

[Get started](#)[Open in app](#)[Follow](#)

582K Followers



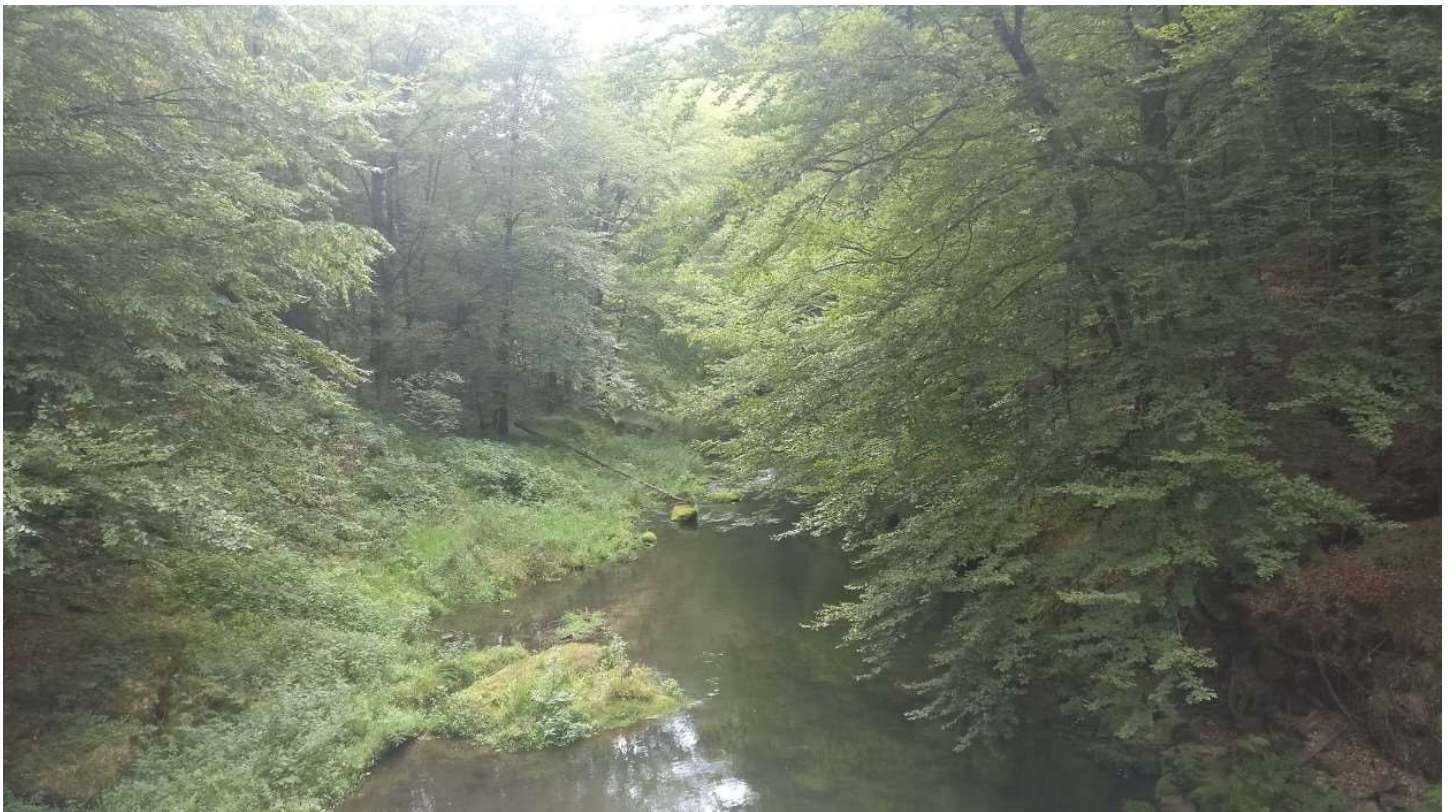
You have **1** free member-only story left this month. [Sign up for Medium and get an extra one](#)

# A very brief introduction to Fuzzy Logic and Fuzzy Systems

“As complexity rises, precise statements lose meaning and meaningful statements lose precision” — Lotfi A. Zadeh



Carmel Gafa · Apr 26, 2020 · 10 min read ★



## Introduction

Get started

Open in app



lifting fragile objects or parking a car. The ability of humans to deal with vague and imprecise data makes such tasks easy for us. Therefore if we aim to replicate the control actions of a human operator, we must be able to model the activities of the operator and not of the plant itself. Our model must be built so that it is capable of dealing with vague information.

