

UNIVERSIDAD DE SANTIAGO DE CHILE  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA



## Informe taller n°2: "lógica difusa"

Integrantes: Juan Fernández Muñoz

Catalina Morales Rojas

Curso: Lógica y teoría de la computación

Profesor(a): Daniel Vega

10 de Julio de 2019

# Tabla de contenidos

<b>1. Introducción</b>	<b>1</b>
<b>2. Resumen del trabajo</b>	<b>2</b>
<b>3. Desarrollo</b>	<b>4</b>
<b>4. Conclusiones</b>	<b>9</b>
4.1. Lógica difusa v/s lógica clásica . . . . .	9
<b>Bibliografía</b>	<b>10</b>

# 1. Introducción

En el taller anterior fue implementada la lógica de primer orden, la cual tenía un poder expresivo mayor a la lógica proposicional, pero que aún así su poder expresivo era limitado. Es por esto que nace la lógica difusa, la cual permite representar de mejor forma la realidad, asignado un grado de pertenencia en el intervalo  $[0,1]$  que es determinada por su respectiva función de pertenencia.

Para lograr esto la lógica difusa cuenta con agentes inteligentes, los que tienen la característica de percibir su entorno y en base a eso realizar acciones.

A continuación se presenta el desarrollo y resultados obtenidos al representar una cafetera mediante lógica difusa y la inferencia de Mamdani, utilizando el lenguaje de programación Python con la librería fuzzy logic.

## 2. Resumen del trabajo

Los pasos a seguir en el trabajo son:

1. Se ingresan los parámetros por consola de la siguiente forma:

**python taller2.py Tipo Tamaño\_taza Intensidad**

La descripción de cada uno es:

- a) Tipo:** Corresponde al tipo de preparación a realizar, las cuales pueden ser espresso, latte, capuccino o mokaccino (ingresado en minúscula).
- b) Tamaño de la taza:** Es la cantidad de mL que puede contener la taza del café que se desea preparar.
- c) Temperatura:** grados °C de la temperatura ambiental.
- d) Intensidad:** grados de intensidad de la preparación.

La temperatura ambiental es seleccionada por el programa mediante la función random.

2. Fusificado: Se reciben las entradas externas (cantidad, tipo e intensidad), las cuales corresponden a valores nítidos y se obtienen los datos difusos con el valor de la función de pertenencia que corresponda junto con el valor lingüístico. En este punto es realizada una interpolación, ya que de esta manera es posible obtener el valor de pertenencia para el dato ingresado por el usuario en caso de que este no se encuentre en el arreglo de la función de pertenencia. Esta interpolación la realizan la librería scikit-fuzzy con sus funciones de antecedentes, consecuencia y control.
3. Reglas: En base a los antecedentes entregados en el enunciado se crean las reglas de inferencia, las cuales indican el comportamiento que tendrá el sistema. Para esto se utiliza la tabla de antecedentes y consecuentes para cada tipo de café. Por ejemplo una regla es:

Si las entradas son un tamaño de taza pequeño, una temperatura ambiental fría, una intensidad de café suave y el café de tipo espresso (antecedentes), entonces se obtiene un poco nivel de agua, cantidad de café y un tiempo de preparación medio (consecuentes). En el código se representa de la siguiente manera:

```
rule1 = ctrl.Rule(tamano_taza['pequeno'] & temperatura_ambiental['frio']
& intensidad_cafe['fuerte'], ( nivel_agua['poca'], cantidad_cafe['poca'], tiem-
po_preparacion['media'])))
```

La función rule lo que realiza internamente es aplicar la inferencia de mamdani, por lo que calcula el mínimo entre los antecedentes y los relaciona con los consecuentes. Fue necesario agregar más reglas debido a que al ingresar algunos valores estos no tenían pertenencia con ninguno de los antecedentes, por lo que al defusificar el área se volvía cero y se producía un error de cálculo.

