



UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE INGENIERÍA

2° CUATRIMESTRE DE 2017

86.05

SEÑALES Y SISTEMAS

Trabajo Práctico n°1
Detección de latidos cardíacos

Integrante:
Garcias, Ezequiel <garciaezequiel91@gmail.com>

Padrón:
93191

20 de noviembre de 2017

Índice

1. Introducción	2
2. Objetivos	2
3. Enunciado	3
4. Desarrollo	5
4.1. Ejercicio 1.	5
4.2. Ejercicio 2.	7
4.3. Ejercicio 3.	8
4.4. Ejercicio 4.	9
4.5. Ejercicio 5.	10
4.6. Ejercicio 6.	11
4.7. Ejercicio 7.	12
4.8. Ejercicio 8.	13
4.9. Ejercicio 9.	13
4.10. Ejercicio 10.	14
4.11. Ejercicio 11.	20
5. Conclusiones	21

1. Introducción

Hoy en día los latidos del corazón se pueden ver por medio de instrumentación médica, la cual nos brinda información acerca de nuestro estado de salud. Sin embargo es muy común el uso de la tecnología para leer señales de la vida cotidiana. La mayoría de los celulares que las personas poseen tienen una gran capacidad de procesamiento. De modo que para poder saber las pulsaciones por minuto de un corazón ya no es necesario utilizar un instrumento muy costoso y de mucha precisión.

2. Objetivos

El objetivo del presente trabajo es, en primer lugar, realizar un análisis en tiempo y frecuencia de la señal FPG obtenida a partir de la grabación de video de un smartpho-
ne. Posteriormente, se implementará una serie de filtros para obtener la componente AC de la señal, la cual se utilizará finalmente durante la implementación del detector de latidos. En lugar de utilizar el video directamente como señal, lo cual resulta demandante con la memoria, se trabajará con las intensidades de color para cada frame del video en cuestión.

3. Enunciado

Detección de latidos cardíacos.

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.
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.
3. Rehacer el punto anterior, pero utilizando DFT. ¿Puede en este caso identificar cuándo se da el cambio de LPM?
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?
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
6. A partir de la respuesta en fase del filtro, calcule su retardo temporal y compare con lo observado en el punto 5c.
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.
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?
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).

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 MA1 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.
- Detector de picos mediante umbral (puede definir como umbral un valor arbitrario).
- Gráfico en superposición de la señal con las marcas de los picos detectados. Sugieren-
cia para el punto e: en general, un pico en la señal producirá la detección de múltiples muestras por encima del umbral. Para reducirlas a sólo una, puede utilizar la función `diff` para evaluar la primera derivada sobre el vector de muestras detectadas (la cual contiene sólo unos y ceros), quedándose solo con las muestras positivas que resulten y buscando, para cada una de estas, la posición del máximo inmediato de la señal original dentro de un intervalo pre-establecido (este deberá contener mínimamente a la duración aproximada de los picos).

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.

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

- 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.
- 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.

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.

4. Desarrollo

4.1. Ejercicio 1.

Como podemos ver tenemos el gráfico de las tres señales superpuestas. Tenemos un total de 2636 muestras con una frecuencia de muestreo de 29.37Hz. Dando como resultado 90 segundos de video.

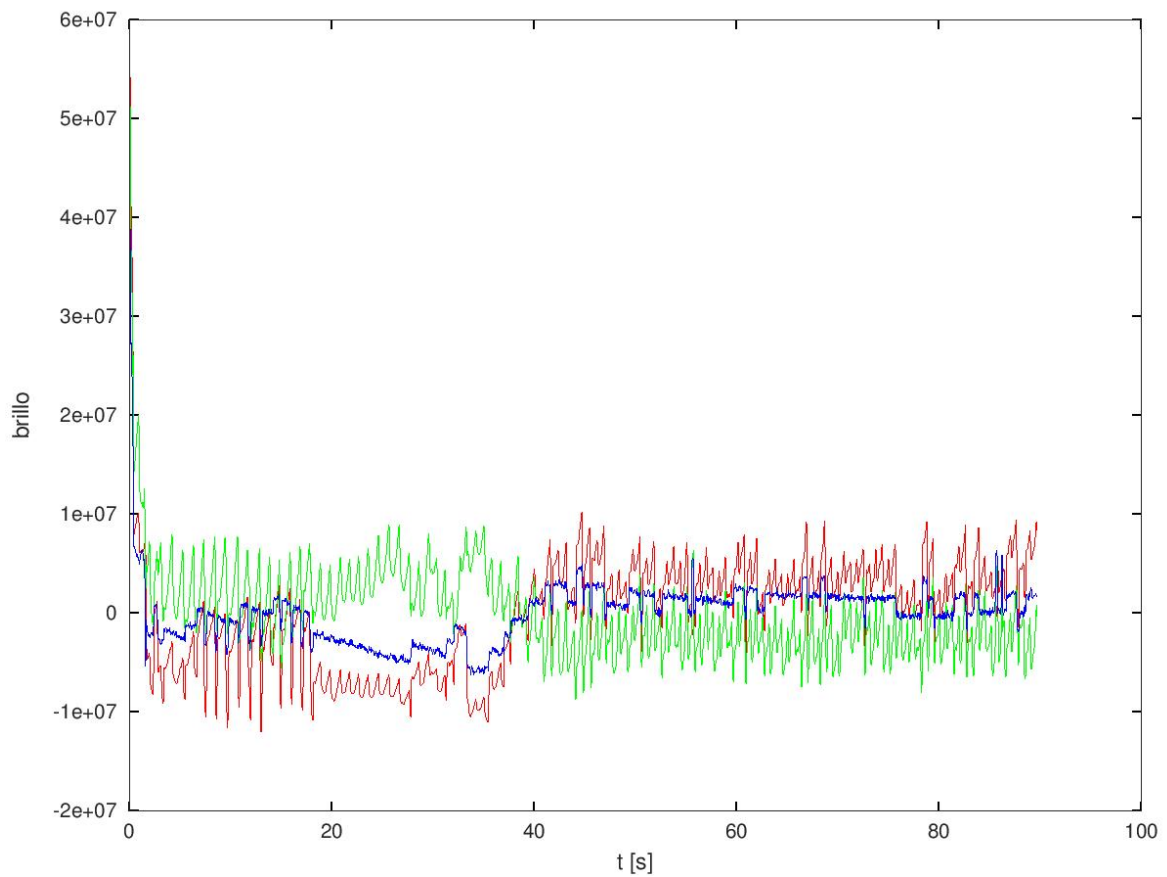


Figura 4.1 – Superposición tres señales Roja Verde y Azul.

La señal azul no presenta la altura de los picos suficientemente marcados para lograr obtener información útil de ella.
Las señal Roja y Verde pueden ser consideradas para obtener el resultado esperado.
Finalmente opte por trabajar con la señal verde.