Fuzzy logic-based systems do precisely that; they excel where systems are particularly complex and have been used successfully in many applications ranging from voice and handwriting recognition to subway train speed control.

This article focuses on the basic ideas of fuzzy sets and systems.

## Crisp Sets and logic

Classical logic is based on the crisp set, where a group of distinct objects are considered as a collection. For example, the colours white and red are both separate objects in their own right, but they can be regarded as a collection using the notation {red, white}. Crisp sets are, by convention designated a capital letter hence the above example can be described by,

$$F = \{\text{red, white}\}$$

A crisp subset can be defined from a more extensive set where the elements of the set belong to the subset according to some condition. For example, set A can be defined as the set of numbers that are greater or equal to 4 and smaller or equal to 12. This statement can be described using the following notation:

$$A = \{i \mid i \text{ is an integer and } 4 \leq i \leq 12\}$$

A graphical representation of the subset above is possible if we introduce the notion of the characteristic or indicator function of a set, that is, in this case, the function defined over the set of integers, that we shall call X, that indicates the membership of elements in subset A in X. This is achieved by assigning a value of 1 to the elements of X in A, and a value of 0 to the elements of X not in A. In our example, therefore, the indicator function for this set is:

$$f(x) = \begin{cases} 1 & \text{if } 4 \leq x \leq 12 \\ 0 & \text{otherwise} \end{cases}$$

Get started

Open in app



Graphically this can be displayed as follows:

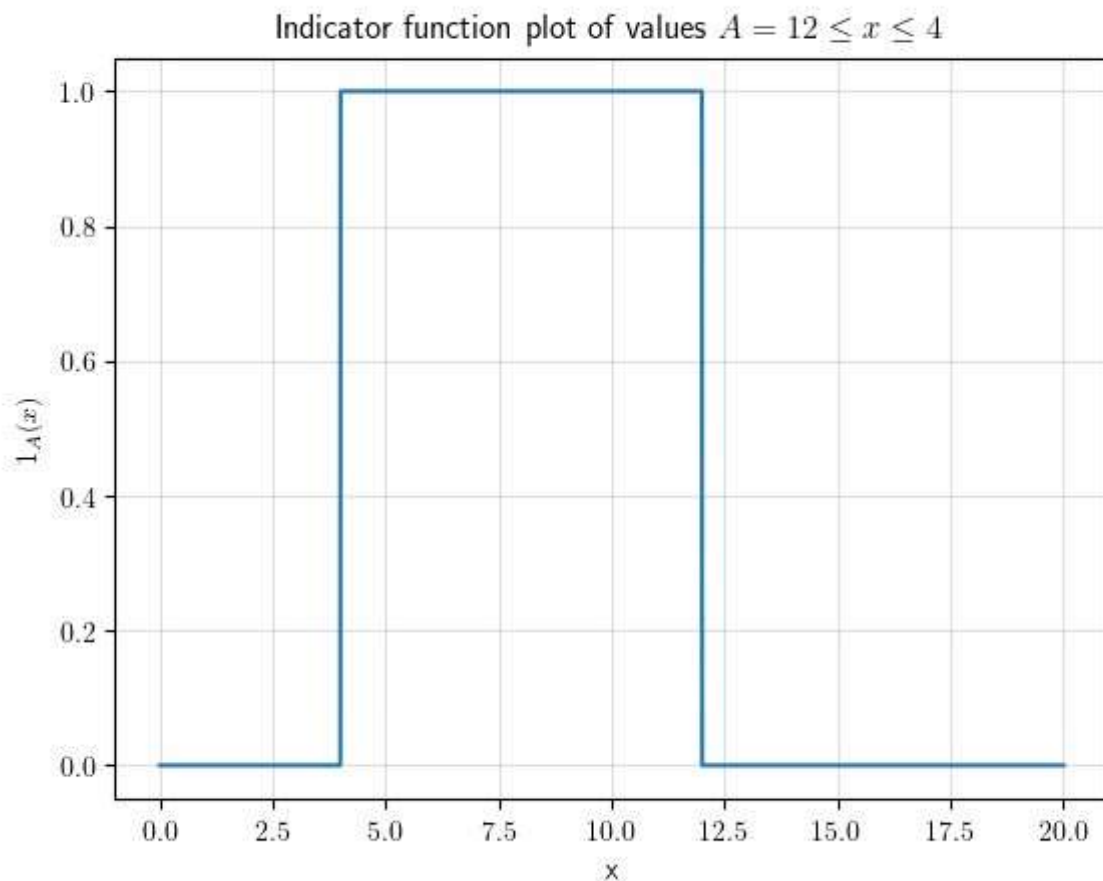
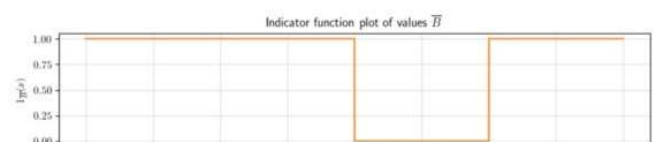
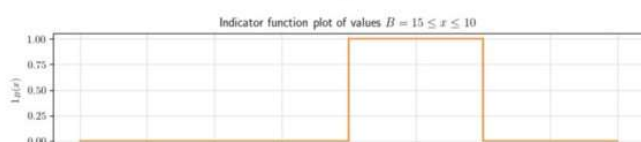
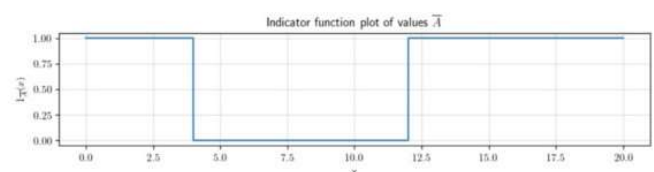
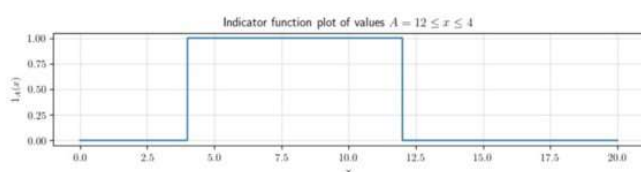


