Logic-Oriented Fuzzy Neural Networks:

Optimization and Applications of Interpretable Models of Machine Learning

By:

Majed Mohammad Alateeq

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

In

Computer Engineering

Department of Electrical and Computer Engineering

University of Alberta

Abstract

With the rapid development of machine learning models along with increasingly complex data structures, it becomes difficult to ground the reliability of models' predictions despite the substantial progress in favor of high approximation properties. The lack of interpretability remains a key barrier in order to fully leverage the tremendous success of intelligent systems since it delivers critical analysis abilities to the end user to achieve efficiency in decision-making processes. The purpose of Interpretability and transparency is to reveal interconnections of intelligent models leading to justifying decision-making process, eliminating vagueness and capturing a factor of uncertainty in data space. Therefore, any advancement made to the interpretability feature will positively impact overall models' performance. In the presented considerations, this work relies on logic-oriented fuzzy neural networks to represent knowledge in a transparent way with the aid of information granules. In the synergistic collaboration with fuzzy logic, neural networks deliver a vast array of learning abilities that can be even augmented with various fuzzy analytical methods to discover hidden data patterns for better interpretability. The high modularity of the constructed networks (leading to multifunctionality and robustness) is inherited from the logic nature of AND/OR neurons. The logic-oriented neurons play a pivotal role in the developed models and realize a logic approximation of experimental data and reflect general decomposition of Boolean function in two-valued logic.

Information granularity is a key component in building abstract concepts to humans for knowledge acquisition and reasoning. In fact, information granules serve as a vehicle to interpret and represent knowledge domain, offering efficient way to describe complex and nonlinear systems. Fuzzy sets, as a form of information granules, adequately handle imprecise and vague

knowledge in systems and consequently are a key in building transparent and interpretable models. Thus, humans can easily comprehend real-world systems or natural phenomena.

The overall model efficiency, expressed in terms of accuracy and interpretability when dealing with the design and validation of AND/OR networks, constitutes a focal point of this research, along with effective quantification of the extracted knowledge especially in the case of high-dimensional input–output space. The primary objective of this dissertation is to analyze and design a cohesive interpretable framework capable of maintaining high approximation capabilities. In this study, we used logic-oriented fuzzy AND\OR networks as a backbone of overall interpretable framework. Starting off with structural analysis of the network, the structure exhibits low efficiency caused by gradient-based learning algorithms. Therefore, other gradient-based learning alternatives are superior in improving convergence due to their adaptive learning mechanisms. We demonstrate that the rate of convergence can be improved significantly by integrating randomized learning techniques through generating random weight values of connectives. Furthermore, we proposed an innovative interpretable method to describe and quantify data using concepts. The approach describes reference information granules positioned in some space (output space) inducing fuzzy sets localized in the input space. The description is realized by running a conditional fuzzy clustering followed by a calibration process completed through logic networks. The synergy between conditional clustering and logic networks presents highly cohesive linguistic dependency between objects and their attributes. As for the interpretability, a thoroughly discussion of interpretation aspects of concept analysis and conceptual clustering is presented as a means for uncertainty quantification and rigorous explainability. Further enhancement of the interpretation framework is proposed by presenting a novel method of conditional clustering. We developed a mathematical model that takes into consideration multi conditions positioned in the output space to induce information granules in input space simultaneously making these types of models more reflective of reality. The experimental studies involve synthetic data machine learning datasets from publicly-available repositories.

Preface

This thesis is an original work by Majed Alateeq under the supervision of Dr. Witold Pedrycz, and it was supported by King Khalid University, Saudi Arabia.

As detailed in the following, some chapters of this thesis have been published or accepted for publication as scholarly articles in which Prof. Witold Pedrycz was the supervisory author and has contributed to concepts formations and the manuscript composition.

Chapter 2 (A) of this thesis has been submitted for publication to *Fuzzy Sets and System* as **M. Alateeq**, and W. Pedrycz, "Logic-oriented Fuzzy Neural Networks: A Survey"

Chapter 3 (B) of this thesis has been published as **M. Alateeq** and W. Pedrycz, "Development of Two-phase Logic-oriented Fuzzy AND/OR Network," In *Neurocomputing* vol. 482 pages 129 – 138, 2022.

The results of our publication, which have been published as **M. Alateeq**, W. Pedrycz, "A comparative analysis of bio-inspired optimization algorithms for automated test pattern generation in sequential circuits," In *Applied Soft Computing*, vol. 101, March 2021, are used in a comparative analysis in Chapter 3.

Chapter 5 of this thesis has been re-submitted for publication after first revision as **M. Alateeq** and W. Pedrycz, "Concept Discovery in Data: A Design and Interpretation Framework of Conditional Clustering and Logic Networks," In *IEEE Transactions on Cybernetics*.

Dedication

To my parents,

"Mohammad" and "Aljawhara"

who never saw this adventure comes to the end…

wish you were here …

Acknowledgment

First and foremost I would like to thank my supervisor Professor Witold Pedrycz, to whom I express my deepest respect and gratitude. I profoundly appreciate all your contributions of ideas, innovations, and time as well as your patience, enthusiasm, and support. I deeply appreciate all your contributions to my formation as a PhD student during my time at the University. It has been an honor being your PhD student.

I would also like to express my thanks to the members of my supervisory committee Professor Marek Reformat, Professor Venkata Dinavahi and the external examiner Professor Keun-Chang Kwak for their brilliant comments and suggestions. I also thank Professor Jie Han for his comments and positive words to improve this work.

I must express my gratitude to my wife for her support and encouragement, my kids for accepting being away from them while working on my thesis.

I would like to convey my gratitude to all my brothers and sisters for their endless positive supports. No words can express my gratitude.

Table of Contents

List of Tables

List of Figures

List of Abbreviations and Acronyms

FL	Fuzzy Logic
FS	Fuzzy Systems
FNNs	Fuzzy Neural Networks
TSK	Takagi-Sugeno-Kang
FCM	Fuzzy C-Means
CFCM	Conditional Fuzzy C-Means
AI	Artificial Intelligence
CI	Computational Intelligence
ML	Machine Learning
NNs	Neural Networks
EC	Evolutionary Computing
FCA	Formal Concept Analysis
DL	Deep Learning
SOP	Sum of Minterm
POS	Product of Maxterm
FRBS	Fuzzy rules-based systems

1. Introduction

Logic, in general, is the study of reasoning, as well as the formation and analysis of arguments. It is therefore one of the fundamental principles of philosophy. Recently, it has been commonly studied in mathematics, law and applied to computer science. The parts that make up a computer chip are often called "logic gates" which has opened up a new era of computing. Hence, the scope of logic can be very broad, ranging from core topics such as the study of fallacies and paradoxes, to specialized analyses of reasoning such as probability, correct reasoning, and arguments involving causality. Logic is also commonly used today in information theory; data mining and artificial intelligence (AI) as the human knowledge could be expressed using logic with mathematical notations even with the complexity of human reasoning. Hence, logic has an important role to play in at least some central areas of AI research

Logics have emerged as an influential driver within AI community due to several distinct characteristics such as reasoning approximation and resilience to vagueness in knowledge domain. Furthermore, logic-based methodologies preserve several strengths, in particular, interpretability, verifiability, and broadly generalization to novel tasks. The verifiability characteristic in logics allows us to identify cases where a system may fail and to formulate verification as a computationally traceable optimization problem [211], [212]. These distinctive features, and some others, of logic-based methods make logic ideal to solve existing problems emerged in the field of Machine Learning (ML). Consequently, logics have been widely incorporated in a plethora of different methods to construct coherent intelligent models such as with Neural Networks (NNs) [5], [6] in what so-called logic-oriented networks to study autonomous behaviors in complex systems. These types of networks have evolved ever since to the point in which sufficient description of qualitative behavior of real-world systems can be modeled. Currently, logics are not necessarily bounded to conventional values such as (0s, 1s) or (*True*, *false*), yet they are crucial components of today's intelligent modeling. Logic interval has been expanded to include infinite logic spectrum for more precise behavior modeling of real-world systems. It has since been devised with an infinite number of "degrees of truth" or "probability of the subjective truth value" such as in Fuzzy Logic (FL) or Bayesian reasoning [11]. This conceptual extension had has augmented the benefit of infinite logic spectrum to be used in different domain such as pattern recognition, control engineering and handling incomplete information and uncertainty [2], [12], [13].

Fuzzy logics [1] emerge as a conceptual augmented framework of infinite logic. It generalizes the fundamental idea of two-valued logic to allow for more detailed analysis and accurate knowledge representation due to its linguistic interpretation and approximation properties. Furthermore, FL is efficient in representing data and extract knowledge to capture uncertainty in the data space. In fact, it constitutes a useful vehicle to quantify data uncertainty. The deviation from classical to non-classical logic in FL has changed the way we perceive knowledge by determining the truth-value of knowledge's components. That is, the value of a logical compound proposition is given a "degree of truth" assuming values between [0, 1]. Consequently, the represented knowledge matches intuition and captures the semantics of the intended representation. The essence of logic networks that revolve around FL comes from the simplicity of extracting data descriptors realized by information granules (fuzzy sets). Each fuzzy set comes with its own well-defined semantics and as such, the granules can be associated with a certain linguistically sound meaning that is well-justified and reflective of the existing experimental data.

The combination of fuzzy logic and logic-oriented networks by taking advantages of fuzzy neurocomputing delivers an array of learning mechanisms that are either supervised or unsupervised learning schemes. Furthermore, the successful synergy between fuzzy sets and a learning mechanism strike the best balance between the fundamental requirements of fuzzy modelling in which the accuracy-interpretability trade-off have to be met to the highest extent. This requirement is commonly visible in constructing fuzzy models in which some models lean toward improving interpretation with very limited learning capabilities while other models compromise interpretability in favor of high accuracy of approximation. Interpretation capabilities of fuzzy logics have solved the black-box nature of generic learning algorithms and eliminated vagueness and uncertainty occur in decision-making processes. The well-structured semantic mapping between the input-output data carried out in logic-oriented fuzzy networks allows for easy translation and extraction of the semantic meaning of the acquired knowledge. The extracted knowledge through interpretation has a pivotal role in revealing essential hidden linkages between input-output pairs. Therefore, the resultant fuzzy models are capable to recognize complicated patterns and provide quantitative decision support. The development of the relationship between fuzzy sets (information granules) or fuzzy relations, in fuzzy modelling serves as a vehicle and a convenient way to describe complex and nonlinear systems. In the construction of fuzzy models, fuzzy clustering techniques are often used, in particular Fuzzy C-Means (FCM) [2], [15]. Clustering is used even more intensively in constructing linguistic models [20] through the formation of fuzzy sets in the output space and subsequently produces induced fuzzy sets positioned in the input space [4], [10]. Linguistic models [10], [16], [17], [18], [19], [20], [26], [27], [28] are inherently dwell upon collections of highly interpretable and user-oriented entities.

Fuzzy neurocomputing [29], [30], [31], [32], [33] utilizes the advantages of Artificial Neural Networks (ANNs), as a learning mechanism, and Fuzzy Systems (FS) for better data mining and knowledge discovery therefore they are efficiently capable of approximating reasoning and handling uncertainty [34]. The main difference between fuzzy theory and neural paradigms is that fuzzy set theory attempts to mimic the human reasoning and thought processes, whereas neural networks attempt to emulate the architecture and information representation scheme of the human brain. It is therefore meaningful to integrate these two distinct paradigms, enhancing their individual capabilities in order to build a more intelligent processing system. This new processing paradigm has become quite dominant models [35], [36], [37], [38] that try to cope with a fundamental challenge of system modeling and strike a balance between accuracy and interpretability as these two requirements have contradictory nature. These two requirements might inconsistent together [25] when designing intelligent systems especially when system complexity increases and nonlinearities become more substantial. In a nutshell, interpretability and learning accuracy are the crucial requirements when designing intelligent model. In spite these requirements are contradictory, they must be met to the highest possible degree to achieve an efficient and optimize architecture.

In light of growing concern about the black-box nature of neural networks, exacerbated by rapidly growing complexity and the size of the constructed models, concept design and discovery has gained substantial momentum to improve interpretability and knowledge extraction and representation. Concept design plays a significant role in numerous fields of machine learning [48] concerning pattern recognition, data representation, prediction, and classification [49], [50], [51]. It aims to capture the essence of data by building experimentally and semantically justifiable relations among data entities. Hence, systems' interpretability and the credibility of classic decision-making models, i.e., the precision of predictions, are enhanced because of their close alliance with how people perceive and structure knowledge [52]. Concept learning [53], [54], [55] is considered as learning unknown data pattern structures and descriptions by utilizing certain methods [56], such as fuzzy concepts [57], [60] and granular computing [57], [58], [59], [191]. Several other approaches [61], [62], [63], [64], [65] have been utilized to characterize explicit concepts to provide a certain linguistic meaning to data collection instead of leaving the interpretation task to the user [61]. Formally, different concept structures with rigorous models have been proposed to specify certain concept meanings and to provide a formal semantics for data analysis [59], [66], [67], [68], [70], [71], [72], [73].

In a more general context, the dissertation is built on a special type of neuro fuzzy systems that is logic-oriented fuzzy AND\OR networks [24]. The logic-oriented fuzzy neurons (AND\OR) [7], [8] are incorporated in neural networks and exhibit superior capabilities to unveil the logic behind decision-making processes to interpret the outcomes linguistically. Furthermore, they show flexibility and sufficient plasticity to allow the network to have learning abilities. The network offers a novel promising processing environment especially when they are integrated with other fuzzy tools such as fuzzy clustering mechanism. We developed several methodologies that utilize fuzzy logic, information granules, neural networks, and fuzzy clustering to improve knowledge acquisition and interpretation.

1.1 Preliminary Definitions

This research revolves around developing explainable, interpretable, transparent and yet accurate intelligent models. The three terms, interpretability, explainability and transparency, might seem undistinguishable in some context due to characteristics similarities. However, the misuse of the terms hinders the establishment of a common ground. Therefore, it is necessary to give fundamental knowledge to remove any potential for ambiguity in the thesis. Generally, interpretability technology, including explainability, transparency, understandability and intelligibility considered as the ability to understand the internal logic, inner workings, and rationale behind predictions [205]. In the following, we present clarifications of the terminologies that we use throughout the context of the thesis [206] [208] [209], [210]:

- **Explainability:** Explainable AI gives insights into how decisions are made by ML algorithms with the capacity to defend their actions and to provide relevant responses to questions.

- **Interpretability:** It is defined as the ability to explain or to provide the meaning in understandable terms to a human. This is a subjective property and user-dependant when evaluating interpretation of a model.
- **Transparency:** A model is transparent if all inner connections can be revealed to understand how variables are related to each other. Usually interpretable models are transparent too.

Deep neural networks are example of complex models that are hard to interpret due to their huge number of parameters and nonlinearity nature. Conversely, rule-based models are typical examples of highly interpretable models. Evidently, some certain models may tolerate a limited performance degradation to achieve a full interpretation operation. In fact, performance and interpretability\explainability are typically considered conflicting objectives [207], [208]. The developed models, in this research, consider meeting these two objectives at the highest possible extent as a primary objective.

1.2 Research objectives and originality

The key objectives of this research are:

- To revisit several logic-oriented fuzzy architectures to analyze and optimize fuzzy networks, and make the resulting structures as robust as possible.
- To address the three fundamental requirements of fuzzy modelling in which the accuracy, interpretability, and transparency trade-off have to be met to the highest extent. The discussion is essential in the era of explainable AI.
- To develop novel strategies to design and analyze concepts and subsequently obtain well-defined semantics with a certain linguistically sound meaning which results in an optimal interpretation of datasets.
- To construct and optimize data clustering mechanism that can be incorporated with logic-oriented structure to improve interpretability\accuracy measures.

This research exhibits several novel facets and addresses new interesting issues. We focus on new directions to utilize logic-oriented networks architecture by addressing the shortcomings of AND\OR networks and integrating several methodologies to improve learning scheme, knowledge extraction and interpretability. Moreover, we take advantage of our study on several optimization algorithms in [9] along with other studies from the literature to assess the implementation of optimization algorithms on logic networks. The thesis sought to stay away from relying on traditional techniques for the purpose of originality in augmenting the effectiveness of logic-oriented networks.

This research exhibits a significant level of originality:

- A new design for concept discovery in data by developing a coherent interpretable model that integrate logic networks into the formation of concepts.
- A better learning approach to overcome the computing overhead of gradient-based learning in logic networks as well as reduce the need for optimization algorithms.
- An innovative fuzzy clustering technique used to enhance and refine the quality of clusters to help in identifying data patterns, or to efficiently reconstruct interpretation models.
- Proposal of a three dimensional taxonomy to clearly lay out a better understanding of the AND\OR architecture to pave the way for new perspectives and possibilities to recognize the full potential of logic-oriented architectures in different realms of Machine Learning.

1.3 Models Development

The dissertation put the logic networks as a focal module in developing interpretable models. Other modules describe parts of data processing and contribute to producing a sound balance between transparency of a system and accuracy of approximation. It is therefore essential modules in forming of concepts in data space.

In Figure 1.1, an overall scheme of the proposed design in this study is depicted.

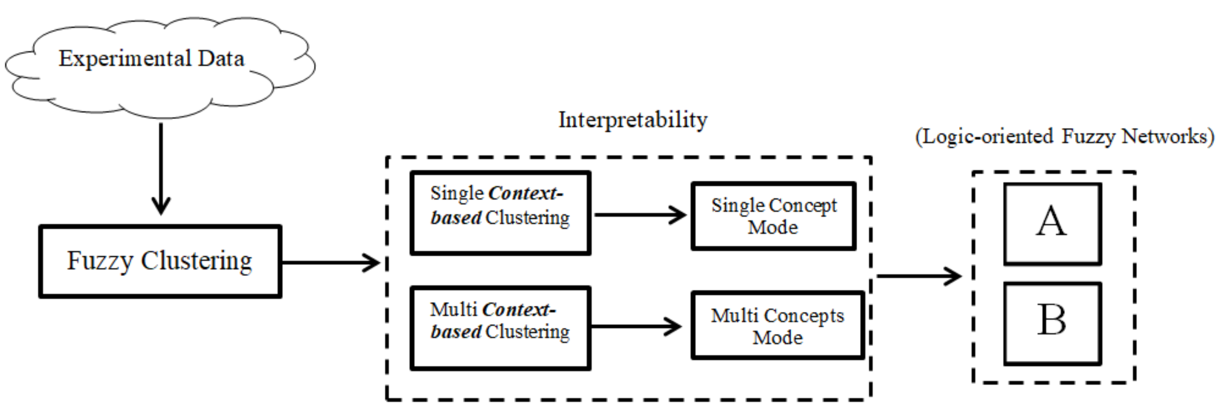


Figure 1-1: Overall scheme for the design and construction of the granular fuzzy models.

As it is displayed in the figure, the scheme starts with experimental data which may come as input-output pairs of numerical synthetic data, or from real-world data sets. After the data is preprocessing, it is fed onto the ensuing modules as follows;

- *Fuzzy Clustering*: Clustering plays a crucial role in linguistic modeling by forming several information granules from numeric data to form the backbone of the model. Fuzzy C- Means is used to find data patterns in the output space and construct fuzzy sets before they are given linguistic landmarks. The data patterns are represented as a partition matrix and given a linguistic description to reference a specific concept in the output space. The ensuing modules concern building fuzzy relations realized as triangular norms in a learning framework.

- *Conditional Fuzzy C-Means*: This algorithm introduces directionality into forming relations between input and output spaces. A condition, from the output space, constraint identifying partitions in the input space and consequently the constructed clusters (fuzzy sets) are focused and more reflective. This module occupies a central role in the development of interpretable model by identifying relationships between output-inputs in the form of *concepts*. Forming concepts might go in a single mode or in multi concepts mode which depends on the form of conditional clustering used. In the multi context-based clustering, several referential information granules are identified and connected to several fuzzy sets in the input space simultaneously. Subsequently, the revealed interpretation is more accurate and informative than the genuine (single) concept mode. An improved conditional clustering algorithm is presented to help carrying out multi-concepts discovery and augment interpretability capability.

- *Concept discovery*: a concept constitutes a pair of reference information granule and its counterpart fuzzy sets already established in the input space through a clustering mechanism. A concept provides a better and yet a clear interpretation of a knowledge domain. All modules contribute in forming and optimizing the concept. Here, concept discovery uses fuzzy clustering as a central method in most of the steps of forming concept since clustering is a key algorithm is forming information granules. A concept is better way for knowledge extraction and representation. The synergy between all other modules presents highly cohesive linguistic dependency between information granules and their counterpart fuzzy sets.

- *Logic-oriented network*: This is probably the pivotal module of the framework. Once the information granules (fuzzy sets) are obtained from the previous modules, interpretation is extracted in the form of logic expressions. It provides quantitative and qualitative relationships of inputs-outputs mapping through a learning mechanism along with logic-oriented fuzzy neurons. In (A), structure discovery is completed based on the formed concepts in the previous module whereas (B) concerns a learning scheme; ideally it is a gradient-based. Typically, gradient algorithms experience several drawbacks which have been discussed thoroughly in this dissertation. Hence, we worked in (B) to improve the accuracy of approximation while decreasing the effects of gradient learning. The improvement is completed by the utilization of feedforward learning side to side with a gradient-based learning in a two-phase design.

While the outcomes of this work are significant to improve interpretation of datasets, interpretability feature remains a subjective property that deals with the issues surrounding algorithmic opacity. This feature, interpretability, is user-oriented and therefore, it depends on the perception of a user to satisfy their own interests in accordance with their position in human-centric models. When we say “subjective property” we mean that there is no exact meaning of what facilitate understanding the relationships in data. In this work, we discuss characteristics that impact interpretability and we try to distinguish interpretability from other interpretation-alike features such as transparency, explainability and knowledge representation. The discussion is essential in designing effective interpretation framework of data.

1.4 Dissertation organization

The dissertation flowchart shows how the document is organized (Figure 1.2). The document starts from Chapter 1, were we introduced an overall picture of fuzzy and linguistic modeling through logic-oriented networks. Then, we move to Chapter 2 to present a literature review of each components used in the design along with identifying several shortcoming of the whole design. Furthermore, we present full background knowledge on construction interpretable, and transparent, linguistic models. We took the advantage of the literature review presented in chapter 2 to present an improved learning mechanism in Chapter 3. Background knowledge of logic-oriented networks, neurocomputing and concept discovery were expanded further in Chapter 3 and Chapter 4 to cover essential information. Subsequently, the logic networks were utilized in Chapter 5 to presents an interpretation framework of data by using fuzzy clustering. Hence, knowledge is extracted and presented in understandable terms (logic expressions) for users. Afterwards, we move to Chapter 6 to present an enhancement made on fuzzy clustering for better interpretation framework of data to clearly reflect the essence of experimental data. Finally, we arrived to chapter 7 to present a brief discussion of what we have presented in the dissertation, conclusions and future directions.

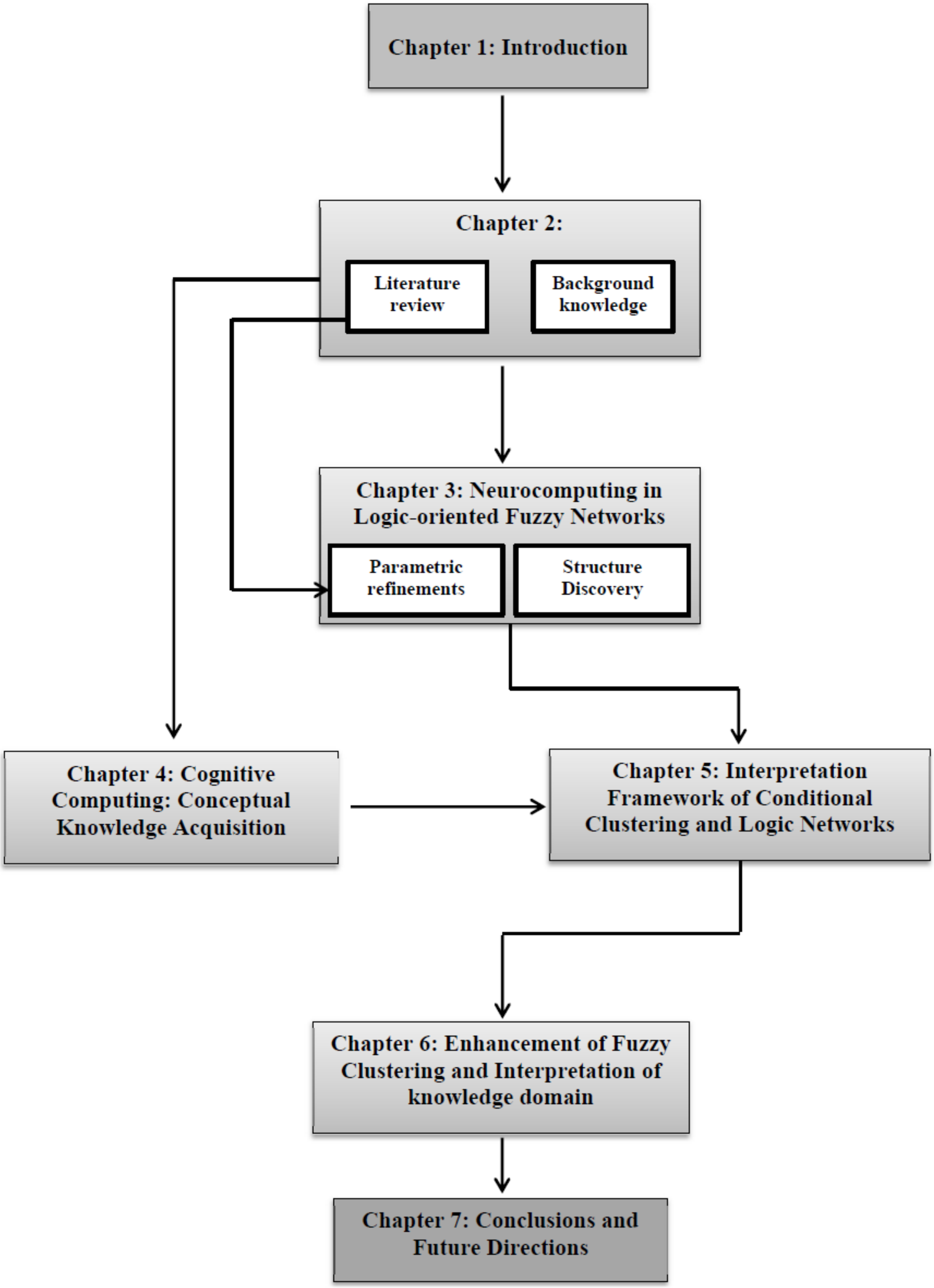


Figure 1-2: Dissertation flowchart

The subsequent chapters are structured as follows

Chapter 2: Literature Review and Background knowledge

In this chapter, we provide a comprehensive review of logic oriented neural networks starting form formal Boolean logic till the construction of logic-oriented neural networks. The review emphasizes one of objective of this thesis which is revisiting and discussing logic-oriented fuzzy networks from different perspectives. Consequently, the development and enhancement presented in this thesis is ensured to exhibit high degree of originality.

Additionally, this chapter presents full background knowledge on construction interpretable, and transparent, linguistic models. We tried not to be repetitive by covering all essential information in this chapter and not repeating background in the ensuing chapters.

Chapter 3: Neurocomputing in Logic-oriented Fuzzy Networks

This chapter is concerned with the adaptive information processing between objects through neuromorphic processes. The underlying techniques of designing logic networks, namely; structure discovery and parametric refinements, are presented. Moreover, we discuss the learning mechanism in-depth to assess the performance of gradient-based learning applied to AND\OR networks and offer alternative solutions to the optimization of the constructed networks. Therefore, the knowledge embedded in the neurocomputing system can be represented in a well-structured form, such as logic expressions or fuzzy rules.

The chapter is divided into three subsections to reflect the design process of logic-oriented fuzzy networks:

3.1 **Construction of Information Granules**: Information granules (Fuzzy sets) serves a vehicle for the construction of the logic-oriented networks, offering a convenient and efficient way to reflect the existing experimental data. It is the initial building block for the construction of the networks.

3.2 **Structure Discovery**: This step aims to set up the logic skeleton of a problem. It utilizes several to find an optimal construct such as especially optimization algorithms.

3.3 **Parametric refinements**: we discussed gradient and non-gradient learning schemes with a complete assessment of several optimizers. Also, we present a better technique to improve parametric refinements by focusing on Feedforward neural networks.

The outcomes of this chapter were utilized in subsequent chapters in the formation of concepts to extract useful knowledge from datasets. Both performance index and interpretability of knowledge domain were carefully assessed to ensure optimal results

Chapter 4: Cognitive Computing: Conceptual Knowledge Acquisition

Cognitive computing implements computational intelligence to solve the problems of imprecision, uncertainty and partial truth in biological systems. The recent advances in several computing disciplines such as neuromorphic processors, big data, predictive modeling, machine learning, are accelerating advances in cognitive science and cognitive computing. The field of cognitive computing has been one of the main pillars in knowledge extraction and representation. Therefore, the intention of this chapter is to discuss cognitive computing systems and the formation of concepts in data by simulating human cognitive and neuromorphic processes. Furthermore, the chapter thoroughly discusses concept analysis and conceptual clustering from interpretability perspective to clearly lay out some fundamentals of knowledge interpretation.

Chapter 5: Concept Discovery: Interpretation Framework of Conditional Clustering and Logic Networks

This chapter describes the overall architecture of a concept discovery model which revolves around a logic-oriented network of intuitively structured collection of information granules pairs. The details of the construction of concepts and their optimization are presented, where we focus on a specialized fuzzy clustering algorithm; a so-called context-based FCM algorithm. This algorithm considers output space when clustering input space which improves the construction of linguistic, interpretable, models.

Chapter 6: Enhancement of Fuzzy Clustering and Interpretation of Knowledge Domain

This chapter introduces an extension of the generic conditional Fuzzy C-Means (CFCM) for better interpretation of datasets. First, we improve the mathematical model of the context-based FCM cluster to interpret several concepts positioned in the output space simultaneously, and in consequences the interpretation has improved to reflect the essence of the data sets. Experimental evidences are presented to assess the method efficiency.

Chapter 7: Conclusions and Future Studies

A number of conclusions are drawn from our research in this chapter, and we suggest a number of directions for future works.

2. Literature Review and Background Knowledge

The development of highly accurate and interpretable models reveals deep understanding of systems' behaviour to support decision-making process. In real-world system modelling, it is difficult to efficiently build highly transparent model even with rigorous mathematical ground due to the fact that only input and output data acquired from running the process is accessible for the design. Therefore, uncertainty and vagueness in predicted outcomes arises as a focal problem in knowledge representation. It is preferable to analyze data in a context of fuzzy sets with the use of such information granules to facilitate precise knowledge extraction and cope with uncertainty in knowledge systems.

In this chapter, we comprehensively review logic-based networks with all other components used in knowledge representation and extraction for better interpretability. Furthermore, the background knowledge presented here aims to cover all different aspects of different techniques used in linguistic modelling through fuzzy logic. Additionally, we covered, in an introductory manner, the contradictory requirements of fuzzy modelling, that is; interpretation capabilities and accuracy of approximation.

2.1 Interpretability and Explainability of Machine Learning

Traditionally, AI practitioners consider system accuracy to be the main metric in their workflows. However, this metric is far from sufficient to reflect the reliability of decisions made by ML models. Several current ML models are found vulnerable to imperceptible attacks [213], biased against underrepresented groups, lacking in user privacy protection which result in degradation of end-user experience and trust [214]. Therefore, various metrics, beyond system accuracy, should be considered to improve models' trustworthiness including but not limited to algorithmic fairness, explainability, and transparency. In important high-stake domains, such as in healthcare, those metrics are vital to the adoption of machine learning which, if not applied properly, may have serious consequences.

The ultimate purposes of interpretability\explainability are to understand the underlying decision-making processes, justify the predictions and to help us to understand how, when, and why predictions are made [211]. Therefore, it can build greater trust and improve the safety of our use of AI models. Practically, AI models are aimed to achieve specific quantitative objectives. On the other hand, explainable AI reveals inner relations of a model to evaluate whether or not an objective has been truly met. Furthermore, decisions that are relevant, easy to understand, and not prone to misrepresentation are highly trusted which lead to better adoption by humans. In case of black-box models, user trust can be manipulated by creating a framework to generate misleading explanations which can be, mistakenly, verified by experts. Therefore, level of trust will rise in conjunction with the level of models' transparency. Several metrics must be utilized when designing ML models in order to arrive to efficient construct. Each of these metric is closely tied to each other and may overlap in their underlying meaning. They are distinct in their desired outcomes and characteristics [211], [215].

- Understandability: The concept of understandability, also known as intelligibility, is the property of the overall model to be understandable without any need for details and explanation of its internal algorithmic structure used by the model. Therefore, the functionality of an AI model must be understandable to humans.
- Comprehensibility: It refers to the ability of a model to represent and convey its learned knowledge in a human-understandable fashion. In general, measuring how well humans can understand explanations is difficult in a nominal sense, but somewhat easier from a relative perspective.
- Interpretability: The goal of interpretability is to describe the structure of a model in a way that is easily understandable by humans. Subsequently, model predictions can be explained.
- Transparency: It helps to comprehend the basis of a model and reveal the internal relations. It is worth mentioning that a model can have different degrees of understandability.

The progress on interpretability has been difficult to measure, as lack of clear consensus definitions have exposed interpretability's inherent subjectivity and field-specific meanings. This has limited the real-world impact of interpretability methods [205], [216], [217].

Working with explainable AI forces us to deal with various types of trade-offs, not only the well-known accuracy-explainability trade-off. For instance, the General Data Protection Regulation (GDPR) introduced by the European Commission [219] enforces data privacy as an inherent consideration in real-world systems. This requirement may limit explainability of the systems which can affect system overall efficiency. It is because the need to hide some necessary private data from being visible in the explanation process to meet privacy-preserving requirement. Therefore, there is a need to address this dilemma when designing privacy-preserving machine learning models. Finally, improving explainability of AI models may limit other performance metrics. Hence, understanding the inner relations and the limitations of different explainable methods may help to meet all requirements, simultaneously, at the highest extent.

2.2 Fuzzy and Probability: The Relationship

In this dissertation, we think it is crucial to start off with a concise clarification of the relationship between fuzzy and probability theories to eliminate any ambiguity that may arise in the subsequent chapters. Although many papers have discussed this topic from different perspectives, the short discussion here is meant to be focused and self-contained.

The relationship between fuzzy set theory and probability theory has been under debate for a long time [11], [196]. As a matter of fact, probability theory and fuzzy set theory are distinct theories with different agendas. Fundamentally, probability theory may be viewed as a formalization of perceptions of likelihood and count; fuzzy set theory may be viewed as a formalization of perceptions of class and similarity [196]. Fuzzy set theory describes partial truths, i.e., degrees of truths. According to the "Principle of Valence", propositions are not only either true or false but they can have intermediate truth-values which are modeled by numbers in the unit interval. Fuzzy sets theory derives that fuzzy logic is logic of partial degrees of truth, imprecise notions, and propositions, which may be more or less true, and expressed through the natural language. Fuzzy sets theory has embraced uncertainty modeling when membership functions have been reinterpreted as possibility distributions [38,39] such that a fuzzy set μ measures the belongingness of x to set A. This conceptual extension was necessary when fuzzy logic was used in several domains such as pattern recognition and control engineering. On the other hand, probability theories describe the degrees of uncertainty when there is lack of knowledge about the truth. The probability $p_A(x)$ measures the knowledge or ignorance of the truth of the event that x belongs to the set A. Therefore, the probability of a proposition, true or false propositions, is the degree of belief on the truth of that proposition.

