mathematics

MDPI

Article

The α-Groups under Condorcet Clustering

Tarik Faouzi [1,*,†], Luis Firinguetti-Limone [1,†], José Miguel Avilez-Bozo [1,†] and Rubén Carvajal-Schiaffino [2,†]

[1] Departamento de Estadística, Universidad del Bío-Bío, Concepción 4051381, Chile; lfiringu@ubiobio.cl (L.F.-L.); jose.avilez1601@alumnos.ubiobio.cl (J.M.A.-B.)
[2] Departamento de Matemática y Ciencia de la Computación, Universidad de Santiago de Chile, Santiago 9170020, Chile; ruben.carvajal@usach.cl
* Correspondence: tfaouzi@ubiobio.cl
† These authors contributed equally to this work.

Abstract: We introduce a new approach to clustering categorical data: Condorcet clustering with a fixed number of groups, denoted α-Condorcet. As *k*-modes, this approach is essentially based on similarity and dissimilarity measures. The paper is divided into three parts: first, we propose a new Condorcet criterion, with a fixed number of groups (to select cases into clusters). In the second part, we propose a heuristic algorithm to carry out the task. In the third part, we compare α-Condorcet clustering with *k*-modes clustering. The comparison is made with a quality's index, accuracy of a measurement, and a within-cluster sum-of-squares index. Our findings are illustrated using real datasets: the feline dataset and the US Census 1990 dataset.

Keywords: *k*-modes; Condorcet clustering; categorical data; quality index; within-cluster sum-of-squares index

MSC: 62H30; 91C20

Citation: Faouzi, T.; Firinguetti-Limone, L.; Avilez-Bozo, J.M.; Carvajal-Schiaffino, R. The α-Groups under Condorcet Clustering. *Mathematics* **2022**, *10*, 718. https://doi.org/10.3390/math10050718

Academic Editors: Giada Adelfio, Elvira Romano and Andrzej Sokołowski

Received: 29 December 2021
Accepted: 22 February 2022
Published: 24 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

In 1909, Jan Czekanowski proposed the first clustering method [1]. This kind of method has become fundamental to many branches of statistics and social sciences. With clustering, we seek to classify a set of objects into relatively homogeneous groups, which are usually referred to as clusters. That is, for a given dataset, the goal of a cluster analysis is to define a set of clusters and to assign to each of them the observations that some distances or similarity measures are close to each other, while observations between clusters are away from each other. There are increasing discussions surrounding the best clustering method, as one can gather from the large number of review articles (see for example [2–6]). Many authors have proposed different clustering algorithms, and most techniques and algorithms deal with quantitative data. However, categorical data are common, particularly in the social sciences [7–12]. As such, applying clustering methods to categorical data is important, and methods have been proposed to deal with these types of data. An extension of the *k*-means approach to clustering, the *k*-modes clustering [13], is prominent among these. In this paper, we present a novel algorithm to group qualitative data: an extension of Condorcet clustering [14]. We demonstrate that, with a fixed number of clusters, a unique partition of the data could be achieved by maximizing a Condorcet´s criterion [14]. We developed a heuristic algorithm that proved to be very useful. Moreover, an adjustment rate index was used to evaluate the quality of the partition of *k*-modes and α-Condorcet on the basis of real datasets. The rest of the paper is organized as follows: in Section 2, we present some related work. In Section 3, we introduce some relevant concepts and definitions. In Section 4, we present some theoretical results. The clustering algorithm is presented in Section 5. Using real data, in Section 6, we compare α-Condorcet clustering to *k*-modes clustering. Finally, our concluding remarks are given in Section 7.

2. Related Work

Clusters may be regarded as crisp or fuzzy. In fuzzy clustering, an observation may belong to more than one cluster with given probabilities, whereas in crisp clustering, an observation belongs to one and only one cluster. Most clustering algorithms, but not all, may be classified into two categories: partitioning and hierarchical algorithms.

k-means is prominent among the partitional methods, and is one of the most popular techniques for clustering quantitative data [15–18]. Given a set of n multivariate observations, $(x_1, x_2, \cdots, x_n)$ where x_i is a d dimensional vector, the *k*-means algorithm partitions the data into $k \leq n$ clusters, $\mathbf{S} = (S_1, S_2, \ldots, S_k)$, such that the sum of squares within each cluster is minimized. That is, *k*-means seeks to minimize:

$$\underset{\mathbf{S}}{\operatorname{argmin}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in \mathbf{S}_i} \|\mathbf{x}_j - \mu_i\|^2, \tag{1}$$

where μ_i is the mean of point in S_i. This algorithm is fast and easy to implement [16,18]. Once the number of clusters is defined, this method chooses, at random, k points in the attribute space as initial values. After that, observations are assigned to the closest cluster and the centroids are updated. Because the algorithm does not guarantee convergence to the global optimum and since it is usually a fast algorithm, it is common to run it multiple times with different starting conditions. This method may, however, be badly affected by outliers.

Several methods have been proposed to deal with qualitative data or mixed data. The *k*-modes and *k*-prototype methods are prominent among these, as proposed by Huang [13], which are extensions to the *k*-means (see Table 1).

k-modes, in particular, is the *k*-means method, but with the Euclidean distance metric substituted by a simple matching dissimilarity measure, where the centers of the clusters are represented by their modes instead of the means. To introduce *k*-modes, let X and Z be two objects described by n categorical attributes. Then, a simple dissimilarity measure between these objects is the total number of mismatches of the corresponding values of the attributes of the two objects. That is

$$d_1(X, Z) = \sum_{j=1}^{n} \delta(x_j, z_j),$$

where

$$\delta(x_j, z_j) = \begin{cases} 0 & if \quad x_j = z_j, \\ 1 & if \quad x_j \neq z_j. \end{cases} \tag{2}$$

Let $\mathbf{S} = (X_1, X_2, \ldots, X_m)$ be a set of m objects described by n categorical attributes denoted by v_j, $j = 1, 2, \ldots, n$. Then a mode of $\mathbf{S}$ is a vector $Q = [q_1, q_2, \ldots, q_n]$ that minimizes

$$D(S, Q) = \sum_{i=1}^{m} d_1(X_i, Q),$$

Q not necessarily an object of $\mathbf{S}$. Finally, the *k*-modes algorithm partitions the set of m objects described by n categorical attributes into k clusters, S_i, $i = 1, \ldots, k$, by minimizing the following expression:

$$D(S, Q) = \sum_{i=1}^{k} \sum_{X \in S_i} d_1(X, Q_i),$$

where Q_i is the mode of cluster S_i. For a survey of *k*-modes see [19]. For a different approach to clustering categorical data, see [20].

Although the *k*-modes method has the advantage of being scalable to very large datasets, the final solution may be influenced by the initialization criterion of using random initial modes as centers. A number of suggestions have been made to overcome the

shortcomings of *k*-modes. For example, Lakshmi [21] propose a different algorithm to overcome the initialization problem of *k*-modes. Moreover, Dorman [22] adapt the Hartigan algorithm for *k*-means and develop several approaches to selects the initial centroids to improve the efficiency of *k*-modes. Two other approaches to initialize the *k*-modes algorithm are given in [23,24]. A fuzzy version of the *k*-modes algorithm is proposed by Huang [25] to improve the performance of *k*-modes. Other fuzzy versions of the *k*-modes method are given in [26–28]. Ng [29] modify a simple matching dissimilarity measure to obtain clusters with intra-similarity and describe extensions of *k*-modes to cluster efficiently large categorical datasets. A different dissimilarity measure is provided by Cao [30].

For different approaches to clustering categorical data, see [20,31–33].

Table 1. Some classical methods of clustering for categorical or mixed data.

