**Proyecto Final**

**Presentado por:**

* Camila Lozano Jiménez - código 20201020161.
* Juan Felipe Rodríguez Galindo - código 20181020158.

1. **Definición de valores para el ejercicio:** En este caso, se toman 7 y 5 como cantidades de antecedentes y consecuentes.
2. Para definir, dentro de los posibles estados, los de la retina que determinan la presencia de la retinopatía diabética tenemos las siguientes convenciones que serán nuestras variables de entrada:
   1. Espesor macular
   2. Quistes intrarretinianos
   3. Elipsoides
   4. Desorganización de las capas de la retina

Y la variable de salida, nivel de la enfermedad, estará determinada por las siguientes condiciones:

* 1. Temprana
  2. Avanzada
  3. Severa
  4. Atrófica

Teniendo en cuenta lo anterior, se realizaron 4 configuraciones para determinar la presencia de la retinopatía diabética, las cuales se presentan a continuación:

1. **CONFIGURACIÓN 1:**

Se realiza el código de Matlab para determinar la presencia de la retinopatía diabética:

% Ejercicio de taller con base en el proyecto de detección de retinopatía

% dibética

% implemmentado en comandos

% Punto 5 Corte 1

close all

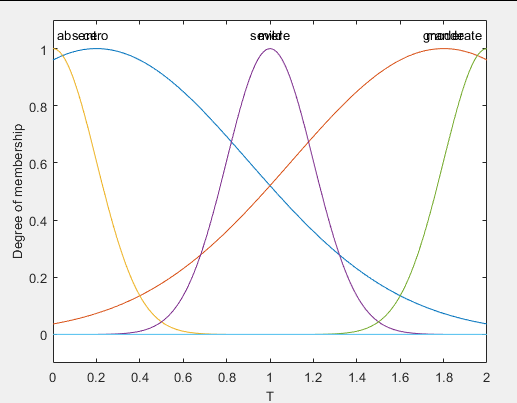
clear all

warning('off')

% Sistema

a = newfis('Retinopatía dibética')

Graficando se puede observar:



Luego se realiza el código en Matlab para cada una de las entradas:

% Variable de entrada : Thickness

a=addvar(a,'input','T',[0 2]); % Rango de espesor

% Funciones de pertenencia

a=addmf(a,'input',1,'cero','gaussmf',[0.7 0.2]);

a=addmf(a,'input',1,'grande','gaussmf',[0.7 1.8]);

plotmf(a,'input',1)

% Variable de entrada : Cyst

a=addvar(a,'input','Q',[0 3]); % Quistes metarretineanos

% Funciones de pertenencia

a=addmf(a,'input',1,'absent','gaussmf',[0.2 0]);

a=addmf(a,'input',1,'mild','gaussmf',[0.2 1]);

a=addmf(a,'input',1,'moderate','gaussmf',[0.2 2]);

a=addmf(a,'input',1,'severe','gaussmf',[0.2 3]);

plotmf(a,'input',1)

% Variable de entrada : EZ

a=addvar(a,'input','EZ',[0 2]); % Elipsoides

% Funciones de pertenencia

a=addmf(a,'input',1,'intact','gaussmf',[0.2 0]);

a=addmf(a,'input',1,'disrupted','gaussmf',[0.2 1]);

a=addmf(a,'input',1,'absent','gaussmf',[0.2 2]);

plotmf(a,'input',1)

% Variable de entrada : DRIL

a=addvar(a,'input','DRIL',[0 2]); % Desaorganización de las capas de la retina

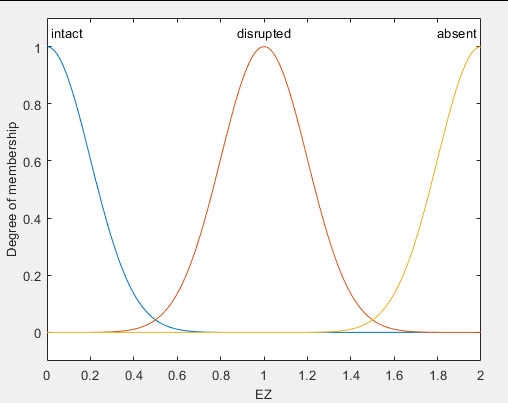
% Funciones de pertenencia

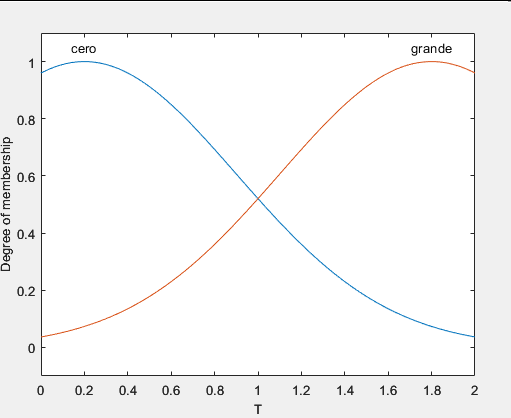
a=addmf(a,'input',1,'whitout','gaussmf',[0.7 0.2]);

a=addmf(a,'input',1,'with','gaussmf',[0.7 1.8]);

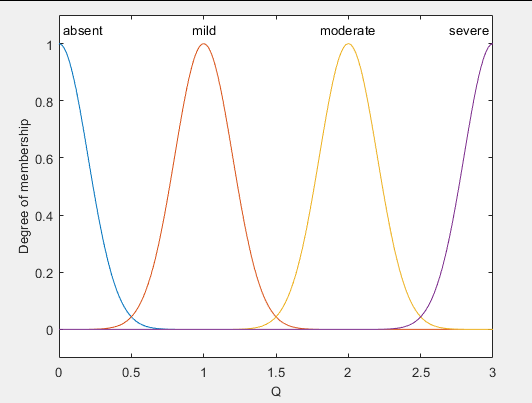
plotmf(a,'input',1)

Graficando se puede observar:



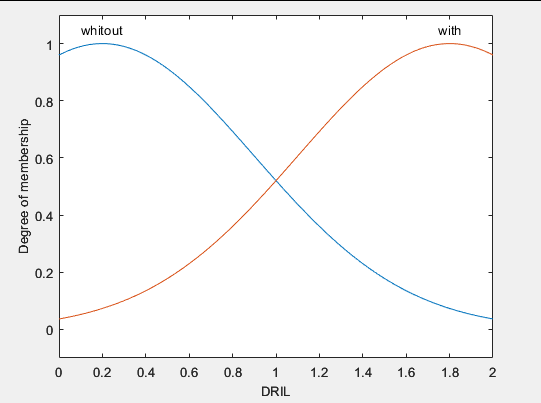


Thickness



Quist

EZ



DRIL

Finalmente, se realiza el nivel de retinopatía diabética

% Variable de salida : Retinopatía diabética

a=addvar(a,'input','nivel de retinopatia',[0 4]); % Quistes metarretineanos

% Funciones de pertenencia

a=addmf(a,'input',1,'absent','trimf',[0 0.5 1]);

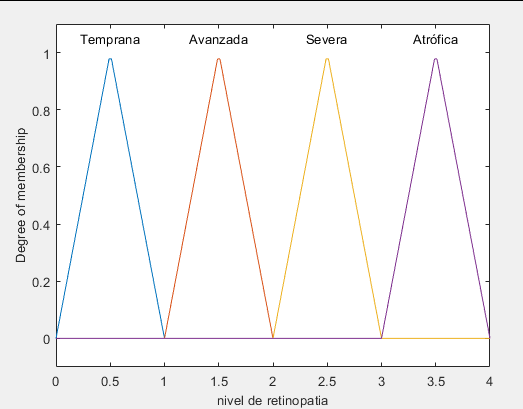
a=addmf(a,'input',1,'mild','trimf',[1 1.5 2]);

a=addmf(a,'input',1,'moderate','trimf',[2 2.5 3]);

a=addmf(a,'input',1,'severe','trimf',[3 3.5 4]);

plotmf(a,'input',1)

y se grafica de la siguiente manera:



Nivel de retinopatía.

Posteriormente, se definen las reglas de inferencia de la siguiente manera:

* Columna 1: Temprana
* Columna 2: Avanzada
* Columna 3: Severa
* Columna 4: Profunda

Estas reglas de inferencia se basaron en [An optical coherence tomography-based grading of diabetic maculopathy proposed by an international expert panel: The European School for Advanced Studies in Ophthalmology classification](https://pubmed.ncbi.nlm.nih.gov/31718271/), TABLA 2

Se definen las siguientes reglas por código:

ruleList = [

1 1 1 1 1 1 1 % Temprana

1 2 1 1 2 1 1 % Avanzada

2 2 1 2 3 1 1 % Severa

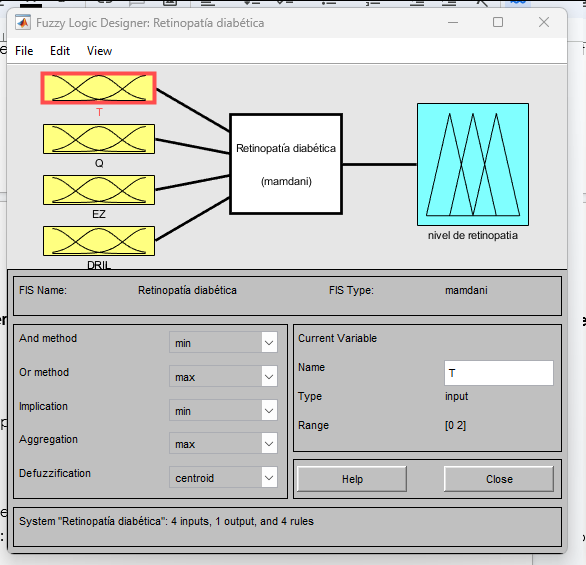
2 3 2 2 4 1 1 % Atrófica

];

Luego, se guarda el sistema difuso en la variable ‘a’:

a = addRule(a,ruleList);

Y se observa el sistema por medio de la interfaz gráfica:



En cuanto a los datos de entrada, se tomarán del dataset propuesto:

**Cálculo del MSE**

La fórmula para calcular el Error Cuadrático Medio (MSE) es:

Donde *Yobservado,i* es el valor observado *Ypredicho,i* es el valor predicho por el modelo para la i-ésima observación.

Se lleva la implementación a MATLAB

%% PRUEBA DE CONCEPTO

% Observaciones del nivel de retinopatía

observaciones = [0.5; 2.0; 1.5; 0.2];

% Datos de entrada

entradas = [

0.3 1.0 0.1 0.2;

1.5 2.0 1.0 1.8;

1.0 3.0 1.5 0.8;

0.2 0.5 0.0 0.1];

% Evaluar el sistema difuso para cada fila de entradas

predicciones = zeros(size(observaciones));

for i = 1:size(entradas, 1)

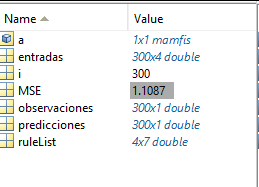
predicciones(i) = evalfis(entradas(i, :), a);

end

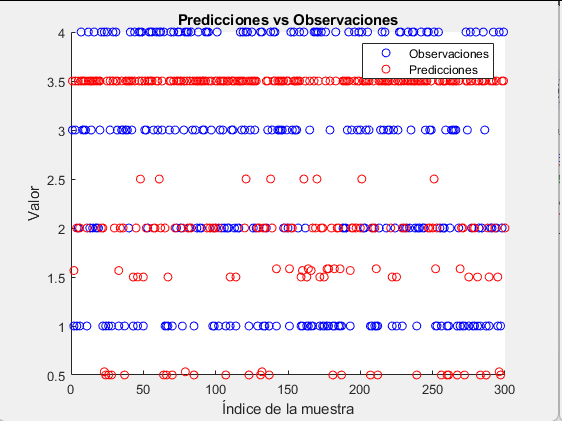
% Calcular el MSE

MSE = mean((observaciones - predicciones).^2);

La cual da como resultado:



Gráfica de predicciones Vs observaciones



1. **CONFIGURACIÓN 2:**

Se realiza la modificación sobre el código de Matlab para determinar la presencia de la retinopatía diabética, se modifica la función de pertenencia de la propiedad “T”:

% Variable de entrada : Thickness

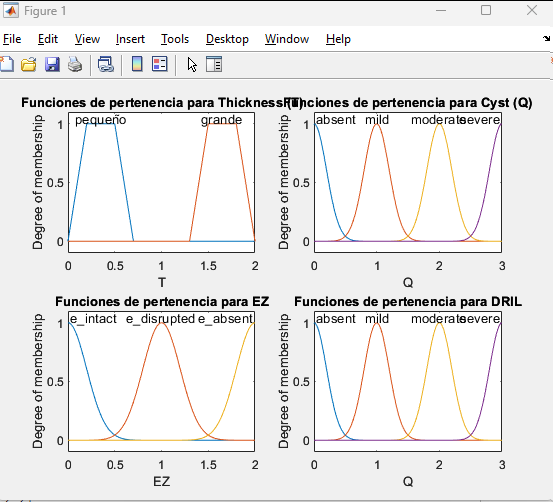
a = addvar(a, 'input', 'T', [0 2]); % Rango de espesor

% Funciones de pertenencia

a = addmf(a, 'input', 1, 'pequeño', 'trapmf', [0 0.2 0.5 0.7]);

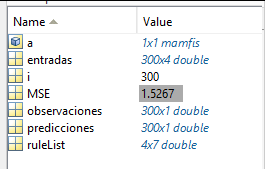
a = addmf(a, 'input', 1, 'grande', 'trapmf', [1.3 1.5 1.8 2]);

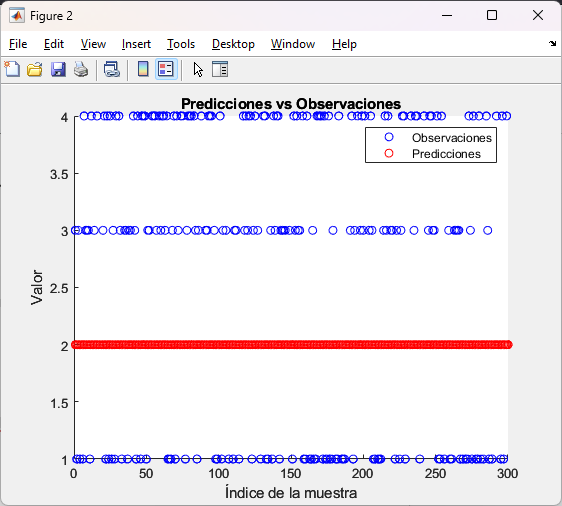
Graficando se puede observar:



Posteriormente, se utiliza el mismo código de clasificación del nivel de retinopatía diabética, las reglas de inferencia y los datos de entrada del dataset utilizado en la CONFIGURACIÓN 1.

Por último se realiza el cálculo del MSE como se mostró anteriormente en la CONFIGURACIÓN 1, dando como resultado en este caso:



Gráfica de predicciones Vs observaciones

1. **CONFIGURACIÓN 3:**

Se realiza la modificación sobre el código de Matlab para determinar la presencia de la retinopatía diabética, se modifica la función de pertenencia:

% Sistema

a = newfis('Retinopatía diabética');

% Variable de entrada : Thickness

a=addvar(a,'input','T',[0 2]); % Rango de espesor

% Funciones de pertenencia

a=addmf(a,'input',1,'cero','gaussmf',[0.7 0.2]);

a=addmf(a,'input',1,'grande','gaussmf',[0.7 1.8]);

% Variable de entrada : Cyst

a=addvar(a,'input','Q',[0 3]); % Quistes metarretineanos

% Funciones de pertenencia

a=addmf(a,'input',2,'absent','gaussmf',[0.2 0]);

a=addmf(a,'input',2,'mild','gaussmf',[0.2 1]);

a=addmf(a,'input',2,'moderate','gaussmf',[0.2 2]);

a=addmf(a,'input',2,'severe','gaussmf',[0.2 3]);

% Variable de entrada : EZ

a=addvar(a,'input','EZ',[0 2]); % Elipsoides

% Funciones de pertenencia

a=addmf(a,'input',3,'e\_intact','gaussmf',[0.2 0]);

a=addmf(a,'input',3,'e\_disrupted','gaussmf',[0.2 1]);

a=addmf(a,'input',3,'e\_absent','gaussmf',[0.2 2]);

% Variable de entrada : DRIL

a=addvar(a,'input','DRIL',[0 2]); % Desorganización de las capas de la retina

% Funciones de pertenencia

a=addmf(a,'input',4,'without','gaussmf',[0.7 0.2]);

a=addmf(a,'input',4,'with','gaussmf',[0.7 1.8]);

% Variable de salida : Retinopatía diabética

a=addvar(a,'output','nivel de retinopatia',[0 4]);

% Funciones de pertenencia

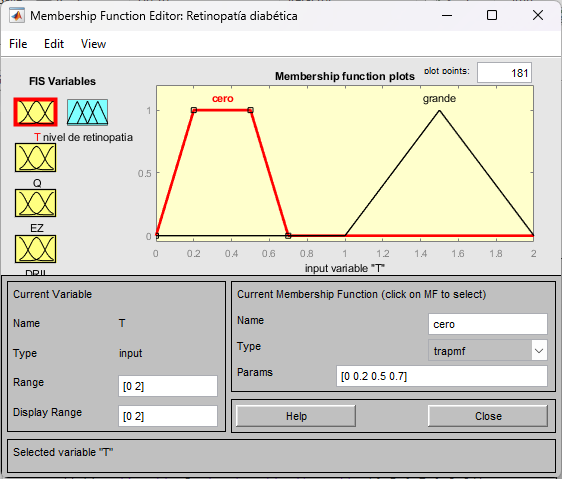
a=addmf(a,'output',1,'Temprana','trimf',[0 0.5 1]);

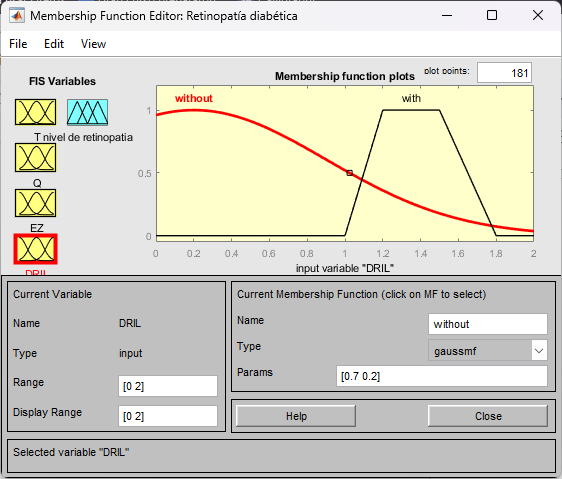
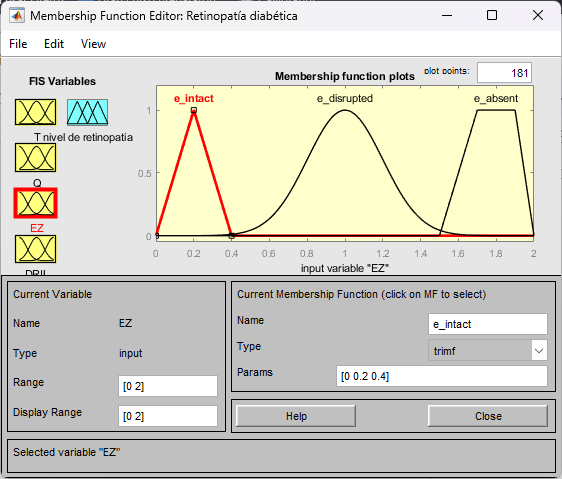
a=addmf(a,'output',1,'Avanzada','trimf',[1 1.5 2]);

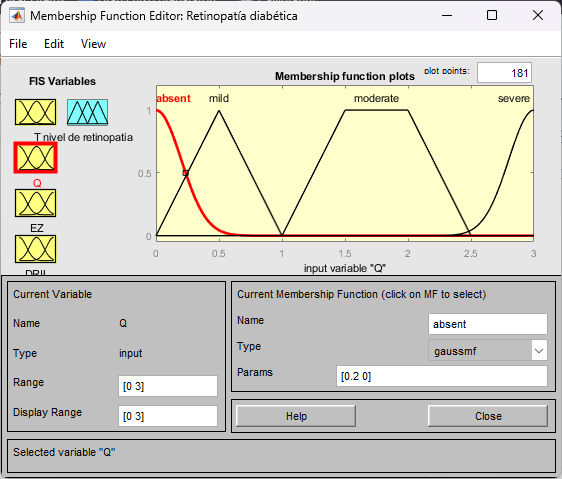
a=addmf(a,'output',1,'Severa','trimf',[2 2.5 3]);

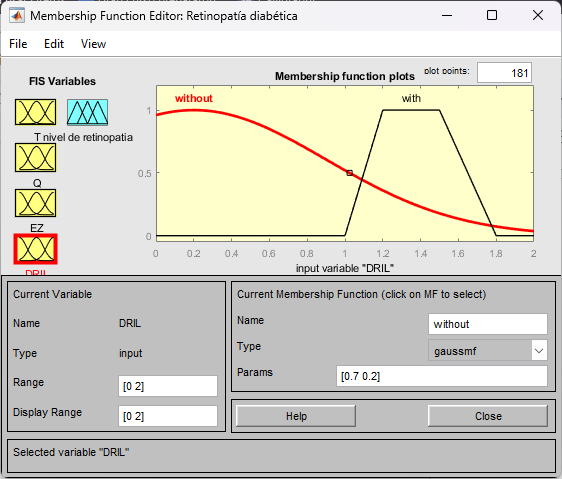
a=addmf(a,'output',1,'Atrófica','trimf',[3 3.5 4]);

Posteriormente, se utiliza el mismo código de clasificación del nivel de retinopatía diabética, las reglas de inferencia y los datos de entrada del dataset utilizado en la CONFIGURACIÓN 1.

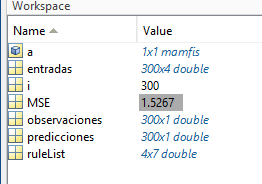




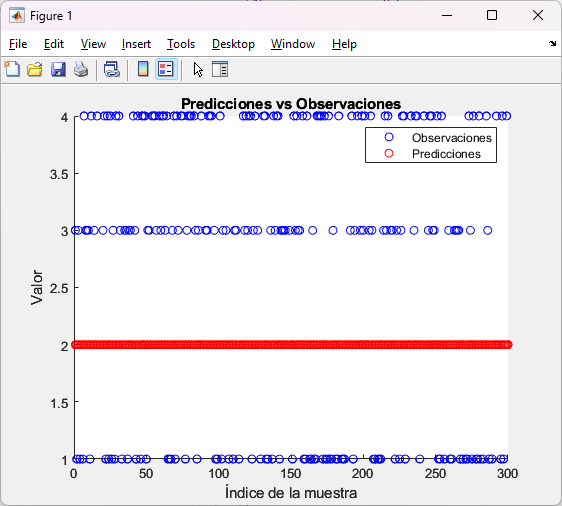




Por último se realiza el cálculo del MSE como se mostró anteriormente en la CONFIGURACIÓN 1, dando como resultado en este caso:



Y el gráfico de correlación entre lo predicho y las observaciones es :



1. **CONFIGURACIÓN 4:**

Se realiza la modificación sobre el código de Matlab para determinar la presencia de la retinopatía diabética, se modifica la función de pertenencia:

% Variable de entrada : Thickness

a = addvar(a, 'input', 'T', [0 2]); % Rango de espesor

% Funciones de pertenencia

a = addmf(a, 'input', 1, 'cero', 'gaussmf', [0.5 0.1]);

a = addmf(a, 'input', 1, 'grande', 'trapmf', [1 1.3 1.7 2]);

% Variable de entrada : Cyst

a = addvar(a, 'input', 'Q', [0 3]); % Quistes metarretineanos

% Funciones de pertenencia

a = addmf(a, 'input', 2, 'absent', 'trimf', [0 0.1 0.2]);

a = addmf(a, 'input', 2, 'mild', 'trapmf', [0.5 0.7 1 1.2]);

a = addmf(a, 'input', 2, 'moderate', 'gaussmf', [0.2 2]);

a = addmf(a, 'input', 2, 'severe', 'trimf', [2.5 2.8 3]);