Regarding the differences between probability theory and fuzzy theory, Dennis Lindley states that probability theory is sufficient for dealing with uncertainty and imprecision of any kind, implying that there is no need for fuzzy set theory and therefore probability is the only sensible description of uncertainty and is adequate for all problems involving uncertainty [197]. However, Zadeh in [198], [199] has a contradiction view to Professor Lindley's assertion by stating that some problems, even simple problems, cannot be solved through the use of standard probability theory. Furthermore, Zadeh discusses probability theory by rooting it to two-valued logic (bivalent-logic-based-theories) where basic concepts are defined as bivalent concepts, with no shades of truth allowed while, in reality, most basic concepts are fuzzy. Hence, the capability of probability theory to deal with real-world problems can be enhanced through the generalization of fuzzy theory. It is therefore better to think about the relationship between probability theory and fuzzy logic as complementary rather than competitive since probability theory and fuzzy logic have distinct agendas and different domains of applicability [204]

2.3 Fuzzy Systems Modelling

Fuzzy systems are grounded on fuzzy logic which was introduced by Lotfi Zadeh in [1] to model the interactions and relationships among variables in vague terms instead of precise Boolean logic. They have proven to be an important tool for modeling nonlinear and complex systems in which classical approaches are unsuccessful due to their complexity. The systems provide an effective conceptual framework to solve the shortcoming of the conventional approach to knowledge representation that is based on two valued logic. Systems built around fuzzy logics exhibit two essential advantages: the capability of fuzzy sets to handle uncertainty and the approximate reasoning of fuzzy logic. Zadeh also proposed the use of linguistic landmarks to allow experts to utilize knowledge through fuzzy logic [26], [27], [28]. In this consideration, The relationships between variables are commonly presented as collections of fuzzy *if-then* rules and each rule consists of one or more rule terms (antecedents, consequents) and unites them using logical terms AND or OR. In this sense, rules can be defined by capturing human expertise and incorporate them into expert systems, or they can be extracted through a learning approach. Subsequently rules present linguistic interpretation of knowledge domain instead of being a black box without sacrificing the accuracy of the constructed model. In a broader sense, fuzzy modelling deals with modes of human reasoning that are approximate rather than exact [39]. Therefore, the essence of fuzzy modeling is concerned with the development of relationships between information granules regarded as fuzzy sets or fuzzy relations. The diversity of existing models is enormous and well documented in the literature with a vast array of methodologies, architectures, and algorithms [10].

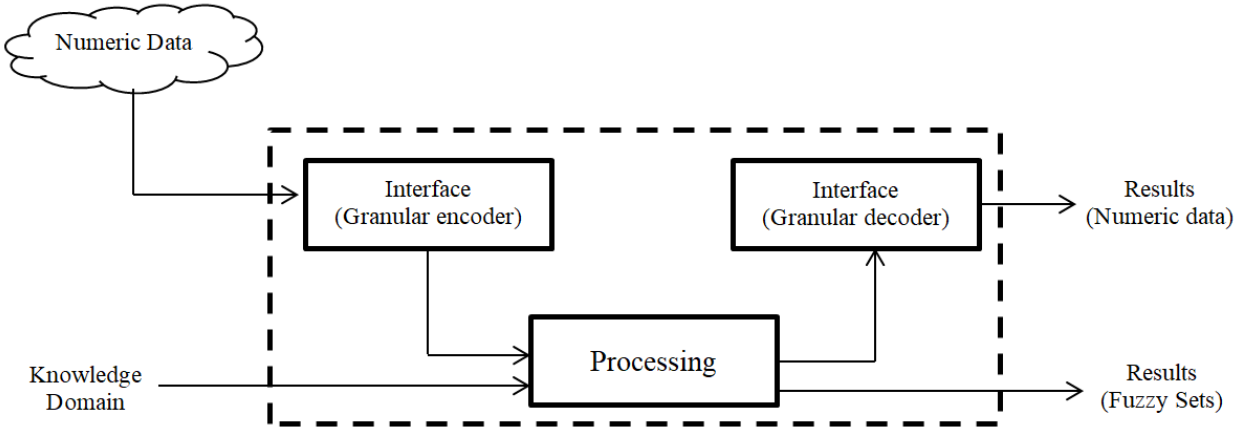


Figure 2-1: A general view of the underlying fuzzy model showing the three main modules. Note that, the results might be in the form of fuzzy sets or numeric data conveyed to the numeric environment.

A general architecture of fuzzy models is depicted in Figure 2.1. The figure shows the three components where each of them comes with well-defined objectives. The architecture components are able to interact efficiently due to their underlying well-defined mechanism. Starting off with the numeric data, it is transferred through the interface and sent to the processing core. The numeric mode is regarded as a multivariable nonlinear input–output mapping resulting numeric values of the corresponding output. In another arrangement, the input-output mapping is completed in the context of rule-based systems which brings substantial level of modeling flexibility. The integration of knowledge domain into fuzzy modelling for the interpretation purposes supports the user-centric facet of fuzzy modeling [7].

Rule-based modelling have been presented and widely adopted in the literature. It requires that any input is transformed and expressed in terms of fuzzy sets. It captures relationships among fuzzy sets and link linguistic terms with their computational realization. Subsequently, it produces highly qualitative linguistic description of some piece of knowledge. The main modelling schemes of rule-based systems are; Mamdani [74] and Takagi-Sugeno-Kang (TSK) [75]. Both presents transparent relationship pertinent to the problem with the main difference between them is that Mamdani-type fuzzy rules consider a linguistic variable in the consequent, whereas TSK fuzzy rules are based on representing the consequent as a polynomial function of the inputs. Mamdani systems are most often classified as a form of approximate reasoning, and defined as “the process or processes by which a possible imprecise conclusion is deduced from a collection of imprecise premises” [47]. This type of modeling is more interpretable because both the premises and consequents of the rules are fuzzy sets while the TSK types are more accurate and computationally efficient. Mamdani fuzzy rules for a fuzzy system can be described as follows:

$$If\ x_1\ is\ M_1 and\ \dots and\ x_n is\ M_n \qquad Then \quad Y = y_1\ is\ M_3$$

where $x_1, \dots, x_n$ are input variables, y_1 is the output variable, $M_1, \dots, M_n$ are fuzzy sets and AND are fuzzy logic operator. "$If\ x_1\ is\ M_1 and\ \dots and\ x_n is\ M_n$" is called the rule antecedent, whereas the remaining part is named the rule consequent.

On the other hand, TSK style rules formalize the ability of rules to predict in a real valued output space, making them candidate solutions to continuous function estimation or regression problems. TSK is composed of several rules in the following format:

$$\text{if } X \text{ is } A_i(\boldsymbol{x}) \text{ then } y = f(\boldsymbol{x}, \boldsymbol{a}_i), i = 1, 2, \dots, c$$

where $\boldsymbol{x} = [x_1, x_2, \dots, x_n]^T$ is a vector of n input variables, y is the rule's output, $A_i(\boldsymbol{x})$ is the membership function of the i^{th} multivariable fuzzy set, c is the number of rules and $f(\boldsymbol{x}, \boldsymbol{a}_i)$ is a local polynomial. The previous formula can be rewritten in another format as follows:

$$If\ X_1\ is\ A_1 and\ \dots and\ X_n is\ A_n THEN\ Y = p_1.X_1 + \dots + p_n.X_N + p_0$$

Note that, in both modelling schemes the rule parts, antecedent and consequent, formed as information granules, conceptual entities, that are semantically sound abstractions. These rule based systems are adapted for different systems and exercise predication properties that make them ideal for several applications in different domains such as geophysics, image processing, pattern recognition and Industrial applications [7], [13], [40], [41], [42], [43], [44].

The two terms "*Fuzzy logic*" and "*Fuzzy sets*" constitute the basis for setting up the linguistic labels and meanings through building the fuzzy relation matrices according to the input-output process data. A fuzzy relation is used to represent and quantify associations between objects; in the case of a Boolean relation, it represents the presence or absence of an association, this concept is generalized in fuzzy relations by allowing various degrees of association between numerical entities [7], [76]. The transformation process, from Boolean to fuzzy, is done through some composition operators. These compositions are realized by using triangular norms [77]. A triangular norm is a binary operation used in the multi-valued logic framework to represent logical conjunction or disjunction in fuzzy logic. The triangular norms (*t-norms*) are used to represent an AND operation, in fuzzy set theory. The t-conorms (*s-norms*) are dual to the t-norms and used to represent an OR operation. The realizations of the *t-norms* and *t-conorms* have direct and significant impact on the approximation capabilities of a modeled system. In this dissertation, we rely on the product and probabilistic sum as a realization *t-norms* and *s-norms*, respectively, to allow for the highest level of interactivity between objects.

	t-norm	t-conorm	Effects
Min\Max	min (a, b)	max (a, b)	Lack of interactivity between operands. Results only reflect the influence of one operand.
Product \ Probabilistic Sum	(atb = ab)	(asb = a + b - ab)	Provide interactivity, but at a high computational cost

The fuzzy operations, AND\OR, are significantly important in constructing logic-oriented fuzzy neural networks. Here, logic-oriented fuzzy operations through AND\OR neurons replace the conventional neurons found in neural networks with substantial increase in advantageous outcomes presented as interpretation capabilities.

2.4 Logic-Oriented Fuzzy Networks

Fuzzy Neural Networks (FNNs) [35], [144] represent a generalization of the McCulloch–Pitts neurons [145] in which inputs and weights are either real or fuzzy values. FNNs extend the benefit of multi valued logic by the integration of expert knowledge into the system, and are considered inherently more understandable because of their use of human-like fuzzy inference. Therefore, the networks are easily comprehended by a user. The networks are built upon the advantages of neural networks and fuzzy systems. They are efficiently capable of approximate reasoning and handling uncertainty due to their composition nature that is structured as fuzzy-sets driven with the aid of logic processing units, fuzzy neurons [6] [8] [24].

This work focuses on a special type of infinite-valued-based architecture that is endowed with outstanding abilities to eliminate ambiguity in knowledge domain through interpretation capabilities. The architecture synergistically integrate AND and OR logic-oriented fuzzy neurons with neural networks to improve, with high efficiency, data representation and knowledge extraction to capture uncertainty in datasets [36], [37], [80]. In fact, it constitutes a useful vehicle to quantify data uncertainty [78], [79]. The network, in its basic architecture, consists of logic-driven AND/OR neurons placed in series in the hidden and output layers of a three layered neural network (Figure 2-2). It combines the learning abilities of neural networks with interpretation capabilities that are the consequences of utilizing AND/OR neurons.

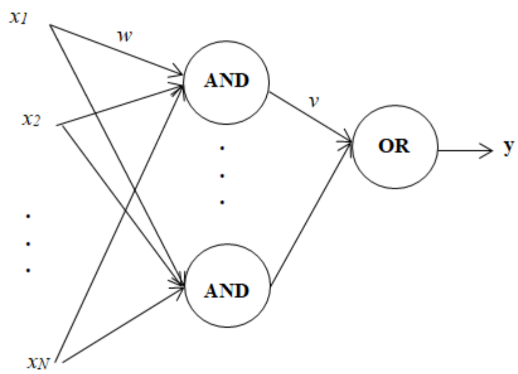


Figure 2-2: General topology of AND/OR network.

The design of fuzzy networks starts at early stages by the determination of the size of fuzzy sets (information granules) to show a level of detail one wants to capture. It affects interpretability as both specificity and coverage could reveal (or hide) some important information. Later, the development of logic-oriented network for the approximation of experimental data concerns two main fundamental facets which are structural discovery and parametric learning. The structure discovery is the process of setting up the logic skeleton of a problem followed by the learning technique, ideally a gradient-based learning, to optimize the connection and complete the development process. The two steps are completed separately. However, both are tied together in which good results of one phase positively impact the other. Initially, we start with constructing a topology of the network and select the realization of *t-norm* and *t-conorm*. Selecting the type of topology means the organization of neurons (AND\OR or OR\AND) followed by choosing the number of neurons in each layer. Parametric learning comes next to adjust the weight of the connections of the neurons. In the learning phase, where interpretation is revealed, the iterative nature of the interpretation process allows a designer to monitor and control interpretation (level of details) and readability through pertinent pruning mechanisms [80], [81].Those values are might not work side by side with performance index since all networks' parameters are independent from each other and thus the network preserve high degree of controllability and plasticity. The development process is carried out in an iterative fashion where the performance is assessed, and if not accepted the network is revised. Once the network meets certain objectives in terms of its accuracy, interpretability and stability, hence the network has reached an optimal design.

In the literature, we can encounter a plethora of different approaches employed to construct AND\OR from numerical data. The paper [80] addresses the interpretability challenge of neurofuzzy system through the underlying logics. It reveals models' transparency through a collection of logic driven processes and shows that the presented logic-oriented networks retain a well-define interpretability and exhibit plasticity by mutually enhancing neurons and connections. This paper could be set as a backbone of later studies on AND\OR architectures as it shows the practical benefits of neurofuzzy computing. In a different setting, Evolutionary Computing (EC) has been used to augment AND\OR construct to help combat high-dimensional problems in which gradient-based learning is vulnerable. Furthermore, EC provides optimization support in the case of large space of structures. Subsequently, only a small subset of features is essential and their reduction may help to develop compact and efficient AND\OR networks [81], [82], [83], [85]. The independency of group-based optimization, such as EC algorithms, from the gradient information makes them more robust and suitable for complex problems. Ideally, it is better to use hybrid techniques where a population-based algorithm is set to find initial structure or fuzzy sets followed by a gradient-based learning to optimize the connectives and lower the computing overhead. The type of EC-based algorithm to implement, or to start with in case of hybridization, is always a problem-dependent issue.

An interesting implementation of AND\OR neurons is with hardware design to take advantages of power consumptions and analog electronics [86], [87], [88]. These types of implementations give a perception of the plethora of areas in which the logic networks can be used. Several other implementations of AND\OR architecture worth mentioning to stress the adoptability of the architecture such as in biomedical pattern classification [89], [90], software engineering for describing software process and products quantitatively [91], data representation and dimensionality reduction [92] and finally Karnaugh map minimization [80]. Additionally, the fuzzy logic operators (AND\OR) offer various other advantages such as transparent data representation which allows introducing a novel granules logic-oriented autoencoder [92].

2.5 Concept Design and Discovery

A convenient definition of concepts is inspired by Port-Royal logic [93], which indicates that a concept consists of its extent and intent. The extent of a concept is a collection of all objects that are covered by the concept, while the intent is the collection of all attributes covered by the concept. Following the Port-Royal logic, a concept structure follows the format such that a concept p_2 is a sub-concept of p_1, which implies that p_1 is a super-concept of p_2. In this manner, a structure of concepts can be built no matter what kind of linguistic meaning we give to the concepts. The logic description of a concept (that is, a pair of (object, attributes)) has inspired several researchers to follow a similar path for knowledge representation and extraction in several domains.

Concept discovery was first incepted in [94] to describe a collection of objects $\boldsymbol{O}$ positioned in a space of attributes A to achieve optimal knowledge acquisition and representation. The literature is rich of methodologies to tackle the design and learning of explicit concepts that preserve a high level of generalization. The ground-breaking paper by Wille [95] is one of the earliest works to describe a concept by defining a set of objects $\boldsymbol{O}$, a set of attributes $\boldsymbol{A}$ and a binary relation in the form $I \subseteq O \times A$, to indicate the relations between an object and its counterpart attributes. A complete concept lattice then formed by the set of all formal concepts of formal contexts with their relations to sub-concepts and super-concepts. This method, called formal concept analysis (FCA), has since been applied in many domains due to its mathematical effectiveness [48], [96]. One limitation of Wille's concepts is that only one set is allowed as an extent (or intent), while more extents (intents) must be disjunction expressions. This limitation was addressed in the monotone concept [107], which allows for a set of unions for the extents and disjunction for the intents.

Concept analysis and discovery has been generalized to a theory on fuzzy relations [93], [97], [98]. Fuzzy formal concepts introduce a structure L to form a complete resituated lattice and provide a scale of truth values with its structure. The binary relation is interpreted as the truth degree to which the object $o \in \boldsymbol{O}$ has the attribute $a \in \boldsymbol{A}$. Other types of concepts with different formation characteristics, such as power concepts [99], property-oriented concepts [72], object-oriented concepts [70], [71], AFS concepts [103] and approximate concepts [62], [104], have been presented to meet different requirements of cognitive knowledge discovery. Those well-defined concepts can be distinguished from one another according to the characteristics of their intensions, whose forms may be conjunctive, disjunctive or mixed [106]. Object-oriented and property-oriented concepts [70], [71], [72] are based on rough set theory and incorporated into FCA to discover useful data patterns. The work in [108] utilizes property-oriented concepts to tackle the problem of uncertainty and complexity in a formal context that is traditionally composed of 0s and 1s. It uses linguistic values to describe the connotation of a decision in the formal context directly, thereby reflecting the importance of group meaning.

AFS concepts, a family of completely distributive lattices, are related to the semantic interpretation of fuzzy attributes. The approach has been applied to fuzzy clustering [109], [110] to meet the requirements of real-world applications. Further improvements on fuzzy AFS concepts were presented in [111] to mimic human reasoning and provide definitive semantic interpretation. Fuzzy sets emphasize the idea of partial membership to offer a possibility of dealing with concepts where the binary view of the underlying concept is not suitable or overly restrictive [112]. They solve the issue of limited interpretation capabilities for a full linguistic description of concepts.

Concept learning aims to acquire and extract valuable knowledge according to human cognitive processes. It is a very active research area with several challenges, especially when processing consistently changing data [105]. Generally, concept learning for a certain structure from certain datasets is computationally expensive [59]. Thus, fuzzy logic and granular computing [112], [113], [114] have been incorporated into concept learning to effectively improve knowledge discovery. Furthermore, concept learning requires less processing time than is required for detailed numeric processing [33], [34], 41]. A concept granular computing system comprises two complete lattices and two operators [31]. The incorporation of granular computing into FCA allows for eliminating redundant concepts [42], [43] from the constructed concept lattice of formal context, and thus, the knowledge processing efficiency of FCA improves.

2.6 Fuzzy Clustering in the Development of Interpretable Models

Fuzzy clustering [7], [20], [46] establishes a solid framework for constructing linguistic models by forming clusters geometry in the feature space and hence clusters become more robust and reflective of knowledge domain. In general, Fuzzy clustering partition a collection of data points into a number of clusters, where the objects inside a cluster show a certain degree of closeness or similarity, defined by a membership grade, to different clusters in a soft clustering fashion. The similarity is measured mathematically by well-defined distance functions to discover boundaries and classify objects into subsets. Fuzzy clustering deals with uncertainty in data in which there is no a priori information on the data distribution given through unsupervised learning. It is, therefore, an essential technique in user-centric modelling where the models should help the user to justify decisions being taken and hence outcomes are readily understood. Furthermore, fuzzy clustering is an effective explanatory analysis technique and can deliver findings at the level of information granules which is an important facet in designing linguistic models where interpretability arises as a focal manifestation.

2.6.1 Fuzzy C-Means

The generic Fuzzy C-Means algorithm was proposed by Dunn [116], was called ISODATA, when extended the hard means of clustering to preliminary concepts fuzzy means. Bezdek [2], [117], [118], [120] improved fuzzy clustering to more a general case by adding fuzzy factor and proposed FCM. To improve convergence rate of FCM, several other variants of the algorithm was proposed. Wei and Xie [140] proposed rival checked fuzzy C-means clustering algorithm (RCFCM) on the basis of competitive learning. RCFCM magnifies the biggest membership degree and suppresses the second biggest membership degree to improve convergence speed. Later, suppressed fuzzy c-means clustering algorithm (S-FCM) is proposed [141]. The new algorithm establishes more natural and reasonable relationships between hard c-means clustering and FCM algorithm by eliminating sensitivity to the fuzzy factor. The work in [142] modifies the objective function of FCM algorithm to compensate for such intensity inhomogeneity induced by the radio-frequency coil in magnetic resonance imaging (MRI). It tailored the FCM algorithm as a bias corrected FCM (BCFCM) with the regularization of the FCM objective function with a spatial neighborhood regularization term. This algorithm was improved in [143] to lower the computational complexity by replacing Euclidean distance with a kernel-induced distance and proposed kernel versions of FCM with spatial constraints.

The generic FCM method has some disadvantages such as the need for a large amount of time to converge and the sensitivity to noise, because of squared-norm to measure similarity between cluster center and data points. The noise in the clustering refers to the outliers of the data set in the clustering which have significant numerical difference from most of the other data or have a large spatial distance with areas with high density of data points [115]. Outliers affect FCM robustness despite the algorithm stability which could compromise the performance of ensuing clusters [123], [124], [125]. This issue has been in the centre of research activities of fuzzy clustering to limit its downsides. The work in [126] proposed a noisy cluster algorithm to assign noisy data points to what is called a noise class to eliminate the effects of outliers. Another technique was proposed in [127] to make the clustering algorithm more robust by using an iteratively reweighted least-squares technique (IRLS) combined with the Mahalanobis distance. Although the algorithm has some disadvantages, it remains one of the most commonly used objective function-based clustering techniques that form a prerequisite for pursuing other processing. It is very efficient when representing the essence of the numerical data.

The way how Fuzzy C-Means algorithm works is by calculating the centers of clusters v and the membership matrix U to partition the data space to minimize the following objective function:

$$Q = \sum_{i=1}^{c} \sum_{k=1}^{N} u_{ik}^{p} \| x_k - v_i \|^2 \tag{2.1}$$

where c is the number of clusters, $\|.\|$ is the Euclidean distance function computed between x_k and the prototype (v_i) of the ith cluster, and p is the fuzzification coefficient (>1) which affects the shape of the produced membership function. We, typically, choose the value to be equal 2. Minimizing the objective function must satisfy one constraint:

$$\sum_{i=1}^{c} u_{ik} = 1 \tag{2.2}$$

The partition matrix and the prototypes are updated iteratively in each context to determine the structure of each dataset falling within a fuzzy set A_i. The iteration mechanism to update the membership matrix follows:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{\|x_k - v_i\|}{\|x_k - v_j\|} \right)^{\frac{2}{p-1}}} \tag{2.3}$$

Typically, the Euclidean distance measure is used. It helps to avoid any bias toward any particular component. The objective function is minimized when large membership values are assigned to input patterns that are close to their nearest cluster centers and low membership values are assigned to those that are far from the cluster center. The cluster centers (prototypes) are calculated as follows:

$$v_i = \frac{\sum_{k=1}^{N} u_{ik}^{p} \, x_k}{\sum_{k=1}^{N} u_{ik}^{p}} \tag{2.4}$$

FCM, sometimes, fails to discover data structure in data because it always takes all the points in given data to calculate the new cluster leading to high sensitivity to outliers. Moreover, FCM has a long computational time in dealing with large scale of data as the number of iterations will significantly increase when data and dimension increase. Furthermore, Fuzzy C-Means is largely influenced by the initialization of the partition matrix. When the algorithm starts with the imperfect initial condition, it would have the risk of getting into local extremum and cannot get the optimal solution. Furthermore, the performance of FCM is always dependent on the dataset even with high adaptability of FCM on a wide range of engineering and scientific disciplines. In a nutshell, FCM algorithm is efficient, robust, and has been extensively (and successfully) implemented in different domains even with some downsides that must be taken into consideration.

2.6.2 Conditional Fuzzy C-Means

A modified algorithm of FCM, called Conditional Fuzzy C-Means (CFCM) [4], is geared towards forming direction-sensitive information granules of fuzzy clusters. In CFCM, the search for data structures (clusters) is narrowed down to a certain context (a subset of output space) making the discovery of structure more promising and the computing more manageable. However, this algorithm does not necessarily reflect the essence of experimental data and could be a realization of developer design because of the added context (condition).

The formation on context (condition) is performed in the output space (the space of condition variables) as an independent task. For a specific condition, clustering is performed to reflect on the pertinent portion of data in the input space (Figure 2-4). By changing the condition, clustering focus on different parts of input space to form different information granules. Conditions in this case are fuzzy sets with well-defined semantics that is experimental evidence. The formulation of clustering differs from the generic fuzzy clustering. Instead of "cluster data set," we formulate the clustering process as "cluster data set in context *A*" as *A* being a fuzzy set defined in the output space. The task of fuzzy clustering can be reformulated to reflect the added condition in the following way:

Determine data structure (cluster) in **X** under condition (context) ***A***.

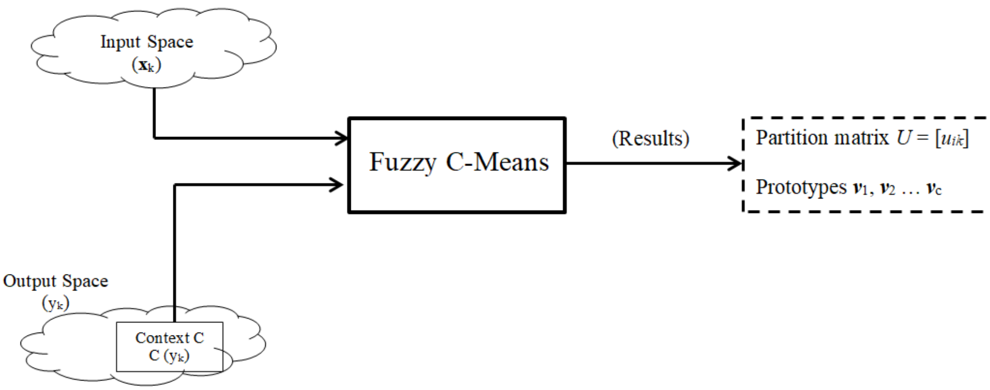


Figure 2-3: Conditional clustering exhibits similar process as in the generic fuzzy clustering

The clustering method in CFCM is the same as FCM that we thoroughly discussed in the previous section. However, the essence of this method is that clustering is carried out in the context of some auxiliary information granule so that the formation of the structure is conditional implied by some additional piece of knowledge. Therefore, it is an example of knowledge-based clustering [46], [169].

The problem is formulated on a dataset composed of pairs ($\boldsymbol{x}_k$, $\boldsymbol{y}_k$), k =1, 2, …, N where $\boldsymbol{x}_k \in \mathbb{R}^n$and $y_k \in \mathbb{R}^p$. The data set **X** is endowed with context **F** meaning that with each $\mathbf{x}_k$ a value of membership grade associated with it, denoted by f_k, is given. Subsequently, we form pairs of objects ($\mathbf{x}_k$, f_k) to be involved in the clustering process. The context values f_k have to be taken into consideration when formulating the problem. The objective function assumes the standard form similar to Eq. (2.1):

$$Q = \sum_{i=1}^{c} \sum_{k=1}^{N} u_{ik}^{p} \|x_k - v_i\|^2$$

The added condition requests that the minimization of the objective function must satisfy the following constraint:

$$\sum_{i=1}^{c} u_{ik} = f_k \qquad (2.5)$$

The partition matrix is incorporate the membership grades f_x that guide the clustering process which leads to the computing of the partition matrix realized in the following way:

$$u_{ik} = \frac{f_k}{\sum_{j=1}^{c}\left(\frac{\|x_k - v_i\|}{\|x_k - v_j\|}\right)^{\frac{2}{p-1}}} \tag{2.6}$$

where f_K describes a degree of membership of x_k in the constructed clusters to be associated with or allocated among the computed membership values of x_k.

As mentioned earlier, the value of the fuzzification coefficient p, in CFCM, affects the shape of the membership functions (clusters) being generated. With lower values of p, the membership functions tend to resemble characteristic functions of sets, whereas higher values of p produce spiky membership functions with a profound rippling effect. It is common in the literature to use $p = 2$ [39].

Conditional fuzzy clustering in the design of the logic-oriented fuzzy AND/OR network aims to solve the problem of low efficiency as the dimensionality of the input–output space increases. A cluster in n-dimensional space is described by prototypes v_1, v_2, … ,v_i. The variable x_1 in the input space has the prototypes v_{11}, v_{21}, …, v_{i1} which are calculated using Equation (2.6).

The general architecture of a CFCM-based AND/OR network, context-based FCM, is based on formalization of fuzzy sets, information granules, in the output space followed by generating v clusters (prototypes) in the input space. Since the model proposed in this scheme is inherently structured around the AND/OR network and aimed to produce semantically sound terms with high interpretability, a collection of fuzzy sets $A_1, A_2, \ldots, A_p$, with linguistic description, are positioned in the output space to form a partition of the output space. Each fuzzy set, information granule, is projected onto the input space and reflected through a collection of fuzzy sets. In a nutshell, for each fuzzy set A there is a certain number of clusters formed: v_1, v_2, …, v_p. The main feature of such a model is that the collection of fuzzy sets in input space is induced by the output fuzzy set, which leads to fragmentation of problem space into a series of context-driven clustering problems, greatly reducing computational overhead [41].

Conditional clustering has been employed in a wide range of domains. CFCM was used in the preprocessing phase to design receptive fields of radial basis neural networks as homogeneous clusters of the network [129]. In [130] conditional fuzzy C-means clustering algorithm with spatial information is employed to improve robustness of the segmentation of Magnetic Resonance Images (MRI) and deal with noise and inhomogeneous features. Moreover, the algorithm was also used to construct granular fuzzy models with the aid of logic-oriented networks [131]. Furthermore, CFCM was employed for blind channel equalization to estimate desire states of an unknown digital communication channels. The clustering technique was used because it exhibits shorter processing time than population-based methods and faster convergence rate along with the reliable estimation accuracy in search of optimal output states [119]. The work [132] used fuzzy equalization [133] to guide the granulation of the output space by utilizing triangular fuzzy sets. In subsequent, a logic-oriented fuzzy network was utilized to refine the resultant fuzzy sets and helped to extract useful hidden knowledge of several datasets because of fuzzy neurons that have learning abilities. Finally, the successful implementations of CFCM in different domain show the effectiveness of the modified clustering algorithm. It even helps to reduce the influence of heavy noise sequences in the underlying clustering procedure [119].

2.7 Interpretability and Accuracy of Fuzzy Models

Interpretability, and transparency, allows perceiving qualitative relationships and explicit representation between variables for sufficient explanation of decision-making process linguistically to an observer. Moreover, transparency in the context of intelligent models reveals the inner description along with input-output mapping to eliminate uncertainty and provides in-depth vision to knowledge domain. Several researches have suggested turning our eyes to and focusing on the importance of interpretability rather than relying only on accuracy to achieve high degree of overall model's performance [21], [22], [23]. In order to improve efficiency of an intelligent model, it is important that a proposed model has to improve interpretability while maintaining high learning performance. Subsequently, the model is capable to produces understandable and trusted outcomes to humans. Recently, there is a tendency toward more attention being placed on approximation capabilities being highly glorified and focused upon while the interpretation abilities are being quietly reduced. This tendency may not be surprising at all since the approximation abilities are easier to quantify and eventually easier to realize [80] whereas the interpretability measures are difficult to define as it depends on several factors such as model structure and the number of features.

The three measures of fuzzy models are related and yet contradiction. Several studies use the terms transparency and interpretability interchangeably. However, we believe that these are different terminologies and suggest different directives in the context of logic networks. An intelligent model can be interpretable, with lower accuracy index, without necessarily being transparent even if both features can reveal and extract hidden knowledge. Having high accuracy index is not necessarily an advantage because more details about a system reveal more information at the expense of complexity [79]. Accuracy concerns a higher similarity between the actual and predicted results while Interpretability, which is a subjective property that depends on the designer perspective, concerns the capability of a model to interpret data linguistically to an observer through the transparency characteristic. This specific characteristic, the transparency, gives the design the ability to reveal all inner and yet hidden knowledge. Transparency in the context of intelligent models reveals the input-output mapping along with inner descriptions to provide in-depth vision to knowledge domain. Subsequently, the model is capable to produce understandable and trusted outcomes to humans (Interpretability). Interpretability and transparency are much related to model reliability. Therefore, decision-making processes are trustworthy due to the simplicity in detecting and assisting in identifying the underlying reasons.

The literature of system modelling defines a challenge of meeting the two requirements (accuracy and interpretability) at the highest degree to be almost impossible [14], [25], [134]. Thus, researchers try to achieve the best trade-off between them depending on the user's requirements that are under consideration when constructing intelligent models [135], [136]. The AND\OR architecture has shown supportive experiment evidences on its capability to optimize the three requirements (instead of two requirements) at highest degree due to the underlying features of added components (neurons and NN). The logic networks, with the aid of AND\OR neurons, have the ability to optimize the three criteria (accuracy, interpretability and transparency) and significantly strike the best trade-off with the potential to even improve models' competency further due to the structure nature.

2.8 Summary

In summary, we presented a thoroughly discussion on the design of linguistic models that retain high degree of interpretability along with approximation capabilities. Those types of models are capable of extracting legitimate knowledge out of dataset and eliminate uncertainty and ambiguity due to their underlying logic-oriented structure. The discussion went through each component separately to show its contribution in the interpretation of intelligent models. This chapter aimed to study the literature, along with background knowledge, to develop several novel strategies to design and improve interpretation of knowledge domain in the ensuing chapters, without negatively affecting approximation capabilities. Finally, the literature has shown that architecture of logic-oriented networks, as an essential neurocomputing paradigm, could offer a novel promising processing environment if they are integrated with other fuzzy tools such as fuzzy clustering.

For a better representation of design and optimization solutions, the alternative strategies will be presented in ensuing chapters to build optimal models for knowledge extraction and representation.

3. Neurocomputing in Logic-oriented Fuzzy Networks

Neurocomputing concerns the adaptive information processing systems that develop associations (transformations or mappings) between objects in response to their environment. It is one of the pillars of computational intelligence that could be trained to produce autonomous behavior [45], [46]. Neurocomputing methods imitate the brain as a network of simple interconnected processing elements corresponding to neurons. The collective processing of artificial neurons drives the chief advantages that being exhibited in the learning capabilities along with adaptability to a changing environment. In knowledge-based neurocomputing, the emphasis is on the use and representation of knowledge domain. The reason is that humans find it difficult to interpret the numeric representation of a neural network. The key assumption of knowledge-based neurocomputing is that knowledge is obtainable from, or can be represented by, a neurocomputing system in a form that humans can understand. That is, the knowledge embedded in the neurocomputing system can also be represented in a symbolic or well-structured form, such as logic expressions, fuzzy rules, or other familiar ways.

This chapter discusses the development and optimization of logic-oriented fuzzy neural networks as a significant form of neurocomputing paradigm. These types of networks combine three main components in one simple construct that is transparent, interpretable and accurate. The three components; fuzzy sets (information granules), fuzzy neurons and neural networks constitute the backbone (blueprint) of linguistic, and interpretable, logic-oriented models (Figure 3-1). We will go in details on how all models' components work cohesively to extract and represent knowledge efficiently.

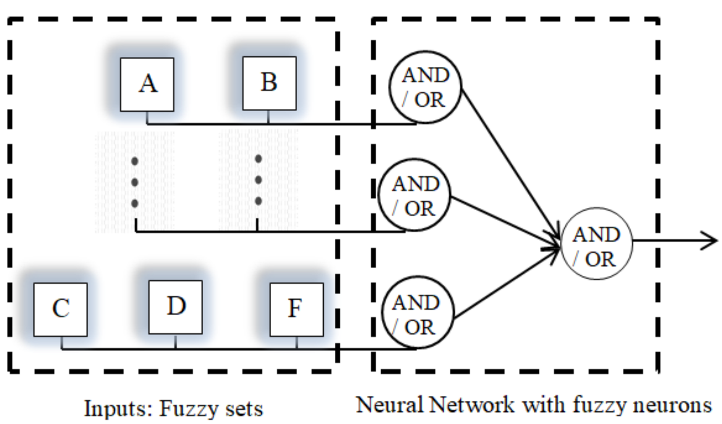


Figure 3-1: The logic oriented fuzzy neural network utilizes three main components for the purpose of constructing an interpretable model.

