



Universidad de Buenos Aires

Facultad de Ingeniería

Proyecto Especial

Detección de latidos cardíacos a partir de la grabación de video de
un smartphone

Señales y Sistemas (66.74 / 66.86)

Segundo cuatrimestre de 2017.

WERNER, Ezequiel Maximiliano - 95922 - Ingeniería en Informática.

Introducción

El proyecto consiste en realizar una serie de análisis de señales biométricas tomadas con dispositivos móviles, con el objetivo de poner en práctica los contenidos de la materia.

El análisis se realizará a través del desarrollo de scripts en Octave.

Las señales biométricas fueron tomadas con la cámara de video y el micrófono de un teléfono celular y representan una sumatoria para cada tiempo de los píxeles de la pantalla. Cada uno de los píxeles tiene una componente roja, una azul, y una verde, y por ende nuestro script tendrá como entrada tres arreglos representando las componentes del video, y un cuarto arreglo representando la grabación de audio. Ninguna de las señales tendrá unidades de magnitud en su representación.

El análisis se compone de una secuencia de pasos sugeridos, por lo que el desarrollo del trabajo práctico consistirá en seguir y explicar dichos pasos.

Desarrollo

Ejercicio 1

Cargar el archivo *intensidad_RGB.mat* y visualizar la señal RGB en superposición (3 canales). Elija alguno de los 3 canales que, según su criterio, posea la señal más útil a los efectos de analizar la dinámica de los latidos y utilícelo para realizar los puntos siguientes.

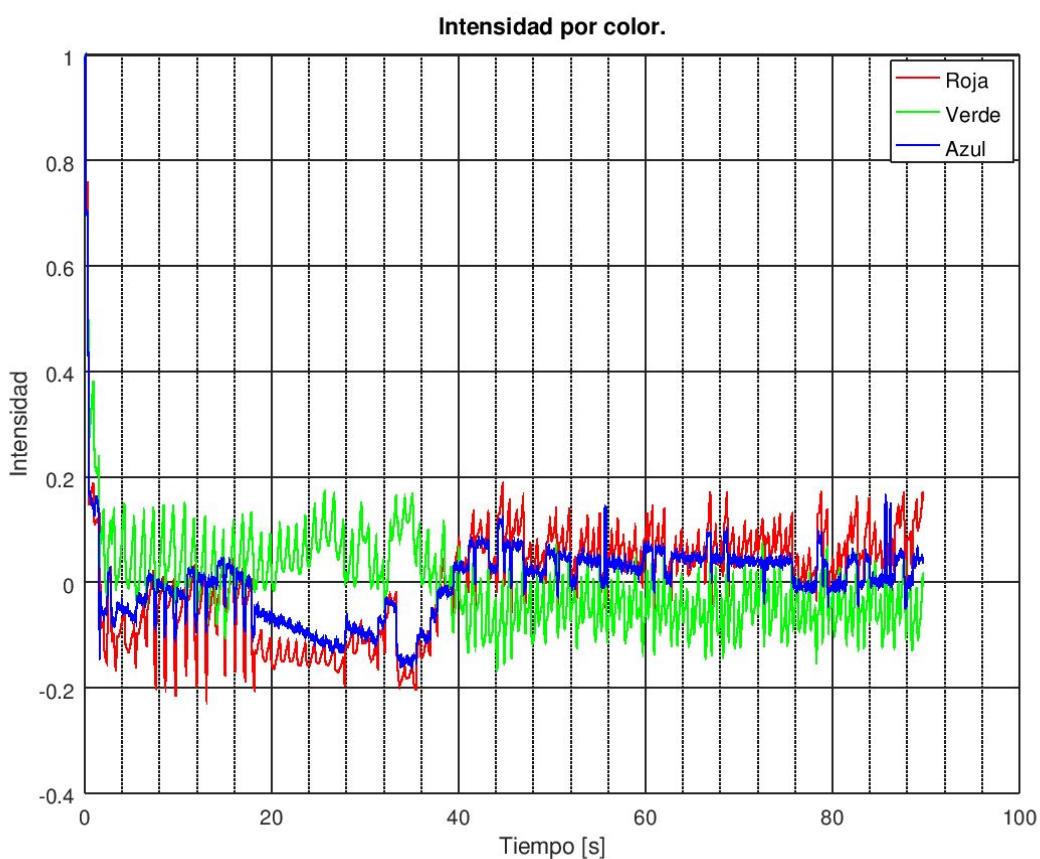


Figura 1. Intensidad normalizada de las señales por color.

En la figura 1 se pueden observar las tres señales que representan el video grabado por el celular. Las señales fueron divididas por su máximo valor para que dicho máximo quede con valor 1 después del escalado. Esta transformación es válida porque la señal no tiene unidades de amplitud y sólo nos interesa la forma. La señal elegida para el resto del análisis será la componente verde, ya que sus picos son más marcados a simple vista.

Ejercicio 2

Del gráfico del punto anterior, estimar aproximadamente los latidos por minuto (LPM). Identifique el momento a partir del cual la frecuencia cardiaca comienza a incrementarse, y en cuánto se incrementa.

La estimación de este ejercicio se automatizó más allá de lo requerido para reducir la información “a ojo” dentro del análisis.

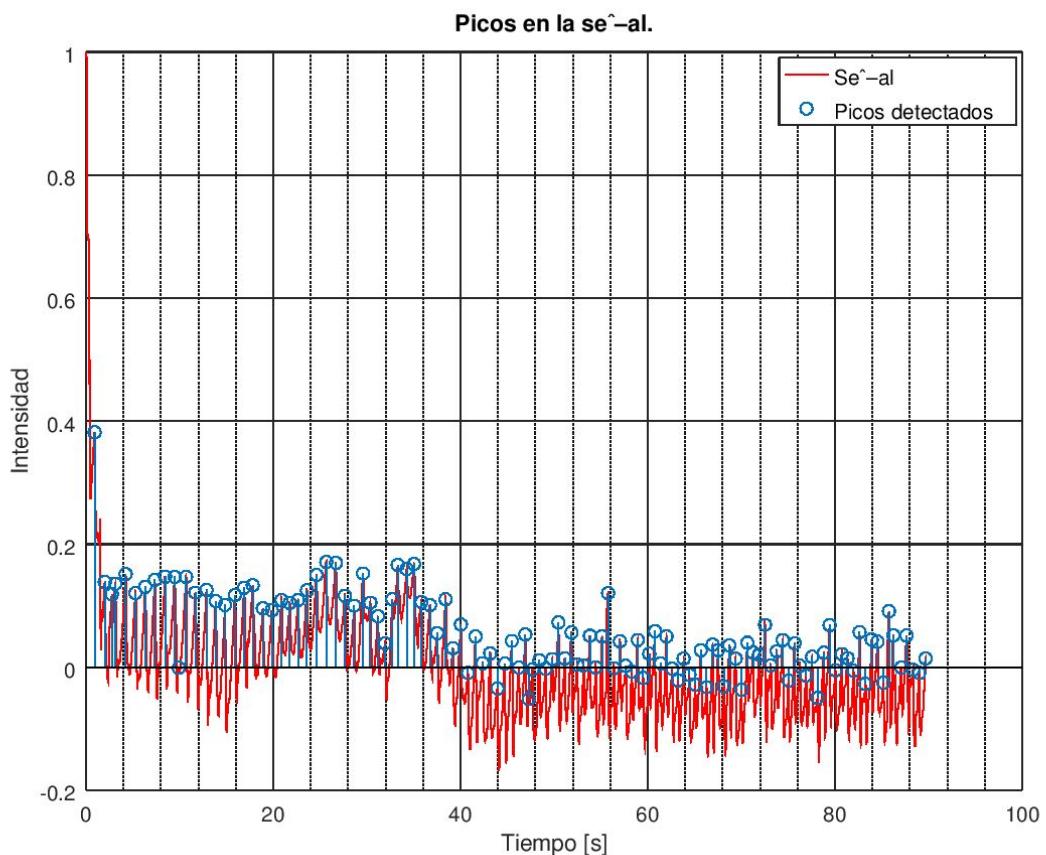


Figura 2. Picos de la señal sin ningún filtro.

Para generar la figura 2 se trasladó la señal hacia arriba ubicando el mínimo de la misma en el cero, y luego se utilizó la función `findpeaks` de Octave variando los parámetros que recibe hasta obtener un resultado descriptivo de la señal. Luego se corrigió dicha traslación y se guardó el resultado.

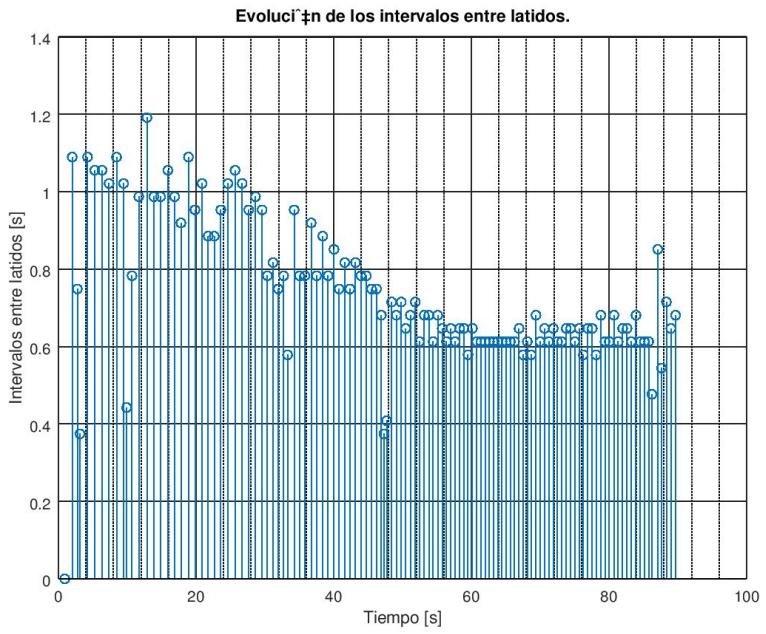


Figura 3. Intervalos entre picos de la señal sin ningún filtro.

El gráfico de la figura 3 muestra una tendencia decreciente de los intervalos (una frecuencia creciente) desde 1.1 segundos hasta 0.73 segundos.

Se podría decir que, entonces, la frecuencia cardíaca va desde unos 65.5 LPM hasta unos 95 LPM. Cuando alcanza este último valor alrededor de los 50 segundos, se estanca y se mantiene pareja hasta terminar la muestra. Dichas frecuencias se pueden visualizar en la figura 4.

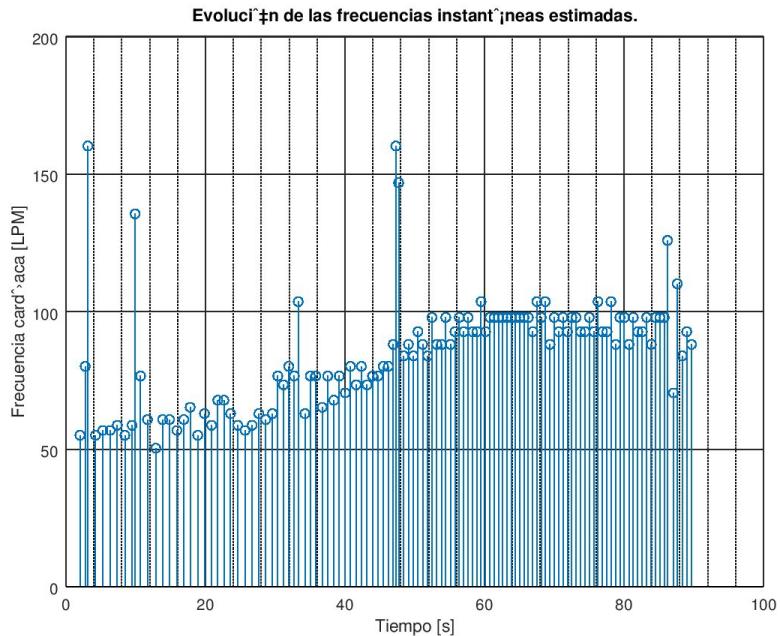


Figura 4. Frecuencia instantánea calculada a partir del gráfico anterior.

Ejercicio 3

Rehacer el punto anterior, pero utilizando DFT. ¿Puede en este caso identificar cuándo se da el cambio de LPM?

