

UNIVERSIDAD DE SANTIAGO DE CHILE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA



Laboratorio N 2: Diseño de Filtros

Integrante: Nicolás Gutiérrez

Curso: Redes de Computadores

Profesor(a): Carlos Gonzalez Cortes

03 de Mayo de 2019

Tabla de contenidos

1. Introducción	1
2. Marco teórico	2
2.1. Filtro paso bajo	2
2.2. Filtro paso alto	2
2.3. Filtro paso banda	2
2.4. Función de Filtro butterworth	3
3. Desarrollo de la experiencia	4
3.1. Obtención del espectro de Amplitud vs Tiempo	4
3.2. Obtención del espectro con un filtro Pasa Bajos	6
3.3. Obtención del espectro con un filtro Pasa Altos	7
3.4. Obtención del espectro con un filtro Pasa Banda	8
3.5. Obtención del nuevo archivo .wav	10
4. Análisis de los resultados	11
4.1. Espectro amplitud versus tiempo	11
4.2. Espectro de la señal Con un filtro Pasa Bajos	11
4.3. Espectro de la señal Con un filtro Pasa Altos	12
4.4. Espectro de la señal Con un filtro Pasa Banda	12
5. Conclusiones	14
Bibliografía	15

1. Introducción

Durante todo el día, todos están expuestos a sonidos los cuales se pueden distinguir por el cómo suenan, se puede distinguir unos gritos de la bocina de un auto. Estos sonidos pueden ser distinguidos ya que tienen una suma de frecuencia distinta entre ellos y con distinta energía o amplitud haciendo que suenen de distinta forma. Incluso escuchando música, sin saberlo se están percibiendo distintas frecuencias las que se interpretan como notas musicales. De esta forma hay frecuencias que los humanos no pueden percibir como por ejemplo las frecuencias mayores a 20.000 Hz.

Como se mencionaba, un sonido está compuesto de frecuencias, de este modo, ¿qué pasaría si se ignora partes de las frecuencias y se deja solo algunas?. Si de un sonido particular se necesita obtener las bajas frecuencias ya que son las importantes, por ejemplo, para predecir cual será la intensidad de un sismo.

Justamente el objetivo de la experiencia que se muestra a continuación es aplicar filtros a un audio en particular, que permitan dejar ciertas frecuencias e ignorar o suavizar las demás para luego , usando el lenguaje Python con algunas de sus librerías, analizar qué frecuencias se pueden percibir, donde están las frecuencias con más energía o amplitud y qué pasaría con el audio después de aplicar el filtro.

Durante este documento se busca responder: si se elimina algunas frecuencias ¿Qué ocurre con el sonido ?, ¿Qué ocurriría si solo se deja la frecuencia con mayor amplitud?, entre otras preguntas. El documento esta dividido de tal forma que se explican los conceptos y funciones a utilizar en la experiencia, pasando después a la parte de exposición de los resultados con su respectivo análisis para finalmente llegar a la conclusión del informe.

2. Marco teórico

2.1. Filtro paso bajo

Este filtro introduce muy poca atenuación a las frecuencias que son menores que una determinada, llamada frecuencia de corte. Las señales de alta frecuencia son bloqueadas o impedidas por este filtro.

2.2. Filtro paso alto

Este filtro permite que las señales de alta frecuencia pasen a través más fácil y las señales de baja frecuencia son atenuadas. Todas las frecuencias superiores a la frecuencia de corte no tendrán atenuación.

2.3. Filtro paso banda

Tipo de filtro electrónico que deja pasar un rango de frecuencias de una señal t atenúa el paso de las señales que están fuera de ese rango establecido.

