

# Manual Técnico de Análisis de Señales

## Guía de Operación: Notebook EAFIT\_DataScience

Versión: 2.0 (Feb 2026) | Entorno: Python 3.12+

Docente: Jorge Iván Padilla-Buriticá  
Universidad EAFIT - Medellín, Colombia

**Resumen:** Este manual detalla el funcionamiento del notebook adjunto para el **Challenge 02**. El objetivo es que el estudiante manipule series de tiempo complejas, aplique filtros digitales y valide modelos predictivos bajo condiciones de ruido real (SNR).

## 1 Guía de Modificación y Juego con el Código

### 1.1 1. ¿Cómo aumentar o reducir el ruido (SNR)?

En el **Bloque 3**, localice la función `add_noise`. El parámetro clave es `snr_db`.

- **Más Ruido:** Reduzca el valor (ej. `snr_db=2`). La señal será casi imperceptible.
- **Menos Ruido:** Aumente el valor (ej. `snr_db=30`). La señal será casi idéntica a la limpia.

### 1.2 2. ¿Cómo cambiar y ajustar los filtros?

El notebook ofrece dos estrategias en el **Bloque 4**:

- **Media Móvil (Moving Average):** Cambie el valor de `window=20`. Una ventana más grande suaviza más, pero introduce un "lag" (retraso) mayor.
- **Filtro Butterworth:** Ajuste el parámetro `cutoff` (frecuencia de corte).
  - *Valor bajo (0.01):* Solo deja pasar tendencias lentas.
  - *Valor alto (0.4):* Deja pasar más ruido.

### 1.3 3. ¿Cómo cambiar el modelo ARIMA?

En el **Bloque 5**, la función `ARIMA(series, order=(p, d, q))` define la estructura:

- **p:** Rezagos de la serie (Autorregresivo).
- **d:** Veces que diferencia la serie (para hacerla estacionaria).
- **q:** Rezagos del error (Media Móvil).

## 2 Paso a Paso: Guía de Análisis para Estudiantes

1. **Generación:** Ejecute el Bloque 1 para poblar el DataFrame `df_clean`. Use `df.head()` para verificar los datos.
2. **Inspección Geo-Temporal:** Use los mapas de Plotly para identificar si el origen del dato (Latitude/Longitude) influye en la señal.

3. **Test de Estacionariedad:** Aplique `adfuller()`. **Regla de Oro:** Si el p-value es  $> 0,05$ , usted NO puede aplicar correlaciones de Pearson ni modelos ARMA simples; debe diferenciar la serie.
4. **Análisis de Frecuencia:** Observe el espectrograma. Identifique la "huella digital" de la señal. Si hay una línea horizontal brillante, ha encontrado una periodicidad (ciclo).
5. **Validación de Residuos:** Tras ajustar su modelo, mire los residuos. Si el gráfico ACF de los residuos tiene barras que sobresalen, su modelo está dejando información "sobre la mesa".

### 3 Uso de Librerías Core

#### Fragmento de Código: Pandas - Manipulación de Datos

Use `df.iloc[fila, columna]` para acceso por posición y `df.loc['index', 'columna']` para acceso por nombre. Para series de tiempo, el ventaneo se logra con `df.rolling(window=X).mean()`.

#### Fragmento de Código: Plotly - Visualización Viva

La clave de la "Gráfica Viva" es el parámetro `frames`. Cada frame es un "snapshot" de la serie que Plotly reproduce secuencialmente. Para manipulación estática elegante, use el `rangeslider` en el layout.

### 4 Checklist de Validación Final

- ✓ ¿He comparado el RMSE del filtro Butterworth vs. el Media Móvil?
- ✓ ¿Mis intervalos de confianza de la predicción crecen con el tiempo (evidencia de incertidumbre)?
- ✓ ¿He documentado por qué elegí los parámetros  $(p, d, q)$  basándome en los gráficos de ACF/PACF?

*Este manual es una guía, su curiosidad es el verdadero motor del análisis."*

**EAFIT - 2026**