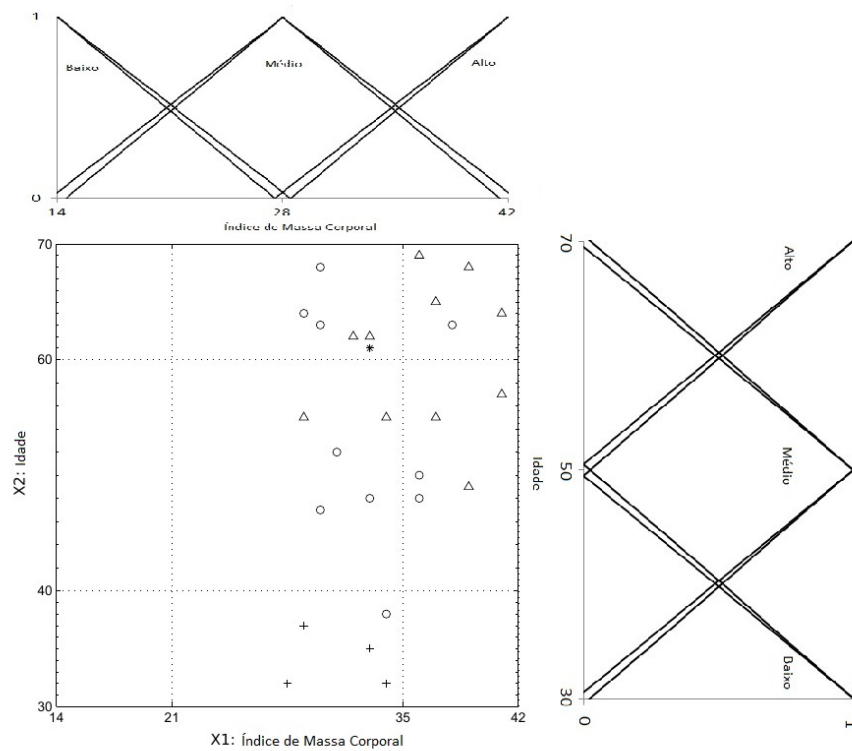


Figura 12: Regras

Tabela 7: Exemplos usados para treinamento

Id Exemplo	$\chi_1 = \text{Massa Corporal}$	$\chi_2 = \text{Idade}$	Classe
1	28	32	1
2	34	32	1
3	33	35	1
4	29	37	1
5	34	38	2
6	30	47	2
7	33	48	2
8	31	52	2
9	29	55	3
10	34	55	3
11	30	63	2
12	33	62	3
13	29	64	2
14	32	62	3
15	30	68	2
16	36	48	2
17	39	49	3
18	36	50	2
19	37	55	3
20	41	57	3
21	38	63	2
22	37	65	3
23	41	64	3
24	39	68	3
25	36	69	3

#### 4.1 Calculando o Grau de Confiança Intervalar para Cada Regra

Nesta subseção construímos o grau de confiança intervalar  $[Cnf_q, \overline{Cnf}_q]$  para cada regra ( $q = 1, \dots, 5$ ). As associações dos valores da Tabela 7 para os conjuntos representados na Figura 11 são dados na Tabela 8.

Tabela 8: Valores Fuzzy Intervalares

Id Exemplo	$\chi_1 = \text{Massa Corporal}$			$\chi_2 = \text{Idade}$			Classe
	Baixo	Médio	Alto	Baixo	Médio	Alto	
1	[0.000, 0.000]	[1.000, 1.000]	[0.000, 0.000]	[0.897, 0.902]	[0.077, 0.122]	[0.000, 0.000]	1.000
2	[0.000, 0.000]	[0.556, 0.586]	[0.407, 0.448]	[0.897, 0.902]	[0.077, 0.122]	[0.000, 0.000]	1.000
3	[0.000, 0.000]	[0.630, 0.655]	[0.333, 0.379]	[0.744, 0.756]	[0.231, 0.268]	[0.000, 0.000]	1.000
4	[0.000, 0.000]	[0.926, 0.931]	[0.037, 0.103]	[0.641, 0.659]	[0.333, 0.366]	[0.000, 0.000]	1.000
5	[0.000, 0.000]	[0.556, 0.586]	[0.407, 0.448]	[0.590, 0.610]	[0.385, 0.415]	[0.000, 0.000]	2.000
6	[0.000, 0.000]	[0.852, 0.862]	[0.111, 0.172]	[0.128, 0.171]	[0.846, 0.854]	[0.000, 0.000]	2.000
7	[0.000, 0.000]	[0.630, 0.655]	[0.333, 0.379]	[0.077, 0.122]	[0.897, 0.902]	[0.000, 0.000]	2.000
8	[0.000, 0.000]	[0.778, 0.793]	[0.185, 0.241]	[0.000, 0.000]	[0.897, 0.902]	[0.077, 0.122]	2.000
9	[0.000, 0.000]	[0.926, 0.931]	[0.037, 0.103]	[0.000, 0.000]	[0.744, 0.756]	[0.231, 0.268]	3.000
10	[0.000, 0.000]	[0.556, 0.586]	[0.407, 0.448]	[0.000, 0.000]	[0.744, 0.756]	[0.231, 0.268]	3.000
11	[0.000, 0.000]	[0.852, 0.862]	[0.111, 0.172]	[0.000, 0.000]	[0.333, 0.366]	[0.641, 0.659]	2.000
12	[0.000, 0.000]	[0.630, 0.655]	[0.333, 0.379]	[0.000, 0.000]	[0.385, 0.415]	[0.590, 0.610]	3.000
13	[0.000, 0.000]	[0.926, 0.931]	[0.037, 0.103]	[0.000, 0.000]	[0.282, 0.317]	[0.692, 0.707]	2.000
14	[0.000, 0.000]	[0.704, 0.724]	[0.259, 0.310]	[0.000, 0.000]	[0.385, 0.415]	[0.590, 0.610]	3.000
15	[0.000, 0.000]	[0.852, 0.862]	[0.111, 0.172]	[0.000, 0.000]	[0.077, 0.122]	[0.897, 0.902]	2.000
16	[0.000, 0.000]	[0.407, 0.448]	[0.556, 0.586]	[0.077, 0.122]	[0.897, 0.902]	[0.000, 0.000]	2.000
17	[0.000, 0.000]	[0.185, 0.241]	[0.778, 0.793]	[0.026, 0.073]	[0.949, 0.951]	[0.000, 0.000]	3.000
18	[0.000, 0.000]	[0.407, 0.448]	[0.556, 0.586]	[0.000, 0.000]	[1.000, 1.000]	[0.000, 0.000]	2.000
19	[0.000, 0.000]	[0.333, 0.379]	[0.630, 0.655]	[0.000, 0.000]	[0.744, 0.756]	[0.231, 0.268]	3.000
20	[0.000, 0.000]	[0.037, 0.103]	[0.926, 0.931]	[0.000, 0.000]	[0.641, 0.659]	[0.333, 0.366]	3.000
21	[0.000, 0.000]	[0.259, 0.310]	[0.704, 0.724]	[0.000, 0.000]	[0.333, 0.366]	[0.641, 0.659]	2.000
22	[0.000, 0.000]	[0.333, 0.379]	[0.630, 0.655]	[0.000, 0.000]	[0.231, 0.268]	[0.744, 0.756]	3.000
23	[0.000, 0.000]	[0.037, 0.103]	[0.926, 0.931]	[0.000, 0.000]	[0.282, 0.317]	[0.692, 0.707]	3.000
24	[0.000, 0.000]	[0.185, 0.241]	[0.778, 0.793]	[0.000, 0.000]	[0.077, 0.122]	[0.897, 0.902]	3.000
25	[0.000, 0.000]	[0.407, 0.448]	[0.556, 0.586]	[0.000, 0.000]	[0.026, 0.073]	[0.949, 0.951]	3.000

