



FACULTAD DE INGENIERÍA

BIOINGENIERÍA

Bioseñales y Sistemas

Valentina Garcia Obando C.C.1000539432

Juan José Alzate Molina. C.C.1007232151

## Informe Proyecto 3: Filtros

### 1. Del artículo:

<https://www.nature.com/articles/s41598-020-59821-7>. Consultar y explicar los dos métodos de reducción de ruido usados en el artículo: Robust LOESS y Non Local Means (10%), mostrar cómo se podrían implementar en Python (5%)

**Robust LOESS (Locally Weighted Scatterplot Smoothing):** El método LOESS es una técnica de suavizado no paramétrica utilizada para eliminar fluctuaciones de baja frecuencia, como el desplazamiento de línea base en señales biomédicas (por ejemplo, ECG). Este método realiza un ajuste de regresión local ponderada que se adapta a las características específicas de los datos y es robusto frente a valores atípicos.

Su funcionamiento es:

**Ajuste local,** LOESS divide los datos en pequeñas regiones (vecindarios) y realiza un ajuste de regresión polinómica (normalmente lineal o cuadrática) en cada región.

**Pesos,** cada punto dentro del vecindario recibe un peso que decrece con la distancia al centro del vecindario. Estos pesos se calculan mediante una función kernel, como una campana gaussiana.

**Iteratividad,** el método ajusta iterativamente los pesos para minimizar la influencia de valores atípicos, aumentando la robustez del suavizado.

**Combinación,** finalmente, se combinan los ajustes locales para obtener una curva suavizada que sigue las tendencias principales de los datos.

Sus ventajas son: Adecuado para señales con tendencias de baja frecuencia, como la línea base del ECG, no requiere suposiciones sobre el modelo global de los datos, su robustez permite manejar valores atípicos sin afectar significativamente el resultado.

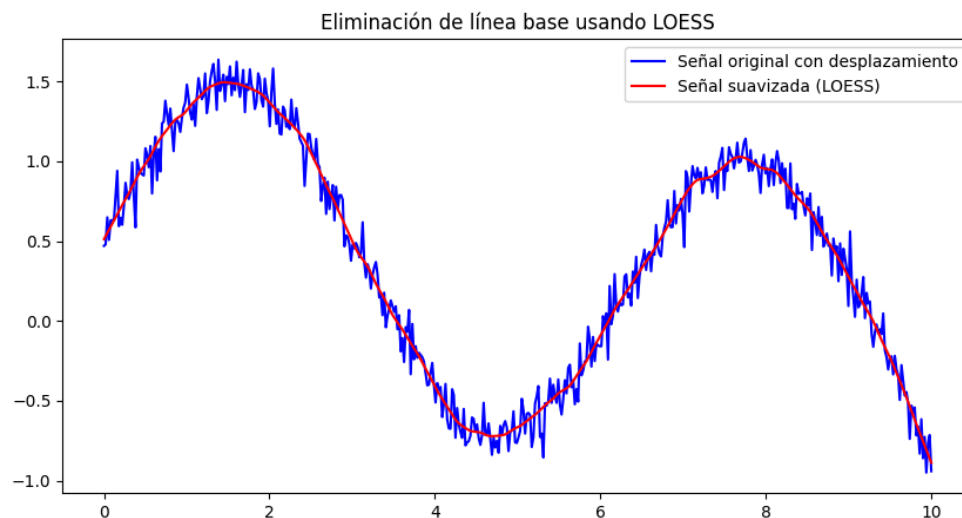
Se muestra cómo se puede aplicar LOESS a una señal con desplazamiento de línea base:

```
import numpy as np
from statsmodels.nonparametric.smoothers_lowess import lowess
import matplotlib.pyplot as plt

# Simulación de señal con desplazamiento de línea base
x = np.linspace(0, 10, 500)
y = np.sin(x) + 0.5 * np.cos(0.2 * x) + 0.1 * np.random.normal(size=500)

# Aplicación de LOESS
y_smooth = lowess(y, x, frac=0.05, it=3)[: , 1]

# Visualización de resultados
plt.figure(figsize=(10, 5))
plt.plot(x, y, label='Señal original con desplazamiento', color='blue')
plt.plot(x, y_smooth, label='Señal suavizada (LOESS)', color='red')
plt.title("Eliminación de línea base usando LOESS")
plt.legend()
plt.show()
```



**Non Local Means (NLM):** El método Non Local Means es una técnica avanzada de filtrado que se utiliza para eliminar ruido preservando los detalles importantes de una señal. Este método fue desarrollado inicialmente para procesar imágenes digitales, pero ha demostrado ser eficaz para señales unidimensionales, como las señales ECG.

Su funcionamiento es:

**Similitud**, para cada punto  $i$ , se compara su vecindario local con los vecindarios de otros puntos  $j$  en la señal.

**Pesos**, los pesos de cada punto  $j$  se calculan en función de la similitud entre sus vecindarios. Los puntos más similares tienen mayor influencia.

**Reconstrucción**, el valor en  $i$  se actualiza mediante un promedio ponderado de los valores de  $j$ , donde los pesos determinan la contribución de cada punto.

Sus ventajas son: Preserva detalles finos, como los picos y complejos QRS en señales ECG, es especialmente útil para señales contaminadas con ruido blanco gaussiano, no requiere supuestos previos sobre el tipo de ruido.

