

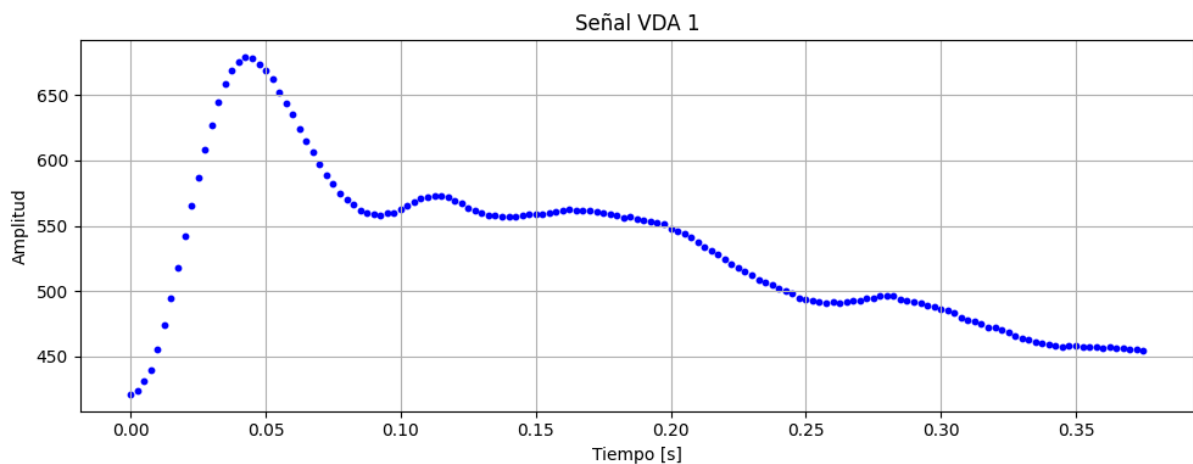
Actividad Sugeno 1a Variación de Diámetro Arterial (VDA)

1. Introducción

El presente trabajo tiene como objetivo principal la aplicación y evaluación de los Modelos Difusos de **Sugeno** para la aproximación de funciones en la señal de **Variación de Diámetro Arterial (VDA)**. Dada su dinámica compleja y naturaleza no lineal, la VDA representa un caso de estudio ideal para la modelización mediante técnicas de inferencia difusa.

Un modelo de Sugeno es un sistema de inferencia difuso que utiliza conjuntos difusos en el antecedente y funciones lineales o constantes en el consecuente, siendo particularmente robusto y adecuado para la **aproximación universal de funciones**.

En este apartado, se procedió a cargar y analizar la señal de VDA, que fue muestreada a una frecuencia de 400 Hz, estableciendo un eje temporal con intervalos de 2.5 ms entre muestras. A continuación, se **entrenaron múltiples modelos de Sugeno**, variando la cantidad de reglas R [**Mencionar si se usó un algoritmo específico para la generación de reglas, como el *clustering* sustractivo**] para evaluar el *trade-off* entre complejidad y precisión, medida por el Error Cuadrático Medio (MSE). Finalmente, el modelo seleccionado se utilizó para **sobremuestrear** la señal, demostrando su capacidad de generalización y predicción de valores intermedios.



2. Metodología

Esta sección describe el procedimiento detallado para el entrenamiento, evaluación y selección de un modelo de Sugeno para aproximar la señal de Variación de Diámetro Arterial (VDA).

Preparación de Datos

1. **Carga de la Señal:** Se cargó el conjunto de datos de la señal de VDA desde el archivo "samplesVDA1.txt".
2. **Generación del Eje Temporal:**
 - La señal fue muestreada a una frecuencia (F_s) de 400 Hz.
 - El intervalo de tiempo (Δt) entre muestras se calculó como $\Delta t = 1/F_s = 1/400$ s = 2.5 ms.
 - Se generó un vector temporal (t) para el eje de entrada, que representa el tiempo en segundos.
3. **Definición de Entradas y Salidas (E/S):** El modelo de Sugeno se entrenó para mapear el tiempo al valor de la señal.
 - **Variable de Entrada (x):** El vector temporal t .
 - **Variable de Salida (y):** El valor de la señal de VDA en ese instante.

```
# Cargar el archivo 'samplesVDA1.txt'
archivo = 'samplesVDA1.txt'
vda_signal = np.loadtxt(archivo)

# Frecuencia de muestreo y eje temporal (en segundos)
fs = 400 # Hz
Ts = 1 / fs # intervalo de muestreo: 2.5 ms
tiempo = np.arange(len(vda_signal)) * Ts # eje x en segundos

# Graficar la señal
plt.figure(figsize=(10, 4))
plt.scatter(tiempo, vda_signal, color='blue', s=10) # s es el tamaño de los puntos
```