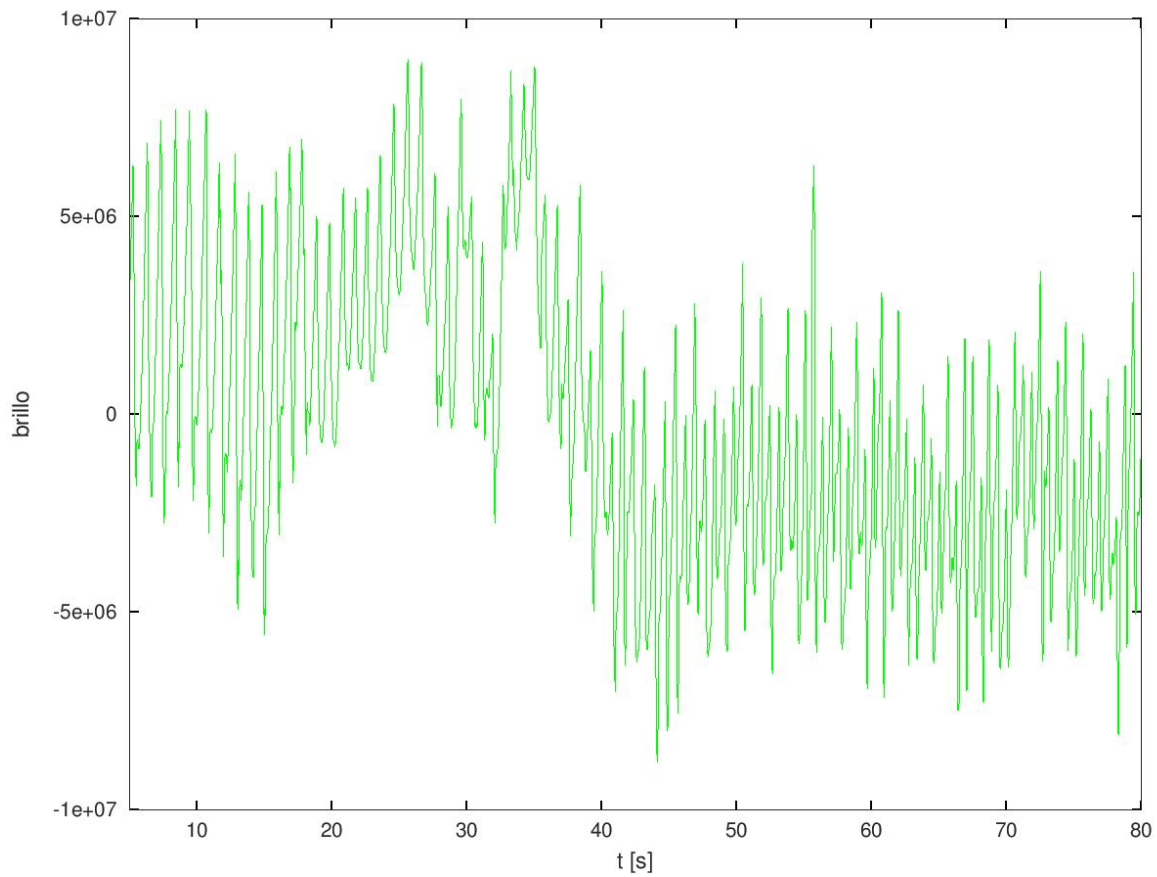


Figura 4.2 – Señal Verde.

4.2. Ejercicio 2.

Para poder observar la variación de latidos opte por graficar la señal verde cada 20 segundos. Finalmente al contar manualmente notamos como cada 20 segundos se observa una aceleración en el ritmo cardíaco.

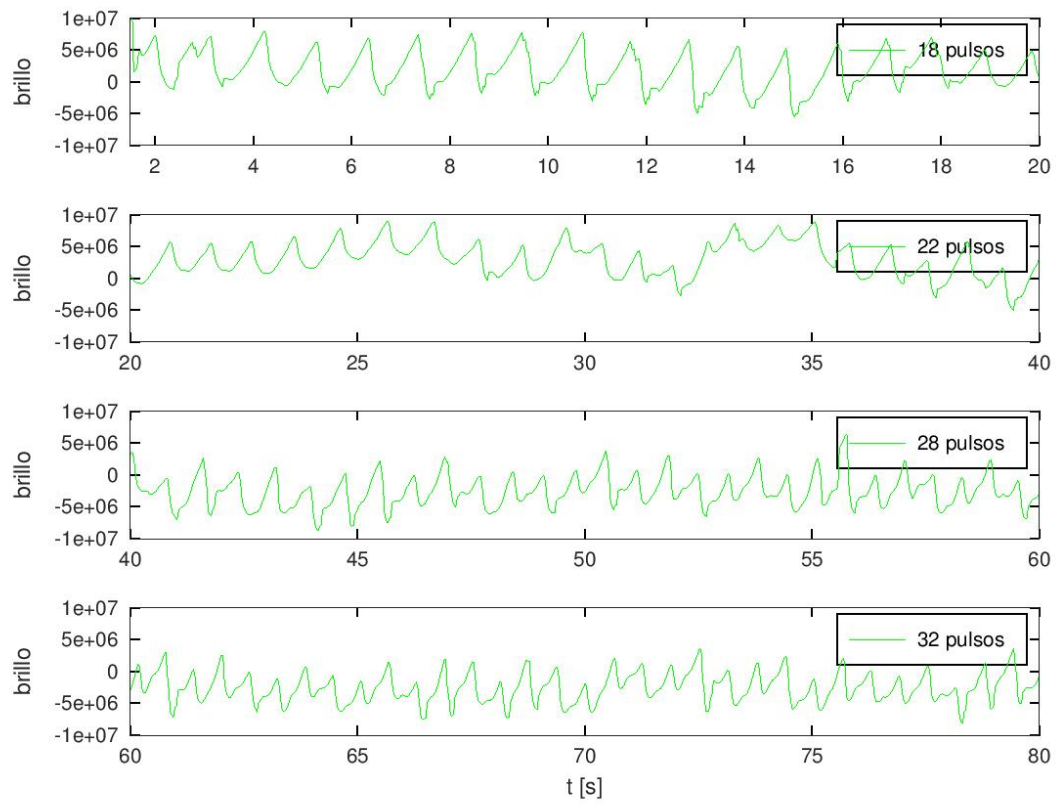


Figura 4.3 – Señal Verde conteo de pulsos.

4.3. Ejercicio 3.

Luego de realizar una DFT a la señal verde, se nos pide hallar información acerca de los latidos por minuto.