Se muestra cómo aplicar NLM para filtrar ruido en una señal, en este código, se ha utilizado `reshape(1, -1)` para convertir la señal unidimensional en una matriz 2D con una sola fila y luego la he convertido de nuevo a 1D usando `flatten()` después de la denoización.

```
from skimage.restoration import denoise_nl_means, estimate_sigma
from skimage.util import random_noise
import matplotlib.pyplot as plt
import numpy as np

# Creación de señal ruidosa
np.random.seed(42)
signal = np.sin(np.linspace(0, 4 * np.pi, 500))
noisy_signal = random_noise(signal, var=0.05**2)

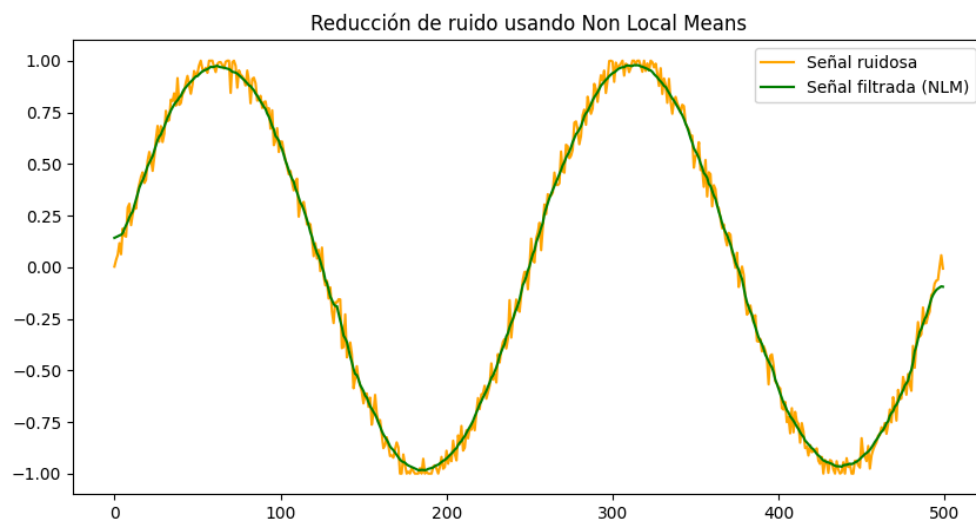
# Estimación de la desviación estándar del ruido
sigma_est = np.sqrt(estimate_sigma(noisy_signal))

# Convertir la señal 1D en 2D (una fila)
noisy_signal_2d = noisy_signal.reshape(1, -1)

# Filtrado con Non Local Means
denoised_signal_2d = denoise_nl_means(noisy_signal_2d, h=1.15 * sigma_est, fast_mode=True, patch_size=5, patch_distance=11)

# Convertir la señal filtrada de nuevo a 1D
denoised_signal = denoised_signal_2d.flatten()

# Visualización de resultados
plt.figure(figsize=(10, 5))
plt.plot(noisy_signal, label='Señal ruidosa', color='orange')
plt.plot(denoised_signal, label='Señal filtrada (NLM)', color='green')
plt.title("Reducción de ruido usando Non Local Means")
plt.legend()
plt.show()
```



2. Consultar qué otros tipos de señales wavelet se pueden usar para el análisis de señales ECG y adaptar el código del filtro wavelet que se entrega en el curso de acuerdo a la consulta (10%)

Según lo presentado por Fars Samann y Thomas Schanzela, en su análisis de diferentes filtros y parámetros para tratar una onda ECG, la transformada Wavelet se optimiza con la selección del tipo de onda madre y el nivel de descomposición. Y entre las diferentes opciones consideradas el mejor resultado se obtuvo con un filtro **sym4** de nivel 3.

La wavelet **sym4** es parte de la familia de wavelets simétricas, conocidas también como Symlets. Fue desarrollada por Ingrid Daubechies como una variante de la familia de wavelets Daubechies, con el propósito de reducir la asimetría que estas tenían. La sym4 presenta un perfil de ondulación suavizado y simétrico, lo cual ayuda a preservar los detalles de la señal durante la descomposición. Esta característica es clave para minimizar la distorsión de los componentes relevantes, manteniendo una buena resolución tanto en el tiempo como en la frecuencia.

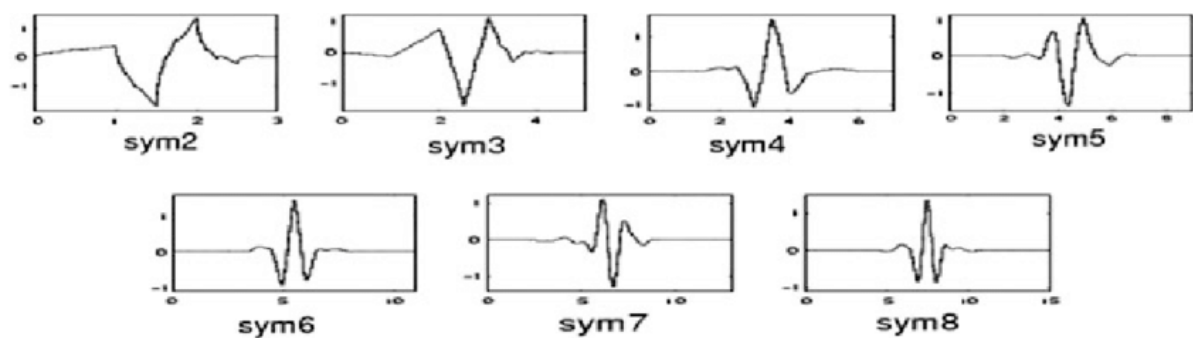


Figura 1. Representación de diferentes tipos de onda wavelet Symlets. Se aprecia la forma de la sym 4 similar al complejo QRS