Espresso					
Antecedentes			Consecuentes		
Tamaño de taza	Temperatura Ambiental	Intensidad del Café	Nivel de Agua	Cantidad de Café	Tiempo de Preparación
Pequeño	Frío	Suave	Poca	Poca	Media
Pequeño	Cálido	Fuerte	Poca	Media	Poca
Mediano	Cálido	Medio	Media	Poca	Poca
Mediano	Caluroso	Fuerte	Media	Media	Poca
Grande	Frío	Suave	Mucha	Media	Media
Grande	Caluroso	Medio	Mucha	Media	Poca

Figura 1: Tabla con las reglas del cafe mokaccino

4. Sistema de inferencia: A los valores difusos obtenidos en el punto uno les son aplicadas las reglas de inferencia definidas, entregando como resultado las consecuencias del sistema (valores difusos).
5. Defusificación: Finalmente, para lograr obtener los valores nítidos de salida es que se aplica el método COA (centro de área). Todo esto es realizado por la función compute.

### 3. Desarrollo

Para el caso particular con entradas:

1. Tipo de café: mokaccino
2. Intensidad: 4
3. Tamaño de la taza: 400 ml

Se generan los siguientes gráficos iniciales.

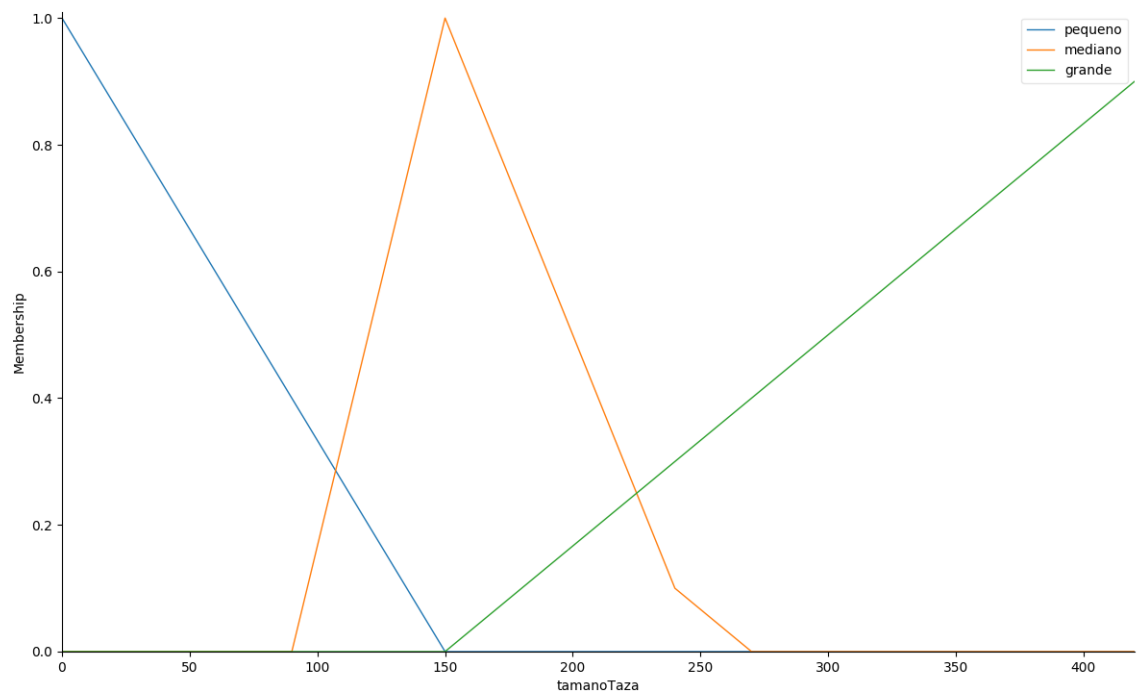


Figura 2: Gráfico tamaño taza con variable lingüística mL

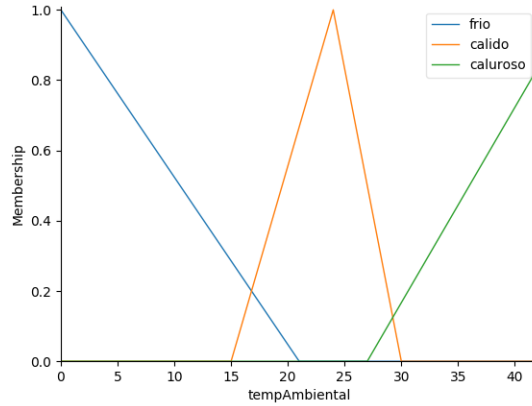


Figura 3: Gráfico temperatura ambiental con variable lingüística temperatura °C

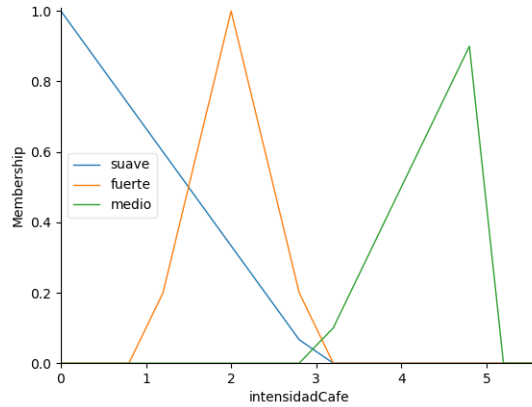


Figura 4: Gráfico intensidad café con variable lingüística grado de intensidad

Estos gráficos representan los valores de pertenencia con respecto a cada uno de los valores lingüísticos de las entradas. Luego de ser aplicadas las reglas de inferencia se obtienen los gráficos de salida de cada uno de los consecuentes, los cuales corresponden a valores difusos. Estos valores son calculados aplicando el mínimo a cada uno de los antecedentes de las reglas (inferencia de mamdani).

