UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

GABRIEL HARRISON FIDELIS TEOTONIO

Variable Weighted Fuzzy Clustering Algorithm For Qualitative Data

Recife

2023

GABRIEL HARRISON FIDELIS TEOTONIO

Variable Weighted Fuzzy Clustering Algorithm For Qualitative Data

A M.Sc. Dissertation presented to the Graduate Program in Computer Science of the Center for Informatics of the Federal University of Pernambuco, as a partial requirement for obtaining the title of Master of Science in Computer Science. Concentration Area: Computational Intelligence.

Advisor: Renata Maria Cardoso Rodrigues de Souza

Co-advisor: Getúlio José Amorim do Amaral

Recife

2023

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

T314v Teotonio, Gabriel Harrison Fidelis

Variable weighted fuzzy clustering algorithm for qualitative data / Gabriel Harrison Fidelis Teotonio. – 2023.

68 f.: il., fig., tab.

Orientadora: Renata Maria Cardoso Rodrigues de Souza.

Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2023.

Inclui referências.

1. Inteligência computacional. 2. Agrupamento. I. Souza, Renata Maria Cardoso Rodrigues de (orientadora). II. Título.

006.31 CDD (23. ed.) UFPE - CCEN 2023-169

Gabriel Harrison Fidelis Teotonio

“Variable Weighted Fuzzy Clustering Algorithm For Qualitative Data”

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 25 de maio de 2023.

BANCA EXAMINADORA

Prof. Dr. Nivan Roberto Ferreira Júnior
Centro de Informática / UFPE

Prof. Dr. Bruno Almeida Pimentel
Instituto de Computação / UFAL

Profa. Dra. Renata Maria Cardoso Rodrigues de Souza
Centro de Informática / UFPE
(**Orientadora**)

To my family and friends.

ACKNOWLEDGMENTS

I am deeply thankful to my family and friends for their unwavering love and support throughout this journey. They have been my source of strength and motivation, and I could not have accomplished this without them. Their encouragement and belief in me has been invaluable and I am eternally grateful for their presence in my life.

I would also like to acknowledge the guidance and support provided by my advisor, Prof. Renata de Souza, and co-advisor, Getúlio do Amaral. Their contributions to this project were greatly appreciated, and I am grateful for their expertise and wisdom.

ABSTRACT

This work focuses on the clustering methods within unsupervised learning, a challenging sub-division of Machine Learning where there is no response variable available. Clustering is a technique for finding groups in a dataset, where the observations in each group are similar to each other and different from those in other groups. The K-Means method, recognized as the most well-known and widely used clustering technique, efficiently handles quantitative variables, like many other existing clustering methods. However, the K-Means algorithm cannot be used with qualitative variables, such as gender or education level. To overcome this limitation, the K-Modes method was proposed, which uses modes instead of means to represent the clusters. The existing partitional clustering algorithms without variable weighting have a limitation in that they assign equal importance to all variables. It can be problematic when clustering high-dimensional, sparse data where the characterization of cluster partitions can be explained by a particular subset of variables. To address this issue, subspace clustering techniques and adaptive distances have been proposed, with the latter being derived from constraints based on the sum and product of the weights relative to the importance of the variables. This work proposes a new fuzzy clustering algorithm for qualitative data based on adaptive distances, which demonstrates improved performance compared to conventional methods. The local adaptive distances, which assign different weights to each variable across the clusters, perform better for datasets with high levels of dispersion and overlap of classes. The results extend the capabilities of existing clustering algorithms based on adaptive distances.

Keywords: clustering; unsupervised learning; adaptive distances; qualitative data.

RESUMO

Este trabalho se concentra nos métodos de agrupamento dentro do aprendizado não supervisionado, uma subdivisão desafiadora da Aprendizagem de Máquina onde não há variável resposta disponível. O agrupamento é uma técnica para encontrar grupos em um conjunto de dados, onde as observações em cada grupo são semelhantes umas às outras e diferentes das observações em outros grupos. O método K-*Means*, reconhecido como a técnica de agrupamento mais conhecida e amplamente utilizada, lida de forma eficiente com variáveis quantitativas, assim como muitos outros métodos de agrupamento existentes. No entanto, o algoritmo K-*Means* não pode ser usado com variáveis qualitativas, como gênero ou nível de educação. Para superar esta limitação, foi proposto o método K-*Modes*, que usa modas em vez de médias para representar os grupos. Os algoritmos de agrupamento particional existentes sem ponderação variável têm a limitação de atribuir importância igual a todas as variáveis. Isso pode ser problemático ao agrupar dados de alta dimensão e esparsos, onde a caracterização das partições do agrupamento pode ser explicada por um subconjunto particular de variáveis. Para abordar este problema, foram propostas técnicas de agrupamento de subespaço e distâncias adaptativas, sendo estas últimas derivadas a partir de restrições baseadas na soma e no produto dos pesos relativos à importância das variáveis. Este trabalho propõe um novo algoritmo de agrupamento difuso para dados qualitativos baseado em distâncias adaptativas, o qual demonstra desempenho melhorado em comparação aos métodos convencionais. As distâncias adaptativas locais, que atribuem pesos diferentes para cada variável em relação aos grupos, apresentam melhor desempenho para conjuntos de dados com altos níveis de dispersão e sobreposição de classes. Os resultados ampliam as capacidades dos algoritmos de agrupamento existentes baseados em distâncias adaptativas.

Palavras-chaves: agrupamento; aprendizado não supervisionado; distâncias adaptativas; dados qualitativos.

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

n	Number of objects
p	Number of variables
c	Number of clusters
Ω	Dataset
ω	Object
$\boldsymbol{x}_i$	Vector that describes i-th object
x_{ij}	Describes i-th object in the j-th variable
A_j	Describes j-th variable
n_{A_j}	Number of categories in the j-th qualitative variable
c_{tj}	Describes a certain t category in the j-th qualitative variable
$\boldsymbol{P}$	Proximity matrix
Q	Hard partition
J	Objective function
$\boldsymbol{v}_i$	Centroid of the i-th cluster
$\boldsymbol{U}$	Matrix of membership degrees
u_{ik}	Membership degree of k-th object in the i-th cluster
m	Fuzziness parameter
$\boldsymbol{\lambda}$	Matrix of relevance weights
λ_{ij}	Relevance weight of the j-th variable in the i-th cluster
β	Influence parameter
$\emptyset$	Empty set
∂	Partial derivative

ρ	Lagrange multiplier
∇	Describes the gradient
T	Maximum number of iterations
t^*	Counter of iterations
ϵ	Minimum improvement in the objective function between two consecutive iterations

CONTENTS

1 INTRODUCTION

1.1 MOTIVATION

The Machine Learning field is divided into two parts: supervised learning and unsupervised learning, as described in (JAMES et al., 2013). Within the sub-division of supervised learning, we have the following scenario: for each observation of the explanatory variables, there is an associated response variable. We want to adjust a model that relates the response to the predictors, with the objective of predicting, with some accuracy, the response for future observations or a better understanding of the relationship between the response and the predictors. In the sub-division of unsupervised learning, we have a more challenging situation, wherein for every observation, we observe a vector of values, but which is not associated with an answer. As such, we are blindfolded when dealing with problems of this nature. This situation is called unsupervised because in which we lack a response variable capable of supervising our analysis. In this work, we are interested in the clustering methods within the unsupervised learning sub-division.

Clustering is a technique for finding groups in a dataset, where the observations in each group are similar to each other and different from those in other groups (JAMES et al., 2013). When we cluster the observations of a dataset, we seek to partition them into distinct groups so that the observations within each group are quite similar to each other, while observations in different groups are quite different from each other.

Most of the existing clustering methods were developed specifically to deal with objects characterized by variables of a quantitative nature. One of the best known and studied in the literature is the K-Means method (MACQUEEN, 1965; HARTIGAN; WONG, 1979), by using the concept of arithmetic mean to group objects. In short, the clusters to be created are represented by the means of the variables that quantify the objects belonging to each cluster. However, The K-Means algorithm cannot be used in the presence of qualitative variables, since its clusters are based on the mean of the analyzed variables. Thus, there is a need for methods to deal with the cases of objects that are measured by qualitative variables, such as gender, education level, social class, etc.

A first effort to get around the K-Means limitation of dealing with qualitative variables, was carried out in (RALAMBONDRAINY, 1995). The idea was to transform the qualitative variables into multiple binary variables, using 0 and 1 to represent the absence or presence of a category, respectively (NETTER; WASSERMAN; KUTNER, 1989). And, after this modification, apply K-Means normally. Yet, in a context of a large number of variables with different categories, it becomes inevitable an increase in computational cost. Furthermore, the new means computed would be given by a real value between 0 and 1, not indicating the original characteristics of the objects.

Thereafter, a new clustering method called K-Modes was proposed to deal with qualitative variables (HUANG, 1998). This method extends K-Means, by using modes instead of means to represent the clusters. Also, the distance measures used in the objective function had to be adapted. A generalization of the K-Modes method was proposed in (HUANG; NG, 1999), based on fuzzy sets (ZADEH, 1965), to create the Fuzzy C-Modes method where there are membership degrees to determine the association of an object to a given cluster.

One limitation of these clustering algorithms is that they assign equal importance to all variables when determining an object's cluster membership. This can be problematic in certain situations, such as when clustering high-dimensional, sparse data where the cluster structure is typically explained by a specific subset of variables instead of the entire set, and clusters with different forms. A more suitable approach is to incorporate appropriate variable weights into the clustering process (TSAI; CHIU, 2008). Subspace clustering techniques aim to find clusters in specific combinations of dimensions. In addition, the significance of each variable in relation to each cluster can vary, meaning that each cluster may have a unique set of important variables.

Thereby, adaptive distances are based on a weight vector for each cluster in a way that the variables' importance is treated to achieve better partitions (KELLER; KLAWONN, 2000; CHAN et al., 2004; FRIGUI; NASRAOUI, 2004). Moreover, the weight expressions have been derived based on two types of constraints. First, it has been stipulated that the sum of the variable weights must equal one. Second, it has been required that the product of the variable weights must equal one.

Determining methods such as K-Modes and Fuzzy C-Modes with variable weighting aims to improve the quality of unsupervised learning techniques, particularly in scenarios with qualitative variables and the need for different variable importance within clusters.

1.2 OBJECTIVE

The main aim of this work is to develop a Fuzzy C-Modes algorithm with variable weighting, wherein the distance measure has an adaptive component to each variable and cluster, with different constraints.

More specifically, this work aims to:

- Propose a new algorithm for qualitative data partitional clustering;
- Consider different adaptive distance measures to calculate the dissimilarity in the objective function;
- Evaluate, based on synthetic and real datasets, the performance between hard and fuzzy methods using adaptive distance measures and other presented clustering methods through literature;
- Implement the proposed methods as a library in the R programming language to be made available to the community in open-source format.

1.3 STRUCTURE OF THE DOCUMENT

This dissertation comprises this introductory chapter and four more chapters. In Chapter 2, basic concepts about qualitative data and partitional clustering are presented. Chapter 3 the contribution of this work is shown: the proposal of a new Fuzzy C-Modes algorithm based on adaptive distances in the objective function. In Chapter 4, a set of experiments performed with both synthetic and real datasets to evaluate the proposed method are presented. Finally, in Chapter 5, the contributions about method proposal and future works are provided.

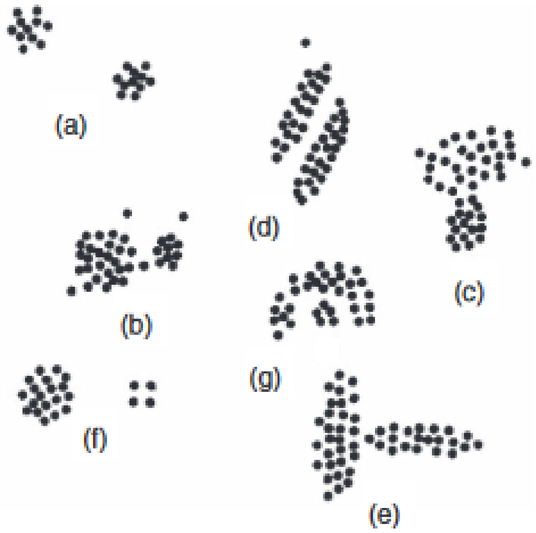
2 OVERVIEW OF THE PROBLEM

In this chapter, the basic concepts of clustering algorithms will be introduced. Then, there will be a description of the types of existing variables and the measures of proximity most used in the literature, for each type of variable presented. Finally, the main types of methods in clustering are listed, headlining the clustering methods of the partitional type.

2.1 BASIC CONCEPTS

Clustering analysis is a type of Machine Learning algorithm whose purpose is to separate objects into clusters, based on the characteristics that these objects have. The basic idea consists of placing objects in the same cluster that is similar according to some predetermined criteria. The clusters obtained must present low internal variance and high external variance (DEBORAH; BASKARAN; KANNAN, 2010). This means that objects of a given cluster must be mutually similar and, preferably, very different from the elements of other clusters. To illustrate the task of clustering objects into clusters, refer to the example of the wide variety of situations in Figure 1.

Figure 1 – Clustering problems



Source: (RUSPINI; BEZDEK; KELLER, 2019)

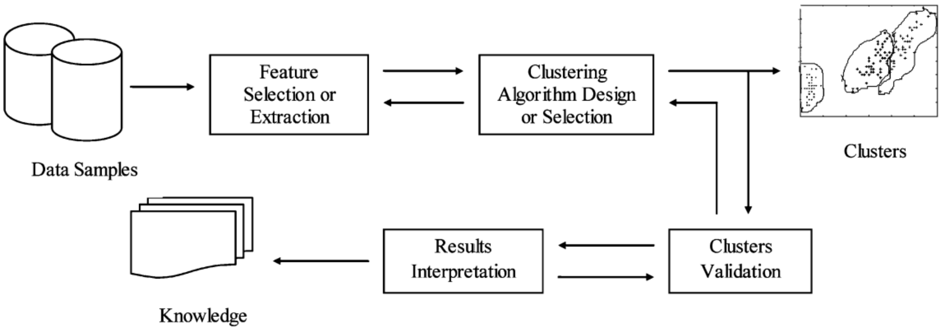
For some datasets such as (a) and (f), it is natural to think of the formation of two clusters. However, for datasets (c) and (g), e.g., it becomes harder thinking to determine a linear separation without overlapping the formation of two or more clusters. Regardless of the ease of naturally identifying possible groups, all the datasets are in a scenario in which there's no pre-existing knowledge about the real classes defined for the objects. With that in mind, there is a need to adopt a similarity criterion and then establish which objects are similar and which ones have little similarity, placing these in different clusters and those in the same cluster. In this way, clustering is a task prior to classification, as there is no knowledge a priori of classes to allocate the studied objects.

The clustering allows, then, to determine the clusters existing in a set of objects. Attention is drawn to the fact that the choice of the total number of clusters is subjective, and it is up to the experimenter to determine it in advance. With these available clusters, it is possible to analyze the objects that compose them, identifying the common characteristics of their respective objects. Thus, one can create a label representing them. With the existence of labels, when receiving a new object, which belongs to the considered universe, it is possible to allocate it correctly.

2.2 MAIN PHASES IN CLUSTERING ALGORITHMS

The basic structure of a clustering analysis can be represented in four steps, as shown in Figure 2, obtained from (XU; WUNSCH, 2005). It should be noted that these phases are not independent of each other. Sometimes, it will be necessary to go back to previous steps to correct and improve later phases.