% Variable de entrada : EZ

a = addvar(a, 'input', 'EZ', [0 2]); % Elipsoides

% Funciones de pertenencia

a = addmf(a, 'input', 3, 'e\_intact', 'trapmf', [0 0.1 0.3 0.4]);

a = addmf(a, 'input', 3, 'e\_disrupted', 'trimf', [0.7 1 1.3]);

a = addmf(a, 'input', 3, 'e\_absent', 'gaussmf', [0.2 2]);

% Variable de entrada : DRIL

a = addvar(a, 'input', 'DRIL', [0 2]); % Desorganización de las capas de la retina

% Funciones de pertenencia

a = addmf(a, 'input', 4, 'without', 'trimf', [0 0.2 0.4]);

a = addmf(a, 'input', 4, 'with', 'gaussmf', [0.7 1.8]);

% Variable de salida : Retinopatía diabética

a=addvar(a,'output','nivel de retinopatia',[0 4]);

% Funciones de pertenencia

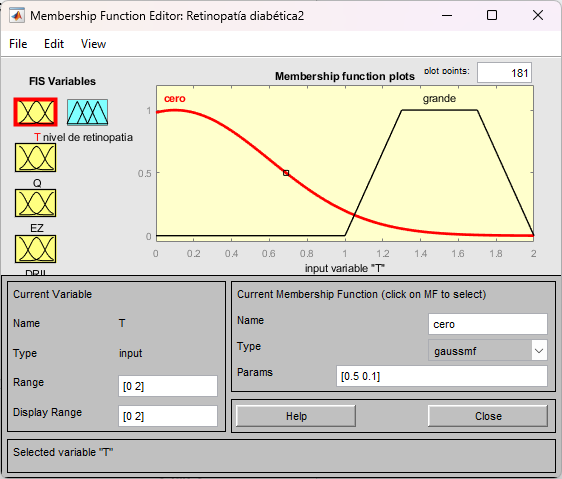
a=addmf(a,'output',1,'Temprana','trimf',[0 0.5 1]);

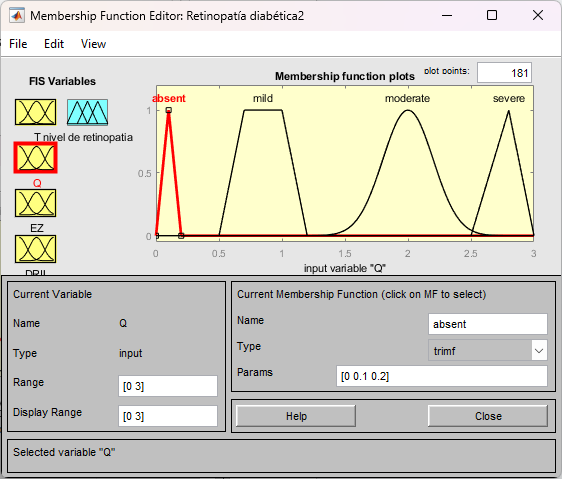
a=addmf(a,'output',1,'Avanzada','trimf',[1 1.5 2]);

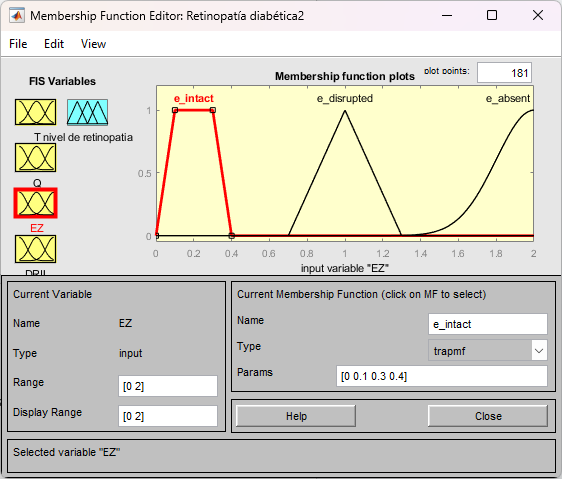
a=addmf(a,'output',1,'Severa','trimf',[2 2.5 3]);

