



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 pixeles de la pantalla. Cada uno de los pixeles 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.

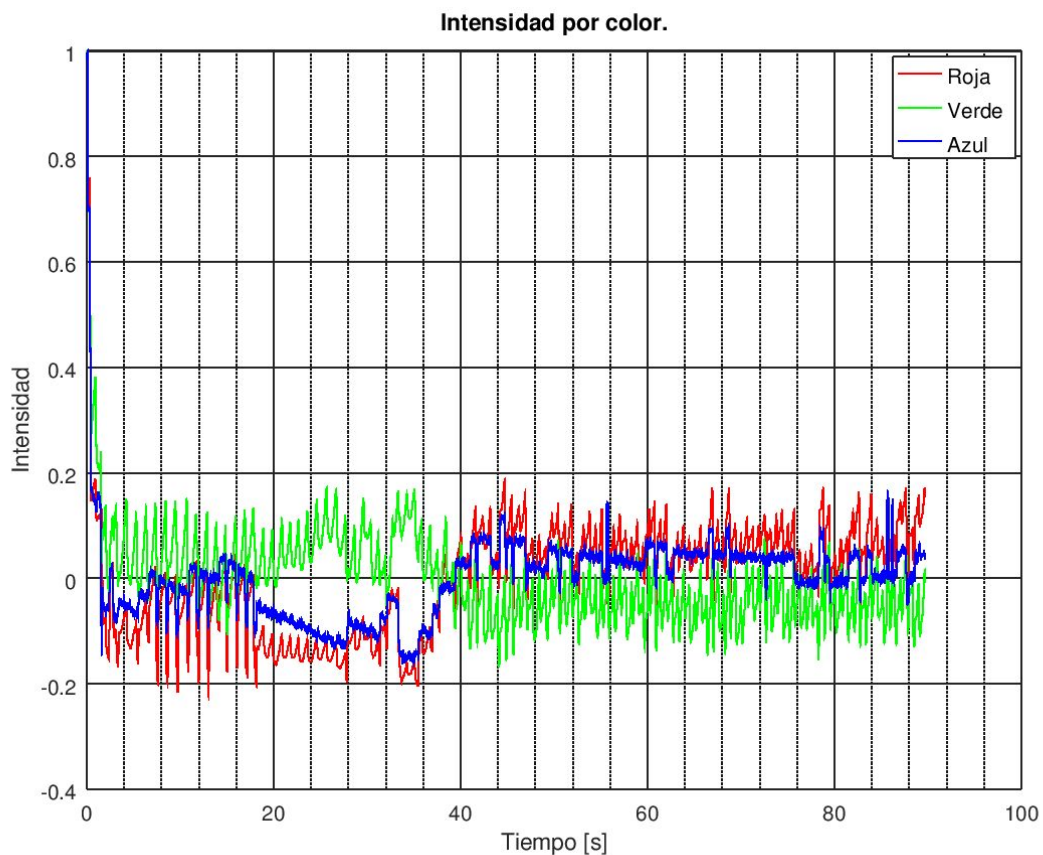


Figura 1. Intensidad normalizada de las señales por color.

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 roja, 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.

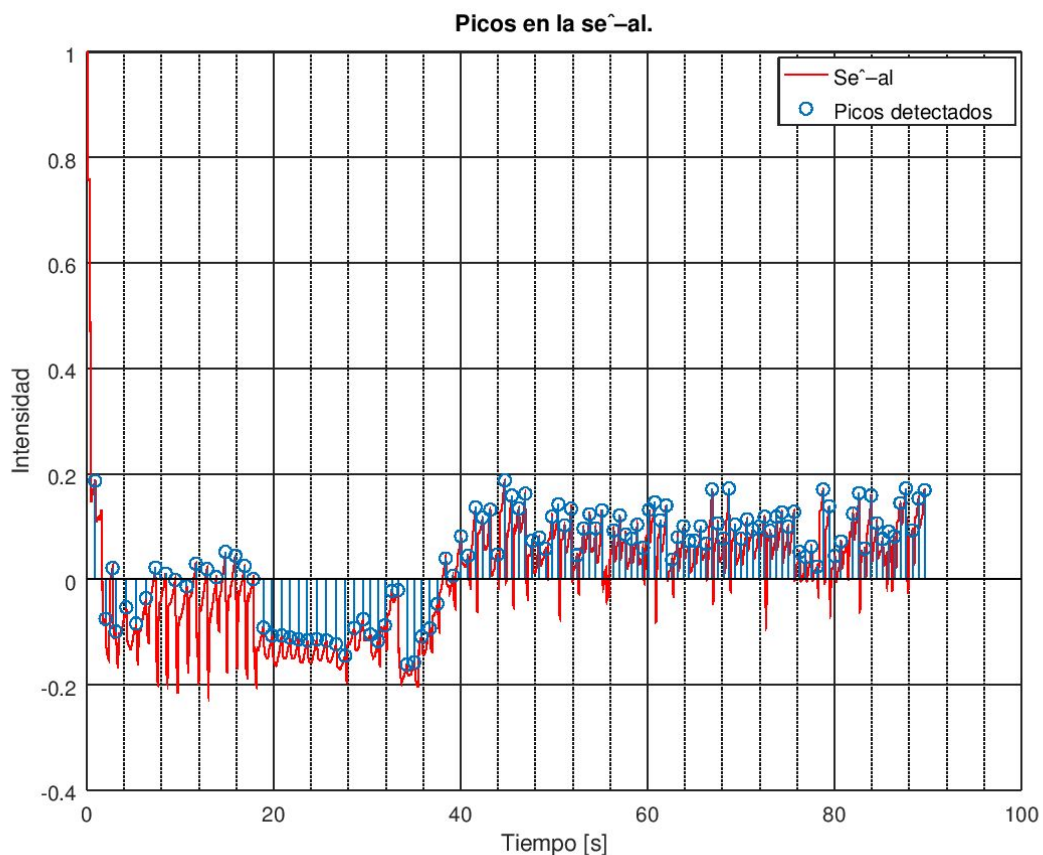


Figura 2. Picos de la señal sin ningún filtro.

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.

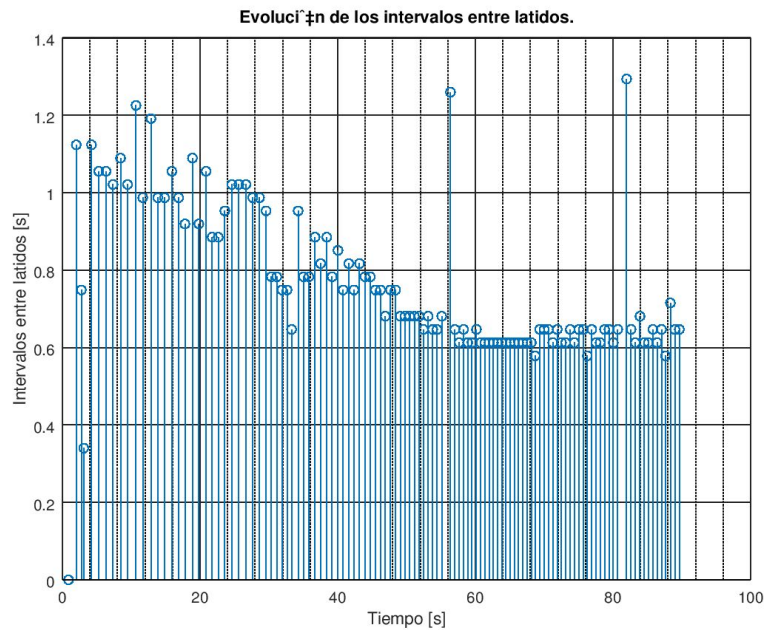


Figura 3. Intervalos entre picos de la señal sin ningún filtro.

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.

