

UNIVERSIDAD POLITECNICA SALESIANA

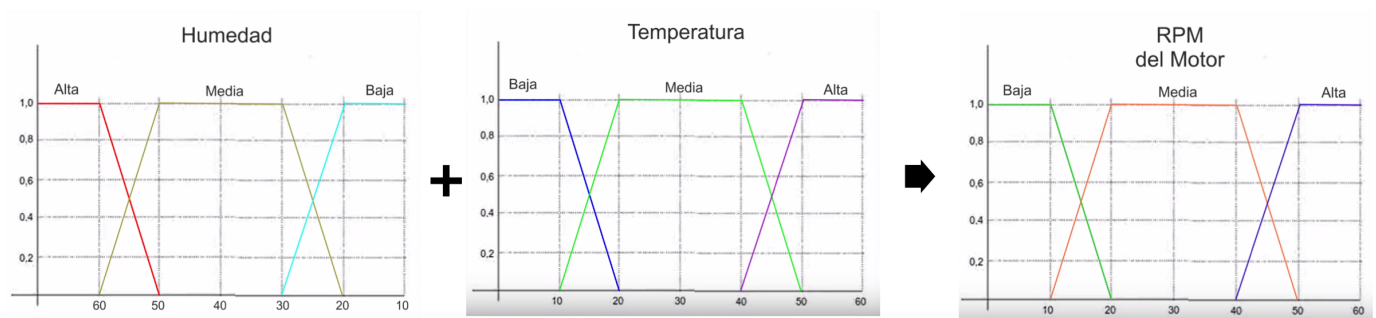
Práctica Lógica Difusa

Alumna: Narcisa Araujo 

Enunciado

En un galpón se tiene una temperatura de 18 grados centígrados, y una humedad de aproximadamente 22 grados centígrados. Según estos valores determinar cual es la velocidad que debería estar funcionando el motor. Para revisar las reglas, función de pertinencia y el proceso revisar el siguiente link:
<https://medium.com/@javierdiazarcal/C3%B3gica-difusa-ejercicio-2-bases-de-la-ia-1a8ae594cc15>
(<https://medium.com/@javierdiazarcal/C3%B3gica-difusa-ejercicio-2-bases-de-la-ia-1a8ae594cc15>) En base a ello, desarrollar e implementar el sistema dentro de Python o Java en donde me permita modificar los valores de la temperatura y humedad, generando así un sistema experto basado en lógica difusa para obtener la velocidad del motor de aire acondicionado. Este sistema deberá tener la opción de poder modificar los valores de la temperatura y humedad con un scroll bar y obtener la velocidad de giro. Además, deberá presentarme las graficas de pertenencia de INPUT/OUTPUT del sistema difuso y como estas varían de acuerdo al cambio de las variables.

Valores del Problema

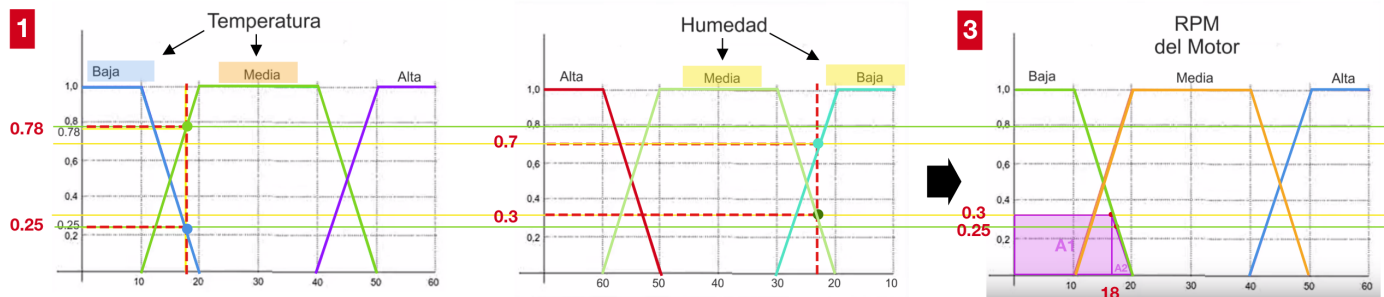


Reglas a considerar

| Temperatura | Humedad | RPM del Motor |
|-------------|---------|---------------|
| Baja | Alta | Baja |
| Media | Alta | Media |
| Alta | Alta | Media |
| Baja | Media | Baja |
| Media | Media | Baja |
| Alta | Media | Media |
| Baja | Baja | Baja |
| Media | Baja | Baja |
| Alta | Baja | Alta |

