



# Monitoreo y Control de Reactor CSTR aplicando técnicas de IA

# Agenda

- Problemática
- Objetivo General
- Objetivos Específicos
- Modelo de Interacción
- Técnicas aplicadas IA
  - Red Neuronal
  - Lógica Difusa
  - Multiagentes
- Evidencias del Trabajo Realizado
- Resultados
- Conclusiones
- Referencias

## Problemática

Al ser tan diferentes, complejas y específicas las reacciones al interior de reactor CSTR (continuous stirred tank reactor), resulta muy difícil realizar un control MIMO con técnicas clásicas para este tipo de sistema

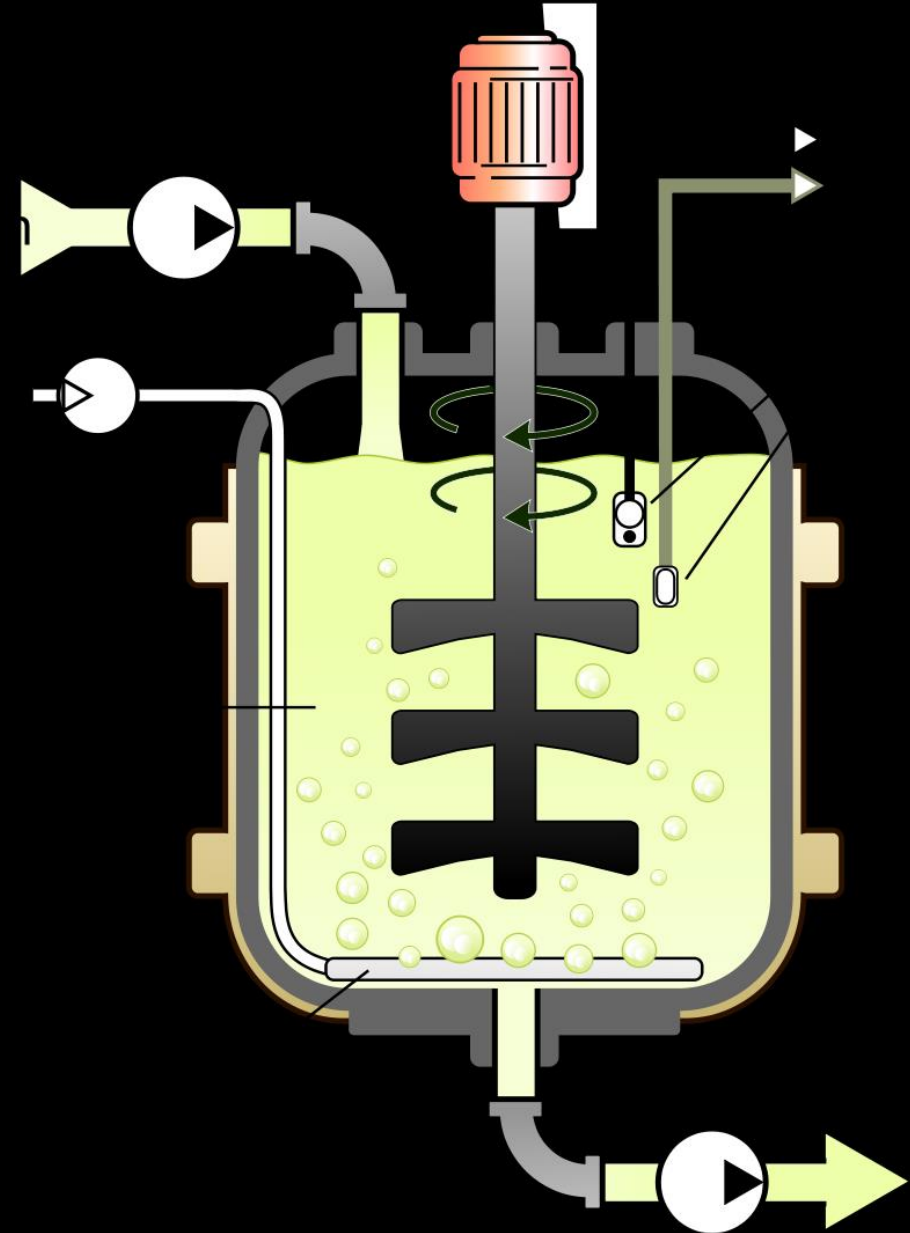


