Implementation Details

Fuzzy logic is simply a way to deal with uncertainties. Rather than true or false, fuzzy logic deals with the degree of truth or false. For example, in Boolean terms a person can be said to be honest (or 1) or dishonest (or 0), whereas fuzzy logic considers degree of honesty or dishonesty to take further action. This logic has various applications such as controlling the lighting of a room through light sensor.

For completion of the coursework a fuzzy logic based controller was designed to simulate an intelligent flat for assisting disabled users. Mat Lab was installed and used to design and simulate the controller and Mamdani inference model was used. Following figure depicts Mat Lab installed in local computer and C:\Users\kccra\OneDrive\Documents\MATLAB set as home directory.

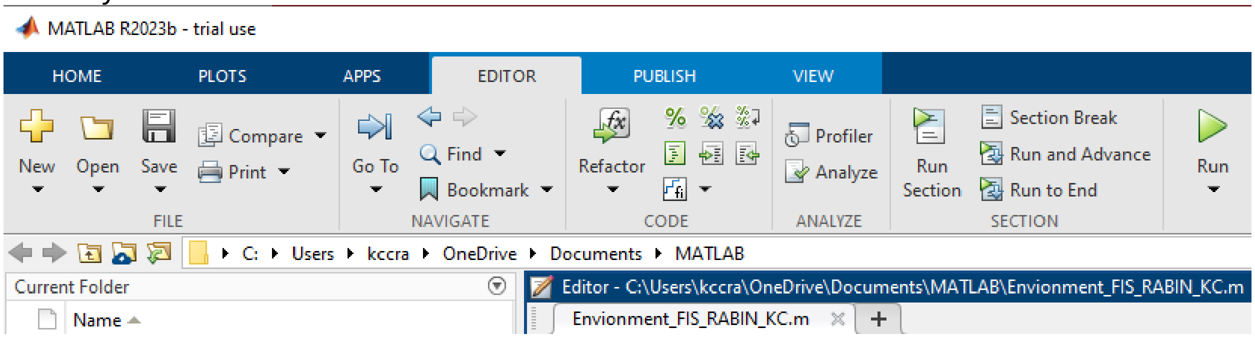


Figure: Mat Lab Home directory in local computer

To develop this intelligent assistive care six (6) parameters were considered as an input. These six (6) parameters include humidity, temperature, light level, user activity, time of the day, and presence of smoke. Following figure shows the creation and selection of Mamdani inference model and parameters considered as input:

```
% creating Mamdani Fuzzy inference System
intAstCareEnvFis = mamfis('Name','IntelligentAsstCare');

% adding input variables
intAstCareEnvFis = addInput(intAstCareEnvFis, [0 10], 'Name', 'humidityLevel', 'NumMFs', 4);
intAstCareEnvFis = addInput(intAstCareEnvFis, [0 48], 'Name', 'temperature', 'NumMFs', 3);
intAstCareEnvFis = addInput(intAstCareEnvFis, [0 10], 'Name', 'lightIntensity', 'NumMFs', 3);
intAstCareEnvFis = addInput(intAstCareEnvFis, [0 1], 'Name', 'activity', 'NumMFs', 2);
intAstCareEnvFis = addInput(intAstCareEnvFis, [0 24], 'Name', 'time', 'NumMFs', 4);
intAstCareEnvFis = addInput(intAstCareEnvFis, [0 1], 'Name', 'smokePresence', 'NumMFs', 2);
```

