Weekly Exercises Fuzzy Logic Week 6

Name**: Babu Pallam**
Pnumber**: P2849288**

Exercise 15

The given problem is to find the amount of food which need to be fed to a cat, where two influencing factors exist, such as size of the cat and the level of the noise (miaows) it makes. To solve this problem using type 1 Mamdani Model of Fuzzy Inference System with the structure given in Fig.1 in the questions, the practical implementation follows mainly four steps, such as Fuzzification, Rule Inference, Rule Aggregation, Defuzzification.

Question 1

Step 1: Fuzzification

Fuzzification is the process of creating fuzzy sets from the crisp sets with the help of Knowledge Base. With the help of membership functions, the inputs and outputs can be fuzzified with appropriate linguistic variables.

Knowledge Base consists of the input's values and output values with the domain specification. In the given example, the knowledge base can be defined as given below.

- Inputs:
  1. size :
     - Defines: the size of the cat,
     - Values range: 1 to 10
  2. miaows:
     - Defines: the level of noise the cat makes, more miaows, the hungrier they are
     - Values range: 1 to 10
- Outputs:
  1. Amount:
     - Defines: the amount of food required to be fed
     - Values range: 20 to 150 (in grams)

With the help of above values and the rules given in the Fig.1, the linguistic variable can be defined as

- Inputs:
  - size:
    - small (indicates small cat)
    - medium (indicates medium cat)
    - large (indicates large cat)
  - miaows
    - intermittent (noise on regular intervals)

        - persistent (continuous noise)
- Output
    - amount:
        - a_bit (small amount of food)
        - average (average amount of food)
        - a_lot (large amount of food)

The inputs and output can be fuzzified using membership function (MF). Commonly used MFs are, triangular, trapezoidal, gaussian, and others.

Assume that MF chosen is triangular MF, and the parameters of each variable are as given below. Each set has three values since we use triangular MF,

[lower bound, peak value, upper bound]

- Inputs:
    - size: [0,10]
        - small [0,2,6]
        - medium [4,7,9]
        - large [7.5,9,10]
    - miaows[0,10]
        - intermittent [0,3,7]
        - persistent [4,8,10]
- Output
    - amount:[20,150]
        - a_bit [20,30,70]
        - average [40,80,100]
        - a_lot [90,130,150]

Based on these given values, a FIS system has been defined using MATLAB code, the resulted graphs are provided below for input and output as Figure 1. [See Appendix A, A1 for the code]

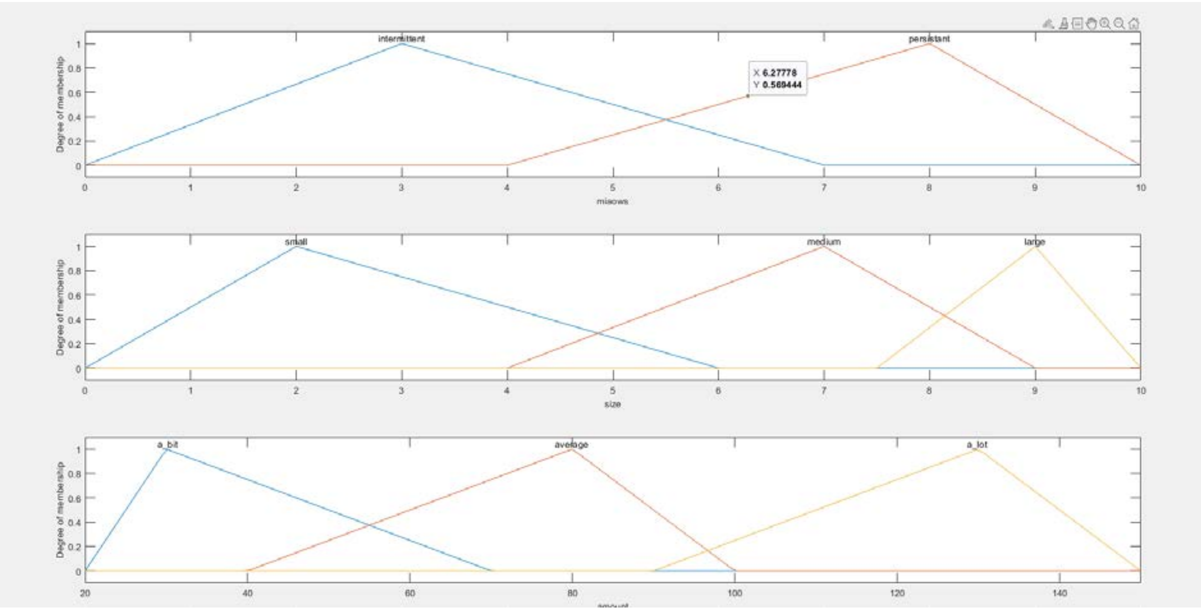
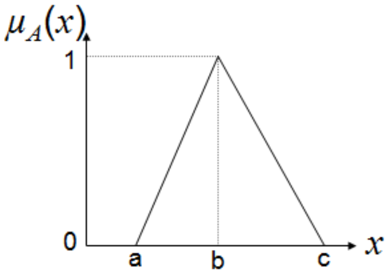