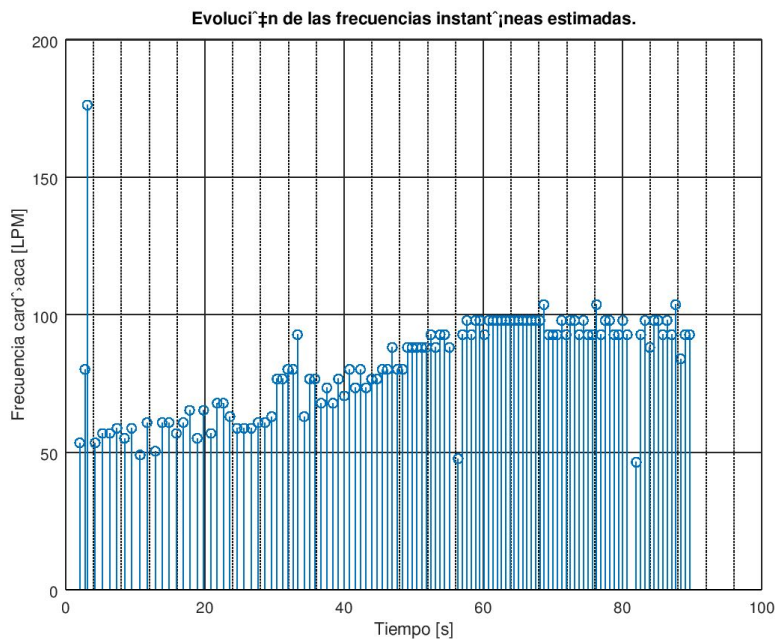


Figura 4. Frecuencia instantánea calculada a partir del gráfico anterior.

Ejercicio 3

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

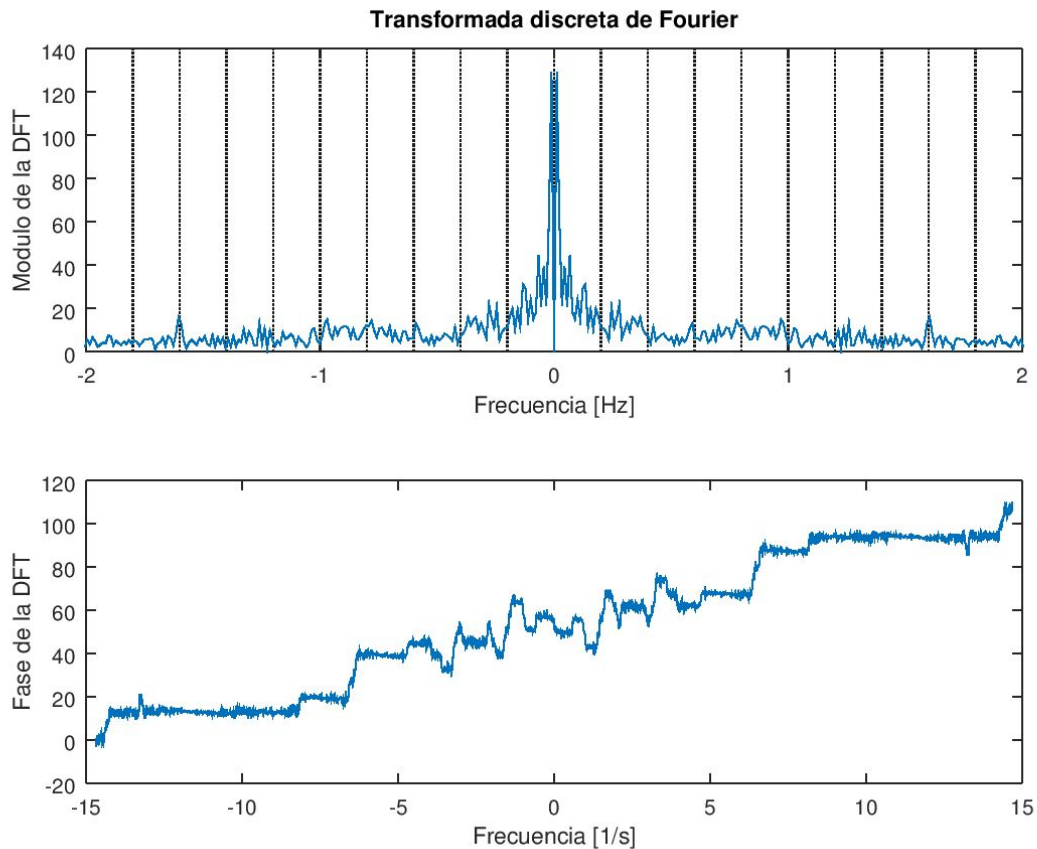


Figura 5. Transformada discreta de Fourier de la señal.

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 6.

Filtro de orden 2

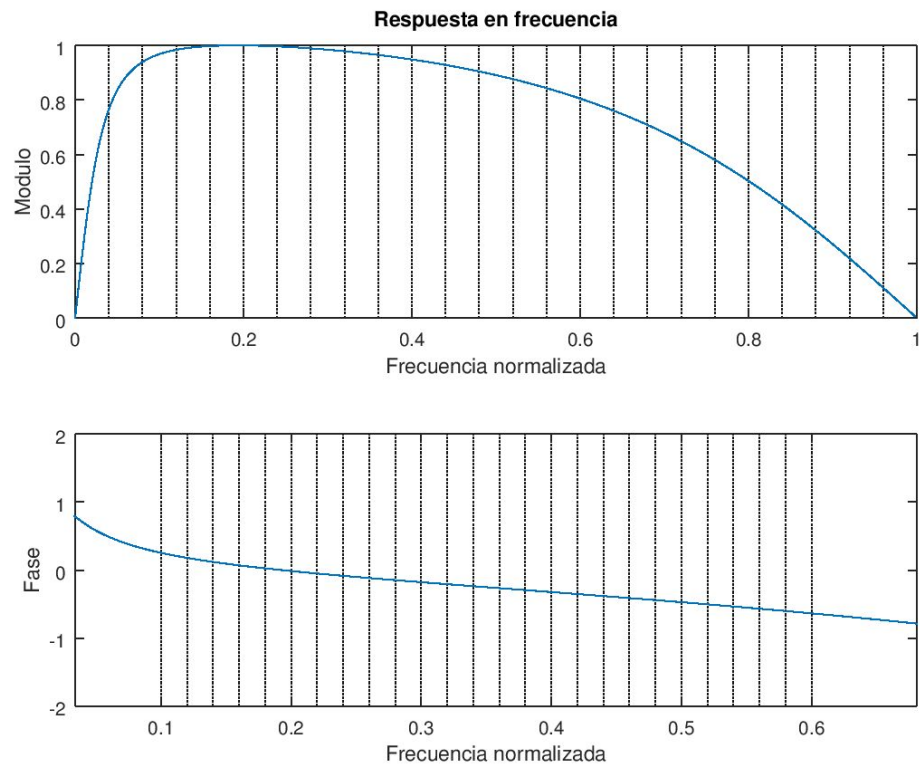


Figura 6. Respuesta en frecuencia del filtro de orden 2.

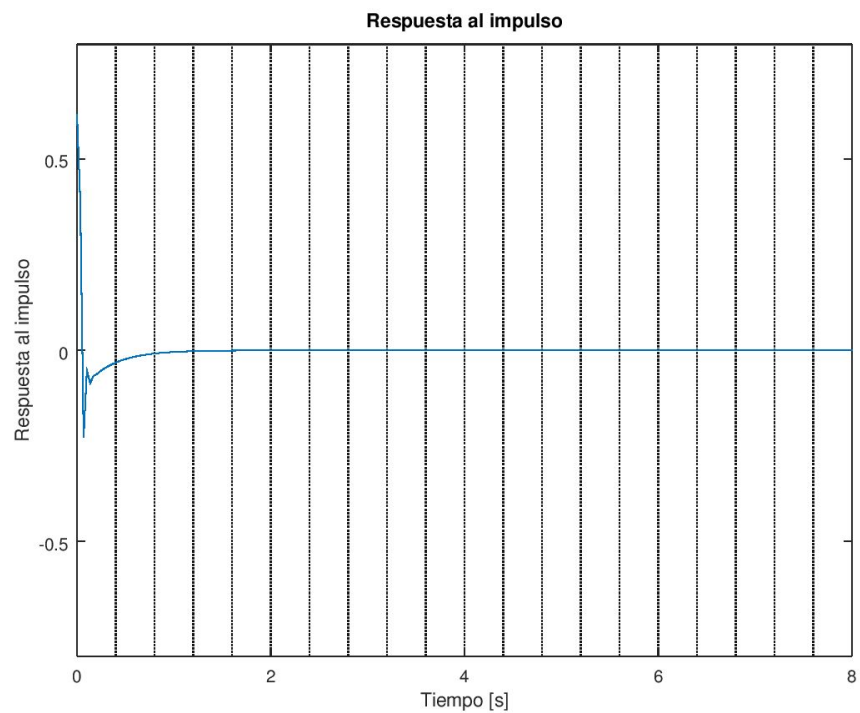


Figura 7. Respuesta al impulso del filtro de orden 2.