3. Escoger 10 señales al azar y aplicar un flujo de procesamiento que conste de:

#### Flujo 1

- Filtro pasa-altas usando filtro IIR a 0.5 Hz. Justificar la elección de parámetros y si se usa FIR o IIR
- Filtro wavelet modificado del punto 3
- Filtrado pasa bajas 50 Hz. Justificar la elección de parámetros y si se usa FIR o IIR

#### Flujo 2

- Detrend
- Filtro wavelet modificado del punto 3
- Filtrado pasa bajas 50 Hz. Justificar la elección de parámetros y si se usa FIR o IIR

### **Flujo 3**

- **Filtro pasa-altas usando filtro IIR a 0.5 Hz. Justificar la elección de parámetros y si se usa FIR o IIR**
- **Filtrado pasa bajas 50 Hz. Justificar la elección de parámetros y si se usa FIR o IIR**

**Describir los resultados obtenidos y decidir si el resto del procesamiento se hace con el flujo 1, el flujo 2 o el flujo 3 (20%)**

Para mantener los procesos de filtrado lo más parecidos posibles y poder observar la diferencia que hacían algunos pasos extra se utilizó el mismo filtro IIR pasa bajas a 0.5 Hz en los Flujos 1 y 3, así como el mismo filtro FIR pasa altas a 50 Hz para los 3 flujos de filtrado.

#### **Filtro Pasa-Altas**

Para el filtro pasa-altas se hizo uso de un filtro IIR (porque estaba especificado en los flujos) con frecuencia de corte a 0.5 Hz y aproximación Butterworth.

Se usa esta frecuencia de corte pues el ruido de línea de base y otros artefactos de baja frecuencia suelen estar por debajo de 0.5 Hz. Por lo tanto, un filtro pasa-altas a 0.5 Hz es adecuado para eliminar estos artefactos, preservando las frecuencias importantes de la señal de ECG que suelen comenzar alrededor de 1 Hz.

Se suele usar un filtro Butterworth en procesamiento de señales biomédicas debido a su respuesta de amplitud plana en la banda de paso, evitando distorsiones en las frecuencias clave de la señal de ECG. Aunque otros filtros como el Chebyshev pueden proporcionar una transición más rápida, pueden generar ondulaciones indeseadas en la banda de paso.

#### **Filtro Pasa-Bajas**

Para el filtro pasa-bajas se hizo uso de un filtro FIR, pues aunque requiere un mayor orden para lograr el filtrado objetivo (aumentando el costo computacional), no genera distorsión en la fase de la señal, lo cual es importante en aplicaciones de tiempo real y monitoreo médico.

La ventana de Hamming se elige porque proporciona una buena relación entre el ancho de banda de transición y la atenuación en la banda de rechazo. Ofrece una reducción de "efecto campana" en las frecuencias de transición y es una elección común en el procesamiento de señales biomédicas debido a su rendimiento equilibrado.

Para el orden del filtro se calculó que para obtener una banda de transición menor a 10 Hz (y así poder asegurar que se atenúa el ruido de la red eléctrica a 60 Hz) debía ser mayor a 160. Para reducir aún más el ancho de banda se utilizó un filtro de orden 200.

### Resultados Filtrado

Luego de utilizar los 3 procesos de filtrado en 10 señales aleatorias se obtuvieron los resultados que se observan a continuación:

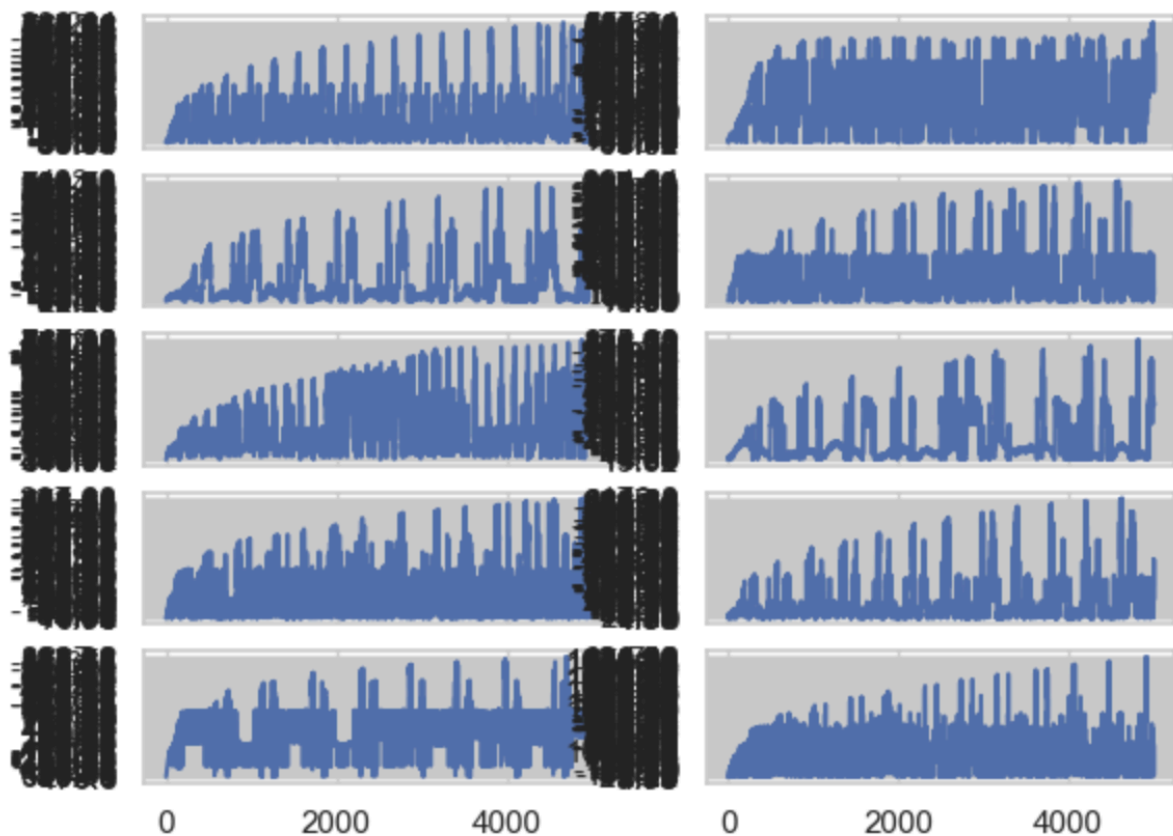


Figura 2. Señales originales con ruido

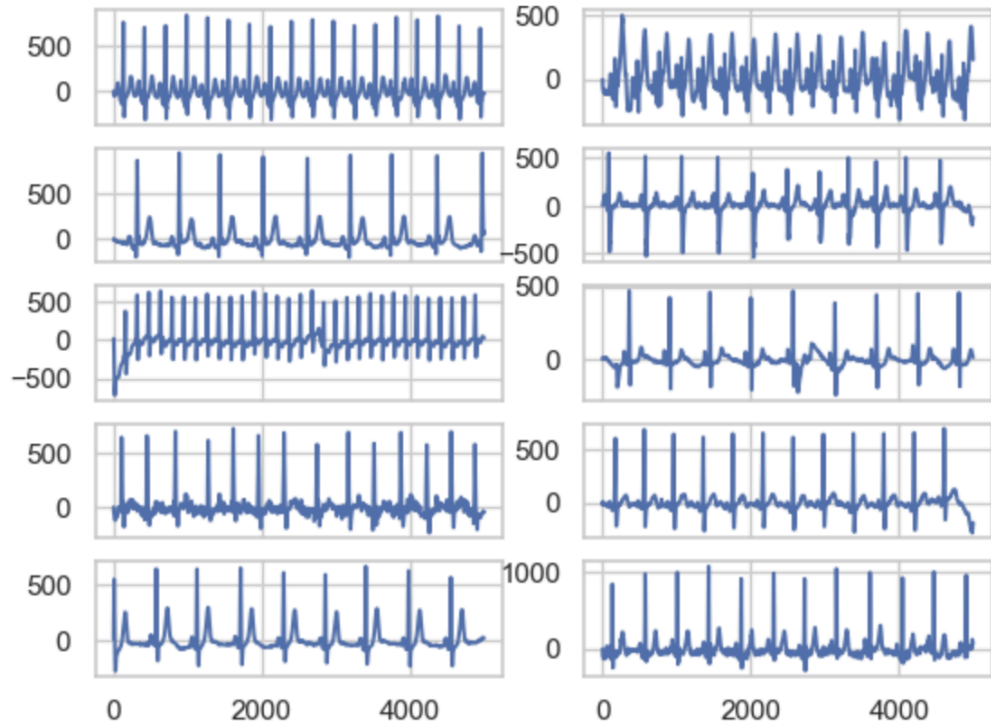


Figura 3. Resultados flujo de filtrado 1 (IIR 0.5Hz - Wavelet sym4 - FIR 50Hz)

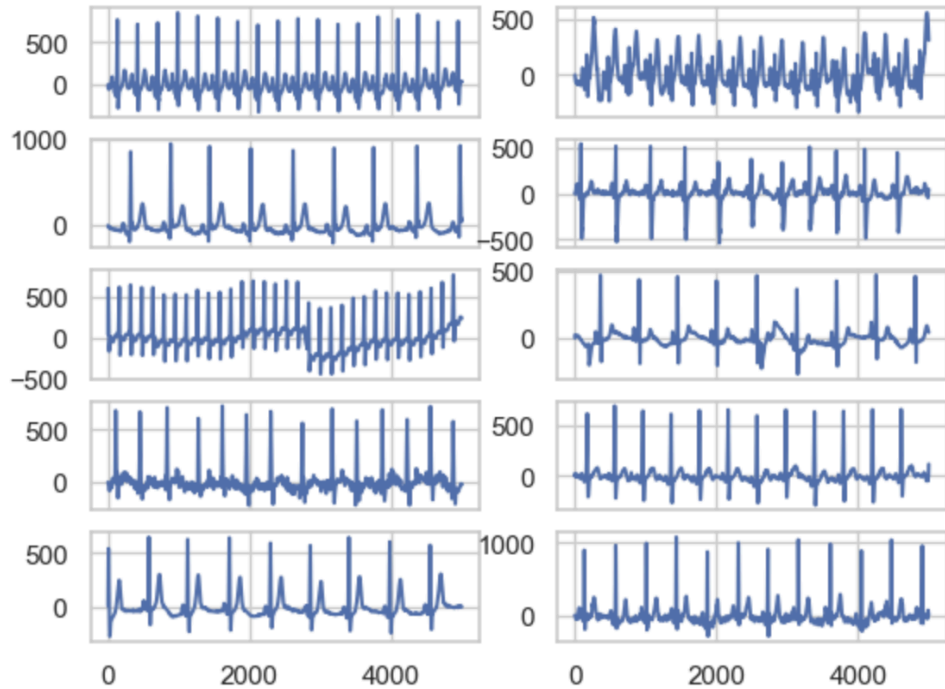


Figura 4. Resultados flujo de filtrado 2 (Detrend - Wavelet sym4 - FIR 50Hz)