Figure 1

Since we use, triangular function to plot the membership function distribution, the fuzzy sets of each input-linguistic variable can be found using the given formula given below[1],

$$\mu_A(x) = \begin{cases} 0 & if\ x \le a \\ \frac{x-a}{b-a} & if\ a \le x \le b \\ \frac{c-x}{c-b} & if\ b \le x \le c \\ 0 & if\ x \ge c \end{cases}$$



For example, the fussy set of size -medium, can be calculated as given below for different values, provided the parameters [0,2,6]

Here a = 0, b=2, and c=6, while applying in the above equation,

Then Fuzzy set describes as $\mu_{medium}(x)$ = {(0,0),(1,0.5),(2,1),(4,0.5),(5,0.25),(6,0)}

Step 2: Rule Inference

This step involves evaluation process of rules, which is assumed to be created in Rule Base.

The **Rule Base** consists of set of rules which says the connection between the input variables and output variables, based on linguistic variable of those. Fig1. given in the question contains four rules, as given below.

1. If **miaows** are **persistent**, then **amount** is **a_lot**
2. If **miaows** are **intermittent** and **size** is **small**, then **amount** id **a_bit**
3. If **miaows** are **intermittent** and **size** is **medium**, then **amount** is **average**
4. If **size** is **large** then **amount** is **a_lot**

Each of the given rule gets evaluated based on the MFs defined in the fuzzification process. This is done in two steps. First, apply fuzzy operator between the fuzzy sets, i.e in between two MFs that denoted the input variable. Second, apply **Mamdani Implication**, which allows us to find the output fuzzy set based on the fuzzy input sets.

For example, take the second rule:

*If **miaows** are **intermittent** and **size** is **small**, then **amount** id **a_bit***

We have two MFs as inputs, **intermittent & small**

Step1. Apply Fuzzy Operator

[1] https://www.researchgate.net/figure/Triangular-membership-functions-representation_fig1_265798833#:~:text=Triangular%20membership%20functions%20feature%2C%20which,are%20obtained%20from%20experts'%20knowledge.

This could be AND or OR operation. Further AND operation can be either minimum or product, and OR operation can be either maximum or probability sum, in Tnorm or Tconorm operators. By applying AND operation minimum in the above rule, the resulted value would be

