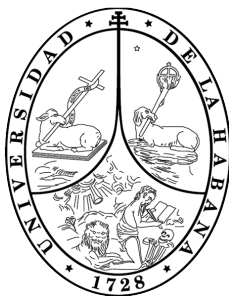


Proyecto de Simulación

LÓGICA E INFERENCIA DIFUSA

30 de noviembre de 2020



Facultad de Matemática y Computación
Universidad de la Habana

Autor: Daniel Alberto García Pérez **C412**
d.garcia@estudiantes.matcom.uh.cu

1. Propuesta del problema a solucionar

Para este proyecto se elaboró un problema ficticio relacionado con un fenómeno real, el cual fue simplificado para facilitar su implementación. Inspirado en la presa *Cauto el Paso*, de la provincia de *Granma*, la cual juega un papel importante en nuestra agricultura, red hidráulica y en el control del río *Cauto*. Se quiere automatizar el proceso de desviación del agua de un río hacia una presa, con el principal objetivo de evitar las crecidas del mismo en temporada lluviosa.

2. Sistema de inferencia difusa propuesto

En nuestro sistema se tienen en cuenta las siguientes variables lingüísticas con sus categorías:

- **Nivel del río:** *bajo, normal, crecido*
- **Probabilidad de lluvia:** *baja, probable, alta*
- **Intensidad de las lluvias:** *leves, intensas*
- **Velocidad de desvío:** *lento, normal, rápido*

Las funciones que indican el grado de pertenencia a cada categoría se muestra a continuación:

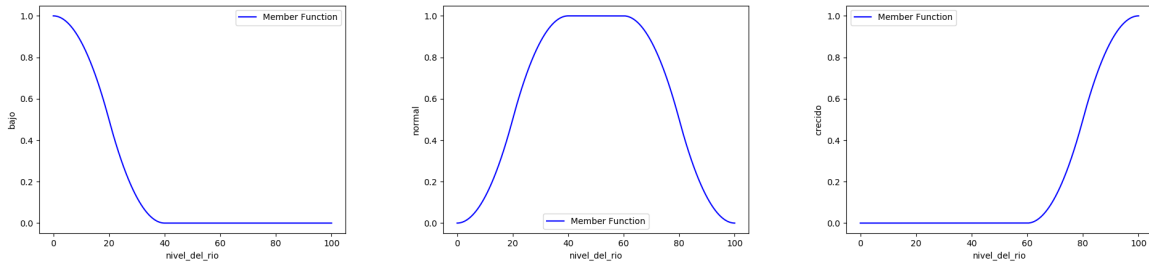


Figura 1: Nivel del río. Indica el porcentaje del nivel total del mismo que está lleno, valores mayores que 100, indican un desborde del río.

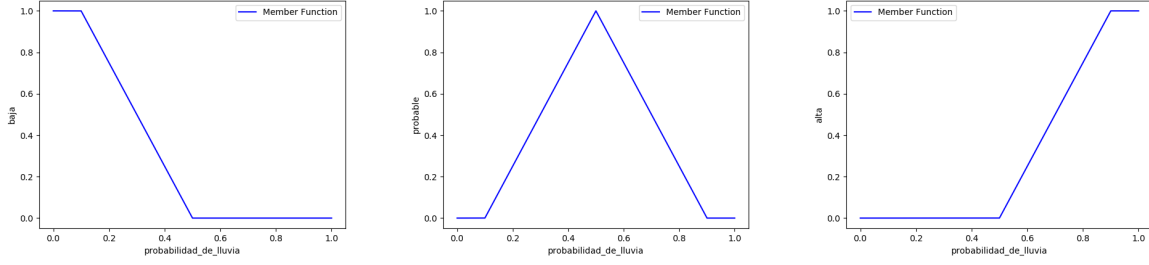


Figura 2: Probabilidad de lluvia. Es sin más un número en el intervalo $(0, 1)$.

