Clustering Approach using Belief Function Theory
Belief K-modes Method

Sarra BEN HARIZ

Advisors: Pr. Khaled MELLOULI (IHEC, Tunisia)
Zied ELOUEDI (ISG, Tunisia)

Masterś Thesis

Computer Science Department
Higher Institute of Management of Tunis

Research Laboratory:
L.A.R.O.D.E.C

July 2006

Acknowledgements

I would like to thank my advisor Professor Khaled Mellouli for introducing me to an interesting area of research. I thank him for his continuous guidance and for all the valuable discussions we have had during the period of this work.

I owe so much to my co-advisor Zied Elouedi in the fulfillment of this master thesis. He has been very patient in supervising this work at all its stages. Your competence and your generosity have allowed me to work in good conditions. Your comments and your observations have always been of great interest. Thank you for your support, advices, constant encouragements and for your endless patience in improving my writing.

I am also thankful to Thierry Denoeux from *Université de Technologie de Compiègne* for replying me each time I asked him questions regarding belief function theory.

A special wink to all members of my laboratory LARODEC for all the good and bad moments we get together.

Heartfelt thanks to all of my friends for their encouragement especially Wided, Hajer, and Hanen.

Especially, I would like to give my special thanks to my family for providing a loving environment for me. Lastly, and most importantly, I wish to thank my parents. They bear me, raise me, support me, teach me, and love me. To them I dedicate this master thesis.

I would like to express my gratitude to all those who gave me the possibility to complete this work.

Contents

List of Figures

List of Tables

Introduction

Data analysis is a process that involves describing, summarizing and comparing data. It has traditionally been a branch of statistics. However, it has now also spread to the fields of computer science, for example machine learning, artificial intelligence, data management, and data mining.

An important technique frequently used in data analysis is **clustering**. The aim of clustering is to group a set of objects into classes of similar objects. It can be used on any datasets consisting of items for which we have a measure of similarity.

Clustering techniques are among the well known machine learning techniques, and the K-modes method [31] is considered as one of the most popular of them. These techniques are used in many domains such as medicine, banking, finance, marketing, security, etc. They work under an unsupervised mode when the class label of each object in the training set is not known a priori.

In addition to these unsupervised classification techniques, there exist those working under a supervised mode when the classes of instances are known in advance helping to construct a model that will be used for classifying new objects. Among them, we mention decision trees, k-nearest neighbor, neural networks, etc.

The capability to deal with datasets containing uncertain attributes is undoubtedly important due to the fact of this kind of datasets is common in real life data mining applications. However, this problem makes most of the standard methods inappropriate for clustering such training objects. In order to overcome this drawback, the idea is to combine clustering methods with theories managing uncertainty, these theories are probabilities theory, fuzzy set theory, possibilitic theory and the belief function theory.

This latter theory as interpreted in the transferable belief model (TBM) [57] presents an effective tool to deal with this uncertainty. It permits to handle partial or even total ignorance concerning classification parameters, and offers interesting means to combine several pieces of evidence. In fact, there are belief classification techniques which have been developed such as belief decision trees (BDT) [21] and belief k-nearest neighbor [13], and which have provided interesting results.

The objective of this work is to develop a new clustering method in an uncertain context that uses the K-modes paradigm and based on the belief function theory, the proposed approach is called the belief K-modes method (BKM). The main contributions of this work are to provide one approach to deal with on one hand the construction of clusters where the values of the attributes of training objects may be uncertain, and in the other hand the classification of new instances charaterized also by uncertain values based on the obtained clusters.

This report is organized in four chapters belonging to two main parts:

Part I: *Theoretical aspects* presents the necessary theoretical aspects regarding the belief function theory and the K-modes method which are detailed respectively in chapter 1 and chapter 2.

Part II: *Belief K-modes Method* details our proposed method namely Belief K-modes method. Chapter 3, details the different parameters that we have developed relatively to the building procedure of the clusters within an uncertain context. Furthermore, the classification step of both certain and uncertain instances using the obtained clusters will be exposed. Chapter 4 deals with implementation and simulations which have been performed in order to analyze and evaluate results given by the proposed belief clustering method.

Finally, a conclusion summarizes all the work presented in this report and proposes further works to improve our method.

An appendix is provided at the end of this report to complete this master thesis. It presents the description of data sets used in simulations.

Part I

Theoretical aspects

Chapter 1

Belief function Theory

1.1 Introduction

The mathematical theory of Evidence [45], also known as Dempster-Shafer theory and theory of belief functions (BF's) is considered as a useful theory for representing and managing uncertain knowledge. This theory is introduced as a model to represent quantified beliefs.

The belief function theory is widely applied to artificial intelligence. More recent variants of Dempster-Shafer theory include the Transferable Belief Model (TBM) [50, 54, 55], the lower probability model [63], and the theory of Hints [36].

In this report, we deal with the interpretation of the belief function theory as explained by the TBM on which our work is based. Belief function theory as interpreted in TBM [50, 54, 55] provides a mathematical tool to treat subjective, and personal judgments on the different parameters of any classification problems and can be easily extended to deal with objective probabilities. It expresses partial beliefs in a much more flexible way than probability functions do. It also permits to handle partial or even total ignorance concerning classification parameters. This theory offers interesting tools to combine several pieces of evidence [47, 49, 52], like the conjunction and the disjunction rules of combination. Furthermore, decision making is solved through the pignistic transformation.

In this chapter, we introduce the basic concepts of this theory. Next, some special belief functions are described. Then, several concepts of the belief function theory are detailed like the combination, and the discounting, Finally, we present the decision process within this theory. All these concepts are illustrated by examples.

1.2 Basic concepts

1.2.1 Frame of discernment

Let Θ be a finite non empty set of elementary events to a given problem, called **the frame of discernment**. It also referred to as **the universe of discourse** or **the domain of reference**. This set contains hypotheses about some problem domain. All the subsets of Θ belong to the power of Θ, denoted by 2^Θ and defined as follows:

$$2^\Theta = \{A : A \subseteq \Theta\} \tag{1.1}$$

Each element of 2^Θ is called **a proposition** or **an event**.

The elements of Θ are called the elementary propositions.

In Shafer's model [45], the frame of discernment is defined to be the set of mutually **exclusive** and **exhaustive** hypotheses which means that the solution to a given problem is unique and is necessarily included in this frame of discernment.

However, in the TBM, Smets contests the exhaustivity of the frame of discernment considering that it is difficult to know a priori all the possible hypotheses for some problems and introduces the **open-**and **closed-world assumptions** [50, 54, 55].

Under the **open-world assumption** the frame is not necessarily exhaustive admitting thus the existence of an unkown proposition (not defined in the frame of discernment) that might be a solution to the considered problem, whereas under the **closed-world assumption** the frame of discernment is exhaustive.

Example 1.1 *Let's treat a classification problem of firmś departments. Suppose that the frame of discernment is defined as follows:*
$\Theta = \{\textit{Finance, Marketing, Accounts}\}$

The corresponding power set of Θ *is:*

$2^\Theta = \{\emptyset, \{Finance\}, \{Marketing\}, \{Accounts\}, \{Finance, Marketing\},$
$\{Finance, Accounts\}, \{Marketing, Accounts\}, \{Finance, Marketing, Accounts\}\}$

1.2.2 Basic belief assignment

The impact of a piece of evidence on the different subsets of the frame of discernment Θ is represented by the so-called **basic belief assignment** (bba), called initially [45] **basic probability assignment**. The bba is defined as follows:

$$m : 2^{\Theta} \mapsto [0,1]$$
$$\sum_{A \subseteq \Theta} m(A) = 1 \qquad (1.2)$$

Each quantity m(A), named **basic belief mass** (bbm), is considered to be the part of belief that supports the event A, and that, due to the lack of information, does not support any strict subset of A.

Shafer has initially imposed the condition $m(\emptyset) = 0$. This condition reflects the fact that no belief ought to be allocated to the emptyset. Such bba is called **a normalized basic belief assignment**.

This condition is relaxed in the TBM, the allocation of a positive mass to the empty set ($m(\emptyset) > 0$) is interpreted as a consequence of the **open-world assumption**. A bba that verifies this condition is said to be **subnormal** or **unformalized**. A mass of belief is assigned to each possible subset of classes.

Example 1.2 *Let's continue with Example 1.1. Suppose an expert expresses a piece of evidence concerning the department nature. The bba related to this expert's evidence is defined as:*
$m(\{Finance\}) = 0.6$;
$m(\{Accounts, Marketing\}) = 0.3$;
$m(\Theta) = 0.1$.
For example, 0.6 represents the part of expert's belief exactly supporting the proposition that the needed department is Finance department.

1.2.3 Focal elements, body of evidence, core

The subsets A of the frame of discernment Θ such that m(A) is strictly positive, are called **the focal elements** of the bba m.

The pair (F,m) is called **a body of evidence** where F is the set of all the focal elements relative to the bba m.

The union of all the focal elements of m are named **the core** and are defined as follows:

$$\varphi = \bigcup_{A:m(A)>0} A \tag{1.3}$$

Example 1.3 *Let's continue with our example (Example 1.2). The subsets* $\{Finance\}$, $\{Accounts, Marketing\}$,Θ *are the focal elements of the bba m.*

So,(F,m) is called the body of evidence such that:
$F = \{\{Finance\}, \{Accounts, Marketing\}, \Theta\}$

The core is defined as follows:
$\varphi = \{Finance\} \cup \{Accounts, Marketing\} \cup \Theta = \Theta$

1.2.4 Belief function

A belief function, denoted bel, corresponding to a specific bba m, assigns to every subset A of Θ the sum of masses of belief committed to every subset of A by m [45]. The belief function bel is defined as follows:

$$bel : 2^{\Theta} \mapsto [0, 1]$$
$$bel(A) = \sum_{\phi \neq B \subseteq \Theta} m(B) \tag{1.4}$$

The belief function bel represents the total belief that one commits to A without being also commited to $\overline{A}$. The bbm $m(\emptyset)$ is not included in bel(A) as $\emptyset$ is both a subset of A and $\overline{A}$.

Properties:

- Sub-additivity:

$$bel(A) + bel(\overline{A}) \leq 1 \tag{1.5}$$

  Contrary to the probability theory the belief function theory increasing beliefs on a proposition A does not necessary require the decrease of beliefs on $\overline{A}$.

- Monotonicity:

$$A \subseteq B \Longrightarrow bel(b) \geq bel(A) \tag{1.6}$$

  Θ will get the highest value of bel, whereas $\emptyset$ will get the lowest value.

- $For A, B \subseteq \Theta, A \cap B = \emptyset,$
$$bel(A \cup B) \geq bel(A) + bel(B) \tag{1.7}$$
- m(A) may be expressed by the values of bel as follows [60]:
$$m(A) = \sum_{B \subseteq A} (-1)^{|A|-|B|} bel(B), \forall A \subseteq \Theta, A \neq \emptyset \tag{1.8}$$
- the bba $m(\emptyset)$ is computed as follows :
$$m(\emptyset) = 1 - bel(\Theta) \tag{1.9}$$

Example 1.4 *The belief function bel corresponding to the same example (Example 1.2) is defined as follows:*
$bel(\emptyset) = 0;$
$bel(\{Finance\}) = 0.6;$
$bel(\{Marketing\}) = bel(\{Accounts\} = 0;$
$bel(\{Finance, Marketing\}) = 0.6;$
$bel(\{Finance, Accounts\}) = 0.6;$
$bel(\{Marketing, Accounts\}) = 0.3;$
$bel(\Theta) = 0.1 + 0.3 + 0.6 = 1$
For example, 0.6 is the total belief committed to the proposition $\{Finance,Marketing\}$*. It is the sum of the bbm's assigned to this set and also to its subsets.*

1.2.5 Plausibility function

The plausibility function pl associated with a mass function m quantifies the maximum amount of belief that could be given to a subset A of the frame of discernment. It is equal to the sum of the bbm's relative to subsets B compatible with A. The plausibility function pl is defined as follows [3]:
$$pl : 2^{\Theta} \mapsto [0, 1]$$
$$pl(A) = \sum_{A \cap B \neq \phi} m(B) \tag{1.10}$$

There is a simple relationship between the belief function bel and the plausibility function pl associated with a mass function m : for $A \subseteq \Theta$
$$pl(A) = bel(\Theta) - bel(\overline{A}) \tag{1.11}$$
and
$$bel(A) = pl(\Theta) - bel(\overline{A}) \tag{1.12}$$
where $\overline{A}$ denotes the complement of A.

Properties:

- Over additivity:

$$pl(A) + pl(\overline{A}) \geq 1 \tag{1.13}$$

- Monotonicity:

$$A \subset B \Longrightarrow pl(B) \geq pl(A) \tag{1.14}$$

- $For A, B \subseteq \Theta, A \cap B = \emptyset,$

$$pl(A \cup B) \leq pl(A) + pl(B) \tag{1.15}$$

- $For A \subseteq \Theta,$

$$bel(A) \leq pl(A) \tag{1.16}$$

Example 1.5 *The plausibility function pl corresponding to the bbm (see Example 1.2) is defined as follows:*
$pl(\emptyset) = 0;$
$pl(\{Finance\}) = 0.6 + 0.1 = 0.7;$
$pl(\{Marketing\} = 0.3 + 0.1 = 0.4;$
$pl(\{Accounts\}) = 0.3 + 0.1 = 0.4;$
$pl(\{Marketing, Accounts\}) = 0.3 + 0.1 = 0.4;$
$pl(\{Finance, Marketing\}) = 0.3 + 0.1 + 0.6 = 1;$
$pl(\{Finance, Accounts\}) = 0.3 + 0.1 + 0.6 = 1;$
$pl(\{\Theta\}) = 0.3 + 0.1 + 0.6 = 1$
For example, 0.7 represents the maximum degree of belief that could be given to the proposition $\{Finance\}$.

1.2.6 Commonality function

Another function related to a basic belief function m is the **commonality function** q. The meaning of the commonality function is not obvious.
However, it may represent the total mass that is free to move to every element of A [3]. It is defined as follows:

$$q : 2^{\Theta} \mapsto [0, 1]$$
$$q(A) = \sum_{A \subseteq B} m(B) \tag{1.17}$$

Properties:

- The commonality value relative to the empty set is defined as follows:

$$q(\emptyset) = 1 \tag{1.18}$$

- the commonality value relative to the whole frame of discernment is defined as follows:

$$q(\Theta) = m(\Theta) \tag{1.19}$$

Example 1.6 *The commonality function q corresponding to the same bbm, as in the previous examples, is defined as follows:*
$q(\{\emptyset\}) = 1;$
$q(\{Finance\}) = 0.6 + 0.1 = 0.7;$
$q(\{Marketing\}) = 0.3 + 0.1 = 0.4;$
$q(\{Accounts\}) = 0.3 + 0.1 = 0.4;$
$q(\{Finance, Marketing\}) = q(\{Finance, Accounts\}) = 0.1;$
$q(\{Marketing, Accounts\}) = 0.3 + 0.1 = 0.4;$
$q(\{\Theta\}) = 0.1$

1.2.7 Belief intervals

The information contained in bel concerning a given subset A may be conveniently expressed by the interval [bel(A),pl(A)] called **belief interval**.

Example 1.7 *Let's consider the setting of examples 1.4 and 1.5.*
We have bel({Finance})=0.6 and pl({Finance})=0.7.
Thus, the belief interval corresponding to the proposition {Finance} is given by the interval [0.6, 0.7].

1.3 Special belief functions

1.3.1 Vacuous belief function

A belief function is said to be **vacuous** if Θ is its unique focal element [45]:

$$m(\Theta) = 1 \; and \; m(A) = 0 \; for \; all \; A \subseteq \Theta, A \neq \Theta \tag{1.20}$$

In other words :

$$bel(\Theta) = 1 \; and \; bel(A) = 0 \; for \; all \; A \subseteq \Theta, A \neq \Theta \tag{1.21}$$

Such bba where Θ is the unique focal element, quantifies the state of *total ignorance* since there is no support given to any strict subset of Θ.

Example 1.8 *Suppose that an expert cannot detect the nature of department. We have a state of total ignorance where the corresponding bba* m_0 *is a vacuous bba defined by:*
$m_0(\Theta) = 1$ *and* $m_0(A) = 0$ *for* $A \neq \Theta$.

1.3.2 Categorical belief function

It is a normalized belief function such that its bba is defined as follows [41]:

$$m(A) = 1 \ for \ some \ A \subset \Theta \ and \ m(B) = 0, \ for \ B \subseteq \Theta, B \neq A \tag{1.22}$$

The unique focal element A is different from the frame of discernment Θ.

Example 1.9 *We obtain a piece of evidence specifying that the department cannot be Finance department. So, the corresponding bbm presents a categorical belief function characterized by:*
m({Marketing,Accounts})=1

1.3.3 Bayesian belief function

A **Bayesian belief function** is a belief function where the focal elements are all singletons, is defined as follows [45]:

$$bel(\emptyset) = 0 \tag{1.23}$$

$$bel(\Theta) = 1 \tag{1.24}$$

$$bel(A \cup B) = bel(A) + bel(B) \ whenever \ A, B \subset \Theta \ and \ A \cap B = \emptyset \tag{1.25}$$

Properties:

- bel becomes a probability distribution
- As in the probability theory :

$$bel(A) + bel(\overline{A}) = 1 \ for \ A \subset \Theta \tag{1.26}$$

- bel = pl

Example 1.10 *The same frame of discernment is considered. The evidence is expressed by:*
$m_b(\{Finance\}) = 0.5;$
$m_b(\{Marketing\}) = 0.3;$
$m_b(\{Accounts\}) = 0.2;$
$m_b(\{\Theta\}) = 0$
The bba m_b is a bayesian belief function since all its focal elements are singletons.

1.3.4 Consonant belief function

A **consonant belief function** is a belief function whose focal elements are nested, such that $A_1 \subset A_2 \subset ... \subset A_n$ [45]. In that case,

$$bel(A \cap B) = min[bel(A), bel(B)] \; \forall \; A, B \; \subseteq \Theta \tag{1.27}$$

or equivalently

$$pl(A \cup B) = max[pl(A), pl(B)] \; \forall \; A, B \; \subseteq \Theta \tag{1.28}$$

Such belief and plausibility functions are called respectively **necessity** and **possibility** measures [19].