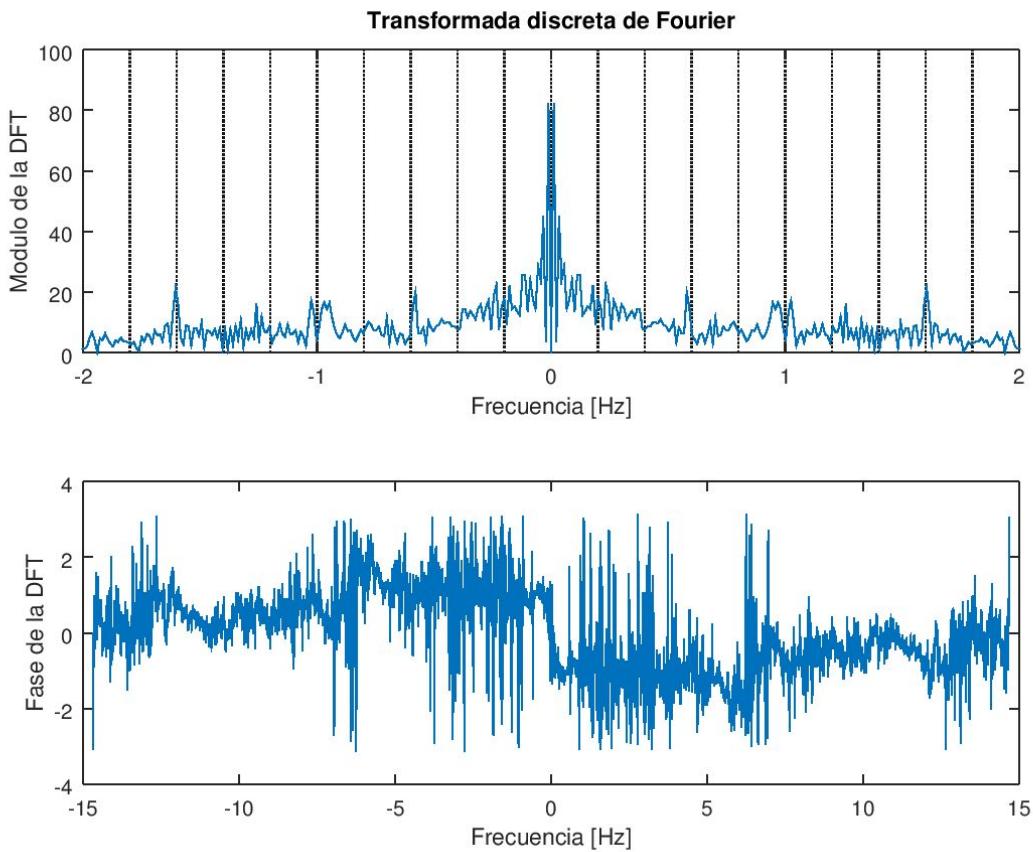


Figura 5. Transformada discreta de Fourier de la señal.

En el gráfico de la figura 5 se ve un pico en unos 1.6Hz (96 LPM) correspondiente a la frecuencia en la que queda la señal luego de los 50 segundos, y también se ven varios picos parecidos entre 40 y 80 LPM, que se condicen con la frecuencia creciente que veíamos antes de dicho tiempo.

El problema de este gráfico es que no aporta información sobre cuándo se ven esas frecuencias (no tiene localización temporal).

Ejercicio 4

Diseñar un filtro pasa-banda tipo Butterworth con banda de paso entre 0.5 Hz y 10 Hz. Graficar respuesta en frecuencia (módulo y fase), diagrama de polos y ceros y respuesta al impulso (sugerencia: usar las funciones butter y fvtool de Matlab). ¿Qué papel juega el orden del filtro seleccionado en su diseño?

En las siguientes páginas se mostrarán cuatro filtros que cumplen con estas especificaciones, con distintos órdenes: 2, 4, 6, y 10.

Se puede observar en las figuras 6 a 17 que a medida que aumenta el orden del filtro las pendientes de la respuesta en frecuencia se hacen más pronunciadas, pero la fase se altera más en los bordes.

También se nota la relación entre el diagrama de polos y ceros con el módulo de la respuesta en frecuencia, ya que cuando hay polos y ceros cercanos hay más pendiente en el diagrama de módulo, y además que haya ceros en 1 y -1 hace que el filtro sea un pasa-banda.

Por último, se observa que el transitorio de la respuesta al impulso es más largo a medida que se aumenta el orden del filtro.

Para el resto de los ejercicios, cuando se necesite un filtro Butterworth se utilizará uno de orden 10.

Filtro de orden 2

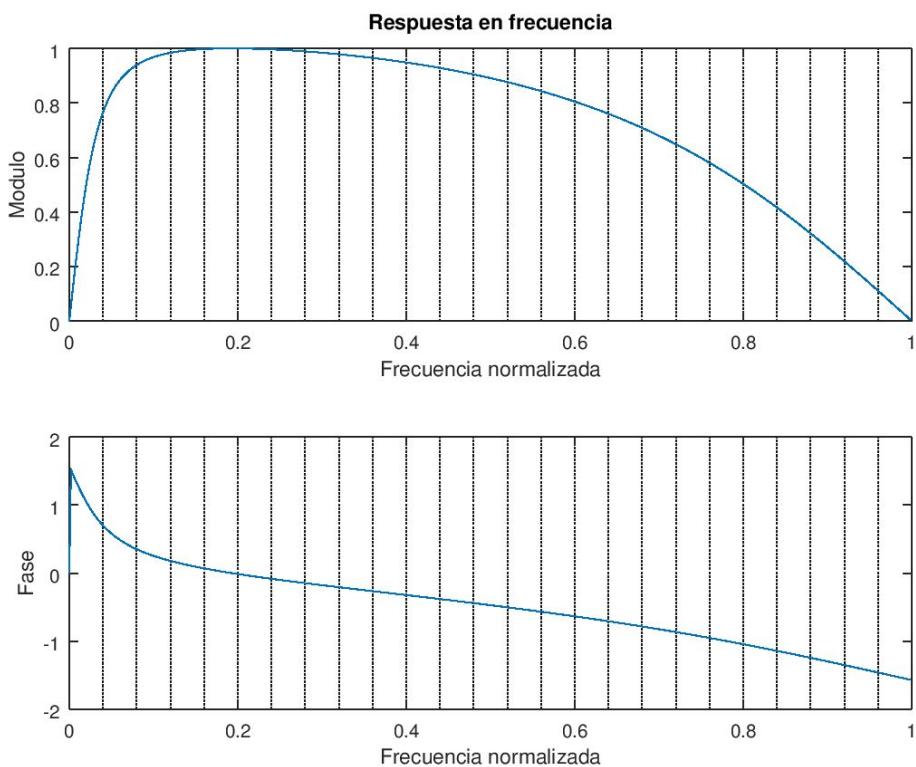


Figura 6. Respuesta en frecuencia del filtro de orden 2.

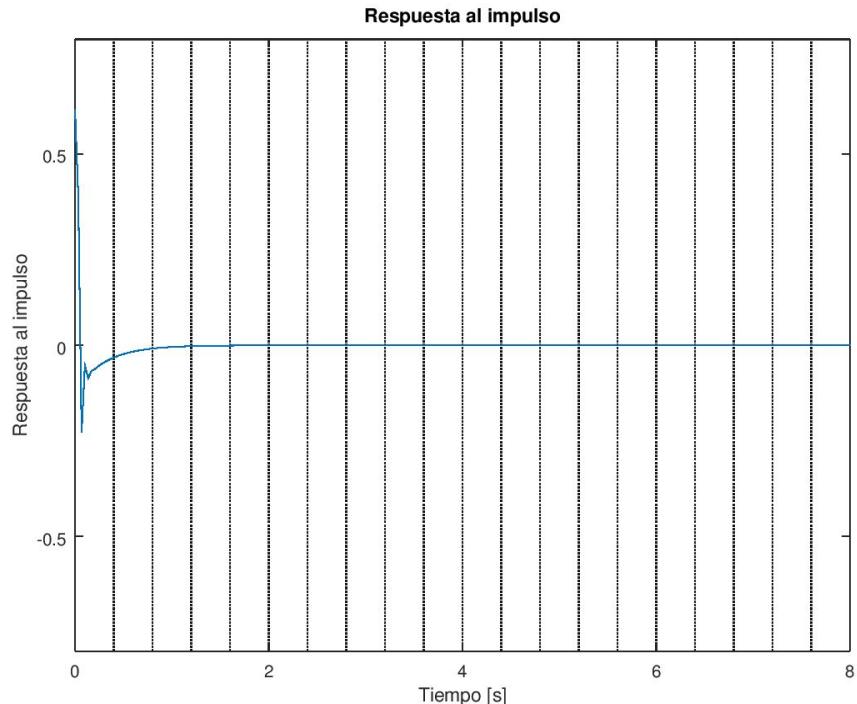


Figura 7. Respuesta al impulso del filtro de orden 2.

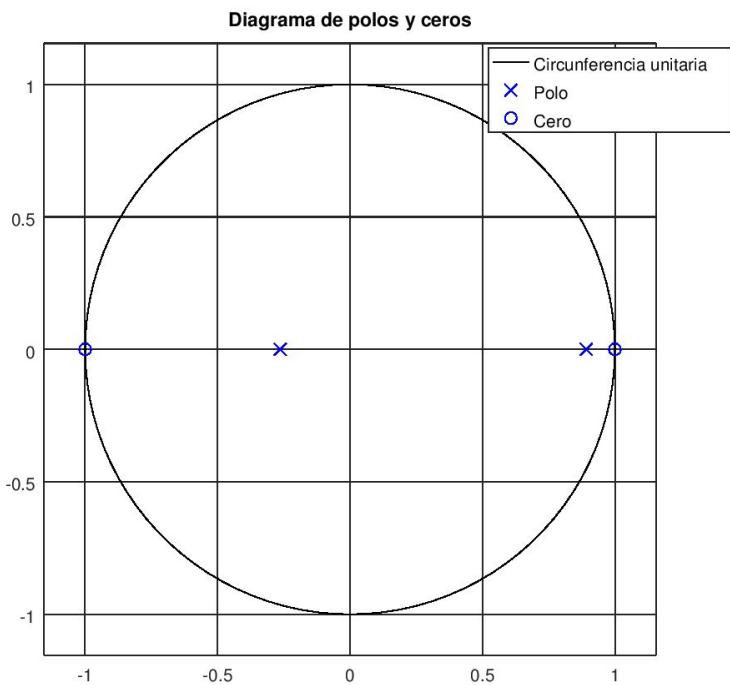


Figura 8. Diagrama de polos y ceros del filtro de orden 2.

Filtro de orden 4

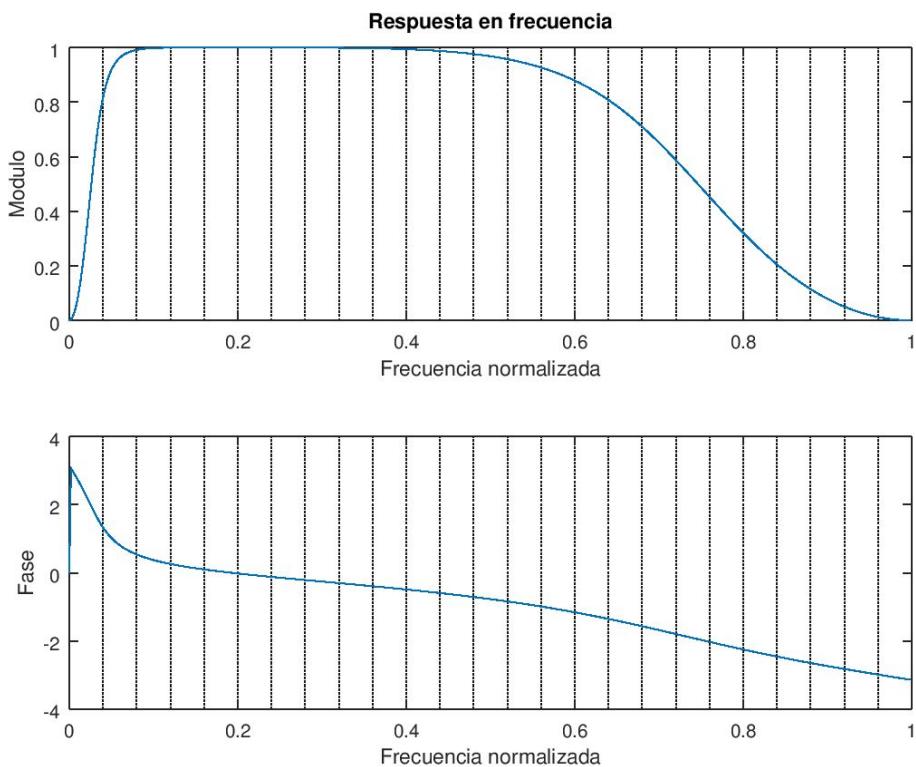


Figura 9. Respuesta en frecuencia del filtro de orden 4.