Desarrollo

1- Ubico los valores indicados en el gráfico tanto en Humedad y Temperatura, para ubicar los grados de pertenencia e indico unos valores de conveniencia. 2- Según las reglas activadas, ubico el valor mínimo de cada una [Método Mamdani] 3- Proyecto los mínimos según las reglas en "RPM del Motor" 4- Defuzzyficación: Aplicamos el método del Centroide



2

| Temperatura | Humedad | RPM del Motor |
|-------------|---------|---------------|
| Baja | Alta | Baja |
| Media | Alta | Media |
| Alta | Alta | Media |
| Baja | Media | Baja |
| Media | Media | Baja |
| Alta | Media | Media |
| Baja | Baja | Baja |
| Media | Baja | Baja |
| Alta | Baja | Alta |

→

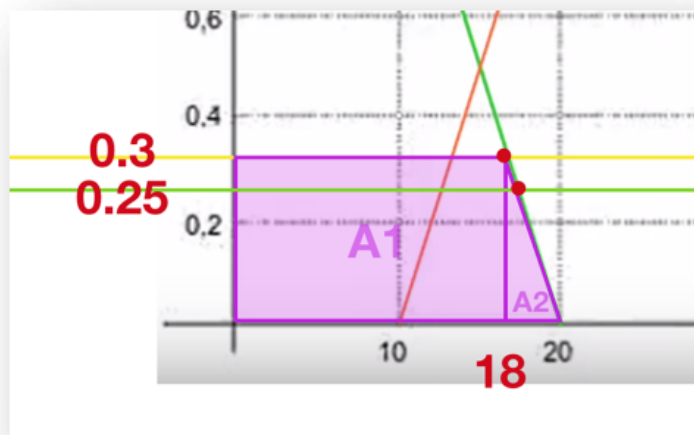
Evalúo la más baja en c/u

0.25

0.30

0.25

0.70



Centroides

$$C1 : (18 / 2) = 9$$

$$C2 : [(20 - 18) / 3] + 18 = 18.66$$

Áreas Parciales

$$A1 : B \times A$$

$$A1 : 18 \times 0.3 = 5.4$$

$$A2 : (B \times A) / 2$$

$$A2 : (2 \times 0.3) / 2 = 0.3$$

Área Total

$$AT = A1 + A2$$

$$AT = 5.4 + 3.0 = 8.4$$

CALCULO DEL CENTROIDE

$$C = \frac{C1 \times A1 + C2 \times A2}{A(total)}$$

$$C = \frac{9 \times 5.4 + 18.66 \times 0.3}{8.4} = 6.38$$

Ejemplo Resuelto en Python

```
In [1]: from tkinter import messagebox, ttk
        from tkinter import *
        import tkinter
        import matplotlib.pyplot as plt
        import numpy as np
        from tkinter.messagebox import showinfo
```

```

In [30]: #VENTANA PRINCIPAL
root = tkinter.Tk()
root.title("LOGICA DIFUSA")
root.geometry("800x800")

nb = ttk.Notebook(root)
nb.pack(fill='both', expand='yes')

p1 = ttk.Frame(nb)
p2 = ttk.Frame(nb)
p3 = ttk.Frame(nb)
nb.add(p1, text='Temperatura')
nb.add(p2, text='Humedad')
nb.add(p3, text='RPM Motor')

campo_de_texto = ttk.Entry(p1)
campo_de_texto.pack()
def grafico_temp():
    print("TEMPERATURA")
    valoresX = [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 10,11,12,
13,14,15,16,17,18
,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,
40,41,42,43,44,45,
46,47,48,49,50, 40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,
56,57,58,59,60]

    valoresY = [1,1,1,1,1,1,1,1,1,1,1,1,0.9,0.8,0.7,0.6,0.5,0.4,0.3,0.2,0.1,0, 0
,0.1,0.2,0.3,0.4,0.5,
0.6,0.7,0.8,0.9,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0.9,0.
8,0.7,0.6,0.5,0.4,
0.3,0.2,0.1,0, 0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1,1,1,1,1,1
,1,1,1,1,1]

    plt.plot(valoresX[:21], valoresY[:21], lw=5,label="Baja")
    plt.plot(valoresX[21:62], valoresY[21:62], lw=5, label="Media")