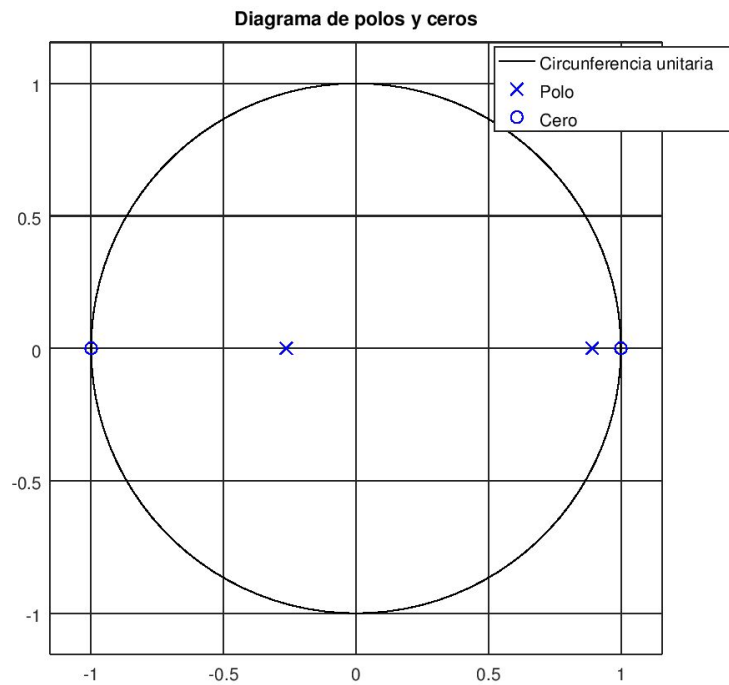


Figura 8. Diagrama de polos y ceros del filtro de orden 2.

Filtro de orden 4

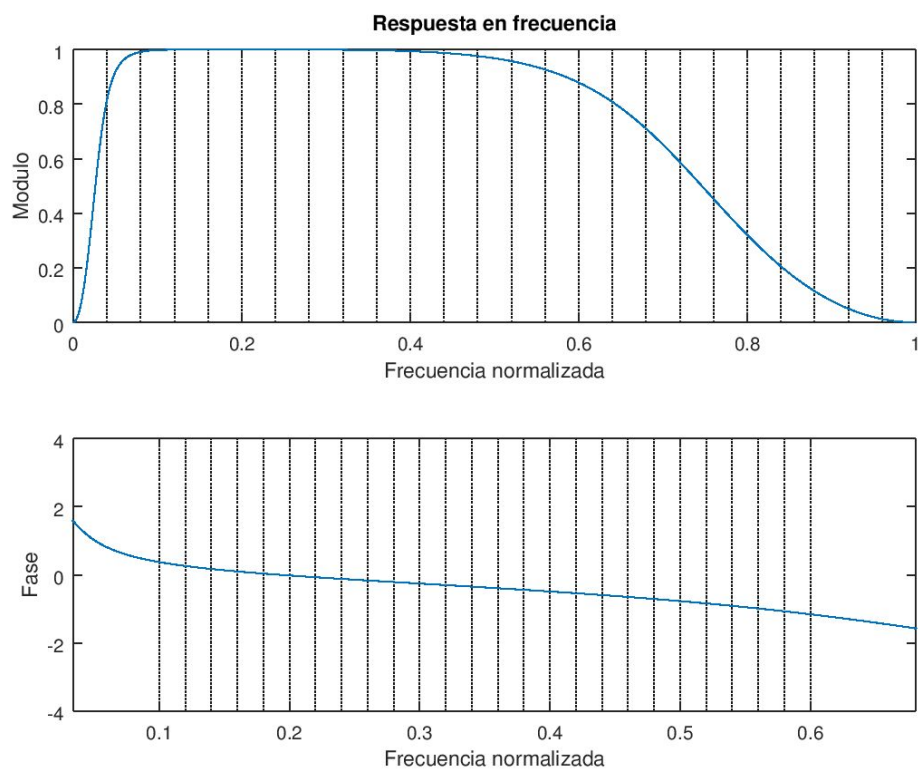


Figura 9. Respuesta en frecuencia del filtro de orden 4.

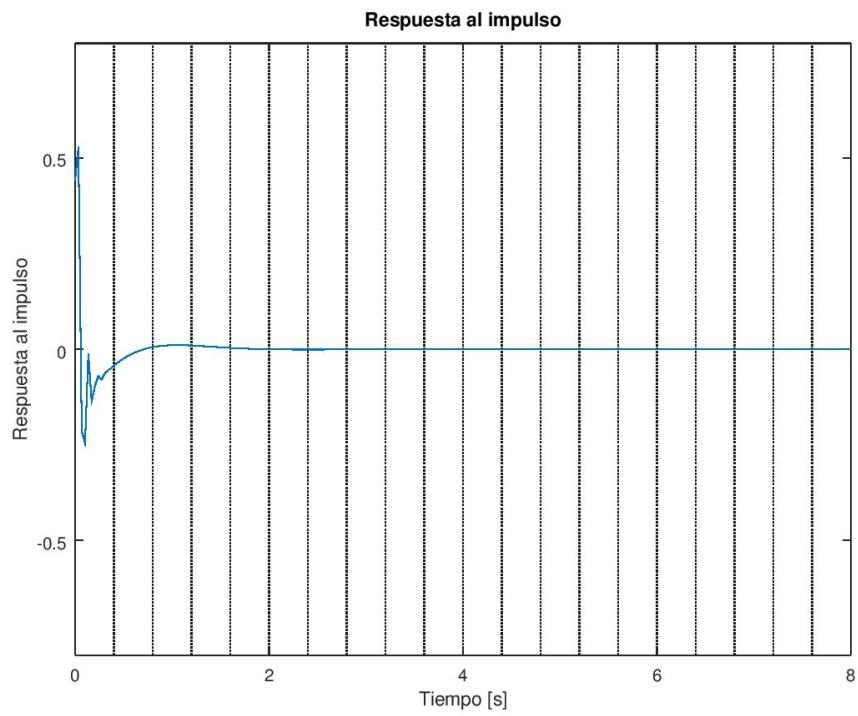


Figura 10. Respuesta al impulso del filtro de orden 4.