Figura1:  
Modelo CSTR



# Objetivo General

- Controlar la energía térmica producida por las reacciones químicas que se dan al interior de un reactor de tanque agitado continuo (CSTR), el volumen dentro del tanque para que se de dicha reacción y la concentración de salida del producto de la reacción, evaluando el desempeño de diferentes técnicas de Inteligencia Artificial y escogiendo de forma desatendida la más conveniente según el caso.

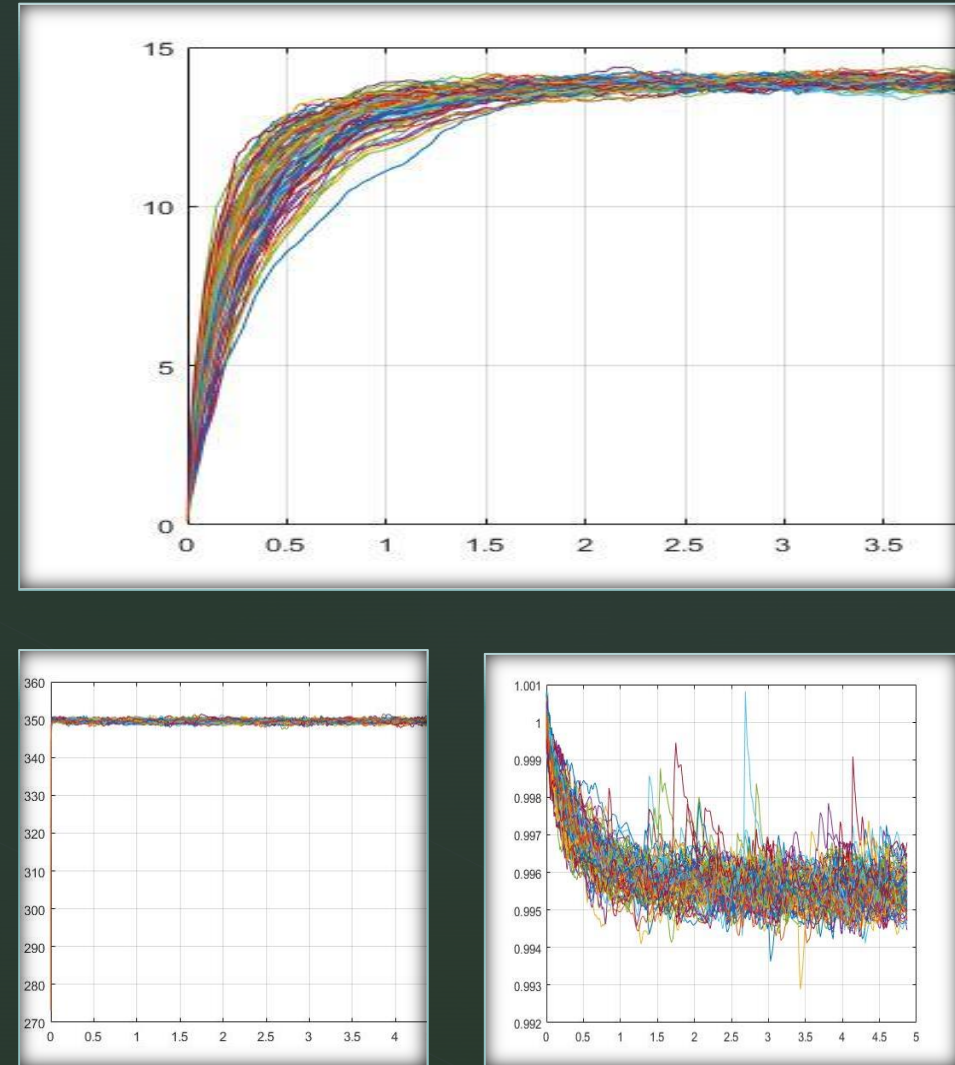


Figura2:  
Resultados de Simulaciones CSTR, Volumen, Temperatura  
y Concentración