Figure 2 – Clustering procedure


Data Samples
Feature Selection or Extraction
Clustering Algorithm Design or Selection
Clusters
Clusters Validation
Results Interpretation
Knowledge

Source: (XU; WUNSCH, 2005)

The phases are described below:

- **Variable selection or extraction**: In the selection step, variables are defined to be used for the clustering, while the extraction step uses transformations to generate useful and new variables from the originals (JAIN; DUBES, 1988). The chosen set of variables should describe the similarity between the objects, in terms relevant to the researched problem. This phase optimizes the computational processing time of the chosen method since the search space is reduced (HORE; HALL; GOLDGOF, 2007). This directly affects the performance of the next step. Furthermore, a good choice or selection of variables directly interferes with the quality of the formed clusters;

- **Development or selection of a clustering algorithm**: This phase is usually combined with the choice of a proximity measure and the definition of an objective function. The proximity measure will quantify how close an object is to another. Different proximity measures are found in the literature and basically depend on the type of variables (quantitative or qualitative) involved in the study. Having chosen the proximity measure, the construction of an objective function makes obtaining the clusters an optimization problem, which is well-defined mathematically. The algorithm will specify, in general, how these functions will be optimized, given the choice of a proximity measure appropriate to the type of the variable used;

- **Clusters validation**: Given a dataset, every clustering algorithm can generate a cluster, no matter what the theoretical or practical sense of it. In addition, there must be criteria to assess the reliability of the results offered by the algorithms. This evaluation can be based on external validation indexes and internal. An external index is used to compare the cluster structure obtained by clustering with an a priori defined structure and an internal index determines whether the grouping structure is appropriate to the data (SOUZA; CARVALHO, 2004);

- **Results interpretation**: In this phase, it is verified how to use the results obtained in the clustering algorithm, so that the problems raised at the beginning of the research are resolved. The need for further analysis and experiments may arise to ensure the reliability of the extracted knowledge.

The steps described form a very useful methodological procedure for carrying out clustering analysis. However, (MANLY; ALBERTO, 2016) warns that there is still no universal and effective for selecting or extracting variables, as well as for choosing the algorithm to be employed. The author also points out that the evaluation indices provide information important about the quality of the generated clusters, but how to choose this same criterion is still a problem that requires more effort. An important consideration to be made is that the choice of clustering algorithm should be based, essentially, on the type of variable related to the objects under study. Many different types of variables require different types of methods to be employed.

2.3 NOTATION

In the following sections and chapters of this work, we assume that the set $\Omega = \{w_1, \ldots, w_n\}$ of n objects are grouped into c clusters. These objects are characterized by a set $\{A_1, \ldots, A_p\}$ of p variables. So the variable A can be understood as a function that associates the result of realization of the observed property $A(w) = x$ with each object $w \in \Omega$.

Each object w_i $(i = 1, \ldots, n)$ will be represented by a variable vector $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})$, where x_{ij} is the result of the realization of the variable j observed in the object w_i. In the case of the set $\{A_1, \ldots, A_p\}$ being formed by variables of the quantitative type, the described elements will be called quantitative objects. Already if $\{A_1, \ldots, A_p\}$ is formed by qualitative variables, the elements described by them will be called qualitative objects. In the possibility of $\{A_1, \ldots, A_p\}$ having both types of variables, the objects will be called mixed objects.

The set of possible results of the realizations of a variable A_j is called the domain of the variable A_j, which will be denoted by DOM(A_j), with $j \in \{1, \ldots, p\}$. If $A_1, \ldots, A_j$ are quantitative variables, DOM$(A_j) = \mathbb{R}$. In the case of qualitative objects, we have that the cardinality of this set, $|$DOM$(A_j)| = n_{A_j}$, will represent the number of possible categories of the j-th qualitative variable. When a certain category t of A_j needs to be specified, it will be specified as c_{tj}, with $t \in$ DOM(A_j). Every object is considered to have exactly p variables.

2.4 VARIABLE TYPES

As mentioned in the previous section, a variable A_j can be understood as a vector of variables of interest, which measures each object in the sample or population. Based on DOM(A_j), the variables are classified into two large groups: variables qualitative and quantitative variables (MORETTIN; BUSSAB, 2017). A variable is qualitative, if DOM(A_j) is a finite set and the elements of this set are categories. A variable is quantitative, if your domain is the set of real numbers $\mathbb{R}$, or DOM$(A_j) \subseteq \mathbb{R}$. The variables qualitative and quantitative are subdivided as follows:

1. Qualitative;
    - Ordinal;
    - Nominal.
2. Quantitative.
    - Discrete;
    - Continuous.

2.4.1 Qualitative variables

The qualitative variables have their domain DOM(A_j) finite and without a numeric meaning. They represent a classification of objects. In the nominal type, there are no sorts in the domain. For instance, gender, ethnicity, and political affiliation are examples of nominal variables. When the domain of these variables has only two categories, which are usually encoded as 0 or 1, i.e. DOM$(A_j) = \{0, 1\}$, these variables are called binary. An example of a binary variable is whether a person has a specific medical condition or not. A variable A_j is qualitative ordinal if its domain is finite and for each pair of objects $x_{ij}, x_{kj} \in$ DOM(A_j) there is a linear order between them, that is, $x_{ij} < x_{kj}$ or $x_{kj} < x_{ij}$. For example, education level is an ordinal variable because it can be ranked from less to more education, such as high school, some college, Bachelor's degree, Master's degree, and Doctorate degree. Other examples of ordinal variables include socioeconomic status and stages of disease progression.

2.4.2 Quantitative variables

Quantitative variables are variables that can be measured on a quantitative scale, that is, they present numerical values that make sense. As shown, they can be discrete or continuous. In the discrete case, DOM(A_j) is a finite or infinite enumerable set of values (JAMES, 2015). It is usually the result of counts event. In the continuous case, the domain is given by values on a continuous scale in $\mathbb{R}$. Usually, they must be measured through some instrument. It is worth noting that a variable, originally quantitative, can be collected in a qualitative manner. For example, the variable age measured in complete years is a quantitative continuous variable.

However, if only the age group is informed (0 to 5 years old, 6 to 10 years old, etc.), it is a qualitative ordinal variable.

(JAMES et al., 2013) draws attention to the fact that it is necessary to standardize the quantitative variables, aiming to mitigate the effects of different scale measures. Such an effect can compromise the final result of the clustering. The chosen distance function can also help mitigate those effects, e.g., the Mahalanobis distance (MAESSCHALCK; JOUAN-RIMBAUD; MASSART, 2000).

2.5 SIMILARITY MEASURES

The use of an appropriate measure relates to the type of object being analyzed. Therefore, the choice of similarity measure should be based on the type of domain of the variables A_j. By using these measures, the criteria are established that define whether two objects $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$ are close, or not. Based on this, objects are allocated to the same cluster or to different clusters.

These measures can be divided into two types: similarity and dissimilarity measures. (GOSHTASBY, 2012) defines that a similarity measure s is considered a metric if it produces a larger value as the dependency between the corresponding values of the observations increases. A dissimilarity measure d is a metric if it produces a smaller value as the dependency between the values corresponding observations decreases. These values we refer to can be the grey level value of the pixels of the compared images. As the price of two real estates located in a city. As an example of a similarity measure, we have the Pearson correlation coefficient (MORETTIN; BUSSAB, 2017), and as a dissimilarity measure, the square of the Euclidean distance (GOSHTASBY, 2012). Notice that a dissimilarity measure can be written as $d = (1 - s)$.

Thus, a similarity s is a function $s : \Omega \times \Omega \to \mathbb{R}^+$, which satisfies the following properties $\forall \boldsymbol{x}_i, \boldsymbol{x}_k \in \Omega \, (i, k = 1, \dots, n)$:

1. $\forall \boldsymbol{x}_i, \boldsymbol{x}_k \in \Omega,\ s(\boldsymbol{x}_i, \boldsymbol{x}_k) \geq 0$;
2. $\forall \boldsymbol{x}_i \in \Omega,\ s(\boldsymbol{x}_i, \boldsymbol{x}_i) \geq max_{\boldsymbol{x}_k} s(\boldsymbol{x}_i, \boldsymbol{x}_k)$;
3. $\forall (\boldsymbol{x}_i, \boldsymbol{x}_k) \in \Omega \times \Omega : s(\boldsymbol{x}_i, \boldsymbol{x}_k) = s(\boldsymbol{x}_k, \boldsymbol{x}_i)$.

A dissimilarity d is a function $d : \Omega \times \Omega \to \mathbb{R}^+$, which satisfies the following properties $\forall \boldsymbol{x}_i, \boldsymbol{x}_k, \boldsymbol{x}_r \in \Omega \, (i, k, r = 1, \dots, n)$:

1. $\forall \boldsymbol{x}_i, \boldsymbol{x}_k \in \Omega,\ d(\boldsymbol{x}_i, \boldsymbol{x}_k) \geq 0$;

2. $\forall \boldsymbol{x}_i \in \Omega$, $d(\boldsymbol{x}_i, \boldsymbol{x}_i) = 0$;

3. $\forall (\boldsymbol{x}_i, \boldsymbol{x}_k) \in \Omega \times \Omega : d(\boldsymbol{x}_i, \boldsymbol{x}_k) = d(\boldsymbol{x}_k, \boldsymbol{x}_i)$.

In addition, a distance f is a dissimilarity function, which also satisfies properties 1. to 3. and, additionally, the property of triangular inequality (LIMA, 2015). Such property guarantees that the distance from an object $\boldsymbol{x}_i$ to an object $\boldsymbol{x}_k$ will never exceed the sum of the distances of these objects to another object $\boldsymbol{x}_r$, i.e.,

4. $\forall (\boldsymbol{x}_i, \boldsymbol{x}_k) \in \Omega \times \Omega : f(\boldsymbol{x}_i, \boldsymbol{x}_k) \leq f(\boldsymbol{x}_i, \boldsymbol{x}_r) + f(\boldsymbol{x}_k, \boldsymbol{x}_r)$.

There are several similarity measures proposed in the literature. As reported, the similarity/dissimilarity between pairs of objects is calculated depending on the type of variable that describes the objects. Next, we list some measures for each type of variable and their characteristics.

2.5.1 Measures for quantitative variables

Most of the measures proposed in the literature are for quantitative variables. It should be noted that in this work, only the three most used will be mentioned. For more details on other measures, it is recommended to consult (XU; WUNSCH, 2005).

The next presented measurements will be defined as distances, which meet properties 1. to 4. Therefore, the distance $f(i, k)$ between the quantitative objects w_i and w_k will be constructed from values of p variables, transformed into vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$.

The most used distances for quantitative objects are:

- Euclidian distance

$$f(i, k) = \sqrt{(x_{i1} - x_{k1})^2 + \cdots + (x_{im} - x_{km})^2}. \tag{2.1}$$

- Manhattan distance

$$f(i, k) = |x_{i1} - x_{k1}| + \cdots + |x_{im} - x_{km}|. \tag{2.2}$$

- Minkowski distance

$$f(i, k) = \sqrt[\lambda]{(|x_{i1} - x_{k1}|)^\lambda + \cdots + (|x_{im} - x_{km}|)^\lambda}. \tag{2.3}$$

The last distance generalizes the first two when $\lambda = \{1, 2\}$. Also, the larger the value of λ, the greater the weight is given to very different (dissimilar) observations.

2.5.2 Measures for qualitative variables

Next, the main measures for the qualitative variables are described. The qualitative binary variable, the particular case of the nominal when we have $|\mathsf{DOM}(A_j)| = 2$, has a different evaluation approach from the general nominal case for reasons of interpretability of the event that was observed by this variable. In general, the comparison between binary objects is more interested in understanding whether or not there was an agreement between them, rather than a simple equality.

2.5.2.1 Binary variables

Binary variables can take only two different values, as mentioned. Then let be the pair of qualitative objects (w_i, w_k), each described by a vector of p binary variables, where $x_{ij} = x_{kj} = 0$ indicates absence for variable j and $x_{ij} = x_{kj} = 1$ indicates presence.

The most used proximity measures between qualitative objects are similarities s, which are generally based on the following quantities:

a: number of variables A_j, $1 \leq j \leq p$, for which both objects (w_i, w_k) assume the value 1;

b: number of variables A_j, $1 \leq j \leq p$, for which the object w_i assumes the value 1 and the object w_k assumes the value 0;

c: number of variables A_j, $1 \leq j \leq p$, for which the object w_i assumes the value 0 and the object w_k assumes the value 1;

d: number of variables A_j, $1 \leq j \leq p$, for which both objects (w_i, w_k) assume the value 0.

Consider the following similarities:

1. Sokal-Michener correspondence coefficient

$$s(i,k) = \frac{a+d}{a+b+c+d}, \quad 0 \leq s(i,k) \leq 1. \tag{2.4}$$

   This similarity represents the proportion of variables A_j in which there is an agreement in the values of individuals i and k (SOKAL; MICHENER, 1975).

2. Jaccard coefficient

$$s(i,k) = \frac{a}{a+b+c}, \quad 0 \leq s(i,k) \leq 1. \tag{2.5}$$

This similarity represents the number of variables A_j, in which both objects have a present value, in relation to the number of variables A_j, where at least one of the objects have a present value (JACCARD, 1901).

3. Gower-Lengendre coefficient

$$s(i,k) = \frac{(a+d)-(b+c)}{a+b+c+d}, \quad -1 \leq s(i,k) \leq 1. \tag{2.6}$$

This similarity makes the difference between agreements and disagreements relative to the total p of observed variables. Unlike 2.4 and 2.5, it can take values negative, situation to occur, if there are more disagreements than agreements, in the variable values for qualitative objects w_i and w_k (GOWER; LEGENDRE, 1986).

The similarity measures s can be easily converted into d dissimilarities measures by computing $d(i,k) = 1 - s(i,k)$.

2.5.2.2 *Nominal variables*

We now have the pair of objects (w_i, w_k) described by a vector of p nominal variables. In general, the measure of dissimilarity between w_i and w_k is understood as the total of disagreements that occurred, when analyzing these two objects, with respect to the p variables (KAUFMAN; ROUSSEEUW, 2009). How much the smaller the number of dislocations, the greater the similarity between the two objects.

Consider,

$$d(i,k) = \sum_{j=1}^{p} \phi(x_{ij}, x_{kj}), \tag{2.7}$$

where

$$\phi(x_{ij}, x_{kj}) = \begin{cases} 0, & \text{if } x_{ij} = x_{kj} \\ \\ 1, & \text{if } x_{ij} \neq x_{kj} \end{cases} .$$

2.5.2.3 *Ordinal variables*

When the qualitative variables are of the ordinal type, the proximity between objects w_i and w_k can be calculated in a very similar way to the calculation used for quantitative objects. The procedure consists of the following steps:

1. For each variable j $(j = 1, \ldots, p)$ its possible categories are enumerated c_{tj}, according to the order existing between them. Let $\{1, \ldots, n_{A_j}\}$ an enumerated list of the categories, for which n_{A_j} is the total number of categories of variable j. For the objects w_i and w_k, each category x_{ij}, x_{kj} is replaced in their respective order r_{ij}, r_{kj} with r_{ij}, $r_{kj} \in \{1, \ldots, n_{A_j}\}$.

2. Since each variable has a different number of categories, a normalization of the data is necessary, and this can be accomplished by doing

$$z_{ij} = \frac{r_{ij} - 1}{n_{A_j} - 1}, \quad \text{with } i = 1, \ldots, n.$$

3. Finally, the dissimilarity between objects w_i and w_k can be computed using the distances 2.1, 2.2 or 2.3 applied to the normalized data vectors $\boldsymbol{z}_i$ and $\boldsymbol{z}_k$.