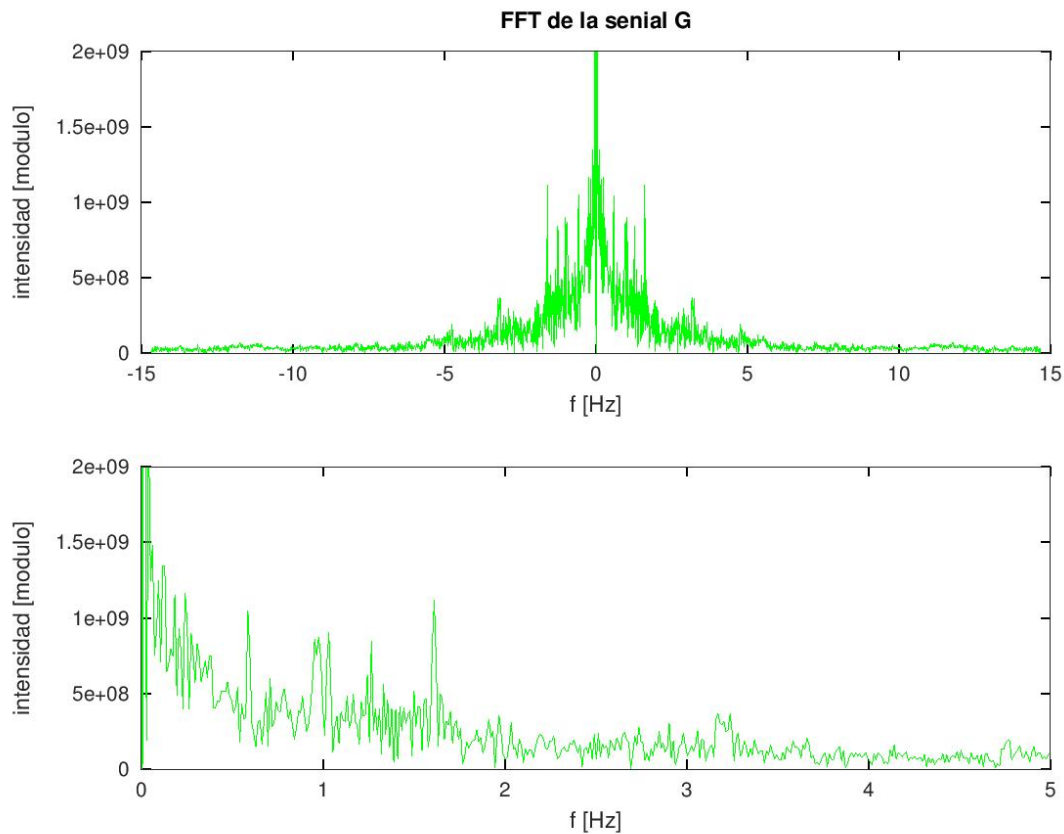


Figura 4.4 – Señal Verde Transformada Rápida de Fourier.

Como podemos observar la información aportada por la DFT solo nos dice cuales son las frecuencias presentes, pero no podemos obtener información sobre el cambio de ritmo cardíaco.

4.4. Ejercicio 4.

Para mejorar la lectura de la señal verde se propone diseñar un filtro pasabanda. El filtro tiene el fin de quitar las componentes de continua, bajas frecuencias y muy altas frecuencias que puede generar una lectura errónea de la información.

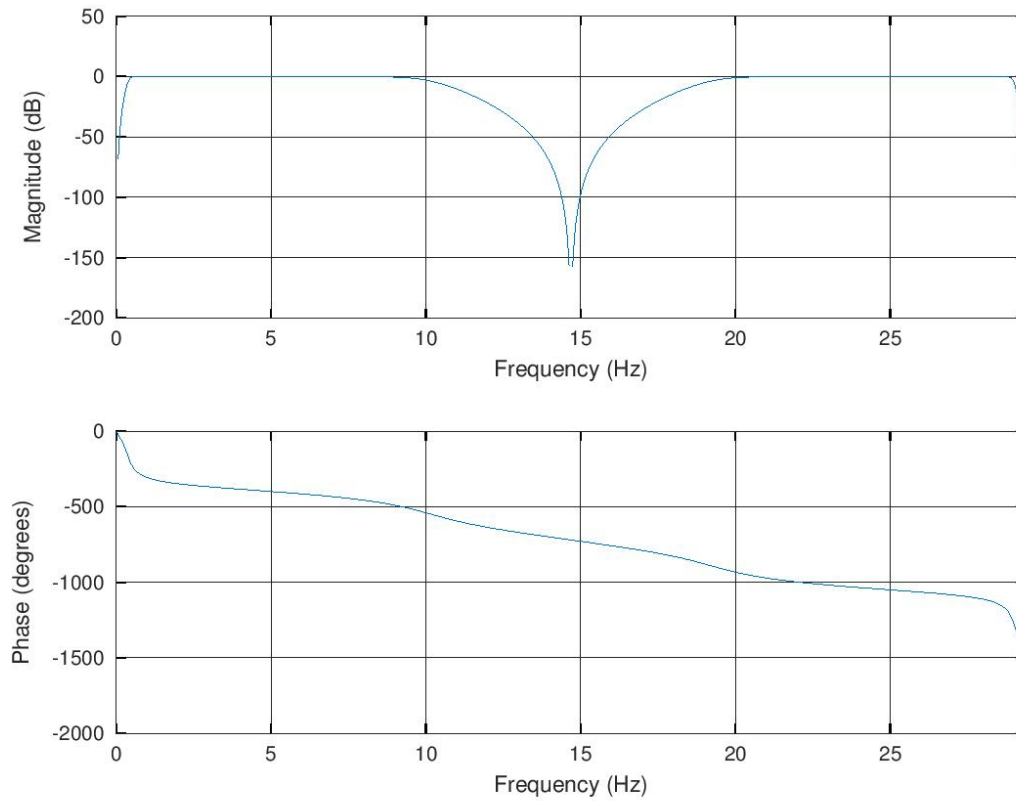


Figura 4.5 – Diagrama de filtro pasa banda.

El orden del filtro genera cambios más abruptos para las frecuencias de corte seleccionadas.