## Objetivos Específicos

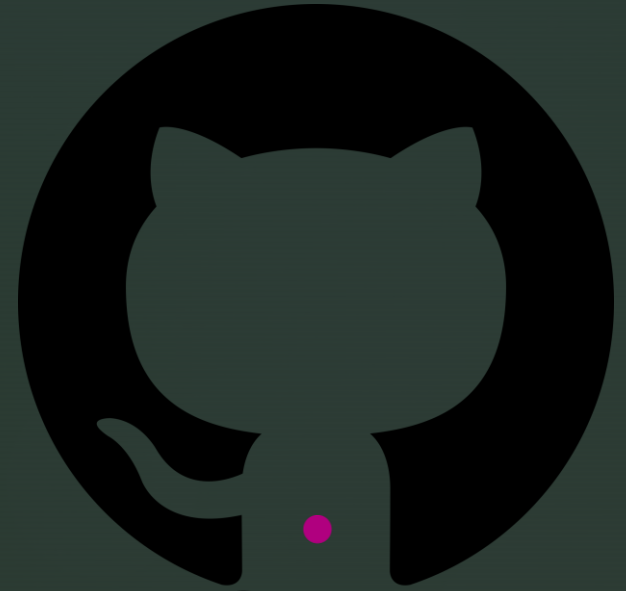
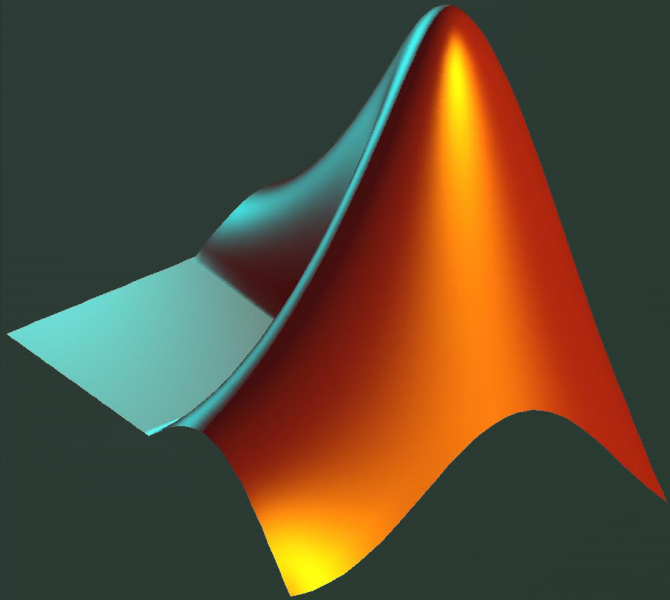
- ▶ Seleccionar por la técnica de Redes Neuronales los valores óptimos de Flujo de Entrada y Flujo en la Chaqueta, para el control de la variable Temperatura a la salida del reactor CSTR.
- ▶ Asociar a las Temperaturas Optimas (variable Critica) los flujos de entrada, para que por medio de la técnica de Lógica Difusa se pueda realizar el control del Volumen y Concentración a la salida del reactor CSTR, según el punto de operación.
- ▶ Diseñar una interfaz para los controladores.
- ▶ Diseñar un control Multi Agente para determinar la evolución del sistema propuesto y que acciones se deben aplicar.