Example 1.11 *Let's continue with the same frame of discernment* $\Theta(see Example 1.1):$

$m(\{Accounts\}) = 0.2;$
$m(\{Accounts, Marketing\}) = 0.6;$
$m(\{\Theta\}) = 0.2$
The focal elements of this bba m are nested. So, it is a consonant belief function.

1.3.5 Certain belief function

A **certain belief function** is a categorical belief function such that it has only one focal element and which should be a singleton. Its corresponding bba is defined as follows:

$$m(A) = 1 \; and \; m(B) = 0 \; for \; all \; B \neq A \; and \; B \subseteq \Theta \tag{1.29}$$

where A is a singleton event of Θ. This function represents a state of total certainty as it assigns all the belief to a unique elementary event.

Example 1.12 *Let's consider the expert's confirmation S_c that the department is a Marketing department.*
The bba m_c corresponding is a certain bba defined as:
$m_c(\{Marketing\}) = 1.$

1.3.6 Simple support function

A belief function is said to be a **simple support function (SSF)** if it has at most one focal element different from the frame of discernment Θ. This focal element is called **the focus** of the SSF.

A SSF is defined as follows:

$$m(X) = \begin{cases} \omega & \text{if X} = \Theta \\ 1-\omega & \text{if X} = \text{A for some } A \subseteq \Theta \\ 0 & otherwise. \end{cases} \quad (1.30)$$

where A is the focus and $\omega \in [0, 1]$

Example 1.13 *Considering the same example (Example 1.1) and a bba which is defined as follows:*
$m(\{Finance, Marketing\}) = 0.7;$
$m(\{\Theta\}) = 0.3$
m is called a simple support function where the focus is the proposition $\{Finance, Marketing\}$.

1.4 Combination rules

One of the most important operations in the Dempster-Shafer theory is the aggregation of several sources of evidence. The belief function theory, as understood in the TBM framework, offers interesting rules for aggregating the basic belief assignments defined over the same frame of discernment and induced from distinct pieces of evidence [51] and provided by two (or more) source of information [47, 49, 52].

We can combine them in at least two ways: conjunctively or disjunctively using respectively the conjunctive or disjunctive rules of combination.

The choice of one of these rules of combination for aggregating pieces of evidence may be guided by meta-belief concerning the reliability of the sources. In fact, if we know that both sources of information are fully reliable, then we combine them conjunctively. However, if we know that at least one of the two sources is reliable, then we combine them disjunctively [52].

1.4.1 Combination of two information sources

Let m_1 and m_2 be two bba's defined on the same frame of discernment Θ. These two bba's are collected by two 'distinct' pieces of evidence and induced from two experts (information sources).

The Conjunctive rule of combination

$$(m_1 \bigcirc\!\!\!\!\cap\, m_2)(A) = \sum_{B,C \subseteq \Theta; B \cap C = A} m_1(B) m_2(C) \tag{1.31}$$

Properties:

The following properties characterize the conjunctive rule of combination:

- Compositionality:
  $(m_1 \bigcirc\!\!\!\!\cap\, m_2)(A)$ is function of A, m_1 and m_2
- Commutativity:
  $$m_1 \bigcirc\!\!\!\!\cap\, m_2 = m_2 \bigcirc\!\!\!\!\cap\, m_1 \tag{1.32}$$
- Associativity:
  $$(m_1 \bigcirc\!\!\!\!\cap\, m_2) \bigcirc\!\!\!\!\cap\, m_3 = m_1 \bigcirc\!\!\!\!\cap\, (m_2 \bigcirc\!\!\!\!\cap\, m_3) \tag{1.33}$$
- Non-idempotency
  $$m \bigcirc\!\!\!\!\cap\, m \neq m \tag{1.34}$$
- Neutral element
  The neutral element within the conjunctive rule of combination is the vacuous basic belief assignment representing the total ignorance.
  $$m \bigcirc\!\!\!\!\cap\, m_0 = m \tag{1.35}$$
  where m_0 is a vacuous bba.

Example 1.14 *In this example, we will consider two distinct experts' evidences S_1 and S_2 concerning the department nature, represented by respectively the two bba's m_1 and m_2 as follows:*
$m_1(\{Finance\}) = 0.4$;
$m_1(\{Accounts\}) = 0.1$;
$m_1(\{Finance, Marketing\}) = 0.3$;
$m_1(\{\Theta\}) = 0.2$;

$m_2(\{Marketing\}) = 0.5$;
$m_2(\{Finance, Marketing\}) = 0.4$;
$m_2(\{\Theta\}) = 0.1$;

Once the conjunctive rule of combination is applied, the final results are:
$(m_1 \textcircled{\cap} m_2)(\{\emptyset\}) = 0.2 + 0.05 + 0.04 = 0.29$;
$(m_1 \textcircled{\cap} m_2)(\{Finance\}) = 0.16 + 0.04 = 0.2$;
$(m_1 \textcircled{\cap} m_2)(\{Accounts\}) = 0.01$;
$(m_1 \textcircled{\cap} m_2)(\{Marketing\}) = 0.1 + 0.15 = 0.25$;
$(m_1 \textcircled{\cap} m_2)(\{Finance, Marketing\}) = 0.08 + 0.12 + 0.02 = 0.22$;
$(m_1 \textcircled{\cap} m_2)(\{\Theta\}) = 0.03$

The Disjunctive rule of combination

The disjunctive rule of combination is the dual of the conjunctive one, it builds the bba representing the impact of two pieces of evidence when we only know that at least one of them is to be accepted, but we do not know which one. This rule is defined as follows [52] :

$$(m_1 \textcircled{\cup} m_2)(A) = \sum_{B,C \subseteq \Theta; B \cup C = A} m_1(B) m_2(C) \tag{1.36}$$

Properties:

The disjunctive rule of combination is commutative and associative.

In this case, we assume that at least one of the two experts (S_1 *and* S_2) is accepted, but we do not know which one and in consequence we will apply the disjunctive rule of combination.

Example 1.15 *Let us consider the same bba's represented in Example 1.14.*
Once the disjunctive rule of combination is applied, we get:
$(m_1 \textcircled{\cup} m_2)(\{Finance, Marketing\}) = 0.2 + 0.1 + 0.16 + 0.08 = 0.54$;
$(m_1 \textcircled{\cup} m_2)(\{Accounts, Marketing\}) = 0.05$;
$(m_1 \textcircled{\cup} m_2)(\{\Theta\}) = 0.15 + 0.12 + 0.04 + 0.02 + 0.03 + 0.04 + 0.01 = 0.41$;

1.4.2 Combination of several information sources

It is easily assured by applying repeatedly the chosen rule (conjunctive or disjunctive rule), since these rules of combination are both commutative and associative.

1.5 Discounting

Experts are not fully reliable, the method of discounting [45] seems imperative to update experts' beliefs by tacking into account their reliability.

Let $(1-\alpha)$ be the degree of trust assigned to the expert, it quantifies the strength of reliability given to this expert.

The expert's opinions become :

$$m^{\alpha}(A) = (1-\alpha)m(A) \; for \; A \subset \Theta \tag{1.37}$$

$$m^{\alpha}(\Theta) = \alpha + (1-\alpha)m(\Theta) \tag{1.38}$$

where α is named **the discounting factor**.

Properties:

- $\alpha = 0$ means that the expert is totally reliable.
- $\alpha = 1$ means that the expert is not reliable at all. Then, his opinions have to be totally ignored.

Example 1.16 *The degree of reliability given to the expert is equal to 0.7. If we consider this bba defined as follows:*
$m(\{Finance\}) = 0.6$
$m(\{Accounts, Marketing\}) = 0.3$
$m(\Theta) = 0.1$

So, we obtain after discounting this bba:
$m_{\alpha}(\{Finance\}) = 0.6 \times 0.7 = 0.42;$
$m_{\alpha}(\{Accounts, Marketing\}) = 0.3 \times 0.7 = 0.21;$
$m_{\alpha}(\Theta) = 0.3 + (1-0.3) \times 0.1 = 0.07$

1.6 Decision Process

1.6.1 Introduction

It is necessary for making a decision, to select the most likely hypothesis. Some solutions are developed to ensure the decision making within the belief function theory. The one of the most used is the pignistic probability proposed within the TBM

[54, 55, 57].

We mention other methods like the maximum of credibility which consists in choosing the hypothesis having the highest value of the belief function bel, that is the most credible hypothesis and the maximum of plausibility, contrary to the maximum credibility criterion, this method consists in supporting the hypothesis having the highest value of the plausibility function pl [3].

1.6.2 Pignistic transformation

The TBM is based on a two level mental models:

- The credal level where beliefs are entertained and represented by belief functions.
- The pignistic level where beliefs are used to make decisions and represented by probability functions called **the pignistic probabilities.**

When a decision must be made, beliefs held at the credal level induce a probability measure at the pignistic measure denoted *BetP* [57].
The link between these two functions is achieved by the pignistic transformation.

$$BetP(A) = \sum_{B \subseteq \Theta} \frac{|A \cap B|}{|B|} \frac{m(B)}{(1 - m(\phi))}, \text{ for all } A \in \Theta \tag{1.39}$$

Example 1.17 *The following bba m are considered at the credal level:*
$m(\{Finance, Marketing\}) = 0.1;$
$m(\{Finance\}) = 0.6;$
$m(\{\Theta\}) = 0.3$

To compute a pignistic probability BetP corresponding to these bba m, is necessary to make a decision.
We get:
$BetP(\{Finance\}) = 0.75;$
$BetP(\{Marketing\}) = 0.15;$
$BetP(\{Accounts\}) = 0.1$
We note that is more probable that the department is a Finance department.

1.7 Conclusion

The basic concepts of the belief function theory as understood in the transferable belief model are presented in this chapter.

These different notions are illustrated by many examples.

This presentation shows that the belief function theory provides a convenient tool to handle uncertainty in classification problems, especially within clustering techniques. The following chapter will deal with these methodes more precisely the K-modes one.

Chapter 2

K-modes Method as Clustering Method

2.1 Introduction

Clustering techniques are considered as efficient tools for partitioning data sets in order to get homogeneous clusters of objects. These techniques [34] are among the well known machine learning techniques, and are used in many domains such as medicine, banking, finance, marketing, security, etc. They work under an unsupervised mode when the class label of each object in the training set is not known a priori. The K-modes method [31] is considered as one of the most popular of them. Its algorithm was proposed to extend the K-means one [39] to tackle the problem of clustering large categorical data sets in data mining.

In this chapter, we firstly study the cluster analysis. We expose several clustering techniques, organized into the following categories: partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods. We are interested especially in the basics of the K-modes method as partitioning clustering method. We focus on its standard version where its principle will be described, then we will present the algorithm, and an example will be detailed to illustrate this method. Two extensions of this method will be described in order to cope with one K-modes method drawback.

The last part of this chapter deals with another kind of this approach under uncertain environment which is briefly exposed combining this method with one theory managing this kind of environment such as fuzzy theory. Finally, we deal with belief clustering.

2.2 Cluster Analysis

2.2.1 Introduction

The process of grouping a set of objects into classes of similar objects is called clustering. A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. A cluster of data objects can be treated collectively as one group and so may be considered as a form of data compression. Although classification is an effective means for distinguishing groups or classes of objects. It is often more desirable to proceed in the reverse direction: First partition the set of data into groups based on data similarity (e.g., using clustering), and then assign labels to the relatively small number of groups. Additional advantage of such a clustering-based process is that it is adaptable to changes and helps single out useful features that distinguish different groups.

Clustering is also called data segmentation in some applications because clustering partitions large data sets into groups according to their similarity. As a data mining function, cluster analysis can be used as a stand-alone tool to gain insight into the distribution of data, to observe the characteristics of each cluster, and to focus on a particular set of clusters for further analysis. Alternatively, it may serve as a pre-processing step for other algorithms, such as characterization, attribute subset selection, and classification, which would then operate on the detected clusters and the selected attributes or features.

Data clustering is under vigorous development. Contributing areas of research include data mining, statistics, machine learning, spatial database technology, biology, and marketing. Owing to the huge amounts of data collected in databases, cluster analysis has recently become a highly active topic in data mining research.

In machine learning, clustering is an example of unsupervised learning. Unlike classification, clustering and unsupervised learning do not rely on predefined classes and class-labeled training examples. For this reason, clustering is a form of learning by observation, rather than learning by examples. In data mining, efforts have focused on finding methods for efficient and effective cluster analysis in large databases. Active themes of research focus on the scalability of clustering methods, the effectiveness of methods for clustering complex shapes and types of data, high-dimensional clustering techniques, and methods for clustering mixed numerical and categorical data in large databases and method for clustering uncertain data sets.

2.2.2 A Categorization of Clustering Methods

A large number of clustering algorithms exist in the literature. It is difficult to provide a crisp categorization of clustering methods since these categories may overlap so that a method may have features from several categories. In general, the major clustering methods can be classified into the following categories.

Partitioning methods:

Given a database of n objects and K, the number of clusters to form, a partitioning algorithm organizes the objects into K partitions ($K \leq n$), where each partition represents a cluster.

The clusters are formed to optimize an objective partitioning criterion, often called a similarity function, such as distance, so that objects within a cluster are similar whereas objects of different clusters are dissimilar in terms of the database attributes.

The most well-known and commonly used partitioning methods are K-means method [39] where each cluster is represented by the mean value of the objects in the cluster and its extension K-modes method [31] to handle categorical data, K-medoids [35], where each cluster is represented by one of the objects located near the center of the cluster and their variations.

Hierarchical methods:

A hierarchical clustering method works by grouping data into a tree of clusters. Hierarchical clustering methods can be further classified into agglomerative and divisive hierarchical clustering, depending on whether the hierarchical decomposition is formed in a bottom-up or top-down fashion.

Agglomerative hierarchical clustering : this bottom-up strategy starts by placing each object in its own cluster and then merges these atomic clusters into larger and larger clusters, until all of the objects are in a single cluster or until certain termination conditions are satisfied.

Divisive hierarchical clustering : this top-down strategy does the reverse of agglomerative hierarchical clustering by starting with all objects in one cluster. It subdivides the cluster into smaller and smaller pieces, until each object forms a cluster on its own or until it satisfied certain termination conditions, such as a desired number of clusters is obtained or the distance between the two closest clusters is above a certain threshold distance.

The methods are introduced in this category are : BIRCH [65], CURE [27]...

Density-based methods:

The general idea is to continue growing the given cluster as long as the density (number of objects or data) in the neighborhood exceeds some threshold; that is, for each data point within a given cluster, the neighborhood of a given radius has to contain at least a minimum number of points. Such as DBSCAN [23], OPTICS [2],etc.

Grid-based methods:

Grid-based methods divide the object space into a finite number of cells that form a grid structure on which all of the operations for clustering are performed. Some typical examples of the grid-based approaches include STRING [62], WAVECLUSTER [48], CLIQUE [1]...

Model-based methods:

These methods attempt to optimize the fit between the given data and some mathematical model. Model-based methods follow two major approaches : a satatistical approach or a neural network approach.

The choice of clustering algorithm depends both on the type of data available and on the particular purpose of the application. If cluster analysis is used as a descriptive or exploratory tool, it is possible to try several algorithms on the same data to see what the data may disclose.

2.3 K-modes method as Partitioning method

2.3.1 Introduction

As is well known, K-means method [39] has been a very popular technique for partitioning large data sets with numerical attributes. However, data mining applications frequently involve many data sets that also consist of categorical attributes. One approach [44] was proposed to using the K-means algorithm to cluster categorical data. This approach consists in converting multiple category attributes into binary attributes (using 0 and 1 to represent either a category absent or present) and treat the binary attributes as numeric in the K-means algorithm. If it is used in data mining, this approach needs to handle a large number of binary attributes because categorical attributes in data sets often have hundreds or thousands of categories. This will increase both computational and space costs of the K-means algorithm. Another drawback is that the cluster means, given by real values between 0 and 1, not really reflect the characteristics of the clusters.

Indeed, in applying K-means method to categorical data, two main problems are encountered, namely, the construction of clusters' centers and the definition of dissimilarity between objects and clusters' centers. So, the K-modes method [31] was proposed to tackle the problem of clustering large categorical data sets in data mining. This method uses the K-means paradigm to cluster data having categorical values.

2.3.2 The K-modes method parameters

The K-modes method extends the K-means [39] one by using a simple matching dissimilarity measure for categorical objects, modes instead of means for clusters, and a frequency-based method to update modes in the clustering process to minimize the clustering cost function.

The mentioned modifications to the K-means algorithm are discussed as follows:

Cluster's mode

Given a cluster $\{X_1, ...X_p\}$ of p categorical objects, with $X_i = (x_{i,1}, .., x_{i,s})$, $1 \leq i \leq p$, its mode $Q = (q_1, .., q_s)$ is defined by assigning q_j, $1 \leq j \leq s$, where s is the number of attributes, the category most frequently encountered in $\{x_{1,j}, ..., x_{p,j}\}$.

However, the mode of cluster is not generally unique. This makes the algorithm unstable depending on mode selection during the clustering process.

The dissimilarity measure

We assume that the set of objects to be clustered is stored in dataset $\mathbf{D}$ defined by a set of attributes $A_1, A_2, ..., A_s$ with domains $D_1, D_2, ..., D_s$, respectively. Each object in $\mathbf{D}$ is represented by a tuple $t \in D_1 * D_2 \times ... \times D_s$.

The distance between two categorical objects $X_1, X_2 \in \mathbf{D}$, with $X_1 = (x_{1,1}, ..., x_{1,s})$ and $X_2 = (x_{2,1}, ..., x_{2,s})$ consisting of categorical values only, can be defined as follows:

$$d(X_1, X_2) = \sum_{j=1}^{s} \delta(x_{1,j}, x_{2,j}), \tag{2.1}$$

where:

$$\delta(x_{1,j}, x_{2,j}) = \begin{cases} 0 & if x_{1,j} = x_{2,j} \\ 1 & if x_{1,j} \neq x_{2,j} \end{cases} \tag{2.2}$$

$\delta(x_{1,j}, x_{2,j})$ is considered as the simple matching respectively to one attribute A_j.
For these two objects, if all attributes have the same values, this distance measure d will be equal to zero. However, when these two objects are totally distinct, it means that all attributes' values are different, so this measure will be equal to the number of attributes s. In other word, $0 \leq d \leq s$.
It is easy to verify that the function d defines a metric space on the set of categorical objects.