2.6 PARTITIONAL CLUSTERING OVERVIEW

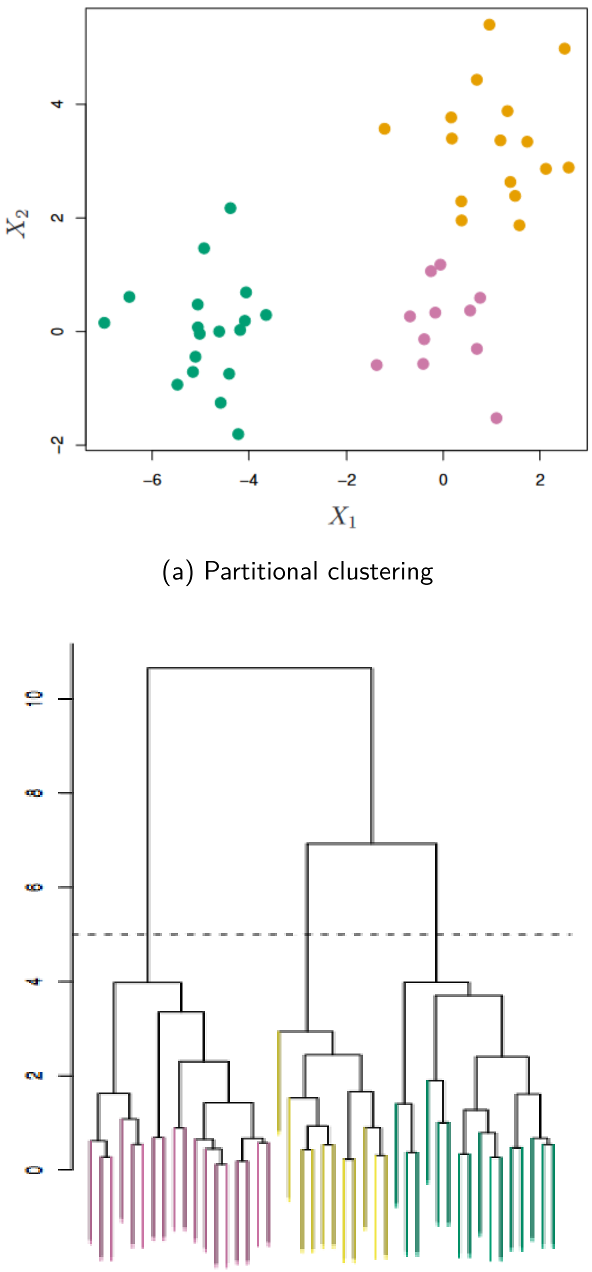
There are proposals in the literature for different methods, with different forms of classification. This fact is related to the number of existing algorithms and techniques, as well as the different possibilities of application. However, the most concerted division, which will be adopted in this work, is to divide the methods in relation to the strategy adopted for the definition of clusters. In this way, the methods are organized into two major groups: partitional and hierarchical (JAMES et al., 2013). In this work, we're focused on the partitional methods. The partitional methods aim to directly decompose objects into a set of disjoint or overlapping clusters, obtaining a partition, which optimizes an objective function. Figure 3 illustrates the difference between the results obtained from the different types of methods.

Let a data set $\Omega = \{w_1, \ldots, w_n\}$. The partitional clustering methods aim to partition this set into a predefined number of c clusters, where $c \leq n$. Usually, these clusters are constituted, through the optimization of some objective function. There are two types of partitional methods: hard and fuzzy ones. In hard clustering, it is assumed that clusters must form a partition Q of the set Ω. This partition can be understood as a family of distinct and non-empty subsets Ω_i, for $i = 1, \ldots, c$, of Ω. In this way, objects that belong to the same cluster as the partition are fully related and objects that belong to distinct clusters are not related, such that:

1. $\Omega_i \neq \emptyset,\ i = 1, \ldots, c$;

2. $\cup_{i=1}^{c} \Omega_i = \Omega$;

Figure 3 – An illustration of clustering approaches



(a) Partitional clustering

(b) Hierarchical clustering

Source: (JAMES et al., 2013)

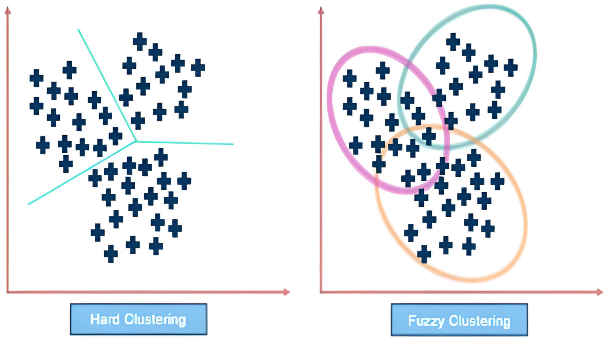
3. $\Omega_i \cap \Omega_j = \emptyset,\ i, j = 1, \ldots, c$ and $i \neq j$.

Thus, in the hard approach, clusters are discrete entities, characterized by a set of properties shared by their members. These clusters are clearly defined, mutually exclusive, and collectively exhaustive. In this way, any object must exclusively belong to one, and only one, of the proposed clusters. In many practical applications, however, a hard partition can be too restrictive, because the same object can share variables present in more than one cluster.

In that case, fuzzy partitional methods extend the notion of hard clustering, allowing to associate an object with all clusters, using a membership degree, $u_{ik} \in [0, 1]$, which represents the membership coefficient of the k-th object in the i-th cluster by satisfying $\sum_{i=1}^{c} u_{ik} = 1 \forall k$ and $\sum_{k=1}^{n} u_{ik} < n$ (BEZDEK, 1981). With the use of the fuzzy approach, the problem is then characterized as a fuzzy clustering problem, whose objective is to obtain a fuzzy partition, of an Ω data set.

Figure 4 illustrates the idea behind fuzzy clustering. The lines separate the dataset into three hard clusters, while a fuzzy partitional algorithm could create three fuzzy clusters represented by ellipses. Thus, the objects will have a membership degree u_{ik} in the interval $[0, 1]$ for each cluster. Note that the image does not faithfully characterize a fuzzy clustering, since all objects belong to all clusters at the same time, with different degrees of membership. High membership values indicate a high degree of association of an object to a cluster. It is worth mentioning that a hard partition can be obtained from a fuzzy partition, by applying a threshold on the membership values.

Figure 4 – Hard clusters versus fuzzy clusters



Source: (KAMOLOV; PARK, 2021)

Among the partitional clustering algorithms, there is a specific field with methods based on the minimization of the square-error criterion, the sum of squared Euclidean distances (Equation 2.1) of points from their closest cluster centroid, is the most commonly used (SISODIA et al., 2012). The most known methods in this scenario are K-Means (MACQUEEN, 1965), K-Medoids or Partitioning Around Medoids (PAM) (KAUFMAN; ROUSSEEUW, 2009), K-Modes (HUANG, 1998), and Clustering Large Applications based on Randomized Search (CLARANS) (HE; XU; DENG, 2005). In this work, as there is a focus on building algorithms that deal with qualitative data, the K-Modes provides an extension of the mathematical problem deemed in the K-Means algorithm for such type of data. After the initial versions, adaptations for fuzzy algorithms were proposed: Fuzzy C-Means (DUNN, 1973) and Fuzzy C-Modes (HUANG; NG, 1999).

Another adaptation proposed by the literature was the insertion of adaptive distances which change at each algorithm iteration and can either be the same for all clusters (global adaptive distances) or different from one cluster to another (local adaptive distances). This kind of dissimilarity measure is suitable to learn the weights of the variables during the clustering process, improving the performance of the algorithms (DIDAY; SIMON, 1976; DIDAY, 1977; FERREIRA; CARVALHO, 2014).

In the next chapter, a set of algorithms based on qualitative data are proposed with the insertion of adaptive distances as an extension of the K-Modes and Fuzzy C-Modes algorithms.

3 FUZZY C-MODES CLUSTERING ALGORITHM WITH VARIABLE WEIGHTING

This chapter provides a description of the proposed Fuzzy C-Modes clustering algorithm with variable weighting.

3.1 OBJECTIVE FUNCTION OVERVIEW

The Fuzzy C-Modes Clustering algorithm (HUANG; NG, 1999) is defined from the minimization of the following objective function

$$J = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^{m} \Psi(\boldsymbol{x}_k, \boldsymbol{v}_i), \tag{3.1}$$

subject to

$$\begin{cases} u_{ik} \in [0, 1] \quad \forall i, k \\ \sum_{i=1}^{c} u_{ik} = 1 \quad \forall k, \end{cases}$$

where u_{ik} are the fuzzy membership degrees of observation k and cluster i, $\boldsymbol{v}_i$ is the centroid of cluster i and $m \in (1, \infty)$ is a parameter that controls the fuzziness of membership for each k. In addition, $\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)$ is the appropriate distance function for a given qualitative dataset. In the simplest scenario, one may assume that $\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) = \sum_{j=1}^{p} \psi(x_{kj}, v_{ij})$ (KAUFMAN; ROUSSEEUW, 2009), where

$$\psi(x_{kj}, v_{ij}) = \begin{cases} 0, & \text{if } x_{kj} = v_{ij} \\ 1, & \text{otherwise.} \end{cases}$$

In this scenario with qualitative variables, it is necessary to adapt the way the similarity (or dissimilarity) measures are calculated among the observations and centroids. In addition to considering measures that have structure for qualitative variables, it is also necessary to direct them to the types: binary, nominal, and ordinal. The proposed algorithm considers the mode as the statistic used to determine the centroids for each cluster (HUANG; NG, 1999). The definition for calculating the mode is following.

Definition 3.1. *The mode for a set of k observations measured by p qualitative variables is an object v which minimizes*

$$\sum_{k=1}^{n} d(x_{kj}, v), \ \textit{for a given } j.$$

Notice that v might not be unique, because two or more categories of a variable can have the same frequency. To calculate the mode described in Definition 3.1, it's considered the following proposition, considering initially a hard clustering.

Proposition 3.1. *Let Ω be a set of n objects characterized by the qualitative variables $A_1, A_2, \ldots, A_p$ and DOM(A_j) $= \{a_j^1, a_j^2, \ldots, a_j^{n_j}\}$, where n_j is the number of categories on variable A_j, for $1 \leq j \leq p$. It's calculated the modes from the clusters represented by $\boldsymbol{v}_i = (v_{i1}, v_{i2}, \ldots, v_{ip})$ for $1 \leq i \leq c$. Then, the quantity*

$$\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)$$

is minimized if and only if $v_{ij} = a_j^{(r)} \in$ DOM(A_j), where $u_{ik} = \{0, 1\}$ and $a_j^{(r)}$ is the category of variable j with greater frequency. Also,

$$|\{u_{ik}|x_{kj} = a_j^{(r)}, u_{ik} = 1\}| \geq |\{u_{ik}|x_{kj} = a_j^{(t)}, u_{ik} = 1\}|$$
$$1 \leq t \leq n_j$$
$$1 \leq j \leq p.$$

Here, $|\boldsymbol{x}|$ denotes the number of elements in set Ω.

Proof. For a given matrix $\boldsymbol{U}$, all the inner sums of

$$\sum_{i=1}^{c}\left[\sum_{k}^{n} u_{ik}\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)\right]$$

are independent and non-negative. Minimizing the quantity above is equivalent to minimizing each inner sum. Thus, write the i-th inner sum as

$$\begin{aligned}\sum_{k=1}^{n} u_{ik}\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) &= \sum_{k=1}^{n} u_{ik}\sum_{j=1}^{p}\psi(x_{kj}, v_{ij})\\ &= \sum_{j=1}^{p}\sum_{k=1}^{n} u_{ik}\psi(x_{kj}, v_{ij})\\ &= \sum_{j=1}^{p}(n - |\{u_{ik}|x_{kj} = v_{ij}, u_{ik} = 1\}|)\\ &= \sum_{j=1}^{p} n\left(1 - \frac{|\{u_{ik}|x_{kj} = v_{ij}, u_{ik} = 1\}|}{n}\right).\end{aligned}$$

The inner sum is minimized if and only if each term $\left(1 - \frac{|\{u_{ik}|x_{kj}=v_{ij}, u_{ik}=1\}|}{n}\right)$ is minimum for $1 \leq j \leq p$. Therefore, the term $|\{u_{ik}|x_{kj} = v_{ij}, u_{ik} = 1\}|$ must be maximum. □

Therefore, by Definition 3.1 and Proposition 3.1, the vector of modes $\boldsymbol{v}_i$ is defined by the higher frequency categories $a_j^{(r)}, j \in [1, p]$, on the set of observations belonging to cluster i. For the fuzzy version the constraint on u_{ik} is slacked such as $0 \leq u_{ik} \leq 1$. With that in mind, the definition of the u_{ik} is required besides the mode in the scenario where each observation can belong to all clusters at the same instant.

Proposition 3.2. *For $m > 1$ and fixed centroids $\boldsymbol{v}_i$, we have that*

$$u_{ik} = \left\{ \sum_{h=1}^{c} \left[\frac{\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)}{\Psi(\boldsymbol{x}_k, \boldsymbol{v}_h)} \right]^{1/(m-1)} \right\}^{-1}.$$

Proof. From the Lagrange multipliers, (STEWART, 2012), to find an optimum value, consider

$$J(\rho) = \sum_{i=1}^{c} u_{ik}^{m} \Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) - \rho(\sum_{i=1}^{c} u_{ik} - 1),$$

that is stationary when the gradient is equal to 0, that is $\nabla J(\rho) = \left[\frac{\partial}{\partial \rho} J(\rho), \frac{\partial}{\partial u_{ik}} J(\rho)\right] = 0.$ Thus, there's the following system of equations:

$$\begin{cases} \frac{\partial J(\rho)}{\partial \rho} = \sum_{i=1}^{c} u_{ik} - 1 = 0 \quad \text{(I)} \\ \frac{\partial J(\rho)}{\partial u_{st}} = m(u_{st})^{m-1} \Psi(\boldsymbol{x}_t, \boldsymbol{v}_s) - \rho = 0. \quad \text{(II)} \end{cases}$$

$$\begin{aligned} \text{(I)} \Rightarrow m(u_{st}^{m-1}) \Psi(\boldsymbol{x}_t, \boldsymbol{v}_s) - \rho =& 0 \\ u_{st}^{m-1} =& \frac{\rho}{m\Psi(\boldsymbol{x}_t, \boldsymbol{v}_s)} \\ u_{st} =& \left(\frac{\rho}{m\Psi(\boldsymbol{x}_t, \boldsymbol{v}_s)} \right)^{\frac{1}{(m-1)}}. \quad \text{(III)} \end{aligned}$$

$$\begin{aligned} \text{(I) and (III)} \Rightarrow \sum_{h=1}^{c} u_{ht} =& \sum_{h=1}^{c} \left(\frac{\rho}{m\Psi(\boldsymbol{x}_t, \boldsymbol{v}_h)} \right)^{\frac{1}{(m-1)}} \\ \Rightarrow& \left(\frac{\rho}{m} \right)^{\frac{1}{(m-1)}} \sum_{h=1}^{c} \left(\frac{1}{m\Psi(\boldsymbol{x}_t, \boldsymbol{v}_h)} \right)^{\frac{1}{(m-1)}} = 1 \\ \left(\frac{\rho}{m} \right)^{\frac{1}{(m-1)}} =& \frac{1}{\sum_{h=1}^{c} \left(\frac{1}{m\Psi(\boldsymbol{x}_t, \boldsymbol{v}_h)} \right)^{\frac{1}{(m-1)}}}. \end{aligned}$$

Backing to (III),

$$\begin{aligned}
u_{st} =& \left(\frac{\rho}{m}\right)^{\frac{1}{(m-1)}} \left(\frac{1}{m\Psi(\boldsymbol{x}_t, \boldsymbol{v}_s)}\right)^{\frac{1}{(m-1)}} \\
=& \frac{1}{\sum_{h=1}^{c} \left(\frac{1}{m\Psi(\boldsymbol{x}_t,\boldsymbol{v}_h)}\right)^{\frac{1}{(m-1)}}} \left(\frac{1}{m\Psi(\boldsymbol{x}_t, \boldsymbol{v}_s)}\right)^{\frac{1}{(m-1)}} \\
=& \left\{ \sum_{h=1}^{c} \left[\frac{\Psi(\boldsymbol{x}_k, \boldsymbol{v}_s)}{\Psi(\boldsymbol{x}_k, \boldsymbol{v}_h)}\right]^{1/(m-1)} \right\}^{-1}.
\end{aligned}$$

□

With the change of membership degrees in the fuzzy version, it is also necessary to define the mode calculation for the scenario in which each observation belongs to all clusters at the same time. Equation 3.3 shows how to get these modes.