# MULTI-AGENTE



## Modelo de Interacción

# Recursos Empleados



SPADE

turi





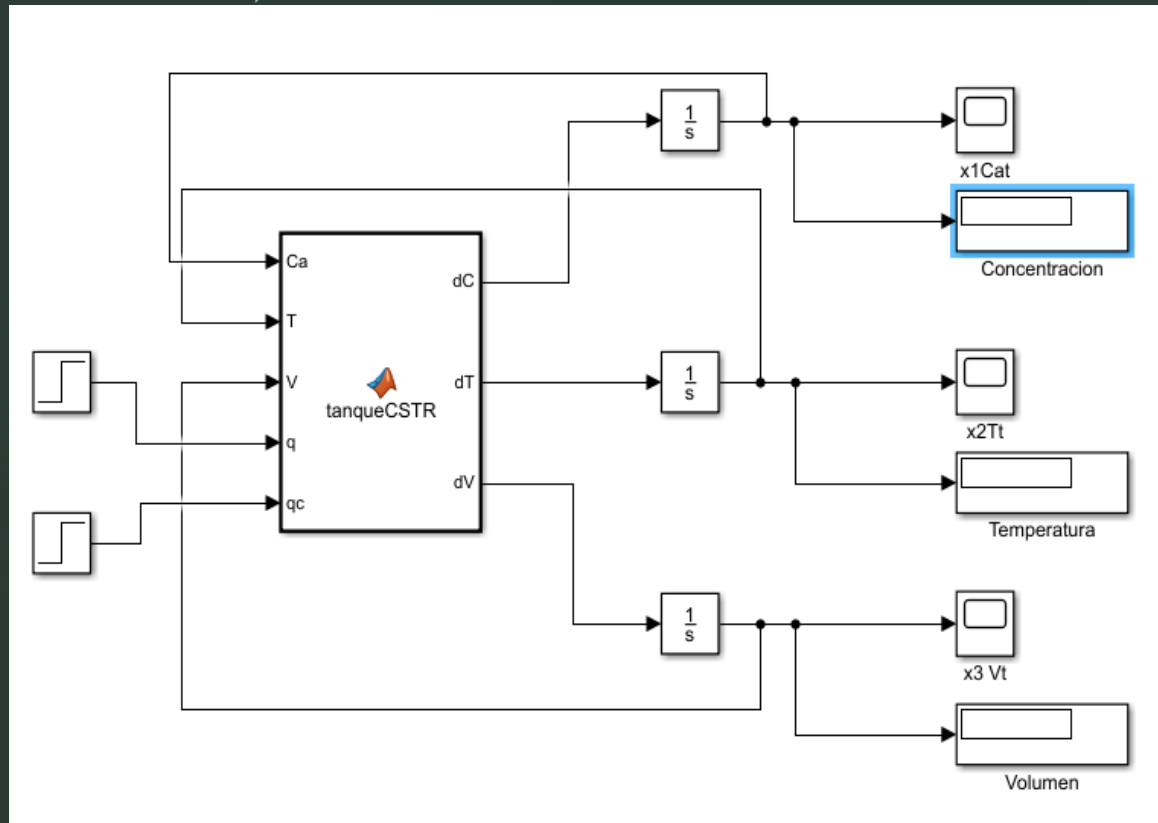


Figura3:  
Modelo Simulación en Matlab del reactor CSTR.

Figura4: Modelo matemático en  
Matlab del reactor CSTR.

$$\frac{dCa(t)}{dt} = \frac{q(t)}{V(t)} (Ca_0 - Ca(t)) - k_0 Ca(t) e^{-\frac{E}{RT(t)}}$$

$$\frac{dT(t)}{dt} = \frac{q(t)}{V(t)} (T_0 - T(t)) + k_1 Ca(t) e^{-\frac{E}{RT(t)}} + k_2 \frac{q(t)}{V(t)} \left( 1 - e^{-\frac{k_2}{q_c(t)}} \right) (T_{c0}(t) - T(t))$$

$$\frac{dV(t)}{dt} = q(t) - k_4 \sqrt{V(t)}$$

# Simulación

Iteration	Examples	Elapsed Time	Training-accuracy	Validation-accuracy	Training-recall@2	Validation-recall@2	Examples/second
1	40200	4.512952	0.011045	0.010467	0.021343	0.024289	890
2	40200	9.183193	0.013781	0.013821	0.026766	0.028354	860
3	40200	13.733645	0.014030	0.013821	0.027687	0.028354	883
4	40200	18.245986	0.014030	0.013821	0.027687	0.028354	890
5	40200	22.784773	0.014030	0.013821	0.027886	0.028354	885
6	40200	27.619832	0.014030	0.013821	0.027886	0.028354	831
7	40200	32.166859	0.014030	0.013821	0.027886	0.028354	884
8	40200	36.648143	0.014030	0.013821	0.027886	0.028354	897
9	40200	41.140933	0.014030	0.013821	0.027886	0.028354	894
10	40200	45.807867	0.014030	0.013821	0.027886	0.028354	861

Figura5:  
Creacion red Neuronal de 10 interacciones.