Method	Data Type	Metric
k-modes	Categorical data	Measure of similarity
k-prototype	Mixed data	Huang cost function
Condorcet	Categorical data	Measure of similarity

Besides the *k*-modes algorithm, Huang [25] also proposes the *k*-prototype, an algorithm that integrates the *k*-means and *k*-modes algorithms to cluster mixed types of objects. The dissimilarity between two mixed-type objects, X and Z, which are described by $v_1^r, v_2^r, \ldots, v_p^r, v_{p+1}^c, \ldots, v_n^c$, may be measured by

$$d_2(X,Z) = \sum_{j=1}^{p}(x_j - z_j)^2 + \gamma \sum_{j=p+1}^{n} \delta(x_j, z_j). \tag{3}$$

Of course, the first term corresponds to the squared Euclidean distance, which is applied to the quantitative attributes and the second term is the simple matching dissimilarity measure, which is applied to the qualitative attributes. γ is a weight used to avoid favoring either type of attribute. Thus *k*-prototype seeks to minimize the following cost function:

$$P(W,\mathbf{Q}) = \sum_{l=1}^{k}\left(\sum_{i=1}^{n} w_{il} \sum_{j=1}^{p}(x_{ij} - q_{lj})^2 + \gamma \sum_{i=1}^{m} w_{il} \sum_{j=p+1}^{n} \delta(x_{ij}, q_{lj})\right), \tag{4}$$

where W is an $n \times k$ partition matrix with elements $w_{ij}, i = 1,2,\ldots,m \quad \text{and} \quad j = 1,2,\ldots,k$; $\mathbf{Q} = \{Q_1, Q_2, \ldots, Q_k\}$ is a set of objects in the same object domain.

Next, Marcotorchino [14], Michaud [34–36] were the first to propose a clustering method for categorical data using a dissimilarity measure. These authors developed the relational analysis theory, and introduced the relation aggregation problem in order to solve the Condorcet' s paradox in the voting system, and relate it to the similarity problem.

This approach consists of using pairwise comparisons and applying the simple majority decision rule. Indeed, aggregating equivalence relations using the simple majority decision rule guarantees optimal solutions under some constraints and without fixing *a priori* the number of groups. In our work, we used the approach introduced by Michaud and Marcotorchino, setting *a priori* the number of groups.

3. Materials and Methods

Let $N = \{v_1, \ldots, v_n\}$ be a set of n variables and $S = \{x_1, \ldots, x_m\}$ a set of m objects. Let C be a Condorcet matrix, with elements $c_{x_i x_j}$, corresponding to the number of variables for which x_i is similar to x_j, denoted by $x_i \overset{v_k}{\sim} x_j$, and $Y = (y_{x_i x_j})_{i,j=1}^{m}$ is a matrix, such that

$$y_{x_i x_j} = \begin{cases} 1 & if \quad x_i \sim x_j, \\ 0 & if \quad x_i \nsim x_j. \end{cases} \tag{5}$$

For two given objects, x_i and x_j, with $x_i \overset{v_k}{\sim} x_j$ we mean that x_i and x_j have the same value for the variable v_k with $k = 1, \ldots, n$, while $x_i \sim x_j$ means that x_i and x_j are similar.

In the relational analysis methodology, Marcotorchino [14] suggest the maximization of Condorcet´s criterion, under some restrictions, given by

$$f(Y) = \sum_{x_j \neq x_i} (c_{x_i x_j} y_{x_i x_j} + \bar{c}_{x_i x_j} \bar{y}_{x_i x_j}), \tag{6}$$

with $y_{x_i x_j} + \bar{y}_{x_i x_j} = 1$, $\bar{c}_{x_i x_j} + c_{x_i x_j} = n$ and $Y = (y_{x_i x_j})_{i=1,j=1}^{m}$ is a matrix that maximizes the function $f(\cdot)$ given in Equation (6). This matrix takes values 0 and 1.

Then, the model associated with the absolute global majority is defined by:

$$\mathcal{P} \quad \begin{cases} \max\limits_{Y} f(Y) \\ y_{x_i x_j} \in \{0, 1\} \\ y_{x_i x_j} + \bar{y}_{x_i x_j} = 1 \\ 0 \leq y_{x_i x_j} + y_{x_j x_k} - y_{x_i x_k} \leq 1, \end{cases} \tag{7}$$

where Y is the matrix of similarities. The first constraint represents the binarity, the second restriction represents the symmetry and the third restriction is the transitivity.

The following example explains how to obtain the matrix Y, which maximizes the Condorcet´s criterion under the restrictions given below.

Let E be a dataset that is composed of three items (x_1, x_2, x_3), with three qualitative variables v_1, v_2, v_3 being measured. The dataset is presented in Table 2.

Table 2. Example of dataset E.

	v_1	v_2	v_3
x_1	1	1	3
x_2	1	2	3
x_3	2	1	2

Using Table 2, we identify the matrix of Condorcet C, which is given by $C = \begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 0 \\ 1 & 0 & 3 \end{pmatrix}$.

Then, the possible solutions Y that satisfy the constraints are

$$Y_1 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, Y_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}, Y_3 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}, Y_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$
$$Y_5 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

Next, we compute the function $f(\cdot)$ for each matrix Y_i, $i = 1, 2, 3, 4, 5$. Indeed, we have $f(Y_1) = 23$, $f(Y_2) = 15$, $f(Y_3) = 19$, $f(Y_4) = 21$ and $f(Y_5) = 15$. We deduce that Y_1 maximizes the Condorcet´s criterion. Finally, we obtain the number of clusters, equal 2, and the clusters are $\{x_1, x_2\}$ and $\{x_3\}$.

Although, the proposed method of clustering does not require fixing the number of classes beforehand, there are instances where this is not convenient, such is the case in a psychometric analysis. In this paper, we take this point of view and assume that the number of clusters is given.

Therefore, in this paper, we fix the number of groups denoted by α, and focus on finding the solution of Equation (7), giving its algorithm and comparing its results to k-modes for some fixed value of α.

Next, let us denote by $S = \cup_{k=1}^{\alpha} S_k$ a partition with respect to the set of objects S. In this partition, the number of clusters is α.

Recall that the matrix C represents the similarities between pairs of objects that we want to cluster. Similarly, we introduce a matrix of dissimilarities between pairs of the same objects and we denote it by $\overline{C}$. Next, we define n categorical variables denoted by $v_k; k = 1, \ldots, n$, and let v_k^i be a modality of v_k assigned to object $i \in S$. Then, we write that each variable v_k is associated with a matrix C^k. As a consequence, we obtain

$$\sum_{k=1}^{n} C^k = C, \tag{8}$$

where the elements of matrix C^k are given by

$$c^k_{x_i x_j} = \begin{cases} 1 \text{ if } x_i \text{ and } x_j \text{ have the same modality of } v_k \\ 0 \text{ otherwise} \end{cases}. \tag{9}$$

By abuse of notation, we write $c_{x_i x_j} = c_{ij}$ and $y_{x_i x_j} = y_{ij}$.

Using Equation (8), the general terms of the collective relational matrix C are given by $c_{ij} = \sum_{k=1}^{n} c_{ij}^k$. Furthermore, we define the general terms of the collective relational matrix $\overline{C}$ as $\overline{c}_{ij} = \sum_{k=1}^{n} \overline{c}_{ij}^k$. Note that $\overline{c}_{ij}$ represents the number of variables for which x_i and x_j are not similar.

4. Main Theoretical Results

4.1. α-Condorcet Criterion Function

Now, we present our first important result: a new Condorcet criterion function.

Definition 1. *Let $(S_k)_{k\in\{1,\cdots,\alpha\}}$ be a partition of a set of objects S. We define a new Condorcet criterion function g as*

$$g(S;\alpha) = \sum_{k=1}^{\alpha} \sum_{1\le i,j\le m} \Big(c_{ij} S_{ik} S_{jk} + \overline{c}_{ij} (\overline{S}_{ik} S_{jk} + S_{ik} \overline{S}_{jk}) \Big).$$

with $i \in S_k$ if $S_{ik} = 1$ and $i \notin S_k$ if $S_{ik} = 0$.