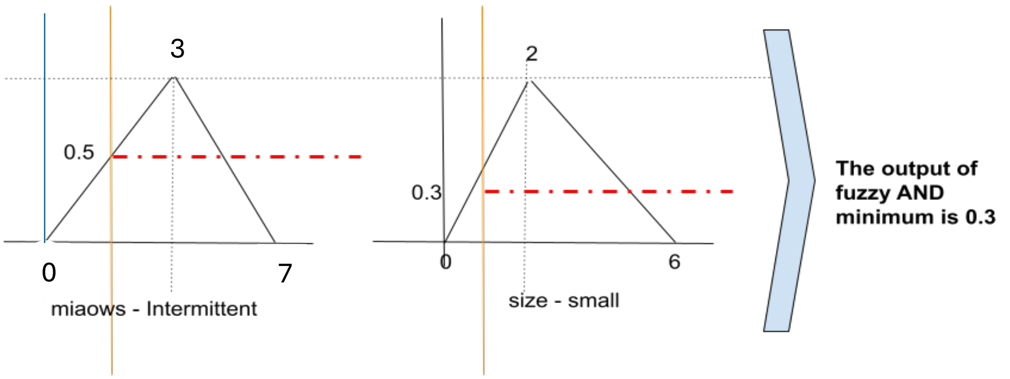


Figure 2

as given in Figure 2. Here values of each MF are 0.5 and 0.3 respectively, then the result is 0.3

Step 2. Apply Implication Operator

This is the step, which is followed by step1, the result of fuzzified input is used to apply Mamdani implication, which can be AND or prod. AND means minimum, which truncates the output MF at the level which is resulted in fuzzy operator.

So the resulted graph would be as given in Figure 3.

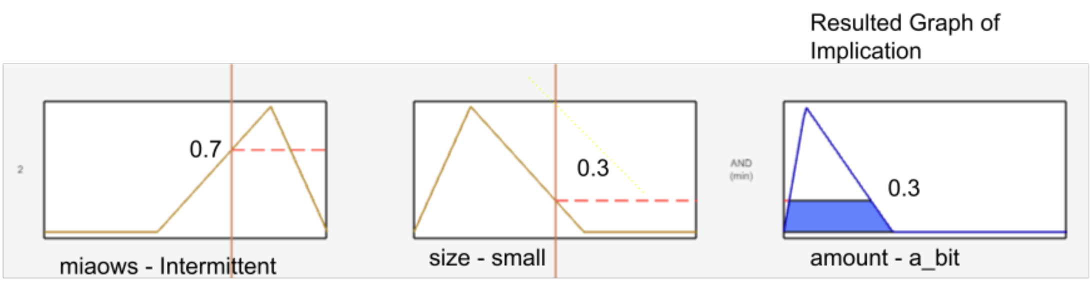


Figure 3

For each rules the implication graph would be different. The main purpose of this step is to find the rule strength of each which need to be added in the next phase of fuzzy interference system.

Step 3: Rule Aggregation

This step involves aggregation of all the outputs which has been received after the evaluation of each rule. To do this, maximum operator can be used. This output would also be in fuzzy. Based on the FIS system implemented using 'Rule Interference' in FIS Toolbox of MATLAB, a sample of the resulted graph has been given below in Figure 4.

