

INTRODUCCIÓN A LOS SISTEMAS DE INFERENCIA DIFUSA USANDO OCTAVE.

SALVADOR DÍAZ MALDONADO
MARZO, 2022
salva.gestion@gmail.com

Contenido

INTRODUCCIÓN.....	1
SISTEMA DE INFERNICIA DIFUSA, SID.	2
Ejemplo 1. Problema básico de la propina de un comensal	2
Mandani	2
Sugeno	12
Ejemplo 2. Segundo problema básico de dar propina	16
Mandani	16
Sugeno	24
Ejemplo 3. Problema de aplicación: diseño de intercambiador de calor.	26
Mandani	26
Sugeno	40
Ejemplo 4. Problema sobre el clima.....	46
Mandani	46
Sugeno	57

INTRODUCCIÓN.

Uno de los paquetes computacionales más comunes para resolver problemas de lógica difusa es Matlab. Por otro lado, durante los últimos años se han desarrollado herramientas de software libre que mejoran continuamente, algunas de las cuales son R, Octave y SciLab, entre otras.

El presente trabajo pretende precisamente introducir a los sistemas de inferencia difusos usando estas herramientas de manera alternativa, además de apoyar en la labor académica, para ello se presentan cuatro ejemplos a continuación, todos resueltos aplicando los modelos de Mamdani y Sugeno. Principalmente se aplica Octave, además de usar algunos de los otros (Matlab, Scilab o R), debido a su gran compatibilidad con Matlab, aunque la interfase gráfica de este es mejor, hasta ahora.

Se incluyen ademas los códigos de Octave y de R, cuando se aplica, con el fin de promover la programación que permita resolver problemas a la medida del usuario.

SISTEMA DE INFERENCIA DIFUSA, SID.

Ejemplo 1¹. Problema básico de la propina de un comensal

Mandani

¿Dado un número de 1 a 10 que representa la calidad del servicio, CS , en un restaurante (donde 10 es excelente) y otro número, también de 1 a 10, que representa la calidad de la comida, CC , en ese restaurante (otra vez, 10 es excelente), cuál debería ser la propina?

El punto de partida es escribir las tres reglas de la asignación de propina por parte del comensal:

1. Si la CC es *Mala* y la CS es *Mala*, entonces la *Propina* es Aprox. 5%.
2. Si la CC es *Mala* y la CS es *Buena*, entonces la *Propina* es Aprox. 10%.
3. Si la CC es *Buena* y la CS es *Mala*, entonces la *Propina* es Aprox. 10%.
4. Si la CC es *Buena* y la CS es *Buena*, entonces la *Propina* es Aprox. 20%.

Suponer que la propina promedio es 10%, una propina generosa es 20% y una propina baja es 5%.

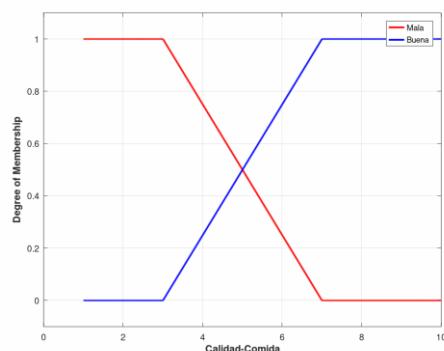
Solución.

Inicialmente se almacenan los conjuntos difusos de las variables de entrada, en este caso CC y CS .

La variable de entrada CC tiene dos valores lingüísticos: *Mala* y *Buena*. Y se comportan como una función de membresía trapezoidal con los siguientes parámetros:

- **Calidad Comida, CC:**

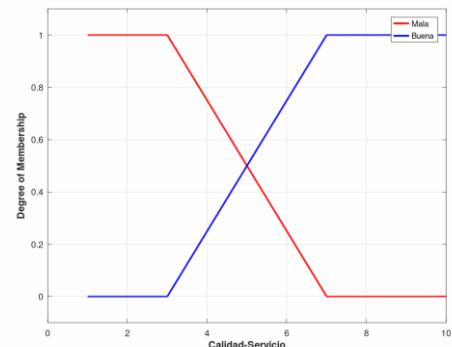
Nombre:	Función de membresía:	a	b	c	d
Mala	Trapezial	0	0	3	7
Buena	Trapezial	3	7	10	10



¹ MathWorks. Adaptado de Fuzzy Logic Toolbox. User Guide Matlab.

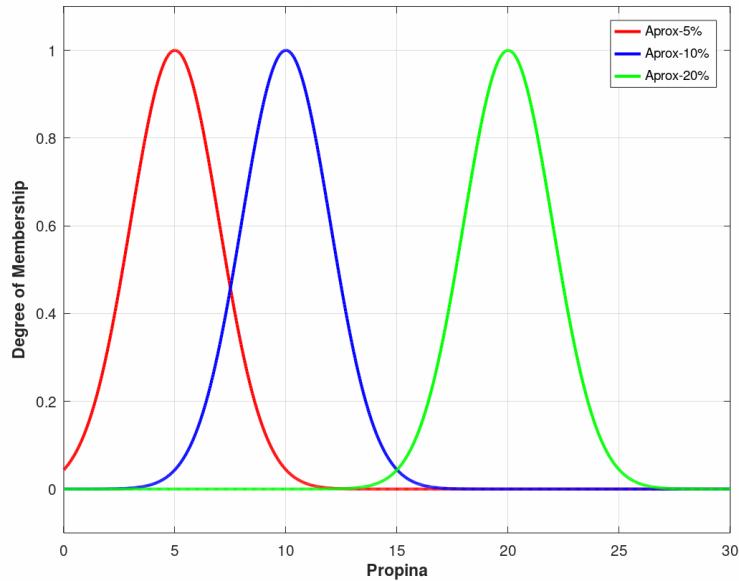
- **Calidad Servicio, CS:**

Nombre:	Función de membresía:	a	b	c	d
Mala	Trapecial	0	0	3	7
Buena	Trapecial	3	7	10	10



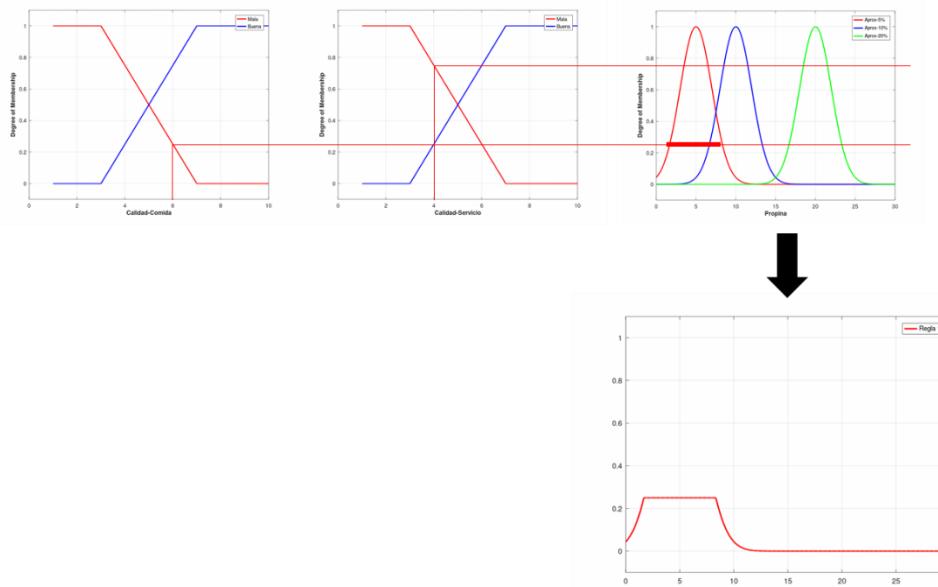
- **Propina:**

Nombre:	Función de membresía:	Parámetro de escala (desviación estándar)	Parámetro de posición (media):
Aprox 5%	Normal (Gauss)	2	5
Aprox 10%	Normal (Gauss)	2	10
Aprox 20%	Normal (Gauss)	2	20

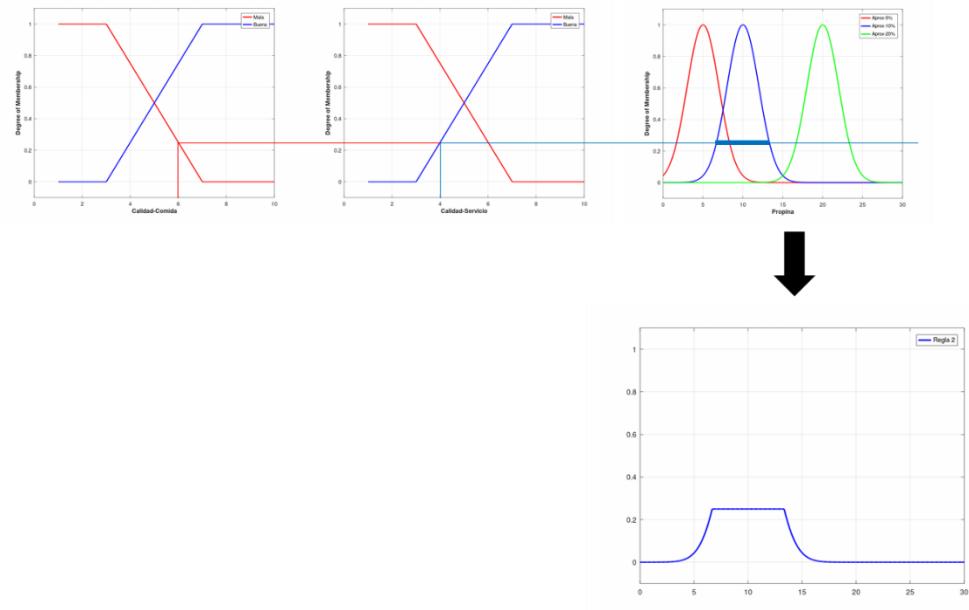


a) Método de defusificación del centroide. Suponer Calidad-Comida, $CC = 6$ y Calidad-Servicio, $CS = 4$.

1. Si la CC es *Mala* y la CS es *Buena*, entonces la *Propina* es Aprox. 5%.



2. Si la CC es *Mala* y la CS es *Buena*, entonces la *Propina* es Aprox 10%.

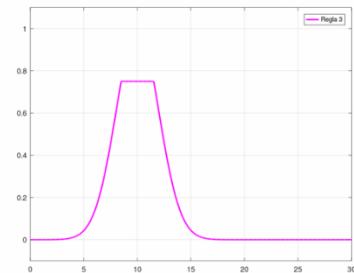
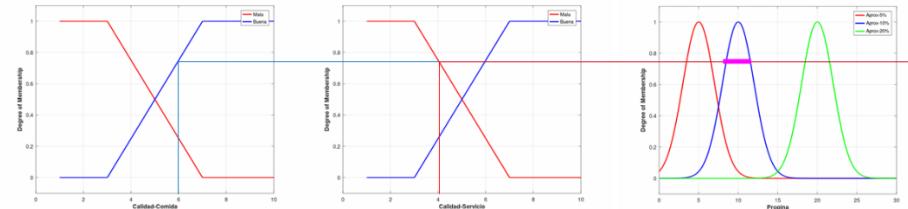


Introducción al Sistema de Inferencia Difusa usando Octave

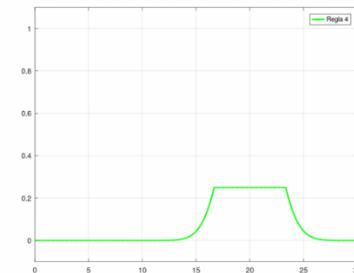
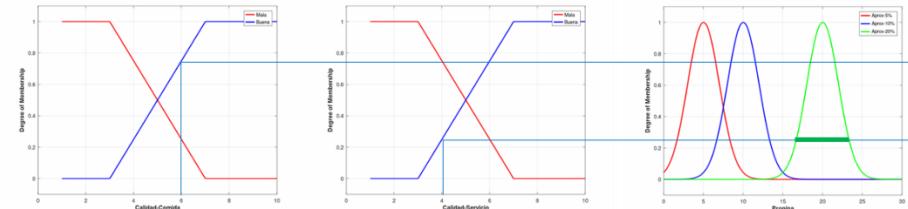
Salvador Díaz Maldonado

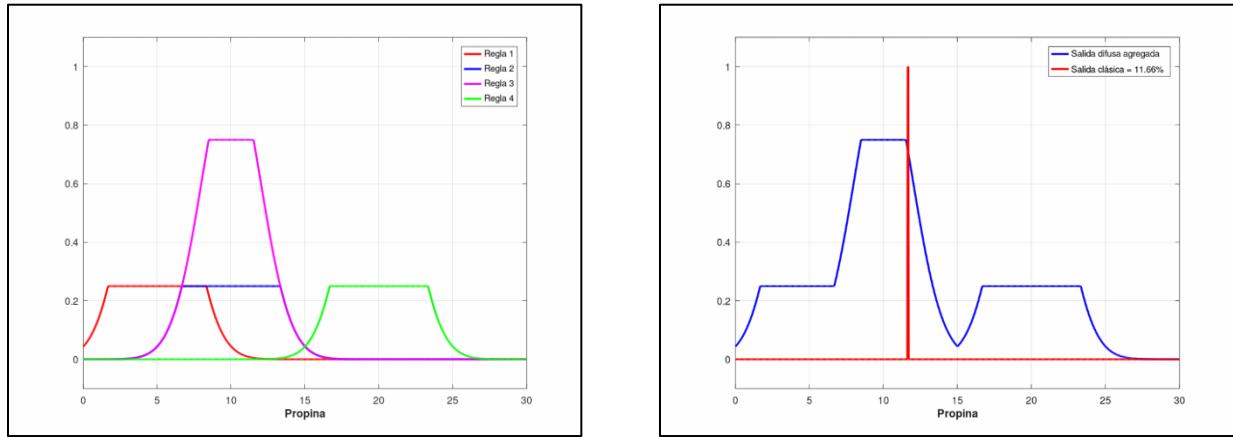
2022.

3. Si la CC es Buena y la CS es Mala, entonces la Propina es Aprox 10%.



4. Si la CC es Buena y la CS es Buena, entonces la Propina es Aprox. 20%.





b) Resolviendo con Octave:

- **Código del programa,**

```
## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_1_Mandani.m
## Última modificación: 11 mar 2022
#
# OJO: Cargar previamente "pkg load fuzzy-logic-toolkit";
clc; clear; close all;
fis=readfis ('Calculadora de propinas Mamdani');
## Trazar las funciones de pertenencia de entrada y salida.
plotmf (fis, 'input', 1)
plotmf (fis, 'input', 2);
plotmf (fis, 'output', 1);
resp='S';
while (resp=='S')
    clc
    ## Mostrar las reglas en formato simbólico.
    puts ("\nReglas de la calculadora de propinas Mamdani:\n\n");
    showrule(fis); disp(' ');disp(' ');
    ## Calcule la propina usando la Calidad Comida y la Calidad Servicio.
    puts ("\nIngresar los valores de Calidad de comida y del Servicio");
    puts ("\n para evaluar con el Sistema de Inferencia Difusa (SID)\n\n");
    CC = input('CC, Calidad de la Comida (1-10)? ');
    CS = input('CS, Calidad del Servicio (1-10)? '); disp(' ');disp(' ');
    [output, rule_input, rule_output, fuzzy_output] = ...
    evalfis ([CC CS], fis, 1001);
    ## Trazar la primera salida (Consejo) de las reglas difusas individuales
    ## en un juego de ejes.
    x_axis = linspace (fis.output(1).range(1), ...
                      fis.output(1).range(2), 1001);
    colors = ['r' 'b' 'm' 'g'];
    figure ('NumberTitle', 'off', 'Name', ...
            'Output of Fuzzy Rules 1-4 for Input = (CC, CS)');
    for i = 1 : 4
        y_label = [colors(i) ";Regla " num2str(i) ";"];
        plot (x_axis, rule_output(:,i), y_label, 'LineWidth', 2);
        ylim ([-0.1, 1.1]);
        grid;
        figure();
    endfor
    for i = 1 : 4
        y_label = [colors(i) ";Regla " num2str(i) ";"];

```

```

plot (x_axis, rule_output(:,i), y_label, 'LineWidth', 2);
grid;
hold on;
endfor
ylim ([-0.1, 1.1]);
xlabel ('Propina', 'FontWeight', 'bold');
grid;
hold;
## Trazar la primera salida difusa agregada y la primera salida difusa
## (Propina) en un juego de ejes.
figure('NumberTitle', 'off', 'Name', ...
    'Agregación y defuzzificación para la entrada = (CC, CS)');
plot (x_axis, fuzzy_output(:, 1), "b;Salida difusa agregada;", ...
    'LineWidth', 2);
hold on;
crisp_output = evalmf(x_axis, output(1), 'constant');
y_label = ["r;Salida clásica = " num2str(output(1)) "%"];
plot (x_axis, crisp_output, y_label, 'LineWidth', 2);
ylim ([-0.1, 1.1]);
xlabel ('Propina', 'FontWeight', 'bold');
grid;
hold;
% Evalúa el FIS;
fprintf('Propina = %6.0f',output(1));disp(' ');
resp=input('Otro cálculo con el SID (S/N)? ','s');disp(' ')
resp=upper(resp);
close all;
endwhile

```

- **Código del Sistema de Inferencia Difusa,**

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_1_Mandani.fis
## Última modificación: 11 mar 2022
#
[System]
Name='Calculadora de propinas Mamdani'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=4
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='Calidad-Comida'
Range=[1 10]
NumMFs=2
MF1='Mala':'trapmf',[0 1 3 7]
MF2='Buena':'trapmf',[3 7 10 11]

[Input2]
Name='Servicio'
Range=[1 10]
NumMFs=2
MF1='Malo':'trapmf',[0 1 3 7]
MF2='Bueno':'trapmf',[3 7 10 11]

[Output1]

```

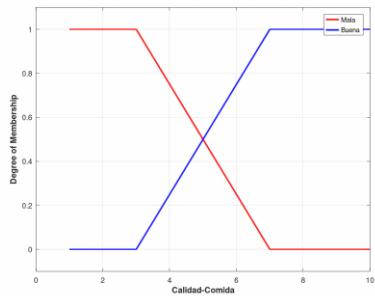
Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

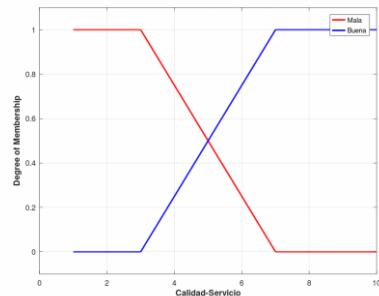
```
Name='Propina'
Range=[0 30]
NumMFs=3
MF1='Aprox-Cinco-Porciente':'gaussmf',[2 5]
MF2='Aprox-Diez-Porciente':'gaussmf',[2 10]
MF3='Aprox-Veinte-Porciente':'gaussmf',[2 20]
```