Using Equations $-c_{ij} + n = \overline{c}_{ij}$ and $S_{ik} + \overline{S}_{ik} = 1$, the formula above can be rewritten as follows

$$g(S;\alpha) = \sum_{k=1}^{\alpha} \sum_{1\le i,j\le m} \Big((3c_{ij} - 2n) S_{ik} S_{jk} + \overline{c}_{ij} (S_{ik} + S_{jk}) \Big).$$

Knowing the exact number of groups, the following model allows to group the objects by similarity in the sense of having common characteristics. Then, we obtain the partition $S = \cup_{k=1}^{\alpha} S_k$ by maximizing the following function

$$\mathcal{P}' \quad \begin{cases} \max\limits_{S} g(S;\alpha) \\ S_{ik} \in \{0,1\} \\ \sum\limits_{i=1}^{m} S_{ik} \ge 1 \\ \sum\limits_{k=1}^{\alpha} S_{ik} = 1 \end{cases} \tag{10}$$

where

$$S_{ik} = \begin{cases} 1 \text{ if } x_j \in S_k, \\ 0 \text{ otherwise.} \end{cases} \tag{11}$$

Using the third restriction of $\mathcal{P}'$, the function g can be simplified to the following expression

$$g(S;\alpha) = \sum_{i,j=1}^{m} \left(\sum_{k=1}^{\alpha} (3c_{ij} - 2n) S_{ik} S_{jk} + 2\bar{c}_{ij} \right).$$

The next theorem ensures the existence of at least one solution to the problem given in Equation (10).

Theorem 1. *Let $S = \cup_{k=1}^{\alpha} S_k = \{x_1, \cdots, x_m\}$ with $S_i \cap S_j = \varnothing \ \forall \alpha \leq m$. Then, there exists at least a partition of S that maximizes $(\mathcal{P}')$.*

Proof of Theorem 1. We know that if the number of objects m is inferior to the number of clusters α, then no solution of $(\mathcal{P}')$ exists.

We now suppose that the number of objects is superior to the number of clusters α. Then, we have d possible partitions, where the parameter d is the number of partitions of the set of m objects into α clusters, which can be expressed as follows

$$d = \sum_{\substack{w_1+w_2+\cdots+w_\alpha=m, \\ w_1 \leq w_2 \leq \cdots \leq w_\alpha}} \frac{m!}{w_1! w_2! \cdots w_\alpha!},$$

where $w_i, i = 1, \cdots, \alpha$, is the number of objects in the cluster S_i. Furthermore, d is a positive integer. Finally, there exists at least a partition of S that maximizes $(\mathcal{P}')$. □

Theorem 2. *We assume that the dataset does not present the Condorcet' s paradox. Then, for some value of α , there exists a unique partition of S that maximizes $(\mathcal{P}')$.*

Proof of Theorem 2. To simplify the calculations, we consider $m = 3$. We suppose that the object x_1 is similar to x_2 and not similar to x_3, then, under the absence of Condorcet' s paradox, we conclude that x_2 is not similar to x_3. Finally, there exists a unique partition $(\{x_1, x_2\}, \{x_3\})$ that maximizes $(\mathcal{P}')$. □

Next, we give some axiomatic conditions studied by Michaud [37], presenting some axiomatic conditions verified by Condorcet´s rule that respond to K. Arrow´s impossibility theorem, presented below.

Theorem 3 (Theorem of K. Arrow)**.** *According to Arrow´s impossibility theorem, it is impossible to formulate a social ordering without violating one of the following conditions:*

1. *Non-dictatorship: the voter´s preference cannot represent a whole community. The wishes of multiple voters should be taken into consideration.*
2. *Pareto efficiency: unanimous individual preferences must be respected. If every voter prefers candidate A over candidate B, candidate A should win.*
3. *Independence of irrelevant alternatives: if a choice is removed, then the others order should not change. If candidate A ranks ahead of candidate B, candidate A should still be ahead of candidate B, even if a third candidate, candidate C, is removed from participation.*
4. *Unrestricted domain: voting must account for all individual preferences.*
5. *Social ordering: each individual should be able to order their choices in a connected and transitive relation.*

Then, Michaud [37] proved that the rule of Condorcet verifies some conditions given in the following Theorem. The following result concerns the verification of these conditions by the α-Condorcet method given in Equation (10).

Theorem 4 (Axiomatic conditions)**.** *In the context of similarity aggregation problems, the rule of Condorcet verifies the following conditions for some values of α:*

1. *Non-dictatorship condition: this condition means that no variable can, by itself, determine an item (individual or object) classification maximizing the Condorcet criterion.*
2. *Pareto pair unanimity condition: if all variables are presented in two items, then these two items must be found in the same cluster.*
3. *Condition of total neutrality: the classification obtained must be independent of the order of individuals or items or variables.*
4. *Condition of coherent union: if two disjointed sets of variables give the same partition, then the union of the two sets will give the same partition.*

4.2. Total Inertia

We now focus on the inertia or within-cluster sum-of-squares. First, we present some preliminaries. Let $(S_k)_{k=\{1,\cdots,\alpha\}}$ be a partition of S. We build a cloud of points in $\mathbb{R}^n$, denoted $N(S)$, in which each dimension corresponds to a category of the variable v_j, $j = 1, \cdots, n$. Let $N(S) = \{(A_i, \mu_i) : i \in S\}$ be a cloud of mass points, where A_i is the coordinate point of object x_i and μ_i is its corresponding mass. In general, the expression of within-cluster sum-of-squares is given by

$$I_w = \sum_{k}^{\alpha} \sum_{i \in S_k} \mu_i \|A_i - G_k\|^2, \tag{12}$$

where G_k is cloud's center of gravity of cluster S_k.

The within-cluster sum-of-squares is a measure of how the objects are similar in each cluster. However, this measure does not allow making decisions regarding the quality of the partition: for values of inertia close to zero, the quality of the partition is better.

Next, let $p_1, \cdots, p_n$ be n modalities, respectively, of variables $v_1, \cdots, v_n$ and let p a fixed parameter, such that $p = \sum_{k=1}^{n} p_k$. Then, we define $\hat{c}_{ij} = \sum_{k=1}^{p} \frac{\vartheta_{ik}\vartheta_{jk}}{\vartheta_{\cdot\, k}}$, where ϑ_{ik} is one if object i has a modality k and zero otherwise, and $\vartheta_{\cdot\, k}$ is the total of objects that have the same modality k.

In the next theorem, we present an expression for inertia given by [38,39]

Theorem 5. *A relational expression of within-cluster sum-of-squares is given by*

$$I_w = \frac{1}{n}\left(p - \sum_{i=1}^{m}\sum_{j=1}^{m} \frac{\hat{c}_{ij} y_{ij}}{y_{i\cdot}}\right), \tag{13}$$

Note that this expression does not require the number of clusters to be specified *a priori*.

The following result is an application of Equation (12) and Theorem 5.

Theorem 6. *A new relational expression of the within-cluster sum-of-squares, with the number of clusters α fixed, is given by*

$$I_w = \frac{1}{n}\left(p - \sum_{i=1}^{m}\sum_{j=1}^{m} \hat{c}_{ij} \sum_{k=1}^{\alpha} \frac{S_{ik} S_{jk}}{|S_k|}\right), \tag{14}$$

where $|S_k|$ is the cardinality of cluster S_k.

Proof of Theorem 6. To prove this result, we need to consider the chi-square metric to find and compute a closed form of the expression of the within-cluster sum-of-squares. First, we define some preliminary elements. Let k_{ij} be a general term given by

$$k_{ij} = \begin{cases} 1 \text{ if } x_i \text{ has the modality } j; \\ 0 \text{ otherwise,} \end{cases} \tag{15}$$

with $k_{i.} = \sum_{j=1}^{p} k_{ij}$, $k_{.j} = \sum_{i=1}^{m} k_{ij}$ and $k_{..} = \sum_{i=1}^{m} k_{i.}$.