Figure 1.

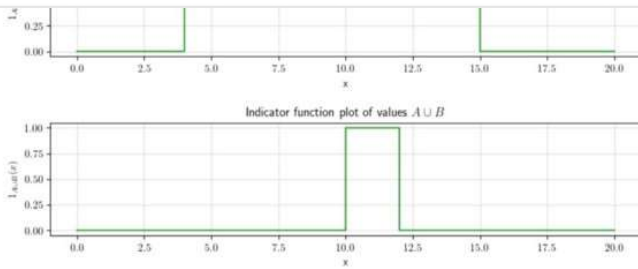
The intersection of two sets is the set containing all elements of that are common to both sets. The union of two sets is the set containing all elements that are in either of the sets.

The negation of a set  $A$  is the set containing all elements that are not in  $A$ .



Get started

Open in app



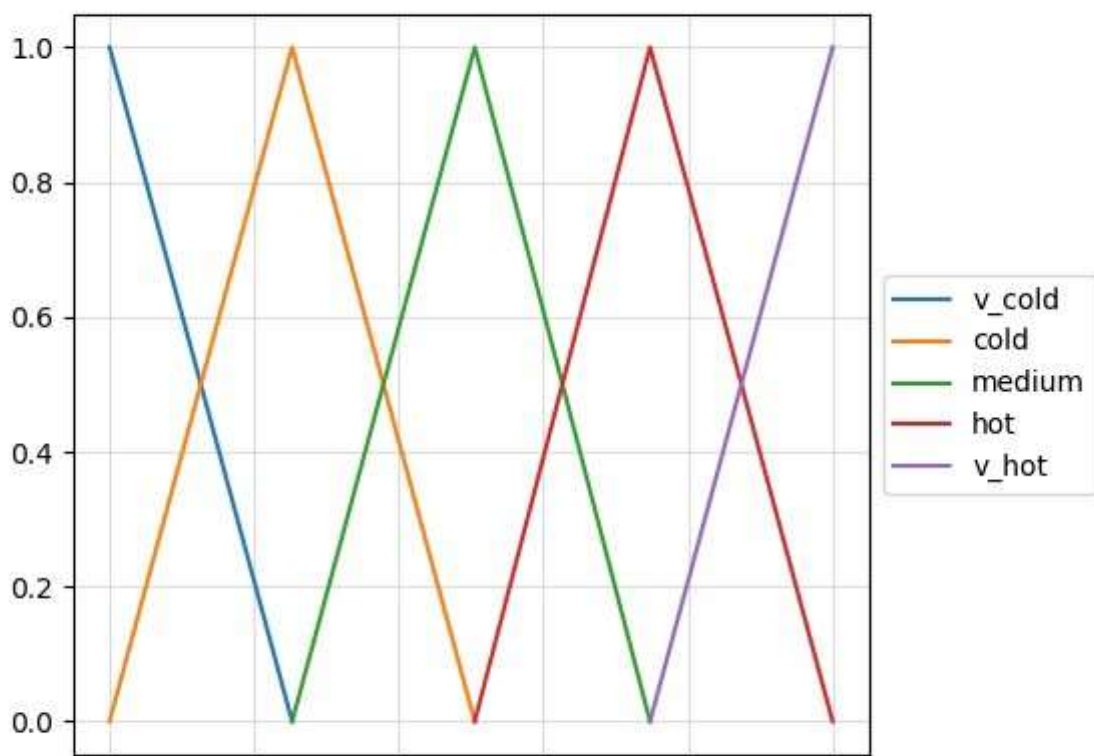
## Fuzzy sets

Fuzzy sets were introduced by Lotfi Zadeh (1921–2017) in 1965.

Unlike crisp sets, a fuzzy set allows partial belonging to a set, that is defined by a degree of membership, denoted by  $\mu$ , that can take any value from 0 (element does not belong at all in the set) to 1 (element belongs fully to the set).

It is evident that if we remove all the values of belonging except from 0 and 1, the fuzzy set will collapse to a crisp set that was described in the previous section.

The membership function of the set is the relationship between the elements of the set and their degree-of-belonging. An illustration of how membership functions can be applied to temperature is shown below.