```
[Rules]
1 1, 1 1 (1) : 1
1 2, 2 2 (1) : 1
2 1, 2 2 (1) : 1
2 2, 3 3 (1) : 1
```

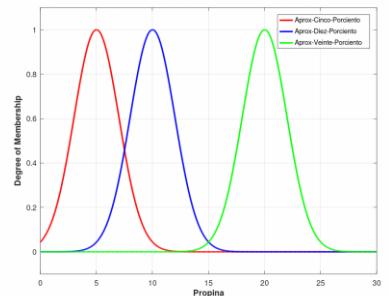
- **Resultados del código Octave,**



Antecedente. Entrada 1



Antecedente. Entrada 2



Consecuente. Salida

Reglas de la calculadora de propinas Mamdani:

1. If (Calidad-Comida is Mala) and (Calidad-Servicio is Mala), then (Propina is Aprox-5%) (1)
2. If (Calidad-Comida is Mala) and (Calidad-Servicio is Buena), then (Propina is Aprox-10%) (1)
3. If (Calidad-Comida is Buena) and (Calidad-Servicio is Mala), then (Propina is Aprox-10%) (1)
4. If (Calidad-Comida is Buena) and (Calidad-Servicio is Buena), then (Propina is Aprox-20%) (1)

Ingresar los valores de Calidad de comida y del Servicio para evaluar con el Sistema de Inferencia Difusa (SID)

CC, Calidad de la Comida (1-10)? 6
CS, Calidad del Servicio (1-10)? 4

sol = 11.660

Otro cálculo con el SID (S/N)? n
=>

Como se aprecia, si $CC = 6$ y $CS=4$, entonces la propina es de aproximadamente 12%.

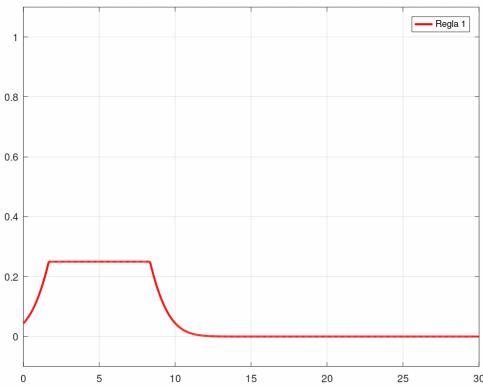
Si se desea otra alternativa para CC y CS , se responde S y se hacen otros cálculos. O se responde N y se termina el programa.

Gráficas de defuzzificación resultantes usando el método del centroide con el código Octave.

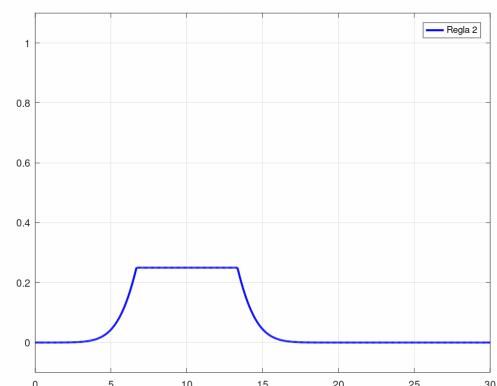
Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

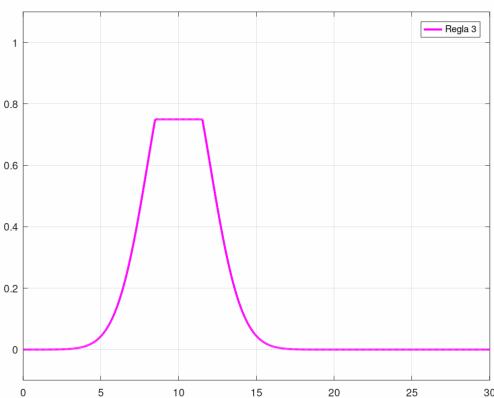
2022.



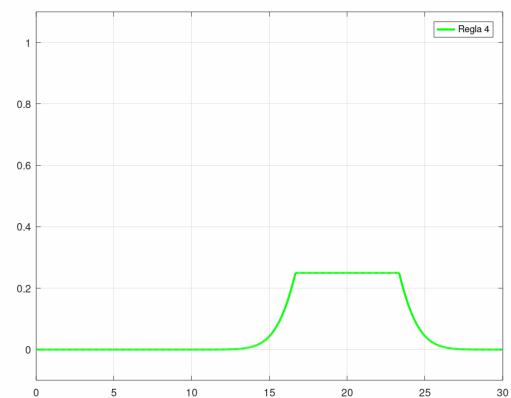
Mamdani. Regla 1



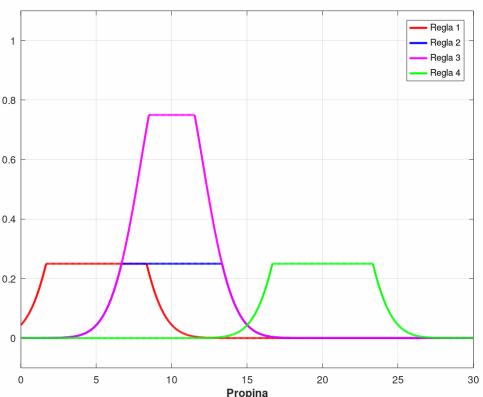
Mamdani. Regla 2



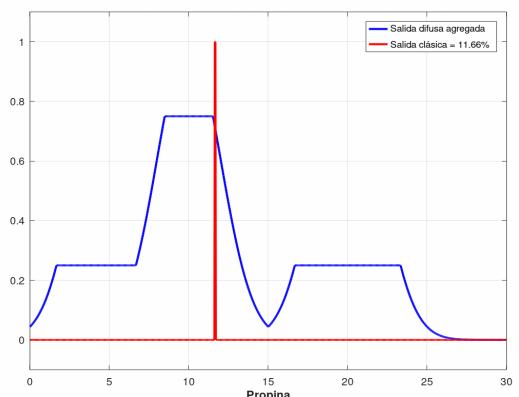
Mamdani. Regla 3



Mamdani. Regla 4



Mamdani. Todas las reglas



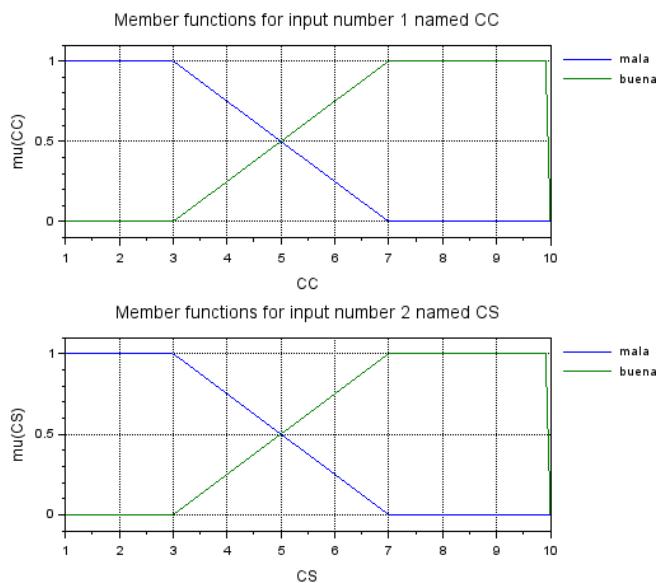
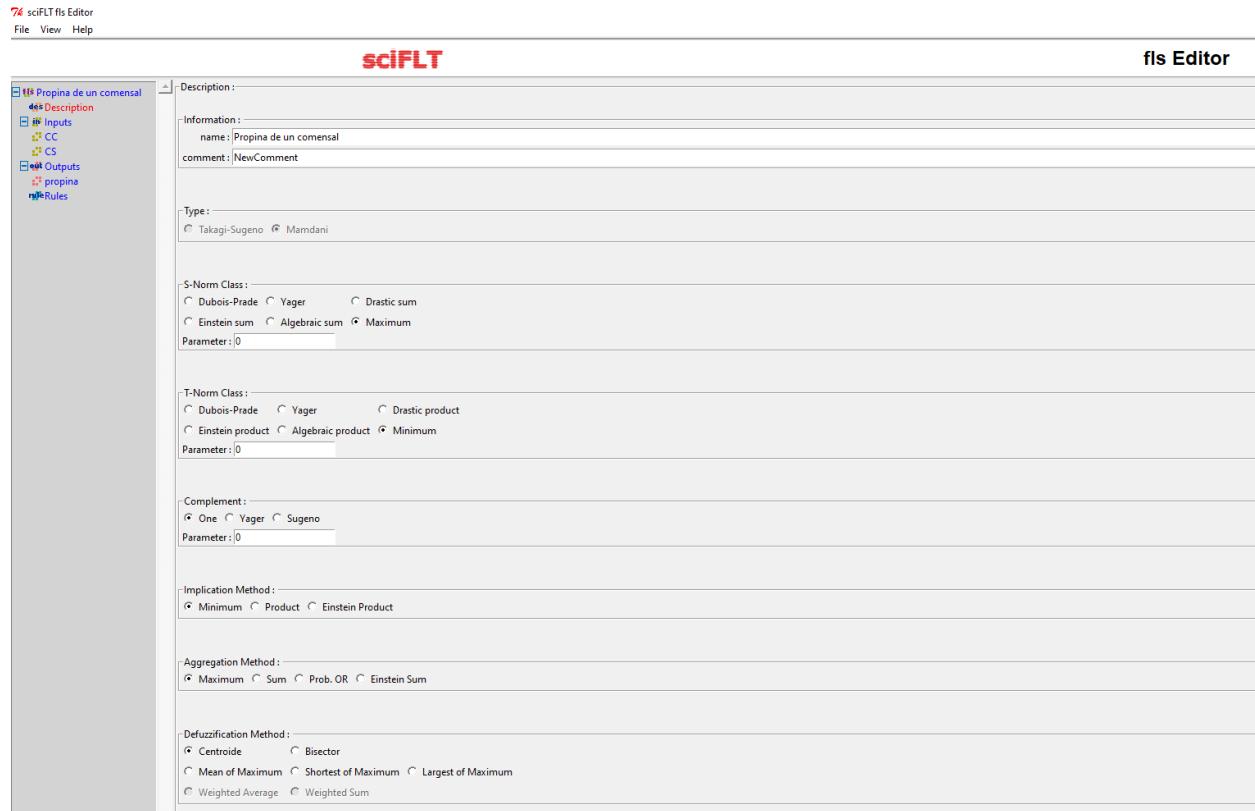
Mandani, centroide

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

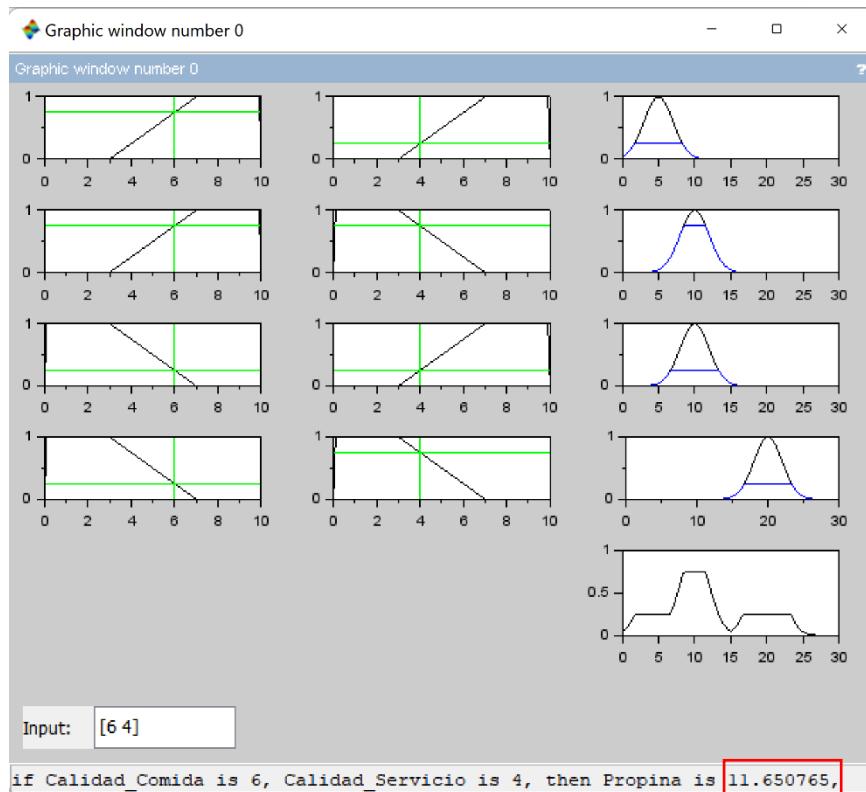
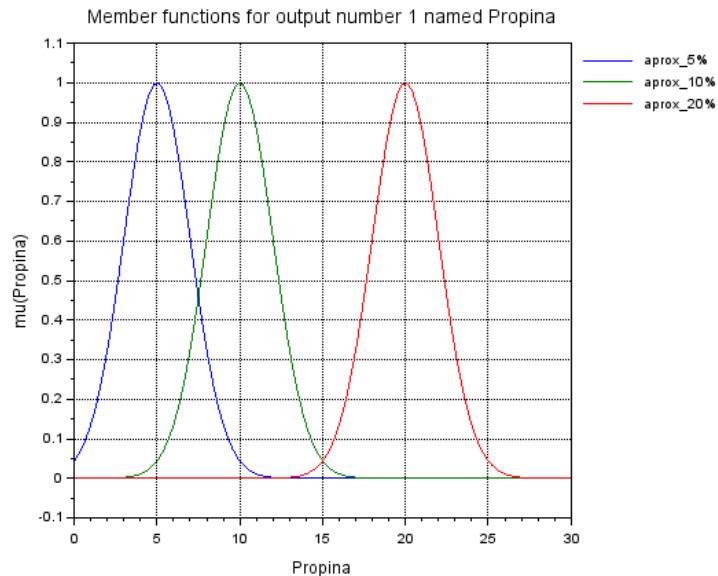
c) Resolviendo con Scilab.



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



Sugeno

a) Resolviendo con Octave.

○ Código

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_1_Sugeno.m
## Última modificación: 10 mar 2022
#
# OJO: Cargar previamente "pkg load fuzzy-logic-toolkit";
clc; close all;
## Lectura del archivo FIS.
fis = readfis ('Ejemplo_1_Sugeno.fis');
## Grafica las funciones de membresía de entrada y salida.
plotmf (fis, 'input', 1);
plotmf (fis, 'input', 2);
plotmf (fis, 'output', 1);
resp='S'
while (resp=='S')
    clc
    ## Mostrar las reglas en formato simbólico.
    puts ("\nReglas de la calculadora de propinas Mamdani:\n\n");
    ## Muestra las reglas.
    showrule (fis);
    ## Calcula la propina.
    puts ("\n\n\nIngresar los valores de Calidad de comida y del Servicio");
    puts ("\n para evaluar con el Sistema de Inferencia Difusa (SID)\n\n");
    CC = input(' CC, Calidad de la Comida (1-10)? ');
    CS = input(' CS, Calidad del Servicio (1-10)? '); disp(' ');disp(' ');
    propina=evalfis([CC CS], fis, 1001);
    fprintf('* Propina pobre    = %6.0f',propina(1));fprintf('\n');
    disp(' ');
    resp=input('Otro cálculo con el SID (S/N)? ','s');disp(' ');
    resp=upper(resp);
    close all;
endwhile

```

○ Sistema de inferencia difusa, FIS

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_1_Sugeno.fis
## Última modificación: 11 mar 2022
# [System]
Name = 'Ejemplo_1_Sugeno'
Type = 'sugeno'
Version = 1.0
NumInputs = 2
NumOutputs = 1
NumRules = 4
AndMethod = 'prod'
OrMethod = 'probOr'
ImpMethod = 'min'
AggMethod = 'max'
DefuzzMethod = 'centroid'
[Input1]
Name = 'Calidad-Alimento'
Range = [0 10]
NumMFs = 2
MF1 = 'Mala' : 'trapmf', [-3 0 3 7]
MF2 = 'Buena' : 'trapmf', [3 7 10 11]

```

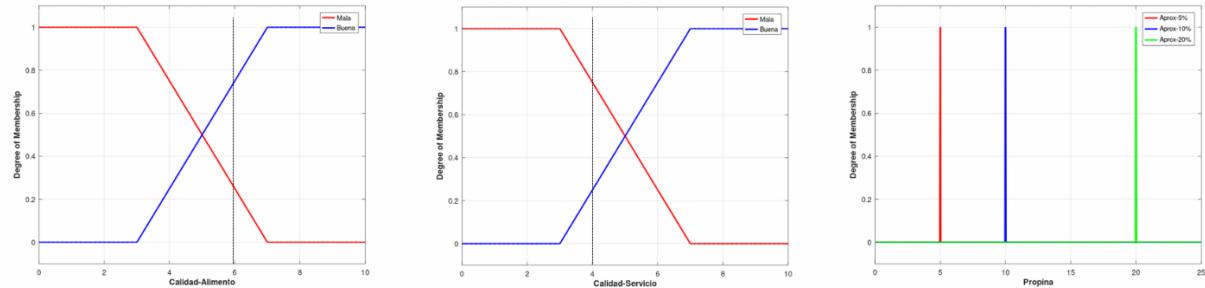
Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

