Table of Contents

[Part 2 - Demo 3](#_Toc140524206)

[Introduction 3](#_Toc140524207)

[Software to Implement AI 4](#_Toc140524208)

[GUI Design 5](#_Toc140524209)

[Plotting Experimental Results 6](#_Toc140524210)

[Neural Network Output 8](#_Toc140524211)

[Cascade Forward Neural Network (CFNN): 8](#_Toc140524212)

[Feedforward Neural Network (FFNN): 10](#_Toc140524213)

[Comparison of Models 12](#_Toc140524214)

[Fuzzy Logic 13](#_Toc140524215)

[Membership Function 13](#_Toc140524216)

[Fuzzy Logic Model A 14](#_Toc140524217)

[Fuzzy Logic Model B 18](#_Toc140524218)

[Justification 22](#_Toc140524219)

[References 23](#_Toc140524220)

Part 2 - Demo

Introduction  
Several sectors, including agriculture, water management, and outdoor recreation planning, rely heavily on weather forecasting. Accurate weather forecasts facilitate informed decision-making and resource allocation optimisation. In this case study, we present a system that integrates a weather prediction model based on neural networks with a water sprinkler control system based on fuzzy logic.

The system uses meteorological data from the past to train a neural network model. Based on the input features, the trained model can predict meteorological conditions. By accurately predicting meteorological conditions, the system can provide insightful information for decision-making.

In addition, the system includes a sprinkler control system based on imprecise logic. It uses fuzzy logic rules and the forecasted weather conditions to determine the appropriate water supply for irrigation. The fuzzy logic controller considers weather forecasts and other factors, such as water temperature, to optimise irrigation operations by adjusting the water supply level.

This solution seeks to improve water management practises and irrigation efficiency by combining a neural network weather prediction model with a fuzzy logic water sprinkler control system. The precise weather forecasts allow for proactive resource allocation and planning, while the fuzzy logic controller ensures efficient water utilisation based on the weather forecast.

# Software to Implement AI

In this case study, we develop and implement the Neural Network Weather Prediction System with Fuzzy Logic Water Sprinkler Control using a variety of tools and software. These instruments offer crucial capabilities for data processing, model training, evaluation, and system implementation. Here is a summary of the utilised tools and software:

MATLAB: It is an advanced programming environment used extensively in scientific and engineering applications. We utilise MATLAB because of its extensive collection of functions and toolboxes designed specifically for machine learning, neural networks, and fuzzy logic systems. The intuitive and user-friendly interface of MATLAB makes it suitable for the development and implementation of complex systems.

MATLAB App Designer: App Designer is a graphical development environment within MATLAB that enables the design and generation of visually interfaced interactive applications. We use App Designer to create the graphical user interface (GUI) for our system, which enables seamless user interaction with the weather prediction and water irrigation control features.

The Neural Network Toolbox: provides a comprehensive set of tools and functions for designing, training, and evaluating neural networks. We utilise this toolkit to design and configure the neural network architecture for weather forecasting. It provides a variety of activation functions, training algorithms, and performance metrics for modelling neural networks.

MATLAB Fuzzy Logic Toolbox: Fuzzy Logic toolset is a specialised toolset for designing, simulating, and implementing fuzzy logic systems in MATLAB. This toolkit is used to define fuzzy membership functions, develop fuzzy rule-based systems, and conduct fuzzy inference for water sprinkler control. The toolset provides a variety of fuzzy logic modelling functions and visualisation tools.

Data Import and Manipulation: To manipulate the dataset and perform data pre-processing tasks, we employ MATLAB's built-in functions and capabilities. MATLAB offers functions for importing data from a variety of file formats, including CSV files, Excel spreadsheets, and databases. Using MATLAB's data manipulation functions, we can also perform data cleansing, normalisation, and dividing.

We can efficiently develop, train, and evaluate the neural network weather prediction model and implement the fuzzy logic-based water sprinkler control system using these tools and software. The combination of MATLAB, App Designer, Neural Network Toolbox, and Fuzzy Logic Toolbox provides a comprehensive environment for developing intelligent systems and enables us to seamlessly address the challenges of accurate weather prediction and optimal water management.

## GUI Design

A screenshot of a computer

Description automatically generated

Figure Water Sprinkler System GUI

First, the user selects the neural network and fuzzy logic to be used to predict the water output of the sprinkler. The user then selects the month and days for which they wish to predict the system's water output. The water sprinkler system predicts the next day's output using data from the previous day.

The user then pushes the start button to simulate the sprinkler system. Then the system receives the data it needs including temperature, pressure, rainfall, and humidity. Then the data will be sent to two neural networks to predict temperature and precipitation. The predicted value is then sent to fuzzy logic, which determines the amount of water the sprinkler will use.