Levando em conta a Tabela 7, a distribuição das regras entre as três classes e a Tabela 8, calculamos os graus de confiança intervalar das regras pelo Algoritmo 1 da seguinte forma:

Passo 1 Consideramos o *overlap* intervalar  $IO = T_p$  e o índice de *overlap* intervalar  $IO_\pi$  dado pela Definição 25; que é, se  $A \in CFI(U)$  e  $B \in CFI(U)$  então

$$IO_\pi(A, B) = \frac{1}{m} \sum_{i=1}^m A(u_i) \cdot B(u_i)$$

Passo 2 Classificamos os padrões (exemplos) em três classes. Neste caso, os primeiros quatro são  $C_1$  (ver Tabela 7)

Passo 3 Para  $q = 1$  até  $p = 5$  (ou seja, para todas as regras), e iniciando com  $q = 1$ :

Passo 4 Usamos o  $s = 4$ , relacionando aos exemplos que pertencem a classe  $C_1$  (ver Tabela 7)

Passo 5 Para  $j = 1$  até  $s = 4$  faça:

Passos 6 e 7 Calculamos o *Interval matching degree* levando em conta os valores da Tabela 8, iniciando o  $j = 1$ , conforme a Equação (39), formando o conjunto *fuzzy* intervalar:

$$\begin{aligned} c_1([\underline{\chi}_1, \bar{\chi}_1]) &= T_p(A_{11}(\chi_{11}), A_{12}(\chi_{12})) = A_{11}(28) \cdot A_{12}(32) = [1.000, 1.000] \cdot [0.897, 0.902] = [0.897, 0.902] \\ c_2([\underline{\chi}_2, \bar{\chi}_2]) &= T_p(A_{11}(\chi_{21}), A_{12}(\chi_{22})) = A_{11}(34) \cdot A_{12}(32) = [0.556, 0.586] \cdot [0.897, 0.902] = [0.499, 0.529] \\ c_3([\underline{\chi}_3, \bar{\chi}_3]) &= T_p(A_{11}(\chi_{31}), A_{12}(\chi_{32})) = A_{11}(33) \cdot A_{12}(35) = [0.630, 0.655] \cdot [0.744, 0.756] = [0.468, 0.495] \\ c_4([\underline{\chi}_4, \bar{\chi}_4]) &= T_p(A_{11}(\chi_{41}), A_{12}(\chi_{42})) = A_{11}(26) \cdot A_{12}(37) = [0.926, 0.931] \cdot [0.641, 0.659] = [0.594, 0.613] \end{aligned}$$

onde  $A_{11}$  é o conjunto Médio e  $A_{12}$  é o conjunto Baixo em  $R_1$ .

Passo 8 Construimos o conjunto *fuzzy* intervalar em  $U$ :

$$\begin{aligned} \mathbb{C}_{14} &= \{(u_1, c_1([\underline{\chi}_1, \bar{\chi}_1])), (u_2, c_2([\underline{\chi}_2, \bar{\chi}_2])), (u_3, c_3([\underline{\chi}_3, \bar{\chi}_3])), (u_4, c_4([\underline{\chi}_4, \bar{\chi}_4]))\} = \\ &= \{(u_1, [0.897, 0.902]), (u_2, [0.499, 0.529]), (u_3, [0.468, 0.495]), (u_4, [0.594, 0.613]), \\ & \dots, (u_{25}, [0.000, 0.000])\} \end{aligned}$$

Passo 9 Para  $l = 1$  até  $m = 25$ , iniciando com  $l = 1$

Passos 10 e 11 Calculamos:

