

Computer Assignment 2  
Rainy Probabilty Prediction

จัดทำโดย

นายรัชพงศ์ ทอหุล 600610769

เสนอ

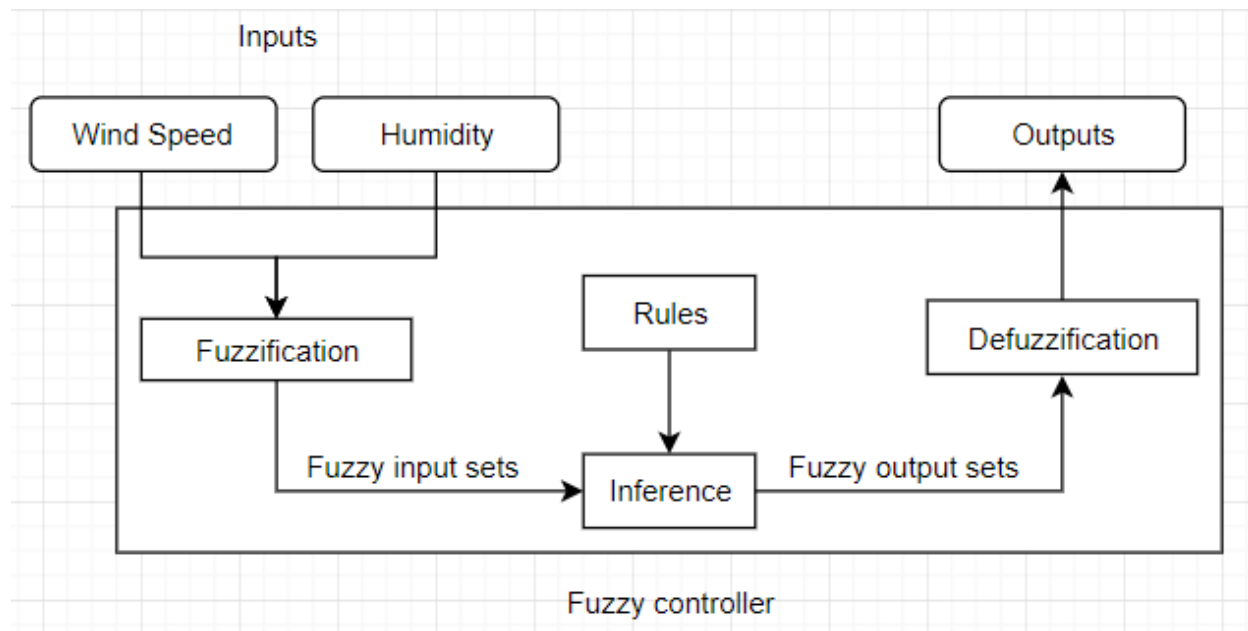
รศ.ดร.ศันสนีย์ เอื้อพันธุ์วิริยะกุล

รายงานนี้เป็นส่วนหนึ่งของวิชา  
CPE 261456 (Introduction to Computational Intelligence)  
ภาคเรียนที่ 1 ปีการศึกษา 2563  
มหาวิทยาลัยเชียงใหม่

## สารบัญ

สารบัญ	1
Method	2
1. Inputs	2
2. Fuzzification	2
2.1 Membership	3
3. Inference	4
4. Defuzzification	4
5. Outputs	5
Result	6
ภาคผนวกโปรแกรม	