```
[Input2]
Name = 'Servicio'
Range = [0 10]
NumMFs = 2
MF1 = 'Malo' : 'trapmf', [-3 0 3 7]
MF2 = 'Bueno' : 'trapmf', [3 7 10 11]
[Output1]
Name = 'Propina'
Range = [0 25]
NumMFs = 3
MF1 = 'Aprox-5%' : 'constant', [5]
MF2 = 'Aprox-10%' : 'constant', [10]
MF3 = 'Aprox-20%' : 'constant', [20]
[Rules]
1 1, 1 (1) : 1
2 1, 2 (1) : 1
1 2, 2 (1) : 1
2 2, 3 (1) : 1
```

○ Resultados.



Reglas de la calculadora de propinas Mamdani:

1. If (Calidad-Alimento is Malo) and (Calidad-Servicio is Malo), then (Propina is Aprox-5%) (1)
2. If (Calidad-Alimento is Buena) and (Calidad-Servicio is Malo), then (Propina is Aprox-10%) (1)
3. If (Calidad-Alimento is Malo) and (Calidad-Servicio is Buena), then (Propina is Aprox-10%) (1)
4. If (Calidad-Alimento is Buena) and (Calidad-Servicio is Buena), then (Propina is Aprox-20%) (1)

Ingresar los valores de Calidad de comida y del Servicio

para evaluar con el Sistema de Inferencia Difusa (SID)

CC, Calidad de la Comida (1-10)? 6

CS, Calidad del Servicio (1-10)? 4

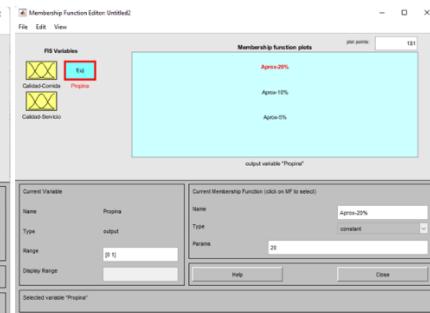
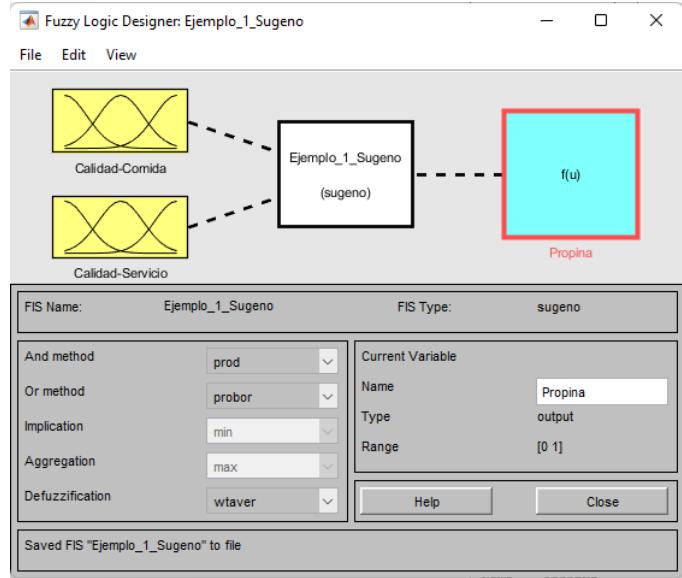
* Propina pobre = **11**

Otro cálculo con el SID (S/N) ?

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

b) Resolviendo con Matlab.



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



Ejemplo 2. Segundo problema básico de dar propina**Mandani**

Este ejemplo utiliza un problema crítico dos entradas, una salida basado en prácticas de dar propina en los Estados Unidos.

¿Dado un número entre 0 y 10 que representa la calidad del servicio, CS , en un restaurante (donde 10 es excelente) y otro número entre 0 y 10 que representa la calidad de la comida, CC , en ese restaurante (otra vez, 10 es excelente), cuál debería ser la propina? El punto de partida es escribir las tres reglas de oro de la asignación de propina:

1. Si la CC es *mala* o la CS es *mala*, entonces la *Propina* es *poca*.
2. Si la CC es *bueno*, entonces la *Propina* es *promedio*.
3. Si la CC es *excelente* o la CS es *excelente*, entonces la *Propina* es *generosa*.

Suponer que la propina promedio es 10%, una propina generosa es 20% y una propina baja es 5%.

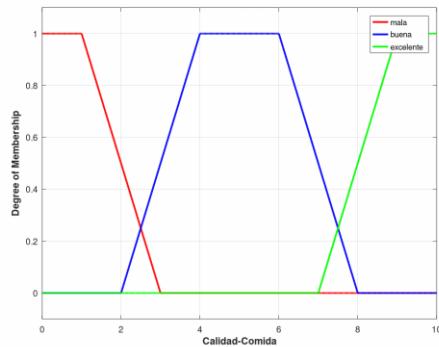
Solución.

Inicialmente se almacenan los conjuntos difusos de las variables de entrada, en este caso CC y $propina$.

La variable de entrada CC tiene dos valores lingüísticos: mala y buena. Y se comportan como una función de membresía trapezoidal con los siguientes parámetros:

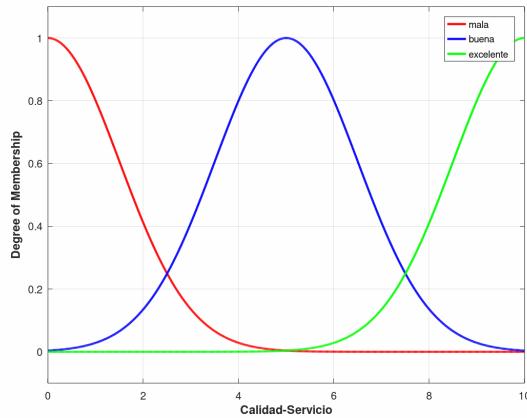
Calidad Comida, CC:

Nombre:	Función de membresía:	a	b	c	d
mala	Trapezial	0	0	1	3
bueno	Trapezial	2	4	6	8
excelente	Trapezial	7	9	10	10



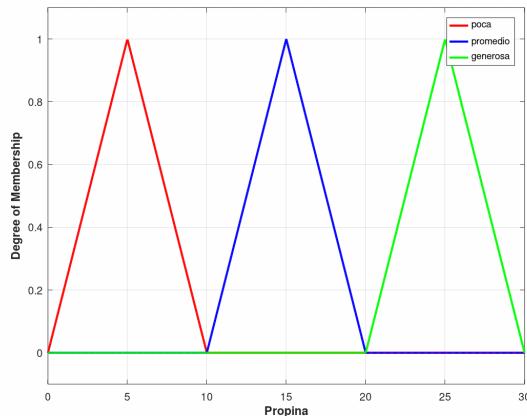
Calidad Servicio, CS:

Nombre:	Función de membresía:	Parámetro de escala (desviación estándar)	Parámetro de posición (media):
mala	Normal (Gauss)	1.5	0
bueno	Normal (Gauss)	1.5	5
excelente	Normal (Gauss)	1.5	10



Propina:

Nombre:	Función de membresía:	a	b	c
pobre	Triangular	0	5	10
promedio	Triangular	10	15	20
generosa	Triangular	20	25	30



a) Resolviendo con Octave.○ **Código,**

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_2_Mandani.m
## Última modificación: 11 mar 2022
#
# OJO: Cargar previamente "pkg load fuzzy-logic-toolkit";
clc; clear; close all;
fis=readfis ('Calculadora de propinas Mamdani 1');
## Trazar las funciones de pertenencia de entrada y salida.
plotmf (fis, 'input', 1)
plotmf (fis, 'input', 2);
plotmf (fis, 'output', 1);
## Graficar la propina y el pago + propina como funciones de la calidad de los
## Alimentos y Servicio.
%gensurf (fis, [1 2], 1);
%gensurf (fis, [1 2], 2);
resp='S';
while (resp=='S')
    clc
    ## Mostrar las reglas en formato simbólico.
    puts ("\nReglas de la calculadora de propinas Mamdani:\n\n");
    showrule(fis); disp(' ');disp(' ');
    ## Calcule la propina y el cheque + propina usando
    ## (Calidad Comida, Servicio.
    puts ("\nIngresar los valores de Calidad de Comida y del Servicio");
    puts ("\n para evaluar con el Sistema de Inferencia Difusa (SID)\n\n");
    CC = input('CC, Calidad de la Comida (1-10)? ');
    CS = input('CS, Calidad del Servicio (1-10)? '); disp(' ');disp(' ');
    [output, rule_input, rule_output, fuzzy_output] = ...
    evalfis ([CC CS], fis, 1001);
    ## Trazar la primera salida (Consejo) de las reglas difusas individuales
    ## en un juego de ejes.
    x_axis = linspace (fis.output(1).range(1), ...
                       fis.output(1).range(2), 1001);
    colors = ['r' 'b' 'm' 'g'];
    figure ('NumberTitle', 'off', 'Name', ...
            'Output of Fuzzy Rules 1-4 for Input = (CC, CS)');
    for i = 1 : 3
        y_label = [colors(i) ";Regla " num2str(i) ";"];
        plot (x_axis, rule_output(:,i), y_label, 'LineWidth', 2);
        ylim ([0, 1]);
        grid;
        figure();
    endfor
    for i = 1 : 3
        y_label = [colors(i) ";Regla " num2str(i) ";"];
        plot (x_axis, rule_output(:,i), y_label, 'LineWidth', 2);
        grid; hold on;
    endfor
    grid;
    ylim ([0, 1]);
    xlabel ('Propina', 'FontWeight', 'bold');
    grid; hold;
    ## Trazar la primera salida difusa agregada y la primera salida difusa
    ## (Propina) en un juego de ejes.
    figure('NumberTitle', 'off', 'Name', ...
           'Agregación y defuzzificación para la entrada = (CC, CS)');
    plot (x_axis, fuzzy_output(:, 1), "b;Salida difusa agregada;", ...
           'LineWidth', 2);
    hold on;
    crisp_output = evalmf(x_axis, output(1), 'constant');
    y_label = ["r;Salida clásica = " num2str(output(1)) "%;"];
    plot (x_axis, crisp_output, y_label, 'LineWidth', 2);
    ylim ([0, 1]);
    xlabel ('Propina', 'FontWeight', 'bold');
    grid;
    hold;
    % Evalúa el FIS;
    sol=output(1)
    disp(' ');

```

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

```
resp=input('Otro cálculo con el SID (S/N)? ','s');disp(' ')
resp=upper(resp);
close all;
endwhile
```

○ Código del sistema de inferencia difusa,

```
## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_2_Mandani.fis
## Última modificación: 11 mar 2022
#
[System]
Name='Calculadora de propinas Mamdani 1'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=3
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

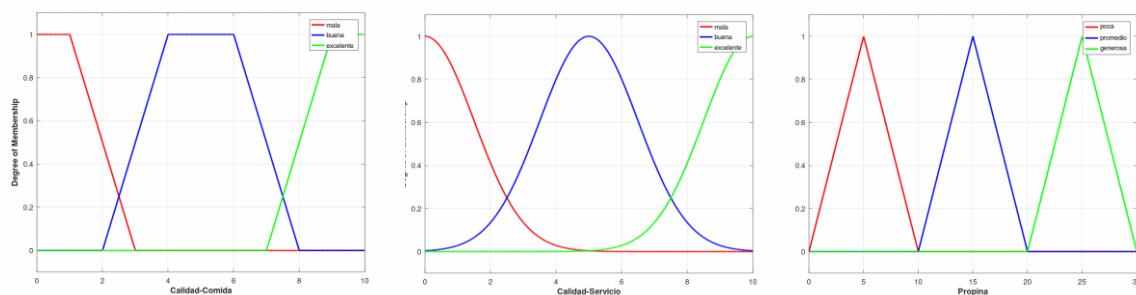
[Input1]
Name='Calidad-Comida'
Range=[0 10]
NumMFs=3
MF1='mala':'trapmf',[ -1 0 1 3]
MF2='buena':'trapmf',[2 4 6 8]
MF3='excelente':'trapmf',[7 9 10 11]

[Input2]
Name='Calidad-Servicio'
Range=[0 10]
NumMFs=3
MF1='mala':'gaussmf',[1.5 0]
MF2='buena':'gaussmf',[1.5 5]
MF3='excelente':'gaussmf',[1.5 10]

[Output1]
Name='Propina'
Range=[0 30]
NumMFs=3
MF1='poca':'trimf',[0 5 10]
MF2='promedio':'trimf',[10 15 20]
MF3='generosa':'trimf',[20 25 30]

[Rules]
1 1, 1 (1) : 2
1 0, 2 (1) : 1
3 3, 3 (1) : 2
```

○ Resultados del código Octave,



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

Reglas de la calculadora de propinas Mamdani:

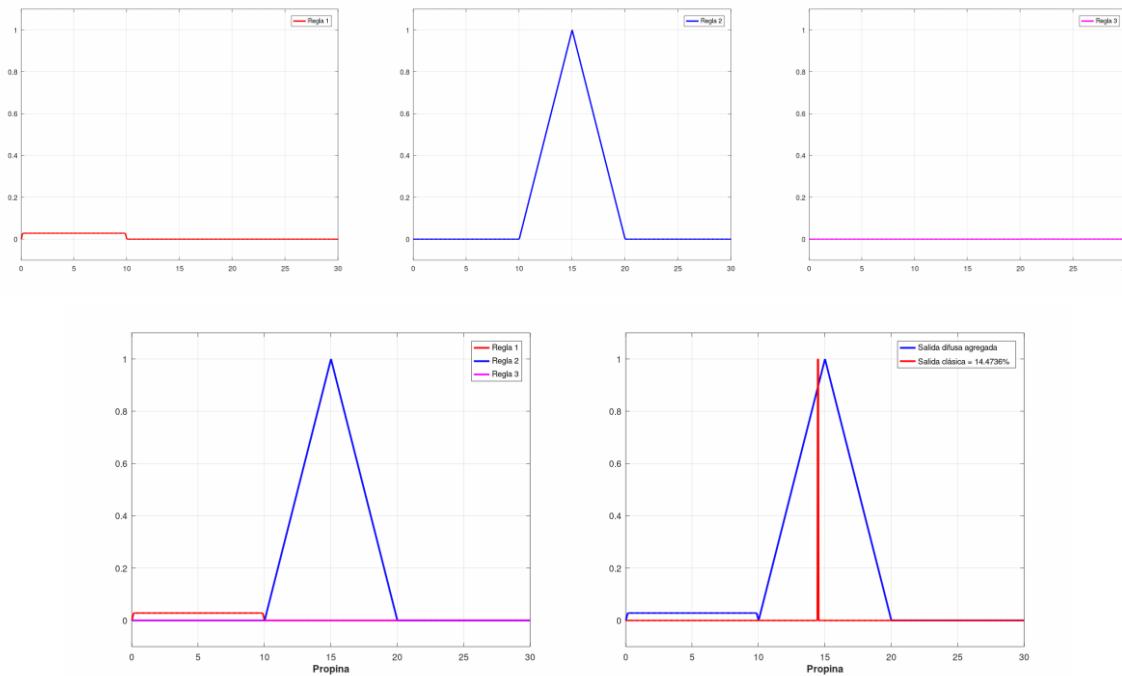
1. If (Calidad-Comida is mala) or (Calidad-Servicio is mala), then (Propina is poca) (1)
2. If (Calidad-Servicio is buena), then (Propina is promedio) (1)
3. If (Calidad-Comida is excelente) or (Calidad-Servicio is excelente), then (Propina is generosa) (1)

Ingresar los valores de Calidad de Comida y del Servicio para evaluar con el Sistema de Inferencia Difusa (SID)

CC, Calidad de la Comida (1-10)? 6
CS, Calidad del Servicio (1-10)? 4

sol = 14.474

Otro cálculo con el SID (S/N)? n
>>



Como se aprecia, si $CC = 6$ y $CS = 4$, entonces la propina es de aproximadamente 14%.

Si se desea otra alternativa para CC y CS , se responde S y se hacen otros cálculos. O se responde N y se termina el programa.

b) Resolviendo con R,○ **Código de R:**

```
### SEGUNDO EJEMPLO
#
graphics.off()
rm(list = ls())
cat("\014")
library(sets)
## define el universo del discurso
sets_options("universe", seq(0, 30, 0.5))
## establece las variables difusas
variables <- set(
  comida = fuzzy_variable(mala =
    fuzzy_trapezoid(corners = c(-1, 0, 1, 3)),
    buena =
      fuzzy_trapezoid(corners = c(2, 4, 6, 8)),
    excelente =
      fuzzy_trapezoid(corners = c(7, 9, 10, 11))),
  servicio = fuzzy_partition(varnames = c(malo = 0, bueno = 5, excelente = 10),
    sd = 1.5),
  propina = fuzzy_partition(varnames = c(poca = 5, promedio = 15, generosa = 25),
    FUN = fuzzy_cone, radius = 5)
)
## establece las reglas
rules <- set(
  fuzzy_rule(comida %is% mala || servicio %is% malo, propina %is% poca),
  fuzzy_rule(comida %is% buena, propina %is% promedio),
  fuzzy_rule(comida %is% excelente || servicio %is% excelente, propina %is% generosa)
)
## combinar en un sistema
modelo <- fuzzy_system(variables, rules)
print(modelo)
plot(modelo)
## hace la inferencia
ejemplo <- fuzzy_inference(modelo, list(comida = 6, servicio = 4))
## defuzifica
gset_defuzzify(ejemplo, "centroid")
## grafica resultados del conjunto difuso
plot(ejemplo)
sets_options("universe", NULL) # Restablecer el universo del discurso
```

○ **Resultados con código R,**

```
> library(sets)
> ## define el universo del discurso
> sets_options("universe", seq(0, 30, 0.5))
> ## establece las variables difusas
> variables <- set(
+   comida = fuzzy_variable(mala =
+     fuzzy_trapezoid(corners = c(-1, 0, 1, 3)),
+     buena =
+       fuzzy_trapezoid(corners = c(2, 4, 6, 8)),
+     excelente =
+       fuzzy_trapezoid(corners = c(7, 9, 10, 11))),
+   servicio = fuzzy_partition(varnames = c(malo = 0, bueno = 5, excelente = 10),
+     sd = 1.5),
+   propina = fuzzy_partition(varnames = c(poca = 5, promedio = 15, generosa = 25),
+     FUN = fuzzy_cone, radius = 5)
+ )
> # ## establece las reglas
> rules <- set(
+   fuzzy_rule(comida %is% mala || servicio %is% malo, propina %is% poca),
+   fuzzy_rule(comida %is% buena, propina %is% promedio),
+   fuzzy_rule(comida %is% excelente || servicio %is% excelente, propina %is% generosa)
+ )
> ## combinar en un sistema
> modelo <- fuzzy_system(variables, rules)
> print(modelo)
```

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

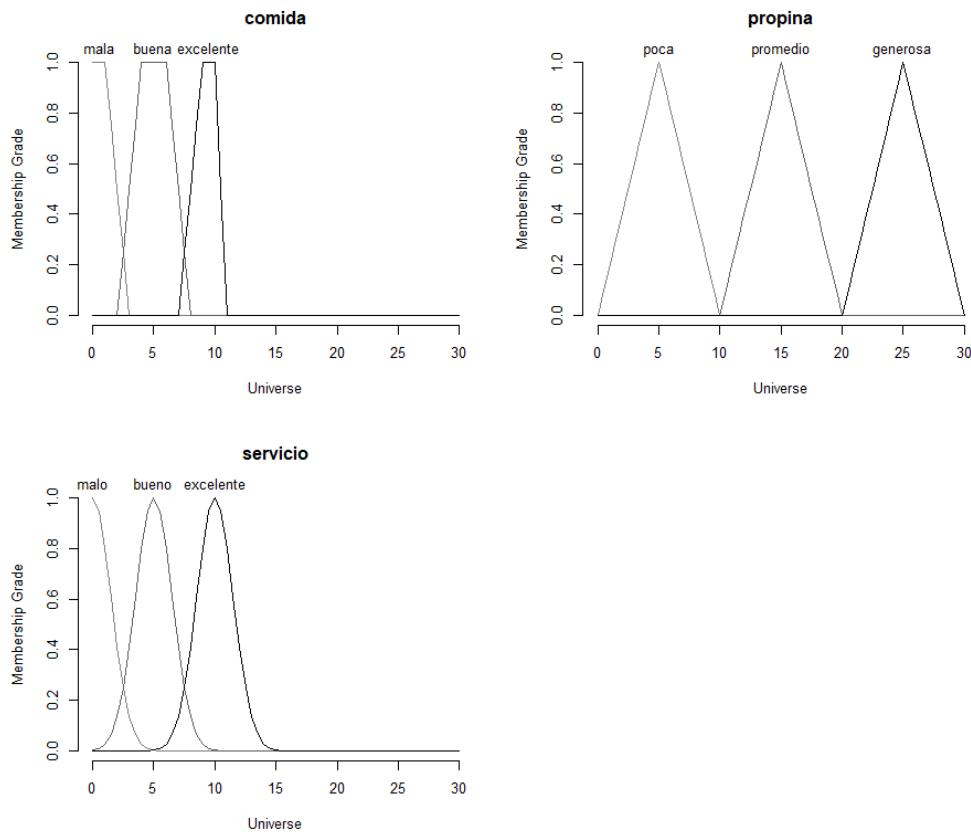
A fuzzy system consisting of 3 variables and 3 rules.

Variables:

```
comida(mala, buena, excelente)
propina(poca, promedio, generosa)
servicio(malo, bueno, excelente)
```

Rules:

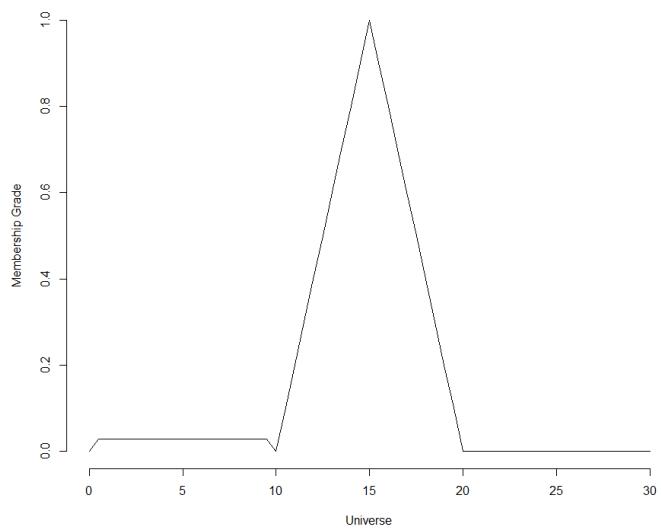
```
comida %is% mala || servicio %is% malo => propina %is% poca
comida %is% excelente || servicio %is% excelente => propina %is% generosa
comida %is% buena => propina %is% promedio
> plot(modelo)
> ## hace la inferencia
> ejemplo <- fuzzy_inference(modelo, list(comida = 6, servicio = 4))
> ## defuzzifica
> gset_defuzzify(ejemplo, "centroid")
[1] 14.49155
> ## grafica resultados del conjunto difuso
> plot(ejemplo)
> sets_options("universe", NULL) # Restablecer el universo del discurso
>
```



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



Sugeno

a) Resolviendo con Octave.