Get started

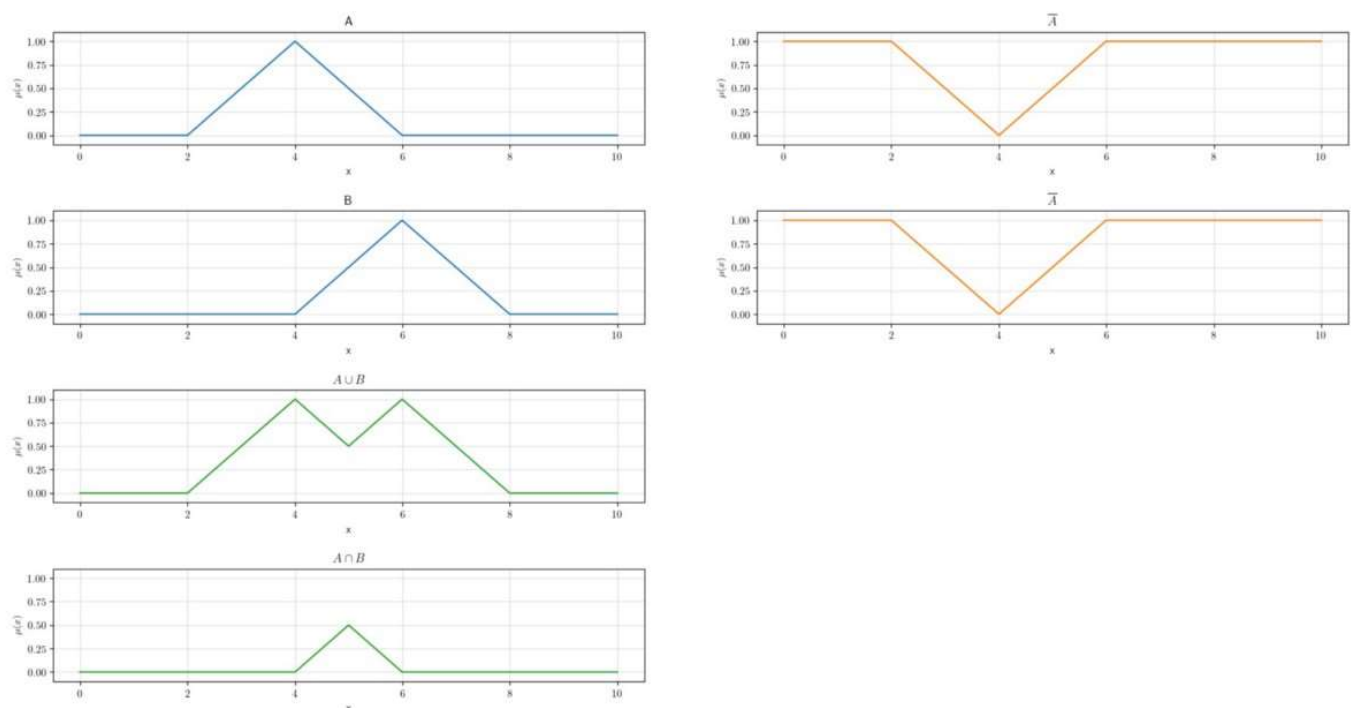
Open in app



In the example above, the fuzzy sets describe temperatures of an engine ranging from very cold to very hot. The value,  $\mu$ , is the amount of membership in the set. One can notice, for example, that at a temperature of 80 degrees, the engine can be described as being **hot** to a factor of 0.2, and **very hot** to a factor of 0.8.

In the previous section, the union, intersection and negation operators of crisp sets were discussed as they provide a way to express conjunction and disjunction (and/or) that are pivotal to reasoning.

The most common method of computing the union of two fuzzy sets is by applying the maximum operator on the sets. Other methods do exist, including the use of the product operator on the two sets. Similarly, the most common method of computing the intersection of two fuzzy sets is by applying the minimum operator on the sets. The complement of a fuzzy set is calculated by subtracting the set membership function from 1.



One crucial observation is that an element can have a degree of belonging both in a set and in the complement of the set. Hence, as an example, element  $x$  can be both in  $A$  and also in 'not- $A$ '.

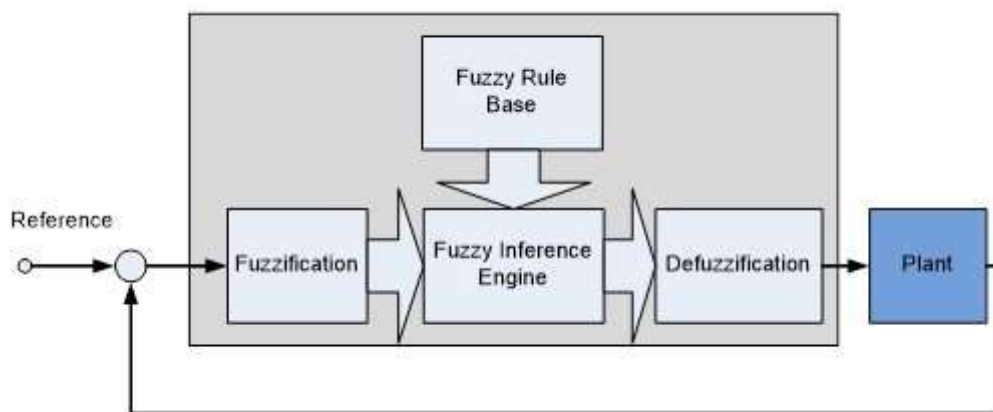
## Fuzzy Inference Systems

[Get started](#)[Open in app](#)

fuzzy membership functions and a set of fuzzy rules, known as the rule-base, having the form:

**IF** (conditions are fulfilled) **THEN** (consequences are inferred)

The basic configuration of a fuzzy system is shown below:



A typical fuzzy system can be split into four main parts, namely a fuzzifier, a knowledge base, an inference engine and a defuzzifier;

The **fuzzifier** maps a real crisp input to a fuzzy function, therefore determining the ‘degree of membership’ of the input to a vague concept. In a number of controllers, the values of the input variables are mapped to the range of values of the corresponding universe of discourse. The range and resolution of input-fuzzy sets and their effect on the fuzzification process are considered as factors affecting the overall performance of the controller.

The **knowledge base** comprises the knowledge of the application domain and the attendant control goals. It can be split into a database of **definitions** used to express linguistic control rules in the controller, and a **rule base** that describes the knowledge held by the experts of the domain. Intuitively, the knowledge base is the core element of a fuzzy controller as it will contain all the information necessary to accomplish its execution tasks. Various researchers have applied techniques to fine-tune a fuzzy

Get started

Open in app



The **Inference Engine** provides the decision making logic of the controller. It deduces the fuzzy control actions by employing fuzzy implication and fuzzy rules of inference. In many aspects, it can be viewed as an emulation of human decision making.