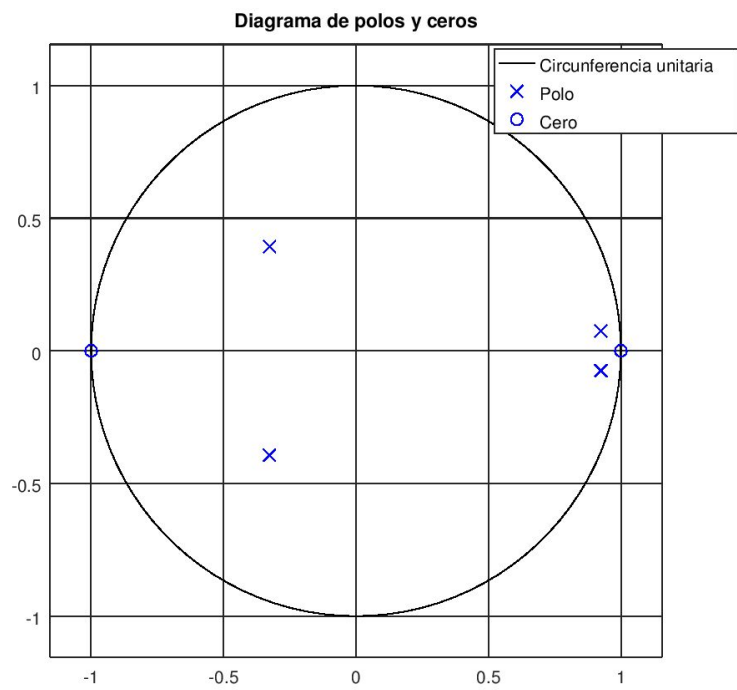


Figura 11. Diagrama de polos y ceros del filtro de orden 4.

Filtro de orden 6

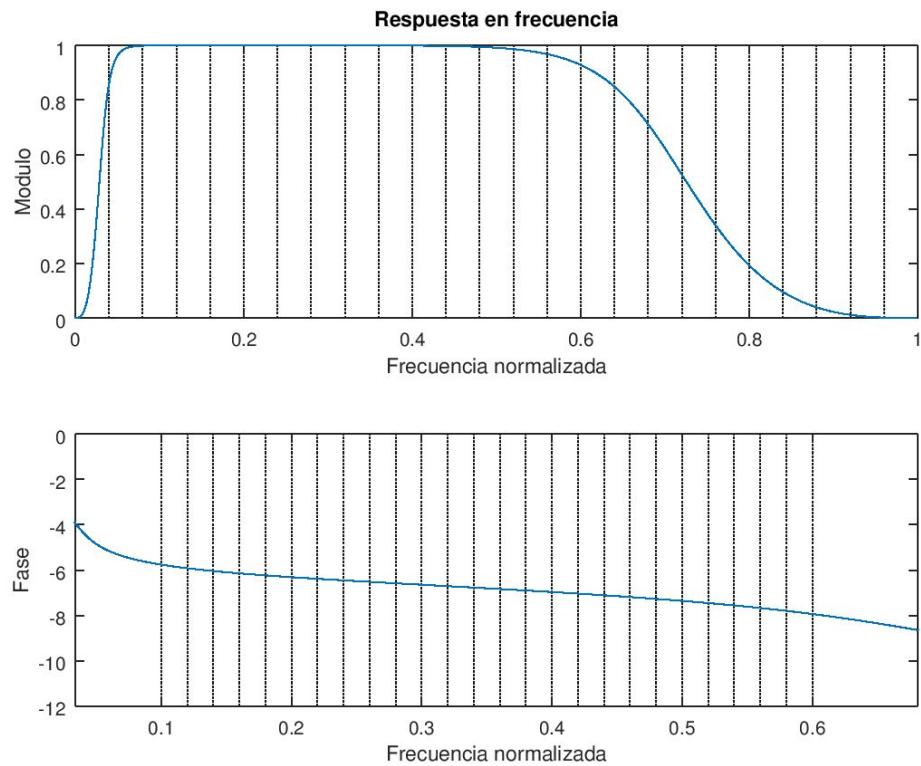


Figura 12. Respuesta en frecuencia del filtro de orden 6.

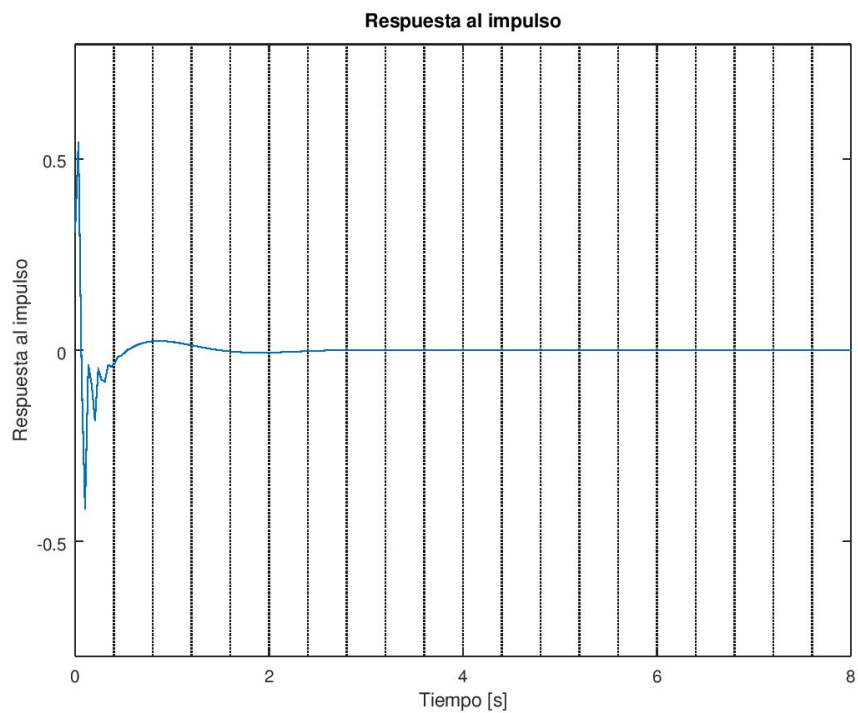


Figura 13. Respuesta al impulso del filtro de orden 6.

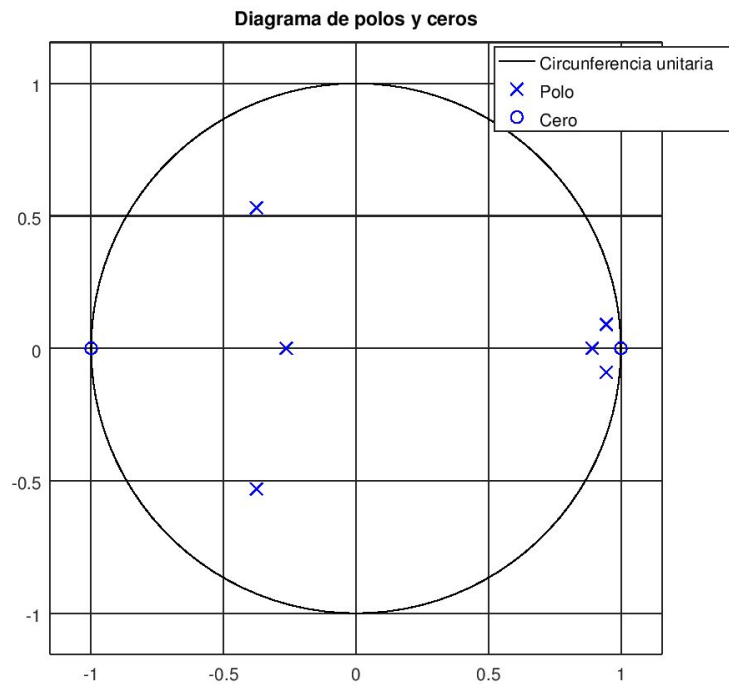


Figura 14. Diagrama de polos y ceros del filtro de orden 6.

