Chapter 7

Fuzzy logic theory

Dr. Gonzalo Nápoles
Tilburg University

In this chapter, I will introduce the *Fuzzy Logic Theory*. Fuzzy logic is a form of many-valued logic that will help us convert numerical values into symbolic ones. After motivating this theory, I will introduce the main concepts and the ensuing mathematical formalism. Moreover, I will elaborate on a case study that illustrates the advantages of reasoning with fuzzy IF-THEN rules.

Introduction

In the previous lectures, we studied how to reason with symbols using Prolog clauses and a knowledge base. It consists of atomic and conditional sentences describing a given problem with the aid of symbols. We could certainly include numbers in a Prolog knowledge base, but then we can argue whether Prolog is the proper tool to perform the reasoning. However, most real-world problems involve a mixture of numerical and nominal variables.

To exploit the advantages of Prolog, we should be able to transform numerical knowledge representations into symbolic ones.

There is another issue directly linked to numerical representations: the detail level. There are problems in which more details about the problem domain leads to more information but less knowledge. Theoretically speaking, this is known as *granularity degree*. More granularity implies more detailed knowledge representations and less abstraction in those representations. For example, real numbers have a higher granularity than integer numbers as they allow more precision when quantifying a certain phenomenon.

Fuzzy logic approximates human reasoning and does a good job of balancing the trade-off between precision and significance. For instance, when warning someone of an object falling toward them, being precise about the exact mass and speed is not necessary. As a matter of fact, we might not be able to process these details after getting hit by such a massive structure. Therefore, in the fuzzy logic context, *precision* refers to granularity level of the knowledge while *significance* refers to the usability of that knowledge.

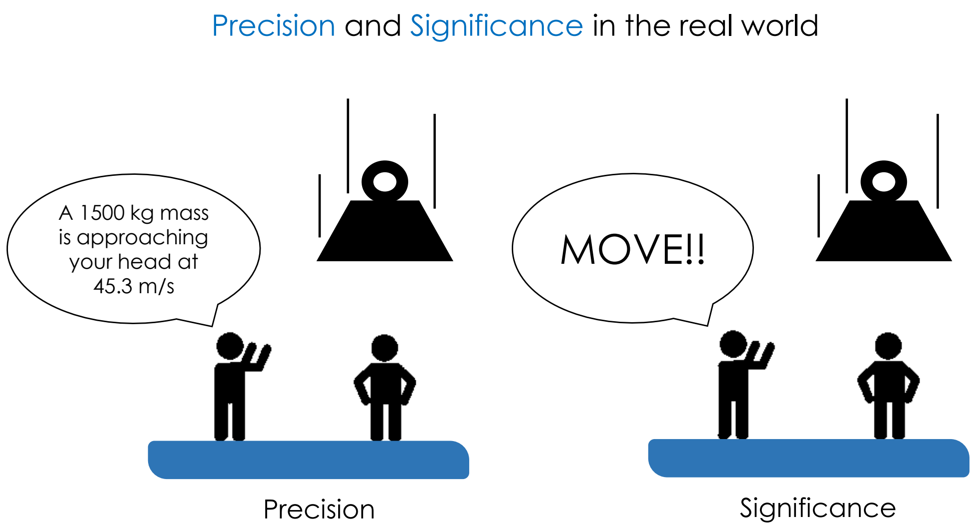


Figure 3: Example of precision versus significance.

However, there are other scenarios where we receive rather qualitative information to compute a numerical outcome. An example of this situation could be determining the appropriate amount to tip your waitperson in a US restaurant based on a symbolic service evaluation.

Fuzzy logic

Fuzzy logic refers to a form of many-valued logic in which the truth values of variables may be any real number between 0 and 1, both inclusive. Such a mathematical theory is quite useful to handle the concept of partial truth, where the truth value may range between completely true and completely false. By contrast, in Boolean logic (also known as Aristotelian logic), the truth values of variables may only be the integer values zero or one.

Fuzzy logic assumes that people make decisions using imprecise and symbolic information. Fuzzy models or sets are mathematical means of representing vagueness and imprecise information (hence the term fuzzy). These models have the capability of recognizing, representing, manipulating, interpreting, and using data and information that are vague and lack certainty.