## Method



รูปที่ 1 ระบบคาดเดาโอกาสฝนตกด้วยฟัซซีคอนโทรล

## การทำงานของระบบโดยรวม

## 1. Inputs

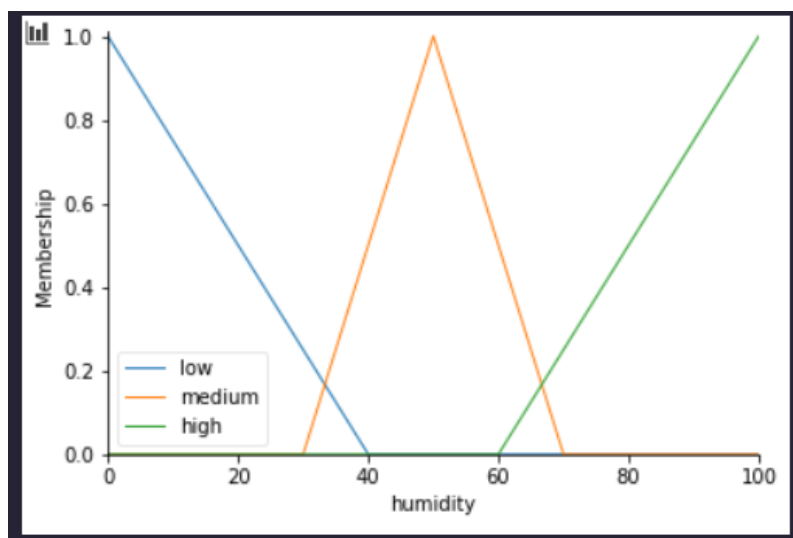
ระบบจะรับอินพุต 2 อินพุต คือ ค่าความชื้นในอากาศ และค่าความเร็วลม ซึ่งทั้งค่าความชื้นในอากาศจะมีค่าอยู่ในช่วงของ  $[0, 100]$  โดยเป็นตัวเลขแสดงค่าเปอร์เซ็นต์เช่นหากอินพุตค่าความชื้นในอากาศเป็น 50 หมายถึงมีความชื้นในอากาศ 50% นั่นเองส่วนค่าความเร็วลมจะมีค่าในช่วง  $[0, 50]$  โดยมีหน่วยเป็น กิโลเมตรต่อชั่วโมง

## 2. Fuzzification

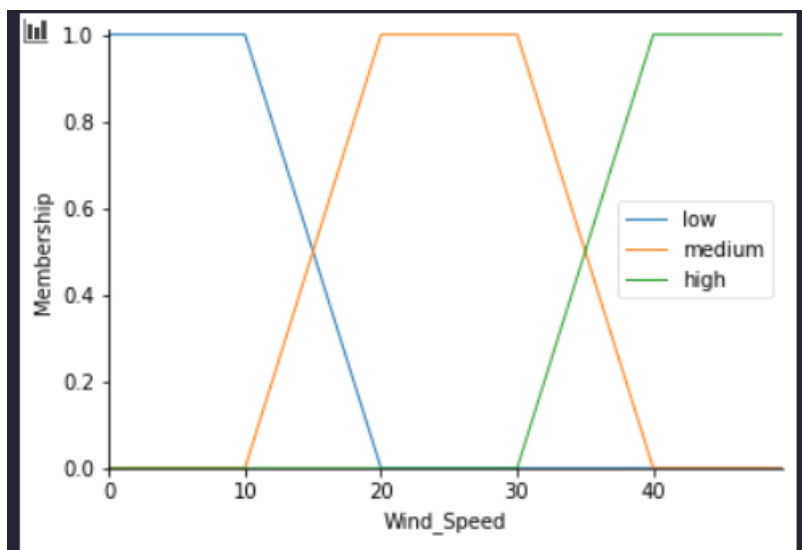
กระบวนการนี้จะนำค่าอินพุตที่ได้จากข้อ 1 มาแปลงค่าจากหน่วยความจริงให้กลายเป็นหน่วยของฟัซซี ซึ่งก็คือค่าความเป็นสมาชิก