- **Código,**

```
## Directorio:      c:/Fuzzy Sets/2022
## Archivo:        Ejemplo_2_Sugeno.m
## Última modificación: 11 mar 2022
#
# OJO: Cargar previamente "pkg load fuzzy-logic-toolkit";
clc; close all;
## Lectura del archivo FIS.
fis = readfis ('Ejemplo_2_Sugeno.fis');
## Grafica las funciones de membresía de entrada y salida.
plotmf (fis, 'input', 1);
plotmf (fis, 'input', 2);
plotmf (fis, 'output', 1);
resp='S'
while (resp=='S')
    clc
    ## Mostrar las reglas en formato simbólico.
    puts ("\nReglas de la calculadora de propinas Mamdani:\n\n");
    ## Muestra las reglas.
    showrule (fis);
    ## Calcula la propina.
    puts ("\n\n\nIngresar los valores de Calidad de comida y del Servicio");
    puts ("\n para evaluar con el Sistema de Inferencia Difusa (SID)\n\n");
    CC = input(' CC, Calidad de la Comida (1-10)? ');
    CS = input(' CS, Calidad del Servicio (1-10)? '); disp(' '); disp(' ');
    propina=evalfis([CC CS], fis, 1001);
    fprintf('* Propina pobre    = %6.0f',propina(1));fprintf('\n');
    disp(' ');
    resp=input('Otro cálculo con el SID (S/N)? ','s');disp(' ')
    resp=upper(resp);
    close all;
endwhile
```

- **Código del sistema de inferencia difusa,**

```
## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_2_Sugeno.fis
## Última modificación: 11 mar 2022
#
[System]
Name='Ejemplo_2_Sugeno'
Type='sugeno'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=3
AndMethod = 'prod'
OrMethod = 'probOr'
ImpMethod = 'min'
AggMethod = 'max'
DefuzzMethod = 'centroid'

[Input1]
Name='Calidad-Comida'
Range=[1 10]
NumMFs=2
MF1='Mala':'trapmf',[0 1 3 7]
MF2='Buena':'trapmf',[3 7 10 11]

[Input2]
Name='Calidad-Servicio'
Range=[1 10]
NumMFs=2
MF1='Mala':'trapmf',[0 1 3 7]
MF2='Buena':'trapmf',[3 7 10 11]
```

Introducción al Sistema de Inferencia Difusa usando Octave

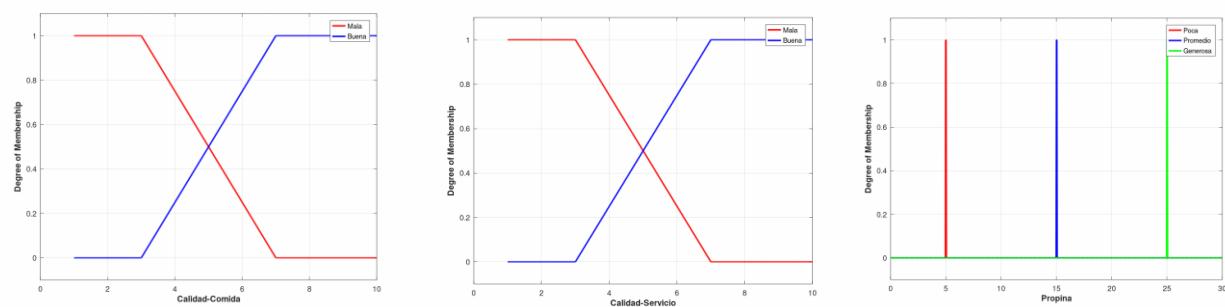
Salvador Díaz Maldonado

2022.

```
[Output1]
Name='Propina'
Range=[0 30]
NumMFs=3
MF1 = 'Poca' : 'constant', [5]
MF2 = 'Promedio': 'constant', [15]
MF3 = 'Generosa': 'constant', [25]

[Rules]
1 1, 1(1) : 1
2 0, 2(1) : 1
2 2, 3(1) : 1
```

○ **Resultados,**



Reglas de la calculadora de propinas Mamdani:

1. If (Calidad-Comida is Mala) and (Calidad-Servicio is Mala), then (Propina is Poca) (1)
2. If (Calidad-Comida is Buena), then (Propina is Promedio) (1)
3. If (Calidad-Comida is Buena) and (Calidad-Servicio is Buena), then (Propina is Generosa) (1)

Ingresar los valores de Calidad de comida y del Servicio para evaluar con el Sistema de Inferencia Difusa (SID)

CC, Calidad de la Comida (1-10)? 6
CS, Calidad del Servicio (1-10)? 4

* Propina pobre = **15**

Otro cálculo con el SID (S/N)?

Ejemplo 3². Problema de aplicación: diseño de intercambiador de calor.**Mandani**

En el diseño de intercambiadores de calor, un análisis de flexibilidad requiere que el diseñador determine si el tamaño del intercambiador de calor es pequeño o grande. Para cuantificar esta vaguedad lingüística del tamaño se forma la ecuación de diseño general para un intercambiador de calor, $Q = AU T_{\log \text{ media}}$, donde es necesario determinar el coeficiente de transferencia de calor U y el área A . La figura 3.1 muestra un esquema de este intercambiador.

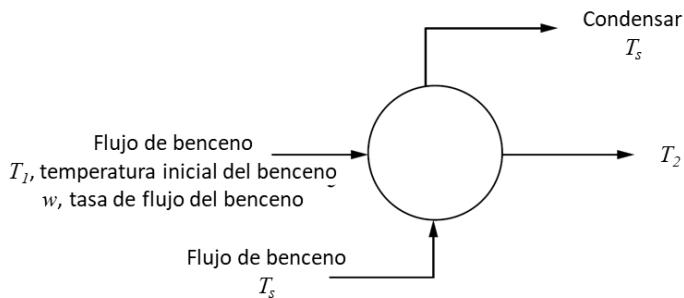


Figure 3.1. Diseño de intercambiador de calor.

$$AU = wC_p \ln \left(\frac{T_s - T_1}{\Delta T_{\text{app}}} \right)$$

donde C_p es la capacidad de calor del benceno (1.7543 kJ/°K kg) y $T_s - T_1 = 72.55$ °K.

Se modelará el flujo de benceno, w , en kg/s y la diferencia de temperatura (ΔT_{app}) en grados kelvin, como entradas, y se modelará el tamaño del intercambiador de calor como salida. Se deducirán las siguientes reglas de inferencia disyuntivas basadas en observaciones del modelo:

- Regla 1: IF w es *flujo grande* y ΔT_{app} es *diferencia pequeña*,
ENTONCES AU es *intercambiador de gran calor*.
- Regla 2: IF w es *flujo pequeño* o ΔT_{app} es *diferencia grande*,
ENTONCES AU es *intercambiador de poco calor*.
- Regla 3: IF w es *flujo pequeño* y ΔT_{app} es *aproximación pequeña*,
ENTONCES AU es *intercambiador de gran calor*.

² Ross, T.J. 2010. Fuzzy Logic with Engineering Applications. Third Edition. Wiley

El equivalente gráfico de estas reglas se muestra en la figura 3.2. Se empleará un método de defuzzificación promedio ponderado para comparar los resultados de un par de entrada para cada uno de los dos métodos de inferencia de Mamdani y de Sugeno.

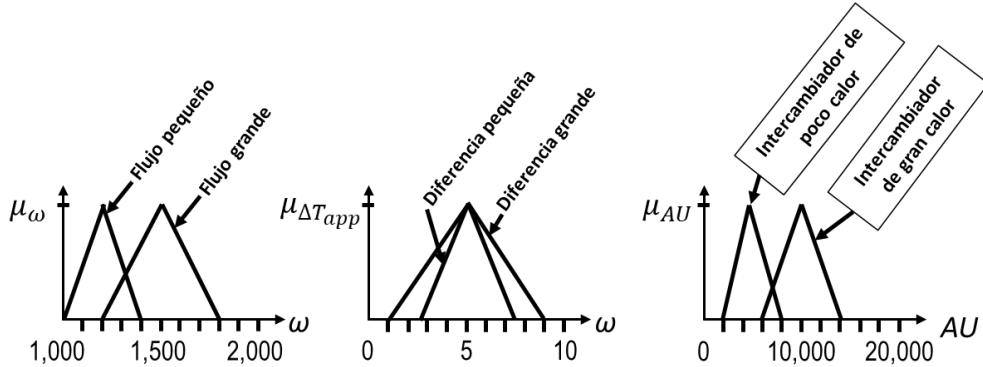


Figura 3.2. Entradas difusas (antecedente) y salida difusa (subsecuente) para utilizar el método Mamdani para las tres reglas.

Se ingresan dos valores clásicos de velocidad de flujo de benceno y temperatura aproximada:

$$w = 1300 \text{ kg/s} \quad \text{y} \quad \Delta T_{app} = 6.5^\circ\text{K}$$

Utilizando el método de inferencia de implicación máxima-mínima de Mamdani, se sabe que

$$\mu_{\tilde{B}^k}(AU) = \max_k \left\{ \min \left[\mu_{A_1^k}(w), \mu_{A_2^k}(\Delta T_{app}) \right] \right\}$$

Usando el enfoque gráfico se obtienen las reglas mostradas en la figura 3.3.

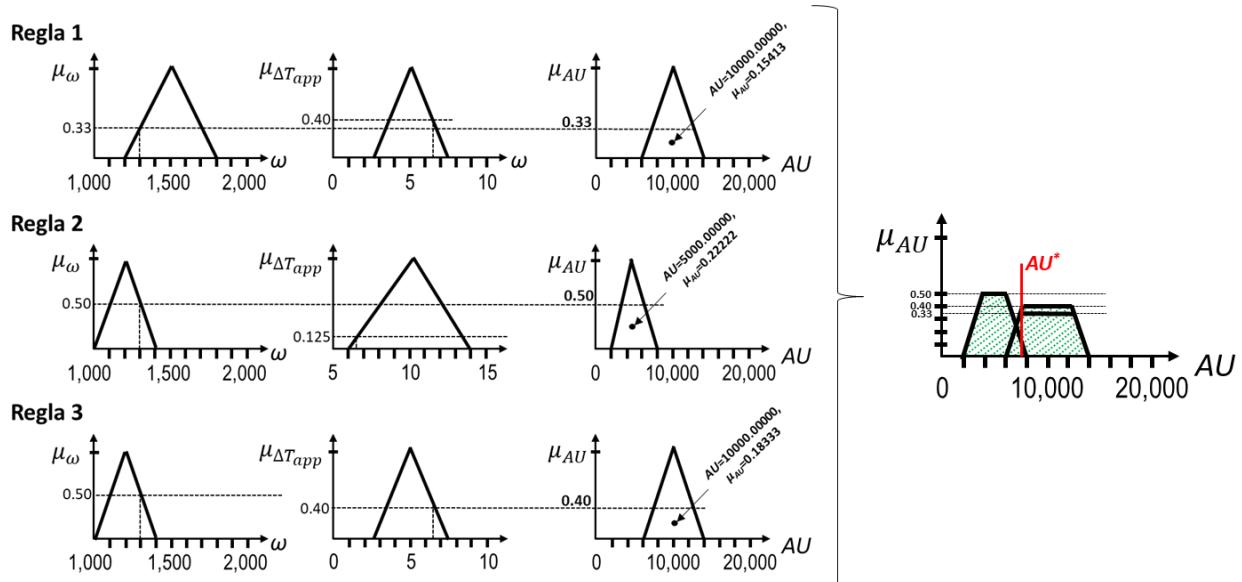
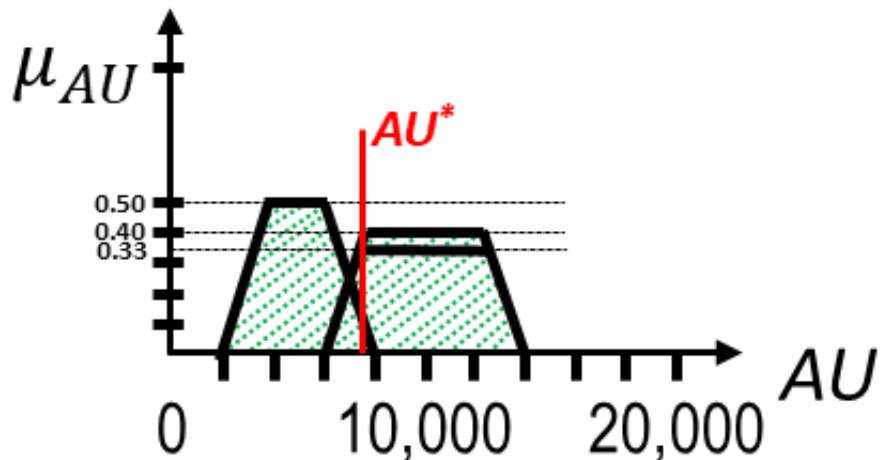


Figura 3.3. Resultado del paso de defuzzificación en el método Mamdani.

Para calcular el centroide (obteniendo las ecuaciones de los segmentos del perímetro de la sección izquierda de la figura 3.3), se aplica la ecuación²,

$$z^* = \frac{\int \mu_{\tilde{C}}(z) \cdot z dz}{\int \mu_{\tilde{C}}(z) dz}, \text{ es decir, } AU^* = \frac{\int \mu_{\tilde{C}}(AU) \cdot AU dAU}{\int \mu_{\tilde{C}}(AU) dAU}$$



Las integrales de cada segmento son (aproximadamente):

$$\int_{2000}^{3500} \left(\frac{0.1(AU - 2000)}{300} \right) (AU) d(AU) = 1,125,000$$

$$\int_{3500}^{6500} 0.5(AU) d(AU) = 7,500,000$$

$$\int_{6500}^{7140} (0.5 - 3.3333 \times 10^{-4}(AU - 6500))(AU) d(AU) = 1,710,012$$

$$\int_{7140}^{7600} (0.2867 + 2.4638 \times 10^{-4}(AU - 7140))(AU) d(AU) = 1,165,970$$

$$\int_{7600}^{12400} 0.4(AU) d(AU) = 19,200,000$$

$$\int_{12400}^{14000} (0.4 - 2.5 \times 10^{-4}(AU - 12400))(AU) d(AU) = 4,138,667$$

Entonces,

$$AU^* = \frac{\int \mu_{\tilde{C}}(z) \cdot z dz}{\int \mu_{\tilde{C}}(z) dz} = \frac{1,125,000 + 7,500,000 + 1,710,012 + 1,166,083 + 19,200,000 + 4,138,667}{375 + 1,500 + 252 + 158 + 1,920 + 320} \\ = \textcolor{red}{7,699.39}$$

Por otro lado, otro método de defusificación muy usado es el de *promedio ponderado*. Aplicando la ecuación² y recurriendo a la figura 3.3,

$$z^* = \frac{\sum(\mu_{\tilde{C}}(\bar{z}) \cdot \bar{z})}{\sum \mu_{\tilde{C}}(\bar{z})}$$

Regla 1: $\begin{cases} \mu(\omega) = 1/3 \rightarrow \left\{ \begin{array}{l} 7,333.33 \\ 12,666.67 \end{array} \right\} \cong 10,000.00 \\ \mu(\Delta T_{app}) = 0.4 \end{cases}$

Regla 2: $\begin{cases} \mu(\omega) = 0.50 \rightarrow \left\{ \begin{array}{l} 3,500.00 \\ 5,000.31 \end{array} \right\} \cong 4,250.16 \\ \mu(\Delta T_{app}) = 0.125 \end{cases}$

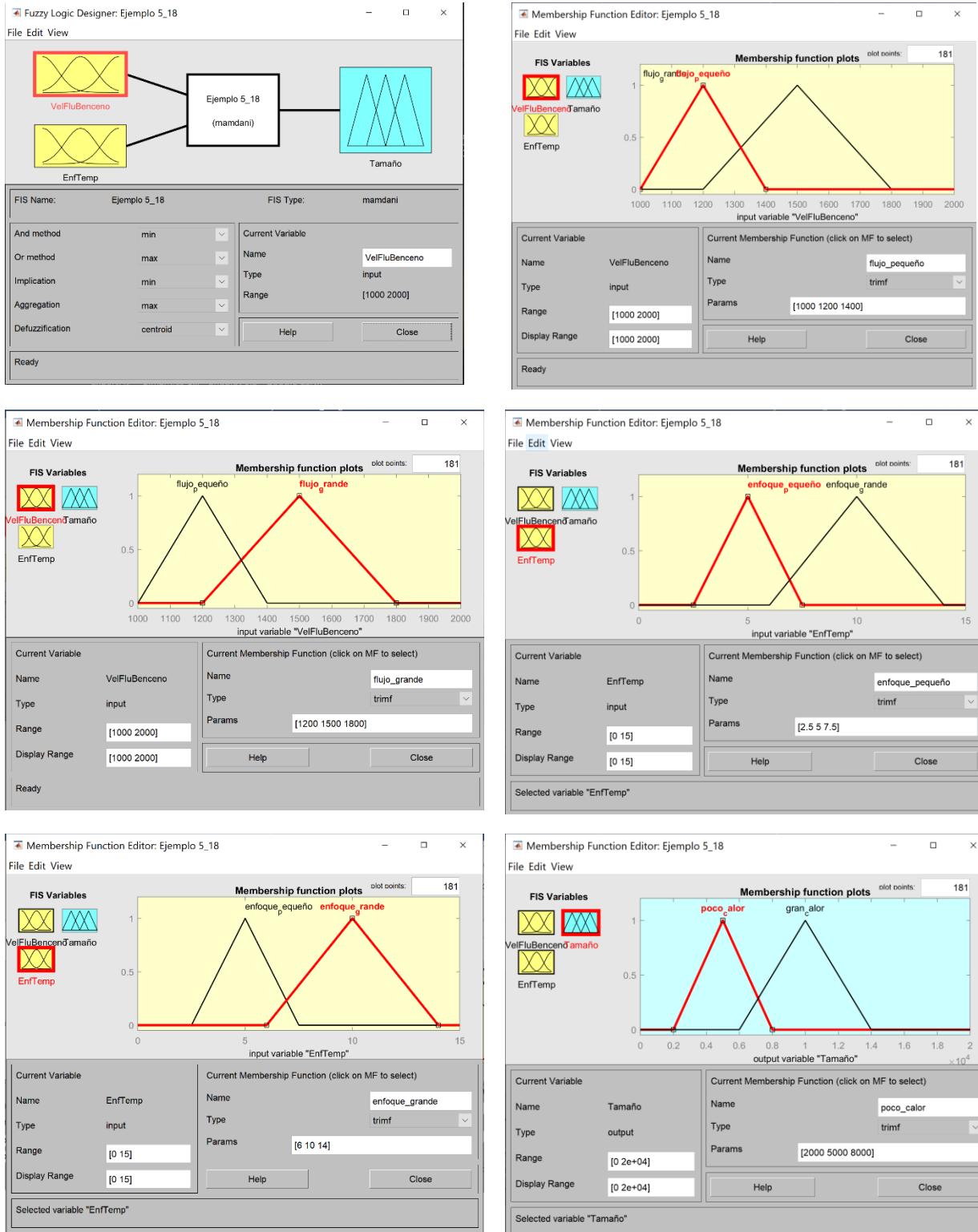
Regla 3: $\begin{cases} \mu(\omega) = 0.50 \\ \mu(\Delta T_{app}) = 0.40 \rightarrow \left\{ \begin{array}{l} 7,600.00 \\ 12,400.00 \end{array} \right\} \cong 10,000.00 \end{cases}$

$$z^* = \frac{\sum(\mu_{\tilde{C}}(\bar{z}) \cdot \bar{z})}{\sum \mu_{\tilde{C}}(\bar{z})} = \frac{0.15413(10000) + 0.22222(5000) + 0.18333(10000)}{0.33 + 0.5 + 0.40} = \textcolor{red}{7,972.97} \frac{\text{kW}}{\text{m}^2 \text{°K}}$$

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

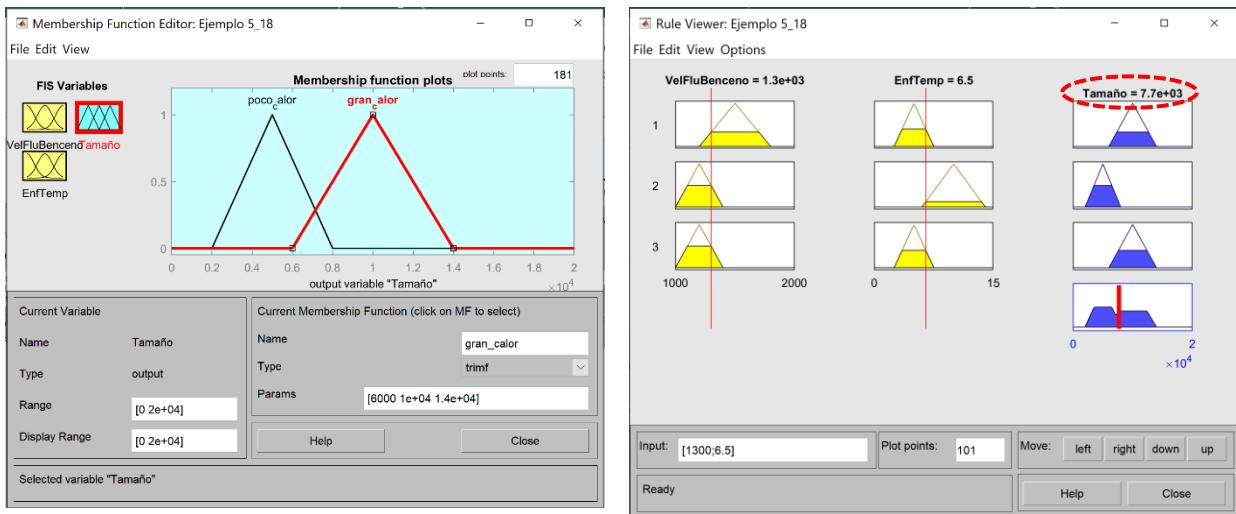
a) Resolviendo con Matlab.