Proposition 3.3. *Let Ω be a set of n objects characterized by the qualitative variables $A_1, A_2, \ldots, A_p$ and DOM(A_j) $= \{a_j^1, a_j^2, \ldots, a_j^{n_j}\}$, where n_j is the number of categories on variable A_j, for $1 \leq j \leq p$. It's calculated the modes from the clusters represented by $\boldsymbol{v}_i = (v_{i1}, v_{i2}, \ldots, v_{ip})$ for $1 \leq i \leq c$. Then, the quantity*

$$\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^m \Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)$$

is minimized if and only if $v_{ij} = a_j^{(r)} \in$ DOM(A_j), where $u_{ik} = [0, 1]$ and $a_j^{(r)}$ is the category of variable j with greater frequency. Also,

$$\sum_{\substack{i \\ x_{kj}=a_j^{(r)}}} u_{ik}^m \geq \sum_{\substack{i \\ x_{kj}=a_j^{(t)}}} u_{ik}^m$$

$$1 \leq t \leq n_j$$

$$1 \leq j \leq p.$$

Proof. For a given matrix $\boldsymbol{U}$, all the inner sums of

$$\sum_{i=1}^{c}\left[\sum_{k=1}^{n} u_{ik}^m \Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)\right]$$

are independent and non-negative. Minimizing the quantity above is equivalent to minimizing each inner sum. Thus, write the i-th inner sum as

$$\begin{aligned}\sum_{k=1}^{n} u_{ik}^{m}\Psi(\boldsymbol{x}_k,\boldsymbol{v}_i) &= \sum_{k=1}^{n} u_{ik}^{m}\sum_{j=1}^{p}\psi(x_{kj},v_{ij})\\ &= \sum_{j=1}^{p}\sum_{k=1}^{n} u_{ik}^{m}\psi(x_{kj},v_{ij})\\ &= \sum_{j=1}^{p}\left(\sum_{t=1}^{n_j}\sum_{\substack{i\\ x_{kj}=a_j^{(t)}}} u_{ik}^{m} - \sum_{\substack{i\\ x_{kj}=v_{ij}}} u_{ik}^{m}\right).\end{aligned}$$

Once u_{ik}^{m} is fixed and non-negative for $1 \leq i \leq c$ and $1 \leq k \leq n$, the quantity

$$\sum_{t=1}^{n_j}\sum_{\substack{i\\ x_{kj}=a_j^{(t)}}} u_{ik}^{m}$$

is fixed and non-negative. Thereby, it follows that $\sum_{k=1}^{n} u_{ik}^{m}\Psi(\boldsymbol{x}_k,\boldsymbol{v}_i)$ is minimized if and only if each term $\sum_{\substack{i\\ x_{kj}=v_{ij}}} u_{ik}^{m}$ is maximum. □

3.1.1 Adaptive distance functions

Conventional clustering methods do not take into account the weights and relevance of variables. That is, these methods consider that all variables are equally important for clustering in the sense that they all have the same relevance weight. However, in most applications we have to deal with, the available data sets have high dimensionality. Thus, some variables may be irrelevant and, among the relevant ones, some may be more or less relevant than others (FERREIRA; CARVALHO, 2014). Also, the relevance weight of each variable for each cluster may be different. As a result, each cluster can have a different set of relevant variables. If we consider there are differences in the relevance weights between the variables and calculate these weights, then the clustering performance may be improved.

For inserting the weights into the clustering algorithm, consider the following distances:

(a) Non-adaptive distance:

$$\Psi(\boldsymbol{x}_k,\boldsymbol{v}_i) = \sum_{j=1}^{p}\psi(x_{kj},v_{ij}). \tag{3.2}$$

(b) Local adaptive distance with the constraint that the sum of the weights of the variables for each cluster must be equal to one:

$$\Psi(\boldsymbol{x}_k,\boldsymbol{v}_i) = \sum_{j=1}^{p}(\lambda_{ij})^{\beta}\psi(x_{kj},v_{ij}), \tag{3.3}$$

where $\boldsymbol{\lambda}_i = (\lambda_{i1}, \ldots, \lambda_{ip})$ is the weights vector related to cluster i subject to

$$\begin{cases} \lambda_{ij} \in [0,1], & \forall i, j \\ \sum_{j=1}^{p} \lambda_{ij} = 1, & \forall i. \end{cases}$$

And $\beta \in (1, \infty)$ is the parameter that controls the degree of influence of the weight of each variable to each cluster:

- If $\beta \to \infty$ all the variables have the same influence on all clusters.
- If $\beta \to 1$ then the influence of the weights of the variables will be the highest.

(c) Global adaptive with the constraint that the sum of the weights of the variables must be equal to one:

$$\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) = \sum_{j=1}^{p} (\lambda_j)^{\beta} \psi(x_{kj}, v_{ij}), \tag{3.4}$$

where $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_p)$ is the weights vector subject to

$$\begin{cases} \lambda_j \in [0,1], & \forall j \\ \sum_{j=1}^{p} \lambda_j = 1. \end{cases}$$

(d) Local adaptive distance with the constraint that the product of the weights of the variables for each cluster must be equal to one:

$$\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) = \sum_{j=1}^{p} \lambda_{ij} \psi(x_{kj}, v_{ij}), \tag{3.5}$$

where $\boldsymbol{\lambda}_i = (\lambda_{i1}, \ldots, \lambda_{ip})$ is the weights vector related to cluster i subject to

$$\begin{cases} \lambda_{ij} > 0, & \forall i, j \\ \prod_{j=1}^{p} \lambda_{ij} = 1, & \forall i. \end{cases}$$

(e) Global adaptive with the constraint that the product of the weights of the variables must be equal to one:

$$\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) = \sum_{j=1}^{p} \lambda_j \psi(x_{kj}, v_{ij}), \tag{3.6}$$

where $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_p)$ is the weights vector subject to

$$\begin{cases} \lambda_j > 0, & \forall j \\ \prod_{j=1}^{p} \lambda_j = 1. \end{cases}$$

From this point forward, all propositions presented form an integral part of the development of this work. The Proposition 3.4 below is an extension of Proposition 3.3 when calculating the mode for the scenarios where $\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)$ can be defined as Equations 3.3, 3.4, 3.5 and 3.6.

Proposition 3.4. *Let Ω be a set of n objects characterized by the qualitative variables $A_1, A_2, \ldots, A_p$ and DOM$(A_j) = \{a_j^1, a_j^2, \ldots, a_j^{n_j}\}$, where n_j is the number of categories on variable A_j, for $1 \leq j \leq p$. It's calculated the modes from the clusters represented by $\boldsymbol{v}_i = (v_{i1}, v_{i2}, \ldots, v_{ip})$ for $1 \leq i \leq c$. Then, the quantity*

$$\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m}\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i),$$

where $\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)$ can be defined as Equations 3.3, 3.4, 3.5, and 3.6, is minimized if and only if $v_{ij} = a_j^{(r)} \in$ DOM(A_j), where

$$\sum_{\substack{i \\ x_{kj}=a_j^{(r)}}} u_{ik}^{m} \geq \sum_{\substack{i \\ x_{kj}=a_j^{(t)}}} u_{ik}^{m}$$

$$1 \leq t \leq n_j$$

$$1 \leq j \leq p.$$

Proof. For the given matrices $\boldsymbol{U}$ and $\boldsymbol{\lambda}$, all the inner sums of

$$\sum_{i=1}^{c}\left[\sum_{k=1}^{n} u_{ik}^{m}\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)\right]$$

are independent and non-negative. Minimizing the quantity above is equivalent to minimizing each inner sum. Thus, write the i-th inner sum as

$$\begin{aligned}
\sum_{k=1}^{n} u_{ik}^{m}\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) &= \sum_{k=1}^{n} u_{ik}^{m}\sum_{j=1}^{p}\lambda_{ij}\psi(x_{kj}, v_{ij}) \quad \text{(similar for } \lambda_i) \\
&= \sum_{j=1}^{p}\lambda_{ij}\sum_{k=1}^{n} u_{ik}^{m}\psi(x_{kj}, v_{ij}) \\
&= \sum_{j=1}^{p}\lambda_{ij}\left(\sum_{t=1}^{n_j}\sum_{\substack{i \\ x_{kj}=a_j^{(t)}}} u_{ik}^{m} - \sum_{\substack{i \\ x_{kj}=v_{ij}}} u_{ik}^{m}\right).
\end{aligned}$$

Once u_{ik}^{m} is fixed and non-negative for $1 \leq i \leq c$ and $1 \leq k \leq n$, the quantities

$$\sum_{t=1}^{n_j}\sum_{\substack{i \\ x_{kj}=a_j^{(t)}}} u_{ik}^{m}, \quad \sum_{j=1}^{p}\lambda_{ij}$$

are fixed and non-negative. Thereby, it follows that $\sum_{k=1}^{n}\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)$ is minimized if and only if each term $\sum_{\substack{i \\ x_{kj}=v_{ij}}} u_{ik}^{m}$ is maximum. □

With the insertion of the weights, it is also necessary to update the definition of the membership degrees u_{ik}. To calculate the membership degrees, it's considered the following proposition.

Proposition 3.5. *For $m > 1$, $\boldsymbol{v}_i$ and $\boldsymbol{\lambda}$ fixed, we have that*

$$u_{ik} = \left\{ \sum_{h=1}^{c} \left[\frac{\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)}{\Psi(\boldsymbol{x}_k, \boldsymbol{v}_h)} \right]^{1/(m-1)} \right\}^{-1},$$

where $\Psi(\boldsymbol{x}_k, \boldsymbol{v}_i)$ can be defined as Equations 3.3, 3.4, 3.5, and 3.6.

Proof. From the Lagrange multipliers (STEWART, 2012), to find a optimum value, consider

$$J(\rho) = \sum_{i=1}^{c} u_{ik}^m \Psi(\boldsymbol{x}_k, \boldsymbol{v}_i) - \rho(\sum_{i=1}^{c} u_{ik} - 1),$$

that is stationary when the gradient is equal to 0, that is $\nabla J(\rho) = \left[\frac{\partial}{\partial \rho} J(\rho), \frac{\partial}{\partial u_{ik}} J(\rho)\right] = 0.$ Thus, there's the following system of equations:

$$\begin{cases} \frac{\partial J(\rho)}{\partial \rho} = \sum_{i=1}^{c} u_{ik} - 1 = 0 \quad \text{(I)} \\ \frac{\partial J(\rho)}{\partial u_{st}} = m(u_{st})^{m-1} \Psi(\boldsymbol{x}_t, \boldsymbol{v}_s) - \rho = 0. \quad \text{(II)} \end{cases}$$

By the proof of Proposition 3.2, we can assume that the quantity $\Psi(\boldsymbol{x}_t, \boldsymbol{v}_s)$ can be as in Equations 3.3, 3.4, 3.5 or 3.6 and the same expressions obtained will be the optimum and maximum point for the problem. □

Besides determining the expressions for the centroids and membership degrees, that were analogs to the non-adaptive scenario, it's necessary to define the quantities that represent the weights used in Equations 3.3, 3.4, 3.5 and 3.6.

Proposition 3.6. *For $\boldsymbol{v}_i$ and $\boldsymbol{U}$ fixed, and $\sum_{j=1}^{p} \lambda_{ij} = 1$:*

1. *For $\beta > 1$ or $\beta \leq 0$, we have that*

   $$\lambda_{ij} = \left[\sum_{g=1}^{s} \left(\frac{J_{ij}}{J_{ig}} \right)^{1/(\beta-1)} \right]^{-1},$$

   where $J_{ij} = \sum_{k=1}^{n} u_{ik}^m \psi(x_{kj}, v_{ij})$ for $J_{ij} \neq 0$, and s is the number of variables which $J_{ij} \neq 0$.

2. *For $\beta = 1$, we have that*

   $$\lambda_{st} = 1 \quad \textit{and} \quad \lambda_{ij} = 0 \quad \textit{for} \quad st \neq ij,$$

   where $J_{st} \leq J_{ij}$.

Proof. Consider,

$$\begin{aligned}\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m}\Psi(\boldsymbol{x}_k,\boldsymbol{v}_i) =& \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m}\sum_{j=1}^{p}\lambda_{ij}^{\beta}\psi(x_{kj},v_{ij})\\ =&\sum_{i=1}^{c} J_i, \text{ where } J_i=\sum_{j=1}^{p}\lambda_{ij}^{\beta}J_{ij}.\end{aligned}$$

Thus, the problem becomes to minimize J_i.

If $J_{ij} = 0$, the j-th variable has a unique category in each cluster, which leads to a degenerated solution. Thus, it's necessary to set $\lambda_{ij} = 0$ for all variables in which $J_{ij} = 0$. For the s left variables ($s \leq p$),

$$F(\rho)=\sum_{j=1}^{s}\lambda_{ij}^{\beta}J_{ij}-\rho\left(\sum_{j=1}^{s}\lambda_{ij}-1\right)$$

from the Lagrange multipliers (STEWART, 2012). $F(\rho)$ is stationary when the gradient is equal to 0, that is $\nabla F(\rho) = \left[\frac{\partial}{\partial\rho}F(\rho), \frac{\partial}{\partial\lambda_{ij}}F(\rho)\right] = 0$. Thus, there's the following system of equations:

$$\begin{cases}\frac{\partial F(\rho)}{\partial\rho} = \sum_{j=1}^{s}\lambda_{ij}-1=0 \quad \text{(I)}\\ \frac{\partial F(\rho)}{\partial\lambda_{it}} = \beta\lambda_{it}^{(\beta-1)}J_{it}-\rho=0. \quad \text{(II)}\end{cases}$$

$$\text{(II)} \Rightarrow \lambda_{it}=\left(\frac{\rho}{\beta J_{it}}\right)^{1/(\beta-1)} \text{(III)}$$

$$\begin{aligned}\text{(I) and (III)} \Rightarrow \sum_{w=1}^{s}\lambda_{iw} =& \sum_{w=1}^{s}\left(\frac{\rho}{\beta J_{iw}}\right)^{1/(\beta-1)}\\ =&\left(\frac{\rho}{\beta}\right)^{1/(\beta-1)}\sum_{w=1}^{s}\left(\frac{1}{J_{iw}}\right)^{1/(\beta-1)}\\ \Rightarrow \left(\frac{\rho}{\beta}\right)^{1/(\beta-1)} =& \frac{1}{\sum_{w=1}^{s}\left(\frac{1}{J_{iw}}\right)^{1/(\beta-1)}}.\end{aligned}$$

Backing to (III),

$$\begin{aligned}\lambda_{it} =& \frac{1}{\sum_{w=1}^{s}\left(\frac{1}{J_{iw}}\right)^{1/(\beta-1)}}\left(\frac{1}{J_{it}}\right)^{1/(\beta-1)}\\ =&\left[\sum_{w=1}^{s}\left(\frac{J_{it}}{J_{iw}}\right)^{1/(\beta-1)}\right]^{-1}.\end{aligned}$$

For $\beta = 1$, we have that

$$J_i = \sum_{j=1}^{p} \lambda_{ij} J_{ij}.$$

It follows that $\sum_{j=1}^{p} \lambda_{ij} J_{ij} \geq J_{st}$, where $J_{st} \leq J_{ij}$ because $\sum_{j=1}^{p} \lambda_{ij}((J_{ij} - J_{st}) \geq 0$ (all the non-negative terms). Thus,

$$\begin{aligned} \Rightarrow \sum_{j=1}^{p} \lambda_{ij}(J_{ij} - J_{st}) &= \sum_{j=1}^{p} \lambda_{ij} J_{ij} - \sum_{j=1}^{p} \lambda_{ij} J_{st} \\ &= \sum_{j=1}^{p} \lambda_{ij} J_{ij} - J_{st} \sum_{j=1}^{p} \lambda_{ij} \\ &= \sum_{j=1}^{p} \lambda_{ij} J_{ij} - J_{st}. \end{aligned}$$

Then, $\lambda_{st} = 1$ and $\lambda_{ij} = 0 \; \forall ij \neq st$. □

Proposition 3.7. *For $\boldsymbol{v}_i$ and $\boldsymbol{U}$ fixed, and $\sum_{j=1}^{p} \lambda_j = 1$*

1. *For $\beta > 1$ or $\beta \leq 0$, we have that*

$$\lambda_j = \left[\sum_{g=1}^{s} \left(\frac{\sum_{i=1}^{c} J_{ij}}{\sum_{i=1}^{c} J_{ig}} \right)^{1/(\beta-1)} \right]^{-1},$$

where $J_{ij} = \sum_{k=1}^{n} u_{ik}^{m} \psi(x_{kj}, v_{ij})$ for $J_{ij} \neq 0$, and s is the number of variables which $J_{ij} \neq 0$.

