Operations on fuzzy sets and relations, classical set Vs fuzzy set

Like classical set theory, fuzzy set theory includes operations union, intersection, complement, and inclusion, but also includes operations that have no classical counterpart, such as the modifiers concentration and dilation, and the connective fuzzy aggregation. In this section, all formulas are written with the assumption that only two sets are considered, but it is possible to extend all to three or more sets fairly easily by mathematical induction.

Fuzzification

In the process of fuzzification, membership functions defined on input variables are applied to their actual values so that the degree of truth for each rule premise can be determined.

Fuzzy statements in the antecedent are resolved to a degree of membership between 0 and 1.

- If there is only one part to the antecedent, then this is the degree of support for the rule.
- If there are multiple parts to the antecedent, apply fuzzy logic operators and resolve the antecedent to a single number between 0 and 1.

Antecedent may be joined by OR; AND operators.

- For OR -- max
- For AND -- min

Fuzzy Inferencing

In the process of inference

- Truth value for the premise of each rule is computed and applied to the conclusion part of each rule.
- This results in one fuzzy set to be assigned to each output variable for each rule.

The use of degree of support for the entire rule is to shape the output fuzzy set. The consequent of a fuzzy rule assigns an entire fuzzy set to the output. If the antecedent is only partially true, (i.e., is assigned a value less than 1), then the output fuzzy set is truncated according to the implication method. If the consequent of a rule has multiple parts, then all consequents are affected equally by the result of the antecedent. The consequent specifies a fuzzy set to be assigned to the output. The implication function then modifies that fuzzy set to the degree specified by the antecedent.

The following functions are used in inference rules.

min or *prod* are commonly used as inference rules.

min: truncates the consequent's membership function

prod: scales it.

Aggregation of all outputs

It is the process where the outputs of each rule are combined into a single fuzzy set.

- The input of the aggregation process is the list of truncated output functions returned by the implication process for each rule.
- The output of the aggregation process is one fuzzy set for each output variable.
  - Here, all fuzzy sets assigned to each output variable are combined together to form a single fuzzy set for each output variable using a fuzzy aggregation operator.

Some of the most commonly used aggregation operators are

- the maximum : point-wise maximum over all of the fuzzy sets
- the sum : (point-wise sum over all of the fuzzy) the probabilistic sum.

Defuzzification

In Defuzzification, the fuzzy output set is converted to a crisp number.

Some commonly used techniques are the *centroid* and *maximum* methods.

- In the *centroid method*, the crisp value of the output variable is computed by finding the variable value of the centre of gravity of the membership function for the fuzzy value.

- In the *maximum method*, one of the variable values at which the fuzzy set has its maximum truth value is chosen as the crisp value for the output variable.

Some other methods for defuzzification are:

- bisector, middle of maximum (the average of the maximum value of the output set), largest of maximum, and smallest of maximum, etc.

There are two types of fuzzy inference systems that can be implemented in the Fuzzy Logic Toolbox: Mamdani-type and Sugeno-type.

Mamdani Fuzzy Model

Mamdani's fuzzy inference method is the most commonly seen fuzzy methodology. Mamdani's method was among the first control systems built using fuzzy set theory. It was proposed in 1975 by Ebrahim Mamdani as an attempt to control a steam engine and boiler combination by synthesizing a set of linguistic control rules obtained from experienced human operators.

To compute the output of this FIS given the inputs, one must go through six steps:

1. Determining a set of fuzzy rules

2. Fuzzifying the inputs using the input membership functions,

3. Combining the fuzzified inputs according to the fuzzy rules to establish a rule strength,

4. Finding the consequence of the rule by combining the rule strength and the output membership function,

5. Combining the consequences to get an output distribution, and

6. Defuzzifying the output distribution (this step is only if a crisp output (class) is needed).

The following is a more detailed description of this process.

