

▼ A7-Introducción a series de tiempo

Jorge Eduardo de León Reyna - A00829759

▼ Problema 1

```

1 import pandas as pd
2 import numpy as np
3
4 ventas_gasolina = [17, 21, 19, 23, 18, 16, 20, 18, 22, 20, 15, 22]
5
6 ventas = pd.Series(ventas_gasolina)
7
8 promedios_moviles = ventas.rolling(window=4).mean()
9
10 errores = ventas[3:] - promedios_moviles[3:]
11 cme_promedios_moviles = (errores**2).mean()
12
13 pronostico_semana_13_promedios_moviles = promedios_moviles.iloc[-1]
14
15 promedios_moviles, cme_promedios_moviles, pronostico_semana_13_promedios_moviles

(0      NaN
 1      NaN
 2      NaN
 3     20.00
 4     20.25
 5     19.00
 6     19.25
 7     18.00
 8     19.00
 9     20.00
10     18.75
11     19.75
dtype: float64,
5.75,
19.75)

1 media_ventas = ventas.mean()
2
3 desviacion_absoluta = abs(ventas - media_ventas)
4
5 pesos = desviacion_absoluta / desviacion_absoluta.sum()
6
7 promedios_moviles_ponderados = (ventas * pesos).rolling(window=4).sum()
8
9 errores_ponderados = ventas[3:] - promedios_moviles_ponderados[3:]
10 cme_promedios_moviles_ponderados = (errores_ponderados**2).mean()
11
12 pronostico_semana_13_promedios_moviles_ponderados = promedios_moviles_ponderados.iloc[-1]
13 promedios_moviles_ponderados, cme_promedios_moviles_ponderados, pronostico_semana_13_promedios_moviles_ponderados

(0      NaN
 1      NaN
 2      NaN
 3     6.64
 4     6.01
 5     6.62
 6     7.03
 7     4.48
 8     6.00
 9     4.52
10     6.47
11     7.99
dtype: float64,
179.45297777777776,
7.99)

1 from sklearn.metrics import mean_squared_error
2 def suavizamiento_exponencial(serie, alpha):
3     resultado = [serie[0]]
4     for n in range(1, len(serie)):
5         resultado.append(alpha * serie[n] + (1 - alpha) * resultado[n-1])

```

```

6     return pd.Series(resultado)
7 alphas = np.linspace(0, 1, 11)
8 cme_valores = []
9 for alpha in alphas:
10     pronosticos = suavizamiento_exponencial(ventas, alpha)
11     cme = mean_squared_error(ventas, pronosticos)
12     cme_valores.append(cme)
13 mejor_alpha_index = np.argmin(cme_valores)
14 mejor_alpha = alphas[mejor_alpha_index]
15 mejor_cme = cme_valores[mejor_alpha_index]
16 pronostico_semana_13_exponencial = suavizamiento_exponencial(ventas,mejor_alpha).iloc[-1]
17 mejor_alpha, mejor_cme, pronostico_semana_13_exponencial

(1.0, 0.0, 22.0)

1 cme_comparacion = {
2     "Promedios Móviles": cme_promedios_moviles,
3     "Promedios Móviles Ponderados": cme_promedios_moviles_ponderados,
4     "Suavizamiento Exponencial": mejor_cme
5 }
6 mejor_metodo = min(cme_comparacion, key=cme_comparacion.get)
7 cme_comparacion, mejor_metodo

({'Promedios Móviles': 5.75,
  'Promedios Móviles Ponderados': 179.45297777777776,
  'Suavizamiento Exponencial': 0.0},
 'Suavizamiento Exponencial')
```

La comparación de los Errores Cuadráticos Medios (CME) entre los tres métodos se presenta de la siguiente manera:

- Promedios Móviles: CME = 5.75
- Promedios Móviles Ponderados: CME = 179.45
- Suavizamiento Exponencial: CME = 0.0 El método de Suavizamiento Exponencial con un valor de $\alpha = 1.0$ ha arrojado el CME más bajo, lo que sugiere que es el modelo más eficaz entre los tres métodos evaluados para esta serie de tiempo en particular. No obstante, es fundamental considerar que un α igual a 1 implica que el modelo otorga un peso completo a la observación más reciente, lo que podría no ser óptimo si la serie de tiempo presenta variabilidad o ruido.