Filtro de orden 10

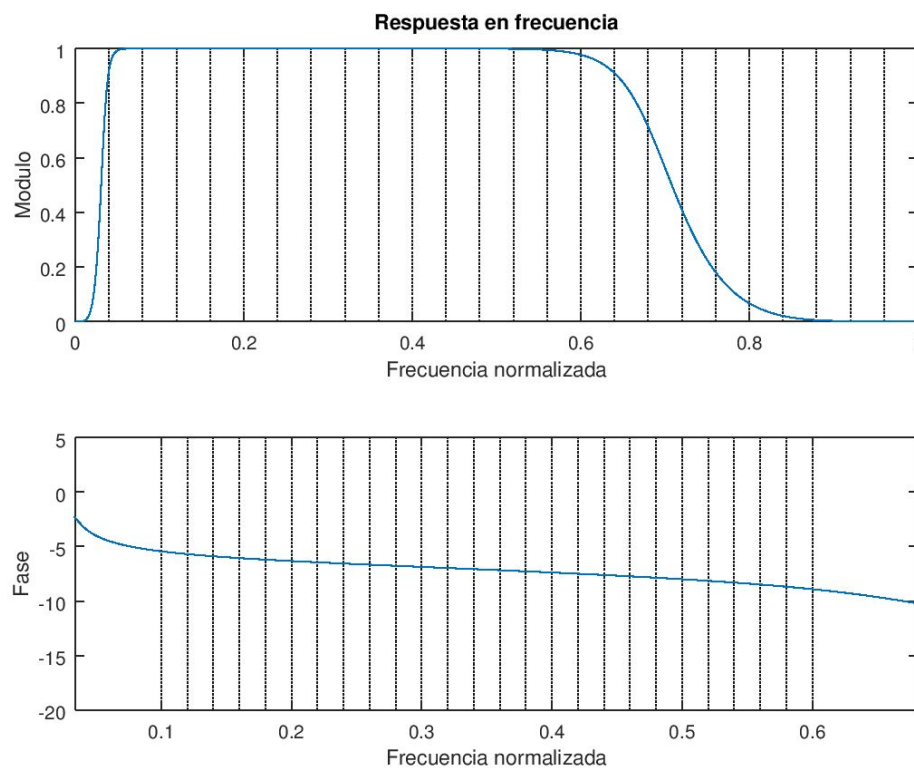


Figura 15. Respuesta en frecuencia del filtro de orden 10.

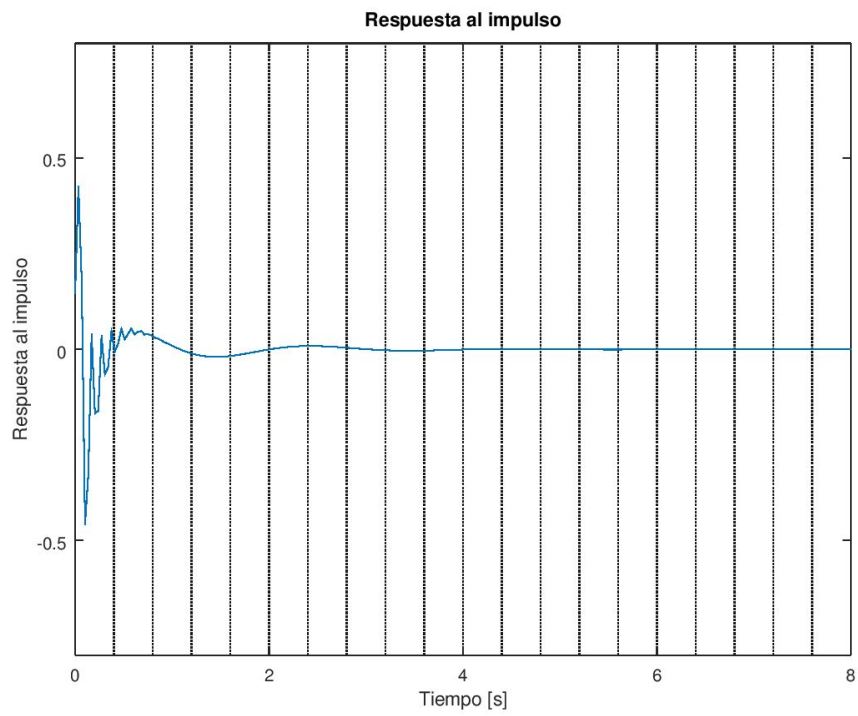


Figura 16. Respuesta al impulso del filtro de orden 10.

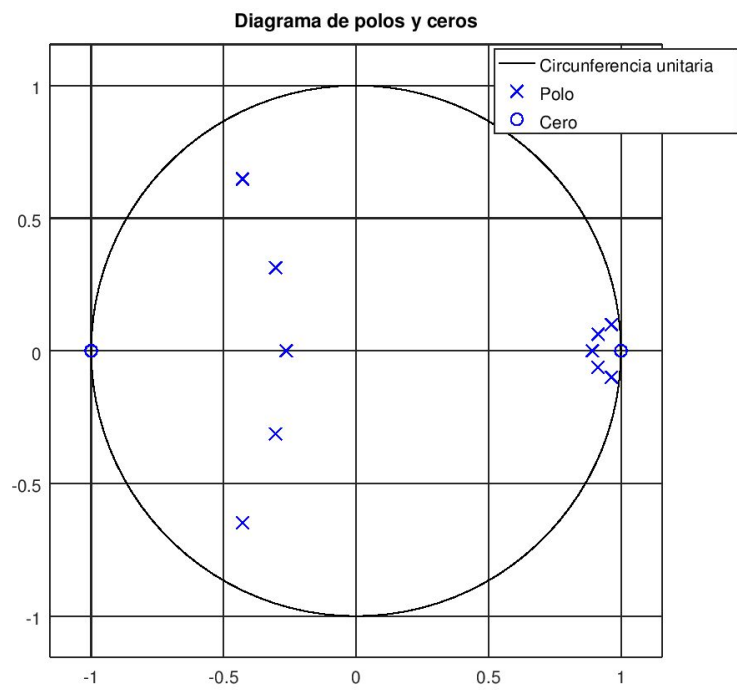


Figura 17. Diagrama de polos y ceros del filtro de orden 10.

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:

- Remoción de derivas
- Cambios en la forma de la señal
- Retardo de la señal filtrada respecto de la original

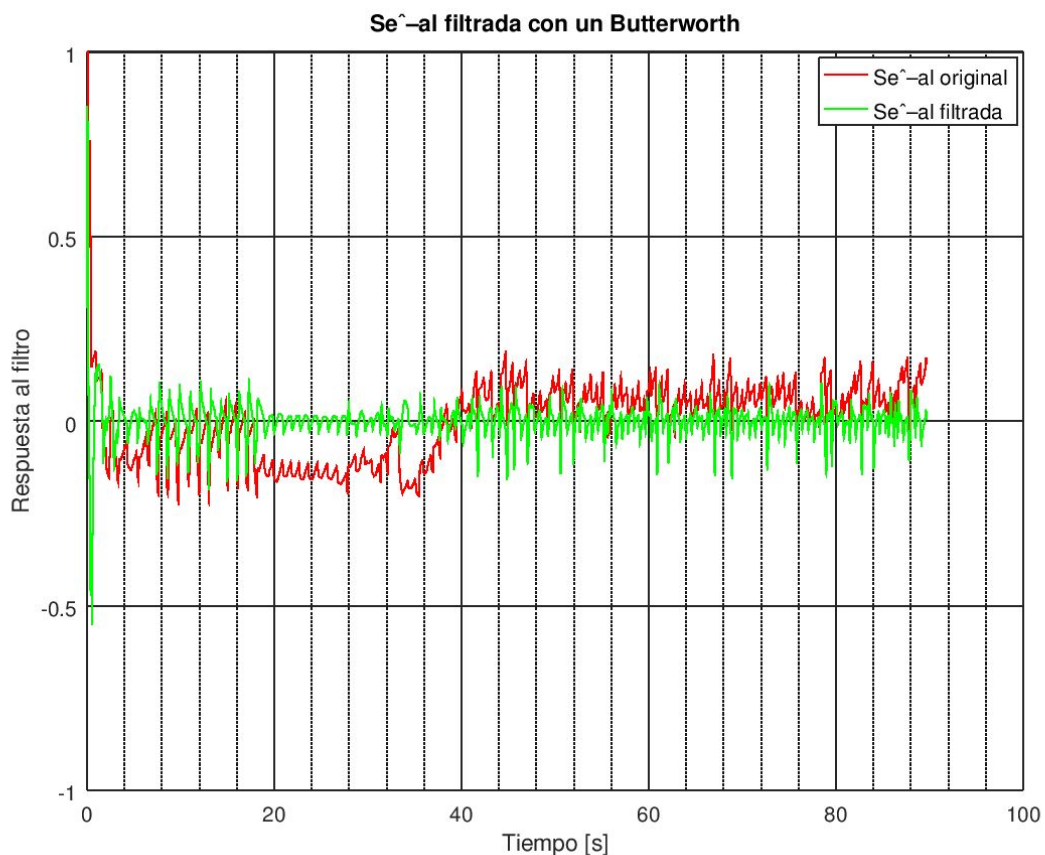


Figura 18. Señal original y su salida del filtro Butterworth en superposición.