Then, we have

$$I_w = \sum_{k=1}^{\alpha} \sum_{i \in S_k} \mu_i \|A_i - G_k\|^2 = \sum_{k=1}^{\alpha} \sum_{i \in S_k} \mu_i \sum_{j=1}^{p} \frac{k_{..}}{k_{.j}} (A_i^j - G_k^j)^2, \tag{16}$$

where $A_i^j = \frac{k_{ij}}{n}$, $G_k^j = \frac{1}{\nu_k} \sum_{i \in S_k} \mu_i A_i^j$ and $\nu_k = \sum_{i \in S_k} \mu_i = \frac{n_k}{m}$ with $n_k = |S_k|$ is the cardinality of class S_k.

Note that μ_i is the mass of each individual given by $\frac{k_{i.}}{\sum_i \sum_j k_{ij}} = \frac{n}{nm} = \frac{1}{m}$.

It follows that

$$\begin{aligned} I_w &= \sum_{k=1}^{\alpha} \sum_{i \in S_k} \frac{1}{m} \sum_{j=1}^{p} \frac{n \times m}{k_{.j}} \left(\frac{k_{ij}}{n} - \frac{m}{n_k \times m} \frac{\sum_i k_{ij}}{n} \right)^2 \\ &= \sum_{k=1}^{\alpha} \sum_{i \in S_k} \sum_{j=1}^{p} \frac{1}{n} \left(\frac{k_{ij}}{\sqrt{k_{.j}}} - \frac{1}{n_k} \frac{\sum_{i \in S_k} k_{ij}}{\sqrt{k_{.j}}} \right)^2 \\ &= \sum_{k=1}^{\alpha} \sum_{j=1}^{p} \frac{1}{n} \left(\frac{\sum_{i \in S_k} k_{ij}}{k_{.j}} - \frac{(\sum_{i \in S_k} k_{ij})^2}{\sqrt{n_k k_{.j}}} \right). \end{aligned} \tag{17}$$

Now, we compute both right hand terms of Equation (17). We have,

$$\sum_{k=1}^{\alpha} \sum_{j=1}^{p} \frac{1}{n} \frac{\sum_{i \in S_k} k_{ij}}{k_{.j}} = \frac{1}{n} \sum_{j=1}^{p} \frac{k_{.j}}{k_{.j}} = \frac{p}{n}, \tag{18}$$

and

$$\begin{aligned} \sum_{k=1}^{\alpha} \sum_{j=1}^{p} \frac{1}{n} \frac{(\sum_{i \in S_k} k_{ij})^2}{\sqrt{n_k k_{.j}}} &= \sum_{j=1}^{p} \frac{1}{n \times k_{.j}} \sum_{k=1}^{\alpha} \frac{1}{n_k} \sum_{i \in S_k} \sum_{i' \in S_k} k_{ij} k_{i'j} \\ &= \sum_{j=1}^{p} \frac{1}{n \times k_{.j}} \sum_{k=1}^{\alpha} \sum_{i=1}^{m} \sum_{i'=1}^{m} k_{ij} k_{i'j} \frac{S_{ik} S_{i'k}}{|S_k|} \\ &= \sum_{i=1}^{m} \sum_{i'=1}^{m} \frac{1}{n} \left(\sum_j \frac{k_{ij} k_{i'j}}{k_{.j}} \right) \sum_{k=1}^{\alpha} \frac{S_{ik} S_{i'k}}{|S_k|} \\ &= \frac{1}{n} \sum_{i=1}^{m} \sum_{j=1}^{m} \hat{c}_{ij} \sum_{k=1}^{\alpha} \frac{S_{ik} S_{jk}}{|S_k|}. \end{aligned} \tag{19}$$

Finally, we obtain

$$I_w = \frac{1}{n} \left(p - \sum_{i=1}^{m} \sum_{j=1}^{m} \hat{c}_{ij} \sum_{k=1}^{\alpha} \frac{S_{ik} S_{jk}}{|S_k|} \right). \tag{20}$$

□

We next introduce a quality index given by

$$Ca = \frac{\max_{S_1, \cdots, S_\alpha} g(S_1, \cdots, S_\alpha; \alpha)}{m^2 n}. \tag{21}$$

This index was given in [14], and measures the quality of the partition.

5. Algorithm

Several algorithms have been given in the past three decades to solve problem (7) by linear programming techniques, when the population under study is relatively small. Unfortunately, the classical linear programming techniques require many restrictions. In this case, the heuristic method has been adopted in order to process large amounts of data. Although these heuristic algorithms are fast, they do not always ensure an optimal solution.

Next, for all points $(x_i, x_j) \in S \times S$, we maximize the expression

$$\sum_{k=1}^{\alpha} \sum_{i,j=1}^{m} (3c_{ij} - 2n) S_{ik} S_{jk}.$$

Note that the term $2\bar{c}_{ij}$ is omitted because it is constant.

The above formula represents the series of links between objects x_i and x_j, denoted by $\mathcal{L}_{ij}$, and we write $\mathcal{L}_{ij} = 3c_{ij} - 2n$. Moreover, we denote the general link by $\mathcal{L} = 3C - 2n \times \mathbf{1}$, where $\mathbf{1}$ is a matrix with elements equal to 1.

The α-Condorcet clustering algorithm is illustrated by some steps given in Algorithm 1. Given a database D of m points in $\mathbb{R}^n$ and partition $S_1, \cdots, S_m$ of S, such that $S_k = \{x_k\}$, $k = 1, \cdots, m$.

Similar to the algorithm given by [40], we compute the following steps.

Algorithm 1. Heuristic Algorithm α-Condorcet.

Input α: Number of partitions
sc: Number of observations
$x_k, \quad k = 1, \cdots, sc$: observations
nv: Number of variables
$D_{sc \times nv}$: Feature Matrix

Output $S_1, \cdots, S_\alpha$: is a partition of α clusters

$C_{sc \times sc} \leftarrow$ GenCondorcetGM($D_{sc \times nv}$): the generation of the Condorcet matrix C
$F \leftarrow 2 \times C_{sc \times sc} - nv \times \mathbf{1}_{sc \times sc}$: where $\mathbf{1}_{sc \times sc}$ is a matrix of ones
$S_k \leftarrow \{x_k\}$
$Ini \leftarrow sc$
$P_1 \leftarrow \cup_{k=1}^{Ini} S_k$
for $j \leftarrow 1$ **to** Ini **do**
  for $i \leftarrow 1$ **to** Ini **do**
    if $(i \neq j)$ **then**
      $L_{i,j} \leftarrow C_{i,j}$
    endif
  endfor
  $K_j \leftarrow \text{Max}(L_{\cdot,j})$ where $L_{\cdot,j}$ is the largest value of the $j-th$ column of matrix L
endfor
  $Posa_2 \leftarrow FirstPosition(\text{Max}(K))$ is the position of the first occurrence of the largest value of vector K
  $S_{Pos_1} \leftarrow x_{Posa_1}, x_{Posa_2}$
  EliminateGroup(S_{Posa_2}): Elimination of the cluster $S_{Posa_2} = \{x_{Posa_2}\}$
  $Ini \leftarrow \alpha - 1$
  $a \leftarrow \text{Max}(K)$
  $C_{Posa_2, Posa_1} \leftarrow C_{Posa_1, Posa_2} \leftarrow Null$
  $b \leftarrow \text{Max}(C_{\cdot, Posa_2})$
  $Posb \leftarrow FirstPosition(\text{Max}(C_{\cdot, Posa_2}))$
  if $(a = b \wedge Ini > \alpha)$ **then**

```
23        G ← (GenCombi(S_posa1, x_posb1, F)): GenCombi gives the combination
                                  which maximizes the link F
24        r ← Card(S_Posa1): Card is the cardinality function
25        if (r < Card(G)) then
26            Ini ← Ini − 1
27            C_Posa2,Posa1 ← C_Posb1,Posb2 ← Null
28            S_Posa1 ← G, goto 5
29        else
30            goto 5
31        endif
32    endif
33    if (a > b ∧ Ini > α) then
34        goto 5
35    endif
36    if (Ini = α) then
37        return (S_1, ..., S_α)
38    endif
```

1. First, we find the largest value in each column of Condorcet´s matrix C, which corresponds to the number of characteristics that a pair of observations share. We then take the position of the largest value, denoted by a. In this case, we put those observations in the same group S_k, with k representing the kth column.
2. Next, we remove the value a in the matrix C and define b as the largest value of the kth column.
3. We distinguish some conditions:
   - 3.1 If $a > b$, we repeat the first point.
   - 3.2 If $a = b$, then the Condorcet´s criterion is applied. We group the elements that maximize the Condorcet criterion.
4. We repeat the process.
5. This process stops when the α groups are identified

5.1. Illustrative Example Using the Heuristic Algorithm

We now consider a dataset D, which is composed of six items $(x_1, x_2, \cdots, x_6)$ with three qualitative variables v_1, v_2, v_3 being measured. The dataset is presented in Table 3.