Fuzzy logic principles were firstly presented by Prof. Lotfi Zadeh [1] in his 1965 seminal paper entitled "*Fuzzy sets*". As a part of his research on computer understanding of natural language, Prof. Zadeh realized that such problems (like many other real-world situations) are difficult to model using bivalent logic. In the beginning, his theory was not warmly received mainly due to the negative connotation of the word "fuzzy" as people tend to associate this word with uncertainty and lack of precision. However, fuzzy logic is about the mathematical quantification of situations that lack precision.

Overall, classical logic only permits conclusions which are either true or false. However, there are also propositions with variable answers, such as one might find when asking a group of people to identify a color. In such instances, the truth appears as the result of reasoning from inexact or partial knowledge in which the sampled answers are mapped on a spectrum.

Fuzzy logic starts with the concept of a fuzzy set. Informally speaking, a fuzzy set is a set without a crisp, clearly defined boundary. It can contain elements with only a partial degree of membership.

To understand what a fuzzy set is, let us analyze the definition of a classical set (often referred to as *crisp* set): *a set that wholly includes or wholly excludes any given element.* For example, the set of days of the week unquestionably includes Monday, Tuesday, Wednesday, Thursday, Friday, Saturday and Sunday. Such a crisp will unquestionably exclude everything else.

[1] Lotfi Aliasker Zadeh (1921-2017) was an Emeritus Professor of Computer Science at the University of California, Berkeley. Prof. Zadeh was best known for proposing fuzzy mathematics consisting of these fuzzy-related concepts: fuzzy sets, fuzzy logic, fuzzy algorithms, fuzzy semantics, fuzzy languages, fuzzy control, fuzzy systems, fuzzy probabilities, fuzzy events, and fuzzy information. Prof. Zadeh is also credited, along with John R. Ragazzini, in 1952, with having pioneered the development of the z-transform method in discrete time signal processing and analysis. These methods are now standard in digital signal processing, digital control, and other discrete-time systems used in industry and research.

This type of set is called a classical set because it has been around for a long time. It was Aristotle who first formulated the Law of the Excluded Middle, which says X must either be in set A or in set not-A. Another version of this law is: *of any subject, one thing must be either asserted or denied.*

To restate this law with annotations: "Of any subject (say Monday), one thing (a day of the week) must be either asserted or denied (I assert that Monday is a day of the week)." This law demands that opposites, the two categories A and not-A, should between them contain the entire universe (the set with all possibilities). Everything falls into either one group or the other. There is no thing that is both a day of the week and not a day of the week.

Now, let us consider the set of days comprising a weekend. Most people will agree that Saturday and Sunday belong in the weekend set, but what about Friday? It feels like a part of the weekend, but it should be technically excluded. Therefore, Friday "straddles the fence." Classical sets do not tolerate this kind of classification. Either something is in a set or it is out of a set. However, human experience suggests something different.

Of course, individual perceptions and cultural background must be considered when you define what constitutes the weekend. Even the dictionary is imprecise, defining the weekend as the period from Friday night or Saturday to Monday morning. You are entering the realm where sharp-edged, yes-no logic stops being helpful. Fuzzy reasoning becomes valuable exactly when you work with how people perceive the concept weekend instead of a simple-minded classification useful for accounting purposes only.

The following statement lays the foundations for fuzzy logic: *In fuzzy logic, the truth of any statement becomes a matter of degree.*

However, in his book entitled "*Razonamiento: significado, incertidumbre y borrosidad*"[2], Prof. Enric Trillas makes an important amendment to this definition: fuzzy logic is a matter of degree and *meaning*.

The major advantage that fuzzy reasoning offers is the ability to reply to a yes-no question with a not-quite-yes-or-no answer. Humans do this kind of thing all the time (think how rarely you get a straight answer to a seemingly simple question), but it is a rather new trick for computers.

