UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Ayushmat Bhardwaj Soni

Reviewing the classification performance of recent neuro-fuzzy systems

Master's Thesis (30 ECTS)

Supervisor: Prof. Stefania Tomasiello
Mr. Muhammad Uzair

Tartu 2024

Reviewing the classification performance of recent neuro-fuzzy systems

Abstract: This thesis explores the classification performance of recent neuro-fuzzy systems, which combine fuzzy logic with neural networks to enhance machine learning applications. Through a meta-analysis of literature from the past decades, the study evaluates various neuro-fuzzy architectures including the classical Adaptive Neuro-Fuzzy Inference System (ANFIS) and their performance in different domains. Comparative analyses with different approaches highlight neuro-fuzzy systems' strengths in handling imprecise and noisy data and their dependency on fuzzy set design and neural architecture. The goal of this work is to offer practical insights for both practitioners and scholars on the selection and management of appropriate methods for classification tasks across various application domains such as medicine and finance. This is achieved through a detailed analysis of diverse approaches, addressing a gap in recent comprehensive reviews of these methods.

Keywords:
Mamdani, Takagi-Sugeno-Kang, Fuzzy rules

CERCS: P170 Computer science, numerical analysis, systems, control

Hiljutiste neuro-hägusüsteemide klassifikatsioonitulemuste ülevaade

Lühikokkuvõte: See lõputöö uurib uusimate neuro-fuzzy süsteemide klassifitseerimisvõimekust, mis ühendavad hägusloogika ja närvivõrgud, et täiustada masinõppe rakendusi. Kümnendi jooksul ilmunud kirjanduse meta-analüüsi kaudu hinnatakse erinevaid neuro-fuzzy arhitektuure, sealhulgas klassikalist Adaptive Neuro-Fuzzy Inference Systemi (ANFIS), ja nende toimivust erinevates valdkondades. Võrdlevad analüüsid erinevate lähenemisviisidega toovad esile neuro-fuzzy süsteemide tugevused ebatäpse ja mürarikka andmete töötlemisel ning nende sõltuvuse hägusate hulga kujundusest ja närvivõrgu arhitektuurist. Selle töö eesmärk on pakkuda praktilisi teadmisi nii praktiseerijatele kui ka teadlastele sobivate meetodite valiku ja haldamise kohta klassifitseerimisülesannete jaoks erinevates rakendusvaldkondades, nagu meditsiin ja rahandus. Seda saavutatakse mitmekesiste lähenemisviiside üksikasjaliku analüüsi kaudu, käsitledes lünka nende meetodite hiljutistes ülevaadetes.

Võtmesõnad:
Mamdani, Takagi-Sugeno-Kang, Fuzzy rules

CERCS: P170 Computer science, numerical analysis, systems, control

Acknowledgements
I am sincerely grateful to my supervisors, Prof. Stefania Tomasiello and Mr. Muhammad Uzair, for their invaluable guidance and support. Their expertise and dedication have shaped my growth, providing insightful feedback and challenging me to excel.

List of Tables

1 Characteristics of datasets used as classification benchmarks in the experiments. . . . 35
2 Haberman's Survival Data Attributes 35
3 Cryotherapy Data Attribute 36
4 Heart Disease Data Attributes 36
5 Autism Screening Data Attributes 37
6 Immunotherapy Data Attributes 37
7 Iris Data Attribute 37
8 Thyroid Disease Data Attribute 38
9 Wine Data Attribute 38
10 Pima Indians Diabetes Dataset 39
11 Balance Scale Dataset 39
12 Lymphography Dataset 40
13 Car Evaluation Dataset 40
14 Neuro fuzzy system and their acronyms 41
15 Accuracy results for binary classification Across Multiple Datasets . . . 41
16 Accuracy Results for multi-classification across multiple datasets 42
17 Parameters used for ANFIS-T and CANFIS-T 42
18 Friedman's ranking for binary classification 43
19 Friedman's ranking for multi-class classification 44

List of Figures

1 Examples of four classes of parameterized MFs: (a) triangle (x; 20, 60, 80); (b) trapezoid (x; 10, 20, 60, 95); (c) Gaussian (x; 50, 20); (d) bell (x; 20, 4, 50) (Taken from researchhubs.com) 10
2 Different activation functions 14
3 An example of feedforward neural network 14
4 Classical neuro-fuzzy systems [43] 17
5 ANFIS architecture (Taken from mathworks.com) 18
6 General model of Mamdani ANFIS 20
7 Classification of the population-based techniques [43] 27
8 Variation of accuracy with C (standard regularisation) for CANFIS-T . . 45
9 Variation of accuracy with *a* (fractional regularisation) for CANFIS-T (C=1) 45
10 Variation of accuracy with *a* (fractional regularisation) for CANFIS-T (C=10) 46

11 Variation of accuracy with a (fractional regularisation) for CANFIS-T (C=100) 46

Contents

1 Introduction **7**
1.1 Problem Overview . . . 7
1.2 Motivation and Main Goals . . . 8

2 Background and Theoretical Framework **9**
2.1 Fuzzy Sets . . . 9
2.1.1 Definition . . . 9
2.1.2 Membership Functions . . . 9
2.1.3 Connectives . . . 11
2.2 Fuzzy Inference Systems . . . 11
2.3 Neural Networks . . . 13
2.3.1 Fundamentals of Neural Networks . . . 13
2.3.2 Architecture . . . 15
2.3.3 Learning Algorithms . . . 15
2.3.4 ANN and Neuro-Fuzzy Systems . . . 16
2.4 Neuro-Fuzzy System Overview . . . 16
2.4.1 Classical neuro-fuzzy systems . . . 17
2.4.1.1 Type-1 Neuro-Fuzzy Systems . . . 17
2.4.1.2 Type-2 Neuro-Fuzzy Systems . . . 21
2.4.2 Different training algorithms used in neuro-fuzzy models . . . 23
2.4.2.1 Least Squares-based Neuro-Fuzzy Systems . . . 23
2.4.2.2 Gradient Based Neuro-Fuzzy System . . . 25
2.4.2.3 Hybrid Neuro-fuzzy Systems . . . 25
2.4.2.4 SVM-based Neuro-fuzzy Systems . . . 26
2.4.2.5 Population-based Neuro-fuzzy Systems . . . 26

3 Literature Review **28**

4 Methodology and Results **32**
4.1 Datasets . . . 33
4.2 Results and Discussion . . . 41

5 Concluding remarks **47**

Appendix **53**
I. Licence . . . 54

1 Introduction

1.1 Problem Overview

Classification is now essential in many areas because of its importance in practical, everyday use. It helps make accurate decisions by organizing data into clear categories. For example, in risk assessment and prevention, classification can predict the chances of certain events happening, allowing us to take action ahead of time [31]. Additionally, it is crucial in quality control processes, where it helps identify defective or non-compliant products to ensure that only high-quality goods are available in the market [34]. Classification is also fundamental in detecting fraudulent activities by distinguishing transactions or behaviours as legitimate or fraudulent, thus protecting individuals and organizations from financial losses [19]. It also plays a key role in sentiment analysis, where it helps to sort text into positive or negative sentiments. This gives us valuable insights into what consumers think and prefer [37].

Numerical data, marked by continuous variables like age, income, or sensor readings, are prevalent in various fields such as finance, healthcare, and marketing [24]. However, applying classification techniques to numerical data requires careful consideration of factors like data distribution, feature engineering, and model selection to achieve accurate and reliable predictions[48]. Despite these challenges, the widespread adoption of machine learning techniques faces resistance in these critical areas, where errors could severely impact human life or financial stability.

The structure of a rule in a Fuzzy Inference System (FIS) can be cumbersome, as it requires each rule to consider all input variables within its antecedent section. This requirement makes fuzzy rules as long as the number of input variables, which is often impractical. This limitation hinders the system's operational effectiveness, interpretation, performance, and scientific reliability [40].

In this thesis, we focus on the Fuzzy Logic system for classification using various data types. Fuzzy logic systems handle approximate reasoning and degrees of truth, allowing for flexible and intuitive handling of complex, real-world situations where information is uncertain or imprecise. We aim to evaluate and compare the performance of different neuro-fuzzy systems, examining their accuracy, computational efficiency, and robustness. The goal is to rank their effectiveness in processing and classifying diverse data sets, providing insights into their practical applications and potential improvements. We have selected different data sets from various domains and assessed their accuracy.

1.2 Motivation and Main Goals

Neural networks and fuzzy inference systems combined have demonstrated potential for solving difficult classification issues. Conventional approaches such as the Adaptive Neuro-Fuzzy Inference System (ANFIS) have enabled the combination of fuzzy system interpretability with neural network learning capabilities [25]. However, these methods often have shortcomings, particularly when developing structured fuzzy rules that need to consider each and every input variable [40]. The primary motivation behind this thesis is to identify the most effective approach for classification tasks. Traditional methods, like ANFIS, work great, but they lose some interpretability and scalability because every rule has to assess every input variable.

Recent advancements in neuro-fuzzy systems have been pivotal in pushing the boundaries of research in artificial intelligence. However, despite considerable progress, challenges in classification performance, such as overfitting, interpretability, and computational efficiency, remain persistent obstacles. Addressing these challenges is crucial for the deployment of neuro-fuzzy systems in critical applications like medical diagnosis, financial prediction, and environmental monitoring.

While existing literature extensively explores individual components of neuro-fuzzy systems, comprehensive reviews on their classification performance—particularly in recent implementations—are sparse.

The thesis aims to achieve the following specific objectives:

1. **Investigate Approaches** Several results from different researches have been reviewed and compared against CANFIS-T and ANFIS-T which have been recently proposed [50], [51] but not tried on classification yet.

2. **Analysis of Performance** Compare the performance of various methodologies in terms of classification accuracy. Identify the strengths and weaknesses of each approach in handling classification tasks across different domains.

To achieve these objectives, this thesis hopes to shed light on the potential benefits of non-structured rules for the delivery of improved translation and performance. The aim is to provide practical insights for practitioners and scholars on selecting and managing appropriate methods for binary classification tasks in different application domains through detailed analysis, as no such review of different approaches has been done recently.

The content within this thesis has been enhanced through the use of OpenAI's ChatGPT-4 model. Specific sections were generated with the assistance of this advanced language model, which also provided support in reviewing and refining the grammar and syntactic structure of the entire document.

2 Background and Theoretical Framework

2.1 Fuzzy Sets

2.1.1 Definition

"Fuzzy sets[18], first introduced by L.A. Zadeh in 1965, extend the classical notion of sets, traditionally known as crisp sets. Unlike crisp sets where the membership of elements is binary, being either included or excluded, fuzzy sets introduce a gradient of membership. This allows elements to possess a degree of association with the set, quantified between 0 and 1. This attribute of fuzzy sets is particularly advantageous for handling the ambiguity and vagueness often encountered in real-world data. Let S be a space of points (objects), with a generic element of S denoted by s, and let A be a set, $A \subset S$ [51].

