UiT The Arctic University of Norway

DTE-2501 AI Methods and Applications

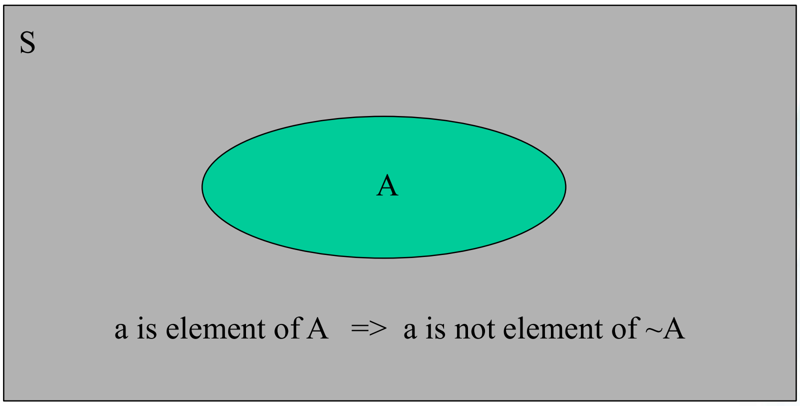
Lecture 2/2 – Boolean logic vs Fuzzy logic

Andreas Dyrøy Jansson
PhD Candidate
C3190
andreas.d.jansson@uit.no

This is lecture 2 of 2 in a series on probabilistics theory and fuzzy logic

Classic Boolean Logic

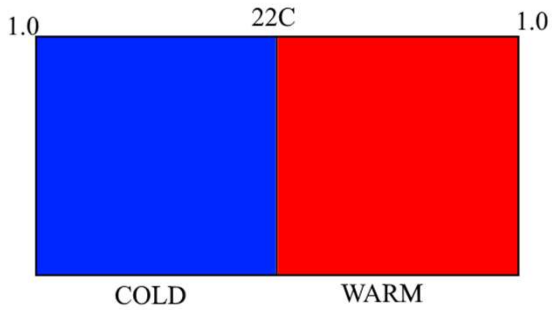
- You are either in or you are out
  - True or false
  - Yes or no



Let's start by taking a look at classic Boolean logic. In Boolean logic, you are working with values that are either true or false, yes or no. You are either part of a set, or not part of a set. Simple as that. This is fine for many applications, but let us consider a simple example.

A practical example: temperature

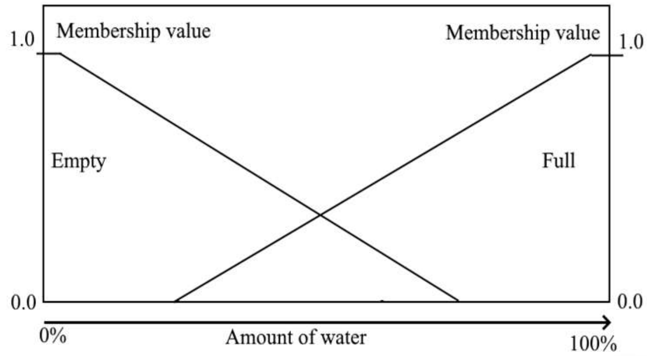
- Strict boolean logic is not always the best way to model a problem
- Example: temperature
  - Everything below 22 C is COLD and everything above is WARM
  - In other words, IF temp > 22 THEN WARM, ELSE COLD



Take temperature. Let us say that we are programming a heater to respond to a certain temperature. If we are working with simple true and false values, we have to determine a set point to work around. If we assume that normal indoor temperature is 22 centigrades, we can program our heater to turn on if it gets below this temperature. In other words, we claim that everything below 22 is cold, and everything above is warm. Then, our heater will do nothing as long as it is warm, and will let the temperature drop until it gets below our set point. Then, it will turn on at full blast, wasting energy.

Is a glass half empty or half full?

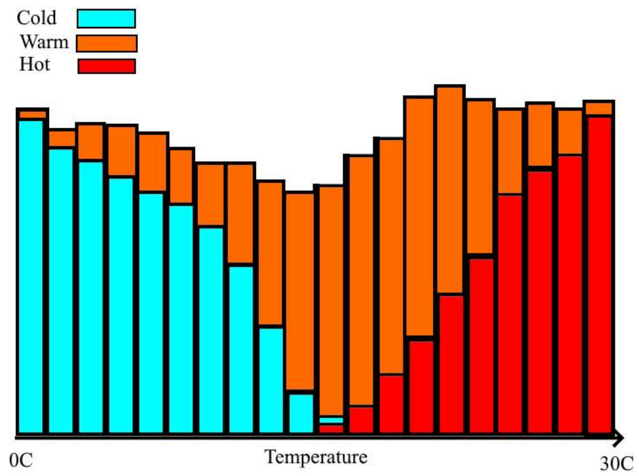
- A possible answer to the age-old debate:
  - Fuzzy sets and fuzzy logic
    - Zadeh, 1965
  - Membership values
  - “Possibility”-theory
  - The glass is part full
    - But also part empty
  - Truth value
    - How “true” is our claim?