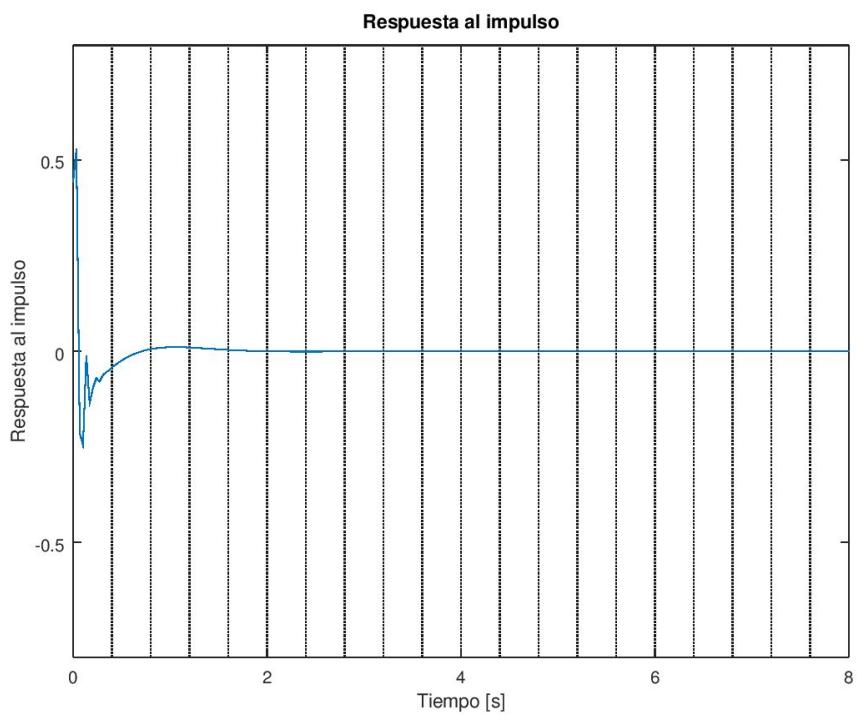


Figura 10. Respuesta al impulso del filtro de orden 4.

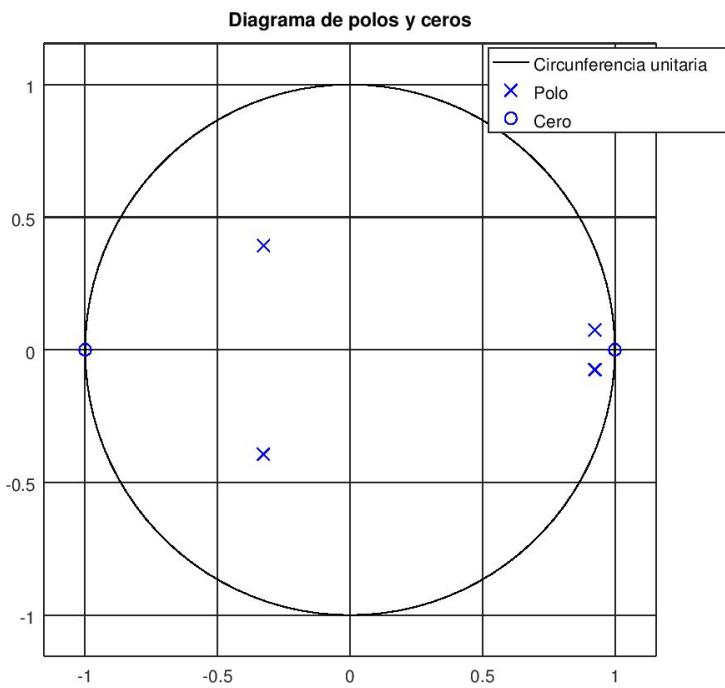


Figura 11. Diagrama de polos y ceros del filtro de orden 4.

Filtro de orden 6

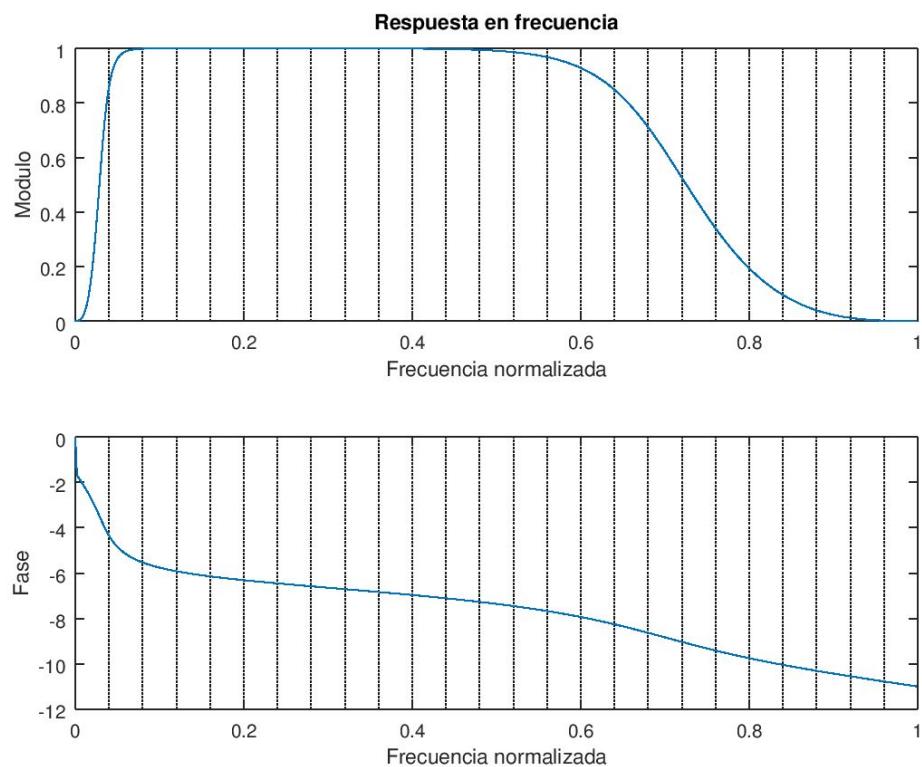


Figura 12. Respuesta en frecuencia del filtro de orden 6.

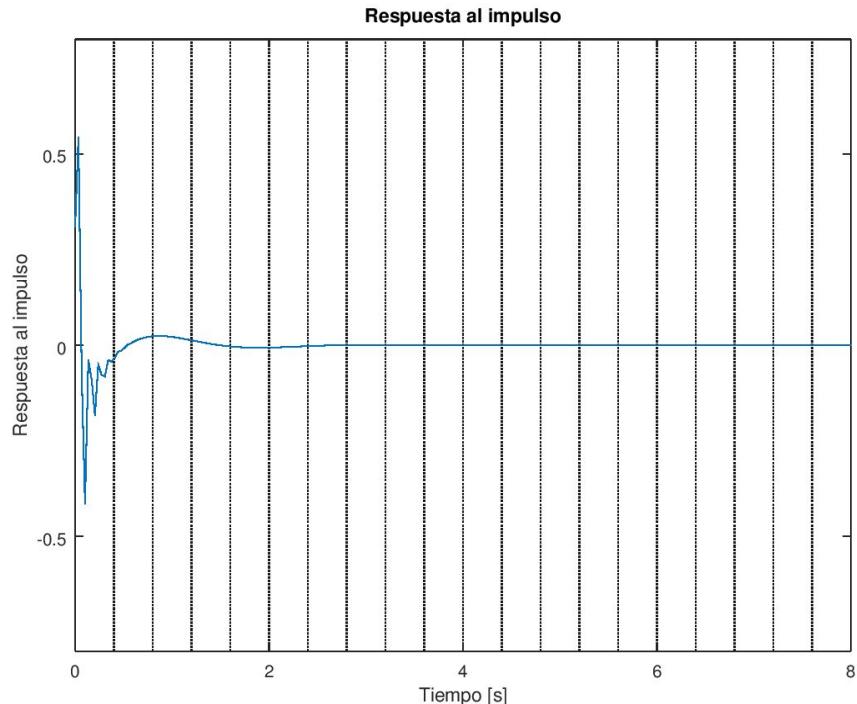


Figura 13. Respuesta al impulso del filtro de orden 6.

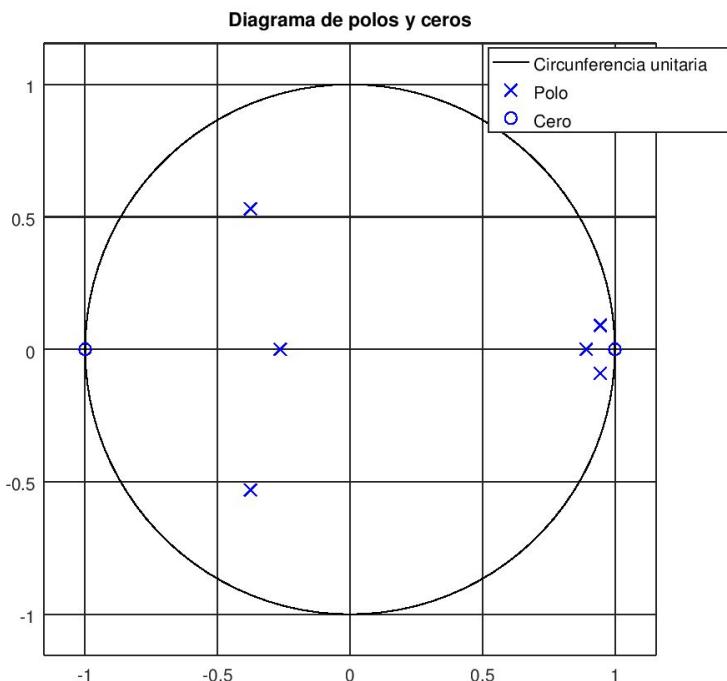


Figura 14. Diagrama de polos y ceros del filtro de orden 6.

Filtro de orden 10

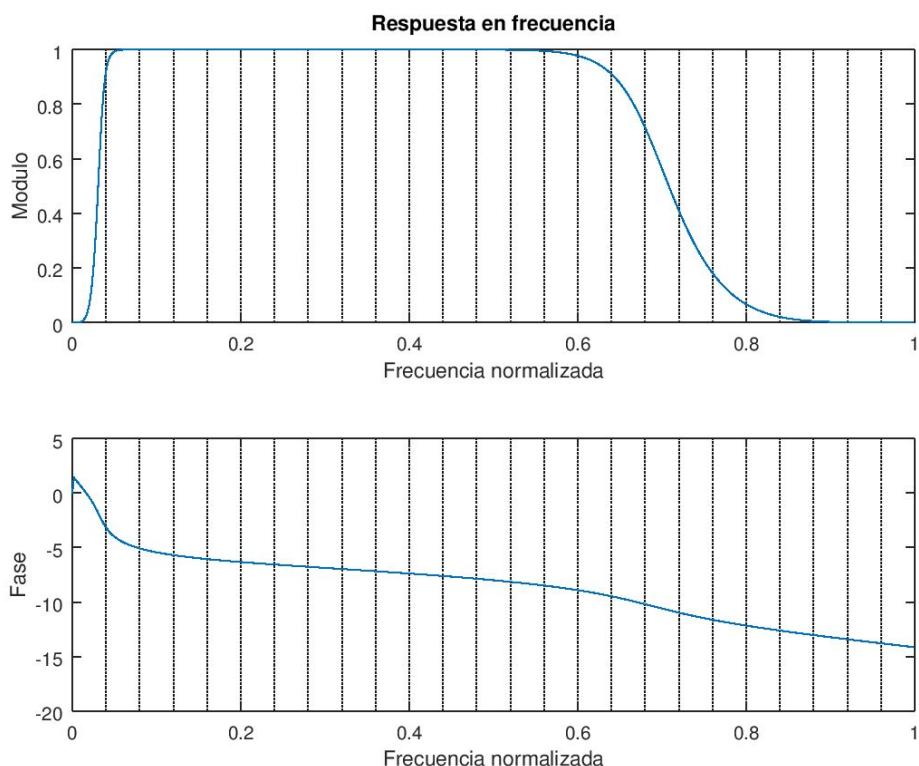


Figura 15. Respuesta en frecuencia del filtro de orden 10.

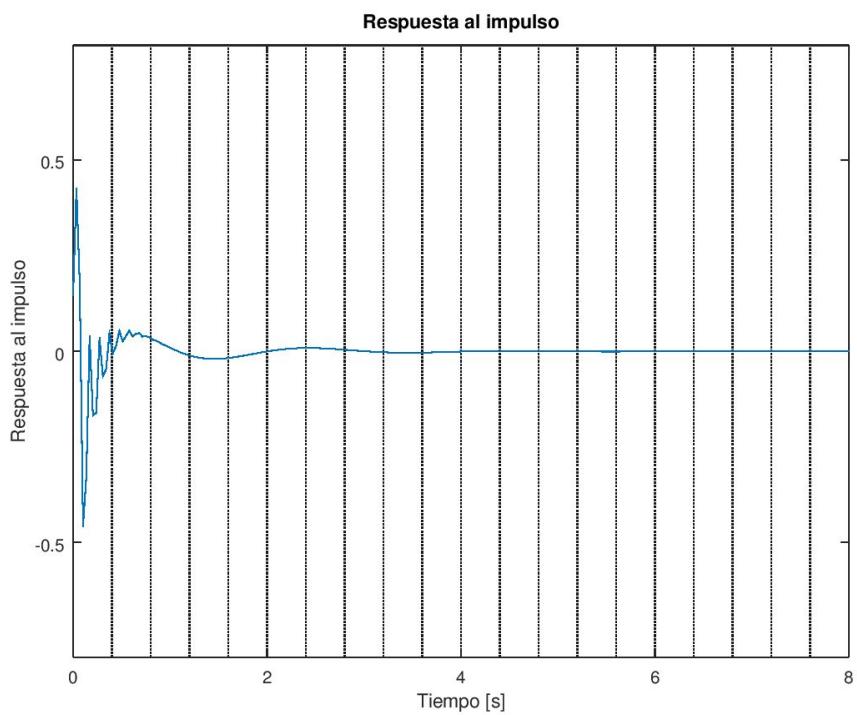


Figura 16. Respuesta al impulso del filtro de orden 10.