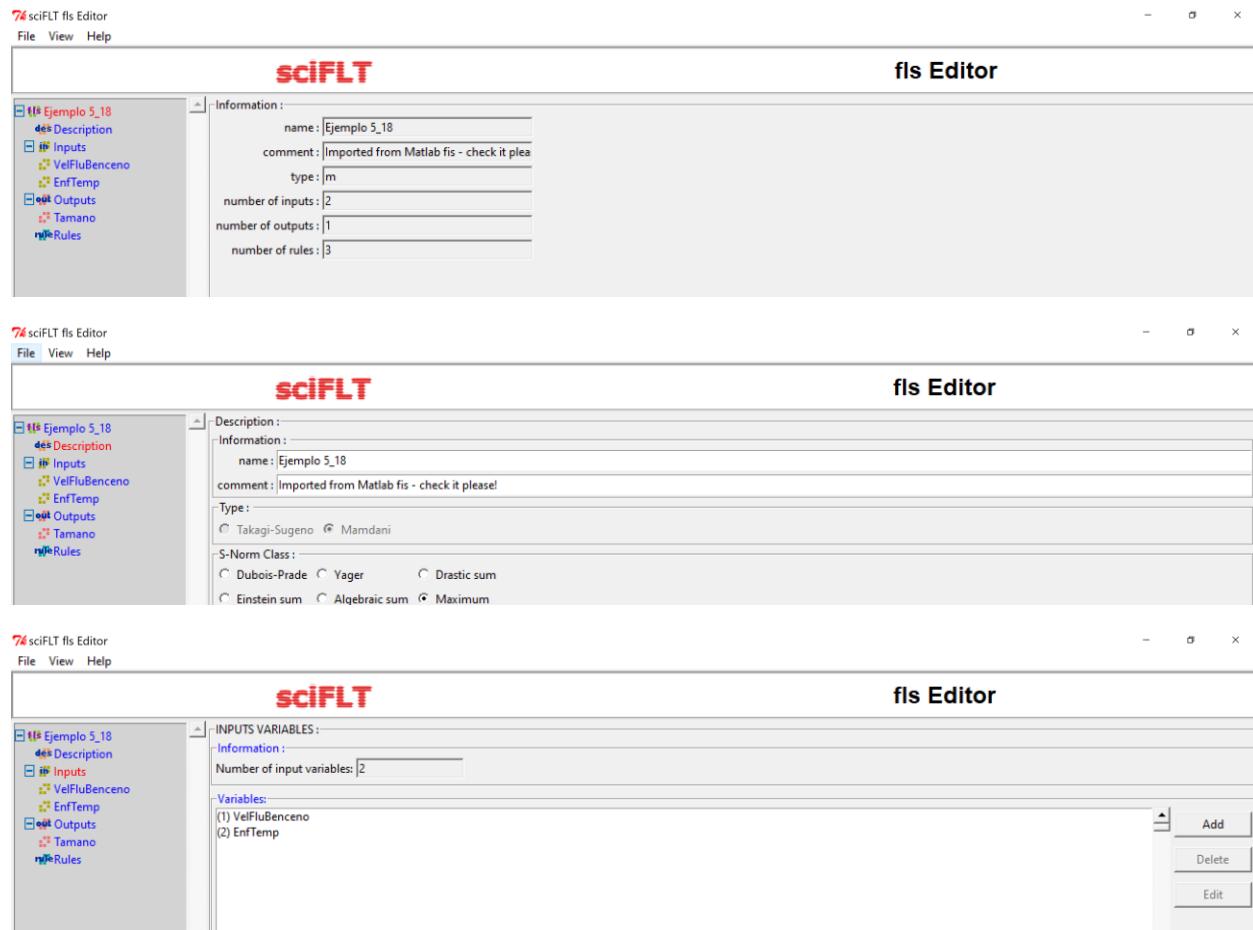
Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



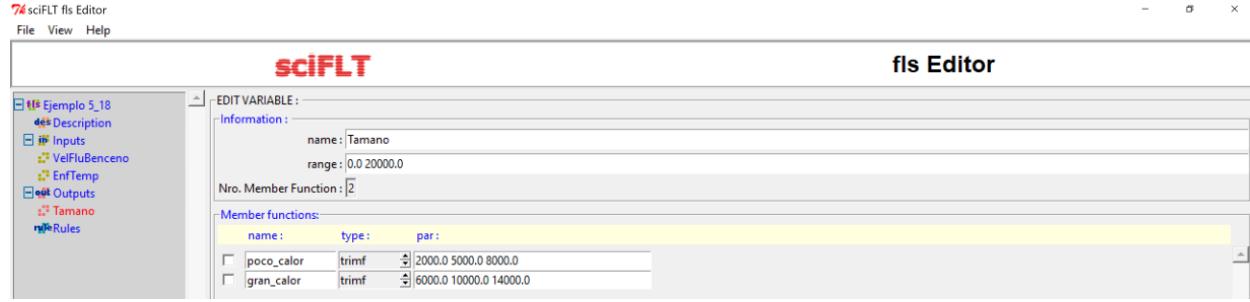
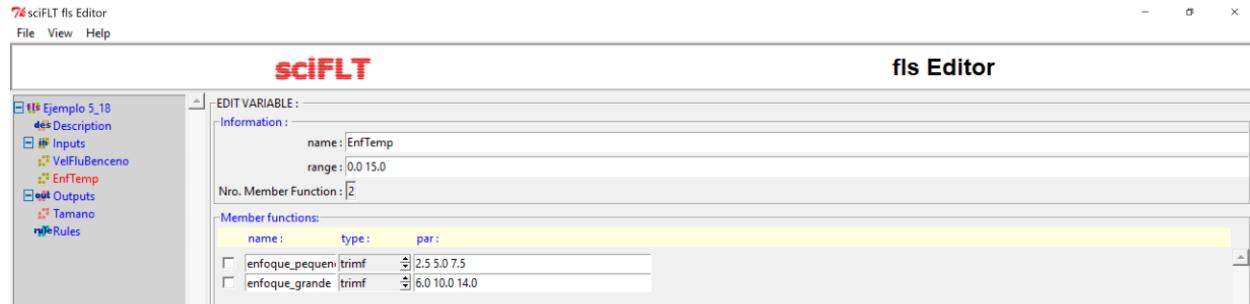
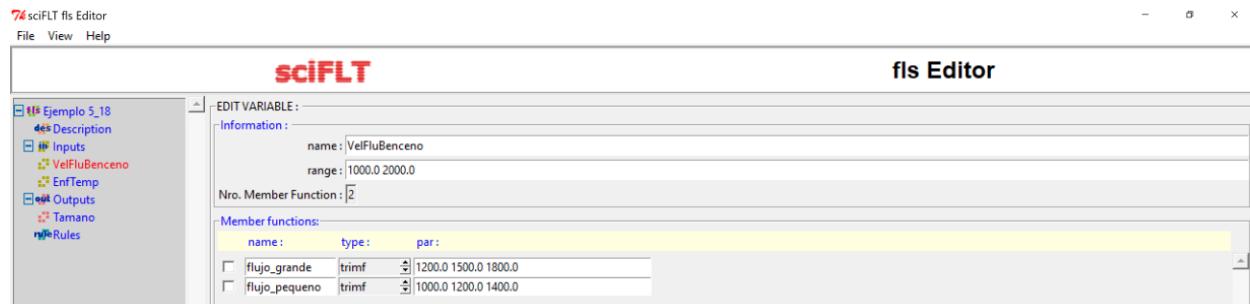
b) Resolviendo con Scilab:



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

sciFLT fIS Editor

File View Help

sciFLT

fIS Editor

Ejemplo 5_18

#Description

#Inputs

VelFluBenceno

#Outputs

Tamano

Rules

R1: IF (VelFluBenceno IS flujo_grande) AND (EnfTemp IS enfoque_pequeno) THEN (Tamano IS gran_calor) weight=1.0
R2: IF (VelFluBenceno IS flujo_pequeno) OR (EnfTemp IS enfoque_grande) THEN (Tamano IS poco_calor) weight=1.0
R3: IF (VelFluBenceno IS flujo_pequeno) AND (EnfTemp IS enfoque_pequeno) THEN (Tamano IS gran_calor) weight=1.0

IF
VelFluBenceno is: EnfTemp is: THEN
(0) ---- (0) ---- [0] ----
 not not not
 AND OR Weight: 1.0

Delete rule Add rule Change rule

Graphic window number 0

File Tools Edit ?

Graphic window number 0

Member functions for input number 1 named VelFluBenceno

mu(VelFluBenceno)

VelFluBenceno

flujo_grande

flujo_pequeno

Member functions for input number 2 named EnfTemp

mu(EnfTemp)

EnfTemp

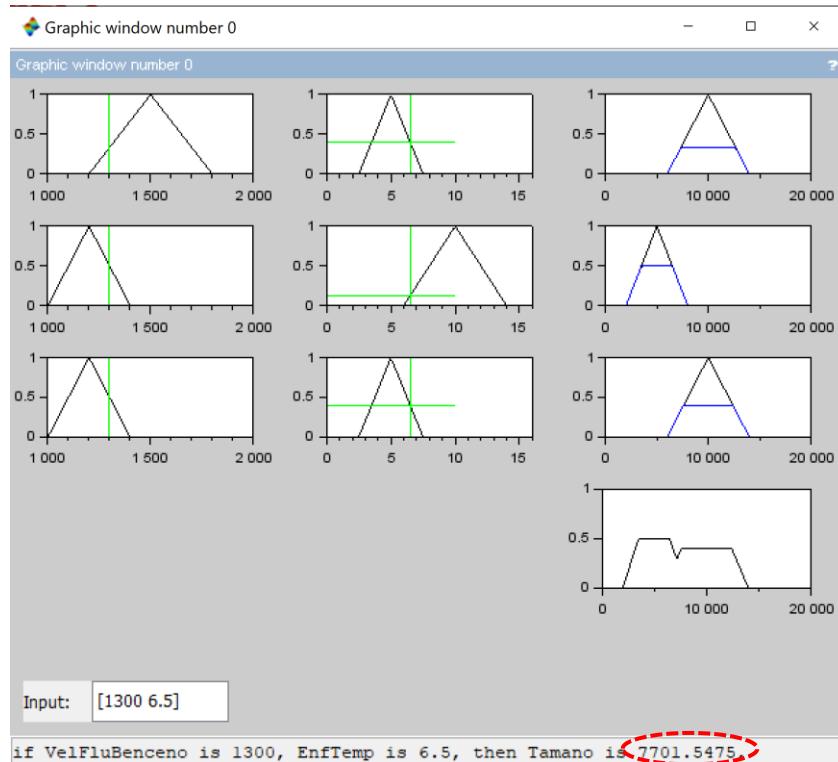
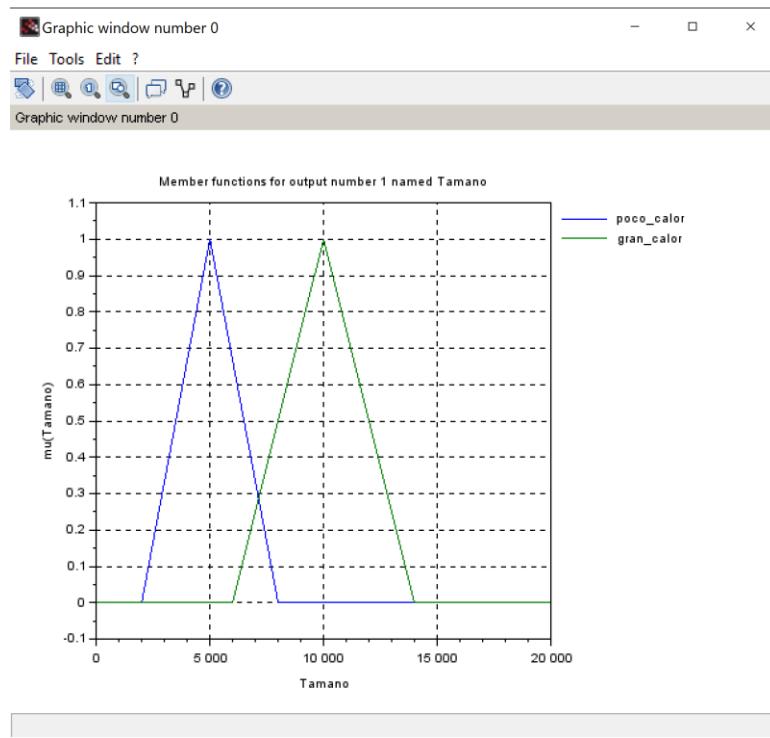
enfoque_pequeno

enfoque_grande

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



c) Resolviendo con Octave.○ **Código Octave:**

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_3_Mandani.m
## Última modificación: 11 mar 2022
#
# OJO: Cargar previamente "pkg load fuzzy-logic-toolkit";
clc; clear; close all;
fis=readfis ('Ejemplo_5_18_Octave');
## Grafica las funciones de membresía de entrada y salida.
plotmf (fis, 'input', 1)
plotmf (fis, 'input', 2);
plotmf (fis, 'output', 1);
grid('on')
resp='S';
while (resp=='S')
    clc
    ## Muestra las reglas en formato simbólico.
    puts ("\nReglas del Ejemplo 5.18:\n\n");
    showrule(fis); disp(' ');disp(' ');
    ## Hace los cálculos de Au
    puts ("\nIngresar los valores de Flu-Benceno y Aprox-Temp");
    puts ("\n para evaluar con el Sistema de Inferencia Difusa (SID)\n\n");
    FluBenceno = input('Flujo de Benceno(1,000 a 1,800)? ');
    AproxTemp = input('Aprox. Temperatura (2.5 - 14)? '); disp(' ');disp(' ');
    [output, rule_input, rule_output, fuzzy_output] = ...
    evalfis ([FluBenceno, AproxTemp], fis, 1001);
    ## Grafic la primera salida (AU)de las reglas difusas inbdividuales
    ## en un conjunto de ejes.
    x_axis = linspace (fis.output(1).range(1), ...
                      fis.output(1).range(2), 1001);
    colors = ['r' 'b' 'm' 'g'];
    figure ('NumberTitle', 'off', 'Name', ...
            'Output of Fuzzy Rules 1-4 for Input = (FluBenceno, AproxTemp)');
    for i = 1 : 3
        y_label = [colors(i) ";Regla " num2str(i) ";"];
        plot (x_axis, rule_output(:,i), y_label, 'LineWidth', 2);
        ylim([-0.1, 1.1]);
        grid;
        figure();
    endfor
    for i = 1 : 3
        y_label = [colors(i) ";Regla " num2str(i) ";"];
        plot (x_axis, rule_output(:,i), y_label, 'LineWidth', 2);
        grid; hold on;
    endfor
    ylim([-0.1, 1.1]);
    xlabel ('Tamaño', 'FontWeight', 'bold');
    ## Trazar la primera salida difusa agregada y la primera salida clásica
    ## (TAU) en un juego de ejes.
    figure('NumberTitle', 'off', 'Name', ...
            'Agregación y defuzzificación para la entrada = (FluBenceno, AproxTemp)');
    plot (x_axis, fuzzy_output(:, 1), "b;Salida difusa agregada;", ...
          'LineWidth', 2);
    hold on;
    crisp_output = evalmf(x_axis, output(1), 'constant');
    y_label = ["r;Salida clásica = " num2str(output(1)) ";"];
    plot (x_axis, crisp_output, y_label, 'LineWidth', 2);
    ylim([-0.1, 1.1]);
    xlabel ('Tamaño', 'FontWeight', 'bold');
    grid;
    hold;

```

```
% Evalúa el FIS;
sol=output(1)
disp(' ');
resp=input('Otro cálculo con el SID (S/N)? ','s');disp(' ')
resp=upper(resp);
close all;
endwhile
```

○ **Sistema de inferencia difusa:**

```
## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_3_Mandani.fis
## Última modificación: 11 mar 2022
#
[System]
Name='Ejemplo_5_18_Octave'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=3
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='FluBenceno'
Range=[1000 2000]
NumMFs=2
MF1='flujo-grande':'trimf',[1200 1500 1800]
MF2='flujo-pequeno':'trimf',[1000 1200 1400]

[Input2]
Name='AproxTemp'
Range=[0 15]
NumMFs=2
MF1='aprox-pequena':'trimf',[2.5 5.0 7.5]
MF2='aprox-grande':'trimf',[6.0 10.0 14.0]

[Output1]
Name='*AU'
Range=[0 20000]
NumMFs=2
MF1='poco-calor':'trimf',[2000 5000 8000]
MF2='gran-calor':'trimf',[6000 10000 14000]