3.1 Logic-Oriented Neural Networks

Neural networks are equipped with significant learning capabilities to the extent that any continuous function can be approximated to any desired accuracy. NNs are well known mechanism for predictive analysis with exceptional performance on many real-world problems [48]. The network models the operation of an unknown system from a given set of input–output data. In basic systems, where no nonlinearities occur, it is easy to understand the properties of the unknown system through simulation a given set of input–output data and handle them properly. Conversely, if the unknown system is nonlinear and complex, involving a large number of variables, advanced system modeling approaches are necessary. Learning deals with the optimization of the parameters through gradient-based techniques to construct the topology of the network. In some cases in which gradients are not possible to be computed, evolutionary algorithms become the best candidate for parametric learning. Due to the lack of underlying semantics of processing carried out at the level of the individual neuron, there is no any mechanism to examine the character of the produced mapping and investigate it. This brings the lack of confidence to the generalization abilities of the neural networks since the construct is neither interpretable nor transparent.

The architecture of AND\OR networks incorporate a hybrid paradigm of Artificial Neural Networks (ANNs), fuzzy sets, and evolutionary computing to efficiently mimic some aspects of human reasoning [137]. The network is built with the aid of conceptually simple neurons that mimic AND and OR logic operations. Connections between neurons, and the set of inputs-outputs, are easily adjusted by a standard gradient-based learning scheme which adds plasticity feature to the network. The network produces a collection of logic expressions similar to rule-based systems. The logic-oriented AND/OR architecture allows for a deeper analysis and better understanding of nonlinear systems because of its linguistic interpretation and approximation properties being the chief advantages. Thus, the network is efficient in representing data and extract knowledge to capture uncertainty in the data space. Although the construct is simple, it has been applied to plethora of applications efficiently and successfully.

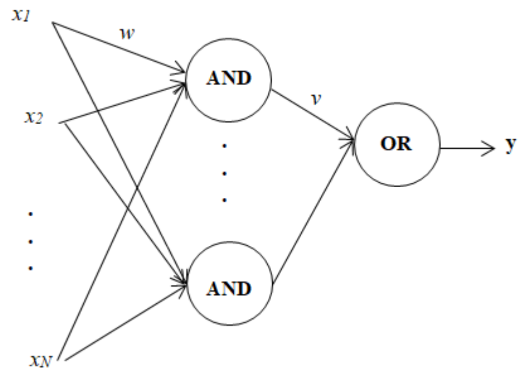


Figure 3-2: A general construct of logic-oriented AND\OR network

Incorporating AND\OR networks in fuzzy modelling utilized as processing core builds a complete qualitative and quantitative interpretable framework (Figure 3-3). Quantitative description concerns the construction of a logic network, with the aid of AND/OR neurons, based on a qualitative description followed by the incorporation of gradient-based learning approach. The learning approach is utilized to quantify the relevance of descriptive variables and interpret the knowledge domain more precisely. The graph in Figure 3-3 is similar to a generic fuzzy model with only difference being the logic network. The interfaces link fuzzy model and logic network with the external world. They convert between numeric to fuzzy sets to be accepted by the model environment. The model may be used in a numeric mode through the transformation of inputs-outputs or in a user-centric mode through the presentation of results in a granular format. The user-centric is more informative and comprehensive than a single numeric quantity [7].

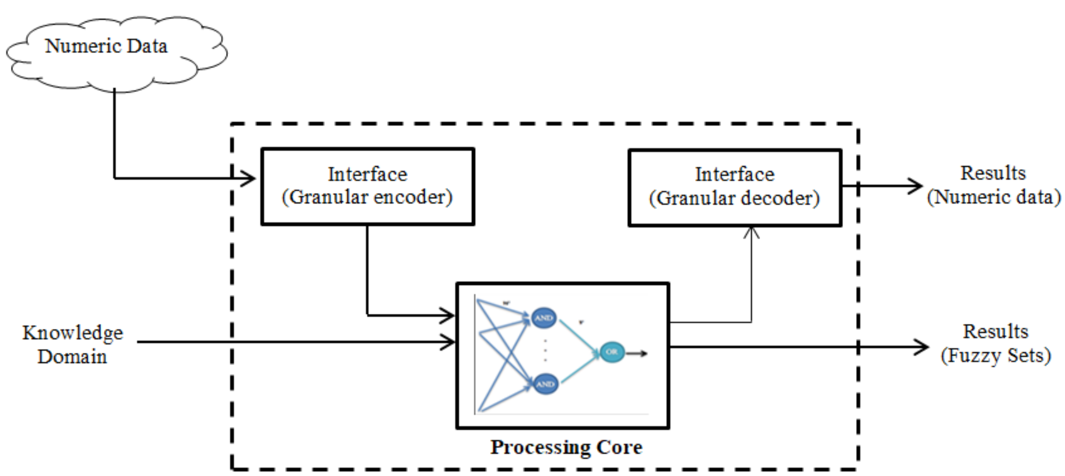


Figure 3-3: A logic-oriented serves as a processing core to build a complete qualitative and quantitative interpretable framework

Designing the AND\OR architecture consists of three main phases namely; a) defining information granules (fuzzy sets) representing semantically sound linguistic landmarks; b) building an initial learning structure that capture the essence of knowledge domain; c) refining the structure by tuning its parameters through parametric learning. Each phase is independent of each other but the efficiency of each phase will be reflected on ensuing phase. Thus, the design process is divided into three sup-processes for optimal results.

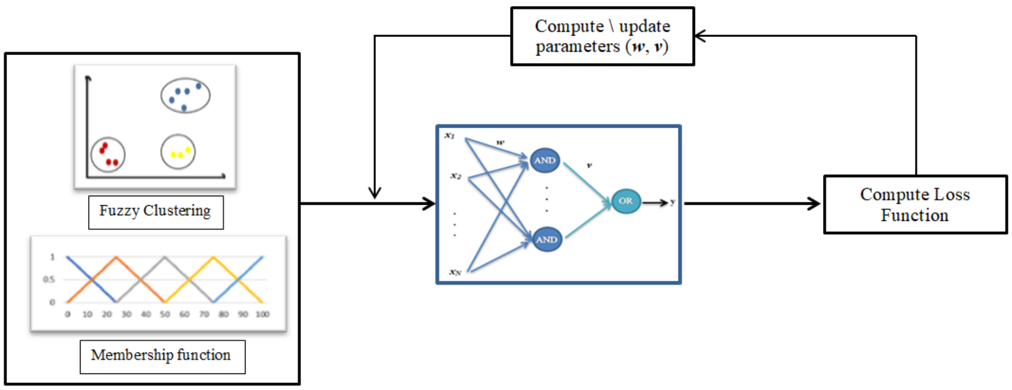


Figure 3-4: The three main phases of constructing logic oriented networks with fuzzy neurons start by defining fuzzy sets followed by building an initial AND\OR structure that equipped with learning capabilities.

3.2 Construction of Information Granules

Construction of information granules (fuzzy sets) is the process to map data in the input–output space (universe of discourse (**X**)) into unit interval [0, 1]. It goes through a membership function to assess data belongingness to a specific fuzzy set. The membership function captures and quantifies the transition that occurs in the description of a problem. Several membership functions can represent a problem in either a continuous or discrete format, which brings the advantage of diversity to fuzzy systems. The diversity of membership functions available for the quantification of information granules is an important operational facet of fuzzy computing. In general, fuzzy sets are constructed in two main ways:

1. **User-driven approach:** fuzzy sets result from a designer's preferences. The designer determines the number of information granules, specificity and coverage for data structure. Subsequently, the results reflect the interest of the user.
2. **Data-Driven approach:** fuzzy sets are determined by a clustering process such as fuzzy clustering or conditional fuzzy clustering, and based on experimental data. Here, the constructed fuzzy sets are experimentally justifiable and legitimate and better in terms of predictive accuracy [146].

Fuzzification (Granulation) of input variables goes through a granular coder that converts numeric inputs coming from the external environment into membership grades of the fuzzy sets or relations. It assign, to each possible value x of the corresponding quantity, a degree of belongingness $A(x)$. However, if the values x_1 and x_2 are close, then the membership degree of these values, $A(x_1)$ and $A(x_2)$, are expected to be close. The resultant fuzzy set generalizes the concept of a single numeric quantity by providing an envelope of possible values it can assume. A membership function defines a fuzzy set A in $\mathbf{X}$ as follows:

$$A(x) : \boldsymbol{X} \rightarrow [0, 1]$$

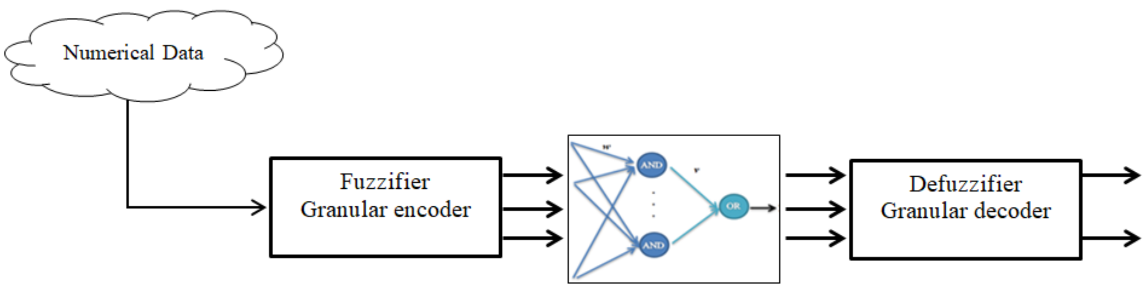


Figure 3-5: Fuzzifier and defuzzifier serve as main interfaces of logic-oriented networks.

Another way to fuzzify the input–output space is by a clustering method such as fuzzy C-means [35], where clusters are represented by their respective centers (centroids). The algorithm partitions data space by calculating the centers of classes and the membership matrix U until an objective function is minimized, leading to the formation of information granules (fuzzy sets) that best describe a system by linguistic terms. Accuracy in the fuzzification of the input–output space results in better extracted logic expressions from the AND/OR network. It clearly states that interpretability of the AND/OR network reflects how data in the input–output space was fuzzified. Additionally, increasing the dimensionality of the input–output space by fuzzification is a reason for the low learning efficiency of the AND/OR network. Therefore, it is better to reduce the dimensionality of the problem first by approximating the relationship between a specific fuzzy set from the output space to the corresponding fuzzy sets in the input space before learning of parameters starts through a clustering mechanism. In Figure 3-6, as dimensionality of the input–output space increases, the number of parameters that must be refined ($\boldsymbol{w}$, $\boldsymbol{v}$) increase as well leading to increasing in model complexity. As complexity increases, and nonlinearity as well, optimization is necessary to handle the AND/OR network.

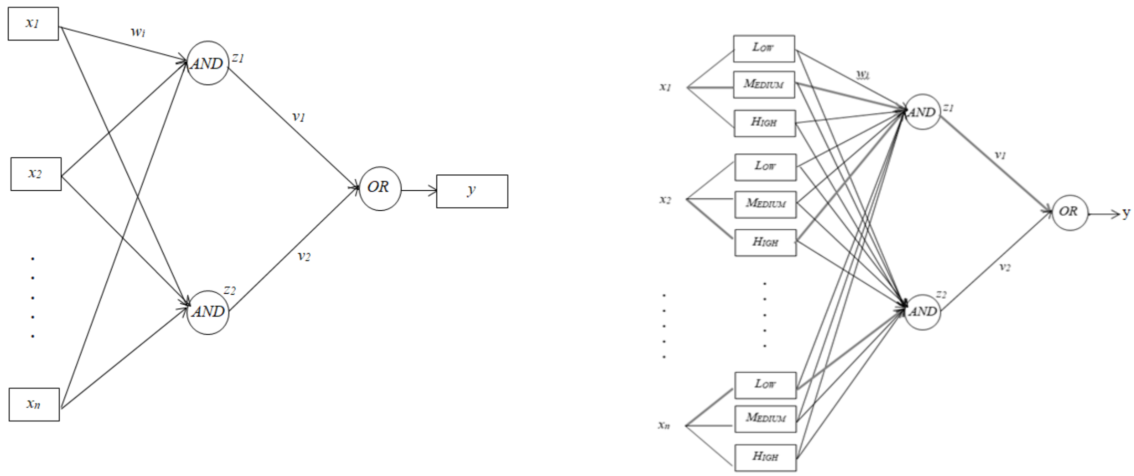


Figure 3-6: AND/OR Network (Left) numeric network (Right) Fuzzified input/output space.

Commonly, expert knowledge is the source to determine and assign linguistic terms (such as *low*, *med*, *hi*) to some interval to build a linguistic model. A family of fuzzy sets defined in same space forms a structure of linguistic terms of some variables (say; temperature). Furthermore, each fuzzy set might be described by a set of fuzzy sets such as (*Med-low*, *High-low*) in same space.

In this work, we rely use both data-driven and user-driven approaches to form information granules. We rely mainly on fuzzy clustering to construct linguistic landmarks. We also used triangular membership function to formalize fuzzy sets of each variable to test and optimize the AND/OR network.

3.3 Structure Design and Discovery

AND\OR architecture differs from other neurocomputing systems by the incorporation of the fuzzy logic-driven AND and OR neurons into the neural networks in a simple construct. Hence, the whole architecture offers simplicity, plasticity along with a useful balance in quantifying accuracy side to side with the interpretation capabilities. The processing scheme of the AND\OR network is augmented by the incorporation of fuzzy neurons into one structure. Each component of network, namely; fuzzy sets, fuzzy neurons and neural networks, plays a unique role in the final output of the networks. Therefore, designing and constructing each component goes through individual process to optimally quantify how well the original data are represented. Building the network at the level of information granules (fuzzy sets) makes the construct more reflective of reality and subsequently the network provides superior understanding of the data under observation.

We will elaborate on the main categories of the fuzzy neurons and their main properties, and then we will move on to the architectures of networks composed of such neurons, and then discuss various facets of network interpretation.

3.3.1 Logic-Oriented Fuzzy Neurons

Fuzzy neurons [8], [24], [139], [148] are endowed with the clearly delineated logic structure. The classes of Logic-based neurons are either (a) aggregative logic neurons which realize OR and AND operations (Figure 3-7), b) the referential logic-based neurons which analyze input signals with respect to a given reference point (through a matching, inclusion, difference or dominance relationship) [31], [46] (Figure 3-8). The neurons, in both classes, solve the lack of underlying semantics of individual neuron processing in neural networks. Consequently, one can directly examine the character of the produced results and investigate it with regard to the data at hand. Consequently, it increases our confidence in the generalization abilities of the network because it alleviates the limitation black-box nature of neural networks by revealing the inner description along with input-output mapping to provide in-depth vision to knowledge domain. Several evident benefits can be carried out by the neurons; first, easy interpretation of results and providing highly description of data. Second, learning mechanism is highly facilitated [46]. Therefore, the fuzzy logic-driven neurons address the issues of interpretability\transparency very efficiently which in result can be translated into a collection of textual semantics that take logic statements form.

Computationally, the use of *t-norm* and *t-conorm* inherit nonlinearity and plasticity characteristics to the AND\OR architecture. The incorporation of this nonlinearity changes the numerical characteristics of the neuron, but its essential logic behavior is maintained. The nonlinearity transformations add interpretability to the architecture because they can be treated as models of linguistic modifiers [79], [138]. The potential plasticity of the neurons becomes critical when incorporating a learning scheme to the networks. Several realization of the logic operations exist with several drawbacks that have to be taken into consideration. When we say "realization of logic operations", we mean the mathematical models translation of AND\OR operations. Those realizations affect the final results as well as the model interpretation by adding interactivity of both neuron operands. In some realization, in particular min\max, the results only reflect the influence of one operand and obviously the true interpretability of such model will not be reflected. Furthermore, the high nonlinearity processing coming from the logic neurons is dependent on the selection of the *t-norms* and *t-conorm*. These forms of triangular norms (*t-norms* and *t-conorm*) are a well-defined robust fuzzy mathematical framework and they meet the minimal set of properties of logic connections which consists of commutativity, associativity, monotonicity and some boundary conditions [69], [105].

A. Aggregative Neurons

The aggregative neurons realize a logic mapping from $[0,1]^n$ to $[0,1]$. The **OR Neuron** realizes and logic operation on inputs x and the corresponding weights w. the computation is realized as:

$$y = \mathrm{OR}(\mathbf{x}; \mathbf{w}) = S_{i=1}^{n}(x_i \; t \; w_i) \qquad (3.1)$$

The logic operator here is modeled via *s*-norm (*t*-conorm) where the typical examples of this particular model are maximum, probabilistic sum and Lukasiewicz or connective. The formula retains a higher degree of flexibility for better interpretability in the following manner:

$$y = (x_1 \; and \; w_1) \; or \; (x_2 \; and \; w_2) \; or \; \dots \; or \; (x_n \; and \; w_n)$$

The weight of inputs impacts the final results by the following relations; lower value of w_i lowers the impact of the corresponding input x_i which implies if the value of a connection equals zero then the corresponding inputs are eliminated.

On the other hand, the **AND Neuron** exhibits converse characteristics from what we observed for the OR neuron. Here, the computation is governed by:

$$y = AND(\boldsymbol{x}; \boldsymbol{w}) = T_{i=1}^{n}(x_i \; s \; w_i) \qquad (3.2)$$

The logic operator presented here is modeled via (*t*-norm) where the typical examples of this model are minimum, product, and Lukasiewicz connective. This formula also preserves higher degree of flexibility and can be interpreted in the following manner:

$$y = (x_1 \; or \; w_1) \; and \; (x_2 \; or \; w_2) \; and \; \dots \; and \; (x_n \; or \; w_n)$$

The final results are impacted by the value of inputs' weights similar to OR neuron. However, the relationship between input and weight is different here as higher value of w_i lowers the impact of the corresponding input x_i which implies that if the value of a connection equals one then the corresponding inputs are eliminated.

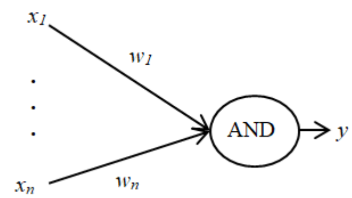


Figure 3-7: An aggregative AND logic Neuron

B. Referential Neurons

This type of neurons deals with processing logic predicates [28, 44]. Predicate logic allows solving the problem of multiple generality due to its nature to analyze a problem (or a statement) into subject and argument in different ways. The computation in this class is first carried out with respect to some given reference point then the results of processing are aggregated in a similar way as encountered in aggregative neurons discussed in the previous sections. The reference neurons are described as follows:

$$y = \mathrm{OR}(\mathrm{REF}(\mathbf{x}; \textbf{reference point}), \mathbf{w}) \qquad (3.3)$$

$$y = \mathrm{AND}(\mathrm{REF}(\mathbf{x}; \textbf{reference point}), \mathbf{w}) \qquad (3.4)$$

The expression *P(x, a)* evaluates the truth value of the predicate (*P*) in which "*a*" is the reference point that the input is analyzed against. There are several analysis mechanisms to compare an input x_i to a reference point to show the degree of satisfaction of the truth value of the predicate (*P*). The four analysis mechanisms are; matching, inclusion, difference and dominance. The expression is translated with regard to the implemented analysis as "*x* is matched to *a*," "*x* is included in *a*" and so on. Therefore, the meaning (interpretation) of an expression depends on the chosen analysis mechanism. Pedrycz in [21], [28], [44] has presented full analysis of referential neurons and discussed its potential in fuzzy systems.

It is worth mentioning that, the impacts of inputs' weights on final results are exactly similar to those of aggregative neurons. These types of neurons reveal specific knowledge about a certain reference which adds more specificity to the network.

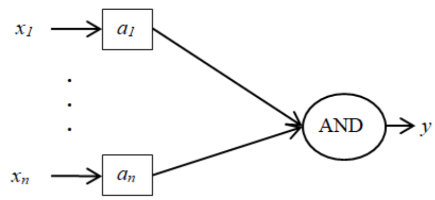


Figure 3-8: A referential AND logic-based neurons. a_1 and a_n serve as reference points in analyzing input data

3.3.2 Logic-Oriented AND\OR Networks

The network, in its basic architecture, consists of logic-driven AND/OR neurons placed in series in the hidden and output layers of a three layered neural network. It combines the learning abilities of neural networks with interpretation capabilities that are the consequences of utilizing AND/OR neurons. The neurons constitute the main computational structure of the logic-driven fuzzy neural networks by modeling the logic operators AND and OR with *t-norms* and *t-conorms* fuzzy set operators respectively [77], [149]. Such fuzzy neurons are also interesting generalizations of the basic structures of logic gates encountered in digital system design. They help in representing data in logical expressions form and extract knowledge through a learning approach within logic networks. These types of networks are capable of being modularized, debugged, and refined because of their construction nature. Those features become substantially important as the complexity of system modeling increases due to increase in nonlinearity. Furthermore, logic networks offer an interesting feasibility to linguistically interpret intelligent models without sacrificing the accuracy which leads to support decision-making process and hence decisions are justifiable and legitimate. Several other features of logic networks, such as adaptability for different systems and predicative properties, have yielded to successful integration with other algorithms in different domains [80], [86], [89], [150].

The logic processing of the logic networks comes from the famous Shannon theorem of decomposition of Boolean functions [151] in which any Boolean function can be represented as sum of min-terms or product of max-terms. The decomposition theorem implies that there are two modes of logic networks which directly affect the construction of the neural networks. In the first construct, a two-layer neural network consists of AND neurons (similar to logic AND gate) in the first layer whose outputs are aggregated by a single OR neuron (gate) (Figure 3-9). The opposite construct occurs for the decomposition mode where OR neurons are placed in the first layer and a single AND neuron aggregates the outputs then produces the final results (Figure 3-10). One can look the two modes of aggregation as a way that gives the architecture a positive flexibility characteristic in knowledge representation. Here, ambiguity and uncertainty in datasets are reduced, and might be eliminated, through a collection of composite rules (referred to as logical expressions).

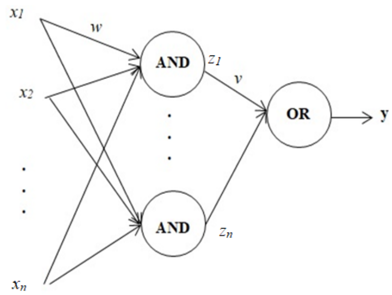


Figure 3-9: The architecture of product of max-term. The decomposition of the first layer is translated as x_1 AND x_2 … AND x_n while the output layer is interpreted as z_1 OR z_n

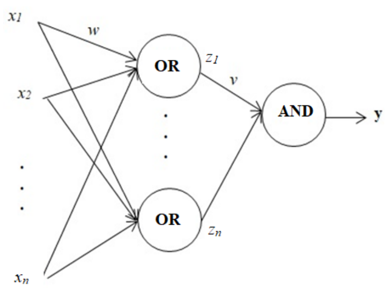


Figure 3-10: The architecture of sum of min-terms. The decomposition of the first layer is translated as x_1 OR x_2 … OR x_n while the output layer is interpreted as z_1 AND z_n

3.3.3 Gradient-based Learning Schemes

Deep learning (DL) [152] in the context of neural networks indicates the multiple processing layers for predictive analysis with multiple level of abstraction. When NNs is designed with AND\OR neurons followed by an implemented DL mechanism, we end up with a complete AND\OR architecture. AND\OR architecture is trained in supervised learning fashion in which weights of connections are refined iteratively by an optimizer [79]. It reserves it's distinguish logic processing feature that offers high degree of parametric flexibility. However, this type of learning is usually slow, especially if the predictive model is updated after all observations have been evaluated, which may result in a local minimum of the performance index. The efficiency of an optimizer is determined by two factors: the speed of convergence and a way on how the trained model performs on new data.

In a standard learning scenario, realized as supervised learning, of AND\OR architecture, a collection of experimental data organized into a family of input-output pairs as the following:

$$\{\mathbf{x}\,(k), target(k)\} \qquad k = 1, 2, \ldots, N$$

A generic AND/OR network comes equipped with the connections that could be easily adjusted through a gradient-based learning. Connections optimization modifies the network's parameters (connection weights) to minimize the performance index Q_N through iterative process that could be schematically described in the form:

$$\boldsymbol{conn}\,(iter + 1) = \boldsymbol{conn}(iter) - \alpha\, \nabla_{\boldsymbol{conn}(iter)}\, Q \qquad (3.5)$$

where $\nabla_{\boldsymbol{conn}(iter)}\, Q$stands for the gradient of Q determined with respect to the connections, α is the positive learning rate. Here, it is necessary to find an optimal learning rate since small values result in slow convergence, whereas large values prevent convergence, making the loss function fluctuate at the minimum.

The learning is carried out in a supervised mode with input-output data. After specifying the form on *t-norm* and *t-conrom* to implement a fuzzy function multi inputs and a single output, the performance index is expressed as a sum of squared error:

$$Q = \sum_{k=1}^{N} (y_k - \, target_k)^2 \qquad (3.6)$$

where y_k is the output of the network and $\hat{y}$ is the target output and N is number of instances.

Updating of connection values are made after the presentation of each pair of input–output data through back-propagation technique which is responsible for the slow convergence, especially when the network is large (i.e., the number of neurons is high). Furthermore, global minimum is not feasible and it is most likely to end up with a local minimum due to gradient-based mechanism. When specifying the form of *t-norm* and *t-conorm* as min and max operators, the final do not realistically reflect the arguments of t-norms and s-norms.

In this work, we assessed several gradient-based optimizers namely; ADAM, AdaMax [121], and AMSGRAD [122]. Although the optimizers share the same idea of updating weights in a back-propagation step, they have shown better convergence rate when implemented on different intelligent models. However, there is a lack of available data of how efficient those optimizers are when implemented to optimize AND/OR networks due to the fact that mathematical operations of AND/OR neurons differ from ordinary neural networks. To close this gap, we evaluate the performance of the AND\OR construct on different learning schemes. These are as follows:

- **ADAM**: an adaptive learning rate optimization method that computes individual learning rates for different parameters and uses moving average of the gradient instead of the gradient itself. The learning scheme is governed by:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \qquad (3.7)$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \qquad (3.8)$$

where m_t is the exponential moving average of the gradient, v_t is the squared gradient which is the estimates of the 1st and 2nd moment, β_1 and β_2 control the exponential decay rate and they equal 0.9 and 0.999 respectively. However, these parameters need to be tuned for each specific application and don't rely on the default parameters.

$$\widehat{m_t} = \frac{m_t}{1-\ \beta_1^t} \qquad (3.9)$$

$$\widehat{v_t} = \frac{v_t}{1-\ \beta_2^t} \qquad (3.10)$$

where $\widehat{v_t}$and $\widehat{m_t}$ are biased corrected estimates of the 1st and 2nd moment respectively. Weights of connectives are updated as follows:

$$w_{t+1} = w_t - \frac{a\,\widehat{m_t}}{\sqrt{\widehat{v_t} +\ \varepsilon}} \qquad (3.11)$$

where α is the positive learning rate, ε is fixed value equal 10^{-8}

- **AdaMax**: an improved version of the Adam optimizer that solves stability problems that occur in special cases when training some models. Both algorithms share similar parameters and the learning goes as following:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \qquad (3.12)$$

$$\widehat{m_t} = \frac{m_t}{1-\ \beta_1^t} \qquad (3.13)$$

$$u_t = \max(\beta_2 u_{t-1}\,, |g_t|) \qquad (3.14)$$

$$w_{t+1} = w_t - \frac{a}{u_t}\widehat{m}_t \qquad (3.15)$$

- **AMSGRAD**: uses the maximum of the past squared gradient rather than the exponential average to update the parameters. This prevents the poor generalization behavior of adaptive learning rate methods caused by the exponential moving average of the past squared gradient.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \qquad (3.16)$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \qquad (3.17)$$

$$\hat{v_t} = \ max\ (v_t, \widehat{v_{t-1}}) \qquad (3.18)$$

$$w_{t+1} = w_t - \frac{a\ m}{\sqrt{\hat{v_t}} = \varepsilon} \qquad (3.19)$$

Table 3-1: Performance of gradient based algorithms on AND\OR Network

Optimizer	Neuron = 2 (*Q*) Training \ Testing	Neuron = 3 (*Q*) Training \ Testing	Neuron = 4 (*Q*) Training \ Testing
MiniBatch	0.0232 \ 0.0301	0.082 \ 0.13	0.062 \ 0.036
ADAM	0.0323\ 0.039	0.072 \ 0.097	0.049 \ 0.048
AdaMax	0.037 \ 0.042	0.073 \ 0.099	0.045 \ 0.045
AMSGrad	0.030 \ 0.036	0.028 \ 0.047	0.029 \ 0.028

As Table 3-1 suggests, we used AMSGrad as a learning optimizer when training AND\OR networks. It is because this optimizer improves the convergence by avoiding large abrupt changes in the learning rate for each input variable. AMSGrad solves the limitation of gradient decent which is the hyperparameter *step size* that is used for all inputs variables whereas AMSGrad uses a separate step size (learning rate) for each parameter in the optimization problem. We re-stress the issue of very slow learning that may occur especially when the size of the network gets large. This problem emphasizes having preliminary knowledge about the structure of the network in what is called structure discovery process [81].

3.4 Interpretation of Logic-Oriented Fuzzy Networks

Knowledge extraction starts once the performance index is minimized and connection values are optimized. The extracted understandable terms that reflect a model's nature are, ideally, in the form of logical expressions similar to Sum of Minterm (SOP) or Product of Maxterm (POS) that are found in canonical form in Boolean algebra. The extracted knowledge exhibit similar feature of Fuzzy rules-based systems (FRBS) and reflect the logic (Boolean) nature of the neurons. The following logic expressions follow the network construct depicted in (Figure 3-9):

Conclusion$_k$ =

condition$_1$ AND condition$_2$ AND . . . condition$_L$

OR

condition$_{1'}$ AND condition$_{2'}$ AND . . . condition$_{L'}$

Rules can be represented in a different scheme if we rely on a different network construct such as in (Figure 3-10):

Conclusion $_k$ =

condition$_1$ OR condition$_2$ OR . . . condition $_L$

AND

condition$_{1'}$ OR condition$_{2'}$ OR . . . condition $_{L'}$

The representation of extracted knowledge (as in the above examples) shows substantial advantages to an observer or a designer since they usually do not require explicit explanations. The neurons along with a learning mechanism play a pivotal role to highlight the significance (or relevance) of the key elements of knowledge domain through the adjustment of connectives values and in consequence, the explanation is constructed accordingly. The well-structured semantic mapping between the input-output data of constructed networks allows for easy translation and extraction of the semantic meaning of the acquired knowledge. Therefore, the resultant networks are capable to recognize complicated patterns and provide quantitative decision support since essential hidden linkages between input-output pairs are revealed.

Table 3-2: Taxonomy of the three contradictory requirements of designing logic-oriented fuzzy architecture

		Accuracy, Interpretability and Transparency
1	**Active Structure**	• Structure is actively changing when it is under learning for better Interpretability. • Learning can take different approaches; supervised learning or evolutionary computing • Transparency alters as a result of structure modification.
2	**Explanation**	• Transparency is revealed as a result of hidden neurons. • Interpretation is presented in terms of knowledge extraction (logic expressions or *if-then* rules).
3	**Interpretability**	• Local: Interpretation of individual prediction. • Global: Interpretation of a model as a whole in terms of fuzzy rules or logic expressions

The taxonomy in Table 3-2 is meant to explore how the three criteria are cohesively tied together in an integrative process. It facilitates the three contradictory requirements and tries to integrate them to optimally construct the network [21]. Furthermore, it shows the AND\OR architecture is being actively modified when it's under learning. In this case, low accuracy is not of vital problematic as the interpretability increases at the expense of accuracy. Quality of interpretation should be discussed with the subject of information granules. It is because specificity and coverage measures affect quality of data blocks since the more specific (more descriptive) the information granules are, the precise interpretability is revealed (Figure 3-11). However, that might not be needed in some cases in which the degree of knowledge is not highly specific. Therefore, a sound balance between specificity and coverage is desired. That leads us to the significance of the formation of information granules (fuzzy sets) in the first place and even before network construction. On the other hand, transparency is discussed away from data entities leading to focusing on the algorithm that being used or the structure itself.

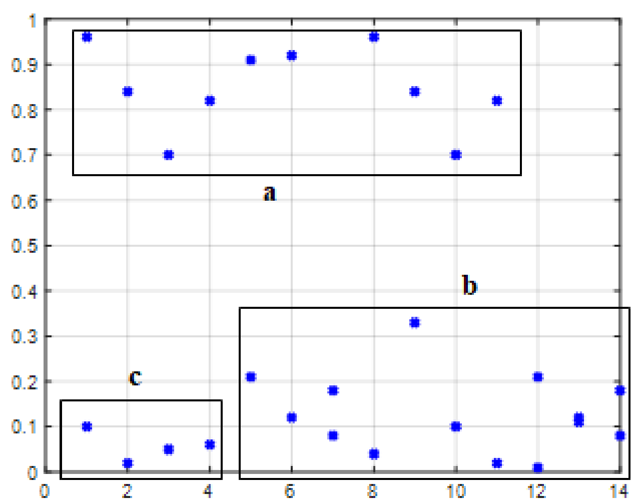


Figure 3-11: Specificity and coverage measures of information granules in 2D output space; (a, b) low specificity and high coverage, (c) high specificity and sound semantic.

The AND\OR architecture not only interprets a dataset as whole and extracts knowledge; it is endowed with the ability to focus on certain information granules in the output space (referential information granules) to extract focused knowledge about a specific concept. This could be utilized with the aid of fuzzy clustering [4], [129], [132] to add better interpretation capabilities. Furthermore, interpretation of individual prediction adds traceability feature to help identify the cause of a certain decision. This type of architecture is able, to some extent, to learn the cause-and-effect structure of a decision that has been made and subsequently could improve the reliability and trustworthiness of machine learning. The mutual advantages of different components, e.g. NNs, neurons, fuzzy sets, … etc., help to achieve high level of overall performance.

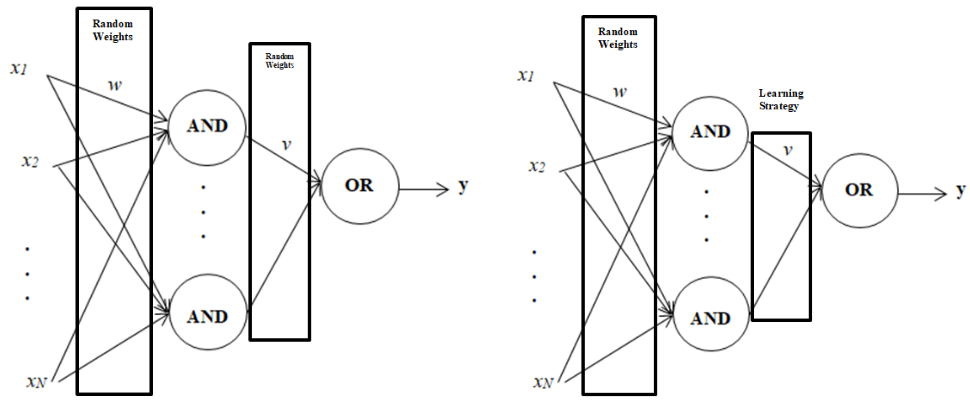
We should emphasize here that logic-oriented fuzzy neural networks could be categorized under the umbrella of Responsible AI by prioritizing and addressing uncertainty, interpretability and fairness in decision-making. Responsible AI framework is essential to evaluate fairness and react on unfair predictions for the benefit of human beings. AI algorithms tend to be biased, which is reflected in deteriorated fairness, since the data considered in the construction of the models are biased as well. More specifically, a prediction model may actually be inherently biased since it learns and preserves historical biases [91]. The architecture under review quantifies the credibility of intelligent models and contributes to the formation of a sound trade-off between accuracy and interpretability. Thus, decision making processes are justified which will eliminate bias. Therefore, fairness of models' outcomes is positively impacted.