$$\begin{aligned}
c_5([\underline{\chi}_5, \overline{\chi}_5]) &= T_p(A_{11}(\chi_{51}), A_{12}(\chi_{52})) = A_{11}(34) \cdot A_{12}(38) = [0.556, 0.586] \cdot [0.590, 0.610] = [0.328, 0.357] \\
c_6([\underline{\chi}_6, \overline{\chi}_6]) &= T_p(A_{11}(\chi_{61}), A_{12}(\chi_{62})) = A_{11}(30) \cdot A_{12}(47) = [0.852, 0.862] \cdot [0.128, 0.171] = [0.109, 0.147] \\
c_7([\underline{\chi}_7, \overline{\chi}_7]) &= T_p(A_{11}(\chi_{71}), A_{12}(\chi_{72})) = A_{11}(33) \cdot A_{12}(48) = [0.630, 0.655] \cdot [0.077, 0.122] = [0.048, 0.080] \\
c_8([\underline{\chi}_8, \overline{\chi}_8]) &= T_p(A_{11}(\chi_{81}), A_{12}(\chi_{82})) = A_{11}(31) \cdot A_{12}(52) = [0.778, 0.793] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_9([\underline{\chi}_9, \overline{\chi}_9]) &= T_p(A_{11}(\chi_{91}), A_{12}(\chi_{92})) = A_{11}(29) \cdot A_{12}(55) = [0.926, 0, 931] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{10}([\underline{\chi}_{10}, \overline{\chi}_{10}]) &= T_p(A_{11}(\chi_{101}), A_{12}(\chi_{102})) = A_{11}(34) \cdot A_{12}(55) = [0.556, 0.586] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{11}([\underline{\chi}_{11}, \overline{\chi}_{11}]) &= T_p(A_{11}(\chi_{111}), A_{12}(\chi_{112})) = A_{11}(30) \cdot A_{12}(63) = [0.852, 0.862] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{12}([\underline{\chi}_{12}, \overline{\chi}_{12}]) &= T_p(A_{11}(\chi_{121}), A_{12}(\chi_{122})) = A_{11}(33) \cdot A_{12}(62) = [0.630, 0.655] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{13}([\underline{\chi}_{13}, \overline{\chi}_{13}]) &= T_p(A_{11}(\chi_{131}), A_{12}(\chi_{132})) = A_{11}(29) \cdot A_{12}(64) = [0.926, 0.931] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{14}([\underline{\chi}_{14}, \overline{\chi}_{14}]) &= T_p(A_{11}(\chi_{141}), A_{12}(\chi_{142})) = A_{11}(32) \cdot A_{12}(62) = [0.704, 0.724] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{15}([\underline{\chi}_{15}, \overline{\chi}_{15}]) &= T_p(A_{11}(\chi_{151}), A_{12}(\chi_{152})) = A_{11}(30) \cdot A_{12}(68) = [0.852, 0.862] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{16}([\underline{\chi}_{16}, \overline{\chi}_{16}]) &= T_p(A_{11}(\chi_{161}), A_{12}(\chi_{162})) = A_{11}(36) \cdot A_{12}(48) = [0.407, 0.448] \cdot [0.077, 0.122] = [0.031, 0.055] \\
c_{17}([\underline{\chi}_{17}, \overline{\chi}_{17}]) &= T_p(A_{11}(\chi_{171}), A_{12}(\chi_{172})) = A_{11}(39) \cdot A_{12}(49) = [0.185, 0.241] \cdot [0.026, 0.073] = [0.005, 0.018] \\
c_{18}([\underline{\chi}_{18}, \overline{\chi}_{18}]) &= T_p(A_{11}(\chi_{181}), A_{12}(\chi_{182})) = A_{11}(36) \cdot A_{12}(50) = [0.407, 0.448] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{19}([\underline{\chi}_{19}, \overline{\chi}_{19}]) &= T_p(A_{11}(\chi_{191}), A_{12}(\chi_{192})) = A_{11}(37) \cdot A_{12}(55) = [0.333, 0.379] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{20}([\underline{\chi}_{20}, \overline{\chi}_{20}]) &= T_p(A_{11}(\chi_{201}), A_{12}(\chi_{202})) = A_{11}(41) \cdot A_{12}(57) = [0.037, 0.103] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{21}([\underline{\chi}_{21}, \overline{\chi}_{21}]) &= T_p(A_{11}(\chi_{211}), A_{12}(\chi_{212})) = A_{11}(38) \cdot A_{12}(63) = [0.259, 0.310] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{22}([\underline{\chi}_{22}, \overline{\chi}_{22}]) &= T_p(A_{11}(\chi_{221}), A_{12}(\chi_{222})) = A_{11}(37) \cdot A_{12}(65) = [0.333, 0.379] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{23}([\underline{\chi}_{23}, \overline{\chi}_{23}]) &= T_p(A_{11}(\chi_{231}), A_{12}(\chi_{232})) = A_{11}(41) \cdot A_{12}(64) = [0.037, 0.103] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{24}([\underline{\chi}_{24}, \overline{\chi}_{24}]) &= T_p(A_{11}(\chi_{241}), A_{12}(\chi_{242})) = A_{11}(39) \cdot A_{12}(68) = [0.185, 0.241] \cdot [0.000, 0.000] = [0.000, 0.000] \\
c_{25}([\underline{\chi}_{25}, \overline{\chi}_{25}]) &= T_p(A_{11}(\chi_{251}), A_{12}(\chi_{252})) = A_{11}(36) \cdot A_{12}(69) = [0.407, 0.448] \cdot [0.000, 0.000] = [0.000, 0.000]
\end{aligned}$$

**Passo 12** Construímos o conjunto *fuzzy* intervalar em  $U$ :

$$\begin{aligned}
\mathbb{C}_{1,25} &= \{(u_1, c_1(\underline{\chi}_1, \overline{\chi}_1)), \dots, (u_{25}, c_m(\underline{\chi}_{25}, \overline{\chi}_{25}))\} = \\
&\{(u_1, [0.897, 0.902]), (u_2, [0.499, 0.529]), (u_3, [0.468, 0.495]), (u_4, [0.594, 0.613]), \\
&(u_5, [0.328, 0.357]), (u_6, [0.109, 0.147]), (u_7, [0.048, 0.080]), (u_8, [0.000, 0.000]), \\
&(u_9, [0.000, 0.000]), (u_{10}, [0.000, 0.000]), (u_{11}, [0.000, 0.000]), (u_{12}, [0.000, 0.000]), \\
&(u_{13}, [0.000, 0.000]), (u_{14}, [0.000, 0.000]), (u_{15}, [0.000, 0.000]), (u_{16}, [0.031, 0.055]), \\
&(u_{17}, [0.005, 0.018]), (u_{18}, [0.000, 0.000]), (u_{19}, [0.000, 0.000]), (u_{20}, [0.000, 0.000]), \\
&(u_{21}, [0.000, 0.000]), (u_{22}, [0.000, 0.000]), (u_{23}, [0.000, 0.000]), (u_{24}, [0.000, 0.000]), \\
&(u_{25}, [0.000, 0.000])\}
\end{aligned}$$

**Passo 13** Pela Eq. (42) temos:

$$\begin{aligned}
[Cnf(R_1), \overline{Cnf}(R_1)] &= \frac{IO(\mathbb{C}_{14}, U)}{IO(\mathbb{C}_{125}, U)} = \\
&\frac{\frac{1}{25} \sum_{i=1}^4 c_i([\chi_i])}{\frac{1}{25} \sum_{i=1}^{25} c_i([\chi_i])} = \\
&\frac{\frac{1}{25} ([0.897, 0.902] + [0.499, 0.529] + [0.468, 0.495] + [0.594, 0.613])}{\frac{1}{25} ([0.897, 0.902] + [0.499, 0.529] + [0.468, 0.495] + [0.594, 0.613] + [0.328, 0.357] + [0.109, 0.147] + [0.048, 0.080] + [0.031, 0.055] + [0.005, 0.018])} = \\
&\frac{[2.458, 2.540]}{[2.979, 3.197]} = \\
&[0.769, 0.853]
\end{aligned}$$



No final do algoritmo; repetimos os passos anteriores para  $q = 2, 3, 4, 5$ , e temos os seguintes resultados para as outras regras:  $\left[ \overline{Cnf(R_2)}, \overline{Cnf(R_2)} \right] = [0.511, 0.619]$ ;  $\left[ \overline{Cnf(R_3)}, \overline{Cnf(R_3)} \right] = [0.464, 0.546]$ ;  $\left[ \overline{Cnf(R_4)}, \overline{Cnf(R_4)} \right] = [0.493, 0.633]$ ;  $\left[ \overline{Cnf(R_5)}, \overline{Cnf(R_5)} \right] = [0.743, 0.900]$  (ver segunda coluna em Tabela 9).