Table 3. Example of dataset D.

	v_1	v_2	v_3
x_1	1	1	3
x_2	1	2	3
x_3	2	1	2
x_4	1	1	3
x_5	2	1	1
x_6	2	1	2

Then, the Condorcet´s matrix C is given in the Table 4.

Table 4. The Condorcet´s matrix C.

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	-	2	1	3	1	1
x_2	2	-	0	2	0	0
x_3	1	0	-	1	2	3
x_4	3	2	1	-	1	1
x_5	1	0	2	1	-	2
x_6	1	0	3	1	2	-

In general, the diagonal of Condorcet´s matrix represents the number of variables measured in this group of observations, it also represents the maximum possible similarity that can occur between two observations x_i and x_j with $i, j = 1, 2, \cdots, 6$. For our heuristic algorithm, we replace the diagonal numbers by zero.

The goal of this example is to create our partition P such that $P = G_i \cup G_j$, $i, j = 1, \cdots, 6$, fixing the number of groups $\alpha = 2$. Before creating this partition, we suppose that each element represents a group and we write $G_i = \{x_i\}$ with $i = 1, \cdots, 6$. Let K be a vector whose elements are the maximum in each vector of Condorcet´s matrix. Then, we have $K = (3, 2, 3, 3, 2, 3)$. So, we identify the first maximum number of the vector K, called a, with $a = 3$, and its position $p = (4, 1)$, which represents the fourth position of the first column.

The following step is to put together the elements x_4 and x_1 in the same group G_1. In the first column, we eliminate the fourth value, and we write:

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	-	2	1	3	1	1
x_2	2	-	0	2	0	0
x_3	1	0	-	1	2	3
x_4	-	2	1	-	1	1
x_5	1	0	2	1	-	2
x_6	1	0	3	1	2	-

Computing the maximum of the first column, we obtain $b = 2$. Comparing the value of both parameters a and b, we find that $a > b$. Thus, we have $K = (3, 2, 3, 3, 2, 3)$. Then, the vector K is recalculated without considering columns 1 and 4, obtaining $K = (0, 2, 3, 0, 2, 3)$. So, we identify the first maximum number of the vector K with $a = 3$, and its position $p = (6, 3)$ that represents the last position of the third column. In the third column, we eliminate the sixth value, and we write:

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	-	2	1	3	1	1
x_2	2	-	0	2	0	0
x_3	1	0	-	1	2	3
x_4	-	2	1	-	1	1
x_5	1	0	2	1	-	2
x_6	1	0	-	1	2	-

Computing the maximum of the third column, we obtain $b = 2$. In this case, $a > b$. Again, the vector K is recalculated without considering columns $1, 3, 4$ and 6, and we have $K = (0, 2, 0, 0, 2, 0)$. The first maximum of vector K is in the second position, and we have $a = 2$, with position in the Condorcet matrix given by $p = (1, 2)$. The last position lead to add the element x_2 to the group G_1. We eliminate the first value of the second column, and write:

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	-	-	1	3	1	1
x_2	2	-	0	2	0	0
x_3	1	0	-	1	2	3
x_4	-	2	1	-	1	1
x_5	1	0	2	1	-	2
x_6	1	0	-	1	2	-

Next, the maximum of the second column is equal to 2, and we have $b = 2$ with a position $(4, 2)$. Both parameters a and b are equal. In this case, it is not necessary to carry out combinatorics between $\{x_1, x_4\}$ and $\{x_2\}$ because the element x_4 of position

$(4,2)$ belongs to G_1. Now, we define a new vector K without the second column given by $K = (0,0,0,0,2,0)$. The first maximum of vector K can be found in position $p = (3,5)$ and $p' = (6,5)$ meaning that the element x_5 can be in the first group or the second group. In this case, we must check which of the two partitions maximizes the Condorcet´s criterion function. After simple calculations, we deduce that x_5 belongs to G_2. Finally, we obtain two groups $G_1 = \{x_1, x_2, x_4\}$ and $G_2 = \{x_3, x_5, x_6\}$.

5.2. Advantage of Heuristic Algorithm

The main goals of this section are threefold. Firstly, we compare the partition quality, given in Equation (21), for the feline dataset using both exact and heuristic algorithms. Secondly, we use the inertia index, given in Equation (13), to compare the exact and heuristic algorithms. Finally, the execution time of the two methods is compared. For each step, we choose the first $m = 5, 7, 9$ felines of the feline dataset.

Table 5 shows that the use of the exact algorithm is ineffective due to some problems that occur when the sample size increases. Furthermore, the inertia and quality indexes of both heuristic and exact algorithms are almost equal.

Finally, observing the last column of Table 5, when the data size is equal to $m = 9$ and $\alpha = 4$, the execution of the exact algorithm takes 15.2 s, while the execution time of the heuristic algorithm is 0.36 s for $m = 30$ and $\alpha = 4$. Fixing again $m = 9$ and $\alpha = 6$, we observe that the execution time of the exact algorithm increases considerably, 586.95 s, compared to the execution time for $\alpha = 4, 5$. Furthermore, for $m = 30$, the quality and inertia indexes cannot be computed for exact algorithm; however, we know that the exact algorithm provide an optimal solution. Then, we can deduce that its quality index is at least as large as the quality of heuristic algorithm. Consequently, we confirm that the exact algorithm is computationally very expensive compared to the heuristic algorithm. Note that for the exact algorithm, the use of large datasets generates two important problems, the first is related to the execution time, while the second is concerned with the temporal storage space of data required by the programs being used at a particular moment (e.g., R-project).

Table 5. Time comparison between an exact algorithm and a heuristic algorithm using the feline dataset.

	Data Size	α	Quality Index	Inertia Index	Time (Second)
Exact algorithm	5	2	0.69	0.55	0.05
		3	0.62	0.32	0.03
		4	0.57	0.15	0.11
Heuristic algorithm		2	0.69	0.55	0.03
		3	0.61	0.28	0.03
		4	0.57	0.15	0.017
Exact algorithm	7	2	0.65	0.83	0.05
		3	0.64	0.56	0.18
		4	0.63	0.41	1.20
		5	0.59	0.26	4.90
		6	0.57	0.11	14.70
Heuristic algorithm		2	0.65	0.83	0.04
		3	0.64	0.56	0.03
		4	0.61	0.36	0.02
		5	0.59	0.21	0.03
		6	0.57	0.11	0.02

Table 5. *Cont.*

	Data Size	α	Quality Index	Inertia Index	Time (Second)
Exact algorithm	9	2	0.64	1.61	0.11
		3	0.64	1.18	0.99
		4	0.64	0.92	15.26
		5	0.62	0.77	111.80
		6	0.60	0.30	586.95
		7	0.58	0.22	2186.42
		8	0.56	0.11	6514.23
Heuristic algorithm		2	0.64	1.61	0.09
		3	0.64	1.18	0.06
		4	0.64	0.92	0.07
		5	0.60	0.65	0.05
		6	0.59	0.30	0.04
		7	0.58	0.22	0.05
		8	0.56	0.11	0.05
Exact algorithm	30	4	$\geq$0.67	NA	NA
Heuristic algorithm		4	0.67	0.90	0.36

6. Comparison between α-Condorcet and k-Modes

Firstly, in this section, we describe the experiments and their results. We ran our algorithm on feline datasets obtained from [14] and presented in Tables A1 and A2 from the Appendix A. We tested the performance of α-Condorcet clustering against the k-modes algorithm. Our algorithms were implemented in R language. The α-Condorcet algorithm was implemented according to the description given above, and for k-modes, we used the algorithm as already implemented in R language. The quality of the partition was compared using the fit rate [14] given by Equation (21)

In previous studies [14,37], the similarity aggregation method gave an optimal solution of four groups. This solution was closer to the classification recognized by zoologists by species and genus (Figure 1).