When the above is used as the dissimilarity measure for categorical objects, the optimization problem for partitioning a set of n objects described by m categorical attributes into k clusters becomes:

$$Minimize\ P(W, Q) = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{s} w_{i,l} \delta(x_{i,j}, q_{l,j}) \tag{2.3}$$

subject to:

$$\sum_{l=1}^{k} w_{i,l} = 1,\ 1 \leq i \leq n,\ 1 \leq l \leq k,\ and\ w_{i,l} \in \{0, 1\}$$

where W is an $n \times k$ partition matrix, $w_{i,l}$ is the degree of membership of the object X_i in the cluster C_l (using 1 and 0 to represent either the object X_i is an element of the cluster C_l or not), and $Q = \{Q_1, Q_2, ..., Q_k\}$.
This function is an indicator of the distance of the n data points from their respective clusters' modes.

2.3.3 The K-modes Algorithm

K-modes method is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume K clusters) fixed a priori.

The main idea is to define K modes, one for each cluster. These modes should be placed in a cunning way because of different location causes different results.

So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest mode using the simple matching dissimilarity measure. When no point is pending, the first step is completed and an early groupage is done. At this point we need to recalculate K new modes of the clusters resulting from the previous step based on frequency-based method.
After we have these K new modes, a new binding has to be done between the same data set points and the nearest new mode. A loop has been generated. As a result of this loop we may notice that the K modes change their location step by step until no more changes are done. In other words modes do not move any more.

Although it can be proved that the procedure will always terminate, the K-modes algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster modes. The K-modes algorithm can be run multiple times to reduce this effect.

This algorithm has as an input a predefined number of clusters, that is the K from its name. K-modes algorithm is a simple, iterative procedure, in which a crucial concept is the one of mode.
Mode is an artificial point in the space of records which represents all objects of the particular cluster. The coordinates of this point are the most frquently of attribute values of all examples that belong to the cluster.

The algorithm is composed of the following steps:

1. Select K initial modes, one for each cluster.

2. Allocate an object to the cluster whose mode is the nearest to it according to the simple matching dissimilarity measure. Update the mode of the cluster after each allocation.

3. Once all objects have been allocated to clusters, retest the dissimilarity of objects against the current modes. If an object is found such that its nearest mode belongs to another cluster rather than its current one, reallocate the object to that cluster and update the modes of both clusters.

4. Repeat 3 until no object has changed clusters after a full cycle test of the whole data set.

Example 2.1 *Let's treat a classification problem of firm's staff. Suppose that we have a training set T, defined by Table 2.1.*

Table 2.1: Training set T relative to the standard K-modes method

Objects	Qualification	Income	Department
X_1	A	High	Finance
X_2	B	Low	Finance
X_3	C	Average	Marketing
X_4	C	Average	Accounts
X_5	B	Low	Marketing
X_6	A	High	Finance
X_7	B	Low	Accounts

Suppose that $K = 3$, *3-partition of T is initialized randomly as follows:* $C_1 = \{X_1\}$, $C_2 = \{X_2\}$, *and* $C_3 = \{X_3\}$.

The three cluster modes , one for each cluster are defined by:
Q_1=*(A,High,Finance).*
Q_2=*(B,Low,Finance).*
Q_3=*(C,Average,Marketing).*

For each object X_i, $i \in \{4, ..., 7\}$, *compute the dissimilarities:* $d(X_i, Q_l)$, $l = 1,...,3$, *using Equation 2.1 and Equation 2.2.*

- *For example for the object* X_4 *the three dissimilarity measures are computed as follows:*
  $d(X_4, Q_1) = \sum_{j=1}^{3} \delta(x_{4,j}, q_{1,j})$
  $= 1 + 1 + 1 = 3.$
  $d(X_4, Q_2) = \sum_{j=1}^{3} \delta(x_{4,j}, q_{2,j})$
  $= 1 + 1 + 1 = 3.$
  $d(X_4, Q_3) = \sum_{j=1}^{3} \delta(x_{4,j}, q_{3,j})$
  $= 0 + 0 + 1 = 1.$
  So, X_4 *is assigned to* C_3, *as* $d(X_4, Q_3)$ *is minimal.*
  It is the same for the other objects.

- X_5 *is assigned to* C_2 *or to* C_3, *as* $d(X_5, Q_2)$ *and* $d(X_5, Q_3)$ *are minimal.*

- X_6 *is assigned to* C_1, *as* $d(X_6, Q_1)$ *is minimal.*

- X_7 *is assigned to* C_2, *as* $d(X_7, Q_2)$ *is minimal.*

We obtain these clusters:
$C_1 = \{X_1, X_6\}$

$C_2 = \{X_2, X_5, X_7\}$
$C_3 = \{X_3, X_4\}$

with updated modes:
Q_1=*(A,High,Finance).*
Q_2=*(B,Low,Finance) or* Q_2=*(B,Low,Marketing) or* Q_2=*(B,Low,Accounts)*
Q_3=*(C,Average,Marketing) or* Q_3=*(C,Average,Accounts).*

The mode of the second cluster Q_2 *is not unique. We take randomly into account the third possible value of* Q_2*. It is the same for the third mode cluster, the first possible value of* Q_3 *is randomly chosen. Then, the process continue until we will obtain the stable partition.*

After retesting the dissimilarity of objects against the current modes, all objects are found that are in the nearst cluster. So, the obtained partition is the final one as it is stable.

Once, the clusters are built, one instance to classify is as follows (Finance, High, C) is affected to first cluster C_1 *since* $d(X, Q_1)$ *is minimal.*

Advantages:

- Simple and understandable.
- The K-modes algorithm converges in a finite number of iterations.

Drawbacks:

This standard version of the K-modes method has some weaknesses:

- The way to initialize the modes was not specified. One popular way to start is to randomly choose K of the samples. So, the produced results depend on the initial values for the modes. The standard solution is to try a number of different starting points.
- The results depend on the value of K. Unfortunately there is no general theoretical solution to find the optimal number of clusters for any given data set. A simple approach is to compare the results of multiple runs with different K classes and choose the best one according to a given criterion such that the clustering cost function.
- The mode of cluster is not generally unique. This makes the algorithm unstable depending on mode selection during the clustering process.

2.3.4 Extensions of the K-modes method

Two alternative extensions of the K-modes algorithm [33, 40] aim at eliminating the mentioned non-uniqueness drawback in the K-modes method by introducing new notions of cluster modes.
In the next section, we expose the K-representatives method and the other extension handling with the above K-modes method limitation.

K-Representatives Method

In applying K-means method to categorical data, two main problems are encountered, namely, how to compute clusters' modes and the measure of dissimilarity between objects and clusters' modes.

The K-Representatives algorithm [40] extends K-modes algorithm by using these following concepts.

- Cluster's mode or representative:

  Given a cluster $C = \{X_1, ..., X_p\}$ of p categorical objects, with:
  $X_i = (x_{i,1}, ..., x_{i,s})$, $1 \leq i \leq p$, denote by D_j the set formed from categorical values $x_{1,j}, ..., x_{p,j}$, $1 \leq j \leq s$.
  Then, the representative of C is defined by $Q = (q_1, ..., q_s)$, with

  $$q_j = \{(c_j, f_{c_j}) | c_j \in D_j\} \tag{2.4}$$

  Where f_{c_j} is the relative frequency of category c_j within C,i.e, $f_{c_j} = nc_j/p$, where nc_j is the number of objects in C having category c_j at attribute A_j.

- Dissimilarity measure:

  The dissimilarity between a categorical object and the representative of a cluster is defined based on simple matching as follows:

  $$d(X, Q) = \sum_{j=1}^{s} \sum_{c_j \in D_j} f_{c_j}.\delta(x_j, c_j) = \sum_{j=1}^{s} \sum_{c_j \in D_j, c_j \neq x_j} f_{c_j} = \sum_{j=1}^{s} (1 - f_{x_j}) \tag{2.5}$$

  where f_{x_j} is the relative frequency of category x_j within C.

For this extension, even if the category x_j of one object corresponding to the attribute A_j has not the highest frequency, this frequency is taken into account in the representation of the mode of the cluster to which this object belongs. So, we will consider this frequency and the matching between the object and the mode respectively to this attribute A_j will be $(1 - f_{x_j})$ instead of zero comparing to the standard K-modes method.

Example 2.2 *Let's continue with our example (Example 2.1). Suppose that the same first partition is considered which is initialized randomly as follows for* $K = 3$*:* $C_1 = \{X_1\}$, $C_2 = \{X_2\}$, *and* $C_3 = \{X_3\}$.

The three cluster modes or K representatives, one for each cluster are defined by:
Q_1=$(\{(A,1),(B,0),(C,0)\};\{(High,1),(Low,0),(Average,0)\};\{(Finance,1),(Marketing,0),(Accounts,0)\})$.
Q_2=$(\{(A,0),(B,1),(C,0)\};\{(High,0),(Low,1),(Average,0)\};\{(Finance,1),(Marketing,0),(Accounts,0)\})$.
Q_3=$(\{(A,0),(B,0),(C,1)\};\{(High,0),(Low,0),(Average,1)\};\{(Finance,0),(Marketing,1),(Accounts,0)\})$.

For each object X_i, $i \in 4,...,7$, *compute the dissimilarities:* $d(X_i, Q_l)$, $l = 1,...,3$ *using Equation 2.5.*

- *For example for the object* X_4 *the three dissimilarity measures are computed as follows:*
  $$d(X_4, Q_1) = \sum_{j=1}^{3}(1 - f_{x_{4,j}}) = 1 + 1 + 1 = 3.$$
  $$d(X_4, Q_2) = \sum_{j=1}^{3}(1 - f_{x_{4,j}}) = 1 + 1 + 1 = 3.$$
  $$d(X_4, Q_3) = \sum_{j=1}^{3}(1 - f_{x_{4,j}}) = 0 + 0 + 1 = 1.$$
  So, X_4 *is assigned to* C_3, *as* $d(X_4, Q_3)$ *is minimal.*

  Continuing the same process for the other objects.

- X_5 *is assigned to* C_2, *as* $d(X_5, Q_2)$ *is minimal.*
- X_6 *is assigned to* C_1, *as* $d(X_6, Q_1)$ *is minimal.*
- X_7 *is assigned to* C_2, *as* $d(X_7, Q_2)$ *is minimal.*

We obtain:
$C_1 = \{X_1, X_6\}$
$C_2 = \{X_2, X_5, X_7\}$
$C_3 = \{X_3, X_4\}$

with updated representatives:
Q_1=($\{(A,1),(B,0),(C,0)\}$;$\{(High,1)$,$(Low,0)$,$(Average,0)\}$;$\{(Finance,1)$, $(Marketing,0)$,$(Accounts,0)\}$).
Q_2=($\{(A,0),(B,1),(C,0)\}$;$\{(High,0)$,$(Low,1)$, $(Average,0)\}$;$\{(Finance,1/3)$,$(Marketing,1/3)$,$(Accounts,1/3)\}$).
Q_3=($\{(A,0),(B,0),(C,1)\}$;$\{(High,0)$,$(Low,0)$, $(Average,1)\}$;$\{(Finance,0)$,$(Marketing,1/2)$,$(Accounts,1/2)\}$).

Recalculate the dissimilarity measures for each object.

Objects have not changed clusters : ***The partition is stable****.*

Recently, another extension of K-modes method was introduced [33] which considers the relative frequencies of attribute values in each cluster mode.

- Cluster's mode:

  The mode of C is defined by $Q = (q_1, ..., q_s)$, with $q_j = (c_j, f_{c_j})|c_j \in D_j$, where f_{c_j} is the relative frequency of the highest frequency category of attribute A_j within C.

- Dissimilarity measure:

  If we consider that d_j is the dissimilarity measure respectively to only one attribute A_j.

$$d_j(X,Q) = \begin{cases} 1 \; - \; f(A_j = q_j) & \text{if } x_j = q_j \\ 1 & \text{if } x_j \neq q_j \end{cases} \tag{2.6}$$

  where f is the relative frequency of x_j in C.
  So, the dissimilarity measure between one object X and one cluster's mode Q will be defined as follows:

$$d(X,Q) = \sum_{j=1}^{s} d_j(X,Q) \tag{2.7}$$

Here, we have just added the frequency, corresponding to each attribute's value of the object,which will be taken into account in the representation of the clusters' modes. Once, this frequency is mentioned and when the object and the mode have the same attribute's value, instead of the zero value, this measure d_j will be equal to one minus this frequency's value.

2.3.5 K-modes method under uncertainty

Standard versions of the K-modes method and its extensions give good results in a context in which everything is known with certainty. However, the reality is connected to uncertainty and imprecision by nature. Such uncertainty may badly affect the classification performance. However, a good classifier must be able to predict the object's class value even when information concerning the object is imperfect.

So, the K-modes is inadequate and badly adapted to ensure its role of classification in an environment characterized by a lot of uncertainty and imprecision. That is why searches are oriented to improvement and extension of this method, in order to adapt it to this kind of environment.

The idea is to combine theories managing uncertainty and imprecision with the K-modes method, these theories are probability theory, fuzzy set theory, belief function theory and possibilistic theory.

Hence, this adaptation of K-modes method to an uncertain environment has led to a new approach, and the fuzzy K-modes method [32] was developed. In this approach, one object does not belong exclusively to a well defined cluster. In fact, it may belong to several clusters with different membership degrees. This extension is briefly presented in what follows.

Fuzzy K-modes method

Fuzzy K-modes method deals with cognitive uncertainty. It can take into account imprecision and fuzziness in object class memberships using fuzzy sets and membership degrees.

Fuzzy K-modes approach is a method of clustering which allows one piece of data to belong to two or more clusters. It uses fuzzy partitioning such that a data point can belong to all groups with different membership grades between 0 and 1.
It is based on minimization of the following objective function (its model is the constrained optimization problem):

$$Minimize\ P(W,Q) = \sum_{l=1}^{k}\sum_{i=1}^{n} w_{i,l}^{\alpha} d(X_i, Q_l) \tag{2.8}$$

Subject to:

$$0 \leq w_{i,l} \leq 1, \qquad 1 \leq i \leq n,\ 1 \leq l \leq k \tag{2.9}$$

$$\sum_{l=1}^{k} w_{i,l} = 1, \qquad 1 \leq i \leq n \tag{2.10}$$

and

$$0 < \sum_{i=1}^{n} w_{i,l} < n, \qquad 1 \leq l \leq k \tag{2.11}$$

Where k($\leq n$) is a known number of clusters, α, any real number greater than 1, is weighting exponent, $w_{i,l}$ is the degree of membership of X_i in the cluster C_l, X_i is the i^{th} of s-dimensional measured data (s is the number of attributes), Q_l is the s-dimension mode of the cluster C_l, and d is the simple matching distance expressing the similarity between any measured data and the mode. Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership $w_{i,l}$ by:

$$w_{i,l} = \begin{cases} 1 & \text{if } X_i = Q_l \ (1) \\ 0 & \text{if } X_i = Q_h,\ h \neq l \ (2) \\ w_{i,l} = \frac{1}{\sum_{h=1}^{K} [\frac{d(X_i,Q_l}{d(X_i,Q_h}]^{\frac{1}{\alpha-1}}}\ , & \text{if } X_i \neq Q_l \text{ and } X_i \neq Q_h,\ 1 \leq h \leq k \ (3) \end{cases} \tag{2.12}$$

(1): It means that the object X_i has the same attributes' values than the mode Q_l.
(2): When the attributes' values of the object X_i are the same than another mode $Q_h \neq Q_l$.
(3): The object X_i is different (the attributes' values) from all clusters' modes.

The cluster's mode Q_l is computed as follows:

$$Q_l = \frac{\sum_{i=1}^{n} w_{i,l}^{\alpha} X_i}{\sum_{i=1}^{n} w_{i,l}^{\alpha}} \tag{2.13}$$

The algorithm is composed of the following steps:

1. Choose an initial point $Q^{(1)}$. Determine $W^{(1)}$ (using Equation 2.12) such that $P(W, Q^{(1)})$ is minimized. Set $t = 1$.

2. Determine $Q^{(t+1)}$ (see Equation 2.13) such that $P(W^{(t)}, Q^{(t+1)})$ is minimized. If $P(W^{(t)}, Q^{(t+1)}) = P(W^{(t)}, Q^{(t)}) < e$ then STOP; otherwise return to step 3.

3. Determine $W^{(t+1)}$ such that $P(W^{(t+1)}, Q^{(t+1)})$ is minimized. If $P(W^{(t+1)}, Q^{(t+1)})$ $= P(W^{(t)}, Q^{(t+1)}) < e$ then STOP; otherwise set $t = t + 1$ and go to step 2.

The fuzzy K-modes algorithm produces a fuzzy partition matrix W. We obtain the cluster membership from W as follows. The record X_i was assigned to the lth cluster if $w_{i,l} = max_{1 \leq h \leq k}\{w_{i,h}\}$. If the maximum was not unique, then X_i was assigned to the cluster of first achieving the maximum.

Remarks:

As presented in the first chapter of this work, the belief function theory as interpreted in the Transferable Belief Model (TBM) [50, 54, 56, 57] provides a mathematical tool to treat subjective, and personal judgments on the different parameters of any classification problems and can be easily extended to deal with objective probabilities. It expresses partial beliefs in a much more flexible way than probability functions do. Besides, it permits to handle partial or even total ignorance concerning classification parameters.

This theory offers interesting tools to combine several pieces of evidence [47, 49, 52], like the conjunction and the disjunction rules of combination, and decision making is solved through the pignistic transformation [60].

Inspite of its several advantages [59, 60], the belief function theory had not yet been applied to the K-modes method.

