



Universidade Federal do Piauí  
Centro de Ciências da Natureza  
Programa de Pós-Graduação em Ciência da Computação

# **Modelo de Classificação *Fuzzy* baseado em Rotulação de grupos e Lógica *Fuzzy***

**Sidiney de Sousa Araujo**

**Teresina-PI, 09 de Abril de 2019**



Sidiney de Sousa Araujo

## **Modelo de Classificação *Fuzzy* baseado em Rotulação de grupos e Lógica *Fuzzy***

**Dissertação de Mestrado** apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFPI (área de concentração: Computação Aplicada), como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

Universidade Federal do Piauí – UFPI

Centro de Ciências da Natureza

Programa de Pós-Graduação em Ciência da Computação

Orientador: Prof. Dr. Vinicius Ponte Machado

Teresina-PI

09 de Abril de 2019

---

Sidiney de Sousa Araujo

Modelo de Classificação *Fuzzy* baseado em Rotulação de grupos e Lógica *Fuzzy*/  
Sidiney de Sousa Araujo. – Teresina-PI, 09 de Abril de 2019-  
42 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Vinicius Ponte Machado

Dissertação (Mestrado) – Universidade Federal do Piauí – UFPI  
Centro de Ciências da Natureza

Programa de Pós-Graduação em Ciência da Computação, 09 de Abril de 2019.

1. Aprendizagem de Máquina. 2. Classificação. 3. Fuzzy. 4. Rotulação I. Prof.  
Dr. Vinicius Ponte Machado. II. Universidade Federal do Piauí. III. Programa de  
Pós-Graduação em Ciência da Computação. IV. Modelo de Classificação *Fuzzy*  
baseado em Rotulação de grupos e Lógica *Fuzzy*.

CDU 02:141:005.7

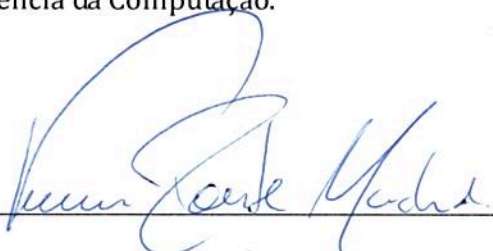
---

“Modelo de Classificação Fuzzy Baseado em Processo de Rotulação  
Automática de Grupos”

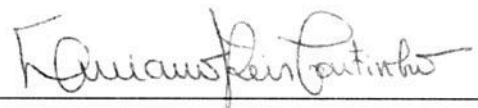
**SIDINEY DE SOUSA ARAÚJO**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Natureza da Universidade Federal do Piauí, como parte integrante dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Aprovada por:



Prof. Vinicius Ponte Machado  
(Presidente da Banca Examinadora)



Prof. Luciano Reis Coutinho  
(Examinador Externo à Instituição)



Prof. Kelson Rômulo Teixeira Aires  
(Examinador Interno)



Prof. Rodrigo de Melo Souza Veras  
(Examinador Interno)

Teresina, 09 de abril de 2019

*Aos meus pais, irmãs, padrinhos e amigos.*

# Agradecimentos

Agradeço a Deus.

Agradeço a minha mãe, Joana e ao meu pai Cantídio pelo carinho, apoio e compreensão, por estarem sempre do meu lado.

As minhas irmãs Laiane, Luana e Josélia por me apoiarem todos esses anos.

Aos meu padrinhos, Raimunda Nery e Raimundo, por me acolherem todos esses anos.

Ao meus primos Miller, Milena (in memory), pelo apoio de sempre.

Agradeço ao meu orientador, Vinícius, pela paciência, ajuda e ensinamentos durante essa jornada.

Ao meu amigo Douglas Lopes, pela carona, lances, almoços, cocas, paciência e apoio em todos esses anos.

À minha amiga Lúcia Emília por toda colaboração no trabalho, paciência e companhia nessa jornada.

Aos meus amigos, "os fi de dotô", pelas horas e horas de diversão.

Aos meus amigos que jogam comigo, mesmo longe sempre apoiando.

Aos meus amigos de graduação, pela paciência.

Aos professores Marvin Lemos, Ricardo Lira e Giovanni por todos os ensinamentos e inspirações.

Aos meus professores de graduação e mestrado, por todo conhecimento e dedicação.

Ao seu Zé por deixar o laboratório sempre limpo.

À Justina pela preocupação de me ver subir as escadas.

Aos integrantes do OASIS pelo café.

À CAPES pelo apoio financeiro para realização deste trabalho de pesquisa.





*“A vida é muito curta para ser pequena.”  
(Benjamin Disraeli)*



# Resumo

As técnicas de agrupamento e classificação de dados são frequentemente utilizadas com a finalidade de extrair padrões e classificar novos elementos, respectivamente. A combinação de tais técnicas pode ser aplicada em bases de dados em que não se conhece o atributo classe, utilizando a interpretação dos grupos obtidos no processo de agrupamento dos dados para identificação de um padrão que auxilie o processo de classificação. Esta interpretação, apesar de depender do problema abordado, requerendo por vezes o auxílio de um especialista, pode ser desempenhada por modelos de rotulação automáticos. Esses modelos são capazes de identificar características relevantes dos grupos e utilizá-las na formação de rótulos. Além da interpretação dos grupos a adição de outra técnica para classificação pode demandar mais tempo de processamento. Baseado em modelos de rotulação automáticos e lógica *Fuzzy*, este trabalho propõe um modelo de classificação no qual os rótulos dos grupos são utilizados para formação de regras e funções de pertinência de um sistema *Fuzzy*. O modelo proposto foi avaliado comparando a acurácia, desvio padrão, índice Kappa e tempo de treinamento com de outros algoritmos de classificação. Além disso, foi analisado a quantidade de regras geradas. Para diferentes bases testadas disponíveis no repositório UCI, os resultados para o índice Kappa foram acima de 0,8, com acurácia e tempo de treinamento similares aos de algoritmos encontrados na literatura.

**Palavras-chaves:** Aprendizado de Máquina. Classificação. Fuzzy. Rotulação.



# Abstract

The techniques of clustering and classification are frequently used to obtain patterns and classify new data. The combination of those techniques can be applied to problems where there is no label, using the clustering process to extract information that will assist the classification process. Usually, the clusters are analyzed by an expert to obtain that information, but this process can also be done by automatic labeling models. Those are models capable of identifying the most relevant characteristics of the clusters and use them to create a label. In addition to the interpretation of the groups, the addition of another technique to classify new elements can further increase the simulation time. Based on the automatic labeling models and the fuzzy logic, this paper proposes a model of classification that uses the clusters labels to compose rules and membership functions of a Fuzzy system. The efficiency of the proposed model was evaluated by comparing the accuracy, standard deviation, Kappa index and training time with other classification algorithms, as well as individually analyzed the amount of rules generated by the algorithms. For different databases available in the UCI repository, the results for the Kappa index were above 0.8, with accuracy and training time similar to the algorithms found in the literature.

**Keywords:** Classification. Machine Learning. Fuzzy. Labeling.



# Lista de ilustrações

Figura 1 – Arquitetura de um Sistema de Inferência <i>Fuzzy</i> . . . . .	10
Figura 2 – Modelo de rotulação proposto por Machado, Ribeiro e Rabêlo (2015). . . . .	12
Figura 3 – Fluxograma do Modelo Proposto. . . . .	19
Figura 4 – Representação das faixas para o atributo $Atr_{12}$ . . . . .	21
Figura 5 – Pertinência dos elementos no $Atr_7$ associados aos grupos. . . . .	23
Figura 6 – Pertinência dos elementos no $Atr_{12}$ associados aos grupos. . . . .	24
Figura 7 – Associação das pertinências dos elementos para o $Atr_7$ . . . . .	24
Figura 8 – Associação das pertinências dos elementos para o $Atr_{12}$ . . . . .	25
Figura 9 – Ajuste de Curva para elementos do $Atr_{12}$ variando grau do polinômio: (a) grau 2; (b) grau 3; (c) grau 4; (d) grau 5 e (e) grau 6. . . . .	25
Figura 10 – Função da partição associada ao atributo $Atr_{12}$ no Grupo 1. . . . .	26
Figura 11 – Função de pertinência do atributo $Atr_7$ . . . . .	26
Figura 12 – Função de pertinência do atributo $Atr_{12}$ . . . . .	27
Figura 13 – Cálculo do GA para atributo $Atr_7$ . . . . .	28
Figura 14 – Cálculo do GA para atributo $Atr_{12}$ . . . . .	28





# Lista de tabelas

Tabela 1 – Base de Dados Exemplo. . . . .	13
Tabela 2 – Matriz $U$ da Base de Dados Exemplo. . . . .	14
Tabela 3 – Elementos selecionados no Grupo 1. . . . .	15
Tabela 4 – Elementos selecionados no Grupo 2. . . . .	15
Tabela 5 – Elementos selecionados no Grupo 3. . . . .	15
Tabela 6 – Faixas iniciais para Base de Dados Exemplo. . . . .	16
Tabela 7 – Rótulos finais da base exemplo. . . . .	16
Tabela 8 – Rótulos para grupos da base <i>Vehicle</i> . . . . .	20
Tabela 9 – Matriz $U$ do Grupo 2 para base <i>Vehicle</i> . . . . .	22
Tabela 10 – Associação das pertinências dos elementos do Grupo 2 para o $Atr_{12}$ . . . . .	23
Tabela 11 – Resultados do modelo proposto com outros algoritmos. . . . .	33
Tabela 12 – Resultados para tempo de treinamento e quantidade de regras. . . . .	34
Tabela 13 – Resultados para base <i>Wine</i> . . . . .	34
Tabela 14 – Resultados do tempo de treinamento para base <i>Wine</i> . . . . .	34



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
<b>1.1</b>	<b>Motivação</b>	<b>1</b>
<b>1.2</b>	<b>Objetivos</b>	<b>2</b>
1.2.1	Objetivos Específicos	2
<b>1.3</b>	<b>Organização do Documento</b>	<b>3</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>5</b>
<b>2.1</b>	<b>Aprendizagem de Máquina</b>	<b>5</b>
2.1.1	Aprendizagem Supervisionada	5
2.1.2	Aprendizagem Não Supervisionada	6
<b>2.2</b>	<b>Rotulação</b>	<b>7</b>
<b>2.3</b>	<b>Sistemas Fuzzy</b>	<b>8</b>
2.3.1	Conjuntos Fuzzy	8
2.3.2	Regras Fuzzy	9
2.3.3	Sistemas de Inferência Fuzzy	10
2.3.4	Fuzzy C-means	11
<b>2.4</b>	<b>Modelo de Rotulação Automático baseado em Conjuntos Fuzzy</b>	<b>12</b>
<b>2.5</b>	<b>Trabalhos Relacionados</b>	<b>16</b>
<b>3</b>	<b>MATERIAS E MÉTODOS</b>	<b>19</b>
<b>3.1</b>	<b>Modelo Proposto</b>	<b>19</b>
3.1.1	Criação dos grupos	19
3.1.2	Rotulação dos Grupos	20
3.1.3	Geração do Classificador	21
<b>3.2</b>	<b>Bases</b>	<b>28</b>
3.2.1	Íris	29
3.2.2	Seeds	29
3.2.3	Liver Disorders - BUPA	29
3.2.4	Vehicle Silhouettes	29
3.2.5	Wine	30
<b>3.3</b>	<b>Algoritmos</b>	<b>30</b>
3.3.1	Árvore de Decisão	30
3.3.2	Algoritmo K-Vizinhos mais Próximos ( <i>K-Nearest Neighbors</i> - KNN)	30
3.3.3	Máquina de Vetor de Suporte ( <i>Support Vector Machine</i> - SVM)	31
<b>3.4</b>	<b>Métricas de Avaliação</b>	<b>31</b>

<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b> . . . . .	<b>33</b>
<b>4.1</b>	<b>Experimentos</b> . . . . .	<b>33</b>
<b>4.2</b>	<b>Avaliação dos Resultados</b> . . . . .	<b>35</b>
<b>5</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS</b> . . . . .	<b>37</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>39</b>

# 1 Introdução

Com o avanço da informatização dos sistemas, grandes bases de dados são geradas e atualizadas a cada momento. Extrair conhecimentos dessas bases é um dos grandes desafios da Aprendizagem de Máquina (AM). A AM é uma subárea da Inteligência Artificial (IA) e segundo [Mitchell et al. \(1997\)](#) pode ser descrita como o desenvolvimento de técnicas computacionais, bem como a construção de sistemas capazes de adquirir o conhecimento de forma automática.

A AM comumente divide-se em dois tipos: aprendizagem supervisionada, na qual as classes dos dados são conhecidas; e não supervisionada, na qual a classe dos dados é desconhecida ou omitida. A aprendizagem supervisionada desempenha tarefas cujo objetivo é a classificação ou regressão de novos dados, enquanto a não supervisionada tem como principal tarefa o agrupamento em que busca-se formar subconjuntos com dados que possuem características similares entre si e dissimilares com outros subconjuntos ([RUSSELL; NORVIG, 2016](#)).

Na modelagem de classificadores utilizando aprendizagem supervisionada tem-se a necessidade de uma classe dos dados. Essa classe identifica um conjunto de exemplos que são utilizados na construção do modelo para inferir a classe de novos exemplos. Em bases que não possuem esse atributo classe podem ser utilizados os modelos de rotulação automáticos que tem como objetivo encontrar rótulos para grupos, de forma a facilitar sua interpretação.

## 1.1 Motivação

Ao longo dos anos surgiram diversas técnicas utilizando AM. Na aprendizagem supervisionada por exemplo, algoritmos baseados em Redes Neurais ([SILVA; SPATTI; FLAUZINO, 2010](#)), Árvores de Decisão ([SAFAVIAN; LANDGREBE, 1991](#)) ou Regras ([NOMURA; HAYASHI; WAKAMI, 1992](#)) são aplicados em problemas de diversos domínios. Algumas técnicas buscam maximizar o acerto, como *Multilayer Perceptron* (MLP) ([SILVA; SPATTI; FLAUZINO, 2010](#)), outras procuram mostrar o conhecimento obtido de forma a facilitar a interpretação dos resultados como as Árvores de Decisão ([MILARÉ; CARVALHO; MONARD, 2002](#)).

A grande variedade de técnicas também está presente na aprendizagem não supervisionada, a exemplo os algoritmos *C-Means* ([MACQUEEN et al., 1967](#)), *Fuzzy C-Means* ([BEZDEK, 1981](#)), *CURE* ([GUHA; RASTOGI; SHIM, 1998](#)), *DBSCAN* ([ESTER et al., 1996](#)), entre outros. O objetivo principal desses algoritmos é a formação de grupos através

da identificação de similaridades e dissimilaridades dos dados.