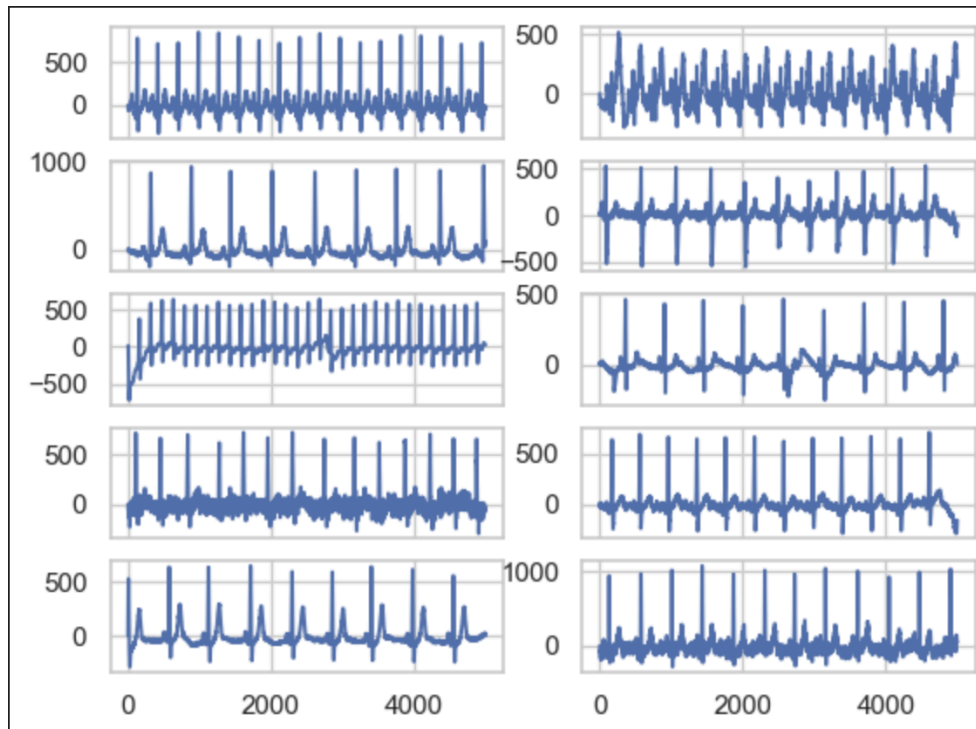


Figura 5. Resultados flujo de filtrado 3 (IIR 0.5Hz - FIR 50Hz)

Analizando el resultado del flujo de filtrado 3, podemos apreciar la importancia del filtro Wavelet para la definición de los complejos de la onda mediante la reducción de ruido en los mismos. En los flujos 1 y 2 se obtiene una onda con mucho menor ruido y complejos mucho más claros.

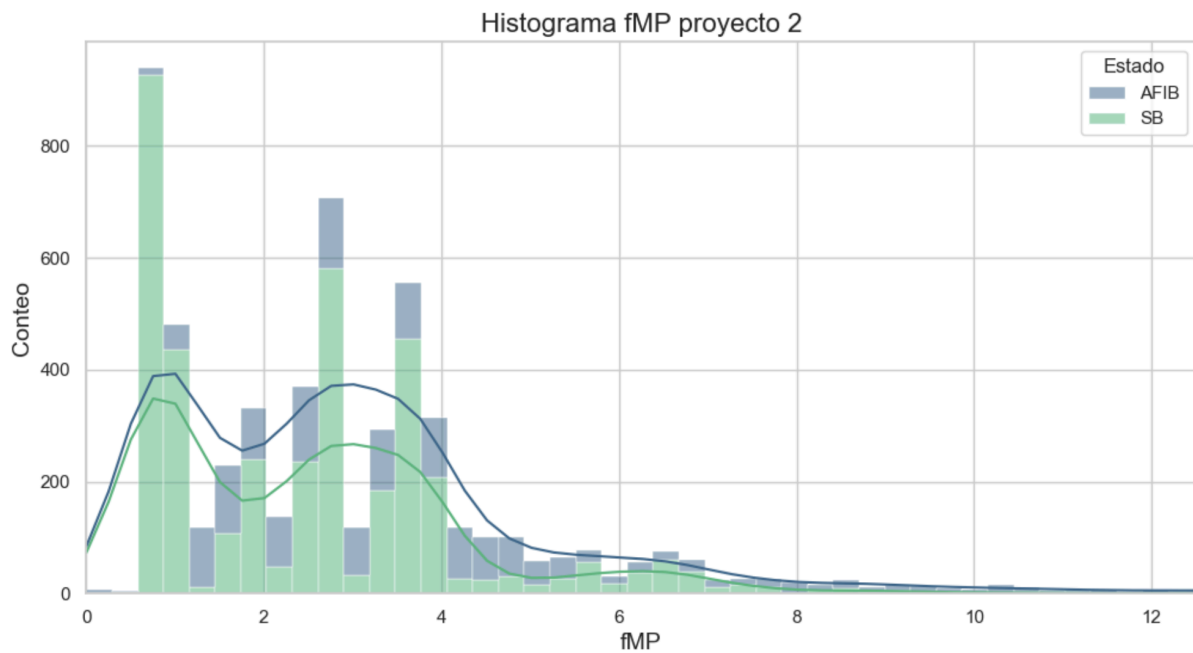
Se toma la decisión de proceder de ahora en adelante con el flujo de filtrado 2. Aunque el flujo 1 y 2 obtienen resultados similares, al evitar el uso de un filtro IIR en el segundo flujo, estamos seguros de que la fase y respuesta frecuencial de la señal no se verán afectadas, más allá del denoising que se busca, además se pudo apreciar que en el proceso de detrend también se atenúan algunas bajas frecuencias, obteniendo resultados similares sin afectar la fase.