Hence, the belief function theory provides a convenient tool to handle uncertainty in any clustering methods.

2.4 EVCLUS as Belief Clustering Method

We have presented in the previous section one extension of K-modes method within uncertainty context, using fuzzy theory in order to adapt it to this kind of environment.

Belief function theory as a theory managing uncertainty was applied to clustering problem and that is the purpose of these works [16, 17, 18].

A novel approach was presented to clustering proximity data, based on the theory of belief functions. This approach is called evidential clustering (EVCLUS) [16]. The allocation of objects to clusters is performed using the concept of basic belief assignment (bba), whereby a mass of belief is assigned to each possible subset of clusters. Having assigned a bba to each object, it is possible to compute, for each two objects the plausibility that they belong to the same cluster.
It was shown that the possibility to assign masses not only to single clusters but also

to subsets of clusters or to the empty set makes the proposed partitioning model more general than the classical hard or fuzzy ones.

Another proposed approach [18] extended the first one from real valued to interval-valued dissimilarity data. It is proposed to relate the upper and lower dissimilarities to two quantities: the plausibility that the objects belong or not to the same cluster. The idea is as follows: if an observed interval-valued dissimilarities matrix is given, we have to find a credal partition. It means that this method attempts to find one configuration of points that match as well as possible the input dissimilarities. To this end, a stress function is introduced and it is at minimize.

Note that, these works deal with only the uncertainty related to the clusters' membership of the objects and not to the attributes values which characterize them.

2.5 Conclusion

In this chapter, we have presented the basic concepts of the K-modes method. The different parameters of this method are discussed and then, we have given an example to explain the K-modes procedure from a given data set.

K-modes method is considered as one of the most known clustering techniques especially in artificial intelligence applications where attributes have categorical values.

However, this method has some drawbacks. So, two extensions [33, 40] of this method were developed to deal especially with the problem of non-uniqueness of the modes within a certain context.

The K-modes method was developed to extend the K-means one to handle categorical data sets. When, we are in presence of mixed numerical and categorical data, the K-prototypes method [29] which integrates the K-means and the K-modes processes, is recommended to use in order to cluster this type of data.

Through this presentation,we conclude that, despite the advantages provided by K-modes method and the improvements given as fuzzy K-modes, many researches are still needed in order to deal with the uncertainty especially the cognitive one, that many occur in the different parameters related to any classification problem.

The belief function theory as understood in the Transferable Belief model (TBM) seems to be one of the appropriate formalism to cope with this kind of uncertainty. In spite of its several advantages, the belief function theory had not yet been applied to the K-modes method to handle uncertainty problem. However, it was already applied to clustering problem as a theory managing uncertainty with classification problem in [16, 17, 18].

Thus, our objective will be the adaptation of this theory to the K-modes method and to develop what we call a belief K-modes method that will be presented in a following part of this work.

Part II

Belief K-modes Method

Chapter 3

Belief K-modes Method

3.1 Introduction

K-modes approach is considered as an efficient clustering method for classification problem. That's why, it is widely applied to a variety of problems in artificial intelligence. Despite its accuracy when precise and certain data are available, the standard K-modes algorithm shows serious limitations when dealing with uncertainty. Such uncertainty may affect the parameters of any classification problem and can appear either in the construction or in the classification phase as discussed in chapter 2.

In this standard method, we have dealt with training sets characterized by certain training objects while their attributes' values are supposed to be known with certainty. However, uncertainty may appear in attributes of instances belonging to the training set that will be used to ensure the construction of clusters. Faced to uncertain parameters, the standard method seems to be unable to provide significant classification results.

To overcome this limitation, we propose as solution to develop what we call a belief K-modes method, a new clustering technique based on the K-modes method within the belief function theory in order to deal with uncertainty that may pervade any classification problems. This theory for uncertainty representation, as understood in the Transferable Belief Model (TBM), provides a convenient framework for managing and manipulating uncertain knowledge, especially the cognitive one.

This chapter is consecrated to the presentation of this new approach under uncertainty. We first give some motivations to develop this method for handling this kind of uncertainty. Then, we define the new structure of the training set within the belief function framework and the different parameters leading to the construction of clusters namely the computation of the cluster modes and the dissimilarity measure to use in such a case. Once the clusters are obtained, the next step which is the classification of new instances that may be uncertain will be detailed. Finally, the algorithm of the belief K-modes method (BKM) with an illustrative example are reported.

3.2 Definition and motivations

Our belief K-modes method which is based on an extension of the K-modes algorithm while taking into account the uncertainty of some parameters related to the classification problem using the belief function theory for building clusters. A belief K-modes method is a K-modes method in an uncertain environment. The uncertainty will be represented and handled by the means of the belief function theory.

Contrary to the standard and non-standard K-modes methods (fuzzy K-modes method) and also the belief clustering method (EVCLUS) where object attribute values in the training sets are known with certainty, in the belief K-modes method, these values may be affected with uncertainty. In the first part, we deal with uncertain training sets, and in the second part, we make our approach able to classify instances that may have some or even all attributes with uncertain values. So, such uncertainty can appear either in the building of the clusters or the classification phase when we have a new object to classify.

Dealing with uncertainty is represented through belief functions in attribute values leads to two main problems:

1. The strategy to apply for computation of the cluster modes in each step of the algorithm. In other words, how to compute attributes' values of the cluster modes.

2. The dissimilarity measure, that means, which distance measure will be used where objects may be characterized by uncertain attributes' values and this uncertainty is represented by the belief function theory.

Hence, the two parameters useful for the BKM method have to be defined, in such a context within uncertainty, such as the strategy to compute the cluster modes and the distance measure.

The main hypotheses of our work can be summarized as follows:
First, we have to define the structure of the training set under this uncertain framework. Second, we must determine a new distance measure within uncertain environment and the strategy to calculate the cluster modes that should be defined.

It consists in the construction of clusters from a training set of objects using successive refinements. This set is the basis of the construction of these clusters and consequently to the classification of new instances. Thus, our approach deals with uncertainty in both building and classification procedures.
Note that all over this report, we only deal with categorical attributes.

3.3 Objectives

The objective of this work is to develop a new concept that we will call belief K-modes method. In addition to the objectives of the standard K-modes method, the belief K-modes one aims at ensuring two major objectives:

- Building K clusters from a given set of training instances characterized by uncertainty on their attributes' values.
- Ensuring the classification of new instances that may be described by uncertain or even unknown attributes' values.

This new approach is based on both the K-modes method and the belief function theory in order to cope with uncertainty problem.

3.4 Notations

We will use the following notations, in this work:

- T: a given data set of objects.
- X_i: an object or instance, $i = 1, ..., n$.
- $A = \{A_1, ..., A_s\}$: a set of s attributes.
- Θ_j: the frame of discernment involving all the possible values of the attribute A_j related to the classification problem, j=1,...,s.
- D_j: the power set of the attribute $A_j \in A$.
- $x_{i,j}$: the value of the attribute A_j for the object X_i.

- $m^{\Theta_j}\{X_i\}$: expresses the beliefs on the values of the attribute A_j corresponding to the object X_i.
- $m_i(c_j)$: denoted the bba given to $c_j \subseteq \Theta_j$ relative to the object X_i.
- $m_l(c_j)$: denoted the bba given to $c_j \subseteq \Theta_j$ relative to the mode Q_l of the cluster C_l.

3.5 Structure of training set

3.5.1 Definition

Generally, objects that belong to the training set are known with accuracy and the value of each one of its attributes is certain.

The structure of the training set may be different from the traditional one, due to the uncertainty introduced here related to training instances' attributes. So, this structure will change. Instead of assigning for each attribute of an object a unique value, it will be labeled by a bba or mass function expressing a belief on the actual attribute value of objects.
Unlike the standard training set, we assume it may contain data where there is some uncertainty in the knowledge of the attribute values. In other words, each attribute of the training instances may be uncertain or even unknown.

We propose to represent the uncertainty on the attributes values of the training instances by a basic belief assignment (bba) defined on the set of possible values considered in the classification problem. This bba, generally given by an expert (or several experts), represents the opinions-beliefs of this expert about the actual values of the attributes for each object in the training set.

Among the advantages of working under the belief function framework, we notice that the two extreme cases, total ignorance and total knowledge which are easily expressed [45].

Example 3.1 *This is an example, to illustrate our new structure of the training set T within the belief function framework (see Table 3.1). We assume there are seven objects $X_i(i \in \{1, ..., 7\})$. Assume we want to classify these objects by taking into account their attributes. The training set instances are characterized by three categorical attributes defined as follows:*

Table 3.1: Structure of training set relative to BKM

Objects	Qualification	Income	Department
X_1	A	High	$m^{\Theta_3}\{X_1\}$
X_2	B	Low	$m^{\Theta_3}\{X_2\}$
X_3	C	Average	$m^{\Theta_3}\{X_3\}$
X_4	C	Average	$m^{\Theta_3}\{X_4\}$
X_5	B	Low	$m^{\Theta_3}\{X_5\}$
X_6	A	High	$m^{\Theta_3}\{X_6\}$
X_7	B	Low	$m^{\Theta_3}\{X_7\}$

- *Qualification with possible values* $\{A, B, C\}$.
- *Income with possible values* $\{High, Low, Medium\}$.
- *Department with possible values* $\{Finance, Accounts, Marketing\}$.

For each attribute A_j *for an object* X_i *belonging to the training set* T*, we assign a bba* $m^{\Theta_j}\{X_i\}$ *expressing beliefs on its assigned attributes' values. These functions are defined respectively on the same frame of discernment* $\Theta_j\{j = 1, 2, 3\}$
$\Theta_1 = \{A, B, C\}$
$\Theta_2 = \{High, Low, Average\}$
$\Theta_3 = \{Finance, Accounts, Marketing\}$

If we consider that only the department attribute is known with uncertainty and the two other attributes (qualification and income) are known with certain and unique value. The structure of the data set T *is be defined in Table 3.1.*

For example, if we have for the object X_2*:* $m^{\Theta_3}\{X_2\}(Marketing) = 0.8$ *and* $m^{\Theta_3}\{X_2\}(\Theta_3) = 0.2$*. It means that 0.8 of beliefs are exactly committed to the Marketing department, whereas 0.2 is assigned to the disjunction of attribute values (department nature), i.e., 0.2 is assigned to the whole frame of discernment (ignorance).*

3.5.2 Special cases

Within the belief function framework, two extreme cases such as **the total ignorance** and **the total knowledge** regarding training instances attributes' values can be easily expressed.

1. When the attributes' values of the object X_i are perfectly known and are unique, it will be represented by **a certain basic belief assignment** (see Section 1.3.5)

having the only focal element this attribute value. This case is referred to as **total knowledge** and corresponds to the classical 'certain' context.

2. When the expert is not able to give any information about the instances especially about attribute values. Thus, the bba will be **a vacuous basic belief assignment** (see Section 1.3.1). This case is referred to as **total ignorance**.

3.6 BKM parameters

3.6.1 Introduction

As with standard K-modes method, building clusters within BKM needs the definition of its fundamental parameters, namely, cluster modes and the dissimilarity measure. These parameters must take into account the uncertainty encountered in the training set and that pervades the attribute values of training objects.

3.6.2 How to compute the cluster modes

Due to the uncertainty and contrary to the traditional training set where it includes only certain instances, the structure of our training set will be represented via bba's respectively to each attribute relative to each object, this training set offers a more generalized framework than the traditional one.

Two extreme cases should be noted, when one attribute is known with certainty, it will be represented by a certain belief function (see Equation 1.29), whereas when it is missing we will use the vacuous belief function (see Equation 1.20).

Within this structure of training set, our belief K-modes method cannot use the strategy used by the standard method which is the frequency-based method to update modes of clusters.

One of the fundamental parameters in the K-modes method (and consequently in a belief K-modes method) is the strategy to calculate the cluster modes. This measure is applied in order to compute at each step the attribute values of the cluster modes.

If we consider that objects belonging to one cluster are sources of information which provide distinct pieces of evidence, we can combine them in at least two ways as conjunctively or disjunctively using respectively the conjunctive or disjunctive rules of combination.

An intuitive definition of the strategy to calculate the cluster modes within the belief function theory context will be the conjunctive rule of combination generally used as an aggregate operator in the belief function framework combining between two or several bba's.

This operator is particularly suitable when distinct sources provide pieces of evidence respectively to the same object. However, in our case, the induced pieces of evidence are related to different objects' attributes. So, the conjunctive (even disjunctive) rule is not appropriate.

Thus, our problem is as follows: what is the appropriate function to use in order to obtain a single representation value of different bba's pertaining to all objects belonging to one cluster. To perform this task, some functions can be used such as median, mean, ...

Within this uncertain environment especially belief function framework, we should first define how the attributes' values of each cluster's mode will be represented and what is the appropriate function used to express our belief corresponding to the attributes' values. We have take into account all distinct bba's of different objects belonging to one cluster.

Thus to solve our problem, the idea is to apply the mean operator to this uncertain context due to its efficiency and its simplicity.

Note that using the mean operator offers many advantages since it satisfies these properties namely the associativity, the commutativity and the idempotency. The latter property is the most important one in our case (it is not satisfied by the conjunctive rule of combination). When we have two or more objects belonging to one cluster which provide the same bbm's corresponding to any uncertain attribute, their cluster's mode should be characterized by these same bbm's (provided by its objects). In fact by applying this operator (the mean operator), we will keep the same bbm's provided by the objects and these values will characterize the attribute's value of their cluster's mode.

So, the mean operator permits combining bba's respectively to each attribute provided by all objects belonging one cluster.

Since, this operator will be applied in a belief framework, we call it **the belief mean**.

In standard version of the K-modes method, the mode of cluster is not generally unique, it means that more than one mode's value can be chosen. This drawback is known as **the non-uniqueness problem**. Note that using the averaging rule (the belief mean) will solve this problem.

Given a cluster $C_l = \{X_1, ..., X_p\}$ of objects, with $X_i = (x_{i,1}, ..., x_{i,s})$, $1 \leq i \leq p$, $1 \leq l \leq k$. Then, the mode of C_l is defined by : $Q_l = (q_{l,1}, ..., q_{l,s})$, with:

$$q_{l,j} = \{(c_j, m_l(c_j))|c_j \in D_j\}, \quad 1 \leq j \leq s \tag{3.1}$$

where $m_l(c_j)$ is the relative bba of attribute value c_j within C_l.

$$m_l(c_j) = \frac{\sum_{\mathrm{i=1}}^{\mathrm{p}} \mathrm{m_i(c_j)}}{|\mathrm{C_l}|} \tag{3.2}$$

with $C_l = \{X_1, X_2, ..., X_p\}$ and $|C|$ is the number of objects in C_l. $m_l(c_j)$ expresses the belief about the value of the attribute A_j corresponding to the cluster mode Q_l.

Example 3.2 *If we consider the same frame of discernment as in previous examples, and the bba's corresponding to the two objects X_1 and X_2 which form one cluster are respectively as follows:*
$m^{\Theta}\{X_1\}(\{Finance\}) = 1$
$m^{\Theta}\{X_2\}(\{Finance, Marketing\}) = 0.5$, $m^{\Theta}\{X_2\}(\{Accounts\}) = 0.5$
The averaging rule is applied and the results are:
$m(\{Finance\}) = 0.5$
$m(\{Accounts\}) = 0.25$
$m(\{Finance, Marketing\}) = 0.25$
It represents the bba's values of the cluster mode which is composed by these two objects X_1 and X_2.

3.6.3 Dissimilarity measure

We have to define one dissimilarity measure which verifies the following basic properties for any distance measures. So, we have studied many distance measures under the belief function framework which will be detailed in following sections. Firstly, let us introduce some basic concepts for distances:

Definition 3.1 *A metric distance defined on the set ξ is a function:*
$d : \xi * \xi \longrightarrow \Re$
$(A,B) \mapsto d(A,B)$,
that satisfies the following requirements for any A and B of ξ:

1. *Non-negativity : $d(A,B) \geq 0$. d is non negative number.*

2. *Non-degeneracy : $d(A,A) = 0$. The distance of an object to itself is 0.*

3. *Symmetry : $d(A,B)=d(B,A)$. The distance is a symmetric function.*

Note that other properties may be mentioned but are not interesting in our case. Many clustering algorithms operate on a dissimilarity matrix which stores a collection of proximities that are available for all pairs of n objects belonging to a data set to be clustered. This dissimilarity matrix is often represented by n-by-n $d(X_1, X_2)$ table. Where $d(X_1, X_2)$ is the measured difference or dissimilarity between two objects X_1 and X_2 verifying all properties defined before. The most used measure distance namely Euclidean distance is discussed throughout this section.

Euclidean distance

In this section, we will discuss how object dissimilarity can be computed for objects described by numerical variables. These dissimilarity data can later be used to compute clusters of objects.
The most popular distance measure is Euclidean distance which is defined as:

$$d(X_1, X_2) = \sqrt{(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 + ... + (x_{1s} - x_{2s})^2} \tag{3.3}$$

where $X_1 = (x_{11}, x_{12}, ..., x_{1s})$ and $X_2 = (x_{21}, x_{22}, ..., x_{2s})$ are two s-dimensional data objects.
This distance should satisfy the mathematic requirements of a distance function presented by Definition 3.1.
Within belief function framework, we define *the belief euclidean distance* as follows:

$$d(X_1, X_2) = \sqrt{\sum_{j=1}^{s} \sum_{c_j \in 2^{\Theta^j}} (m_1(c_j) - (m_2(c_j))^2} \tag{3.4}$$

where $X_1 = \vec{m}_1$ and $X_2 = \vec{m}_2$ are two s-dimensional data objects.
$\vec{m}_1$=$(m_1(A_1), m_1(A_2), ..., m_1(A_s))$ and $\vec{m}_2$=$(m_2(A_1), m_2(A_2), ..., m_2(A_s))$.
And $m_i = \{m_i(c_j) | c_j \in 2^{\Theta_j}, 1 \leq j \leq s\}$, $i = \{1, 2\}$.