[2]This book was written in Spanish. The translation of this tittle to English would be "Reasoning: meaning, uncertainty and fuzziness"

Fuzzy set formalism

Now, let us give mathematical shape to the above-discussed ideas. Firstly, it is important to mention that fuzzy sets operate with *linguistic variables* (also referred to as *symbolic variables*). While variables in mathematics usually take numerical values, in fuzzy logic applications, non-numeric values are often used to facilitate the expression of rules and facts. A linguistic variable such as age may accept values such as young and its antonym old. The values for linguistic variables are termed either *linguistic term* or *symbolic value*. For example, *blonde* is a possible linguistic term for the symbolic variable *hair color*.

Because natural languages do not always contain enough terms to express a fuzzy value scale, it is common practice to modify linguistic values with adjectives or adverbs. For example, we can use *rather* and *somewhat* to construct the additional values *rather old* or *somewhat young*.

A fuzzy set is an extension of a classical set. If X is the universe of discourse and its elements are denoted by x, then a fuzzy set A in X is defined as a set of ordered pairs $A = \{x, \mu_A(x) \mid x \in X\}$. In this definition, $\mu_A(x)$ represents the *membership function* (to be discussed in the following section).

Fuzzification operations map mathematical input values into fuzzy membership functions. And the opposite *de-fuzzifying* operations can be used to map a fuzzy output membership function into a crisp output value that can be then used for decision-making or control purposes.

Fuzzification refers to the process of assigning the numerical input of a system to fuzzy sets with some degree of membership. This is done by evaluating the element to be fuzzified with the membership function. This degree of membership may be anywhere within the $[0, 1]$ interval. If the membership value is 0, then the element does not belong to the given fuzzy set. If the membership value is 1, then the element completely belongs to the fuzzy set. Any value between 0 and 1 represents the degree of uncertainty that the value belongs to the set. These fuzzy sets are typically described by words or linguistic terms. Therefore, we can reason with it in a linguistically natural manner by assigning the system input to fuzzy sets. Moreover, we can use this principle to translate numbers into symbolic representations with little effort.

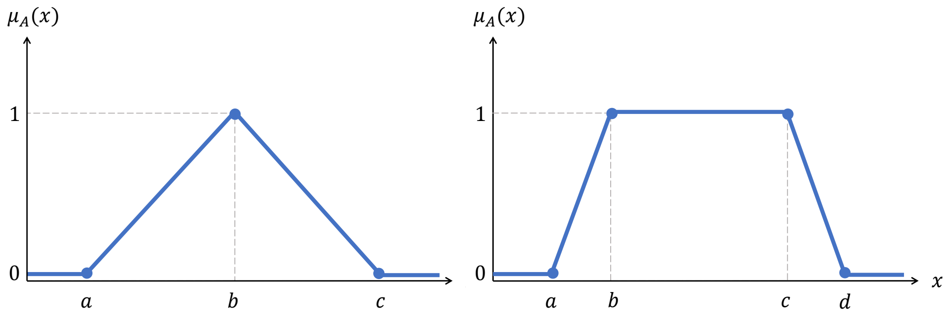
De-fuzzification refers to a set of methods to perform the inverse process. For example, given a fuzzy set A and a membership value $\mu_A(x)$, we can find the crisp value x corresponding to that relationship.

Membership functions

It seems apparent that membership functions are a pivotal component of fuzzy systems. This can be defined as follows:

A *membership function* $\mu_A(x)$ is a function that defines how each point x in the input space X is mapped to a membership value (or degree of membership) between 0 and 1. The input space is sometimes referred to as the *universe of discourse*, a fancy name for a simple concept. Moreover, A corresponds to a certain linguist term. This suggests that we will have a membership function for each term describing a linguistic variable.

Membership functions are normally defined by the human modeler taking into account the granularity level (i.e., the number of symbolic terms) and the domain of variables. However, the literature includes generic transfer functions such as the triangular and the trapezoidal ones.



(a) triangular

(b) trapezoidal

Figure 4: Simple membership functions.

These functions are considered simple membership functions since they are formed using straight lines. The simplest one is the triangular membership function. This function is completely defined by three points (a, b and c) forming a triangle. In contrast, the trapezoidal membership function has a flat top resulting from truncating a triangle. It involves four points to be defined by the modeler (a, b, c and d). These straight-line membership functions have the advantage of simplicity and transparency. It should be noted that, unlike the triangular membership function, the trapezoidal function has a flat top where several elements can have maximal membership to the set.