- A classical (crisp) set A is a finite collection of elements $s \in S$. Every single element can either belong to or not belong to A. The member elements are defined by means of the characteristic function $\chi_A : S \rightarrow \{0, 1\}$ (1 - membership; 0 - nonmembership).
- A fuzzy set A is uniquely defined by the membership function $\mu_A : S \rightarrow [0, 1]$, which allows a continuum of degrees of membership for the elements of a given set; $\mu_A(s)$ is the degree of membership of s in A. " [52]

2.1.2 Membership Functions

"The design of membership functions (MFs) is crucial as they define how each element's degree of membership is calculated. Typical forms of membership functions include:

- **Triangular MF**, providing straightforward and computationally efficient representations

$$\mu_A = \begin{cases} \frac{s-s_l}{s_c-s_l}, & s \in (s_l, s_c] \\ \frac{s_r-s}{s_r-s_c}, & s \in (s_c, s_r) \\ 0, & otherwise \end{cases} \tag{1}$$

- **Gaussian MF**, known for its smooth and symmetrical curve, it represents the typical parametric MF used in neuro-fuzzy systems

$$\mu_A = \begin{cases} \exp(-(\frac{s-s_c}{s_l})^2), & s \in [s_c - \delta, s_c + \delta] \\ 0, & otherwise \end{cases} \tag{2}$$

  where δ is the spread. It models the linguistic expression about s_c." [52]

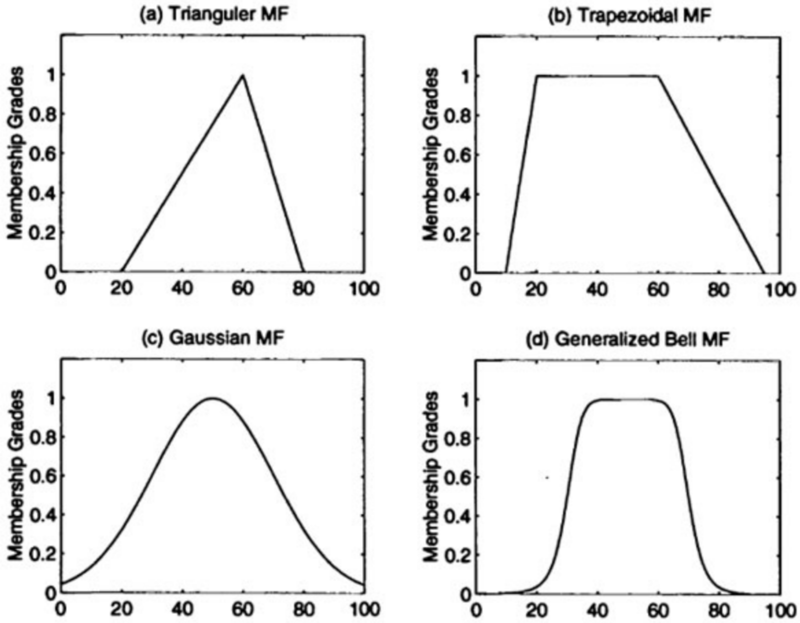

Figure 1. Examples of four classes of parameterized MFs: (a) triangle (x; 20, 60, 80); (b) trapezoid (x; 10, 20, 60, 95); (c) Gaussian (x; 50, 20); (d) bell (x; 20, 4, 50) (Taken from researchhubs.com)

2.1.3 Connectives

"Given two fuzzy sets A and B, the **t-norm** is represented by the function

$$\underline{\cap} : [0,1] \times [0,1] \to [0,1],$$

which aggregates two membership grades. The t-norm has to be regarded as a general class of intersection operators for fuzzy sets. Types of t-norms [49]:

- $\underline{\cap}_1(x, y) = \min\{x, y\} = x \cap y$ (minimum-operator);
- $\underline{\cap}_2(x, y) = xy$ (algebraic product);
- $\underline{\cap}_3(x, y) = \max\{0, x + y - 1\}$ (bounded product);
- $\underline{\cap}_4(x, y) = \begin{cases} x & if \quad y = 1 \\ y & if \quad x = 1 \\ 0 & otherwise \end{cases}$ (drastic product).

The **t-conorm** operator is represented by the function:

$$\overline{\cup} : [0,1] \times [0,1] \to [0,1]$$

. This operator models the logical connective **or**. Types of t-conorms [49]:

- $\overline{\cup}_1 = \max\{x, y\} = x \cup y$ (max-operator);
- $\overline{\cup}_2 = x + y - xy$ (algebraic sum);
- $\overline{\cup}_3 = \min\{1, x + y\}$ (bounded sum);
- $\overline{\cup}_4 = \begin{cases} x & if \quad y = 0 \\ y & if \quad x = 0 \\ 1 & otherwise \end{cases}$ (drastic sum).

This foundational overview of fuzzy sets prepares the ground for further discussion on neuro-fuzzy Systems." [52]

2.2 Fuzzy Inference Systems

"Fuzzy Inference Systems (FIS), as detailed in [33], represent a pivotal application of fuzzy logic and fuzzy set theory. These systems effectively utilize fuzzy set theory to transform inputs into outputs, thereby facilitating accurate predictions across various applications. FIS are particularly noted for their robust performance and exceptional generalization capabilities, which enable them to deliver optimal solutions across diverse scenarios.

FIS can be divided into two primary categories: the Mamdani model and the Takagi-Sugeno-Kang model. Each type has distinct characteristics and applications.

Mamdani Model. This model is well-known for its intuitive approach, using linguistic rules for both the antecedents (conditions) and the consequents (conclusions). It mimics human reasoning by applying a comprehensive set of fuzzy IF-THEN rules. These rules are thoroughly assessed and formulated based on expert knowledge or empirical data to establish a robust rule base. The Mamdani model is celebrated for the transparency and clarity of its rule base, which contributes significantly to its popularity in applications requiring interpretable models.

It is represented by:

$$R^i: \text{ If } x_1 \text{ is } A_1^i \text{ and } x_2 \text{ is } A_2^i \ldots \text{ and } x_n \text{ is } A_n^i, \text{ then } y^i \text{ is } B^i$$

The function R^i where $i = 1, 2, \ldots, l$, represents the i-th fuzzy rule, x_j (where $j = 1, 2, \ldots, n$) indicates the input, and y^i is the output of the fuzzy rule R^i. Furthermore, A_j^i and B^i (where $i = 1, 2, \ldots, l$) denote the fuzzy membership functions commonly linked with linguistic descriptors.

Takagi-Sugeno-Kang Model. Unlike the Mamdani model, the consequents in this model are typically mathematical functions of the input variables, rather than fuzzy sets. This model is suited for systems where precise mathematical modelling is necessary, and it excels in applications that require more rigorous computational approaches.

In Takagi-Sugeno-Kang (TSK) systems, the rules are typically expressed in a format that highlights their structured nature. Consider the generic rule R_k:

$$\textbf{Rule } R_k: \text{ If } x_1 \text{ is } A_{k1}, \ldots, \text{ and } x_n \text{ is } A_{kn} \text{ then } y \text{ is } y_k = g_k(x_1, \ldots, x_n),$$

where $k = 1, \ldots, r$.

The final output y of the system is computed as a weighted average of the outputs from each rule:

$$y = \frac{\sum_{j=1}^{r} \overline{\omega}_j g_j(x_1, x_2, \ldots, x_n)}{\sum_{j=1}^{r} \overline{\omega}_j},$$

The weights $\overline{\omega}_j$ are determined by the firing strength of each rule, which is the aggregation of the membership values of the inputs using a t-norm $\underline{\cap}$:

$$\overline{\omega}_j = \mu_{A_{j1}}(x_1) \underline{\cap} \mu_{A_{j2}}(x_2) \underline{\cap} \ldots \underline{\cap} \mu_{A_{jn}}(x_n),$$

When the functions g_i are linear, the system is specifically referred to as a Takagi–Sugeno (TS) system. This linear representation allows the model to efficiently handle complex datasets with a form of linear regression tailored to each fuzzy region defined by the rules." [52]

2.3 Neural Networks

Neural Networks (NN) [27] or Artificial Neural Networks (ANN) are a cornerstone of modern computational intelligence, laying the groundwork for more complex systems, including neuro-fuzzy systems. This section discusses the fundamental structure of neural networks, how they learn, and their practical applications, emphasizing how these aspects serve as precursors to the integration of neural networks with fuzzy logic.

2.3.1 Fundamentals of Neural Networks

Neural networks are inspired by the biological nervous systems, specifically the human brain. They comprise interconnected nodes or "neurons," collectively processing information using a network-like architecture.

Neuron A neuron is the building block of a neural network. It takes in information, processes it, and then produces an output. Each neuron carries out a basic computation on the input data. Typically, a neuron has several inputs, a bias, an activation function, and an output. To refine the input data, there can be many hidden layers between the input and the final output. The input χ of a generic node k is given by the linear sum of incoming signals x_i from the preceding layer, through the weights w_{ik}, plus a bias term b_k, that is

$$\chi_k = \sum_i w_{ik} x_i + b_k$$

Notice that $x_i = f(\chi_i)$, where f is the activation function.

Activation Functions These are crucial functions in neural networks that decide whether a neuron should be activated. They help the network learn complex patterns during training. The most common functions are :

- **Linear** $f(\chi_i) = c\chi_i$, that is the identity function when $c = 1$; it can be used in the output node.
- **Sigmoid (logistic)** $f(\chi_i) = \frac{1}{1+\exp(-\chi_i)}$, whose range is between 0 and 1.
- **Hyperbolic Tangent** $f(\chi_i) = \frac{2}{1+\exp(-2\chi_i)} - 1$, whose range is between -1 and 1.
- **Rectified Linear Unit (ReLU)** $f(\chi_i) = \max(0, \chi_i)$, which has the advantage of a lower processing time.

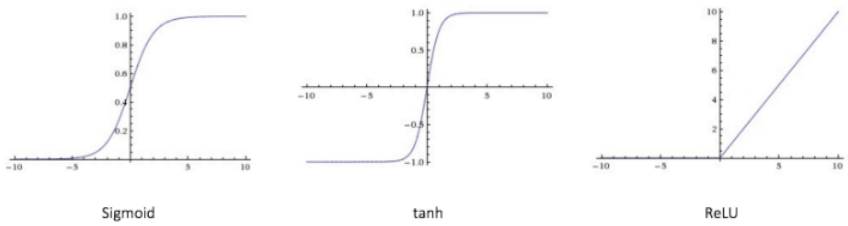

Figure 2. Different activation functions

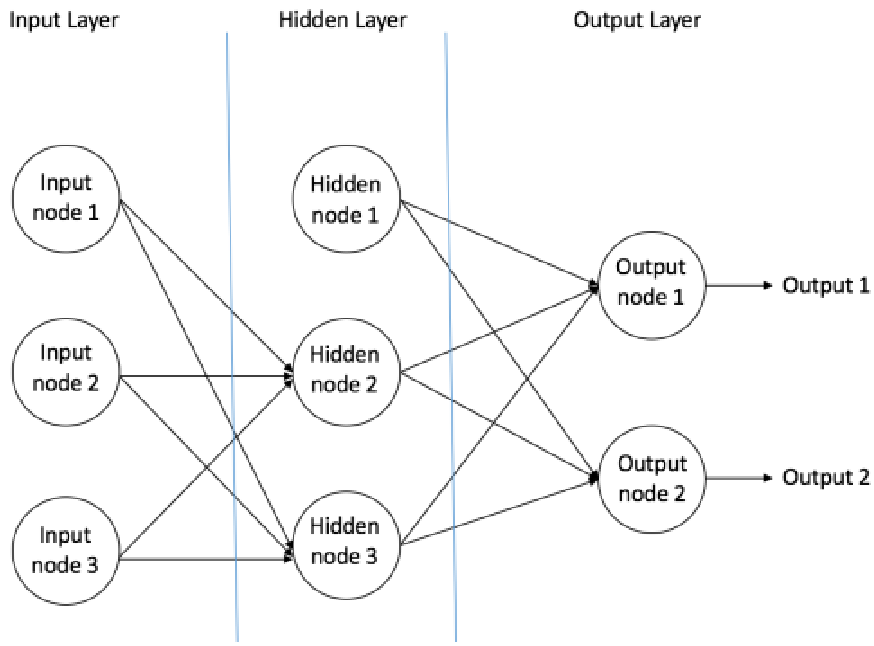

Figure 3. An example of feedforward neural network

2.3.2 Architecture

The architecture of an Artificial Neural Network (ANN) is determined by the number and arrangement of neurons within its layers, defining the network's overall structure. Typically, the configuration of an ANN's architecture is predefined by the user. There are principally two types of network architectures:

- **Feedforward Networks**, which have a unidirectional flow of information, from the input to the output layer, without any intra-layer connections. This setup ensures that data moves through the network in a single pass from inputs to outputs, as depicted in Figure 3;
- **Feedback Networks**, which, unlike feedforward networks, permit the circulation of information where outputs from one or more layers may recur as inputs to the same or preceding layers. This class includes recurrent neural networks, known for their ability to process inputs in sequences by maintaining a state that reflects previous inputs.