Figure: selection of model and input parameters

Following figures depicts the inputs graphically

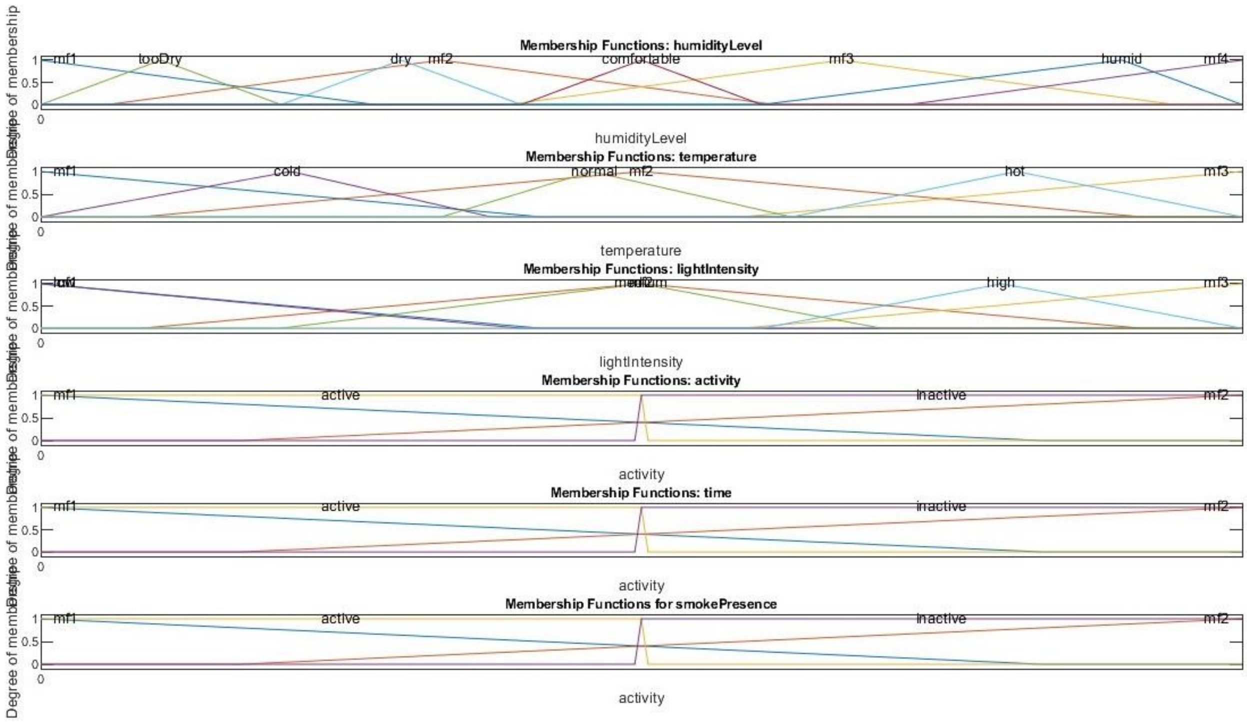


Figure: Input variables, membership functions

Here, triangular membership functions were used for humidity level and temperature, whereas light intensity was Gaussian. Details of input variables, membership functions and membership values are tabulated below:

INPUT PARAMETER	MEMBER FUNCTION	MEMBERSHIP FUNCTION VALUES
HUMIDITY LEVEL	Triangular	Too dry, Dry, Comfortable and Humid
TEMPERATURE	Triangular	Cold, Normal, and Hot
LIGHT INTENSITY	Triangular	Low, Medium, and High
USER ACTIVITY	Trapezoidal	Inactive and Active
TIME OF DAY	Gaussian	Morning, Day Evening and Night
SMOKE	Trapezoidal	Present, and Absent

After input parameters, three (3) output parameters were selected. These three (3) output parameters include door, light switch and heater. Following figure shows selection of output parameters:

```
% adding output variables
intAstCareEnvFis = addOutput(intAstCareEnvFis, [0 10], 'Name', 'door', 'NumMFs', 2);
intAstCareEnvFis = addOutput(intAstCareEnvFis, [0 10], 'Name', 'lightSwitch', 'NumMFs', 2);
intAstCareEnvFis = addOutput(intAstCareEnvFis, [0 10], 'Name', 'heater', 'NumMFs', 3);
```