# Plotting Experimental Results

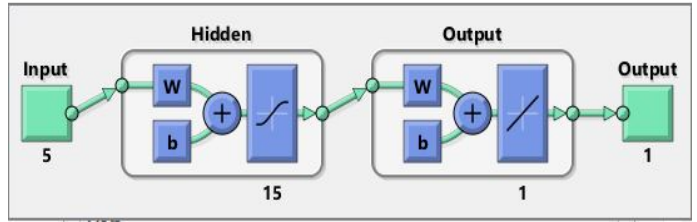
Neural networks have generated considerable interest in the fields of artificial intelligence and machine learning because of their capacity to recognise complex patterns and make accurate predictions. CFNN (Cascade Forward Neural Network) and FFNN (Feedforward Neural Network) are three popular architectures among the numerous types of neural networks. Each of these architectures has its own applications and characteristics.

A diagram of a diagram

Description automatically generated

Cascade Forward Neural Network (CFNN): Cascade forward neural network is the neural network that I chose for predicting the output (temperature, rainfall, and water level) that will be provided to fuzzy logic. It is a neural network like a feed-forward network, but with more connections from the input to each additional layer. The input connects not only to the concealed layer but also to the output layer. As with feed forward neural networks, the back propagation of a CFNN consists of three phases: input pattern, error counting, and weight adjustment. Similarly, to a feed-forward network, a CFNN with two or more layers may discover the relationship between the input and output if given sufficient hidden layers.

Three cascade forward neural networks, each with five unique inputs and one output, were created for the water irrigation system. All networks were equipped with sixteen hidden layers and 8783 hours of input and target data for training. As stated previously, the network utilises the previous hour's data to predict the next hour's data.



**Feedforward Neural Network (FFNN):** FFNN architecture is the simplest and most prevalent form of neural network. It consists of multiple layers of interconnected neurons, with one-way information transmission from the input layer to the output layer. Levenberg-Marquardt backpropagation is used for network training. There are three networks below that are used to predict rainfall and temperature separately. They are trained with 8783 distinct inputs and outputs. The model is trained to predict the next day's rainfall and temperature using data from the previous day.

In conclusion, CFNN and FFNN are two prominent neural network architectures, each suited for distinct data types and duties. CFNNs excel at image and signal processing, whereas FFNNs are adaptable and extensively used for a variety of machine learning tasks. Understanding the characteristics and capabilities of these architectures is essential for selecting the optimal neural network for a specific problem domain.

## Neural Network Output

## Cascade Forward Neural Network (CFNN):

Rainfall

A graph of a number of days and months

Description automatically generated

A graph of a test

Description automatically generated

Figure CFNN- Prediction and MSE of Rainfall

The graph compares the outputs of the predicted rainfall with the actual rainfall over a 31-day period. The actual outputs for the first eight days were stable, and an abrupt change in values and neural network was observed to accurately predict the pattern and the rainfall. Even though the outputs of the predicted rainfall were not identical to the outputs of the actual rainfall, the neural network accurately predicted both the pattern and the rainfall. As depicted in Figure 5, the mean squared error (MSE) for this neural network is 0.13622.

Temperature

A graph of a temperature

Description automatically generated

A graph with a red line and green line

Description automatically generated

Figure 6 CFNN- Prediction and MSE of Temperature

The graph depicts a 31-day comparison between the predicted and actual temperature outputs. Even though the outputs of the predicted temperature were not identical to the outputs of the actual temperature, the neural network was able to predict the pattern as well as the temperature with a higher degree of accuracy than the rainfall. As shown in Figure 7, the mean squared error (MSE) for this neural network is 0.59349.

## Feedforward Neural Network (FFNN):

Rainfall

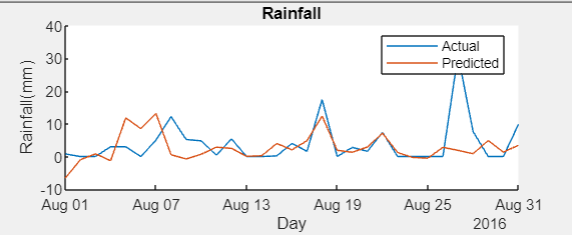


Figure 7 FFNN- Prediction and MSE of Rainfall

The graph represents actual and forecasted precipitation according to the Feed Forward Function Fitting Neural Network model. The graph represents 31 days. The graph demonstrates that the model is not entirely accurate; however, it follows the pattern of the actual data, which is encouraging.