In feedforward networks, the total number of connections, N_c, between the n^{th} and $(n+1)^{th}$ layers is mathematically expressed as:

$$N_c(n) = \text{units}(n) \times \text{units}(n+1)$$

where units(n) and units$(n+1)$ represent the number of neurons in the n^{th} and $(n+1)^{th}$ layers, respectively. Connections between layers are characterized by a weight matrix, $W(n)$, associated with the links between the n^{th} and $(n+1)^{th}$ layers. This matrix is pivotal in the network's learning process as it adjusts to minimize error and enhance predictive accuracy.

2.3.3 Learning Algorithms

Neural networks learn through algorithms that adjust the network's weights and biases based on the error of its predictions.

Backpropagation This is the most widely used algorithm for training neural networks. Backpropagation works by propagating the error back through the network, from the output towards the input, thereby allowing the network to update the weights and minimize the error in subsequent predictions.

Stochastic Gradient Descent (SGD) SGD is a variant of the gradient descent algorithm that updates the weights of the network incrementally for each training example rather than for the entire dataset at once. This method helps to speed up the learning process and can lead to better convergence, especially in large datasets.

2.3.4 ANN and Neuro-Fuzzy Systems

The development of Neuro-Fuzzy Systems (NFS) [54] represents a significant stride in the field of intelligent systems, leveraging both computational models to create systems capable of complex decision-making and learning from data. This passage explores how ANNs and Neuro-Fuzzy Systems, including Adaptive Neuro-Fuzzy Inference Systems (ANFIS), interrelate and complement each other, particularly focusing on the learning mechanisms and system architecture involved.

Integration of ANN and Fuzzy Inference Systems ANNs are distinguished by their ability to perform machine learning without the necessity for human-derived rules. They automatically adjust their internal parameters (weights) based on the input-output data presented during training. This is largely due to the homogeneous structure of ANNs, where layers of neurons adjust their synaptic weights to minimize error rates between predicted and actual outputs. However, a significant challenge with ANNs is the extraction of interpretable knowledge from the network, as the learning is embedded intrinsically in the form of weight adjustments and neuron connections.

Neuro-Fuzzy Systems and Human Expertise Contrasting with ANNs, Neuro-Fuzzy Systems like ANFIS are designed to utilize human expertise by encoding it into a set of fuzzy rules and membership functions within their framework. This design allows NFS to perform fuzzy reasoning to infer outputs, providing a system that not only learns from data but also incorporates human-like reasoning through its rule-based structure. The challenge, however, lies in systematically transforming human expertise into an effective rule base, a task that often lacks a formalized method and relies heavily on the subjective insights of experts. This challenge has been addressed with Neuro-Fuzzy Systems (NFS), which aim to bridge this gap by combining human-like reasoning with automated learning techniques.

2.4 Neuro-Fuzzy System Overview

Introduction Neuro-fuzzy systems integrate the dynamic learning abilities of neural networks with the soft reasoning capabilities of fuzzy logic, addressing complex challenges characterized by uncertainty and ambiguity. This combination results in sophisticated systems adept at interpreting vague data and managing indeterminate conditions efficiently.

Historical Context Neuro-fuzzy systems emerged in the early 1990s from the convergence of quantitative neural network approaches with qualitative fuzzy logic methods. Initiated by efforts to utilize the adaptive learning of neural networks alongside the interpretative power of fuzzy logic, this hybrid methodology has developed over the decades. It now serves effectively in areas where binary logic systems falter, owing to its enhanced ability to process varied informational structures (Jang, 1993; Lin and Lee, 1996).

Types of Neuro-Fuzzy Systems In this section, different types of Neuro-Fuzzy systems are explained. In the paper [43], the authors divided all the Neuro-Fuzzy systems based on the algorithm used and fuzzy method.

2.4.1 Classical neuro-fuzzy systems

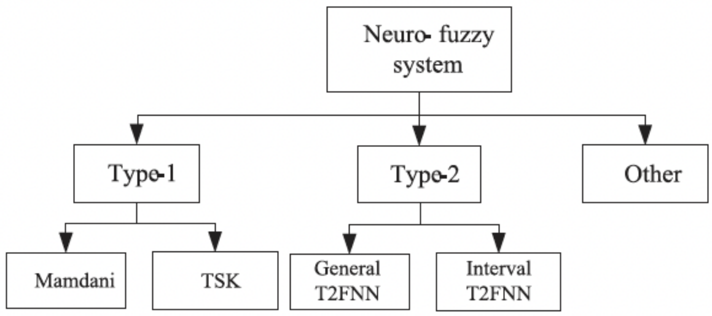

Figure 4. Classical neuro-fuzzy systems [43]

2.4.1.1 Type-1 Neuro-Fuzzy Systems

TSK-based NFS Offers a functional output as a linear combination of input variables, making it suitable for dynamic system modelling and control. ANFIS is one of the most popular methods in this class.

ANFIS A critical aspect in the realm of fuzzy sets lies in the selection of appropriate intersection operators, commonly known as t-norms. These operators facilitate the amalgamation of membership grades in diverse manners, including the utilization of minimum, algebraic products, bounded product, and drastic product methods. [30].

The Adaptive Network-based Fuzzy Inference System or ANFIS was introduced by Jang [25]. It is a five-layer network architecture representing the first-order fuzzy inference system. The term "adaptive" indicates that the nodes in this network adjust their outputs based on the parameters of the node. The common learning algorithm is gradient descent. To minimize error, parameter tuning is specified by the learning process.

Backpropagation with the least-squares method is typically used to adjust membership function parameters. The ANFIS architecture is depicted in Figure 5.

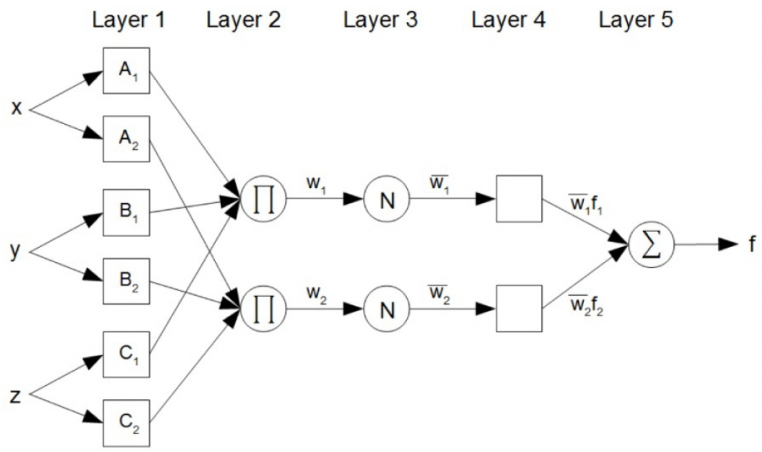

Figure 5. ANFIS architecture (Taken from mathworks.com)

In general, ANFIS operates on IF-THEN rules with functional consequent parts, adjusting antecedents and consequent parameters. The network comprises five layers ($S1$, $S2$, …, $S5$). Outputs from layer $S4$ are summed to produce the final output z_0. The output of $S1$ represents the membership degree of the input to the fuzzy set. Layer $S2$ has fixed nodes, and each node's output signifies the firing strength of a rule. Nodes in $S2$ utilize product-type t-norms for computation (to model AND). For each layer S_i:

$$
\begin{aligned}
&S1 : O_i = \mu_{Bi}(\xi_i);\\
&S2 : \omega_r = \prod_{i=1}^{n} \mu_{Bi}(\xi_i)^{\Omega_{ri}};\\
&S3 : \omega_r = \frac{\omega_r}{\sum_{j=1}^{R} \omega_j};\\
&S4 : \rho_r = \omega_r \left(\sum_{i=1}^{n} \alpha_{ir}\xi_i + \alpha_{i(n+1)} \right);\\
&S5 : z_0 = \sum_{r=1}^{R-1} \rho_r,
\end{aligned}
$$

where α_{ir} are the consequent parameters. The membership function $\mu_{Bi}(\xi)$ can take various forms, including the generalized bell-shaped function:

$$\mu_{Bi}(\xi) = \left(1 + \left(\frac{\xi - c_{ir}}{a_{ir}}\right)^{2b_{ir}}\right)^{-1}, \tag{3}$$

and the classical bell-shaped function:

$$\mu_{Bi}(\xi) = \exp\left(-\frac{(\xi - c_{ir})^2}{a_{ir}}\right). \tag{4}$$

Hybrid learning is based on backpropagation and the least-squares-based algorithm is utilized in the ANFIS. The output from all nodes proceeds until layer $S4$, and the least-squares method is applied to ascertain the consequent parameters for fixed antecedent parameters.

Matrix equation $X\delta = Y$ is derived using the training data, where δ collects the unknown parameters and Y corresponds to target values. The least-squares (LS) method is formulated as:

$$\min_{\delta} \|X\delta - Y\|^2 \tag{5}$$

With the solution,

$$\delta^* = (X^T X)^{-1} X^T Y, \tag{6}$$

where $\overline{X} = (X^T X)^{-1} X^T$ is the pseudoinverse of X. Backpropagation is then utilized to adjust the antecedent parameters. ANFIS with fractional regularisation is discussed in the following sections with least squares-based neuro-fuzzy systems.

This system is suitable for scenarios where input and output uncertainties are manageable with standard fuzzy logic, which operates on precise membership functions. These systems are commonly used due to their simplicity and effectiveness in various applications where high precision in uncertainty modelling is not critical.

Mamdani-based NFS utilizes fuzzy logic for decision-making and is appreciated for its interpretability and ease of application in human-like reasoning contexts. M-ANFIS is one such type of model shown in Figure 6.

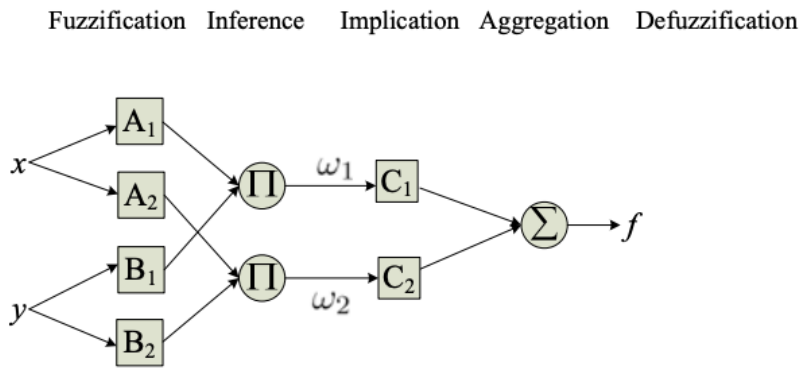

Figure 6. General model of Mamdani ANFIS

The M-ANFIS architecture is structured into multiple layers, each corresponding to a different stage in the fuzzy inference process.

1. **Fuzzification Layer**, which converts crisp inputs into degrees of membership using fuzzy sets. For each input variable x, the membership function $\mu_A(x)$ is calculated, typically using generalized bell functions:

$$\mu_{A_i}(x) = \frac{1}{1 + \left(\frac{x-c_i}{d_i}\right)^{2b_i}}$$

   where b_i, c_i, d_i are adjustable parameters that define the shape and position of the membership function.