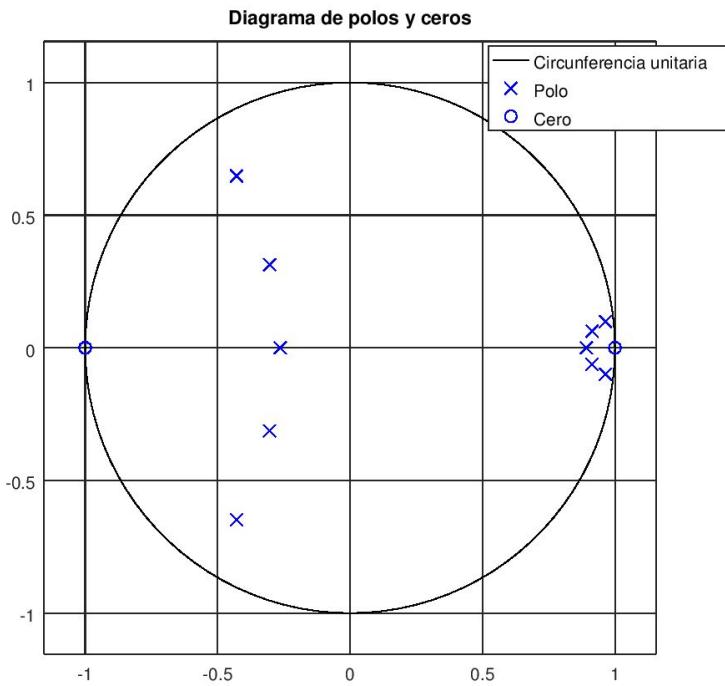


Figura 17. Diagrama de polos y ceros del filtro de orden 10.

Ejercicio 5

Filtrar la señal FPG utilizando el filtro diseñado en el punto anterior mediante la función filter. Grafique en superposición la señal original con la filtrada y comente acerca de:

- a. Remoción de derivas
- b. Cambios en la forma de la señal
- c. Retardo de la señal filtrada respecto de la original

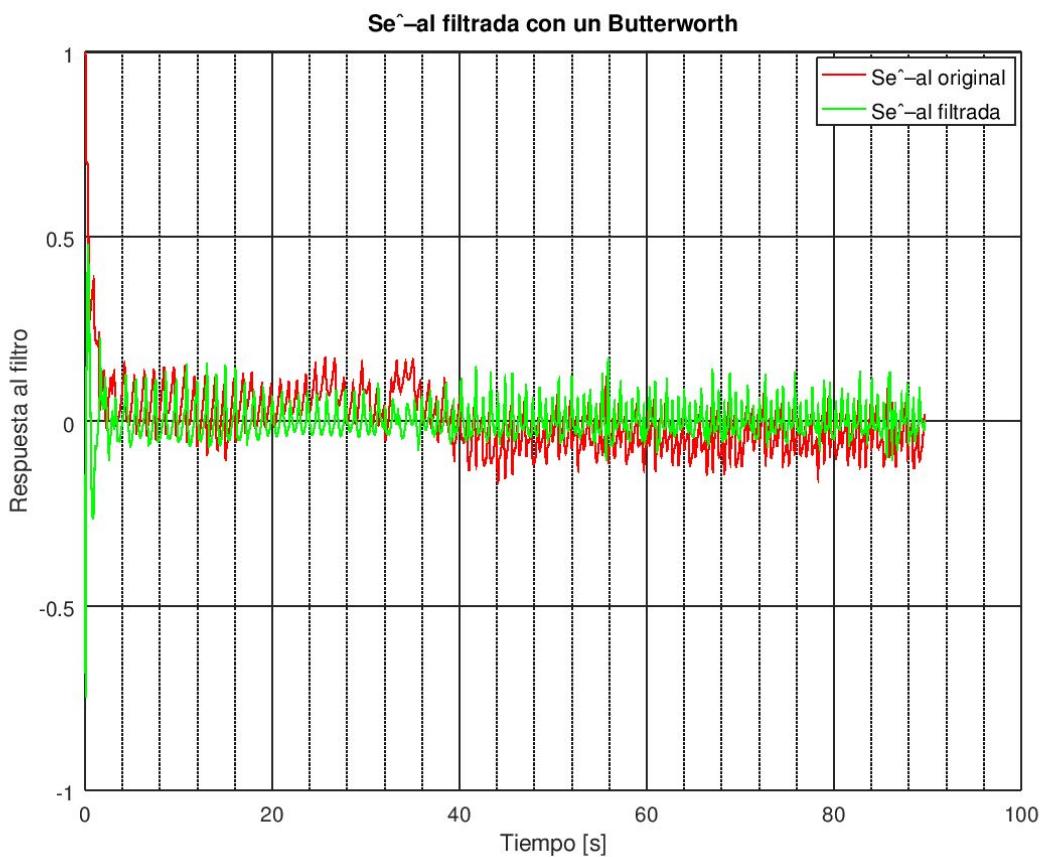


Figura 18. Señal original y su salida del filtro Butterworth en superposición.

En cuanto a la remoción de derivas, en la figura 18 se ve que desaparecen de la señal las componentes de frecuencia baja (sobre las que parece montada la señal original).

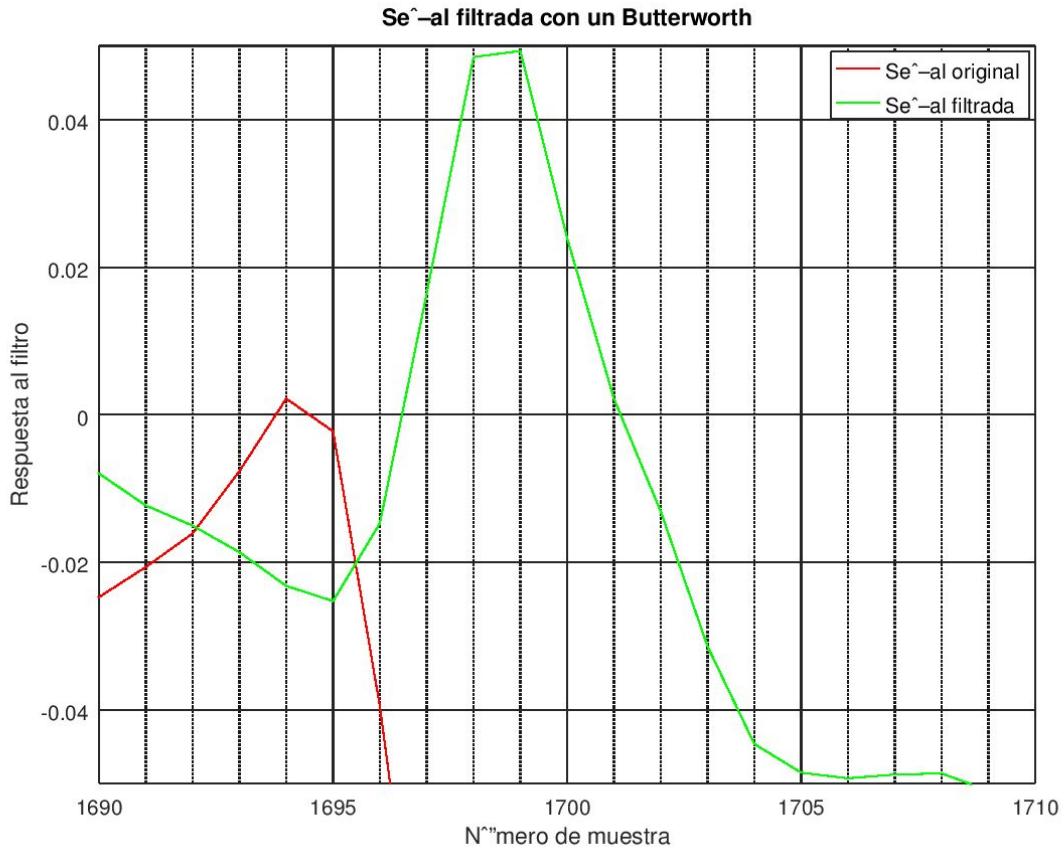


Figura 19. Zoom de la figura anterior, con el eje de abscisas en número de muestra.

Además, como se muestra en la figura 19, se observa una deformación de la señal, dada por lo mucho que varía la fase en las altas frecuencias, que son las que le dan forma a los picos. También se puede observar en varias zonas de dicha figura que la señal filtrada tiene un retardo de aproximadamente 4 muestras respecto de la original.

Ejercicio 6

A partir de la respuesta en fase del filtro, calcule su retardo temporal y compare con lo observado en el punto 5c.

El retardo temporal se define como $-\frac{d}{d\Omega}\phi(\Omega)$, con $\phi(\Omega) = \text{Arg}(H(\Omega))$ la fase del filtro:

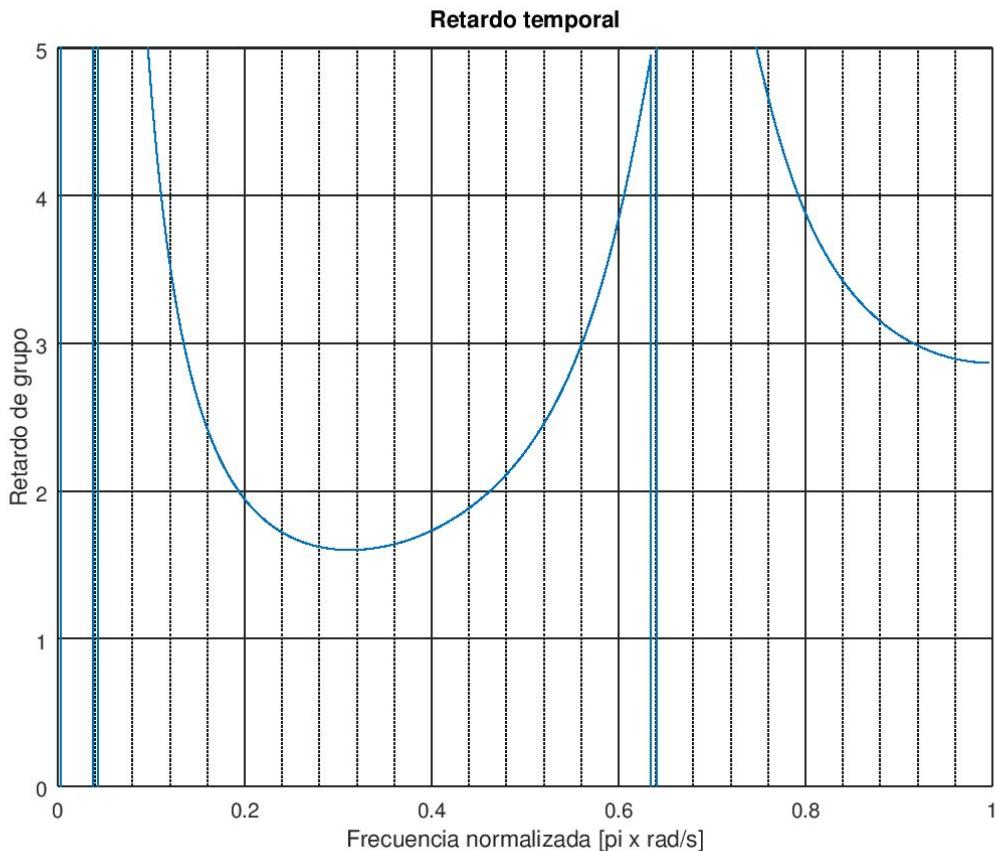


Figura 20. Retardo temporal calculado a partir de la fase del filtro.

En la figura 20 se ve que para las frecuencias altas, la fase es de aproximadamente unas cuatro muestras, lo que se condice con lo observado en el punto 5-c, ya que para estimar ese valor se habían comparado las posiciones de los picos.

NOTA IMPORTANTE: En el gráfico se ve una variación que parece muy grande, pero en realidad el eje de ordenadas está escalado para mostrar sólo un rango de 5 muestras, y casi todos los valores del retardo de grupo caen en ese rango.

Ejercicio 7

Implementar un filtrado IIR ida y vuelta para anular la fase del filtro (puede utilizar la función `filtfilt` de Matlab). Justificar teóricamente el funcionamiento de este tipo de filtrado y cuál resulta su ventaja. Filtrar nuevamente la señal FPG y comparar el resultado con lo obtenido en el punto anterior, particularmente en la forma de la señal y su retardo.