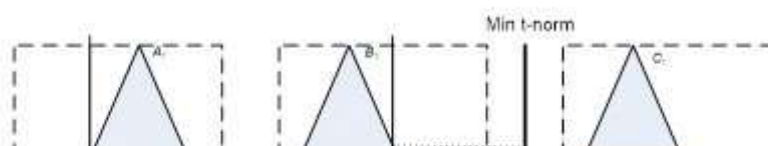
The **Defuzzification** process converts fuzzy control values into crisp quantities, that is, it links a single point to a fuzzy set, given that the point belongs to the support of the fuzzy set. There are many defuzzification techniques, the most famous being the centre-of-area or centre-of-gravity.

$$\bar{x} = \frac{\sum_i x_i \mu(x_i)}{\sum_i \mu(x_i)}$$

Other defuzzification methods include first of maxima and mean of maxima.

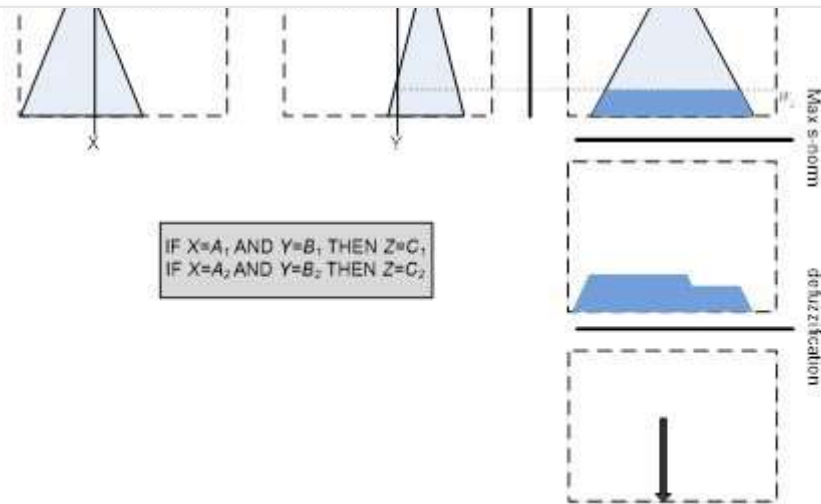
Several inferencing models use fuzzy sets to reason the outputs of a system given inputs. One of the most popular methods was devised by Professor Abe Mamdani that used fuzzy sets to control a steam engine. Another popular model was developed by Professor Tomohiro Takagi and Professor Michio Sugeno.

In Mamdani inferencing, the antecedents and consequents of a fuzzy rule are fuzzy sets. The inference is based on Generalised Modus Ponens, which states that the degree of truth of the consequent of a fuzzy rule is the degree of truth of the antecedent. In the case where more than one antecedent clause is present, the individual degrees of membership are joined using a min t-norm operator. If the fuzzy system contains several rules, their output is combined using a max s-norm operator. Defuzzification is necessary so that the consequent action can be expressed in terms of a crisp value. A graphical representation of this process is shown below.



Get started

Open in app



In the Takagi-Sugeno inferencing model, the consequents are functions that map crisp input values to the rule's crisp output. Hence fuzzy rules are of the form:

**IF**  $x$  **IS**  $X$  **and**  $y$  **IS**  $Y$  **THEN**  $z=f(x, y)$

where  $f$  is generally a linear function in  $X$  and  $Y$ . In contrast to Mamdani fuzzy systems, the rules are not combined using a max -operator but are combined by finding a weighted average, where the weight of a given rule is the degree of membership of its antecedent. Therefore Takagi-Sugeno systems do not require any defuzzification.

## Design of a Fuzzy System

In this section, a simple example system will be constructed and executed to visualise the design and execution of a fuzzy inference system. The hypothetical system considered here controls the speed of a fan has according to the environment's temperature and humidity. Therefore, our system consists of two inputs, temperature and humidity and a single output, that is the fan speed.