Figure: output variables

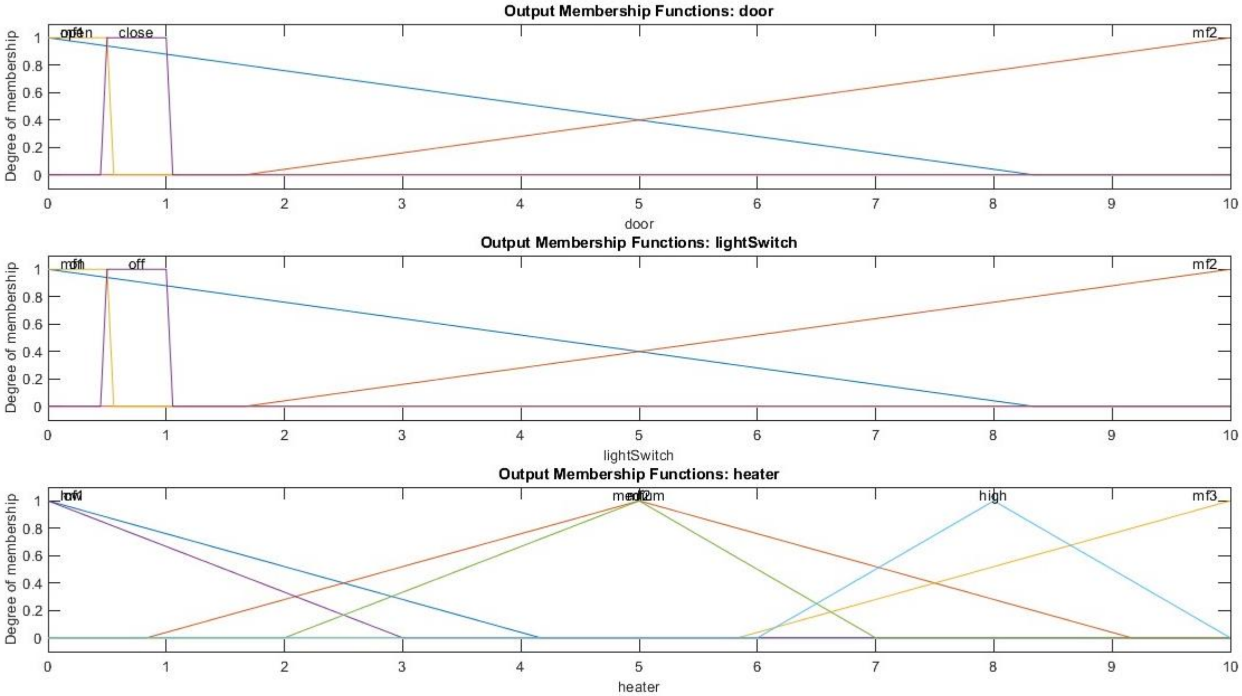


Figure: graphical representation of output variables and membership functions

Details of output variables, membership functions and membership values are tabulated below:

OUTPUT PARAMETER	MEMBER FUNCTION	MEMBERSHIP FUNCTION VALUES
DOOR	Trapezoidal	Open and Close
LIGHT SWITCH	Triangular	On and Off
HEATER	Trapezoidal	Low, Medium and high

After creation of input and output variables and membership functions fuzzy rules were specified and added to the model. Some of the rules are shown below:

'1. If (humidityLevel is comfortable) and (temperature is normal) then (door is close) (1) '
'2. If (humidityLevel is comfortable) and (temperature is normal) then (heater is medium) (1)'
'3. If (temperature is hot) then (door is open) (1) '
'4. If (temperature is hot) then (heater is low) (1) '
'5. If (activity is active) and (time is day) then (door is open) (1) '
'6. If (activity is active) and (time is night) then (door is close) (1) '
'7. If (activity is active) and (time is day) then (lightSwitch is off) (1) '
'8. If (activity is active) and (time is day) then (lightSwitch is off) (1) '
'9. If (lightIntensity is low) and (activity is active) then (lightSwitch is on) (1) '
'10. If (smokePresence is present) then (door is open) (1) '
'11. If (lightIntensity is low) and (time is evening) then (lightSwitch is on) (1) '
'12. If (lightIntensity is high) and (time is day) then (lightSwitch is off) (1) '

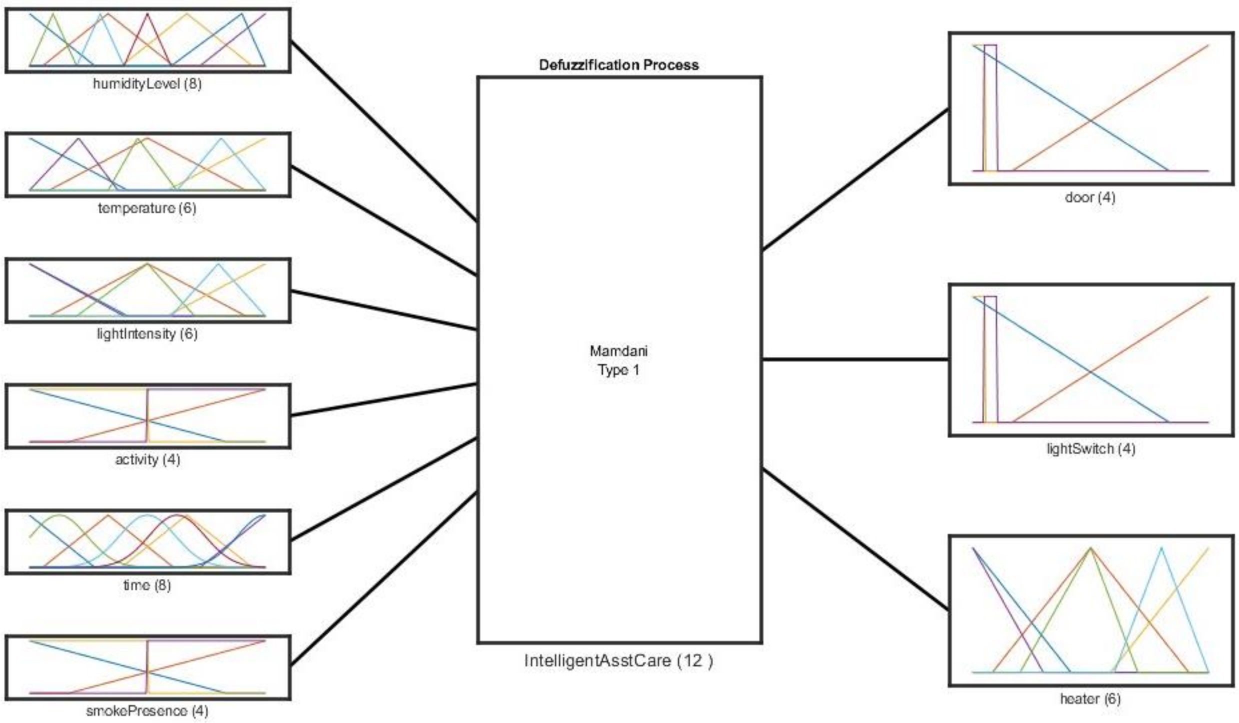
Now, the input variable set was fed into the model and evaluated to obtain the model as shown below:

```
%setting input values
environmentalInputs = [5, 22, 4, 1, 12, 1];

% evaluating output
environmentalOutputs = evalfis(intAstCareEnvFis,environmentalInputs);

%visualizing output results
disp(environmentalOutputs);
```

Command Window

```
    0.5000    5.0000    4.6667
```

Finally, defuzzification process was performed and output were obtained, as depicted from figure below:



Justification

Discuss and justify your design decisions for the choice of fuzzy sets - membership functions, fuzzy rules, FLC inference mechanism selected, and defuzzification method that was chosen. Back up your explanations with evidence in the form of appropriate diagrams and screenshots.

Choice of membership functions

The membership functions chosen for input and outputs are depicted in following tables below

INPUT PARAMETER	MEMBER FUNCTION	MEMBERSHIP FUNCTION VALUES

HUMIDITY LEVEL	Triangular	Too dry (0 to 2), Dry(2 to 4), Comfortable(4 to 6) and Humid(6 to 10)
TEMPERATURE	Triangular	Cold (0 to 18), Normal (16 to 30), and Hot (30 to 48)
LIGHT INTENSITY	Triangular	Low (0 to 4), Medium(2 to 7), and High(6 to 10)
USER ACTIVITY	Trapezoidal	Inactive (0 0 0.5 0.5) and Active (0.5 0.5 1 1)
TIME OF DAY	Gaussian	Morning(2.831 3), Day(2.831 12), Evening(2.831 15) and Night (2.831 15)
SMOKE	Trapezoidal	Present (0 0 0.5 0.5), and Absent (0.5 0.5 1 1)

OUTPUT PARAMETER	**MEMBER FUNCTION**	**MEMBERSHIP FUNCTION VALUES**
DOOR	Trapezoidal	Open (0 0 0.5 0.5) and Close (0.5 0.5 1 1)
LIGHT SWITCH	Triangular	On (0 0 0.5 0.5) and Off (0.5 0.5 1 1)
HEATER	Trapezoidal	Low(0 to 3), Medium(2 to 7) and high(6 to 10)

As shown in table above, humidity level as measured on a scale of ten, 0 being too dry and 10 being humid. Temperature was considered on scale of zero (0) to forty eight (48) degrees. Along with that light intensity was measured on scale of ten (10). User activity was considered as trapezoidal and four input points for inactive and active were selected. Similarly time of day was measured as 24 hours in Gaussian member function with morning, evening, day. finally night and presence or absence of smoke was measured using trapezoidal function each value representing four points of trapezoid.