Let’s consider another example. Is a glass half full or half empty? A possible answer to this can be given using fuzzy logic. The term was introduced by Azerbaijani scientist Lofti Zadeh in 1965, but has roots back to the 1920s. Fuzzy logic is based on the concept of membership values and possibility theory. In other words, we can say that the glass is part full, while still being part empty. Membership values, or truth values, gives us a number between zero and one which says how true a claim is.

Back to the temperature example

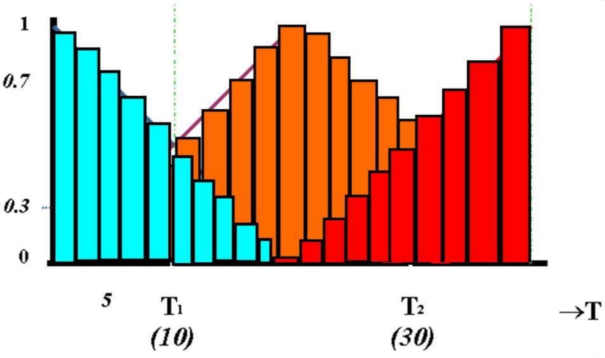
- What is «cold»?
- What is «warm»?
  - Who are we asking?
  - In what context?
- If we ask a group of people, we may get something like this:
- This is our fuzzy set
  - There is no «hard» edge between cold, warm and hot



Let us go back to the temperature example. How can we determine what is cold and warm? It will depend on who we are asking, and in what context. Let's say we ask a group of people to label temperatures from 0 to 30 as cold, warm and hot. If we were to visualize the data, we might get something like this figure. This is our fuzzy set, as there are several overlaps between classes, and there is no hard edge between cold, warm and hot.

Modelling this as a fuzzy system

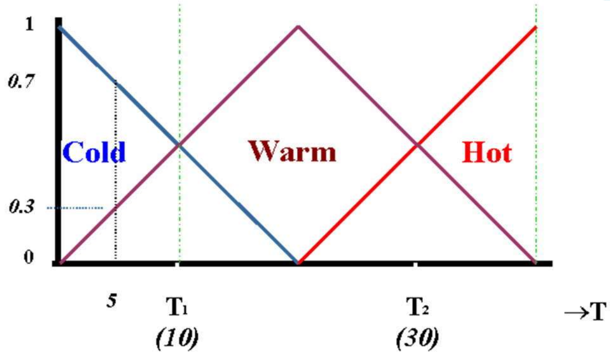
- Three states: Cold, Warm and Hot
- We assign membership values based on our findings
  - We calculate the probability for each state based on our findings – this becomes the truth values for each temperature
- This is called «fuzzification»



We can use our data to model a fuzzy input system. We are working with three states: cold, warm, and hot. We assign the membership values to each class based on our findings, as probabilities of a temperature belonging to a certain class. This also becomes the truth value for each class at a given temperature. This process is called fuzzification. We then get the following fuzzy input system.

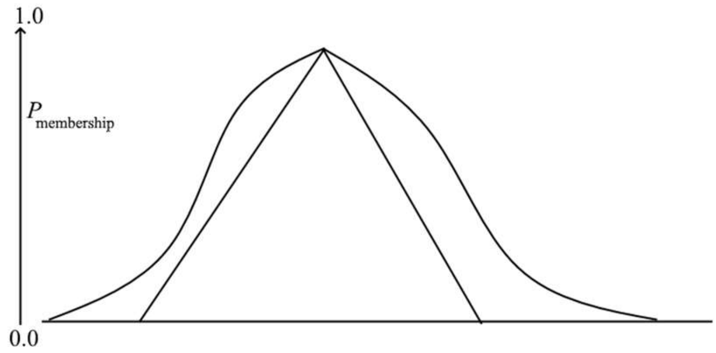
Modelling this as a fuzzy system

- Three states: Cold, Warm and Hot
- We assign membership values based on our findings
  - We calculate the probability for each state based on our findings – this becomes the truth values for each temperature
- This is called «fuzzification»



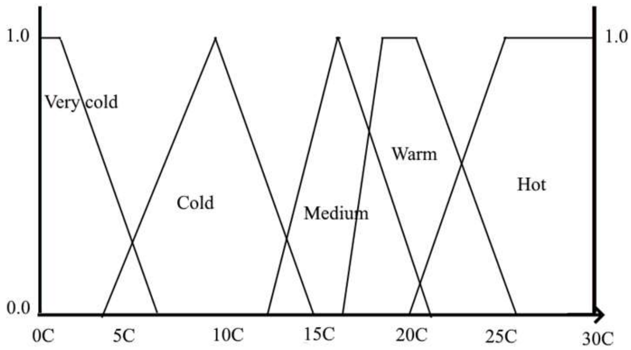
Different shapes express differences in fuzzy set membership

- In essence, we are interested in the probability of an input belonging to one or more of our classes



Depending on what method we use to calculate the probability of truth value for each class, our function may make take on different shapes. In essence, we are only interested in the probability that an input belongs to one or more of our classes.

We can add more classes as desired:



We can even add more classes if we want. For temperature, we can enhance our input set with classes like very cold, medium, and so on.

The construction procedure for a fuzzy inference system

1. Express the states to be monitored in real world terms
   1. i.e., VERY COLD, HOT, etc.
2. Fuzzificate by establishing the fuzzy sets for these input states applying a suitable set of functions
3. Define the reaction modes in real world terms
   1. i.e., EMPTY, FULL, HIGH, LOW etc.
4. Fuzzificate similarly – using suitable functions

In summary, we have the following steps for constructing a fuzzy inference system:

First, start, by expressing the states to be monitored in real world terms. So far we have looked at classes like Very cold, hot, et cetera. Then, we fuzzificate by establishing the fuzzy sets for these input states applying a suitable set of functions. We then define the reaction modes in real world terms, like empty, full, high, low, and so on.

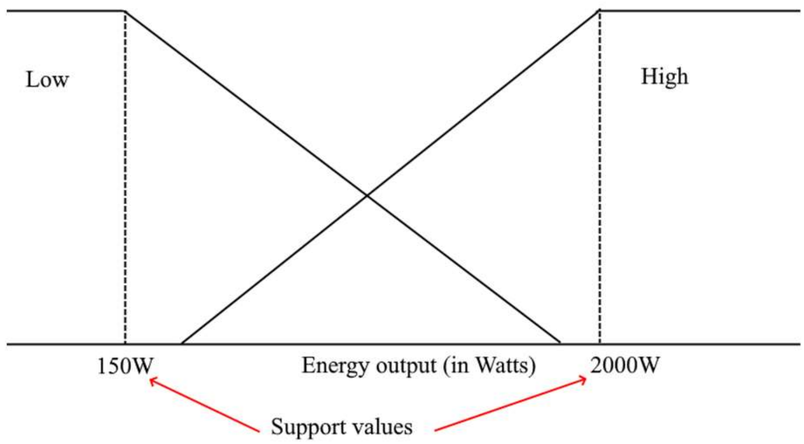
Finally, we use the same fuzzification process on the outputs using suitable functions.

Using fuzzy logic for control

- Example: heater output
  - Base logic:
    - IF temperature IS COLD THEN HIGH OUTPUT
    - IF temperature IS WARM THEN LOW OUTPUT
  - Again: what is «cold» and what is «warm»?
    - How to deal with uncertainty?

Alright, but why do we do this? Let’s look at a practical use case: heater output. How can we program a heater to output the appropriate wattage for any given situation? If we were to simply use a rule like the first we looked at, we would get a rather poor system.

Heater control based on fuzzy logic



Let’s fuzzificate the heater’s output. The heater has two main modes: high output and low output.

Support values

- Each output mode is associated with a support value
  - A parameter for one type of output
- We usually select from the middle of the output set – where the associated truth value is high
- Used for defuzzification

The heater in our example can output between 150 watts and 2000 watts. These will be our support values, associated with each of the output modes. We usually select from the middle of the output set, in other words, where the associated truth value is high. Support values are used in the defuzzification step.

Defuzzification

- We must defuzzicate in order to reach a conculsion
- **Regular Mamdani**: center of gravity of resulting output set

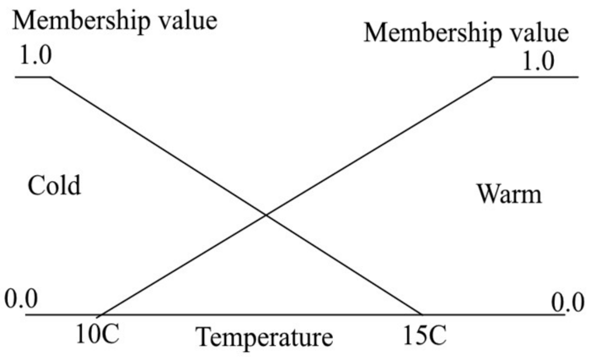
$$X = \frac{\sum \mathbf{x} * \mu_x \Delta x}{\sum \mu_x * \Delta x}$$

- **Sugeno** using simple singletons (support values for output set)
  - Non scaled: SUM for all i (input truth value$_i$ * support value$_i$)
  - Scaled: NonScaled output / SUM for all i(input truth value$_i$)

In order to actually reach a conclusion we must defuzzicate. Defuzzification methods include the regular Mamdani, which calculates the center of gravity of our resulting output set, and Sugeno, which use the support values for the output set, called singletons.