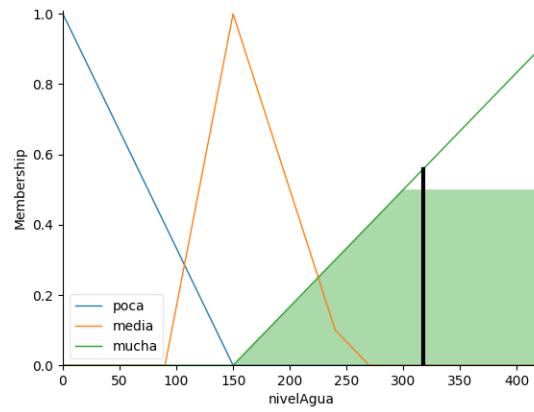


Figura 5: Gráfico cantidad de agua luego de aplicar las reglas

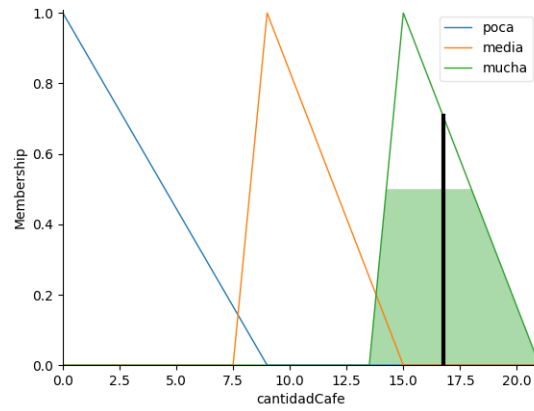


Figura 6: Gráfico cantidad de café luego de aplicar las reglas



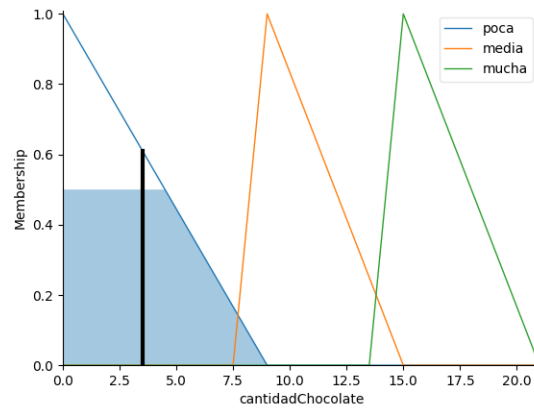


Figura 7: Gráfico cantidad de chocolate luego de aplicar las reglas

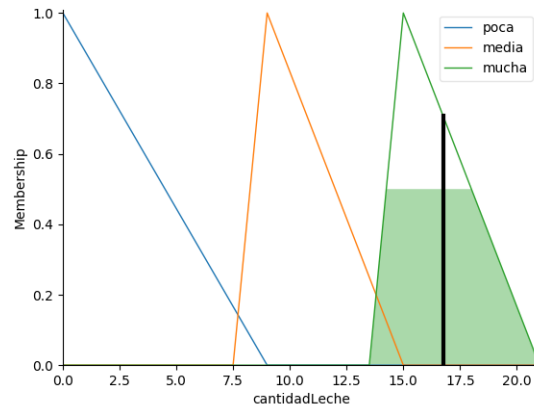


Figura 8: Gráfico cantidad de leche luego de aplicar las reglas

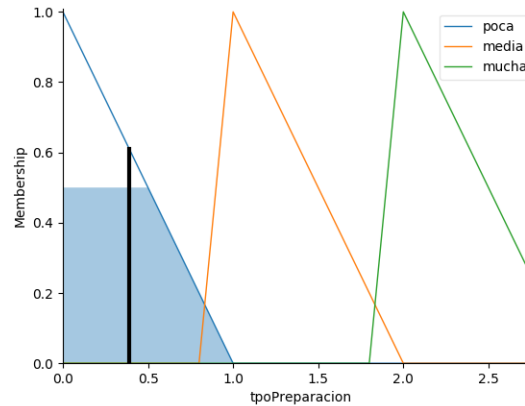


Figura 9: Gráfico tiempo de preparación luego de aplicar las reglas

Es posible determinar que los valores de pertenencia que se obtienen para cada uno de las variables son:

- Nivel de agua: 0.5 de pertenencia a mucha
- Cantidad de café: 0.5 de pertenencia a mucha
- Cantidad de chocolate: 0.6 de pertenencia a poca
- Cantidad de leche: 0.7 de pertenencia a mucha
- Tiempo de preparación: 0.6 de pertenencia a poca

Finalmente, es aplicada la defusificación, en donde se obtienen los valores en un archivo de salida con formato: **Cafe\_400.0\_mokaccino\_4.0\_10.0.txt** Con los valores nítidos:

- Nivel de agua: 330.0ml
- Cantidad de cafe: 18.0grs
- Cantidad de leche: 18.0grs
- Cantidad de chocolate: 4.5grs
- Tiempo de preparacion: 0.4minutos