The first step in the design of our system is to define fuzzy sets to describe the input and output variables. For simplicity's sake each variable will be characterised by three



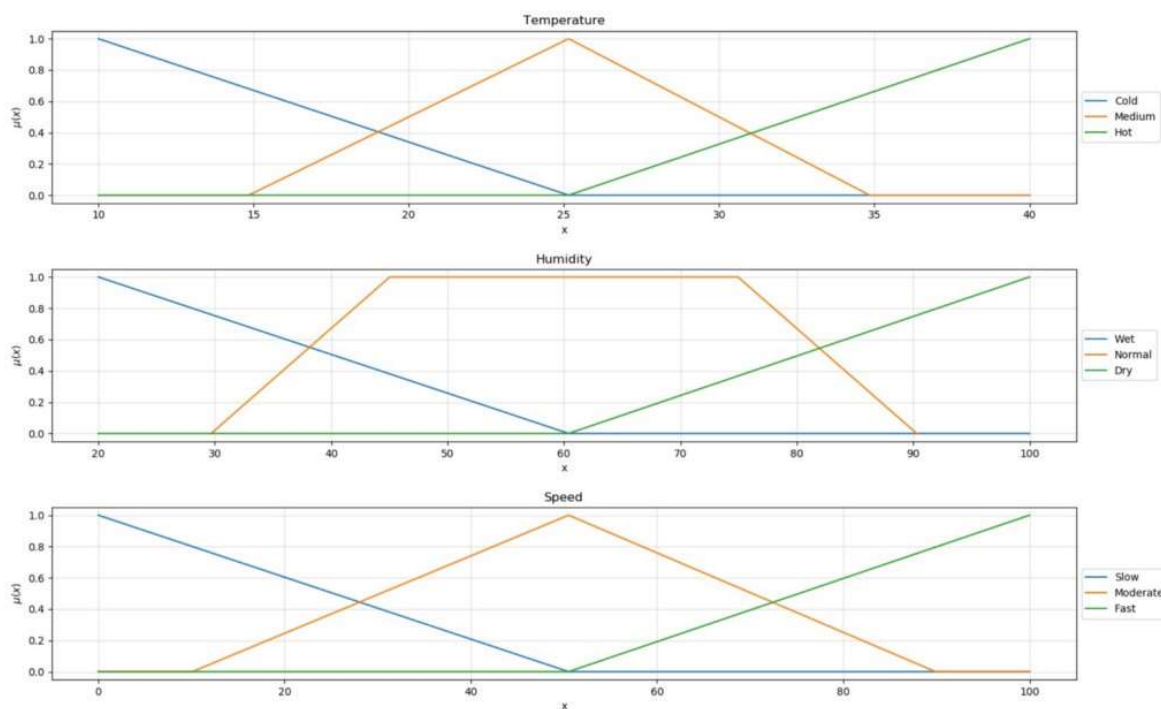
Get started

Open in app



Temperature: Cold, Medium, Hot  
 Humidity: Dry, Normal, Wet  
 Fan Speed: Slow, Moderate, Fast

The diagram below shows a graphical representation of the input and output variables of our system and their respective sets.



It can be noted that triangular sets were used to describe most of the sets of this system; however, 'normal' humidity is specified using a trapezoidal set. Fuzzy sets reflect the knowledge is the user designing the system, so they can take a wide variety of shapes.

Note that the output was described using fuzzy sets as well; therefore the system that is being considered is a Mamdani-type, that links fuzzy sets related to the inputs of the system to fuzzy sets associated with the output of the system using fuzzy rules.

A total of nine rules are used to describe the knowledge necessary to operate our fan:

If Temperature is Cold and Humidity is Dry Then Fan Speed is Slow  
 If Temperature is Medium and Humidity is Dry Then Fan Speed is Slow  
 If Temperature is Cold and Humidity is Dry Then Fan Speed is Slow

Get started

Open in app



If Temperature is Cold and Humidity is Wet then Fan Speed is Moderate

If Temperature is Hot and Humidity is Normal Then Fan Speed is Fast

If Temperature is Hot and Humidity is Wet Then Fan Speed is Fast

If Temperature is Medium and Humidity is Wet Then Fan Speed is Fast

These rules can be visualised if we use a combined fuzzy rule base, that is a grid where the input-fuzzy sets occupy the edges so that each cell in the grid defines a rule. The following diagram shows the rule base for this system.