2. **Rule Layer** computes the firing strength of each rule through a T-norm operator, typically the product:

$$\omega_i = \prod_{j=1}^{n} \mu_{A_j}(x_j)$$

   where ω_i is the firing strength of the i-th rule, and n is the number of inputs.

3. **Normalization Layer**, which normalizes the firing strengths to ensure that their sum is 1.

$$\bar{\omega}_i = \frac{\omega_i}{\sum_{k=1}^{m} \omega_k}$$

   where m is the number of rules.

4. **Output Layer** where each rule's output is calculated by applying a weighted average of the rule's consequent parameters, adjusted by the normalized firing strengths:
$$f_i = \bar{\omega}_i \cdot (p_i \cdot x + q_i \cdot y + r_i)$$
where p_i, q_i, r_i are the consequent parameters of the i-th rule.

5. **Defuzzification Layer**, where the final system output is the summation of all rule outputs, providing a crisp value:
$$F = \sum_{i=1}^{m} f_i$$

Learning Algorithm The learning process in M-ANFIS involves adjusting the parameters $b_i, c_i, d_i, p_i, q_i, r_i$ using a hybrid learning algorithm that combines gradient descent and least squares methods:

- **Premise Parameters** adjusted by gradient descent to minimize the overall system error.
- **Consequent Parameters** estimated using a least squares approach to optimize the linear combination in the rule consequents.

The error measure used to guide the learning is typically the mean squared error between the predicted outputs and the actual target values from the training data.

2.4.1.2 Type-2 Neuro-Fuzzy Systems Type-2 neuro-fuzzy systems represent a significant evolution from type-1 fuzzy systems by effectively modelling and managing the uncertainties inherent in fuzzy logic systems. These uncertainties are often overlooked in type-1 systems, which employ precise and crisp membership functions. Type-2 systems enhance the capability to handle uncertainties in both the rule and membership function definitions, which are crucial when dealing with ambiguous and imprecise information.

Interval Type-2 Neuro-Fuzzy Systems (IT2NFS) Interval Type-2 Neuro-Fuzzy Systems (IT2NFS) are the most common form of type-2 fuzzy systems. They utilize interval type-2 fuzzy sets in the antecedent and consequent parts of the rules, providing a robust mechanism for handling data and operational uncertainties.

General Type-2 Neuro-Fuzzy Systems (GT2NFS) General Type-2 Neuro-Fuzzy Systems (GT2NFS) extend the modelling capabilities of IT2NFS by representing uncertainties in a three-dimensional space, as opposed to the two-dimensional approach in IT2NFS. This approach allows for a more nuanced handling of fuzzy set uncertainties.

GT2NFS are less common and more complex but are gaining traction due to their enhanced predictive and adaptive capabilities, particularly when coupled with methods like α-plane representation and centroid-flow algorithms for type reduction [224, 225, 226].

2.4.2 Different training algorithms used in neuro-fuzzy models

2.4.2.1 Least Squares-based Neuro-Fuzzy Systems

Core Principle Least Squares-based Neuro-Fuzzy Systems incorporate the least squares method to optimize the parameters of fuzzy inference models effectively. This integration facilitates precise modelling and forecasting capabilities, making these systems invaluable in complex, data-driven applications. These systems typically employ a neuro-fuzzy framework that integrates the adaptive capacity of neural networks with the intuitive reasoning of fuzzy logic systems. The least squares method is used primarily to fine-tune the consequent parameters of the fuzzy rules within the system, enhancing the model's accuracy and predictive performance.

A novel variant of the ANFIS has been developed, incorporating the least-squares method alongside fractional Tikhonov regularization. This approach eschews traditional backpropagation and grid partitioning techniques. Fractional regularization, a sophisticated generalization of standard regularization methods, is applied to refine the ANFIS learning process. This innovation results in a streamlined rule base that comprises a minimal number of rules. Such an enhancement significantly reduces the computational demands typically associated with handling models that have a large number of input variables. Consequently, this advanced system maintains high accuracy while efficiently managing complex problems, thereby optimizing performance with reduced computational overhead [50].

Fractional regularization Fractional regularization introduces fractional order differentiation or integration into the regularization process of a model, allowing it to capture more complex patterns and long-range dependencies in the data. This approach is particularly useful for dealing with data that exhibits non-local or fractal-like properties, where traditional methods may struggle.

Mathematically, fractional regularization can be formulated as follows.

Let us consider a regularized optimization problem of the form:

$$\min_{\theta} \sum_{i=1}^{n} L(y_i, f(x_i; \theta)) + \lambda R(\theta)$$

where,

- θ represents the parameters of the model.
- L is a loss function measuring the discrepancy between the model predictions $f(x_i; \theta)$ and the true labels y_i.
- $R(\theta)$ is the regularization term, which helps to prevent overfitting.

- λ is the regularization parameter controlling the strength of regularization.

In traditional regularization methods like L_1 or L_2 regularization, $R(\theta)$ typically penalizes the magnitude of the parameters to prevent overfitting [29][38].

ANFIS with regularization In the proposed variant, the learning algorithm is only LS-based and uses a kind of regularization, both in standard and fractional versions as described in references [50]. The standard regularization method, as a generalization of the standard LS method, is well-known for solving discrete ill-posed inverse problems. The minimization problem is formulated as follows [39]:

$$\min_{\delta} \|X\delta - Y\|^2 + \frac{1}{C}\|\delta\|^2, \tag{7}$$

C here denotes the regularization parameter. The solution is given by:

$$\delta* = (X^T X + \frac{1}{C} I)^{-1} X^T Y \tag{8}$$

I denotes the identity matrix.

The fractional method is based on solving the following minimization problem,

$$\min_{\delta} \|X\delta - Y\|_Q^2 + \frac{1}{C}\|\delta\|^2, \tag{9}$$

where $\|\delta\|_Q = (\delta^T Q \delta)^{0.5}$ and Q is defined as:

$$Q = (X^T X)^{a/2-1/2}, \tag{10}$$

Q is a symmetric matrix that is positive semi-definite with $a \in (0, 1)$, which means fractional power.

Let u_i and v_i denote the column vectors of the matrices U and V, respectively, obtained from the singular value decomposition (SVD) of the matrix X, where $X = UDV^T$, and D is the diagonal matrix containing the singular values σ_i arranged in decreasing order. The solution can then be expressed as [39]:

$$\delta* = \sum_{i=1}^{q} \frac{\sigma_i^a}{\sigma_i^{a+1} + \frac{1}{C}} (u_i^T Y) v_i \tag{11}$$

Extreme Learning Machines is an important type of such methods that provide an efficient learning algorithm primarily for single-layer feedforward neural networks, where the hidden nodes need not be tuned. This simplifies the training process significantly as it focuses on linear models that can be solved analytically.

Application ELM-based methods in neuro-fuzzy systems can rapidly construct a model that is robust to noise and capable of handling large-scale data sets, making them suitable for tasks that require quick deployment with reasonable generalization performance. ELM-based neuro-fuzzy systems have been applied in various practical applications such as medical diagnosis, weather prediction, and complex system control, demonstrating their versatility and effectiveness in handling real-world problems with uncertainties.

2.4.2.2 Gradient Based Neuro-Fuzzy System

Core Principle These methods primarily rely on the concept of gradient descent, where the parameters of the neuro-fuzzy system are adjusted iteratively based on the gradient of a loss function. The objective is to minimize the error between the system's outputs and the desired outputs. It is particularly effective for finding the minimum of a cost function by iteratively moving towards the steepest descent direction determined by the negative of the gradient. This method is essential for training models like neural networks, where the goal is to adjust parameters (weights and biases) to minimize the difference between predicted and actual outputs.

Application Widely used in training neural networks, these methods are effective for continuous adjustment of membership functions and rule parameters in a neuro-fuzzy system, leading to fine-tuned performance in tasks like regression and classification. These models are widely used in the fields of Control Systems, Pattern Recognition and Prediction and Forecasting in financial markets.

2.4.2.3 Hybrid Neuro-fuzzy Systems

Core Principle Hybrid methods in neuro-fuzzy systems integrate techniques from different computational approaches, such as combining fuzzy logic with evolutionary algorithms or neural networks with decision trees. This integration aims to leverage the strengths of each method to improve system performance.

Application An example is a neuro-fuzzy system enhanced with a decision tree for feature selection, improving the interpretability and efficiency of the model, particularly useful in complex scenarios where data interpretation is crucial. ANFIS, FALCON, GARIC, and SONFIN are a few examples of it [53]. A few applications of this model listed in [42] are,

Monitoring Sensors in Nuclear Power Plants An ANFIS-based PSO technique is employed to enhance safety and operational efficiency in nuclear facilities [36].

Business Failure Prediction This approach provides critical insights into financial stability and risk management [44].

Channel Equalization Improves signal quality and reduces transmission errors in telecommunications [17].

Electricity Price Forecasting A method developed for more accurate forecasting of electricity prices [47].

Wind Power Forecasting Essential for effective energy resource management [16].

Financial Forecasting Enhances the accuracy of financial forecasting using a quantum-behaved PSO approach [15].

2.4.2.4 SVM-based Neuro-fuzzy Systems

Core Principle Support Vector Machines (SVM) are known for their ability to find a hyperplane that best separates different classes by maximizing the margin between the closest points of the classes (support vectors). In neuro-fuzzy contexts, SVMs can optimize the classification boundaries using fuzzy membership functions.

Application SVM-based neuro-fuzzy systems are effective in classification tasks where clear margin separation is beneficial, such as in image recognition or biometric authentication, providing a robust mechanism against overfitting in high-dimensional spaces.

2.4.2.5 Population-based Neuro-fuzzy Systems

Core Principle These methods use a set of potential solutions to explore the solution space. Each member of the population represents a possible solution, and through processes like mutation, crossover, and selection, the population evolves towards an optimal set of parameters. These algorithms are effective in noisy or uncertain environments.

Types of Population-Based Algorithms

- **Genetic Algorithms (GA)** These algorithms use processes such as selection, mutation, and crossover. They are advantageous because they operate on encoded data, which allows them to function independently of the data's continuity and derivative requirements.

- **Differential Evolution (DE)** Known for its simplicity and effectiveness, DE employs strategies involving mutation, crossover, and selection without needing derivative information, making it suitable for complex optimization problems.