Se ve que desaparece la señal de frecuencia baja (sobre la que parece montada la señal original), y que desaparecen los picos más bruscos, ya que se están filtrando las frecuencias altas.

También se observa una deformación de la señal, dada por lo mucho que varía la fase.

Comentar abc...

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 $-\frac{d}{d\Omega}\phi(\Omega)$. Como en este caso en la zona que nos interesa la fase se ve lineal, tomando dos puntos de la recta se puede calcular la derivada:

$$-\frac{d}{d\Omega}\phi(\Omega) \approx 5.29 \quad ? \quad \text{Esto no es temporal....}$$

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.

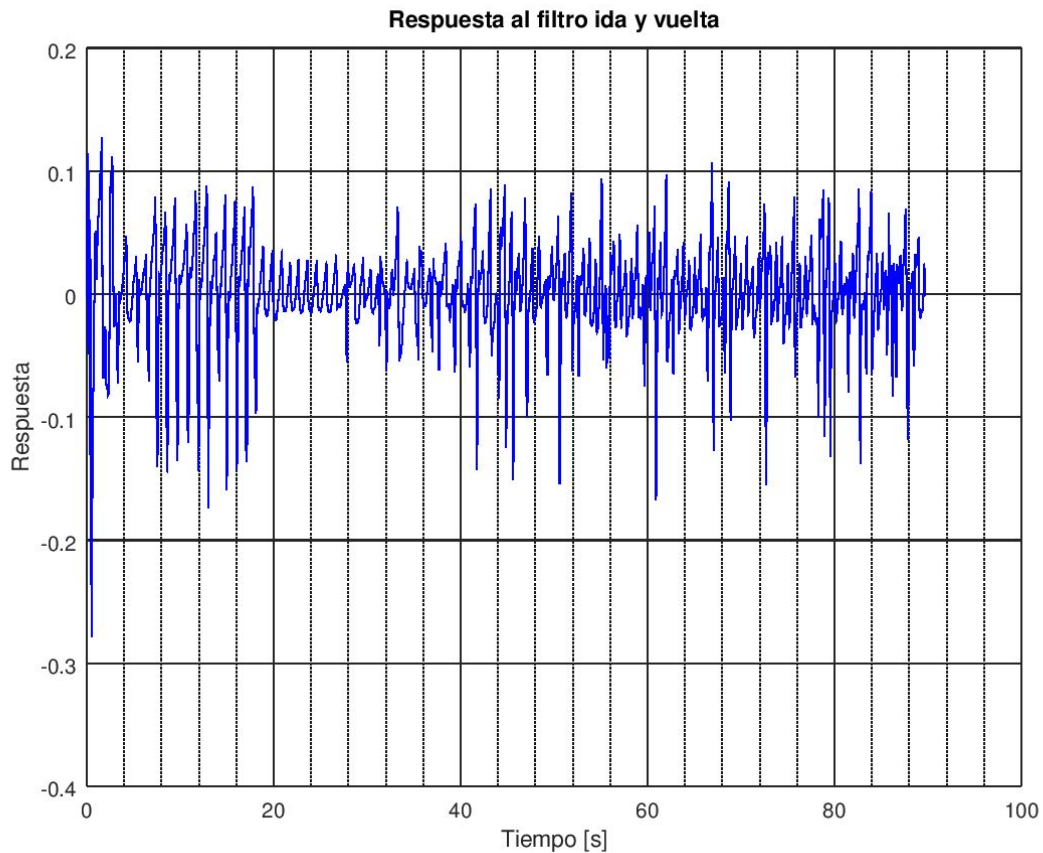


Figura 19. Respuesta al filtrado IIR.

A diferencia de la señal filtrada con el Butterworth, esta señal conserva la forma de la señal original, porque $H \times H^* = |H|^2$, que es real (su fase es igual a cero). Que la fase sea igual a cero también hace que la señal de salida no tenga retardos.

Agregar una justificación más completa...

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?

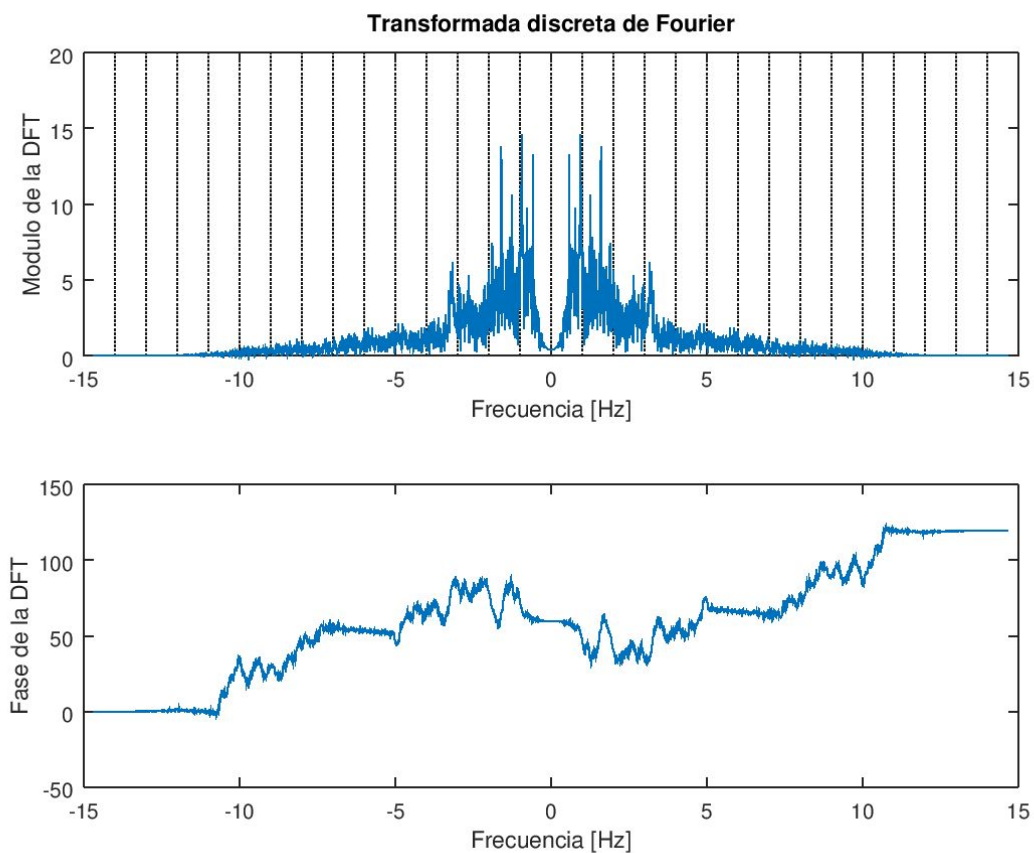


Figura 20. Transformada discreta de Fourier de la señal filtrada.

En la figura 20 podemos ver que las componentes más importantes de frecuencia están en una banda de 3.5 Hz aproximadamente, por lo que las ventanas de los espectrogramas siguientes tendrán un ancho relacionado con ese valor.

Para obtener mejor resolución en frecuencia debería usar una ventana 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.