This measure provides one mean to compute the dissimilarity for objects described by bba's respectively to their attributes' values (s is the number of attributes). However, this adapted measure is not appropriate in our case, since, it gives a biased result where all the subsets of Θ have the same bbm's without taking into account similarity among them as proved by the following counter example.

Counter-example 3.1 *For example, the subset {Finance} is "closer" to {Finance,Accounts} than is {Marketing}. In particular, if we consider three bodies of evidence, such as that the body of evidence is defined by Equation(1.3):*

$(\varphi,X)=\{[(Finance),0.8], [\Theta,0.2]\}$.
$(\varphi,Q_1)=\{[(Finance,Accounts),0.8], [\Theta,0.2]\}$.
$(\varphi,Q_2)=\{[(Marketing),0.8], [\Theta,0.2]\}$.

If we consider the Euclidean distance, then we have $d(X,Q_1)=d(X,Q_2)=1.13$, *even if we expect* $d(X,Q_1) < d(X,Q_2)$.

In fact, the dissimilarity measure should take into account the bbm's for each attribute for all training set objects, and compute the distance between any object and each cluster mode (represented by bba's). The used measure must be defined that describes the similarity (or likeness) between the subsets of Θ. The requirements for this distance d are then:

- d must define a metric distance.
- d must take into account the similarity among the subsets of Θ.
- d must satisfy $d(m, n_1) < d(m, n_2)$, if n_1 is closer to m than n_2 is.

Several works are concentrated on the definition of a distance between two basic belief assignments (bba's) within belief function theory framework. So, a very important number of measures were developed which can be classified into two kinds which are detailed as follows.

Distance measures based on pignistic transformation

The first category of distance measures within belief function context are those based on the pignistic transformation.
For these distances [5, 22, 61, 64], one unavoidable step is the pignistic transformation of the bba's (transformation from the power set to the frame of discernment).

This kind of distance may lose information given by the initial bba's. Besides, we can obtain the same pignistic probabilities by applying the pignistic transformation on two different bba's. So, the distance between the two obtained results does not reflect the actual similarity between the starting bba's values.

Counter-example 3.2 *Let m_1 and m_2 be two bba's defined on the same frame of discernment $\Theta_{Department} = \{Finance, Accounts, Marketing\}$.*

The two bba's are respectively:
$m^{\Theta}\{X_1\}(\{Finance\}) = 0.2$, $m^{\Theta}\{X_1\}(\{Finance, Marketing\}) = 0.3$,
$m^{\Theta}\{X_1\}(\{Accounts, Finance\}) = 0.2$, $m^{\Theta}\{X_1\}(\{\Theta_{Dept}\}) = 0.3$

$m^{\Theta}\{X_2\}(\{Finance\}) = 0.2$, $m^{\Theta}\{X_2\}(\{Finance, Marketing\}) = 0.2$,
$m^{\Theta}\{X_2\}(\{Marketing\}) = 0.1$, $m^{\Theta}\{X_2\}(\{Accounts\}) = 0.05$, $m^{\Theta}\{X_2\}(\{\Theta_{Dept}\}) = 0.45$

The pignistic transformation is applied and the results are:
$BetP_1(\{Finance\}) = 0.45$, $BetP_1(\{Accounts\}) = 0.2$, *and* $BetP_1(\{Marketing\}) = 0.35$
$BetP_2(\{Finance\}) = 0.45$, $BetP_2(\{Accounts\}) = 0.2$, *and* $BetP_2(\{Marketing\}) = 0.35$

If we will consider the two following distinct belief functions and after applying the pignistic transformation, the obtained probabilities can be the same. Hence, the distance between the two results computed via the pignistic transformation when it is equal to zero does not reflect the real distance's value between the initial belief distributions which are different. This explains the fact that this kind of distance is not suitable within this context.

Distances measures between bba's

The second kind of belief distance measures are those which are applied directly to bba's and not to the pignistic probabilities. These measures are defined on the power set [11, 24].

The second one developed by Fixen and Mahler [24] is defined as follows:
Let (β_1, m_1) and (β_2, m_2) be two bodies of evidence (see Section 1.2.3), the distance between these two bodies of evidence is defined by:

$$d_{\alpha}^2(\beta_1, \beta_2) = \alpha_q(\beta_1, \beta_1) - 2\alpha(\beta_1, \beta_2) + \alpha_q(\beta_2, \beta_2) \tag{3.5}$$

where $\alpha_q(\beta_1, \beta_2)$ is the scalar product defined as follows:

$$\alpha_q(\beta_1, \beta_2) = \sum_{A \in \beta_1} \sum_{B \in \beta_2} m_1(A) m_2(B) \frac{q(A \cap B)}{q(A) q(B)} \tag{3.6}$$

Assuming q is uniform $q(A) = |A|/|\Theta|$, for all A subsets of Θ. Equation 3.6 reduces to:

$$\alpha_q(\beta_1, \beta_2) = \sum_{A \in \beta_1} \sum_{B \in \beta_2} m_1(A) m_2(B) |\Theta| \frac{|A \cap B|}{|A|.|B|} \tag{3.7}$$

Similarity matrix D defining to this distance measure is then given by its elements:

$$D_\alpha(A, B) = \frac{|A \cap B|}{|A|.|B|}, \ \forall \ A, B \in 2^\Theta \tag{3.8}$$

The result is a pseudo-metric, since the condition of non-degeneracy of one distance metric (see Definition 3.1) is not respected, this means that $(\beta_1, m_1) \neq (\beta_2, m_2)$ exists such that $d_\alpha(\beta_1, \beta_2) = 0$.

Our idea is to adapt the belief distance defined by [11] to this uncertain clustering context to compute the dissimilarity between any object and each cluster mode since it lies in the non-degeneracy which not respected by the previous presented distance and it verifies all presented properties in Definition 3.1.
This distance measure takes into account both the bba's distributions provided by the objects and one similarity matrix D which is based on the cardinalities of the subsets of the corresponding frame of one attribute and those of the intersection and union of these subsets.

The elements of the matrix D satisfy the follow equation :

$$D(A, B) = \frac{|\mathrm{A} \cap \mathrm{B}|}{|\mathrm{A} \cup \mathrm{B}|}, \ A, B \ \in 2^\Theta \tag{3.9}$$

Let m_1 and m_2 be two bba's on the same frame of discernment Θ, the distance between m_1 and m_2 is :

$$d(m_1, m_2) = \sqrt{\frac{1}{2}(\vec{m_1} - \vec{m_2}) D (\vec{m_1} - \vec{m_2})} \tag{3.10}$$

D is an $2^\Theta \times 2^\Theta$ matrix whose elements are defined by Equation 3.9, another way to write d is:

$$d(m_1, m_2) = \sqrt{\frac{1}{2}(\|\vec{m_1}\|^2 + \|\vec{m_2}\|^2 - 2 < \vec{m_1}, \vec{m_2} >)} \tag{3.11}$$

where $< \vec{m_1}, \vec{m_2} >$ is the scaler product defined by:

$$< \vec{m_1}, \vec{m_2} >= \sum_{w=1}^{2^{\Theta_j}} \sum_{z=1}^{2^{\Theta_j}} m_1(B_w) m_2(B_z) \frac{|\mathrm{B_w} \cap \mathrm{B_z}|}{|\mathrm{B_w} \cup \mathrm{B_z}|} \tag{3.12}$$

with $B_w, B_z \in D$ for w,z = 1,...,2^{Θ}, and $\| \vec{m} \|^2$ is then the square norm of $\vec{m}$: $\|\vec{m}\|^2 = < \vec{m}, \vec{m} >$
Thus, the dissimilarity measure between any object X_i and each mode Q can be defined as follows:

$$D(X_i, Q) = \sum_{j=1}^{s} d(m^{\Theta_j}\{X_i\}, m^{\Theta_j}\{Q\}) \tag{3.13}$$

where $m^{\Theta}\{X_i\}$ and $m^{\Theta}\{Q\}$ are the relative bba of the attribute A_j provided by respectively the object X_i and the mode Q.

3.7 The BKM algorithm

If we will consider the proposed distance measure, detailed in the previous section, and the averaging rule (the belief mean) for computation the modes of the clusters and applying the K-modes method, we will obtain what we call belief K-modes method to handle categorical data within uncertainty represented by belief function theory concepts.

3.7.1 Building phase

The BKM algorithm has the same skeleton as standard K-modes method. The different construction steps of our approach are described as follows:

1. Giving K, the number of clusters to form.
2. Partition objects of the training set T in K nonempty subsets.
3. Compute seed points as the clusters' modes of the current partition using the averaging rule of combination (see Equation 3.2).
4. Assign each object to the cluster with the nearest seed point after computing its distance measures respectively to all clusters' modes defined in Equation 3.13.
5. Go back to step 3, stop when no more new assignment. In other word, all objects have no changed clusters.

Table 3.2: Data set T relative to BKM

Objects	Qualification	Income	Department
X_1	A	High	$m^{\Theta_3}\{X_1\}$
X_2	B	Low	$m^{\Theta_3}\{X_2\}$
X_3	C	Average	$m^{\Theta_3}\{X_3\}$
X_4	C	Average	$m^{\Theta_3}\{X_4\}$
X_5	B	Low	$m^{\Theta_3}\{X_5\}$
X_6	A	High	$m^{\Theta_3}\{X_6\}$
X_7	B	Low	$m^{\Theta_3}\{X_7\}$

Example 3.3 *Let us illustrate our method by a simple example. Assume that a firm wants to group its staff by taking into account their attributes.*
Let T be a training set (see Table 3.2) composed of seven instances characterized by three categorical attributes:

- *Qualification with possible values* $\{A, B, C\}$.
- *Income with possible values* $\{High, Low, Average\}$.
- *Department with possible values* $\{Finance, Accounts, Marketing\}$.

For each attribute A_j *for an object* X_i *belonging to the training set T, we assign a bba* $m^{\Theta_j}\{X_i\}$ *expressing beliefs on its assigned attributes values, defined respectively on:*
$\Theta_1 = \{A, B, C\}$
$\Theta_2 = \{High, Low, Average\}$
$\Theta_3 = \{Finance, Accounts, Marketing\}$.

If we consider that only the department attribute is known with uncertainty. The structure of the data set T is defined as in Table 3.2.

Where:

- $m^{\Theta_3}\{X_1\}(\{Finance\}) = 0.5$; $m^{\Theta_3}\{X_1\}(\{Finance, Accounts\}) = 0.3$ *and* $m^{\Theta_3}\{X_1\}(\Theta_3) = 0.2$
- $m^{\Theta_3}\{X_2\}(\{Finance\}) = 0.8$ *and* $m^{\Theta_3}\{X_2\}(\Theta_3) = 0.2$
- $m^{\Theta_3}\{X_3\}(\{Marketing\}) = 0.8$; $m^{\Theta_3}\{X_3\}(\{Finance, Accounts\}) = 0.1$ *and* $m^{\Theta_3}\{X_3\}(\Theta_3) = 0.1$
- $m^{\Theta_3}\{X_4\}(\{Accounts\}) = 0.8$ *and* $m^{\Theta_3}\{X_4\}(\Theta_3) = 0.2$

- $m^{\Theta_3}\{X_5\}(\{Marketing\}) = 0.8$ *and* $m^{\Theta_3}\{X_5\}(\Theta_3) = 0.2$
- $m^{\Theta_3}\{X_6\}(\{Finance, Accounts\}) = 0.8$ *and* $m^{\Theta_3}\{X_6\}(\Theta_3) = 0.2$
- $m^{\Theta_3}\{X_7\}(\{Accounts\}) = 0.8$ *and* $m^{\Theta_3}\{X_7\}(\Theta_3) = 0.2$

Let us now try to construct the clusters using our approach relative to the training set T. The first step is to specify K the number of clusters to form, and select the initial modes.

Suppose that $K = 2$, *2-partition of T is initialized randomly as follows:* $C_1 = \{X_1\}$, *and* $C_2 = \{X_2\}$.

For each object X_i, $i \in 3, ..., 7$, *compute the dissimilarities:* $d(X_i, Q_l)$, l = *1,...,2, using the dissimilarity measure defined by Equation 3.13.*

In order to compute the distance measures, we have first to define the matrix D that describes the similarity between the subsets of Θ_3, *its elements are calculated by Equation 3.9.*
Note that the Finance department is represented by F, Accounts by Ac and Marketing by M.
D is defined as follow:

	$\emptyset$	$Finance$	$Accounts$	$Marketing$	F, Ac	F, M	Ac, M	Θ
$\emptyset$	0	0	0	0	0	0	0	0
$Finance$	0	1	0	0	1/2	1/2	0	1/3
$Accounts$	0	0	1	0	1/2	0	1/2	1/3
$Marketing$	0	0	0	1	0	1/2	1/2	1/3
F, Ac	0	1/2	1/2	0	1	1/3	1/3	2/3
F, M	0	1/2	0	1/2	1/3	1	1/3	2/3
Ac, M	0	0	1/2	1/2	1/3	1/3	1	2/3
Θ	0	1/3	1/3	1/3	2/3	2/3	2/3	1

We have to compute the distance measures relatively to all objects $\{X_3, X_4, X_5, X_6, X_7\}$ *corresponding to the 2 initial modes.*

Let us now compute the distance measures of the object X_3 *over the two clusters' modes by the following equation (see Equation 3.13).*

$D(X_3, Q_l) = \sum_{j=1}^{3} d(m^{\Theta_j}\{X_3\}, m^{\Theta_j}\{Q_l\})$, *for* $l = \{1,2\}$.

- *For l=1*
  $D(X_3, Q_1) = \sum_{j=1}^{3} d(m^{\Theta_j}\{X_3\}, m^{\Theta_j}\{Q_1\})$

  - *For j=1: This attribute is known with certainty, so our belief dissimilarity measure is equivalent to the simple matching measure (see Equation 2.2). The object and the cluster mode have different values for this attribute (A $\neq$ C), then d=1.*

  - *For j=2: It is the same, the income attribute is a certain attribute. d=1, since High $\neq$ Average.*

  - *For j=3:*
    To simplify the expression, we will replace $m^{\Theta_3}\{X_3\}$ *by* m_3 *and* $m^{\Theta_3}\{Q_1\}$ *by* m_1.
    $d(m_3, m_1) = \sqrt{\frac{1}{2}(\|\vec{m_3}\|^2 + \|\vec{m_1}\|^2 - 2 < \vec{m_3}, \vec{m_1} >)}$
    where $\|\vec{m_3}\|^2$, $\|\vec{m_1}\|^2$, *and* $< \vec{m_3}, \vec{m_1} >$ *are the scaler products defined by Equation 3.12:*

    $\|\vec{m_3}\|^2 = (0.5 \times 0.5 \times 1) + (0.5 \times 0.3 \times 1/2) + (0.5 \times 0.2 \times 1/3) + (0.3 \times 0.3 \times 1) + (0.3 \times 0.5 \times 1/2) + (0.3 \times 0.2 \times 2/3) + (0.2 \times 0.2 \times 1) + (0.2 \times 0.5 \times 1/3) + (0.2 \times 0.3 \times 2/3) = 0.676$

    $\|\vec{m_1}\|^2 = (0.8 \times 0.8 \times 1) + (0.8 \times 0.1 \times 0) + (0.8 \times 0.1 \times 1/3) + (0.1 \times 0.8 \times 0) + (0.1 \times 0.1 \times 1) + (0.1 \times 0.1 \times 2/3) + (0.1 \times 0.8 \times 1/3) + (0.1 \times 0.1 \times 2/3) + (0.1 \times 0.13 \times 1) = 0.727$

    $< \vec{m_3}, \vec{m_1} >= (0.5 \times 0.8 \times 0) + (0.5 \times 0.1 \times 1/2) + (0.5 \times 0.1 \times 1/3) + (0.3 \times 0.8 \times 0) + (0.3 \times 0.1 \times 1) + ((0.3 \times 0.1 \times 2/3) + (0.2 \times 0.8 \times 1/3) + (0.2 \times 0.1 \times 2/3) + (0.2 \times 0.1 \times 1) = 0.178$

    So, $d(m_3, m_1) = \sqrt{1/2(0.727 + 0.676 - 2 \times 0.178)} = 0.723.$

  Hence, $D(X_3, Q_1) = 1 + 1 + 0.723 = 2.723$

- *For l=2*
  $D(X_3, Q_2) = \sum_{j=1}^{3} d(m^{\Theta_j}\{X_3\}, m^{\Theta_j}\{Q_2\})$
  - *For j=1: d=1 (B $\neq$ C).*
  - *For j=2: d=1 (Low $\neq$ Average).*
  - *For j=3: The same procedure is applied for this attribute as explained before, and we obtain d=0.776.*

  As a result, we have $D(X_3, Q_2) = 1 + 1 + 0.776 = 2.776$

Thus, for the object X_3*, and after computing its distance measures over the two clusters' modes, it is assigned to* C_1 *since* $d(X_3, Q_1) < d(X_3, Q_2)$*. It is the same for the fourth other objects.*

After that all objects have been assigned to appropriate clusters, the following clusters are obtained:
$C_1 = \{X_1, X_3, X_6\}$*, and* $C_2 = \{X_2, X_5, X_4, X_7\}$.
Next, we have to update clusters' modes. The same steps will be applied until no object has changed clusters.

We finally obtain these clusters:
$C_1 = \{X_1, X_3, X_4, X_6\}$*, and* $C_1 = \{X_2, X_5, X_7\}$*, with the corresponding modes :*
$Q_1 = (\{(A, 0.5), (C, 0.5)\}; \{(H, 0.5), (Av, 0.5)\};$
$\{(F, 0.125), (Ac, 0.2), (M, 0.2), (\{F, A\}, 0.2), (\{F, A, M\}, 0.275)\})$*, and*
$Q_2 = (\{(B, 1)\}, \{(L, 1)\}; \{(F, 0.267), (Ac, 0.267), (M, 0.266), (\{F, A, M\}, 0.2)\})$

3.7.2 Classification phase

Once the clusters' construction is done, the classification of a new object that may be characterized by uncertain attributes' values is guaranteed by our method. We have to assign it to the most similar cluster based on its distance corresponding to the obtained clusters' modes resulting from the construction phase using (see the previous section) the distance measure defined by Equation 3.13.

In standard K-modes method, each instance to classify is represented by a s-uple containing the different attribute values (e.g(Finance,High,A)), whereas, in the belief

extension, an instance is described as follows:
For each attribute , we assign a bba to express the uncertainty on the real attribute's values. Given the set of attributes A which characterize the objects, we describe the instance to classify by a vector of belief values $\vec{m} = (m(A_1), .., m(A_s))$. We notice these two following extreme cases:

- An attribute A_j whose value is known with certainty has exactly one value $c_j \in D_{A_j}$, such that $m(c_j) = 1$, and all other values $c'_j \in D_{A_j}$-$\{c_j\}$, $m(c'_j) = 0$. This is a certain bba.
- An attribute A_j whose value is totaly ignored is represented as follows: m(Θ)=1, and for all other values $c_j \in D_{A_j}$-$\{\Theta\}$, $m(c_j) = 0$.

In our approach, a new instance to classify may be certain or uncertain. The BKM approach ensures the classification of instances whose attributes' values are represented via belief functions.

It is easy to classify a new uncertain instance within BKM approach, it consists in computing the distance measures between this instance and all obtained clusters' modes (the results of the building step) and assign it to the nearest one.

Example 3.4 *Let us consider Example 3.3 and assume a new object X to classify using the belief K-modes approach. More precisely, we will use the results obtained from the building phase.*

The standard instance X_h whose attributes' values are certain will be represented by the following table.

Table 3.3: A certain instance

$m_{Qualification}$		m_{Income}		$m_{Department}$	
$\emptyset$	0	$\emptyset$	0	$\emptyset$	0
A	1	High	1	Finance	1
B	0	Low	0	Accounts	0
C	0	Average	0	Marketing	0
A,B	0	High,Lo	0	Finance,Accounts	0
A,C	0	High,Average	0	Finance,Marketing	0
B,C	0	Low,Average	0	Accounts,Marketing	0
$\Theta_{Qualification}$	0	Θ_{Income}	0	$\Theta_{Department}$	0

Once the two clusters are fixed (the final partition which is obtained by applying the building procedure), and since our method consists in a generalization of standard

K-modes method, it must be able to ensure the standard classification that means classify instances whose attributes' values are certainly known.

If we would classify a new certain instance X_h as represented before in Table 3.3. Note that the two clusters' modes Q_1 and Q_2 were already obtained in the building step in Example 3.3 as follows:
$Q_1 = (\{(A, 0.5), (C, 0.5)\}; \{(H, 0.5), (Av, 0.5)\};$
$\{(F, 0.125), (Ac, 0.2), (M, 0.2), (\{F, A\}, 0.2), (\{F, A, M\}, 0.275)\})$, *and*
$Q_2 = (\{(B, 1)\}, \{(L, 1)\}; \{(F, 0.267), (Ac, 0.267), (M, 0.266), (\{F, A, M\}, 0.2)\})$

Let us compute the dissimilarity measures of this object respectively to the two clusters' modes as follows (see Equation 3.11 and 3.13):
$D(X_h, Q_l) = \sum_{j=1}^{3} d(m^{\Theta_j}\{X_h\}, m^{\Theta_j}\{Q_l\})$, *for* $l = \{1,2\}$.

As a result, we assign it to the first cluster C_1 *since* $D(X_h, Q_1) < D(X_h, Q_2)$.

Suppose that we would classify a new object X_i characterized by certain and exact values for its qualification and income attributes which are respectively the values B and Low. However, there is some uncertainty in the value of the department attribute as represented in Table 3.4.

Table 3.4: An uncertain instance

$m_{Qualification}$		m_{Income}		$m_{Department}$	
∅	0	∅	0	∅	0
A	0	High	0	Finance	0.4
B	1	Low	1	Accounts	0
C	0	Average	0	Marketing	0
A,B	0	High,Lo	0	Finance,Accounts	0
A,C	0	High,Average	0	Finance,Marketing	0.3
B,C	0	Low,Average	0	Accounts,Marketing	0
$\Theta_{Qualification}$	0	Θ_{Income}	0	$\Theta_{Department}$	0.3

As a result, we obtain that the new instance to classify has as distances respectively to the two clusters (see Equation 3.11 and 3.13):
$D(X_i, Q_1) = \sum_{j=1}^{3} d(m^{\Theta_j}\{X_i\}, m^{\Theta_j}\{Q_1\}) = 1.355$, *and*
$D(X_i, Q_2) = \sum_{j=1}^{3} d(m^{\Theta_j}\{X_i\}, m^{\Theta_j}\{Q_2\}) = 0.300$.

So, this object is assigned to the second cluster C_2 *since* $D(X_i, Q_2) < D(X_i, Q_1)$.

3.8 Conclusion

We have presented our method which consists in developing a new K-modes method in an uncertain environment for partitioning categorical data bases. Our method is based on the belief function theory in order to represent the uncertainty relative to the parameters of the classification problem precisely about attributes' values.

In this chapter, the belief K-modes approach was developed. We have exposed what kind of uncertainty is handled by this approach as well as its different parameters namely the computation of the clusters' modes and the measure of dissimilarity allowing to assign objects to appropriate clusters. Then, we have exposed the BKM algorithm allowing the construction of such clusters.

Once clusters are built, the result will be used to classify new instances with unknown attributes' values. This task is known as the classification procedure. As mentioned in the beginning of this chapter, uncertainty may pervade both the building and the classification steps.

In the next chapter, we will present the implementation for checking the performance of our belief clustering method called BKM.

Then, we will show different results obtained from simulations and that have been performed on real databases in this uncertain context.

Chapter 4

Implementation and simulation

4.1 Introduction

Implementing our belief K-modes approach (BKM) is important since it allows us to have an idea concerning the feasibility of our proposed method.

In this chapter, we present the implementation of our new method, the belief K-modes method (BKM). To this end, we have developed all programs in MATLAB V6.5 .

Once the different programs are implemented, for checking the feasibility of our approach regarding belief K-modes method and judging its performances, we have performed several tests and simulations on several real databases obtained from the U.C.I. repository [42]. Different results will be presented and analyzed in order to evaluate our proposed method.

In fact, this chapter is composed of two parts:

- The first part deals with the implementation of the belief K-modes method where the major variables and programs are detailed. The principal BKM algorithms are also exposed.
- The second one is interested to the simulation phase. Then, results over data from a real world problem are presented with an analysis of them. Note that the objective of this simulation is the feasibility of our belief K-modes algorithm.

4.2 Implementation

4.2.1 Framework

In order to ensure the implementation of our approach, we have developed programs in Matlab V6.5.
Obviously, we have implemented the building algorithm as well as the classification one which are detailed in the previous chapter.

These programs have as inputs:

1. An ordinary data set with certain objects, i.e., objects whose attributes are represented by single certain values as in standard K-modes method.
2. The different attributes that will be uncertain.
3. The different values of the parameter p (uncertainty degree respectively to each attribute) allowing the generation of uncertain attributes whose masses will be equal to $1 - f$, where f is a randomly number that must be less or equal to p.
4. The percent of data that will be generated within uncertain context.
5. K which represents the number of clusters to form.
6. An uncertain instance to classify.

The outputs of our programs are:

1. The obtained K modes, one for each cluster.
2. The obtained clusters' memberships of all objects belonging to the data set to cluster.
3. The K clusters' modes and their corresponding attributes' values.
4. The accuracy of the belief K-modes method given by the Percent of Correct Classification (PCC) (a most used criterion for measuring the accuracy of classifiers).

4.2.2 Principal variables

Our programs use the following variables to implement our proposed method:

- *data*: a matrix whose first row contains the different attributes' labels including the class label obtained from the ordinary data set. It contains the ordinary data set with certain objects.
- *all_val*: a matrix whose first column contains the different attributes' labels including the class label in the last one, and in the remaining columns the corresponding values.
- *nbreobj*: the number of objects which compose the real data set.
- *nbreatt*: the number of attributes which characterize the data set instances.
- *att*: a cell array which contains the different Θ respectively to all data attributes.
- *attrib*: a cell array which contains the different 2^{Θ} respectively to all data attributes.
- *cardatt*: a vector of the different attribute cardinalities.
- *cardattrib*: a vector of the different 2^{Θ} cardinalities respectively to all data attributes.
- *val_clus*: a cell contains the real class values.
- *degr*: a vector which contains the corresponding uncertainty degree p of each attribute.
- *percent*: the percent of uncertain data to generate.
- *cluster*: a vector of n elements, such as n is equal to *nbreobj*, of cluster membership of each data object.
- *center*: the obtained K clusters' centers or modes after applying the BKM procedure: a matrix of K rows which contains the different attributes belief functions (bba's).
- *mat*: a matrix of *nbreobj* rows which indicates the certain/uncertain data set represented via a belief function.
- *result*: a matrix which contains generated uncertain data set and in its last column we mention the obtained clusters from the building step.

- n_1: the number of correctly classified instances.
- n_2: the number of incorrectly classified instances.
- PCC: the percent of correctly classified instances.
- instance_to_classify: a vector which specifies the different attributes' values represented via belief functions of one new instance to classify.

4.2.3 Main programs

Many programs are developed to ensure the construction of our software, we will present them in this section regrouped as follows in the three distinct parts.

Data set creation

The idea is to assign to each attribute, a bba over the set of remaining attributes of this object, based on the set of their possible values . After precise the uncertainty degree respective to each attribute, we affect $1 - f$, where f is a randomly number that must be less or equal to p (p is the uncertainty degree) as bbm to the certain attribute's value and f to the frame of discernment corresponding to this attribute.

Once belief degrees are fixed for each attribute and the percent of uncertain data is indicated, we use the belief K-modes method to cluster the uncertain data set which is obtained from one real data set in the known number of clusters K.

In order to respect the instances' representation that we have proposed in advance, we have to deal with an existing ordinary data set in which instances are characterized by certain (single) attribute values and then create, artificially, belief functions (**bba**) on attributes' values of instances. To create such instances, we have developed these programs:

- **Generate_cert_data_set:** generates a certain data set from an ordinary data set: it creates for each attribute of a given data instance (from the ordinary data set) a certain bba, i.e., each created bba has only one element with basic belief mass equal to 1 and the others with bbm equal to 0.
- **Generate_uncert_data_set:** generates an uncertain data set from an ordinary data set. Artificially, a bba on possible values of each attribute of the data instance is created in order to add uncertainty about the attribute values.
  - **Get_p:** allows to get the value of the parameter p, the uncertainty degree respectively to each attribute.

  - **Get_percent:** allows to get the percent of uncertain data objects.

The two previous procedures (generate certain and uncertain datasets) use the following one.

- **Init_data:** allows to load the data set from which we will apply the BKM method.

Building Clusters

The implementation of the building clusters procedure relative to the belief K-modes approach represents an important task. Many programs have been developed to ensure this purpose. Allowing the building of clusters from a certain/uncertain generated data set from loaded one, these main programs are defined as follow:

- **Get_K:** allows to get the value of the parameter K, number of clusters to form.

- **Belief_clustering:** is an iterative procedure allowing to build the clusters and compute the corresponding modes relatively to the generated data set from the loaded one and to the introduced respective value of p for each attribute and the percentage of uncertain data objects. This procedure uses the following main function:

  - Beliefkm: this procedure has as output the K modes and the cluster's memberships of different generated uncertain objects which form the data set. It uses this function:

    * Beldist: it consists in computing the distance measure between each object and anyone cluster mode. It is an iterative procedure which is repeated s times (s is the number of attributes).

Classification

Once clusters and their corresponding modes are generated, we have to develop programs that will ensure the classification of such instances represented via bba's on their different attributes' values to corresponding constructed clusters. These programs are the following ones:

- **Classify_instance:** allows the classification of a certain/uncertain instance within a BKM results. This procedure uses the Beldist procedure to compute the distance of this instance respectively to all obtained modes.

- **Eval_BKM:** evaluates the BKM by computing its PCC: the percent of correctly classified data instances presented to the BKM.

4.2.4 Belief K-modes algorithms

In this section, we will present the major algorithms relative to the belief K-modes method, namely, the building and the classification algorithms.

The building procedure

Algorithm $BKM(nbre)$
Output: PCC

```
begin
(* Load data set to cluster *)
[data, all_val, filename, path] ← Init_data;
[a,b] ← size(all_val);
(* b is the number of attributes+1 (cluster column)*)
(* Define the uncertainty degrees of all attributes *)
for i ← 1 to b-1 do
  degr(i) ← Get_p;
end for
(* Define the percent of uncertain objects *)
percent ← Get_percent;
SPCC ← 0
MPCC ← 0
for iter ← 1 to nbre do
  (* Generate uncertain data set from loaded one *)
  [mat] ← Generate_uncert_data_set(data, all_val, filename, path, degr, percent);
  (* The actual clusters' values*)
  l ← 1;
  val_cluster(l) ← all_val(2,b);
  for i ← 3 to a do
    if all_val(i,b) not exist (val_cluster) then
      l← l + 1;
      val_cluster(l)← all_val(i,b);
    end if
  end for
```

```
  (* Give the number of clusters to form*)
  K ← Get_K;
  (* The obtained uncetain data set with the corresponding clusters' membership and the different K modes*)
  [result, center] ← Belief_clustering(mat, K, data, all_val, filename, path, val_cluster);
  [n1,n2] ← Evaluate_BKM(data, result);
  (* Computing PCC *)
  PCC ← n1/nbreobj * 100;
  SPCC ← SPCC+PCC;
end for
MPCC ← SPCC/nbre
end.
```

Algorithm *Generate_uncert_data_set*
Input: data, all_val, filename path, degr, percent
Output: mat

```
begin
(* Generate the different power sets of all attributes and compute the respective cardinalities*)
[attrib,cardattrib] ← generate_all_val(all_val);
[nl, nc] ← size(data);
nbreobj ← nl-1;
nbattrib ← nc-1;
uncert_data ← (percent*nbreobj)/100;
(* Generate the uncertain part of data*)
for i ← 1 to uncert_data do
  for j ← 1 to nbreattrib do
    indcour ← indexval(data(i,j),attrib(j));
    pred ← 0;
    for l ← 1 to cardattrib(j)-1 do
      if l = indcour then
        x← rand;
        While x > degr(j) do
          x ← rand;
        end while
```

```
        mat(i,l+pred)← 1-x;
      else
        mat(i,l+pred) ← 0;
      end if
      mat(i,cardattrib(j)+pred) ← x;
    end for
    k ← j+1;
    pred ← cardattrib(k-1);
  end for
end for
(* Generate the certain part of data*)
for i ← uncert_data+1 to nbreobj do
  for j ← 1 to nbreattrib do
    indcour ← indexval(data(i,j),attrib(j));
    pred ← 0;
    for l ← 1 to cardattrib(j) do
      if l = indcour then
        mat(i,l+pred)← 1;
      else
        mat(i,l+pred) ← 0;
      end if
    end for
    k ← j+1;
    pred ← cardattrib(k-1);
  end for
end for
end.
```

Algorithm *Belief_clustering*
Input: mat, data, K, all_val, filename, path, val_cluster
Output: result, center

```
begin
[a,b] ← size(data);
nbreobj ← a-1;
```

```
[result] ← data(2:a,1:b-1);
for h ← 1 to nbreobj do
  col(h,1) ← ' ';
end for
[result] ← [result,col];
[attrib,cardattrib] ← generate_all_val(all_val);
[cluster,center] ← beliefkm(mat,attrib,cardattrib,K);
for h ← 1 to nbreobj do
  result(i,b) ← cluster(i);
end for
end.
```

Algorithm *Beliefkm*
Input: mat, attrib, cardattrib, K
Output: cluster, center

```
begin
[nbreobj,nbrecol] ← size(mat);
  if K = 1 then
    center← mean(mat,1);
    cluster← ones(1,nbreobj);
  else
    ii ← randperm(nbreobj);
    center ← mat(ii(1:K),:);
    DD ← zeros(nbreobj,K);
    cl ← ones(1,nbreobj);
    Fin ← 0;
    While not Fin do
      for i ← 1 to nbreobj do
        for l ← 1 to K do
          for j ← 1 to nbreattrib do
            DD(i,l) ← DD(i,l)+beldist(mat(i,j),center(l,j));
          end for
        end for
      end for
```

```
      [Dmin,cluster] ← min(DD);
      for l ← 1 to K do
        iii ← Find(cluster=l);
        center(l,:) ← mean(mat(iii,:),1);
      end for
      Fin ← (cl==cluster)
    end while
  end if
end.
```

The classification procedure

Algorithm $Classify\_instance$
Input: cluster, center, instance_to_classify
Output: membership

```
begin
mode ← center;
[nbrecluster, nbrecol] ← size(mode);
(* Computing the distance between this instance to classify and all modes *)
for i ← 1 to nbrecluster do
(*Computing the distance corresponding to each attribute*)
  for j ← 1 to nbreattribut do
    D(i) ← D(i)+beldist(instance_to_classify(j),mode(i,j))
  end for
end for
(*The obtained cluster to which this instance belongs*)
[Dmin, membership] ← min(D)
end.
```

Algorithm $Eval\_BKM$
Input: data, result
Output: n1, n2

```
begin
(* Initializing the number of correctly classified instances to 0 *)
n1 ← 0;
```

```
(* Initializing the number of incorrectly classified instances to 0 *)
n2 ← 0;
[nbreobj, nbrcol] ← size(result);
for i ← 1 to nbreobj do
(* comparing the actual cluster value and the obtained one by applying our method *)
   if data(i,nbrecol) = result(i,nbrecol) then
     n1← n1 + 1;
   else
     n2 ← n2 + 1;
   end if
end for
end.
```

4.3 Simulations and results

4.3.1 Experimental setup

The implementation of our BKM algorithm will be useful in the simulation phase. We have performed several tests and simulations on real databases obtained from the U.C.I. repository. Different results carried out from these simulations will be presented and analyzed in order to evaluate our proposed method.