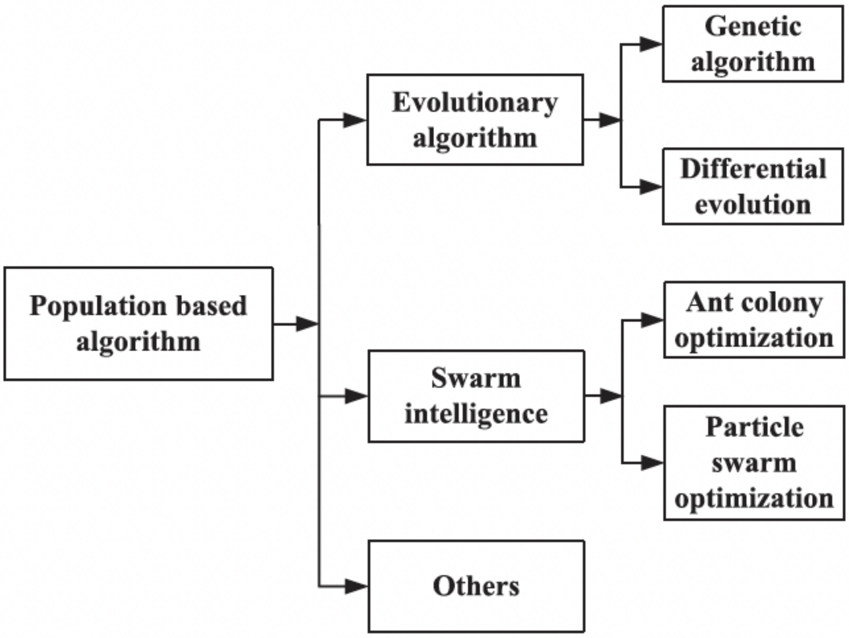

Figure 7. Classification of the population-based techniques [43]

- **Particle Swarm Optimization (PSO)** Inspired by social behaviours like bird flocking, PSO is noted for its easy implementation, minimal memory requirements, and rapid convergence.
- **Grey Wolf Optimizer (GWO)** This algorithm draws inspiration from the social structure and hunting strategies of grey wolves, presenting a novel approach to optimization. GWO embodies the hierarchical organization observed within wolf packs, where a leader, known as the alpha wolf, dictates the pack's movements and decisions. Supporting the alpha are beta wolves, acting as advisors, followed by delta and omega wolves, each with distinct roles in the hunting process. In the pursuit of prey, alpha, beta, and delta wolves identify vulnerable targets, while the omega wolves coordinate their efforts based on the leaders' cues. By encircling and wearing down their prey before launching a coordinated attack, the pack maximizes its hunting success. a compelling choice for optimization tasks in diverse domains
- **Ant Colony Optimization (ACO)** Based on the foraging behaviour of ants, ACO excels in problems with multiple optimum solutions due to its collective problem-solving approach.

Application Genetic algorithms are a typical example, used to optimize the rule base of a fuzzy system. They are particularly useful in environments where the search space is large and the global optimum is difficult to find using traditional methods.

3 Literature Review

To conduct a comprehensive review, we began by selecting recent research papers that focused on classification using neuro-fuzzy systems. We first performed a database search, targeting major academic databases such as Scopus and Web of Science. The search was conducted using specific keywords, including *"neuro-fuzzy systems," "classification,"* and *"fuzzy logic."* We then applied our inclusion criteria, selecting studies published within the last 10 years to ensure that our analysis focused on the most recent methods and trends in the field. Only papers that used commonly adopted datasets and reported performance metrics like accuracy were included. Conversely, we excluded papers if their titles were misleading, if they did not employ neuro-fuzzy methods for classification, or if their content was not closely aligned with the specific topics we were examining.

Many different approaches have been studied to improve the performance and interpretability of neuro-fuzzy logic approaches in other domains. Many researchers have compared the performance of their model with others. However, they are limited to one domain like health. An extensive comparison of performance through several datasets is still missing.

In [32], the authors used TSK-based fuzzy systems to tune the parameters of Particle Swarm Optimisation (PSO), as many nature-inspired algorithms like Genetic Algorithm (GA), Neural Networks, Swarm Intelligence failed to provide any explainable intelligence [13].

In a comparative study, [35] assesses the performance of three optimization algorithms —type-2 fuzzy particle swarm optimization (PSO), bee colony optimization (BCO), and bat algorithm (BA)—in the optimization of fuzzy controllers. Through experiments conducted on benchmark functions, the authors analyze factors such as convergence speed, solution quality, and robustness to parameter variations. The study reveals that each algorithm exhibits distinct strengths and weaknesses. Type-2 fuzzy PSO demonstrates robust convergence properties and resilience to parameter variations, making it well-suited for complex optimization tasks. Bee colony optimization demonstrates competitive performance, particularly in solution quality and exploration-exploitation balance. While the Bat algorithm may be less effective in some scenarios, it offers unique exploration capabilities. This comparative analysis provides valuable insights into the efficacy of different optimization algorithms for improving the performance of fuzzy controllers. Such insights are crucial for selecting the most appropriate algorithm based on the specific requirements of a given optimization task. The study concluded that PSO outperforms other algorithms in controlling the trajectory of an autonomous robot. Since PSO is one of the most common Population-based methods used and GWO outperforms it, it is imperative to mention the idea behind GWO here [13].

[13] presents a technique for the development of Mamdani fuzzy rules to obtain accurate data classification based on a fuzzy evolutionary system using the GWO algorithm. GWO algorithm is developed after the hunting behaviour of gray wolves that mimics the patterns of movements by packs and takes guidance from leader wolves to identify optimal positions for prey encirclement and attack. [13] also evaluated the performance of PSO and concluded that GWO outperforms PSO as it converges in a lesser number of iterations and less time, along with having better accuracy.

The numerical section of [13], where the authors have performed a comparative analysis of the GWO and PSO algorithms across 15 datasets to evolve fuzzy rules. A fundamental distinction between PSO and GWO lies in the updating mechanism: in PSO, the position of the swarm particles updates based on local neighbours and global leaders. The document does not delve into PSO extensively as no recent studies focus on it. According to [13], the findings indicate that the GWO algorithm requires fewer iterations to converge and achieves higher accuracy compared to PSO.

In their paper, [41] introduced FAME (Fuzzy logic-based Adaptive selection of Operators Multi-objective Evolutionary Algorithm) that used fuzzy logic to dynamically adapt the selection of evolutionary operators based on the current state of the optimization process. The authors tested this algorithm on a number of problems and concluded that FAME achieves a competitive performance when compared to other popular evolutionary algorithms. In particular, FAME illustrates

1. higher-quality solutions are produced by FAME for a variety of multi-objective optimization issues.

2. increased speed of convergence, the method converges effectively to near-optimal or optimum solutions.

3. robustness, FAME performs and is stable in a variety of problem domains and optimization settings.

To explain the behaviour of an unmanned aerial vehicle when it deviates from its planned path while in the incognito mode, writers in [28] developed fuzzy rules using a neuro-fuzzy technique. The authors used interpretable and explainable artificial intelligence (XAI) models for unmanned aerial vehicles (UAVs) to show how AI systems used in high-stake sectors need to be transparent and interpretable.

In [26], authors argued that the problem of inferring over a large set of rules is not widely discussed. The authors believed the problem was noticed until huge datasets such as big data were used [22][23]. They came up with a solution to this problem by avoiding reviewing all the rules and focusing only on the 'best' rule in the neighbourhood.

Predictions or choices based on fuzzy logic in the context of fuzzy rule-based inference systems require a few crucial steps. Initially, distinct states or degrees of a phenomenon are represented by linguistic variables and the fuzzy domains that go along with them. Fuzzy rules are created by combining these variables, each of which specifies criteria and results together with weights that indicate how important they are. After the construction of rules, a subset—typically derived from rule-learning algorithms—is chosen for inference. The inference process starts by assessing how well the new data fits the requirements of each rule. The degree of alignment between each fuzzy label and the data is shown by the minimum of membership values, which is one way to evaluate this fit. The procedure then combines each rule's weight and fit using the proper operator, taking into account rule weights that indicate their relative importance. This makes it easier to choose the rule that best fits the available data while taking its relevance and fit into account.

In the end, the inference procedure is similar to looking for the best rule that maximises the combined fit and weight and most closely matches the available data. Even with complex datasets, this optimisation approach guarantees that the best rule is chosen for predictions or choices based on fuzzy logic principles. This brief introduction to fuzzy rule-based inference highlights the need to optimise rule selection for best results [26]. In the paper, the authors argued that to improve the inference process, it is imperative to improve the search process. They analysed the performance of 4 different algorithms: Linear search, Linear Search with pruning, Backtracking search with pruning in the neighbourhood of the example and Heuristic backtracking search with pruning in the nearby neighbourhood of the example to see the difference in performance.

50 classification problems were selected and arranged in 4 different groups using the χ algorithm based on the number of variables and rules. According to the study, the recursive technique works better in scenarios with a low number of continuous variables, whereas the sequential approach performs better when there are few rules and many continuous variables. Furthermore, to address cases with a large number of rules and continuous variables, a hybrid inference algorithm integrating both techniques was devised, leading to increased efficiency.

To address the problem of Interpretability, authors of [21] proposed null-unineuron, a novel logical fuzzy neuron built on the idea of a null-uninorm. This novel neuron was included in a dynamic neuro-fuzzy model architecture EFNN-NullUni, which allowed complex fuzzy rules with AND and OR connections of antecedents to be extracted. The model improved the interpretability and comprehension of the analyzed problem by permitting these flexible links.

The three-layer suggested design uses an evolving weighted fuzzification technique and progressive data partitioning techniques to extract knowledge. Within this architecture, the null-unineurons are trained incrementally and online for the classification of binary and multiclass patterns.

In the evolving data partitioning algorithm, weights were assigned based on feature importance levels. These weights facilitated the automatic adjustment of distance calcula-

tions, particularly diminishing the impact of unimportant input directions (features). This adaptive mechanism contributed to a softer dimension reduction and mitigated the risk of overfitting. This architecture proposed by authors was subject to pattern classification tests, being more efficient compared to related (evolving) neuro-fuzzy models in the literature. The authors came to a conclusion that this approach is solely designed for classification problems and does not perform an explicit (iterative) selection of neurons. The authors of [21] also worked on an advanced interpretable Fuzzy Neural Network model based on uni-nullneuron constructed from n-uninorms UNInull FNN in [20]. The paper presented a fuzzy logic neuron architecture that leverages n-uninorms to construct uni-nullneurons. This architecture, termed a fuzzy neural network (FNN), offers enhanced flexibility by seamlessly integrating nullnorms and uninorms at different stages, thereby enhancing model accuracy and facilitating more intricate rule connections, including both *AND* and *OR* connections within a single rule. The incorporation of such flexible connections enables experts and operators to glean deeper insights from data, thereby enriching the knowledge extraction process. Both these approaches are discussed later in detail with experimental results.
In the paper, [43], the authors reviewed different neuro-fuzzy systems based on the classification of research articles from 2000 to 2017. The primary goal of this survey was to give the reader different neuro-fuzzy systems to facilitate the identification of methods that align with their specific research interests. The authors made a detailed summary of models based on learning algorithms, fuzzy methods and structure.

4 Methodology and Results

In this section, we carry out a quantitative analysis of different fuzzy systems used for classification tasks. Here's a breakdown of the steps we followed in the analysis:

1. **Preparation of Datasets** The datasets used in the experiments were obtained from the UCI Machine Learning Repository and Kaggle. The preprocessing of these datasets mentioned in Table 1 the following steps were involved:

   - **Data Cleaning** All string-based values were converted to integers to ensure compatibility with the algorithms used. For datasets with continuous variables, floating-point values were appropriately scaled or normalized.
   - **One-Hot Encoding** For multi-class classification tasks, we used one-hot encoding to transform the output labels. This means we converted each categorical class label into a binary vector, where each class was represented by its own column, with binary values indicating whether or not a data point belongs to that class.

     The one-hot encoding representation would look like this:

     – Class 1: $[1, 0, 0]$
     – Class 2: $[0, 1, 0]$
     – Class 3: $[0, 0, 1]$

     In this representation:

     – $[1, 0, 0]$ indicates that the instance belongs to Class 1.
     – $[0, 1, 0]$ indicates that the instance belongs to Class 2.
     – $[0, 0, 1]$ indicates that the instance belongs to Class 3.

   The preprocessing steps for each dataset, along with their descriptions, are provided in a later section.

2. **Experimental Setup** In our experimental setup, we systematically tested the performance of the ANFIS-T and CANFIS-T across various datasets. The setup involved:

   - **Parameter Tuning** We varied the regularization parameters to identify the best combinations that yield the highest accuracy.

     – Standard Regularization Parameter C: Tested values were $C \in \{10^{-1}, 1, 10, 10^2\}$.
     – Fractional Regularization Parameter a: Tested values were $a \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$.