Um problema relacionado ao agrupamento é a interpretação dos grupos formados que pode ser dificultada pela quantidade de instâncias e atributos presentes na base e também pela estrutura de saída dos algoritmos. Para resolução deste problema apresentam-se os modelos de rotulação, que buscam gerar rótulos que identifiquem de forma única cada grupo a fim de facilitar sua interpretação. Os modelos de rotulação, de modo geral, dão-se pela combinação de técnicas de AM já existentes. Por exemplo, [Lopes et al. \(2016\)](#) utiliza Redes Neurais Artificiais para encontrar as principais características do grupo, ao passo que [Machado, Ribeiro e Rabêlo \(2015\)](#) utiliza conjuntos *Fuzzy* para selecionar elementos com maiores pertinências para formar os rótulos.

Outro problema relacionado ao agrupamento é a associação de novos elementos aos grupos formados. Este cenário torna propícia a utilização de técnicas baseadas em aprendizagem supervisionada, em que os dados já agrupados são utilizados no treinamento do algoritmo. Apesar de eficiente, o uso de técnicas de classificação em agrupamentos pode não ser satisfatório uma vez que não se conhece quais similaridades formaram os grupos, retornando ao problema de interpretação. Neste sentido, os modelos de rotulação podem ser utilizados como auxílio para análise do agrupamento, fornecendo informações úteis para o treinamento de um modelo de classificação.

## 1.2 Objetivos

O objetivo desse trabalho é propor um modelo de classificação utilizando os rótulos gerados pelos modelos de rotulação automáticos e a lógica *Fuzzy*. O modelo proposto gera regras e funções de pertinência baseada nos rótulos, e a matriz de pertinência resultante do agrupamento, otimizando o tempo de execução da classificação e facilitando a interpretação dos resultados.

### 1.2.1 Objetivos Específicos

Para esse trabalho foram estabelecidos alguns objetivos específicos:

- Implementação e teste do modelo proposto para verificar sua eficiência em diferentes bases;
- Verificar a eficiência dos classificadores através da precisão obtida pelo conjunto de teste.

## 1.3 Organização do Documento

Este trabalho foi organizado com base nos principais assuntos abordados: Aprendizagem de Máquina, Modelos de Rotulação e Sistemas *Fuzzy*. O embasamento teórico dos assuntos mencionados possibilitou o desenvolvimento do modelo proposto e sua validação.

Além desta introdução, o trabalho está dividido em 5 capítulos e organizado como segue: no Capítulo 2 são descritos conceitos sobre aprendizagem de máquina como também suas subdivisões: aprendizagem supervisionada e não supervisionada. O problema da rotulação como um modelo baseado em aprendizagem supervisionada é apresentado na Seção 2.2. Também são apresentados conceitos sobre os Sistemas *Fuzzy* na Seção 2.3.

No Capítulo 3 é detalhado as etapas do modelo de classificação proposto. No Capítulo 4 explica como foram realizados os testes do modelo e resultados obtidos. E por fim, no Capítulo 5 é realizada a conclusão do trabalho e proposta para trabalhos futuros.





## 2 Referencial Teórico

Neste capítulo é apresentado o embasamento teórico para auxiliar na compreensão do trabalho proposto. Inicialmente é feita uma introdução sobre AM, mais especificamente sobre aprendizado supervisionado e não supervisionado. Em seguida é apresentado o problema da rotulação como também o modelo de rotulação que é utilizado neste trabalho. Em outra seção são apresentadas algumas definições relacionadas aos Sistemas *Fuzzy*. Ao final, são destacados alguns trabalhos relacionados ao modelo proposto.

### 2.1 Aprendizagem de Máquina

De acordo com [Mitchell et al. \(1997\)](#) a aprendizagem de máquina refere-se ao desenvolvimento de técnicas computacionais, de forma autônoma, em uma determinada tarefa por meio de experiências. Diferente das metodologias computacionais comumente utilizadas, a AM lida com o problema de modo que a própria máquina irá encontrar, após a aprendizagem, uma hipótese que melhor define o problema.

A AM relaciona-se com a manipulação do conhecimento prévio e com novas observações que possam adquirir novos conhecimentos. Conhecida como inferência indutiva ([MICHALSKI, 1983](#)), em que a forma de raciocínio parte de um conceito específico e o generaliza. A inferência indutiva divide-se em supervisionada e não supervisionada. A supervisionada tem como objetivo a criação de um modelo preditivo para novos dados a partir dos já existentes, enquanto na não supervisionada o objetivo é encontrar características que possam resumir os dados dividindo-os em grupos.

#### 2.1.1 Aprendizagem Supervisionada

Nos métodos baseados em aprendizagem supervisionada, a classe do conjunto de amostras já é conhecida. O objetivo é encontrar uma hipótese capaz de classificar novas observações entre as classes já existentes ou fazer regressões.

Geralmente os métodos para classificação dividem-se em duas etapas: a primeira etapa está relacionada ao treinamento, onde o método constrói seu modelo para classificação baseado no conjunto de dados onde se conhece a classe. Na segunda etapa, os dados que não foram utilizados no treinamento, também chamados conjunto de teste, são submetidos ao modelo para cálculo da acurácia ou outras medidas de avaliações. Em alguns métodos, à medida que aumenta o volume de dados é necessário uma atualização do modelo para extrair novos conhecimentos inseridos na base, nesse caso é repetida novamente a primeira etapa de treinamento.

Na resolução do problema de classificação incluem aplicações de redes neurais, árvores de decisão e métodos estatísticos. Métodos como *K-Nearest Neighbors* (KNN) (COVER; HART, 1967) utilizam as informações das classes no conjunto de dados atual para classificar novos exemplos, atribuindo rótulos a estes exemplos por meio de medidas de similaridade. Também surgiram variações dos métodos de classificação que realizam a indução de regras *Fuzzy* a partir de um conjunto de dados como Sistemas Neuro-*Fuzzy* e árvores de decisão *Fuzzy*.

Barber (2012) apresenta uma definição formal para o aprendizado supervisionado. Dado um conjunto de dados  $D = (x_n, y_n), n = 1, \dots, N$  contendo  $N$  elementos, o objetivo é criar um modelo preditivo capaz de aprender a relação existente entre os valores de entradas  $x$  e os valores de saída  $y$ , de modo que quando uma nova entrada  $x^*$  é fornecida a predição de sua saída  $y^*$  é correta.

### 2.1.2 Aprendizagem Não Supervisionada

Nos métodos baseados em aprendizagem não supervisionada o conjunto dos dados não possui o atributo classe. Dessa forma, no processo de aprendizado se busca detectar padrões existentes nos dados com o objetivo de representá-los de forma resumida. Assim, o problema consiste em alocar os exemplos em grupos, de tal forma que os elementos do mesmo grupo possuam alto grau de similaridade, e a dissimilaridade entre grupos diferentes seja a maior possível sendo caracterizado principalmente pela escolha da medida de similaridade (ATILGAN; NASIBOV, 2016).

Variando as características dos dados e a finalidade da aplicação são escolhidas as medidas que vão definir quão similar um exemplo é de outro. Por exemplo, em dados numéricos geralmente são utilizadas medidas baseadas em distância dentre elas destacam-se as distâncias Euclidiana e *Manhattan*.

A aprendizagem não supervisionada possui algumas desvantagens devido a sua forte dependência com as medidas de similaridade e normalização dos dados. Outra desvantagem é a necessidade da definição de alguns parâmetros como quantidade de grupos (*C-Means*, *Fuzzy C-Means*), raio para cálculo da vizinhança (*Density Based Spatial Clustering of Application with Noise*) entre outros.

Barber (2012) define a aprendizagem não supervisionada como: dado um conjunto de dados  $D = (X_n), n = 1, \dots, N$  contendo  $N$  elementos, o objetivo é determinar uma descrição plausível e compacta dos dados. De modo específico, o aprendizado não supervisionado pode ser visualizado a execução de técnicas para formação de agrupamentos, em que pretende-se atribuir classes aos elementos de uma base de dados.

## 2.2 Rotulação

As técnicas de aprendizagem não supervisionada propõem-se a formar grupos de acordo com as medidas de similaridades ou dissimilaridades dos dados (ATILGAN; NASIBOV, 2016), de modo que a interpretação dos mesmos fica à critério de um especialista. Entretanto, à medida que aumenta a quantidade de dados, maior a dificuldade de interpretação destes grupos. Neste contexto, a rotulação automática apresenta-se como uma técnica capaz de prover auxílio ao especialista. Lopes et al. (2016) definem que o problema da rotulação consiste em apresentar uma identificação resumida para os grupos, nomeando-os de acordo com suas características. A definição formal do problema de rotulação é apresentada a seguir.

**Problema de Rotulação:** Dado um conjunto de grupos  $C = \{c_1, \dots, c_K | K > 1\}$ , de modo que cada grupo contém um conjunto de elementos  $\{c_i = \vec{e}_1, \dots, \vec{e}_{n^{(c_i)}} | n^{(c_i)} \geq 1\}$  que podem ser representados por um vetor de atributos definidos em  $\mathbb{R}^m$  e expresso por  $\vec{e}_j^{(c_i)} = (a_1, \dots, a_m)$ , e que  $c_i \cap c_{i'} = \{\emptyset\}$  com  $1 \leq i, i' \leq K$  e  $i \neq i'$ ; o objetivo consiste em apresentar um conjunto de rótulos  $R = \{r_{c_1}, \dots, r_{c_K}\}$  no qual cada rótulo específico é dado por um conjunto de pares de valores: atributo e seu respectivo intervalo,  $r_{c_i} = \{(a_1, [p_1, q_1]), \dots, (a_{m^{(c_i)}}, [p_{m^{(c_i)}}, q_{m^{(c_i)}}])\}$  tal que  $r_{c_i} - r_{c_{i'}} \neq \emptyset$  e  $r_{c_i} \not\supseteq r_{c_{i'}}$ , seja capaz de melhor expressar o grupo  $c_i$  associado.

A fim de esclarecimento, tem-se que:

- $K$  é o número de grupos;
- $c_i$  é um grupo qualquer;
- $n^{(c_i)}$  é o número de elementos do grupo  $c_i$ ;
- $\vec{e}_j^{(c_i)}$  se refere ao  $j$ -ésimo elemento pertencente ao grupo  $c_i$ ;
- $m$  é a dimensão do problema;
- $[p_{m^{(c_i)}}, q_{m^{(c_i)}}]$  representa o intervalo de valores  $a_{m^{(c_i)}}$  onde  $p_{m^{(c_i)}}$  é o limite inferior e  $q_{m^{(c_i)}}$  é o limite superior;
- $m^{(c_i)}$  é a quantidade de atributos presente em um rótulo referente ao grupo  $c_i$ .

Em outras palavras a rotulação busca auxiliar na interpretação dos grupos, em que as características dos grupos são apresentadas na forma atributo e faixa de valores. Por exemplo, suponha que tenha um grupo de pessoas e esse grupo possui alguns atributos (Nome, Idade, Sexo, Curso). Rotular esse grupo seria informar que o grupo é formado por pessoas de Idade entre 20 e 35 anos. Apesar de não saber como o grupo foi construído o rótulo apresenta uma característica relevante para o grupo avaliado. Os modelos de rotulação buscam formar esses rótulos de forma automatizada facilitando assim o trabalho com os grupos.

Diferentes modelos para resolução do problema da rotulação são propostos na lite-

ratura. Por exemplo, Lopes et al. (2016) utilizam Redes Neurais Artificiais para selecionar atributos relevantes e encontrar as respectivas faixas de valores que os representam no grupo. Já Machado, Ribeiro e Rabêlo (2015) utilizam a matriz de pertinência resultante do algoritmo *Fuzzy C-Means* para selecionar os elementos cujos intervalos de valores irão compor o rótulo.

## 2.3 Sistemas *Fuzzy*

Sistemas *Fuzzy* são sistemas baseados nas teorias de conjuntos *Fuzzy* e lógica *Fuzzy*, para construção de um raciocínio aproximado, permitindo a expressão da ambiguidade e subjetividade presentes no raciocínio humano (PEDRYCZ; GOMIDE, 1998).

### 2.3.1 Conjuntos *Fuzzy*

A teoria dos conjuntos *Fuzzy* é considerada uma extensão da teoria clássica de conjuntos (ZIMMERMANN, 1996), na qual um elemento pode apenas pertencer ou não a um conjunto ou classe. Este tipo de conjunto é denominado conjunto *crisp*. Contudo, em muitas situações do mundo real a teoria clássica pode ser insuficiente para representar os elementos, uma vez que um elemento pode pertencer a um conjunto de acordo com certo grau de pertinência, denominando-os conjuntos *Fuzzy*.

O cálculo da pertinência de um elemento em um conjunto *Fuzzy* é feito através das funções de pertinência, representadas por qualquer função que produza valores no intervalo  $[0, 1]$ , em que 0 representa nenhuma pertinência e 1, total pertinência ao conjunto (ZADEH et al., 1965). Esta abordagem permite o estabelecimento de valores parciais de verdade, ou seja, valores entre o completamente verdadeiro e completamente falso, o que constitui a lógica *Fuzzy*.

As funções mais utilizadas são apresentadas no formato linear (triangular, trapezoidal), quadrática ou gaussiana. Em geral as funções lineares são mais populares devido sua simplicidade e o fato do custo computacional requeridos pelas outras funções não afetar, em geral, na melhoria significativa dos valores de saída dos Sistemas *Fuzzy* (YEN; LANGARI, 1999).

As operações definidas na teoria clássica dos conjuntos também são estendidas nos conjuntos fuzzy. Segundo Thole, Zimmermann e Zysno (1979) as operações básicas com conjuntos fuzzy são realizadas sobre suas respectivas funções de pertinência, com a utilização de operadores máximo e mínimo. A seguir são apresentados alguns operadores em que  $X$  é o domínio.

**Complemento** - O grau de pertinência  $\mu_{\bar{A}}(x)$  de um conjunto fuzzy normalizado

$A$  é definido pela subtração de  $\mu_A(x)$ , como mostra a Equação 2.1.