2. *For $\beta = 1$, we have that*

$$\lambda_t = 1 \quad \textit{and} \quad \lambda_j = 0 \quad \textit{for} \quad t \neq j,$$

where $J_t \leq J_j$.

Proof. The result follows from Proposition 3.6. Notice that λ_j can be written as

$$\lambda_j = \left\{ \sum_{g=1}^{s} \left[\frac{J_i(\lambda_{i1}, \ldots, \lambda_{ij})}{J_i(\lambda_{i1}, \ldots, \lambda_{ig})} \right]^{1/(\beta-1)} \right\}^{-1}.$$

□

Proposition 3.8. *For $\boldsymbol{v}_i$ and $\boldsymbol{U}$ fixed, and $\prod_{j=1}^{p} \lambda_{ij} = 1$, we have that*

$$\lambda_{ij} = \frac{(\prod_{g=1}^{s} J_{ig})^{1/s}}{J_{ij}},$$

where $J_{ij} = \sum_{k=1}^{n} u_{ik}^{m} \psi(x_{kj}, v_{ij})$ for $J_{ij} \neq 0$, and s is the number of variables which $J_{ij} \neq 0$.

Proof. If $J_{ij} = 0$, the j-th variable has a unique category in each cluster, which leads to a degenerated solution. Thus, it's necessary to set $\lambda_{ij} = 0$ for all variables in which $J_{ij} = 0$. For the s left variables ($s \leq p$),

$$F(\rho) = \sum_{j=1}^{s} \lambda_{ij} J_{ij} - \rho \Bigg(\prod_{j=1}^{s} \lambda_{ij} - 1 \Bigg)$$

from the Lagrange multipliers (STEWART, 2012). $F(\rho)$ is stationary when the gradient be equal to 0, that is $\nabla F(\rho) = \left[\frac{\partial}{\partial \rho} F(\rho), \frac{\partial}{\partial \lambda_{ij}} F(\rho) \right] = 0$. Thus, there's the following system of equations:

$$\begin{cases} \frac{\partial F(\rho)}{\partial \rho} = \prod_{j=1}^{s} \lambda_{ij} - 1 = 0 \quad \text{(I)} \\ \frac{\partial F(\rho)}{\partial \lambda_{it}} = \lambda_{it} J_{it} - \rho = 0. \quad \text{(II)} \end{cases}$$

$$\text{(II)} \Rightarrow \lambda_{it} = \frac{\rho}{J_{it}} \text{ (III)}$$

$$\begin{aligned} \text{(I) and (III)} \Rightarrow \prod_{w=1}^{s} \frac{\rho}{J_{iw}} &= \frac{\rho^s}{\prod_{w=1}^{s} J_{iw}} = 1 \\ \Rightarrow \rho^s &= \prod_{w=1}^{s} J_{iw} \\ \rho &= (\prod_{w=1}^{s} J_{iw})^{(1/s)} \end{aligned}$$

Backing to (III),

$$\lambda_{it} = \frac{(\prod_{w=1}^{s} J_{iw})^{1/s}}{J_{it}}.$$

□

Proposition 3.9. *For $\boldsymbol{v}_i$ and $\boldsymbol{U}$ fixed, and $\prod_{j=1}^{p} \lambda_j = 1$, we have that*

$$\lambda_j = \frac{\left[\prod_{g=1}^{s} (\sum_{i=1}^{c} J_{ig})^{1/s} \right]}{\sum_{i=1}^{c} J_{ij}},$$

where $J_{ij} = \sum_{k=1}^{n} u_{ik}^{m} \psi(x_{kj}, v_{ij})$ for $J_{ij} \neq 0$, and s is the number of variables which $J_{ij} \neq 0$.

Proof. The result follows from Proposition 3.8. □

3.1.2 Algorithm

The Fuzzy C-modes algorithm with variable weighting is summarized in Algorithm 1. Notice that the fuzzy version is easily turned into a hard version by setting the constraints for the membership degrees $u_{ik} = \{0, 1\}$ and $m \to 1$.

According to the propositions in this chapter, the computational complexity of the Algorithm 1 is $O(TCNP)$, where T is the total number of iterations required, and N, P, C indicate the number of objects, variables, and clusters respectively. For storage, it needed memory to keep:

- The dataset objects (NP);
- The membership matrix (CN);
- The variable weight matrix (CP).

Algorithm 1 Fuzzy C-Modes with variable weighting Algorithm

```
Input
    The set $\Omega = \{w_1, \dots, w_n\}$;
    The number $c$ of clusters $(2 \leq c \leq n)$;
    The parameter $T$ (maximum of iterations);
    The parameter $t^*$ (counter of iterations);
    The threshold $\epsilon > 0$ and $\epsilon \ll 1$.
Initialization
    Set $t^* = 0$;
    Randomly select $c$ distinct centroids $\boldsymbol{v}_i \in \Omega$ with $i = \{1, \dots, c\}$;
    Randomly initialize the matrix of membership degrees $\boldsymbol{U}$ such that $u_{ik} \geq 0$ and $\sum_{i=1}^{c} u_{ik} = 1$;
    Initialize the matrix of relevance weights $\boldsymbol{\lambda}$ with the restriction as in Equations 3.2, 3.3, 3.4, 3.5 or 3.6;
    Compute the $J$ according to the Equation 3.1.
Repeat
    Set $J_{t^*-1} = J_{t^*}$;
    Set $t^* = t^* + 1$.
      1. Representation
         For $i = 1, \dots, c$ and $j = 1, \dots, p$, compute the component $v_{ij}$ of the centroid $\boldsymbol{v}_i = (v_{i1}, v_{i2}, \dots, v_{ip})$ according to the Proposition 3.4.
      2. Weighting
         Compute the elements $\lambda_{ij}$ (or $\lambda_j$) of the matrix of relevance weights $\boldsymbol{\lambda}$, according to Propositions 3.6, 3.8, 3.7, and 3.9.
      3. Allocation
         Compute the elements $u_{ik}$ of the matrix of membership degrees $\boldsymbol{U}$, according to the Proposition 3.5.
    Compute the $J_{t^*}$ according to the Equation 3.1;
    Until $|J_{t^*} - J_{t^*-1}| < \epsilon$ or $t^* > T$.
```

4 EXPERIMENTAL EVALUATION

This chapter provides the performance evaluation of the proposed algorithms and a comparison with literature methods. A series of experiments were performed with various datasets, including synthetic and real ones. A set of measures used to evaluate both hard and fuzzy partition algorithms (RODRíGUES, 2018) are described in the following sections. The measures in this work are in the context of two types of indexes: internal and external (LIU et al., 2010).

4.1 INTERNAL INDEXES

Internal indexes obtain the quality of clustering from the information of the set Ω itself, without any external information. Usually, an internal index analyzes whether the positions of objects in an obtained clustering match the proximity matrix $\boldsymbol{P}$ (JAIN; DUBES, 1988). Therefore, in this case, the method is evaluated by measuring the deviation between the structure generated by it and the original set Ω. This type of index is most used in the context of real data.

4.1.1 Partition coefficient

Given a membership matrix $\boldsymbol{U}$, with n objects and c clusters as dimensions, it was attempted to define a performance measure based on minimizing the overall content of pairwise fuzzy intersection in the partition matrix $\boldsymbol{U}$. (BEZDEK, 1981) proposed a clustering validity index for fuzzy clustering named Partition Coefficient (PC). The index was defined as

$$V_{PC} = \frac{1}{n} \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^2. \tag{4.1}$$

The PC index indicates the average relative amount of membership sharing done between pairs of fuzzy subsets in $\boldsymbol{U}$, by combining into a single number, the average contents of pairs of algebraic products. The index values range in $[1/c, 1]$, for c clusters, the closer the value of PC to 1, the harder are the clustering partitions.

4.1.2 Partition entropy coefficient

(BEZDEK, 1975) proposed the Partition Entropy (PE) defined as:

$$V_{PE} = -\frac{1}{n} \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik} \log u_{ik}. \tag{4.2}$$

The PE index is a scalar measure of the amount of fuzziness in a given $\boldsymbol{U}$. The PE index values range in $[0, \log c]$, the closer the value of PE to 0, the harder are the clustering partitions. The index value close to the upper bound indicates the absence of any clustering structure in the datasets or the inability of the algorithm to extract it.

4.1.3 Modified partition coefficient

Both PC and PE possess monotonic evolution tendencies with the number of clusters. According to (BEZDEK, 1981), that characteristic can be attributed to their apparent monotonicity and an extent, to the heuristic nature of the rationale underlying its formulation. The modification of the V_{PC} index can reduce the monotonic tendency and is defined as

$$V_{MPC} = 1 - \frac{c}{c-1}(1 - V_{PC}). \tag{4.3}$$

The range of the MPC index is the unit interval $[0, 1]$, where MPC $= 0$ corresponds to maximum fuzziness and MPC $= 1$ to a hard partition.

4.1.4 Silhouette index

To define this criterion, consider a data object $\boldsymbol{x}_k$ belonging to cluster $i \in \{1, \ldots, c\}$. In the context of hard partitions produced by a centroid-based clustering algorithm. This means that the object $\boldsymbol{x}_k$ is closer to the centroid of cluster i than to any other centroid. Let the average distance of object $\boldsymbol{x}_i$ to all other objects belonging to cluster i be denoted by a_{ik}. Also, let the average distance of this object to all objects belonging to another cluster s, $s \neq i$, be called by d_{sk}. Finally, let b_{ik} be the minimum d_{sk} computed over $s = 1, \ldots, c$, which represents the dissimilarity of object $\boldsymbol{x}_k$ to its closest neighboring cluster. Then, the silhouette of object $\boldsymbol{x}_k$ is defined as

$$s_k = \frac{b_{ik} - a_{ik}}{max(a_{ik}, b_{ik})}, \tag{4.4}$$

where the denominator is used just as a normalization term. The higher s_k, the better the assignment of object $\boldsymbol{x}_k$ to cluster i. In case i is a singleton, then the silhouette of this object is defined as $s_k = 0$. This prevents the Silhouette index, defined as the average of s_k over $k = 1, \ldots, n$, to find the trivial solution $c = n$, with the object of the dataset forming a cluster on its own. This way, the best partition is achieved when the S is maximized, which implies minimizing the intra-cluster distance (a_{ik}) while maximizing the inter-cluster distance (b_{ik})

(HRUSCHKA; CASTRO; CAMPELLO, 2004).

$$\mathsf{S} = \frac{1}{n}\sum_{k=1}^{n} s_k. \tag{4.5}$$

4.2 EXTERNAL INDEXES

External indexes evaluate a clustering according to external information, usually a researcher's intuition about the structure present in the data or a cluster built by a domain expert. In this case, the result of the method is evaluated by comparing it with a predefined structure, which is imposed on the set Ω, and which represents the actual structure of the data in clusters.

4.2.1 Adjusted Rand index

The Adjusted Rand Index (ARI) is the corrected-for-chance version of the Rand Index (RAND, 1971). Though the Rand index may only yield a value between 0 and 1, the ARI can yield negative values if the index is less than the expected index (HUBERT; ARABIE, 1985). The Adjusted Rand Index evaluates how close are two partitions: a prior partition and a final partition obtained from a clustering algorithm.

The index values on the interval $[-1, 1]$ indicate the perfect matching between the two partitions, when 1 is observed, and a random matching when 0 (or negative) is observed. Let $\boldsymbol{A} = \{a_1, \ldots, a_i, \ldots, a_R\}$ and $\boldsymbol{B} = \{b_1, \ldots, b_j, \ldots, b_S\}$ be two partitions of the same data having respectively R and S clusters. The Adjusted Rand Index is

$$ARI = \frac{\sum_{i=1}^{R}\sum_{j=1}^{S}\binom{n_{ij}}{2} - \binom{n}{2}^{-1}\sum_{i=1}^{R}\binom{n_{i.}}{2}\sum_{j=1}^{S}\binom{n_{.j}}{2}}{\frac{1}{2}\Big[\sum_{i=1}^{R}\binom{n_{i.}}{2} + \sum_{j=1}^{S}\binom{n_{.j}}{2}\Big] - \binom{n}{2}^{-1}\sum_{i=1}^{R}\binom{n_{i.}}{2}\sum_{j=1}^{S}\binom{n_{.j}}{2}}, \tag{4.6}$$

where $\binom{n}{2} = \frac{n(n-1)}{2}$, n_{ij} represents the number of objects that are in partition a_i and b_j. While, $n_{i.}$ and $n_{.j}$ represents the number of objects that are in partition a_i and b_j, respectively.

4.3 PARAMETERS AND ALGORITHM SETTINGS

To evaluate the performance of the proposed algorithms a series of experiments were performed with various datasets, synthetic and real ones. Also, for each dataset considered there is a comparison of the fuzzy and hard versions and the variable-weighted versions. For each dataset, the PC (4.1), PE (4.2), and MPC (4.3) indexes were computed for the fuzzy versions and the ARI (4.6) and S (4.5) indexes were computed for the hard versions in the framework of a Monte Carlo (HAMMERSLEY, 2013) simulation with 100 replicas. In each replica, the clustering algorithms were run, until the convergence to a stationary value of the adequacy criterion, 100 times, and the best result for each method was selected according to the adequacy criterion. The fuzziness parameter m and the degree of influence parameter β were set equal to 2 to limit the search and evaluation space of the algorithms. It is expected that the variation of these parameters may lead to different results and characterizations of the clustering. We set $\epsilon = 10^{-10}$ as the tolerance for the convergence of the adequacy criterion and $T = 100$ as the number of iterations. From the fuzzy partitions given by these clustering algorithms, it is obtained a hard partition by assigning each object to a hard cluster, allowing the computation of the ARI thereafter (FERREIRA; CARVALHO, 2014). The hard partition is created by assigning an object $\boldsymbol{x}_k$ to a cluster i if $u_{ik} = \max_h(u_{hk})$, for $1 \leq h \leq c$. The Algorithm 1 was implemented in the R programming language (R Core Team, 2023) and is available via package [1].

4.4 EXPERIMENTS ON SYNTHETIC DATASETS

The synthetic datasets in this section followed the rationale presented in (MINGOTI; MATOS, 2012), in which the authors present an intense simulation study, based on Monte Carlo experiments, to compare five important qualitative methods. The simulated data were generated in order to offer different degrees of difficulty for the algorithms, on different types of data. In total, 8 datasets were generated for different scenarios. Such scenarios can be classified into 2 general cases, with the following differentiating characteristics: the level of dispersion of the variables and, consequently, the level of superposition (overlapping) of classes.

The level of dispersion concerns the frequency distribution of the categories of variables, for each of the classes, while the level of superposition is related to the constructed similarities between classes. The more similar the classes are, variables with equal dispersion for each of the classes, the greater the overlap between them. That is, a certain category of a variable with the same frequency in different classes. This characteristic imposed on the data makes the clustering problem more difficult to be solved by the algorithms of the methods.