- **Algorithm Implementation** The selected neuro-fuzzy systems were implemented in Scilab, with accuracy computed for each configuration.

3. **Performance Evaluation** The classification performance of each system was evaluated using standard metrics such as accuracy as it was the one metric common in all papers. Accuracy in Table 15 and Table 16 defined as:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{12}$$

where TP denotes true positives, TN denotes true negatives, FP denotes false positives, FN denotes false negatives.

The different neuro-fuzzy systems were then ranked by their performance using Friedman's test, which is discussed in more detail later in Table 18 and 19.

4.1 Datasets

The datasets are listed in Table 1. In description, "D" stands for multivalued discrete and "C" for continuous.

1. **Haberman's Survival Dataset** [5] contains survival data of breast cancer patients post-surgery from a study conducted at the University of Chicago's Billings Hospital between 1958 and 1970. No adjustments were made to the data before running the experiments.

2. **Cryotherapy Dataset** [4] provides data on cryotherapy treatment outcomes for 90 patients, focusing on wart removal effectiveness. The column storing time was rounded off to the nearest integer value.

3. **Heart Disease Dataset** [6] includes four databases related to heart disease diagnosis, with all numerical attributes, aiding in disease prediction models. The Cleveland dataset is selected for the purpose of running the tests.

4. **Autism Screening Data for Toddlers** [1] includes behavioural features from the Q-Chat-10 assessment and additional characteristics effective in autism spectrum disorder (ASD) detection in toddlers. All the string values, such as ethnicity, gender, and output, were coded as integer values as part of preprocessing.

5. **Immunotherapy Dataset** [7] is from the field of health and medicine that contains immunotherapy results for 90 patients. The time column for patients was rounded to the nearest integer value.

6. **Iris Dataset** [8] features data from three Iris plant species, classified into three different classes based on morphological data. The classes mentioned in the dataset were converted from string to one-hot encoding for experiments for CANFIS-T.

7. **Thyroid Dataset** [11] is provided by Garavan Institute having different datasets for thyroid data. ANN data was selected for experiments and floating values were multiplied with a constant value along with representing the classes with three columns for one hot encoding.

8. **Wine Dataset** [12] presents chemical analysis results of wines from three different cultivars in the same Italian region, detailing 13 constituents in each wine type. The floating values in the dataset were multiplied with a constant and classes were represented as three new columns for one hot encoding.

9. **Pima Indians Diabetes Dataset** [10] comprises multiple medical predictor variables alongside a single target variable, labeled "Outcome." The independent variables encompass the patient's number of pregnancies, Body Mass Index (BMI), insulin levels, and age, among other factors. The values in BMI were rounded to the nearest integer value and DiabetesPedigreeFunction were multiplied by a factor of 1000 as part of preprocessing.

10. **Balance Scale Dataset** [2] was created to model psychological experimental results. Each example is classified as having the balance scale tip to the right, tip to the left, or remain balanced. The attributes include the left weight, left distance, right weight, and right distance. The correct classification is determined by comparing (left distance * left weight) and (right distance * right weight). If the values are equal, the scale is balanced. The classes were represented as one hot encoding and four new columns were added for the same as part of preprocessing before using it with CANFIS-T.

11. **Lymphography Dataset** [9] was obtained from the University Medical Centre, Institute of Oncology in Ljubljana, Yugoslavia. For preprocessing the classes were replaced with one hot encoding and four new columns were added for the same before using it with CANFIS-T.

12. **Car Evaluation** [3], derived from a hierarchical model by M. Bohanec and V. Rajkovic, evaluates car acceptability based on six attributes: buying price, maintenance cost, number of doors, passenger capacity, luggage boot size, and safety. This dataset is useful for testing constructive induction and structure discovery methods. The original dataset contained values as string that were coded to integers and the class column was replaced with 4 more columns for one hot encoding before using it with CANFIS-T.

Table 1. Characteristics of datasets used as classification benchmarks in the experiments.

Dataset	Acronym	Classes	Features
Haberman	HAB	2	3
Cryotherapy	CRI	2	6
Heart	HEA	2	13
Autism	AUT	2	18
Immunotherapy	IMU	2	7
PIMA Diabetes	PIM	2	9
Iris	IRI	3	4
Thyroid	THY	3	5
Wine	WIN	3	13
Balance Scale	BAL	5	4
Lymphography	LYM	4	18
Car Evaluation	CAR	4	6

Table 2. Haberman's Survival Data Attributes

Variable	**Description**	**Type**
Age	Age of patient at time of operation	D
Year	Patient's year of operation	D
Nodes	Number of positive axillary nodes detected	D
Status	Survival status: 1. 1 = the patient survived 5 years or longer 2. 2 = the patient died within 5 years	D

Table 3. Cryotherapy Data Attribute

Variable	Description	Type
Sex	-	D
Age	-	D
Time	-	C
Number of Wrats	-	D
Type	-	D
Area	-	D
Result	-	D

Table 4. Heart Disease Data Attributes

Variable	Description	Type
Age	Age of patient	D
Sex	Gender	D
cp	-	D
trestbps	Resting blood pressure (on admission to the hospital)	D
chol	Serum cholesterol in mg/dl	D
fbs	Fasting blood sugar > 120 mg/dl (1 = true; 0 = false)	D
restecg	-	D
thalach	Maximum heart rate achieved	D
exang	Exercise induced angina (1 = yes; 0 = no)	D
oldpeak	ST depression induced by exercise relative to rest	D
slope	-	D
ca	Number of major vessels (0-3) colored by fluoroscopy	D
thal	3 = normal; 6 = fixed defect; 7 = reversible defect (Thalassemia)	D
num	Diagnosis of heart disease	D

Table 5. Autism Screening Data Attributes

Variable	**Description**	**Type**
Case No	-	D
A1	-	D
A2	-	D
A3	-	D
A4	-	D
A5	-	D
A6	-	D
A7	-	D
A8	-	D
A9	-	D

Table 6. Immunotherapy Data Attributes

Variable	**Description**	**Type**
Sex	-	D
Age	-	D
Time	-	C
Number of Wrats	-	D
Type	-	D
Area	-	D
Induration diameter	-	D
Result	-	D

Table 7. Iris Data Attribute

Variable	**Description**	**Type**
sepal length	-	C
sepal width	-	C
petal length	-	C
petal width	-	C
class	class of iris plant	D

Table 8. Thyroid Disease Data Attribute

Variable	Description	Type
Class	-	D
Attribute1	-	D
Attribute2	-	C
Attribute3	-	C
Attribute4	-	C
Attribute5	-	D

Table 9. Wine Data Attribute

Variable	Description	Type
Alcohol	-	C
Malic acid	-	C
Ash	-	C
Alcalinity of ash	-	C
Magnesium	-	C
Total phenols	-	C
Flavanoids	-	C
Nonflavanoid phenols	-	C
Proanthocyanins	-	C
Color intensity	-	C
Hue	-	C
OD280/OD315 of diluted wines	-	C
Proline	-	C

Table 10. Pima Indians Diabetes Dataset

Variable	**Description**	**Type**
Pregnancies	Number of times pregnant	D
Glucose	Plasma glucose concentration	D
BloodPressure	Diastolic blood pressure	D
SkinThickness	Triceps skin fold	D
Insulin	2-Hour serum insulin	D
BMI	Body mass index	C
Diabetespedigreefunction	Diabetes pedigree function	C
Age	Age in years	D
Outcome	1 or 0	D

Table 11. Balance Scale Dataset

Variable	**Description**	**Type**
right-distance	L, B, R	D
right-weight	1, 2, 3, 4, 5	D
left-distance	1, 2, 3, 4, 5	D
left-weight	1, 2, 3, 4, 5	D
class	1, 2, 3, 4, 5	D

Table 12. Lymphography Dataset

Variable	Description	Type
Class	-	D
lymphatics	-	D
block of affere	-	D
bl. of lymph. c	-	D
bl. of lymph. s	-	D
by pass	-	D
extravasates	-	D
regeneration	-	D
early uptake in	-	D
lym.nodes dimin	-	D
lym.nodes enlar	-	D
changes in lym.	-	D
defect in node	-	D
changes in node s	-	D
changes in stru	-	D
special forms	-	D
dislocation of	-	D
exclusion of no	-	D
no. of nodes	-	D

Table 13. Car Evaluation Dataset

Variable	Description	Type
buying	v-high, high, med, low	D
maintenance	v-high, high, med, low	D
doors	2, 3, 4, 5-more	D
persons	2, 4, more	D
lug boot	small, med, big	D
safety	ow, med, high	D
class	unacc, acc, good, vgood	D

4.2 Results and Discussion

Table 14. Neuro fuzzy system and their acronyms

Acronym	Neuro fuzzy System
EFFN: NullUni	Evolving fuzzy neural networks - null-unineuron
UNInull FNN	Evolving fuzzy neural networks- unieuron
ALMMo-0*	Autonomous Learning Multiple-Model Zero-Order Classifier
ANFIS-T	Adaptive Neuro-Fuzzy Inference System with Fractional Tikhonov regularization
EFNHN	Evolving fuzzy neural hydrocarbon networks
EFNN	Evolving fuzzy neural networks
UNFIS-c	Unstructured neuro-fuzzy inference system
Mamdani-GWO	Mamdani FIS with Grey Wolf Optimizer
Mamdani-PSO	Mamdani FIS with Particle Swarm Optimization
CANFIS-T	Co-Adaptive Neuro-Fuzzy Inference System with Fractional Tikhonov regularization

Table 15 and Table 16 present the accuracy results for binary and multi-classification across multiple datasets, respectively. The performance of the ANFIS-T algorithm, tried for the first time in this thesis, is detailed in the context of binary classification, while the performance of the CANFIS-T algorithm, also tried for the first time in this thesis, is highlighted in the multi-classification context. These results are compared with those obtained from other researchers (Table 14), providing a comprehensive view of the technique's accuracy. Since several of the reviewed papers did not provide the number of rules for each dataset, this information has been excluded from the tables.
Table 17 shows the values of C (standard regularisation parameter) and a (fractional regularisation parameter) that resulted in the best accuracy.

Table 15. Accuracy results for binary classification Across Multiple Datasets

Methods	HAB	CRI	HEA	AUT	IMU	PIM
ANFIS-T [50]	**99**	79.44	80.12	**99**	78.88	**77.18**
EFNN: NullUni [21]	76.06	81.26	80.12	95.98	**86.14**	-
UNInull FNN [20]	73.36	82.46	66.17	76.37	80.12	-
ALMMo-0* [14]	71.25	78.25	71.12	79.99	78.25	-
EFNHN [46]	58.07	70.98	63.25	77.61	65.06	-
EFNN [45]	73.11	81.48	78.11	81.36	76.29	-
UNFIS-c [40]	74.60	**84.69**	**81.26**	98.03	73.45	-
Mandami-GWO [13]	74.84	-	-	-	-	71.11
Mandami-PSO [13]	73.18	-	-	-	-	69.62

Table 16. Accuracy Results for multi-classification across multiple datasets

Methods	IRI	WIN	THY	LYM	BAL	CAR
EFNN: NullUni [21]	86.14	61.13	89.54	-	-	-
UNInull FNN [20]	90.12	65.34	85.07	-	-	-
ALMMo-0* [14]	78.25	62.66	81.85	-	-	-
EFNN [45]	79.16	47.32	82.46	-	-	-
UNFIS-c [40]	73.25	84.44	**94.46**	-	-	-
Mamdani-GWO [13]	**96.29**	56.24	84.86	11.38	69.90	73.44
Mamdani-PSO [13]	94.29	55.82	82.86	11.39	68.96	72.05
CANFIS-T [51]	66.67	**97.64**	88.27	**98.57**	**80.48**	**86.10**

Given the limited information available in the reviewed literature, it was not possible to provide a complete report on the number of rules used. However, in the study [13] that utilized the Iris dataset, it is clearly stated that 8 rules were applied. On the other hand, CANFIS-T used only 3 rules for the same dataset, as shown in Table 17. Interestingly, Mamdani-GWO [13] emerged as the top performer on the Iris dataset, surpassing all other methods, while CANFIS-T [51] fell short, delivering the weakest results.