### Most frequent items from <SArray>

Value	Count	Percent
-1	39.527	98,82%
1	472	1,18%

Figura6:  
Clasificación de los datos.

# Red Neuronal

FlujoCh	FlujoRx	TCalificacion	Tempera
104.75	80.16	-1	273.15
106.81	83.468	-1	281.12
104.11	83.129	-1	287.85
103.81	82.207	-1	293.53
104.66	84.865	-1	298.55
106.06	80.703	-1	304.24
106.56	80.74	-1	312.91
105.53	83.209	-1	316.32
107.21	80.343	-1	322.3
102.2	80.212	-1	324.87

[39999 rows x 4 columns]

FlujoCh	FlujoRx	TCalificacion	Tempera
105.25	80.616	1	350.0
103.47	82.316	1	350.0
103.8	84.503	1	350.0
105.39	79.191	1	349.0
107.77	82.101	1	350.0
103.34	84.051	1	350.0
103.93	80.83	1	350.0
107.11	83.932	1	350.0
106.02	83.464	1	349.0
102.43	83.468	1	350.0

[472 rows x 4 columns]

Figura7:  
Tablas de clasificación de los datos y sus respectivos flujos.

# Fuzzy

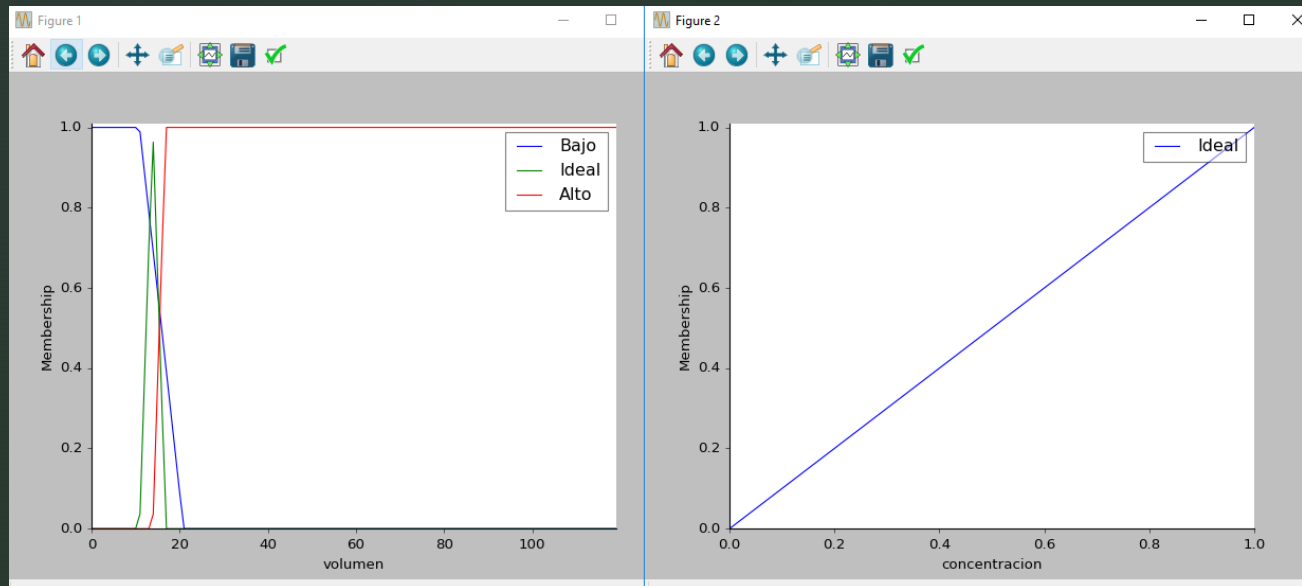
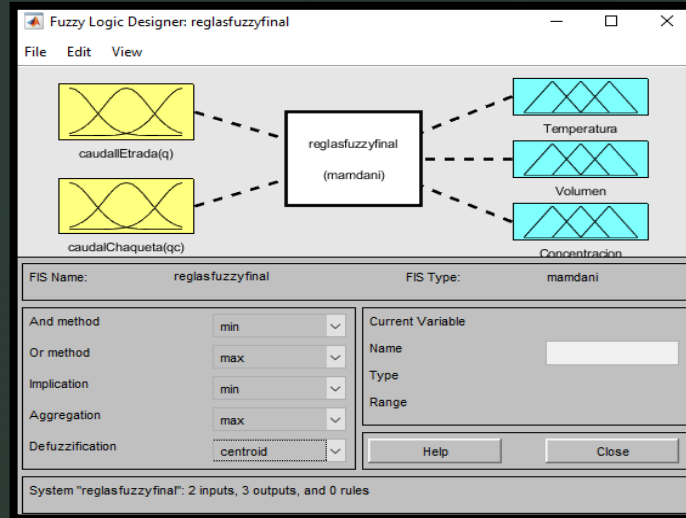