Equations (7.1) and (7.2) mathematically define the triangular and trapezoidal membership functions, respectively.

$$\mu_A(x) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a < x \leq b \\ \frac{c-x}{c-b} & b < x < c \\ 0 & x \geq c \end{cases} \tag{7.1}$$

$$\mu_A(x) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a < x \leq b \\ 1 & b < x \leq c \\ \frac{d-x}{d-c} & c < x < d \\ 0 & x \geq d \end{cases} \tag{7.2}$$

Although these membership functions are widely used, we can build smoother functions based on the Gaussian function. The following equation shows the mathematical formalization of this function,

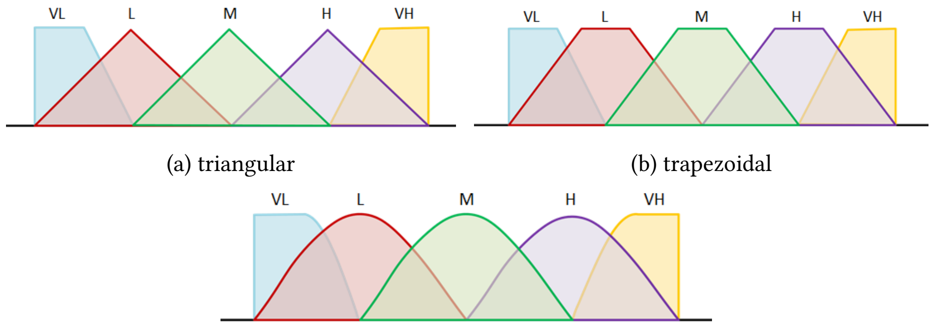
$$\mu_A(x) = e^{-\frac{(x-m)^2}{2k^2}} \tag{7.3}$$

where m is the central value (media) and $k > 0$ is the standard deviation (amplitude). This function allows for smoother transitions between intervals when compared to a triangular or trapezoidal membership function.

Example. Let us suppose that we have a numerical variable and we want to describe such a variable using the following linguistic terms: *very low* (VL), *low* (L), *medium* (M), *high* (H) and *very high* (VH).

Firstly, it would be convenient to normalize the variables such that the universe of discourse will be contained in the $[0, 1]$ interval. Secondly, we should build a membership function for each fuzzy set (linguistic term). For example, we can build a triangular membership function for the fuzzy set *medium* such that $a = 0.25$, $b = 0.5$ and $c = 0.75$. Similarly we could build a Gaussian membership function for that fuzzy set such that $m = 0.5$ and $s = 0.1$.

An important rule when designing these functions is that consecutive membership functions (e.g., the functions associated to *low* and *medium*) should overlap. Otherwise, the sets would be crisp instead of fuzzy. The extent to which these functions will overlap depends on the problem being modeled. Finally, when determining the most adequate linguistic term for a given (normalized) numerical value, we often select the term reporting the largest membership value. This is know as the *maximal membership principle*.



(a) triangular (b) trapezoidal

(c) Gaussian

Figure 5: Fuzzy sets associated with five linguistic terms.

A closer inspection of the figure shows that the membership functions associated with *very low* and *very high* have a different shape. This denotes that numbers lower than zero will still be very low and that numbers higher than one will still be very high. So, it is a matter of design.

From numbers to symbols

Although the previous example illustrated how to transform numbers into symbols, it seems convenient to further discuss the advantages of using fuzzy logic. I will do that with the aid of an example.

One of the most commonly used examples of a fuzzy set is the set of tall people. In this case, the universe of discourse is all potential heights, say from 54.64 cm to 2.72 m, according to the Guinness World Records (accessed in December 2020). In this example, the linguistic term *tall* will have a membership function defining the degree to which any person is tall.

If the set of tall people is given the well-defined (crisp) boundary of a classical set, you might say all people taller than 180 cm are considered tall. That distinction, although technically correct, is rather absurd. It is unreasonable to call one person short and another one tall when they differ in height by the width of a hair. Figure 6 illustrates the drama.