Table 17. Parameters used for ANFIS-T and CANFIS-T

Dataset	Method	Classification	C	a	Number of rules
HAB	ANFIS-T	Binary	0.5	1	3
CRI	ANFIS-T	Binary	0.5	1	3
HEA	ANFIS-T	Binary	0.5	1	3
AUT	ANFIS-T	Binary	1	1	3
IMU	ANFIS-T	Binary	0.5	1	3
PIM	ANFIS-T	Binary	0.5	1	3
IRI	CANFIS-T	Multi-class	1	1	3
WIN	CANFIS-T	Multi-class	0.3	0.9	3
THY	CANFIS-T	Multi-class	1	1	4
LYM	CANFIS-T	Multi-class	0.3	0.3	3
BAL	CANFIS-T	Multi-class	0.6	1	5
CAR	CANFIS-T	Multi-class	0.4	1	3

The ANFIS-T algorithm demonstrated high and consistent accuracy across all binary classification datasets, confirming its robustness and reliability. This makes ANFIS-T a strong choice for binary classification tasks.

CANFIS-T showed variable performance across different multi-classification datasets, indicating strong potential for specific applications but also suggesting the need for further refinement to enhance consistency.

We applied Friedman's method to rank various neuro-fuzzy system methods for both binary and multi-class classification based on their performance across different datasets. This non-parametric statistical test is suitable for our data, which involves repeated measures across multiple conditions.

For binary classification, we selected the datasets HAB, CRI, HEA, AUT, and IMU to evaluate the performance of the following methods ANFIS-T, EFNN: NullUni, UNIfull FNN, ALMMo-0, EFNHN, EFNN, and UNFIS-c. The analysis using Friedman's method revealed significant differences between the neuro-fuzzy systems evaluated. Specifically, EFNN: NullUni and ANFIS-T demonstrated superior performance across all datasets.

Table 18. Friedman's ranking for binary classification

Methods	Rank
EFNN: NullUni	1
ANFIS-T	2
UNFIS-c	2
EFNN	4
UNIfull FNN	5
ALMMo-0	6
EFNHN	7

For multi-class classification, we assessed the methods EFNN NullUni, UNINull, ALMMoO-9a, EFNN+, EFNN, UNFIS-c, Mamdani-GWO, Mamdani-PSO, and CANFIS-T using the datasets IRI, WIN, and THY. Although the results of Friedman's method indicated that the differences are not statistically significant at the 0.05 level, the rankings still provide valuable insights. CANFIS-T exhibited superior performance across all datasets.

Table 19. Friedman's ranking for multi-class classification

Methods	Rank
CANFIS-T	1
UNFIS-c	2
UNIfull	3
EFNN: NullUni	4
Mamdani-GWO	5
Mamdani-PSO	5
ALMMoO-9a	7
EFNN	8

The variation of accuracy in different datasets with varying C with a=1 is shown in Figure 8 and variation in accuracy with varying a with different values of C is shown in Figure 9, Figure 10 and Figure 11. The number of terms was also tested to see the impact on accuracy. The number of terms with the best accuracy is listed in Table 17. The number of terms were selected from 2 to 6. It can be seen from the graphs, variation in C after C=1, does not show any additional benefit in achieving more accuracy.

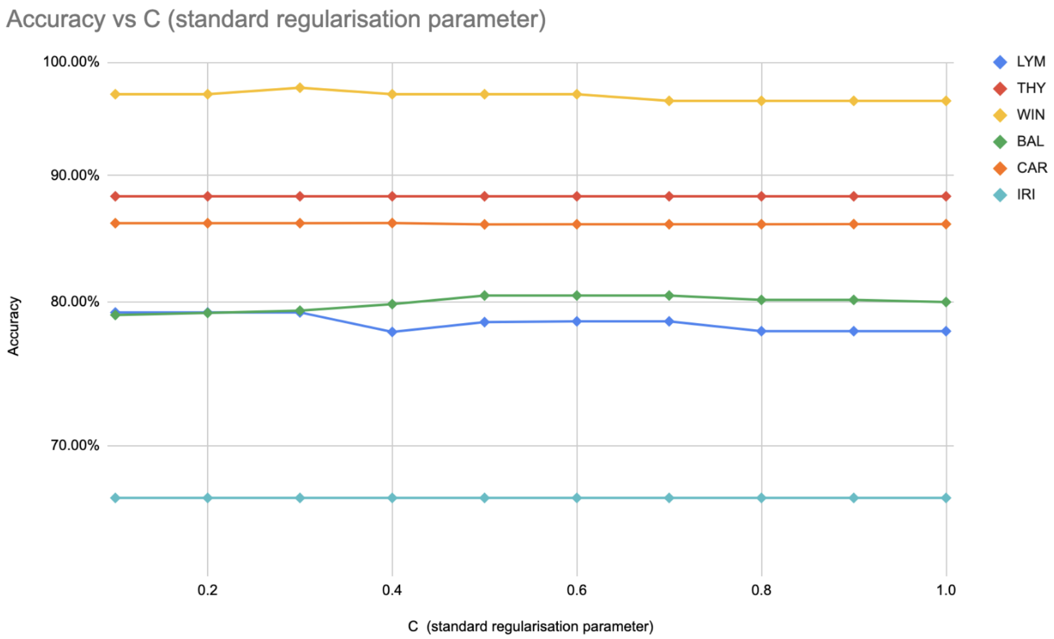

Figure 8. Variation of accuracy with C (standard regularisation) for CANFIS-T

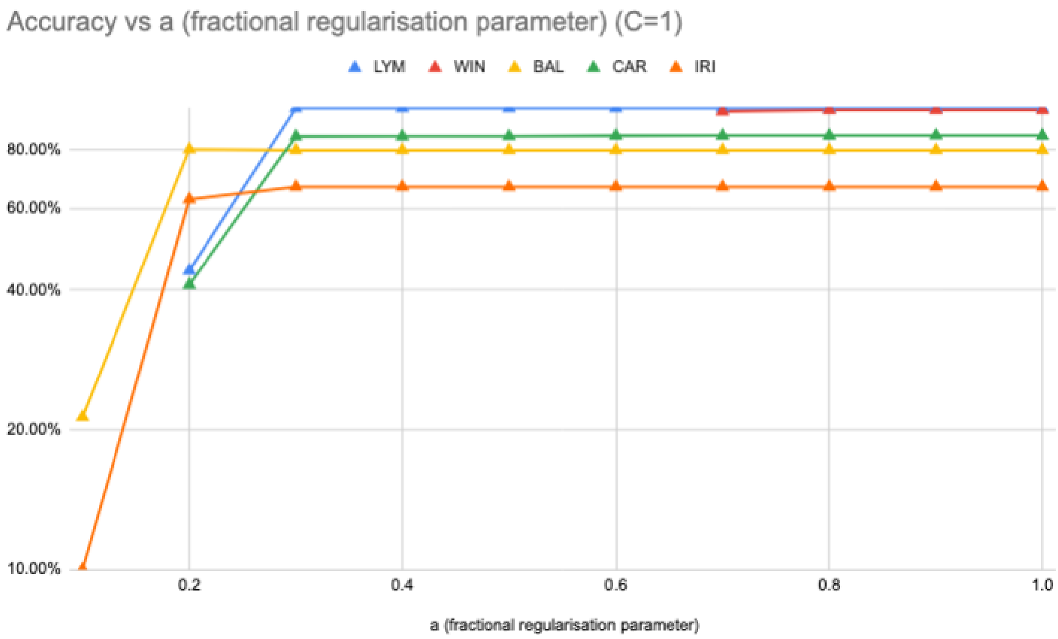

Figure 9. Variation of accuracy with *a* (fractional regularisation) for CANFIS-T (C=1)

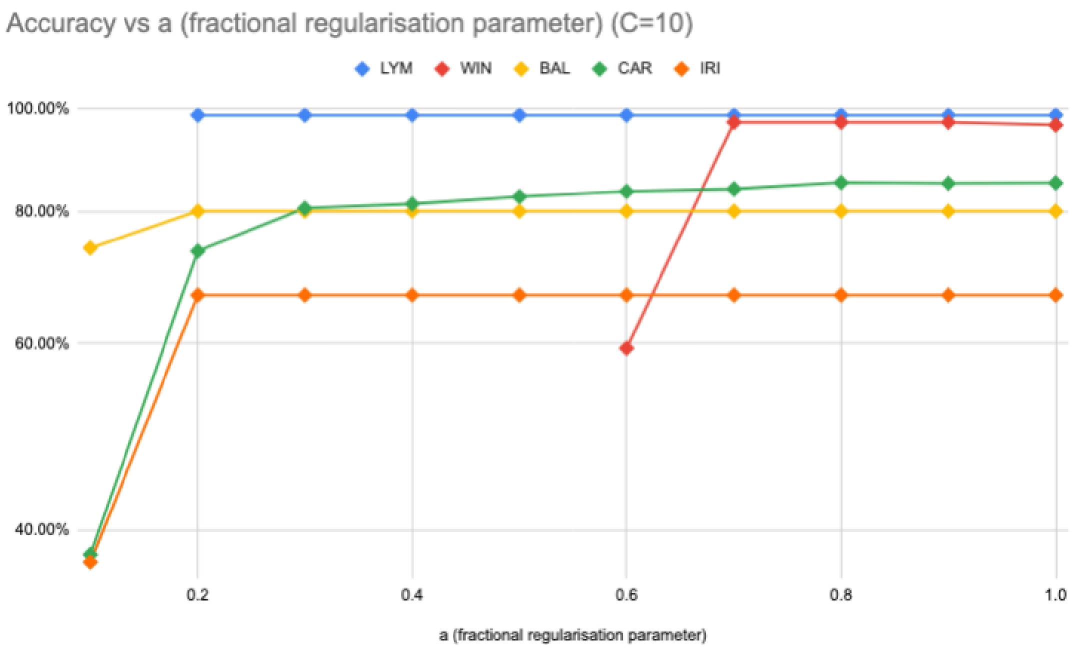

Figure 10. Variation of accuracy with *a* (fractional regularisation) for CANFIS-T (C=10)

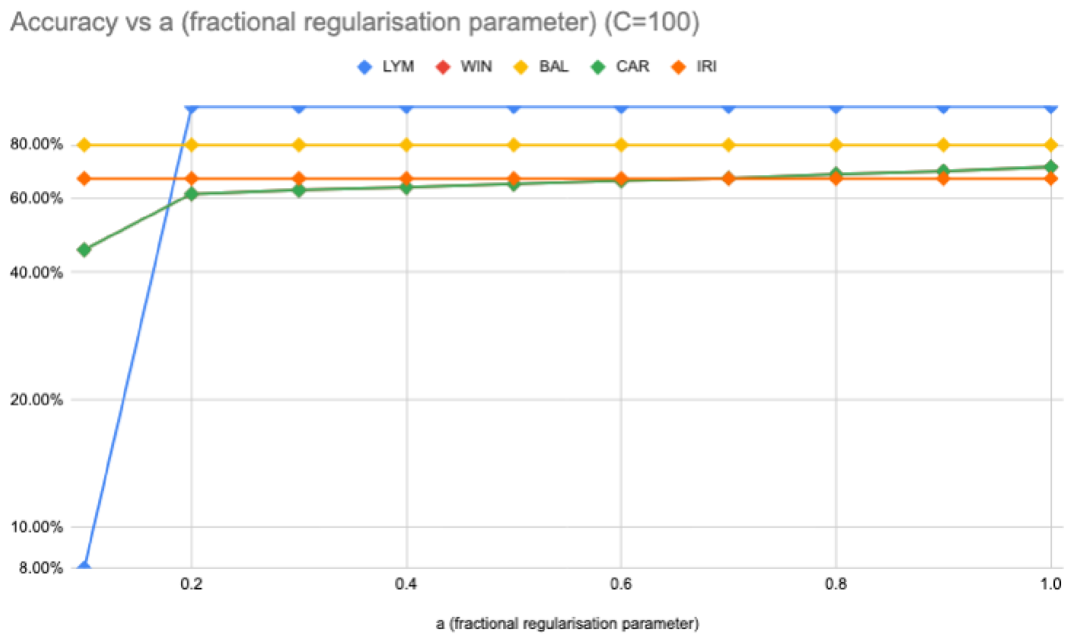

Figure 11. Variation of accuracy with *a* (fractional regularisation) for CANFIS-T (C=100)

5 Concluding remarks

In summary, this thesis has explored different fuzzy inference systems for solving the challenging classification problem. Twelve publically available datasets (Table 1) from different domains were selected and the accuracy of the most recent approaches was compared to help the user make a proper decision when selecting an approach for his problem. ANFIS-T and CANFIS-T were also tested on the same datasets for binary classification and multi-class classification respectively and the variation in accuracy was studied with varying values of C, standard regularization parameter, and a, fractional regularization parameter.