**4. Comparar los resultados de fMP del proyecto 3 con los del proyecto 2 usando estadística descriptiva: gráficos y pruebas de hipótesis (15%)  
Discuta si hay más diferencias entre los tipos de señales con el flujo de procesamiento propuesto respecto al que se trabajó en el proyecto 2.**

Se obtuvieron gráficos para el fMP de ambas poblaciones en ambos casos de estudio (proyectos 2 y 3), cómo se muestra a continuación:



a)



b)

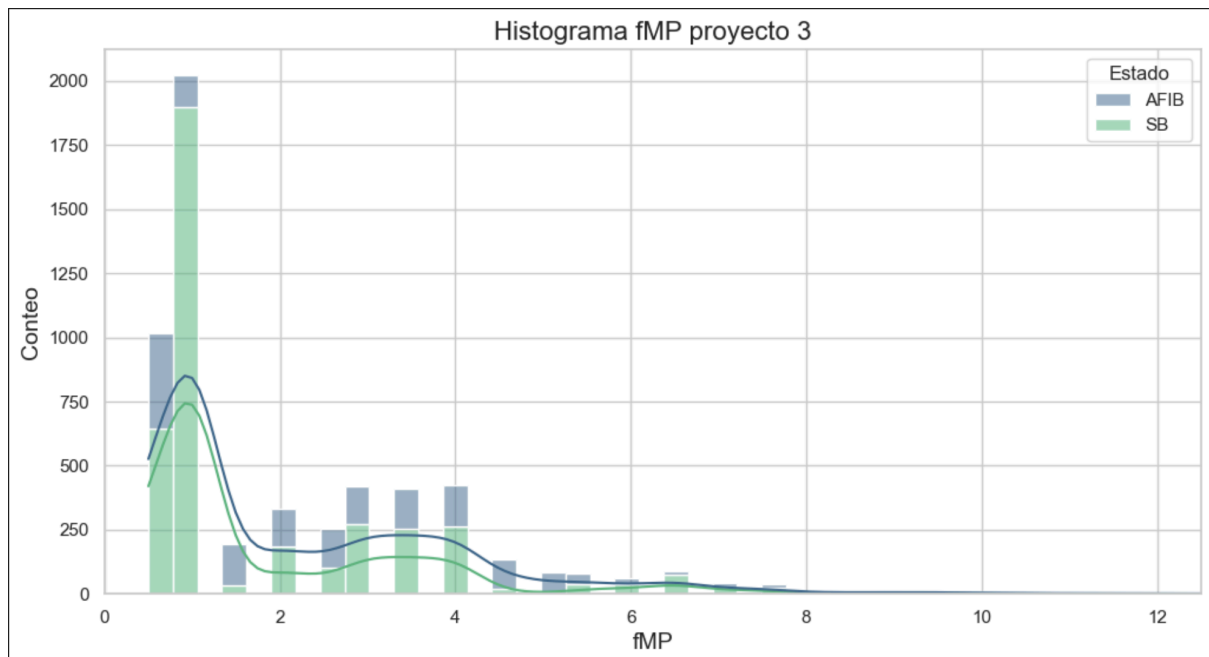
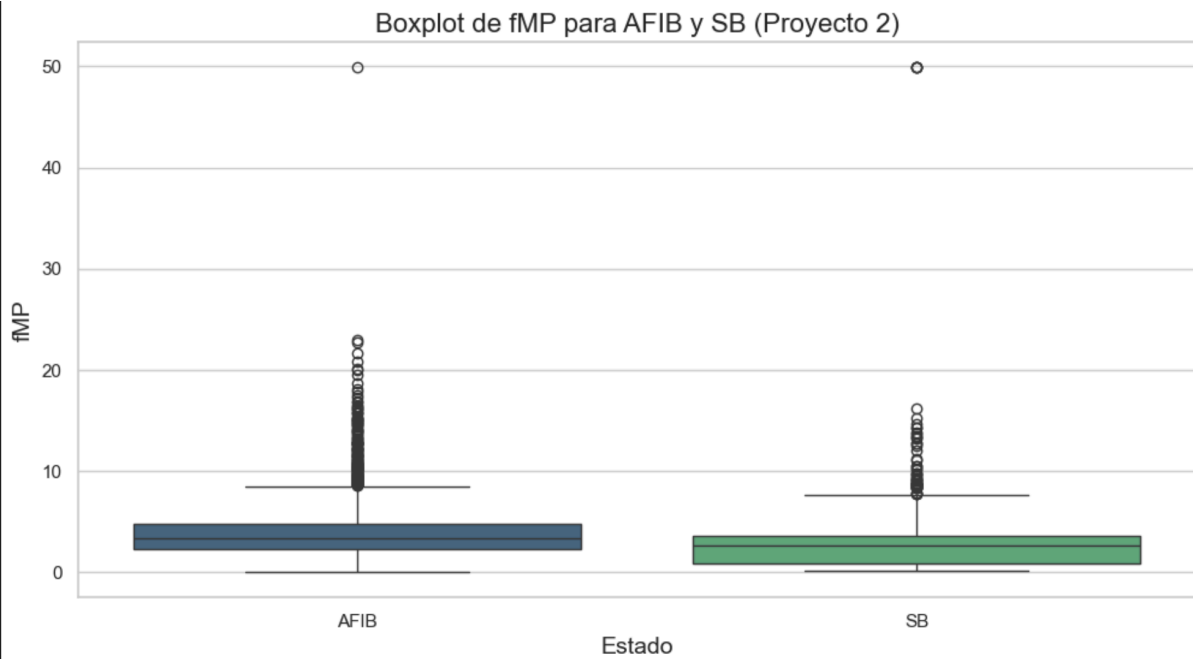