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x), \quad x \in X. \quad (2.1)$$

**Interseção** - O grau de pertinência  $\mu_C(x)$  de uma interseção  $C = A \cap B$  é definido pelos valores mínimos entre  $\mu_A(x)$  e  $\mu_B(x)$ , conforme Equação 2.2.

$$\mu_C(x) = \min\{\mu_A, \mu_B\}, \quad x \in X. \quad (2.2)$$

**União** - O grau de pertinência  $\mu_D(x)$  de uma união  $D = A \cup B$  é definido pelos valores máximos entre  $\mu_A(x)$  e  $\mu_B(x)$ , conforme equação 2.3. Em casos de regras do tipo *or* essa operação é utilizada.

$$\mu_D(x) = \max\{\mu_A, \mu_B\}, \quad x \in X. \quad (2.3)$$

Outra definição utilizada nos conjuntos *Fuzzy* refere-se às variáveis linguísticas, que segundo Zimmermann (1996) são aquelas cujos valores são palavras ou sentenças em linguagem natural em vez de números. Uma vez que os seres humanos costumam representar o conhecimento de forma linguística, o que torna complexa a interpretação pelos computadores, as variáveis linguísticas representam matematicamente tais conhecimentos, tornando-os computacionalmente manipuláveis. Por exemplo, a variável linguística temperatura pode ter seu domínio granularizado nos valores linguísticos *baixa*, *média* e *alta*.

### 2.3.2 Regras Fuzzy

Os pares variáveis linguísticas e valor linguístico são utilizados como termos para modelagem dos sistemas *Fuzzy* através de regras *Fuzzy* da forma:

**Se antecedente então consequente.**

As regras são comumente criadas pelos especialistas, os quais possuem conhecimento do problema estudado. Tal representação permite ao sistema *Fuzzy* maior transparência e compreensibilidade.

As regras *Fuzzy* utilizam as variáveis de entrada do Sistema *Fuzzy* como condições e as variáveis de saída como consequência. Assim, sendo A e B variáveis linguísticas de entrada, definidas respectivamente em seu conjunto de termos  $\{A_1, A_2, \dots, A_n\}$  e  $\{B_1, B_2, \dots, B_n\}$ , e C uma variável linguística de saída definida no conjunto de termos  $\{C_1, C_2, \dots, C_n\}$ . De forma geral, a regra *Fuzzy* terá o seguinte formato:

Se  $A$  é  $A_i$  e  $B$  é  $B_j$  então  $C$  é  $C_k$

No modelo proposto de classificação as variáveis  $A$  e  $B$  serão representadas pelos atributos da base, e  $C$  representa o grupo. Os termos  $A_i$  e  $B_j$  representam os valores da classe e  $C_k$  o grupo associado.

### 2.3.3 Sistemas de Inferência Fuzzy

Segundo [Jamshidi, Vadiie e Ross \(1993\)](#) os Sistemas *Fuzzy* possuem três etapas: *Fuzzificação*, Processo de Inferência e *Defuzzificação*. A Figura 1 é ilustrada a arquitetura básica de um Sistema de Inferência *Fuzzy*.

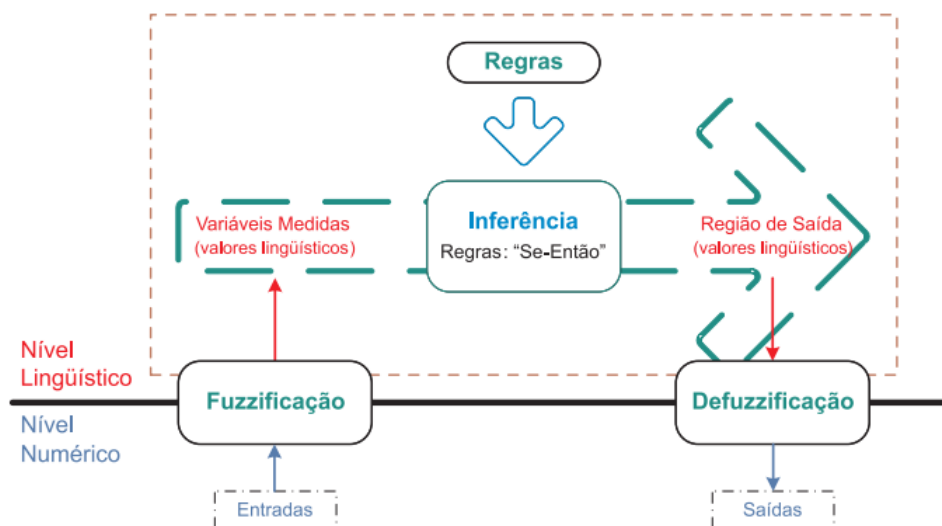


Figura 1 – Arquitetura de um Sistema de Inferência *Fuzzy*.

**Fuzzificação** - realizado o mapeamento do domínio de entrada dos dados numéricos para o domínio lingüístico. Por exemplo, um valor de temperatura igual à 35 (numérica) é mapeada como quente (lingüístico).

**Inferência** - avalia o conjunto de regras que descrevem a relação entre as variáveis de entrada e saída, ou antecedente e consequente. Por exemplo, a variável temperatura nessa etapa são executadas as regras presente na base como: Se temperatura é quente então chuva é sim.

**Defuzzificação** - transformação do valor *Fuzzy* para uma saída numérica. Na literatura estão presentes algumas técnicas de defuzzificação como: centróide, *First-of-Maxima*, *Middle-of-Maxima* e critério de máximo. Retornando ao exemplo da temperatura, nessa etapa o valor de saída resultante da inferência, que no caso é sim para chuva, é transformado em numérico.

### 2.3.4 Fuzzy C-means

Conjuntos *Fuzzy* podem ser formados com uma base de dados numérica através de seus agrupamentos. Os grupos de dados dão origem às funções de pertinência que transmitem uma visão mais abstrata dos dados globais disponíveis. Neste contexto, o *Fuzzy C-Means* (FCM) é dos mecanismos comumente utilizados.

Dado um conjunto de dados  $n$ -dimensional  $\{\mathbf{x}_k, k = 1, 2, \dots, N\}$ , a tarefa é determinar sua estrutura, uma coleção de  $c$  grupos, que é expressado como uma minimização da seguinte função objetivo  $\mathcal{Q}$  sendo considerada como a soma das distâncias quadradas:

$$\mathcal{Q} = \sum_{i=1}^c \sum_{k=1}^N u_{ik}^m \|\mathbf{x}_k - \mathbf{v}_i\|^2, \quad (2.4)$$

onde  $\mathbf{v}_i$  são os  $n$ -dimensional protótipos dos grupos;  $i = 1, 2, \dots, c$  e  $U = [u_{ik}]$  representa uma matriz de partição que expressa uma forma de alocação dos dados aos grupos correspondentes; e  $u_{ik}$  é o grau de pertinência do dado  $\mathbf{x}_k$  no  $i$ -enésimo grupo. A distância entre o dado  $n_k$  ao protótipo  $\mathbf{v}_i$  é denotado por  $\|\cdot\|$ . O parâmetro  $m (>1,0)$ , chamado de *fuzzifier* ou expoente de ponderação, expressa o impacto do grau de pertinência nos grupos individuais. Para caso de  $m = 1$ , as atribuições do grupo permanecem semelhantes ao *Hard C-Means*(DUNN, 1973). Valores mais elevados para o  $m$  tornam os limites entre os grupos mais suaves, normalmente é escolhido  $m = 2$ .

A matriz de partição satisfaz duas propriedades importantes:

$$0 < \sum_{k=1}^N u_{ik} < N, \quad i = 1, 2, \dots, c \quad (2.5)$$

$$\sum_{i=1}^c u_{ik} = 1, \quad k = 1, 2, \dots, N. \quad (2.6)$$

As restrições dadas nas Equações (2.5) e (2.6) têm de ser satisfeitas para graus de pertinência em  $U$ . A restrição (2.5) afirma que cada grupo não pode ser vazio e diferente do conjunto total. A segunda restrição (2.6) estabelece que a soma dos graus de pertinência deve ser limitada a 1.

A função objetivo  $\mathcal{Q}$  é otimizada por meio das seguintes equações, e os graus de pertinência são otimizados com os protótipos definidos. Em seguida, os protótipos são otimizados por meio dos graus de pertinência, dadas pelas Equações (2.7) e (2.8).

$$u_{ik} = \frac{1}{\sum_{l=1}^c \left( \frac{d_{ik}^2}{d_{lk}^2} \right)^{\frac{1}{m-1}}} \quad (2.7)$$

$$\mathbf{v}_i = \frac{\sum_{k=1}^N u_{ik}^m \mathbf{x}_k}{\sum_{k=1}^N u_{ik}^m} \quad (2.8)$$

onde  $d_{ik}^2 = \|\mathbf{x}_k - \mathbf{v}_i\|^2$ .

De acordo com [Bezdek \(2013\)](#) os graus de pertinência têm de ser escolhidos de acordo com a Equação (2.7), que é independente da medida de distância escolhida. Caso exista um grupo  $i$  com distância zero de um dado  $x_k$ ,  $u_{ik} = 1$  e  $u_{lk} = 0$  para todos os grupos  $l \neq i$ . A Equação (2.7) mostra o caráter relativo do grau de pertinência, no qual não depende apenas da distância do dado  $x_k$  para o grupo  $i$ , mas também das distâncias entre este dado e os outros grupos.

A Equação (2.8) apresenta a fórmula para calcular os centros dos grupos, em que os graus de pertinências calculados são utilizados na otimização dos protótipos ([BEZDEK, 2013](#)).

## 2.4 Modelo de Rotulação Automático baseado em Conjuntos Fuzzy

O modelo proposto por [Machado, Ribeiro e Rabêlo \(2015\)](#) para resolução do problema de rotulação segue o fluxograma da Figura 2.



Figura 2 – Modelo de rotulação proposto por [Machado, Ribeiro e Rabêlo \(2015\)](#).

Uma base de dados exemplo apresentada na Tabela 1 é utilizada para demonstrar



o funcionamento do modelo. A base é composta por 30 elementos, representados por um identificador e dois atributos:  $Atr_1$  e  $Atr_2$ .

Tabela 1 – Base de Dados Exemplo.

#	$Atr_1$	$Atr_2$	#	$Atr_1$	$Atr_2$
1	4,3	6,0	16	8,9	4,0
2	9,7	6,5	17	7,8	7,7
3	4,7	5,7	18	7,8	4,6
4	7,0	8,0	19	3,9	3,6
5	4,3	4,7	20	8,5	5,1
6	4,9	4,8	21	7,6	6,9
7	3,1	5,7	22	5,9	8,5
8	4,1	5,5	23	9,1	6,4
9	5,8	7,5	24	7,0	7,3
10	9,4	6,0	25	6,3	7,4
11	9,8	3,5	26	5,3	3,8
12	8,0	4,0	27	7,5	8,2
13	9,0	6,7	28	5,9	6,9
14	4,5	3,7	29	3,8	4,6
15	8,8	4,5	30	6,8	7,9

O modelo é iniciado com o carregamento da base de dados, a definição dos parâmetros GS (Grau de Seleção) e IGS (Incremento do Grau de Seleção). O parâmetro GS consiste em um número que é utilizado para seleção dos elementos mais significativos na formação dos rótulos. O IGS consiste em um valor de incremento do parâmetro GS a cada interação. Após definição dos parâmetros é aplicado o algoritmo de agrupamento não supervisionado *Fuzzy C-Means* que fornece como saída a matriz  $U$ . Esta matriz atribui a cada elemento um grau de pertinência em cada um dos grupos formados. O grau de pertinência é atribuído de tal forma que, quanto mais próximo o elemento estiver do centróide de um grupo, maior é seu grau de pertinência em relação ao grupo. Executando o *Fuzzy C-Means* com a quantidade de grupos igual à 3 na base exemplo obteve-se a matriz  $U$  apresentada na Tabela 2.

Com a formação da matriz  $U$ , o modelo de rotulação seleciona os elementos que possuem um grau de pertinência maior que o parâmetro GS. Com isto, em cada grupo selecionado são extraídos os valores máximo e mínimo de cada atributo. Esses valores correspondem as faixas de valores de cada grupo. Na base exemplo os elementos inicialmente escolhidos em cada grupo considerando um GS igual à 0,5 são apresentados nas Tabelas 3, 4 e 5. A Tabela 6 apresenta as faixas iniciais, tendo por base os elementos selecionados.

Ao final, é verificado se existem interseções entre as faixas de valores pertencentes a um mesmo atributo. Caso exista interseção estas faixas de valores não são consideradas nos rótulos e a análise parte para outro conjunto de faixas. Isto é necessário pois faixas de valores que compõem interseção são ambíguas e incapazes de representar um único grupo.

Tabela 2 – Matriz U da Base de Dados Exemplo.

Id	Grau de Pertinência		
	Grupo 1	Grupo 2	Grupo 3
1	0,1269	0,0535	0,8196
2	0,2383	0,6920	0,0697
3	0,0973	0,0458	0,8569
4	0,9758	0,0140	0,0102
5	0,0009	0,0007	0,9984
6	0,0286	0,0217	0,9497
7	0,1062	0,0565	0,8373
8	0,0401	0,0213	0,9386
9	0,8411	0,0598	0,0991
10	0,1329	0,8230	0,0441
11	0,0983	0,8230	0,0787
12	0,0890	0,8220	0,0890
13	0,3388	0,5891	0,0721
14	0,0549	0,0565	0,8886
15	0,0142	0,9764	0,0094
16	0,0476	0,9148	0,0376
17	0,8693	0,0920	0,0387
18	0,0954	0,8273	0,0773
19	0,0589	0,0561	0,8850
20	0,0135	0,9797	0,0068
21	0,7885	0,1567	0,0548
22	0,8425	0,0681	0,0894
23	0,2381	0,7011	0,0608
24	0,9791	0,0126	0,0083
25	0,9436	0,0261	0,0303
26	0,0944	0,1165	0,7891
27	0,9095	0,0576	0,0329
28	0,7602	0,0859	0,1539
29	0,0162	0,0117	0,9721
30	0,9871	0,0071	0,0058

Tabela 3 – Elementos selecionados no Grupo 1.

Id	Atr <sub>1</sub>	Atr <sub>2</sub>	Grau de Pertinência		
			Grupo 1	Grupo 2	Grupo 3
4	7,0	8,0	0,9758	0,0140	0,0102
9	5,8	7,5	0,8411	0,0598	0,0991
17	7,8	7,7	0,8693	0,0920	0,0387
21	7,6	6,9	0,7884	0,1567	0,0548
22	5,9	8,5	0,8425	0,0681	0,0894
24	7,0	7,3	0,9792	0,0126	0,0083
25	6,3	7,4	0,9436	0,0261	0,0303
27	7,5	8,2	0,9095	0,0576	0,0329
28	5,9	6,9	0,7602	0,0859	0,1539
30	6,8	7,9	0,9872	0,0071	0,0058

Tabela 4 – Elementos selecionados no Grupo 2.