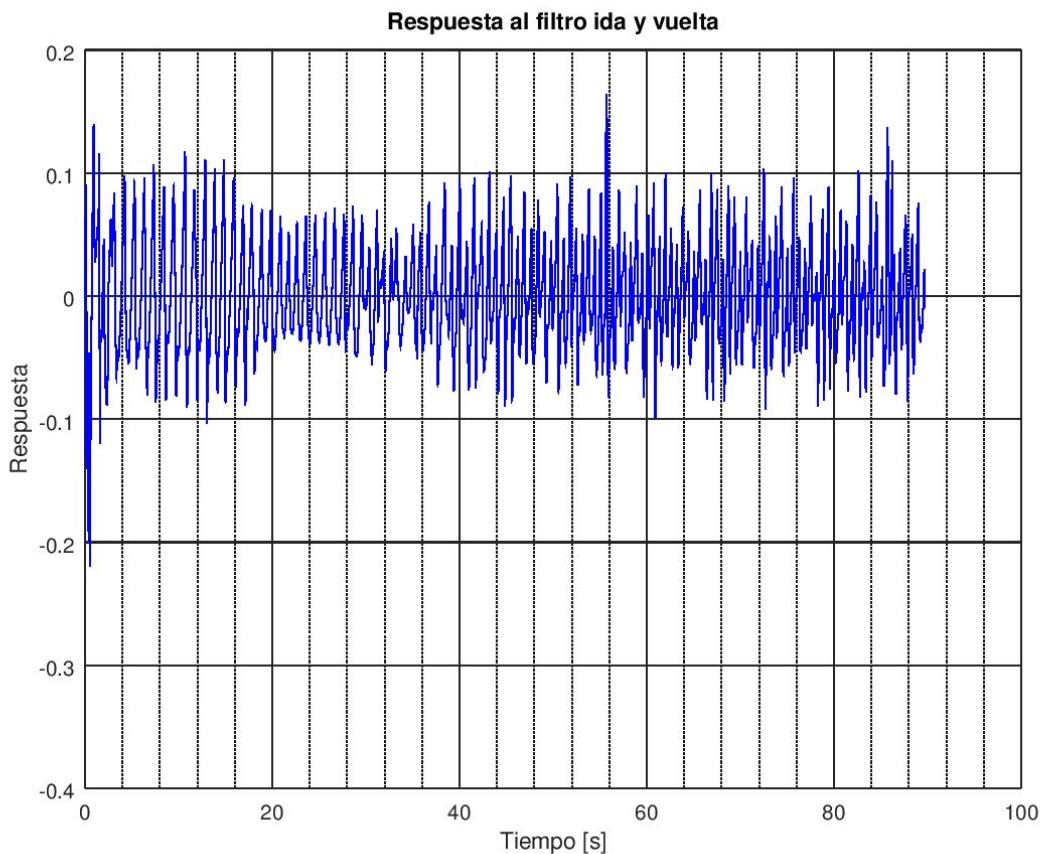


Figura 21. Respuesta al filtrado IIR.

A diferencia de la señal filtrada con el Butterworth, esta señal conserva la forma de la señal original, su respuesta en frecuencia es real, y por ende su fase es nula. Que la fase sea igual a cero también hace que la señal de salida no tenga retardos.

La idea detrás de estos filtros es usar la paraconjugación sobre la transferencia del mismo.

La paraconjugación es:

$$\widehat{H}(z) = H^* \left(\frac{1}{z^*} \right), \text{ ROC } \left\{ \widehat{H}(z) \right\} = \frac{1}{\text{ROC} \{ H(z) \}}$$

Si $H(z)$ es racional:

$$H(z) = \left(\frac{b_0}{a_0}\right) \frac{\prod_{k=1}^M (1 - c_k z^{-1})}{\prod_{k=1}^N (1 - p_k z^{-1})}$$

y entonces:

$$\hat{H}(z) = \left(\frac{b_0}{a_0}\right) \frac{\prod_{k=1}^M (1 - c_k^* z)}{\prod_{k=1}^N (1 - p_k^* z)}$$

Sea G tal que:

$$G(z) = H(z)\hat{H}(z) = H(z)H^*\left(\frac{1}{z^*}\right), \text{ ROC}\{G(z)\} \supseteq \text{ROC}\{\hat{H}(z)\} \cap \text{ROC}\{H(z)\}$$

Entonces:

$$G(z) = \left(\frac{b_0}{a_0}\right)^2 \frac{\prod_{k=1}^M (1 - c_k z^{-1})(1 - c_k^* z)}{\prod_{k=1}^N (1 - p_k z^{-1})(1 - p_k^* z)}$$

Si la ROC de G incluye la circunferencia de radio 1, lo cual se da cuando H es estable, tenemos que sobre dicha circunferencia:

$$G(z)|_{e^{j\omega}} = |H(e^{j\omega})|^2$$

Esta transferencia es real, por lo que su fase es nula.

Ejercicio 8

Realizar un espectrograma de la señal antes y después de filtrar, mediante la función `spectrogram` de Matlab (sugerencia: utilice la función `caxis` para saturar los colores del espectrograma y lograr una mejor visualización). Justificar la longitud de ventana elegida y comente acerca del resultado obtenido, relacionándolo con los puntos 2 y 3. Calcule la resolución en frecuencia de la ventana mediante DFT en Matlab. ¿Cómo haría para obtener mejor resolución en frecuencia y qué se pierde con esto?

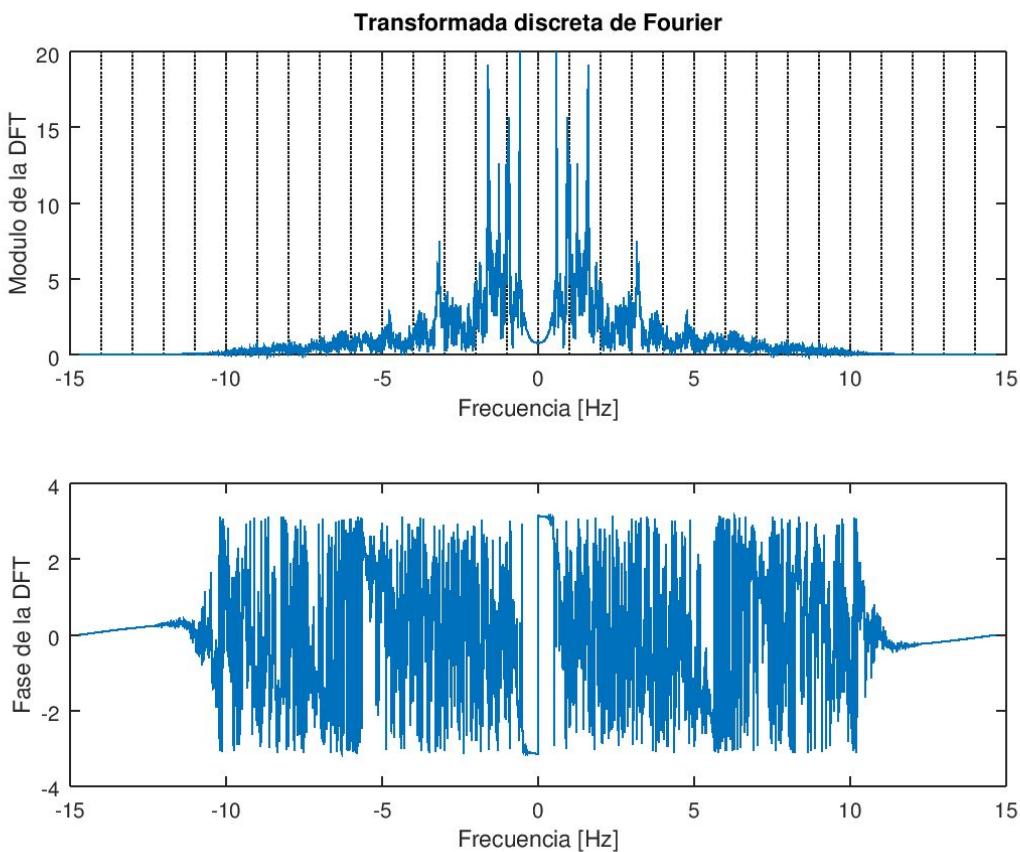


Figura 22. Transformada discreta de Fourier de la señal filtrada.

En la DFT se pueden observar picos separados por unos 0.4Hz, que queremos que en el espectrograma se vean en un color cálido, separados por al menos un color frío en el medio. Por ende, se deberá usar una ventana mayor a $2/0.4\text{Hz} = 5\text{s}$. Se usará entonces una ventana de 6s para los espectrogramas.

Para obtener mejor resolución en frecuencia debería usar una ventana de tiempo más larga, pero con eso se pierde localización temporal: se puede decir que si

usamos una ventana del largo de la señal, tenemos los mismos datos que en la DFT, pero representados de otra manera.

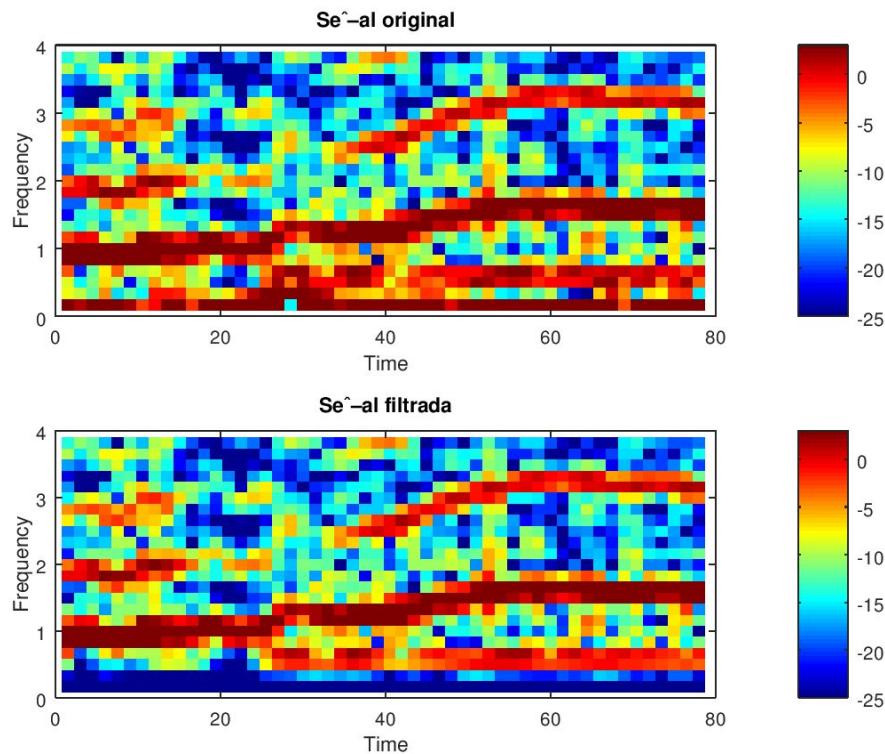


Figura 23. Espectrogramas de la señal original y de la filtrada usando una ventana de Hamming de 6 segundos.

Se puede ver que la evolución de la componente de frecuencia que ronda 1 Hz se corresponde con lo observado en el segundo ejercicio. Entonces, con estos datos, podemos decir que la visualización de dicho análisis tenía sentido.

La diferencia entre la señal original y la filtrada se ve en las frecuencias bajas, que en la filtrada tienen un color mucho más frío.

También se utilizó una ventana de Hanning del mismo largo, pero la única diferencia entre los resultados se veía en el contraste de las curvas que se forman: la ventana de Hamming genera curvas de mayor contraste, mientras que las curvas generadas por la de Hanning son más gruesas.

Ejercicio 9

Identificar en el espectrograma la zona donde el pulso se acelera. Observar con detenimiento los componentes de frecuencia que posee la señal y justificar el origen de cada uno (para esto último, necesitará hacer uso de la señal audio_det.mat para explicar todos los componentes observados).

Al igual que en el análisis preliminar, en el espectrograma se ve mucho ruido al principio, y luego una frecuencia creciente que arranca en 1Hz aproximadamente a los 20 segundos, y después de los 50 segundos se estanca en unos 1.6Hz.

Más arriba se puede observar otra componente que arranca en un poco menos de 2Hz y se estanca en 3.2Hz, por lo que se puede explicar como un armónico de la frecuencia cardíaca.

Para explicar la frecuencia más baja que se observa en el espectrograma, la información que analizamos hasta ahora no nos alcanza:

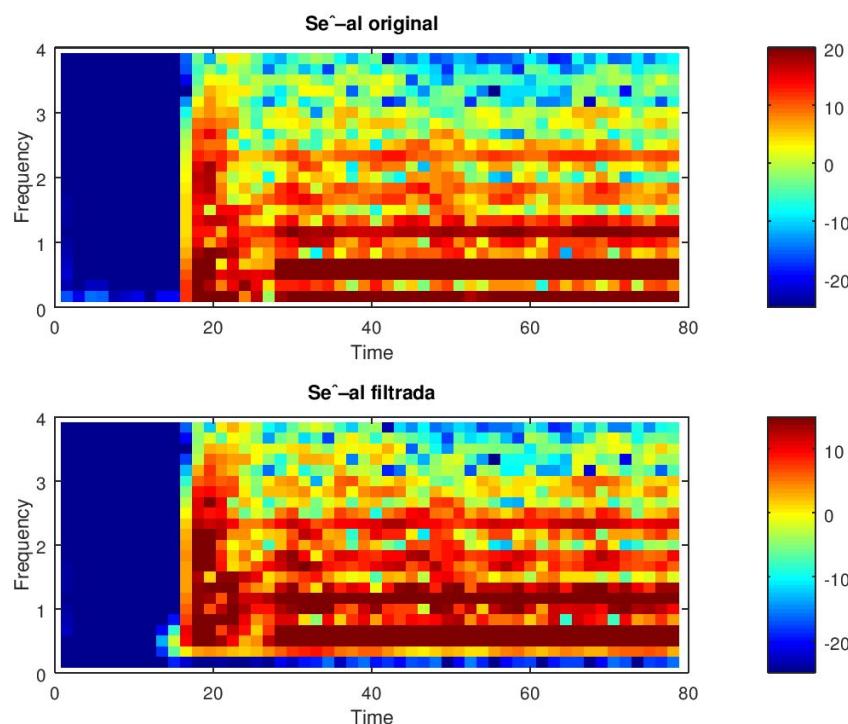


Figura 24. Espectrogramas de la señal de audio y su salida del mismo filtro que se usa para la señal de video.

En la figura 24 podemos observar las componentes de la señal de audio, que comienzan en la misma zona en la que se empieza a acelerar la señal de video. Esto explica la componente en aproximadamente 0.6Hz que faltaba explicar en el espectrograma de la figura 23.

Ejercicio 10

Realizar un detector automático de latidos. El mismo debe tomar como entrada la señal FPG y producir como salida un vector de tiempos, donde cada tiempo corresponde a la detección de un latido en la señal. Para esto, se sugiere implementar los siguientes pasos:

- a. Filtrado pasa-banda de la señal, utilizando el filtrado del ejercicio 7.
- b. Filtro de derivada, implementado con un filtro FIR $h(n)=[-2 -1 0 1 2]$.
- c. Normalización con energía instantánea: primero calcular la energía instantánea de la señal mediante un filtro MA 1 de la señal del punto 10a elevada al cuadrado; luego dividir la señal del punto b por el vector obtenido. Esto tiene como objeto reducir el impacto de la presión sanguínea sobre el nivel de señal.
- d. Sobre-muestreo en un factor 4 para obtener mayor resolución temporal: implemente el sobre-muestreo utilizando la función `upsample` y diseñe un filtro interpolador FIR utilizando la herramienta `fdatool` de Matlab. Grafique: respuesta en frecuencia del filtro en módulo y fase, y señal original y sobre-muestreada en superposición.

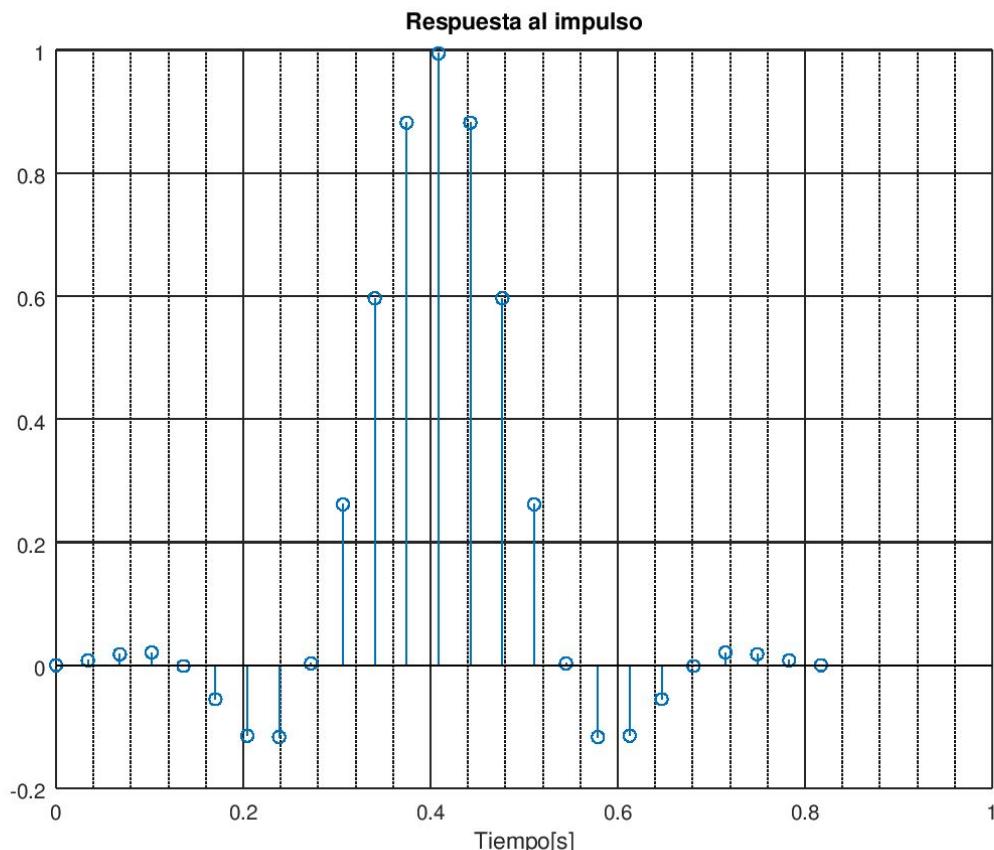


Figura 25. Respuesta al impulso del filtro a utilizar para la interpolación. Una sinc de 25 puntos.

El filtro interpolador cuya respuesta al impulso se puede observar en la figura 25 fué diseñado como un pasa-bajos de orden 25 con una frecuencia de corte de $\pi/4$. Para obtenerlo se utilizó la función fir1 de Octave, de la siguiente manera:

```
hInterpolator = fir1(24, 1/4, 'low')/(1/4);
```

El factor de escala sirve para corregir el escalamiento que resulta de llamar a la función.

En la figura 26 se puede observar la respuesta en frecuencia del filtro.

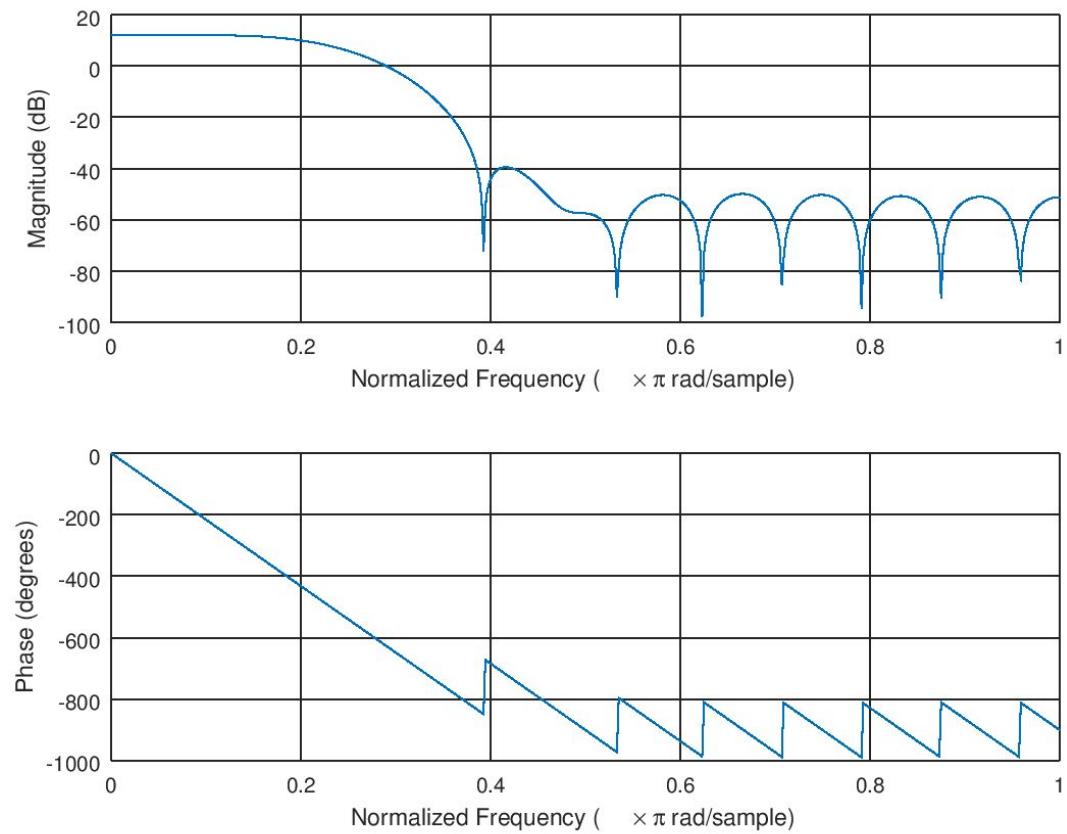


Figura 26. Respuesta en frecuencia del filtro interpolador.

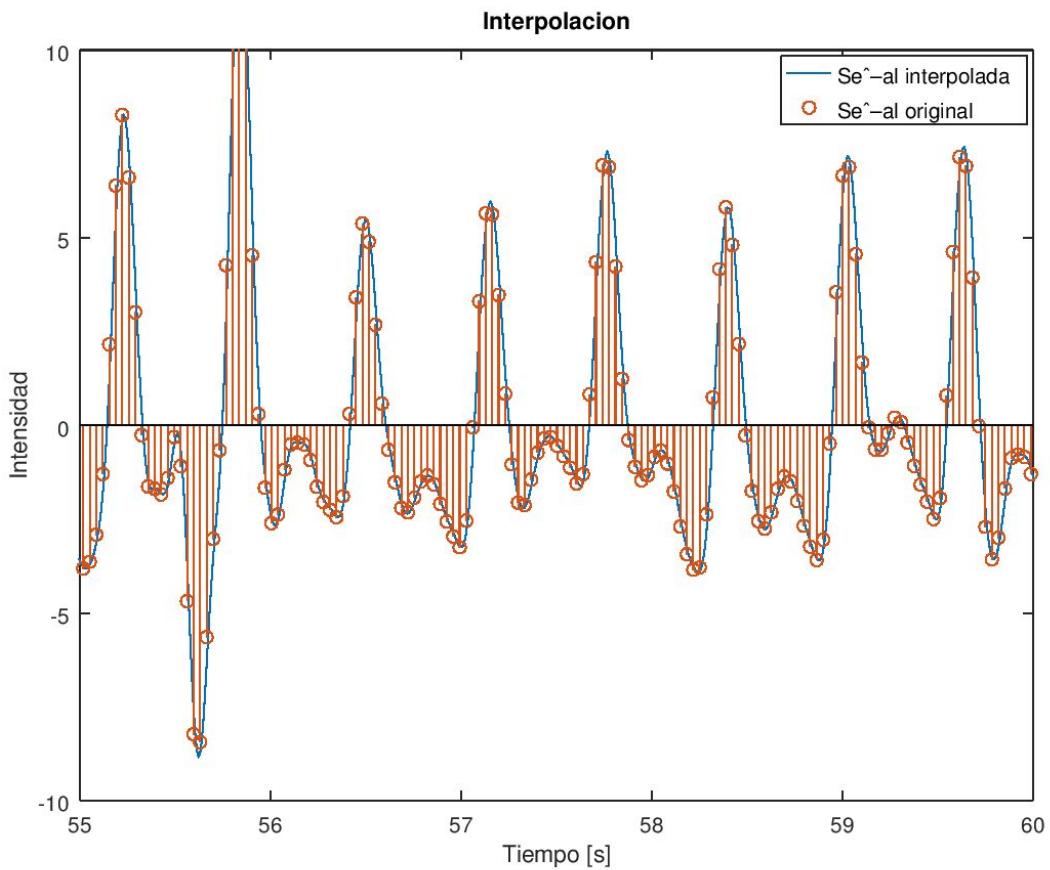


Figura 27. Visualización de la interpolación en un segmento de la señal.

Con el objetivo de corroborar el correcto funcionamiento del filtro interpolador, se muestra en la figura 27 la señal a interpolar y el resultado de la interpolación superpuestas.

- e. *Detector de picos mediante umbral (puede definir como umbral un valor arbitrario).*
- f. *Gráfico en superposición de la señal con las marcas de los picos detectados.*

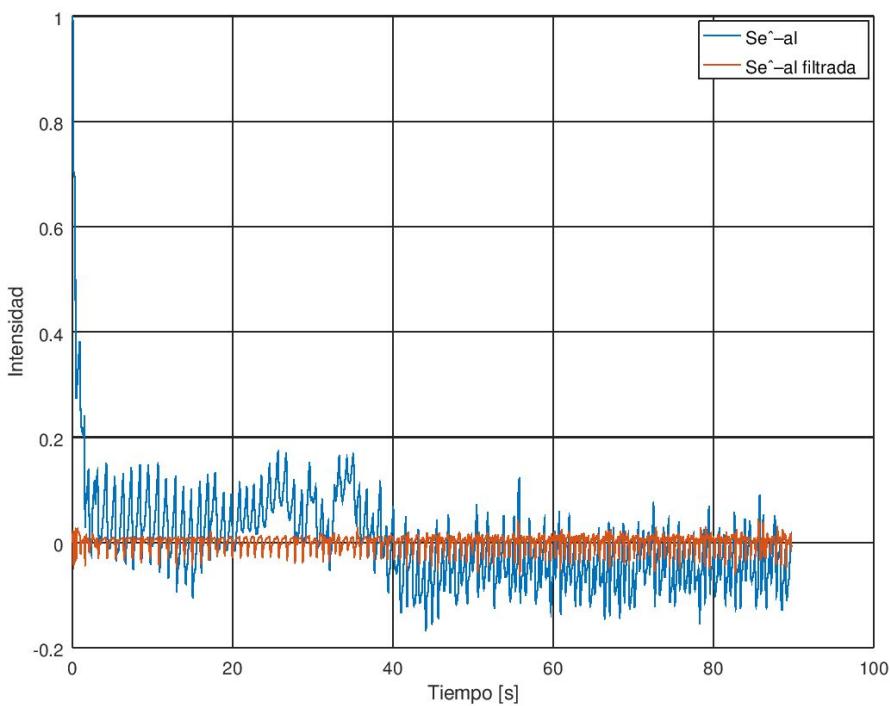


Figura 28. Señal original y filtrada en superposición.

Podemos observar en la figura 28 cómo queda la señal luego de ser filtrada, y en la 29 los picos marcados en la señal filtrada.

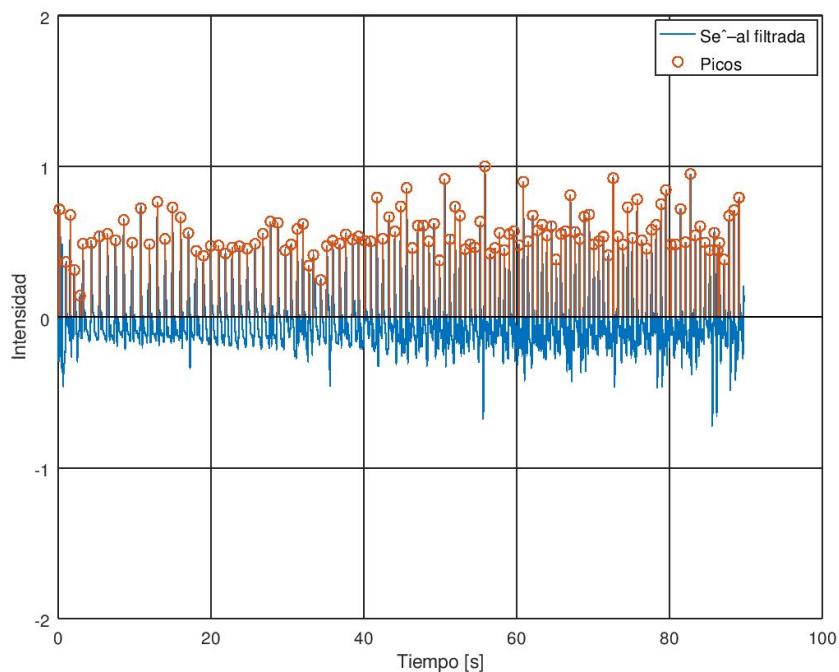


Figura 29. Señal filtrada en superposición con sus picos.

Para conseguir una mejor visualización de los picos marcados, se incluye en la figura 30 un segmento en el que se puede observar con mayor facilidad el funcionamiento del procedimiento.

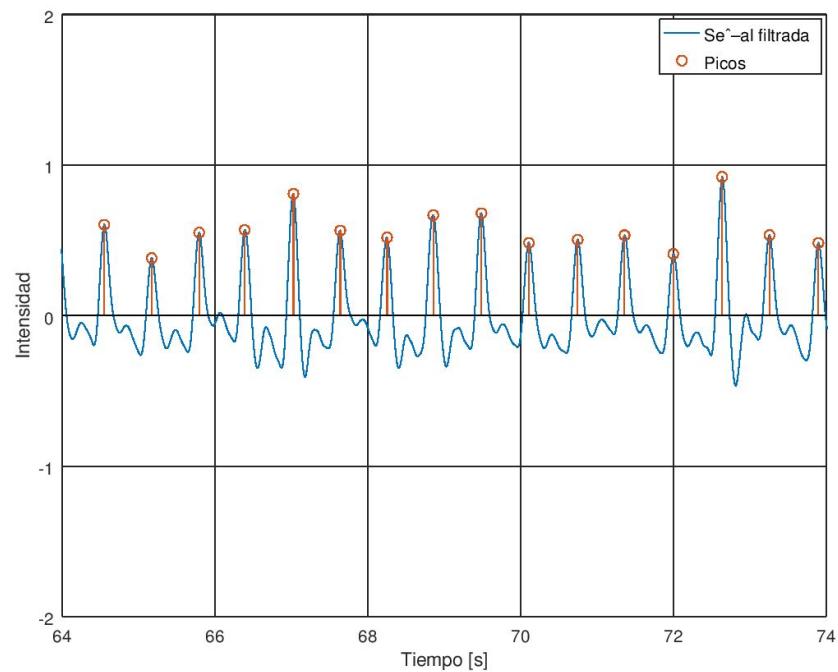


Figura 30. Un segmento de la señal con los picos marcados.

Ejercicio 11

En base a los resultados del punto anterior, calcule y grafique el intervalo temporal instantáneo entre latidos (IBI: inter-breath interval) y los LPM instantáneos.

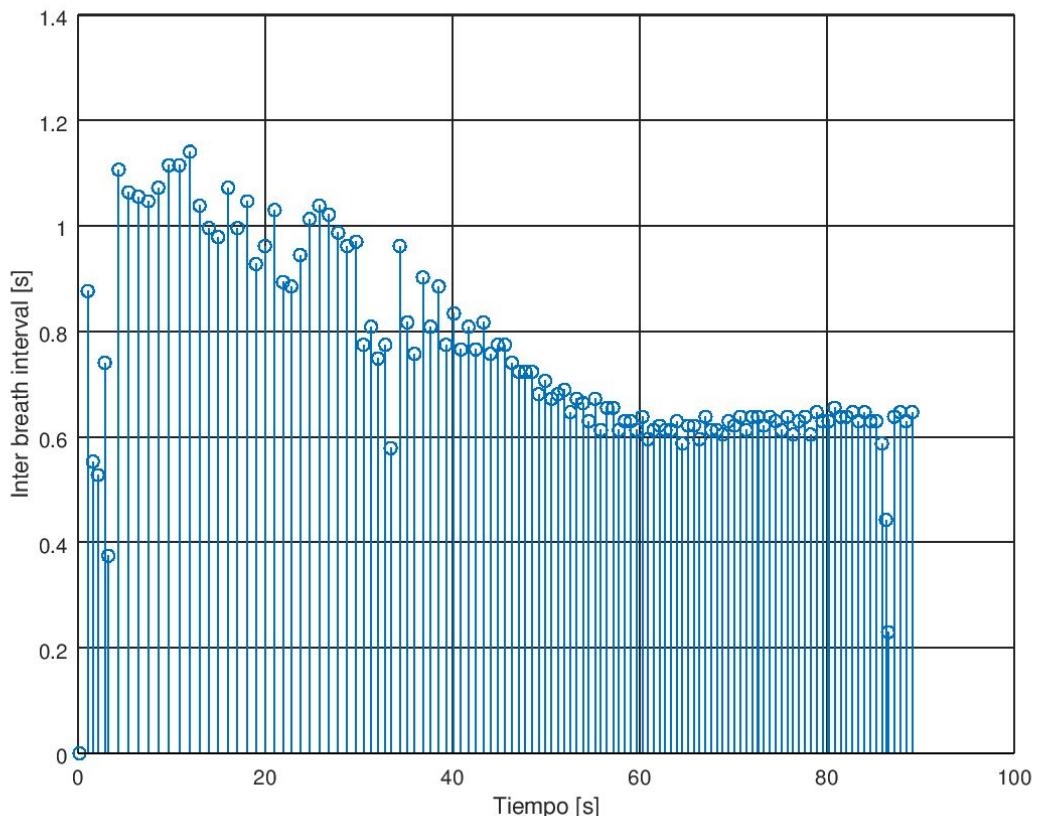


Figura 31. Evolución del intervalo entre latidos en función del tiempo.

En la figura 31 vemos una evolución similar a la observada en el análisis preliminar del ejercicio 2, pero más suavizada debido a los filtros que eliminaron frecuencias que no se querían tener en cuenta.

Ejercicio 12

Opcional. Mejorar el detector de latidos aplicando las reglas de [5]:

- a. *Establecer como regla que, si dos latidos se detectaron con una separación temporal menor a 200ms, sobrevive sólo aquel que corresponda al pico de la señal mayor entre ambos.*
- b. *Establecer como regla que si el IBI instantáneo aumenta repentinamente en al menos 1.5 veces entre muestras consecutivas, puede haberse perdido la detección de un latido. Cuando este sea el caso, realice una nueva detección de picos dentro del intervalo correspondiente, utilizando un umbral la mitad del nominal. Si de esta manera se halla un nuevo pico, distanciado al menos 360ms de la detección precedente, entonces clasificarlo como latido.*

Ambas mejoras fueron implementadas en la detección de picos del ejercicio 10.

Ejercicio 13

Opcional. Obtener su propia señal FPG mediante la cámara de su celular y aplicar los análisis y algoritmos desarrollados a esta señal. Para esto, realice una grabación tal como se explicó en la sección 1 y utilice los scripts auxiliares de Matlab para extraer los archivos de intensidades y audio del video. Nota: para un mejor resultado en la señal registrada, utilizar el dedo índice aplicando una leve presión sobre la lente, evitando resultar excesivamente suave o fuerte.

Usando el script provisto

Tomando la señal con mi celular, la DFT quedó más marcada en una frecuencia de casi 1.2Hz:

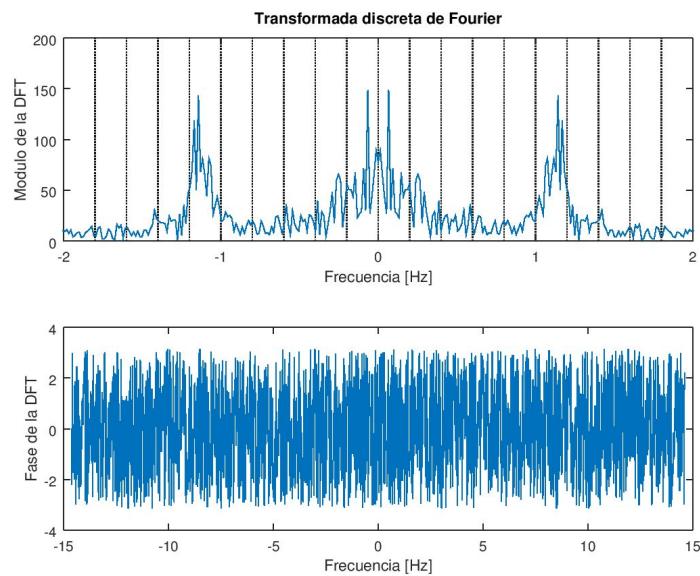


Figura 32. DFT de la señal tomada con el teléfono celular sin filtrar.

Utilizando los mismos filtros y algoritmos que antes, la señal filtrada se comporta mejor que en la señal del resto del desarrollo. Para la grabación de la señal se intentó reducir al máximo las fuentes de ruido (luces apagadas, ventanas cerradas, etcétera), y la respiración se intentó mantener al mismo ritmo.

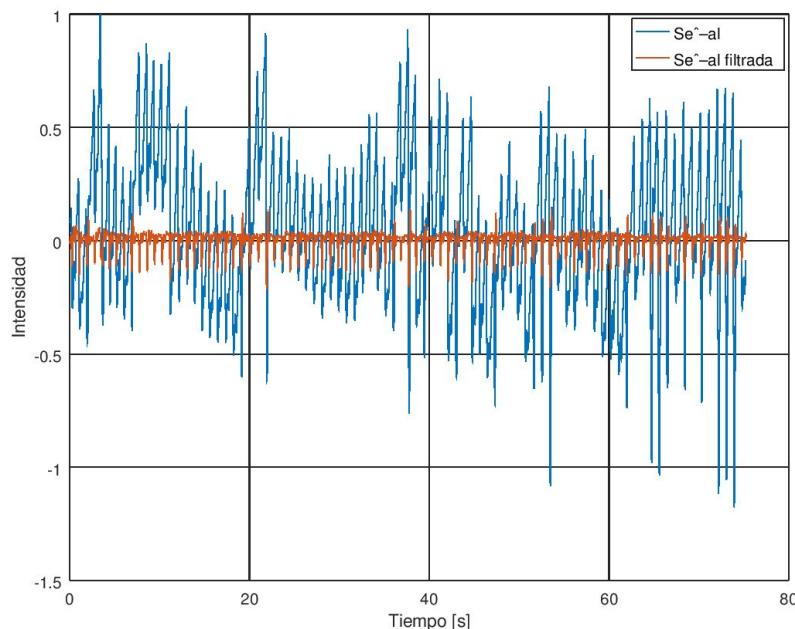


Figura 33. Señal tomada con el celular, antes y después del filtrado.

