



Actividad 5: Lógica Difusa

CE Inteligencia Artificial y Big Data
Modelos de Inteligencia Artificial
2024/2025

Daniel Marín López

Índice

1. Control de temperatura en una ducha	3
2. Evaluación de la satisfacción del cliente	7

1. Control de temperatura en una ducha

1. a) Fase de Modelado:

Primero definimos las variables:

- Temperatura del agua (T)
- Tiempo en la ducha (t)

Domínios:

Para la temperatura T tenemos lo siguiente:

- Rango Universal: $[0^{\circ}\text{C}, 60^{\circ}\text{C}]$
- Etiquetas:
 - Fría: $[0^{\circ}\text{C}, 20^{\circ}\text{C}]$
 - Templada: $[15^{\circ}\text{C}, 40^{\circ}\text{C}]$
 - Caliente: $[30^{\circ}\text{C}, 60^{\circ}\text{C}]$

Para el tiempo t tenemos lo siguiente:

- Rango Universal: $[0\text{ min}, 30\text{ min}]$
- Etiquetas:
 - Corto: $[0\text{ min}, 10\text{ min}]$
 - Medio: $[5\text{ min}, 20\text{ min}]$
 - Largo: $[15\text{ min}, 30\text{ min}]$

Fase de Fuzzificación:

Definimos las funciones que usaremos tanto para T como para t:

$$T_{\text{Fría}}(x) = \begin{cases} 0, & x \leq 0 \\ \frac{20-x}{20-0}, & 0 < x \leq 20 \\ 0, & x > 20 \end{cases}$$

$$T_{Templada}(x) = \begin{cases} 0, & (x < 15) \vee (x > 40) \\ \frac{x-15}{20-15}, & 15 \leq x \leq 20 \\ 1, & 20 \leq x \leq 40 \\ \frac{40-x}{40-20}, & 20 \leq x \leq 40 \end{cases}$$

$$T_{caliente}(x) = \begin{cases} 0, & x \leq 30 \\ \frac{60-x}{60-30}, & 30 < x \leq 60 \\ 0, & x > 60 \end{cases}$$

$$t_{corto}(x) = \begin{cases} 0, & x \leq 0 \\ \frac{10-x}{10-0}, & 0 < x \leq 10 \\ 0, & x > 10 \end{cases}$$

$$t_{largo}(x) = \begin{cases} 0, & x \leq 15 \\ \frac{30-x}{30-15}, & 15 < x \leq 30 \\ 0, & x > 30 \end{cases}$$

$$t_{Medio}(x) = \begin{cases} 0, & (x < 5) \vee (x > 20) \\ \frac{x-5}{10-5}, & 5 < x \leq 10 \\ 1, & 10 \leq x \leq 20 \\ \frac{20-x}{20-10}, & 10 \leq x \leq 20 \end{cases}$$

Grados de pertenencia:

$$\bullet T_{Fria}(25) = 0$$

$$\bullet T_{Templada}(25) = \frac{40-25}{40-20} = \frac{15}{20} = 0.75$$

$$\bullet T_{caliente}(25) = 0$$

$$\bullet t_{corto}(15) = 0$$

$$\bullet t_{Medio}(15) = \frac{20-15}{20-10} = \frac{5}{10} = 0.5$$

$$\bullet t_{largo}(15) = 0$$

Fase de Motor de Inferencia:

Definimos reglas y evaluamos.

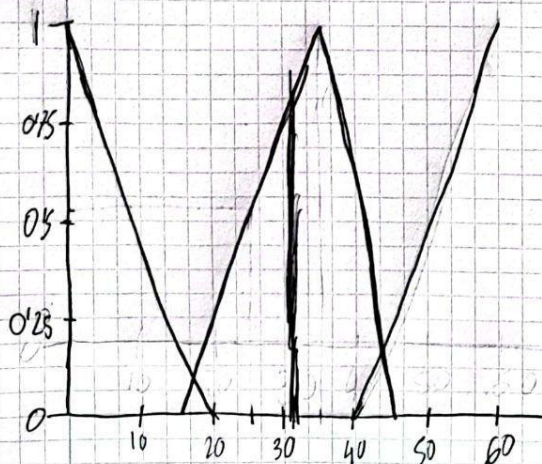
- Si T es Fría y t es Corto \rightarrow Baja temperatura
- Si T es Templada y t es Medio \rightarrow Temperatura Mod.
- Si T es Caliente y t es Alto \rightarrow Alta temperatura

Evaluación de reglas:

- Regla 1: $T_{\text{Fría}}(25) \wedge t_{\text{Corto}}(15) = \min(0, 0) = 0$
- Regla 2: $T_{\text{Templada}}(25) \wedge t_{\text{Medio}}(15) = \min(0,75, 0,15) = 0,15$
- Regla 3: $T_{\text{Caliente}}(25) \wedge t_{\text{Alto}}(15) = \min(0, 0) = 0$

Fase de Desfuzzificación:

La salida solo viene influenciada por la Regla 2 (Temperatura Moderada) con un grado de 0,15. Si asumimos que el conjunto difuso para la salida (Temp) es triangular con un valor central moderado de 35°C , el resultado estará ligeramente desplazado a la derecha.



La temperatura del agua debería ser de 31°C .

```

import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl

temperatura_agua = ctrl.Antecedent(np.arange(0, 61, 1), 'temperatura_agua')
tiempo_ducha = ctrl.Antecedent(np.arange(0, 31, 1), 'tiempo_ducha')
salida_temperatura = ctrl.Consequent(np.arange(0, 61, 1), 'salida_temperatura')

temperatura_agua['fria'] = fuzz.trimf(temperatura_agua.universe, [0, 0, 20])
temperatura_agua['templada'] = fuzz.trapmf(temperatura_agua.universe, [15, 20, 40, 40])
temperatura_agua['caliente'] = fuzz.trimf(temperatura_agua.universe, [30, 60, 60])

tiempo_ducha['corto'] = fuzz.trimf(tiempo_ducha.universe, [0, 0, 10])
tiempo_ducha['medio'] = fuzz.trapmf(tiempo_ducha.universe, [5, 10, 20, 20])
tiempo_ducha['largo'] = fuzz.trimf(tiempo_ducha.universe, [15, 30, 30])

salida_temperatura['baja'] = fuzz.trimf(salida_temperatura.universe, [0, 0, 20])
salida_temperatura['moderada'] = fuzz.trimf(salida_temperatura.universe, [15, 35, 45])
salida_temperatura['alta'] = fuzz.trimf(salida_temperatura.universe, [40, 60, 60])

rule1 = ctrl.Rule(temperatura_agua['fria'] & tiempo_ducha['corto'], salida_temperatura['baja'])
rule2 = ctrl.Rule(temperatura_agua['templada'] & tiempo_ducha['medio'],
salida_temperatura['moderada'])
rule3 = ctrl.Rule(temperatura_agua['caliente'] & tiempo_ducha['largo'], salida_temperatura['alta'])

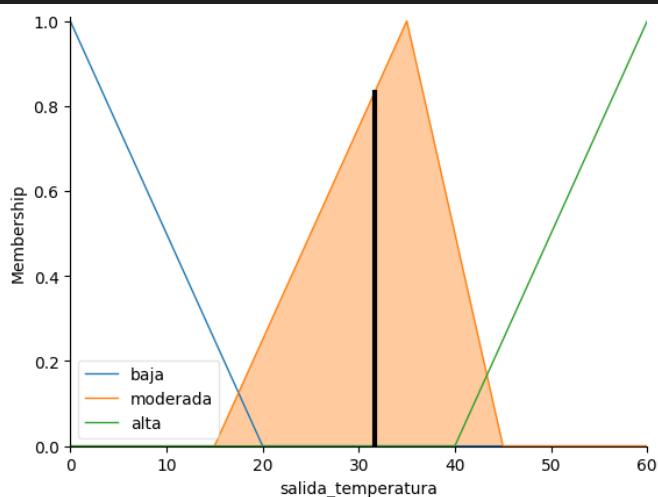
sistema_control = ctrl.ControlSystem([rule1, rule2, rule3])
simulador = ctrl.ControlSystemSimulation(sistema_control)

simulador.input['temperatura_agua'] = 25 # Temperatura del agua (°C)
simulador.input['tiempo_ducha'] = 15 # Tiempo de ducha (min)

simulador.compute()

print(f"Salida de temperatura: {simulador.output['salida_temperatura']:.2f} °C") # 31.67 °C
print("-"*35)
salida_temperatura.view(sim=simulador)

```



2. Evaluación de la satisfacción del cliente

2. a) Fase de Modelado:

Primero definimos las variables:

- Tiempo de respuesta (t)
- Calidad del producto (C)

Dominios:

- Rango Universal para t : $[0 \text{ min}, 30 \text{ min}]$
- Etiquetas para t :
 - Rápido: $[0 \text{ min}, 10 \text{ min}]$
 - Moderado: $[5 \text{ min}, 20 \text{ min}]$
 - Lento: $[15 \text{ min}, 30 \text{ min}]$
- Rango Universal para C : $[0, 10]$
- Etiquetas para C :
 - Baja: $[0, 5]$
 - Media: $[3, 7]$
 - Alta: $[5, 10]$

Fase de Fuzzificación:

$$t_{\text{rapido}}(x) = \begin{cases} 0, & x \leq 0 \\ \frac{10-x}{10-0}, & 0 < x \leq 10 \\ 0, & x > 10 \end{cases} \quad t_{\text{lento}}(x) = \begin{cases} 0, & x \leq 15 \\ \frac{30-x}{30-15}, & 15 < x \leq 30 \\ 0, & x > 30 \end{cases}$$

$$t_{\text{mod}}(x) = \begin{cases} 0, & (x < 5) \vee (x > 20) \\ \frac{x-5}{10-5}, & 5 \leq x \leq 10 \\ 1, & 10 \leq x \leq 20 \\ \frac{20-x}{20-10}, & 10 \leq x \leq 20 \end{cases}$$

$$C_{Baja}(x) = \begin{cases} 0, & x \leq 0 \\ \frac{5-x}{5-0}, & 0 < x \leq 5 \\ 0, & x > 5 \end{cases} \quad C_{Alta}(x) = \begin{cases} 0, & x \leq 7 \\ \frac{10-x}{10-7}, & 7 < x \leq 10 \\ 0, & x > 10 \end{cases}$$

$$C_{Media}(x) = \begin{cases} 0, & (x < 3) \vee (x > 7) \\ \frac{x-3}{5-3}, & 3 \leq x \leq 5 \\ 1, & 5 \leq x \leq 7 \\ \frac{7-x}{7-5}, & 5 \leq x \leq 7 \end{cases}$$

$$\cdot t_{Rápido}(15) = 0$$

$$\cdot t_{Mod.}(15) = \frac{20-15}{20-10} = \frac{5}{10} = 0.5$$

$$\cdot t_{Lento}(15) = 0$$

$$\cdot C_{Baja}(6) = 0$$

$$\cdot C_{Media}(6) = \frac{7-6}{7-5} = \frac{1}{2} = 0.5$$

$$\cdot C_{Alta}(6) = 0$$

Fase del Motor de Inferencia:

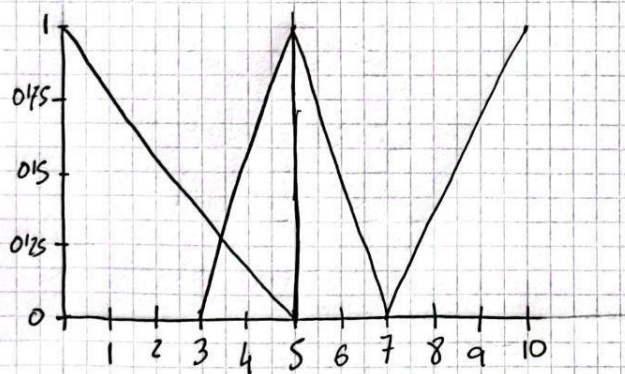
- Si t es rápido y C es alta \rightarrow Muy satisfecho
- Si t es moderado y C es media \rightarrow Satisfecho
- Si t es lento y C es baja \rightarrow Insatisfecho

Evaluamos:

- Regla 1: $t_{\text{rápido}}(15) \wedge C_{\text{alta}}(6) = \min(0, 0) = 0$
- Regla 2: $t_{\text{mod.}}(15) \wedge C_{\text{media}}(6) = \min(0.5, 0.5) = 0.5$
- Regla 3: $t_{\text{lento}}(15) \wedge C_{\text{baja}}(6) = \min(0, 0) = 0$

Fase de Desfuzzificación:

De nuevo, la salida se ve influenciada por la Regla 2 ("Satisfecho") con un grado de 0.5.



Es muy posible que el sistema difuso devuelva 5.

```

tiempo_respuesta = ctrl.Antecedent(np.arange(0,31,1), "tiempo_respuesta")
calidad_producto = ctrl.Antecedent(np.arange(0,11,1), "calidad_producto")
satisfaccion = ctrl.Consequent(np.arange(0,11,1), "satisfaccion")

tiempo_respuesta['rapido'] = fuzz.trimf(tiempo_respuesta.universe, [0,0,10])
tiempo_respuesta['moderado'] = fuzz.trapmf(tiempo_respuesta.universe, [5, 10, 20, 20])
tiempo_respuesta['lento'] = fuzz.trimf(tiempo_respuesta.universe, [15, 30, 30])

calidad_producto['baja'] = fuzz.trimf(calidad_producto.universe, [0,0,5])
calidad_producto['media'] = fuzz.trapmf(calidad_producto.universe, [3, 5, 7, 7])
calidad_producto['alta'] = fuzz.trimf(calidad_producto.universe, [5, 10, 10])

satisfaccion['insatisfecho'] = fuzz.trimf(satisfaccion.universe, [0,0,5])
satisfaccion['satisfecho'] = fuzz.trimf(satisfaccion.universe, [3, 5, 7])
satisfaccion['muy_satisfecho'] = fuzz.trimf(satisfaccion.universe, [7, 10, 10])

rule1_2 = ctrl.Rule(tiempo_respuesta['rapido'] & calidad_producto['alta'],
satisfaccion['muy_satisfecho'])
rule2_2 = ctrl.Rule(tiempo_respuesta['moderado'] & calidad_producto['media'],
satisfaccion['satisfecho'])
rule3_2 = ctrl.Rule(tiempo_respuesta['lento'] & calidad_producto['baja'],
satisfaccion['insatisfecho'])

sistema_control_2 = ctrl.ControlSystem([rule1_2, rule2_2, rule3_2])
simulador_2 = ctrl.ControlSystemSimulation(sistema_control_2)

simulador_2.input['tiempo_respuesta'] = 15 # Tiempo de respuesta (min)
simulador_2.input['calidad_producto'] = 6 # Calidad del producto

simulador_2.compute()

print(f"Satisfacción del cliente: {simulador_2.output['satisfaccion']:.2f}") # 5
print("-"*35)
satisfaccion.view(sim=simulador_2)

```