4.5. Ejercicio 5.

Se aplica el filtro diseñado anteriormente a la señal original.
Dando como resultado el siguiente gráfico:

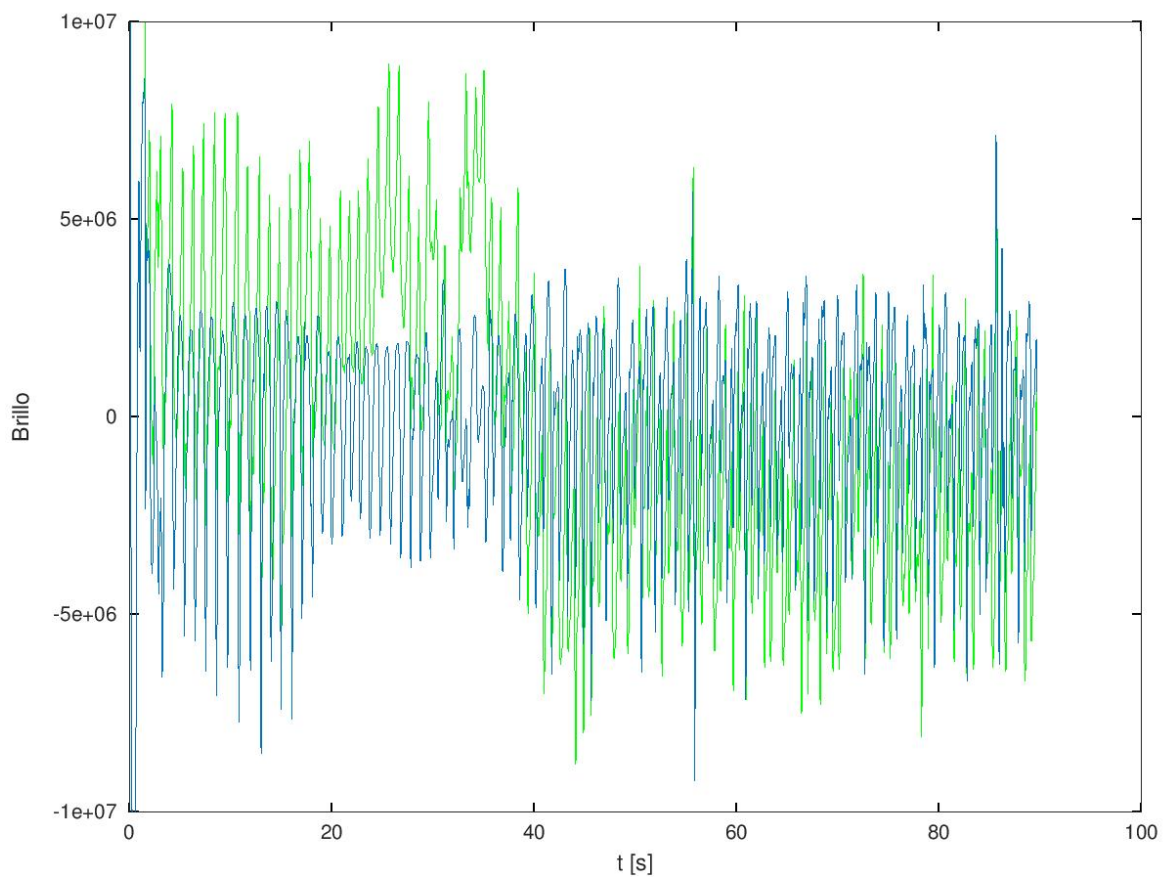


Figura 4.6 – Superposición de señales original y filtrada.

Como podemos notar se eliminó la componente de muy baja frecuencia sobre la cual estaba montada la señal original.

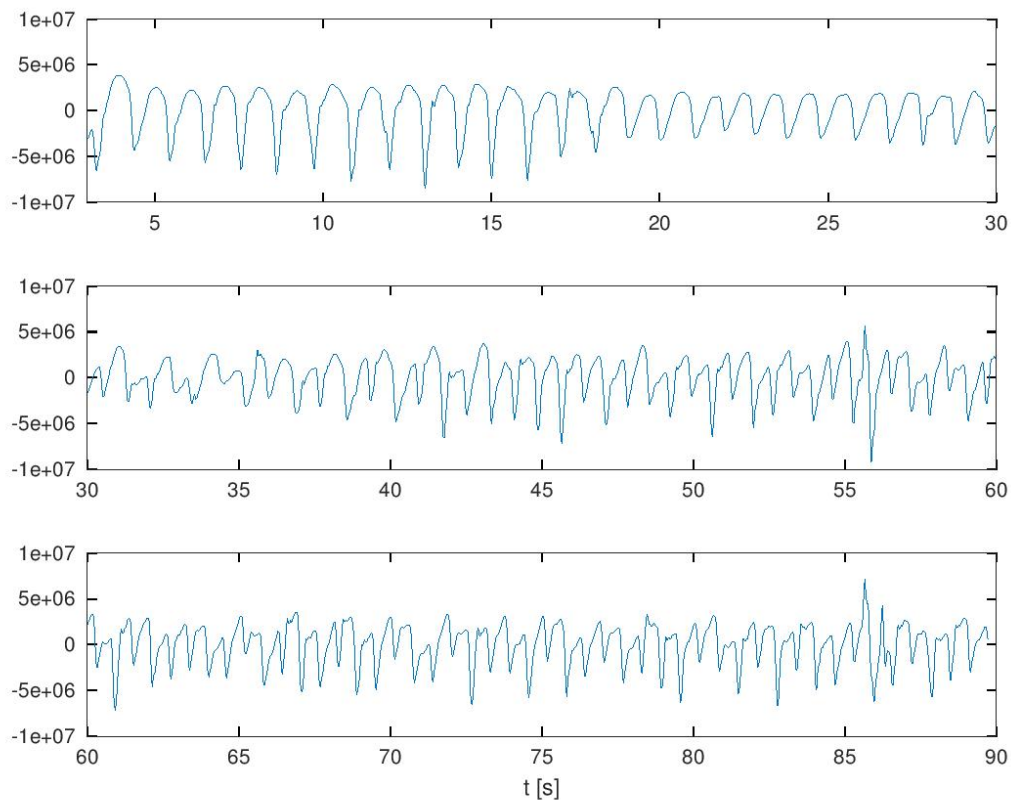


Figura 4.7 – Señal filtrada.

a)

Teniendo una vista más detallada en la señal filtrada, se puede apreciar como hubo una remoción de derivas al no tener saltos sobre los valores promedios de la señal.

b)

Los cambios en la forma de la señal se deben a dos motivos del filtrado, la quita de frecuencias y la deformación causada por el cambio de fase.

c)

La señal filtrada tiene un retardo en fase respecto a la señal original causado por la respuesta del filtro para diferentes frecuencias.

4.6. Ejercicio 6.

Te lo debo.