## 2.1 Membership

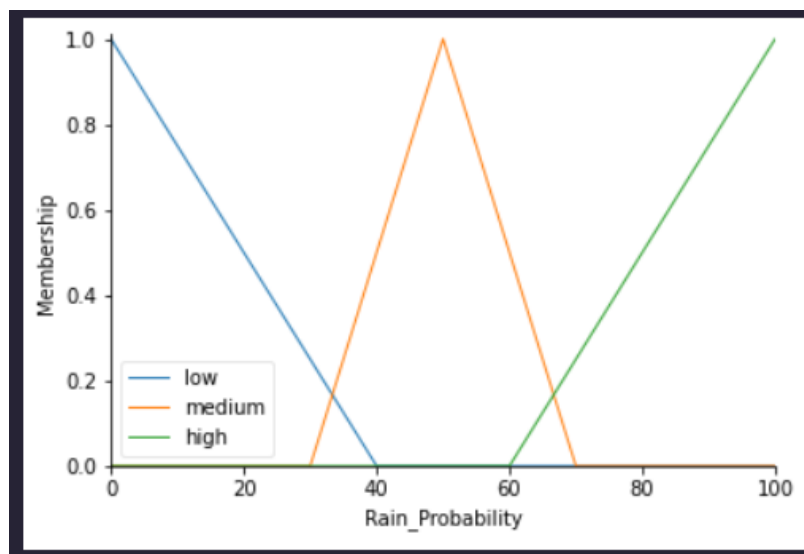
กำหนดค่าความเป็นสมาชิกของอินพุตที่ 1 ใช้ฟังก์ชันสามเหลี่ยม ,อินพุตที่ 2 ใช้ฟังก์ชันสี่เหลี่ยมคางหมู และเอาต์พุตใช้ฟังก์ชันสามเหลี่ยม ในการกำหนดค่าความเป็นสมาชิกดังรูปต่อไปนี้



รูปที่ 2 ค่าความเป็นสมาชิกของอินพุตความชื้นในอากาศ



รูปที่ 3 ค่าความเป็นสมาชิกของอินพุตความเร็วลม



รูปที่ 4 ค่าความเป็นสมาชิกของเอาต์พุตโอกาสเกิดฝนตก

### 3. Inference

จะเป็นกระบวนการอนุมานผลลัพธ์จากกฎที่มีอยู่ใน Fuzzy Controller โดยวิธี Mamdani และ ประมวลผลกับฟัซซีอินพุต โดยวิธี Max-min composition ซึ่งกฎมีดังนี้

Rain_Probability	Humidity['low']	Humidity['medium']	Humidity['high']
Wind_Speed['low']	low	low	medium
Wind_Speed['medium']	low	medium	high
Wind_Speed['high']	medium	high	high

โดยเอาต์พุตจะมีค่าอยู่ในช่วง 0% ถึง 100% ค่า low medium high เป็นดังนี้

Low คือ โอกาสฝนตก 0%

Medium คือ โอกาสฝนตก 50%

High คือ โอกาสฝนตก 100%

### 4. Defuzzification

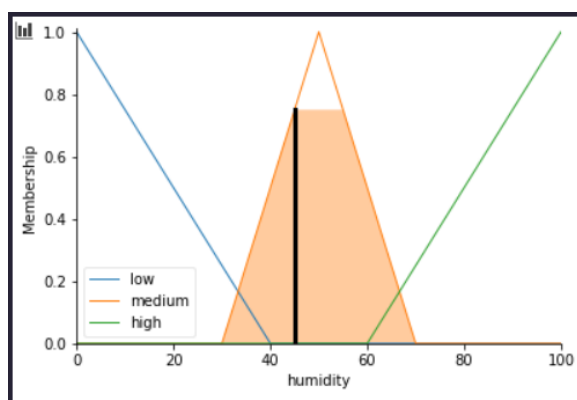
เป็นการแปลงจากค่าเอาต์พุตที่เป็นฟัซซีให้เป็นค่าปกติโดยเลือกค่าสูงสุด หรือสรุปหาเหตุผลจากหลายๆ เซตมาเพียงค่าเดียว

## 5. Outputs

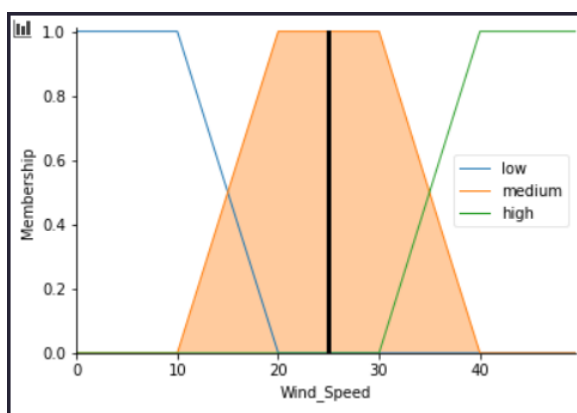
เอาต์พุตที่ได้ออกมาก็จะเป็นโอกาสที่จะเกิดฝนตกเป็นเปอร์เซ็นต์นั่นเอง

