# G53FUZ - Designing and Tuning a Fuzzy Inference System

*by Matheus Serpellone*

## Introduction

The goal of this coursework is to create a Fuzzy Inference System for a hypothetical scenario to recommend a patient to look for help based on a their temperature and headache levels. The system was created using R and the Fuzzy Toolkit by the University of Nottingham. Various fuzzy systems were considered and a simple evaluation method was used to select the best amongst them.

## The System

This was the base system used to test the parameters. The first two variables “temperature” and “headache” are the input variables, and “emergency” is the output variable.

| **Variable** | **Membership Functions** | | | | |
|---|---|---|---|---|---|
| Temperature | Very Low | Low | Normal | High | Very High |
| Headache | Normal | Mild | Moderate | Severe | Profound |
| Emergency | None | Mild | Careful | Worry | Urgency |

The rules were very simple, as I noticed that adding too many rules made the system hard to handle and the surface didn’t reflect normal reasoning on what the expected emergency was supposed to be. This was a simple solution that worked and wasn’t over complicated.

*If t*emperature *is* normal *and* headache *is* none *then* emergency *is* none
*If* temperature *is* very_high *then* emergency *is* urgent
*If* temperature is very_low *then* emergency *is* urgent
*If* headache *is* profound *then* emergency *is* urgent

# Evaluation System

I used a simple comparison system to determine the quality of a FIS. I set up some values for each input variable on the FIS and also for their expected outcome, and compared this expected outcome with the actual prediction from the fuzzy system.

After normalising and summing all the errors, I obtained a value of how good the fuzzy system was compared to the test data. The goal was then to minimise this value with different settings for the FIS.

$$J = \sum_{i \in test\ data} \sqrt{(expected(i) - predicted(i))^2}$$

Test data:

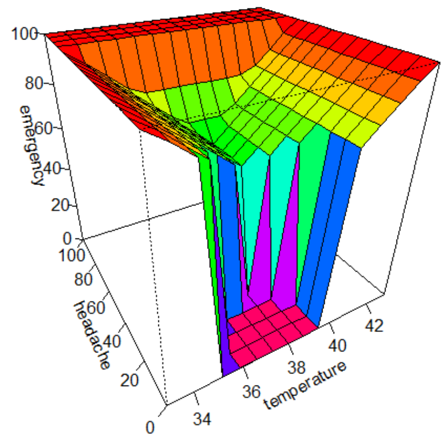
| Temperature | Headache | Emergency |
|---|---|---|
| 43 | 100 | 80 |
| 37 | 0 | 0 |
| 39 | 20 | 50 |
| 35 | 20 | 50 |
| 33 | 0 | 80 |
| 43 | 0 | 80 |
| 36 | 20 | 15 |

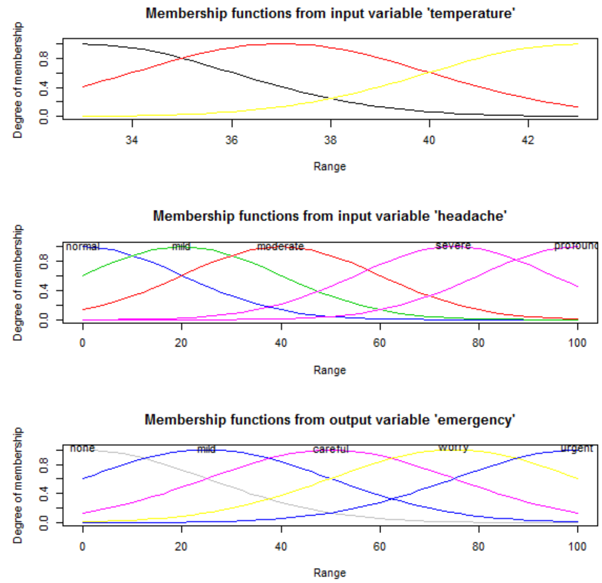
# Results

The Fuzzy Toolkit could only operate under the Mamdani Fuzzy Systems, so that was the only one tested.

| Defuzzification | And/or operators | FIS type | # Inputs | # Outputs | J (cost function) |
|---|---|---|---|---|---|
| centroid | min/max | Mamdani | 5/5 | 5 | 92 |
| bisector | min/max | Mamdani | 5/5 | 5 | 95 |
| medium of max | min/max | Mamdani | 5/5 | 5 | 113 |
| smallest of max | min/max | Mamdani | 5/5 | 5 | 87 |
| largest of max | min/max | Mamdani | 5/5 | 5 | 164 |
| | | | | | |
| smallest of max | prod/prod | Mamdani | 5/5 | 5 | 87 |
| | | | | | |
| smallest of max | prod/prod | Mamdani | 3/5 | 5 | 87 |
| smallest of max | prod/prod | Mamdani | 3/3 | 5 | 141 |
| | | | | | |
| smallest of max | prod/prod | Mamdani | 3/5 | 3 | 94 |

According to the Occam's Razor principle, we should choose the simplest FIS that minimises the cost function. Which is a Mamdani FIS with the product operator as the AND/OR evaluators, smallest of maxima defuzzification method, 3 membership functions for the temperature input variable, 5 membership functions for the headache variable and 5 membership functions for the emergency output variable.




# Conclusions

Fuzzy Systems are very versatile and simple to begin using, but tuning and optimising them to your precise needs require extra work. There are a lot of parameters and combinations and it may be overwhelming to test all combinations if the application is too large.

Regarding the rules, my impression was that only the most basic rules should be written into the system, usually the edge case scenarios, e.g. the normal and urgent emergency on this coursework's output. Adding more rules in this case made the system noisy and the surface plot did not represent the meaning that is desired for the FIS.

The distribution of the membership functions could be tested programmatically. However it would be very computationally expensive to test all values for all MF. Besides, most points aren't relevant to the membership functions, for example it doesn't make sense to test a "cold" membership function that is a triangle centered at 40℃ so instead some manual, reasonable values were tested.