Figura 6. Histogramas para fMP de los grupos AFIB y SB. a) Proyecto 2 (ECGDataDenoised). b) Proyecto 3 (ECGData)

Se puede apreciar un cambio en la distribución con la potencia más concentrada en las frecuencias bajas en los resultados del proyecto 3. Esto puede estar relacionado con la ausencia del filtro específico pasa-altas, y su reemplazo con un detrend, sin embargo, también se aprecia una menor dispersión en las frecuencias de interés, lo cual puede facilitar el análisis y clasificación

a)



b)

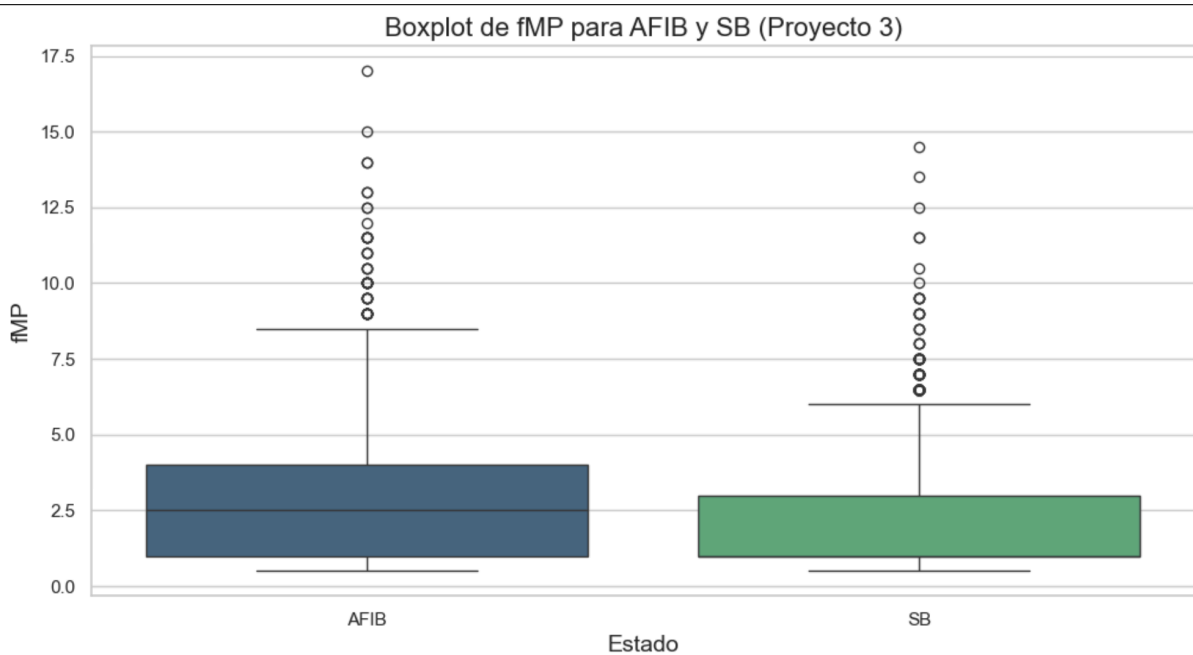


Figura 7. Boxplots para fMP de los grupos AFIB y SB. a) Proyecto 2 (ECGDataDenoised). b) Proyecto 3 (ECGData)

Se puede apreciar una diferencia importante, sobre todo en la presencia de outliers. El comportamiento es similar entre ambos estudios, pero nuestro flujo de filtrado del proyecto 3 permite la eliminación de frecuencias outliers por encima de 20, lo cual se puede asociar directamente con el filtro FIR, y permite apreciar más claramente la diferencia en media y cuartiles entre los 2 grupos.