    plt.plot(valoresX[62:], valoresY[62:], lw=5, label="Alta")
    plt.axis([min(valoresX),max(valoresX),min(valoresY),max(valoresY)])
    plt.grid()
    plt.legend(loc="lower right", title="Legend Title", frameon=False)
    a = campo_de_texto.get()
    plt.axvline(int(a), label='pyplot vertical line')
    #if int(a)>= 0 and int(a)>= 0
    plt.scatter(int(a), 0.25, color='black' )
    plt.scatter(int(a), 0.78, color='green' )
    plt.show()

boton_tem = Button(p1, text="TEMPERATURA", command=grafico_temp)
boton_tem.pack()

def grafico_hume():
    print("HUMEDAD")
    valoresX = [70, 60, 50,60, 50, 30, 20, 30,20,10]

```

```

valoresY = [1, 1, 0, 0, 1, 1, 0, 0, 1, 1]
plt.plot(valoresX[:3], valoresY[:3], lw=5, label="Alta")
plt.plot(valoresX[3:7], valoresY[3:7], lw=5, label="Media")
plt.plot(valoresX[7:], valoresY[7:], lw=5, label="Baja")
plt.axis([max(valoresX), min(valoresX), min(valoresY), max(valoresY)])
plt.grid()
plt.legend(loc="lower right", title="Legend Title", frameon=False)
a=campo_de_texto.get()
plt.axvline(int(a), label='pyplot vertical line')
plt.scatter(int(a), 0.3, color='black')
plt.scatter(int(a), 0.7, color='green')
plt.show()

```

```

boton_hum = Button(p2, text="HUMEDAD", command=grafico_hume)
boton_hum.pack()

```

```

def grafico_RPM():
    print("RPM")
    valoresX = [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 10,11,12,
    13,14,15,16,17,18
    ,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,
    40,41,42,43,44,45,
    46,47,48,49,50, 40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,
    56,57,58,59,60]

    valoresY = [1,1,1,1,1,1,1,1,1,1,1,1,0.9,0.8,0.7,0.6,0.5,0.4,0.3,0.2,0.1,0, 0
    ,0.1,0.2,0.3,0.4,0.5,
    0.6,0.7,0.8,0.9,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0.9,0.
    8,0.7,0.6,0.5,0.4,
    0.3,0.2,0.1,0, 0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1,1,1,1,1,1
    ,1,1,1,1,1]

    plt.plot(valoresX[:21], valoresY[:21], lw=5, label="Baja")
    plt.plot(valoresX[21:62], valoresY[21:62], lw=5, label="Media")
    plt.plot(valoresX[62:], valoresY[62:], lw=5, label="Alta")

    plt.axis([min(valoresX), max(valoresX), min(valoresY), max(valoresY)])
    plt.grid()
    plt.legend(loc="lower right", title="Legend Title", frameon=False)
    a = campo_de_texto.get()
    plt.axvline(int(a), label='pyplot vertical line')

    plt.scatter(int(a), 0.25, color='black' )
    plt.scatter(int(a), 0.3, color='green' )
    plt.scatter(int(a), 18, color='red' )

    plt.show()