Along with input variables output variables such as door, switch and heater were considered to be Trapezoidal, Triangular, and Trapezoidal respectively. The power on the heater was measured as low for value between zero (0) to three (3), Medium for values between two (2) to seven (7) and high for six (6) to ten (10).

The fuzzy rules were as follows

1. If (humidityLevel is comfortable) and (temperature is normal) then (door is close) (1) '
2. If (humidityLevel is comfortable) and (temperature is normal) then (heater is medium) (1)'
3. If (temperature is hot) then (door is open) (1)
4. If (temperature is hot) then (heater is low) (1)
5. If (activity is active) and (time is day) then (door is open) (1)
6. If (activity is active) and (time is night) then (door is close) (1)
7. If (activity is active) and (time is day) then (lightSwitch is off) (1)
8. If (activity is active) and (time is day) then (lightSwitch is off) (1)
9. If (lightIntensity is low) and (activity is active) then (lightSwitch is on) (1)
10. If (smokePresence is present) then (door is open) (1)
11. If (lightIntensity is low) and (time is evening) then (lightSwitch is on) (1)
12. If (lightIntensity is high) and (time is day) then (lightSwitch is off) (1)

The rules are straightforward for instance, if there is no humidity and temperature is normal then the heater can be set to normal as shown in rule two (2) above. Similarly, if the user is active and time is day then the door needs to be opened as he needs to pass through door multiple times in his activity during day as in rule five (5).

To achieve output inference from the input variables, the input variables were initialized and evaluated. For this crisp (non fuzzy) inputs were provided, all the rules were evaluated in parallel using fuzzy reasoning, then the results were combined and selected, and finally the crisp output was selected.

For defuzzification process made use of centroid. For instance the aggregate fuzzy set of humidity level of 5, temperate of 22 degrees, light of 4 units, user activity 1(inactive) and smoke present produced a defuzzified number of 0.5 for door, which means to close the door.

Analysis

Following table shows the simple inputs provided

INPUT PARAMETER	**MEMBER FUNCTION**
HUMIDITY LEVEL	5 (comfortable)
TEMPERATURE	22 (normal)
LIGHT INTENSITY	4 (low-medium)
USER ACTIVITY	1 (inactive)
TIME OF DAY	12 (day)
SMOKE	1 (present)

The output values from the Mamdami based fuzzy controller for provided input (as shown above) were as follows:

OUTPUT PARAMETER	**MEMBER FUNCTION**
DOOR	0.5
LIGHT SWITCH	5
HEATER	4.7

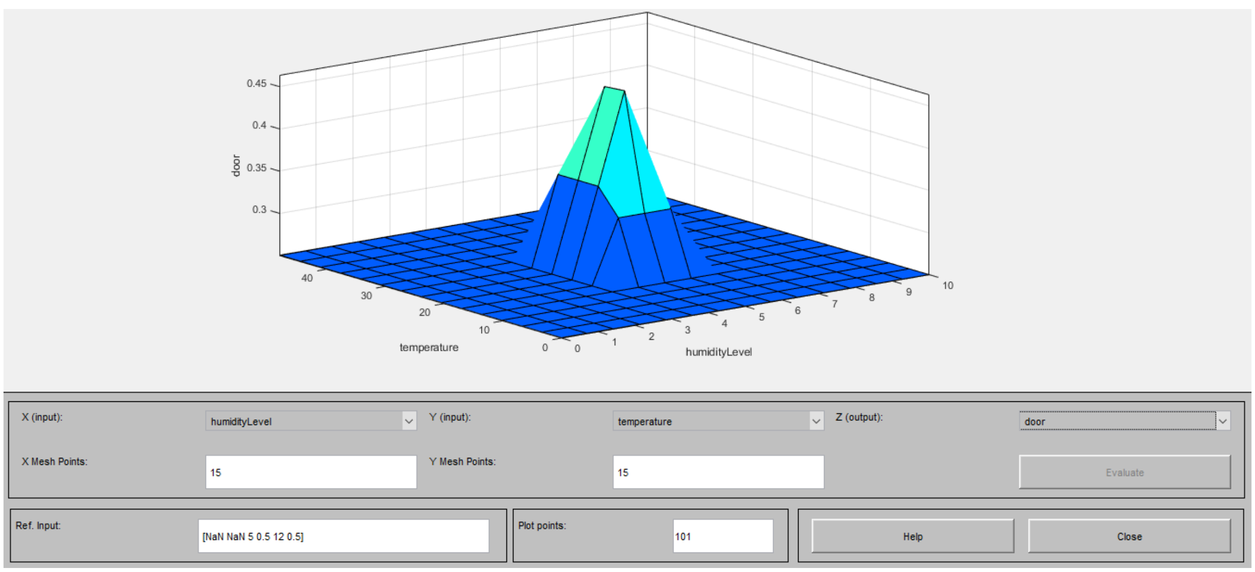
This input produced following output and triggered rules as shown in figure below:

```
%visualizing output results
disp(environmentalOutputs);
disp(['Door: ', num2str(outputValues(1))]);
disp(['LightSwitch: ', num2str(outputValues(2))]);
disp(['Heater: ', num2str(outputValues(3))]);


%visualizing triggered rules based on output membership values
[~, doorRule] = max(outputValues(1));
[~, lightSwitchRule] = max(outputValues(2));
[~, heaterRule] = max(outputValues(3));

disp(['Triggered rule for door: ', num2str(doorRule)]);
disp(['Triggered rule for light Switch : ', num2str(lightSwitchRule)]);
disp(['Triggered rule for heater : ', num2str(heaterRule)]);
```