Temperature

A graph of a temperature

Description automatically generated

A graph of a graph

Description automatically generated

Figure 8 FFNN- Prediction and MSE of Temperature

Actual and predicted temperatures are shown on the graph, with the value of the predicted temperature being predicted by the Feed Forward Function Fitting Neural Network model. The lines were plotted for 31 days. The graph demonstrates that the model follows the pattern of the actual data, which is positive. MSE value is 0.59474 for the Feed Forward Function Fitting Neural Network model with a hidden layer size of 15.

## Comparison of Models

A graph of different weather conditions

Description automatically generated

A graph of different weather conditions

Description automatically generated

Figure FFNN Model

Figure CFNN Model

The diagrams above for the output rainfall is shown of two distinct neural networks over a 31-day period. Both the cascade forward neural network and the function fitting neural network are moderately capable of predicting the pattern. It is demonstrated that the FFNN model predicts results that are relatively like the actual data.

The temperature variables above are compared the outputs of two distinct neural networks predicting the temperature and the actual temperature over a period of 31 days. The cascade forward neural network and Feed Forward Function Fitting neural network model has close accurate predictions with promising results and similar pattern recognition to the actual data.

# Fuzzy Logic

A fuzzy logic system is the standard of truth for a particular scenario. It is used to simulate the way the human brain makes decisions (GeeksforGeeks, 2019). For instance, a fuzzy logic system can determine whether the temperature is hot or chilly, by using fuzzy logic, instead of using 1 and 0 to identify the weather in a Boolean manner, we can output any value between 0 and 1, 0.45, and other numbers (GeeksforGeeks, 2019).

## Membership Function

Membership functions are fundamental to fuzzy logic, a mathematical framework for addressing uncertainty and imprecision in decision-making. A membership function determines an element's level of membership in an ambiguous set. It converts each input value to a membership value between 0 and 1, denoting the input's degree of membership in the fuzzy set.

**The triangular membership function** is one of the most frequently employed membership functions. It has a triangular shape with a defined centre and two slopes that determine the membership value's rate of increase and decrease. The triangular membership function determines the membership value of an input x based on its position relative to the three parameters. The function increases gradually from 0 at its lower limit (a), reaches its maximal value of 1 at its centre (b), and then decreases gradually to 0 at its upper limit (c). Both sides of the centre have symmetrical gradients.

Another type of membership function commonly employed in fuzzy logic systems is the **trapezoidal membership function**. It is characterised by a trapezoid-shaped curve in which the membership value is constant within a specific range and abruptly changes at the boundaries.

The **Gaussian membership function** is another frequently used membership function. It has a bell-shaped curve in which the membership value decreases gradually as the input value moves away from the curve's centre.

## Fuzzy Logic Model A

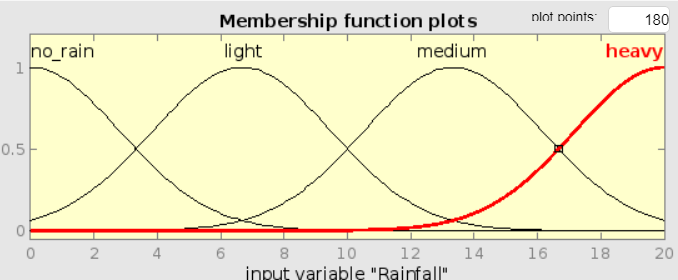
A screenshot of a computer

Description automatically generated

Figure Fuzzy Logic Model A

Matlab's fuzzy logic toolset is displayed in the figure shown above. The fuzzy logic arsenal consists of three primary components: input, rule base, and output. The above fuzzy logic model outlined in the previous section has two inputs. The rainfall value is the first input, followed by the temperature value.

The inputs are obtained from the neural network's outputs. Each of the input contents has its own membership function with unique characteristics to map the input content accordingly.



A diagram of a function

Description automatically generated

Figure 12 MF for intput Variable

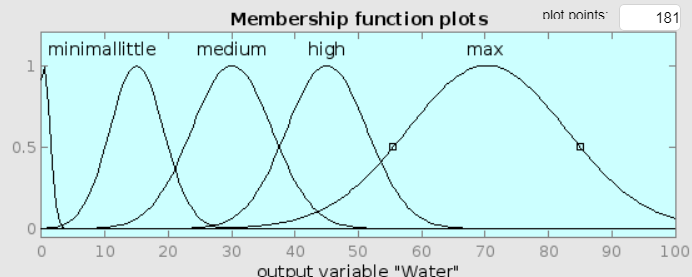


Figure MF for Output Variable

The diagram above shows the membership function used to generate output data from the fuzzy logic model. The output of the data will be the quantity of water that will be passed to the sprinklers. In millilitres, the quantity of water will be measured. The water discharge will vary between 0 and 100. The range begins at 0 because, during frigid weather, the model indicates that sprinklers should receive no water.