```

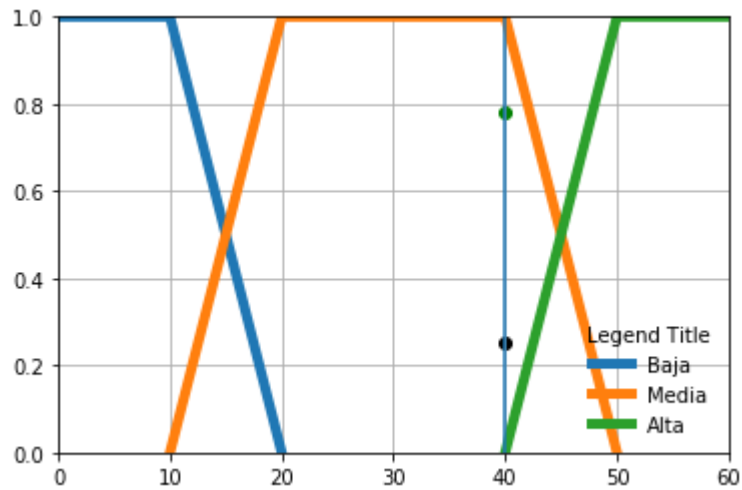
```

boton_rpm = Button(p3, text="RPM", command=grafico_RPM)
boton_rpm.pack()

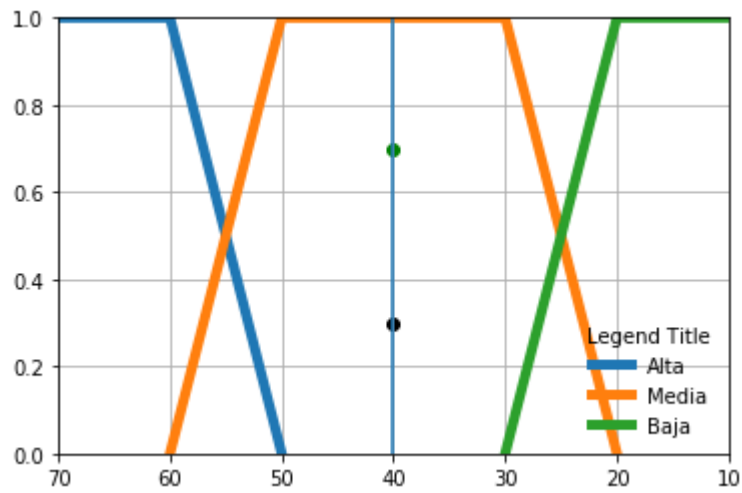
```

```
root.mainloop()
```

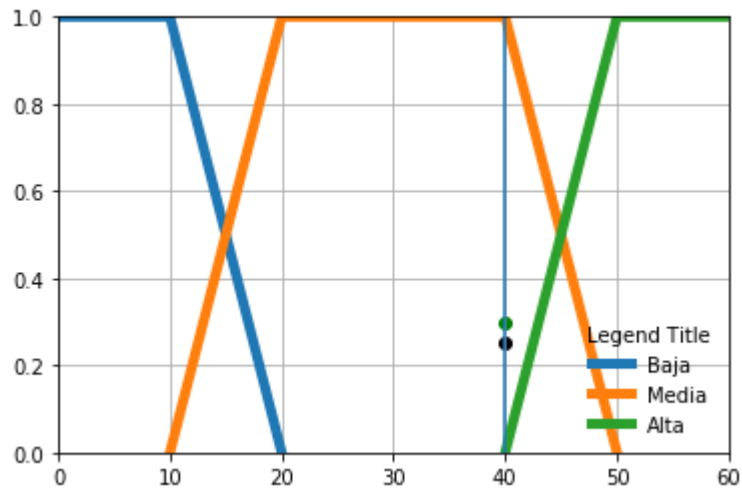
TEMPERATURA



HUMEDAD



RPM



Bibliografia

<https://medium.com/@javierdiazarca/l%C3%B3gica-difusa-ejercicio-2-bases-de-la-ia-1a8ae594cc15>
(<https://medium.com/@javierdiazarca/l%C3%B3gica-difusa-ejercicio-2-bases-de-la-ia-1a8ae594cc15>)

In []: