

## **Actividad Sugeno 1b Valores Accionarios del S&P 500 (SPY)**

### **1. Introducción**

El presente trabajo tiene como objetivo principal la aplicación y evaluación de los Modelos Difusos de Sugeno para la aproximación y predicción de series temporales financieras, específicamente los valores de las acciones del índice S&P 500 (SPY). Dada la naturaleza volátil y estocástica de las series de precios, este conjunto de datos representa un desafío único para la modelización mediante técnicas de inferencia difusa, buscando capturar tendencias a pesar del ruido inherente al mercado.

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 una herramienta robusta para la aproximación de funciones no lineales y la modelización de series temporales.

En este apartado, se procedió a cargar y analizar el conjunto de datos del S&P 500 utilizando un subconjunto de diez años, específicamente el periodo comprendido entre enero de 2013 y enero de 2023. A continuación, se entrenaron múltiples Modelos Sugeno de Primer Orden, variando la cantidad de reglas  $R$ . Esta variación se logró barriendo el parámetro de radio del Clustering Sustractivo, el algoritmo utilizado para generar las reglas y sus centros. Esto permitió evaluar el *trade-off* entre complejidad y precisión, medida por el Error Cuadrático Medio (MSE). Finalmente, el modelo seleccionado se utilizó para la extrapolación de los precios futuros de las acciones. La tarea de sobremuestreo se omitió, dado que el subconjunto de diez años ya contenía una alta densidad de datos 2518 días, lo que hacía que el sobremuestreo no aportara un valor adicional significativo a la visualización de la aproximación de la función. El enfoque se centró en la capacidad predictiva a corto plazo del modelo.