No Passo 1 do Algoritmo 1, usamos os índices de *overlap* intervalares construídos com o Definição 25 da seguinte forma:

1.  $\mathcal{IO}(A, B) = \sqrt{AB}$  e  $M$  a média aritmética; isso é

$$\mathcal{IO}_{\sqrt{}}(A, B) = \frac{1}{m} \sum_{i=1}^m \sqrt{A(u_i) \cdot B(u_i)},$$

2.  $\mathcal{IO}(A, B) = \frac{\sqrt{AB}}{\sqrt{AB+1-AB}}$  e a média aritmética; isso é,

$$\mathcal{IO}_{rational}(A, B) = \frac{1}{m} \sum_{i=1}^m \frac{\sqrt{A(u_i) \cdot B(u_i)}}{\sqrt{A(u_i) \cdot B(u_i) + 1 - A(u_i) \cdot B(u_i)}},$$

então recuperamos o grau de confiança intervalar  $Cnf$  apresentado nas colunas 3 e 4 respectivamente da Tabela 9.

Tabela 9: Confiança com diferentes *overlaps* intervalares

Regra	$\mathcal{IO}_{\pi}$	$\mathcal{IO}_{\sqrt{}}$	$\mathcal{IO}_{rational}$
1	[0.769, 0.853]	[0.648, 0.707]	[0.638, 0.741]
2	[0.511, 0.619]	[0.464, 0.534]	[0.445, 0.562]
3	[0.464, 0.546]	[0.391, 0.445]	[0.386, 0.478]
4	[0.493, 0.633]	[0.466, 0.574]	[0.443, 0.608]
5	[0.743, 0.900]	[0.705, 0.821]	[0.673, 0.861]

## 4.2 Inferência

Através do conjunto de regras abaixo:

- Regra  $R_1$ : Se  $\chi_1$  é Médio e  $\chi_2$  é Baixo então Classe  $C_1$  com  $Cnf_1$   
 Regra  $R_2$ : Se  $\chi_1$  é Médio e  $\chi_2$  é Médio então Classe  $C_2$  com  $Cnf_2$   
 Regra  $R_3$ : Se  $\chi_1$  é Médio e  $\chi_2$  é Alto então Classe  $C_2$  com  $Cnf_3$   
 Regra  $R_4$ : Se  $\chi_1$  é Alto e  $\chi_2$  é Médio então Classe  $C_3$  com  $Cnf_4$   
 Regra  $R_5$ : Se  $\chi_1$  é Alto e  $\chi_2$  é Alto então Classe  $C_3$  com  $Cnf_5$

pretendemos determinar à qual classe pertence o seguinte registro, para isso usamos o Algoritmo 2:

Índice de massa corporal = 33 e Idade = 61

Na Fig. (12) o novo registro a ser classificado (IMC = 33 e Idade = 61) é representado com o asterisco e, julgando por sua posição acreditamos que deva pertencer à classe três, já que é cercado de outros exemplos desta classe.

A partir da Fig. (11):

1. construímos o conjunto  $A'_1$  e  $A'_2$  como segue:

$$A'_1 = \{(14, 0), (15, 0), \dots, (32, 0), (33, 1), (34, 0), \dots, (42, 0)\}$$

$$A'_2 = \{(30, 0), (31, 0), \dots, (60, 0), (61, 1), (62, 0), \dots, (70, 0)\}$$

2. calculamos os valores de pertinência intervalar do IMC 33 aos conjuntos Baixo, Médio e Alto. Os resultados são:

$$\text{Baixo}(33) = [0.000, 0.000]; \text{Médio}(33) = [0.630, 0.655] \text{ e } \text{Alto}(33) = [0.333, 0.379]$$

3. calculamos os valores de pertinência intervalar da variável idade para os conjuntos; ou seja,

$$\text{Baixo}(61) = [0.000, 0.000]; \text{Médio}(61) = [0.436, 0.463] \text{ e } \text{Alto}(61) = [0.538, 0.561].$$

Para executar o Algoritmo 2 temos:

- $IO = T_p$ ,
- $M = \max$ ,
- conjunto de índices de *overlap* intervalar  $S = 3 \{IO_\pi, IO_\surd, IO_{rational}\}$ ,
- e os graus de confiança intervalar  $Cnf_j$  ( $j = 1, \dots, 5$ ) calculados com Algoritmo 1 e representados nas colunas 2, 3 e 4 na Tabela 9.

Apresentamos os resultados obtidos com o Algoritmo 2 neste o experimento na Tabela 12.

### 4.3 Consenso

Uma vez que o Algoritmo 2 foi executado com a função de agregação  $M_1 = \max = m_1$ , executamos este algoritmo mais duas vezes. Um com a agregação  $M_2$  dada pelo operador *OWA* correspondente ao quantificador *pelo menos uma metade*; isto é,  $\omega = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0, 0, 0)$  e outro com  $M_3$  dado pelo operador *OWA* correspondente ao quantificador *o maior número possível*; isto é,  $\omega = (0, 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ .

Em seguida, executamos o algoritmo de consenso, Algoritmo 3, com a função *penalty* dada na Eq. (51) e as tuplas construídas calculando as permutações com a repetição

Tabela 10:  $M$  Agregações

	Classe 2	Classe 3
M1	0,2105	0,3358
M2	0,1297	0,3694
M3	0,1309	0,1974

Tabela 11: Resultado Algoritmo 3 para o ponto médio do intervalo

C1	C2	C3	C1	C2	C3	Total
M1	M2	M3	0	0,1297	0,1974	0,3271
M2	M2	M3	0	0,1297	0,1974	0,3271
M3	M2	M3	0	0,1297	0,1974	0,3271

de  $(M_1, M_2, M_3)$  para o ponto médio do intervalo. Então nós temos 27 combinações possíveis das 3 funções de agregação. Na Tabela 11 apresentamos os valores numéricos obtidos com a função *penalty* utilizada para cada uma das 3-tuplas consideradas. Observe que o mínimo é dado por 0,3271 correspondente à combinação de agregações:  $(M_1, M_2, M_3)$

$$P_{\nabla}(C_1, \dots, C_M, (m_{\sigma(1)}^{C_1}, \dots, m_{\sigma(M)}^{C_M})) = \sum_{i=1}^M \left( \sum_{j=1, \dots, r; L=1, \dots, S} |K_{C^i f_{nj} \sigma_L}^{C_i} - m_{\sigma(i)}^{C_i}| \right)^2 \quad (51)$$

Então temos:  $(\mathbb{K}^{C_1} = 0, \mathbb{K}^{C_2} = 0.1297, \mathbb{K}^{C_3} = 0, 1974)$ . Pelo Passo 5 no Algoritmo 3 o resultado da classe é  $C_3$ .