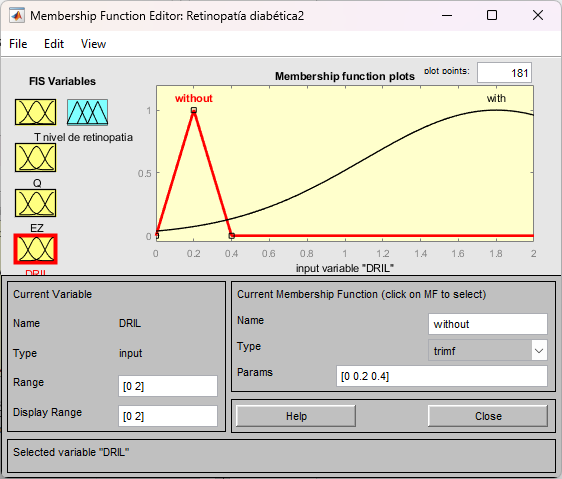
a=addmf(a,'output',1,'Atrófica','trimf',[3 3.5 4]);

Posteriormente, se utiliza el mismo código de clasificación del nivel de retinopatía diabética, las reglas de inferencia y los datos de entrada del dataset utilizado en la CONFIGURACIÓN 1.

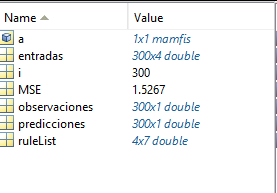




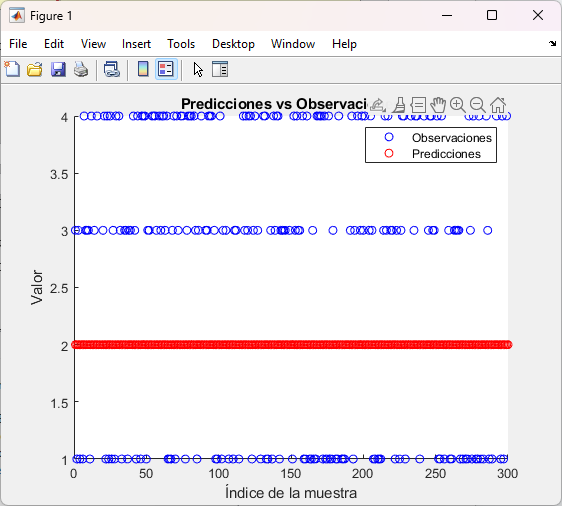




Por último se realiza el cálculo del MSE como se mostró anteriormente en la CONFIGURACIÓN 1, dando como resultado en este caso:



Se realiza la gráfica de correlación de las predicciones con lo observado:



**Análisis de Funciones de Membresía**

Las funciones gaussianas, trapezoidales y triangulares tienen diferentes propiedades que afectan la forma en que el sistema difuso mapea las entradas a las salidas. Por ejemplo, las funciones gaussianas son suaves y continuas, lo que podría ser más adecuado para modelar sistemas donde se espera una transición gradual entre estados.

Por otro lado, las funciones trapezoidales y triangulares tienen bordes más abruptos, lo que podría ser más adecuado para sistemas con transiciones más claras y definidas.

**Ajuste del Modelo**

No lograr un mejor ajuste después de cambiar la configuración inicial podría indicar que las funciones de membresía originales estaban más alineadas con la naturaleza del problema que estás tratando de resolver. Esto podría ser un indicativo de que la elección de funciones de membresía debe hacerse en función de un conocimiento más profundo del sistema que se está modelando.

**Importancia de la Experimentación**

La experimentación es crucial en la modelización difusa. Cambiar las funciones de membresía y observar cómo afectan al sistema puede proporcionar intuiciones valiosas sobre la estructura del problema. Sin embargo, es importante tener en cuenta que cada cambio debe ser validado con datos reales o simulaciones para asegurar que el modelo sigue siendo representativo del sistema real.

**CONCLUSIÓN:**

La elección de funciones de membresía en un sistema difuso es un aspecto crítico que puede influir significativamente en la calidad del modelo. No lograr un mejor ajuste tras cambiar las funciones sugiere que la configuración inicial podría haber sido más apropiada para el problema específico en cuestión. Este ejercicio subraya la importancia de entender tanto las matemáticas detrás de las funciones de membresía como la física o la lógica del sistema que se está modelando. En última instancia, la modelización difusa es tanto un arte como una ciencia, y encontrar el equilibrio adecuado entre diferentes tipos de funciones de membresía puede requerir una combinación de experimentación empírica y comprensión teórica.