Entrenamiento y Evaluación

1. **Algoritmo de Entrenamiento:** Se utilizó el algoritmo de Clustering Sustractivo para generar las reglas del Modelo de Sugeno de Primer Orden.
2. **Variación del Parámetro Clave:** Se entrenó una serie de modelos de Sugeno de primer orden, variando el parámetro de radio de vecindad (Radii). Cada radii determinó automáticamente la cantidad de clusters y en consecuencia la cantidad de reglas (R).
3. **Evaluación:** Para cada modelo entrenado, se calculó el Error Cuadrático Medio (MSE) entre los valores predichos por el modelo y los valores reales del conjunto de datos.

```

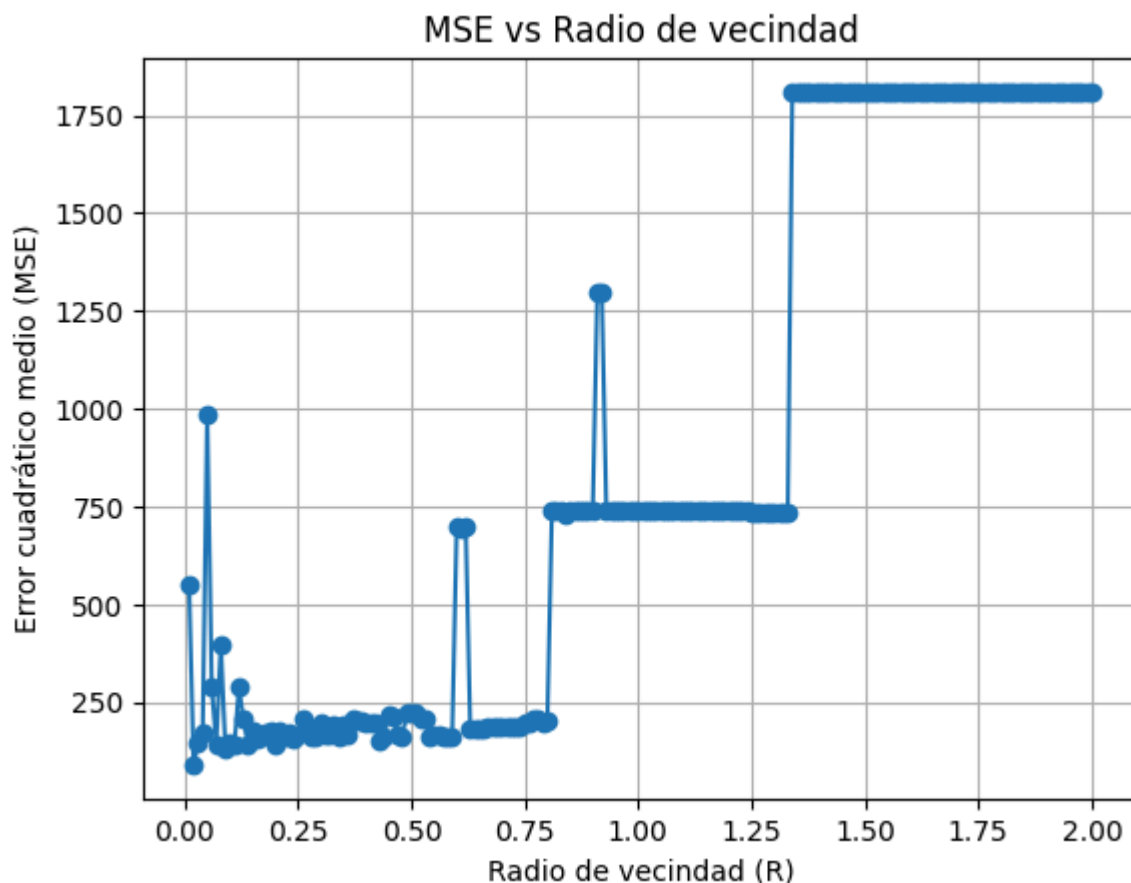
radii_values = np.linspace(0.01, 2.0, 200)
mse_values = []

for r in radii_values:
    modelo = fis(muestra=False)
    modelo.genfis(data, radii=r)
    y_pred = modelo.evalfis(data[:, 0].reshape(-1,1))
    y_true = data[:, 1]
    mse = np.mean((y_true - y_pred)**2)
    mse_values.append(mse)
    print("MSE=",mse," CantCluster=", len(modelo.rules)," radii=",r)

plt.figure()
plt.plot(radii_values, mse_values, marker='o')
plt.xlabel('Radio de vecindad (R)')
plt.ylabel('Error cuadrático medio (MSE)')
plt.title('MSE vs Radio de vecindad')
plt.grid(True)
plt.show()

```

4. **Graficación:** Se generó la gráfica que relaciona el MSE (eje y) versus la complejidad R (eje x), permitiendo visualizar el compromiso entre ambos.