In most cases, the highest accuracy is achieved when C is set between 0.5 and 1, and a is set to 1. Increasing the value of C beyond this range does not provide any additional benefit, while lower values of C lead to a significant drop in accuracy.

To enable readers to make a more informed decision when selecting an approach for their classification problem, it is important to compare various metrics such as precision, F-measure, and computational time for different approaches. This detailed comparison will provide a clearer understanding of each method's performance and suitability for specific use cases, but this could not be reported in this thesis due to missing values.

References

[1] Autism screening dataset. https://www.kaggle.com/datasets/fabdelja/autism-screening-for-toddlers/data.

[2] Balance scale dataset. https://archive.ics.uci.edu/dataset/12/balance+scale.

[3] Car evaluation dataset. https://archive.ics.uci.edu/dataset/19/car+evaluation.

[4] Cryotherapy dataset. https://archive.ics.uci.edu/dataset/41/cryotherapy.

[5] Haberman's survival dataset. https://archive.ics.uci.edu/dataset/43/haberman+s+survival.

[6] Heart disease dataset. https://archive.ics.uci.edu/dataset/45/heart+disease.

[7] Immunotherapy dataset. https://archive.ics.uci.edu/dataset/428/immunotherapy+dataset.

[8] Iris dataset. https://archive.ics.uci.edu/dataset/53/iris.

[9] Lymphography dataset. https://archive.ics.uci.edu/dataset/63/lymphography.

[10] Pima indians diabetes database. https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database.

[11] Thyroid disease dataset. https://archive.ics.uci.edu/dataset/102/thyroid+disease.

[12] Wine dataset. https://archive.ics.uci.edu/dataset/109/wine.

[13] Musbah Abdulgader and Devinder Kaur. Evolving mamdani fuzzy rules using swarm algorithms for accurate data classification. *IEEE Access*, 7:175907–175916, 2019.

[14] S Aeberhard, D Coomans, and O De Vel. Comparison of classifiers in high dimensional settings. *Dept. Math. Statist., James Cook Univ., North Queensland, Australia, Tech. Rep*, 92(02), 1992.

[15] A. Bagheri, H. Mohammadi, and M. Akbari. Financial forecasting using anfis networks with quantum-behaved particle swarm optimization. *Expert Systems with Applications*, 41(14):6235–6250, 2014.

[16] J. P. S. Catalao, H. M. I. Pousinho, and V. M. F. Mendes. Hybrid wavelet-pso-anfis approach for short-term electricity prices forecasting. *IEEE Transactions on Power Systems*, 26(1):137–144, 2011.

[17] M. Y. Chen. A hybrid anfis model for business failure prediction utilizing particle swarm optimization and subtractive clustering. *Information Sciences*, 220:180–195, 2013.

[18] Trinity College. Fuzzy sets and rule bases.

[19] Andrea Dal Pozzolo et al. Credit card fraud detection: A realistic modeling and a novel learning strategy. *IEEE Transactions on Neural Networks and Learning Systems*, 29(8):3784–3797, 2017.

[20] Paulo Vitor de Campos Souza and Edwin Lughofer. Identification of heart sounds with an interpretable evolving fuzzy neural network. *Sensors*, 20(22):6477, 2020.

[21] Paulo Vitor de Campos Souza and Edwin Lughofer. Efnn-nulluni: An evolving fuzzy neural network based on null-uninorm. *Applied Soft Computing*, 67:285–295, 2022.

[22] S. del Río, V. López, J.M. Benítez, and F. Herrera. A mapreduce approach to address big data classification problems based on the fusion of linguistic fuzzy rules. *International Journal of Computational Intelligence Systems*, 8(3):422–437, 2015.

[23] M. Elkano, M. Galar, J. Sanz, and H. Bustince. Chi-bd: A fuzzy rule-based classification system for big data classification problems. *Fuzzy Sets and Systems*, 348(1):75–101, 2018.

[24] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: Data mining, inference, and prediction*. Springer, 2009.

[25] Jyh-Shing Roger Jang. Anfis: Adaptive-network-based fuzzy inference system. *IEEE transactions on systems, man, and cybernetics*, 23(3):665–685, 1993.

[26] Leonardo Jara, Rubén Ariza-Valderrama, Juan Fernández-Olivares, Antonio González, and Raúl Pérez. Efficient inference models for classification problems with a high number of fuzzy rules. *Applied Soft Computing*, 115:108164, 2022.

[27] Ujjwal Karn. A quick introduction to neural networks, 2016. Accessed: 2024-05-06.

[28] B. M. Keneni, D. Kaur, A. Al Bataineh, V. K. Devabhaktuni, A. Y. Javaid, J. D. Zaientz, and R. P. Marinier. Evolving rule-based explainable artificial intelligence for unmanned aerial vehicles. *IEEE Access*, 7:17001–17016, 2019.

[29] Anatoly A Kilbas, Hari M Srivastava, and Juan J Trujillo. *Theory and Applications of Fractional Differential Equations*. Elsevier, 2006.

[30] George J. Klir and Bo Yuan. *Fuzzy sets and fuzzy logic: theory and applications*. Prentice Hall, 1995.

[31] Sotiris B Kotsiantis. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, pages 3–24, 2007.

[32] Patricia Melin, Francisco Olivas, Oscar Castillo, Fevrier Valdez, Jesus Soria, and Martha Valdez. Optimal design of fuzzy classification systems using pso with dynamic parameter adaptation through fuzzy logic. *Expert Systems with Applications*, 40(8):3196–3206, 2013.

[33] Sushmita Mitra and Yoichi Hayashi. Neuro-fuzzy rule generation: Survey in soft computing framework. *IEEE Transactions on Neural Networks*, 11(3):748–768, 2000.

[34] Douglas C Montgomery. *Introduction to statistical quality control*. John Wiley & Sons, 2017.

[35] F Olivas, L Amador-Angulo, J Perez, C Caraveo, F Valdez, and O Castillo. Comparative study of type-2 fuzzy particle swarm, bee colony and bat algorithms in optimization of fuzzy controllers. *Algorithms*, 10(3):101, 2017.

[36] M. V. Oliveira and R. Schirru. Applying particle swarm optimization algorithm for tuning a neuro-fuzzy inference system for sensor monitoring. *Progress in Nuclear Energy*, 51:177–183, 2009.

[37] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1-2):1–135, 2008.

[38] Igor Podlubny. *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*. Academic press, 1999.

[39] Jaynath Prakash, David Sanny, Siva K. Kalva, Manojit Pramanik, and Prasanna K. Yalavarthy. Fractional regularization to improve photoacoustic tomographic image reconstruction. *IEEE Transactions on Medical Imaging*, 38(8):1935–1947, August 2019. Epub 2018 Dec 24.

[40] Armin Salimi-Badr. Unfis: A novel neuro-fuzzy inference system with unstructured fuzzy rules. *Neurocomputing*, 579:127437, 2024.

[41] A. Santiago, B. Dorronsoro, A. J. Nebro, J. J. Durillo, O. Castillo, and H. J. Fraire. A novel multi-objective evolutionary algorithm with fuzzy logic based adaptive selection of operators: Fame. *Information Sciences*, 471:233–251, January 2019.

[42] Siti Mariyam Shamsuddin, Zubaidah Ismail, and Abdul Kadir Marsono. A review of neuro-fuzzy systems and its applications in structural engineering. *International Journal of Computational Methods*, 11(01):1330001, 2014.

[43] KV Shihabudheen and GN Pillai. Recent advances in neuro-fuzzy system: A survey. *Knowledge-Based Systems*, 2018:–, 2018.

[44] M. A. Shoorehdeli, M. Teshnehlab, A. K. Sedigh, and M. A. Khanesar. Identification using anfis with intelligent hybrid stable learning algorithm approaches and stability analysis of training methods. *Applied Soft Computing*, 9:833–850, 2009.

[45] Eduardo Soares, Plamen Angelov, and Xiaowei Gu. Autonomous learning multiple-model zero-order classifier for heart sound classification. *Applied Soft Computing*, 94:106449, 2020.

[46] Paulo Souza, Hiram Ponce, and Edwin Lughofer. Evolving fuzzy neural hydrocarbon networks: A model based on organic compounds. *Knowledge-Based Systems*, 203:106099, 2020.

[47] R. L. Squares, F. Link, A. Neural, and R. B. Function. Pso tuned anfis equalizer based on fuzzy c-means clustering algorithm. *International Journal of Electronics and Communications (AEÜ)*, 70:799–807, 2016.

[48] Yijun Sun, Albert KC Wong, and Mohamed S Kamel. Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(04):687–719, 2009.

[49] Stefania Tomasiello, Witold Pedrycz, and Vincenzo Loia. *Contemporary Fuzzy Logic, A Perspective of Fuzzy Logic with Scilab*. Springer, 2022.

[50] Stefania Tomasiello, Witold Pedrycz, and Vincenzo Loia. On fractional tikhonov regularization: Application to the adaptive network-based fuzzy inference system

for regression problems. *IEEE Transactions on Fuzzy Systems*, 30(11):4717–4727, 2022.

[51] Stefania Tomasiello, Muhammad Uzair, Yang Liu, and Evelin Loit. Data-driven approaches for sustainable agri-food: Coping with sustainability and interpretability. *Journal of Ambient Intelligence and Humanized Computing*, 2023.

[52] Muhammad Uzair. *Soft Decision making for Agri-foof 4.0*. PhD thesis, University of Tartu, 2024.

[53] J. Vieira, F. M. Dias, and A. Mota. Neuro-fuzzy systems: a survey. *WSEAS Transactions on Systems*, 3(2):414–419, 2004.

[54] José Vieira, Fernando Morgado Dias, and Alexandre Mota. Artificial neural networks and neuro-fuzzy systems for modelling and controlling real systems: A comparative study. *Engineering Applications of Artificial Intelligence*.

Appendix

I. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Ayushmat Bhardwaj Soni**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Reviewing the classification performance of recent neuro-fuzzy systems,

   supervised by Dr. Stefania Tomasiello and Mr. Muhammad Uzair.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work, from date until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Ayushmat Bhardwaj Soni
date August 13, 2024