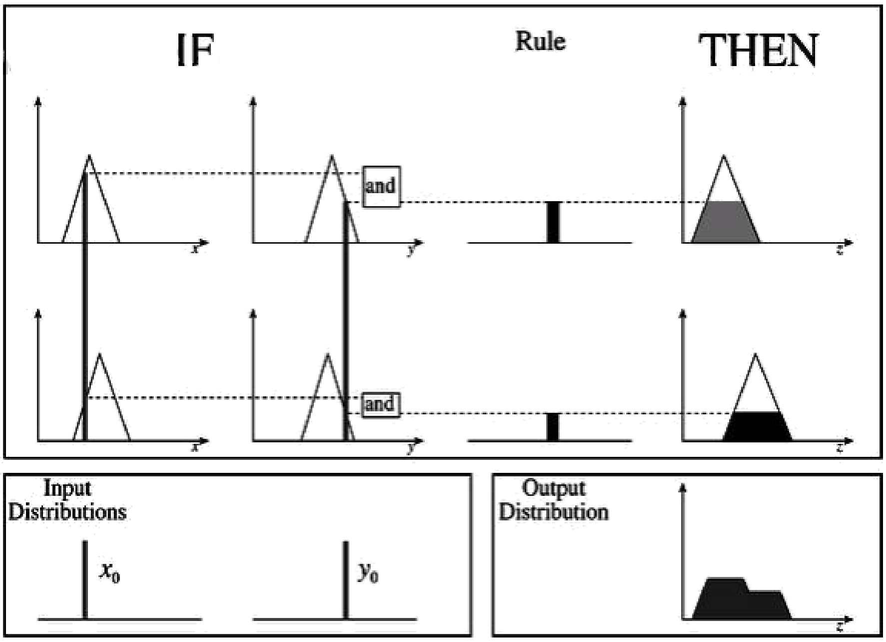


Fig 5. A two input, two rule Mamdani FIS with crisp inputs

Creating fuzzy rules

Fuzzy rules are a collection of linguistic statements that describe how the FIS should make a decision regarding classifying an input or controlling an output. Fuzzy rules are always written in the following form:

if *(input1 is membership function1)* ***and/or*** *(input2 is membership function2)* ***and/or then*** *(output n is output membership function n)*

Example

$$\begin{cases} \text{If } X \text{ is small then } Y \text{ is small.} \\ \text{If } X \text{ is medium then } Y \text{ is medium.} \\ \text{If } X \text{ is large then } Y \text{ is large.} \end{cases}$$

Another Example

if *temperature is high* ***and*** *humidity is high* ***then*** *room is hot.*

There would have to be membership functions that define what we mean by high temperature (input1), high humidity (input2) and a hot room (output1). This process of taking an input such as temperature and processing it through a membership function to determine what we mean by "high" temperature is called fuzzification.Also, we must define what we mean by "and" / "or" in the fuzzy rule. This is called fuzzy combination.

Fuzzification

The purpose of fuzzification is to map the inputs from a set of sensors (or features of those sensors such as amplitude or spectrum) to values from 0 to 1 using a set of input membership functions. In the example shown in the above figure, there are two inputs, x0 and y0 shown at the lower left corner. These inputs are mapped into fuzzy numbers by drawing a line up from the inputs to the input membership functions above and marking the intersection point.

These input membership functions, as discussed previously, can represent fuzzy concepts such as "large" or "small", "old" or "young", "hot" or "cold", etc. When choosing the input membership functions, the definition of what we mean by "large" and "small" may be different for each input.

Fuzzy Combinations

In making a fuzzy rule, we use the concept of "and", "or", and sometimes "not". The sections below describe the most common definitions of these "fuzzy combination" operators. Fuzzy combinations are also referred to as "T-norms".

a) Fuzzy "and"

The fuzzy "and" is written as:

$$u_{A \cap B} = T(u_A(x), u_B(x))$$

where μ A is read as "the membership in class A" and μB is read as "the membership in class B". There are many ways to compute "and". The two most common are:

1. Zadeh - min(uA(x), uB(x)) This technique, named after the inventor of fuzzy set theory simply computes the "and" by taking the minimum of the two (or more) membership values. This is the most common definition of the fuzzy "and".

2. Product - ua(x) times ub(x)) This techniques computes the fuzzy "and" by multiplying the two membership values.

Both techniques have the following two properties:

T(0,0) = T(a,0) = T(0,a) = 0

T(a,1) = T(1,a) = a

One of the nice things about both definitions is that they also can be used to compute the Boolean "and". The table below shows the Boolean "and" operation. Notice that both fuzzy "and" definitions also work for these numbers. The fuzzy "and" is an extension of the Boolean "and" to numbers that are not just 0 or 1, but between 0 and 1.