## Result

เริ่มจากการทดลองใส่ค่าอินพุตเข้าไปในระบบก่อนโดยใส่ค่าอินพุต ความชื้นในอากาศเป็น 45% และค่าความเร็วลมเป็น 25 กิโลเมตรต่อชั่วโมงจากนั้นระบบฟัซซีฟิเคชันจะแปลงค่าความจริงเป็นค่าฟัซซีโดยความชื้นในอากาศจะแปลงเป็นค่าความเป็นสมาชิก 'medium' เท่ากับ 0.79 และค่าความเร็วลมจะแปลงเป็นค่าความเป็นสมาชิก 'medium' เท่ากับ 1.0 ดังรูปประกอบต่อไปนี้



รูปที่ 5 ค่าความเป็นสมาชิกของฟัซซีความชื้นที่ 45%

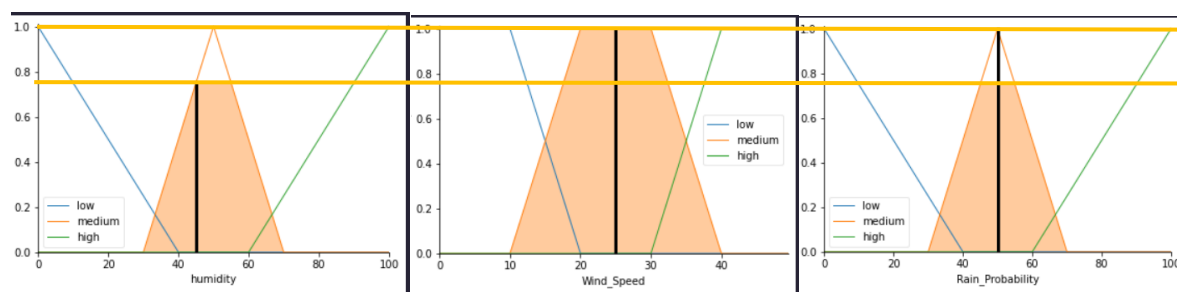


รูปที่ 6 ค่าความเป็นสมาชิกของฟัซซีความเร็วลมที่ 25 กม/ชม

จากนั้นเป็นการอนุมานผลลัพธ์จากกฎที่มีอยู่ใน Fuzzy Controller โดยวิธี Mamdani ซึ่งค่าอินพุตที่ใส่มาสอดคล้องกับกฎในตารางดังนี้

Rain_Probability	Humidity['low']	Humidity['medium']	Humidity['high']
Wind_Speed['low']	low	low	medium
Wind_Speed['medium']	low	medium	high
Wind_Speed['high']	medium	high	high

จะได้การอนุมานว่าผลลัพธ์มีโอกาสเป็น 'medium' และเมื่อทำการหาค่าความเป็นสมาชิกด้วยวิธี **max-min composition** จะได้ค่าความเป็นสมาชิก 'medium' เท่ากับ 49.99 ดังนี้



รูปที่ 7 แสดงกระบวนการหาค่าความเป็นสมาชิกของเอาต์พุต

ก็จะได้อาต์พุตมาดังนี้

```
[34] ▶ ▶ M1
      print(rainyProbability.output['Rain_Probability'])
      49.99999999999999
```