Fig8: Diagrama de la aplicación de la Técnica de Lógica Difusa al Modelo del tanque CSTR.

Temperatura			
FlujoINC\FlujoINR	Bajo	Ideal	Alto
Bajo	MuyBajo	MedioBajo	LigeraBajo
Ideal	MedioBajo	Ideal	MedioAlto
Alto	LigeraBajo	MedioAlto	MuyAlto

Concentración			
FlujoINC\FlujoINR	Bajo	Ideal	Alto
Bajo	Cero	Cero	Cero
Ideal	Cero	Ideal	Ideal
Alto	Cero	Ideal	Ideal

Volumen			
FlujoINC\FlujoINR	Bajo	Ideal	Alto
Bajo	Bajo	Ideal	Alto
Ideal	Ideal	Alto	Alto
Alto	Alto	Alto	Alto

# Multi-agente

Jupyter Entrega Last Checkpoint: 3 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Help Python 2

```
# valores aleatorios de Flujo Ch y Flujo Rx
m=Valores_Clasificados['FlujoCh']
n=Valores_Clasificados['FlujoRx']

print('caudal entrada')
print(choice(n)) #flujo Rx
print('caudal chaqueta')
print(choice(m)) #flujo Ch
```

FlujoCh	FlujoRx	TCalificacion	Temperatura
103.19	80.964	1	350.0
105.25	80.616	1	350.0
107.52	84.94	1	350.0
103.47	82.316	1	350.0
103.8	84.503	1	350.0
105.39	79.191	1	349.0
107.77	82.101	1	350.0
107.11	83.932	1	350.0
107.2	83.908	1	350.0
106.02	83.464	1	349.0

[491 rows x 4 columns]  
Note: Only the head of the SFrame is printed.  
You can use print\_rows(num\_rows=m, num\_columns=n) to print more rows and columns.  
caudal entrada  
79.111  
caudal chaqueta  
104.51

```
PS C:\Users\D A Santana\Downloads> python SuperAgente.py
Invalid debugflag given: always
[0ma@127.0.0.1:[spade] [31mThere is no SPADE platform at 127.0.0.1 . Agent dy
Invalid debugflag given: always
[0ma@127.0.0.1:[spade] [31mCould not register agent a@127.0.0.1 , error[0m
[0ma@127.0.0.1:[spade] [33;1mDid not send 'unavailable' presence: 'Red' object
[0ma@127.0.0.1:[spade] [32mStopping agent a@127.0.0.1 , ok[0m
Traceback (most recent call last):
  File "SuperAgente.py", line 65, in <module>
    a = Red("@"+host,"secret")
  File "C:\Users\D A Santana\Anaconda2\lib\site-packages\spade-2.3.3-py2.7.egg\s
    raise AgentNotRegisteredError
spade.Agent.AgentNotRegisteredError
PS C:\Users\D A Santana\Downloads> python SuperAgente.py
Fuzzy arrancada
La fuzzy recibio el mensaje:
<presence xmlns="jabber:client" from="a@127.0.0.1/spade"><priority>0</priority>
y" stamp="20180529T14:15:54" /></presence>
LANZADOR DE LA RED ENCENDIDO
La red sera lanzada
Ingreso Caudal Entrada: 106.95
Ingreso caudal de la chaqueta: 82.346
Se enviaron las cosas al fuzzy:
[106.95, 82.346]
La fuzzy recibio el mensaje:
[106.95, 82.346]
```