3.5 Design of Two-phase Logic–Oriented Fuzzy Networks

The development of two-phase AND\OR networks is meant to close the performance gap of approximation capabilities between fully deep random AND/OR networks and deep AND/OR networks. The gap is caused by high computation overhead of back-propagation algorithm. Rather than relying fully on gradient-based learning, feedforward neural networks are efficient in approximating any continuous function to a desired degree of accuracy which makes feedforward networks suitable to model nonlinear fuzzy systems especially when the system is hard to be mathematically described.

3.5.1 Feedforward Neural Networks

The speed of learning is one of the biggest advantages of feedforward neural networks along with good nonlinear approximation capabilities [84]. Additionally, simplicity of implementation makes the networks attractive to be dominant in leaning scheme of AND\OR networks. Feedforward AND/OR networks exhibit the behavior of neural systems where the majority of connections are randomized or more generally fixed according to some specific heuristic, either in a stochastic or a deterministic fashion [153]. To overcome the drawback of gradient-based learning algorithms such as local minima, slow convergence, and poor sensitivity to learning setting, it would be practical to turn to randomized algorithms to train the AND/OR networks.



(A) (B)

Figure 3-12: A simple model for feedforward AND\OR network, (A) Pure random. (B) Semi random

In this random scheme of learning algorithms, the weights feeding into the hidden layer can be randomly assigned with no need to be tuned. Then, the weights connected to the output layer are calculated by solving a simple linear regression problem (Figure 3-12 (A)) [154]. In a simpler form, we could rely on fully random weight generator for both layers (Figure 3-12 (B)). However, the learning will certainly suffer local minimum problem. The benefit of relying on random training is that it will outperform gradient-based methods in terms of speed of convergence which is a significant advantage in AND\OR networks. It is therefore a preferable way to be able to increase the size of knowledge domain since time of convergence is reduced.

3.5.2 Development of Two-phase Network

In the two-phase AND\OR network, two duplicated AND\OR models where the first model is trained with feedforward neural network whereas the second model is trained with a gradient-based method (Figure 3-13). Here, AMSGrad optimizer is used to train the second model because of its faster convergence due to adaptive learning mechanism. The benefit of this technique is that the first model will optimize the weight of connections in a short time since there is no backpropagation step. Afterwards, the second model will start from the best weights values (***w, v***) from the previous model to find a better solution "better local minimum" with a possibility to arrive to a global minimum. In nutshell, the objective the first phase is to find a better starting point in relatively short time to efficiently train a model whereas the objective of the second phase is to further refine weight values to better optimize the network. Figure 3-14 explains how the second model will start from near optimal weight values to meet these objectives.

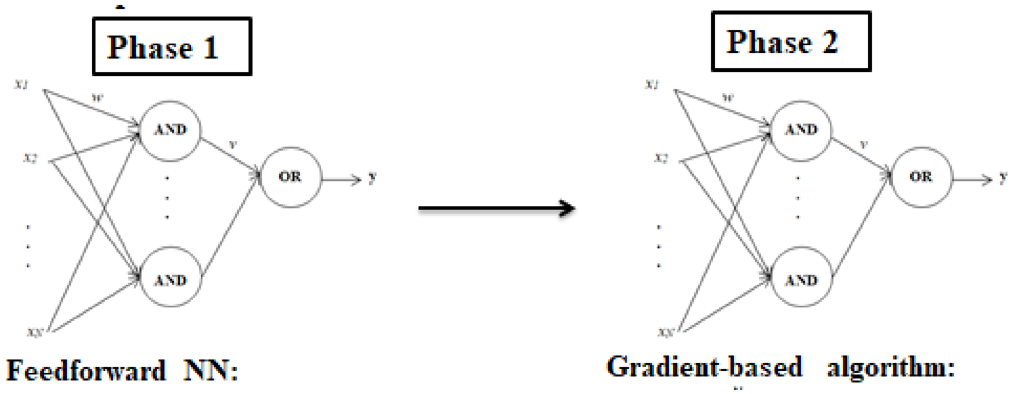


Figure 3-13: The design consists of two duplicated AND\OR networks.

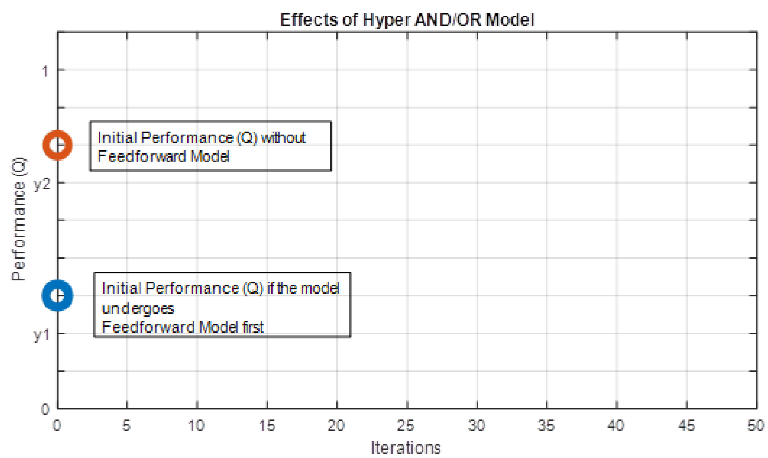


Figure 3-14: Effects of implementing tow-phase AND\OR model

As stated before, the two-phase AND/OR network starts with randomly generated weight values for all connections in all network's layers for n number of iterations in Feedforward mechanism with only using Equations (3.1, 3.2) for OR neuron and AND neuron respectively. A targeted performance index (Q_{taeget}) can be either pre-defined based on a user preferences or it can be left undefined to let the model complete the maximum number of iterations (max (n)). The benefit of setting up the values of (Q_{taeget}) is that the number of iteration n can be shortened if the targeted performance is reached before max(n) which results in reducing the training time. In the other hand, specifying the value of (Q_{taeget}) might limit the efficiency of the model since there is no guarantee that the obtained performance value is optimized. Therefore, it is better to leave (Q_{taeget}) undetermined and let the model completes the feedforward step for (max (n)).

In the second phase, new iteration value (m) is introduced. Since the computation overhead increases rapidly due to calculation associated with gradient-based algorithm, number of iteration

(m) must be significantly lower than n to make the presented model optimal. In the development process, the following condition applies:

$$\max(m) \leq \frac{1}{2}n \qquad (3.20)$$

Since the second model start form a better known initial performance index, the developed model's performance will meet two objectives; the performance will either and outperform a stand-alone gradient-based model or it will reach a stable state faster where no further improvements can be gained. Both objectives are reachable, and within boundaries, if the first phase successfully finds ideal weights values.

3.5.3 Experimental Studies

The experiments have been conducted on both numeric and fuzzy datasets of three datasets, (See Appendix A). Results are organized in two sub-sections to clearly report the model effectiveness in all cases. To illustrate the model effectiveness, AND/OR networks were constructed with different size of hidden layer (Figure 3-12 (A)). Number of AND neurons in the hidden layer equals 2 and 4 whereas number of OR neuron in the output layer always equals 1. The OR neuron aggregates the outputs of AND neurons, in the hidden layer, to produce the final logic expression (such as sum of products in Boolean algebra). The quality of learning is measured by monitoring the value of the performance index. The fuzzy quantization involves three fuzzy sets for each variable which results increase in input-output dimensionality. We assure that the intent of this study it to investigate the performance index when implementing the two-phase model. Optimization of fuzzy sets will certainly lead to refine the extracted fuzzy rules.

A. Numerical Experiments

The results of developed model are presented for each phase individually to show how improvements are obtained. Figure 3-15 shows results of both phases and compare them with stand-alone gradient-based algorithm. In phase one, number of iteration n equals 1000 and it consumed 15% of training time. While in phase two, number of iteration m equals 500 and it consumed 85% of training time. It is worth mentioning that number of iteration, n and m, is a problem specific parameter and it needs to be re-adjusted for other applications. In comparison between two phase model and stand-alone gradient-based algorithm in training AND/OR network, there is 15% increase in training time. However, there is 16%

improvement in performance index (Q). The increase in training time of two-phase model comes from phase one which can be reduced by reducing the number of iteration. Figures 3-16, 3-17, 3-18, 3-19, 3-20, 3-21, 3-22, 3-23 and 3-24 show refined results of both models.

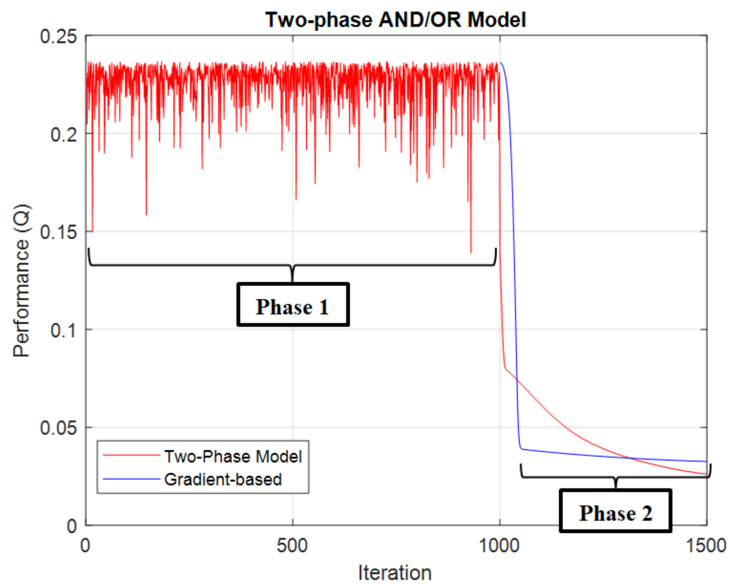


Figure 3-15: Demonstration of (Red) two-phase model, (Blue) Gradient-based algorithm. (Blue) line is offset by 1000 to clearly show the comparison.

All experiments have been conducted with similar settings, in terms of number of iterations and number of AND/OR neurons, on three different datasets. Results have shown that the phase one, which randomly set the values of connections weights for n number of iteration, is effective in reaching better performance index. The Tables 3-3, 3-4 and 3-5 show clearly the improvement in performance index when implementing the proposed model. The increase in training time between 15% to 20% is noticed as results of the first phase.

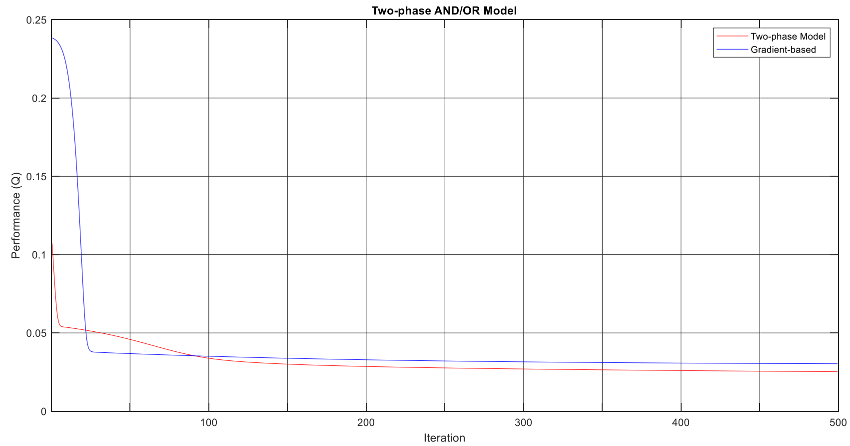


Figure 3-16: Improvements in performance index (Red) two-phase model (Blue) gradient-based. AND/OR network with two AND neurons (numeric Boston housing dataset)

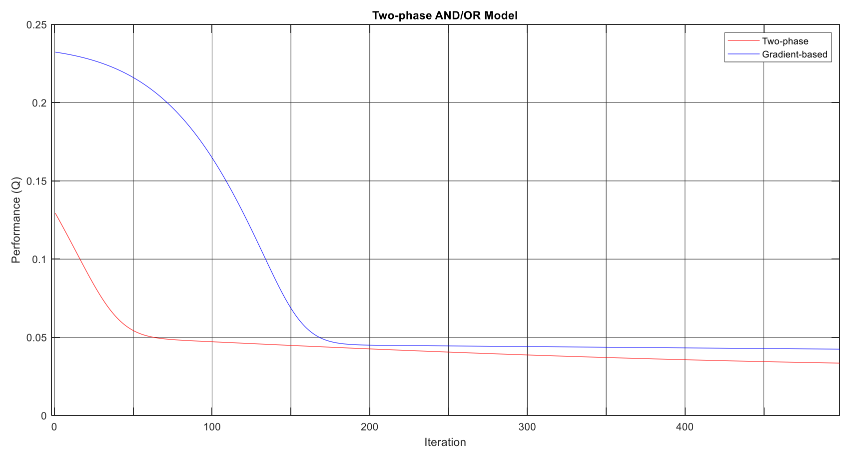


Figure 3-17: Improvements in performance index (Red) two-phase model (Blue) gradient-based. AND/OR network with four AND neurons. (Numeric Boston housing dataset)

Table 3-3: Results of training/ testing AND/OR network (numeric Boston housing dataset)

Method	**Neuron = 2** (*Q*) **Training\Testing**	**Neuron = 4** (*Q*) **Training\Testing**
Two-phase Model	0.0252\0.028	0.0335\0.0409
Gradient-based	0.0303\0.0356	0.0424\0.0425

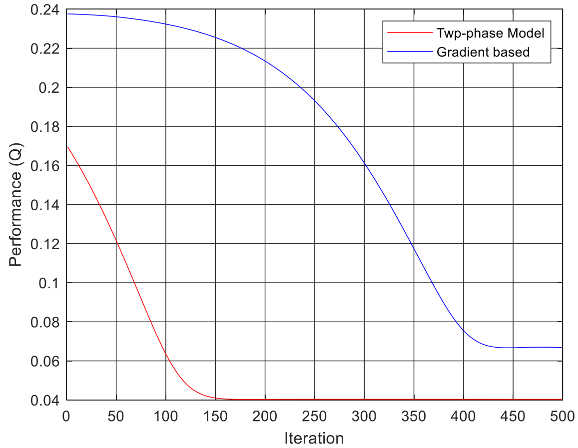


Figure 3-18: Improvements in performance index (Red) two-phase model (Blue) gradient-based. AND/OR network with two AND neurons. (Numeric Diabetes dataset)

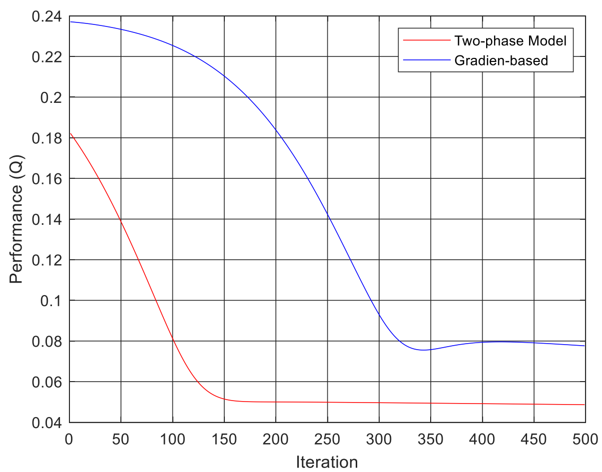


Figure 3-19: Improvements in performance index (Red) two-phase model (Blue) gradient-based. AND/OR network with four AND neurons. (Numeric Diabetes dataset)

Table 3-4: Results of training/ testing AND/OR network (numeric Diabetes dataset)

Method	**Neuron = 2** (*Q*) **Training\Testing**	**Neuron = 4** (*Q*) **Training\Testing**
Two-phase Model	0.0402 / 0.05023	0.0487 / 0.0536
Gradient-based	0.0618 /0.07	0.0755 / 0.0812

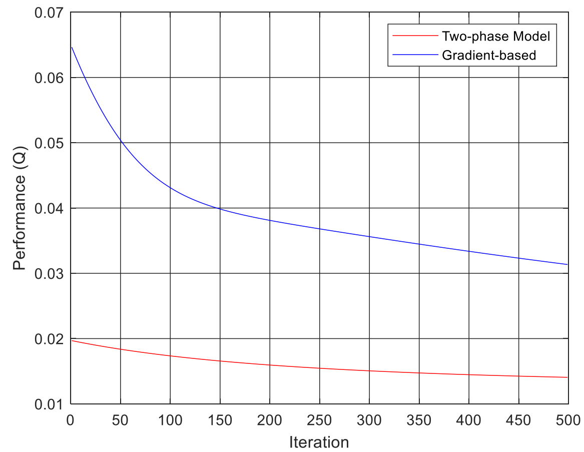


Figure 3-20: Improvements in performance index (Red) two-phase model (Blue) gradient-based. Both models are AND/OR network with two AND neurons. (Numeric Iris plants dataset)

Table 3-5: Results of training/ testing AND/OR network (Numeric Iris plants dataset)

Method	**Neuron = 2** **(*Q*) Training\Testing**
Two-phase Model	0.0141/ 0.0392
Gradient-based	0.03134 / 0.1073

B. Fuzzy-set based Experiments

As pointed out earlier, we rely on experts to assign, to each possible value x of the corresponding quantity, a degree of belongingness $A(x)$. Triangular membership function (Equation 3.8) is used to formalize fuzzy sets of each variable in the Boston housing dataset and Diabetes dataset to test and optimize the AND/OR network. Linguistic landmarks (*High*, *Med*, *Low*) were given to each fuzzy set. The membership function of a triangular fuzzy set is given by the following way:

$$A(x\colon a,m,b) = \begin{cases} 0, & x \le 0 \\ \dfrac{x-a}{m-a}, & x \in [a,m] \\ 1-\dfrac{b-x}{b-m}, & x \in [m,b] \\ 0, & x \ge b \end{cases} \tag{3.8}$$

In fuzzy-set based AND/OR experiments, we use similar settings from numeric experiments for the purpose of consistency. The number of iterations in phase one (n) equals 1000 whereas the number of iterations in the second phase (m) equals 500. We obtained ~ 17% improvement in terms of the performance index (Q) in many cases. Tables 4 and 5 show the advantage of the proposed method quantified by the performance index.

Table 3-6: Results of training/ testing rule-based AND/OR network (Boston hosing dataset)

Method	**Neuron = 2** **(Q) Training\Testing**	**Neuron = 4** **(Q) Training\Testing**
Two-phase Model	0.041\0.0754	0.0365\0.0680
Gradient-based	0.048\0.0999	0.043\0.045

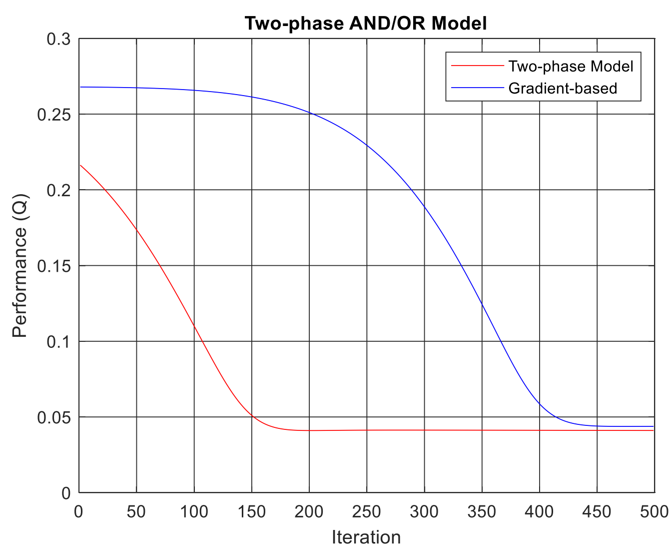


Figure 3-21: Improvements in performance index (Red) two-phase model (Blue) gradient-based. AND/OR network with two AND neurons (Boston hosing dataset)

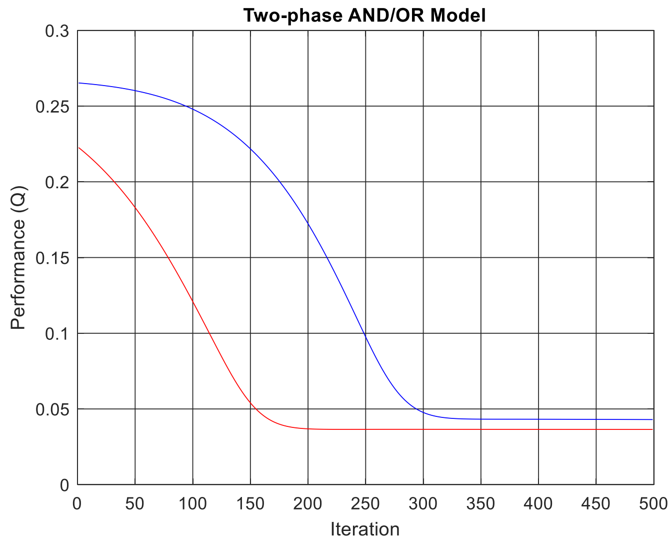


Figure 3-22: Improvements in performance index (Red) two-phase model (Blue) gradient-based. AND/OR network with four AND neurons (Boston hosing dataset)

Table 3-7: Results of training/ testing rule-based AND/OR network (Diabetes dataset)

Method	**Neuron = 2** **(Q) Training\Testing**	**Neuron = 4** **(Q) Training\Testing**
Two-phase Model	0.0839\ 0.0946	0.1021\0.1169
Gradient-based	0.11\0.13	0.1067\0.115

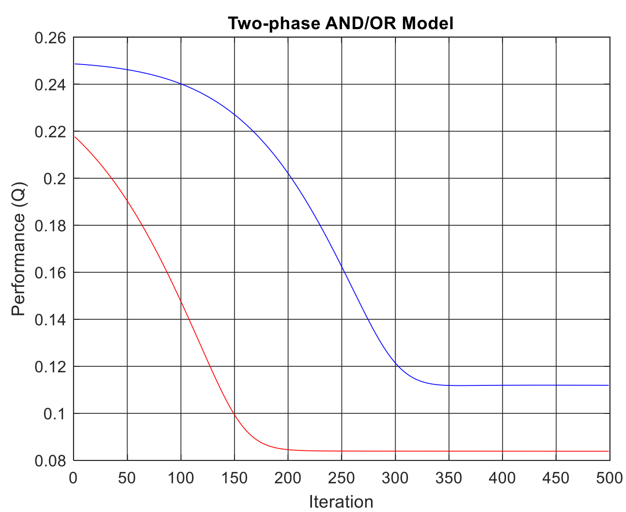


Figure 3-23: Improvements in performance index (Red) two-phase model (Blue) gradient-based. AND/OR network with two AND neurons (Diabetes dataset)

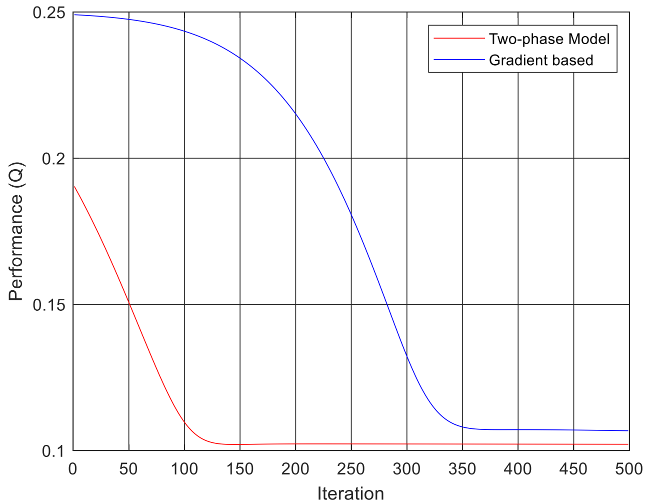


Figure 3-24: Improvements in performance index (Red) two-phase model (Blue) gradient-based. (Right) AND/OR network with four AND neurons (Diabetes dataset)

To perform fuzzy rule extraction, we introduce threshold values to prune weak connections [81]. Pruning of connections mechanism works whenever a connection weight is higher/lower than a threshold value which depends on the form of the neuron. For the OR neuron, the higher the value of some connection, the more significance the corresponding input is. To show this effect, a threshold value λ is introduced. The following formula returns the original value of the connection (***v***) if it exceeds its value or it is equal to λ:

$$v_\lambda = \begin{cases} v, & if\ v \geq \lambda \\ 0, & otherwise \end{cases} \tag{3.21}$$

On the other hand, for the AND neuron, the lower the value of some connection, the more significance the corresponding input is. To show this effect, a threshold value μ is introduced. The following formula returns the original value of the connection (***w***) if its value is lower than or it is equal to μ:

$$w_\mu = \begin{cases} w, & if\ w \leq \mu \\ 1, & otherwise \end{cases} \tag{3.22}$$

	Rule	Confidence level
Neuron = 2 $\mu = 0.5$ $\lambda = 0.6$	**IF** [(CRIM is LOW)$_1$ AND (INDU is HI)$_0$ AND (NOX is MED)$_{0.38}$ AND (DIS is HI)$_0$] **THEN** [MEDV is MED]	0.81
Neuron = 4 $\mu = 0.7$ $\lambda = 0.2$	**IF** [(CRIM is MED)$_{0.79}$ AND (CRIM is HI)$_{0.21}$ AND (INDU is LOW)$_{0.86}$ AND (NOX is LOW)$_{0.89}$ AND (NOX is MED)$_{0.11}$ OR [(CRIM is HI)$_1$] **THEN** (MEDV is LOW)	0.92

	Rule	Confidence level
Neuron = 2 $\mu = 0.5$ $\lambda = 0.5$	**IF** [(BP is MED)$_{0.35}$] **THEN** [Diabetes is LOW]	0.93
Neuron = 2 $\mu = 0.5$ $\lambda = 0.5$	**IF** [(AGE is MED)$_{0.26}$ AND (BMI is HI)$_{0.6}$ AND (TC is LOW)$_{0.54}$ AND (GLU is MED)$_{0.46}$ **THEN** (Diabetes is MED)	0.56
Neuron = 4 $\mu = 0.4$ $\lambda = 0.8$	**IF** (BMI is MED)$_{0.1}$ AND (BP is LOW)$_{0.36}$ AND (GLU is MED)$_{0.1}$ AND (GLU is LOW)$_{0.37}$ **OR** (AGE is MED)$_{0.36}$ AND (BMI is MED)$_{0.27}$ **THEN** (Diabetes is LOW)	0.69

3.5.4 Accuracy and Interpretability of Two-phase Model

Although the main objective of two-phase model is to improve accuracy and computation overhead, the other modelling requirements interpretability, and readability of extracted knowledge, were improved as an indirect advantage of the model. The presented design improves accuracy of approximation in several ways; optimal performance index Q values can be reached faster with fewer number of iterations which means less computing overhead. Even if number of iterations is equal in both models, the proposed two-phase model and the stand-alone gradient-based model, the training time of two-phase model has no negative impact on computation overhead or network efficiency since it mainly comes from feedforward learning mechanism where calculations are straightforward. Based on the experiments, improved performance index is always reached which reflects the efficiency of two-phased model in training AND/OR network. It is worth mentioning that, number of AND/OR neurons is a problem-specific parameter and it differs for each dataset. This specific parameter exhibits a direct relation with interpretation capabilities since the interpretation of datasets is derived from the underlying logic of fuzzy neurons. It is always a good practice to keep the number of neurons as low as possible to improve readability of extracted knowledge. We stress that the presented model should improve the overall results regardless of the number of neurons. It also improve overall efficiency and convergence rate of the learning process. Furthermore, interpretation of datasets, as in tables above, exhibit sound meaning that is well-justified and reflective of the existing experimental data.

3.5.5 Comparative Analysis of Results

The obtained results of the proposed learning mechanism were promising and encouraging but further analysis is necessary to legitimate the results. We have worked with several optimization algorithms in [9] to assess their efficiency and complexity when solving complex problems. Optimization problems, in general, involve maximization or minimization of an objective function, often subject to multiple constraints. High computational complexity hinders the use of optimization algorithms in applications that have limited power resources while slow convergence speed makes it unsuitable for time critical applications. The number of parameters, such as mutation rate and crossover, which must be tuned, adds complexity and computation overhead. In this regard, calculating the complexity of some optimization algorithms, say Genetic Algorithms (GA), requires computing the "*big O*" of all other operators (mutation, crossover, … etc). Adding that to the computation complexity of gradient-based learning which generally depends on number of iterations, number of samples and number of features will result in very high computation complexity. Eventually, efficiency of AND\OR networks degrade substantially [200].

The presented learning technique is augmented by the simplicity of feedforward neural networks in which weights of connectives are randomly applied. It eliminates the use of optimization algorithms to reduce the computing overhead which is a substantial improvement in the efficiency of AND\OR networks. Speed of learning coming from the randomness nature of feedforward learning allows a designer to try almost all possible fuzzy sets instead of trying to optimize fuzzy sets in a preprocessing stage. The randomness of weight generation is the reason of high accuracy of approximation exhibited by the presented design.

3.6 Summary

In summary, this chapter introduced the construction of logic-oriented networks that utilizes fuzzy neurons for better interpretation capabilities. The network strikes the best balance between the fundamental requirements of fuzzy modelling in which the accuracy-interpretability trade-off have to be met to the highest extent. The requirements of fuzzy modelling were discussed and presented in a taxonomy to show how accuracy, transparency and interpretability are cohesively tied together in an integrative process. Furthermore, we proposed a two-phase model to train the logic-oriented network composed of functional modules of AND/OR neurons. Training of the network aims to refine weight connections to improve performance index. The improvements were demonstrated on numeric data along with fuzzy sets to show that the design can be adopted in different cases with different datasets. We stress on the advantages of feedforward propagation in a random mode on the accuracy of approximation. Generating weights randomly can be implemented as a stand-alone learning algorithm on AND/OR networks. However, the integration of feedforward neural networks on AND/OR architectures with a gradient-based learning scheme ensures better values of performance index (Q) which result in better interpretation of datasets.

4. Cognitive Computing: Conceptual Knowledge Acquisition

Cognitive computing [160] is an essential part of artificial intelligence that provides a novel and effective methods for knowledge and concept discovery problems such as classification, image annotation and rule extraction [51], [54], [155], [156], [157], [158]. This emerging computing paradigm implements computational intelligence by perfectly deal with problems of uncertainty and partial truth to stimulates human cognitive processes such as learning and problem solving [59], [160], [161], [162], [163]. Hence, cognitive computing handles complex human problems, which are dynamic, information-rich, sometimes conflicting, and requires understanding of contexts.

A concept is a basic cognitive entity of knowledge and inferences to represent intrinsic regularities to explain and predict behaviors of an observed phenomena [201], [202], [203]. Formally, it has two constituent parts: its extension, which consists of all objects belonging to the concept, and its intension, which comprises all attributes shared by the objects. An important notion in formal concept analysis is thus a formal concept, which is a pair consisting of a set of objects (the extension) and a set of attributes (the intension) such that the intension consists of exactly those attributes that the objects in the extension have in common, and the extension contains exactly those objects that share all attributes in the intension [172]. A wide range of machine learning algorithms have been utilized in constructing concepts due to its efficiency in data analysis, information retrieval, extracting and visualizing hierarchies. It is worth mentioning that concept is somehow similar to granule computing as it can stimulate human thinking patterns and transform complex structures into simpler ones, and from coarser to finer piece of knowledge [59], [104], [172]. They both seek an approximation scheme to solve a complex problem effectively.

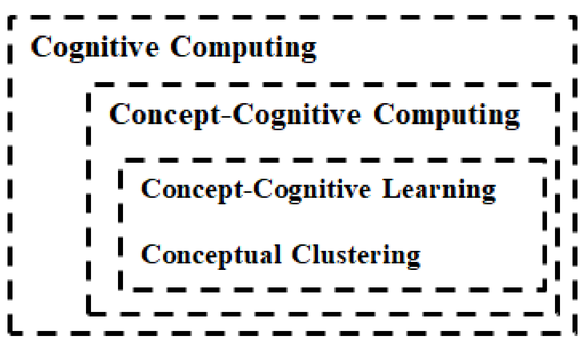


Figure 4-1: The relationships between several sub-fields of concept formation and learning

4.1 Concept- cognitive learning

Concepts are the foundation of human cognition, and humans gain knowledge rapidly through continuous learning of concepts. Thus, carrying on cognitive learning by concepts to solve a variety of practical problems has attracted the attention of many scholars [12]. A concept represents an abstraction of a type of thing in the world. For instance, the concept of a certain fruit, say *orange*, is an abstract description of all *oranges* containing information about an orange such as typical size, color and taste. Concept formation, learning of concepts, concerns observing unlabeled examples one after another and try to find a clustering of these examples into categories, which are then organized in a conceptual hierarchy. It is legitimate to say that concept-cognitive learning models mainly focus on how to gradually acquire knowledge and its properties. This type of learning, called Incremental learning, has been widely used in simulating the human brain to learn concepts through a computer system. Apart from ordinary machine learning approaches where learning is performed using stochastic gradient descent (SGD) optimization in batch mode where all training data are given at the same time, concept learning follows incremental scheme in which after each observation, or observations, the system needs to update its conceptualization. It is because the observations are made one after another. This dynamic environment of acquiring new knowledge force the decision-makers to quickly, and efficiently, react to ever-changing information by imitating the mechanisms of human brain. The area of incremental learning, also called continual learning [128], has been rapidly expanding, fueled by the potential utility of deploying continual learning algorithms for applications such as medical diagnosis or autonomous driving [100], [101].

Incremental learning concerns the learning process that takes place whenever new example(s) emerges and adjusts what has been learned according to the new example(s) [182]. In the context of concept learning, learning can reflect the cognitive process of a human brain by effectively updating the corresponding concept lattice as well as the embodied implication rules [179]. We could, therefore, list the main types of incremental learning:

- **Concept-Incremental learning**: this is to provide a new set of concepts such as new face images for a face recognition system.
- **Object-Incremental learning**: this is when adding one object to a knowledge base such as adding a new user in a face recognition system

- **Attribute-Incremental learning**: this is when attributes are provided after a learning system is trained.

The new added concepts, objects and attributes dynamically change the existing model. A system might need to "forget" the old structure and adjust what have been induced. Consequently, the mechanism of incremental learning faces some shortcomings, such as the model having to adapt gradually [175] and preservation of previously acquired knowledge without the effect of catastrophic forgetting [156], [170], [176], [177]. This specific limitation, the catastrophic forgetting, is due to the unavailability of the entire dataset. The unavailability of data is due to "temporal causes" in the old data cannot be reused once new data comes [192].

Concept space contains a set of all possible concepts. The space grows exponentially as number of concepts increases which then becomes an *NP*-hard problem construct and compute all concept. To demonstrate this difficulty, Let's consider the information table, Table 4-1, from [181] which shows three attributes "height", "hair" and "eyes" with respect to the objects. Given an object set {1, 3, 4, 5, 6}, the corresponding attributes shared by all objects is {*eyes* = *blue*}. Then the pair ({1, 3, 4, 5, 6}, {*eyes* = *blue*}) is called a concept. Now let's consider the concept space of Table 4-1, there are $2^8 - 1$ possible object sets. To construct a single object, we need to compute C_8^1object sets. That is {1}, {2}, …, {8}. Similarly, C_8^2needs to be computed when considering two objects, that is {1, 2}, {1, 3}, …, {7, 8}. Thus, it becomes *NP*-hard problem to construct a set of all concepts as it is very high resource consuming process to search the entire search space.