Humidity	Dry	Moderate	Fast	Fast
	Normal	Slow	Moderate	Fast
	Wet	Slow	Slow	Moderate
		Cold	Medium	Hot
		Temperature		

The following steps take place when an input combination is fed to the system, let us as an example say that we have a temperature of 18 degrees and humidity of 60%:

- The degree of membership for each set of the input variables is determined. Hence we can say that a temperature of 18 degrees is

```
0.48 Cold
0.29 Medium
0.00 Hot
```

and humidity of 60% is

Get started

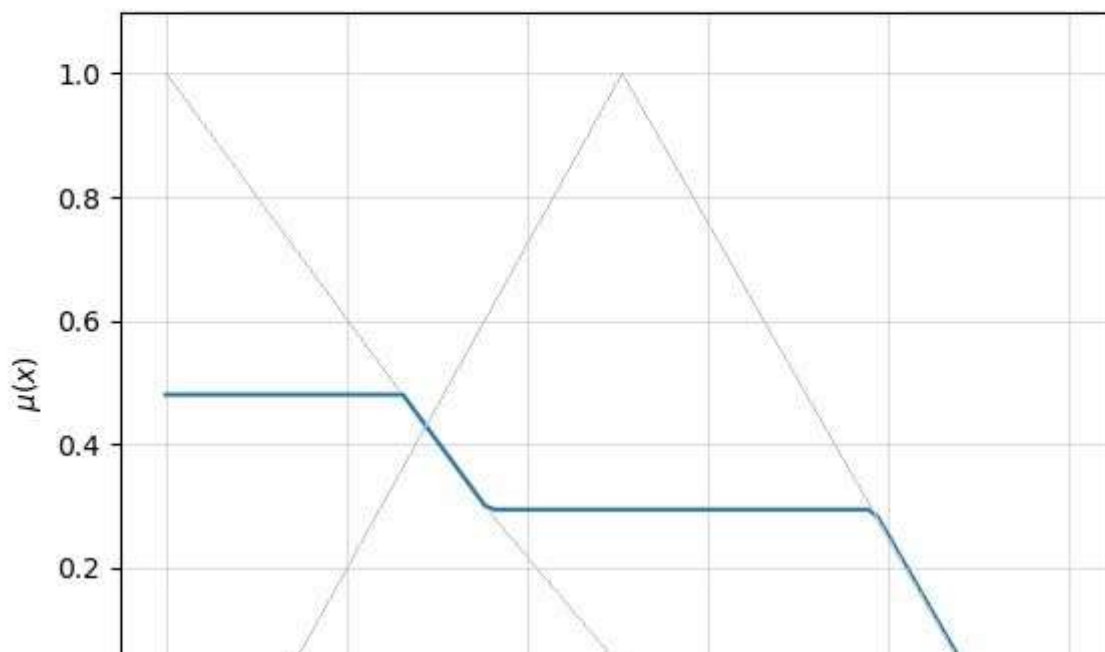
Open in app



- With this input combination, two rules are fired with a degree higher than zero, as can be seen in the updated fuzzy rule base below:

Humidity	Dry(0.0)	Moderate	Fast	Fast
	Normal(1.0)	Slow	Moderate	Fast
	Wet(0.0)	Slow	Slow	Moderate
		Cold (0.48)	Medium(0.29)	Hot(0.0)
		Temperature		

And therefore our fuzzy output will consist of the speed Slow, that has activation of 0.48 and Moderate that has 0.29 activation. The combined effect of the two rules or the fuzzy output of the system is displayed below:



[Get started](#)[Open in app](#)

The output set is finally defuzzified using centre-of-gravity defuzzification and a crisp value of 36.814 is obtained to drive the fan.

## Conclusion

In this article, a brief introduction to fuzzy sets and fuzzy inferencing was presented. It is shown how control of systems can be achieved using linguistic terms to represent human knowledge. In the next article, a fuzzy inference system will be constructed using python from scratch.

Thanks to Yenson Lau.

---

## Sign up for The Variable

By Towards Data Science

Every Thursday, the Variable delivers the very best of Towards Data Science: from hands-on tutorials and cutting-edge research to original features you don't want to miss. [Take a look.](#)

Your email

---

Get this newsletter

By signing up, you will create a Medium account if you don't already have one. Review our [Privacy Policy](#) for more information about our privacy practices.

Fuzzy Logic   Fuzzy

[About](#) [Help](#) [Legal](#)

Get the Medium app



Get started

Open in app