Then what is the right way to define the set of tall people? Using fuzzy logic. Figure 7 depicts a smoothly varying membership curve (the sigmoid function, defined as $\mu_A(x) = \frac{1}{1+e^{-x}}$) that passes from not-tall to tall. The output-axis is a number known as the membership value between 0 and 1. As discussed, this is achieved with the aid of a membership function.

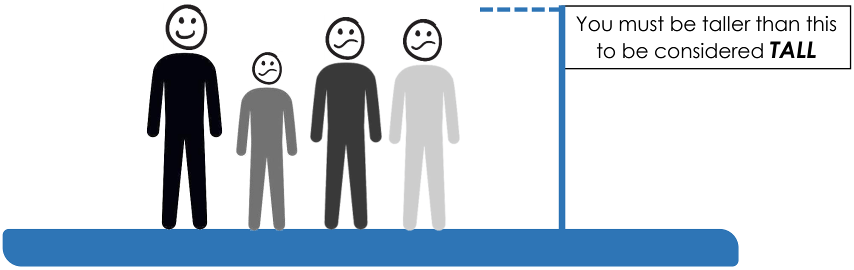


Figure 6: Example of the problem of using bivalent logic.

Figure 7 shows both crisp and smooth tall membership functions. In the top plot, the two people are classified as either entirely tall or entirely not-tall. In the bottom plot, the smooth transition allows for different degrees of tallness. Both people are tall to some degree, but one is less tall than the other. The taller person, with a tallness membership of 0.95 is definitely a tall person, but the person with a tallness membership of 0.3 is not very tall.

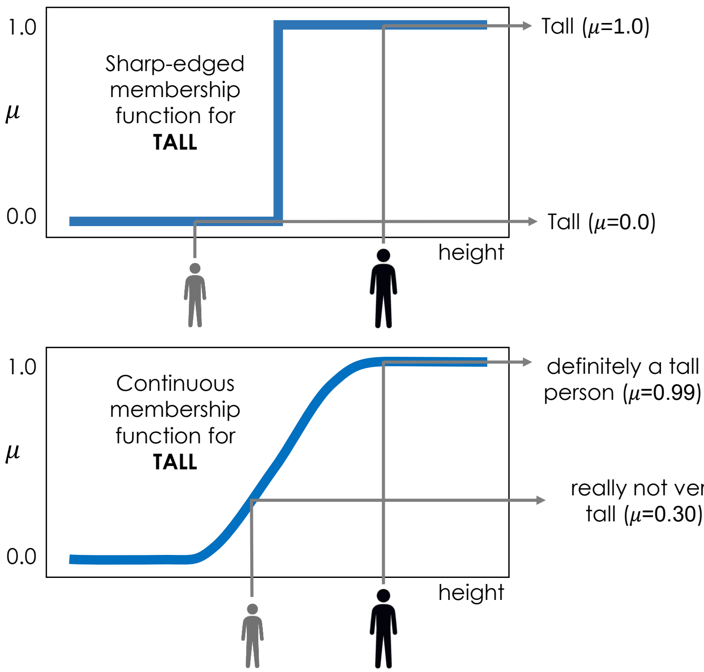


Figure 7: Crisp versus fuzzy sets when modeling the height attribute.

Defuzzification

It is already clear that we can transform a crisp numerical value into a fuzzy one using membership functions. But how could we perform the inverse process? This section will discuss how to obtain the crisp numerical quantity from one or several fuzzy sets and confidence values.

Defuzzification refers to the process of obtaining a crisp quantity given one or several fuzzy sets and confidence values.

Defuzzification is similar to computing the inverse of a continuous function. However, there exist several defuzzification operators that might lead to different results. This means that the defuzzified value is not unique and that selecting the proper method depends on the problem domain.

Let us suppose that we are provided with three fuzzy sets L (low), M (medium) and H (high) that describe a numerical variable. These fuzzy sets are characterized by membership functions $\mu_L(x)$, $\mu_M(x)$ and $\mu_H(x)$, respectively. We would like to know the crisp value having certainty y.