TABLE 4-1: Information table for concept construction

Object	Height	Hair	Eyes	Class
1	Short	Blond	blue	A
2	Short	Blond	Brown	B
3	Tall	Red	Blue	A
4	Tall	Dark	Blue	B
5	Tall	Dark	Blue	B
6	Tall	Blond	Blue	A
7	Tall	Dark	Brown	B
8	Short	Blond	Brown	B

Concept classification in conceptual hierarchies comes as a way to organize concept in readable and understandable way to a user. It reflects how humans would classify groups of entities. The idea is to order concepts from a "super-concept" to "sub-concept" hierarchy. For instance, let's consider a knowledge base that represents mode of transportation, for the set of objects {car, train, airplane} and the attributes {land, air}. We could draw conceptual hierarchy as in Figure 4-2 below:

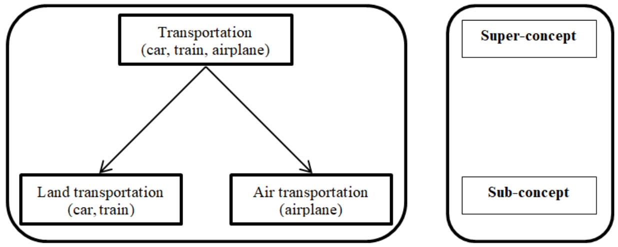


Figure 4-2: A Conceptual hierarchy

Hence, the hierarchies grow as concepts added in an incremental manner. Here, the challenge remains to ensure true retention of knowledge by balancing between plasticity (the ability to adapt to new knowledge) and stability (the ability to retain prior knowledge). Extreme plasticity can cause forgetting the prior information when learning a new concept. On the other hand, extreme stability, incremental concept learning may become more difficult. This phenomenon is known as the stability-plasticity dilemma [102].

The above explanation and discussion of cognitive learning shows the complexity of acquiring and leaning new knowledge when adopting concept learning scheme. Although the hierarchy scheme is sufficient in knowledge representation, it does not necessarily exhibit interpretation capabilities. Knowledge representation could reveal some hidden knowledge and introduce readability to knowledge domain but that it will certainly fail to manifest in revealing essential input-output interconnections.

4.2 Conceptual Clustering

Conceptual clustering focuses on dividing concepts into different categories. Rather than the conventional clustering analysis methods that are based on the similarity between geometric distances of data objects like K-means clustering, conceptual clustering are assembled into a single cluster not because of their pairwise similarity, but because together they represent a concept from a predefined set of concepts [170], [171]. Therefore, conceptual clustering has two crucial tasks: concept classification and concept generation.

In conceptual clustering, a system must know a priori knowledge "concepts" about the dataset. In this regard, the similarity between data points, x_1 and x_2, is a function of these points, context E, and a set of predefined concepts C;

$$\text{Similarity}\ (x_1, x_2) = f(x_1, x_2, E, C)$$

Groups, in conceptual clustering, are divided based on the quality of the resulting concept descriptions. Those groups have some understandable descriptions that characterize their belongingness. Entities are grouped together if they belong to the same concept, even if physically located "far" from each other [184], [185] [186]. This clustering mechanism solves the limitation of conventional clustering which is the lack of interpretation capabilities, by forming a class only if it is describable by a concept from a predefined concept class.

To illustrate the significance of priori knowledge in conceptual clustering, consider the Figure 4-3 which is taken from [184]. Conventional clustering method, say K-means, would consider the configuration of points as a collection of independent points. Therefore, the points (x_1 and x_1) would likely be placed into the same group based on the distances between point pair or similarity feature. On the other hand, a person would place the two points into two different groups. Here, the data points would be partitioned into groups not on the basis of pairwise distance between points, but on the basis of "concept membership". This method represents a significant difference between conventional clustering and conceptual clustering which is that points placed in same cluster represent the same concept. Therefore, the created groups would conform that the concepts are the letter A and the letter B. Taking advantages of available priori knowledge will allow the clustering algorithm to select groupings with meaningful concept descriptions which results in better interpretation of datasets.

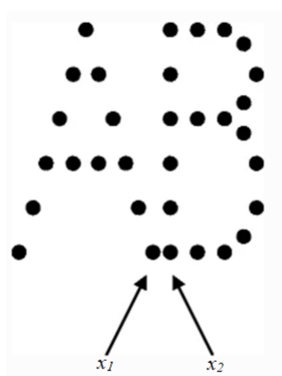


Figure 4-3: Conceptual clustering; two different concepts. The letter A and the letter B

Formally, conceptual clustering is defined as follows:

Let $O = \{o_1, o_2, \ldots, o_n\}$ be a collection of objects described in terms of a set of attributes $A = \{a_1, a_2, \ldots, a_m\}$. The collection O is organized into a set of clusters $G = \{g_1, g_2, \ldots, g_k\}$ such that the three conditions are fulfilled:

- Each cluster, g_i , has an extensional and intentional associated descriptions.

- Every object in O belongs to at least on cluster. That is

$$\forall\ o_i \in O, i = 1, \dots, n, \exists\ g_j \in G, j = 1, \dots, k, such\ that\ o_i \in g_j$$

- Each cluster in the solution contains at least one object of the collection.

Generally, classes in conceptual clustering are presented in hierarchies in top-down scheme. In this form, the set of objects is divided into a small number of classes, each of which may be divided into subclasses, iterating until a termination condition is met. On the other hand, bottom-up scheme may be used such that each object is initially considered to be in its own class; they are then grouped together, and the resulting groups are then brought together into super classes, until the top level is reached. However, since incremental scheme is widely used which means that clustering analysis observation one at a time, creating and modifying classes is based on the new information which, in return, will lower learning efficiency.

4.3 Interpretability of Conceptual Clustering

In general, clustering algorithms focus only in organizing the collection of objects into a set of clusters, leaving the interpretation of those clusters to the user. There has been limited success in addressing cluster interpretability in the original feature space [174]. Conceptual clustering algorithms, in addition to the list of objects belonging to the clusters, provide for each cluster one or several concepts, as an explanation of the clusters. These algorithms could be suitable for applications in which the users do not only wish to know the clusters existing in the data but some additional information that explains why certain clusters are formed [173]. However, conceptual clustering experiences several limitations that could make it difficult to be applied on real problems. These limitations are mainly related with a high computational complexity, several parameters tuning, the inefficient updating of the clustering after changes in the collection, the high complexity of the language used for building the concepts and the impossibility of processing objects described by mixed and incomplete data; these drawbacks make the study of conceptual clustering algorithms a research area of concern nowadays. Hereinafter, we will talk indistinctly about conceptual clustering algorithm and conceptual algorithm to refer to the same term.

Conceptual clustering forms a classification tree (conceptual hierarchy) where a concept is specified through three types of conditional probability distributions:

- Predictability $p(value|concept)$ represents the distribution of possible feature values for members of the concept: If you know that the observation x belongs to the concept C, which kind of feature values v do you expect it to have?
- Predictiveness $p(concept|value)$ this probability tells us how indicative a certain feature value is for concept membership.
- Relative frequency $p(concept|parent)$: indicates the frequency of a concept in comparison to its sibling in the hierarchy. It basically the the likelihood of the observation x to belong to a concept c if the observation x belongs to the parent concept p.

The interpretation is mainly based on Boolean relation that represents the presence or absence of an association through concept lattice. In concept formation and concept hierarchies, concept lattice is an essential component to describe the relationship between the objects and the attributes. It is a common graph structure with excellent mathematical properties which can clearly describe generalization and specialization relations between concepts. The concept lattice normally contains a plenty of redundant information especially when processing uncertain information [183]. Therefore, the reduction of lattice, relationship, improves information retrieval and knowledge representation. The generalization of ordinary concept lattice to fuzzy lattice through fuzzy concepts has led to a major disadvantage that is a fuzzy context contains a considerable number of fuzzy concepts which makes it very time-consuming to generate a fuzzy concept lattice, and therefore it is difficult to find important concepts [195].

We cannot discuss the interpretation of conceptual clustering without relating it to the basics of concept since it is a basic knowledge unit that provides some sort of explanation to a user. Let's recall that a concept can be identified by its extent and intent. A formal interpretation of a concept would look similar to the following:

"object g has attribute m".

Let's consider the example in Figure 4-4 that identifies a concept of "Bird". In this case, we could choose an object g to be "*Sparrow*" with the attributes "winged, feathered and bipedal". Subsequently, a single concept is formed with duality feature which means each of the two parts, extent and intent, determines each other. Let's write that in concise way:

({Sparrow}, {winged, feathered and bipedal })

The above explanation is explicit but it does not necessarily distinguish between the bird "*sparrow*" from other birds since other birds, objects, might share similar attributes. This binary relation determines whether the object has the attribute or not, it is usually represented by 1 or 0. However, the situation is different when we deal with a real-life dataset due to the complexity of objective things and the fuzziness of human thinking, which makes people prefer to express with natural language when they evaluate things. Therefore, we need to turn our attention to different alternatives that linguistically and semantically describe knowledge domain.

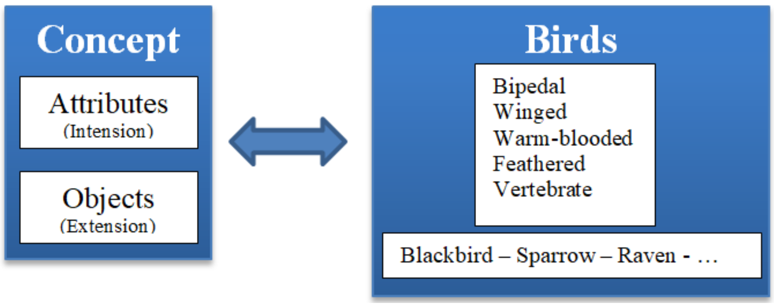


Figure 4-4: A concept presents a cognitive unit of meaning

Interpretability of formal concepts does not suggest the importance of separate attributes. It provides interpretable patterns a priori since it deals with such understandable patterns as sets of attributes to describe both classes and clusters. Therefore, several methods such as Shapley value from Cooperative Game Theory tries to cope with this issue by ranking separate attributes based on the importance of such attribute through some combinatorial formulas [187], [188]. However, some methods are unfeasible for large sets of attributes and hence we should use some attribute reduction techniques [189], [190]. In this case, it is preferable to direct our attention towards attribute aggregation by similarity to keep semantic and meaningful attributes even if there is some degree of redundancy or extra complexity paid for their processing. In the above concept extracted from Figure 4-4:

({Sparrow}, {winged, feathered and bipedal})

We might transform the concept to some linguistic expressions such as (*Sparrow* has *small* wings) which can be done by adding several attributes that explain the parent attribute (winged) Figure (4.5). This mechanism can go iteratively until clear linguistic explanation is obtained.

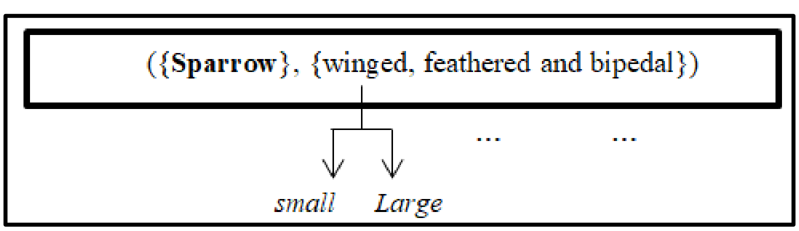


Figure 4-5: Adding more attributes to a parent attribute to provide linguistic explanations

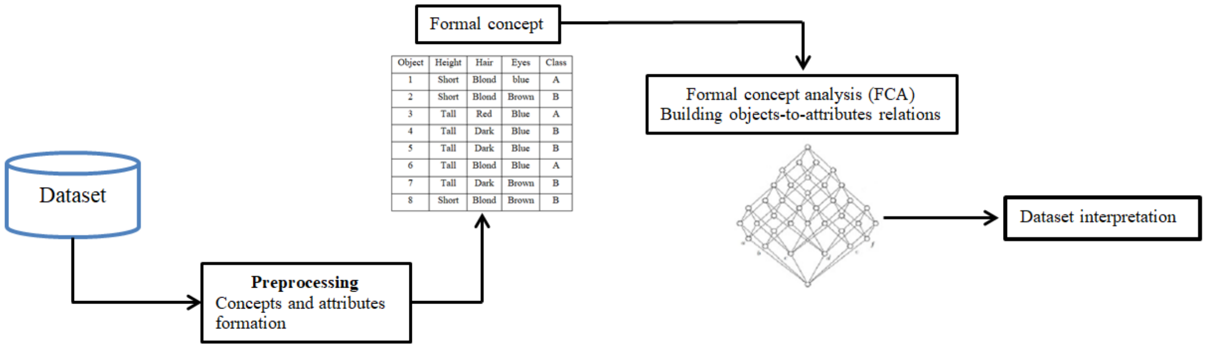


Figure 4-6: The overall scheme of interpreting data through concept formation and analysis

Overall, it may be said, interpretation of concept lacks the capability of presenting conceptually, and semantically, meaningful sound knowledge. Due to the incremental nature of building conceptual framework, the structure will arrive to a point in which the structure is complex and therefore we, as observers, will experience vagueness when interpreting the data. It is important to model concepts linguistically because the acquired conceptual knowledge is difficult to understand due to the substantial amount of linguistic concept knowledge in the concept lattice with imprecise linguistic information [193]. Therefore, it is necessary to process qualitative information by some fuzzy linguistic approaches that utilize concept lattice with linguistic values and manage fuzzy linguistic data.

4.4 Logic-oriented fuzzy concept discovery

Logic has been regarded as the core of concept-cognitive learning. It uses judgment, comparison, analysis and generalization to learn rational thinking to guarantee the accurate cognition to the objective world. Following the logical cognitive mechanism, these concept-cognitive systems highly rely on the patterns and methods of logical thinking in the concept-cognitive process, which leads to rigid concept-cognitive results. Logic-based cognition allows building logical consequences models, taking into account the linguistic relationships (objects and attributes). Although logic-based concept formation is rigorous, logic may cause cognitive process to stagnate or even to interrupt when humans are faced with complex problems. Therefore, it is better to utilize fuzzy cognition to find better concept-cognitive results. In fact, such action is more consistent with the cognitive laws of human brain, which can provide a new cognitive perspective for intelligent computing [178], [194]. Experimental evidence supports the notion that human behavior is highly consistent with fuzzy, and probabilistic, inference in both the sensory and cognitive domain.

Constructing a model of associations, that is built as a network through some logic-oriented fuzzy approaches, can support interpretation of data, which then become easy to translate the knowledge domain into human-like language.

4.5 Interpretability Analysis: A Discussion

Generally, there is no a clear and a specific definition of interpretability since there can be many different perspectives of what constitutes interpretability of different applications. Interpretation is a user-driven and domain-specific measure to convey why a system has predicted a certain outcome. Concept formation and analysis along with other conceptual-related algorithms do not provide interpretation, explanation of datasets. They only deal with data representation by doing some sort of organization or matching between objects and attributes. The mechanism of matching is inaccurately referred to as, sometimes in the literature, interpretability. To see the difference, it is much easier to detect bias or unfairness with an interpretable model. However, the same is not true with concept discovery due to the way the concepts are organized. I would argue that concepts could reveal some necessary knowledge to some extent which leads to eliminating some ambiguity but that does not mean it interprets the dataset.

If one would associate concept discovery with interpretability feature, it is only because concept design obeys structural knowledge of the domain and allows a view of how some variables interact jointly rather than individually. Take an example of a formal interpretation of a concept:

"object *g* has attribute *m*".

This is not an interpretation. Rather, it is a piece of knowledge that is directly extracted from the dataset without iteratively and deeply learning the cognitive entities. Generally speaking, the most accurate predictors with meaningful interpretation are constructed through an iterative process to refine the data. In practice, the performance of machine learning algorithms would be augmented by the ability to interpret results. Concept design and analysis does not provide some necessary characteristics of interpretability such as causality, which is tracing causal effects from inputs to outputs in a supervised learning algorithm [218], [219]. Furthermore, the ability to extract knowledge, provide readable and understandable quantitative and qualitative expressions are pivotal requirements when presenting interpretation of a dataset. Thus, interpretability\explainability is not a feature of concept analysis or conceptual clustering.

4.6 Summary

In this chapter, we thoroughly discussed concept formation and learning from interpretability point of view. Concept discovery, as a substantial method in knowledge acquisition and representation, is essential in supporting decision making processes due to the dynamic nature that resemble real-world scenarios. Therefore, interpretation is an important facet that needs to be addressed carefully. Although the literature that discusses concept analysis is rich, we firmly believe that concepts have to be associated with a certain linguistically sound meaning to produce more specific sound view of the data.

In the subsequent chapters, we utilize concept formation in logic-oriented networks for better interpretation that match human cognitive and produce human-like language. The representation of knowledge domain in the form of understandable expressions is essential in user-centric modelling in which the models should help the user justify decisions being taken.

5. Concept Discovery: Interpretation Framework of Conditional Clustering and Logic Networks

Concept formation is not enough to produce sufficient interpretation of datasets unless it is augmented with iterative learning scheme to refine acquired knowledge followed by efficient knowledge representation mechanism to produce sound conclusions. In this consideration, concepts can be associated with certain linguistically sound meanings to achieve a better and clear understanding of newly acquired knowledge. Starting with a simple knowledge entity represented as a concept which is regarded as pairs of items in the form:

C = (reference information granule, description).

Then, precise knowledge is acquired through logic or rational iterative learning by the implemented logic-oriented fuzzy neural networks. Concepts are built upon a context reside in the output space called referential information granules. Those information granules are focal pieces of knowledge commonly provided externally (by the user or designer) or through some initial data analysis. For instance, they could include some simple terms such as *high* inflation, *low* temperature, *high* approximation error, …etc, or more advanced composed in the form (*low* temperature and *high* pressure), etc. The description is provided (expressed) in terms of other independent variables. The construction of the concept is completed in two phases by invoking conditional fuzzy clustering followed by the use of logic networks. Both are essential to reveal concepts in the form of logic expressions. These expressions are realized and calibrated using logic AND and OR neurons studied in fuzzy neurocomputing. These fuzzy logic-driven neurons are generic processing units that offer a high level of parametric flexibility and are endowed with significant interpretation abilities.

The approach of concept formation and knowledge extraction (Figure 5-1) describes reference information granules positioned in some space (output space) inducing fuzzy sets localized in the input space. Reference information granules are either provided by designers that articulate the individual perspectives or are based on experimentally justifiable facts. The description is realized by running a conditional fuzzy C-means algorithm followed by a calibration process completed through logic networks. The synergy between conditional clustering and logic networks presents highly cohesive linguistic dependency between objects and their attributes, adding originality and providing with a graphical representation of concepts. The design performance is evaluated using both synthetic and machine learning datasets.

We provide full description of *concepts* in chapter 2 and chapter 4. We relate the contents of this chapter to what was presented before. In this chapter, we propose a method to describe and quantify data using concept. Logic-oriented network is incorporated to provide qualitative and quantitative description of knowledge domain.

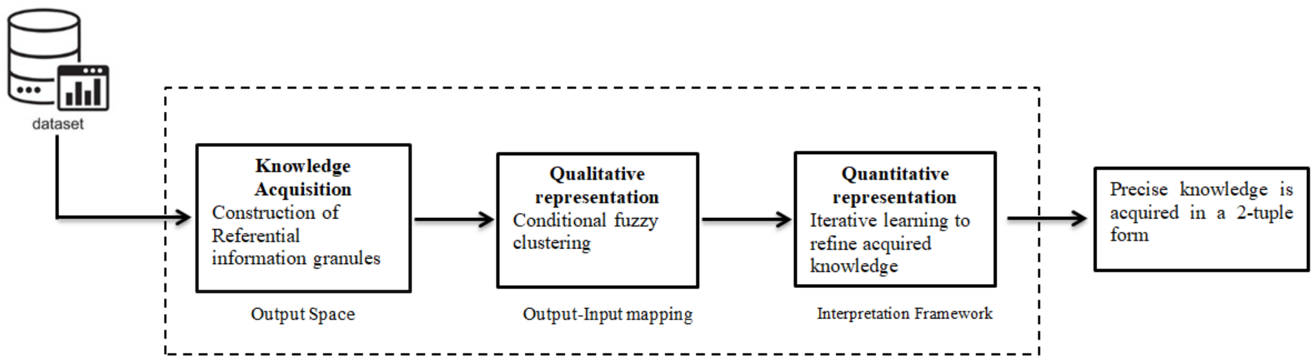


Figure 5-1: General scheme of concept formation and knowledge extraction

5.1 Underlying design of Concepts

In the basic setting of concept discovery, input variables concern descriptive attributes applied to some objects (reference information granules) that reside in the output space. The determination of reference information granules is an independent task and is defined in a certain feature space, and $\boldsymbol{Y}$ is a universe of discourse in which the referential fuzzy sets $G_1, \ldots, G_n \in F(\boldsymbol{Y})$. The term $F(\boldsymbol{Y})$ stands for the set of all fuzzy sets on $\boldsymbol{Y}$ in which all fuzzy sets have some semantics, such as "*low*" for G_1 and "*high*" for G_2, and so on. The resultant is several fuzzy sets defined over a space of reference information granules involving a small number of terms, say $\{low, medium, high\}$.

An object of a concept is more reflective when it is described by more than one descriptor in the input space. Then, each individual variable (descriptor) is granulated into c clusters to reflect the nature of the variable. For example, the reference information granule (*high inflation rate*) is better described with more detailed interpretability by *production cost* and *money supply.* In this case, centers of clusters are projected onto the descriptor variables, and a Cartesian product of clusters c defined in the individual input spaces as $c_1 \times c_2 \times \ldots . \times c_i$. In the sequel, the relationship between clusters of descriptive variables and a single referential information granule can be translated as a family of logic rules containing a collection of output-inputs pairs.

An object of a concept can be a single information granule or multiple information granules to describe the entire output space in which they can be connected by logic lattices, as shown in Figure 5-2. The connectedness of concepts provided by logic networks in later stages is crucial for interpretation and reasoning [164]. One should note that the number of referential fuzzy sets (or length of information granules) impacts the specificity criterion. As the number of elements covered by the information granules increases, the specificity measure decreases and thus becomes less detailed. The underlying design of reference information granules suggests that the meaning of an object must be obvious; otherwise, it may be difficult to discover or reveal its interpretation.

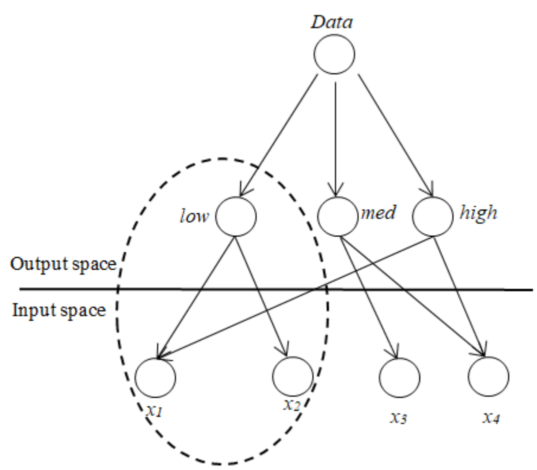


Figure 5-2: A single concept within a dataset

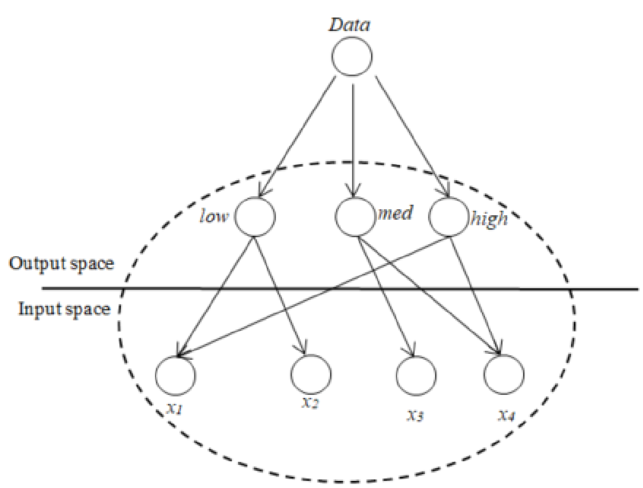


Figure 5-3: A dataset is composed of multi-concepts.

It is crucial not to rely only on the designer's expert-driven point of view when defining a concept that may not reflect the essence of experimental data. A data-centric approach, by using a clustering method, is an alternative way to construct information granules, or objects, that are experimentally justifiable and legitimate. In fact, a data-driven approach is better in terms of predictive accuracy [146]. In this case, the formation of data entities is accomplished in one of two ways: either based upon qualitative guidelines provided by the designer, in what is called designer-oriented data analysis [129] or based on experimentally justifiable facts through fuzzy clustering methods. Typically, a sufficient number of concepts is necessary for building semantically sound relations to reveal meaningful interpretation that reflects high transparency measures. This process is employed through conditional fuzzy clustering algorithm to introduce directionality to the formation of a frame of reference. In light of this, the quality of objects and their relations to the descriptive variables depends on experimental evidence and the designer's perspective.

5.2 Construction of Reference Information Granules

Any information granule $\mathcal{G}$ can be represented through a family of reference information grapnels (frame of reference) that is constructed based on some data **X** and defined in the corresponding output space, $G_i\colon \mathbf{Y} \rightarrow [0, 1]$. The family of reference information granules helps to clearly express any information granules with different semantics based on their elements with different levels of granularity [46], [105], [165].

The reference information granules are constructed in two main ways:

a. **Expert-oriented information granules**: the reference information granules result from a designer's preferences. The designer determines the number of information granules, specificity and coverage for data structure discovery. Subsequently, the results are expressed in terms of reference information granules and directly reflect the interest of the user.
b. **Algorithmic information granules**: the reference information granules are determined by a clustering process, such as conditional fuzzy clustering, based on experimental data from a previous dataset [46]. Here, the concepts are described in terms of the already existing referential information granules.

The granularity of the frame of reference is associated with the specificity measure because the number of building blocks (granularity) controls the level of revealed details. A less specific frame of reference delivers a general look at a model, while a more detailed model requires more referential information granules.

5.3 Qualitative and Quantitative Description of Concepts

This section provides a qualitative and quantitative description of concepts by taking advantage of the perceptual task of logic-driven AND/OR neurons.

5.3.1 Qualitative Description

Similar to linguistic modelling design, the qualitative description of concepts is built on two main pillars, fuzzy clustering and linguistic terms [10], which are contained in a coherent framework for better interpretability. The qualitative description provides different conceptual constructs with meaningful landmarks, such as (*high* or *low* price). In this manner, the entire output space is divided into a frame of reference where a designer might employ a clustering method or certain specifications to construct information granules. However, in the case of a few reference information granules defined in the output space, some crucial data structures might be unintentionally neglected. Alternatively, an experimentally generated frame of reference is justifiable and semantically sound. Specificity and coverage measures should be considered to reflect the essence of the dataset and reveal a practically sound, meaningful description.

Qualitative description of a reference information granule starts with the construction of c clusters of descriptive variables with linguistic landmarks. Subsequently, centers of clusters are projected onto the output space through conditional clustering to reveal the projection of the prototypes to their individual input space. Figure 5-4 shows how a qualitative description is revealed; a symbolic descriptor (L) is given for the large centroid values v_{Large}, and (S) is given for the small centroid values v_{Small}. Subsequently, centers of clusters are joined through an AND operation to generate prototypes (v_i) and aggregated through an OR operation in the following way:

$$\boxed{\begin{array}{c}(x_1)_{\text{Large}}\ \text{AND}\ (x_2)_{\text{Small}}\\ \text{OR}\\ (x_1)_{\text{Small}}\ \text{AND}\ (x_2)_{\text{Large}}\end{array}}$$

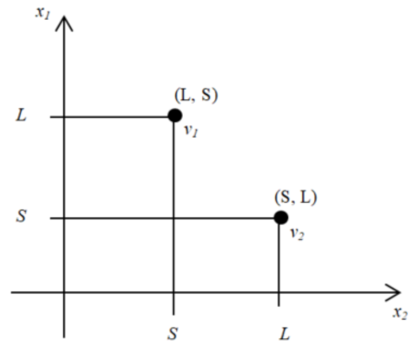


Figure 5-4: The projection of the prototypes to their individual input space, where S and L are small high labels of the prototypes

Although the extracted qualitative description includes all input variables, the qualitative description lacks relevance quantification of descriptive variables to a reference information granule. Thus, a precise description that relies only on linguistic landmarks is not feasible. Hence, describing a frame of reference quantitatively and incorporating a learning approach can eliminate unnecessary descriptors for an optimal concept design.

5.3.2 Qualitative Concept Description

Quantitative description concerns the construction of a logic network, and AND/OR neurons [147], based on a qualitative description followed by the incorporation of gradient-based learning approach (Figure 5-5). The learning approach is utilized to quantify the relevance of descriptive variables and interpret a frame of reference more precisely. In the basic setting of quantitative concept description, a logic network is constructed in which prototypes (v_i) are formed with the aid of the logic-oriented fuzzy AND neuron while an OR neuron aggregates the prototypes to reveal the description of a reference information granule.

The computational operations by the logic-driven operators have no negative effects on the information granules (clusters and concepts) because they are stable in the sense of holding their meaning across any experimental evidence. In multi-dimensional data, it is not possible to depict a prototype's projection onto output space due to limitations in visualization techniques. Thus, linguistic interpretation through concept design can perfectly reflect the nature of the dataset. In a nutshell, the constructed granules preserve a high degree of stability when building logic networks, extracting knowledge and constructing logic expressions.

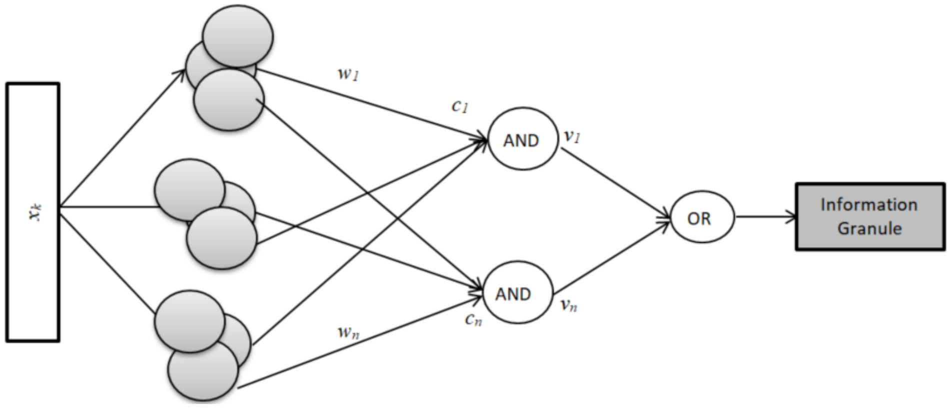


Figure 5-5: Quantitative description of concepts through a logic network. The network describes a single concept.

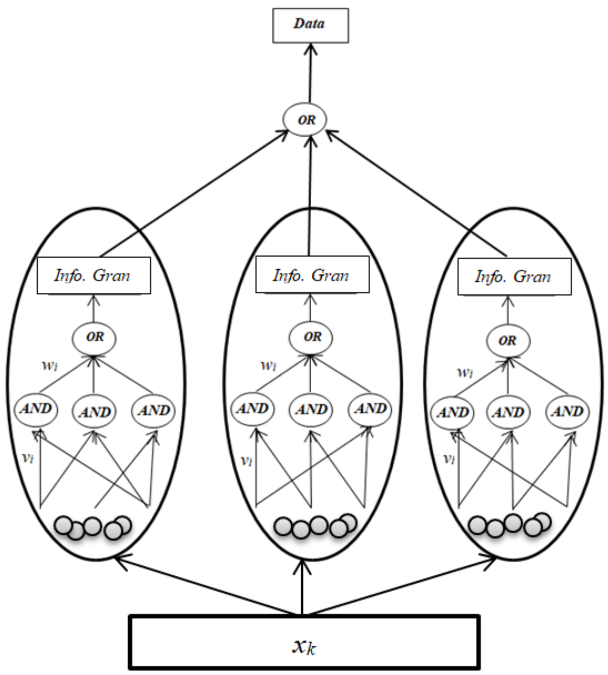


Figure 5-6: Several reference information granules describing a dataset quantitatively by adding an OR neuron.

The description of concepts is quantified through a series of network connection refinements in a deep learning manner. The refinement of the connection weights helps to unveil the interpretation of information granule-descriptors mapping by modifying the network's parameters to minimize the performance index Q. In this work, we use an accumulated performance index Q_N to express the approximation error for all training data. Root-mean-square error (*RMSE)* is used in training the networks:

$$RMSE = Q_N = \sqrt{\frac{1}{N}\sum_{K=1}^{N}(y_k - \hat{y}_k)^2} \qquad (5.1)$$

The minimization of error requires a back-propagation algorithm to update the weight of the connection. The detailed logic expressions depend on the realization of the *t*-norm (AND neuron) and the *t*-conorm (OR neuron). In this work, we adopted the product realization of the *t-norm* (a t $b= ab$) and the probabilistic sum realization of the *t-conorm* (a s $b = a + b$), where a and b are neuron's connections, to allow for a higher degree of interactivity between connections to optimally interpret the system. However, that approach is accompanied by high computation costs that can be reduced by embracing different learning approaches [79].

The computation aspects of AND neurons in logic networks realize the nonlinear transformation from $[0, 1]^n \rightarrow [0, 1]$ in the following way:

$$y = AND(\boldsymbol{x}; \boldsymbol{w}) = T_{i=1}^{n}(x_i \; s \; w_i) \qquad (5.2)$$

where x_i is the ith input, w_i denotes the associated weight, and T is the *t*-norm [166]. The network can be interpreted with a series of logic expressions given the semantics of logic operators. Its main role in the context of concept discovery is to connect clusters of input variables and generate prototypes (Figure 5-5). The logic representation of Equation (5.2) shows the processing and underlines the available parametric flexibility of the module residing within its connections. As the interpretation suggests, Equation (5.2) has the following interpretation:

y = (x_1 or w_1) and (x_1 or w_1) and … and (x_n or w_n).

In contrast, an OR neuron is slightly different when it is interpreted. The neuron realizes a nonlinear transformation from $[0, 1]^n \rightarrow [0, 1]$ as:

$$y = OR(\boldsymbol{x}; \boldsymbol{w}) = S_{i=1}^{n}(x_i \; t \; w_i) \qquad (5.3)$$

where x_i is the ith input, w_i denotes the associated weight, and S is the *t*-conorm [166]. The OR neuron works as an aggregator of different prototypes and information granules and produces the most relevant ones. Equation (5.3) preserves the parametric flexibility needed to better interpret a dataset conceptually in the following manner:

y = (x_1 and w_1) or (x_1 and w_1) or … or (x_n and w_n).

The nonlinearity transformations add interpretability to the architecture because they can be treated as models of linguistic modifiers [138]. The network model implicitly encodes a set of logic expressions in its recurrent multilayer topology. The topology suggests a relationship between the network structure and the concept formation to extract linguistic knowledge after implementing a learning scheme through a neural network. Hence, fuzzy-based architecture provides a high level of interpretability of revealed knowledge.

The interpretability framework of the design is introduced through pertinent pruning mechanisms [80], [81]. The pruning of the weakest (unnecessary) connections helps to facilitate the process of interpretation and reduce the structure of the detailed logic expressions to their essential substructures with the most meaningful topology. The connections-pruning mechanism works whenever a connection weight is higher/lower than a threshold value that depends on the form of the neuron. For the OR neuron, the higher the value of a connection is, the more significant the corresponding input is. A threshold value λ is introduced to highlight this effect. The following formula returns the original value of the connection (v) if it exceeds its value or it is equal to λ:

$$v_{\lambda} = \begin{cases} v, & if\ v \geq \lambda \\ 0, & otherwise \end{cases} \tag{5.4}$$

Meanwhile, for the AND neuron, the lower the value of some connection, the more significant the corresponding input is. A threshold value μ is introduced to show this effect:

$$w_{\mu} = \begin{cases} w, & if\ w \leq \mu \\ 1, & otherwise \end{cases} \tag{5.5}$$

Pruning of connectives has a crucial role in controlling specificity and level of abstraction of the revealed knowledge. By tuning the threshold values (μ, λ), some concepts and clusters will be dropped, which results in higher or lower specificity or coverage. The final results presents an optimal interpretation framework (and build an AND/OR model) and extract logic expressions that linguistically interpret the knowledge domain.