[Rules]
1 1, 2 (1) : 1
2 2, 1 (1) : 2
2 1, 2 (1) : 1
```

○ **Resultados:**

Reglas del Ejemplo 5.18:

1. If (FluBenceno is flujo-grande) and (AproxTemp is aprox-pequena), then (*AU is gran-calor) (1)
2. If (FluBenceno is flujo-pequeno) or (AproxTemp is aprox-grande), then (*AU is poco-calor) (1)
3. If (FluBenceno is flujo-pequeno) and (AproxTemp is aprox-pequena), then (*AU is gran-calor) (1)

Ingresar los valores de Flu-Benceno y Aprox-Temp
para evaluar con el Sistema de Inferencia Difusa (SID)

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

Flujo de Benceno(1,000 a 1,800)? **1300**
Aprox. Temperatura (2.5 - 14)? **6.5**

sol = **7699.9**

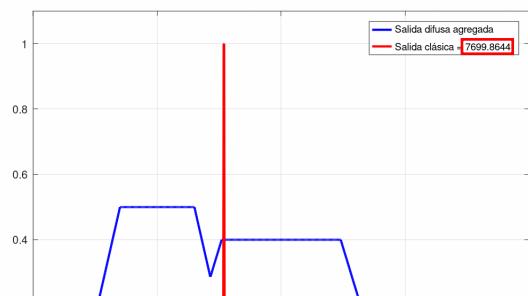
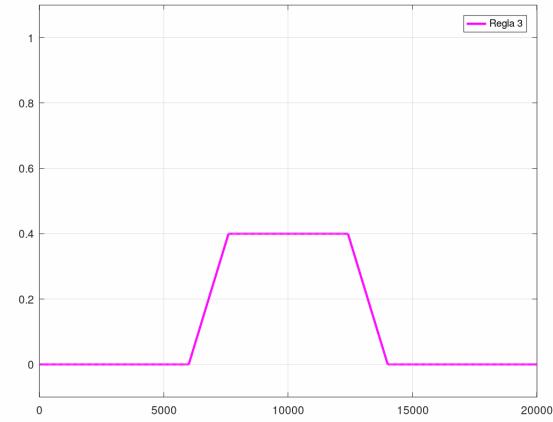
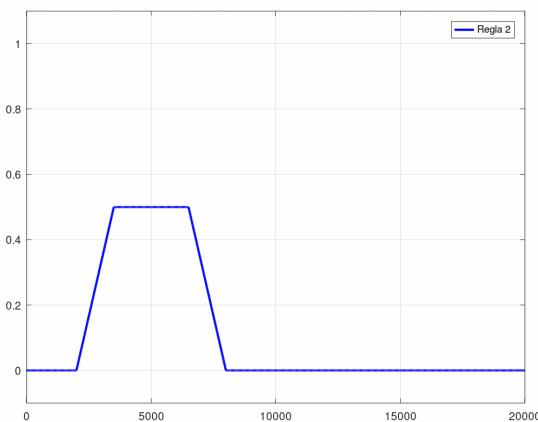
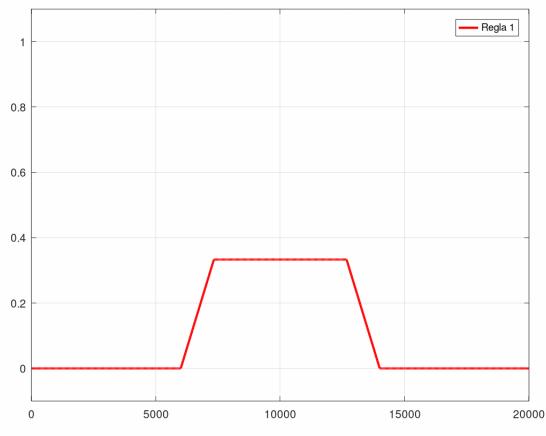
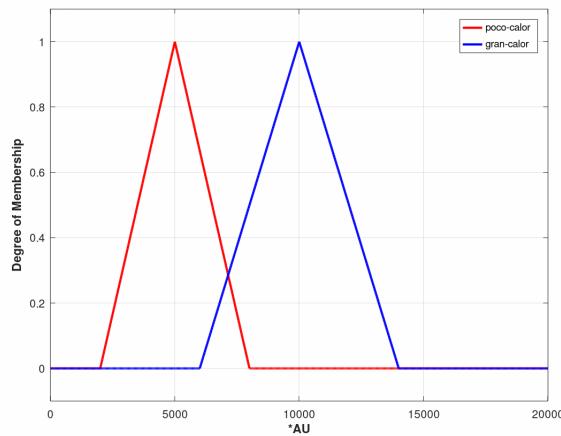
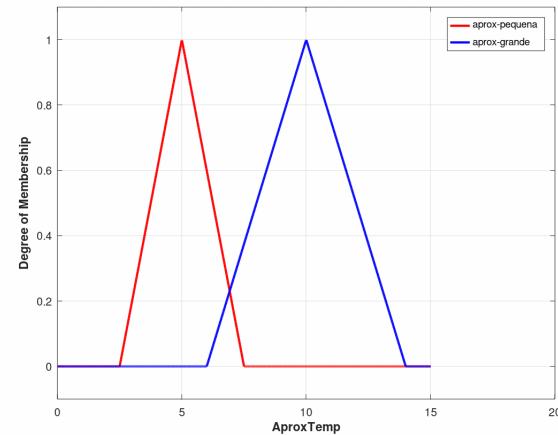
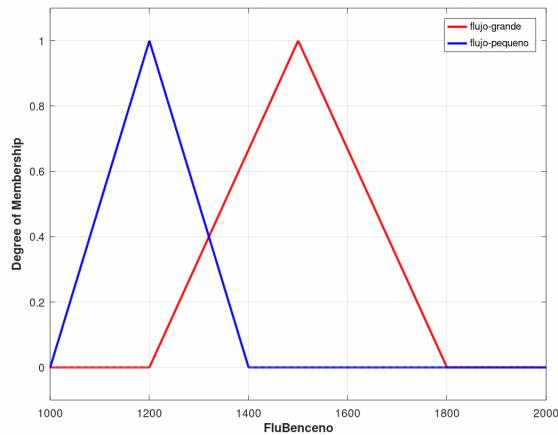
Otro cálculo con el SID (S/N)? n

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

>>



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

Sugeno

Para el método de inferencia difusa de Sugeno, se tiene experiencia en el diseño de intercambiadores de calor que da las siguientes expresiones en forma polinomial para los dos consecuentes (intercambiadores de calor pequeños y grandes):

$$AU_{pequeño} = 3.4765w - 210.5\Delta T_{app} + 2103$$

$$AU_{grande} = 4.6925w - 526.2\Delta T_{app} + 2631$$

Tomando el valor de membresía mínimo para la conjunción de entrada "y" de las Reglas 1 y 3, y el valor máximo para la disyunción de entrada "o" en la Regla 2, el valor de membresía de cada uno de los consecuentes será (figura 3.3, Mandani):

$$\text{Regla1: } \mu(AU)_1 = 0.33,$$

$$\text{Regla 2: } \mu(AU)_2 = 0.50,$$

$$\text{Regla 3: } \mu(AU)_3 = 0.40.$$

Entonces

$$AU_{pequeño} = 5,254 \text{ m}^2 \text{ kW/m}^2 \text{ }^\circ\text{K} \quad y \quad AU_{grande} = 5,311 \text{ m}^2 \text{ kW/m}^2 \text{ }^\circ\text{K}$$

Finalmente, el valor defuzzificado del tamaño del intercambio de calor es (usando el método de defuzificación promedio ponderado),

$$AU^* = \frac{\mu(AU)_1 \cdot AU_{pequeño} + \mu(AU)_2 \cdot AU_{grande} + \mu(AU)_3 \cdot AU_{pequeño}}{\mu(AU)_1 + \mu(AU)_2 + \mu(AU)_3}$$

$$AU^* = \frac{(0.33)(5311 \text{ m}^2 \text{ kW/m}^2 \text{ }^\circ\text{K}) + (0.50)(5254 \text{ m}^2 \text{ kW/m}^2 \text{ }^\circ\text{K}) + (0.40)(5311 \text{ m}^2 \text{ kW/m}^2 \text{ }^\circ\text{K})}{0.33 + 0.50 + 0.40}$$

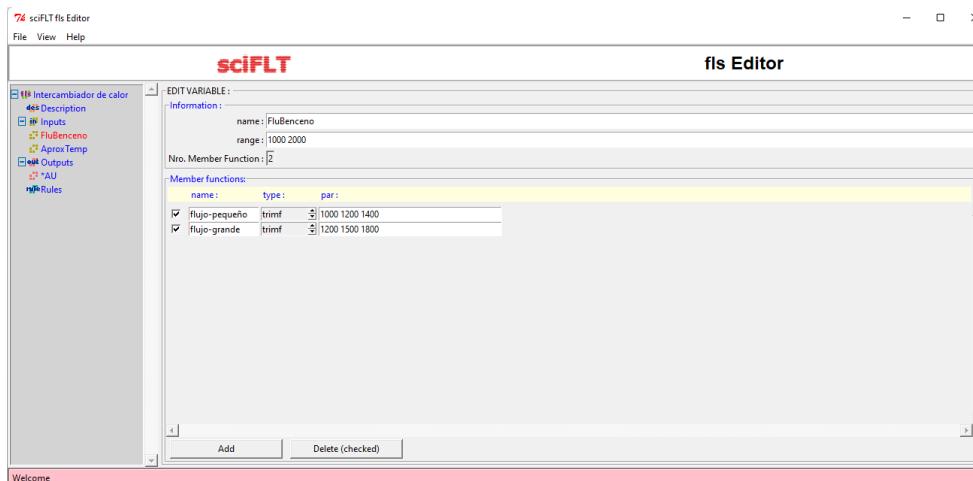
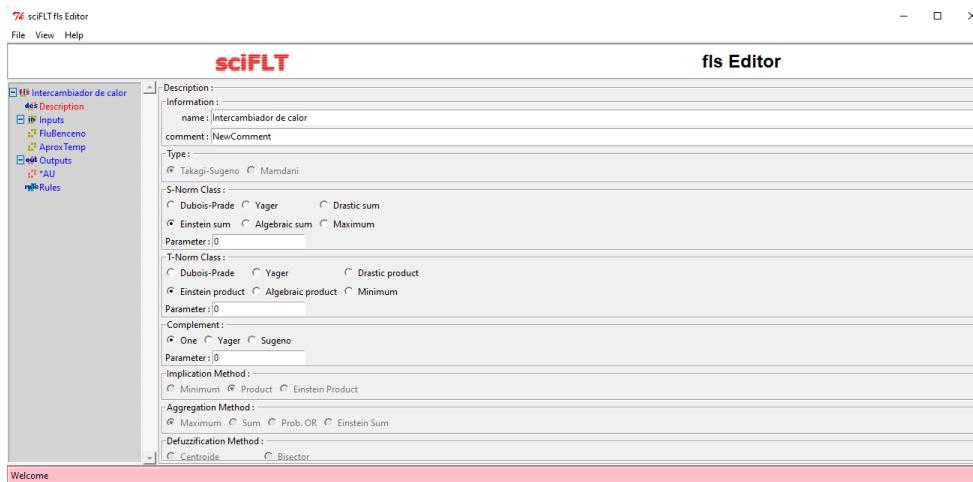
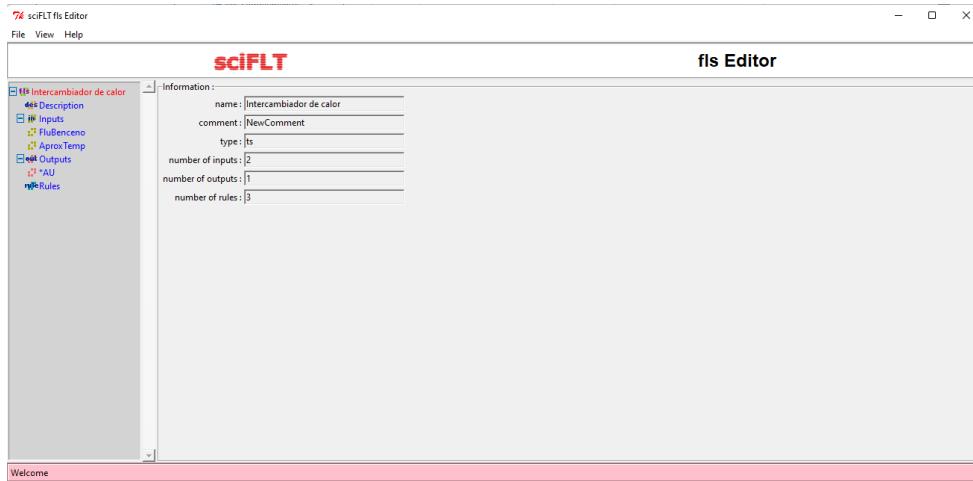
$$AU^* = \mathbf{5,287.8} \text{ m}^2 \text{ kW/m}^2 \text{ }^\circ\text{K}$$

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

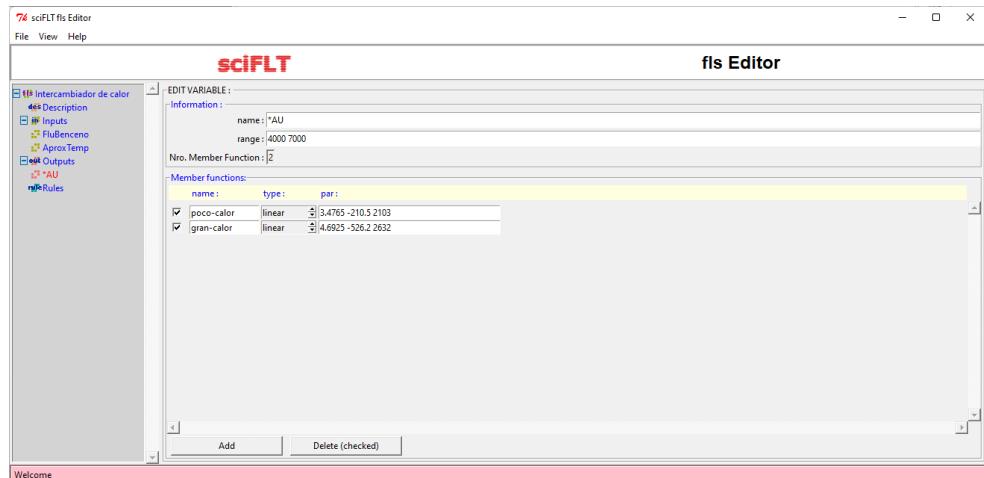
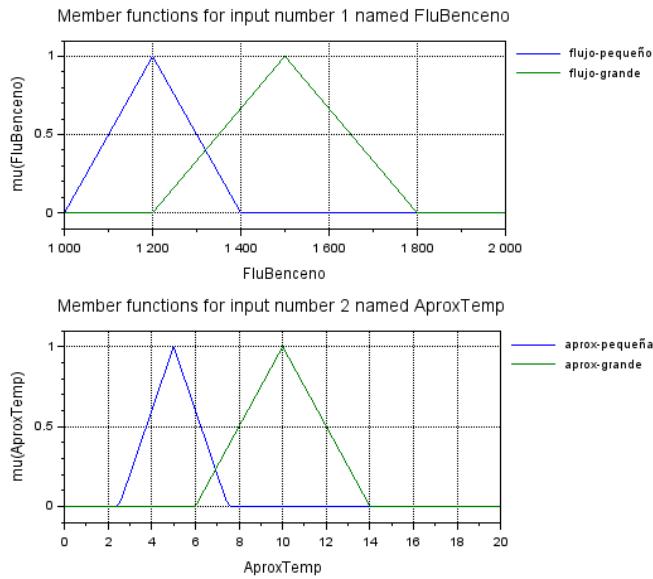
a) Usando Scilab:



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

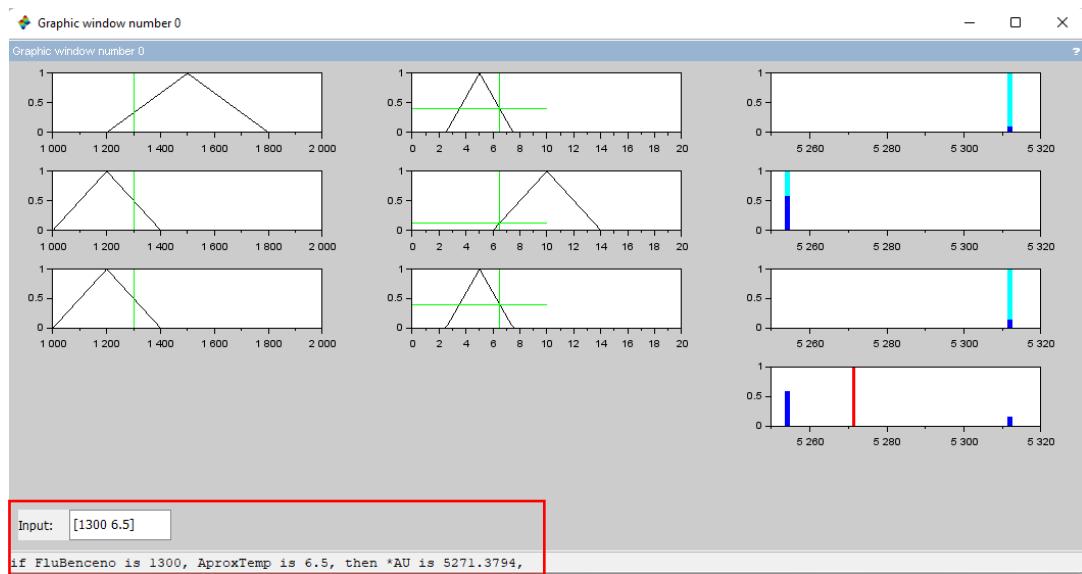
2022.



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



b) Usando Octave○ **Código Octave:**

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_3_Sugeno.m
## Última modificación: 11 mar 2022
#
# # OJO: Cargar previamente "pkg load fuzzy-logic-toolkit";
#
clc; close all;clear;
## Lee el archivo FIS.
fis = readfis ('Ejemplo_5_18_Sugeno');
## Grafica las funciones de membresía de entrada y salida.
plotmf (fis, 'input', 1);
plotmf (fis, 'input', 2);
resp='S'
while (resp=='S')
    clc
    ## Mostrar las reglas en formato simbólico.
    puts ("\nReglas del Ejemplo 5_18 Sugeno:\n\n");
    ## Muestra las reglas.
    showrule (fis);
    ## Estima *AU con datos de FlujoBenceno y AproxTemp
    puts ("\n\n\nIngresar los valores de FlujoBenceno y AproxTemp");
    puts ("\n para evaluar con el Sistema de Inferencia Difusa (SID)\n\n");
    FC = input(' Flujo de Benceno(1,000 a 1,800) ? ');
    AT = input(' Aprox. Temperatura (2.5 - 14) ? '); disp('');disp('');
    AU=evalfis([FC,AT], fis, 1001);
    fprintf('* AU    = %6.0f',AU);fprintf('\n');
    disp('');
    resp=input('Otro cálculo con el SID (S/N) ? ','s');disp('');
    resp=upper(resp);
    close all;
endwhile

```

○ **Código del sistema de inferencia difusa:**

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_3_Sugeno.fis
## Última modificación: 11 mar 2022
#
[System]
Name='Ejemplo_5_18_Sugeno'
Type='sugeno'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=3
AndMethod = 'einstein_product'
OrMethod = 'einstein_sum'
ImpMethod = 'prod'
AggMethod = 'sum'
DefuzzMethod = 'centroid'

[Input1]
Name='FluBenceno'
Range=[1000 2000]
NumMFs=2
MF1='flujo-grande':'trimf',[1200 1500 1800]
MF2='flujo-pequeño':'trimf',[1000 1200 1400]

[Input2]
Name='AproxTemp'

```

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

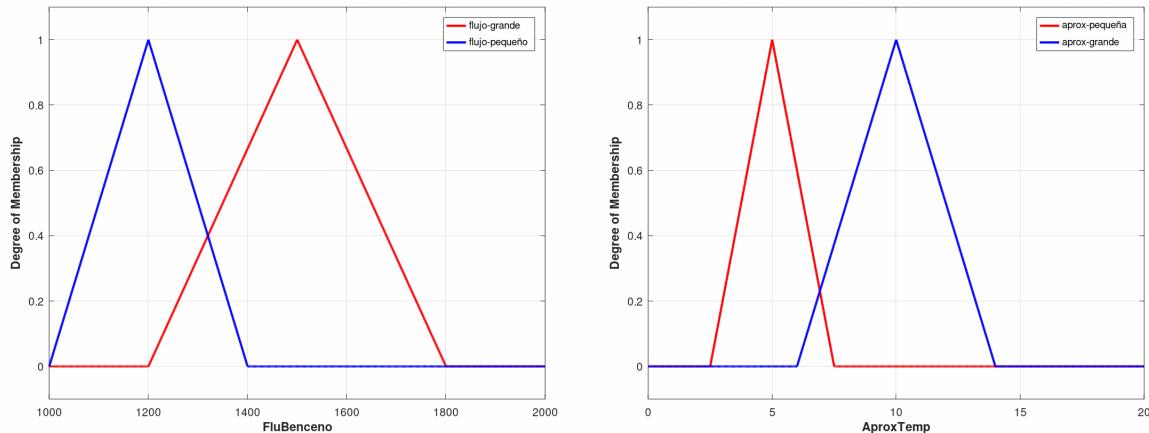
```

Range=[0 20]
NumMFs=2
MF1='aprox-pequeña':'trimf',[2.5 5 7.5]
MF2='aprox-grande':'trimf',[6 10 14]

[Output1]
Name='*AU'
Range=[4000 7000]
NumMFs=2
MF1='AUgrande':'linear',[4.6925 -526.2 2631]
MF2='AUpequeño':'linear',[3.4765 -210.5 2103]

[Rules]
1 1, 1 (1) : 1
2 2, 2 (1) : 2
2 1, 1 (1) : 1

```



Reglas del Ejemplo 5_18 Sugeno:

1. If (FluBenceno is flujo-grande) and (AproxTemp is approx-pequeña), then (*AU is AUgrande) (1)
2. If (FluBenceno is flujo-pequeño) or (AproxTemp is approx-grande), then (*AU is AUpequeño) (1)
3. If (FluBenceno is flujo-pequeño) and (AproxTemp is approx-pequeña), then (*AU is AUgrande) (1)

Ingresar los valores de FluBenceno y AproxTemp
para evaluar con el Sistema de Inferencia Difusa (SID)

Flujo de Benceno(1,000 a 1,800)? 1300
Aprox. Temperatura (2.5 - 14) ? 6.5

* AU = **5271**

Otro cálculo con el SID (S/N) ?

Ejemplo 4³. Problema sobre el clima.**Mandani**

En este ejercicio, se muestra el proceso de creación de un sistema difuso en R. Este sistema difuso usará el clima como ejemplo y su propósito es determinar qué tan "malo", "bueno" o "perfecto" es el clima, es decir, el sistema asignará una puntuación a cada una de estas categorías. Las reglas difusas no están basadas de ninguna manera en algo científico o serio, es solo una guía.

$$\text{clima} = f(\text{temperatura}, \text{humedad}, \text{precipitación})$$

• Variables de entrada:

- Temperatura ($^{\circ}\text{F}$) -> función de membresía Normal: fría(6,5), agradable(25,5) y caliente(35,5)
- Humedad (%) -> función de membresía Normal: seco(30,3), normal(60,3) y húmedo(80,3)
- Precipitación (mm) -> función de membresía Normal: sin lluvia(5,5), poca lluvia(25,5) y con lluvia(50,5)

• Variable de salida:

- clima -> función de membresía Triangular: malo(30,10), bueno(60,10) y perfecto(90,10)

• Reglas difusas:

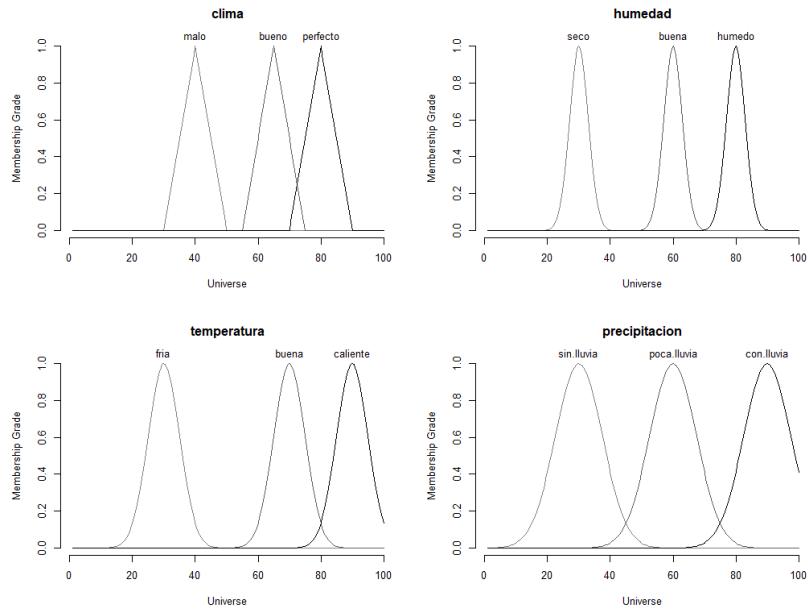
- Si la temperatura es agradable y la humedad es seca y la precipitación es sin lluvia ENTONCES el clima es perfecto.
- Si la temperatura es caliente y la humedad es húmedo y la precipitación es con lluvia ENTONCES el clima es malo.
- Si la temperatura es fría ENTONCES el clima es malo.
- Si la temperatura es agradable o la humedad es normal o la precipitación es poca lluvia ENTONCES el clima es bueno.
- Si la temperatura es caliente y la precipitación es poca lluvia ENTONCES el clima es bueno.
- Si la temperatura es caliente y la humedad es seco y la precipitación es poca lluvia ENTONCES el clima es bueno.

³ http://juandes.github.io/FuzzyLogic-R/docs/fuzzy_tutorial

a) Resolviendo con R.○ **Código.**