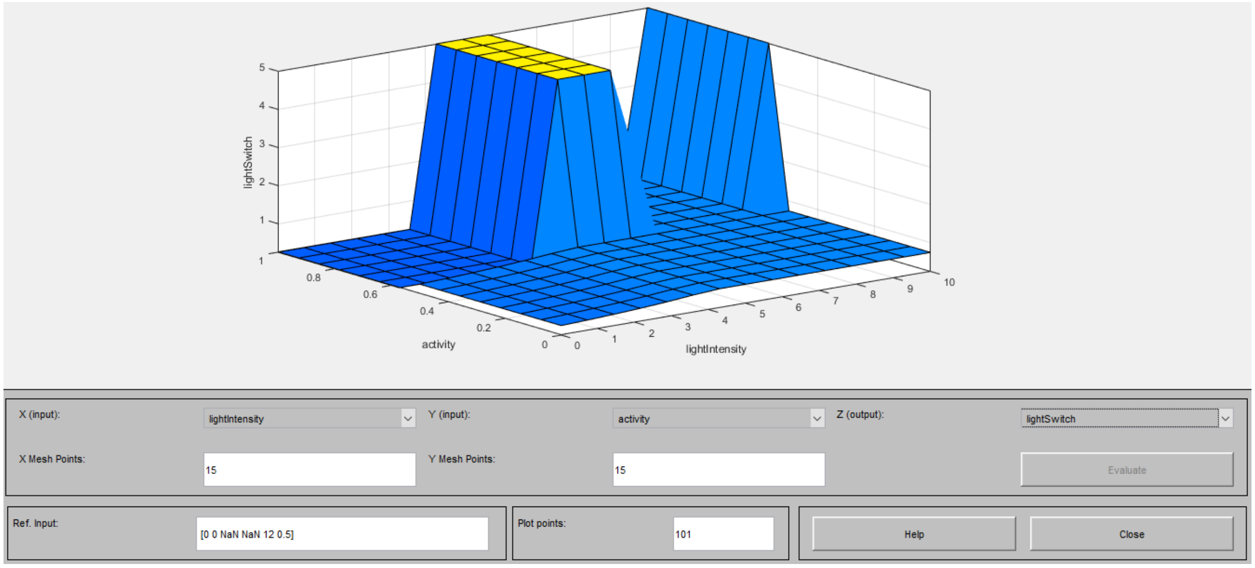
Command Window

```
    0.5000    5.0000    4.6667

Door: 87
LightSwitch: 9.6667
Heater: 50
Triggered rule for door: 1
Triggered rule for light Switch : 1
Triggered rule for heater : 1
```