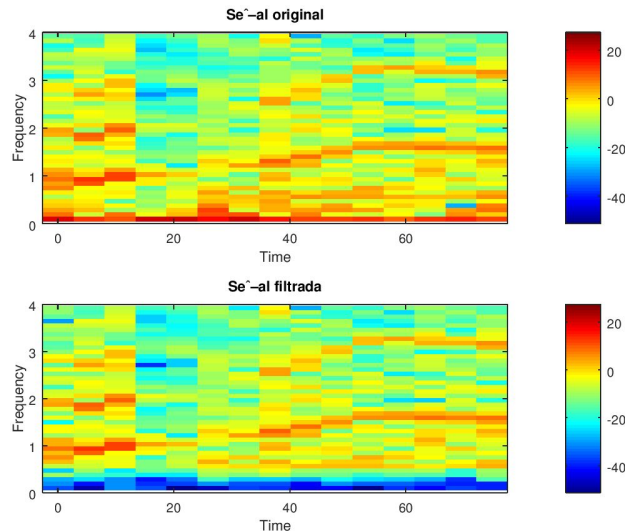


Figura 21. Espectrogramas de la señal original y de la filtrada usando una ventana de Hamming de longitud tal que quepa un rango de 3.5Hz.

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. La diferencia entre los resultados usando ventana de Hamming y de Hanning se ve 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.

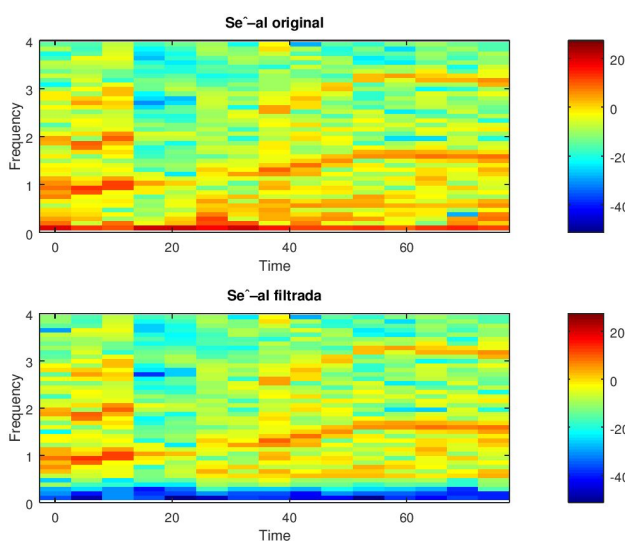


Figura 22. Espectrogramas de la señal original y de la filtrada usando una ventana de Hanning de longitud tal que quepa un rango de 3.5Hz.

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.

Falta analizar qué pasa con la señal de audio y la otra componente que se ve en el espectrograma...

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:

- Filtrado pasa-banda de la señal, utilizando el filtrado del ejercicio 7.
- Filtro de derivada, implementado con un filtro FIR $h(n)=[-2 \ -1 \ 0 \ 1 \ 2]$.
- 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.
- 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.

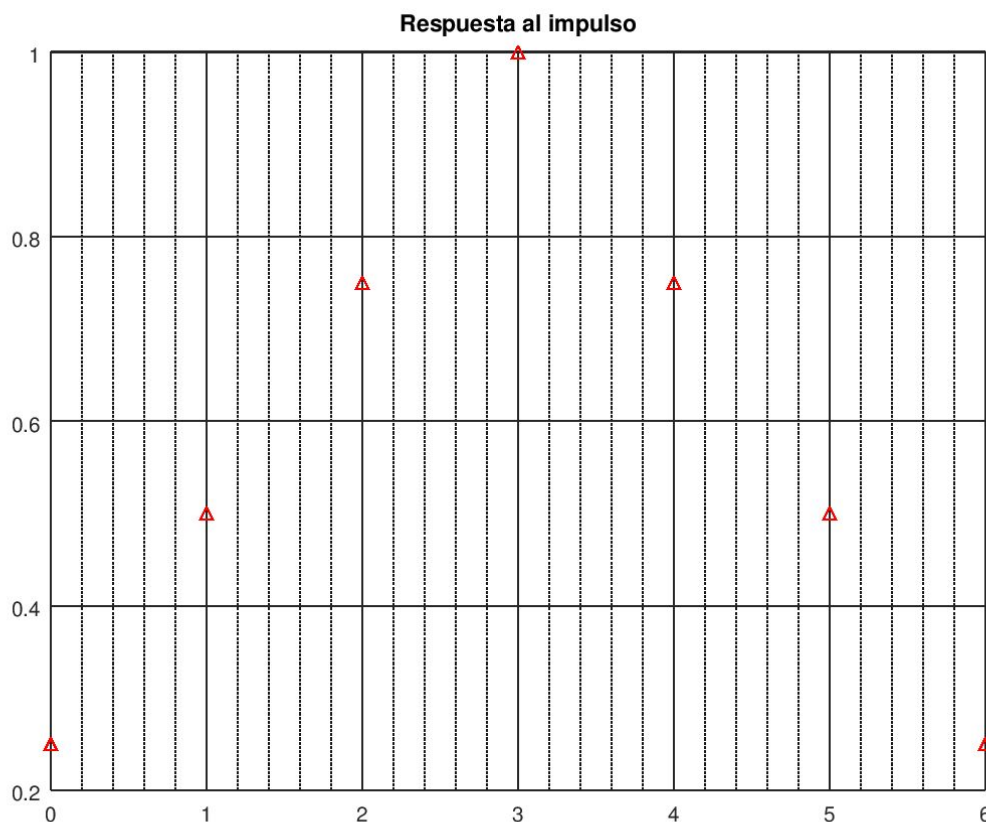


Figura 23. Respuesta al impulso del filtro a utilizar.

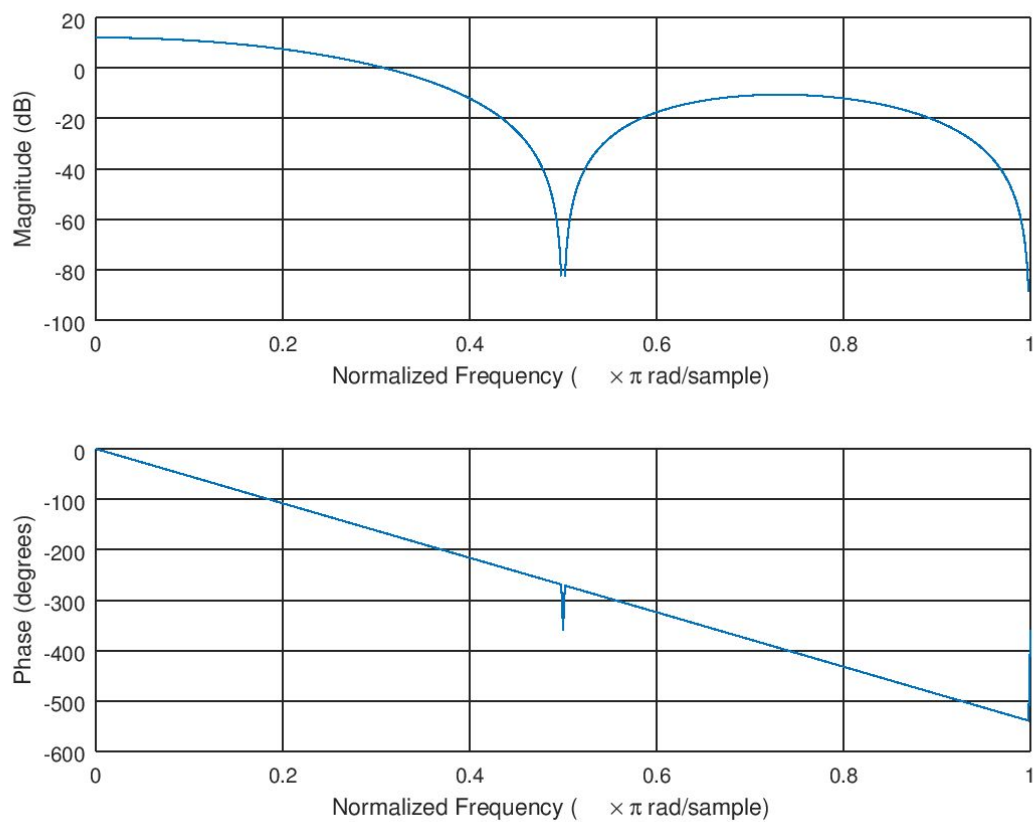


Figura 24. Respuesta en frecuencia del filtro interpolador.