Fig9: Aplicación de la técnica Multiagentes.

# Evidencia

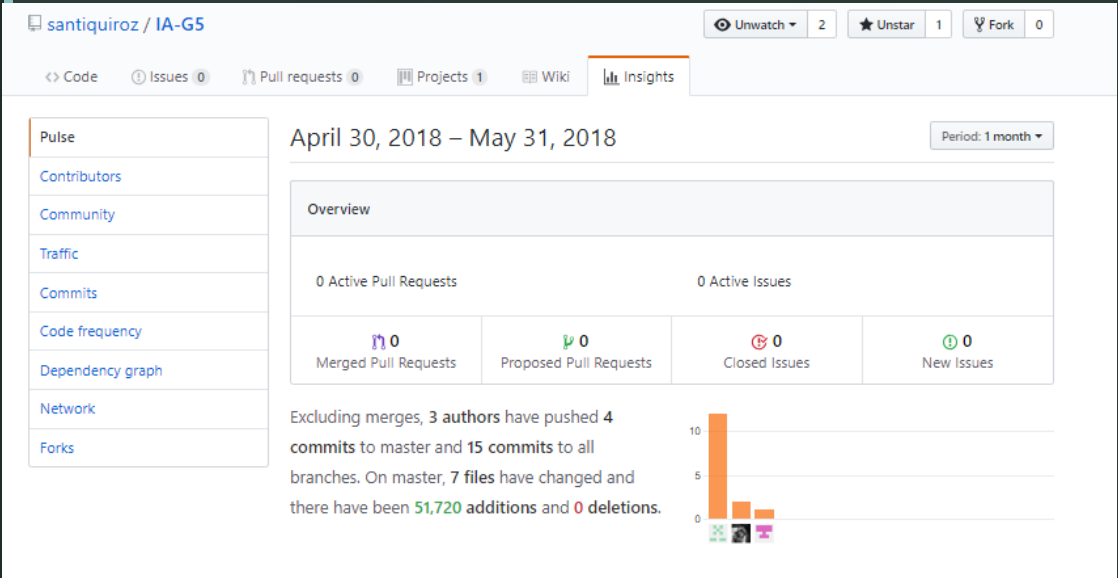
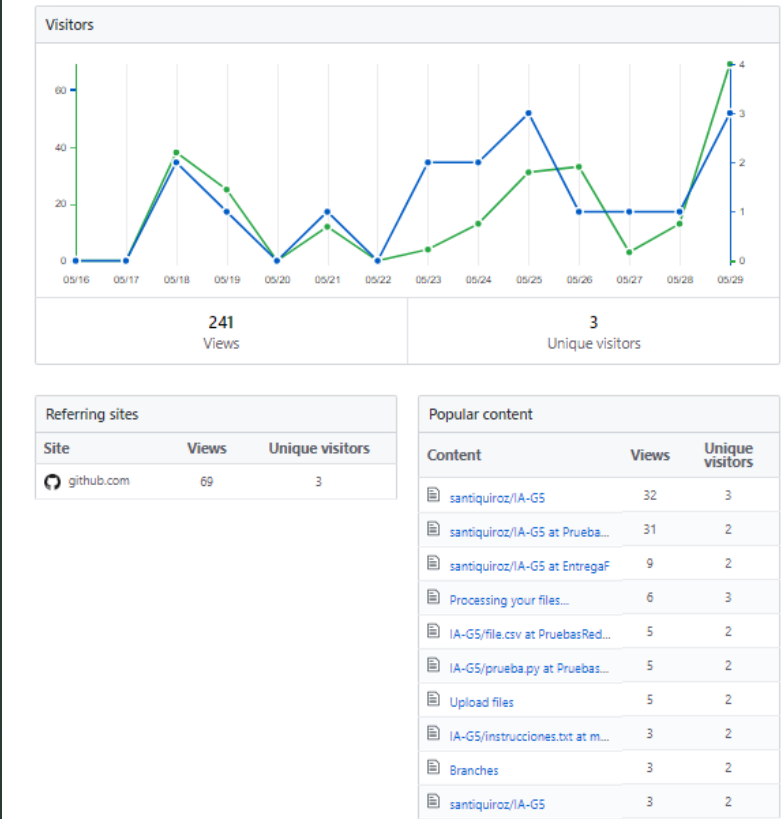