4.3.2 Artificial uncertainty creation in the training set

Training sets are generally composed with certain objects described by known attribute values and classes. Instances with partially known values are usually eliminated from the databases and they are not considered in learning process, probably because it is not easy to know what to do with these objects since the users are in pain to precise exactly their characteristics.

The belief K-modes method is essentially developed to handle uncertain objects where their uncertainty is represented by a bba given on the set of possible attribute values. So, the question is how construct these bba's to obtain uncertain data sets, since there is not real databases within the belief framework.

Attributes' values of training instances are perfectly known in the standard K-modes method and also in the fuzzy extension as well as in EVCLUS method. However, in this work, uncertainty is introduced in the values of attributes and it is presented through bba's concept. Thus, in the case of total certainty of attributes' values we deal with certain bba's and this case is equivalent to the standard version of the K-modes method.

These bba's are created artificially. They take into account these following basic parameters:

- The real attributes' values of the training instances.
- Degree of uncertainty. Since, p (one for each attribute): it will vary in [0, 1] interval. The fixed value of p has a direct effect on the quality of results. In fact, for a large value of p, the number of the correctly classified instances will decrease. The two extreme cases occur when:
  - $p = 0$, no uncertainty, we recover a standard K-modes method and this is the certain case.
  - $p = 1$, we obtain the total ignorance.

We will consider these four different intervals of p for our simulations:

- Level 1: we take $0 < p \leq 0.25$
- Level 2: we take $0.25 < p \leq 0.5$
- Level 3: we take $0.5 < p \leq 0.75$
- Level 4: we take $0.75 < p \leq 1$

Thus, the choice of the value of p for a given attribute is crucial.

The resulting bba's are the bba's which describe our belief about the value of the actual attributes' values which the object has.

Each bba has 2 focal elements:

1. The first is the actual attribute's value of the object with bbm , m(A)= 1-f (f is a probability generated randomly which must be less or equal to p).
2. The second is Θ such as m(Θ)= f.

4.3.3 Evaluation criteria

Huang [31] proposed a measure of clustering results called the clustering accuracy r computed as follows: $r = \frac{\sum_{i=1}^{k} a_i}{n}$, where n is the number of instances in the dataset, k is the number of clusters, a_i is the number of instances occurring in both cluster C_i and its corresponding labeled class. This criterion is equivalent to the most well one, the PCC expressing the percent of the correctly classified instances. These two measures are equivalent. To evaluate our belief K-modes method, we will consider the last mentioned criterion, which is the most relevant one.

So, the accuracy of our method is determined by measuring the number of instances correctly classified among the total number of data instances presented to the classifier which is a most used performance indicator namely **Percent of Correct Classification (*PCC*)**. Let us give the definition of this criterion:
The PCC represents the percent of correct classification of the instances which are classified according to the BKM procedure. It is given by:

$$PCC = \frac{number\ of\ well\ classified\ instances}{total\ number\ of\ classified\ instances} * 100 \tag{4.1}$$

The PCC is computed as follows: for each instance, we make comparison between its real class (its class in the real data set) and the class given by the belief K-modes method. Hence, the number of well classified instances represents the number of instances for which the class obtained by the BKM is the same as their real class.

When the obtained PCC is equal to 100%, it means that this classifier is 'an excellent classifier', whereas a *'null'* classifier has a PCC equal to 0%.

4.3.4 Validation procedure

In our simulations, in order to obtain an unbiased estimation of the PCC, we have used a certain number of tests and after that we will calculate the final PCC as the average of all obtained ones.

This method consists in randomly permutation of the integers from 1 to n of a given data set such as n is the number of objects that compose this data set (from which the clusters will be built) and the K first objects will be extracted to compute the initial clusters' modes. The procedure is repeated x times, each time using another K first instances as the initial clusters' modes.

Obviously, in each fold, we compute the corresponding PCC and the final PCC is given by the mean of the computed PCCs. It is $M.PCC$.

4.3.5 Simulations on the real databases

Description of the databases

For the evaluation of the proposed BKM method, we have developed programs which consist in both building and the classification procedures corresponding to our approach. Then, we have applied these programs to real databases obtained from the U.C.I repository of Machine Learning databases [42]. We have modified these databases by introducing uncertainty in the attributes' values of their instances as explained before. A brief description of these databases is presented in Table 4.1.

Table 4.1: Description of databases

Database	#intances	#attributes	#clusters
Congressional voting records database	497	16	2
Balance scale database	625	4	3
Wisconsin breast cancer database	699	8	2

Experimental results

Two parameters will be considered in our simulations, namely the percent of uncertain data set and the uncertainty degree p for each attribute.

In order to evaluate the BKM approach, for each data set, we run the algorithm several times. The accuracy of our results is measured according to the mean PCC criterion (M.PCC) of the obtained ones. We were interested by the impact of varying these parameters on the $M.PCC$ of each BKM result.

Our simulations will be performed for two cases, namely, the certain case and the uncertain case.

1. **The certain case :** The first case tests the efficiency of our method when there is no uncertainty in attributes' values, it means that each attribute is known with certainty and it has a unique value, and compares the results with ones obtained by apply the standard K-modes method. It shows that this case is equivalent to the standard K-modes method.

2. **The uncertain case :** The second case tests the efficiency of the proposed method with different level of uncertainty according to the uncertain degrees p corresponding to all attributes which characterize the data set to cluster. We take also into account the percent of uncertainty in the data set.

Results of the certain case:

Table 4.2 summarizes different results relative to Wisconsin breast cancer, Balance Scale weight and Congressional voting databases for the certain case.

Table 4.2: Experimental results : certain case

Database	M.PCC(Standard K-modes)	M.PCC (BKM)
Congressional voting records database	86.52	88.35
Balance scale database	79.20	79.20
Wisconsin breast cancer database	71.39	72.48

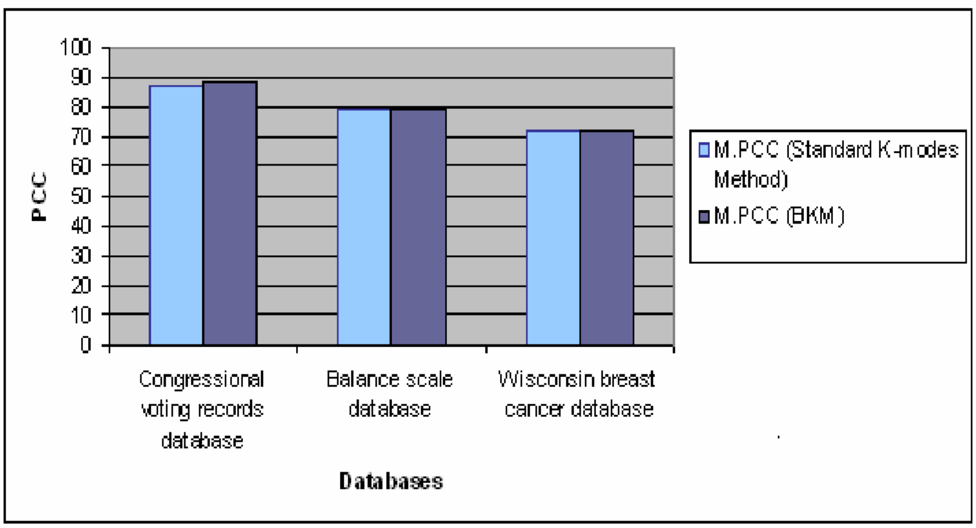

Figure 4.1: M.PCC on the certain case

In order to compare our method's results to obtained ones by applying the Huang's algorithm, we have developed programs (also in MATLAB) implementing Huang's algorithm.

When there is no uncertainty in the attributes' values of instances (which represents our case here), we can make comparison between the two results: the first one corresponds to the our method and the second is obtained from the standard K-modes method given by the Huang algorithm.

From this table, we can conclude that our belief clustering method in certain case has a good results. There are an improvement of the PCC of the results in two databases. But, for Balance Scale weight database, The obtained PCC is the same for the two methods (79.20%).

Results have shown that our approach can outperform the standard K-modes method. For example, for the congressional voting records database the mean PCC (M.PCC) of the standard K-modes is equal to 86.52% whereas, we have obtained larger PCC's equal to 88.35%.

Similarly, for the W.breast cancer database, the PCC have leapt from 71.39% to 72.48% when using our BKM approach instead of the standard one.

As explained in the chapter 2, one problem encountered in the K-modes method and its extensions is the choice of the initial modes which has an effect on the produced results. For both methods, the Huang method and the proposed one, the initial modes are chosen randomly, it explains the obtained results, which are different, although the two methods are applied in the same context namely the certain framework.

We should mention that within the standard version of the K-modes method, the clusters' modes can be non-unique which makes the algorithm unstable. However, our method solves this problem. Thus, our PCC's are, for the most of databases, better than the ones obtained by applying the standard approach.

Results of the uncertain case:

In this section, we will present different results carried out from applying our proposed BKM approach relative to the Wisconsin breast cancer and Balance Scale weight databases on uncertain case.

The following tables (Table 4.3, Table 4.4 and Table 4.5) summarize different results carried out from testing our method to the same three datasets respectively to the four intervals of uncertainty degree p defined before and the different values of the uncertainty percent of dataset instances which are defined as follows : 25%, 50%, 75% and 100%.

These uncertainty percents represent the percent of generated uncertain objects of one given dataset, where as the other objects are known with certainty. For example, if we fix this percent at 25%, it means, for one database which contains 100 objects, that 25 instances will be generated with uncertainty (in their attributes) and the 75 others are exactly know and they have unique attributes' values.

These values of the uncertainty parameters (the uncertainty degrees and the uncertainty percent) allow us to generate the uncertain databases.

Table 4.3: Experimental results (Congressional voting records, uncertain case)

Percent / Degree	25%	50%	75%	100%
$0 < p \leq 0.25$	91.25%	89.13%	88.15%	86.23%
$0.25 < p \leq 0.5$	90.11%	87.52%	87.48%	81.13%
$0.5 < p \leq 0.75$	88.82%	86.03%	85.57%	76.29%
$0.75 < p \leq 1$	85.17%	83.57%	84.12%	71.63%
Mean	88.84%	86.56%	86.33%	78.82%

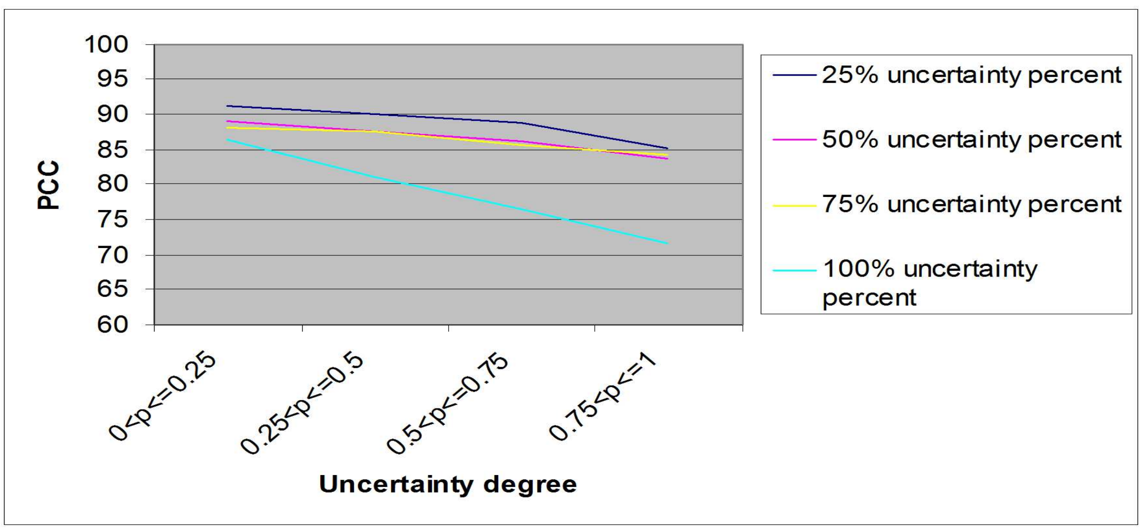

Figure 4.2: M.PCC of the uncertain case (Voting)

As with the certain case, the results show that the proposed approach deals with uncertain instances as good as with certain instances. For instance, if we analyze the results shown in Figure 4.1 (relative to the voting database), we remark that the PCC (uncertain case) remains high (larger than 85% for most cases) in average as in the certain case. These results, certify that our proposed approach is also well adapted to classify instances with uncertain attributes' values.

Table 4.4: Experimental results (Balance scale, uncertain case)

Degree \ Percent	25%	50%	75%	100%
$0 < p \leq 0.25$	82.26%	80.12%	79.29%	77.22%
$0.25 < p \leq 0.5$	79.89%	79.95%	78.56%	77.15%
$0.5 < p \leq 0.75$	78.23%	76.30%	75.59%	74.35%
$0.75 < p \leq 1$	75.83%	76.01%	74.44%	71.23%
Mean	79.05%	78.09%	76.97%	74.98%

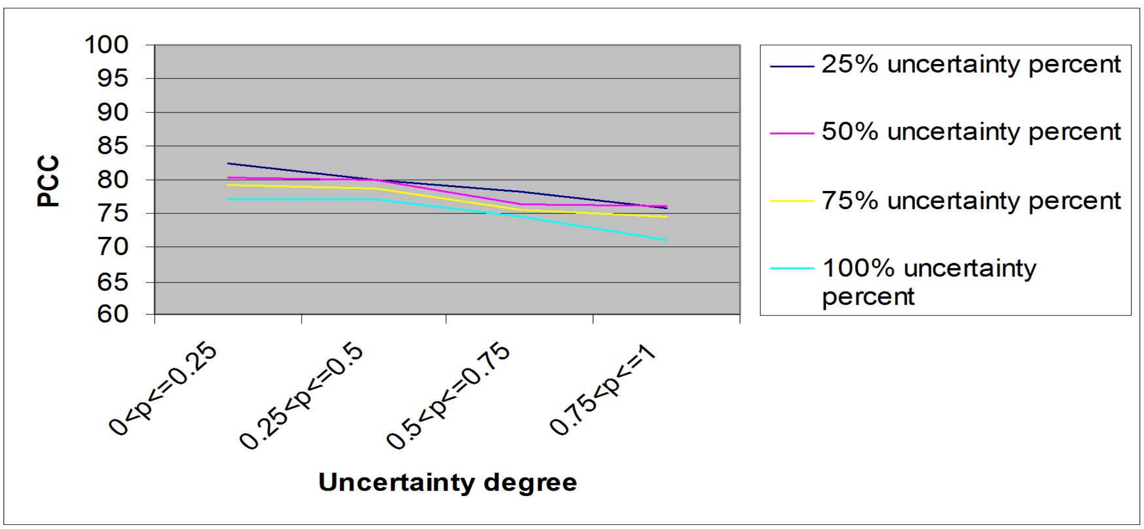

Figure 4.3: M.PCC of the uncertain case (Balance)

Note that the PCC's decrease in average where uncertainty in parameters (the uncertainty percent and the uncertainty degrees) increases as shown in Figures 4.2, 4.3, and 4.4.

For example for the Congressional voting records database, the PCC value becomes 71.63% when the uncertainty percent is at 100% and the uncertainty degrees are in [0.75, 1[, which is considered as a high uncertainty, comparing to 91.25% with

Table 4.5: Experimental results (Wisconsin breast cancer, uncertain case)

Degree \ Percent	25%	50%	75%	100%
$0 < p \leq 0.25$	75.24%	75.13%	74.08%	73.91%
$0.25 < p \leq 0.5$	73.55%	73.21%	72.42%	71.51%
$0.5 < p \leq 0.75$	71.03%	70.81%	68.69%	68.88%
$0.75 < p \leq 1$	70.99%	69.18%	68.87%	68.02%
Mean	72.70%	72.08%	71.01%	70.58%

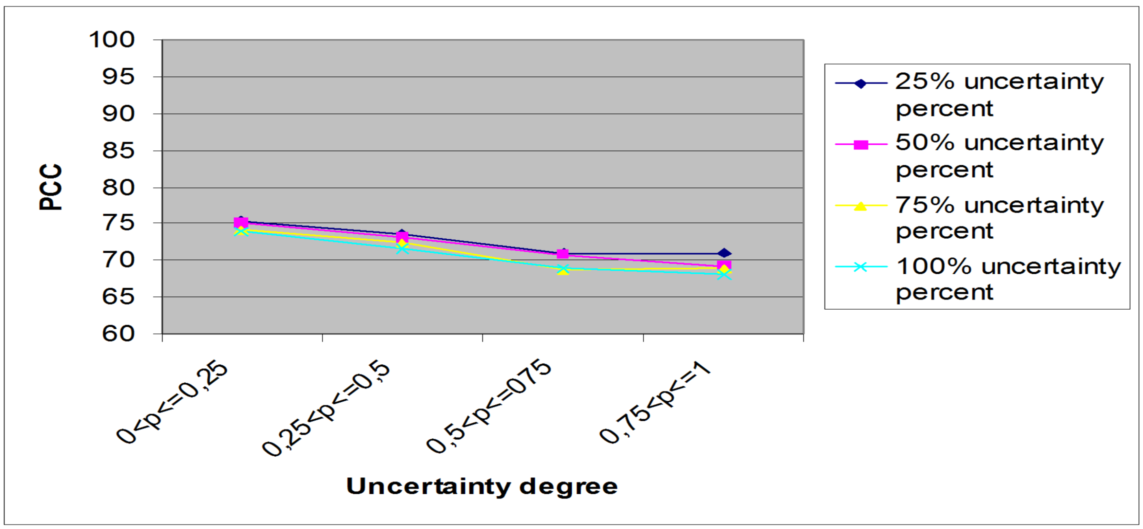

Figure 4.4: M.PCC of the uncertain case (Cancer)

a low uncertainty (25% of uncertainty percent and the uncertainty degrees of the attributes are defined as follows: $0 < p <= 0.25$) as shown in Table 4.3.

Furthermore, we notice that for each database, there is a specific values of the uncertainty parameters for which we obtain the highest PCC.

As a conclusion, we have to note that these values are purely experimental and that depend on the used database and even on the used uncertainty degree and the considered uncertainty percent within a given database.

