Instituto Politécnico Nacional

Neurofuzzy Systems

PRACTICE 1. DISCRETE AND CONTINUOUS MEMBERSHIP FUNCTIONS

Carlos Josafath López Favila
Teacher Blanca Tovar Corona
Delivery Date: February 28th, 2023

I. Introduction

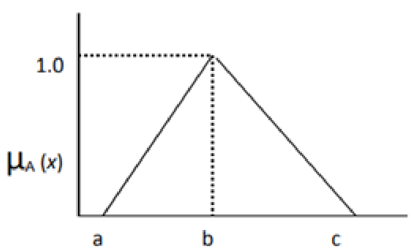
The membership function of a fuzzy set is a generalization of the indicator function in classical sets. In fuzzy logic, it represents the degree of truth as an extension of valuation. Degrees of truth are often confused with probabilities, although they are conceptually distinct, because fuzzy truth represents membership in vaguely defined sets, not likelihood of some event or condition.
A single fuzzy set by itself can model ordering information, but the main reason for employing fuzzy logic and developing fuzzy systems is to infer decisions and information from several pieces of expert knowledge.
The fuzzy set A can be alternatively denoted as follows:

- If X is discrete then $A = \frac{\sum \mu_A(x_i)}{x_i}$
- If X is continuous then $A = \int \frac{\mu_A(x)}{x}$

Here, $\mu_A(x)$ is the "membership function". Value of this function is between 0 and 1. This value represents the "degree of membership" (membership value) of element x in set A. The members of a fuzzy set are members to some degree, known as a membership grade or degree of membership.

A. Triangular Membership Function

Let a, b and c represent the x coordinates of the three vertices of $\mu_A(x)$ in a fuzzy set A (a: lower boundary and c: upper boundary where membership degree is zero, b: the centre where membership degree is 1).

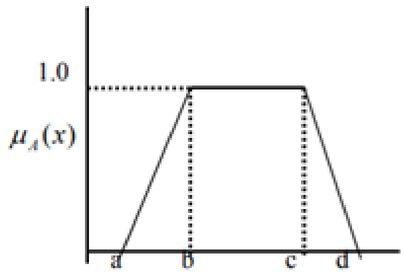


$$\mu_A(x) = \begin{cases} 0 & if \quad x \leq a \\ \frac{x-a}{b-a} & if \quad a < x < b \\ \frac{c-x}{c-b} & if \quad x \geq c \end{cases} \tag{1}$$

B. Discrete Membership Functions

1. Trepezoidal Membership Function

Let a, b, c and d represents the x coordinates of the membership function. Then:

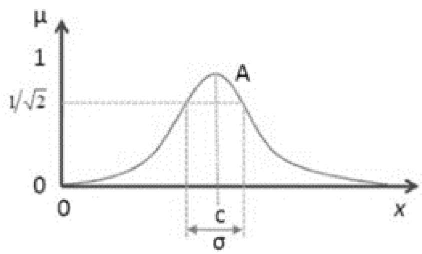


$$\mu_A(x) = \begin{cases} 0 & if \quad x \leq a \\ \frac{x-a}{b-a} & if \quad a < x < b \\ 1 & if \quad b \leq x \leq c \\ \frac{d-x}{d-c} & if \quad c \leq x \leq d \\ 0 & if \quad x \geq d \end{cases} \tag{2}$$

C. Continuous Membership Functions

1. Gaussian Membership Function

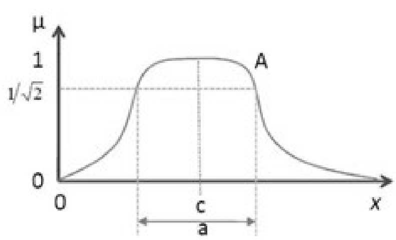
The Gaussian membership function is usually represented as Gaussian(x:c,σ) where c, σ represents the mean and standard deviation.



$$\mu_A(x : c, \sigma) = e^{-\frac{1}{2}(\frac{x-c}{\sigma})^2} \tag{3}$$

2. Bell Membership Function

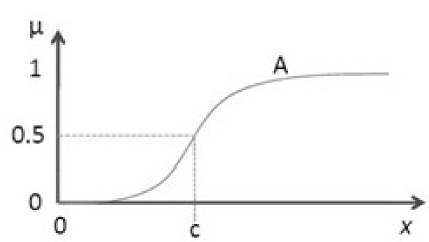
A generalized bell membership function has three parameters: a –responsible for its width, c –responsible for its center and b –responsible for its slopes. Mathematically is:



$$\mu_A(x : c, \sigma) = \frac{1}{1 + |\frac{x-c}{a}|^{2b}} \tag{4}$$

3. Sigmoid Membership Function

A sigmoidal membership function has two parameters: a responsible for its slope at the crossover point x = c. The membership function of the sigmoid function can be represented as Sigmf (x:a, c) and it is:



$$\mu_A(x : c, \sigma) = \frac{1}{1 + e^{-a(x-c)}} \tag{5}$$

II. Objective

In this practice I will develop and use a function in Python (which is my selected platform to work) for each membership functions, that generates two lists, the universe, and the values of the membership degree, using the equations and mathematical descriptions of the membership function described previously.

