

CS4047 Assignment: A Fuzzy Rule Based System

The aim of this assignment is for you to implement a simple general purpose Fuzzy Rule Based System, test it thoroughly, and analyse and explain the results from/ operation of the system.

The hand in date is: **Friday 17th November 2017 at 12 noon**

NB: This assignment is about being able to build and fuzzy rule based system and critically evaluate its operation. It is not about building and fancy UI. You may build one, but that will not gain you any extra marks. I am only interested in your ability with respect to Computational Intelligence, not HCI!

You may perform the implementation in a **language of your choice**. The only restriction is that I must be able to run it . I am a Mac user with access to Unix/Linux (I do not use a pc). I have Java, Lisp, and Python installed on my machine, and can access C and C++ (there may be others but you would have to give adequate instructions on running your system).

You should use the example we went over in the lectures to test your system initially. Your solution should, however, be general purpose as I may try it with different scenarios.

The assignment is divided into the following tasks, perfect completion of which will earn you the associated mark.

1. Examine the lecture notes and from those describe the structure of a fuzzy rule based system including how it is decomposed it into a suitable number of parts. For this part you should hand in a high level description of the problem, and how it may be decomposed. **25%**
2. Create a design document for your system identifying the main features and their functionality (e.g. what approach are you going to take for each part). This part should consist of a brief but fairly detailed design description. **20%**
3. Implement each of the components you designed in part 2 and test their operation. (You could use the intermediate calculations presented in the lectures/ tutorials for this purpose). For this part you should hand in the code of your components, with a brief description of each part and some sample results from your testing of each component. **20%**
4. Provide a complete fuzzy rule-based system made up from the components you developed in part 3. This system should be able to read in a fuzzy rule base, a set of variables and their fuzzy values, and a set of measurements (see below for details of the format) and output the required defuzzified value for the relevant variable. You may use the example discussed in the lectures to test your system (though you are free to make up your own examples). For this part you should hand in any additional code not already given in part 3, a description of your system and the results of your testing. **20%**
5. Do a more thorough testing of your system and analyse the results. You may wish to exhaustively check the response to all possible values of the inputs (for the example in the lectures these are the Current and Temperature). Discuss your findings.

For this part you should hand in a description of the additional examples you ran, a discussion of the results you obtained - including any improvements that might be made to your system or the specification of the rule base or fuzzy sets. 15%

Some points to take note of in doing this assignment.

- While there are a number of ways in which a fuzzy set can be represented, **you must use the 4 tuple representation** presented in the lecture.
- To keep your system simple you should use only one type of logical connective (AND or OR) in a rule (if you use any). You may use as many connectives in the rule as required, but do not mix them.

Precise submission instructions will follow nearer the time.

The format of the data to be input to your system must be as follows. (For testing the system I will expect that these data can be read in from a text file.)

For the rulebase:

<RuleBaseName>

Rule 1: if <variable1> is <value1> [and|or] [<variablen> is <valuen>]
then <variablei> is <valuej>

Rule 2: if <variable2> is <value2> [and|or] [<variablen> is <valuen>]
then <variablei> is <valuej>

...

For setting up the fuzzy sets for each variable:

<variableName1>

<valueName1> <4Tuple1>

...

<valueName1> <4Tuple1>

And for the measurements:

<variableName1> = <RealValue1>

...

<variableNamen> = <RealValuen>