Fig10: Actividad en Github





## Resultados

- Comunicación entre los agentes involucrados en el sistema.
- Evaluación y Pre-clasificación por parte de la red neuronal.
- Clasificación y acciones de control por medio de lógica difusa.

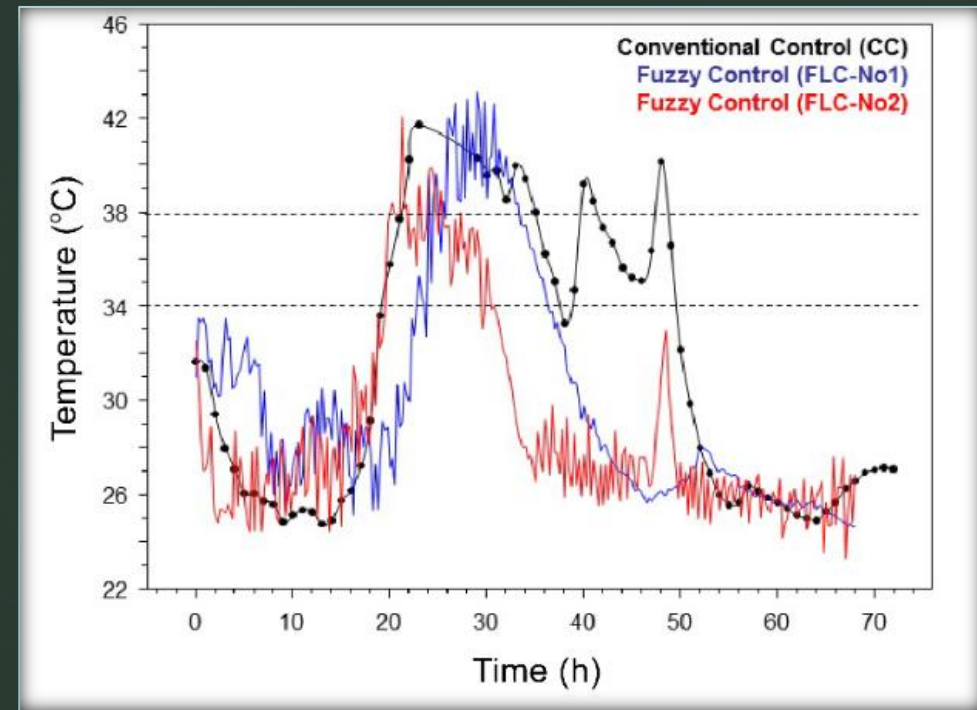


Fig11: Caso de aplicación de control fuzzy con diferentes reglas.

# Conclusiones

- El control MIMO es mas fácil de implementar utilizando técnicas de IA
- Si fue posible lograr los objetivos planteados con las herramientas vistas en clase, aunque no son los mas adecuados para el problema.
- Se debe ser asertivo o asesorarse por un experto para asignar las funciones implementadas y así lograr el objetivo.
- Las técnicas aplicados no fueron los óptimos para objetivo alcanzado.

# Fuentes

- ❑ Control and Monitoring of Chemical Batch Reactors. ISBN 978-0-85729-195-0
- ❑ <https://github.com/santiquiroz/IA-G5/tree/PruebasRedNeuronal/Documentos/Referencias>



■ Gracias.