Id	Atr <sub>1</sub>	Atr <sub>2</sub>	Grau de Pertinência		
			Grupo 1	Grupo 2	Grupo 3
2	9,7	6,5	0,2383	0,6920	0,0697
10	9,4	6,0	0,1329	0,8229	0,0441
11	9,8	3,5	0,0983	0,8229	0,0787
12	8,0	4,0	0,0890	0,8220	0,0890
13	9,0	6,7	0,3388	0,5891	0,0721
15	8,8	4,5	0,0142	0,9764	0,0094
16	8,9	4,0	0,0476	0,9148	0,0376
18	7,8	4,6	0,0954	0,8273	0,0773
20	8,5	5,1	0,0135	0,9797	0,0068
23	9,1	6,4	0,2381	0,7011	0,0608

Tabela 5 – Elementos selecionados no Grupo 3.

Id	Atr <sub>1</sub>	Atr <sub>2</sub>	Grau de Pertinência		
			Grupo 1	Grupo 2	Grupo 3
1	4,3	6,0	0,1269	0,0535	0,8196
3	4,7	5,7	0,0973	0,0458	0,8569
5	4,3	4,7	0,0009	0,0007	0,9984
6	4,9	4,8	0,0286	0,0217	0,9498
7	3,1	5,7	0,1062	0,0565	0,8372
8	4,1	5,5	0,0401	0,0213	0,9386
14	4,5	3,7	0,0549	0,0565	0,8886
19	3,9	3,6	0,0589	0,0561	0,8850
26	5,3	3,8	0,0944	0,1165	0,7891
29	3,8	4,6	0,0162	0,0117	0,9721

Tabela 6 – Faixas iniciais para Base de Dados Exemplo.

	<b>Grupo 1</b>	<b>Grupo 2</b>	<b>Grupo 3</b>
<i>Atr</i> <sub>1</sub>	<b>5,8 ~ 7,8</b>	<b>7,8 ~ 9,8</b>	3,1 ~ 5,3
<i>Atr</i> <sub>2</sub>	6,9 ~ 8,5	<b>3,5 ~ 6,7</b>	<b>3,6 ~ 6,0</b>

Para a base exemplo observando a Tabela 6 as faixas que possuem interseções estão em negrito.

Caso nenhum atributo possua pelo menos uma faixa de valor capaz de representar um dos grupos, o parâmetro GS é incrementado pelo parâmetro IGS e o processo de seleção de elementos é refeito utilizando o novo valor de GS gerando novas faixas. No qual a condição de parada é a existência de pelo menos uma faixa de valor sem interseção em cada grupo. Considerando IGS igual à 0,0001 após 319 iterações o modelo obteve faixas sem interseções para cada grupo na base exemplo. Os rótulos são exibidos na Tabela 7 que mostra as faixas de valores correspondentes a seu grupo e atributo.

Tabela 7 – Rótulos finais da base exemplo.

	<b>Grupo 1</b>	<b>Grupo 2</b>	<b>Grupo 3</b>
<i>Atr</i> <sub>1</sub>	5,8 ~ 7,8	8,5 ~ 8,9	3,1 ~ 4,9
<i>Atr</i> <sub>2</sub>	7,3 ~ 8,5	-	-

## 2.5 Trabalhos Relacionados

Nos últimos anos diversos trabalhos foram publicados apresentando abordagens que combinam técnicas baseadas em aprendizagem não supervisionada e supervisionada. Outro tópico comumente relacionado a estas combinações é a interpretação dos grupos. A seguir são apresentados alguns trabalhos relacionados que mostram a combinação de técnicas baseadas em aprendizagem de máquina, geração de regras utilizando algoritmos presentes na literatura e também são apresentados outros modelos de rotulação.

Em Wang e Mendel (1992) é desenvolvido um método para gerar regras *Fuzzy* de dados numéricos. O método proposto pode ser usado de maneira geral para combinar dados numéricos e informações linguísticas dentro de uma framework em comum. Os dados de entradas e saídas são divididos em regiões *Fuzzy* e formadas as regras. As regras geradas dessa maneira são *and* e do tipo *IF – THEN*. Entretanto neste trabalho é necessário conhecer as saídas para construção das partições *Fuzzy*.

Lopes e Camargo (2012) apresentam uma abordagem para gerar regras *Fuzzy* utilizando agrupamento semissupervisionado. O algoritmo utilizado foi o *Fuzzy C-Means* e um mecanismo de *boosting* (FREUND; SCHAPIRE, 1997) para realizar a rotulação dos dados. A utilização de algoritmos semissupervisionados mesmo que em menor quantidade

ainda necessita de alguns dados já rotulados. A seguir são apresentados trabalhos que apresentam modelos de rotulação.

Diversos trabalhos apresentam modelos híbridos, como em [Esmin e Lambert-Torres \(2010\)](#), em que o modelo e as partições das funções *Fuzzy* são otimizadas utilizando *Particle Swarm Optimization Algorithm* (PSO). [Shobha e Anandhi \(2017\)](#) apresenta uma variante de algoritmos genéticos para descobrimento de regras de classificação.

[Otero, Freitas e Johnson \(2013\)](#) propõem uma estratégia para descobrimento de regras de classificação utilizando algoritmos de colônia de formigas. Onde a principal motivação do trabalho é evitar que regras que forem descobertas afetem no descobrimento de novas regras modificando o espaço de busca. No modelo proposto as regras são formadas separadamente para cada grupo não interferindo em outras regras.

No trabalho de [Elyasigomari et al. \(2015\)](#) é proposta uma abordagem para classificação de tipos de câncer baseada em agrupamentos e otimização. Neste trabalho é mostrado que a classificação de tipos de câncer requer análise de um número extensivo de genes, uma tarefa extremamente complexa se realizada manualmente. A solução apresentada é a seleção de genes realizada por um algoritmo de agrupamento, utilizando uma junção do algoritmo de otimização Cuckoo (COA) e algoritmo genético. Após o agrupamento são selecionados os 20 melhores genes de cada grupo para serem utilizados pelas técnicas de classificação MLP e *Support Vector Machine* (SVM).

No trabalho apresentado por [Costa et al. \(2016\)](#) é proposta uma abordagem para classificação não supervisionada de dados *streams* utilizando o conceito de *Typicality and Eccentricity Data Analytics* (TEDA). São formadas regras utilizando o conceito de sistemas *Fuzzy AnYa*, onde o antecedente da regra relaciona as informações analisadas a um *data cloud*. A parte consequente da regra pode ser da mesma forma que *Mamdani* ou *Takagi-Sugeno*.

[Biscarri et al. \(2017\)](#) apresentam um *framework* que utiliza agrupamentos para auxiliar na classificação automática do consumo de energia. Neste trabalho é realizado uma seleção dos dados similares dos consumidores com base no seus consumos e outros critérios de economia, submetendo-os à diferentes algoritmos de agrupamento, como algoritmos de agrupamento hierárquico, *C-means*, Diana, PAM, Clara, Fanny, SOM, algoritmos baseados em modelos e SOTA. A validação do agrupamento é feita pelo especialista da companhia de energia e então submetida para classificação.

[Gan et al. \(2018\)](#) propõem uma forma de incrementar a performance de classificação para os métodos baseados em aprendizagem de grafos. O trabalho introduz o algoritmo chamando Classificação de Quadrados Mínimos Regularizados baseado em Agrupamentos Supervisionados (SuperRLSC). O modelo inicialmente aplica um algoritmo *C-means* supervisionado para descobrir estruturas presentes nos dados. Após esta descoberta,

as estruturas são utilizadas para criação dos grafos e, por fim, aplica-se um termo de regularização baseado em grafos (RLSC) para classificação. Os resultados obtidos são comparados com outros algoritmos de classificação presentes no estado da arte como RLSC, KMSE, SVM, MR, DRGV e DRLSC.

Em [Yang e Quyen \(2018\)](#) é apresentado um *framework* para análise de dados em bases em que não há atributo classe. Para isso, é realizada a seleção de atributos de interesse da base, compondo uma base auxiliar  $\mathbf{Q}$ . Os demais atributos compõem uma segunda base auxiliar  $\mathbf{X}$ . A partir de então são aplicadas técnicas de agrupamento na base  $\mathbf{Q}$ , cujos resultados obtidos são utilizados pelas técnicas de classificação na base  $\mathbf{X}$ . Entretanto, são necessários ajustes de parâmetros para as técnicas utilizadas visando o balanço entre a otimização do agrupamento e classificação. O ajuste é feito aplicando o algoritmo de otimização NSGAI-SCC.

Além do modelo apresentado por [Machado, Ribeiro e Rabêlo \(2015\)](#) que utiliza conjuntos *Fuzzy* para formação dos rótulos existem outros modelos de rotulação automáticos como proposto por [Lopes et al. \(2016\)](#), no qual os rótulos são formados utilizando algoritmos baseados em aprendizagem supervisionada. O trabalho proposto por [Imperes Filho \(2018\)](#) utiliza algoritmos de agrupamento baseados em distância no processo de rotulação. Após agrupar, é calculada a pertinência de cada elemento para todos os grupos e em seguida selecionados utilizando um parâmetro de seleção. São gerados os rótulos utilizando os elementos selecionados e verifica-se possui interseções entre os intervalos do mesmo atributo. Caso exista, o parâmetro de seleção é incrementado e selecionados novos elementos. Caso não possua interseções não são realizadas mais seleções e os rótulos finais são gerados.

[Araújo et al. \(2018\)](#) utiliza o algoritmo de agrupamento DAMICORE para auxiliar a rotulação de grupos. Neste trabalho os testes foram realizados utilizando o modelo de rotulação proposto por [Lopes et al. \(2016\)](#), no qual na etapa de agrupamento utilizou-se o DAMICORE mostrando uma maior acurácia em comparação ao *C-Means*.

Também existem trabalhos relacionados à rotulação de grupos hierárquicos ([GLOVER et al., 2002](#)), ([CHUANG; CHIEN, 2004](#)), ([POPESCU; UNGAR, 2000](#)), ([MAQBOOL; BABRI, 2005](#)), entretanto se caracterizam por trabalhar exclusivamente com informações textuais.

As abordagens citadas apresentam estruturas similares, consistindo, de modo geral, em: pré-processamento, agrupamento, análise dos grupos e classificação. Dentre as etapas destacamos a interpretação dos grupos, ora desempenhada por modelos matemáticos, ora por especialistas, dependendo de recursos e tempo em ambos os casos. Neste sentido, no modelo proposto neste trabalho a interpretação dos grupos é desempenhada de forma direta na geração do classificador.

## 3 Materias e Métodos

Este capítulo descreve o modelo proposto neste trabalho, as bases, os algoritmos e métricas de avaliação utilizadas.

### 3.1 Modelo Proposto

O objetivo é mostrar um modelo de classificação que utiliza os rótulos para formar regras e funções de pertinência *Fuzzy*. A Figura 3 mostra o fluxograma do funcionamento do modelo que está dividido em três partes: agrupamento (I), rotulação dos grupos (II) e geração do classificador (III).

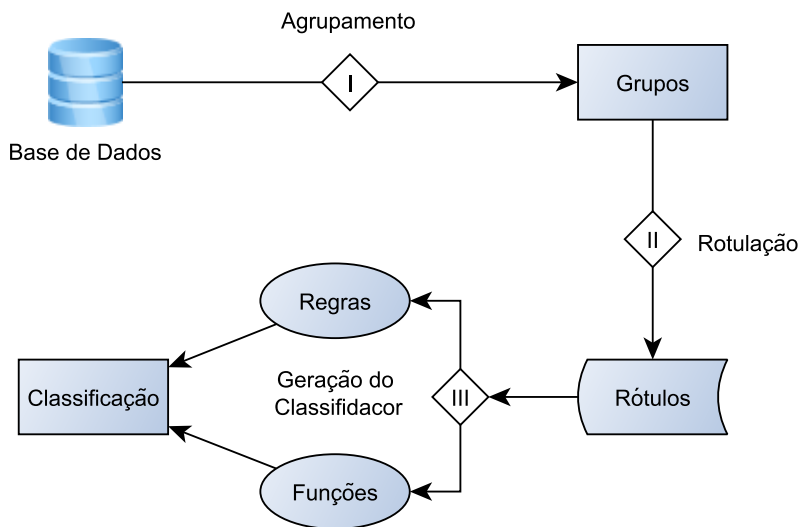


Figura 3 – Fluxograma do Modelo Proposto.

#### 3.1.1 Criação dos grupos

Na formação dos grupos é aplicado o algoritmo *Fuzzy C-means* e se faz necessário a escolha dos parâmetros  $k$  (quantidade de grupos) e  $m$  (fator de *fuzzificação*). Como discutido na Seção 2.3, existe um problema referente a escolha dos melhores valores para os parâmetros, entretando a execução do modelo parte dos rótulos já formados por isso a

definição dos parâmetros deve ser realizada antes da execução do modelo, po isso não é considerada nesse trabalho.

A escolha do algoritmo *Fuzzy C-Means* se dá pelo modelo de rotulação utilizado e pela matriz de pertinência (Matriz  $U$ ) resultante do agrupamento utilizada na geração das funções de pertinência do sistema proposto.

### 3.1.2 Rotulação dos Grupos

Nesta etapa, o modelo de rotulação é aplicado nos grupos criados, no qual o objetivo é encontrar intervalos de valores e seus respectivos atributos que representam cada grupo. Como definido por [Lopes et al. \(2016\)](#), os intervalos obtidos não deve ter interseções, de modo que cada rótulo criado seja único. Para o modelo de classificação proposto é utilizado o Modelo de Rotulação Automática ([MACHADO; RIBEIRO; RABÊLO, 2015](#)) o qual utiliza o *Fuzzy C-Means* como algoritmo de agrupamento.

Como apresentado na Seção 2.4 após o agrupamento são selecionados elementos com pertinência maior que o GS e verifica-se a existência de interseções das faixas calculadas para os atributos. Caso não exista nenhuma faixa que represente os grupos, o valor GS é incrementado e é realizado novamente o cálculo das faixas tendo como critério de parada pelo menos uma faixa identificando cada grupo.

Para exemplificar o processo de rotulação utilizou-se a base de dados *Vehicle Silhouettes Data Set*<sup>1</sup> disponível no repositório do *UCI Machine Learning*. Esta base é composta por 846 instâncias e 18 atributos. Também é informado que a base pode ser dividida em 4 classes.

Inicialmente é aplicado o algoritmo *Fuzzy C-Means* com os parâmetros  $k$  igual à 4 e  $m$  igual à 2. Após a etapa de agrupamento são atribuídos os valores para os parâmetros GS e IGS, sendo 0,5 e 0,01. Os rótulos para os grupos são apresentados na Tabela 8.

Tabela 8 – Rótulos para grupos da base *Vehicle*.

	<b>Grupo 1</b>	<b>Grupo 2</b>	<b>Grupo 3</b>	<b>Grupo 4</b>
$Atr_7$	[174 ~ 203]	-	-	[204 ~ 265]
$Atr_{12}$	[458 ~ 624]	[191 ~ 304]	[305 ~ 428]	[629 ~ 1018]

Ao observar a Tabela 8 nota-se a presença do  $Atr_{12}$  em todos os grupos sendo nos grupos 2 e 3 representados por somente este, o que demonstra sua importância para a interpretação dos grupos.