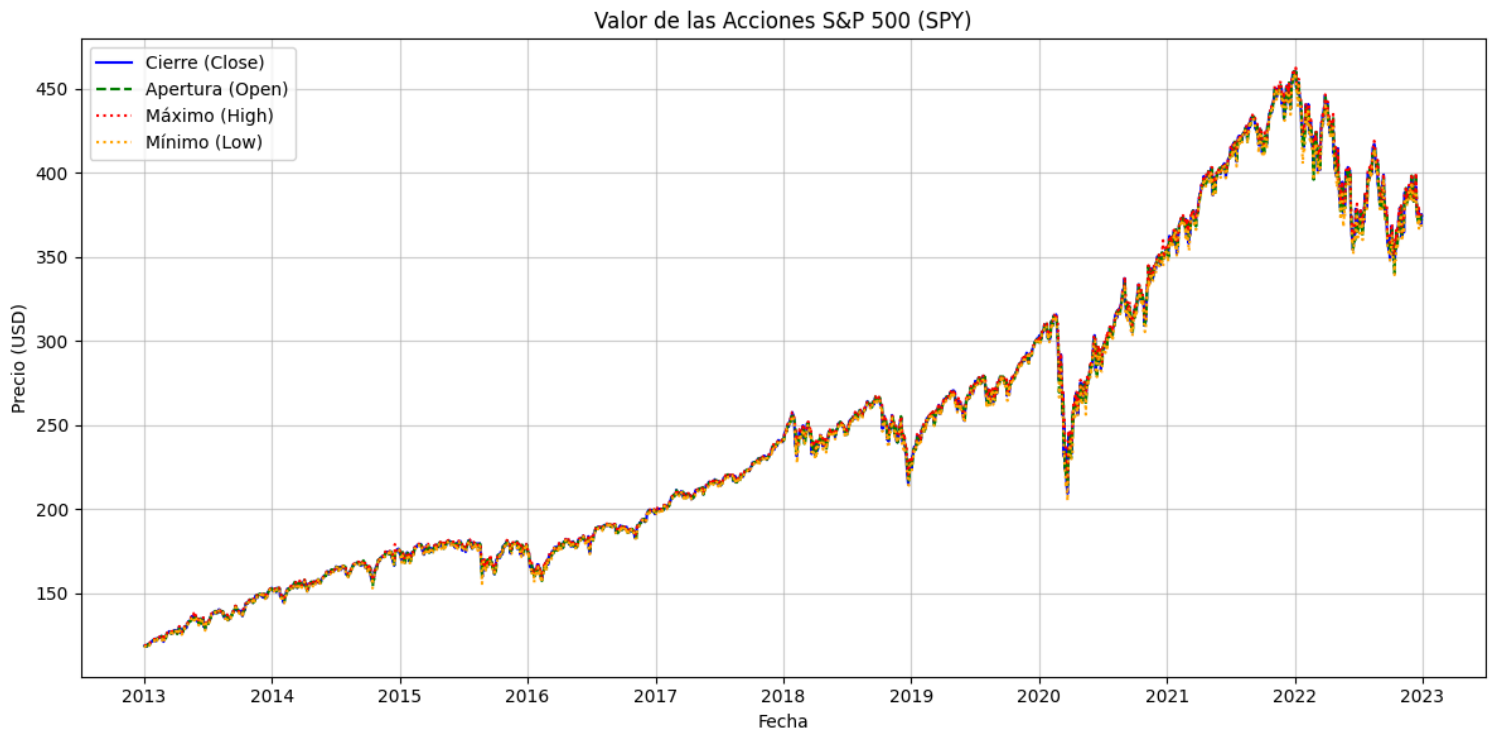
### **2. Metodología**

Esta sección describe el procedimiento detallado para el entrenamiento, evaluación y selección de un modelo de Sugeno para la aproximación y extrapolación de la serie de precios del índice S&P 500 (SPY).

#### **Preparación de Datos**

1. Carga y Selección del Intervalo: Se cargó el conjunto de datos de los precios del SPY a partir del archivo "spy.csv". Para el análisis, se seleccionó un subconjunto de diez años, específicamente el periodo comprendido entre enero de 2013 y enero de 2023.
2. Visualización: Se graficó la serie temporal, utilizando el Precio de Cierre (*Close*) del día como variable principal de interés.

3. Definición de Entradas y Salidas (E/S): El objetivo inicial fue la aproximación de función sobre el periodo histórico, modelando el precio en función del tiempo de ocurrencia.
- Variable de Entrada (x): El índice numérico del tiempo (o el número de día/muestra) dentro del subconjunto seleccionado.
  - Variable de Salida (y): El precio de cierre correspondiente a ese día, Close.



### Entrenamiento y Evaluación

1. **Algoritmo de Entrenamiento:** Se utilizó un **Modelo Sugeno de Primer Orden**. La cantidad de reglas (R) y la ubicación de los conjuntos difusos se determinaron mediante el algoritmo de **Clustering Sustractivo** (*Subtractive Clustering*).
2. **Variación del Parámetro Clave:** Se entrenó una serie de modelos variando el **parámetro de radio de vecindad (Ra)**. Cada valor de Ra generó un número distinto de reglas R.
3. **Evaluación:** Para cada modelo entrenado, se calculó el **Error Cuadrático Medio (MSE)** entre la aproximación del modelo y los valores reales del SPY en el periodo histórico. Se generó la gráfica que relaciona el MSE vs. R.

```

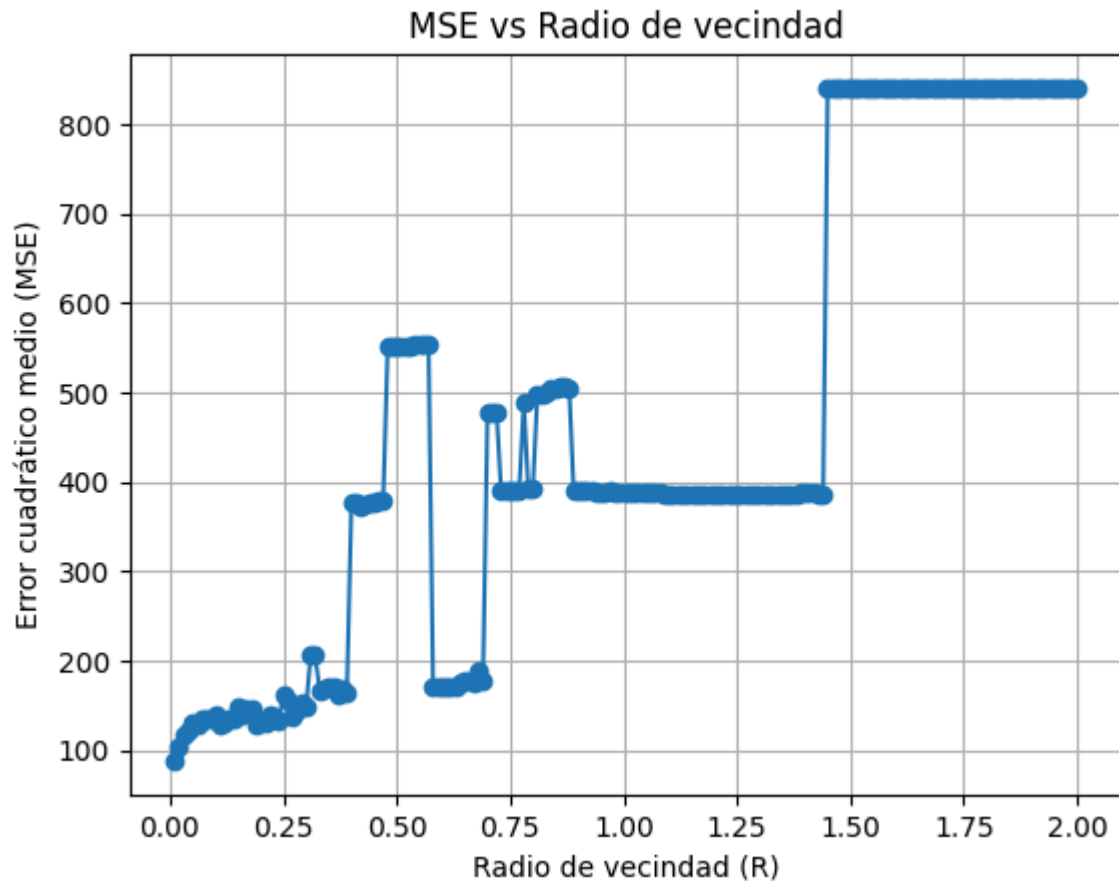
radii_values = np.linspace(0.01, 2.0, 200)
mse_values = []

for r in radii_values:
    modelo_spy = fis(muestra=False)
    modelo_spy.genfis(data_spy, radii=r)
    y_pred = modelo_spy.evalfis(data_spy[:, 0].reshape(-1,1))
    y_true = data_spy[:, 1]
    mse = np.mean((y_true - y_pred)**2)
    mse_values.append(mse)
    print("MSE=",mse," CantCluster=", len(modelo_spy.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 a partir del análisis de la curva MSE vs. R. Inicialmente, se entrenaron aproximadamente **200 modelos** al barrer continuamente el parámetro de radio (Ra) del *Clustering* Sustractivo. El análisis de esta población total reveló que el MSE caía drásticamente con un alto número de reglas ( $R > 100$ ), pero el *trade-off* de complejidad resultaba inaceptable.