Firstly, we should aggregate the membership functions $\mu_L(x)$, $\mu_M(x)$ and $\mu_H(x)$ using the given certainty value y. In this context, aggregation is the process by which the fuzzy sets are combined into a single fuzzy set. The input of the aggregation process is the list of truncated membership functions according to the certainty value. The output of the aggregation process is a combined membership function that describes the aggregated fuzzy set.

Secondly, we need to apply the defuzzification method to the aggregated membership function. The literature includes several defuzzification methods such as the centroid, the middle of maximum (the average of the maximum membership values), the largest of maximum, and the smallest of maximum. Perhaps the most popular defuzzification method is the centroid operator, which returns the center of the area under the aggregate fuzzy set.

Equation (7.4) shows the centroid defuzzification method, which returns the center of gravity of the fuzzy set along the x-axis. In this equation, $\mu(x_i)$ is the membership value for point x_i in the universe of discourse. If you think of the area as a plate with uniform thickness and density, the centroid is the point along the x-axis about which the fuzzy set would balance.

$$x = \frac{\sum_i \mu(x_i)x_i}{\sum_i \mu(x_i)} \tag{7.4}$$

Notice that the centroid defuzzification method is nothing but a weighted average. In this calculation, the weights are the membership values to which the crisp values belong to each fuzzy set.

So far, we have assumed that all membership functions share the same confidence value. However, in real-world applications, it is more likely for each membership function to have its own confidence value. The confidence value can be informally understood as the extent to which a certain observation can be associated with a fuzzy concept. For example, our perception about the statement "0.3 is low" might be different from the statement "0.95 is high", assuming a unitary interval. If this situation comes to light, then we would need to truncate each membership function as indicated by its confidence value. The defuzzification procedure is not altered by this modification.

Example. Two scientists from the European Commission are asked to assess the likely effectiveness of several vaccines. To do that, they can use five linguistic terms: very low (VL), low (L), medium (M), high (H), and very high (VH). In addition, they must say the extent to which they are confident about their assessment. The commission has a limited budget, so they need a fair scoring system to rank the vaccines according to their effectiveness. The most promising ones will be funded. Let us design the scoring system.

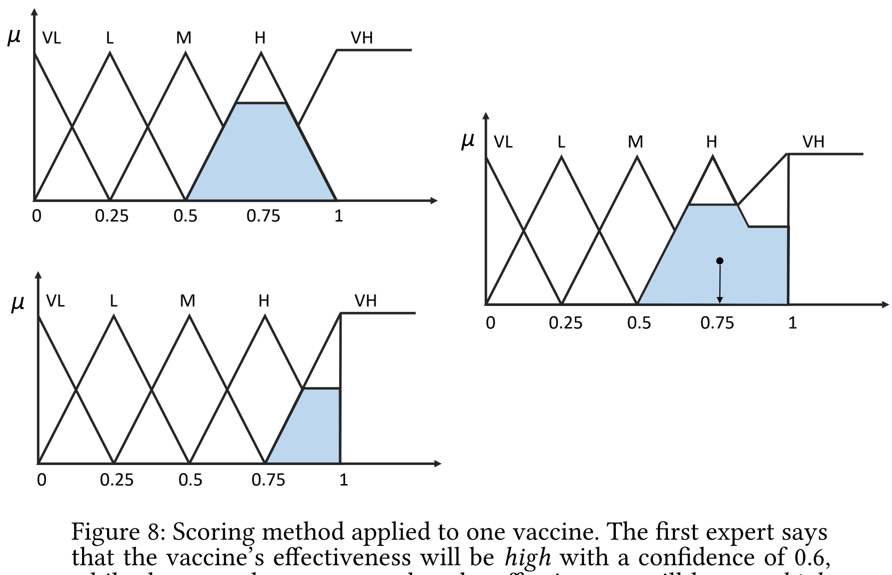


Figure 8: Scoring method applied to one vaccine. The first expert says that the vaccine's effectiveness will be *high* with a confidence of 0.6, while the second expert says that the effectiveness will be *very high* with a confidence of 0.5. After truncating an aggregating these functions, the centroid method reports a score of 0.75.