Vemos también que los IBI son más parejos y cercanos a 0.8s, dado que no se mantuvo una respiración tranquila durante la grabación del video:

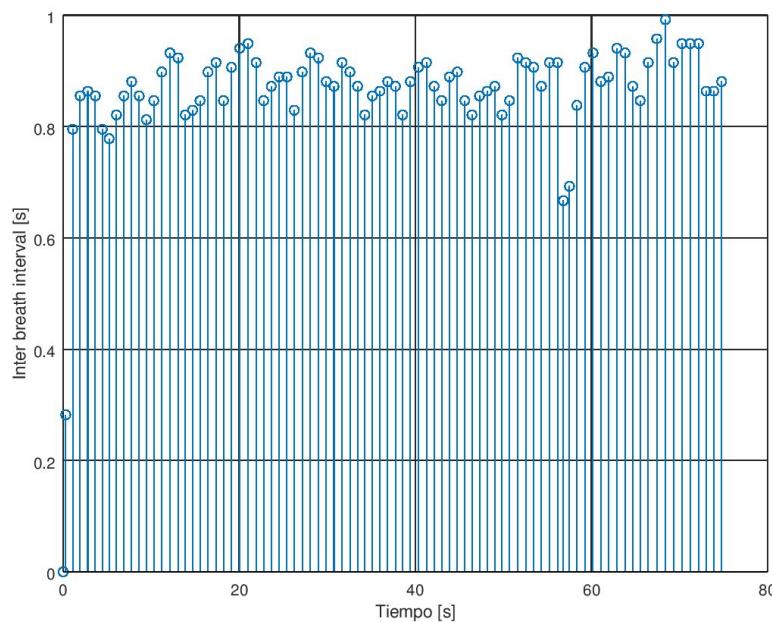


Figura 34. Intervalos entre latidos obtenidos con el video del celular.

En la figura 35 vemos el funcionamiento del algoritmo de detección de picos en la señal tomada con el celular.

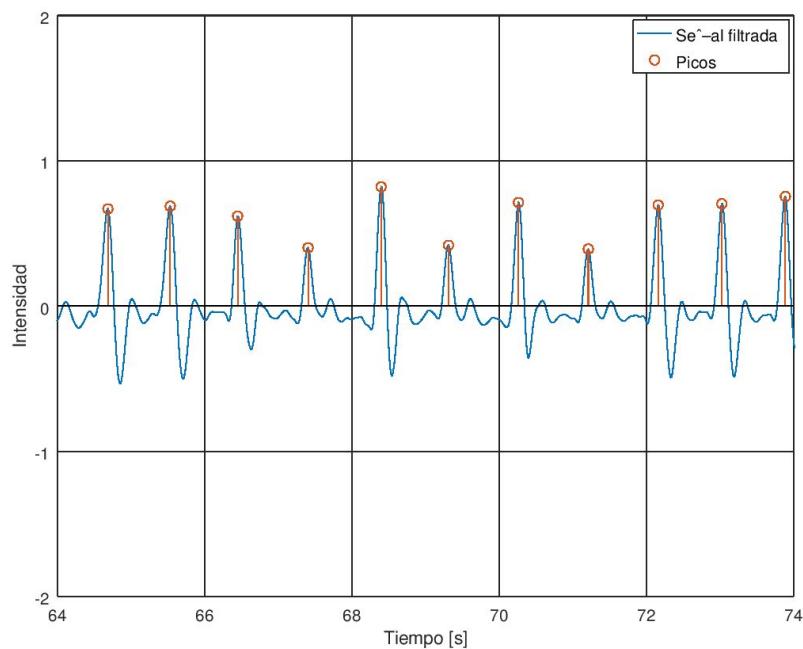


Figura 35. Picos marcados en la señal en un intervalo.

Usando el script modificado

También se analizó la señal modificando el script de transformación de video a archivo .mat. La modificación consistió en reemplazar la media (promedio) como estimador del color de cada frame por la mediana.

Los cambios en los resultados para el video filmado no fueron perceptibles, pero en caso de que haya píxeles que se apartan mucho de la media en algunos instantes, la media variaría mucho más que la mediana, que es un estimador más estable. Esto se puede dar si alguien aparta el dedo de la cámara y entra luz por los bordes, sin interferir con el dedo. A continuación, se muestran los resultados luego de aplicar esta modificación:

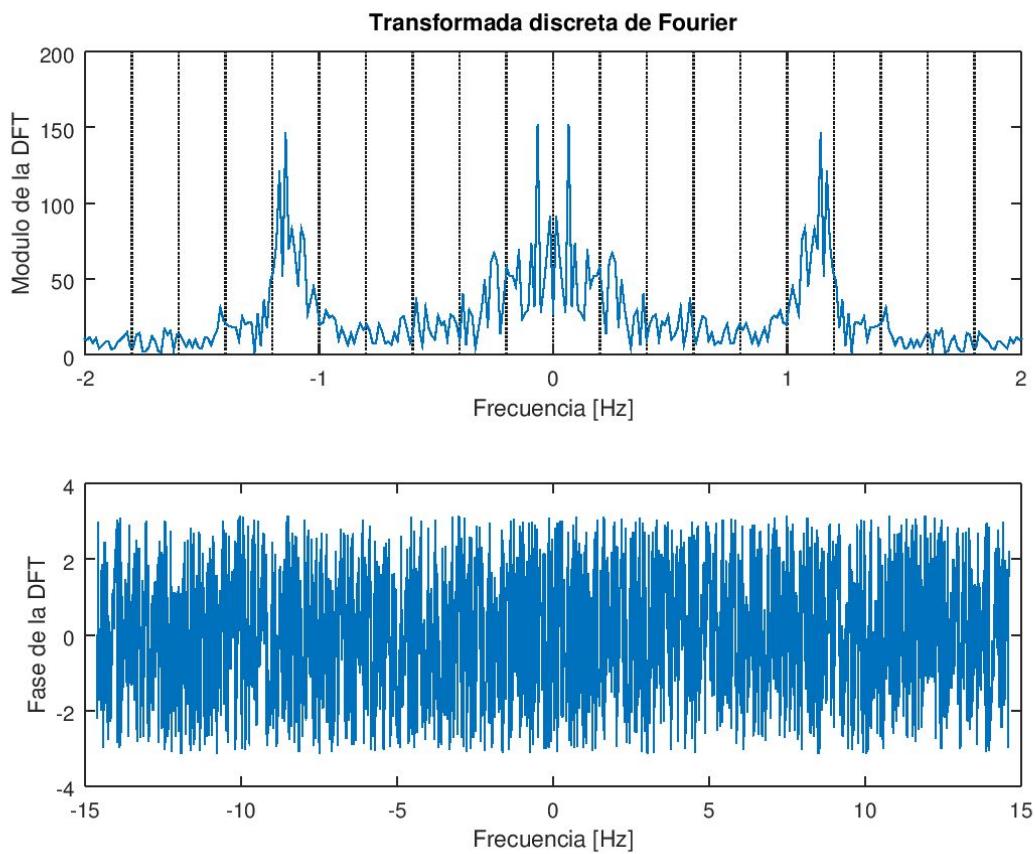


Figura 36. DFT de la señal tomada con el teléfono celular sin filtrar y usando la mediana como estimador.

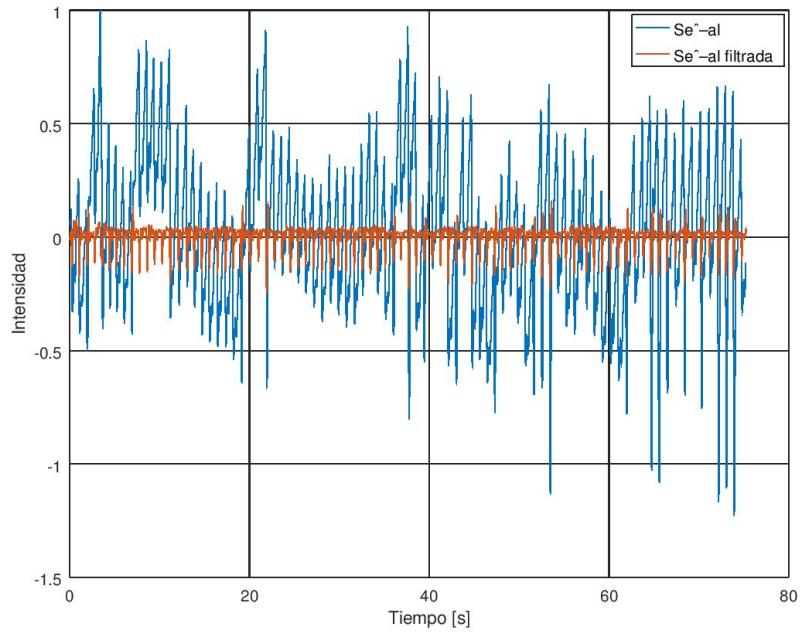


Figura 37. Señal tomada con el celular y usando mediana como estimador del color del frame, antes y después del filtrado.

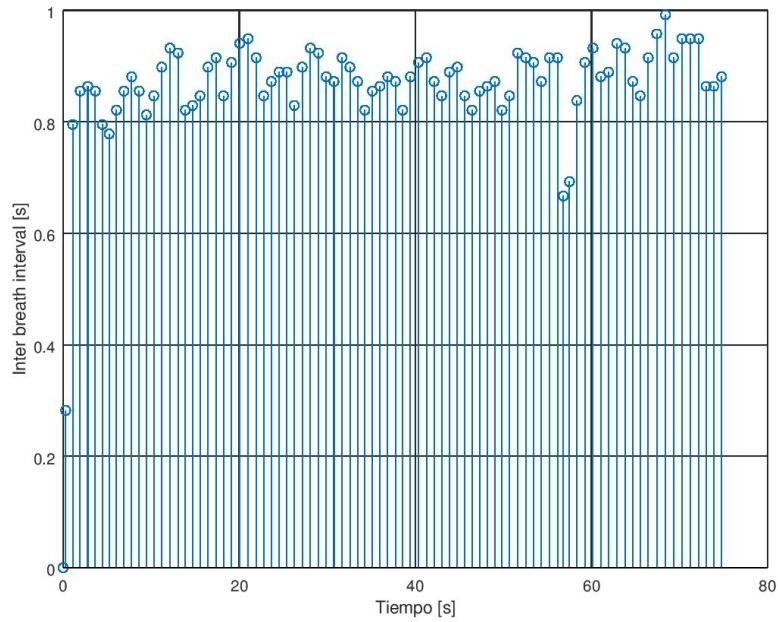


Figura 38. Intervalos entre latidos obtenidos con el video del celular.

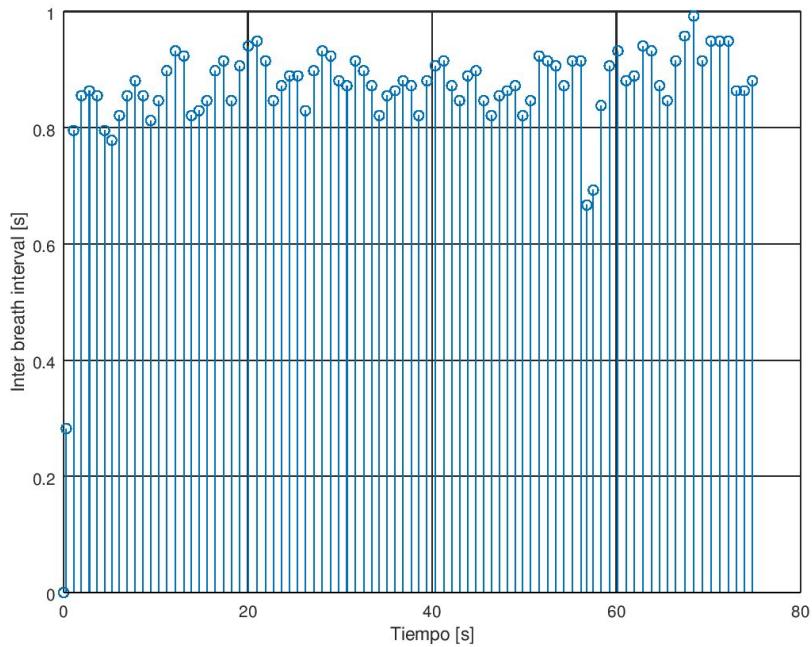


Figura 39. Picos marcados en la señal en un intervalo.

Conclusiones

- Un análisis a priori, como el que se hizo en el segundo ejercicio, ayudó a detectar errores rápidamente y a ir corroborando los resultados intermedios durante el desarrollo del resto de los ejercicios.
- El análisis en frecuencia de la señal es esencial para diseñar los filtros que se aplicarán sobre ella.
- El filtrado de fase cero es necesario para señales en las que importa la forma, como en los ECG. Tal vez para la detección de picos no hacía falta, pero si se quiere observar la forma de los latidos, la fase podría entorpecer algún diagnóstico médico.
- Podría ser muy útil para diagnósticos remotos que cualquier paciente con un smartphone pueda enviar un video de su dedo y que su médico pueda ver la forma de sus latidos.