In this work, the AND/OR network may experience an overfitting case when there is a minority representation of instances for a certain context that deteriorates the quality and accuracy of results. Adding a regularization term has changed the cost function to the following:

$$Q_N = \sum_{K=1}^{N}\left(target_k - NN(x_k, w)\right)^2 + \lambda \sum_{\substack{all \\ weights}} w_i \tag{5.6}$$

5.3.3 Visualization of Concepts

Logic expressions can be visually illustrated to present simplicity and better concept representation [167]. Larger text size or darker color means the expressions are more relevant, while smaller text size or lighter color means the expression is less relevant. The font size and color are determined by the weight value of a neuron connective to reflect the connective's significance. Quantitative description allows for better concept visualization due to the nature of logic networks while the qualitative description provides simple concept description without prioritizing concept's relevance. The visual illustration follows the following rules:

- For the AND neuron, the lower the value of some connection, the darker the color of font is to reflect the input's high relevance.
- For the OR neuron, the higher the value of some connection, the larger the font is to reflect the input's high relevance.

The following example illustrates the visualization of rule relevance:

(a_3) AND (a_2) AND (a_1) AND …. AND (   )

OR

(b_1) AND (b_3) AND (b_2) AND …. AND (b_n)

In this regard, the first term ((a_3) AND (a_2) AND (a_1) AND …. AND (   )) is more significant than the second term of the logic expression which is reflected by the larger font size. On the other hand, the variable (a_3) within the first term of the logic expression is more significant than other variables and that is reflected by the darker font color.

5.4 Experimental Studies

To illustrate the design process of the concept discovery carried out by the logic network model and conditional clustering, we discuss in the following sections several experiments conducted on synthetic data and machine learning datasets. Datasets are chosen based on different size scales to better represent the concept design and its interpretations.

Two important points must be taken into consideration:

- Imbalanced data were considered when training the model to avoid poor data interpretation. The work follows the data-level approach by over-sampling the minority in a pre-processing step.
- Logic expressions are arranged successively based on their decreasing relevance.

A. Synthetic Data

A two-dimensional synthetic dataset with multi-input, single-output data is generated using the function $y = sin(10(x^2 + y^2))/10$ over the range [−10, 10]. This dataset contains 550 data points using a uniform distribution. The data are split into 80% and 20% to train and test the model. The frame of reference in the output space is formed with different linguistic labels. In the experiments, p, which denotes the number of reference information granules in the output space, is set for different values, p = 2, 3 and 5. Furthermore, the number of clusters c $\in\{2,\ 3,\ 5\}$. We present detailed results for some cases that better interpret the dataset while we report other results more concisely.

In the experiments, the assigned linguistic terms are meant to easily define and distinguish the frame of reference. The matching of the frame of reference with their descriptive input variables in the input space is employed through the construction of the logic network. Input variables are ANDed to construct a list of prototypes in a straightforward process. Meanwhile, an OR neuron aggregates the frame of reference and reveals the dominant information granule of a dataset.

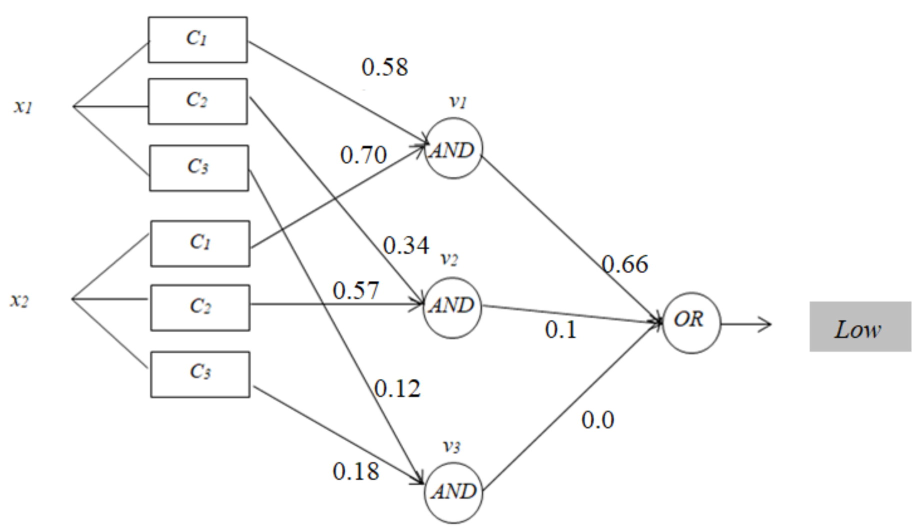


Figure 5-7: Quantitative representation of the reference information granules (*low*)

The three reference information granules are interpreted as follows along with the performance index of the learning:

Low **(**Train = 0.095– Test = 0.1754**)**

$[X_1(c_1)$ **AND** $X_2(c_1)]$ **OR** $[X_1(c_2)$ **AND** $X_2(c_2)]$ **OR** $[X_1(c_3)$ **AND** $X_2(c_3)]$

Medium **(**Train = 0.108 – Test = 0. 163**)**

$[X1(c2)$ **AND** $X2(c3)]$ **OR** $[X_1(c_1)$ **AND** $X_2(c_2)]$ **OR** $[X_1(c_3)$ **AND** $X_2(c_1)]$

High **(**Train = 0.129 – Test = 0.251**)**

$[X_1(c_1)$ **AND** $X_2(c_1)]$ **OR** $[X_1(c_3)$ **AND** $X_2(c_3)]$ **OR** $[X_1(c_2)$ **AND** $X_2(c_2)]$

To describe the whole dataset (that is, the frame of reference generated out of the information granule in output space), we further train the three reference information granules to better interpret the dataset and to explicitly present the dominant concept of the dataset.

After further training of the network, the resultant logic expression is obtained:

DATA = *Low* $_{(0.64)}$ **OR** *Med* $_{(0.56)}$ **OR** *High* $_{(0.15)}$

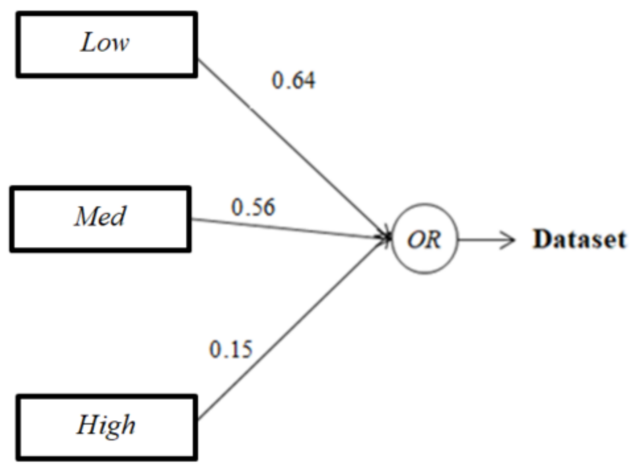


Figure 5-8: Quantitative description of multi-concepts of a dataset

The above logic expression is re-rewritten for better graphical representation and readability as:

DATA = *Low* **OR** *Med* **OR** *High*

Note that the values associated with each connection in (Figure 5-8) indicate the significance of the related concept. The final logic expression implies that (*low*) is more significant in interpreting the dataset. In other words, the following expression exhibits relevance over other logic expressions:

[*X1(c1)* **AND** *X2(c1)*] **OR** [$X_1(c_2)$**AND** $X_2(c_2)$] **OR** [$X_1(c_3)$**AND** $X_2(c_3)$]

In the subsequent cases, we extended the experiment for more reference information granules to pursue better interpretability. Note that the linguistic term (*low*) when $p = 3$ differs from the linguistic term (*low*) when $p = 5$.

Following a similar process from the previous experiment to determine which reference information granule has the highest relevance, we further train a logic network with $p = 5$ {*low*, *low-med*, *med*, *med-high*, *high*}. We arrived at the following logic relationship:

DATA = *low* **OR** *low-med* **OR** *med-high* **OR** *high* **OR** *med*

B. Selected Machine Learning Data

The following experiments have been conducted on several real-world datasets [168]. A thorough process description is provided for the Boston Housing Dataset to clearly present the design and clarify the explicit knowledge extraction in concept-based modelling. For other datasets, brief results are presented to show the design efficiency.

1. Boston Housing Dataset

The Boston Housing Dataset was chosen because of its popularity in the literature, which makes it easier to measure design competence. Three referential information granules are generated out of the output variable (*MEDV*) with different meaningful linguistic landmarks: *Low*, *Med* and *High*. The triangular membership function was employed to form and describe the frame of reference (Figure 5-9). Once again, we try to rely on experimentally justifiable and legitimate concept generation rather than expert orientation. As a result, the decision-making process will be trustworthy because the degree of model interpretability increases.

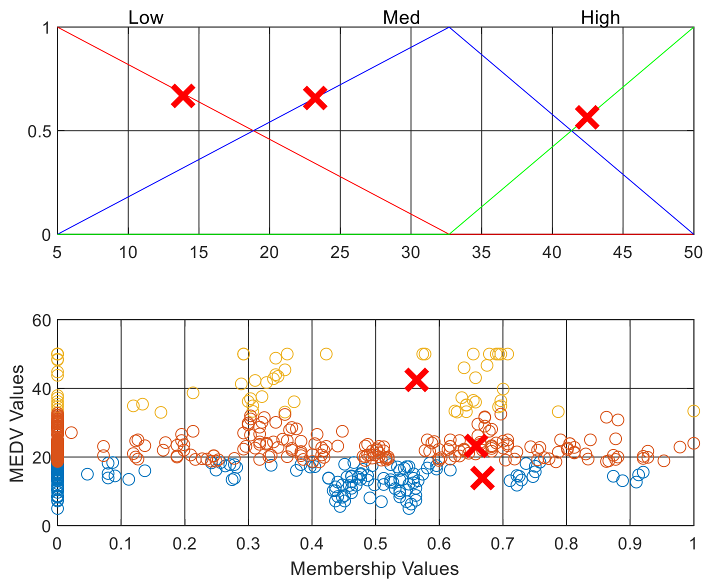


Figure 5-9: Visualization of the output space for three reference information granules.

Conditional fuzzy clustering is applied to input space variables to build concept pair relations and extract the qualitative description of the knowledge domain. For the object (*high*), as in (Figure 5-9), f_k is all membership values of the object. These values work as a constraint when generating membership matrices U of clusters in the input space. Following the same procedure, each centroid is given a symbolic value (linguistic landmark) to qualitatively describe the dataset and the relations between the centroids. In the case of $c = 2$, prototypes and symbolic labels are depicted in the following radar figure (Figure 5-10):

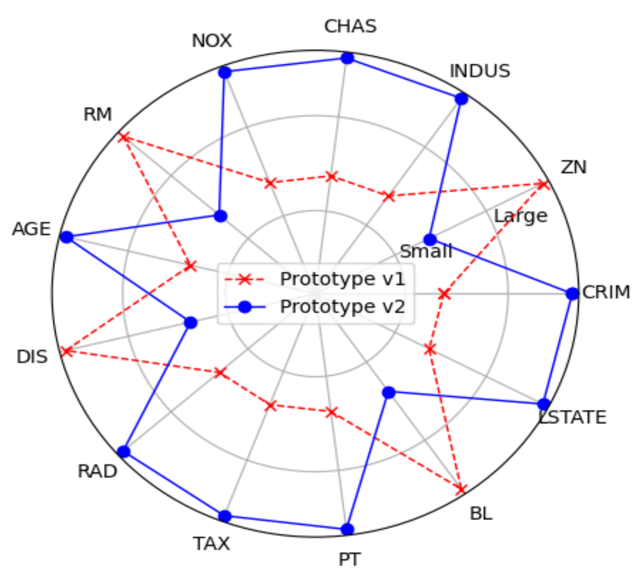


Figure 5-10: Qualitative description of the object (*high*), $c = 2$.

The construction of prototypes deals with the linguistic terms. Given the letter (L) for the large centroid values and (S) for the small centroid values, it is a straightforward process to extract the prototypes from (Figure 5-10). As an example, the prototype $\boldsymbol{v}_1$ (the blue line in Figure 5-10) of the concept **MEDV** $_{high}$ is:

$(L)_{CRIM}$ AND $(S)_{ZN}$ AND $(L)_{INDUS}$ AND $(S)_{CHAS}$ AND $(L)_{NOX}$ AND $(S)_{RM}$ AND $(L)_{AGE}$ AND $(S)_{DIS}$ AND $(L)_{RAD}$ AND $(L)_{TAX}$ AND $(L)_{PT}$ AND $(S)_{BL}$ AND $(L)_{LSTATE}$

Centroid values certainly affect the data structure of each cluster, which affects the logic expressions extracted from the knowledge domain. Thus, data points will have different meanings when associated with different concepts because the objects of a concept change the structure of a cluster even if the cluster resides in the same region of another concept. Subsequently, the linguistic landmark (Large L) has a different meaning when we change the value of c from 2 to 3 (Figure 5-11).

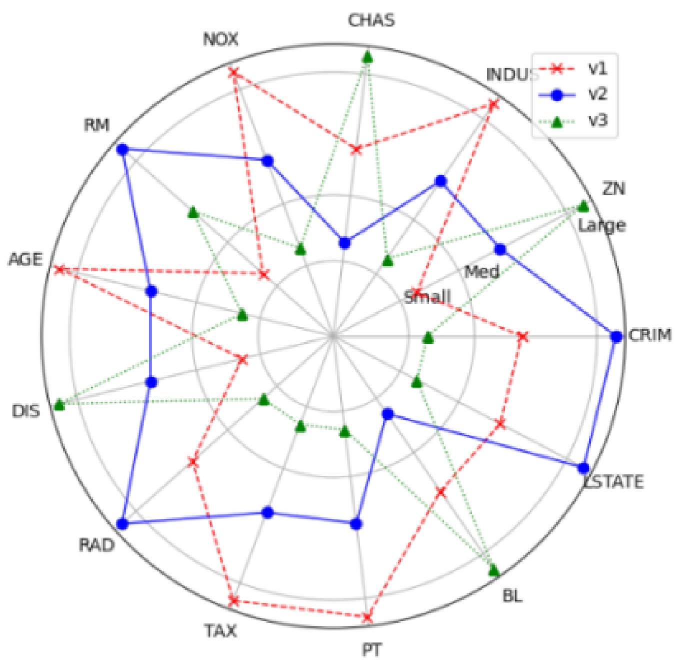


Figure 5-11: Qualitative description of the object (*high*) with $c = 3$, $v = 3$

Other experiments were conducted on the case when $p = 3$ and $c = 3$. The processes of generating prototypes are similar. However, because the number of prototypes and linguistic terms is higher, interpretation of concepts and knowledge domain differ. Increasing the number of clusters has a positive impact on the interpretation of concepts because of the diversity added to the formation of prototypes.

A similar process can be carried out for other concepts to build the initial relations between different entities in the form of logic expressions. However, knowledge extraction from the dataset has not yet been completed. Revealing the meaningful linguistic description is a substantial process. We only have completed the generation of reference information granules, clusters and the formation of prototypes in the input space.

In the training of a network with all relative descriptive variables for a certain reference information granule, we measure the performance index (Q) (Equations 5.1 and 5.6) to find the relevance of constructed prototypes to that information granule in the output space. The results indicate the significance of extracted logic expressions to a certain concept. Pruning conditions (λ, μ) (Equations 5.4 and 5.5) are applied on weights connectives for better interpretability of concepts. Pruning is a well-regulated method and it is essential when building relations between concept pairs to explicitly show the most significant linguistic terms. Tuning the parameter values can vary the level of abstraction (generality) based on the designer's perspective without changing the meaning of concepts (information granules). If the pruning parameter of AND neurons is set to high, you might end up with few descriptive terms, and the description becomes more specific.

2. Computer Hardware Dataset

This dataset contains data describing the relative central processing unit (CPU) performance of computer hardware (ftp://ftp.ics.uci.edu/pub/machinelearning-databases/cpu-performance/). The dataset is a small-scale dataset where the frame of reference is relatively limited. Unless the number of information granules is defined by a designer, experiments show that the output space can be linguistically divided into any values of {2. 3} as shown in (Figure 5-12). In our experiments, the frame of reference is defined with three linguistic terms.

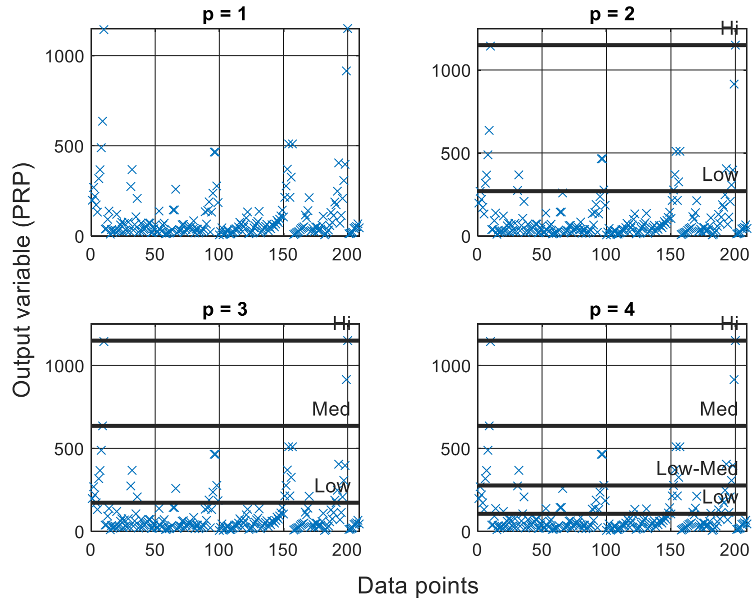


Figure 5-12: Generating frame of reference in the output space

Qualitative descriptions are drawn directly from Figure 5-13. Because there are only two clusters (*c* =2), we could assign the linguistic terms {*Small*, *High*} to each cluster and then connect the first cluster of each input variable with an AND neuron to extract the description. The referential information granules (*Low*) are qualitatively described by:

$\text{MYCT}_{\text{Small}}$ **AND** $\text{MMIN}_{\text{High}}$ **AND** $\text{MMAX}_{\text{High}}$ **AND** $\text{CACH}_{\text{High}}$ **AND** $\text{CHMIN}_{\text{High}}$ **AND** CHMAX_{Hi}

OR

$\text{MYCT}_{\text{High}}$ **AND** $\text{MMIN}_{\text{Small}}$ **AND** $\text{MMAX}_{\text{Small}}$ **AND** $\text{CACH}_{\text{Small}}$ **AND** $\text{CHMIN}_{\text{Small}}$ **AND** $\text{CHMAX}_{\text{Small}}$

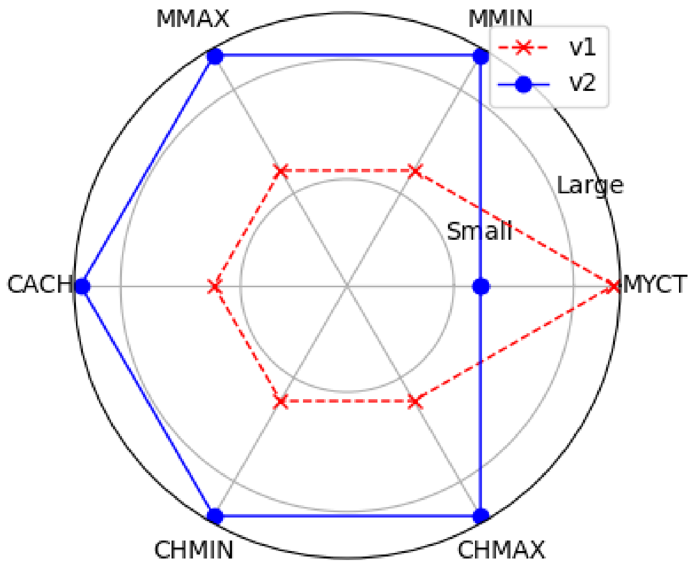


Figure 5-13: Construction of prototypes of the referential information granules (*Low*)

A similar procedure can be used to extract qualitative descriptions of a specific information granule. Then, logic AND/OR networks can highlight the most significant terms through gradient-based learning. The following quantitative description reveals the relevance descriptors to the reference information granule (*Low*):

[(CACH is Low) **AND** (CHMIN is Low) **AND** (CHMIN is Low) **AND** ( ) **AND** ( ) **AND** ( )]

OR

[(MYCT is Low) **AND** ( ) **AND** (MMAX is Hi) **AND** ( ) **AND** ( ) **AND** (CHMIN is Hi)]

Interpretability of the concept discovery framework is a high-efficiency feature as it can precisely match a region of output space to a specific cluster in the input space. Logic expressions have shown high explainability measures of the knowledge domain and are linguistically and precisely represented in a way that a human can understand.

3. Delta Elevators Dataset

This dataset is larger than other datasets in this work, with 9,517 instances, six inputs and a single output. We followed the same procedures used with other datasets. The instances are split into 80/20 training and testing sets. The output values are within a short range [−0.014, 0.013]. Once a reference information granule has been formed, coverage and specificity are evaluated. The number of concepts does not necessarily correlate with the number of output instances. Thus, we considered the frame of reference p = 2, 3 and 4 with different values of clusters for input variables. After testing the dataset for different numbers of referential information granules (and linguistic terms) with descriptive variables, (Figure 5-14) describes the output space in an experimentally justified manner.

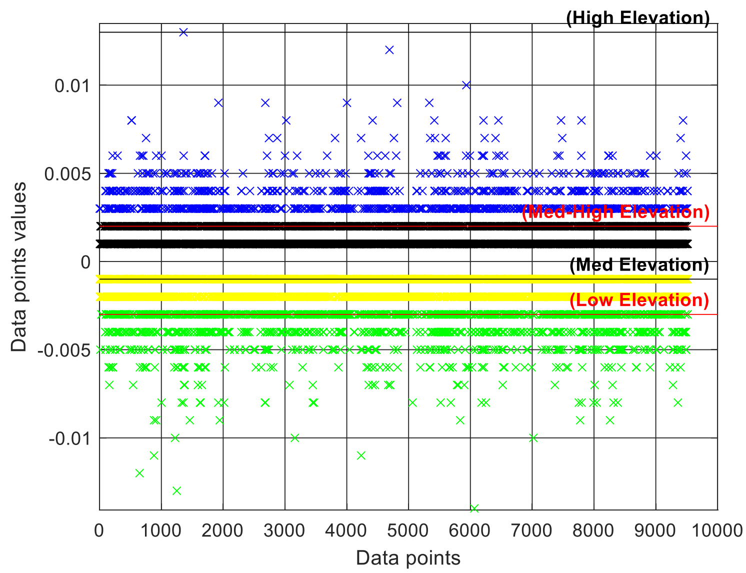


Figure 5-14: Clustering of output space to generate a frame of reference with the linguistic descriptors (*Low*, *Med*, *Med-high*, *High*).

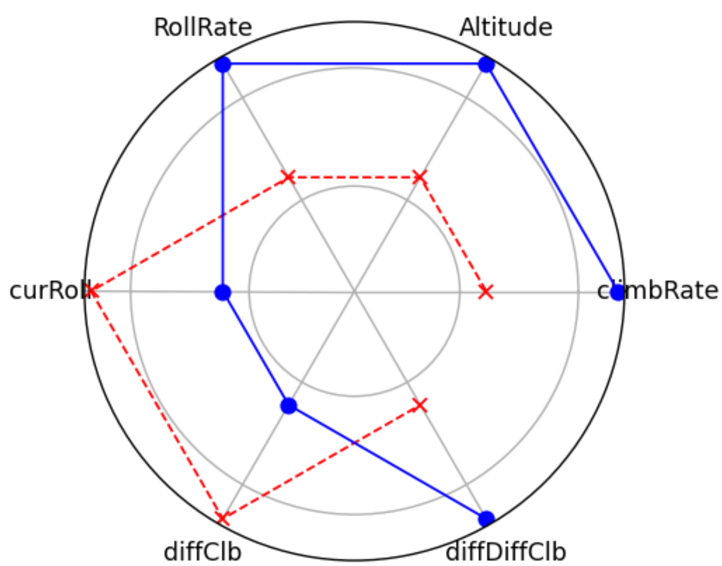


Figure 5-15: Qualitative Description of (*Med elevation*). Knowledge (Interpretations) is easily extracted from the figure, where each line represents a prototype.

The construction of a logic network of the frame of reference ensues. In a similar procedure, several numbers of clusters were constructed for each variable to obtain insight into which input variables have a higher impact on a specific object. Finally, the logic network reveals which concept is more significant than the others. The most significant reference information granule that describes the output space has the following interpretation:

Med-Elevation = [(climbRate is High) AND (Altitude is High) AND ( ) AND (curRoll is Low) AND (diffClb is Low) AND (diffDiffClb is High)]

OR

[(climbRate is Low) AND (Altitude is Low) AND (RollRate is Low) AND ( ) AND(diffClb is High) AND ( )]

The following logic expression delivers a final interpretation of the data. Note the ordering of logical terms which represents terms significance represented by font size.

Output Space = (*Med-Elevation*) OR (*Med-Low Elevation*) OR (*Low-Elevation*) OR (Hi-Elevation)

5.5 Summary

In summary, a conceptual interpretation framework to quantify datasets and extract useful knowledge through the utilization of conditional fuzzy clustering and logic networks is presented. Referential information granules are focal pieces of knowledge in the design, while the description of such information granules is realized in terms of other independent variables. The design exercises a direction-sensitive process by starting from the output space to form the relations with the descriptive input variables and construct logic networks with the aid of AND/OR neurons. This characteristic gives the design the ability to reveal all inner and yet hidden knowledge. Hence, the model is parametrically flexible due to the nature of logic networks. Outcomes of the design are expressed in terms of linguistic terms and logic expressions that are readable and understandable.

The process of concept formation and knowledge extraction involves several major activities summarized in Figure 5-1:

- Knowledge acquisition: it involves the acquisition of knowledge from human experts, sensors, or computer systems.
- Knowledge representation: The acquired knowledge is organized into a knowledge representation. This activity involves preparation of a knowledge map through the prototypes construction.
- Interpretation and justification: This involves the design of an interpretation capability to answer questions like why and how a specific piece of information was obtained.

Although the work presents a promising design to extract useful knowledge in the form of concept, we believe the design can be improved further by improving the clustering mechanism. In the following chapters, we will introduce an improved conditional clustering mechanism which in return should improve knowledge extraction and interpretation of dataset.

6. Enhancement of Fuzzy Clustering and Interpretation of Knowledge Domain

Fuzzy clustering, as an essential tool in granular modelling, helps to convert numeric data into information granules regardless of some possible hints of knowledge domain. Clustering, as a basic algorithmic framework, operates on a single data set and concerns with building partitions of data sets on the basis of some performance index. It is therefore, a data-driven analytical approach that revolves around available data aimed at revealing a structure in data and it is prerequisite for a slew of detailed algorithms. This clustering mechanism is conceptually sound and algorithmically appealing vehicle to construct fuzzy sets by determining their membership grades (functions).

Generally speaking, the generic fuzzy clustering, Fuzzy C-Means, forms data structures based on the concepts of closeness (resemblance). Typically, clustering dose not distinguish between input and output variables. So, the final product of cluster analysis results in a collection of clusters as concise descriptors of data. The generic fuzzy clustering can be referred to as condition-free clustering in which clustering can be formulated as follows:

Determine structure in data **X**

Most practical models are directional constructs that represents a certain mapping from independent to dependent variables. Thus, the role of variables (either inputs or outputs) is critical in constructing interpretable models. Conditional clustering develops clusters by focusing on a certain portion of the original data which adds directional features to the process. This gives rise to the decomposition of clustering and helps carry out more focused data analysis. The quest for conditional clustering can be formulated in a uniform manner as follows:

Determine structure in data **X** under condition (context) *D*

In this case, the clustering algorithm focuses only on a subset of X conditioned by the context D (Figure 6-1).

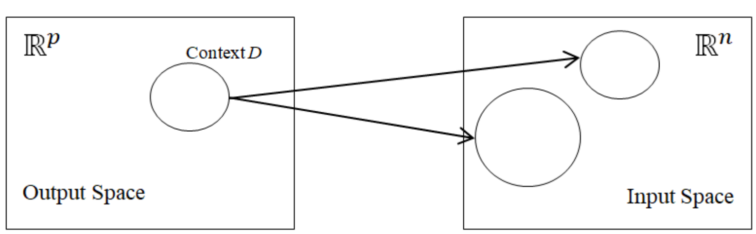


Figure 6-1: Conditional fuzzy clustering operates on a subset of data.

As you can see in Figure 6-1, other regions of output space are not considered when forming clusters in the input space. As a result, it may not be possible to reveal some important hidden data structures. This leads us to reconstruct the clustering mechanism in order to obtain a better interpretation framework of data.

In this chapter, we study the conditional fuzzy clustering with the attention to enhance the clustering mechanism to improve the process of building knowledge entities. Although conditional fuzzy clustering is capable of describe ambiguity in data which result in better interpretation, the fact that it focus only on one context limits its efficient functionality in revealing data patterns. The presented algorithm takes into consideration all possible contexts when clustering input space in either user-centric mode or data-driven mode. Furthermore, the improved conditional clustering mechanism could be augmented with logic-oriented fuzzy neural network to guide knowledge discovery in datasets.

6.1 Problem Formulation: Enhancement on Conditional Fuzzy Clustering

Most studies rely on fuzzy clustering when constructing linguistic models to express the uncertainty in the classification and the empirical complexity. Conditional clustering operates on a subset of data and neglects the effects of other parts (context) of output space which certainly does not reflect the essence of knowledge domain. For instance, if we are interested in modeling a phenomenon of *low* inflation rate on housing prices, we would define a number of fuzzy sets of context focused on low values of inflation rate. Subsequently, the clustering is only focused on the pertinent portion of data in the input space. In this case, a question that one might ask, what would clusters look like if we are interested in modeling *low* and *high* inflation rates on housing prices at the same time. The clustering has to be customizable by identifying pursuits of fuzzy clustering on the pertinent portion of data in the input space.

In an enhanced version of conditional clustering, we regard several contexts simultaneously to decompose the problem. Given several contexts (conditions), each context is mapped to its corresponding subsets of **X** while taken into consideration the effects of other contexts on the formation of prototypes in the input space. Here, the task can be reformulated, as opposed to conditional clustering, in the following way:

Determine structures in data **X** under conditions (contexts) D_1 AND D_2 AND, …, AND D_p

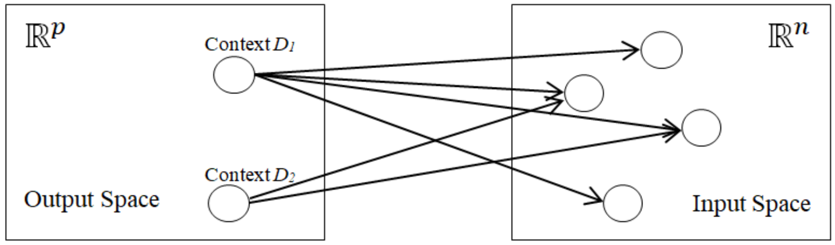


Figure 6-2: A detailed view at the enhanced conditional clustering in case of two contexts

6.2 Mathematical Model of Enhanced Conditional Clustering

This method comes as a modification of the conditional fuzzy clustering (CFCM) which is also a modification of the generic version of Fuzzy C-Means. In general, fuzzy clustering tries to minimize an objective function through iterative mechanism. Let's recall the objective function encountered in FCM:

$$Q = \sum_{i=1}^{c}\sum_{k=1}^{N} u_{ik}^{p}\|x_k - v_i\|^2 \tag{6.1}$$

where c stands for the number of clusters, U=[u_{ik}] is a partition matrix, $p > 1$ is the fuzzification coefficient expresses the impact of the membership grades on the individual clusters and implies the shape of membership function, u_{ik} is the membership degree of data $\mathbf{x}_k$ to the ith cluster and $\|.\|$ is a distance function between $\boldsymbol{x}_k$ and the prototype ($\boldsymbol{v}_i$) of the ith cluster. Here, the performance index (Q) is minimized by modifying the values of clusters, viz prototypes and a partition matrix.