4.7. Ejercicio 7.

Se aplica la función `filtfilt` para el filtro diseñado anteriormente a la señal original. Dando como resultado el siguiente gráfico:

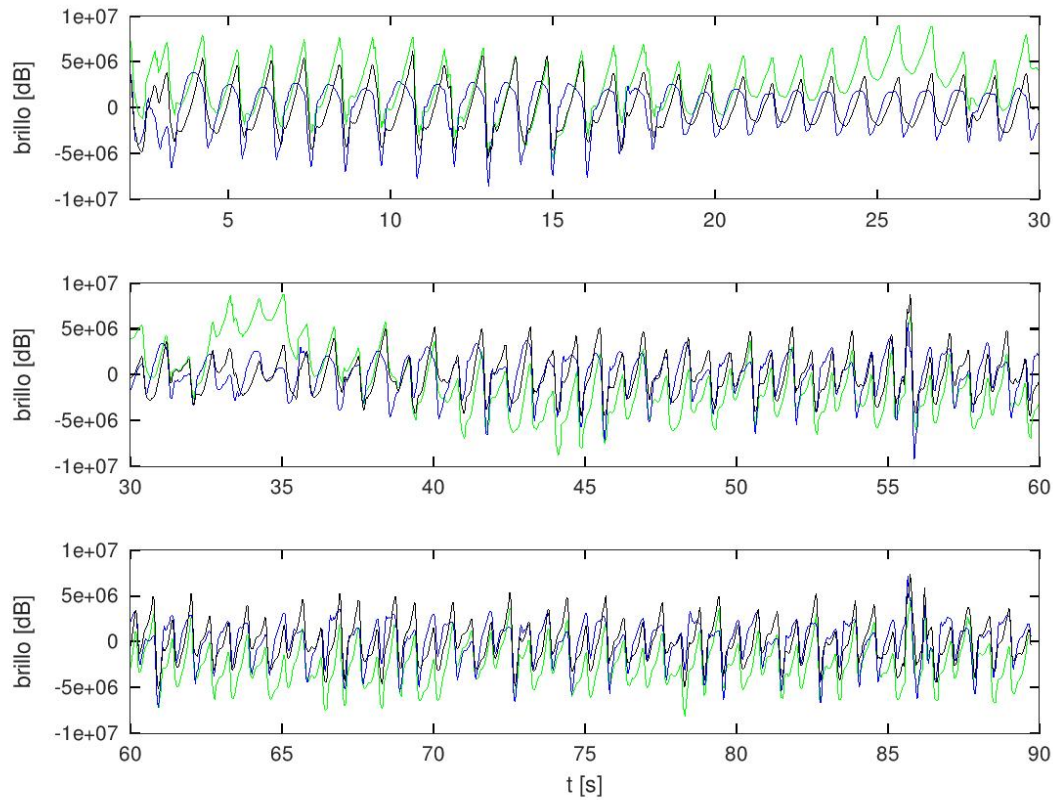


Figura 4.8 – Señales original, filtrada y filtrada con ida y vuelta.

Estamos viendo el resultado de hacer la operación de la señal pasando por el filtro y luego el filtro para transformado, dando como resultado una anulación del retardo de fase.

De modo que la señal luego del filtro no sufre deformación debido a la fase.

4.8. Ejercicio 8.

Se busca hallar el espectrograma de la señal, con una ventana de Hamming de 128 puntos, con un solapamiento de 127 puntos para tener buena resolución en frecuencia.

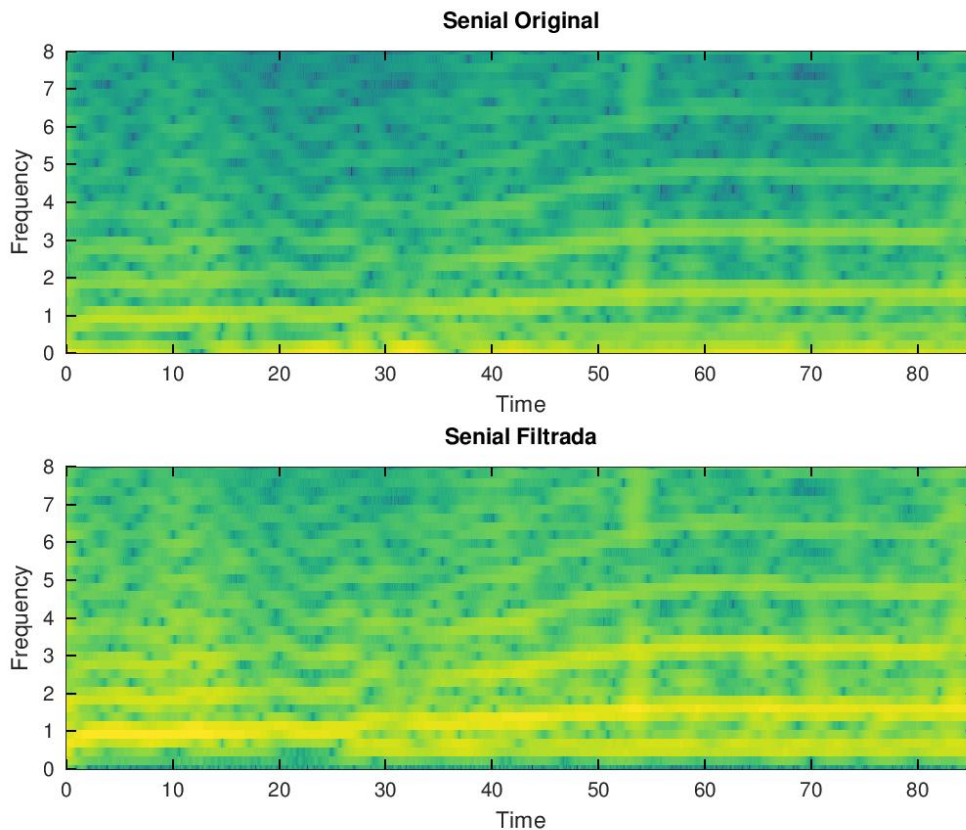


Figura 4.9 – Espectrograma de la señal original y la con ida y vuelta.

Se utilizo una ventana del tipo Hamming de 128 puntos.
A su vez esta ventana se desplazo en 1 punto para obtener un solapamiento de 127 y poder obtener la máxima resolución en frecuencia posible.
En diferencia al punto 2 y 3, ahora si podemos notar la presencia de diferentes frecuencias para diferentes tiempos de la señal.

4.9. Ejercicio 9.

Podemos ver dos espectrogramas, el primero corresponde a la señal original, el segundo corresponde la señal filtrada.
La presencia del ruido no nos permite ver claramente las frecuencias presentes en diferentes tiempos.
Finalmente la segunda imagen nos muestra claramente como hay un primer cambio de ritmo en el segundo 20. Luego desde el segundo 40 al 60 se acelera el ritmo constantemente hasta alcanzar un ritmo cardíaco constante en el segundo 60 hasta el 90.

4.10. Ejercicio 10.

se implementan las ultimas instancias para obtener los resultados buscados.

a)

Se utiliza el filtro diseñado en el punto 7, para finalmente obtener el siguiente resultado.