<sup>1</sup> <https://archive.ics.uci.edu/ml/datasets/Statlog+%28Vehicle+Silhouettes%29>



### 3.1.3 Geração do Classificador

Antes de iniciarmos a descrição de como são formadas as funções e regras, vamos fazer uma discursão sobre os rótulos e como podem ser utilizados na classificação. Para esta discursão serão utilizados os rótulos da base *Vehicle*, apresentados na Tabela 8.

Uma interpretação que pode ser utilizada se observar o rótulo de um grupo é que se um elemento tiver o atributo e a faixa de valores do rótulo então este elemento pertence ao grupo, visto que não é possível pertencer a dois ou mais rótulos devido a sua unicidade. Por exemplo, se o valor de um elemento para o  $Atr_{12}$  é igual à 200 pode-se supor que ele pertença ao Grupo 2 representado pelo rótulo  $\{(Atr_{12}, [191 \sim 304])\}$ . A interpretação utilizada nesse exemplo assemelha-se a teoria clássica dos conjuntos, no qual o elemento pertence ou não a um rótulo e conseqüentemente a um grupo. Uma forma de representar esta interpretação é que se o valor de um atributo está contido no rótulo sua pertinência para aquela faixa é igual a 1. A Figura 4 apresenta essa representação para o atributo  $Atr_{12}$ . Para padronizar, as faixas são nomeadas como  $faixa_{m,i}$ , no qual é  $m$  o atributo e  $i$  o grupo que a faixa está associada. Por exemplo, a  $faixa_{12,1}$  presente na Figura 4 é referente a faixa  $[305 \sim 428]$  do atributo  $Atr_{12}$  associada ao rótulo do Grupo 1.

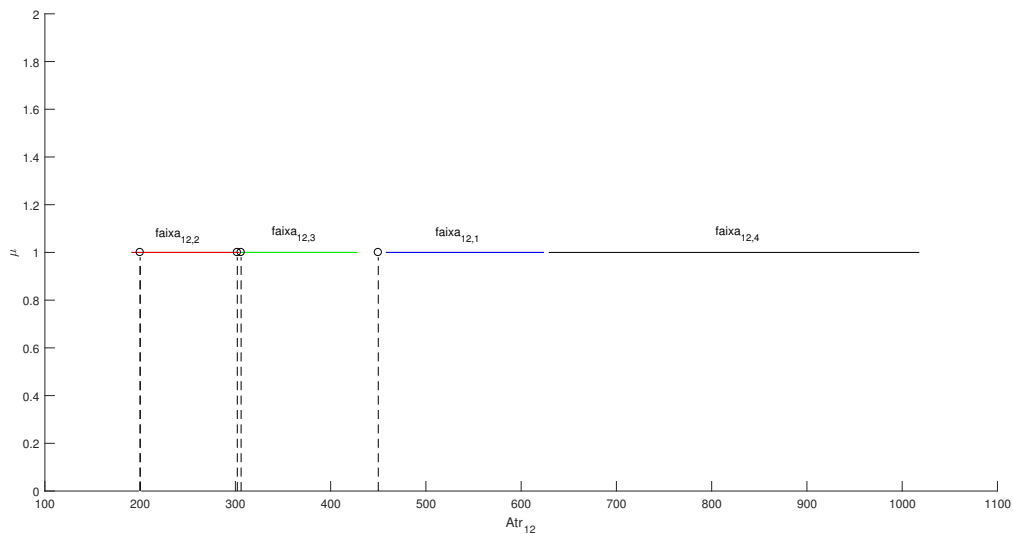


Figura 4 – Representação das faixas para o atributo  $Atr_{12}$ .

Ao observar a Figura 4 notam-se espaços entre as faixas, no qual surge uma dúvida sobre qual grupo o elemento será associado visto que não pertence a nenhum rótulo. Os possíveis grupos para esse elemento são os representados pelas faixas próximas ao espaço. Novamente utilizando o atributo  $Atr_{12}$  agora com valor igual à 450, como pode ser observado na Figura 4 ele não está contido em nenhuma faixa, mas está entre as faixas  $faixa_{12,3}$  e  $faixa_{12,1}$  que representam o Grupo 3 e Grupo 1.

Como discutido inicialmente nesta seção, a interpretação de um elemento que

possui valor contido nas faixas pertença a um grupo assemelha-se a teoria clássica dos conjuntos. Dito isto, um problema presente na teoria clássica também é encontrado nesta interpretação. Supondo dois elementos com valores para o atributo  $Atr_{12}$  igual à 302 e 306, apesar de possuírem valores próximos ambos são classificados em grupos diferentes por pertencerem a faixas diferentes. Neste caso, uma possível solução é que os elementos tenha uma pertinência para cada faixa o que poderia melhorar a representação para os elementos. O algoritmo de agrupamento *Fuzzy C-Means* por exemplo calcula a pertinência de cada elemento para todos os grupos, que é expresso pela Matriz  $U$ .

Baseado nesta interpretação foi desenvolvido um classificador inicial, apresentado na Qualificação, em que as funções de pertinência eram formadas pelas faixas e dois segmentos de retas. Uma partindo do menor elemento ou do ponto médio da faixa anterior ao início da faixa e a segunda partindo do final da faixa até o ponto médio da próxima faixa ou maior elemento presente na base.

Neste trabalho, a etapa de geração do classificador implementa uma solução para os espaços entre as faixas e cálculo da pertinência para os elementos que estão em outras faixas. Esta solução é a criação de uma função de pertinência, em que as faixas são transformadas em partições *Fuzzy*. A seguir é apresentado o passo a passo para criação das funções de pertinência.

Para cada atributo presente em algum rótulo uma função de pertinência é formada, utilizando cada intervalo de valores para auxiliar na formação das partições *Fuzzy*. A quantidade de partições que uma função de pertinência possui para um atributo é igual ao número de faixas presentes nos rótulos. Por exemplo, para os rótulos da Tabela 8 são formadas duas funções de pertinência, uma para o atributo  $Atr_7$  e outra para o atributo  $Atr_{12}$ , sendo duas e quatro partições respectivamente.

Como definido na Seção 2.2 o rótulo de um grupo tem como objetivo representar o grupo, baseado nisto assume-se que a pertinência dos elementos em um grupo  $C_i$  para os atributos que compõem o rótulo  $r_{c_i}$  é igual à pertinência do elemento  $x_j$  para o grupo  $C_i$ . As Tabelas 9 e 10 demonstram essa associação para o atributo  $Atr_{12}$  no Grupo 2.

Tabela 9 – Matriz  $U$  do Grupo 2 para base *Vehicle*.

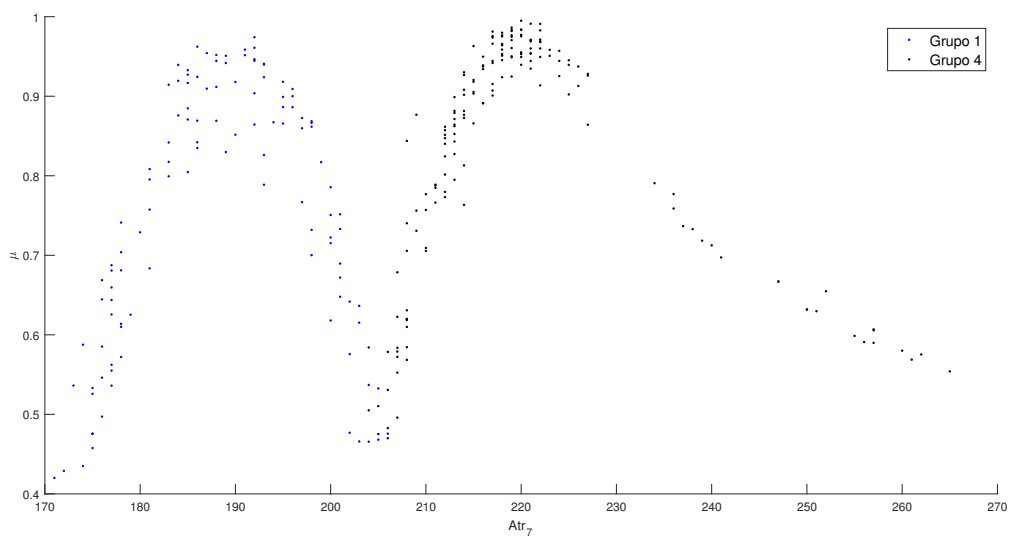
					Grau de Pertinência			
$Atr_1$	$Atr_2$	...	$Atr_{12}$	$Atr_{13}$	Grupo 1	Grupo 2	Grupo 3	Grupo 4
86	34		223	112	0,0204	0,8594	0,1108	0,0094
86	36		266	127	0,0064	0,9344	0,0565	0,0027
90	34		224	118	0,0181	0,8730	0,1006	0,0083
89	36		206	125	0,0267	0,8314	0,1290	0,0129
87	39		307	141	0,0228	0,4975	0,4713	0,0084

Realizando esta associação de atributo presente no rótulo e o seu grupo associado é possível visualizar como os dados de cada atributo estão distribuídos nos grupos. Para

Tabela 10 – Associação das pertinências dos elementos do Grupo 2 para o  $Atr_{12}$ .

$Atr_{12}$	Pertinência G2
223	0,8594
266	0,9344
224	0,8730
206	0,8314
307	0,4975

exemplificar as Figuras 5 e 6 mostram como estão distribuídos os dados dos atributos  $Atr_7$  e  $Atr_{12}$  para a base *Vehicle*.

Figura 5 – Pertinência dos elementos no  $Atr_7$  associados aos grupos.

Nas Figuras 5 e 6 é possível observar a pertinência dos elementos para um grupo até o início do próximo. Entretanto, é necessário conhecer como os elementos dos grupos próximos se relacionam com o grupo analisado. Como a Matriz  $U$  informa a pertinência dos elementos para todos os grupos, torna possível associar a pertinência dos elementos pertencentes a outros grupos com o grupo analisado e, ao ordenar as faixas de valores para um atributo é possível conhecer em que grupo os elementos das faixas ao lado pertence. Essa relação e identificação dos elementos podem ser observadas nas Figuras 7 e 8.

Na Figura 7 observa-se que o atributo  $Atr_7$  está presente em dois rótulos representando os Grupos 1 e 4. Os pontos em azuis na Figura 7 são os elementos pertencentes aos Grupos 1 e 4, pois são considerados próximos, associados a pertinência para o Grupo 1.

Neste ponto tem-se os elementos que representam cada partição sendo agora necessária uma função que represente esses elementos. Para o cálculo desta função foi utilizada ajuste de curva, no qual recebe como parâmetros um conjunto de valores e o grau do polinômio. Com estes parâmetros o ajuste de curva gera um polinômio de acordo com

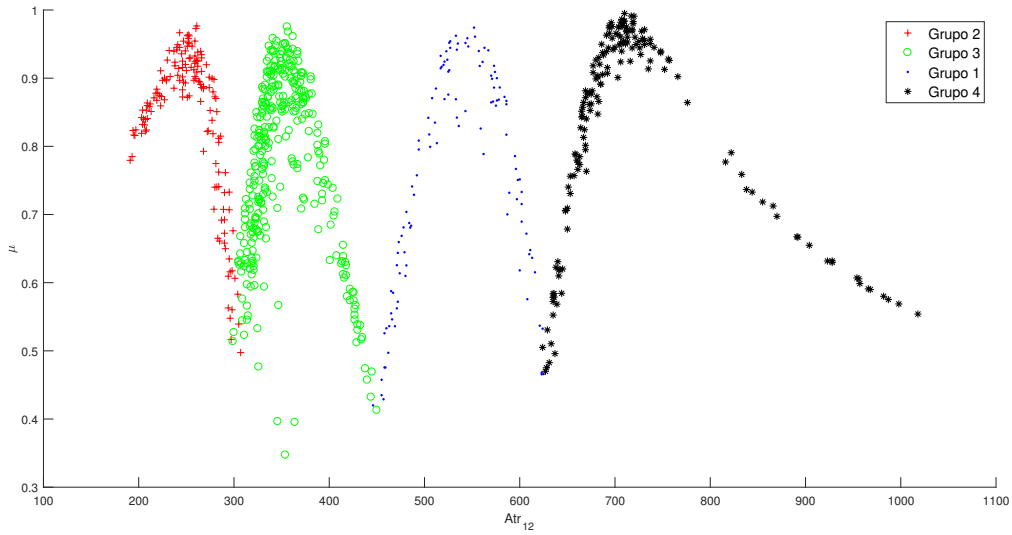


Figura 6 – Pertinência dos elementos no  $Atr_{12}$  associados aos grupos.