#### 4.4 Resultados

Ao longo desta dissertação, propusemos um método para resolver os problemas de ignorância na coleta dos dados, ou imprecisão dos valores, para posterior tomada de decisão, então a Tabela 12 e o Algoritmo 2 nos permitem estudar os resultados da classificação de um dado fato de duas maneiras diferentes:

- por meio do algoritmo clássico (por colunas);
- ou por meio de um família de diferentes medidas para representar a sobreposição entre os antecedentes das regras e o consequente (fato) considerado (por linhas).

A partir da análise da primeira coluna da Tabela 12 deduzimos que o consequente (fato) que estamos considerando neste exemplo pertence a  $C_2$ . No entanto, a partir da análise das outras colunas (exceto a quarta), deduzimos que o fato pertence à classe  $C_3$  ( $C = \arg \max \mathbb{K}^C$ ).

Neste experimento, por simplicidade, adotamos os dados decompostos em dois conjuntos de dados, com os valores do limite inferior e superior do intervalo, como citado an-

Tabela 12: Resultados intervalares do Algoritmo 2

Classe	$C_{nf}$ Construído a partir do Algoritmo 2 usando $ZO_{\pi}$				$C^F = C_{nf}$ Construído a partir do Algoritmo 2 usando $ZO_{rational}$			
	$IO_{\pi}$	$ZO_{\gamma}$	$ZO_{rational}$	$ZO_{\pi}$	$IO_{\gamma}$	$ZO_{rational}$	$ZO_{rational}$	
C1	$K_{ChI\sigma_{\pi}}^{C1} = 0$	$K_{ChI\sigma_{\gamma}}^{C1} = 0$	$K_{ChI\sigma_{rational}}^{C1} = 0$	$K_{ChI\sigma_{\pi}}^{C1} = 0$	$K_{ChI\sigma_{\gamma}}^{C1} = 0$	$K_{ChI\sigma_{rational}}^{C1} = 0$	$K_{ChI\sigma_{rational}}^{C1} = 0$	
C2	$K_{ChI\sigma_{\pi}}^{C2} = V(0.1402, 0.1878), [0.1572, 0.2007]$	$K_{ChI\sigma_{\gamma}}^{C2} = V(0.2677, 0.3408), [0.2699, 0.3311]$	$K_{ChI\sigma_{rational}}^{C2} = V(0.2097, 0.2793), [0.2130, 0.2726]$	$K_{ChI\sigma_{\pi}}^{C2} = V(0.1221, 0.1707), [0.1310, 0.1757]$	$K_{ChI\sigma_{\gamma}}^{C2} = V(0.2330, 0.3099), [0.2254, 0.2898]$	$K_{ChI\sigma_{rational}}^{C2} = V(0.2330, 0.3099), [0.2254, 0.2898]$	$K_{ChI\sigma_{rational}}^{C2} = V(0.1826, 0.2539), [0.1776, 0.2385]$	
C3	$K_{ChI\sigma_{\pi}}^{C3} = V(0.0716, 0.1112), [0.1334, 0.1915]$	$K_{ChI\sigma_{\gamma}}^{C3} = V(0.1878, 0.2653), [0.3148, 0.4152]$	$K_{ChI\sigma_{rational}}^{C3} = V(0.1474, 0.2201), [0.2456, 0.3429]$	$K_{ChI\sigma_{\pi}}^{C3} = V(0.0644, 0.1069), [0.1207, 0.1831]$	$K_{ChI\sigma_{\gamma}}^{C3} = V(0.1690, 0.2551), [0.2819, 0.3970]$	$K_{ChI\sigma_{rational}}^{C3} = V(0.1690, 0.2551), [0.2819, 0.3970]$	$K_{ChI\sigma_{rational}}^{C3} = V(0.1326, 0.2223), [0.1326, 0.2223]$	
Classe	C3		C3	C2		C3	C3	

Tabela 13: Resultados do Algoritmo 2 Limite Inferior do Intervalo

Classe	$C_{nf}$ Construído a partir do Algoritmo 2 usando $O_{\pi}$				$C_{nf}$ Construído a partir do Algoritmo 2 usando $O_{rational}$			
	$O_{\pi}$	$O_{\gamma}$	$O_{rational}$	$O_{\pi}$	$O_{\gamma}$	$O_{rational}$	$O_{rational}$	
C1	$K_{ChI\sigma_{\pi}}^{C1} = 0$	$K_{ChI\sigma_{\gamma}}^{C1} = 0$	$K_{ChI\sigma_{rational}}^{C1} = 0$	$K_{ChI\sigma_{\pi}}^{C1} = 0$	$K_{ChI\sigma_{\gamma}}^{C1} = 0$	$K_{ChI\sigma_{rational}}^{C1} = 0$	$\mathbb{K}^{C1} = 0$	
C2	$K_{ChI\sigma_{\pi}}^{C2} = V(0.1402, 0.1572)$	$K_{ChI\sigma_{\gamma}}^{C2} = V(0.2677, 0.2699)$	$K_{ChI\sigma_{rational}}^{C2} = V(0.2097, 0.2130)$	$K_{ChI\sigma_{\pi}}^{C2} = V(0.1221, 0.1310)$	$K_{ChI\sigma_{\gamma}}^{C2} = V(0.2330, 0.2250)$	$K_{ChI\sigma_{rational}}^{C2} = V(0.1826, 0.1776)$	$\mathbb{K}^{C2} = 0.2699$	
C3	$K_{ChI\sigma_{\pi}}^{C3} = V(0.0716, 0.1334)$	$K_{ChI\sigma_{\gamma}}^{C3} = V(0.1878, 0.3148)$	$K_{ChI\sigma_{rational}}^{C3} = V(0.1474, 0.2456)$	$K_{ChI\sigma_{\pi}}^{C3} = V(0.0644, 0.1207)$	$K_{ChI\sigma_{\gamma}}^{C3} = V(0.1690, 0.2849)$	$K_{ChI\sigma_{rational}}^{C3} = V(0.1326, 0.2223)$	$\mathbb{K}^{C3} = 0.3148$	
Classe	C2		C3	C2		C3	C3	