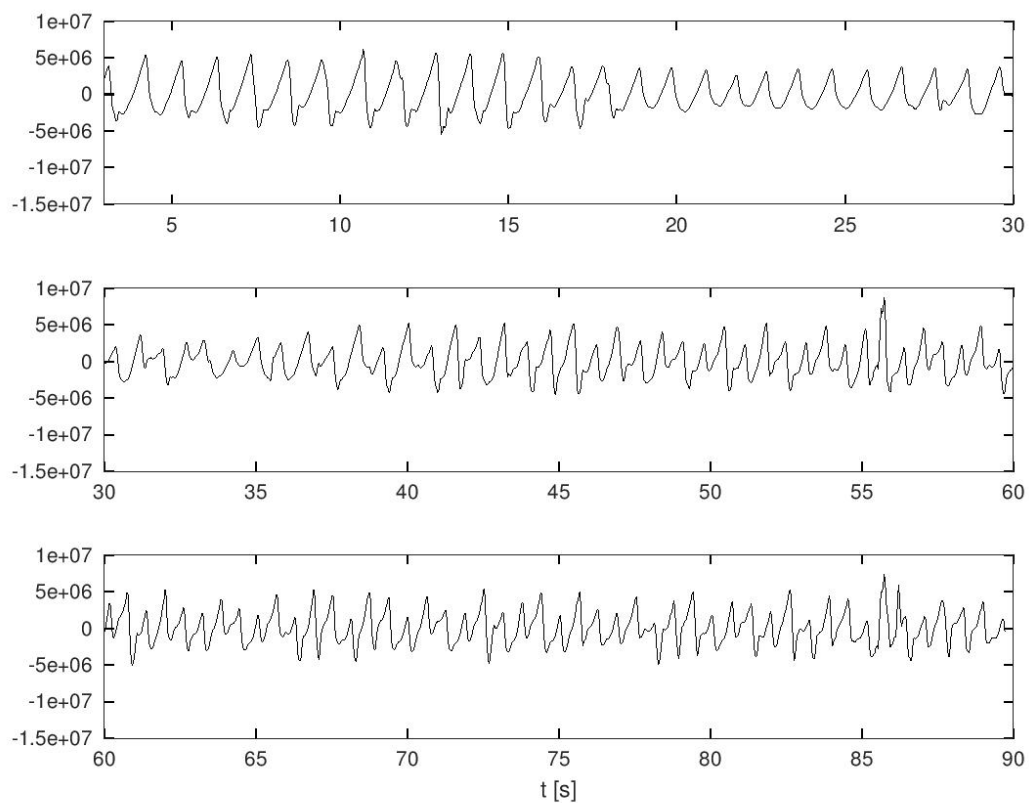


Figura 4.10 – Señal filtrada por el pasabanda.

b)

Se implementa el filtro de derivada, dando como resultado lo siguiente.

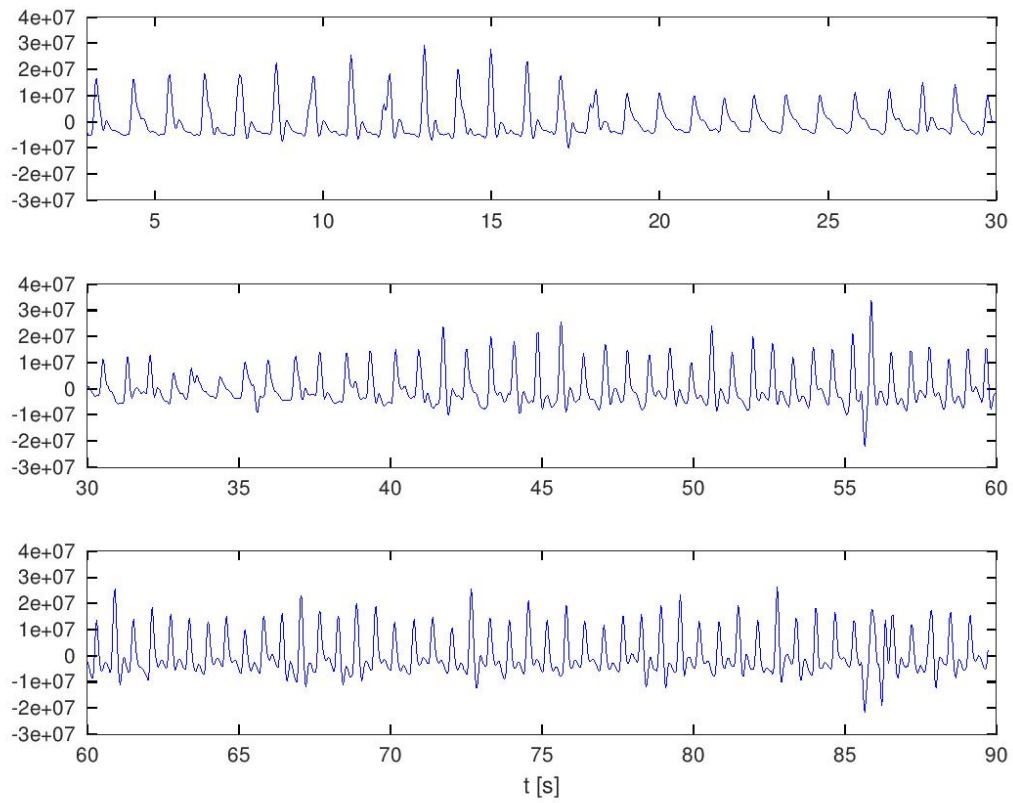


Figura 4.11 – Señal derivada por filtro.

La señal derivada se asemeja mucho más al pulso cardíaco.

c)
Normalización con energía instantánea.

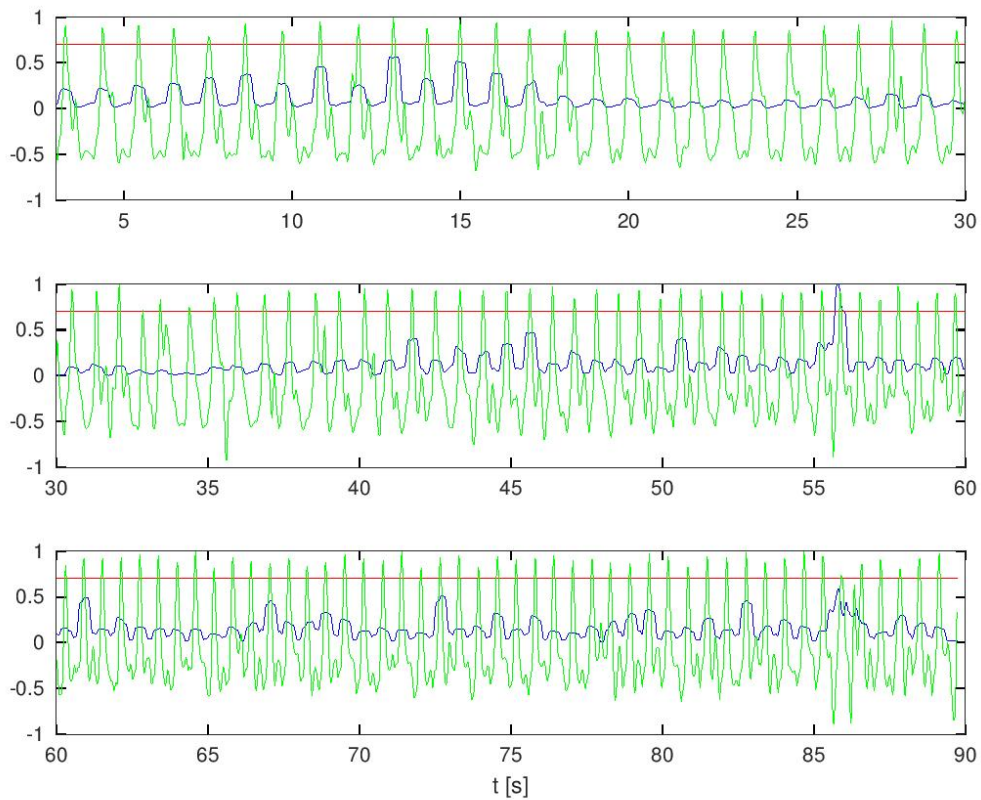


Figura 4.12 – Señal normalizada por la energía.