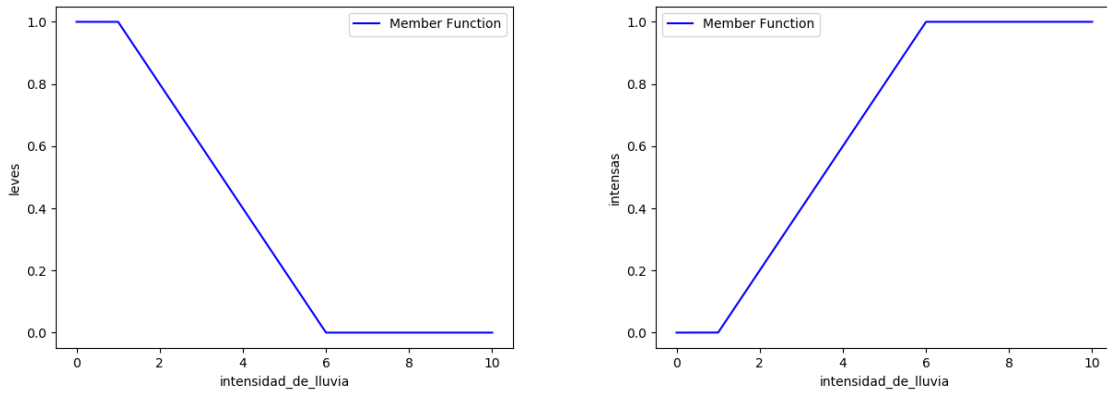


Figura 3: Intensidad de las lluvias. Es un valor que indica el porciento(del nivel total) en que aumenta el nivel del río cuando llueve.

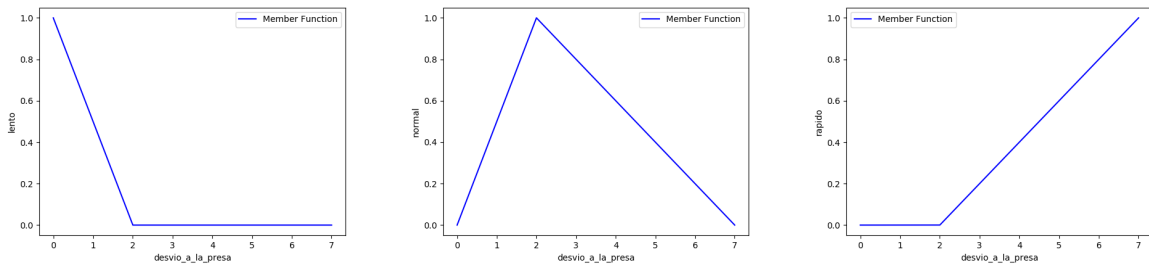


Figura 4: Velocidad de desvío. Indica el porciento(del nivel total) del agua del río que se desvía en un día.

Variables de entrada: *Nivel del río, probabilidad de lluvia e intensidad de las lluvias.*

Variable de salida: *Velocidad de desvío.*

Reglas:

- Si el nivel del río es bajo entonces la velocidad de desvío es lenta,
- Si el nivel del río es normal y, la probabilidad de lluvia es poca o probable entonces la velocidad de desvío es lenta,
- Si el nivel del río es normal, la probabilidad de lluvia es alta y la intensidad de las lluvias es leve entonces la velocidad de desvío es lenta,
- Si el nivel del río es normal, la probabilidad de lluvia es alta y la intensidad de las lluvias es intensa entonces la velocidad de desvío es normal,
- Si el nivel del río es crecido entonces la velocidad de desvío es rápida.

3. Métodos de inferencia, agregación y defusificación

Para solucionar el problema anteriormente propuesto, y como parte del proyecto, se implementó un módulo en Python dotado con herramientas básicas para el trabajo y la resolución de sistemas de inferencia difusa.

Se tiene en el archivo `fuzzy_number.py` varias definiciones de funciones (números difusos) que son frecuentemente usados como función de pertenencia de un conjunto difuso, tales como: `FuzzyTrapezoidal`, `FuzzyTriangular`, `FuzzySigmoidal`, etc. Aquí también se encuentran algunas operaciones básicas para operar con estas funciones, las cuales nos serán útiles a la hora de la inferencia y la agregación.

En el fichero `fuzzy_logic.py` es donde se implementan una serie de clases que nos servirán para crear predicados difusos, mediante el uso de las conjunciones y disyunciones de conjuntos difusos. Muy útil pues, el antecedente de cada regla lo podemos expresar con ambas operaciones, en cambio de limitarnos solo a la conjunción.

Finalmente, en archivo `fuzzy_inference.py`, tenemos la clase `FuzzySet` donde se maneja el concepto de *conjunto difuso*, hereda de `FuzzyPredicate`, pues pertenecer a un conjunto difuso, es el más simple de los predicados, y a partir de estos podemos construir predicados más complejos. Los objetos de tipo `FuzzySet` constan de una función de grado de pertenencia, el nombre del dominio de dicha función y la categoría a la que se mide la pertenencia; también se proveen en esta clase los tres métodos de defusificación más conocidos, el método de máximos, centroide y bisección.