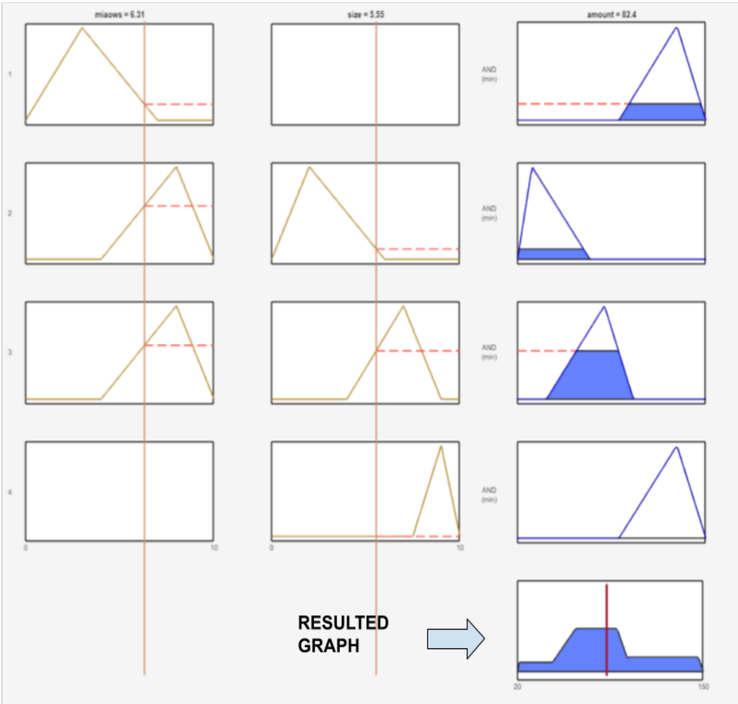


Figure 4

Step 4: Defuzzification

The output we received in step 3, is in fuzzy, which cannot be an appropriate solution, so to make it into crisp solution, defuzzification process helps. That is Defuzzification does the conversion of the aggregated output into a number. In this example, the amount of food would be the output.

There are several methods available, such as Centre of gravity, Centre of sum, etc.

For inputs miaows = 6 and size = 5

1. Example 1. While applying centre of gravity to the aggregated function, the amount resulted is 83.1, which is given in figure 5. In figure 5, the influence of rest of the rules

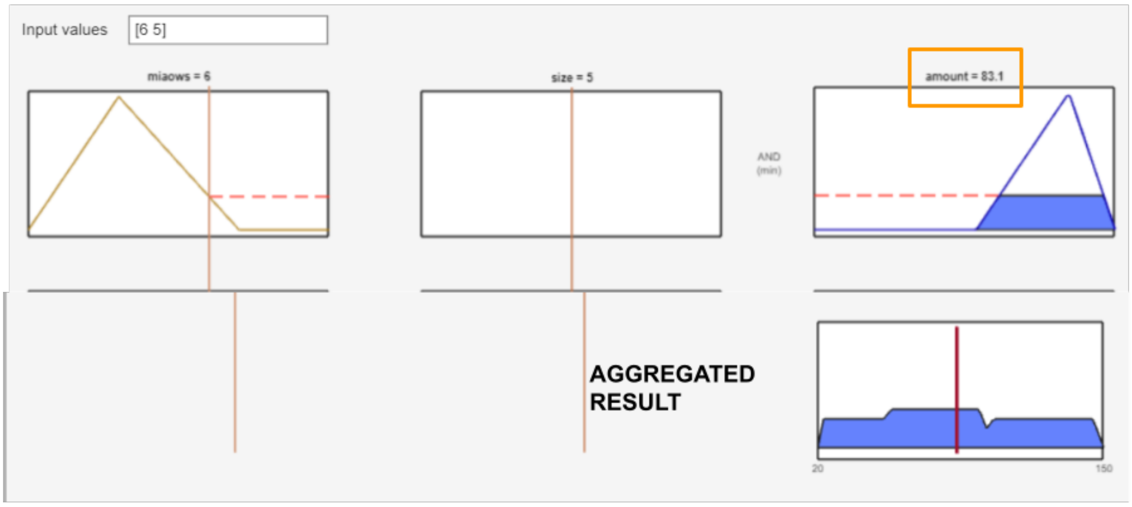


Figure 5

has been hidden in the figure to make an understanding.

2. Example 2. While applying **middle of maximum** to the aggregated function, the amount resulted is 73.3, which is given in figure 6.