▼ Problema 2

```

1 from datetime import datetime, timedelta
2 fechas = pd.to_datetime(["2022-08-24", "2022-08-25", "2022-08-26", "2022-08-29", "2022-08-30", "2022-08-31", "2022-09-01", "2022-09-02", "2022-09-03", "2022-09-04", "2022-09-05", "2022-09-06", "2022-09-07", "2022-09-08", "2022-09-09", "2022-09-10", "2022-09-11", "2022-09-12", "2022-09-13", "2022-09-14", "2022-09-15", "2022-09-16", "2022-09-17", "2022-09-18", "2022-09-19", "2022-09-20", "2022-09-21", "2022-09-22", "2022-09-23", "2022-09-24", "2022-09-25", "2022-09-26", "2022-09-27", "2022-09-28", "2022-09-29", "2022-09-30"])
3 precios = [81.32, 81.10, 80.38, 81.34, 80.54, 80.62, 79.54, 79.46, 81.02, 80.98, 80.80, 81.44, 81.48, 80.75, 80.48, 80.01, 80.3, 80.5, 80.6, 80.7, 80.8, 80.9, 81.0, 81.1, 81.2, 81.3, 81.4, 81.5, 81.6, 81.7, 81.8, 81.9, 82.0, 82.1, 82.2, 82.3, 82.4, 82.5, 82.6, 82.7, 82.8, 82.9, 83.0, 83.1, 83.2, 83.3, 83.4, 83.5, 83.6, 83.7, 83.8, 83.9, 84.0, 84.1, 84.2, 84.3, 84.4, 84.5, 84.6, 84.7, 84.8, 84.9, 85.0, 85.1, 85.2, 85.3, 85.4, 85.5, 85.6, 85.7, 85.8, 85.9, 86.0, 86.1, 86.2, 86.3, 86.4, 86.5, 86.6, 86.7, 86.8, 86.9, 87.0, 87.1, 87.2, 87.3, 87.4, 87.5, 87.6, 87.7, 87.8, 87.9, 88.0, 88.1, 88.2, 88.3, 88.4, 88.5, 88.6, 88.7, 88.8, 88.9, 89.0, 89.1, 89.2, 89.3, 89.4, 89.5, 89.6, 89.7, 89.8, 89.9, 90.0, 90.1, 90.2, 90.3, 90.4, 90.5, 90.6, 90.7, 90.8, 90.9, 91.0, 91.1, 91.2, 91.3, 91.4, 91.5, 91.6, 91.7, 91.8, 91.9, 92.0, 92.1, 92.2, 92.3, 92.4, 92.5, 92.6, 92.7, 92.8, 92.9, 93.0, 93.1, 93.2, 93.3, 93.4, 93.5, 93.6, 93.7, 93.8, 93.9, 94.0, 94.1, 94.2, 94.3, 94.4, 94.5, 94.6, 94.7, 94.8, 94.9, 95.0, 95.1, 95.2, 95.3, 95.4, 95.5, 95.6, 95.7, 95.8, 95.9, 96.0, 96.1, 96.2, 96.3, 96.4, 96.5, 96.6, 96.7, 96.8, 96.9, 97.0, 97.1, 97.2, 97.3, 97.4, 97.5, 97.6, 97.7, 97.8, 97.9, 98.0, 98.1, 98.2, 98.3, 98.4, 98.5, 98.6, 98.7, 98.8, 98.9, 99.0, 99.1, 99.2, 99.3, 99.4, 99.5, 99.6, 99.7, 99.8, 99.9, 100.0]
4 acciones = pd.DataFrame(data={'Dia': fechas, 'Precio': precios})
5 acciones
```

```

    Dia Precio
0 2022-08-24 81.32
1 acciones['Promedio Móvil 3 Días'] = acciones['Precio'].rolling(window=3).mean()
2 pronostico_pm = acciones['Promedio Móvil 3 Días'].iloc[-1]
3 alpha = 0.6
4 acciones['Suavizado Exponencial'] = acciones['Precio'].ewm(alpha=alpha,adjust=False).mean()
5 pronostico_se = acciones['Suavizado Exponencial'].iloc[-1]
6 acciones, pronostico_pm, pronostico_se

```

	Dia	Precio	Promedio Móvil 3 Días	Suavizado Exponencial
0	2022-08-24	81.32	NaN	81.320000
1	2022-08-25	81.10	NaN	81.188000
2	2022-08-26	80.38	80.933333	80.703200
3	2022-08-29	81.34	80.940000	81.085280
4	2022-08-30	80.54	80.753333	80.758112
5	2022-08-31	80.62	80.833333	80.675245
6	2022-09-01	79.54	80.233333	79.994098
7	2022-09-02	79.46	79.873333	79.673639
8	2022-09-06	81.02	80.006667	80.481456
9	2022-09-07	80.98	80.486667	80.780582
10	2022-09-08	80.80	80.933333	80.792233
11	2022-09-09	81.44	81.073333	81.180893
12	2022-09-12	81.48	81.240000	81.360357
13	2022-09-13	80.75	81.223333	80.994143
14	2022-09-14	80.48	80.903333	80.685657
15	2022-09-15	80.01	80.413333	80.280263
16	2022-09-16	80.33	80.273333	80.310105,
			80.27333333333334,	
			80.31010514598606)	

a) Promedio Móvil de Tres Días: En la tabla original, se ha incorporado la serie de tiempo suavizada junto con los promedios móviles de tres días en la columna "Promedio Móvil 3 Días". Al emplear el último promedio móvil calculado, se proyecta un precio de cierre para el 19 de septiembre cercano a 80.27.

b) Suavizamiento Exponencial con $\alpha=0.6$: En la tabla original, se ha integrado la serie de tiempo suavizada en la columna "Suavizado Exponencial". La estimación para el precio de cierre del 19 de septiembre, utilizando el valor suavizado más reciente, ronda los 80.31.

c) Comparación y Preferencia:

Ambos métodos han proporcionado pronósticos similares para el precio de cierre del 19 de septiembre, aunque existen diferencias en su enfoque de suavización temporal. El promedio móvil otorga igual ponderación a los últimos tres días, mientras que el suavizamiento exponencial asigna mayor peso a las observaciones más recientes. La elección entre estos dos métodos puede depender del contexto específico y de la importancia otorgada a los datos recientes en comparación con los datos más antiguos. Si se considera que los cambios recientes en el precio son más representativos de los futuros precios, se podría preferir el suavizamiento exponencial. Si se busca igualar la importancia de los últimos tres días, se podría optar por el promedio móvil.