La clase `LinguisticVariable` representa a las variables lingüísticas, constan de un nombre, y varias categorías, cada una con una función de pertenencia asociada. `FuzzyRule` se usa para representar una regla, consta de un antecedente(predicado) y un consecuente(conjunto difuso). Para representar un sistema difuso se tiene la clase `FuzzySystem`, la

cual se construye pasando como parámetros las variables lingüísticas que forman parte del mismo, indicando en cada caso si son de entrada o salida, para agregar una regla se usa el método `add_rule` o mediante el operador `%=`. Dicha clase esta dotada de los métodos de inferencia de *Mamdani* y *Larsen*, al ser invocados, se infieren cada una de las reglas, obteniendo conjuntos difusos los cuales se agregan utilizando el operador de máximos, resultando en un conjunto difuso(**FuzzySet**) por cada variable de salida, el cual como ya mencionamos, esta dotado de los métodos de defusificación.

4. La simulación

El código de la simulación se encuentra en `cauto.py`, donde se puede ver con claridad como usar el módulo para definir las variables, el sistema, y las reglas del mismo.

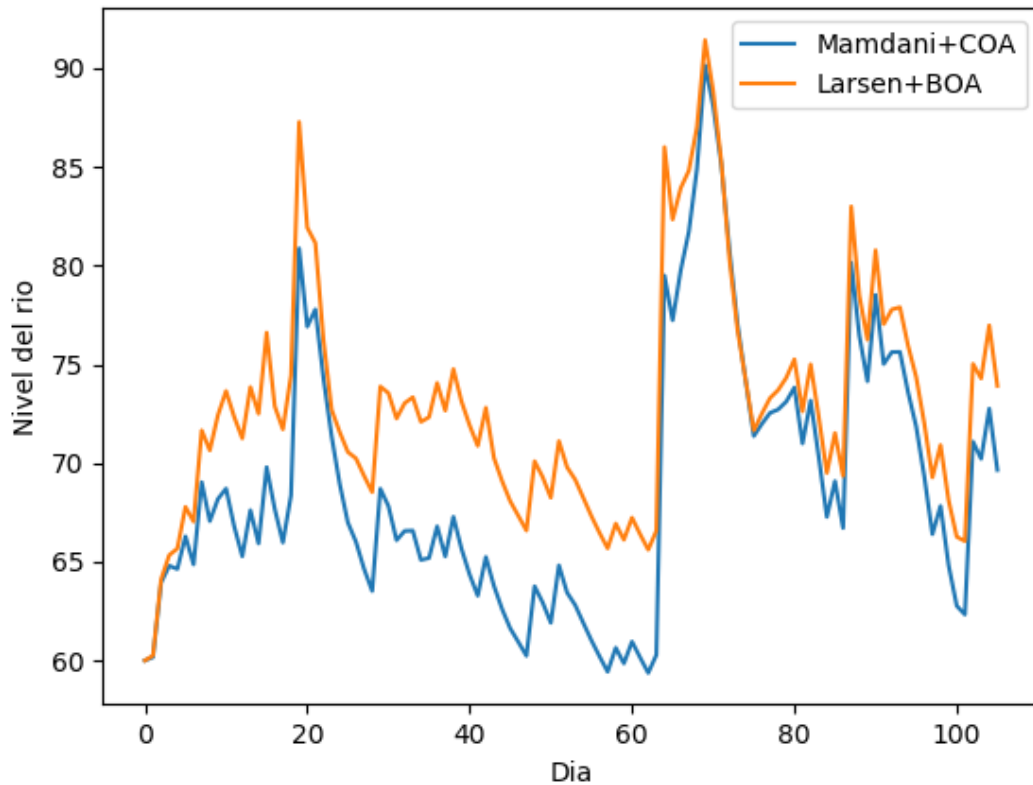


Figura 5: Simulación en temporada de lluvias

La simulación se realizó para 105 días, en los primeros 42, las lluvias son probables(0,5) y no tan intensas. En los siguientes 21 días la probabilidad aumenta hasta 0,7, pero las lluvias serán leves, en los últimos 42 días, la probabilidad se incrementa hasta 0,8 y las

lluvias se vuelven intensas. Se utilizó una *Bernoulli* con parámetro la probabilidad para decidir si llovía, la cantidad de agua que aumentaba el río se generó a partir de una exponencial con parámetro, el inverso del valor esperado de la intensidad de las lluvias.

Los resultados que se muestran corresponden a la misma simulación, en la cual se comparan los resultados si se toma *Mamdani* como inferencia y *centroide* como defusificación, o si en cambio se utiliza *Larsen* y el método de bisección.

Como se nota, la segunda fórmula, aunque no muy diferente, tiende a tener en río más crecido, pero en general, son efectivos, y detectan cuando las lluvias se vuelven frecuentes e intensas, para entonces acelerar la desviación.

El código fuente de la implementación se puede encontrar [aquí](#).

Índice

1. Propuesta del problema a solucionar	2
2. Sistema de inferencia difusa propuesto	2
3. Métodos de inferencia, agregación y defusificación	4
4. La simulación	5