The proposed fuzzy clustering mechanism modifies the objective function to include all porotypes that reside in all clusters. Formally speaking, a dataset composed of pairs ($\boldsymbol{x}_k$, $\boldsymbol{y}_k$), k =1, 2, …, N where $\boldsymbol{x}_k \in \mathbb{R}^n$and $y_k \in \mathbb{R}^p$. Several reference information granules D_1, D_2, …, D_p are defined in $\mathbb{R}^p$ and constraint the process of building clusters in $\mathbb{R}^n$. In this work, we adhere to the practice of using square brackets to identify a certain context such that [ii] denotes the iith context where ii = 1, 2, …, p. The objective function for the iith context, D[ii], is minimized as follows:

$$Q[ii] = \sum_{i=1}^{c}\sum_{k=1}^{N} u_{ik}^{p}[ii]\|x_k - v_i[ii]\|^2 - \alpha \sum_{ii=1}^{p}\sum_{j=1}^{c}\sum_{i=1}^{c}\sum_{k=1}^{N} u_{ik}^{p}[ii]\left\|v_j[jj] - v_i[ii]\right\|^2 \tag{6.2}$$

where $v_i[ii]$and $v_j[jj]$ stands for the iith, jjth contexts respectively, α is a problem-specific learning parameter that needs to tuned. Here, the objective function includes the distance between any prototype $v_i[ii]$ of the current context $D[ii]$ and all other prototypes in different context $D[jj]$ which is guided by the learning parameter α. The optimization of $Q\ [ii]$ completed with respect to the membership matrix $U\ [ii]$ and the prototypes $\boldsymbol{v}_1\ [ii]\ \ldots,\ \boldsymbol{v}_c\ [ii]$ indicates that there are two individual optimization tasks to be carried out separately for the partition matrix and the prototypes. Firstly, we concentrate on optimizing the following:

$$\text{Minimize } Q\ [ii]$$

$$\text{with respect to } V\ \{v_1[ii] \ldots v_c[ii]\} \in \mathbb{R}^n$$

Determining the prototypes is impacted by the selection of the distance function. We consider the Euclidean distance that is $\|x_k - v_i[ii]\|^2 - \sum_{s=1}^{n}(x_{ks} - v_{is}[ii])^2$. Now the optimization of prototypes $v_i\ [v_{il}]$, $l = 1,2, \ldots n$, reads as:

$$Q[ii] = \sum_{k=1}^{N} u_{ik}^2[ii] \sum_{s=1}^{n}(x_{ks} - v_{is}[ii])^2 - \alpha \sum_{ii=1}^{p}\sum_{i=1}^{c}\sum_{k=1}^{c} u_{ik}^2\,[ii] \sum_{s=1}^{n}\left(v_{js}[ii] - v_{is}[ii]\right)^2 \tag{6.3}$$

The gradient of the objective function with respect to v_{il}

$$\frac{\partial Q}{\partial v_{il}} = \sum_{k=1}^{N} u_{ik}^2[ii] \frac{\partial}{\partial v_{il}[ii]} \sum_{s=1}^{n} \frac{(x_{ks} - v_{is}[ii])^2}{\partial s^2} - \alpha \sum_{ii=1}^{p}\sum_{i=1}^{c}\sum_{k=1}^{c} u_{ik}^2\,[ii] \frac{\partial}{\partial v_{il}[ii]} \sum_{s=1}^{n} \frac{\left(v_{js}[ii] - v_{is}[ii]\right)^2}{\partial s^2} \tag{6.4}$$

$$\frac{\partial}{\partial v_{il}[ii]} \sum_{s=1}^{n} \frac{(x_{ks} - v_{is}[ii])^2}{\partial s^2} = 2\frac{x_{kl} - v_{il}[ii]}{\partial l^2} \tag{6.5}$$

$$\frac{\partial}{\partial v_{il}[ii]} \sum_{s=1}^{n} \frac{\left(v_{js}[jj] - v_{is}[ii]\right)^2}{\partial s^2} = -2\,\frac{(v_{jl}[jj] - v_{il}[ii])}{\partial l^2} \tag{6.6}$$

To achieve the minimum of V for $k = 1, 2, \ldots, N$, we set the following conditions:

$$\frac{\partial Q[ii]}{\partial v_{il}[ii]} = 0$$

$$\sum_{k=1}^{N} u_{ik}^2(x_{kl} - v_{il}[ii]) = \alpha \sum_{k=1}^{N}\sum_{ii=1}^{p}\sum_{j=1}^{c} u_{ik}^2[ii](v_{il}[jj] - v_{il}[ii]) \tag{6.7}$$

We arrive to the final expression:

$$\sum_{k=1}^{N} u_{ik}^{m}[ii]x_{kl} - v_{il}[ii]\sum_{k=1}^{N} u_{ik}^{2}[ii] = \alpha\sum_{k=1}^{N}\sum_{ii=1}^{p}\sum_{j=1}^{c} u_{ik}^{2}[ii]v_{jl}[ii] - \alpha v_{il}[ii]\sum_{k=1}^{N}\sum_{ii=1}^{p}\sum_{i=1}^{c} u_{ik}^{2}[ii] \tag{6.8}$$

Setting up auxiliary expressions for readability purposes, we arranges the following expressions

$$A_{il[ii]} = \sum_{k=1}^{N} u_{ik}^{2} x_{kl} \tag{6.9}$$

$$B_{i}[ii] = \sum_{k=1}^{N} u_{ik}^{m}[ii] \tag{6.10}$$

$$D_{il}[ii] = \sum_{k=1}^{N}\sum_{ii=1}^{p}\sum_{j=1}^{c} u_{ik}^{m} v_{jl}[ii] \tag{6.11}$$

$$F_{il}[ii] = \sum_{k=1}^{N}\sum_{ii=1}^{p}\sum_{i=1}^{c} u_{ik}^{m}[ii] \tag{6.12}$$

The prototypes are computed as:

$$v_{il}[ii] = \frac{A_{il}[ii] - \alpha D_{il}[ii]}{B_{i}[ii] - \alpha F_{il}[ii]} \tag{6.13}$$

To produce a concise form of calculations of prototypes, we plug in (6.9), (6.10), (6.11) and (6.12) in (6.13):

$$v_{il}[ii] = \frac{\sum_{k=1}^{N} u_{ik}^{2}[ii]x_{kl} - \alpha \sum_{k=1}^{N}\sum_{ii=1}^{p}\sum_{j=1}^{c} u_{ik}^{m}[ii]\, v_{jl}[ii]}{\sum_{k=1}^{N} u_{ik}^{m}[ii] - \alpha \sum_{k=1}^{N}\sum_{ii=1}^{p}\sum_{i=1}^{c} u_{ik}^{m}[ii]} \tag{6.14}$$

The second optimization task involves the optimization of partition matrix. We introduce Lagrange multiplier μ to transform the problem with constraint into its constraint-free version. We form an augmented objective function formulated for each data point

$$V[ii] = \sum_{i=1}^{c} u_{ik}^{2}[ii]d_{ik}^{2} - \alpha\sum_{ii=1}^{p}\sum_{j=1}^{c}\sum_{i=1}^{c} u_{ik}^{2}[ii]F_{ii,jj,i,j} + \mu \sum_{i=k}^{c} u_{ik}[ii] - G_{k}[ii] \tag{6.15}$$

where

$$d_{ik}^{2} = \|x_{k} - \boldsymbol{v}_{i}[ii]\|^{2}$$

$$F_{ii,jj,i,j}[ii] = \|\boldsymbol{v}_{j}[jj] - \boldsymbol{v}_{i}[ii]\|^{2}$$

We set the following condition:

$$\frac{\partial V[ii]}{\partial u_{st}[ii]} = 0$$

where s =1, 2, …, c, t =1, 2, …, N

Now, we calculate the partial derivative of V with respect to the elements of the partition matrix in the following way:

$$\frac{\partial V[ii]}{\partial u_{st}[ii]} = 2\, u_{st}[ii] d_{st}^2 - 2\, \alpha \sum_{jj=1}^{p} \sum_{j=1}^{c} u_{st}[ii]\, f_{ii,jj\,,s\,,j}\,[ii] + \mu \tag{6.16}$$

$$-\mu = u_{st}^2[ii](d_{st}^2 - \alpha \sum_{jj=1}^{p} \sum_{j=1}^{c} f_{ii,jj\,,s\,,t}\,[ii]) \tag{6.17}$$

The membership grade u_{st} is equal to

$$u_{st}[ii] = -\mu \frac{1}{2d_{st}^2 - 2\, \alpha \sum_{jj=1}^{p} \sum_{j=1}^{c} f_{ii,jj\,,s\,,j}\,[ii]} \tag{6.18}$$

We introduce the term $P_{st}[ii]$ for better readability of the equations

$$2P_{st}[ii] = 2d_{st}^2 - 2\, \alpha \sum_{jj=1}^{p} \sum_{j=1}^{c} f_{ii,jj\,,s\,,j}\,[ii] \tag{6.19}$$

Given the requirement$\sum_{s=1}^{c} u_{st}[ii] = G_t[ii]$, and plug in it in (6.18)

$$-\mu \sum_{s=1}^{c} \frac{1}{P_{st}[ii]} = G_t[ii] \tag{6.20}$$

Now, plug in (6.20) in (6.17) the above expression

$$2\, u_{st}[ii] P_{st}[ii] \frac{1}{P_{st}[ii]} = G_t[ii] \tag{6.21}$$

We obtain the entries of the partition matrix

$$u_{ik}[ii] = \frac{G_t[ii]}{\sum_{l=1}^{c} \frac{P_{ik}[ii]}{P_{lk}[ii]}} \tag{6.22}$$

It is evident that the computing of the partition matrix and the prototypes is intertwined: to calculate the partition matrix, we need the prototype and vice versa. Overall, the algorithm is completed through a sequence of iterations involving (6.14) and (6.22) where we start from some random allocation of data to clusters (a certain randomly initialized partition matrix) and carry out the following updates by adjusting the values of the partition matrix and the prototypes. In summary, the overall iterative scheme is outlined as follows:

- Define contexts $G[1]$, $G[2]$, …, $G[p]$
- Randomly initialize partition matrix $U[1]$, $U[2]$, …, $U[p]$
    - Repeat
        - $ii = 1, \dots p$
        - update prototypes using Equation (6.14).
        - update partition matrix Equation (6.22).
        - Calculate performance index $Q[1]$, $Q[2]$, …, $Q[p]$ (Equation (6.2)).
    - until a certain termination criterion has been satisfied.

The iterative process improves interpretation and knowledge representation as it tries to find the optimal prototypes locations. Note that, the learning parameter α is significant in the clustering process. It controls the interactions between prototypes from different contexts. Therefore, this parameter should be tuned till prototypes are fully optimized.

6.3 Numerical Studies

Two experiments are reported in this section to illustrate the design process of the enhanced conditional clustering. The first one shows the details of the algorithm, whereas the second example alludes to linguistic data mining of multidimensional data.

Example 1. A two-dimensional synthetic dataset consisting of 600 points is randomly generated by using a normal distribution with 6 clusters.

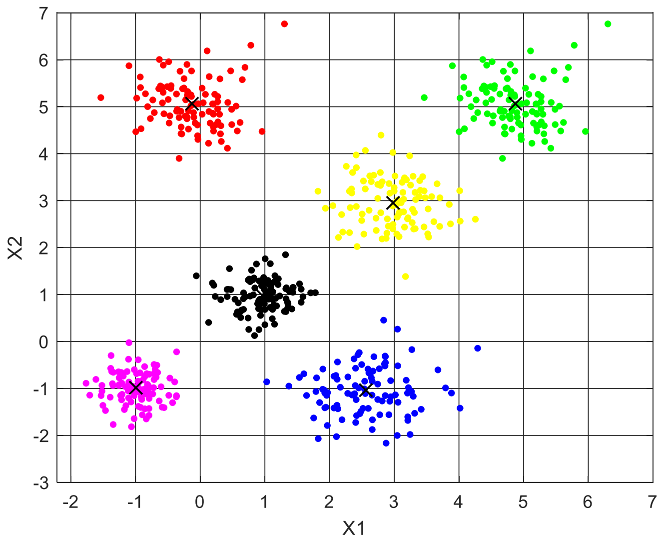


Figure 6-3: Two dimensional data set consisting 6 clusters.

Outputs were separately generated using the distribution N(μ = 2, σ = 0.18) and N(μ = 4, σ = 0.15). Two contexts were generated (D_1 and D_2) following Gaussian membership (Figure 6-4):

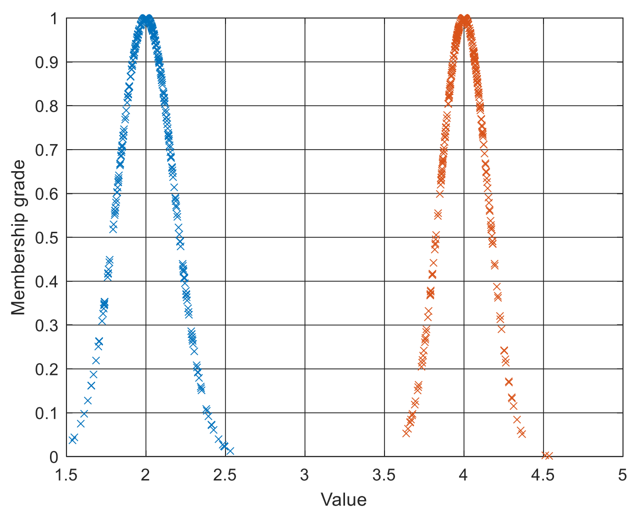


Figure 6-4: Representation of the contexts D_1 and D_2

Testing cases

The presented clustering algorithm is implemented for different values of α. This parameter is problem-dependent that effects the data structure. However, we assume that α = 0.05 will show the effects of the algorithm on the generated clusters. For better visualization of the algorithm, we use α = 0.06 and 0.03

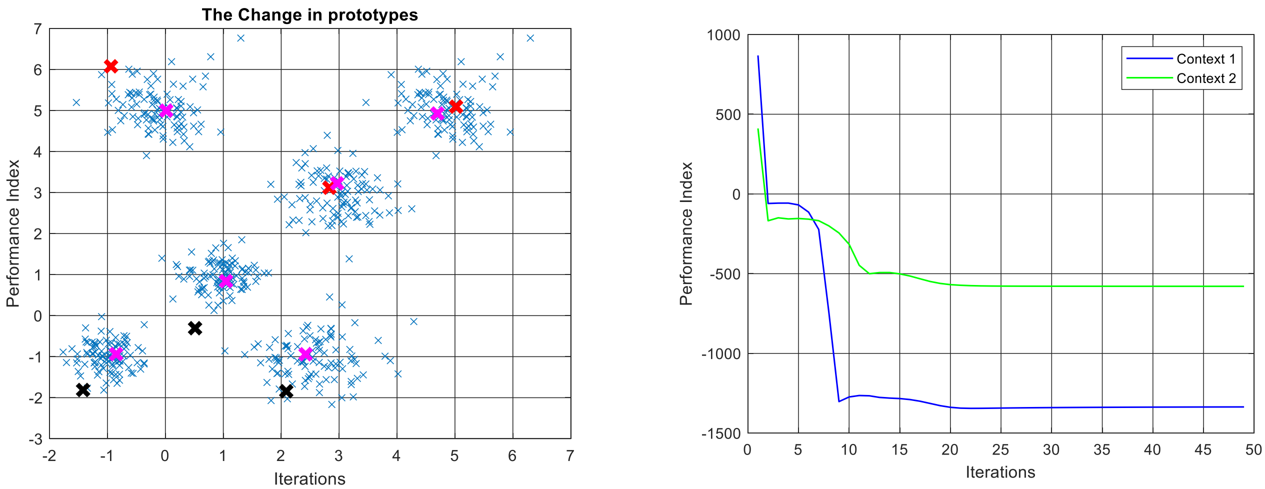
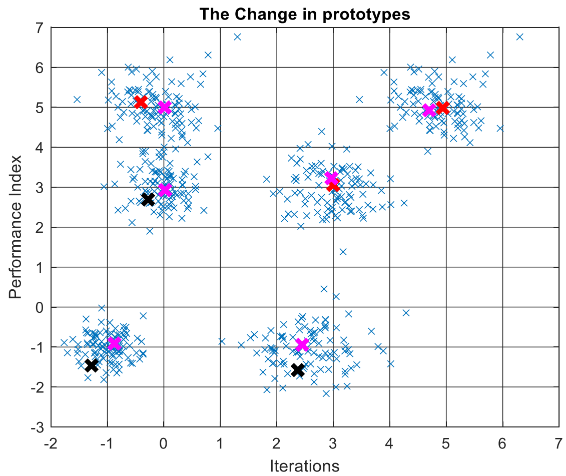


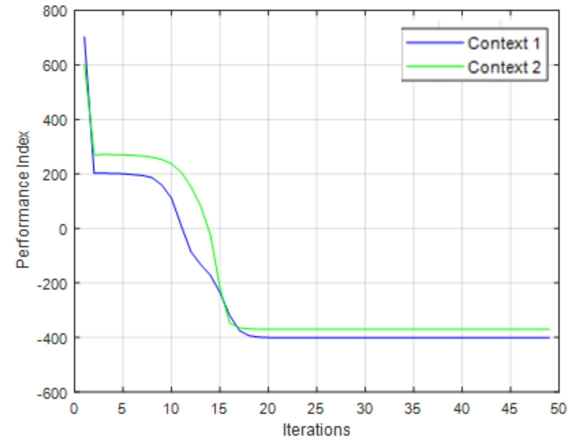
Figure 6-5: (Left) Results of multi-conditional fuzzy clustering into three clusters (α = 0.06). (Right) The performance index. Note that: Pink color indicates the prototypes without conditioning effect. Black\Red color of the prototypes indicate the new location after conditioning effects.

Table 6-1: The effects of the proposed fuzzy clustering method. The results show the change in prototypes positions after implementing multi-condition FCM. α = 0.06, c = 2, p =2

Multi-conditional clustering with *simultaneous* condition effects			
Context 1	prototype 1	-0.8271	5.8427
	prototype 2	2.9370	2.9603
	prototype 3	5.0816	4.9355
Context 2	prototype 1	2.1427	-2.0867
	prototype 2	-1.5705	-1.9911
	prototype 3	-0.5194	2.2883
Fuzzy clustering with *separate* condition effects			
Context 1	prototype 1	0.0144	4.9964
	prototype 2	2.9661	3.225
	prototype 3	4.6978	4.9262
Context 2	prototype 1	2.449	-0.9447
	prototype 2	-0.8758	-0.9128
	prototype 3	0.0214	2.929



(a)



(b)

Figure 6-6: (Left) Results of multi-conditional fuzzy clustering into three clusters (α = 0.03). (Right) The performance index. Note that: Pink color indicates the prototypes without conditioning effect. Black\Red color of the prototypes indicate the new location after conditioning effects.

Table 6-2: The effects of the proposed conditional clustering. The results show the change in prototypes positions after implementing multi-condition FCM. α = 0.03, c = 2, p =2

Multi-conditional clustering with *simultaneous* condition effects			
Context 1	prototype 1	-0.4065	5.1286
	prototype 2	2.9977	3.056
	prototype 3	4.94	4.984
Context 2	prototype 1	2.3713	-1.5769
	prototype 2	-1.2812	-1.4627
	prototype 3	-0.2794	2.6929
Fuzzy clustering with *separate* condition effects			
Context 1	prototype 1	0.0144	4.9964
	prototype 2	2.9661	3.225
	prototype 3	4.6978	4.9262
Context 2	prototype 1	2.449	-0.9447
	prototype 2	-0.8758	-0.9128
	prototype 3	0.0214	2.929

The clustering reveals different data structures that exhibit different organization of the patterns. The effects of clustering are shown explicitly in the Figures 6-1 and 6-2. The effects, which depend of the value of α, are maximal when new prototypes locations move to the boundaries of clusters (Figure 6-5).

Example 2.

Now let's consider the New York air quality measurements dataset [180]. A full description is provided in (Appendix B). The dataset is characterized by the four-dimensional feature vector:

ozone - mean ozone level (parts per billion),

solar - solar radiation,

wind - average wind speed,

temperature - maximal daily temperature.

We will treat *temperature* as the *context* then we will generate three linguistic terms modelled through the triangular membership function. The enhanced conditional clustering is carried out for the feature space composed of the three features (*ozone, solar* and *wind*)

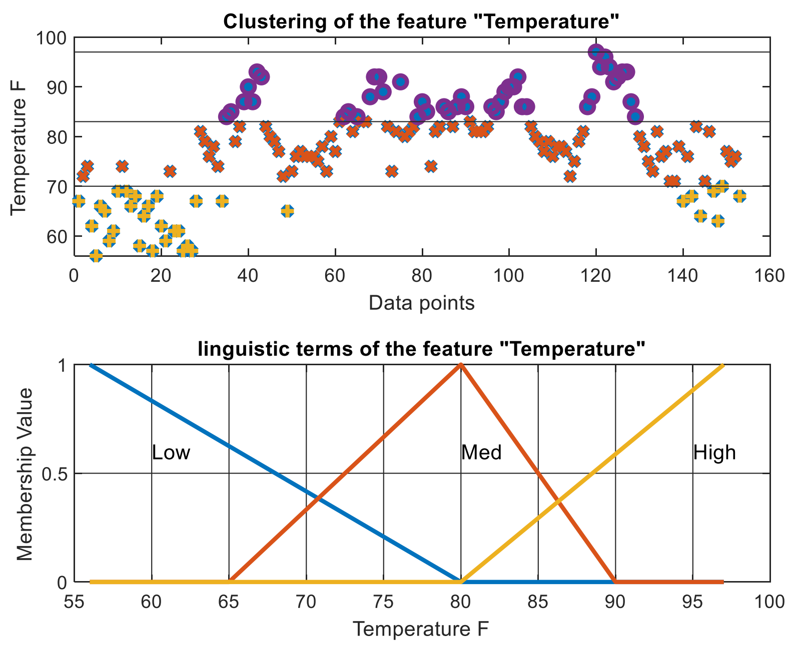


Figure 6-7: Visualization of contexts generation. Linguistic terms were given to each context.

For the comparison purposes, we report the results of clustering in Tables 6-3 and 6-4 which shows the difference if we apply the condition separately or simultaneously. Results show that the proposed method exhibits a visible impact on the locations of the prototypes.

Table 6-3: The effects of the proposed conditional clustering. The results show the change in prototypes positions after implementing multi-condition FCM. α = 0.06, c = 2, p =2

Conditional Fuzzy clustering with *separate* condition effects				
Context 1	prototype 1	38.09412	242.2084	10.3784
Low	prototype 2	29.6264	112.6668	9.5844
Context 2	prototype 1	15.6291	59.9641	13.2491
Med	prototype 2	23.4303	270.7182	11.8380
Multi-conditional clustering with *simultaneous* condition effects				
Context 1	prototype 1	36.2795	225.9680	8.7966
Low	prototype 2	26.8374	72.9598	8.4079
Context 2	prototype 1	9.7887	17.9727	12.1039
Med	prototype 2	20.3820	266.5977	10.7885

Table 6-4: The effects of the proposed conditional clustering. The results show the change in prototypes positions after implementing multi-condition FCM. α = 0.03, c = 2, p =2

Conditional Fuzzy clustering with *separate* condition effects				
Context 1	prototype 1	38.09412	242.2084	10.3784
Low	prototype 2	29.6264	112.6668	9.5844
Context 2	prototype 1	15.6291	59.9641	13.2491
Med	prototype 2	23.4303	270.7182	11.8380
Multi-conditional clustering with *simultaneous* condition effects				
Context 1	prototype 1	37.1232	234.3362	9.5663
Low	prototype 2	28.0671	91.9237	8.9499
Context 2	prototype 1	12.5903	37.9606	12.5688
Med	prototype 2	21.8769	268.4123	11.2730

The simultaneous conditioning exhibits a visible impact on the location of the prototypes. Consider two classes, c = 2. With only the effect of one condition, one of the prototypes ([15.6291 59.9641 13.2491]) comes as an effect of the condition patterns and thereafter gets moved to a different location as an effect of simultaneous conditioning which result a different data structure.

Interpretation mechanism differs from the generic conditional clustering to reflect the added condition to the conditional clustering. In the proposed algorithm, a designer is giver p number of conditions to be applied the dataset. Therefore, it is articulated to reveal dependencies as follows:

Determine structures in data **X** given D_1 is *Low* AND D_2 is *Med*

where *low* and *Med* are fuzzy sets with linguistic descriptions, **X** is a set of inputs in the feature space. Note that, data structures changes as results of adding more condition to reflect some linguistic terms of interest defined for each variable.

6.4 Summary

In this chapter, we extend the generalization of conditional fuzzy clustering to involve several conditions to guide clustering mechanism in a concurrent way. In a direction-sensitive process, the proposed clustering method starts from the predefined information granules in the output space to form the relations with the descriptive input variables. The proposed method improves the process of revealing data structures and hence improves interpretation of dataset. Moreover, of considering several regions of output space in direction-sensitive clustering will certainly help in eliminating vagueness in linguistic models especially when implemented in multi disciplines environment that requires interpreting several concepts jointly and cohesively.

The proposed method can be further augmented by utilizing logic-oriented fuzzy networks to extract quantitative descriptions with the effects of multiple conditions that coming from the fuzzy clustering.

7. Conclusions and Future Studies

This thesis presented several methodologies to augment interpretability of datasets through the utilization of logic-oriented fuzzy neural networks. The primary objective was to build a cohesive framework capable of coping with contradictory requirements of fuzzy modelling. To achieve this objective along with other objectives, state-of-the-art logic-oriented fuzzy components were utilized; also, various analytical methods were proposed to extract useful knowledge, quantify the extracted knowledge and present semantically sound linguistic expressions. Consequently, vagueness is significantly reduced from datasets by presenting knowledge domain linguistically that is easily comprehended by the end user.

Starting off with a comprehensive review of all related logic-oriented fuzzy components to re-assess the design and reach an optimal construct, the rate of convergence of the AND\OR architecture was noticeably improved by not relying on gradient-based learning. The presented two-phase learning scheme reduced the computation overhead by implementing a randomized learning technique. Optimization algorithms were not heavily utilized in this work to avoid extensive parameters alternation for each application. Furthermore, to reach optimally good approximation capabilities along with relevant interpretations, several optimizers were tested on different numeric and granule datasets. The results were employed throughout the dissertation to ensure better experimental evidences.

Interpretability, also explainable AI, stands as a core objective in modelling trustworthy intelligent systems. In this regard, we discussed the interpretability of formal concept analysis in terms of their capability in knowledge extraction and representation. Although concept analysis offers interesting knowledge representation, it lacks several characteristics that make it optimally interpretable such casualty and linguistic interpretation. Therefore, we presented an interpretable framework of data that utilizes directional fuzzy clustering along with a supervised learning scheme. The design exercises a direction-sensitive process by starting from the output space to form the relations with the descriptive input variables and construct logic networks with the aid of AND/OR neurons. This characteristic gives the design the ability to reveal all inner and yet hidden fuzzy relations. Hence, the model is parametrically flexible due to the nature of logic networks. We also presented a modified way to visualize fuzzy rules (or logic expressions) for the purpose of simplicity and better concept representation. The visualization of logic expressions takes advantages of underlying logic processing of fuzzy AND\OR neurons.

Apparently, connectives of the logic network are given weighted values before producing the final result. These values are used in an automated fashion to generate logic expressions, similar to bag-of-word model used in natural language processing.

Additionally, we studied conditional fuzzy clustering with the attention to enhance the clustering mechanism to improve the process of building knowledge entities. Evidently, conditional clustering focuses only on one context which limits its efficient functionality in revealing data patterns. Eventually, the clustering is only focused on the pertinent portion of data in the input space. We presented an innovative version of conditional clustering that takes into consideration all possible contexts, from the output space, when clustering input space. The clustering is linguistically customizable by identifying pursuits of fuzzy sets on the pertinent portion of data in the input space. The most robust aspect of the presented method is the underlying interpretation capabilities in which a designer takes controllability of input-output spaces by considering criteria of coverage and specificity. If the specificity of a context in the output space is at minimum and the coverage is maximal then we end up with a generic fuzzy clustering. Increasing specificity identify some areas of interest leading to uncover some interesting relations between input-output spaces. It is intuitively apparent that these two characteristics are associated: the increase in one of then implies a decrease in another: an information granule that "covers" a lot of data cannot be overly specific and vice versa.

7.1 Research Limitations

Although the research outcomes are encouraging, some limitations have to be taken into considerations; a plethora of research have discussed interpretability of machine learning especially with the growth of models' complexity leading to extremely complex black-box functionality. As such, the functions used to make decisions may be too complex for humans to comprehend; and therefore, it may not be possible to completely understand the full decision-making rationale. It becomes an open question that at what extent a possible interpretation is capable of closing the expectation gap in machine learning. Because humans can handle at most 7 ± 2 cognitive entities at once, practitioners from different domains can't agree whether produced explanations actually useful or not. In an interdisciplinary working environment, more and different cognitive entities are required to cover a certain knowledge domain and satisfy different disciplines requirements. The proposed multi-conditional clustering might seem an ideal candidate to work in an overlapping conceptual structures leading to meeting interdisciplinary requirements.

Additionally, logic-oriented fuzzy networks requires tuning several parameters such as learning rate, number of neurons and the type of decomposition mode of Boolean representation. One might think of this as positive characteristics of the construct because of the added flexibility. However, this could lead to missing the optimal combination of parameters values. These reported limitations are starting points for further future investigations.

7.2 Future Research Directions

First and foremost, the presented multi-conditional fuzzy clustering algorithm has a potential to open up a new research direction for building more interpretation-dedicated linguistic models. The newly developed algorithm, which is geared towards forming direction-sensitive information granules of fuzzy clusters, should reveal hidden knowledge for better data analysis in several domains of machine learning such as pattern recognition and classification.

There are other interesting venues deserve further study directly linked with the undertaken research. The well-known interpretability-accuracy trade-off is worth more investigation. Given what has been said in this thesis, the ability to interpret, then reprocess, dataset could lead to improvements in performance, which could be done through an iterative process. Even though, all recent publications imply, implicitly or explicitly, that the trade-off between interpretability and accuracy generally exists, iterative knowledge discovery could lead to some sort of interpretability that can be imbued directly into models without losing accuracy.

Although we intentionally overlooked optimization algorithms in this work, there is a potential of integrating optimization algorithms in the interpretation framework of concept presented in Chapter 5. This could be done in a preprocessing stage when constructing information granular. The results could have positive impacts on the presented multi-conditional clustering presented in Chapter 6 leading to optimized knowledge extraction process.

We envision from the presented work the following research directions as an extension of this thesis:

- Choosing a better representative level of information granular in the theoretical development of linguistic modeling.

- Optimization algorithms could improve the construction of referential information granules to extract the semantic meaning of the acquired knowledge.
- Coverage and specificity criteria could be further discussed in the context of multi-conditions fuzzy clustering. Both inputs and output spaces are a collection of several fuzzy sets and hence we need to search for the best level of granularity to avoid any detrimental effect on the fuzzy clusters and information granules being developed and optimized.
- Constructing and optimization logic-oriented networks in high-dimensionality problems without suffering from premature convergence and low optimization precision.
- Type-2 information granules have not been sufficiently studied in knowledge discovery and representation and are a good candidate for better concept interpretability.

References

[1] L. A. Zadeh, “Fuzzy sets”, In Information and control, vol. 8, no. 3, pp. 338-353, 1965.

[2] J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms. Springer-Verlag, Plenum Press, New York , 1981

[3] L.A. Zadek, Fuzzy sets and their applications to classification and clustering. In: J. van Ryzin, Ed., Classification and Clustering. Academic Press, New York, pp. 251-299.

[4] W. Pedrycz, “Conditional Fuzzy C-Means”, Pattern Recognition Letters, vol. 17, pp. 625-631, May 1996.

[5] J. Zhao, S. Xu, R. Wang, B. Zhang, G. Guo, D. Doermann, D. Sun, “Data-adaptive binary neural networks for efficient object detection and recognition,” In Pattern Recognition Letters, vol. 153, pp. 239- 245, 2022,

[6] Jie Sun, Abd AlRahman AlMomani, Erik Bollt, “Data-driven learning of Boolean networks and functions by optimal causation entropy principle”, In Patterns, vol. 3, no. 11, 2022,

[7] W. Pedrycz and F. Gomide, “Fuzzy Systems Engineering: Toward Human-Centric Computing,” Wiley-IEEE Press, 2007.

[8] W. Pedrycz, “Neurocomputations in relational systems,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, Issue 3, 1991, Pp. 289-297.

[9] M. Alateeq, W. Pedrycz, “A comparative analysis of bio-inspired optimization algorithms for automated test pattern generation in sequential circuits”, In Applied Soft Computing, vol. 101. 2021

[10] W. Pedrycz, K. Kwak, “Linguistic Models as a Framework of User-Centric System Modeling,” IEEE transactions on systems, man, and cybernetics—part a: systems and humans, vol. 36, no. 4, July 2006

[11] P. L. Gentili, “Establishing a New Link between Fuzzy Logic, Neuroscience, and Quantum Mechanics through Bayesian Probability: Perspectives in Artificial Intelligence and Unconventional Computing,” In Molecules, vol. 26, no. 19, 2021

[12] D. Dubois, H. Prade, “Possibility Theory and its Applications: a Retrospective and Prospective view,” In Decision Theory and Multi-Agent Planning. CISM International Centre for Mechanical Sciences, vol. 482, Springer, 2006

[13] H. Wang, Z. Xu, W. Pedrycz. “An overview on the roles of fuzzy set techniques in big data processing: Trends, challenges and opportunities”, In Knowledg-based Systems, vol. 118, pp. 15-30, 2017.

[14] M.J. Gacto, R. Alcalá, F. Herrera, “Interpretability of linguistic fuzzy rule-based systems: an overview of interpretability measures,” Information. Science, vol. 181, no. 20, pp. 4340–4360, 2011.

[15] J. C. Bezdek, R. Erhlich, W. Full, “FCM: The fuzzy c-means clustering algorithm,” Computers & Geoscience 10, Pp. 191–203, 1984.

[16] O. Cordon and F. Herrera, “A proposal for improving the accuracy of linguistic modeling,” IEEE Transactions on Fuzzy Systems, Vol. 8, No. 3, Pp. 335-344, Jun. 2000.

[17] O. Cordon, F. Herrera, and I. Zwir, “Linguistic modeling by hierarchical systems of linguistic rules,” IEEE Transactions on Fuzzy Systems, Vol. 10, No. 1, Pp. 2-20, Feb. 2002.

[18] C. Porcel, A. G. López-Herrera, and E. Herrera-Viedma, “A recommender system for research resources based on fuzzy linguistic modeling,” Expert Systems with Applications, Vol. 36, No. 3, part 1, Pp. 5173-5183, Apr. 2009.

[19] K. Trawinski, O. Cordon, L. Sanchez, and A. Quirin, “A Genetic Fuzzy Linguistic Combination Method for Fuzzy Rule-Based Multiclassifiers,” IEEE Transactions on Fuzzy Systems, Vol. 21, No. 5, Pp. 950-965, Oct. 2013.

[20] W. Pedrycz and A. V. Vasilakos, “Linguistic models and linguistic modeling,” IEEE Trans. Syst., Man, Cybern., vol. 29, no. 6, pp. 745–757, Dec. 1999.

[21] Y. Zhang, P. Tino, A. Leonardis, K. Tang, “A Survey on Neural Network Interpretability,” IEEE Transactions on Emeerging Topics in Computational Intelligence, Vol. 5, no. 5, Oct. 2021.

[22] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, “A survey of methods for explaining black box models,” ACM Comput. Surv., vol. 51, 2018.

[23] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, ““Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”, In Information Fusion, vol. 58, pp. 82-115, 2020

[24] K. Hirota and W. Pedrycz, "OR/AND neuron in modeling fuzzy set connectives," in *IEEE Transactions on Fuzzy Systems*, vol. 2, no. 2, pp. 151-161, May 1994.

[25] H. Ishibuchi, Y. Nojima, "Discussions on interpretability of fuzzy systems using simple examples", In Proceedings of IFSA-EUSFLAT World Congress, pp. 1649–1654, 2009

[26] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning—I,” In Information Sciences, vol. 8, no. 3, pp. 199-249, 1975

[27] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning—II,” In Information Sciences, vol. 8, no. 4, pp. 301-357, 1975

[28] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning—III,” In Information Sciences, vol. 9, no. 1, pp. 43-80, 1975

[29] Z. Sun, K.-F. Au, and T.-M. Choi, “A neuro–fuzzy inference system through integration of fuzzy logic and extreme learning machine,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 37, no. 5, pp. 1321–1331, Oct. 2007.

[30] S. Mitra and Y. Hayashi, “Neuro–fuzzy rule generation: Survey in soft computing framework,” IEEE Trans. Neural Netw., vol. 11, no. 3, pp. 748–768, May 2000

[31] W. Pedrycz, “Fuzzy neural networks and neurocomputations”, In Fuzzy Sets and Systems, vol. 56 no. 1, pp. 1-28, 1993.

[32] J. . -S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," in IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, no. 3, pp. 665-685, May-June 1993.

[33] D. Nauck, F. Klawon and R. Kruse, “Foundations of Neuro-Fuzzy Systems”, Hoboken, NJ, John Wiley & Sons, 1997.

[34] H. Wang, Z. Xu, W. Pedrycz. “An overview on the roles of fuzzy set techniques in big data processing: Trends, challenges and opportunities”, In Knowledg-based Systems, vol. 118, pp. 15-30, 2017.

[35] W. Pedrycz, “Computational Intelligence: An Introduction”, Boca Raton, FL, CRC Press, 1997.

[36] W. Pedrycz and F. Gomide, “An Introduction to Fuzzy Sets; Analysis and Design”, Cambridge, MA, MIT Press, 1998.

[37] W. Pedrycz, “Fuzzy Sets Engineering”, Boca Raton, FL, CRC Press, 1995.

[38] W. Pedrycz, M. Reformat, “Evolutionary Fuzzy Modeling”, IEEE Trans on Fuzzy Systems, vol. 11, pp. 652-665, 2003.

[39] I. Perfilieva, V. Novak, *Discovering the World with Fuzzy Logic*. 1st ed., Heidelberg , Germany, Physica-Verlag Publishers, Nov. 2000.

[40] W. Pedrycz, Fuzzy Modelling: Paradigms and Practice,1st ed., Kluwer Academic Publishers, 1996.

[41] A. B´ardossy, and L. Duckstein, Fuzzy rule-based modeling with application to geophysical, biological and engineering systems, 1st ed., USA, CRC Press, April 1995.

[42] Z. Chi, H. Yan, and T. Pham, Fuzzy algorithms: with applications to image processing and pattern recognition, Singapore, World Scientific Publishing, 1996

[43] K. Hirota, Industrial applications of fuzzy technology, Springer-Verlag Publication, 2012.

[44] V. G. Moudgal, W. A. Kwong, K. M. Passino and S. Yurkovich, "Fuzzy learning control for a flexible-link robot," in IEEE Transactions on Fuzzy Systems, vol. 3, no. 2, pp. 199-210, May 1995.

[45] R. Hecht-Nielsen, "Neurocomputing: picking the human brain," In IEEE Spectrum, vol. 25, no. 3, pp. 36-41, March 1988.

[46] W. Pedrycz, Knowledge-based Clustering, John Wiley & Sons, Hoboken, NJ, 2005.

[47] S. K. Pal, D. Mandal “Fuzzy Logic and Approximate Reasoning: An Overview,” In IETE Journal of Reseach, vol. 37, no 5-6, pp. 548-560, 2015

[48] S. O. Kuznetsov, “Machine learning and formal concept analysis,” in Proc. Int. Conf. Formal Concept Anal. (ICFCA), pp. 23–26, Feb. 2004.