Fuzzy rules

Being able to transform numerical representations into symbolic ones allows us to obtain fuzzy IF-THEN rules. Doing that with numbers would be unfeasible due to the large number of rules to be produced. This means that we can create a symbolic Prolog knowledge base for problems involving numerical variables with relatively little effort. Moreover, we can reason directly with the fuzzy rules by implementing a *fuzzy inference system*. These reasoning systems are beyond the scope of this course. However, their foundations will help illustrate (through an example) how convenient fuzzy logic can be.

Tipping problem. Given an integer from 0 through 10 rating the service at a restaurant (where 10 is excellent), what should the tip be? This problem is based on tipping as it is typically practiced in the United States. An average tip for a meal in the United States is 15%, though the actual amount can vary depending on the quality of the service provided.

Let us model this problem using the bivalent logic. The social convention dictates that the minimum tip should be 10% of the bill, 15% if the service was good, and up to 25% if the service was exceptional. However, being a computer scientist sometimes is a curse. We want to be polite but we want to be fair with our wallet. Since the service is rated on a scale from 0 through 10, the tip could increase linearly from 5% if the service is bad to 25% if the service is excellent. This strategy does what we want it to do, and is straight forward. However, we may want the tip to reflect the quality of the food as well.

Extended tipping problem. Given two sets of numbers from 0 through 10 (where 10 is excellent) that respectively represent the quality of the service and the quality of the food at a restaurant, what should the tip be?

We can still apply the same strategy but combining variables to compute the right tip. Figure 9 shows the resulting hyper-plane. Now, let us suppose that you want the service to be a more important factor than the food quality. We can specify that service accounts for 80% of the overall tipping grade and the food makes up the other 20%. This would lead to a weighted sum. Figure 10 shows the resulting hyper-plane after this modification.

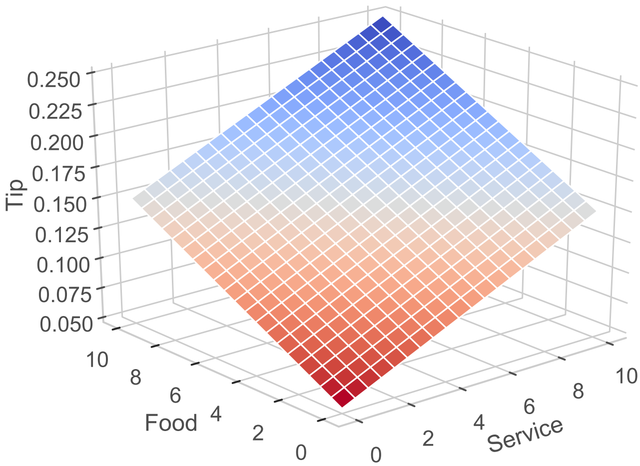


Figure 9: Function that calculates the tip considering the quality of the food and the quality of service. In this model, both components have the same weight when computing the tip.

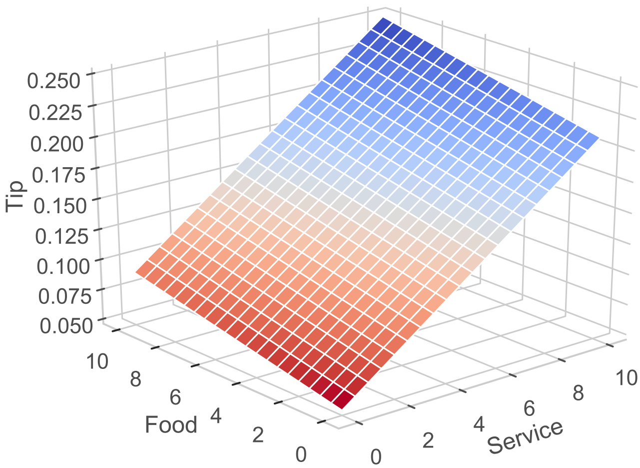


Figure 10: Function that calculates the tip considering the quality of the food and the quality of service. In this weighted model, service accounts for 80% of the overall tipping amount.

The function is still somehow too uniformly linear. Suppose that we want more of a flat response in the middle, that is, we want to give a 15% tip in general, but want to also specify a variation if the service is exceptionally good or bad. This means that the previous linear mappings no longer apply. Therefore, we will add a use piece-wise linear construction.

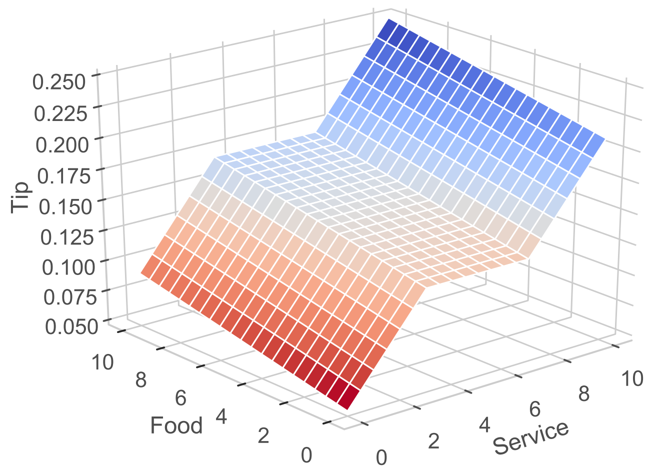


Figure 11: Function that calculates the tip considering the quality of the food and the quality of service. In this weighted model, service accounts for 80% of the overall tipping amount. In addition, we want to give a 15% tip in general.

Now let us solve this problem using fuzzy logic. Overall, we want to capture the essentials of this problem, leaving aside all the factors that could be arbitrary. If we make an inventory of what really matters in this problem, we will end up with the following set of fuzzy IF-THEN rules. R1) *If service is poor or the food is rancid, then tip is cheap*, R2) *If service is good, then tip is average*, R3) *If service is excellent or food is delicious, then tip is generous.*

These three rules are the core of your solution and they correspond to the rules for a fuzzy logic system. When you give mathematical meaning to the linguistic variables (what is an average tip, for example) we will have a complete *fuzzy inference system*. The methodology of fuzzy logic must also consider i) How are the rules combined? ii) How do we mathematically define what an average value is? However, answering that questions is outside the scope of this course since it moves beyond knowledge representation.

After implementing the fuzzy inference system we can obtain a function as depicted below. What I wanted to show with this example is that the function operating with the fuzzy IF-THEN rules is smoother than the ones computed before. It is worth recalling that the shape of our fuzzy function will depend on the granularity degree (number of fuzzy sets), the membership functions attached to each fuzzy set, and the aggregation operators.

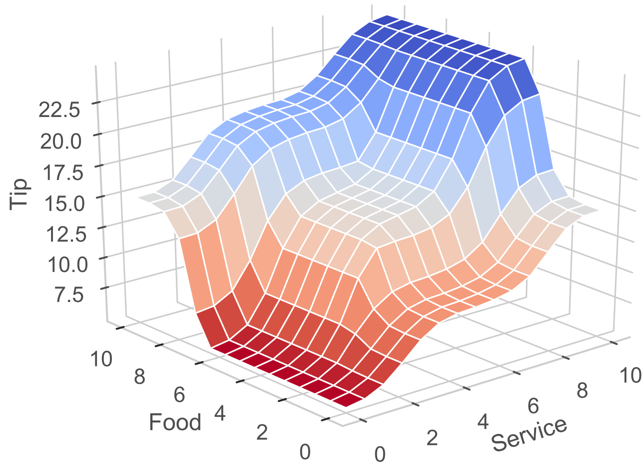


Figure 12: Function that calculates the tip considering the quality of the food and the quality of service. In this model, the tip is calculated by the aggregation of pre-defined fuzzy rules.

In a sense, fuzzy inference systems translate symbolic rules involving many fuzzy variables into either symbolic or numerical representations. For example, each rule in a Mamdani system produces a fuzzy set, while each rule in a Takagi-Sugeno-Kang system produces a linear function together with a confidence degree. However, as already mentioned, these fuzzy inference systems will not be studied in this course. They were mentioned for the sake of the completeness of these notes. Since this course concerns knowledge representation, we will only cover up the fuzzy rule construction.

The practical session will go over the concepts discussed in this chapter. In particular, we will practice the fuzzyfication and de-fuzzyfication operations using different type of membership functions.