Para refinar la elección, se graficaron y compararon específicamente **cuatro candidatos** que representaban diferentes balances entre complejidad y error, como se detalla en la siguiente tabla:

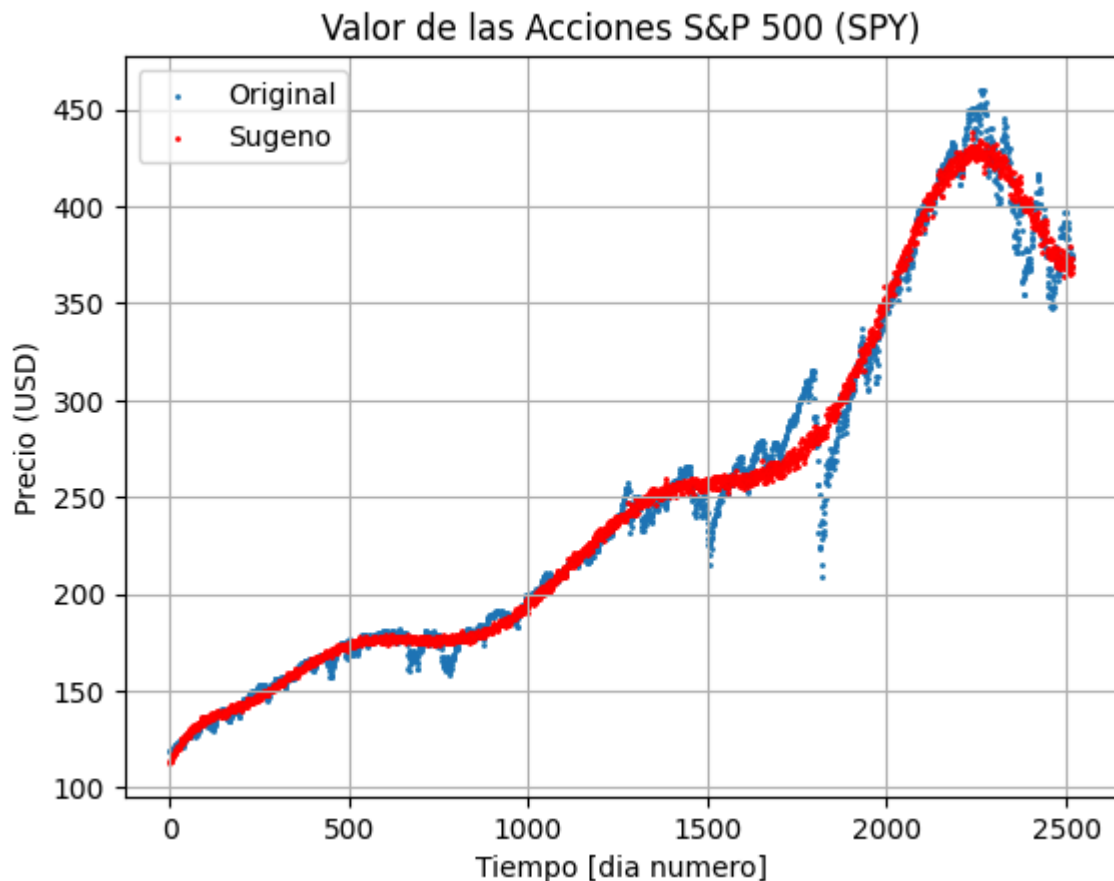
Radii(Ra)	Clusters(R)	MSE	Comentario
0.11	33	129.53	R muy alto pero buen MSE
0.19	17	129.35	Óptimo balance entre R y MSE
0.09	46	135.10	Mayor R con pero MSE
<b>0.37</b>	<b>7</b>	<b>162.48</b>	<b>Elegido: Muy simple 7 reglas</b>

El candidato 2 ofrece el mismo error que los candidatos 1 y 3 y resulta mucho más simple, únicamente 17 reglas contra 33 o 46.

Luego el candidato 4 comparado con el candidato 2 incrementa el MSE un 25,61% pero simplifica el modelo de 17 a 7 reglas, lo que representa un 41,18% menor cantidad de reglas.