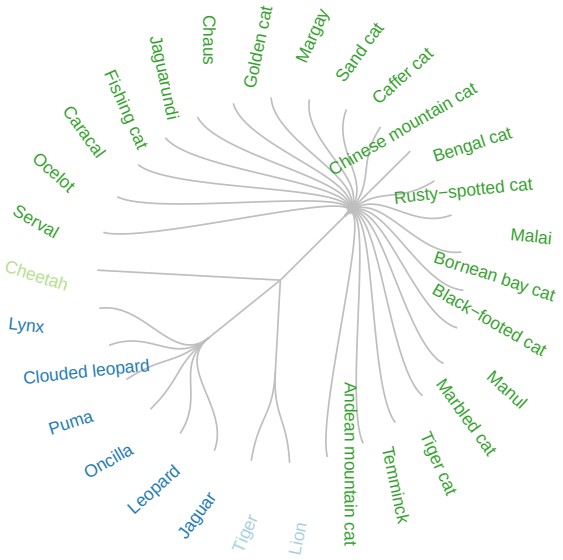

Figure 1. Optimal solution by the Condorcet method.

On the other hand, in the partition into 4 = four groups, applying the k-modes algorithm, it is observed that certain species belong to more than one group and that it does not agree with the classification recognized by zoologists (Figure 2).

The accuracy of measurement, given in Equation (22), of both solutions given in Figures 1 and 2 , is 1 and 0.83, respectively.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}. \tag{22}$$

Next, a comparison was made between the α-Condorcet method and the k-modes method for different values of α in order to find the best method that fit the feline data through the within-class inertia index and adjustment rate given in Equations (13) and (21) respectively.

Figure 3, contrasts the quality of groupings by means of the adjustment rate. Therefore, it is observed that the α-Condorcet method presents a better quality of partition than the k-modes method for different values of α.

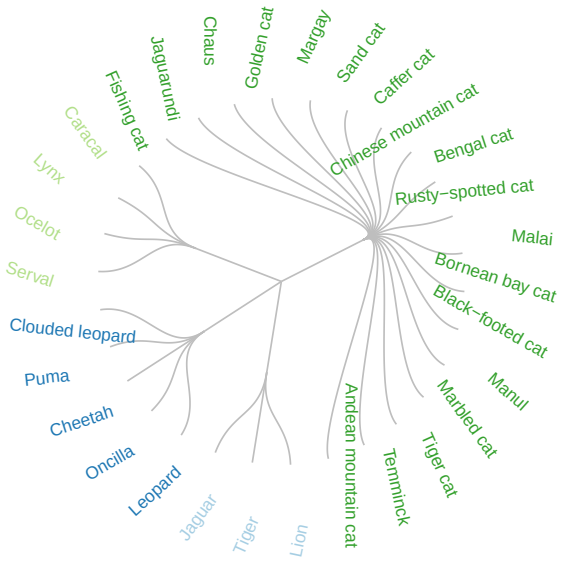

Figure 2. Optimal solution by k-modes, fixing the number of groups at four.

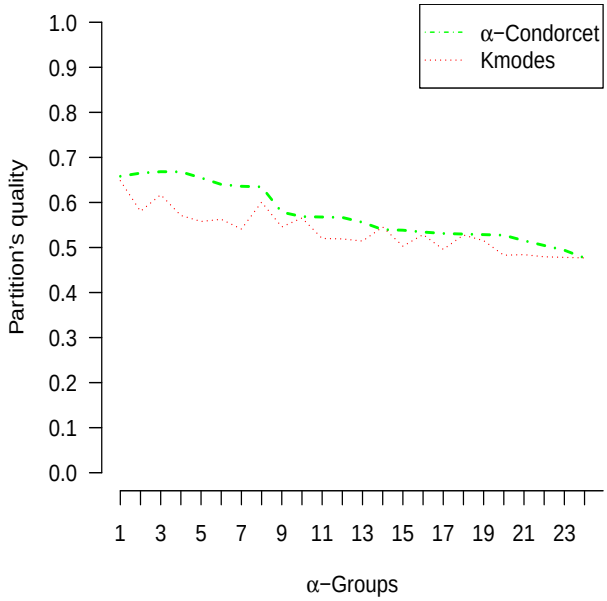

Figure 3. The partitioning quality of feline data under both k-modes and α-Condorcet methods.

Figure 4, contrasts the quality of clustering fit through inertia, in the same dataset. In this figure, it is concluded that the intra–class inertia is almost the same for both methods with different values of α.

We now use the 1990 US Census dataset to compare the heuristic with k-modes algorithm. This dataset contains a 1% sample of the public use microdata sample person records drawn from the full 1990 census sample. For further references, see https://archive.ics.uci.edu/ml/datasets/US+Census+Data+%281990%29 (accessed on 20 December 2021). The comparisons between both methods were made with 50, 100, 150, and 200 observations.

Table 6 shows that the inertia index is almost the same for both algorithms. However, we observe that the heuristic algorithm is better than k-modes algorithm from the point of view of the quality index.

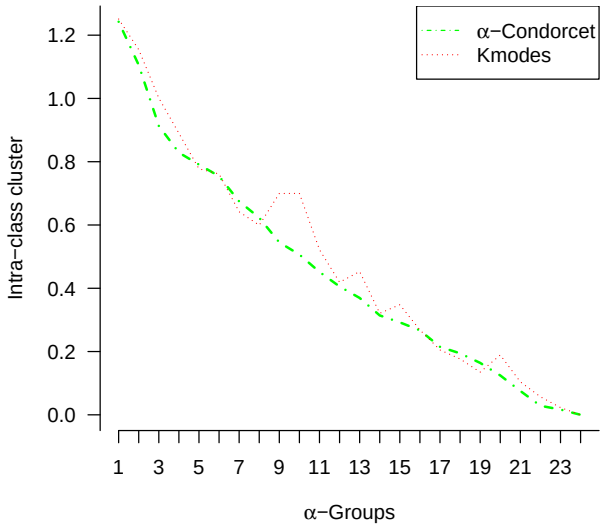

Figure 4. The optimal solution minimizing within-cluster sum-of-squares under both k-modes and α-Condorcet methods.

Table 6. Comparison between k-modes and heuristic algorithm for different sample sizes of the US Census 1990 dataset with $\alpha = 4, 5, 6, 7, 8$.