```
##### http://juandes.github.io/FuzzyLogic-R/docs/fuzzy_tutorial
#
graphics.off()
rm(list = ls())
cat("\014")
#
### Configurar el sistema difuso
library(sets)
sets_options("universe", seq(1, 100, 0.5))
#
# variables de entrada: temperatura, humedad y precipitación
# variable de salida: clima
variables <- set(
  # función de membresía Normal (desv. estandar=5.0)
  temperatura = fuzzy_partition(varnames = c(fria = 30, agradable = 70, caliente = 90),
                                 sd = 5.0),
  humedad = fuzzy_partition(varnames = c(seco = 30, normal = 60, humedo = 80),
                            sd = 3.0),
  precipitacion = fuzzy_partition(varnames = c(sin.lluvia = 30, poca.lluvia = 60,
                                                con.lluvia = 90), sd = 7.5),
  clima = fuzzy_partition(varnames = c(malo = 40, bueno = 65, perfecto = 80),
                           FUN = fuzzy_cone, radius = 10)
)
#
# las reglas difusas
rules <- set(
  fuzzy_rule(temperatura %is% agradable && humedad %is% seca &&
             precipitacion %is% sin.lluvia, clima %is% perfecto),
  fuzzy_rule(temperatura %is% caliente && humedad %is% humedo &&
             precipitacion %is% con.lluvia, clima %is% malo),
  fuzzy_rule(temperatura %is% fria, clima %is% malo),
  fuzzy_rule(temperatura %is% agradable || humedad %is% normal ||
             precipitacion %is% poca.lluvia, clima %is% bueno),
  fuzzy_rule(temperatura %is% caliente && precipitacion %is% poca.lluvia,
             clima %is% bueno),
  fuzzy_rule(temperatura %is% caliente && humedad %is% seco &&
             precipitacion %is% poca.lluvia, clima %is% bueno)
)
#
# el sistema de inferencia difusa
model <- fuzzy_system(variables, rules)
print(model)
plot(model)
#
# Ejemplos
# Ejemplo 1: Temperatura = 75, humedad = 0 y precipitación = 70
example.1 <- fuzzy_inference(model, list(temperatura = 75, humedad = 0,
                                         precipitacion = 70))
gset_defuzzify(example.1, "centroid")
plot(example.1)
# Ejemplo 2: Temperatura = 30, humedad = 0 y precipitación = 70
example.2 <- fuzzy_inference(model, list(temperatura = 30, humedad = 0,
                                         precipitacion = 70))
gset_defuzzify(example.2, "largestofmax") # No se puede con el centroide
plot(example.2) # porque la salida son dos figuras
sets_options("universe", NULL) # Reset the universe
```

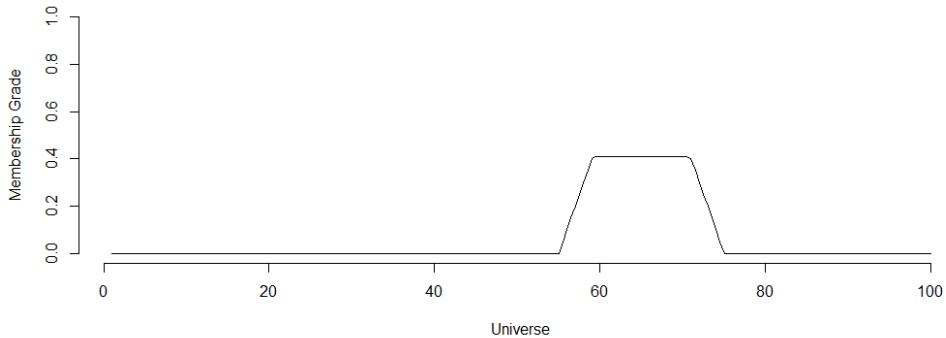
- **Resultados:**



- ❖ Primer escenario: temperatura = 75°F, humedad = 0%, precipitación = 70 mm

```
> example.1 <- fuzzy_inference(model, list(temperatura = 75, humedad = 0,
+                                         precipitacion = 70))
> gset_defuzzify(example.1, "centroid")
[1] 65
```

Entonces, **clima = 65 (bueno)**.

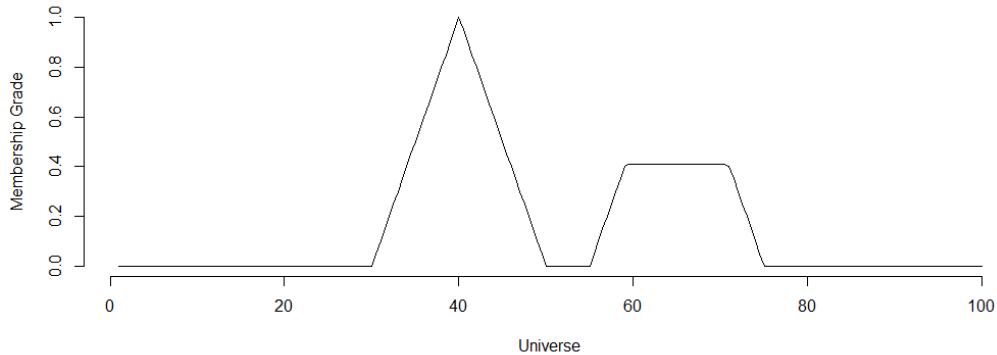


- ❖ Segundo escenario: temperatura = 30°F, humedad = 0%, precipitación = 70 mm

```
> example.2 <- fuzzy_inference(model, list(temperatura = 30, humedad = 0,
```

```
+ precipitation = 70))  
> gset_defuzzify(example.2, "largestofmax")  
[1] 40
```

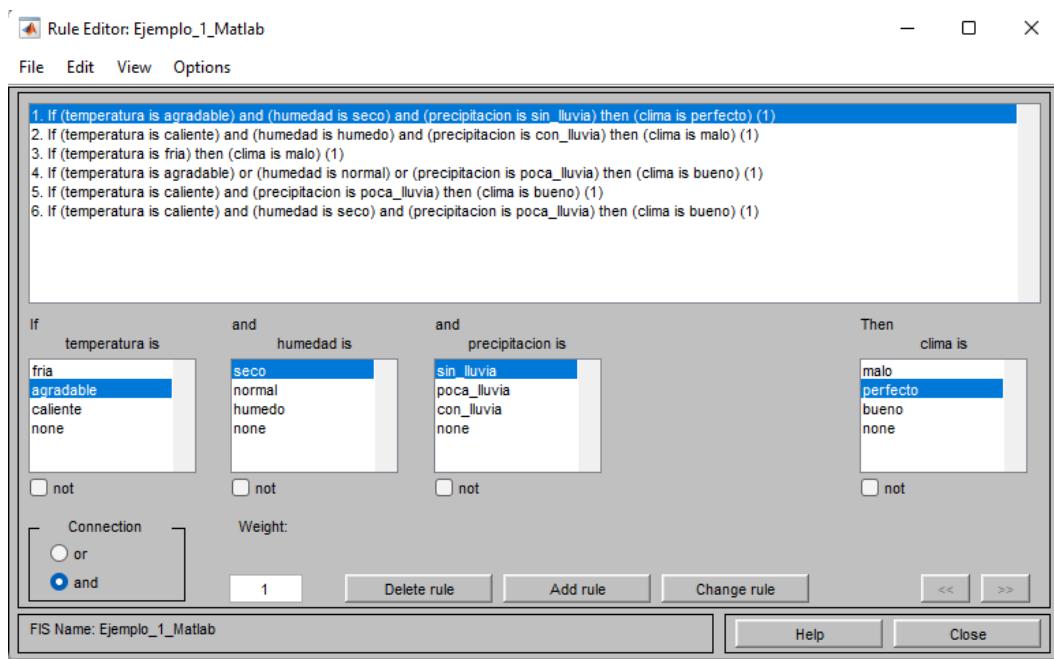
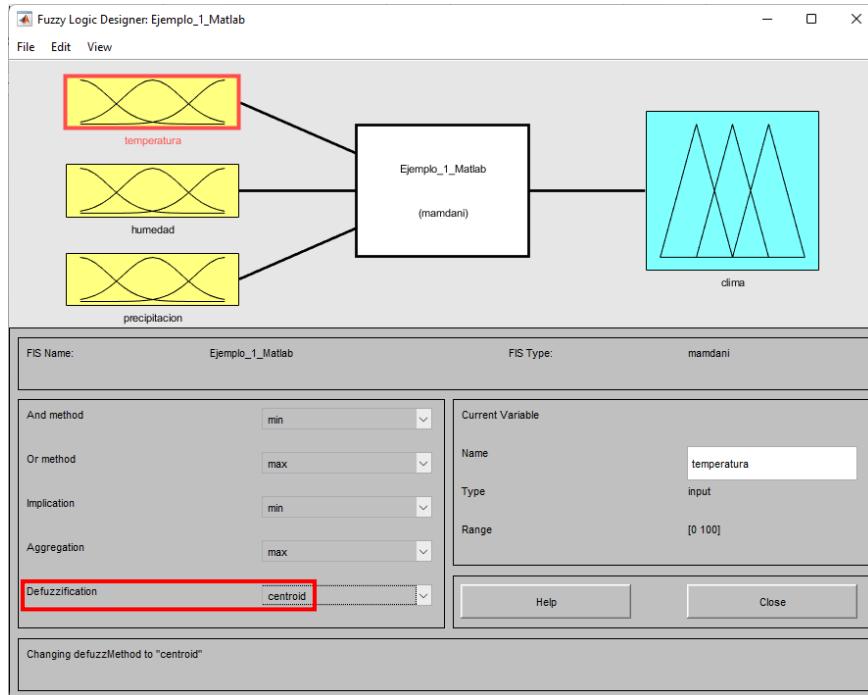
Clima = 40 (malo).



- NOTA: Aquí no se puede defuzificar con el centroide porque son dos figuras separadas en la salida, por lo que sustituye por el método de *mayor del máximo* (*lom*: "largest of maximum")

Se observa que al bajar la temperatura, el modelo reduce la cantidad de clima "bueno" a alrededor de 0.4 y crea un nuevo "pico suave" con un máximo global de 40°F, lo que significa que el clima es "malo".

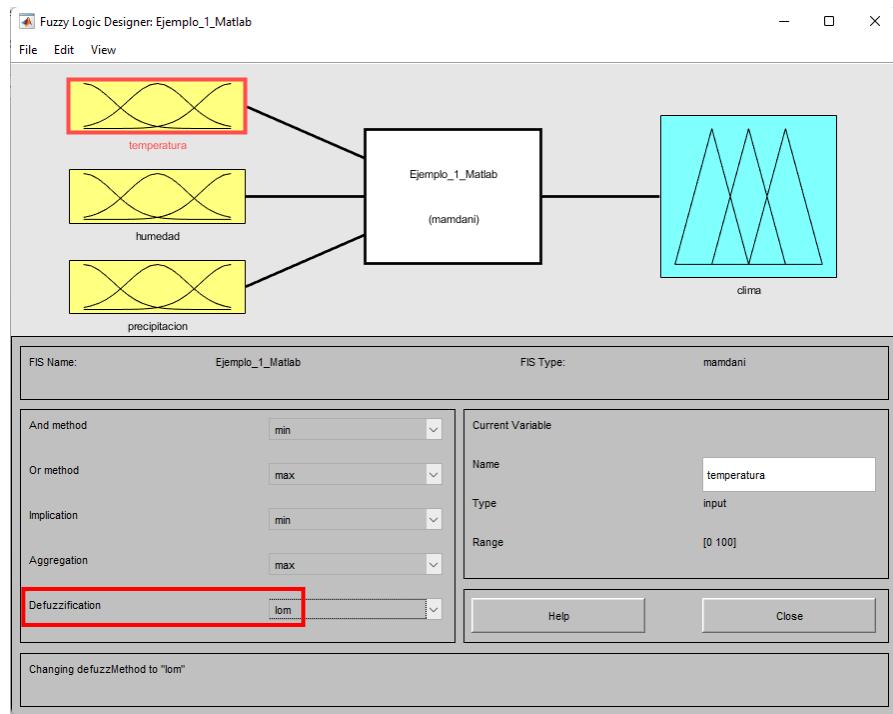
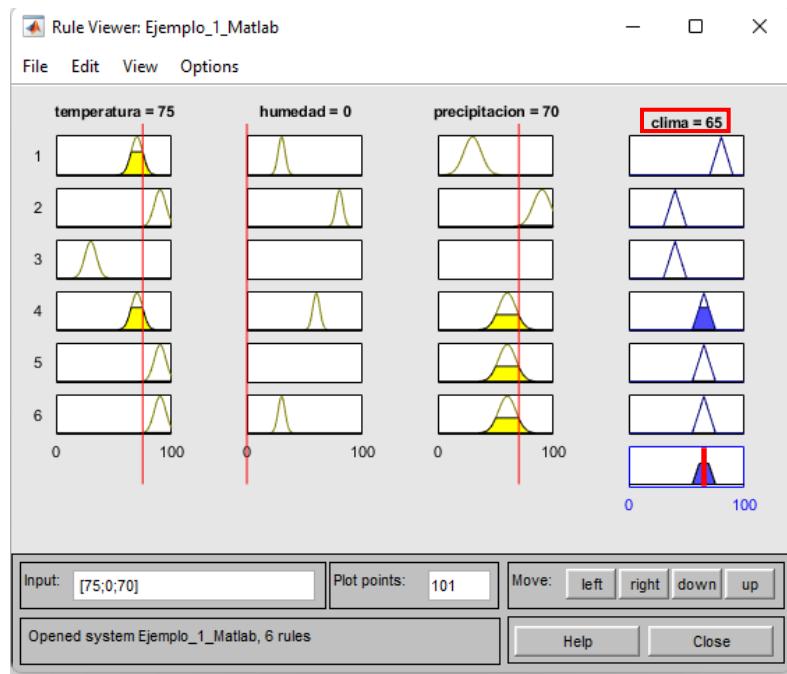
b) Resolviendo con Matlab.



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

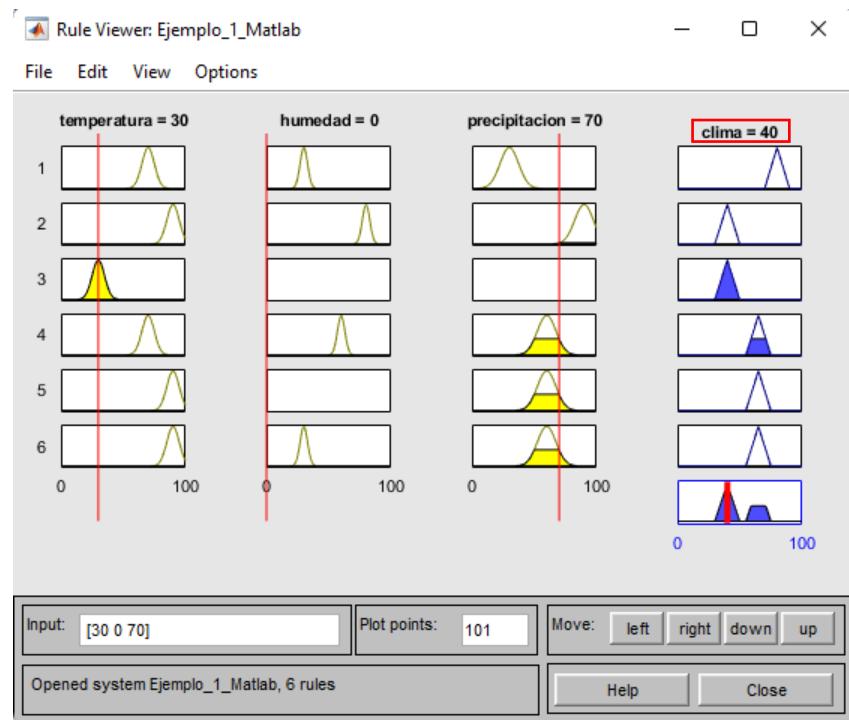
2022.



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.



c) Resolviendo con Octave.

○ *Código en Octave:*

```
% Evalúa el FIS;
fprintf('Clima = %6.0f',output(1));
disp(' '); disp(' ')
resp=input('Otro cálculo con el SID (S/N)? ','s'); disp(' ')
resp=upper(resp);
close all;
endwhile
```

- **Código del sistema de inferencia difusa:**

```
## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_4_Mandani.fis
## Última modificación: 11 mar 2022
#
[System]
Name='Ejemplo_4_Mandani'
Type='mamdani'
Version=2.0
NumInputs=3
NumOutputs=1
NumRules=6
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='temperatura'
Range=[0 100]
NumMFs=3
MF1='fria':'gaussmf',[5 30]
MF2='agradable':'gaussmf',[5 70]
MF3='caliente':'gaussmf',[5 90]

[Input2]
Name='humedad'
Range=[0 100]
NumMFs=3
MF1='seco':'gaussmf',[3 30]
MF2='normal':'gaussmf',[3 60]
MF3='humedo':'gaussmf',[3 80]

[Input3]
Name='precipitacion'
Range=[0 100]
NumMFs=3
MF1='sin_lluvia':'gaussmf',[7.5 30]
MF2='poca_lluvia':'gaussmf',[7.5 60]
MF3='con_lluvia':'gaussmf',[7.5 90]

[Output1]
Name='clima'
Range=[0 100]
NumMFs=3
MF1='malo':'trimf',[30 40 50]
MF2='bueno':'trimf',[55 65 75]
MF3='perfecto':'trimf',[70 80 90]