In addition to these characteristics, the scenarios differ from each other by the number of categories' variables, which vary between 2, 3, 4, and 5. There are situations in which the

[1] Available at https://github.com/gabrielteotonio/cModes.

quantity of categories is the same for all, for a part, or, for no variables. In addition, it varies also the degree of control over the process (number of fixed variables), which is 50% of the variables for every dataset. So many scenarios were outlined, trying to expose the methods to the most diverse types of data. The main motivation for creating the different scenarios is to compare the same configuration in relation to the number of classes, variables, and categories with the absence or presence of overlapping. This allows the generalization of the results and a more detailed analysis of the entire simulation process. The scenarios used are presented next.

1. Without overlapping

   It was created 4 datasets for this scenario with the following configurations:

   - Number of classes: 2; Number of variables: 2; Overlapping degree: no overlapping in the first variable; Number of categories on the remaining variables: 2 **(Case 1 - I)**;
   - Number of classes: 3; Number of variables: 2; Overlapping degree: no overlapping in the first variable; Number of categories on the remaining variables: 5 **(Case 1 - II)**;
   - Number of classes: 3; Number of variables: 4; Overlapping degree: no overlapping in the first and second variables; Number of categories on the remaining variables: 5 **(Case 1 - III)**;
   - Number of classes: 5; Number of variables: 4; Overlapping degree: no overlapping in the first and second variables; Number of categories on the remaining variables: 5 **(Case 1 - IV)**.

2. With overlapping

   It was created 4 datasets for this scenario with the following configurations:

   - Number of classes: 2; Number of variables: 2; Overlapping degree: overlapping in the first variable; Number of categories on the remaining variables: 2 **(Case 2 - I)**;
   - Number of classes: 3; Number of variables: 2; Overlapping degree: overlapping in the first variable; Number of categories on the remaining variables: 5 **(Case 2 - II)**;
   - Number of classes: 3; Number of variables: 4; Overlapping degree: overlapping in the first and second variables; Number of categories on the remaining variables: 5 **(Case 2 - III)**;

- Number of classes: 5; Number of variables: 4; Overlapping degree: overlapping in the first and second variables; Number of categories on the remaining variables: 5 **(Case 2 - IV)**.

Notice that for every dataset, 50% of the variables are deterministic to ensure the overlapping degree of the variables' categories, and consequently, the superposition among the classes. The number of observations in each dataset generated is determined by the number of classes defined, 50 observations for each class. Also, it's assumed that all the variables in the synthetic datasets are nominal. In the following section, it's described the method used to generate the non-deterministic variables for the creation of the datasets.

4.4.1 Qualitative random variables generator

In this work, the beta distribution is adopted for generating the non-deterministic variables, according to (MINGOTI; MATOS, 2012) which set a prior empirical distribution with the desired behavior and then applies the inverse transform method. Thus, for each class c and variable j, with $j \in \{1, \ldots, p\}$, let c_{tj} be a possible category of A_j, where $c_{tj} \in \text{DOM}(A_j)$. The steps for the random variables generation, by using the beta distribution, are the following:

1. Determine the probability of occurrence of each of the categories of the variable A_j, $p_{c_{tj}}$, obtained as occurrences of the Beta(1;0.1) distribution (GUPTA; NADARAJAH, 2004);

2. Normalize, obtaining the new probabilities $p^*_{c_{tj}}$. So, $\mathbb{P}[A_j = c_{tj}] = p^*_{c_{tj}}$, $\forall c_{tj} \in \text{DOM}(A_j)$ and $\sum_{c_{tj}} p^*_{c_{tj}} = 1$;

3. Randomly generate the observations of the distribution of $p^*_{c_{tj}}$. These observations are equivalent to the occurrences of the categories of the variable A_j.

For step 3, the inverse transformation method is used for discrete variables, which consists of:

1. From the probabilities $p^*_{c_{tj}}$, find the cumulative distribution;

2. Generate a random number U, from the Uniform(0;1) distribution (ROSS, 2010);

3. Compare the obtained value U and assign it to the appropriate category of A_j, that is,

$$A_j = \begin{cases} c_{1j}, & \text{if } 0 \leq U \leq p^*_{c_{1j}} \\ c_{2j}, & \text{if } p^*_{c_{1j}} < U \leq p^*_{c_{1j}} + p^*_{c_{2j}} \\ \vdots & \\ c_{tj}, & \text{if } p^*_{c_{1j}} + \cdots + p^*_{c_{(t-1)j}} < U \leq 1 \end{cases} .$$

4.4.2 Results and analysis

The results obtained by the executions of the algorithm versions are following, separated by dataset scenarios.

4.4.2.1 Non-overlapping datasets

Tables 1 and 2 show the evaluation metrics results for the datasets with 2 variables. Among the hard algorithm versions, the methods with adaptive distances usage perform worst than the classical methods. The results for the fuzzy algorithm versions are quite different. In Case 1 - I, there's no performance difference between the proposed ones and the classical ones. On the other hand, in Case 1 - II, the fuzzy proposed algorithms performed better than the classical method for all internal evaluation metrics. The fuzzy version with local adaptive-sum distance has the best performance.

Table 1 – Algorithm evaluation metrics results for non-overlapping dataset Case 1 - I

Version	ARI	S	PC	PE	MPC
k-modes	1	0.6620	-	-	-
fuzzy c-modes	0.2630	-	0.7650	0.3258	0.5300
k-modes with local adaptive-sum	0.3080	0.4426	-	-	-
k-modes with local adaptive-product	0.3080	0.4426	-	-	-
k-modes with global adaptive-sum	0.3070	0.3567	-	-	-
k-modes with global adaptive-product	0.1850	0.3710	-	-	-
fuzzy c-modes with local adaptive-sum	0.3080	-	0.7688	0.3220	0.5376
fuzzy c-modes with local adaptive-product	0.3080	-	0.7660	0.3248	0.5319
fuzzy c-modes with global adaptive-sum	0.3066	-	0.7650	0.3258	0.5300
fuzzy c-modes with global adaptive-product	0.3296	-	0.7650	0.3258	0.5300

Source: The author (2023)

For Case 1 - III and Case 1 - IV the results are similar to Case 1 - II on the hard and fuzzy algorithm versions, as in Tables 3 and 4. And the fuzzy version with local adaptive-sum has the best performance again, and in Case 1 - IV it has scored the theoretical bound of the evaluation metrics. Among the cases for non-overlapping datasets, Case 1 - I was the only one without improvement when considering the adaptive distances. It may be related to the number of categories on the random variables, which was 2 for this particular case, while it was equal to 5 for the remaining.

The ARI values for all Case I scenarios show that the classical methods perform better than the proposed ones. In the latter three scenarios, the gap between the classical hard and fuzzy is smaller. For the proposed methods, there's little fluctuation among the weighting settings on each hard and fuzzy algorithm. Except for fuzzy versions in Case 1 - II, wherein the local adaptive versions overcome the global adaptive ones.

Table 2 – Algorithm evaluation metrics results for non-overlapping dataset Case 1 - II

Version	ARI	S	PC	PE	MPC
k-modes	0.6840	0.3890	-	-	-
fuzzy c-modes	0.5445	-	0.5398	0.7616	0.3098
k-modes with local adaptive-sum	0.2460	0.0882	-	-	-
k-modes with local adaptive-product	0.2580	0.0895	-	-	-
k-modes with global adaptive-sum	0.2710	0.1109	-	-	-
k-modes with global adaptive-product	0.2490	0.1017	-	-	-
fuzzy c-modes with local adaptive-sum	0.4067	-	0.7006	0.4153	0.5509
fuzzy c-modes with local adaptive-product	0.4067	-	0.6937	0.4247	0.5406
fuzzy c-modes with global adaptive-sum	0.2415	-	0.6933	0.4251	0.5400
fuzzy c-modes with global adaptive-product	0.2468	-	0.6933	0.4251	0.5400

Source: The author (2023)

Table 3 – Algorithm evaluation metrics results for non-overlapping dataset Case 1 - III

Version	ARI	S	PC	PE	MPC
k-modes	0.9410	0.5138	-	-	-
fuzzy c-modes	0.9009	-	0.4628	0.8983	0.1942
k-modes with local adaptive-sum	0.2460	0.0929	-	-	-
k-modes with local adaptive-product	0.2470	0.0836	-	-	-
k-modes with global adaptive-sum	0.2510	0.0833	-	-	-
k-modes with global adaptive-product	0.2430	0.0691	-	-	-
fuzzy c-modes with local adaptive-sum	0.2367	-	0.7800	0.3443	0.6700
fuzzy c-modes with local adaptive-product	0.3554	-	0.6965	0.4496	0.5447
fuzzy c-modes with global adaptive-sum	0.2418	-	0.7158	0.4157	0.5737
fuzzy c-modes with global adaptive-product	0.2435	-	0.7158	0.4157	0.5737

Source: The author (2023)

4.4.2.2 Overlapping datasets

Tables 5 and 6 show the evaluation metrics results for the datasets with 2 variables. Among the hard algorithm versions, the methods with adaptive distances usage perform worst than the classical methods. However, the results for the fuzzy algorithm versions are different. In

Table 4 – Algorithm evaluation metrics results for non-overlapping dataset Case 1 - IV

Version	ARI	S	PC	PE	MPC
k-modes	0.8110	0.4178	-	-	-
fuzzy c-modes	0.5422	-	0.3075	1.3879	0.1344
k-modes with local adaptive-sum	0.0595	-0.0012	-	-	-
k-modes with local adaptive-product	0.0718	-0.0005	-	-	-
k-modes with global adaptive-sum	0.0846	0.0187	-	-	-
k-modes with global adaptive-product	0.0986	0.0279	-	-	-
fuzzy c-modes with local adaptive-sum	0	-	1.0000	0.0000	1.0000
fuzzy c-modes with local adaptive-product	0.1008	-	0.5485	0.7628	0.4356
fuzzy c-modes with global adaptive-sum	0.0924	-	0.5473	0.7651	0.4341
fuzzy c-modes with global adaptive-product	0.0879	-	0.5473	0.7651	0.4341

Source: The author (2023)

Case 2 - I, there's no performance difference between the proposed ones and the classical ones, except for the local adaptive versions. On the other hand, in Case 2 - II, all the fuzzy proposed algorithms performed better than the classical methods for all evaluation metrics. Again, as observed for the non-overlapping datasets, the fuzzy version with local adaptive-sum distance has the best performance.

Table 5 – Algorithm evaluation metrics results for overlapping dataset Case 2 - I

Version	ARI	S	PC	PE	MPC
k-modes	1	0.6776	-	-	-
fuzzy c-modes	0.2424	-	0.7750	0.3119	0.5500
k-modes with local adaptive-sum	0.4310	0.4913	-	-	-
k-modes with local adaptive-product	0.4310	0.4913	-	-	-
k-modes with global adaptive-sum	0.2230	0.3947	-	-	-
k-modes with global adaptive-product	0.3300	0.3868	-	-	-
fuzzy c-modes with local adaptive-sum	0.4305	-	0.8229	0.2622	0.6458
fuzzy c-modes with local adaptive-product	0.4305	-	0.7884	0.2983	0.5769
fuzzy c-modes with global adaptive-sum	0.3781	-	0.7750	0.3119	0.5500
fuzzy c-modes with global adaptive-product	0.1068	-	0.7750	0.3119	0.5500

Source: The author (2023)

For Case 2 - III and Case 2 - IV the results are similar to Case 2 - II on the hard and fuzzy algorithm versions, as in Tables 7 and 8. And the fuzzy version with local adaptive-sum has the best performance again, and in Case 2 - IV, as observed in Case 2 - IV, it has scored the theoretical bound of the evaluation metrics. Unlike what happened in the scenarios without overlapping, all scenarios with datasets that allowed overlapping had some improvement in the metrics observed in the fuzzy versions of the algorithm.

The ARI values for Case 2 - I and Case 2 - III scenarios show that the proposed fuzzy methods perform better than the fuzzy classical one. In Case 2 - I, this overcoming is from the local adaptive versions, while in Case 2 - III is from the global adaptive sum version. For Case 2 - II and Case 2 - IV scenarios, the classical methods perform better than the proposed

Table 6 – Algorithm evaluation metrics results for overlapping dataset Case 2 - II

Version	ARI	S	PC	PE	MPC
k-modes	0.4130	0.2754	-	-	-
fuzzy c-modes	0.4753	-	0.5560	0.7344	0.3340
k-modes with local adaptive-sum	0.2470	0.0983	-	-	-
k-modes with local adaptive-product	0.2420	0.0932	-	-	-
k-modes with global adaptive-sum	0.2440	0.1194	-	-	-
k-modes with global adaptive-product	0.2400	0.0934	-	-	-
fuzzy c-modes with local adaptive-sum	0.2660	-	0.8794	0.1981	0.8192
fuzzy c-modes with local adaptive-product	0.3446	-	0.8271	0.2683	0.7407
fuzzy c-modes with global adaptive-sum	0.2662	-	0.6811	0.4688	0.5217
fuzzy c-modes with global adaptive-product	0.3002	-	0.6811	0.4688	0.5217

Source: The author (2023)

ones. In addition, the same fluctuation among the weighting settings on each hard and fuzzy algorithm observed for non-overlapping datasets is seen here for Case 2 - II and Case 2 - IV.

Table 7 – Algorithm evaluation metrics results for overlapping dataset Case 2 - III

Version	ARI	S	PC	PE	MPC
k-modes	0.2580	0.1587	-	-	-
fuzzy c-modes	0.1461	-	0.4087	0.9828	0.1131
k-modes with local adaptive-sum	0.0662	0.0860	-	-	-
k-modes with local adaptive-product	0.1330	0.0842	-	-	-
k-modes with global adaptive-sum	0.1180	0.0871	-	-	-
k-modes with global adaptive-product	0.1720	0.1068	-	-	-
fuzzy c-modes with local adaptive-sum	0.1202	-	0.7435	0.4112	0.6152
fuzzy c-modes with local adaptive-product	0.1443	-	0.7211	0.4455	0.5816
fuzzy c-modes with global adaptive-sum	0.1754	-	0.7202	0.4468	0.5803
fuzzy c-modes with global adaptive-product	0.1448	-	0.7202	0.4468	0.5803

Source: The author (2023)

Table 8 – Algorithm evaluation metrics results for overlapping dataset Case 2 - IV

Version	ARI	S	PC	PE	MPC
k-modes	0.1800	0.1135	-	-	-
fuzzy c-modes	0.1045	-	0.2444	1.5154	0.0556
k-modes with local adaptive-sum	0.0725	0.0229	-	-	-
k-modes with local adaptive-product	0.0703	0.0242	-	-	-
k-modes with global adaptive-sum	0.0659	0.0071	-	-	-
k-modes with global adaptive-product	0.0743	0.0256	-	-	-
fuzzy c-modes with local adaptive-sum	0	-	1.0000	0.0000	1.0000
fuzzy c-modes with local adaptive-product	0.0853	-	0.6293	0.6122	0.5366
fuzzy c-modes with global adaptive-sum	0.0577	-	0.5947	0.6645	0.4933
fuzzy c-modes with global adaptive-product	0.0714	-	0.5947	0.6645	0.4933

Source: The author (2023)

In order to compare these methods, Student's t tests for independent samples with 5% of significance are performed. Tables 9 and 10 give the values of the p-value. In these tables, μ_1, μ_2, μ_3, μ_4, μ_5, μ_6, μ_7, μ_8, μ_9, and μ_{10} are, respectively, the average of the ARI and MPC indexes for K-Modes, Fuzzy C-Modes, K-Modes with local adaptive-sum, K-Modes with local adaptive-product, K-Modes with global adaptive-sum, K-Modes with global adaptive-product, Fuzzy C-Modes with local adaptive-sum, Fuzzy C-Modes with local adaptive-product, Fuzzy C-Modes with global adaptive-sum, and Fuzzy C-Modes with global adaptive-product.