	Data Size	α	Quality Index	Inertia Index
k-modes	50	4	0.57	3.95
		5	0.56	3.86
		6	0.54	3.66
		7	0.52	3.82
		8	0.54	2.91
Heuristic algorithm		4	0.61	3.49
		5	0.60	3.32
		6	0.58	3.22
		7	0.57	3.10
		8	0.56	3.03
k-modes	100	4	0.58	5.39
		5	0.57	5.08
		6	0.55	4.87
		7	0.53	4.77
		8	0.51	4.80
Heuristic algorithm		4	0.60	4.63
		5	0.60	4.54
		6	0.59	4.45
		7	0.59	4.37
		8	0.58	4.34

Table 6. *Cont.*

	Data Size	α	Quality Index	Inertia Index
k-modes	150	4	0.53	6.04
		5	0.52	5.97
		6	0.52	6.09
		7	0.53	6.06
		8	0.53	5.84
Heuristic algorithm		4	0.60	5.71
		5	0.60	5.65
		6	0.59	5.57
		7	0.59	5.52
		8	0.58	5.44
k-modes	200	4	0.54	6.97
		5	0.52	6.88
		6	0.53	6.82
		7	0.51	6.76
		8	0.51	6.70
Heuristic algorithm		4	0.59	6.89
		5	0.59	6.81
		6	0.59	6.75
		7	0.58	6.70
		8	0.58	6.60

7. Conclusions

In clustering categorical data, many researchers have succeeded in developing unsupervised classification methods without fixing the number of classes *a priori*. Fixing the number of clusters beforehand, may be major drawback.

However, sometimes it is convenient to identify beforehand the number of groups, as, for instance, in psychometrics. Several methods have been proposed with a known number of clusters. We believe, however, that these methods do not always provide optimal solutions. For this reason, we proposed a new method with a fixed number of groups. This new approach is an extension of the Condorcet method. Although, the exact algorithm of this new approach gives an optimal solution, it consumes too much time. Hence, the heuristic algorithm was introduced. Table 5 shows that the proposed algorithm produces almost the same values of quality and inertia indexes as the exact algorithm.

Next, comparing our approach with *k*-modes for the feline data, we found that the accuracy index gave better result for the heuristic algorithm. In this case, the comparison was made with the precision index because we know *a priori* that the number clusters, α, is 4. This comparison was also made with the US Census 1990 data, using both partition quality and intra-class inertia indexes. The results in Table 6 show that both methods have almost the same inertia. However, the heuristic algorithm shows an improvement over *k*-modes in terms of partition quality. Consequently, the following may be concluded:

1. We proposed a heuristic algorithm as an alternative algorithm to the exact one. This gives the same or an approximate solution as the exact one.
2. From the simulations presented in Table 5, we can conclude that the heuristic algorithm is faster than the exact algorithm.
3. The heuristic algorithm produces similar (or even better) results to *k*-modes.
4. We conclude that α-Condorcet is a valid technical competitor with respect to the *k*-modes clustering technique.

Author Contributions: Conceptualization: T.F. and L.F.-L.; methodology: T.F. and L.F.-L.; software: R.C.-S., T.F. and J.M.A.-B.; formal analysis, writing—review, and editing: T.F. and L.F.-L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by FONDECYT (grant 11200749) and the University of Bío-Bío (grant DIUBB 2020525 IF/R). Partial support was provided by the university of Bío-Bío to Luis Firinguetti-Limone (grant DIUBB 183808 3/R).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

The dataset of Table A1 has 30 felines and 14 variables that describe the characteristics of each feline. The full name and the modalities of each variable are described in the following Table A2.

Table A1. The feline dataset is a multivariate dataset introduced by P. Michaud and F. Marcotorchino in their articles [14,34,37].

French Name	English Name	Tipopiel	Longpoill	Retract	Comport	Orielles	Larynx	Tailler	Poids	Longueurs	Queue	Dents	Typproie	Arbre	Chasse
Lion	Lion	1.00	0.00	1.00	1.00	1.00	1.00	3.00	3.00	3.00	2.00	1.00	1.00	0.00	1.00
Tigre	Tiger	3.00	0.00	1.00	3.00	1.00	1.00	3.00	3.00	3.00	2.00	1.00	1.00	0.00	0.00
Jaguar	Jaguar	2.00	0.00	1.00	2.00	1.00	1.00	3.00	3.00	2.00	1.00	1.00	1.00	1.00	0.00
Leopardo	Leopard	2.00	0.00	1.00	3.00	1.00	1.00	3.00	3.00	2.00	2.00	1.00	2.00	1.00	0.00
Once	Oncilla	2.00	1.00	1.00	1.00	1.00	1.00	2.00	2.00	2.00	3.00	1.00	2.00	1.00	0.00
Guepardo	Cheetah	2.00	0.00	0.00	1.00	1.00	0.00	3.00	2.00	2.00	3.00	0.00	2.00	0.00	1.00
Puma	Puma	1.00	0.00	1.00	2.00	1.00	0.00	2.00	3.00	2.00	3.00	1.00	2.00	1.00	0.00
Nebul	Clouded leopard	4.00	0.00	1.00	3.00	1.00	1.00	2.00	2.00	2.00	3.00	1.00	3.00	1.00	0.00
Serval	Serval	2.00	0.00	1.00	1.00	2.00	0.00	2.00	2.00	2.00	1.00	0.00	3.00	1.00	1.00
Ocelot	Ocelot	2.00	0.00	1.00	2.00	1.00	0.00	2.00	2.00	2.00	2.00	0.00	3.00	1.00	0.00
Lynx	Lynx	2.00	1.00	1.00	2.00	2.00	0.00	2.00	2.00	2.00	1.00	1.00	2.00	1.00	0.00
Caracal	Caracal	1.00	0.00	1.00	2.00	2.00	0.00	2.00	2.00	1.00	1.00	0.00	3.00	1.00	1.00
Viverrin	Fishing cat	2.00	0.00	1.00	2.00	1.00	0.00	1.00	1.00	2.00	2.00	0.00	3.00	0.00	0.00
Yaguarun	Jaguarundi	1.00	0.00	1.00	2.00	1.00	0.00	1.00	2.00	2.00	3.00	0.00	3.00	1.00	0.00
Chaus	Chaus	1.00	1.00	1.00	3.00	2.00	0.00	1.00	2.00	1.00	2.00	0.00	3.00	1.00	0.00
Dore	Golden cat	1.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	0.00
Merguay	Margay	2.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	0.00
Margerit	Sand cat	1.00	1.00	1.00	2.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	0.00	0.00
Cafer	Caffer cat	3.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	1.00
Chine	Chinese mountain cat	1.00	0.00	1.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	0.00	3.00	1.00	0.00
Bengale	Bengal cat	2.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	0.00
rouilleu	Rusty spotted cat	2.00	0.00	1.00	2.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	0.00
Malais	Malai	1.00	1.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	1.00	0.00	3.00	1.00	0.00
Borneo	Bornean bay cat	1.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	0.00
Nigripes	Black footed cat	2.00	0.00	1.00	2.00	1.00	0.00	1.00	1.00	1.00	1.00	0.00	3.00	1.00	1.00
Manul	Manul	1.00	1.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	1.00	0.00	3.00	1.00	0.00
Marbre	Marbled cat	4.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	3.00	0.00	3.00	1.00	0.00
Tigrin	Tiger cat	2.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	0.00
Temminck	Temminck	1.00	0.00	1.00	3.00	1.00	0.00	1.00	1.00	1.00	2.00	0.00	3.00	1.00	0.00
Andes	Andean mountain cat	2.00	1.00	1.00	3.00	1.00	0.00	1.00	1.00	2.00	2.00	0.00	2.00	1.00	0.00

Table A2. Description of variables given in Table A1.