Defuzzification example

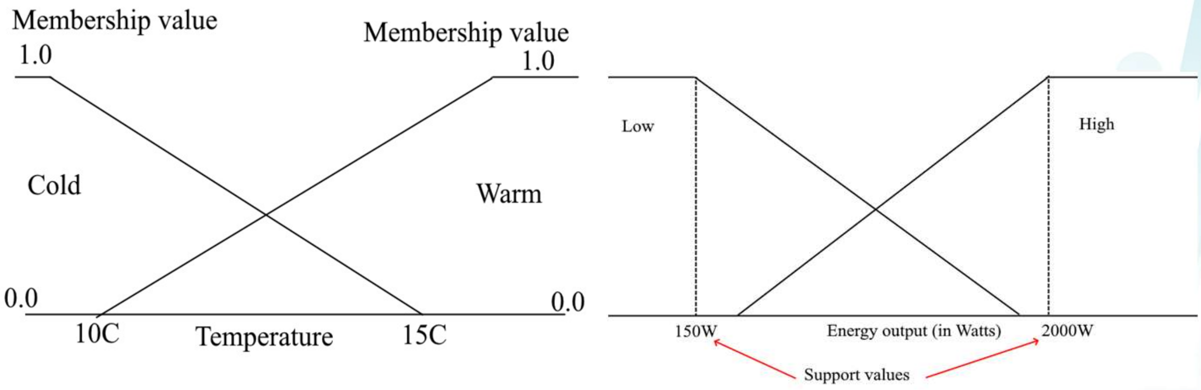
- Given the following inputs and outputs:



Let’s go back to our heater control example. On the input side, we have the following. Similarly, for the heater output, we have this.

Defuzzification example

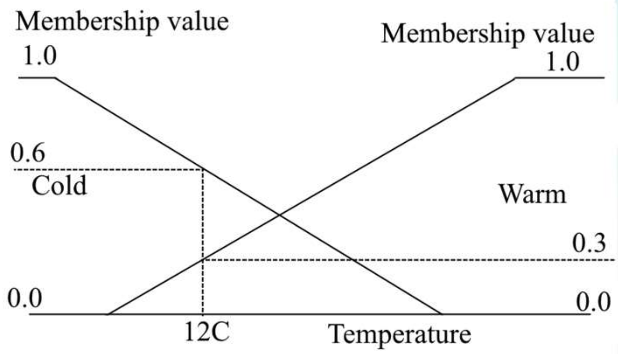
- Given the following inputs and outputs:



Let’s go back to our heater control example. On the input side, we have the following. Similarly, for the heater output, we have this.

Defuzzification example

- Remember:
  - IF temperature IS COLD THEN HIGH OUTPUT
  - IF temperature IS WARM THEN LOW OUTPUT
- Input: 12C
  - This means that it is COLD with a truth value of 0.6
  - But it is also slightly WARM, with a truth value of 0.3



The heater has the following outputs. IF temperature is COLD then output is HIGH. And if temperature is WARM then output is LOW. Let's say that we measure a temperature of 12 degrees. Looking at out input, this means that it is COLD with a truth value of 0.6. But it is also a little bit WARM, with a truth value of 0.3

Defuzzification example

- From this, we can infer that:
  - Heater must provide HIGH output with a truth value of 2000 * 0.6 = 1200W
  - Heater must provide LOW output with a truth value of 150 * 0.3 = 45W
- This is harmonized as part of the defuzzification process:
  - (1200 + 45) / (0.3 + 0.6) = 1383 Watt
- We therefore conclude that the heater should output 1383 Watt of heat when the temperature is 12C

From this, we can infer that the heater must provide HIGH output with a truth value of 2000 times 0.6, which is 1200 watts. In addition the heater must also provide low output with a truth value of 150 times 0.3 equals 45 watts.

This is harmonized as part of the defuzzification process:

1200 plus 45 divided by 0.3 plus 0.6 equals 1383 watts.

We can therefore conclude that the heater should output 1383 watt of heat when the temperature is 12 degrees.

Real-world applications of Fuzzy control

- Nissan: fuzzy automatic transmissions
- Nissan: fuzzy anti-skid braking system
- Subaru/Honda: continuously variable (fuzzy) transmission
- Mitsubishi: fuzzy air conditioner (FC 110) Toshiba: Camcorder
- Electrolux: Fuzzy home appliances
- NASA: Fuzzy control in space station docking
- Sendai, Japan: Subway train control
- And many, many more…..

Fuzzy controllers are used in many different applications in the real world, some examples include automatic transmissions for cars, air conditioners and heaters, home appliances, subway train control, and many more.