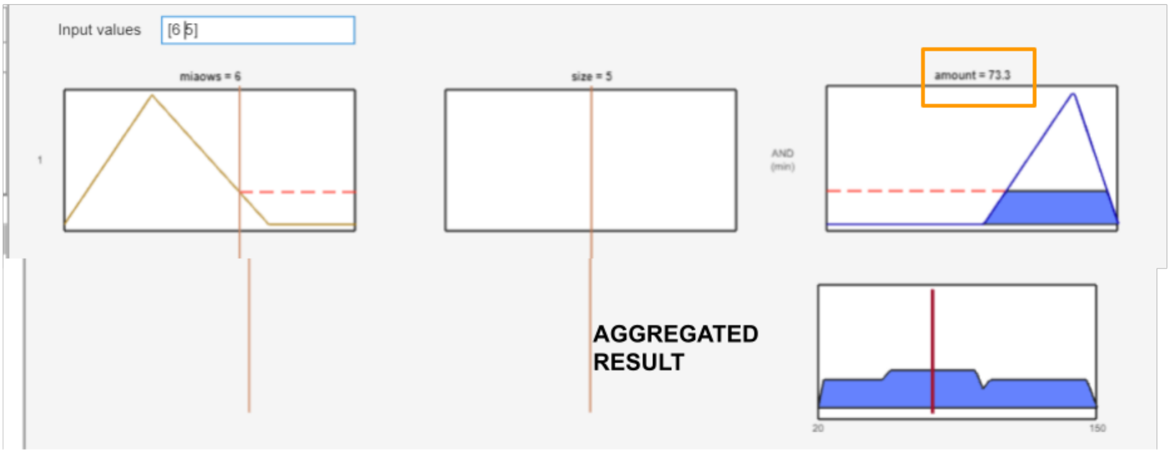


Figure 6

With different difuzzification techniques, the result would be different.

Question 2

There are certain issues or can be seen as challenges while implementing a FIS system.

- **Limits in Domain Specification**
  - In real world problems, the number of parameters that influence the system would be very huge, which increases the complexity of the FIS system.
  - While making a solution for a problem within a certain bounded time, can cause the elimination of parameters which might have a significant influence in the uncertainty of the solution.

- **Accuracy**
    - In terms of accuracy, FIS system gives a poor performance, for those systems where the output cannot be predicted well. If the rules are not well defined with proper bound of inputs, then the accuracy might fall into least.
- **Complexity of Rules**
    - Complexity exist while defining rules is something which is considered as a major challenge.
    - The reason is this.
        - If inputs increases the number of rules needed to be examined will also increase.
        - If number of linguistic variable increases the number of rules needed to be examined will also increase.
        - For example,
            - for 3 inputs with 3 linguistic terms, total 27 rules can be there for examination.
            - for 4 inputs with 4 linguistic terms, total 64 rules can be there for examination.
    - So, time consumption will also increase as complexity increases.
- **Selection of Membership Function**
    - Selection of membership function is critical while evaluating the rules. The criteria is flexible, or no fixed definition exist in such a way that the accuracy can be increased. That means that more "try and error approach by tuning the parameters with the help of pattern observations from the optimized problems" is favourable while doing the evaluation.
- **Rule Evaluation**
    - The use of operators also create uncertainty since the techniques used for finding the consequences of the rules as well as the defuzzification methods resulted into different outputs for various combinations.
    - So, optimum result cannot be predicted even with tuning until and unless we know which is the optimum result.

To conclude, FIS is useful for those systems where uncertainty and fuzziness largely exist, where the result bound to be determined is mantatory.

Exercise 17

Based on General Definition of Fuzzy Decision-making model of Bellman and Zadeh[2], Assume we have n goals, such as G_1, G_2 ... G_n, and m constraints such as C_1, C_2... C_m. Then the decision can be formulated as the intersection of n goals and m constraints.

Decision $D = G_1 \cap G_2 \cap \ldots \cap G_n \cap C_1 \cap C_2 \cap \ldots \cap C_m$

In the same way for membership values

$\mu_D = \mu_{G1} \wedge \mu_{G2} \wedge \ldots \wedge \mu_{Gn} \wedge \mu_{C1} \wedge \mu_{C2} \wedge \ldots \wedge \mu_{Cm}$ ---------------------**(1)**

[2] Bellman, R.E. and Zadeh, L.A. (1970). Decision-Making in a Fuzzy Environment. Management Science, 17(4), p.B-141-B-164. doi:https://doi.org/10.1287/mnsc.17.4.b141.