Tabela 14: Resultados do Algoritmo 2 Limite Superior do Intervalo

Classe	$C_{nf}$ Construído a partir do Algoritmo 2 usando $O_{\pi}$				$C_{nf}$ Construído a partir do Algoritmo 2 usando $O_{rational}$			
	$O_{\pi}$	$O_{\gamma}$	$O_{rational}$	$O_{\pi}$	$O_{\gamma}$	$O_{rational}$	$O_{rational}$	
C1	$K_{ChI\sigma_{\pi}}^{C1} = 0$	$K_{ChI\sigma_{\gamma}}^{C1} = 0$	$K_{ChI\sigma_{rational}}^{C1} = 0$	$K_{ChI\sigma_{\pi}}^{C1} = 0$	$K_{ChI\sigma_{\gamma}}^{C1} = 0$	$K_{ChI\sigma_{rational}}^{C1} = 0$	$\mathbb{K}^{C1} = 0$	
C2	$K_{ChI\sigma_{\pi}}^{C2} = V(0.1878, 0.2007)$	$K_{ChI\sigma_{\gamma}}^{C2} = V(0.3408, 0.3311)$	$K_{ChI\sigma_{rational}}^{C2} = V(0.2793, 0.2726)$	$K_{ChI\sigma_{\pi}}^{C2} = V(0.1707, 0.1757)$	$K_{ChI\sigma_{\gamma}}^{C2} = V(0.3099, 0.2898)$	$K_{ChI\sigma_{rational}}^{C2} = V(0.2539, 0.2385)$	$\mathbb{K}^{C2} = 0.3408$	
C3	$K_{ChI\sigma_{\pi}}^{C3} = V(0.1112, 0.1915)$	$K_{ChI\sigma_{\gamma}}^{C3} = V(0.2653, 0.4152)$	$K_{ChI\sigma_{rational}}^{C3} = V(0.2201, 0.3429)$	$K_{ChI\sigma_{\pi}}^{C3} = V(0.1069, 0.1831)$	$K_{ChI\sigma_{\gamma}}^{C3} = V(0.2551, 0.3970)$	$K_{ChI\sigma_{rational}}^{C3} = V(0.2116, 0.3279)$	$\mathbb{K}^{C3} = 0.4152$	
Classe	C2		C3	C3		C3	C3	

teriormente, como vemos nas Tabelas 13 e 14, se analisarmos os dados por colunas, então veremos que cada coluna corresponde aos dados obtidos com o algoritmo de classificação usando somente uma medida, ou seja, fixando um grau de confiança intervalar para as regras e escolhendo o conjunto de *overlap* intervalar com uma única sobreposição. Somente essa forma de classificar não nos permite tomar a decisão final, já que entre o topo e o piso a coluna  $O_\pi$  da medida de confiança  $Cn.f_{O_{rational}}$  possui divergência de classificação.

Se analisarmos as Tabelas 13 e 14 por linhas, o resultado é aquele obtido por meio do Algoritmo 2 com dois graus de confiança intervalar e três de *overlap* intervalar. Do nosso ponto de vista, devemos usar Algoritmo 2 e entender as Tabelas 13 e 14 por linhas, pois nos conjuntos de classificação (*ensembles*) sempre existem diferentes tipos de interações entre consequentes (fatos) e antecedentes, e devemos levar em conta todos deles. Isso não acontece nos algoritmos clássicos, que medem apenas a sobreposição de uma maneira.

A discussão que propusemos nesta dissertação foi desenvolver o mecanismo de raciocínio *fuzzy* intervalar que instrumentalize essa visão entre linhas e colunas, que podem ser aplicadas para processar os resultados quando um conjunto de classificadores (*ensembles*) são utilizados. Por ser necessário processar diversas funções de medidas, pode haver divergência nos resultados classificados e nestes casos um MRF-I, que possua um método de tratamento da ignorância inicial e, que leve em consideração diversas medidas *fuzzy* intervalar para tratar o conjunto das relações entre antecedentes e consequentes e posteriormente através da aplicação de uma função de consenso final trazer novamente a saída para uma classe específica, pode trazer uma melhor interpretação do sistema e resultado de decisão. Em consequência às razões previamente discutidas, e do que justificamos como consenso, nesse exemplo de classificação a saída com menor penalidade, ou seja  $C3$  é a que consideramos como solução.

## 5 CONCLUSÃO

As recentes obras aplicadas relacionadas aos conjuntos *fuzzy* intervalares (CFI), provaram ser uma ferramenta apropriada para modelar as incertezas do sistema e a ignorância na definição dos termos *fuzzy*. Nesta dissertação analisamos o comportamento do SCBRF, utilizando uma base de conhecimento (BC) simples, estendendo o trabalho realizado em (ELKANO et al, 2017), mantendo o método de aprendizado de regras e gerenciando as incertezas derivadas por uma margem de erro arbitrária em 0,5, buscamos resolver esses problemas de ignorância da coleta dos dados do sistema por meio da teoria CFI. Além disso, também consideramos apropriado fazer um estudo exaustivo usando diferentes operações da aritmética intervalar para calcular o grau de certeza das regras. Para isso, desenvolvemos 2 algoritmos relacionados a construção de SCBRFIs:

1. O Algoritmo 1, que implementa um método para construir medidas de confiança e suporte intervalar, com base em índices de *overlap*, que, por sua vez, podem ser construídos por funções de *overlap* intervalar;
2. O Algoritmo 2 implementa um novo MRF-I para SCBRF, usando diferentes índices de *overlap* com intervalos e funções de agregação para obter a classe selecionada através de um treinamento;
3. O Algoritmo 3 que gera um consenso da classificação através de uma função de *penalty* (penalidade).

O estudo experimental determinou que esta metodologia é uma solução adequada, utilizando como métodos de aprendizagem de regras os algoritmos Chi et al., obtendo um desempenho muito próximo do obtido no algoritmo clássico. Os resultados obtidos mostraram que o modelo proposto de MRF com o uso de intervalos (MRF-I) melhora o comportamento dos modelos básicos de SCBRF, reduzindo o impacto da ignorância na coleta das informações e mesmo assim gerenciando/classificando itens *fuzzy* com a mesma qualidade das soluções obtidas pelos sistemas *fuzzy* padrão. Isso nos permite concluir que a introdução de CFI melhora o comportamento de algoritmos que são conhecidos por já terem um bom desempenho com sistemas *fuzzy*.

Como resultados teóricos, tem-se os seguintes conceitos introduzidos: Definição 25, Equações (34), (35), (36), (37), (38), (44), (45), (46), (47) e Tabela 6.

Como trabalhos futuros podemos validar se o comportamento se mantém em outros tipos de conjuntos de dados, com maiores números de classes e regras, incluir novas medidas *fuzzy* e o uso de funções de (pré)-agregação no SCBRF-I.

## REFERÊNCIAS

AGGARWAL, C. C. **Data classification: algorithms and applications**, Minneapolis, USA: CRC Press, 2014.

BEDREGAL, B.; BUSTINCE, H.; PALMEIRA, E.; DIMURO, G. P.; Fernandez, J. **Generalized interval-valued OWA operators with interval weights derived from interval-valued overlap functions**, International Journal of Approximate Reasoning, v. 90, p. 1-16, 2017.

BEDREGAL, B. C.; DIMURO, G. P.; BUSTINCE, H.; BARRENECHEA, E. **New Results on Overlap and Grouping Functions**, [S.l.], Information Sciences, v. 249, p. 148-170, 2013.