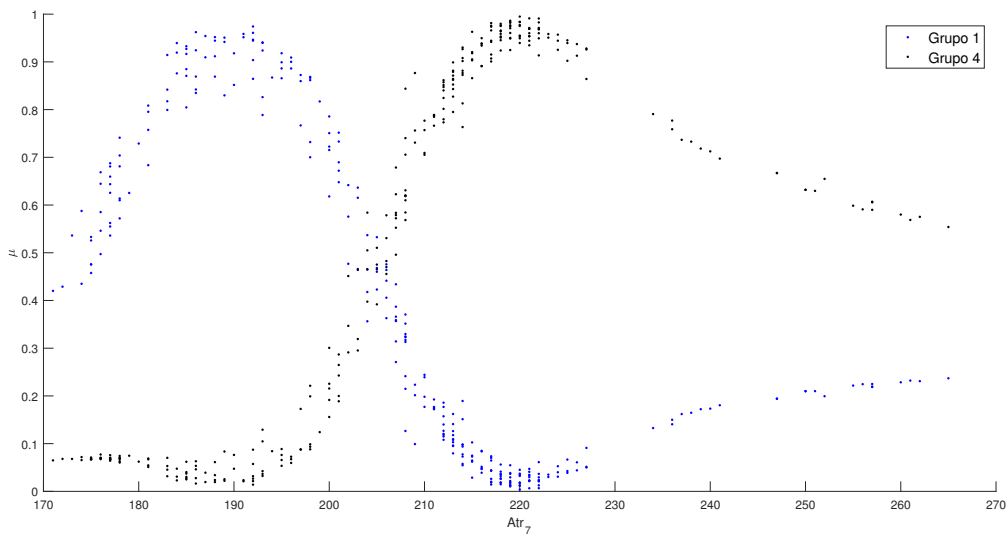


Figura 7 – Associação das pertinências dos elementos para o  $Atr_7$

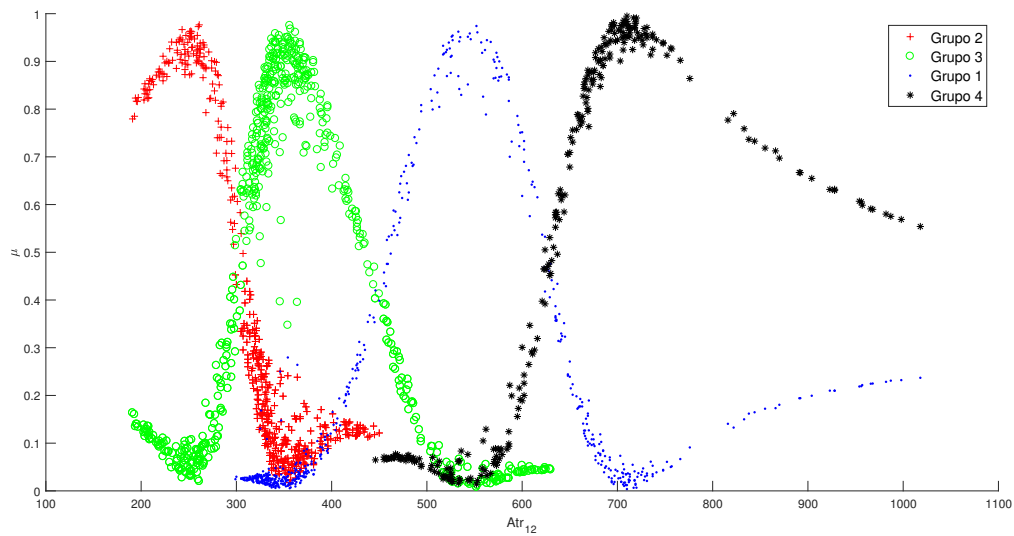


Figura 8 – Associação das pertinências dos elementos para o  $Atr_{12}$

o grau informado. Os testes iniciais considerando todos os elementos da partição podem ser observados na Figura 9.

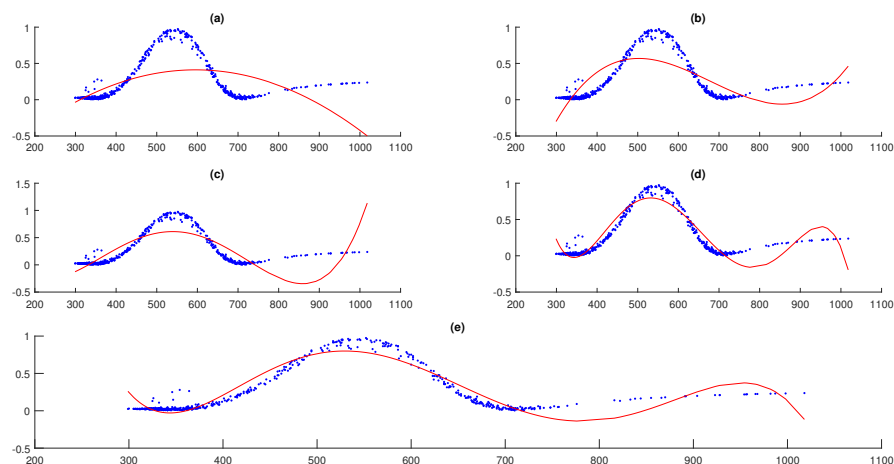


Figura 9 – Ajuste de Curva para elementos do  $Atr_{12}$  variando grau do polinômio: (a) grau 2; (b) grau 3; (c) grau 4; (d) grau 5 e (e) grau 6.

Observando a Figura 9 nota-se que à medida que aumenta o grau do polinômio mais a função calculada se ajusta aos elementos. Entretanto, o aumento do grau do polinômio necessita de mais tempo computacional e uma quantidade maior de elementos. A partir disto o cálculo da função para as partições foi dividido em duas partes. Primeiramente identifica-se o elemento que possui maior pertinência, que é o valor do atributo presente no centróide do grupo analisado. Visto que o algoritmo de agrupamento é baseado em centróides, quanto mais próximo do centróide maior a pertinência do elemento. Identificado o ponto de máximo divide-se os elementos em dois subconjuntos: um com elementos de

valores menores que o ponto de máximo e o outro com os elementos que possui valores maiores. Em seguida é aplicado o ajuste de curva com grau do polinômio igual à 3 nos dois subconjuntos. A função da partição final pode ser observada na Figura 10.

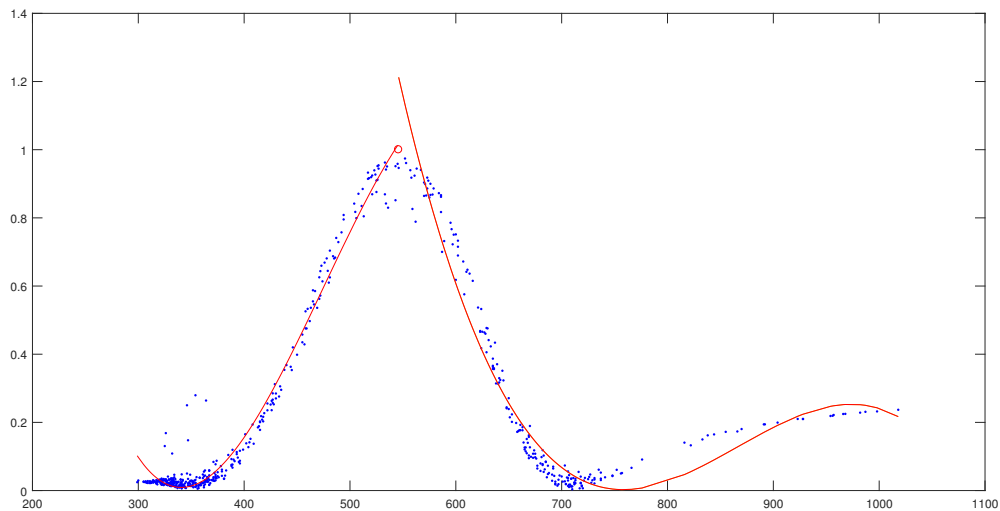


Figura 10 – Função da partição associada ao atributo  $Atr_{12}$  no Grupo 1.

Esse processo é aplicado em todas as partições de cada atributo formando as funções de pertinência. Para o exemplo da base *Vehicle* as funções de pertinência dos atributos  $Atr_7$  e  $Atr_{12}$  podem ser observadas nas Figuras 11 e 12 sendo duas partições para o atributo  $Atr_7$  e quatro para  $Atr_{12}$ .

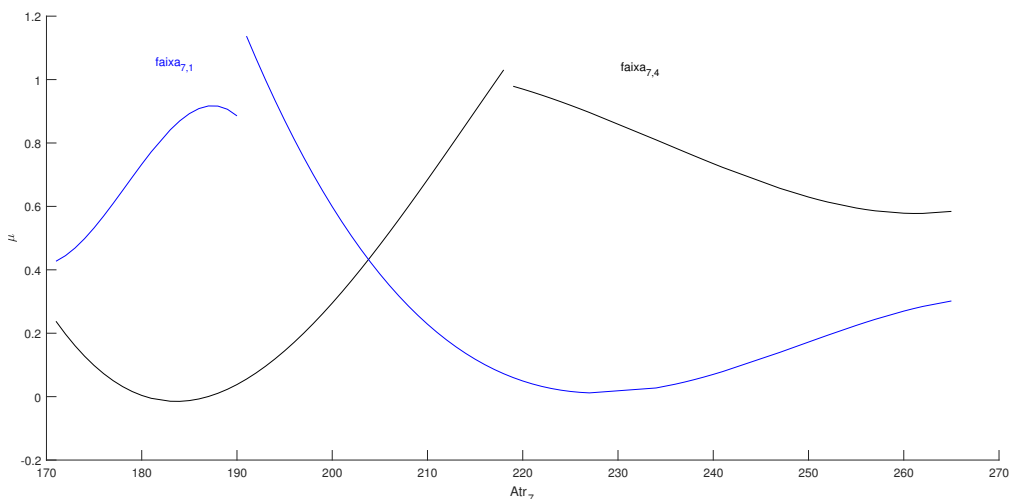


Figura 11 – Função de pertinência do atributo  $Atr_7$ .

Outro ajuste necessário ao observar as funções de pertinências é que para alguns valores o resultado é maior que 1 ou menor que 0. Nestes casos, nos quais os valores possuem resultados maiores que 1, a pertinência é igual a 1, e resultados menores que 0 a pertinência passa a ser igual a 0.

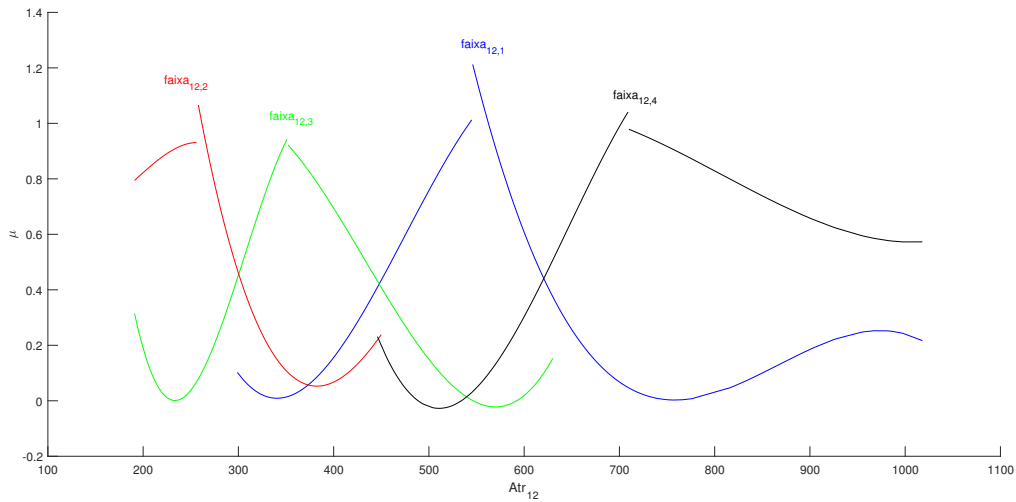


Figura 12 – Função de pertinência do atributo  $Atr_{12}$ .

As regras para inferência também são formadas usando os intervalos dos rótulos, onde os intervalos e atributos compõem o antecedente da regra e o grupo o consequente. Quando o grupo possui rótulo com dois ou mais intervalos é usado o conectivo *e* para formação das regras. As regras a seguir exemplificam esse processo para a base *Vehicle* utilizando os rótulos dispostos na Tabela 8:

- **Regra Grupo 1:** Se  $Atr_7 = faixa_{7,1}$  e  $Atr_{12} = faixa_{12,1}$  então Grupo 1;
- **Regra Grupo 2:** Se  $Atr_{12} = faixa_{12,2}$  então Grupo 2;
- **Regra Grupo 3:** Se  $Atr_{12} = faixa_{12,3}$  então Grupo 3;
- **Regra Grupo 4:** Se  $Atr_7 = faixa_{7,4}$  e  $Atr_{12} = faixa_{12,4}$  então Grupo 4.

Para classificação de um novo exemplo são avaliadas todas as regras, atribuindo-o ao grupo cuja regra obtiver o maior grau de ativação (GA). O GA é o valor resultante da pertinência de uma regra no processo de inferência. Caso haja mais de um componente no antecedente da regra, o GA será determinado por aquele que apresentar menor valor de pertinência.

Supondo um novo exemplo com  $Atr_7 = 210$  e  $Atr_{12} = 490$ , o modelo proposto irá calcular o grau de ativação para cada regra como mostra a seguir:

- $GA(R1) = \min \{ \mu_{faixa_{7,1}}(210), \mu_{faixa_{12,1}}(490) \} = \min\{0,22, 0,69\} = 0,22$ ;
- $GA(R2) = \min \{ \mu_{faixa_{12,2}}(490) \} = 0$ ;
- $GA(R3) = \min \{ \mu_{faixa_{12,3}}(490) \} = 0,19$ .
- $GA(R4) = \min \{ \mu_{faixa_{7,4}}(210), \mu_{faixa_{12,4}}(490) \} = \min\{0,67, 0\} = 0$ ;

Neste caso o novo exemplo será associado ao Grupo 1, pois obteve o maior grau

de ativação. Nas Figuras 13 e 14 podem ser observados os pontos calculados em cada partição.

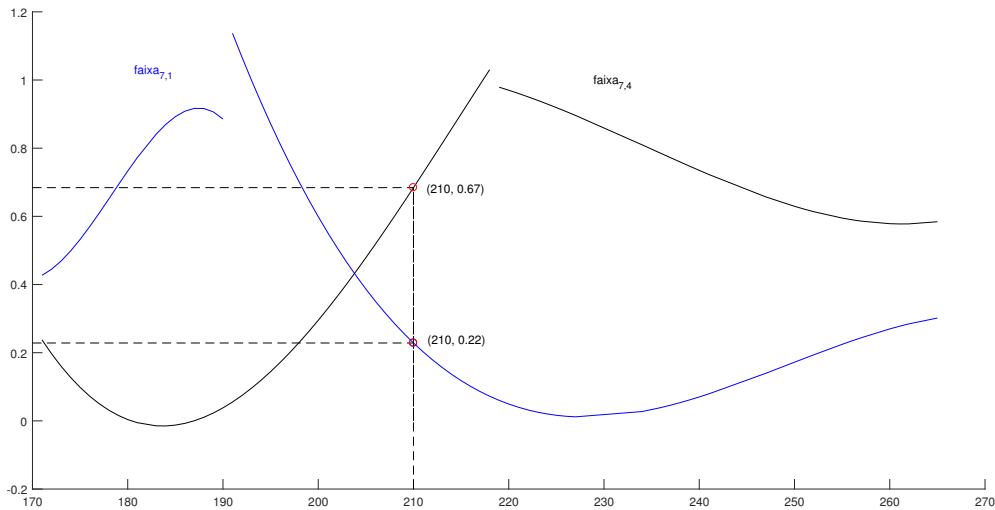


Figura 13 – Cálculo do GA para atributo  $Atr_7$ .