The given fuzzy sets are

Constraint 1: Interest value in each job

C1 = {(0.4, a), (0.6, b), (0.8, c), (0.6, d)}.

Constraint 2: Driving distance to each job

C2 = {(0.1, a), (0.9, b), (0.7, c), (1, d)}.

Fussy set of Goal G, from the given Membership function definition,

For different salaries,

x = f(a) = 30,000 = > μ_G(30,000) = 1- 0.00125*((30000/1000)-40)^2 = 0.875

x = f(b) = 25,000 = > μ_G(25,000) = 1- 0.00125*((25000/1000)-40)^2 = 0.719

x = f(c) = 20,000 = > μ_G(20,000) = 1- 0.00125*((20000/1000)-40)^2 = 0.5

x = f(d) = 15,000 = > μ_G(15,000) = 1- 0.00125*((15000/1000)-40)^2 = 0.21875

So G, the high salary can be defined as

G = {(0.875, a), (0.71875, b), (0.5, c), (0.21875, d)}.

From C1, C2, and G, set of membership function of each can be derived as,

μ_{C1} = {0.4,0.6,0.8,0.6}

μ_{C2} = {0.1,0.9,0.7,1}

μ_G = {0.875,0.719, 0.5, 0.219}

With the above, decision can be calculated using equation (1)

$\mu_D = \mu_{G} \wedge \mu_{C1} \wedge \mu_{C2}$ = {min{0.4,0.1,0.875}, min{0.6,0.9,0.719}, min{0.8,0.7,0.5}, min{0.6,1,0.219}}

= {0.1,0.6,0.5,0.219}

From the above $\mu_{D,}$

D = {(0.1, a), (0.6, b), (0.5, c), (0.219, d)}

Since no job has full membership degree since the constraints and goal is conflicting with one another.

Based on Bellman and Zadeh decision making model,

Optimum solution is of x with maximum membership degree, so from D,

The best job is “b”, with a membership value 0.6

Exercise 19

Given, the secondary member functions of x in type 2 fuzzy set $\bar{A}$ and $\bar{B}$ are,

$$\mu_{\bar{A}(x)} = 0.8/0.5 + 0.7/0.3 + 0.2/0.8$$

$$\mu_{\bar{B}(x)} = 0.6/0.2 + 0.4/0.3 + 0.2/0.6$$

$\underline{\bar{A} \sqcup \bar{B}}$; Join Operation

Join: $$\mu_{\tilde{A}(x)} \sqcup \mu_{\tilde{B}(x)}(u) = \sup_{\substack{u=\max\{v,w\} \\ v,w \in X}} \min\{\mu_{\tilde{A}(x)}(v), \mu_{\tilde{B}(x)}(w)\}.$$

$$\mu_{\bar{A}(x)} \sqcup \mu_{\bar{B}(x)}(u) = \left(0.8/0.5 + 0.7/0.3 + 0.2/0.8\right) \sqcup \left(0.6/0.2 + 0.4/0.3 + 0.2/0.6\right)$$

Apply join operation on all combinations,

$$= \frac{\min(0.8, 0.6)}{0.5 \vee 0.2} + \frac{\min(0.8, 0.4)}{0.5 \vee 0.3} + \frac{\min(0.8, 0.2)}{0.5 \vee 0.6} + \frac{\min(0.7, 0.6)}{0.3 \vee 0.2} + \frac{\min(0.7, 0.4)}{0.3 \vee 0.3} + \frac{\min(0.7, 0.2)}{0.3 \vee 0.6}$$

$$+ \frac{\min(0.2, 0.6)}{0.8 \vee 0.2} + \frac{\min(0.2, 0.4)}{0.8 \vee 0.3} + \frac{\min(0.2, 0.2)}{0.8 \vee 0.6}$$

$$= 0.6/0.5 + 0.4/0.5 + 0.2/0.6 + 0.6/0.3 + 0.4/0.3 + 0.2/0.6 + 0.2/0.8 + \frac{0.2}{0.8} + 0.2/0.8$$