รูปที่ 8 แสดงผลลัพธ์ของเอาต์พุต



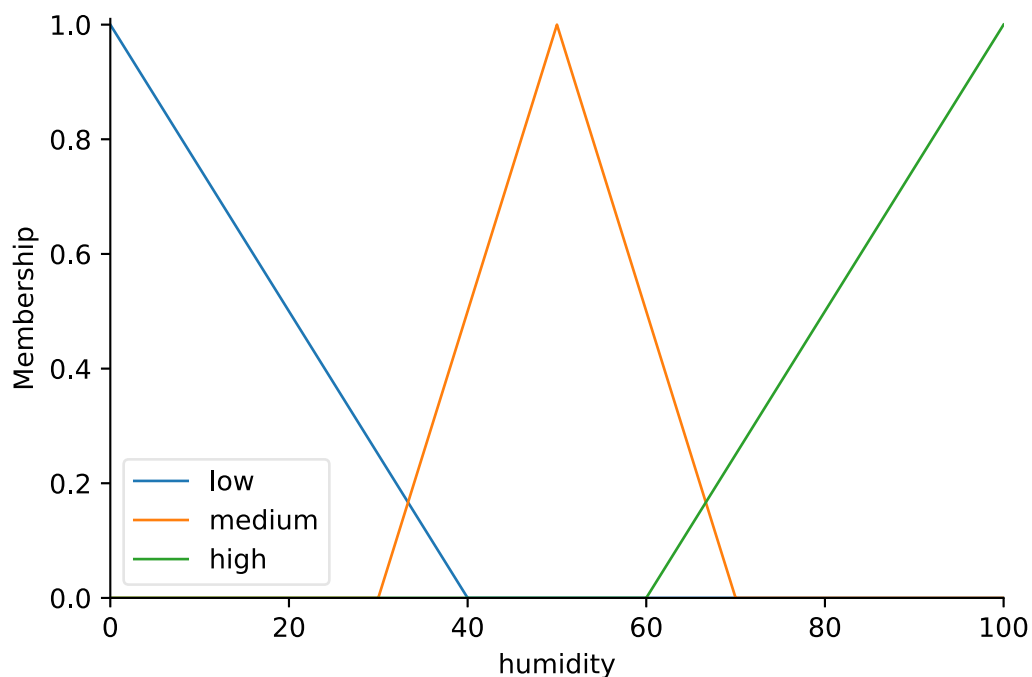
## ภาคผนวกโปรแกรม

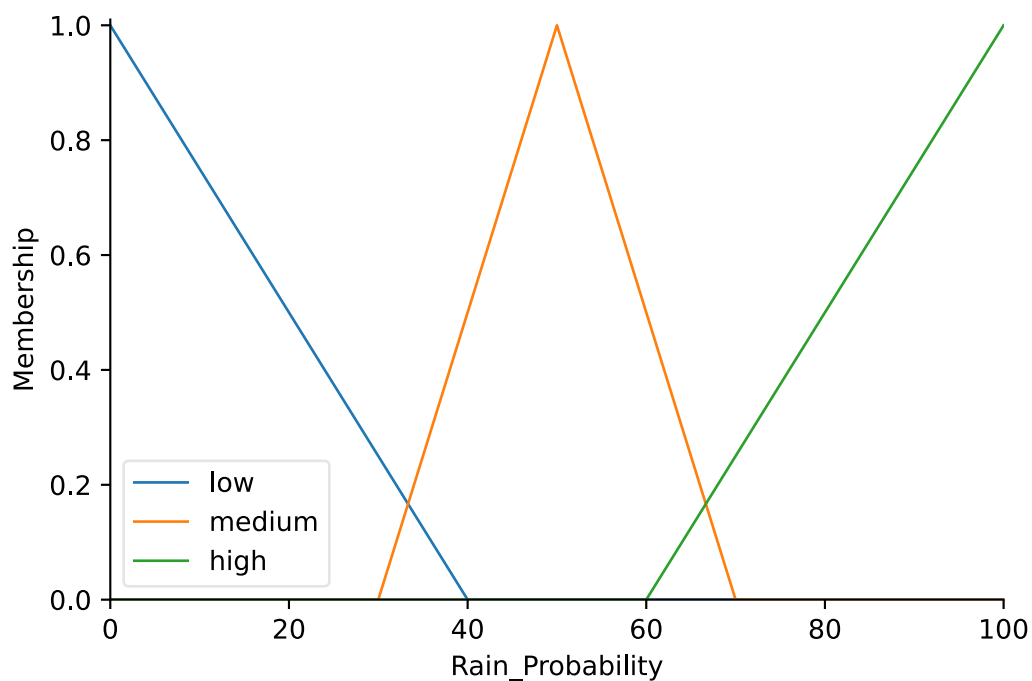
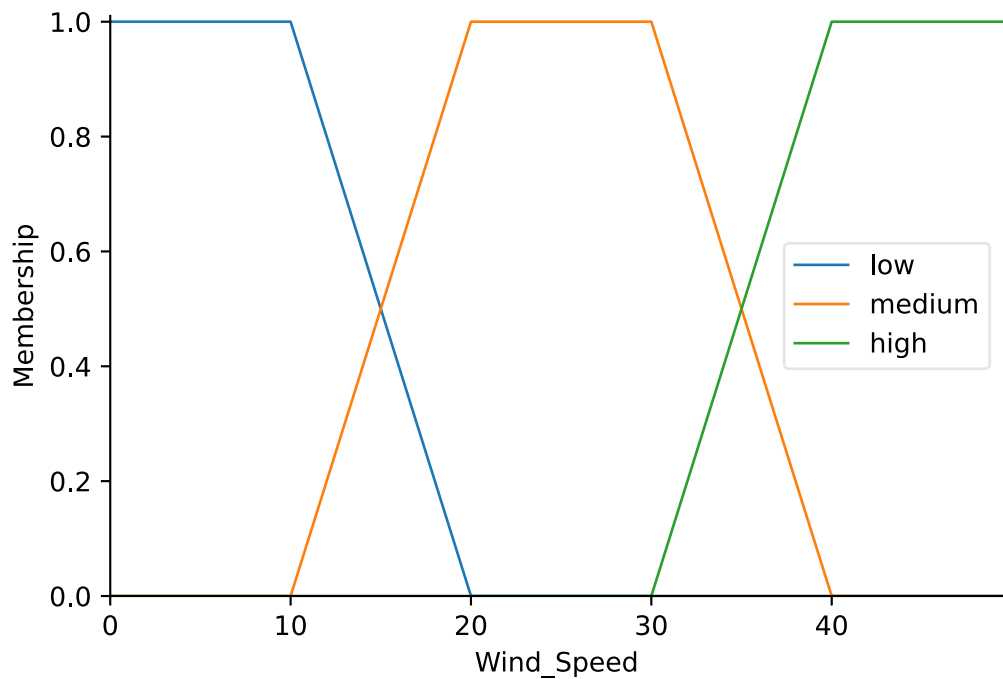
```
In [1]: # import library
import numpy as np
import matplotlib.pyplot as plt
import skfuzzy as fuzz
from skfuzzy import control as ctrl
```

```
In [2]: # define fuzzy universe
humid = ctrl.Antecedent(np.arange(0,101,1), 'humidity')
wSpeed = ctrl.Antecedent(np.arange(0,50,0.5), 'Wind_Speed')
rainy = ctrl.Consequent(np.arange(0,101,1), 'Rain_Probability')
```

```
In [3]: #define fuzzy membership function
#triangle function
humid['low'] = fuzz.trimf(humid.universe, [0,0,40])