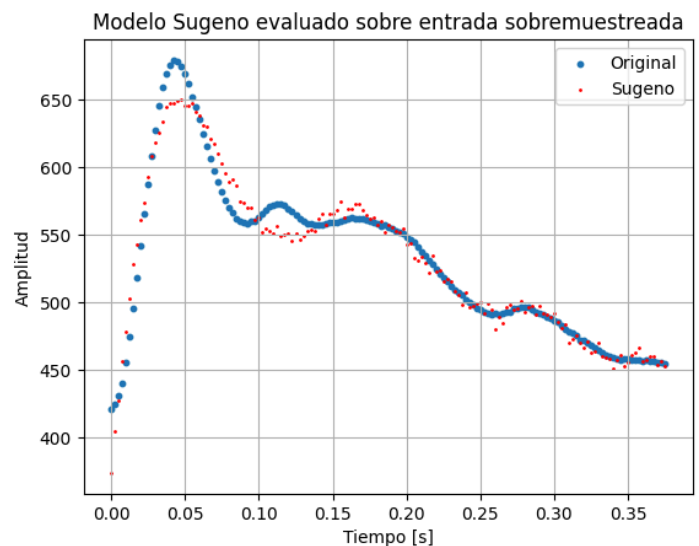
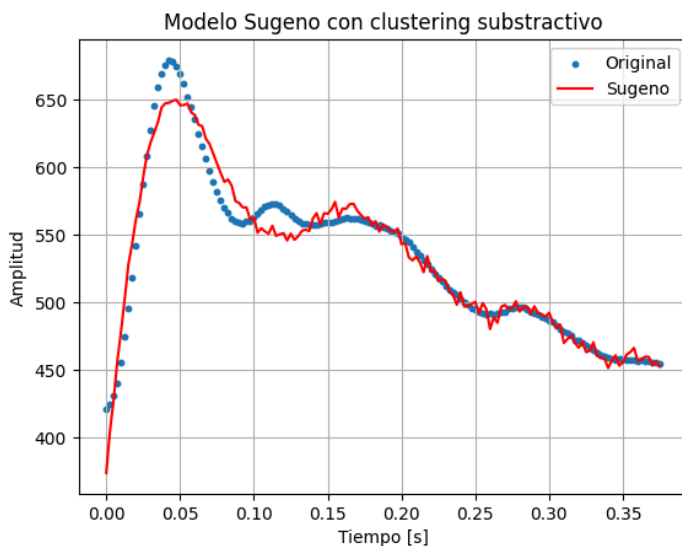


Selección del Modelo

- El modelo óptimo se seleccionó basándose en el análisis del gráfico MSE vs. R (Reglas), evaluando el rendimiento de múltiples modelos generados al barrer el parámetro de radio (Ra) del *Clustering* Sustractivo. El criterio de selección adoptado fue la búsqueda de la mejor relación entre la simplicidad (bajo R) y la robustez del modelo, priorizando la evitación del sobreajuste (o *overfitting*) sobre la consecución del mínimo MSE absoluto.
- De todos los resultados se seleccionaron 4 candidatos:

Radii (Ra)	Cluster (R)	MSE	Comentario
0.09	43	133.00	Mínimo MSE, pero alta complejidad
0.20	21	139.51	Compromiso, aunque todavía complejo
0.43	9	151.97	Elegido: Óptimo balance entre simplicidad y precisión
0.63	5	182.26	Demasiada pérdida de precisión

- Se eligió el Candidato 3, con R=9 reglas** (obtenidas con Ra=0.43). Aunque los candidatos 1 y 2 arrojaron valores de MSE ligeramente inferiores, un número tan elevado de *clusters* (43 y 21) se interpretó como un riesgo potencial de sobreajuste para la naturaleza de los datos. El modelo de R=9 representa un punto donde el incremento en el MSE (pasando de 139.51 a 151.97) es aceptable a cambio de una reducción drástica en la complejidad (21 a 9 reglas). Esta elección asegura un modelo más simple, interpretable y con mayor potencial de generalización a datos no vistos.



Sobremuestreo

Para validar la capacidad de **generalización e interpolación** del modelo Sugeno (R=9), se realizó un sobremuestreo de la señal.

1. **Generación de Datos:** Se creó un vector de entrada temporal t' que abarcó el rango de datos original, pero con una **resolución mucho mayor** (1000 puntos).
2. **Evaluación:** El modelo se evaluó en estos 1000 nuevos puntos (`modelo.evalfis`) para generar una **aproximación densificada** (salida_densa).
3. **Resultado:** La visualización de la predicción densificada superpuesta a los datos originales demostró que el modelo es capaz de generar una **función continua y suave** que describe fielmente la dinámica de la señal VDA.