A screenshot of a text

Description automatically generated

Figure 14 Fuzzy A- The 14 Fuzzy Rules

The above diagram represents the rules utilised by the fuzzy logic inference engine. In total, fourteen principles were implemented for all potential outcomes. The principles were constructed utilising both "and" and "or" gates logic.

A screenshot of a graph

Description automatically generated

Figure 15 Fuzzy A- The Fuzzy Rules

There are a total of 14 rules that were passed into the fuzzy logic's inference engine. The values transmitted into the fuzzy logic model are derived from one of the developed neural networks. As illustrated in the diagram, three parameters will be transmitted to the model. The first parameter is the amount of rainfall, and the second is the temperature. The inputted temperature is 15.4 degrees Celsius, and the inputted rainfall is 5.5 millimetres. Based on the supplied inputs, the output water volume is 5.02 litres. This is merely an illustration; additional testing will be provided throughout the presentation.

A graph of a graph

Description automatically generated

Figure Fuzzy A Surface

This three-dimensional curve illustrates the relationship between average temperature and relative rainfall and final water output. This graph demonstrates that the temperature and rainfall have a direct correlation with the discharge water volume. Similarly, when a temperature and precipitation map is depicted on a three-dimensional map. The greater the rainfall, the lower the temperature, and the less water will be produced. In this three-dimensional graph, the two curve conditions are displayed in a manner that makes them readily apparent. But if the three-dimensional mapping concurrently appears on a three-dimensional map. His display results could be extremely poor.

## Fuzzy Logic Model B

A screenshot of a computer

Description automatically generated

Figure Fuzzy Logic Model A

Matlab's fuzzy logic toolset is displayed in the figure shown above. The fuzzy logic arsenal consists of three primary components: input, rule base, and output. The above fuzzy logic model outlined in the previous section has two inputs. The rainfall value is the first input, followed by the temperature value. The inputs are obtained from the neural network's outputs. Each of the input contents has its own membership function with unique characteristics to map the input content accordingly.

A graph of rain and rain

Description automatically generated

A diagram of a function

Description automatically generated

Figure 18 MF for input Variables

A diagram of a function

Description automatically generated

Figure MF for Output Variable

The diagram above shows the membership function used to generate output data from the fuzzy logic model. The output of the data will be the quantity of water that will be passed to the sprinklers. In millilitres, the quantity of water will be measured. The water discharge will vary between 0 and 100. The range begins at 0 because, during frigid weather, the model indicates that sprinklers should receive no water.

A screenshot of a text

Description automatically generated

Figure Fuzzy B- The 12 Fuzzy Rules

The above diagram represents the rules utilised by the fuzzy logic inference engine. In total, twelve principles were implemented for all potential outcomes. The principles were constructed utilising both "and" and "or" gates logic.

A screenshot of a graph

Description automatically generated

Figure Fuzzy B- The Fuzzy Rules

There are a total of 14 rules that were passed into the fuzzy logic's inference engine. The values transmitted into the fuzzy logic model are derived from one of the developed neural networks. As illustrated in the diagram, three parameters will be transmitted to the model. The first parameter is the amount of rainfall, and the second is the temperature. The inputted temperature is 15.4 degrees Celsius, and the inputted rainfall is 5.5mm. Based on the supplied inputs, the output water volume is 2.26 litres. This is merely an illustration; additional testing will be provided throughout the presentation.

A graph showing different colors of the same color

Description automatically generated

Figure Fuzzy A Surface

This three-dimensional curve illustrates the relationship between average temperature and relative rainfall and final water output. This graph demonstrates that the temperature and rainfall have a direct correlation with the discharge water volume. Similarly, when a temperature and precipitation map is depicted on a three-dimensional map. The greater the rainfall, the lower the temperature, and the less water will be produced. In this three-dimensional graph, the two curve conditions are displayed in a manner that makes them readily apparent. But if the three-dimensional mapping concurrently appears on a three-dimensional map. His display results could be extremely poor.

# Justification

The outputs of both fuzzy logic models are shown in the above diagram. The first fuzzy logic-A has a total of 14 principles. whereas the second fuzzy logic-B includes only 12 principles. Both fuzzy logic models will accept the same input. The first input is rainfall, followed by temperature. These data were provided by the developed neural network. The outputs will be the same, with the quantity of water passing through the sprinkler depending on the weather.

Both models of fuzzy logic have unique rules. When the temperature is 15.4 degrees Celsius, and the rainfall is 5.5mm, the amount of water sent to watering for the trees will be 5.08 litres. With the same values provided to the second fuzzy logic model, the sprinklers will be supplied with  2.26 litres of water.

In conclusion, the second fuzzy logic model is preferable to the first fuzzy logic model for implementation with a neural network system since less water is provided at low temperatures whereas the first fuzzy logic model provides a large quantity of water during cold seasons.

# References