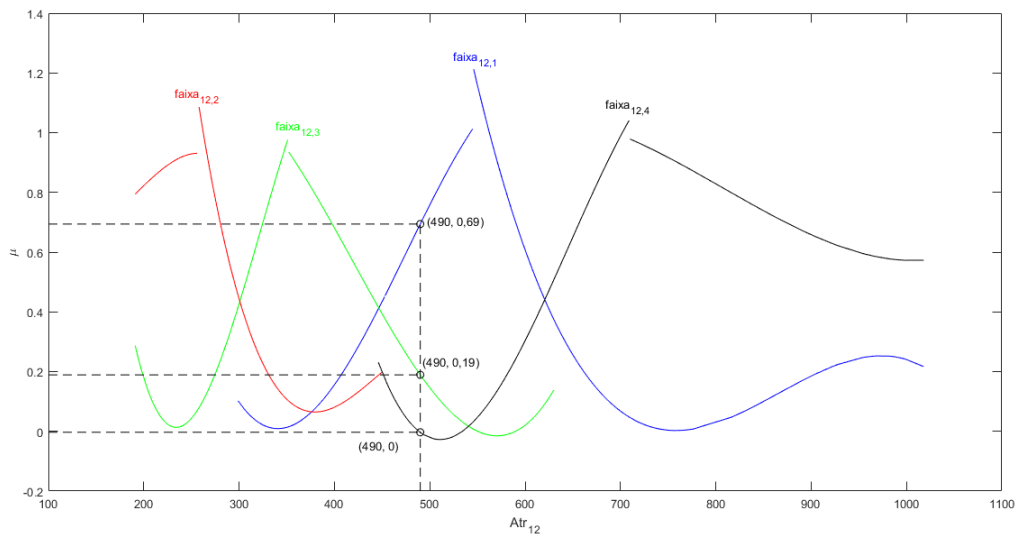


Figura 14 – Cálculo do GA para atributo  $Atr_{12}$ .

## 3.2 Bases

As bases utilizadas para testes foram: *Íris*, *Seeds*, *BUPA*, *Vehicle* e *Wine*; todas disponíveis no repositório do *UCI Machine Learning*.



### 3.2.1 Íris

A base *Íris* é formada por um conjunto de dados de três tipos diferentes da planta *Íris: Setosa, Versicolor e Virginica*. A base é composta por 50 elementos de cada tipo da planta.

Os 150 elementos presentes nesta base são representados por 4 características: comprimento da sépala (CS), largura da sépala (LS), comprimento da pétala (CP) e largura da pétala (LP). Todos esses valores são expressos em centímetros.

### 3.2.2 Seeds

Esta base refere-se à identificação de sementes de trigo e foi apresentada por Kulczycki e Charytanowicz ([KULCZYCKI; CHARYTANOWICZ, 2011](#)). A base é composta por três diferentes tipos de trigo: *Kama, Rosa e Canadian*.

Os 210 elementos presentes nesta base são representados por 7 características geométricas que formam o conjunto de entrada: área, perímetro, densidade, comprimento da semente (CS), largura da semente (LS), coeficiente de assimetria (CA) e comprimento do sulco da semente (CSS).

### 3.2.3 Liver Disorders - BUPA

A base Doenças do Fígado foi criada pelo *BUPA Medical Research*, e contém informações sobre exames de sangue consideradas sensíveis a problemas no fígado. A base possui informações de 345 pacientes do sexo masculino, representados por 5 características do exames e 1 referente ao número de bebidas alcólicas consumidas por dia, sendo essas: MCV, Alkphos, SGPT, SGOT, Gammagt e Drinks, respectivamente.

### 3.2.4 Vehicle Silhouettes

A base *Vehicle Silhouettes* foi criada com o objetivo de reconhecer objetos 3D a partir de uma imagem 2D. Com isso, pretende-se classificar a silhueta de um determinado tipo de veículo utilizando as características extraídas das imagens as quais continham representações dos objetos em vários ângulos distintos.

A base é composta por 846 elementos que podem ser classificadas em 4 classes distintas sendo elas: *bus*, um ônibus de dois andares (240 elementos); *van*, um modelo específico de van fabricado pela montadora Chevrolet (226 elementos); Saab 9000 (240 elementos) e Opel Manta 400 (240 elementos). Os elementos são descritos por 18 atributos obtidos por meio de um processamento de imagens, são eles: Densidade, Circularidade, Distância de Circularidade, Proporção do Raio, Proporção da Tela no Eixo Principal, Comprimento Máximo da Proporção na Tela, Taxa de Dispersão, Alongamento, Retangu-

laridade do Eixo Principal, Retangularidade do Comprimento Máximo, Variância Escalada ao Longo do Eixo Maior, Variância Escalada ao Longo do eixo Menor, Raio Escalado de Rotação, Distorção Sobre o Eixo Maior, Distorção Sobre o Eixo Menor, Achatamento Sobre o Eixo Maior, Achatamento Sobre o Eixo Menor e Proporção de Cavidades.

### 3.2.5 Wine

Esta base contém informações do Vinho Verde, um produto único da região Minho em Portugal. A base é composta por 4898 elementos e representada por 12 atributos: *fixed acidity*, *volatile acidity*, *citric acid*, *residual sugar*, *chlorides*, *free sulfur dioxide*, *total sulfur dioxide*, *density*, *pH*, *sulphates*, *alcohol* e *quality* sendo esta uma nota entre 0 e 10 para o vinho.

## 3.3 Algoritmos

Nos testes realizados neste trabalho são utilizados alguns algoritmos para comparação com o modelo proposto, a seguir são detalhados estes algoritmos:

### 3.3.1 Árvore de Decisão

Árvore de Decisão é um dos métodos de aprendizado simbólico mais amplamente utilizados e práticos. dos métodos de aprendizagem conceitual, são mais robustas a ruídos nos dados e também pode ser representada por regras (Se - Então).

As Árvores, de um modo geral, são estruturas de dados formadas por um conjunto de elementos que armazenam informações chamadas nós. Além disso, toda árvore possui um nó chamado raiz, que possui o maior nível hierárquico (o ponto de partida) e ligações para outros elementos, denominados filhos. Esses filhos podem possuir seus próprios filhos que por sua vez também possuem os seus. O nó que não possui filho é conhecido como nó folha ou terminal.

Para os testes foi utilizada uma função do MATLAB chamada *fitctree* que retorna uma árvore de classificação binária para problemas com múltiplas classes.

### 3.3.2 Algoritmo K-Vizinhos mais Próximos (*K-Nearest Neighbors* - KNN)

O K-Vizinhos mais Próximos é um algoritmo de aprendizado supervisionado que foi introduzido na literatura por [Aha, Kibler e Albert \(1991\)](#). O objetivo do KNN é encontrar os k vizinhos mais próximos do exemplo não classificado e, baseado no rótulo desses exemplos mais próximos, determinar a classe do exemplo não-rotulado. O KNN requer pouco custo computacional na etapa de treinamento. Em contrapartida, o esforço

para classificar um novo exemplo é relativamente alto, pois no pior caso o novo exemplo é comparado com todos os exemplos contidos no conjunto de treinamento.

A função no MATLAB responsável pelo treinamento do KNN é chamada *fitcknn*.

### 3.3.3 Máquina de Vetor de Suporte (*Support Vector Machine* - SVM)

As máquinas de vetor de suporte (SVM) foram propostas por Vapnik (2013) e têm obtido sucesso em um grande número de aplicações. As SVM's são utilizadas na classificação de padrões, em que seu objetivo é construir um hiperplano de decisão de tal forma que a margem de separação entre os exemplos positivos e negativos sejam a máxima possível (CAMPBELL, 2002).

De início a SVM constrói hiperplano para problemas linearmente separáveis para casos em que isto não ocorre uma solução é utilizar uma função *kernel*(CRISTIANINI; SHAWE-TAYLOR et al., 2000). Com esta função, é possível mapear o espaço de características do problema. Então, as máquinas de vetor de suporte constroem um hiperplano de separação ótimo neste espaço. A medida que se aumenta o espaço da dimensão do problema, também aumenta a probabilidade desse problema se tornar linearmente separável em relação à um espaço de baixa dimensão.

Para classificação com múltiplas classes utilizando SVM's no MATLAB é utilizada a função *fitcecoc*.

## 3.4 Métricas de Avaliação

Para avaliação do modelo proposto foram utilizadas as métricas Acurácia (Acc), Desvio Padrão (DP), Índice *Kappa* (K), Tempo de Execução (TE) e Quantidade de Regras (QR). A Acurácia é utilizada para medir quão frequentemente o classificador está correto. A acurácia é calculada utilizando a quantidade de elementos que o classificador acertou pela quantidade total de elementos usados para testes. Como nos testes são realizadas 10 testes para cada base, o Desvio Padrão é calculado para verificar a variação das acurácias para os 10 casos de testes.

Outra medida utilizada foi o índice *Kappa* que é recomendado como outra medida além da acurácia. O cálculo do índice *Kappa* leva em consideração todos os elementos presentes no teste ao invés de somente os elementos classificados corretamente o que acontece no cálculo da Acurácia. Para Landis e Koch (1977) o valor de K assume valores entre 0 e 1. O resultado é qualificado de acordo com o valor de K no qual:  $K \leq 0,2$ : Ruim;  $0,2 \leq K \leq 0,4$ : Razoável;  $0,4 \leq K \leq 0,6$ : Bom;  $0,6 \leq K \leq 0,8$ : Muito Bom;  $K > 0,8$ : Excelente.

No modelo proposto é considerado o tempo de execução, que é o tempo necessário

para selecionar os elementos que representam cada partição e o tempo para cálculo das funções. Para o algoritmo baseado em Árvore de Decisão e modelo proposto foi calculado a quantidade de regras geradas.

## 4 Resultados e Discussão

Neste capítulo são apresentados os testes feitos utilizando o modelo de classificação proposto, verificando a eficiência em diversas bases e as regras formadas pelo modelo. Ao final é realizada uma discussão sobre os resultados apresentados.

O modelo proposto foi implementado utilizando a ferramenta *MATLAB Math Works*<sup>1</sup>. Uma ferramenta para teste de modelos, que permite manipulação de matrizes e possui uma variedade de *toolboxes* que facilita a construção dos algoritmos.

Na etapa de agrupamento foi utilizado o algoritmo *Fuzzy C-Means* com parâmetro  $m$  igual a 2 e o valor de  $k$  foi definido pela quantidade de classes presentes na base. Para bases em que não se conhece a quantidade de classe o valor de  $k$  varia entre 2 e 4. O modelo de rotulação utilizado foi o proposto por Machado, Ribeiro e Rabêlo (2015) descrito na Seção 2.4.

Os testes de comparação do modelo com os algoritmos de classificação presentes na literatura foi realizado usando *cross-validation* com 10 *folds*. Nos testes assume-se como classe do elemento o grupo ao qual ele está associado.

### 4.1 Experimentos

Os testes iniciais foram realizados utilizando bases em que a classe já são conhecidas que são utilizadas como parâmetro para o algoritmo de agrupamento. As bases selecionadas são: *Íris*, *Seeds*, *BUPA* e *Vehicle Silhouettes*. Os resultados obtidos são apresentados na Tabela 11.

Tabela 11 – Resultados do modelo proposto com outros algoritmos.

Bases	Árvore			KNN			SVM			Modelo		
	Acc	K	DP	Acc	K	DP	Acc	K	DP	Acc	K	DP
Íris	96,7	0,95	4,56	92,74	0,88	9,48	<b>97,37</b>	0,96	4,62	94,01	0,90	6,3
Seeds	96,7	0,94	3,03	96,57	0,94	3,44	<b>98,44</b>	0,97	2,5	91,79	0,87	5,86
BUPA	85,55	0,38	5,43	88,14	0,38	3,41	89,52	0,51	5,05	<b>96,27</b>	0,84	3,15
Vehicle	97,88	0,97	1,72	93,38	0,90	1,90	<b>98,69</b>	0,98	1,41	98,35	0,97	1,46

Outras medidas utilizadas para comparação foi o tempo de treinamento e a quantidade de regras. Para essas medidas a Tabela 12 apresenta os resultados obtidos.

Ao observar a Tabela 11, a taxa de acurácia obtidas mostraram-se semelhantes, sendo a SVM com as maiores taxa de acurácia para as base *Íris*, *Seeds* e *Vehicle* e KNN

<sup>1</sup> <https://www.mathworks.com/products/matlab.html>

Tabela 12 – Resultados para tempo de treinamento e quantidade de regras.

Bases	Árvore		KNN	SVM	Modelo	
	TE	QR	TE	TE	TE	QR
Íris	<b>0,035</b>	5,2	0,121	0,119	0,041	<b>3</b>
Seeds	<b>0,029</b>	5,9	0,032	0,106	0,038	<b>3</b>
BUPA	<b>0,023</b>	17	0,028	2,218	0,03	<b>2</b>
Vehicle	0,031	6,5	<b>0,024</b>	0,204	0,068	<b>4</b>

as menores taxa de acurácia. Entretanto, o tempo de execução para a SVM observado na Tabela 12, se mostra superior aos outros algoritmos em todas as bases.

Considerando a medida tempo de execução, os algoritmos Árvore, KNN e modelo proposto obtiveram tempos similares entre si. Vale destacar na Tabela 12 a quantidade de regras obtidas pela Árvore e pelo modelo proposto. Nos resultados a Árvore apresenta uma quantidade de regras superior ao modelo. Na base BUPA a Árvore obteve em média 15 regras enquanto o modelo proposto somente 2.

Para a base *Wine* não é informada a quantidade de classes, nesse caso os testes foram realizados variando o parâmetro  $k$  entre 2 e 4. Os resultados da acurácia, desvio padrão e índice  $kappa$  para base *Wine* podem ser observados na Tabela 13.

Tabela 13 – Resultados para base *Wine*.

k	Árvore			KNN			SVM			Modelo		
	Acc	$K$	DP	Acc	$K$	DP	Acc	$K$	DP	Acc	$K$	DP
2	99,44	0,98	0,30	92,73	0,85	0,85	<b>99,91</b>	0,99	0,14	97,97	0,95	0,81
3	98,75	0,98	0,39	84,52	0,76	1,36	<b>99,85</b>	0,99	0,16	95,74	0,93	1,22
4	98,30	0,97	0,32	78,54	0,70	1,95	<b>99,71</b>	0,99	0,17	93,87	0,91	0,87

Como realizado nas outras bases, os testes contabilizando tempo de treinamento e quantidade de regras foi feito para base *Wine* como observado na Tabela 14.

Tabela 14 – Resultados do tempo de treinamento para base *Wine*.

k	Árvore		KNN	SVM	Modelo	
	TT	QR	TE	TE	TE	QR
2	0,026	25	0,027	1,475	<b>0,013</b>	<b>2</b>
3	0,030	37,3	0,017	4,523	<b>0,010</b>	<b>3</b>
4	0,032	47,3	0,016	2,185	<b>0,0131</b>	<b>4</b>

Os resultados para *Wine* são semelhantes aos das outras bases com SVM medindo maiores taxas de acurácias e maior tempo de execução. O mesmo acontece quando comparado a quantidade de regras geradas pela árvore e modelo proposto no qual a Árvore gera cerca de 10 vezes mais a quantidade de regras.