Apply maximum operator, over the membership values of same element

$$= \frac{\max\{0.6, 0.4\}}{0.5} + \frac{\max\{0.2, 0.2\}}{0.6} + \frac{\max\{0.6, 0.4\}}{0.3} + \frac{\max\{0.2, 0.2, 0.2\}}{0.8}$$

$$= 0.6/0.3 + 0.6/0.5 + 0.2/0.6 + 0.2/0.8$$

$\bar{A} \sqcap \bar{B}$: Meet Operation:

The formula is as given below

$$\text{Meet:} \quad \mu_{\tilde{A}(x)} \sqcap \mu_{\tilde{B}(x)}(u) = \sup_{\substack{u=\min\{v,w\} \\ v,w \in X}} \min\{\mu_{\tilde{A}(x)}(v), \mu_{\tilde{B}(x)}(w)\}.$$

$$\mu_{\bar{A}(x)} \sqcap \mu_{\bar{B}(x)}(u) = \left(0.8/0.5 + 0.7/0.3 + 0.2/0.8 \right) \sqcap \left(0.6/0.2 + 0.4/0.3 + 0.2/0.6 \right)$$

$$= \frac{\min(0.8, 0.6)}{0.5 \wedge 0.2} + \frac{\min(0.8, 0.4)}{0.5 \wedge 0.3} + \frac{\min(0.8, 0.2)}{0.5 \wedge 0.6} + \frac{\min(0.7, 0.6)}{0.3 \wedge 0.2} + \frac{\min(0.7, 0.4)}{0.3 \wedge 0.3} + \frac{\min(0.7, 0.2)}{0.3 \wedge 0.6}$$

$$+ \frac{\min(0.2, 0.6)}{0.8 \wedge 0.2} + \frac{\min(0.2, 0.4)}{0.8 \wedge 0.3} + \frac{\min(0.2, 0.2)}{0.8 \wedge 0.6}$$

$$= 0.6/0.2 + 0.4/0.3 + 0.2/0.5 + 0.6/0.2 + 0.4/0.3 + 0.2/0.3 + 0.2/0.2 + \frac{0.2}{0.3} + 0.2/0.6$$

Apply maximum operator over different membership function values of same element

$$= \frac{\max\{0.6, 0.6, 0.2\}}{0.2} + \frac{\max\{0.4, 0.4, 0.2, 0.2\}}{0.3} + \frac{0.2}{0.5} + \frac{0.2}{0.6}$$

$$= 0.6/0.2 + 0.4/0.3 + 0.2/0.5 + 0.2/0.6$$

APPENDEX A

A1: MATLAB Code for Figure 1

```
L = mamfis('Name',"catLover");

L = addInput(L,[0 10],'Name',"miaows");

L = addMF(L,"miaows","trimf",[0,3,7],'Name',"intermittent");
L = addMF(L,"miaows","trimf",[4,8,10],'Name',"persistant");

L = addInput(L,[0 10],'Name',"size");

L = addMF(L,"size","trimf",[0,2,6],'Name',"small");
L = addMF(L,"size","trimf",[4 7 9],'Name',"medium");
L = addMF(L,"size","trimf",[7.5,9,10],'Name',"large");

L = addOutput(L, [20,150],'Name',"amount") %g

L = addMF(L,"amount","trimf",[20,30,70],'Name',"a_bit");
L = addMF(L,"amount","trimf",[40,80,100],'Name',"average");
L = addMF(L,"amount","trimf",[90 130 150],'Name',"a_lot");

subplot(3,1,1), plotmf(L,'input',1)
subplot(3,1,2), plotmf(L,'input',2)
subplot(3,1,3), plotmf(L,'output',1)
```

A2: '.fis' file used for implementation of FIS system with FIS Toolbox

carLover2.fis

find the attachement with this pdf