Variable	Description	Modalities
Typpel	Appearance of the coat	Unblemished, plain
		Spotted
		Striped
		Marble
Longpoill	Fur	Short hairs
		Long hairs
Retract	Retractable claws	Yes
		No
Comport	Predatory behavior	Diurnal
		Diurnal or nocturnal
		Nocturnal
Orielles	Type of ears	Round or rounded
		Pointed
Larynx	Presence of hyoid bone	Yes
		No
Taille	Waist at the withers	Small
		Average
		Big
Poids	Weight	Low
		Middle
		Heavy
Longueur	Body length	Small
		Middle
		Big
Queue	The relative length of the tail	Short
		Medium
		Long
Dents	Developed fangs	Yes
		No
Typproie	Type of prey	Big
		Big or small
		Small
Arbres	Climb tree	Yes
		No
Chasse	On the run or on the lookout (prowl)	Yes
		No

References

1. Czekanowski, J. Zur Differentialdiagnose der Neandertalgruppe. *Korespondentblatt der Deutschen Gesellschaft für Anthropologie Ethnologie und Urgeschichte* **1909**, *XL*, 44–47.
2. Harkanth, S.; Phulpagar, B.D. A survey on clustering methods and algorithms. *Int. J. Comput. Sci. Inf. Technol.* **2013**, *4*, 687–691.
3. Madhulatha, T.S. An overview on clustering methods. *arXiv* **2012**, arXiv:1205.1117.
4. Madhulatha, T.S. An overview of clustering methods. *Intell. Data Anal.* **2007**, *11*, 583–605. [CrossRef]
5. Xu, R.; Wunsch, D. Survey of clustering algorithms. *IEEE Trans. Neural Netw.* **2005**, *16*, 645–678. [CrossRef] [PubMed]
6. Xu, D.; Tian, Y. A comprehensive survey of clustering algorithms. *Ann. Data Sci.* **2015**, *2*, 165–193. [CrossRef]
7. Ahlquist, J.S.; Breunig, C. Model-based clustering and typologies in the social sciences. *Political Anal.* **2010**, *20*, 325–346. [CrossRef]
8. Aldenderfer, M.S.; Blashfield, R.K. A review of clustering methods. *Clust. Anal.* **1984**, 33–61. [CrossRef]
9. Díaz-Costa, E.; Fernández-Cano, A.; Faouzi, T.; Henríquez, C.F. Validación del constructo subyacente en una escala de evaluación del impacto de la investigación educativa sobre la práctica docente mediante análisis factorial confirmatorio. *Rev. Investig. Educ.* **2015**, *33*, 47–63. [CrossRef]
10. Díaz-Costa, E.; Fernández-Cano, A.; Faouzi-Nadim, T.; Caamaño-Carrillo, C. Modelamiento y estimación del índice de impacto de la investigación sobre la docencia. *Revista Electrónica Interuniversitaria de Formación del Profesorado* **2019**, *22*, 211–228.
11. Fonseca, J.R.S. Clustering in the field of social sciences: That is your choice. *Int. J. Soc. Res. Methodol.* **2013**, *16*, 403–428. [CrossRef]
12. Rice, P.M.; Saffer, M.E. Cluster analysis of mixed-level data: Pottery provenience as an example. *J. Archaeol. Sci.* **1982**, *9*, 395–409. [CrossRef]

13. Huang, Z. Extensions to the *k*-means algorithm for clustering large data sets with categorical values. *Data Min. Knowl. Discov.* **1998**, *2*, 283–304. [CrossRef]
14. Marcotorchino, F.; Michaud, P. Agregation de similarites en classification automatique. *Rev. Stat. Appl.* **1982**, *30*, 21–44.
15. Bock, H.-H. Origins and extensions of the *k*-means algorithm in cluster analysis. *Electron. J. Hist. Probab. Stat.* **2008**, *4*, 1–18.
16. Forgy, E.W. Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. *Biometrics* **1965**, *21*, 768–769.
17. Jain, A.K. Data clustering: 50 years beyond K-means. *IEEE Pattern Recognit. Lett.* **2010**, *31*, 651–666. [CrossRef]
18. MacQueen, J. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*; University of California Press: Oakland, CA, USA, 1967; Volume 281–297.
19. Goyal, M.; Aggarwal, S. A Review on K-Mode Clustering Algorithm. *Int. J. Adv. Res. Comput. Sci.* **2017**, *8*, 1615–1620. [CrossRef]
20. Xiong, T.; Wang, S.; Mayers, A.; Monga, E. DHCC: Divisive hierarchical clustering of categorical data. *Data Min. Knowl. Discov.* **2012**, *24*, 103–135. [CrossRef]
21. Lakshmi, K.; Visalakshi, N.K.; Shanthi, S.; Parvathavarthini, S. Clustering categorical data using K-Modes based on cuckoo searsh optimization algorithm. *ICTACT J. Soft Comput.* **2017**, *8*, 1561–1566. [CrossRef]
22. Dorman, K.S.; Maitra, R. An Efficient *k*-modes Algorithm for Clustering Categorical Datasets. *Stat. Anal. Data Min. ASA Data Sci. J.* **2022**, *15*, 83–97. [CrossRef]
23. Ali, D.S.; Ghoneim, A.; Saleh, M. K-modes and Entropy Cluster Centers Initialization Methods. In *ICORES*; SciTePress: Setúbal, Portugal, 2017; pp. 447–454.
24. Khan, S.S.; Kant, S. Computation of Initial Modes for K-modes Clustering Algorithm Using Evidence Accumulation. In *IJCAI*; Morgan Kaufmann Publishers: San Francisco, CA, USA, 2007; pp. 2784–2789.
25. Huang, Z.; Ng, M.K. A fuzzy *k*-modes algorithm for clustering categorical data. *IEEE Trans. Fuzzy Syst.* **1999**, *7*, 446–452. [CrossRef]
26. Gan, G.; Ma, C.; Wu, J. *Data Clustering: Theory, Algorithms, and Applications*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2020.
27. Jiang, Z.; Liu, X. A novel consensus fuzzy *k*-modes clustering using coupling DNA-chain-hypergraph P system for categorical data. *Processes* **2020**, *8*, 1326. [CrossRef]
28. Kim, D.-W.; Lee, K.H.; Lee, D. Fuzzy clustering of categorical data using fuzzy centroids. *Pattern Recognit. Lett.* **2004**, *25*, 1263–1271. [CrossRef]
29. Ng, M.K.; Li, M.J.; Huang, J.Z.; He, Z. On the impact of dissimilarity measure in *k*-modes clustering algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 503–507. [CrossRef] [PubMed]
30. Cao, F.; Liang, J.; Li, D.; Bai, L.; Dang, C. A dissimilarity measure for the *k*-modes clustering algorithm. *Knowl.-Based Syst.* **2012**, *26*, 120–127. [CrossRef]
31. Hazarika, I.; Mahanta, A.K.; Das, D. A New Categorical Data Clustering Technique Based on Genetic Algorithm. *Int. J. Appl. Eng. Res.* **2017**, *12*, 12075–12082.
32. Khandelwal, G.; Sharma, R. A simple yet fast clustering approach for categorical data. *Processes* **2015**, *120*, 25–30. [CrossRef]
33. Seman, A.; Bakar, Z.A.; Sapawi, A.M.; Othman, I.R. A medoid-based method for clustering categorical data. *J. Artif. Intell.* **2013**, *6*, 257. [CrossRef]
34. Michaud, P.; Marcotorchino, F. Modèles d'optimisation en analyse des données relationnelles. *Math. Sci. Hum.* **1979**, *67*, 7–38.
35. Michaud, P. Condorcet: A man of the avant–garde. *Appl. Stoch. Model. Data Anal.* **1987**, *3*, 173–189. [CrossRef]
36. Michaud, P. The true rule of the Marquis de Condorcet. In *Compromise, Negotiation and Group Decision*; Springer: New York, NY, USA, 1988; pp. 83–100.
37. Michaud, P. *Agrégation à la Majorité II: Analyse du Résultat d'un Vote*; Centre Scientifique IBM France: Paris, France, 1985.
38. Hägele, G.; Pukelsheim, F. Llul's writings on electoral systems. *Stud. Lul.* **2001**, *41*, 3–38.
39. Marcotorchino, F. *Liaison Analyse Factorielle-Analyse Relationnelle: Dualité Burt-Condorcet*; IEEE Centre Scientifique IBM France: Paris, France, 1989.
40. Lebbah, M.; Bennani, Y.; Grozavu, N.; Benhadda, H. Relational analysis for clustering consensus. *Mach. Learn.* **2010**, 45–59. [CrossRef]