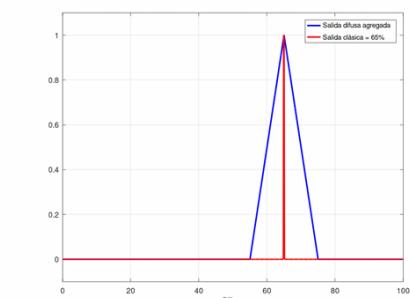
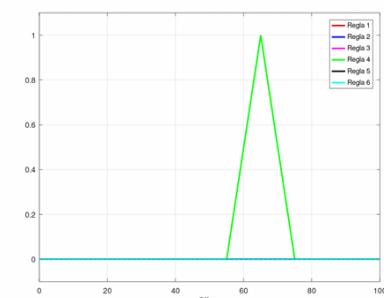
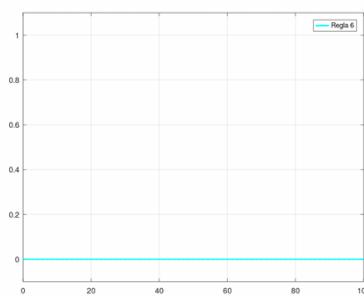
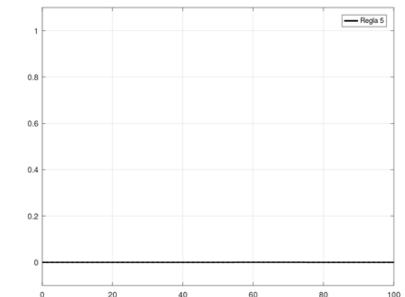
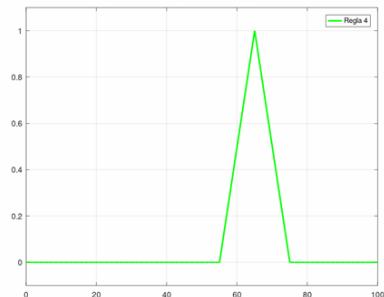
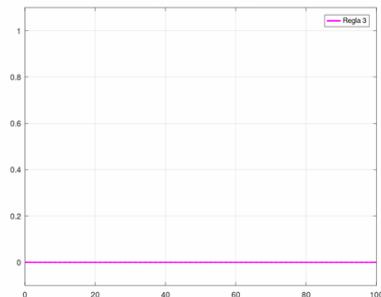
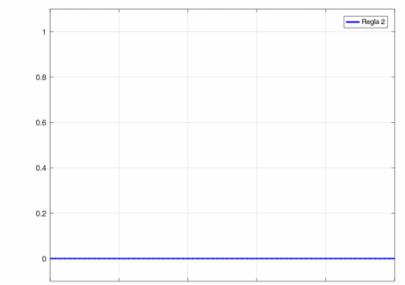
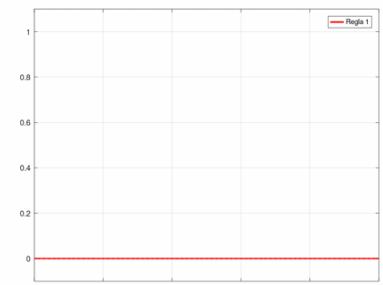
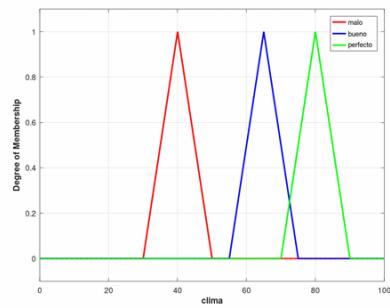
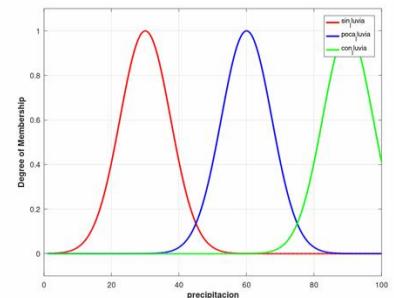
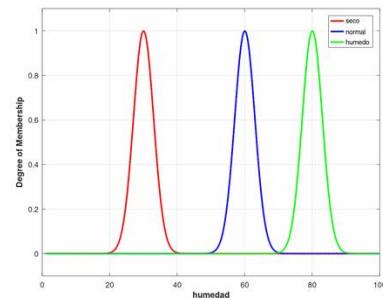
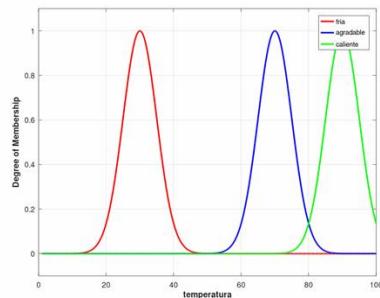
[Rules]
2 1 1, 3 (1) : 1
3 3 3, 1 (1) : 1
1 0 0, 1 (1) : 1
2 2 2, 2 (1) : 2
3 0 2, 2 (1) : 1
3 1 2, 2 (1) : 1
```

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

- **Resultados:**



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

Reglas de la calculadora de Clima Mamdani:

1. If (temperatura is agradable) and (humedad is seco) and (precipitacion is sin_lluvia), then (clima is perfecto) (1)
2. If (temperatura is caliente) and (humedad is humedo) and (precipitacion is con_lluvia), then (clima is malo) (1)
3. If (temperatura is fria), then (clima is malo) (1)
4. If (temperatura is agradable) or (humedad is normal) or (precipitacion is poca_lluvia), then (clima is bueno) (1)
5. If (temperatura is caliente) and (precipitacion is poca_lluvia), then (clima is bueno) (1)
6. If (temperatura is caliente) and (humedad is seco) and (precipitacion is poca_lluvia), then (clima is bueno) (1)

Ingresar los valores de Temperatura, Humedad y Precipitación para evaluar con el Sistema de Inferencia Difusa (SID)

```
T, temperatura (1-100) ? 75
H, humedad (%) (1-100) ? 0
P, precipitación (mm) (1-100)? 70
```

Clima = 65

Otro cálculo con el SID (S/N)? s

Siguiente escenario ($T = 30^{\circ}\text{F}$, $H = 0\%$, $P = 70 \text{ mm}$). Se cambia el método de defuzzificación en el código del sistema de inferencia difuso: (`DefuzzMethod='lom'`)

Reglas de la calculadora de propinas Mamdani:

1. If (temperatura is agradable) and (humedad is seco) and (precipitacion is sin_lluvia), then (clima is perfecto) (1)
2. If (temperatura is caliente) and (humedad is humedo) and (precipitacion is con_lluvia), then (clima is malo) (1)
3. If (temperatura is fria), then (clima is malo) (1)
4. If (temperatura is agradable) or (humedad is normal) or (precipitacion is poca_lluvia), then (clima is bueno) (1)
5. If (temperatura is caliente) and (precipitacion is poca_lluvia), then (clima is bueno) (1)
6. If (temperatura is caliente) and (humedad is seco) and (precipitacion is poca_lluvia), then (clima is bueno) (1)

Ingresar los valores de Calidad de comida y del Servicio para evaluar con el Sistema de Inferencia Difusa (SID)

```
T, temperatura (°F) 0-100) ? 30
H, humedad (%) (0-100) ? 0
P, precipitación (mm) (0-100)? 70
```

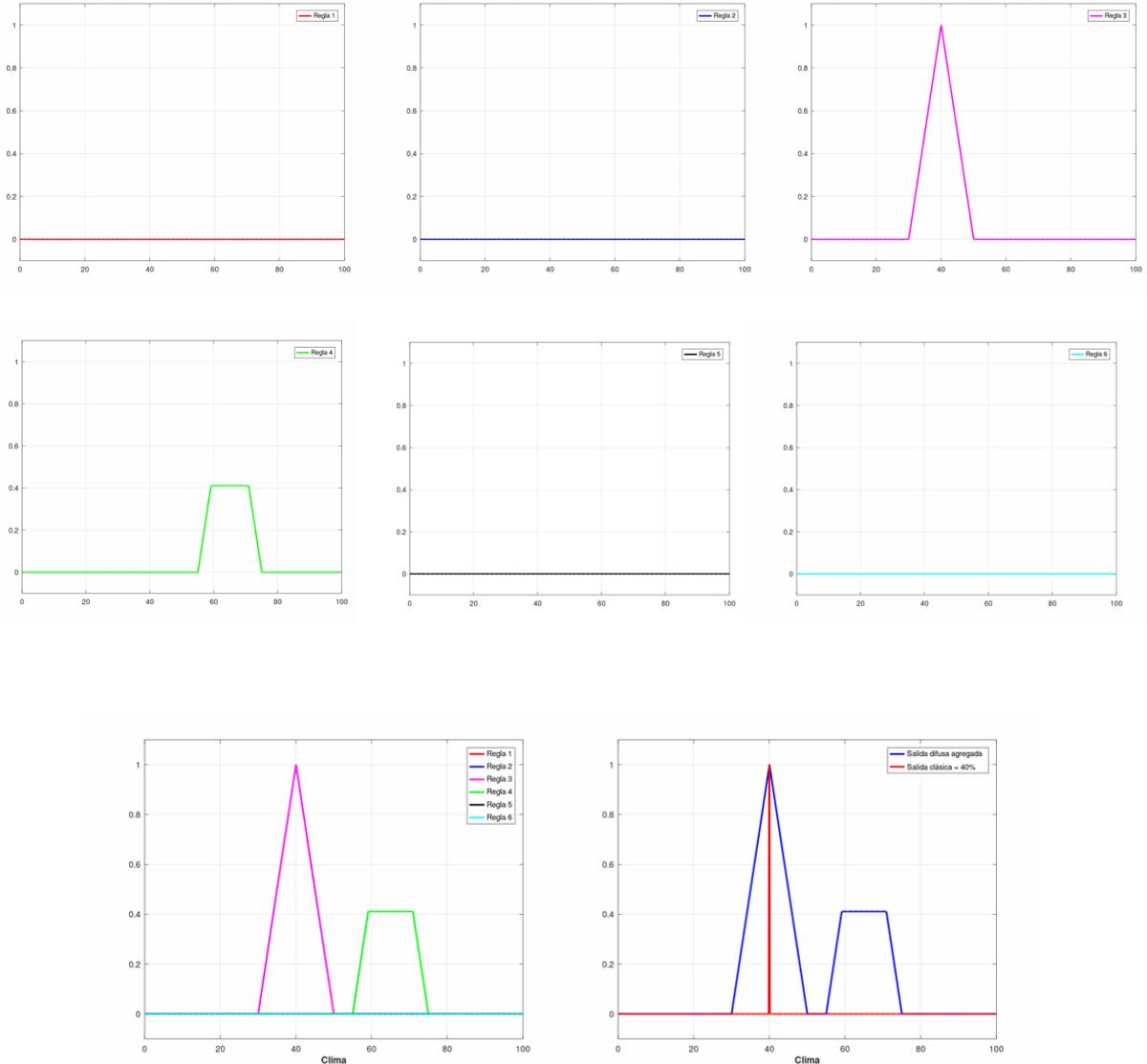
Clima = 40

Otro cálculo con el SID (S/N)? n

>>

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.



Sugeno

a) Resolviendo con Octave.

○ Código en Octave:

```
## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_4_Sugeno.m
## Última modificación: 11 mar 2022
#
# OJO: Cargar previamente "pkg load fuzzy-logic-toolkit";
clc; close all;clear;
fis = readfis ('Ejemplo_4_Sugeno');
## Grafica las funciones de membresía de entrada y salida.
plotmf (fis, 'input', 1)
plotmf (fis, 'input', 2);
plotmf (fis, 'input', 3);
plotmf (fis, 'output', 1);
resp='S'
```

```

while (resp=='S')
clc
## Mostrar las reglas en formato simbólico.
puts ("\\nReglas del Ejemplo 5_18 Sugeno:\\n\\n");
## Muestra las reglas.
showrule (fis);
## Calcula el clima con temperatura, humedad t precipitación
puts ("\nIngresar los valores de Temperatura, Humedad y Precipitación\n");
puts ("\n para evaluar con el Sistema de Inferencia Difusa (SID)\\n\\n");
T = input(' T, temperatura (1-100) ? ');
H = input(' H, humedad (%) (1-100) ? ');
P = input(' P, precipitación (mm) (1-100)? '); disp(' ');disp(' ');
[output, rule_input, rule_output, fuzzy_output] = ...
evalfis ([T H P], fis, 1001);
fprintf('* Clima = %6.0f',output(1));fprintf('\\n');
disp(' ');
resp=input('Otro cálculo con el SID (S/N)? ','s');disp(' ')
resp=upper(resp);
close all;
endwhile

```

- Código del sistema de inferencia difusa.

```

## Directorio:      c:/Fuzzy Sets/2022/GitHub
## Archivo:        Ejemplo_4_Sugeno.fis
## Última modificación: 11 mar 2022
#
[System]
Name='Ejemplo_4_Sugeno'
Type='mamdani'
Version=2.0
NumInputs=3
NumOutputs=1
NumRules=6
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='temperatura'
Range=[0 100]
NumMFs=3
MF1='fria':'gaussmf',[5 30]
MF2='agradable':'gaussmf',[5 70]
MF3='caliente':'gaussmf',[5 90]

[Input2]
Name='humedad'
Range=[0 100]
NumMFs=3
MF1='seco':'gaussmf',[3 30]
MF2='normal':'gaussmf',[3 60]
MF3='humedo':'gaussmf',[3 80]

[Input3]
Name='precipitacion'
Range=[0 100]
NumMFs=3
MF1='sin_lluvia':'gaussmf',[7.5 30]
MF2='poca_lluvia':'gaussmf',[7.5 60]
MF3='con_lluvia':'gaussmf',[7.5 90]

[Output1]
Name='clima'
Range=[0 100]
NumMFs=3
MF1 = 'malo' : 'constant', [40]
MF2 = 'bueno': 'constant', [65]
MF3 = 'perfecto': 'constant', [80]

[Rules]

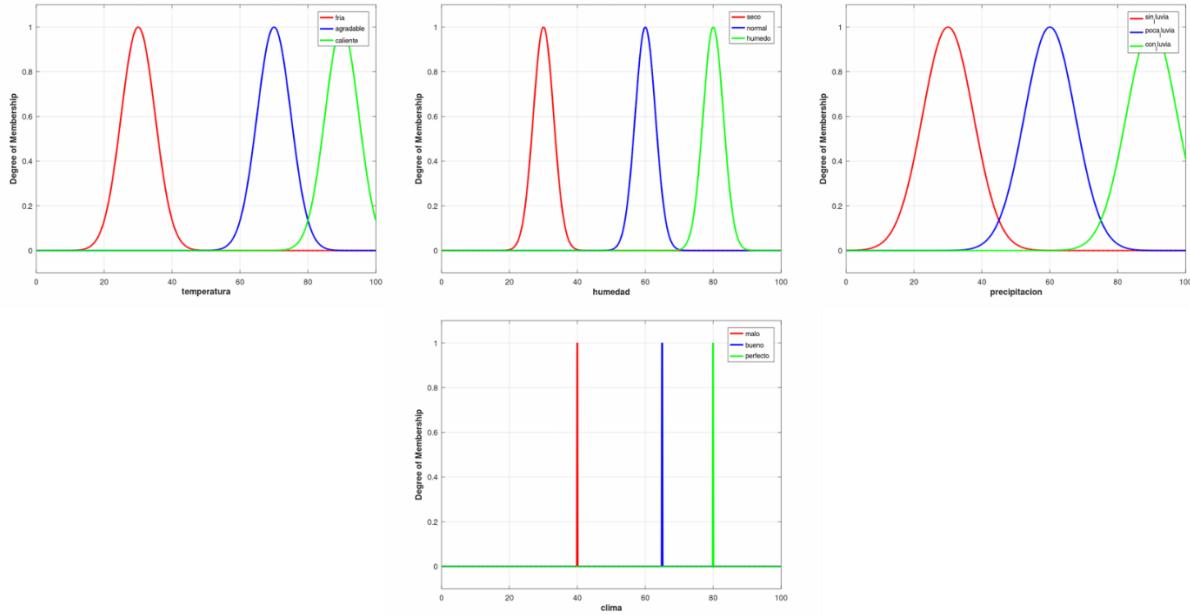
```

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

```
2 1 1, 3 (1) : 1
3 3 3, 1 (1) : 1
1 0 0, 1 (1) : 1
2 2 2, 2 (1) : 2
3 0 2, 2 (1) : 1
3 1 2, 2 (1) : 1
```

○ Resultados.



➤ Escenario: T = 75°F, H = 0 %, P = 70 mm

Reglas del Ejemplo 5_18 Sugeno:

1. If (temperatura is agradable) and (humedad is seco) and (precipitacion is sin_lluvia), then (clima is perfecto) (1)
2. If (temperatura is caliente) and (humedad is humedo) and (precipitacion is con_lluvia), then (clima is malo) (1)
3. If (temperatura is fria), then (clima is malo) (1)
4. If (temperatura is agradable) or (humedad is normal) or (precipitacion is poca_lluvia), then (clima is bueno) (1)
5. If (temperatura is caliente) and (precipitacion is poca_lluvia), then (clima is bueno) (1)
6. If (temperatura is caliente) and (humedad is seco) and (precipitacion is poca_lluvia), then (clima is bueno) (1)

Ingresar los valores de Temperatura, Humedad y Precipitación

para evaluar con el Sistema de Inferencia Difusa (SID)

```
T, temperatura (1-100)      ? 75
H, humedad (%) (1-100)       ? 0
P, precipitación (mm) (1-100) ? 70
```

* Clima = 65

Otro cálculo con el SID (S/N)? s

➤ Escenario: T = 30°F, H = 0 %, P = 70 mm

Reglas del Ejemplo 5_18 Sugeno:

1. If (temperatura is agradable) and (humedad is seco) and (precipitacion is sin_lluvia), then (clima is perfecto) (1)
2. If (temperatura is caliente) and (humedad is humedo) and (precipitacion is con_lluvia), then (clima is malo) (1)
3. If (temperatura is fria), then (clima is malo) (1)

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

4. If (temperatura is agradable) or (humedad is normal) or (precipitacion is poca_lluvia), then (clima is bueno) (1)
5. If (temperatura is caliente) and (precipitacion is poca_lluvia), then (clima is bueno) (1)
6. If (temperatura is caliente) and (humedad is seco) and (precipitacion is poca_lluvia), then (clima is bueno) (1)

Ingresar los valores de Temperatura, Humedad y Precipitación

para evaluar con el Sistema de Inferencia Difusa (SID)

```
T, temperatura (1-100)      ? 30  
H, humedad (%) (1-100)      ? 0  
P, precipitación (mm) (1-100)? 70
```

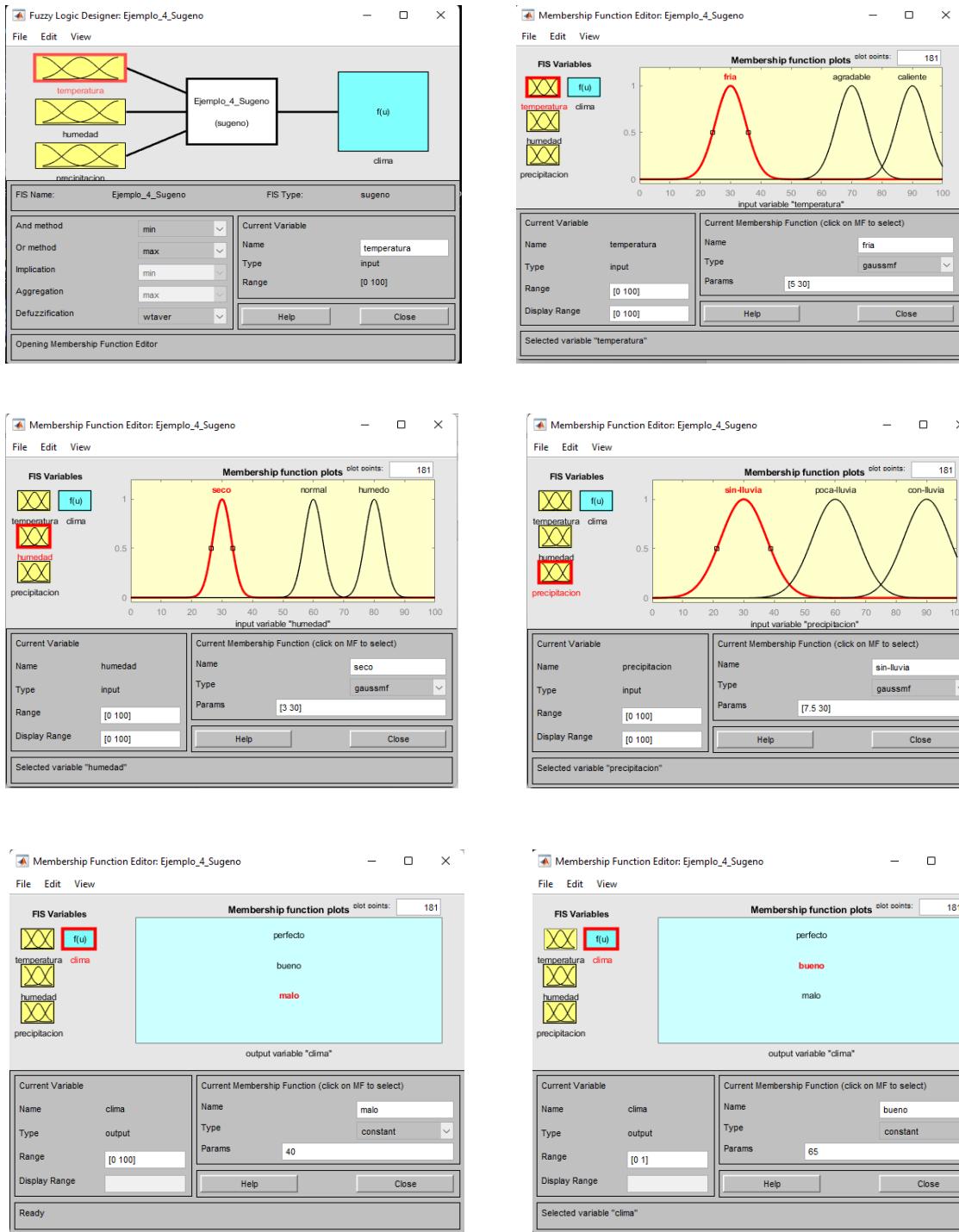
* Clima = 47

Otro cálculo con el SID (S/N)? n

Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado
2022.

b) Usando Matlab



Introducción al Sistema de Inferencia Difusa usando Octave

Salvador Díaz Maldonado

2022.