Input1 (A)	**Input2 (B)**	**Output (A "and" B)**
0	0	0
0	1	0
1	0	0
1	1	1

The Boolean "and"

b) Fuzzy “or”

The fuzzy "or" is written as:

$$u_{A \cup B} = T(u_A(x), u_B(x))$$

Similar to the fuzzy "and", there are two techniques for computing the fuzzy "or":
1. Zadeh - max(uA(x), uB(x)) This technique computes the fuzzy "or" by taking the maximum of the two (or more) membership values. This is the most common method of computing the fuzzy "or".

2. Product - uA(x)+ uB(x) - uA(x) uB(x) This technique uses the difference between the sum of the two (or more) membership values and the product of the membership values.

Both techniques have the following properties:

T(a,0) = T(0,a) = a

T(a,1) = T(1,a) = 1

Similar to the fuzzy "and", both definitions of the fuzzy "or" also can be used to compute the Boolean "or". The table below shows the Boolean "or" operation. Notice that both

fuzzy "or" definitions also work for these numbers. The fuzzy "or" is an extension of the Boolean "or" to numbers that are not just 0 or 1, but between 0 and 1.

Input1 (A)	**Input2 (B)**	**Output (A "or" B)**
0	0	0
0	1	1
1	0	1
1	1	1

The Boolean "or"

Fig 5

c) Consequence

The consequence of a fuzzy rule is computed using two steps:
1. Computing the rule strength by combining the fuzzified inputs using the fuzzy combination process discussed previously. This is shown in Fig 5. Notice in this example, the fuzzy "and" is used to combine the membership functions to compute the rule strength.

2. Clipping the output membership function at the rule strength. Once again, refer to Fig 5. to see how this is done for a two input, two rule Mamdani FIS.

d) **Combining Outputs into an Output Distribution**

The outputs of all of the fuzzy rules must now be combined to obtain one fuzzy output distribution. This is usually, but not always, done by using the fuzzy "or". Figure 5 shows an example of this. The output membership functions on the right hand side of the figure are combined using the fuzzy "or" to obtain the output distribution shown on the lower right corner of the figure.

e) **Defuzzification of Output Distribution**

In many instances, it is desired to come up with a single crisp output from a FIS. For example, if one was trying to classify a letter drawn by hand on a drawing tablet, ultimately the FIS would have to come up with a crisp number to tell the computer which letter was drawn. This crisp number is obtained in a process known as defuzzification. There are two common techniques for defuzzifying:

1. **Center of mass** - This technique takes the output distribution found previously and finds its center of mass to come up with one crisp number. This is computed as follows:

$$z = \frac{\sum_{j=1}^{q} Z_j u_c(Z_j)}{\sum_{j=1}^{q} u_c(Z_j)}$$

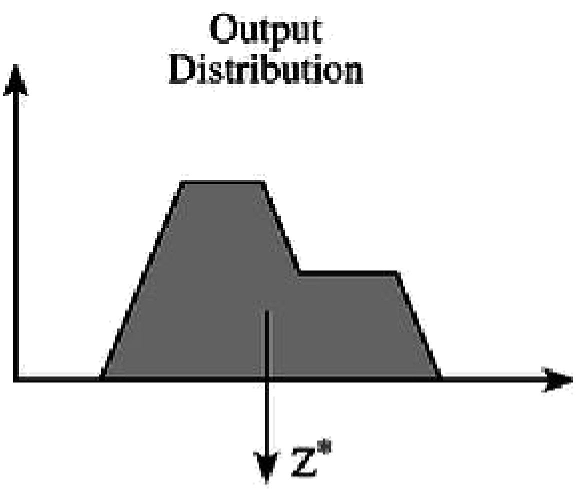
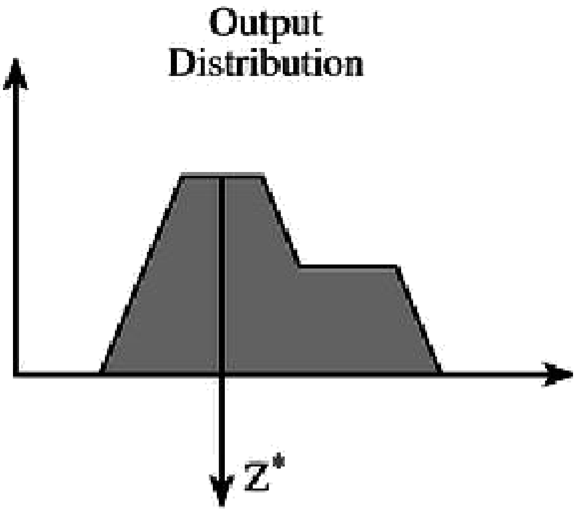


Fig 6. Defuzzification Using the Center of Mass

where z is the center of mass and uc is the membership in class c at value zj. An example outcome of this computation is shown in Fig 6.