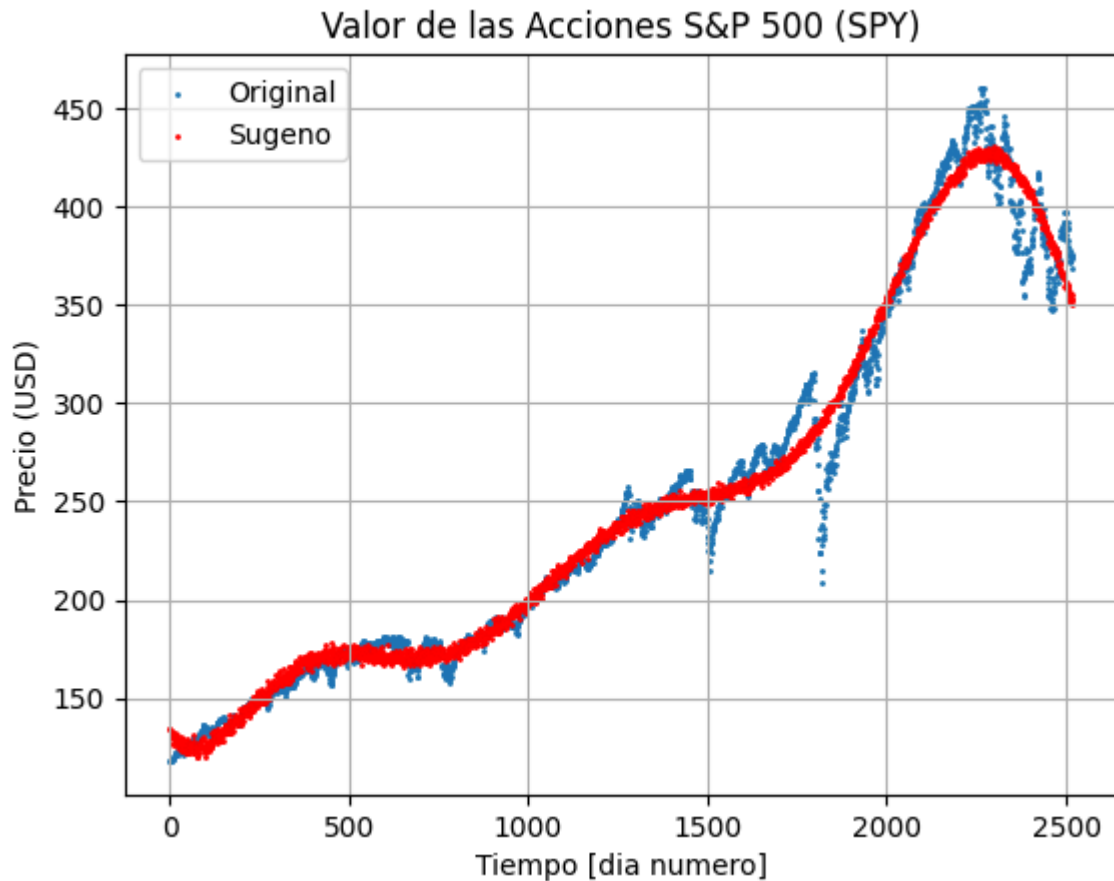
Luego procedí a graficar los modelos 2 y 4 para compararlos gráficamente. Pero no se puede apreciar mucha diferencia entre ambas curvas.

Modelo 2, radii 0.19, reglas 17, mse 129.35



### **Modelo elegido**

Modelo 4, radii 0.37, reglas 7, mse 162.48



### Sobremuestreo

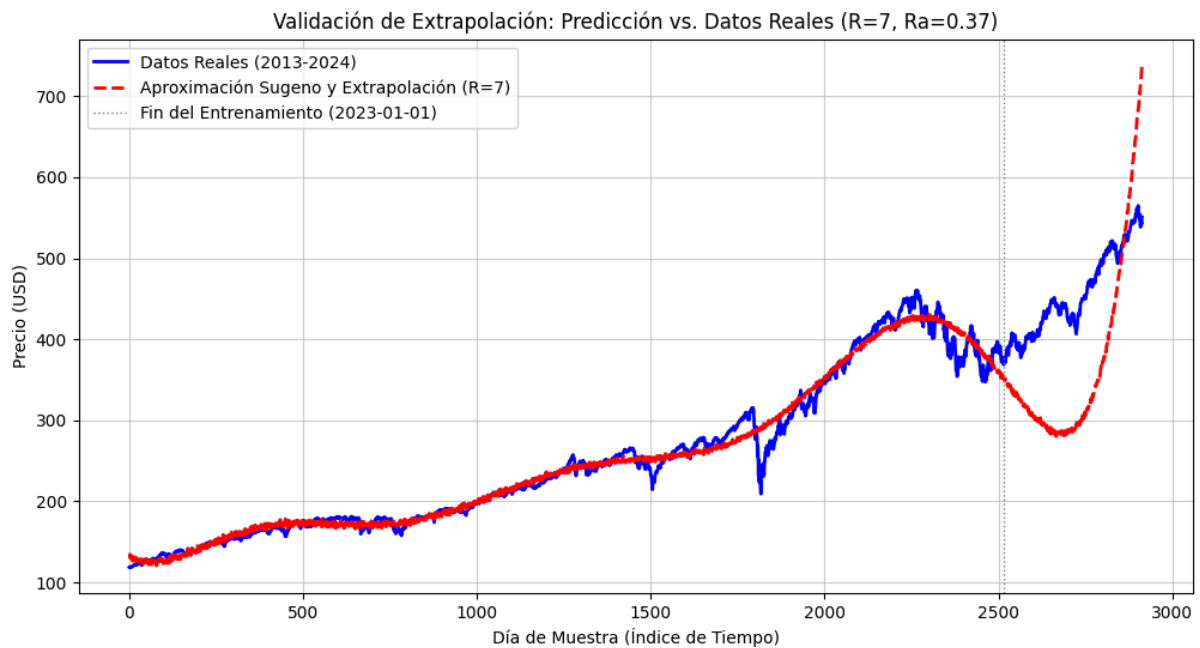
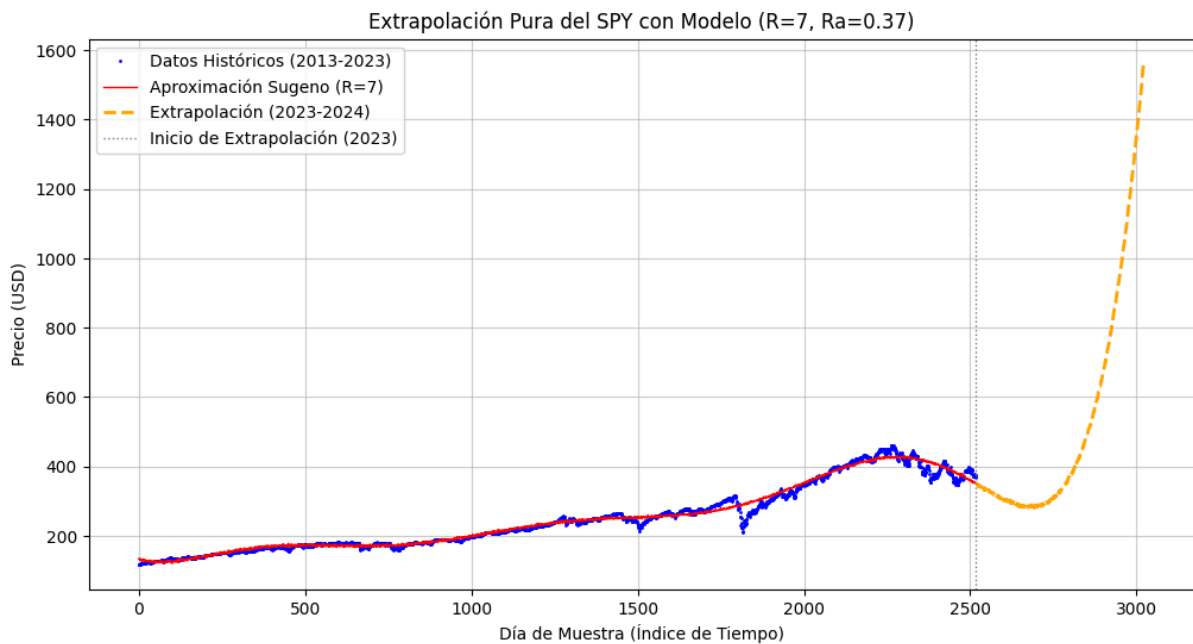
Se omitió el sobremuestreo debido a la **alta densidad** de la serie temporal utilizada, que ya incluye aproximadamente **2500 datos** correspondientes al periodo histórico de diez años. El sobremuestreo busca típicamente hacer visible la función continua subyacente en *datasets* escasos; sin embargo, en este caso, la densidad de los datos originales es suficiente para la visualización.