## 4.2 Avaliação dos Resultados

Os resultados observados do modelo proposto podem ser discutidos em dois pontos. Primeiro em relação à acurácia do modelo em comparação aos outros algoritmos. Nas bases Iris, Seeds, Vehicle e Wine o modelo proposto obteve acurácia semelhantes aos demais algoritmos. A SVM obteve as melhores taxas de acurácias para estas bases e KNN as menores taxas. Na base BUPA o modelo proposto obteve acurácia superior aos outros algoritmos, cerca de 6,5 % a mais que a SVM. Isto mostra que o modelo proposto obtêm acurácias similares ou superiores aos algoritmos presentes na literatura e valor de índice *Kappa* considerados excelentes.

O segundo ponto a ser discutido é em relação ao tempo de treinamento e quantidade de regras observadas. A SVM apesar de obter melhores taxas de acurácias teve tempo de treinamento superior aos outros algoritmos elevando-se para bases com maiores quantidade de elementos. O tempo de execução entre Árvore de Decisão, KNN e modelo proposto se mostraram similares. O algoritmo KNN apesar de obter tempo similar ao modelo foi o algoritmo com menores taxas de acurácias. A comparação entre o algoritmo baseado em Árvore e o modelo proposto fica em relação às regras formadas sendo o algoritmo baseado em Árvore gerando cerca de 10 vezes mais regras, o que ocorreu na base *Wine*. Para o modelo proposto a quantidade de regras geradas é igual à quantidade de grupos formados. Uma grande quantidade de regras pode dificultar o trabalho de interpretação do conhecimento gerado pelas Árvores de Decisão.





## 5 Conclusões e Trabalhos Futuros

Neste trabalho, foi proposto um modelo de classificação *Fuzzy* baseado no processo de rotulação de grupos. Inicialmente é aplicado um modelo de rotulação sobre os agrupamentos para extração dos rótulos a fim de melhor interpretar os grupos. Após a rotulação, os rótulos são utilizados para gerar regras e funções para os grupos podendo classificar novos elementos, além de facilitar a interpretação do conhecimento adquirido.

O modelo proposto utiliza o modelo de rotulação baseados em conjuntos *Fuzzy* proposto por Machado, Ribeiro e Rabêlo (2015) para rotulação dos grupos. Após os grupos rotulados são formadas funções de pertinências as quais são utilizados os rótulos e a matriz de pertinência resultante do agrupamento. São selecionados os elementos que representam cada partição e utilizada uma função de ajuste de curva para calcular uma função que represente os elementos. O cálculo da função é dividida em duas partes: uma função para elementos maiores que o ponto de máximo e outra para elementos menores, ambas polinômios de grau três.

A classificação de novos elementos é feita executando todas as regras para o novo elemento e calculando o grau de ativação de acordo com as funções de pertinências geradas. A regra que obtiver o maior grau de ativação será o qual o elemento irá pertencer.

Foram utilizadas cinco bases de dados para testes citadas no Capítulo 4 sendo analisadas a Acurácia, Desvio Padrão e Índice *Kappa* para avaliar a eficácia do modelo em relação aos algoritmos presentes na literatura. Outras medidas foram calculadas para verificar o tempo de processamento dos algoritmos sendo elas tempo de execução e quantidade de regras quando comparado ao algoritmo baseado em Árvore e modelo proposto.

A execução do modelo para as bases *Íris*, *Seeds*, *Vehicle* e *Wine* obtiveram acurácia similares sendo SVM possuindo maiores taxas. Na base BUPA o modelo proposto obteve acurácia superior aos outros algoritmos. O índice *Kappa* para o modelo proposto em todas as bases foi maior que 0,8 o que é considerado Excelente.

Os testes considerando o tempo de treinamento mostraram que a SVM possui tempo de treinamento superior aos outros algoritmos. Para os algoritmos baseados em Árvore de Decisão, KNN e modelo proposto o tempo de treinamento foi similar. A comparação entre a quantidade de regras geradas pelo algoritmo baseado em Árvore e o modelo proposto mostrou que o algoritmo baseado em Árvore gerou uma quantidade de regras superior ao modelo. Na base *Wine* a quantidade de regras geradas pelo algoritmo baseado em Árvore foi 10 vezes mais que o modelo.

Os resultados obtidos utilizando o modelo proposto mostraram-se promissores, o

fato de utilizar informações já disponíveis (Matriz U e rótulos) oriundas do agrupamento e do modelo de rotulação otimiza o tempo necessário para construção do modelo de classificação. A métrica de acurácia mostrou que utilizando os rótulos como representantes dos grupos possui taxa similar aos outros algoritmos. Destaca-se também a quantidade de regras geradas pelo modelo proposto que mostrou-se eficaz na classificação e em menor quantidade, o que pode auxiliar na interpretação dos resultados.

## Trabalhos Futuros

Considerando os experimentos realizados, alguns trabalhos futuros são propostos. Tais como avaliação do tipo dos dados utilizados pelo algoritmo de agrupamento. O impacto que *outliers* podem causar no modelo. Realizar uma análise sobre o grau de separabilidade dos grupos e que mudanças podem ocorrer na seleção dos elementos para formar as partições. Pretendemos também aplicar outras formas para calcular as funções de pertinência como por exemplo interporlações. Por fim, aplicar um método que incorpore os novos elementos para auxiliar o classificador.

# Referências

AHA, D. W.; KIBLER, D.; ALBERT, M. K. Instance-based learning algorithms. *Machine learning*, Springer, v. 6, n. 1, p. 37–66, 1991. Citado na página 30.

ARAÚJO, F. N. de et al. Automatic cluster labeling based on phylogram analysis. In: IEEE. *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2018. p. 1–8. Citado na página 18.

ATILGAN, C.; NASIBOV, E. A memory efficient distributed fuzzy joint points clustering algorithm. In: IEEE. *2016 IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*. [S.l.], 2016. p. 1–5. Citado 2 vezes nas páginas 6 e 7.

BARBER, D. *Bayesian reasoning and machine learning*. [S.l.]: Cambridge University Press, 2012. Citado na página 6.

BEZDEK, J. C. Objective function clustering. In: *Pattern recognition with fuzzy objective function algorithms*. [S.l.]: Springer, 1981. p. 43–93. Citado na página 1.

BEZDEK, J. C. *Pattern recognition with fuzzy objective function algorithms*. [S.l.]: Springer Science & Business Media, 2013. Citado na página 12.

BISCARRI, F. et al. Electricity clustering framework for automatic classification of customer loads. *Expert Systems with Applications*, Elsevier, v. 86, p. 54–63, 2017. Citado na página 17.

CAMPBELL, C. Kernel methods: a survey of current techniques. *Neurocomputing*, Elsevier, v. 48, n. 1-4, p. 63–84, 2002. Citado na página 31.

CHUANG, S.-L.; CHIEN, L.-F. A practical web-based approach to generating topic hierarchy for text segments. In: ACM. *Proceedings of the thirteenth ACM international conference on Information and knowledge management*. [S.l.], 2004. p. 127–136. Citado na página 18.

COSTA, B. S. J. et al. Unsupervised classification of data streams based on typicality and eccentricity data analytics. In: IEEE. *International Conference on Fuzzy Systems (FUZZ-IEEE)*. [S.l.], 2016. p. 58–63. Citado na página 17.

COVER, T.; HART, P. Nearest neighbor pattern classification. *Transactions on information theory*, IEEE, v. 13, n. 1, p. 21–27, 1967. Citado na página 6.

CRISTIANINI, N.; SHAWE-TAYLOR, J. et al. *An introduction to support vector machines and other kernel-based learning methods*. [S.l.]: Cambridge university press, 2000. Citado na página 31.

ELYASIGOMARI, V. et al. Cancer classification using a novel gene selection approach by means of shuffling based on data clustering with optimization. *Applied Soft Computing*, Elsevier, v. 35, p. 43–51, 2015. Citado na página 17.

ESMIN, A. A. A.; LAMBERT-TORRES, G. Generate and optimize fuzzy rules using the particle swarm algorithm. In: IEEE. *International Conference on Systems Man and Cybernetics (SMC)*. [S.l.], 2010. p. 4244–4250. Citado na página 17.

ESTER, M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Kdd*. [S.l.: s.n.], 1996. v. 96, n. 34, p. 226–231. Citado na página 1.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, Elsevier, v. 55, n. 1, p. 119–139, 1997. Citado na página 16.

GAN, H. et al. On using supervised clustering analysis to improve classification performance. *Information Sciences*, Elsevier, v. 454, p. 216–228, 2018. Citado na página 17.

GLOVER, E. et al. Inferring hierarchical descriptions. In: ACM. *Proceedings of the eleventh international conference on Information and knowledge management*. [S.l.], 2002. p. 507–514. Citado na página 18.

GUHA, S.; RASTOGI, R.; SHIM, K. Cure: an efficient clustering algorithm for large databases. In: ACM. *ACM Sigmod Record*. [S.l.], 1998. v. 27, n. 2, p. 73–84. Citado na página 1.

IMPERES Filho, F. d. C. *Rotulação de Grupos em Algoritmos de Agrupamento Baseados em Distância Utilizando Grau de Pertinência*. 2018. Dissertação - Universidade Federal do Piauí. Citado na página 18.

JAMSHIDI, M.; VADIEE, N.; ROSS, T. *Fuzzy logic and control: software and hardware applications*. [S.l.]: Pearson Education, 1993. v. 2. Citado na página 10.

KULCZYCKI, P.; CHARYTANOWICZ, M. A complete gradient clustering algorithm. In: SPRINGER. *International Conference on Artificial Intelligence and Computational Intelligence*. [S.l.], 2011. p. 497–504. Citado na página 29.

LANDIS, J. R.; KOCH, G. G. The measurement of observer agreement for categorical data. *biometrics*, JSTOR, p. 159–174, 1977. Citado na página 31.

LOPES, L. A. et al. Automatic labelling of clusters of discrete and continuous data with supervised machine learning. *Knowledge-Based Systems*, Elsevier, v. 106, p. 231–241, 2016. Citado 5 vezes nas páginas 2, 7, 8, 18 e 20.

LOPES, P. de A.; CAMARGO, H. de A. Automatic labeling by means of semi-supervised fuzzy clustering as a boosting mechanism in the generation of fuzzy rules. In: IEEE. *13th International Conference on Information Reuse and Integration (IRI)*. [S.l.], 2012. p. 279–286. Citado na página 16.

MACHADO, V. P.; RIBEIRO, V. P.; RABÊLO, R. de A. L. Rotulação de grupos utilizando conjuntos fuzzy. In: *Simpósio Brasileiro de Automação Inteligente-SBAI*. [S.l.: s.n.], 2015. p. 2. Citado 8 vezes nas páginas 13, 2, 8, 12, 18, 20, 33 e 37.

- MACQUEEN, J. et al. Some methods for classification and analysis of multivariate observations. In: OAKLAND, CA, USA. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. [S.l.], 1967. v. 1, n. 14, p. 281–297. Citado na página 1.
- MAQBOOL, O.; BABRI, H. Interpreting clustering results through cluster labeling. In: IEEE. *Proceedings of the IEEE Symposium on Emerging Technologies*. [S.l.], 2005. p. 429–434. Citado na página 18.
- MICHALSKI, R. S. A theory and methodology of inductive learning. In: *Machine learning*. [S.l.]: Springer, 1983. p. 83–134. Citado na página 5.
- MILARÉ, C. R.; CARVALHO, A. C. D. L. D.; MONARD, M. C. An approach to explain neural networks using symbolic algorithms. *International Journal of Computational Intelligence and Applications*, World Scientific, v. 2, n. 04, p. 365–376, 2002. Citado na página 1.
- MITCHELL, T. M. et al. Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, v. 45, n. 37, p. 870–877, 1997. Citado 2 vezes nas páginas 1 e 5.
- NOMURA, H.; HAYASHI, I.; WAKAMI, N. A learning method of fuzzy inference rules by descent method. In: IEEE. *IEEE International Conference on Fuzzy Systems*. [S.l.], 1992. p. 203–210. Citado na página 1.
- OTERO, F. E.; FREITAS, A. A.; JOHNSON, C. G. A new sequential covering strategy for inducing classification rules with ant colony algorithms. *IEEE Trans. Evolutionary Computation*, v. 17, n. 1, p. 64–76, 2013. Citado na página 17.
- PEDRYCZ, W.; GOMIDE, F. *An introduction to fuzzy sets: analysis and design*. [S.l.]: Mit Press, 1998. Citado na página 8.
- POPESCU, A.; UNGAR, L. H. Automatic labeling of document clusters. 2000. Citado na página 18.
- RUSSELL, S. J.; NORVIG, P. *Artificial intelligence: a modern approach*. [S.l.]: Malaysia; Pearson Education Limited, 2016. Citado na página 1.
- SAFAVIAN, S. R.; LANDGREBE, D. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, IEEE, v. 21, n. 3, p. 660–674, 1991. Citado na página 1.
- SHOBHA, T.; ANANDHI, R. Classification rule discovery using variant genetic algorithm. In: IEEE. *International Conference on Circuits, Controls, and Communications (CCUBE)*. [S.l.], 2017. p. 222–225. Citado na página 17.
- SILVA, I. d.; SPATTI, D. H.; FLAUZINO, R. A. Redes neurais artificiais para engenharia e ciências aplicadas. *São Paulo: Artliber*, v. 23, n. 5, p. 33–111, 2010. Citado na página 1.
- THOLE, U.; ZIMMERMANN, H.-J.; ZYSNO, P. On the suitability of minimum and product operators for the intersection of fuzzy sets. *Fuzzy Sets and systems*, Elsevier, v. 2, n. 2, p. 167–180, 1979. Citado na página 8.
- VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer science & business media, 2013. Citado na página 31.

- WANG, L.-X.; MENDEL, J. M. Generating fuzzy rules by learning from examples. *Transactions on systems, man, and cybernetics*, IEEE, v. 22, n. 6, p. 1414–1427, 1992. Citado na página 16.
- YANG, C.-L.; QUYEN, N. T. P. Data analysis framework of sequential clustering and classification using non-dominated sorting genetic algorithm. *Applied Soft Computing*, Elsevier, 2018. Citado na página 18.
- YEN, J.; LANGARI, R. *Fuzzy logic: intelligence, control, and information*. [S.l.]: Prentice Hall Upper Saddle River, NJ, 1999. v. 1. Citado na página 8.
- ZADEH, L. A. et al. Fuzzy sets. *Information and control*, New York, v. 8, n. 3, p. 338–353, 1965. Citado na página 8.
- ZIMMERMANN, H.-J. Fuzzy control. In: *Fuzzy Set Theory—and Its Applications*. [S.l.]: Springer, 1996. p. 203–240. Citado 2 vezes nas páginas 8 e 9.