[49] G. Bouritsas, P. Koutras, A. Zlatintsi, and P. Maragos, “Multimodal visual concept learning with weakly supervised techniques,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 4914–4923, Jun. 2018.

[50] Z. Y. Guan, L. J. Zhang, J. Y. Peng, and J. Fan, “Multi-view concept learning for data representation,” IEEE Trans. Knowl. Data Eng., vol. 27, no. 11, pp. 3016–3028, Nov. 2015.

[51] Y. Shi, Y. Mi, J. Li, W. Li, "Concurrent concept-cognitive learning model for classification," In *Information. Science*, pp. 65–81, vol. 496, 2019.

[52] A. Bargiela and W. Pedrycz, "Granular mappings," In IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 35, no. 2, pp. 292-297, March 2005.

[53] Z. Y. Guan, L. J. Zhang, J. Y. Peng, and J. Fan, “Multi-view concept learning for data representation,” IEEE Trans. Knowl. Data Eng., vol. 27, no. 11, pp. 3016–3028, Nov. 2015.

[54] M. Cintra, H. Camargo, M. Monard, “Genetic generation of fuzzy systems with rule extraction using formal concept analysis,” In Information Sciences, vol. 349-350, pp. 199-215, 2016.

[55] D. Angluin, “Queries and concept learning,” Mach. Learn.,vol.2, pp. 319–342, Apr. 1988.

[56] Y. Mi, Y. Shi, J. Li, W. Liu and M. Yan, "Fuzzy-Based Concept Learning Method: Exploiting Data With Fuzzy Conceptual Clustering," in IEEE Transactions on Cybernetics, vol. 52, no. 1, pp. 582-593, Jan. 2022,

[57] W. H. Xu and W. T. Li, “Granular computing approach to two-way learning based on formal concept analysis in fuzzy datasets,” IEEE Trans. Cybern., vol. 46, no. 2, pp. 366–379, Feb. 2016.

[58] G. F. Qiu, J. M. Ma, H. Z. Yang, and W. X. Zhang, “A mathematical model for concept granular computing systems,” Sci. China Inf. Sci., vol. 53, pp. 1397–1408, Jul. 2010.

[59] J. H. Li, C. L. Mei, W. H. Xu, and Y. Qian, “Concept learning via granular computing: A cognitive viewpoint,” Inf. Sci., vol. 298, pp. 447–467, Mar. 2015.

[60] R. Bˇelohlávek, V. Slenáˇr, and J. Zacpal, “Crisply generated fuzzy concepts,” in Proc. 3rd Int. Conf. Formal Concept Anal., pp. 269-284, France, 2005.

[61] A. Pérez-Suárez, J. F. Martínez-Trinidad, and J. A. Carrasco-Ochoa, “A review of conceptual clustering algorithms,” Artif. Intell. Rev., vol. 52, pp. 1267–1296, Aug. 2019.

[62] W. X. Zhang and W. H. Xu, “Cognitive model based on granular computing,” Chin. J. Eng. Math., vol. 24, pp. 957–971, Jun. 2007.

[63] E. Yan, C. Yu, L. Lu, W. Hong. C. Tang, “Incremental concept cognitive learning based on three-way partial order structure”, In Knowledge-based Systems, vol. 230, 2021.

[64] B. J. Fan, E. C. C. Tsang, W. H. Xu, D. Chen, and W. Li, “Attribute-oriented cognitive concept learning strategy: A multi-level method,” Int. J. Mach. Learn. Cybern., vol. 10, pp. 2421–2437, Sep. 2019.

[65] R. Shivhare and A. K. Cherukuri, “Three-way conceptual approach for cognitive memory functionalities,” Int. J. Mach. Learn. Cybern., vol. 8, pp. 21–34, Feb. 2017.

[66] R. Wille, “Restructuring lattice theory: An approach based on hierarchies of concepts,” in Ordered Sets, I. Rival, Ed., Dordecht, The Netherlands: Springer, pp. 445–470, Sep. 1982.

[67] L. Wang, X. Liu, “Concept analysis vis rough set and AFS algebra”, In Information Sciences, vol. 178, no. 21, pp. 4125, 4137, 2008

[68] J. Li, C. Mei, Y. Lv, “Incomplete decision contexts: approximate concept construction, rule acquisition and knowledge reduction”, In International Journal of Approximate Reasoning, vol. 54, no. 1, pp. 149-165, 2013

[69] W. Pedrycz and A. Rocha, “Fuzzy-set based models of neurons and knowledge-based networks”, IEEE Trans. Fuzzy Systems, vol. 1, no. 4, pp. 254-266, Nov. 1993.

[70] Y. Y. Yao, "Concept lattices in rough set theory," IEEE Annual Meeting of the Fuzzy Information, 2004. Processing NAFIPS '04., pp. 796-801 vol.2, 2004.

[71] JM. Ma, MJ. Cai, and CJ. Zou, “Concept acquisition approach of object-oriented concept lattices,” Int. Journal of Mach. Learn. & Cyber., vol. 8, pp. 123–134, 2017.

[72] N. Duntsch and G. Gediga, "Modal-style operators in qualitative data analysis," In Proceeding of 2002 IEEE International Conference on Data Mining, pp. 155-162, 2002.

[73] R. Bˇelohlavek, "Concept lattices and order in fuzzy logic", In Annals of Pure and Applied Logic, , vol. 128, no.1-3, pp. 277-298, 2004.

[74] E. Mamdani, S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller", International Journal of Man-Machine Studies, vol. 7, no.1, pp. 1–13, Jan. 1975

[75] T. Takagi, M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control”, IEEE Trans. on Systems, Man, and Cybernetics (SMC), vol. 15 no. 1, pp. 116–132, 1985.

[76] G. J. Klir and B. Yuan “Fuzzy Sets and Fuzzy Logic: Theory and Applications,” Prentice Hall PTR, New Jersey, 1995.

[77] B. Schweizer and A. Sklar, “Associative functions and statistical triangle inequalities,” Publ. Mathematical Debrecen, Issue 8, 1961, Pp. 313-334.

[78] W. Pedrycz, “neurcomputations in relational systems,” In IEEE Transactions on pattern Analysis and Machine Intelligence, vol. 13, pp. 289-296, 1991.

[79] M. Alateeq, W. Pedrycz, “Development of Two-phase Logic-oriented Fuzzy AND/OR Network”, In Neurocomputing, vol. 482, pp.129-138, 2022

[80] W. Pedrycz, “Heterogeneous Fuzzy Logic Networks: Fundamentals and Development Studies”, In IEEE Trans. On Neural Networks, vol. 15 no. 6 Nov. 2004

[81] W. Pedrycz, M. Reformat, K. Li, "OR/AND neurons and the development of interpretable logic models", IEEE Trans. Neural Networks. vol. 17, no. 3, pp. 636–658. 2006.

[82] W. Pedrycz, M. Reformat, “Evolutionary Fuzzy Modeling”, IEEE Trans on Fuzzy Systems, vol. 11, pp. 652-665, 2003.

[83] W. Pedrycz, M. Reformat, “Genetically optimized logic models”, Fuzzy Sets Systems, vol. 150, no. 2, pp. 351–371, March 2006.

[84] S. Scardapane, D. Wang, “Randomness in neural networks: an overview”, WIREs Data Mining and Knowledge Discovery, vol. 7, no. 2, 2017.

[85] A. F. Gobi, W. Pedrycz, “Fuzzy modelling through logic optimization”, Int. Journal of Approximate Reasoning, vol. 45, n. 3, pp. 488-510, Aug. 2007.

[86] A. F. Gobi, W. Pedrycz, “The potential of fuzzy neural networks in the realization of approximate reasoning engines”, In Fuzzy Sets and Systems, vol. 157, Issue 22, pp. 2954 – 2973, Nov. 2006.

[87] R. Dlugosz, W. Pedrycz, "Łukasiewicz fuzzy logic networks and their ultra low power hardware implementation", In, Neurocomputing, vol. 73, pp. 1222-1234, 2010.

[88] A.F. Gobi, W. Pedrycz, "Logic Minimization as an Efficient Means of Fuzzy Structure Discovery", IEEE Transactions on Fuzzy Systems, vol. 16, no. 3, June 2008.

[89] N. Retnakaran and N. J. Pizzi, "Biomedical pattern classification using an optimized fuzzy adaptive logic network," Canadian Conference on Electrical and Computer Engineering, pp. 382-385, 2005.

[90] N. J. Pizzi, A. Demko and W. Pedrycz, "Variance analysis and biomedical pattern classification," International Conference on Fuzzy Systems, 2010, pp. 1-8, 2010.

[91] W. Pedrycz, A. Breuer, N. Pizzi, "Fuzzy adaptive logic networks as hyvrid models of quantitative software engineering," In Intellgient automation and soft computing, vol. 12, no. 2, pp. 189-209, 2006.

[92] R. Al-Hmouz, W. Pedrycz, A. Balamash and A. Morfeq, "Logic-Oriented Autoencoders and Granular Logic Autoencoders: Developing Interpretable Data Representation," in IEEE Transactions on Fuzzy Systems, vol. 30, no. 3, pp. 869-877, March 2022.

[93] R. Bˇelohlavek, "Fuzzy Relational Systems: Foundations and Principles", IFSR International Series on Systems Science and Engineering, Dordrecht: Kluwer Academic Publishers, vol. 20, 2002.

[94] R. S. Michalski, "Knowledge acquisition through conceptual clustering: A theoretical framework and algorithm for partitioning data into con- junctive concepts," Int. J. Policy Anal. Inf. Syst., vol. 4, pp. 219–244, Mar. 1980.

[95] R. Wille, "Restructuring lattice theory: An approach based on hierarchies of concepts," in Ordered Sets, I. Rival, Ed., Dordecht, The Netherlands: Springer, pp. 445–470, Sep. 1982.

[96] J. Atif, C. Hudelot, I. Bloch, "Explanatory Reasoning for Image Understanding Using Formal Concept Analysis and Description Logics", IEEE Trnas. On Sys. Man and Cyberne. Sys., vol. 44, no. 4, 2014

[97] Y. Mi, Y. Shi, J. Li, W. Liu and M. Yan, "Fuzzy-Based Concept Learning Method: Exploiting Data With Fuzzy Conceptual Clustering," in IEEE Transactions on Cybernetics, vol. 52, no. 1, pp. 582-593, Jan. 2022,

[98] C. A. Kumar, S. C. Mouliswaran, P. Amriteya, and S. R. Arun, "Fuzzy formal concept analysis approach for information retrieval," in Proc. 5th Int. Conf. Fuzzy Neuro Computing (FANCCO), pp. 255–271, 2015.

[99] L.K. Guo, F.P. Huang, Q.G. Li, and G.Q. Zhang, "Power contexts and their concept lattices", In Discrete Mathematics, vol. 311, no. 18-19, pp. 2049–2063, 2011

[100] C. S. Lee, A. Lee. "Clinical applications of continual learning machine learning," In *The Lancet Digital Health*, vol. 2, no. 6, pp e279-e281, 2020.

[101] K. Shaheen, M. Hanif, O. Hasan, M. Shafique "Continual learning for real-world autonomus systems: algorithms, challenges and frameworks," *In Journal of Intelligent & Robotics Systems*, vol. 105, 2022.

[102] S. Grossberg, "Studies of mind and brain : neural principles of learning, percption, development, cogniton and motor control", 1982.

[103] L. Wang, X. Liu, "Concept analysis vis rough set and AFS algebra", In Information Sciences, vol. 178, no. 21, pp. 4125, 4137, 2008

[104] J.H. Li, C.C. Huang, W.H. Xu, et al., "Cognitive concept learning via granular computing for big data," In International Conference on Machine Learning and Cybernetics, ICMLC, pp. 289–294, 2015.

[105] A. Bargiela, W. Pedrycz, Granular Computing, An Introduction, Boston, MA, USA, Kluwer Academic Publications, 2003.

[106] J. H. Li, C. C. Huang, J. J. Qi, Y. Qian, and W. Liu, "Three-way cognitive concept learning via multi-granularity," Inf. Sci., vol. 378, pp. 244–263, Feb. 2017.

[107] J. Deogun, and J. Saguer, "Monotone concepts for formal concepts analysis", In Discrete Applied Mathematics, vol. 144, pp. 70-78, 2004.

[108] H. Cui, G. Yue, L. Zou, X. Liu, and A. Deng, "Multiple multidimensional linguistic reasoning algorithm based on property-oriented linguistic concept lattice", In International Journal of Approximate Reasoning, vol. 131, pp. 80-92, 2021

[109] X. Liu, W. Wang and T. Chai, "The Fuzzy Clustering Analysis Based on AFS Theory", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 35, no. 5, pp. 1013-1027, Oct. 2005.

[110] Y. Zhang, X. Liu and X. Wang, "A Novel Weighted Fuzzy Clustering Analysis Based on AFS Theory," 2009 Ninth International Conference on Hybrid Intelligent Systems, pp. 346-350, 2009.

[111] Y. Ren, M. Song, and X. Liu, "New approaches to the fuzzy clustering via AFS theory", International Journal of Information and Systems Science 3, pp. 307–325, 2007.

[112] W. Xu and W. Li, "Granular Computing Approach to Two-Way Learning Based on Formal Concept Analysis in Fuzzy Datasets," in IEEE Transactions on Cybernetics, vol. 46, no. 2, pp. 366-379, Feb. 2016.

[113] G. Qiu, J. Ma, H. Yang, and W. Zhang, "A mathematical model for concept granular computing systems", In Science China Information Sciences, vol. 53, pp. 1397–1408, 2010

[114] M. Jian-Min, "Concept Granular Computing Systems", In 2009 6th International Conf. on Fuzzy Sys. And Knowledge Disc., pp. 150-154, 2009

[115] S. Deng, "Clustering with Fuzzy C-means and Common Challenges," In Journal of Physics: Conference Series, vol. 1453, 2019.

[116] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," Journal of Cybernetics, vol. 3, no.3, pp. 32-57, 1974.

[117] J .C. Bezdek, "Cluster validity with fuzzy sets," Journal of Cybernetics, vol. 3, no. 3, pp. 58–72, 1973. DOI: 10.1080/01969727308546047.

[118] J. C. Bezdek, "Mathematical models for systematics and taxonomy," Procceedings of the 8th International Conference on Numerical Taxonomy, Freeman, San Francisco, CA, pp. 143–166, 1975.

[119] S. Han, S. Park, W. Pedrycz, "Conditional fuzzy clustering for blind channel equalization", Applied Soft Computing, vol. 11, no. 2, pp. 2777-2786, March 2011.

[120] J. C. Bezdek, R. Erhlich, W. Full, "FCM: The fuzzy c-means clustering algorithm," Computers & Geoscience 10, pp. 191–203, 1984.

[121] D. P. Kingma and J. L. Ba, “Adam : A method for stochastic optimization”, The 3rd International Conference for Learning Representations, San Diego, CA 2015. arXiv:1412.6980v9

[122] S. J. Reddi, S. Kale, S. Kumar “On the Convergence of Adam and Beyond”, 8th International Conference on Learning Representations, Vancouver, BC, Canada 2018.

[123] R. Krishnapuram and J. M. Keller, “A possibilistic approach to clustering,” IEEE Transactions on Fuzzy Systems, vol. 1, no. 2, pp. 98-110, 1993.

[124] K. Kummamuru, A. Dhawale, and R. Krishnapuram, “Fuzzy co-clustering of documents and keywords,” Fuzzy Systems, 2003. FUZZ '03. The 12th IEEE International Conference on, vol. 2, pp. 772-777, 2003.

[125] C.-H. Oh, K. Honda, and H. Ichihashi, “Fuzzy clustering for categorical multivariate data,” IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th, vol. 4, pp. 2154-2159, 2001

[126] R. Dave “Characterization and detection of noise in clustering,” in Pattern Recognition Letters, vol. 12, no. 11, pp. 657-664, Nov 1991

[127] H. Ichihashi and K. Honda, “Fuzzy c-Means Classifier for Incomplete Data Sets with Outliers and Missing Values,” International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, vol. 2, pp.457,464, 2005.

[128] G. M. Van de Ven, T. Tuytekaars, A. Tolias, “Three types of incremental learning,” In Nature Machine Intellignece, vol. 4, pp. 1185-1197, 2022.

[129] W. Pedrycz, "Conditional Fuzzy Clustering in the Design of Radial Basis Function Neural Networks", IEEE Trans. on Neural Network, vol. 9, n.4, pp. 601-612, July 1998.

[130] S. Adhikaria, J. Sing, D. Basu, M. Nasipuri, “Conditional spatial fuzzy C-means clustering algorithm for segmentation of MRI images”, Applied Soft Computing, vol. 34, pp. 758-769, Sep 2015

[131] O. F. Reyes-Glaviz, W. Pedrycz, “Granular fuzzy models: Analysis, design, and evaluation”, International Journal of Approximate Reasoning, vol. 64, pp. 1-19, Sep. 2015

[132] X. Liang, W. Pedrycz, “Fuzzy Logic-Based Networks: A Study in Logic Data Interpretation,” in Intelligent Systems, vol. 21, no. 12, pp. 1249-1267, 2006

[133] W. Pedrycz, “Fuzzy equalization in the construction of fuzzy sets,” In Fuzzy sets and systems, vol. 119, no. 2 pp. 329-335, April 2001.

[134] Md. Ahmed, N. Mat Isa, “Knowledge base to fuzzy information granule: A review from the interpretability-accuracy perspective,” In Applied Soft Computing, vol. 54, pp.121-141, 2017

[135] Casillas, O. Cordón, F. Herrera, L. Magdalena, Accuracy improvements in linguistic fuzzy modeling, Studies in Fuzziness and Soft Computing, vol. 129, Springer, 2003.

[136] J. Casillas, O. Cordón, F. Herrera, L. Magdalena, Studies in Fuzziness and Soft Computing, vol. 128, Springer, 2003.

[137] Russell C. Eberhart and Yuhui Shi, Computational Intelligence: Concept to Implementation, 1st ed, Burlington, MA, USA, Morgan Kaufmann Publishers, Aug. 2007.

[138] R. Ballini, F. Gomide, "Gradient Projection Method and Equality Index in Recurrent Neural Fuzzy Network", 10th International Fuzzy Systems Association World Congress, Istanbul, Turkey, 2003.

[139] W. Pedrycz, "Processing in relational structures: fuzzy relational equations," Fuzzy Sets and Systems, vol. 40, no. 1, pp.77–106, 1991.

[140] L. M. Wei, W.X. Xie, "Rival checked fuzzy c-means algorithm", Acta Electronica Sinica, vol. 28, no. 7, pp. 63–66, 2000 (in Chinese).

[141] J. Fan, W. Zhen, W. Xie, “Suppressed fuzzy c-means clustering algorithm.” In Pattern Recognition Letters, vol. 24, no. 9-10, pp. 1607-1612, 2003.

[142] M. N. Ahmed, S. M. Yamany, N. Mohamed, A. A. Farag and T. Moriarty, "A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data," in IEEE Transactions on Medical Imaging, vol. 21, no. 3, pp. 193-199, March 2002.

[143] S. Chen, D. Zhang, "Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 34, no. 4, pp. 1907-1916, Aug. 2004.

[144] S.C. Lee and E.T. Lee, "Fuzzy neurons and automata," in Proc. 4th Princeton Conf. Information Science Systems, pp. 381-385, 1970.

[145] W.S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity, Bulletin of Mathematical Biophysics," vol. 5, pp. 115-133, 1943.

[146] M. Malgwi, M. Schlogl and M. Keiler, “Expert-based versus data-driven flood damage models: A comparative evaluation for data-scarce regions.” In International Journal of Disaster Risk Reduction, vol. 57, April 2021.

[147] W. Pedrycz, "Neurocomputations in relational systems," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 289-297, March 1991.

[148] A. Ciaramella, R. Tagliaferri, W. Pedrycz, "The genetic development of ordinal sums," Fuzzy Sets and Systems, vol. 151, no. 2, pp. 303-325, 2005

[149] B. Schweizer, A. Sklar, Probabilistic Metric Spaces, North Holland, Amsterdam, 1983

[150] P. V. de Campss Souza, E. Lughofer, “Identification of Heart Sounds with an Interpretable Evolving Fuzzy Neural Network”, Sensors, vol. 20, no. 22, pp. 6477, Nov. 2020.

[151] W. Pedrycz, R. A. Aliev, “Logic-oriented neural networks for fuzzy neurocomputing,” In Neurocomputing, vol, 73, pp. 10-23, 2009.

[152] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, pp. 436–444, 2015.

[153] C. Gallicchiom S. Scardapane, “Deep randomized neural networks”, arXive: arxiv.org/abs/2002.12287v1, 2020.

[154] F. LI, J. QIAO, H. Han, C. Yang, “A self-organizating cascade neural network with random weights for nonlinear system modeling,” Applied Soft Computing, vol. 42, pp. 184-193, May 2016.

[155] Y. Mi, W. Liu, Y. Shi and J. Li, "Semi-Supervised Concept Learning by Concept-Cognitive Learning and Concept Space," In *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 5, pp. 2429-2442, 1 May 2022

[156] Y. Shi, Y. Mi, J. Li and W. Liu, "Concept-Cognitive Learning Model for Incremental Concept Learning," In *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 2, pp. 809-821, Feb. 2021.

[157] G. H. Gu, Y. Y. Cao, D. Cui, and Y. Zhao, “Object image annotation based on formal concept analysis and semantic association rules,” *Acta Automatica Sinica*, vol. 46, no. 4, pp. 767–781, 2020

[158] M. Cintra, H. Camargo, M. Monard, “Genetic generation of fuzzy systems with rule extraction using formal concept analysis,” In *Information Sciences*, vol. 349-350, pp. 199-215, 2016.

[159] D. S. Modha et al., “Cognitive computing,” Commun. ACM, vol. 54, pp. 62–71, Aug. 2011.

[160] Y. X. Wang et al., “Cognitive informatics: Towards cognitive machine learning and autonomous knowledge manipulation,” *International Journal of Cognitive Informatics and Natural Intelligence*, vol. 12, no. 1, pp. 1–13, Jan. 2018

[161] Y. X. Wang, “On cognitive computing,” *International Journal of Software Science and Computational Intelligence*, vol. 1, no. 3, pp. 1–15, 2009.

[162] J. Feldman, “Minimization of Boolean complexity in human concept learning,” *Nature*, vol. 407, pp. 630–632, Oct. 2000.

[163] Y. X. Zhao, J. H. Li, W. Q. Liu, and W. H. Xu, “Cognitive concept learning from incomplete information,” *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 1, pp. 159–170, Feb. 2017.

[164] Xindong Wu et al., "Knowledge Engineering with Big Data," in IEEE Intelligent Systems, vol. 30, no. 5, pp. 46-55, Sept.-Oct. 2015.

[165] W. Pedrycz, "Computing and Clustering in the Environment of Order-2 Information Granules," in IEEE Transactions on Cybernetics, vol. 53, no. 9, pp. 5414-5423, Sept. 2023.

[166] E. Klement, R. Mesiar and E. Pap, “Logical, Algebraic, Analytic and Probabilistic Aspects of Triangular Norms,” pp. 17, 80, Elsevier Science B.V., 2005

[167] R. Bekkerman and J. Allan, "Using bigrams in text categorization," Technical Report IR-408, Department of Computer Science, University of Massachusetts, MA, 2004

[168] K. Bache, M. Lichman, UCI machine learning repository [Online], Available: http://archive.ics.uci.edu/ml

[169] W. Pedrycz, An Introduction to Computing with Fuzzy Sets; Analysis, Design, and Applications, Intelligent Systems Reference Library, vol. 190, Switzerland, Springer, 2021.

[170] R. French “Catastrophic forgetting in connectionist networks,” In *Trends in Coginitve Sciences*, vol. 3, no. 4 pp. 128-135, 1999.

[171] W. Xu, Y. Chen, "Multi-attention concept-cognitive learning model: A perspective from conceptual clustering," In Knowldge-based Systems, vol. 252, 2022

[172] W. -Z. Wu, Y. Leung and J. -S. Mi, "Granular Computing and Knowledge Reduction in Formal Contexts," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 10, pp. 1461-1474, Oct. 2009,

[173] A. Pérez-Suárez, J. Martínez-Trinidad, J. Carrasco-Ochoa, "A review of conceptual clustering algorithms," In Artificial Intelligence Review, vol. 52, pp. 1267-1269, 2019

[174] D. Bertsimas, A. Orfanoudaki, H. Wiberg, “Interpretable clustering: an optimization approach,” In *Machien Language*, vol. 110, no, 1, pp. 89-138, 2021

[175] V. Losing, B. Hammer, and H. Wersing, “Incremental on-line learning: A review and comparison of state of the art algorithms,” In *Neurocomputing*, vol. 275, pp. 1261–1274, Jan. 2018.

[176] G. Parisi, R. Kemker, J. Part, C. Kanan, and S. Wermter, “Continual lifelong learning with neural networks: A review,” In *Neural Networks*, vol. 113, pp. 54–71, 2019.

[177] H. Li, W. Dong and B. -G. Hu, "Incremental Concept Learning via Online Generative Memory Recall," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 3206-3216, July 2021

[178] K. Guo, J. Li, X. Zhang, “Notes on the improvement of concept-cognitive learning accuracy,” In *International Journal of Approximate Reasoning*, vol. 156, pp. 77-96, 2023

[179] T. Zhang, He-he Li, M. Liu, M. Rong, “Incremental concept-cognitive learning based on attribute topology,” In International Journal of Approximate Reasoning, vol. 118, pp. 173-189, 2020

[180] J. M. Chambers, W. S. Cleveland, B. Kleiner, P. A. Tukey, Graphical Methods for Data Analysis, Wadsworth Int. Group, Belmont, CA 1983

[181] Y. Mi, Z. Wang, H. Liu, Y. Qu, G. Yu, Y. Shi, “Divide and conquer: A granular concept-cognitive computing system for dynamic classification decision making,” In *European Journal of Operational Research*, vol. 308, pp. 255-273, 2023

[182] X. Geng, K. Smith-Miles, Incremental Learning, In: Li, S.Z., Jain, A. (eds) Encyclopedia of Biometrics, pp. 731-735, Springer, Boston, MA, 2009.

[183] Q. Zhang, Y. Qu, A. Deng and R. Zwiggelaar, "A clustering reduction algorithm for fuzzy concept lattice," *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, Guilin, China, 2017, pp. 1894-1899,

[184] Kaufman, K.A.. Conceptual Clustering. In: Seel, N.M. (eds) Encyclopedia of the Sciences of Learning. Springer, Boston, MA, 2012.

[185] D. H. Fisher, “Knowledge acquisition via incremental conceptual clustering” In *Machine Learning*, vol. 2, pp.139–172, 1987

[186] R. S. Michalski, R. E. Stepp "LEARNING FROM OBSERVATION: CONCEPTUAL CLUSTERING," In Machine Learning: An Artificial Intelligence Approach, vol. I, pp. 331-363, 1983

[187] L. Kwuida, D. Ignatov, “On Interpretability and Similarity in Concept-Based Machine Learning,” *International Conference on Analysis of Images, Social Networks and Texts*, vol 12602, pp. 28-54, Springer, 2021

[188] L. Shapley, "A Value for n-Person Games". Contributions to the Theory of Games (AM-28), vol. II, pp. 307-318, Princeton: Princeton University Press, 1953,

[189] J. Konecny, "On attribute reduction in concept lattices: methods based on discernibility matrix are outperformed by basic clarification and reduction" In *Information Sciences*, vol. 415-416, pp. 199–212, 2017.

[190] J. Konecny, P. Krajca, “On attribute reduction in concept lattices: Experimental evaluation shows discernibility matrix based methods inefficient,” In *Information Sciences*, vol. 467, pp. 431–445, 2018

[191] V. Loia, F. Orciuoli, W. Pedrycz, “Towards a granular computing approach based on Formal Concept Analysis for discovering periodicities in data,” In *Knowledge-Based Systems*, vol. 146, pp. 1-11, 2018

[192] G. Wei, L. Xiu, “Knowledge Lock: Overcoming Catastrophic Forgetting in Federated Learning,”, In *Advances in Knowledge Discovery and Data Mining*, Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 601-612, 2022

[193] K. Pang, P. Liu, S. Li, L. Zou, M. Lu, L. Martinez, “Concept lattice simplification with fuzzy linguistics information based on three-way clustering,” In *International Jouranl of Approximate Reasoning*, vol. 154, 149-175, 2023.

[194] V. P. Savnykh, V. Tsvetkov, “Cognitive logic’s princeples,” In Artificial Intelligence in Intelligent Systems, Computer Science On-Line Conference CSOC 2021, vol. 229, pp. 228-296, 2021.

[195] X. Zhang, D. Miao, H. Fujita, “Variable-precision three-way concept in L-contexts,” *In International Journal of Approximate Reasoning*, vol. 130, pp. 107-125, 2021.

[196] L. Zadeh, "Fuzzy Set Theory and Probability Theory: What is the Relationship?," International Encyclopedia of Statistical Science pp. 563–566, 2014.

[197] D. Lindley “The probability approach to the treatment of uncertainty in artificial intelligence and expert systems,” Statistical Science, vol. 2, no. 1, pp. 17–24, 1987

[198] L. Zadeh, “Is probability theory sufficient for dealing with uncertainty in AI: A negative view,” *In Macine Intellignevce and Pattern Recognition*, vol. 4, pp. 103-116, 1986

[199] LA. Zadeh, "Probability theory and fuzzy logic – a radical view," In *Journal of American Statistical Association*, vol. 99, no. 467 pp. 880–881, 2004

[200] M. Sohail, M. Bin Saeed, S. Rizvi, M. Shoaib, A. Ul Haq Sheikh, "Low-Complexity Particle Swarm Optimization for Time-Critical Applications," arXiv:1401.0546v1, 2014

[201] W. Peng, J. Gero, “Concept formation in scientific knowledge discovery from a Constructivist view,” In Scientific Data Mining and Knowledge Discovery, pp. 91 – 109, Springer, Berlin, Germany, 2009.

[202] Y. Wang, D. Gafurov, “The cognitive process of conprehnsion: a formal description”, *In International Journal of Cognitive informatics and Natural Intelligence*, vol. 4, no. 3, pp. 44-58, July 2010

[203] Yingxu Wang and D. Gafurov, "The cognitive process of comprehension," The Second IEEE International Conference on Cognitive Informatics, 2003. Proceedings., pp. 93-97, London, UK, 2003.

[204] L. Zadeh, "Discussion: Probability Theory and Fuzzy Logic Are Complementary Rather Than Competitive," In American statistical association and the american society for quality control, vol. 37, no. 3, 1995.

[205] M. W. Shen, “Trust in AI: Interpretability is not necessary or sufficient, while black-box interaction is necessary and sufficient,” arXiv:2202.05302v1, Feb. 2022.

[206] D. S. Watson, "Conceptual challenges for interpretable machine learning," Synthese, vol. 200, March 2022.

[207] A. Renda, P. Ducange, F. Marcelloni, D. Sabella, M. C. Filippou, G. Nardini, G. Stea, A. Virdis, D. Micheli, D. Rapone, L. Baltar, "Federated Learning of Explainable AI Models in 6G Systems: Towards Secure and Automated Vehicle Networking," Information, vol. 13, no. 8:395, 2022.

[208] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera,"Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," vol. 58, pp. 82-115, June 2020

[209] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," Nature Machine Intelligence, vol. 1, pp. 206-215, 2019.

[210] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter and L. Kagal, "Explaining Explanations: An Overview of Interpretability of Machine Learning," *IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 80-89, Turin, Italy, 2018.

[211] U. Kamath, J. Liu, Introduction to Interpretability and Explainability. In: Explainable Artificial Intelligence: An Introduction to Interpretable Machine Learning, Springer, Cham, 2021, pp. 1-26

[212] M. Brundage et al., “Toward trustworthy AI development: Mechanisms for supporting verifiable claims,” arXiv:2004.07213v2, 2020

[213] X. Sun, S. Sun, “Adversarial robustness and attacks for multi-view deep models,” *Engineering Applications of Artificial Intelligence*, vol. 97, 2021

[214] B. Li, P. Qi, B. Liu, S. Di, J. Liu, J. Pei, J. Yi, B. Zhou, “Trustworthy AI: From Principles to Practices,” ACM Computing Surveys, vol. 55, no. 9, pp. 1-46, 2023.

[215] C. Molnar, “Interpretable Machine Learning: A Guide For Making Black Box Models Explainable,” 2019

[216] M. Krishnan, “Against Interpretability: a Critical Examination of the Interpretability Problem in Machine Learning,” Philosophy and Technology, 2020, https://doi.org/10.1007/s13347-019-00372-9

[217] Z. Lipton, “The Mythos of Model Interpretability: In machine learning, the concept of interpretability is both important and slippery,” In *Queue*, vol. 16, no. 3, pp. 31-57, 2018

[218] D. Janzing, L. Minorics, P. Bloebaum, "Feature relevance quantification in explainable AI: A causal problem, in Proceedings of the Twenty Third International Conference on Artificial Intelligence and

Statistics," In Proceedings of Machine Learning Research (PMLR, 2020), ed. by S. Chiappa, R. Calandra, vol. 108. , pp. 2907–2916. 2020

[219] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, H. Muller, "Causability and explainability of artificial intelligence in medicine,"In Wiley Data Min Knowl Discov, vol. 9, no.4, 2019

[220] Ethics Guidelines for Trustworthy AI, Technical Report. European Commission. High Level Expert Group on AI. 2019. Available online: https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai (accessed on 20 August 2023).

Appendix

A. Appendix A

This section presents full description of datasets used in conducting experiments in Chapter 3.

Table 0-1: Description of Boston housing dataset

Number of instances	506
Attributes	1. CRIM: per capita crime rate by town 2. ZN: proportion of residential land zoned for lots over 25,000 sq.ft. 3. INDUS: proportion of non-retail business acres per town 4. CHAS: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise) 5. NOX: nitric oxides concentration (parts per 10 million) 6. RM: average number of rooms per dwelling 7. AGE: proportion of owner-occupied units built prior to 1940 8. DIS: weighted distances to five Boston employment centres 9. RAD: index of accessibility to radial highways 10. TAX: full-value property-tax rate per $10,000 11. PTRATIO: pupil-teacher ratio by town 12. B: 1000(Bk - 0.63)^2 where Bk is the proportion of black people by town 13. LSTAT: % lower status of the population
Target	MEDV: Median value of owner-occupied homes in $1000's

Table 0-2: Description of Diabetes dataset

Number of instances	442
Number of attributes	1. Age: age in years 2. sex: 3. BMI: body mass index 4. BP: average blood pressure 5. S1 TC: total serum cholesterol 6. S2 LDL: low-density lipoproteins 7. S3 HDL: high-density lipoproteins 8. S4 TCH: total cholesterol / HDL 9. S5 LTG:, possibly log of serum triglycerides level 10. S6 GLU: blood sugar level
Target	measure of disease progression one year after baseline

Table 0-3: Description of Iris plants dataset

Number of instances	150
Attributes	1. sepal length 2. sepal width 3. petal length 4. petal width
Classes	1. Setosa 2. Versicolour 3. Virginica

A. Appendix B

Daily readings of the following air quality values from May 1, 1973 to September 30, 1973. The data were obtained from the New York State Department of Conservation (ozone data) and the National Weather Service (meteorological data).

This dataset was used in Chapter 6

Table 0: Description of New York air quality measurements dataset

Number of instances	156
Attributes	**Ozone**: Mean ozone in parts per billion from 1300 to 1500 hours. **Solar.R**: Solar radiation in the frequency band 4000–7700 Angstroms from 0800 to 1200 hours. **Wind**: Average wind speed in miles per hour at 0700 and 1000 hours. **Temp**: Maximum daily temperature in degrees Fahrenheit.