III. Develompent

Once you have the 5 functions generate the follwing figures:
Figure 1: Overlap the following 3 membership functions with a universe [0 200], step=0.5

- Triangular [50 100 150]
- Trapezoidal [0 0 45 80]
- Trapezoidal [120 155 200 200]

Figure 2: Overlap the following 3 membership functions with a universe [0 200], step=0.5

- Bell [a=40, b=10, c=0]
- Gaussian [c=100, sigma=25]
- Sigmoid [a=0.1, c=140]

IV. Results

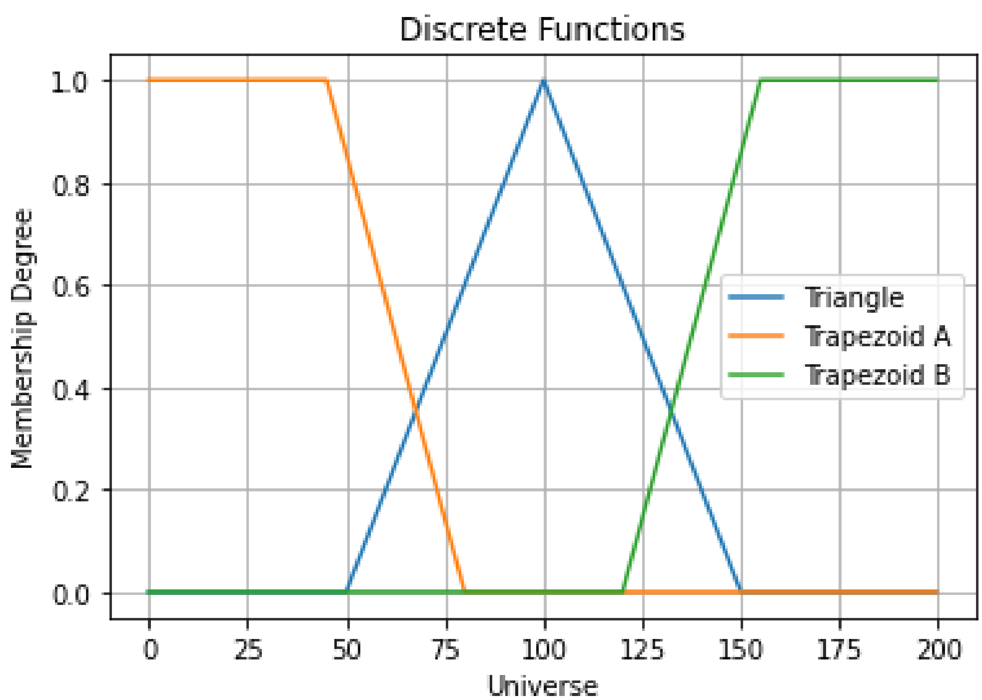


Figure 1: Discrete Membership Functions Overlaped

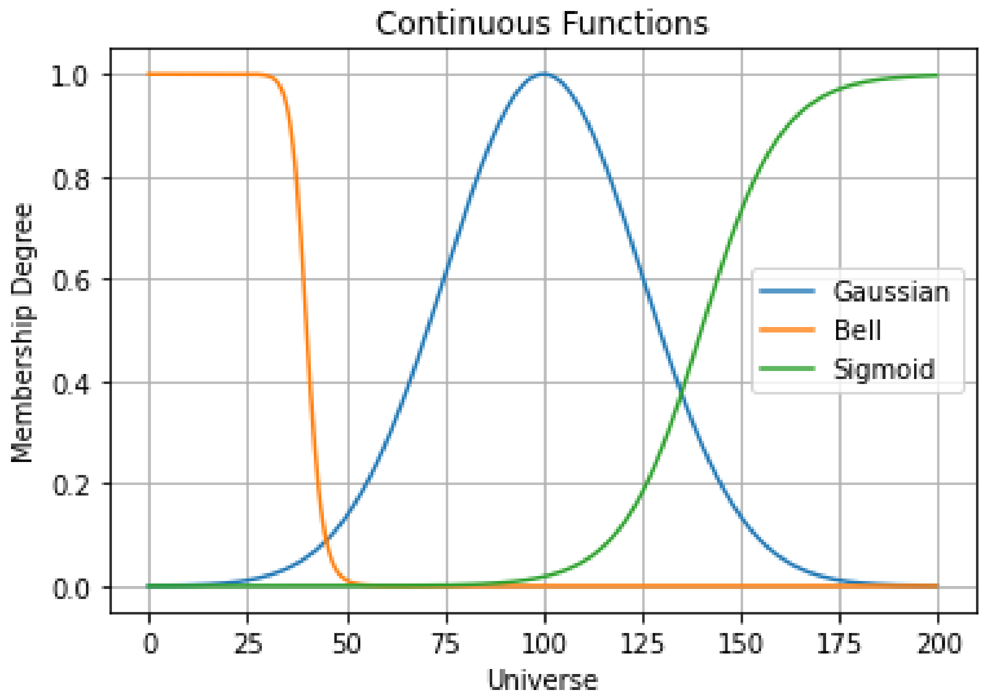


Figure 2: Continuous Membership Functions Overlaped

V. Conclusions

In this practice I learn how to represent membership functions in code graphing them using their mathematical description.

VI. References

- Samanta,D. & Indian Institute of Technology Kharagpur. Fuzzy Membership Function Formulation and Parameterization. https://cse.iitkgp.ac.in/∼dsamanta/courses/archive/sca/Archives/Chapter%203%20Fuzzy %20Membership%20Functions.pdf
- Brown, M.(1996). An Introduction to Fuzzy and Neurofuzzy Systems. http://libgen.is/book/index.php?md5=6EDA1FBA008F9F84B376003C79D0B25B

VII. Apenddix

A. Code

```

import matplotlib.pyplot as plt
import numpy as np


def triangle(ini,fin,a,b,c,step):  #-------------------------TRIANGLE----------------------
```

```
  u=np.arange(ini,fin,step)
  t=[]

  for k in u:
    if (-ini<=k<a):
      t.append(0)
    elif(a<=k<b):
      t.append(((k-b)/(b-a))+1)
    elif(b<=k<c):
      t.append(-(k-c)/(c-b))
    else:
      t.append(0)

  return ([u,t])

def trapezoid(ini,fin,a,b,c,d,step):   #-------------------------TRAPEZOID----------------------
  u=np.arange(ini,fin,step)
  tr=[]