La utilidad de generar la normalización es para encerrar toda la señal sobre un nivel de alturas donde resulte más fácil realizar un algoritmo de búsqueda de picos.

d)
Sobremuestreo con un factor de 4.

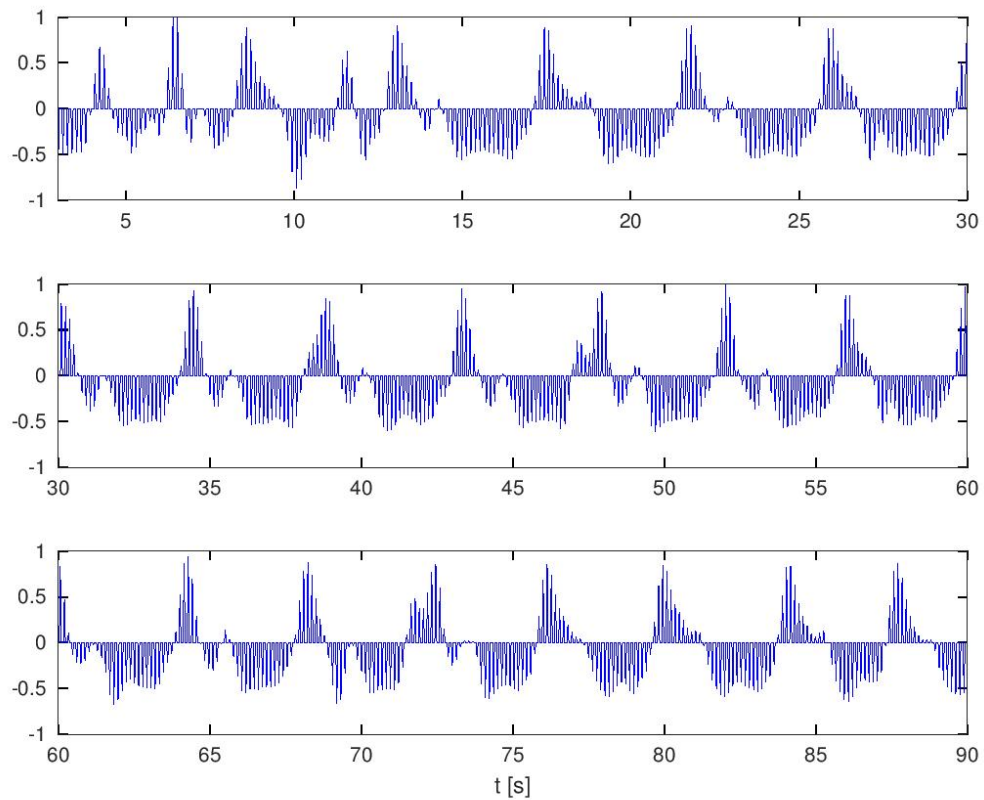


Figura 4.13 – Señal con ceros agregados.

Se realiza un sobremuestreo de factor 4 para poder obtener una mayor resolución al momento de buscar por medio de un umbral de altura, debido a que en la señal original podemos ver como existe una función continua que pasa por el umbral seleccionado, sin embargo la resolución de la señal puede producir que no exista ningún punto de coincidencia.

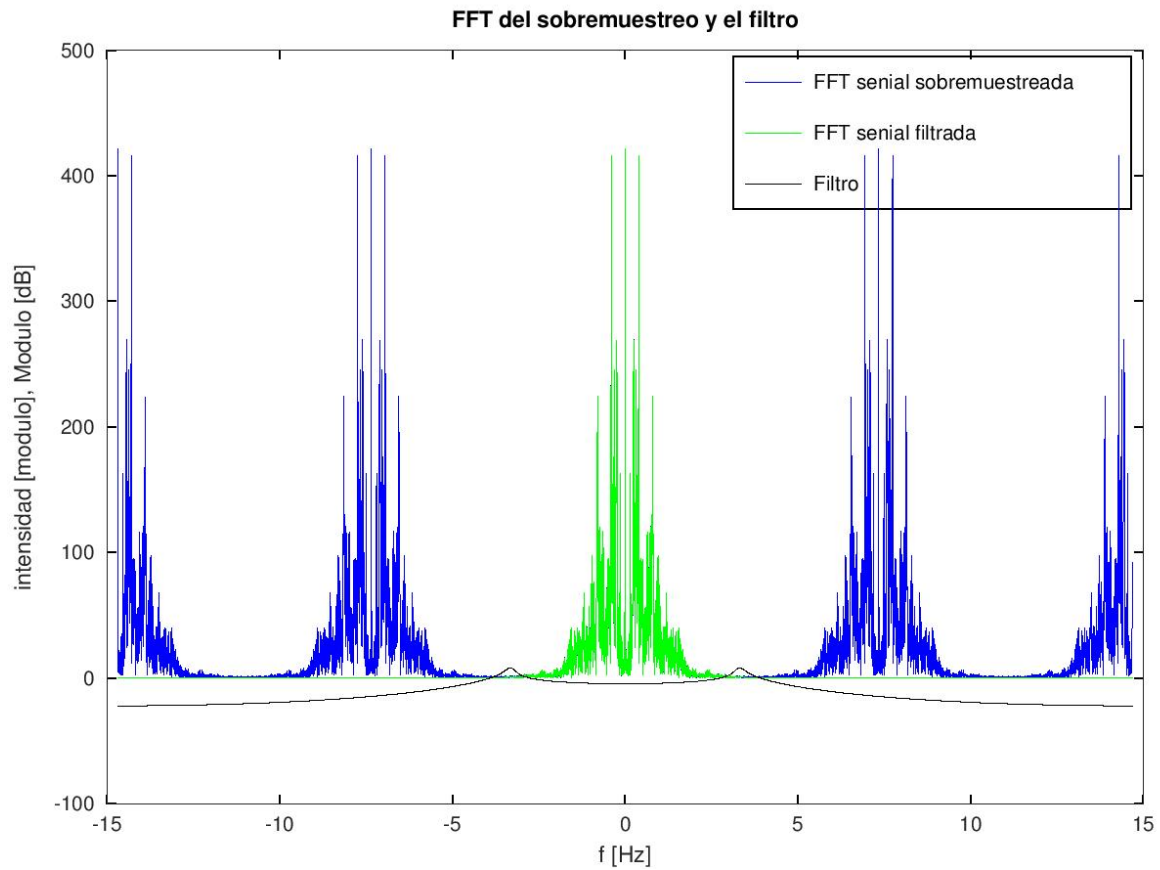


Figura 4.14 – FFT de la señal sobremuestreada, el filtro pasa bajo y la señal resultante.