BEDREGAL, B. C.; TAKAHASHI, A. **The Best Interval Representation of T-Norms and Automorphisms**, Fuzzy Sets and Systems, 157, 3220-3230, 2006.

BELIAKOV, G.; CALVO, T.; JAMES, S. **Consensus measures constructed from aggregation functions and fuzzy implications**, [S.l.], Knowledge-Based Systems, v. 55, p. 1-8, 2014.

BELIAKOV, G.; JAMES, S. **A penalty-based aggregation operator for non-convex intervals**, [S.l.], Knowledge-Based Systems, v. 70, p. 335-344, 2014.

BELIAKOV, G.; PRADERA, A.; CALVO, T. **Aggregation Functions: A Guide for Practitioners**, Berlin, Springer, 2007.

BUSTINCE, H. **Interval-valued Fuzzy Sets in Soft Computing**. International Journal of Computational Intelligence Systems, v. 3, n.2, p. 215-22, 2010.

BUSTINCE, H.; BARRENECHEA, E.; CALVO, T.; JAMES, S.; BELIAKOV, G. **Consensus in multi-expert decision making problems using penalty functions defined over a Cartesian product of lattices**, [S.l.], Information Fusion, v. 17, p. 56-64, 2014.

BUSTINCE, H.; BARRENECHEA, E.; PAGOLA, M. **Generation of interval-valued fuzzy and Atanassov's intuitionistic fuzzy connectives from fuzzy connectives and**



**from Ka operators: laws for conjunctions and disjunctions, amplitude**, [S.l.], International Journal of Intelligent Systems, v. 23, n. 6, p. 680-714, 2008.

BUSTINCE, H.; BELIAKOV, G.; DIMURO, G. P.; BEDREGAL, B.; MESIAR, R. **On the definition of penalty functions in data aggregation**, [S.l.], Fuzzy Sets and Systems, v. 323, p. 1-18, 2016.

BUSTINCE, H.; FERNANDEZ, J.; MESIAR, R.; MONTERO, J.; ORDUNA, R. **Overlap Functions**, [S.l.], Nonlinear Analysis: Theory, Methods & Applications, v. 72, n. 3-4, p. 1488-1499, 2010.

BUSTINCE, H.; FERNANDEZ, J.; MESIAR, R.; MONTERO, J.; ORDUNA, R. **Overlap Index, Overlap Functions and Migrativity**, [S.l.], Proceedings of IFSA/EUSFLAT Conference, p. 300-305, 2009.

BUSTINCE, H.; FERNANDEZ, J.; KOLESAROVA, A.; MESIAR, R. **Generation of linear orders for intervals by means of aggregation functions**, [S.l.], Fuzzy Sets and Systems, 2012.

BUSTINCE, H.; FERNANDEZ, J.; MESIAR, R.; PRADERA, A.; BELIAKOV, G. **Restricted dissimilarity functions and penalty functions**, Amsterdam, Proceedings of 2011 EUSFLAT-IFSA Joint Conference, n. 1, p. 79-85, 2011.

BUSTINCE, H.; PAGOLA, M.; MESIAR, R.; HULLERMEIER, E.; HERRERA, F. **Grouping, Overlaps, and Generalized Bientropic Functions for Fuzzy Modeling**, [S.l.], IEEE Transactions on Fuzzy Systems, v. 20, n. 3, p. 405-415, 2012.

CALVO, T.; BELIAKOV, G. **Aggregation functions based on penalties**, [S.l.], Fuzzy Sets and Systems, v. 161, n. 10, p. 1420-1436, 2010.

CALVO, T.; MESIAR, R.; YAGER, R. R. **Quantitative weights and aggregation**, [S.l.], IEEE Transactions on Fuzzy Systems, v. 12, n. 1, p. 62-69, 2004.

CHOQUET, G. **Theory of capacities**, [S.l.], l'Institut Fourier, v. 5, p. 131-295, 1953.

Cornelis, C.; Deschrijver, G.; Kerre, E. E. **Advances and challenges in interval-valued fuzzy logic**, Fuzzy Sets and Systems, v.157(5), p.622–627, 2006.

DE-BARROS, L. C.; BASSANEZI, R. C. **Temas de lógica fuzzy e biomatemática**, Grupo de Biomatemática, Instituto de Matemática, Estatística e Computação Científica (IMECC), Universidade Estadual de Campinas (UNICAMP)., 2010.

DENNEBERG, D. **Non Additive Measure and Integral**, Dordrecht, Kluwer Academic Publishers, 1994.

DESCHRIJVER, G.; CORNELIS, C.; KERRE, E. E. **On the representation of intuitionistic fuzzy t-norms and t-conorms**, [S.l.], IEEE Transactions on Fuzzy Systems, v. 12, n. 1, p. 45-61, 2004.

DESCHRIJVER, G. **A representation of t-norms in interval-valued L-fuzzy set theory**, Fuzzy Sets and Systems, v. 159, p. 1597-1618, 2008.

DIMURO, G. P. **On Interval Fuzzy Numbers**, Los Alamitos, IEEE, p. 3-8, 2011.

DIMURO, G. P.; BEDREGAL, B. **Archimedean overlap functions: The ordinal sum and the cancellation, idempotency and limiting properties**, [S.l.], Fuzzy Sets and Systems, v. 252, p. 39-54, 2014.

DIMURO, G. P.; BEDREGAL, B. **On residual implications derived from overlap functions**, [S.l.], Information Sciences, v.312, p. 78-88, 2015.

DIMURO, G. P.; BEDREGAL, B. **On the laws of contraposition for residual implications derived from overlap functions**, Los Alamitos, IEEE, p. 1-7, 2015.

DIMURO, G. P.; BEDREGAL, B.; BUSTINCE, H.; ASIAN, M. J.; MESIAR, R. **On additive generators of overlap functions**, [S.l.], Fuzzy Sets and Systems, v. 287, p. 76-96, 2016.

DIMURO, G. P.; BEDREGAL, B.; BUSTINCE, H.; MESIAR, R.; ASIAN, M. J. **On Additive Generators of Grouping Functions**, [S.l.], Information Processing and Management of Uncertainty in Knowledge-Based Systems, v. 444, n. 252-261, 2014

DIMURO, G. P.; BEDREGAL, B.; BUSTINCE, H.; JURIO, A.; BACZYNSKI, M.; MIS, K. **QL-operations and QL-implication functions constructed from tuples (O,G,N) and the generation of fuzzy subethood and entropy measures**, [S.l.], International Journal of Approximate Reasoning, v. 82, p. 170-192, 2017.

DIMURO, G. P.; BEDREGAL, B.; SANTIAGO, H. N. **On  $(G, N)$ -implications derived from grouping functions**, [S.l.], Information Sciences, v. 279, p. 1-17, 2014.

