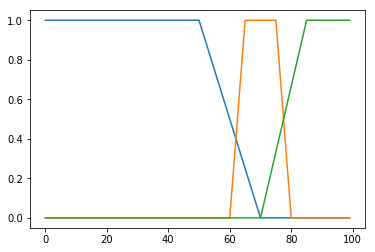
Propuesta de Problema:

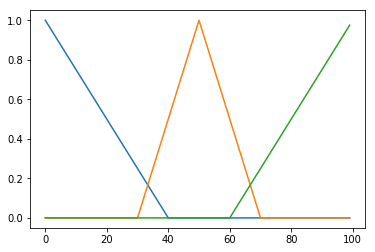
Es tarea de un director técnico saber cuándo tiene que sacar a un jugador del partido. Esta decisión puede afectar directamente el resultado del mismo, así como afectar la condición del equipo para el resto de la temporada. Las variables que pueden afectar esta decisión pueden ser varias (rendimiento del jugador en el partido, forma del jugador a lo largo de la temporada, marcador del partido, estado anímico del jugador, etc.) Dada la importancia de esta decisión, se propone un sistema difuso, con el objetivo de inferir si un jugador debe mantenerse en el campo a partir de datos de entrada que representen su rendimiento en el partido y su estado de forma a lo largo de la temporada, expresados en porciento.

Características del Sistema de Inferencia Propuesto:

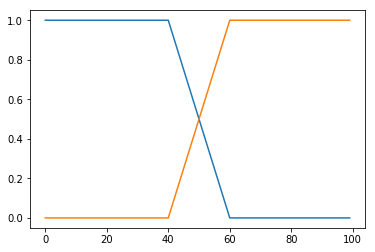
Para la modelación de este problema se utilizó un sistema de inferencia difusa tipo Mamdani, usando min. como como sustituto del and, max. como sustituto del or y 1-x como sustituto de la negación. El método de desdifusificacion usado fue el del centro de gravedad. Se utilizaron 2 variables lingüísticas de las anteriormente mencionadas en la parte antecedente, el rendimiento del jugador y su forma a lo largo de la temporada. La primera variable, “Rendimiento” está definida por los conjuntos difusos “Bajo”, “Medio” y “Alto”, mientras que la segunda variable “Forma” está definida por los conjuntos difusos “Mala”, “Regular” y “Buena”. Para la parte consecuente del sistema se tiene la variable lingüística “Decisión”, definida por los conjuntos difusos “Sacar” y “Dejar”.



La grafica anterior pertenece a la variable “Rendimiento”, donde las funciones azul, naranja y verde corresponden a las funciones de membresía de los conjuntos “Bajo”, “Medio” y “Alto” respectivamente.



La grafica anterior pertenece a la variable “Forma”, donde las funciones azul, naranja y verde corresponden a las funciones de membresía de los conjuntos “Mala”, “Regular” y “Buena” respectivamente.



La grafica anterior pertenece a la variable “Decisión”, donde las funciones azul y naranja corresponden a las funciones de membresía de los conjuntos “Sacar” y “Dejar”.

Para la inferencia de esta decisión se recopilaron una serie de reglas:

* Si el rendimiento es malo, entonces saca al jugador.
* Si el rendimiento es medio y la forma es baja, entonces saca al jugador.
* Si el rendimiento es alto, entonces deja al jugador.
* Si el rendimiento es medio y la forma es alta, entonces deja al jugador.
* Si el rendimiento es medio y la forma es regular, entonces deja al jugador.

A partir de las cuales se aplicaría el método de Mamdani, y luego se procedería a desdifusificar utilizando el método del centro de gravedad.

Principales Ideas Seguidas para la Implementación del Sistema

* Las funciones de membresía de los conjuntos difusos están representadas con arreglos, donde a la posición i, le corresponde f(i)
* Para almacenar los grados de pertenencia de una variable a un conjunto difuso, se utilizaron diccionarios por cada variable lingüística, que tenían como llaves los nombres de los conjuntos difusos de dicha variable, para ganar en naturalidad.
* Se implementaron los métodos de agregación de Mamdani, Takagi-Sugeno-Kang y Tsukamoto.
* Mamdani:
  + Recibe como parámetros los coeficientes asociados a los consecuentes, sus funciones de membresía y una función para desdifusificar (Se implementaron todas las variantes de los máximos, el bisector del área y dos variantes del centro de gravedad, una que utiliza la integral y otra las coordenadas cartesianas de los puntos)
  + Uso de las bibliotecas numpy y scipy.
  + Uso de la función numpy.fmin para truncar las funciones de membresía de los consecuentes por sus respectivos coeficientes. (En caso de que un consecuente tenga varios coeficientes, nos quedamos con el máximo entre ellos para no hacer operaciones innecesarias en la fase de agregación)
  + Uso de la función numpy.fmax para la agregación de todas las funciones, ya truncadas, de los consecuentes.
  + Uso de la función scipy.integrate.trapz para calcular la integral cuando se necesario.
* TSK:
  + Recibe como parámetros los coeficientes de los consecuentes, así como una lista de funciones lineales, una por cada consecuente.
* Tsukamoto:
  + Recibe como parámetros los coeficientes de los consecuentes y las funciones miembro de los mismos.
* Uso del módulo matplotlib.pyplot, para la traficación de las diferentes funciones.