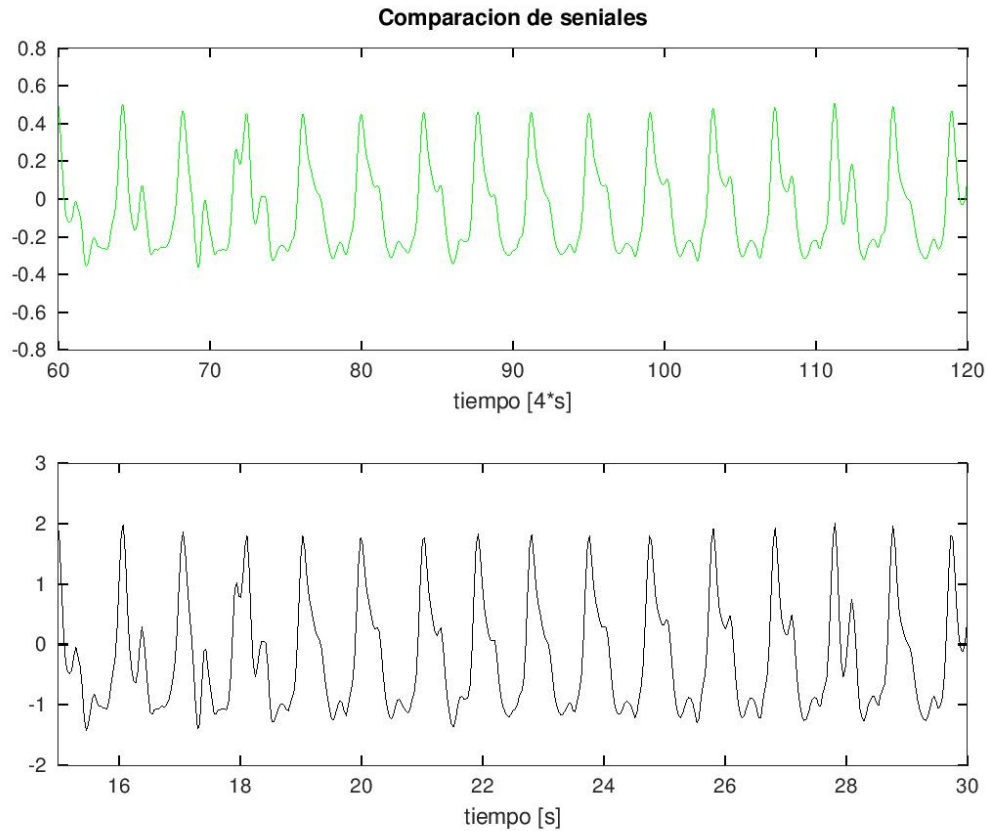


Figura 4.15 – Comparación de la señal y la señal sobremuestreada.

e)

Se implementa un algoritmo de búsqueda de picos con un umbral establecido arbitrariamente en 70 % del máximo de la señal con un margen de tolerancia de 8 %.

f)

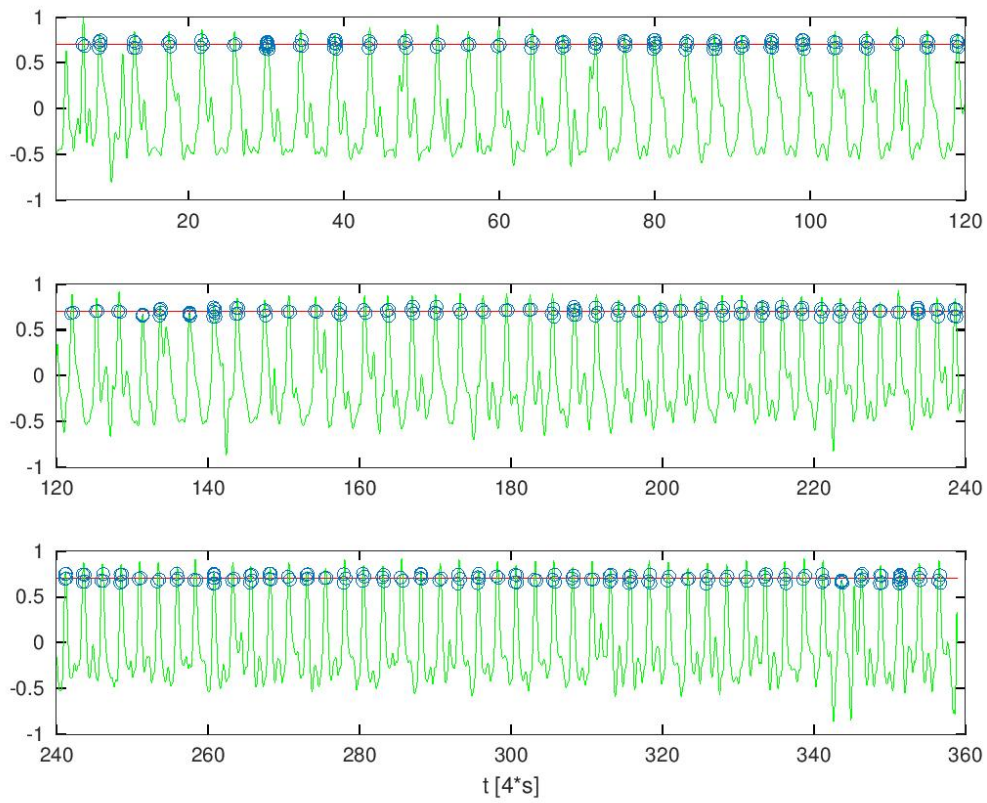


Figura 4.16 – Señal sobremuestreada, con el umbral y sus puntos de coincidencia.

4.11. Ejercicio 11.

SACAR LA CUENTA.

5. Conclusiones

Como conclusión podemos decir que es muy sencillo diseñar filtros digitales para procesar señales.

La correcta implementación de los filtros nos permiten, quitar frecuencias que no son de utilidad y ver con mayor exactitud un fenómeno físico que fue medido por medio de un método analógico.

El poder de procesamiento de una computadora combinado con el algoritmo correcto puede dar como resultado un programa eficiente para poder interpretar las señales que se encuentran presentes en la vida cotidiana.

De este modo, fue posible realizar una medición de los latidos por minuto de una persona sin necesidad de una herramienta de uso médico.