DUBOIS, D.; OSTASIEWICZ, W.; PRADE, H. **Fuzzy Sets: History and Basic Notions**, Boston, Fundamentals of Fuzzy Sets, p. 21-124, 2000.

DUBOIS, D.; PRADE, H. **Fundamentals of Fuzzy Sets**, Boston, Kluwer, 2000.

ELKANO, M.; GALARA, M.; SANZ, J. A.; SCHIAVO, P. F.; PEREIRA, S. JR.; DIMURO, G. P.; BORGES, E. N.; BUSTINCE, H. **Consensus via penalty functions for decision making in ensembles in fuzzy rule-based classification systems**, [S.l.], Applied Soft Computing, v. 67, p. 728-740, 2018.

GAGOLEWSKI, M. **Spread measures and their relation to aggregation functions**, [S.l.], European Journal of Operational Research, v. 241, n. 2, p. 469-477, 2015.

GALAR, M.; BUSTINCE, H.; BARRENECHEA, E.; FERNANDEZ, A; HERRERA, F. **An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes**, Pattern Recognition, v. 44, n. 8, p. 1761-1776, 2011.

GARCIA-JIMENEZ, S.; BUSTINCE, H.; HULLERMEIER, E.; MESIAR, R.; PAL, N. R.; PRADERA, A. **Overlap indices: Construction of and application to interpolative fuzzy systems**, [S.l.], IEEE Transactions on Fuzzy Systems, v. 23, n. 4, p. 1259-1273, 2015.

GRABISH, M.; MUROFUSHI, T.; SUGENO, M. **Fuzzy Measures and Integrals**, Heidelberg, Physica-Verlag, 2000.

HANSS, M. **Applied Fuzzy Arithmetic: An Introduction with Engineering Applications**, Berlin, Springer-Verlag, 2005.

HERRERA, F.; JESUS, D. **A proposal on reasoning methods in fuzzy rule-based classification systems**, International Journal of Approximate Reasoning, v.20, n. 1, p. 21-45, 1999.

ISHIBUCHI, H.; YAMAMOTO, T.; NAKASHIMA, T. **Hybridization of fuzzy GBML approaches for pattern classification problems**, [S.l.], IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), v. 35, n. 2, p. 359-365, 2005.

JURIO, A.; BUSTINCE, H.; PAGOLA, M.; PRADERA, A.; YAGER, R. R. **Some properties of overlap and grouping functions and their application to image thresholding**, [S.l.], Fuzzy Sets and Systems, v. 229, p. 69-90, 2013.

KANDEL, A.; PACHECO, R.; MARTINS, A. E KHATOR, S. **The Foundations of Rule-Based Computations In Fuzzy Models**. Em [Pedrycz, W. 1996, Fuzzy Modelling Paradigms and Practice.], International Series in Intelligent Technologies, v.7, Springer, Boston, 1996.

KLEMENT, E. P.; MESIAR, R.; PAP, E. **Triangular Norms**, Kluwer, Dordrecht, 2000.

KURDILA, A. J.; ZABARANKIN, M. **Convex Functional Analysis**. Berlin, Springer, 2005.

LOWEN, R. **Fuzzy set theory: basic concepts, techniques, and**, Dordrecht, Kluwer Academic Publisher, 1996.

MAYOR, G.; TRILLAS, E. **On the representation of some aggregation functions**, Los Alamitos, Proceedings of IEEE International Symposium on Multiple-Valued Logic, IEEE, 1986, pp 111-114.

Mesiar, R.; Novák, V. **Fuzzy Sets: Theory and Applications**, Mathematical Institute, Slovak Academy of Sciences, Tatra Mountains mathematical publications, issn.1210-3195, 1997.

MONTERO, J.; GOMEZ, D.; BUSTINCE, H. **On the relevance of some families of fuzzy sets**, [S.l.], Fuzzy Sets and Systems, v. 158, p. 2429-2442, 2007.

MOORE R. E. **Methods and Applications of Interval Analysis**. Philadelphia, Society for Industrial and Applied Mathematics, 1979.

RIID, A.; RUSTERN, E. **Adaptability, interpretability and rule weights in fuzzy rule-based systems**, [S.l.], Information Sciences, v. 257, p. 301-312, 2014.

SANZ, J. A.; FERNANDEZ, A.; BUSTINCE, H.; HERRERA, F. **Improving the performance of fuzzy rule-based classification systems with interval-valued fuzzy sets and genetic amplitude tuning**, [S.l.], Information Sciences, v. 180, n. 19, p. 3674-3685, 2010.

SANZ, J. A.; FERNANDEZ, A.; BUSTINCE, H.; HERRERA, F. **IVTURS: A Linguistic Fuzzy Rule-Based Classification System Based On a New Interval-Valued Fuzzy Reasoning Method With Tuning and Rule Selection**, IEEE Transactions on Fuzzy Systems, v. 21, n. 3, p. 399-411, 2013.

SILVEIRA, M.M.T. , **Teoria Fuzzy Intervalar: Uma proposta de Integração da Matemática Intervalar à Teoria Fuzzy**, Tese de Mestrado, DIMAp/UFRN, 2002.

SUGENO, M. **Theory of Fuzzy Integrals and It's Applications**, Tokyo, Tese Ph.D., 1974.

TSOUKALAS, L. H.; UHRIG, R. E. **Fuzzy and neural approaches in engineering**, Publisher John Wiley, 1997.

TURKSEN, I. B. **Type 2 representation and reasoning for computing with words**, [S.l.], Fuzzy Sets and Systems, v. 127, p. 17-36, 2002.

WANG, Z.; KLIR, G. **Fuzzy Measure Theory**, New York, Plenum Press, 1992.

WILKIN, T.; BELIAKOV, G. **Weakly monotonic averaging functions**, [S.l.], International Journal of Intelligent Systems, v. 30, n. 2, p. 144-169, 2015.

XU, Z. S.; YAGER, R. R. **Some geometric aggregation operators based on intuitionistic fuzzy sets**, [S.l.], International Journal of General Systems, v. 35, p. 417-433, 2006.

YAGER, R. R. **Toward a general theory of information aggregation**, [S.l.], Information Sciences, v.68, n. 3, p. 191-206, 1993.

YAGER, R. R.; RYBALOV, A. **Understanding the median as a fusion operator**, [S.l.], International Journal of General Systems, v. 26, n. 3, p. 239-263, 1997.

ZADEH, L. A. **Fuzzy Sets**. Information and Control, v. 8, p. 338-353, 1965.

ZADEH, L. A. **Fuzzy sets as a basis for a theory of possibility**, [S.l.], Fuzzy Sets and Systems, v. 1, n. 1, p. 3-28, 1978.