Table 9 – Statistical tests comparing K-Modes method for synthetic datasets

Dataset	**Statistical test**			
Case 1 - I	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (0.0295) Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (0.0295) Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (0.1636) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (0.3798) Not Reject H_0
Case 1 - II	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (1) Not Reject H_0
Case 1 - III	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (1) Not Reject H_0
Case 1 - IV	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (1) Not Reject H_0
Case 2 - I	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (0.2034) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (0.0537) Not Reject H_0
Case 2 - II	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (1) Not Reject H_0
Case 2 - III	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (1) Not Reject H_0
Case 2 - IV	$H_0 : \mu_1 \geq \mu_3$ $H_1 : \mu_1 < \mu_3$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_4$ $H_1 : \mu_1 < \mu_4$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_5$ $H_1 : \mu_1 < \mu_5$ (1) Not Reject H_0	$H_0 : \mu_1 \geq \mu_6$ $H_1 : \mu_1 < \mu_6$ (1) Not Reject H_0

Source: The author (2023)

Values in these tables support the hypothesis that the fuzzy adaptive versions were superior to the classical fuzzy version for all datasets, except Case 1 - I and Case 2 - I. For the hard versions, only the local adaptive versions were superior to the classical hard version and only for Case 1 - I and Case 2 - I datasets.

Table 10 – Statistical tests comparing Fuzzy C-Modes method for synthetic datasets

Dataset	Statistical test			
Case 1 - I	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (1) Not Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (1) Not Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (1) Not Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (1) Not Reject H_0
Case 1 - II	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (2.2×10^{-16}) Reject H_0
Case 1 - III	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (2.2×10^{-16}) Reject H_0
Case 1 - IV	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (2.2×10^{-16}) Reject H_0
Case 2 - I	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (0.2705) Not Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (0.4291) Not Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (1) Not Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (1) Not Reject H_0
Case 2 - II	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (2.2×10^{-16}) Reject H_0
Case 2 - III	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (2.2×10^{-16}) Reject H_0
Case 2 - IV	$H_0 : \mu_2 \geq \mu_7$ $H_1 : \mu_2 < \mu_7$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_8$ $H_1 : \mu_2 < \mu_8$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_9$ $H_1 : \mu_2 < \mu_9$ (2.2×10^{-16}) Reject H_0	$H_0 : \mu_2 \geq \mu_{10}$ $H_1 : \mu_2 < \mu_{10}$ (2.2×10^{-16}) Reject H_0

Source: The author (2023)

Table 11 – Average and standard deviation (in parentheses) of convergence time (in seconds) for methods and datasets

Version	Case 1 - I	Case 1 - II	Case 1 - III	Case 1 - IV	Case 2 - I	Case 2 - II	Case 2 - III	Case 2 - IV
k-modes	0.2224 (0.1300)	0.1610 (0.0585)	0.3132 (0.1280)	0.5217 (0.2110)	0.1077 (0.0367)	0.1757 (0.0664)	0.3977 (0.1790)	0.4687 (0.1880)
fuzzy c-modes	0.1687 (0.0978)	0.1682 (0.0466)	0.3220 (0.0749)	0.7394 (0.1970)	0.0868 (0.0453)	0.1931 (0.0618)	0.4184 (0.1500)	0.8222 (0.2040)
k-modes with local adaptive-sum	0.2955 (0.1500)	6.5422 (4.8300)	22.8728 (1.9300)	47.3100 (7.1100)	0.1426 (0.0545)	11.7642 (1.8700)	27.9531 (5.2500)	45.9034 (2.0500)
k-modes with local adaptive-product	6.5101 (7.5800)	6.7734 (4.7200)	22.8655 (1.4900)	47.6947 (6.6100)	2.2193 (3.0500)	11.2770 (3.4400)	28.3822 (5.0900)	45.8122 (2.2700)
k-modes with global adaptive-sum	1.5747 (3.2900)	3.8905 (3.1900)	28.6841 (3.8600)	60.1904 (12.8000)	0.6381 (0.6290)	8.9936 (4.2700)	35.3214 (6.7200)	60.7948 (2.6300)
k-modes with global adaptive-product	1.2406 (1.7500)	2.9530 (2.3600)	22.7089 (3.0900)	47.8269 (7.1900)	0.6802 (0.7060)	7.1671 (4.9000)	28.4810 (4.7400)	46.5743 (2.2200)
fuzzy c-modes with local adaptive-sum	2.3126 (2.9600)	18.4204 (3.4800)	29.2313 (3.4500)	71.7670 (21.9000)	1.7073 (2.7100)	18.9102 (2.3300)	35.7915 (6.5400)	74.2699 (13.2000)
fuzzy c-modes with local adaptive-product	0.9781 (1.1000)	18.4999 (4.7900)	29.0133 (3.4000)	80.6084 (5.8200)	0.4587 (0.4500)	18.7804 (2.3500)	35.0942 (4.7300)	77.0562 (5.2200)
fuzzy c-modes with global adaptive-sum	0.4282 (0.2360)	19.6765 (4.2800)	34.8433 (3.9600)	95.5565 (5.4900)	0.2123 (0.0778)	19.8358 (2.4600)	41.6811 (4.6300)	92.3950 (5.4000)
fuzzy c-modes with global adaptive-product	0.4109 (0.3030)	18.3652 (4.1600)	29.0566 (3.5300)	80.6121 (6.7700)	0.2010 (0.0831)	18.5455 (2.3700)	35.4544 (4.9300)	77.1271 (4.6900)

Source: The author (2023)

In order to compare the performance of each method using the computational cost, a study was made concerning the number of iterations and the convergence time for each algorithm. Regarding time complexity, the non-adaptive methods don't need to compute the relevance weights matrix, while the adaptive methods do. Therefore, in terms of time complexity, methods based on adaptive distances have a greater computational cost than methods with non-adaptive distances.

The time (in seconds) was noted for each algorithm until convergence. After 100 replications, the average and standard deviation of these measures were calculated. Table 11 shows values of the average and standard deviation of the time. From the results presented in this table, we can observe that the adaptive methods are slower than the classical ones, with this difference being greater as the number of variables and observations increases. For fuzzy adaptive methods, the global adaptive-sum version had the longest time in most datasets. While the local adaptive versions were faster. As for hard adaptive methods, the global adaptive-sum version is the slowest for datasets with 4 variables. While for datasets with 2 variables, the global adaptive methods present a faster time.

4.5 EXPERIMENTS ON REAL DATASETS

Experiments with real data were carried out with the objective of illustrating the practical use of the proposed methods addressed in this work. The real datasets chosen were obtained from the UCI Machine Learning Repository [2] (ASUNCION; NEWMAN, 2007). The variables that represent the class to which an object belongs have been ignored during the clustering. Four datasets were selected in which the variables are exclusively qualitative and, with the purpose of example, binary or ordinal.

4.5.1 Results and analysis

4.5.1.1 Congressional Voting Records

The original database (SCHLIMMER, 1987) of this experiment contains 435 US congressmen ranked according to their political position: Democrat (class 1) or Republican (class 2). The objective of this application is to trace the voting profile of one and another class, based on 16

[2] Available at https://archive.ics.uci.edu/ml/index.php.

questions. Congressmen evaluated them and voted against or in favor of the subject matter. Thus, all variables contain only two categories: yes or no, that is, DOM$(A_2) = \cdots =$ DOM$(A_{17}) = 2$.

The metadata of this dataset is presented below:

- **Number of observations**: 435;
- **Class distribution**:

  Class 1: 267 (61%);

  Class 2: 168 (39%).
- **Number of variables**: a class indicator variable and 16 more;
- **Variables description**

  A_1. class name

  A_2. handicapped-infants

  A_3. water-project-cost-sharing

  A_4. adoption-of-the-budget-resolution

  A_5. physician-fee-freeze

  A_6. el-salvador-aid

  A_7. religious-groups-in-schools

  A_8. anti-satellite-test-ban

  A_9. aid-to-nicaraguan-contras

  A_{10}. mx-missile

  A_{11}. immigration

  A_{12}. synfuels-corporation-cutback

  A_{13}. education-spending

  A_{14}. superfund-right-to-sue

  A_{15}. crime

  A_{16}. duty-free-exports

  A_{17}. export-administration-act-south-africa

Table 12 shows the evaluation metrics results obtained for the congressional dataset. There is a slight performance difference for the hard and fuzzy algorithm versions when including an adaptive distance by the observed values of the ARI. Showing higher values for the global adaptive-product and local adaptive-sum distances. The fuzzy algorithm versions showed no performance difference by the internal indexes.

Table 12 – Algorithm evaluation metrics results for congressional voting records dataset

Version	ARI	S	PC	PE	MPC
k-modes	0.5034	0.4874	-	-	-
fuzzy c-modes	0.5300	-	0.6728	0.4976	0.3457
k-modes with local adaptive-sum	0.5166	0.4854	-	-	-
k-modes with local adaptive-product	0.5100	0.4873	-	-	-
k-modes with global adaptive-sum	0.5166	0.4870	-	-	-
k-modes with global adaptive-product	0.5232	0.4883	-	-	-
fuzzy c-modes with local adaptive-sum	0.5435	-	0.6728	0.4976	0.3457
fuzzy c-modes with local adaptive-product	0.5300	-	0.6728	0.4976	0.3457
fuzzy c-modes with global adaptive-sum	0.5367	-	0.6728	0.4976	0.3457
fuzzy c-modes with global adaptive-product	0.5435	-	0.6728	0.4976	0.3457

Source: The author (2023)

4.5.1.2 Hayes-Roth

This dataset consists of a list of observations evaluated by 3 variables: age, educational level, and marital status. The dataset was originally considered in (HAYES-ROTH; HAYES-ROTH, 1977).

The metadata of this dataset is presented below:

- **Number of observations**: 132;
- **Class distribution**:

  Class 1: 51 (39%);

  Class 2: 51 (39%);

  Class 3: 30 (22%).
- **Number of variables**: a class indicator variable and 3 more;
- **Variables description**

  A_1. class name, where DOM$(A_1) = 3$

  A_2. age, where DOM$(A_2) = 4$

A_3. educational-level, where DOM$(A_3) = 4$

A_4. marital-status, where DOM$(A_4) = 4$

Table 13 shows the evaluation metrics results obtained for the Hayes-Roth dataset. The ARI values showed, for both classical and proposed hard and fuzzy versions, the absence of pattern matching in the clustering. For hard versions, the S values showed that the classical version performs better than the proposed ones. On the other hand, the fuzzy proposed algorithms performed better than the classical one for all internal metrics. The fuzzy version with local adaptive-sum distance has the best performance.

Table 13 – Algorithm evaluation metrics results for Hayes-Roth dataset

Version	ARI	S	PC	PE	MPC
k-modes	-0.0139	0.3870	-	-	-
fuzzy c-modes	-0.0120	-	0.5210	0.7925	0.2820
k-modes with local adaptive-sum	0.0944	0.243	-	-	-
k-modes with local adaptive-product	0.0891	0.180	-	-	-
k-modes with global adaptive-sum	-0.0085	0.2740	-	-	-
k-modes with global adaptive-product	-0.0145	0.2890	-	-	-
fuzzy c-modes with local adaptive-sum	0	-	1	0	1
fuzzy c-modes with local adaptive-product	0.0150	-	0.7769	0.3570	0.6653
fuzzy c-modes with global adaptive-sum	-0.0041	-	0.7751	0.3596	0.6626
fuzzy c-modes with global adaptive-product	-0.0066	-	0.7751	0.3596	0.6626

Source: The author (2023)

4.5.1.3 Car Evaluation

This dataset consists of a list of cars, separated into 4 classes, where there is a classification based on the purchase condition of the vehicle (BOHANEC; RAJKOVIC, 1988), evaluated by 6 variables considered helpful when decision-making about buying a car.

The metadata of this dataset is presented below:

- **Number of observations**: 1,728;
- **Class distribution**:

Class 1: 1,210 (70%);

Class 2: 384 (22%);

Class 3: 69 (4%);

Class 4: 65 (4%).

- **Number of variables**: a class indicator variable and 6 more;
- **Variables description**

A_1. class name, where DOM$(A_1) = 4$

A_2. buying, where DOM$(A_2) = 4$

A_3. maint, where DOM$(A_3) = 4$

A_4. doors, where DOM$(A_4) = 4$

A_5. persons, where DOM$(A_5) = 3$

A_6. lug-boot, where DOM$(A_6) = 3$

A_7. safety, where DOM$(A_7) = 3$

Table 14 shows the evaluation metrics results obtained for the car evaluation dataset. The ARI and S values showed, for both classical and proposed hard and fuzzy versions, the absence of pattern matching in the clustering. The same is observed for the fuzzy algorithm versions, where the PE values reached the theoretical upper bound.

Table 14 – Algorithm evaluation metrics results for car evaluation dataset

Version	ARI	S	PC	PE	MPC
k-modes	0.0118	0.1055	-	-	-
fuzzy c-modes	0.0748	-	0.2705	1.3480	0.0273
k-modes with local adaptive-sum	-0.1119	-0.0109	-	-	-
k-modes with local adaptive-product	0.0222	0.1024	-	-	-
k-modes with global adaptive-sum	0.0318	0.0983	-	-	-
k-modes with global adaptive-product	0.0114	0.1099	-	-	-
fuzzy c-modes with local adaptive-sum	0.0273	-	0.2714	1.3460	0.0286
fuzzy c-modes with local adaptive-product	0.0452	-	0.2699	1.3490	0.0266
fuzzy c-modes with global adaptive-sum	0.0757	-	0.2705	1.3480	0.0273
fuzzy c-modes with global adaptive-product	0.0736	-	0.2705	1.3480	0.0273

Source: The author (2023)

4.5.1.4 Balance Scale

This dataset was generated to model the experimental results of cognitive tests (SIEGLER, 1976) with weights and distances, aimed at children. The study used a scale, in which each experiment contained different blocks of weights and distances between them, in each part of the balance. The children were asked to predict the outcome of placing certain numbers of weights at various distances to the left or right of the scale. Therefore, we obtained three classes, which vary depending on the actual balance result: class 1 (on the right), class 2 (left) and class 3 (balanced). All variables have a domain equal to 5, except the variable A_1, which represents the number of classes.

The metadata of this dataset is presented below:

- **Number of observations**: 625;
- **Class distribution**:

  Class 1: 288 (46%);

  Class 2: 288 (46%);

  Class 3: 49 (8%).
- **Number of variables**: a class indicator variable and 4 more;
- **Variables description**

  A_1. class name

  A_2. left-weight

  A_3. left-distance

  A_4. right-weight

  A_5. right-distance

Table 15 shows the evaluation metrics results obtained for the balance scale dataset. The S values showed no difference between the classical and proposed hard algorithm versions, while the ARI values showed that the classical method had a better performance than the proposed. The internal indexes showed no performance difference for the fuzzy algorithm versions.