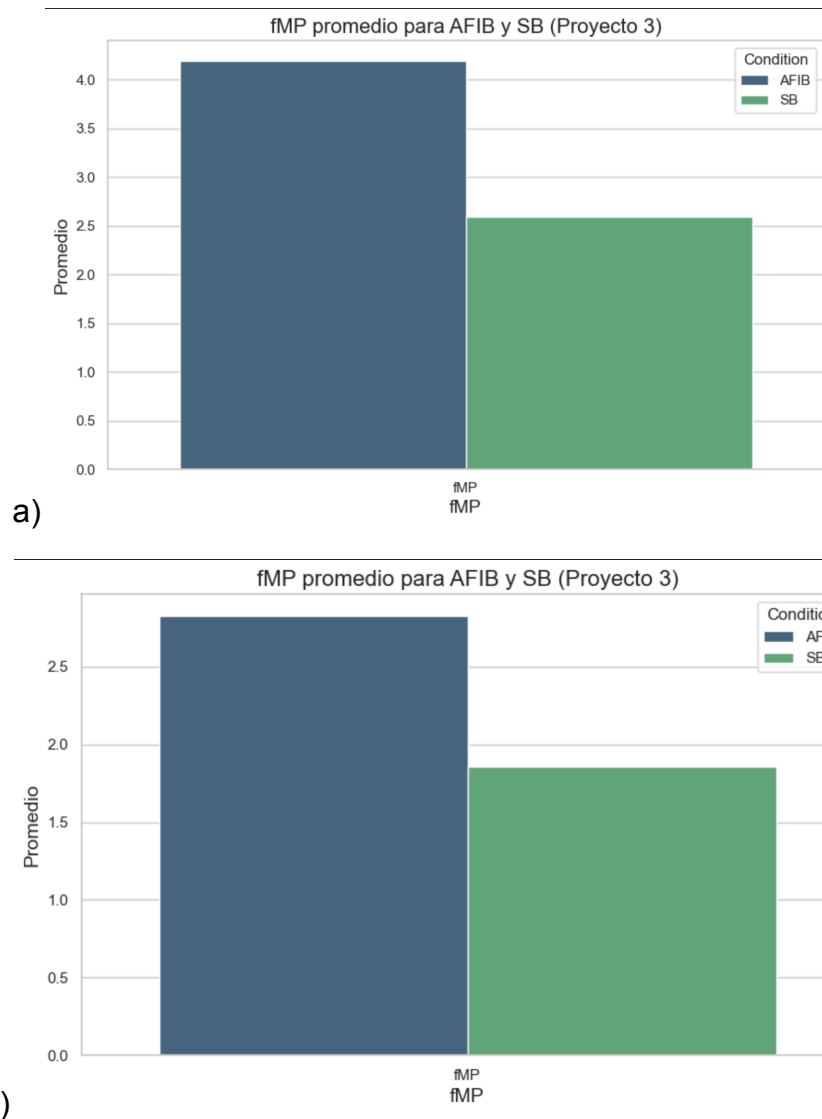


Figura 8. Gráficos de barras para fMP promedio de los grupos AFIB y SB. a) Proyecto 2 (ECGDataDenoised). b) Proyecto 3 (ECGData)

Como se pudo evidenciar en el boxplot, la diferencia en el promedio de fMP de los 2 grupos se mantiene entre los estudios, sin embargo, se puede apreciar que en el proyecto 3, el valor nominal de estos promedios es menor para ambos grupos, esto se puede relacionar también con la distribución vista previamente en los histogramas, donde para el proyecto 3 las bajas frecuencias eran aún más predominantes, y sabemos también que en el proyecto 3 se lograron filtrar algunos outliers que pueden también afectar el valor promedio de la métrica.

Por último las pruebas de hipótesis confirmaron los resultados previamente obtenidos en el proyecto 2, pues para ninguno de los grupos (AFIB o SB) los datos

de fMP siguieron una distribución normal. Y al igual que en el primer estudio, la prueba t-student permitió descartar la hipótesis nula (p-value tiende a 0), la cual consiste en que no existe una diferencia estadística entre estos 2 grupos cuando se evalúa la métrica fMP, por lo cual se llega a la misma conclusión que en el proyecto anterior.

## Conclusiones

- El proceso de filtrado de una señal presenta un alto grado de variabilidad y es responsabilidad del encargado del proceso definir frecuencias, parámetros y prioridades a la hora de abordarlo. Diferentes técnicas de filtrado, filtros individuales o flujos de filtrado tendrán siempre puntos fuertes y debilidades frente a otras técnicas, y la decisión del mecanismo óptimo dependerá de que exista esa definición de características previas del estudio por parte del encargado.
- Para aplicaciones médicas y especialmente para ECG (monitoreo en tiempo real) es recomendable usar filtros FIR y evitar el uso de filtros IIR, pues estos últimos a medida que aumenta el orden (y por ende la capacidad de asemejarse a un filtro ideal), aumentan la distorsión de la fase, lo cual puede a la larga afectar el estudio en frecuencia de la señal objetivo.
- Aunque muy probablemente se realizó un proceso de filtrado distinto al original, con el que se obtuvieron los datos para el proyecto 2, se puede concluir que el filtro cumplió con el objetivo de retirar la mayor cantidad de ruido posible de la señal objetivo, y esto se ve reflejado en la similitud estadística de los resultados. Aunque se presentaron cambios en la distribución o el valor nominal, las conclusiones que se lograron obtener para la métrica de fMP son homólogas en ambos estudios

## Referencias

[1] Zheng, J., Chu, H., Struppa, D. *et al.* Optimal Multi-Stage Arrhythmia Classification Approach. *Sci Rep* 10, 2898 (2020).

<https://doi.org/10.1038/s41598-020-59821-7>

[2] Samann, Fars and Schanze, Thomas. "An efficient ECG Denoising method using Discrete Wavelet with Savitzky-Golay filter" *Current Directions in Biomedical Engineering*, vol. 5, no. 1, 2019, pp. 385-387.

<https://doi.org/10.1515/cdbme-2019-0097>

**[3] Jarrín Ochoa, Diego Andrés. "Diseño e implementación de filtros digitales FIR e IIR utilizando el microcontrolador XMEGA de Atmel para tratamiento de señales de audio". Dirigido por Oñate Cadena, Luis Germán. Junio 2016.**