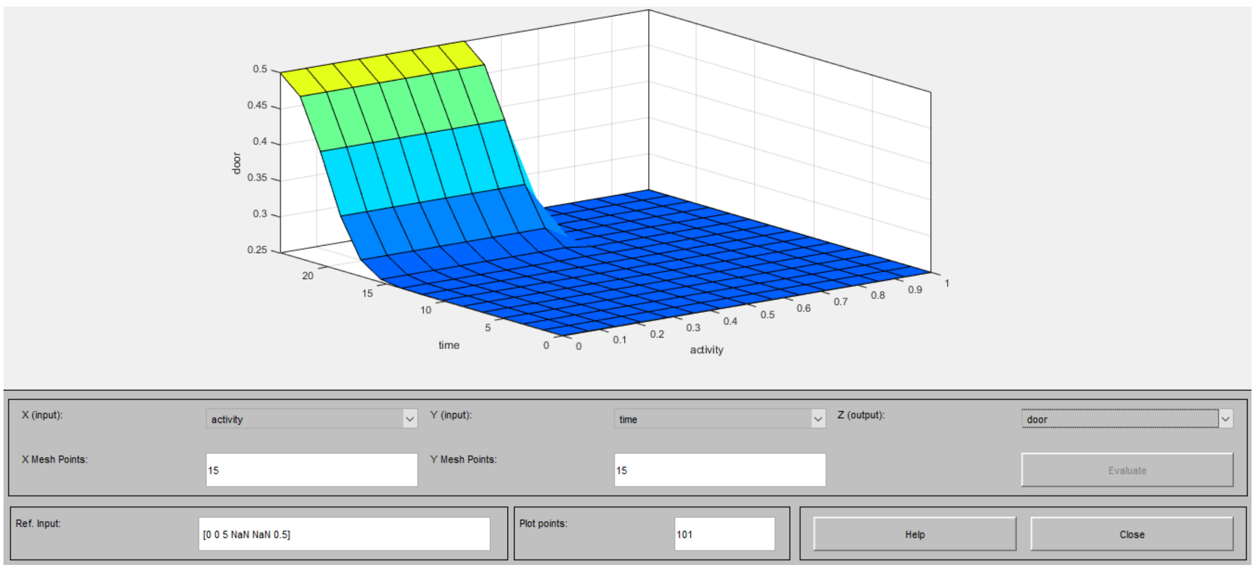
The control surface plots are depicted below:



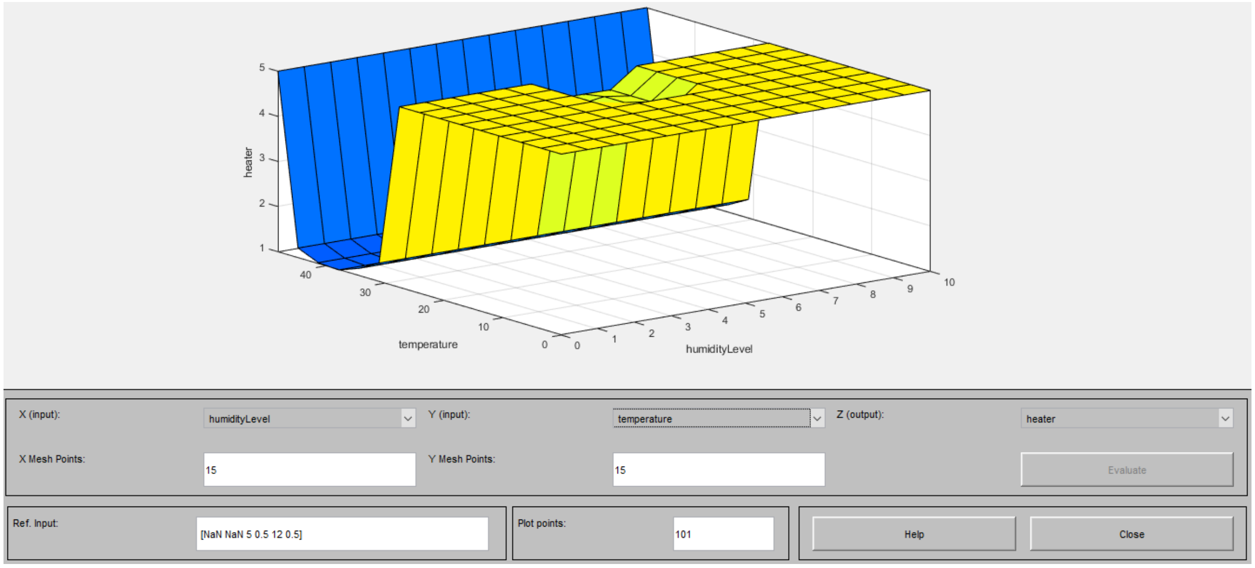
Control surface showing relation of temperature and humidity with door.



Control surface showing relationship between light intensity and activity with light switch



Control surface showing relationship between activity and time with door



Control surface showing relationship between light level and temperature with heater