Table 15 – Algorithm evaluation metrics results for balance scale dataset

Version	ARI	S	PC	PE	MPC
k-modes	0.1582	0.1470	-	-	-
fuzzy c-modes	0.0265	-	0.3770	1.0329	0.0655
k-modes with local adaptive-sum	0.0677	0.1523	-	-	-
k-modes with local adaptive-product	0.0651	0.1430	-	-	-
k-modes with global adaptive-sum	0.0466	0.1564	-	-	-
k-modes with global adaptive-product	0.1030	0.1457	-	-	-
fuzzy c-modes with local adaptive-sum	0.0090	-	0.3768	1.0327	0.0652
fuzzy c-modes with local adaptive-product	0.0020	-	0.3789	1.0298	0.0683
fuzzy c-modes with global adaptive-sum	0.0013	-	0.3770	1.0329	0.0655
fuzzy c-modes with global adaptive-product	0.0013	-	0.3770	1.0329	0.0655

Source: The author (2023)

Table 16 – Statistical tests comparing K-Modes method for real datasets

Dataset	**Statistical test**			
Congressional Voting Records	$H_0 : \mu_1 \geq \mu_3$	$H_0 : \mu_1 \geq \mu_4$	$H_0 : \mu_1 \geq \mu_5$	$H_0 : \mu_1 \geq \mu_6$
	$H_1 : \mu_1 < \mu_3$	$H_1 : \mu_1 < \mu_4$	$H_1 : \mu_1 < \mu_5$	$H_1 : \mu_1 < \mu_6$
	(0.7444)	(0.4380)	(0.9943)	(0.7150)
	Not Reject H_0	Not Reject H_0	Not Reject H_0	Not Reject H_0
Hayes-Roth	$H_0 : \mu_1 \geq \mu_3$	$H_0 : \mu_1 \geq \mu_4$	$H_0 : \mu_1 \geq \mu_5$	$H_0 : \mu_1 \geq \mu_6$
	$H_1 : \mu_1 < \mu_3$	$H_1 : \mu_1 < \mu_4$	$H_1 : \mu_1 < \mu_5$	$H_1 : \mu_1 < \mu_6$
	(1)	(1)	(0.3416)	(0.2348)
	Not Reject H_0	Not Reject H_0	Not Reject H_0	Not Reject H_0
Car Evaluation	$H_0 : \mu_1 \geq \mu_3$	$H_0 : \mu_1 \geq \mu_4$	$H_0 : \mu_1 \geq \mu_5$	$H_0 : \mu_1 \geq \mu_6$
	$H_1 : \mu_1 < \mu_3$	$H_1 : \mu_1 < \mu_4$	$H_1 : \mu_1 < \mu_5$	$H_1 : \mu_1 < \mu_6$
	(1)	(0.7638)	(0.1391)	(0.0928)
	Not Reject H_0	Not Reject H_0	Not Reject H_0	Not Reject H_0
Balance Scale	$H_0 : \mu_1 \geq \mu_3$	$H_0 : \mu_1 \geq \mu_4$	$H_0 : \mu_1 \geq \mu_5$	$H_0 : \mu_1 \geq \mu_6$
	$H_1 : \mu_1 < \mu_3$	$H_1 : \mu_1 < \mu_4$	$H_1 : \mu_1 < \mu_5$	$H_1 : \mu_1 < \mu_6$
	(1)	(1)	(0.2587)	(0.5303)
	Not Reject H_0	Not Reject H_0	Not Reject H_0	Not Reject H_0

Source: The author (2023)

In order to compare these methods, Student's t tests for independent samples with 5% of significance are performed. Tables 16 and 17 give the values of the p-value. In these tables, μ is the same as defined in the synthetic datasets section. Values in these tables show that the fuzzy local adaptive versions were superior to the classical fuzzy version for Car Evaluation, Balance Scale, and Hayes-Roth datasets. This late presents superior performance for all fuzzy adaptive versions. While the hard adaptive versions weren't superior to the classical hard version for all datasets.

Table 17 – Statistical tests comparing Fuzzy C-Modes method for real datasets

Dataset	**Statistical test**			
Congressional Voting Records	$H_0 : \mu_2 \geq \mu_7$	$H_0 : \mu_2 \geq \mu_8$	$H_0 : \mu_2 \geq \mu_9$	$H_0 : \mu_2 \geq \mu_{10}$
	$H_1 : \mu_2 < \mu_7$	$H_1 : \mu_2 < \mu_8$	$H_1 : \mu_2 < \mu_9$	$H_1 : \mu_2 < \mu_{10}$
	(1)	(1)	(0.9280)	(2.2×10^{-16})
	Not Reject H_0	Not Reject H_0	Not Reject H_0	Reject H_0
Hayes-Roth	$H_0 : \mu_2 \geq \mu_7$	$H_0 : \mu_2 \geq \mu_8$	$H_0 : \mu_2 \geq \mu_9$	$H_0 : \mu_2 \geq \mu_{10}$
	$H_1 : \mu_2 < \mu_7$	$H_1 : \mu_2 < \mu_8$	$H_1 : \mu_2 < \mu_9$	$H_1 : \mu_2 < \mu_{10}$
	(2.2×10^{-16})	(2.2×10^{-16})	(2.2×10^{-16})	(2.2×10^{-16})
	Reject H_0	Reject H_0	Reject H_0	Reject H_0
Car Evaluation	$H_0 : \mu_2 \geq \mu_7$	$H_0 : \mu_2 \geq \mu_8$	$H_0 : \mu_2 \geq \mu_9$	$H_0 : \mu_2 \geq \mu_{10}$
	$H_1 : \mu_2 < \mu_7$	$H_1 : \mu_2 < \mu_8$	$H_1 : \mu_2 < \mu_9$	$H_1 : \mu_2 < \mu_{10}$
	(2.8×10^{-9})	(0.0001)	(0.9834)	(1)
	Reject H_0	Reject H_0	Not Reject H_0	Not Reject H_0
Balance Scale	$H_0 : \mu_2 \geq \mu_7$	$H_0 : \mu_2 \geq \mu_8$	$H_0 : \mu_2 \geq \mu_9$	$H_0 : \mu_2 \geq \mu_{10}$
	$H_1 : \mu_2 < \mu_7$	$H_1 : \mu_2 < \mu_8$	$H_1 : \mu_2 < \mu_9$	$H_1 : \mu_2 < \mu_{10}$
	(2.6×10^{-6})	(0.0012)	(0.7168)	(0.8401)
	Reject H_0	Reject H_0	Not Reject H_0	Not Reject H_0

Source: The author (2023)

5 CONCLUSION

In this chapter, conclusions of this Fuzzy C-Modes clustering algorithm with variable weighting are presented, as well as future work directions.

5.1 CONTRIBUTIONS

It was proposed a new fuzzy clustering algorithm for qualitative data based on adaptive distances as dissimilarities measure to compose the objective function. The presented theoretical results extend the clustering algorithms based on adaptive distances already known in the literature when quantitative data is available. In comparison with the conventional methods, the proposed algorithm presented a better performance on the evaluation metrics by adding to its distance function a weight associated with the variable. For datasets with a higher level of dispersion of the variable and superposition of classes, the local adaptive distances had a superior performance since they have different weights for each variable across the clusters.

Experiments on synthetic datasets demonstrated that the Fuzzy C-Modes algorithm versions with variable weighting presented higher modified partition coefficient values and lower partition entropy values for almost all datasets, except for the ones with two variables only. Also, the fuzzy algorithm version with local adaptive sum distance achieved the theoretical bound of the evaluation metrics when the level of superposition of classes was at the maximum for both overlapping and non-overlapping datasets. Differently, the K-Modes algorithm versions with variable weighting had a lower performance when compared to the original algorithm. The internal index presented values below those considered by the literature that would deviate the partition found by the algorithm from a random matching. Whereas the external index suggested an associated matching result from the known label and clustering partition.

The algorithms were also applied to real datasets, from the UCI Machine Learning repository, with the objective of illustrating the practical use of the proposed methods with different types of qualitative data. In particular, the resulting fuzzy partitions obtained from the Congressional and Hayes- Roth datasets presented a better performance for the adaptive distances algorithm versions, according to external and internal indexes respectively.

The generation of the algorithms, creation of random data, and evaluation indexes were implemented in a library in the programming language R that was made accessible with an open-source repository.

5.2 FUTURE WORKS

The nature of the data may vary a lot, depending on the research domain, and the existence of datasets with different types of qualitative is likely. With that in mind, the extension of the proposed algorithms with mixed distance functions to be able to handle binary, nominal, and ordinal variables at once is a point of improvement in this work.

Moreover, another possible improvement point is to evaluate the insertion of a weight in the membership wherein each object, cluster, and variable has a suitable weight. (PIMENTEL; SOUZA, 2016) proposed this approach as the multivariate with weighting for quantitative data. This could lead to better resulting fuzzy partitions, reducing the ambiguity in the membership values.

REFERENCES

ASUNCION, A.; NEWMAN, D. *UCI machine learning repository*. [S.l.]: Irvine, CA, USA, 2007.

BEZDEK, J. C. Mathematical models for systematics and taxonomy. In: FREEMAN. *Proceedings of the 8th International Conference on Numerical Taxonomy, 1975*. [S.l.], 1975.

BEZDEK, J. C. Pattern recognition with fuzzy objective function algorithms. Springer, 1981.

BOHANEC, M.; RAJKOVIC, V. Knowledge acquisition and explanation for multi-attribute decision making. In: AVIGNON FRANCE. *8th intl workshop on expert systems and their applications*. [S.l.], 1988. p. 59–78.

CHAN, E. Y.; CHING, W. K.; NG, M. K.; HUANG, J. Z. An optimization algorithm for clustering using weighted dissimilarity measures. *Pattern recognition*, Elsevier, v. 37, n. 5, p. 943–952, 2004.

DEBORAH, L. J.; BASKARAN, R.; KANNAN, A. A survey on internal validity measure for cluster validation. *International Journal of Computer Science Engineering Survey*, v. 1, n. 2, p. 85–102, 2010.

DIDAY, E. Classification automatique avec distances adaptatives. *RAIRO Informatique Computer Science*, v. 11, n. 4, p. 329–349, 1977.

DIDAY, E.; SIMON, J. Clustering analysis. *Digital pattern recognition*, Springer, p. 47–94, 1976.

DUNN, J. C. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. Taylor & Francis, 1973.

FERREIRA, M. R.; CARVALHO, F. D. A. D. Kernel fuzzy c-means with automatic variable weighting. *Fuzzy Sets and Systems*, Elsevier, v. 237, p. 1–46, 2014.

FRIGUI, H.; NASRAOUI, O. Unsupervised learning of prototypes and attribute weights. *Pattern recognition*, Elsevier, v. 37, n. 3, p. 567–581, 2004.

GOSHTASBY, A. A. *Image Registration: Principles, Tools and Methods*. [S.l.]: Springer Publishing Company, Incorporated, 2012.

GOWER, J. C.; LEGENDRE, P. Metric and euclidean properties of dissimilarity coefficients. *Journal of classification*, Springer, v. 3, p. 5–48, 1986.

GUPTA, A. K.; NADARAJAH, S. *Handbook of beta distribution and its applications*. [S.l.]: CRC press, 2004.

HAMMERSLEY, J. *Monte carlo methods*. [S.l.]: Springer Science & Business Media, 2013.

HARTIGAN, J. A.; WONG, M. A. A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, v. 28, n. 1, p. 100–108, 1979.

HAYES-ROTH, B.; HAYES-ROTH, F. Concept learning and the recognition and classification of exemplars. *Journal of Verbal Learning and Verbal Behavior*, v. 16, n. 3, p. 321–338, 1977.

HE, Z.; XU, X.; DENG, S. Scalable algorithms for clustering large datasets with mixed type attributes. *International Journal of Intelligent Systems*, Wiley Online Library, v. 20, n. 10, p. 1077–1089, 2005.

HORE, P.; HALL, L. O.; GOLDGOF, D. B. Single pass fuzzy c means. In: IEEE. *2007 IEEE International Fuzzy Systems Conference*. [S.l.], 2007. p. 1–7.

HRUSCHKA, E.; CASTRO, L. de; CAMPELLO, R. Evolutionary algorithms for clustering gene-expression data. In: . [S.l.: s.n.], 2004. p. 403–406.

HUANG, Z. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, Springer, v. 2, n. 3, p. 283–304, 1998.

HUANG, Z.; NG, M. A fuzzy k-modes algorithm for clustering categorical data. *IEEE Transactions on Fuzzy Systems*, v. 7, n. 4, p. 446–452, 1999.

HUBERT, L.; ARABIE, P. Comparing partitions. *Journal of classification*, Springer, v. 2, n. 1, p. 193–218, 1985.

JACCARD, P. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bull Soc Vaudoise Sci Nat*, v. 37, p. 547–579, 1901.

JAIN, A. K.; DUBES, R. C. *Algorithms for clustering data*. [S.l.]: Prentice-Hall, Inc., 1988.

JAMES, B. R. *Probabilidade: um curso em nível intermediário*. [S.l.: s.n.], 2015.

JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. *An introduction to statistical learning*. [S.l.]: Springer, 2013. v. 112.

KAMOLOV, A. A.; PARK, S. Prediction of depth of seawater using fuzzy c-means clustering algorithm of crowdsourced sonar data. *Sustainability*, v. 13, n. 11, 2021. ISSN 2071-1050.

KAUFMAN, L.; ROUSSEEUW, P. J. *Finding groups in data: an introduction to cluster analysis*. [S.l.]: John Wiley & Sons, 2009.

KELLER, A.; KLAWONN, F. Fuzzy clustering with weighting of data variables. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, World Scientific, v. 8, n. 06, p. 735–746, 2000.

LIMA, R. F. de. *Topologia e Análise no Espaço Rn*. [S.l.: s.n.], 2015.

LIU, Y.; LI, Z.; XIONG, H.; GAO, X.; WU, J. Understanding of internal clustering validation measures. In: *2010 IEEE International Conference on Data Mining*. [S.l.: s.n.], 2010. p. 911–916.

MACQUEEN, J. Some methods for classification and analysis of multivariate observations. In: *Proc. 5th Berkeley Symposium on Math., Stat., and Prob*. [S.l.: s.n.], 1965. p. 281.

MAESSCHALCK, R. D.; JOUAN-RIMBAUD, D.; MASSART, D. The mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, v. 50, n. 1, p. 1–18, 2000.

MANLY, B. F.; ALBERTO, J. A. N. *Multivariate statistical methods: a primer*. [S.l.]: Chapman and Hall/CRC, 2016.

MINGOTI, S. A.; MATOS, R. A. Clustering algorithms for categorical data: a monte carlo study. *International Journal of Statistics and Applications*, v. 2, n. 4, p. 24–32, 2012.

MORETTIN, P. A.; BUSSAB, W. O. *Estatística básica*. [S.l.]: Saraiva Educação SA, 2017.

NETTER, J.; WASSERMAN, W.; KUTNER, M. Applied linear regression models: Regression, analysis of variance, and experimental designs. *Holmwood, IL: Irwin*, 1989.

PIMENTEL, B. A.; SOUZA, R. M. de. Multivariate fuzzy c-means algorithms with weighting. *Neurocomputing*, Elsevier, v. 174, p. 946–965, 2016.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2023.

RALAMBONDRAINY, H. A conceptual version of the k-means algorithm. *Pattern Recognition Letters*, Elsevier, v. 16, n. 11, p. 1147–1157, 1995.

RAND, W. M. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, Taylor & Francis, v. 66, n. 336, p. 846–850, 1971.

RODRíGUES, S. I. R. A fuzzy partitional clustering algorithm with adaptative euclidean distance and entropy regularization. *MSc dissertation*, Universidade Federal de Pernambuco, Brasil, 2018.

ROSS, S. *A first course in probability*. [S.l.]: Pearson, 2010.

RUSPINI, E. H.; BEZDEK, J. C.; KELLER, J. M. Fuzzy clustering: A historical perspective. *IEEE Computational Intelligence Magazine*, v. 14, n. 1, p. 45–55, 2019.

SCHLIMMER, J. C. Concept acquisition through representational adjustment. *PhD thesis*, University of California, United States, 1987.

SIEGLER, R. S. Three aspects of cognitive development. *Cognitive psychology*, Elsevier, v. 8, n. 4, p. 481–520, 1976.

SISODIA, D.; SINGH, L.; SISODIA, S.; SAXENA, K. Clustering techniques: a brief survey of different clustering algorithms. *International Journal of Latest Trends in Engineering and Technology (IJLTET)*, Citeseer, v. 1, n. 3, p. 82–87, 2012.

SOKAL, R. R.; MICHENER, C. D. A statistical method for evaluating systematic relationships. *Multivariate statistical methods, among-groups covariation*, p. 269, 1975.

SOUZA, R. M. de; CARVALHO, F. d. A. D. Clustering of interval data based on city–block distances. *Pattern Recognition Letters*, Elsevier, v. 25, n. 3, p. 353–365, 2004.

STEWART, J. *Single Variable Calculus, Volume 2*. [S.l.]: Cengage Learning, 2012. ISBN 9781133419440.

TSAI, C.-Y.; CHIU, C.-C. Developing a feature weight self-adjustment mechanism for a k-means clustering algorithm. *Computational statistics & data analysis*, Elsevier, v. 52, n. 10, p. 4658–4672, 2008.

XU, R.; WUNSCH, D. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, v. 16, n. 3, p. 645–678, 2005.

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