humid['medium'] = fuzz.trimf(humid.universe, [30,50,70])
humid['high'] = fuzz.trimf(humid.universe, [60,100,100])
#trapezoid function
wSpeed['low'] = fuzz.trapmf(wSpeed.universe, [0,0,10,20])
wSpeed['medium'] = fuzz.trapmf(wSpeed.universe, [10,20,30,40])
wSpeed['high'] = fuzz.trapmf(wSpeed.universe, [30,40,50,50])
#triangle function
rainy['low'] = fuzz.trimf(rainy.universe, [0,0,40])
rainy['medium'] = fuzz.trimf(rainy.universe, [30,50,70])
rainy['high'] = fuzz.trimf(rainy.universe, [60,100,100])
```

```
In [4]: #view fuzzy membership function
humid.view()
wSpeed.view()
rainy.view()
plt.show()
```





```
In [5]: #define fuzzzi rule
rule1 = ctrl.Rule(antecedent=((humid['low'] & wSpeed['low']) |
                              (humid['low'] & wSpeed['medium']) |
                              (humid['medium'] & wSpeed['low'])),
                 consequent=rainy['low'], label='low')

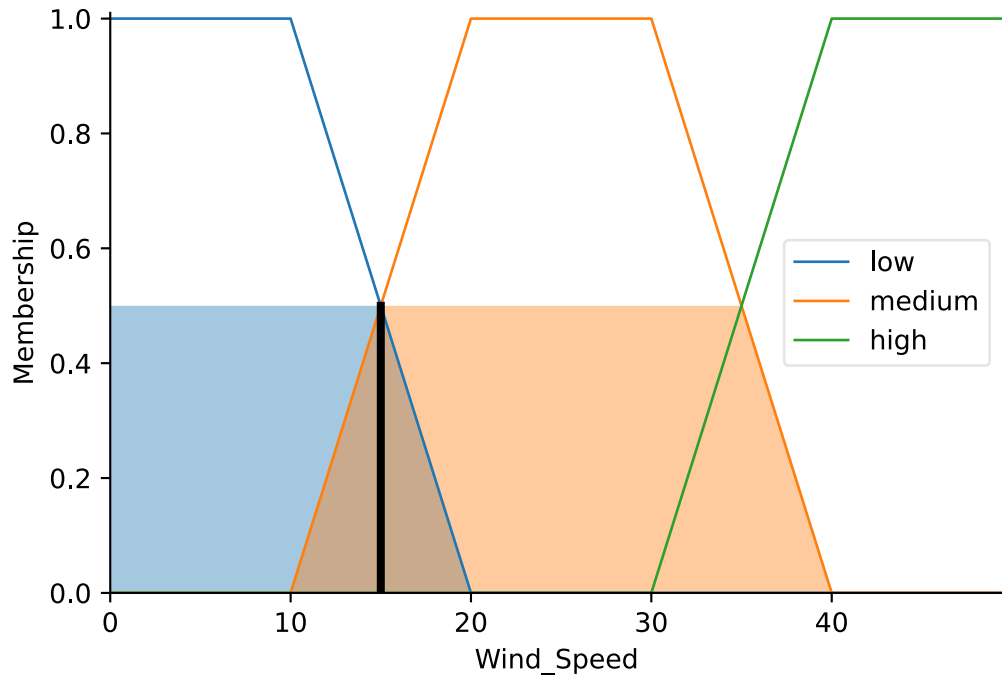
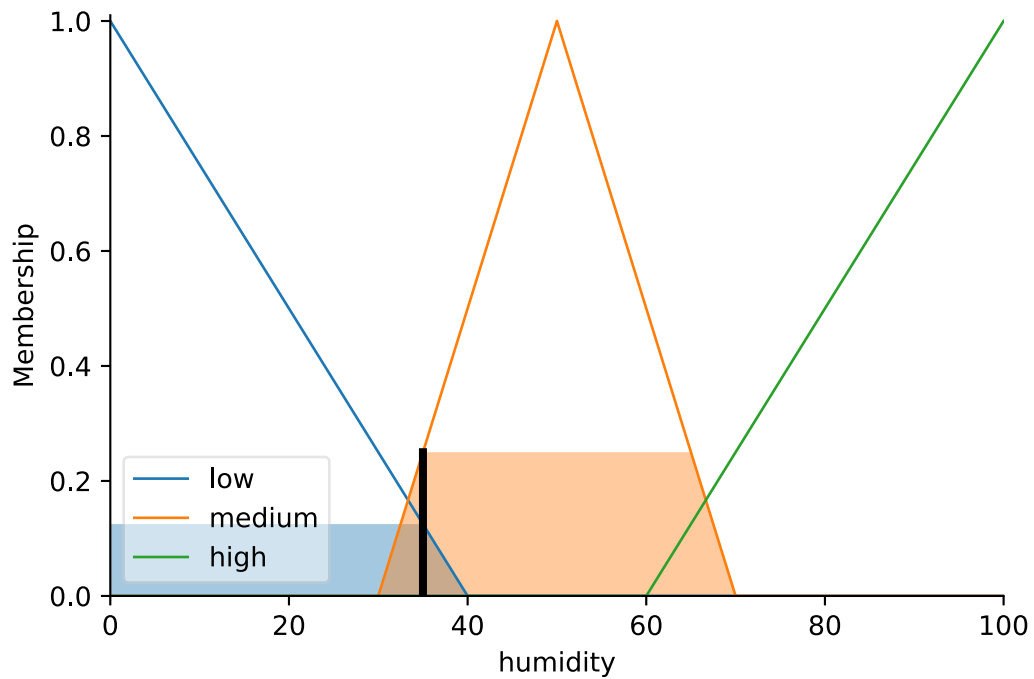
rule2 = ctrl.Rule(antecedent=((humid['low'] & wSpeed['high']) |
                              (humid['medium'] & wSpeed['medium']) |
                              (humid['high'] & wSpeed['low'])),
                 consequent=rainy['medium'], label='medium')

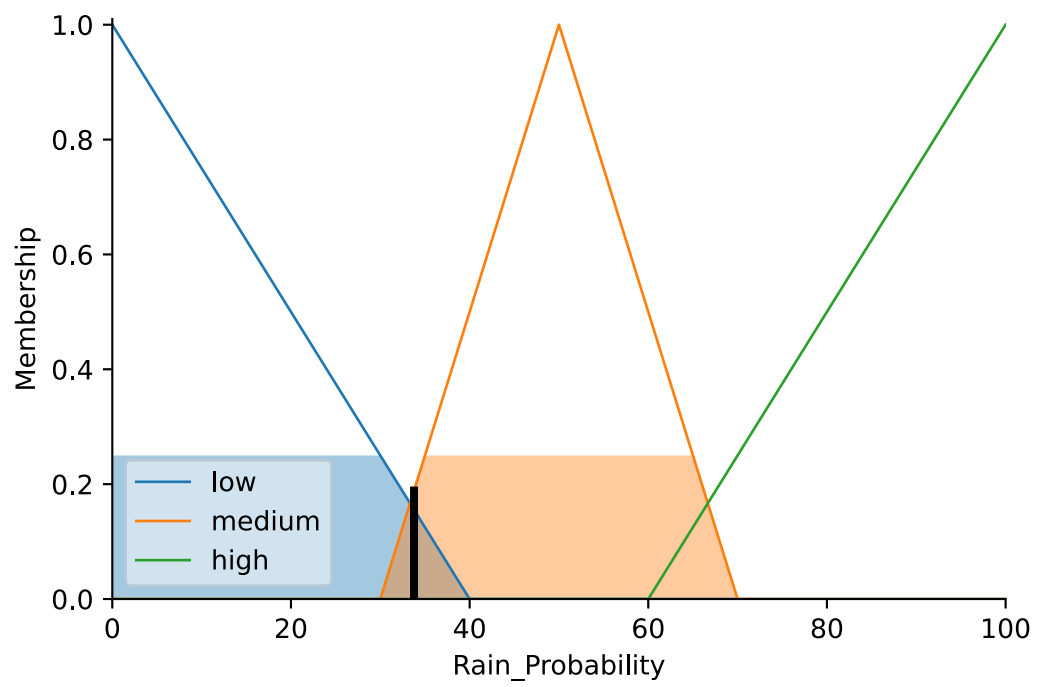
rule3 = ctrl.Rule(antecedent=((humid['medium'] & wSpeed['high']) |
                              (humid['high'] & wSpeed['medium']) |
                              (humid['high'] & wSpeed['high'])),
                 consequent=rainy['high'], label='high')
```

```
In [6]: # add rules to control system
rainy_prediction_ctrl = ctrl.ControlSystem([rule1,rule2,rule3])
rainyProbability = ctrl.ControlSystemSimulation(rainy_prediction_ctrl)
```

```
In [7]: rainyProbability.input['humidity'] = 35  
rainyProbability.input['Wind_Speed'] = 15  
rainyProbability.compute()
```

```
In [8]: humid.view(sim=rainyProbability)  
wSpeed.view(sim=rainyProbability)  
rainy.view(sim=rainyProbability)  
plt.show()
```





```
In [9]: print(rainyProbability.output['Rain_Probability'])
```

33.7776111944028

```
In [ ]:
```