  for k in u:
    if (ini<=k<a):
      tr.append(0)
    elif(a<=k<b):
      tr.append(((k-b)/(b-a))+1)
    elif(b<=k<c):
      tr.append(1)
    elif(c<=k<=d):
      tr.append(-(k-d)/(d-c))
    else:
      tr.append(0)

  return ([u,tr])

def gaussian(ini,fin,c,sig,step): #-------------------------GAUSSIAN----------------------
  u=np.arange(ini,fin,step)
  gs=[]

  for k in u:
    gs.append(np.exp(-(1/2)*((k-c)/sig)**2))
  return([u,gs])

def bell(ini,fin,a,b,c,step):  #-------------------------BELL----------------------
  u=np.arange(ini,fin,step)
  bll=[]

  for k in u:
    bll.append(1/(1+(np.abs((k-c)/a))**(2*b)))

  return([u,bll])

def sigmoid(ini,fin,a,c,step):  #-------------------------SIGMOID----------------------
  u=np.arange(ini,fin,step)
  sgm=[]

  for k in u:
    sgm.append(1/(1+np.exp(-a*(k-c))))

  return([u,sgm])
```

```
#--------------------------------- BUILDING GRAPHS-----------------------


universe1,triangle=triangle(0,200,50,100,150,0.5)
universe2,trapezoid1=trapezoid(0,200,0,0,45,80,0.5)
universe3,trapezoid2=trapezoid(0,200,120,155,200,200,0.5)

plt.figure(1)
plt.plot(universe1,triangle,universe2,trapezoid1,universe3,trapezoid2)
plt.legend(["Triangle", "Trapezoid A", "Trapezoid B"],loc="best")
plt.title("Discrete Functions")
plt.ylabel("Membership Degree")
plt.xlabel("Universe")
plt.grid(True)

universe4,gauss=gaussian(0,200,100,25,0.5)
universe5,bell=bell(0,200,40,10,0,0.5)
universe6,sigmoid=sigmoid(0,200,0.1,140,0.5)

plt.figure(2)
plt.plot(universe4,gauss,universe5,bell,universe6,sigmoid)
plt.legend(["Gaussian","Bell","Sigmoid"])
plt.title("Continuous Functions")
plt.ylabel("Membership Degree")
plt.xlabel("Universe")
plt.grid(True)
```