### Metodología de Extrapolación

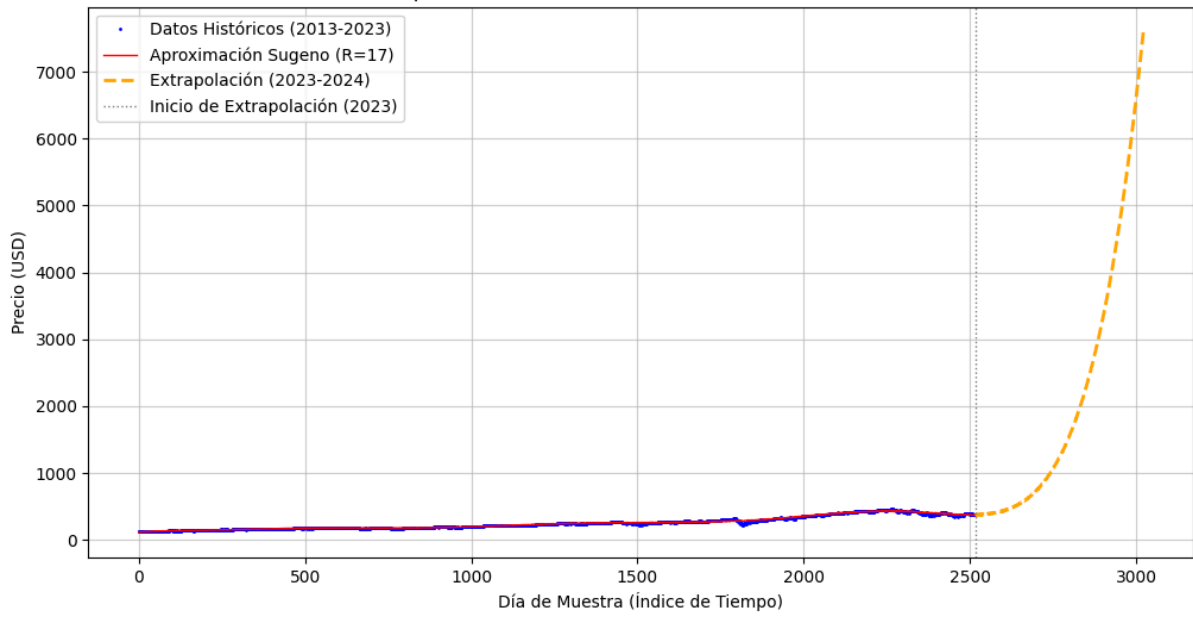
Dado que el modelo fue entrenado para mapear la función Tiempo→Precio, la extrapolación se realizó extendiendo el eje de entrada de la variable tiempo más allá del último día de entrenamiento (Día ≈2517, 1 de enero de 2023).

1. **Generación del Eje Futuro:** Se extendió el eje temporal por 504 días de mercado adicionales, cubriendo aproximadamente los años 2023 y 2024.
2. **Predicción:** Los modelos de Sugeno seleccionados se evaluaron sobre este nuevo rango temporal extendido.
3. **Validación Visual:** Se generó un segundo gráfico comparando la predicción del modelo con los datos reales del S&P 500 (SPY) disponibles hasta finales de 2024. Este contraste permitió validar la robustez de la función extrapolada frente a la volatilidad del mercado real.

De los 4 modelos mencionados arriba solo se hicieron predicciones para el modelo 2 (que tiene 17 clusters), modelo 3(que tiene 46 clusters) y modelo 4(que tiene 7 clusters).



Extrapolación Pura del SPY con Modelo ( $R=17$ ,  $R_a=0.19$ )



Validación de Extrapolación: Predicción vs. Datos Reales ( $R=17$ ,  $R_a=0.19$ )