- 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.*

Podemos observar en la figura 25 cómo queda la señal luego de ser filtrada, y en la 26 los picos marcados en la señal filtrada.

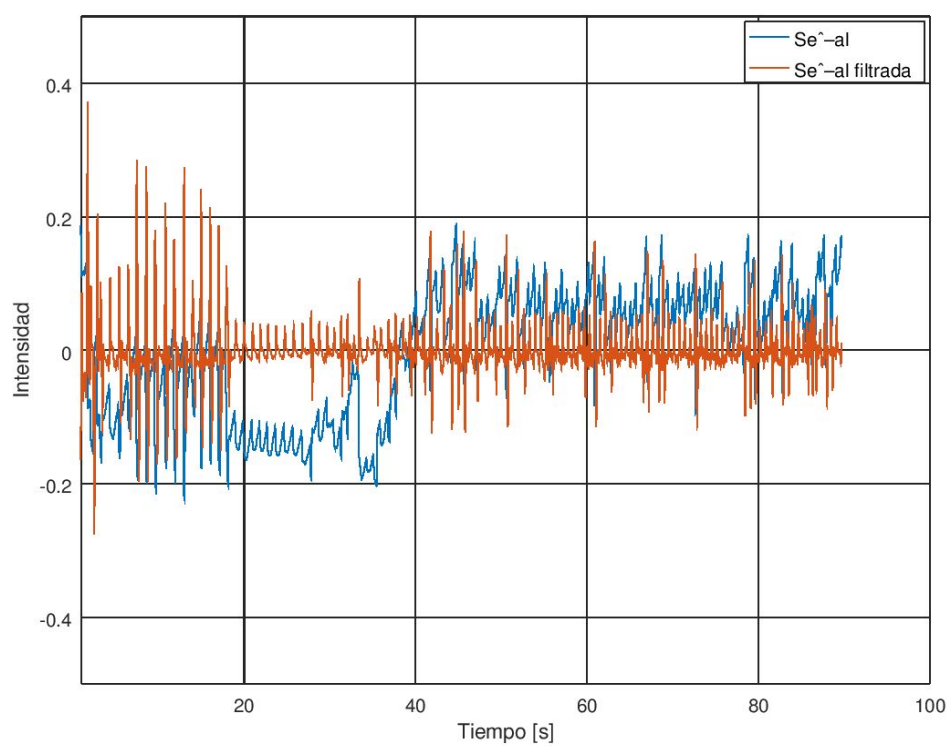


Figura 25. Señal original y filtrada en superposición.

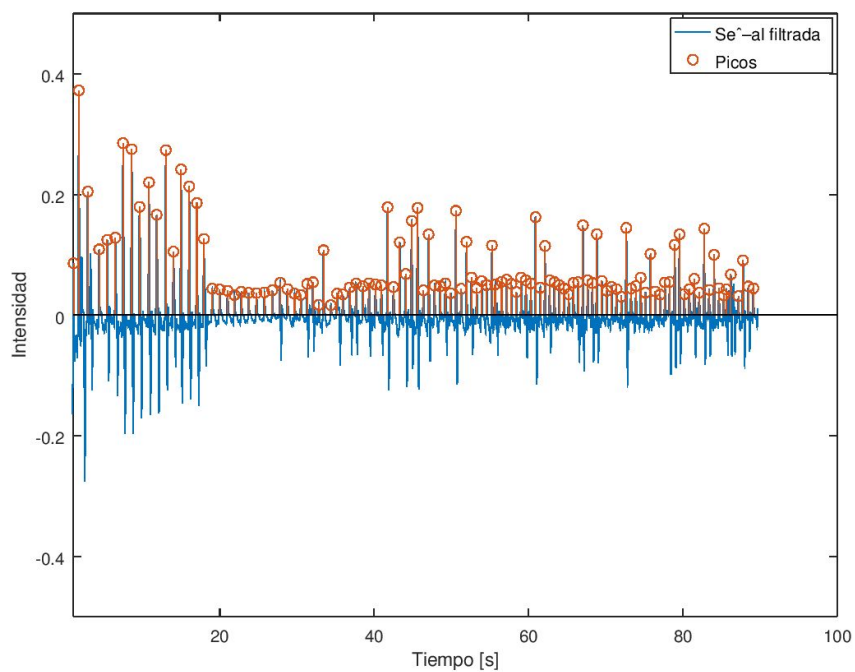


Figura 26. Señal filtrada en superposición con sus picos.

Ejercicio 11

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

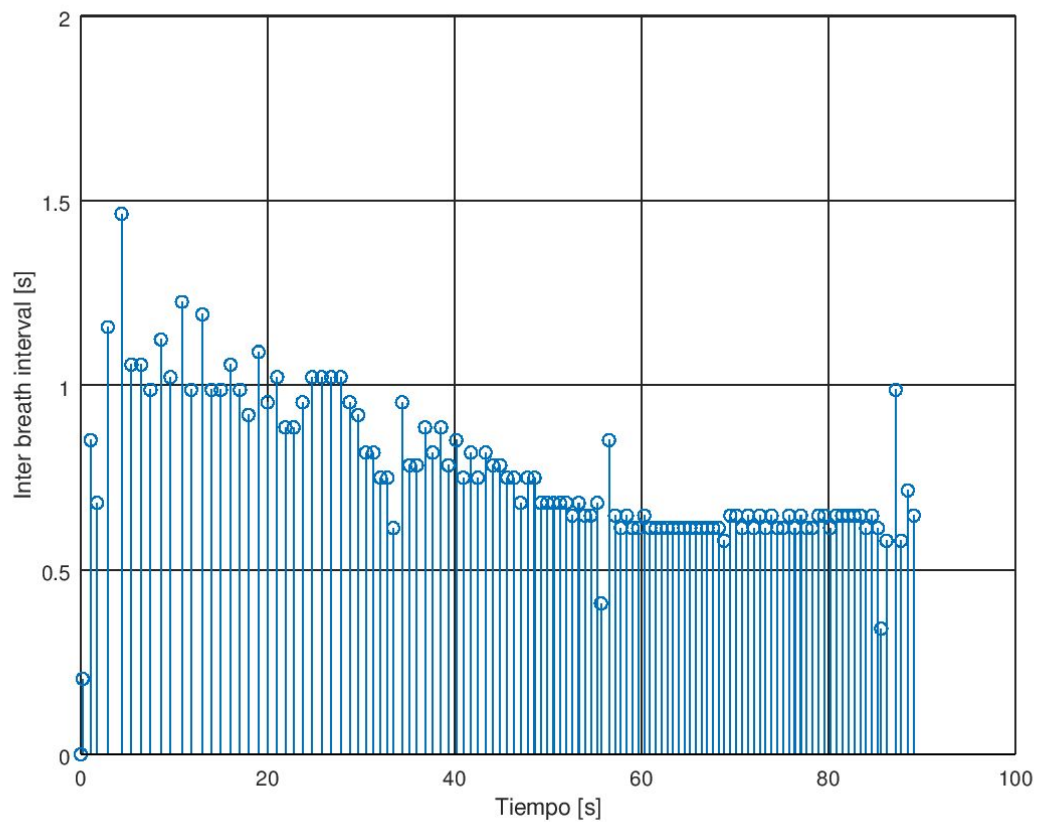


Figura 27. Evolución del intervalo entre latidos en función del tiempo.

En la figura 27 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.*

En la detección de latidos del ejercicio 11 están implementadas estas dos mejoras y una más, que consiste en que desde que se supera el umbral, y hasta que la señal vuelve a bajar, sólo puede haber un pico. Los gráficos mostrados antes ya tienen en cuenta estas mejoras.

Ejercicio 13

Estaría interesante hacerlo para ver qué pasa si en vez de la sumatoria que se usa en el script para generar la entrada se usa la mediana de los valores de los píxeles.

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.

-