2. **Mean of maximum** - This technique takes the output distribution found previously and finds its mean of maxima to come up with one crisp number. This is computed as follows:

$$z = \sum_{j=1}^{l} \frac{z_j}{l}$$



where z is the mean of maximum, zj is the point at which the membership function is maximum, and l is the number of times the output distribution reaches the maximum level. An example outcome of this computation is shown in Figure 7.

Fig 7. Defuzzification Using the Mean of Maximum

Fuzzy Inputs

In summary, Fig 5 shows a two input Mamdani FIS with two rules. It fuzzifies the two inputs by finding the intersection of the crisp input value with the input membership function. It uses the minimum operator to compute the fuzzy "and" for combining the two fuzzified inputs to obtain a rule strength. It clips the output membership function at the rule strength. Finally, it uses the maximum operator to compute the fuzzy "or" for combining the outputs of the two rules.

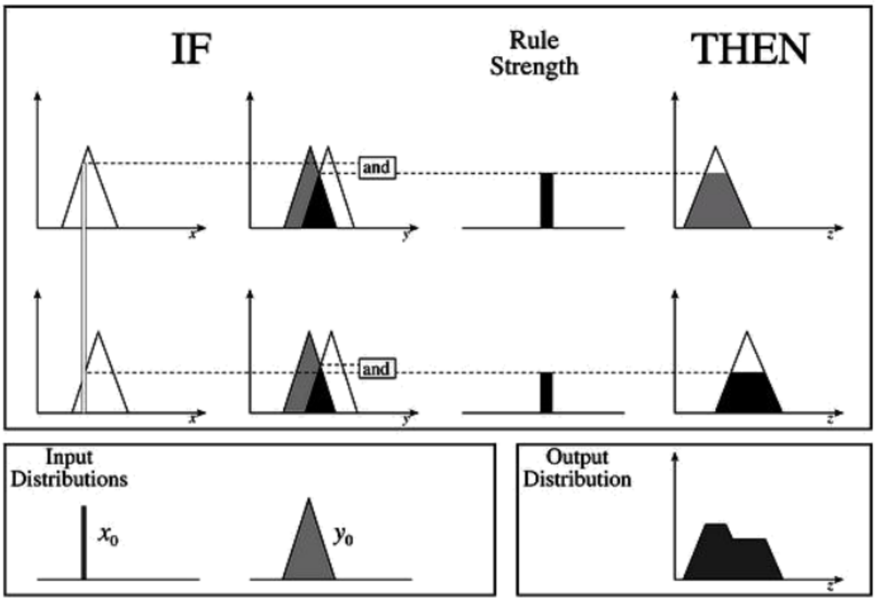


Fig 8. A two Input, two rule Mamdani FIS with a fuzzy input

Fig 8 shows a modification of the Mamdani FIS where the input y0 is fuzzy, not crisp. This can be used to model inaccuracies in the measurement. For example, we may be measuring the output of a pressure sensor. Even with the exact same pressure applied, the sensor is measured to have slightly different voltages. The fuzzy input membership function models this uncertainty. The input fuzzy function is combined with the rule input membership function by using the fuzzy "and" as shown in Fig 8.

References

De Cock, Martine; Bodenhofer, Ulrich; Kerre, Etienne E. (1–4 October 2000). Modelling Linguistic Expressions Using Fuzzy Relations. Proceedings of the 6th International Conference on Soft Computing. Iizuka, Japan. pp. 353–360

Klaua, D. (1965) Über einen Ansatz zur mehrwertigen Mengenlehre. Monatsb. Deutsch. Akad. Wiss. Berlin 7, 859–876. A recent in-depth analysis of this paper has been provided by Gottwald, S. (2010). "An early approach toward graded identity and graded membership in set theory". Fuzzy Sets and Systems.

Liang LR, Lu S, Wang X, Lu Y, Mandal V, Patacsil D, Kumar D. FM-test: a fuzzy-set-theory-based approach to differential gene expression data analysis. BMC Bioinformatics. 2006