It is found that the clustering results produced by the proposed method are very high in accuracy. The PCC's show that our method presented interesting results.

So, These results confirm that our approach is well appropriate within the uncertain context.

4.4 Conclusion

In this chapter, the major variables and the main programs that we have used in order to implement our belief clustering method, are detailed. The principal BKM algorithms are also presented.

Experimental results, provided in this chapter, show that our new approach (described in chapter 3) gives better results than the standard version of this method within a certain context.
These results are interesting since they show that with data bases having uncertain attributes' values, we can cluster it.

The obtained results from simulations that have been performed on real databases in both certain and uncertain cases are mentioned and have shown encouraging results.

Note that the major results of this work are developed in [6].

Conclusion

Datasets containing uncertain information are common in real life data mining applications. However, standard versions of clustering methods are badly adapted to ensure their role in such environment. Thus, the need of the development of appropriate approaches to this kind of environment is vital.

This master dissertation has contributed to the development of one clustering method under uncertainty in belief function theory framework. Our method called Belief K-modes Method (BKM), using the K-modes paradigm, was developed to cope with the problem of clustering data sets with uncertain attributes.

After reviewing the K-modes method, we noticed that the standard version is not adapted to an uncertain framework. Consequently, our idea was to propose a new clustering method appropriate to this kind of environment. For this purpose, we were interested in belief function theory which presents an appropriate framework to deal with uncertain classification problems. So, we have developed what we call Belief K-modes Method (BKM), a combination between the K-modes method and the belief function theory.

In order to cluster dataset objects characterized by uncertain attributes' values, where this uncertainty is represented via basic belief assignments (bba's), we have defined two major parameters namely the computation of the clusters' modes (the belief mean) and the dissimilarity measure.

The proposed method (BKM) allows us to construct clusters within objects having uncertain attributes values. Another advantage of BKM method is that once the clusters are fixed, the classification of new instances that may be uncertain is possible.

We have performed simulations on real databases in order to evaluate the performance of our belief clustering method.
Using the PCC as an evaluation criterion, the results of the experiments on certain

and uncertain data sets show the efficiency of our approach.

Our work can be extended on different manners. In fact, our method can be improved by applying one technique to select the initial modes instead of the current one which consists in choosing randomly K objects from the dataset to cluster as the initial modes.

An interesting future work is to make our method able to cluster datasets characterized by continuous attributes. Thus, the proposed method will be more flexible to handle mixed numerical and categorical databases.

In addition to the uncertainty on attribute values, another line of research will be to assume that each object in the training set may belong to more than one cluster, this uncertainty in the cluster membership can be represented via belief functions.

Appendix A

Data bases used for simulations

A.1 Introduction

In our experiments, we have used these three databases : the Wisconsin breast cancer database, the Congressional voting database and the Balance scale database [42] to evaluate our method.

A.2 The Wisconsin breast cancer database

1. Title: Wisconsin Breast Cancer Database (January 8, 1991).

2. Source Information:

   (a) Dr. William H. Wolberg (physician) University of Wisconsin Hospitals Madison, Wisconsin USA.

   (b) Donor: Olvi Mangasarian (mangasarian@cs.wisc.edu) Received by David W. Aha (aha@cs.jhu.edu).

   (c) Date: 15 July 1992.

3. Relevant Information:

   Samples arrive periodically as Dr. Wolberg reports his clinical cases. The database therefore reflects this chronological grouping of the data. This grouping information appears immediately below, having been removed from the data itself:
   Group 1: 367 instances (January 1989)
   Group 2: 70 instances (October 1989)

Group 3: 31 instances (February 1990)
Group 4: 17 instances (April 1990)
Group 5: 48 instances (August 1990)
Group 6: 49 instances (Updated January 1991)
Group 7: 31 instances (June 1991)
Group 8: 86 instances (November 1991)
So the total = 699 points (as of the donated datbase on 15 July 1992).

4. Number of Instances:
   699 (as of 15 July 1992).

5. Number of Attributes:
   8 attributes + class name = 9.

6. Class Distribution:
   Benign: 458 (65.5%).
   Malignant: 241 (34.5%).

A.3 Balance scale database

1. Title: Balance Scale Weight & Distance Database.

2. Source Information:

   (a) Source: Generated to model psychological experiments reported by Siegler, R. S. (1976). Three Aspects of Cognitive Development. Cognitive Psychology, 8, 481-520.

   (b) Donor: Tim Hume (hume@ics.uci.edu).

   (c) Date: 22 April 1994.

3. Relevant Information:

   This data set was generated to model psychological experimental results. Each example is classified as having the balance scale tip to the right, tip to the left, or be balanced. The attributes are the left weight, the left distance, the right weight, and the right distance. The correct way to find the class is the greater of (left-distance * left-weight) and (right-distance * right-weight). If they are equal, it is balanced.

4. Number of Instances:
   625 (49 balanced (B), 288 left (L), 288 right (R)).

5. Number of Attributes:
   4 attributes + class name = 5.

6. Attribute Information:

   (a) Class Name: 3 (L, B, R).
   (b) Left-Weight: 5 (1, 2, 3, 4, 5).
   (c) Left-Distance: 5 (1, 2, 3, 4, 5).
   (d) Right-Weight: 5 (1, 2, 3, 4, 5).
   (e) Right-Distance: 5 (1, 2, 3, 4, 5).

7. Class Distribution:
   1. 46.08 percent are L.
   2. 07.84 percent are B.
   3. 46.08 percent are R.

A.4 Congressional voting records database

1. Title: 1984 United States Congressional Voting Records Database.

2. Source Information:

   (a) Source: Congressional Quarterly Almanac, 98th Congress, 2nd session 1984, Volume XL: Congressional Quarterly Inc. Washington, D.C., 1985.
   (b) Jeff Schlimmer (Jeffrey.Schlimmer@a.gp.cs.cmu.edu).
   (c) Date: 27 April 1987.

3. Relevant Information:

   This data set includes votes for each of the U.S. House of Representatives Congressmen on the 16 key votes identified by the CQA. The CQA lists nine different types of votes: voted for, paired for, and announced for (these three simplified to yea), voted against, paired against, and announced against (these three simplified to nay), voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known (these three simplified to an unknown disposition).

4. Number of Instances:
   435 (267 democrats, 168 republicans).

5. Number of Attributes:
   16 + class name = 17.

6. Attribute Information:

   (a) Class Name: 2 (democrat, republican).
   (b) handicapped-infants: 2 (y,n).
   (c) water-project-cost-sharing: 2 (y,n).
   (d) adoption-of-the-budget-resolution: 2 (y,n).
   (e) physician-fee-freeze: 2 (y,n).
   (f) el-salvador-aid: 2 (y,n).
   (g) religious-groups-in-schools: 2 (y,n).
   (h) anti-satellite-test-ban: 2 (y,n).
   (i) aid-to-nicaraguan-contras: 2 (y,n).
   (j) mx-missile: 2 (y,n).
   (k) immigration: 2 (y,n).
   (l) synfuels-corporation-cutback: 2 (y,n).
   (m) education-spending: 2 (y,n).
   (n) superfund-right-to-sue: 2 (y,n).
   (o) crime: 2 (y,n).
   (p) duty-free-exports: 2 (y,n).
   (q) export-administration-act-south-africa: 2 (y,n).

7. Class Distribution:
   45.2 percent are democrat.
   54.8 percent are republican.

A.5 Conclusion

In this appendix, we have shown the description of the data sets which we have used in the simulation phase.

Note that in our experiments, we have created the uncertain versions of these data bases via the belief function theory as explained in the last chapter.

Bibliography

[1] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. SIGMOD'98, 1998.

[2] M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure, SIGMOD99, 1999.

[3] J.A.Barnett: Calculating Dempster-Shafer plausibility. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(6), 599-602, 1991.

[4] M.Barni, V.Cappellini and A.Mecocci: Comments on A Possibilistic Approach to Clustering, IEEETrans. Fuzzy Systems, 4(3), 393-396, 1996.

[5] M.Bauer: Approximations for efficient computation in the theory of evidence, Arttif.Intell. 61(2), 315-329, 1993.

[6] S. Ben Hariz, Z. Elouedi, and K. Mellouli: Clustering Approach using Belief Function Theory. Accepted in the Twelfth International Conference on Artificial Intelligence: Methodology, Systems, Applications, 2006.

[7] J.C.Bezdek: Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.

[8] J.C.Bezdek, N.R.Pal, and R.Hemasinha: Uncertainty measures for evidential reasoning I:A review. International Journal of Approximation Reasoning, 7: 165-183, 1992.

[9] J.C.Bezdek, N.R.Pal, and R.Hemasinha: Uncertainty measures for evidential reasoning II:New measure of total uncertainty. International Journal of Approximation Reasoning, 8: 1-16, 1993.

[10] J.C.Bezdek , R.Nikhil and P.Kuhu: A mixed c-means clustering model. FUZZ'IEEE , Barcelone, Spain, July 1-5, 1997.

[11] E.Bosse, A-L.Jousselme, D.Grenier: A new distance between two bodies of evidence. In Information Fusion 2, 91-101, 2001.

[12] R.N.Dave: Clustering relational data containing noise and outliers. In FUZZ'IEEE 98, 1411-1416, 1998.

[13] T.Denoeux: A k-nearest neighbor classification rule based on Dempster-Shafer theory. IEEE Transactions on Systems, Man and Cybernetics, 25(5), 804-813, 1995.

[14] T.Denoeux: Reasoning with imprecise belief structures. International Journal of Approximate Reasoning, 20: 79-111, 1999.

[15] T.Denoeux: A neural network classifier based on Dempster-Shafer theory. IEEE Transactions on Systems, Man and Cybernetics - Part A, 30(2), 131-150, 2000.

[16] T.Denoeux and M.Masson: Clustering of proximity data using belief functions. Proceedings of IPMU'2002, Vol I: 609-616, Annecy, France, July 2002.

[17] T.Denoeux and M.Masson: EVCLUS: Evidential Clustering of Proximity Data. IEEE Transactions on Systems, Man and Cybernetics B, Vol 34, Issue 1, 95-109, 2003.

[18] T.Denoeux and M.Masson: Clustering Interval-valued Data using Belief Functions. Pattern Recognition Letters, Vol 25, Issue 2, 163-171, 2004.

[19] D.Dubois and H.Prade: Representation and combination of uncertainty with belief functions and possibility measures. Computational Intelligence, 4: 244-264, 1988.

[20] J.C.Dunn: A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, Journal of Cybernetics 3: 32-57, 1973.

[21] Z. Elouedi, K. Mellouli, Ph. Smets: Belief Decision trees: Theoretical foundations. International Journal of Approximat Reasoning, Vol 28, Issues 2-3, 91-124, 2001.

[22] Z. Elouedi, K. Mellouli, Ph. Smets: Assessing sensor reliability for multisensor data fusion within the transferable belief model. IEEE Trans Syst Man Cybernetics, Part B, 34(1),782-787, 2004.

[23] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu: A density-based algorithm for discovering clusters in large spatial databases. KDD'96, 1996.

[24] D.Fixen, R.P.S.Mahler: The modified Dempster-Shafer approach to classification, IEEE Trans.Syst.Man Cybern. A 27(1), 96-104, 1997.

[25] J.Gordon and E.H.Shortliffe: The Dempster-Shafer theory of evidence In B. G. Buchanan and E. H. Shortliffe (Eds.), Rule-based expert systems: the MYCIN experiments of the Stanford Heuristic Programming Project, 272-292. Addison-Wesley, Reading, MA, 1984.

[26] J.W.Guan, and D.A.Bell: A generalization of the Dempster-Shafer theory. In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, 10-25. Chamery, France: IJCAI'93, Morgan Kaufman, 1993.

[27] S. Guha, R. Rastogi, and K. Shim. Cure: An efficient clustering algorithm for large databases. SIGMOD'98, 1998.

[28] R.J.Hathaway, J.W.Davenport, and J.Cbezdek: Relational duals of the c-means clustering algorithms. Pattern recognition, 22(2): 205-211, 1989.

[29] Z.Huang: Clustering large data sets with mixed numeric and categorical values. Proceedings of the First Pacific Asia Knowledge Discovery and Data Mining Confrence, Singapore: World Scientific, 21-34, 1997.

[30] Z.Huang: A fast clustering algorithm to cluster very large categorical data sets in data mining. Proceedings of the SIGMOD Workshop on Research Issues on Data Minng and knowledge Discovery, Dept of Computer science, the University of British Columbia, Canada, 1-8, 1997.

[31] Z.Huang: Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Mining Knowl.Discov., Vol.2, No.2,283-304, 1998.

[32] Z.Huang and M.K.Ng: A fuzzy K-modes algorithm for clustering categorical data. IEEE Transaction on Fuzzy Systems,7(4), 446-452, 1999.

[33] Z.He, S.Deng and X.Xu: Improving K-modes algorithm considering frequencies of attribute values in mode. ,2005

[34] A.K.Jain and R.C.Dubes: Algorithms for clustering data. Prentice-Hall, Englewood cliffs, NJ, 1988.

[35] L.Kaufman and P.J.Rousseeuw: Finding groups in data. Wiley New-York, 1990.

[36] J. Kohlas, P. A. Monney: A mathematical theory of hints: An approach to Dempster-Shafer theory of evidence. Lecture Notes in Economics and Mathematical Systems 425, Springer-Verlag, 1995.

[37] R.Krishnapuram and J.M.Keller: A possibilistic approach to clustering". IEEE Transactions on Fuzzy Systems, 1: 98-110, 1993.

[38] R.Krishnapuram and J.M.Keller: The Possibilistic C-Means algorithm: Insights and recommendations. IEEE Transactions on Fuzzy Systems, 4: 385-393, 1996.

[39] J.MacQueen: Some methods for classification and analysis of multivariate observations. In: Proc.of the Fifth Berkeley Symposium on Math, Stat. and Prob Vol 1, 281-296, 1967.

[40] O.Mar San, VN.Huynh, and Y.Nakamori: An alternative extension of the k-means algorithm for clustering categorical data. AMCS, Vol.14, No.2, 241-247, 2004.

[41] K.Mellouli: On the propagation of beliefs in networks using the Dempster-Shafer theory of evidence. PhD thesis, School of business, University of Kansas, 1987.

[42] P.M. Murphy and D.W. Aha. Uci repository databases. http://www.ics.uci.edu/mlear., 1996.

[43] S.Petit-Renaud: Application de la thorie des croyances et des systmes flous l'estimation fonctionnelle en présence d'informations incertaines ou imprécises, Ph.D. thesis, Universit de Technologie de Compiègne, December 1999.

[44] H.Ralambondrainy: A conceptual version of the K-means algorithm. Pattern Recognition Letters, Vol.15, No.11, 1147-157, 1995.

[45] G.Shafer: A mathematical theory of evidence. Princeton Univ. Press. Princeton, NJ.30, 1976.

[46] G.Shafer: Belief functions and parametric models. J. Roy. Statist.Soc. B44 322-352, 1982.

[47] G.Shafer: The combination of evidence. International Journal of Intelligent Systems, 1, 155-180, 1986.

[48] G. Sheikholeslami, S. Chatterjee, and A. Zhang. WaveCluster: A multi-resolution clustering approach for very large spatial databases. VLDB98, 1998.

[49] Ph.Smets: The combination of evidence in the transferable belief model. IEEE-Pattern analysis and Machine Intelligence, 12: 447-458, 1990.

[50] Ph.Smets: The Transferable Belief Model and Other Interpretations of Dempster-Shafer's Model. in Bonissone P.P., Henrion M., Kanal L.N. and Lemmer J.F. eds. Uncertainty in Artificial Intelligence 6.North Holland, Amsteram, 375-384, 1991.

[51] Ph.Smets: The concept of distinct evidence. IPMU 92 Proceedings. 789-794, 1992.

[52] Ph.Smets: Belief functions: The disjunctive rule of combination and the generalized bayesian theorem. International Journal of Approxi-mate Reasoning, 9, 1-35, 1993.

[53] Ph.Smets: Quantifying beliefs by belief functions: An aixomatic justification. In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, 598-603. Chambry, France: IJCAI'93, Morgan Kaufmann, 1993.

[54] Ph.Smets and R.Kennes: The transferable belief model. Artificial Intelligence, 66: 191-234, 1994.

[55] Ph.Smets and R.Kruse: The transferable belief model for belief representation. In A. Motro and P. Smets (Eds.), Uncertainty in Information Systems: From needs to solutions, 343-368. Boston, MA: Kluwer, 1997.

[56] Ph.Smets: The application of the transferable belief model to diagnostic problems. International Journal of Intelligent Systems, 13, 127-157, 1998.

[57] Ph.Smets: The transferable belief model for quantified belief representation. In D. M. Gabbay and P. Smets (Eds.), Handbook of defeasible reasoning and uncertainty management systems, Vol 1, 267-301, 1998.

[58] Ph.Smets: The transferable belief model for expert judgments and reliability problems. Reliability Engineering and System Safety, 38, 59-66, 1999.

[59] Ph.Smets: Practical uses of belief functions. In K. B. Laskey and H. Prade (Eds.), Proceedings of the Fifteenth Uncertainty in Artificial Intelligence, 612-621. Morgan Kaufman, San Francisco, Ca, 1999.

[60] Ph.Smets: Decision making in a context where uncertainty is represented by belief functions. In R. P. Srivastava (Ed.), Belief functions in business decisions. Physica-Verlag, Forthcoming, 2002.

[61] B.Tessem: Approximation algorithms and decision making in the Dempster-Shafer theory of evidence - an empirical study, Int.J.Approx.Reason. 17(2-3) 217, 1997.

[62] W. Wang, Yang, R. Muntz. STING: A Statistical Information grid Approach to Spatial Data Mining, VLDB97, 1997.

[63] P. Walley: Statistical reasoning with imprecise probabilities. Chapman and Hall, London, 1991.

[64] L.M.Zouhal, T.Denoeux: An evidence-theory k-NN rule with parameter optimization, IEEE Trans.Syst.Man Cybern. C 28(2), 263-271, 1998.

[65] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH : an efficient data clustering method for very large databases. SIGMOD'96, 1996.