## 4. Conclusiones

### 4.1. Lógica difusa v/s lógica clásica

Al momento de comparar la lógica difusa con la lógica clásica, las diferencias son evidentes. El hecho de tener un mayor rango de opciones en los cuales puede pertenecer un valor otorga mayor flexibilidad a la lógica difusa al momento de tomar decisiones, entregando a su vez mayor cantidad de opciones ante una cantidad  $n$  de entradas.

A su vez, se puede considerar que la lógica difusa, a diferencia de la lógica clásica, busca entregar resultados tratando de satisfacer la mayor cantidad de opciones posibles al momento de tomar una decisión, a diferencia de la lógica clásica, en la que al tener solo opciones binarias, las opciones intermedias quedan totalmente sesgadas.

Con respecto al taller 1, la lógica difusa es más compleja de implementar y si es que las reglas no cumplen con todos los casos posibles es necesario añadir una mayor cantidad, tal como ocurrió en el caso del presente laboratorio. A pesar de esto, los valores de salida entregada no están predefinidos y los resultados son intuitivos debido a que se asemeja al mundo real.

Finalmente, en base al trabajo realizado es posible concluir que la lógica difusa logra una mejor representación de la realidad, debido a que permite comprender un mayor rango de posibilidades cosa que con la lógica clásica no se puede obtener, además fue posible lograr desarrollar la máquina de café aplicando lógica difusa y comprender sus ventajas aplicativas.

## Bibliografía

[doc] Scikit-fuzzy. [Online] <https://pythonhosted.org/scikit-fuzzy/overview.html>.