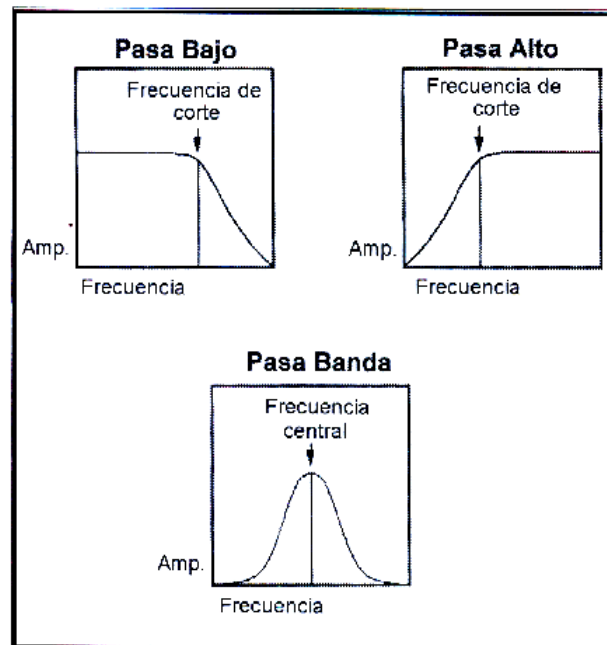


Figura 1: Explicación Teoría de de los filtros

2.4. Función de Filtro butterworth

Para poder hacer los filtros necesarios al audio Handel, es necesario aplicar la función *butter()* de la librería *Scipy*. La función Butter es una función que aplica un filtro con un degrade (no corta la señal sino que va disminuyendo su amplitud) la cual recibe los siguientes parámetros :

1. Orden: El orden del filtro hace referencia al suavizado con el que se aplica el filtro, es decir, que tanto decae el filtro. A mayor orden la señal será cortada abruptamente, en cambio con un orden mas bajo, la señal será cortada con mas suavizado o degrade. Un corte abrupto implica que es un filtro ideal, sin embargo esto no es posible ya que al aplicarlo habría perdida de información.
2. Limites de corte: La función necesita saber los limites de corte, y estos deben ser pasado en frecuencia [Hz], a la función se le puede pasar tanto un limite, como un arreglo de dos limites (para el caso de pasa banda).
3. Tipo de señal: La función soporta tanto señales analógicas como digitales. Para este caso se utilizó para una señal digital.
4. Frecuencia de Muestreo: Si la configuración esta en señal digital, es necesario entregarle la frecuencia de muestreo con la que se esta trabajando.
5. Tipo de filtro : Es necesario indicarle que tipo de filtro se necesita (High, low, band).

3. Desarrollo de la experiencia

Para poder desarrollar la experiencia fue necesario utilizar Python 3 con las librerías *Numpy*, *Matplotlib*, *Scipy*. Estas librerías son para el manejo de números complejos, gráficas y funciones de conversión.

La idea de esta experiencia es tomar un archivo .wav el cual contiene el canto "aleluya", obtener la señal en el tiempo del mismo archivo, posterior graficar su espectrograma y aplicarle los filtros Pasa-Altos, Pasa-Bajos, Pasa-Banda para analizar que frecuencias posee la señal filtrada. Cuando se tenga los datos se le debe aplicar la transformada de Fourier y graficar para ver que frecuencias han sobrevivido. Al aplicar la transformada inversa de Fourier a cada una de las señales filtradas, se debe también guardar el archivo .wav resultante después del filtro, sin antes haber graficado su espectrograma.

Cabe destacar que para las pruebas se utilizó un orden de 8, un límite inferior de 1500 y un límite superior de 2000.

La forma en como se ha desarrollado esta experiencia se divide en pasos o etapas las cuales son:

3.1. Obtención del espectro de Amplitud vs Tiempo

Para esta parte se analiza cómo se comporta la señal de sonido en el tiempo, por lo que es necesario crear un gráfico que muestre la amplitud o energía que se obtuvo en un instante de tiempo t . Gracias a la función *read()* de la librería *scipy.io.wavfile* que recibe el nombre del archivo .wav y retorna un arreglo de dos elementos, se puede obtener los datos necesarios para poder crear el gráfico. El primer elemento del arreglo (*arreglo[0]*) da la información del número de las muestras percibidas en un segundo, el segundo elemento del arreglo (*arreglo[1]*) es la *data* del archivo.

Al aplicar la transformada de Fourier a estos datos se tiene :

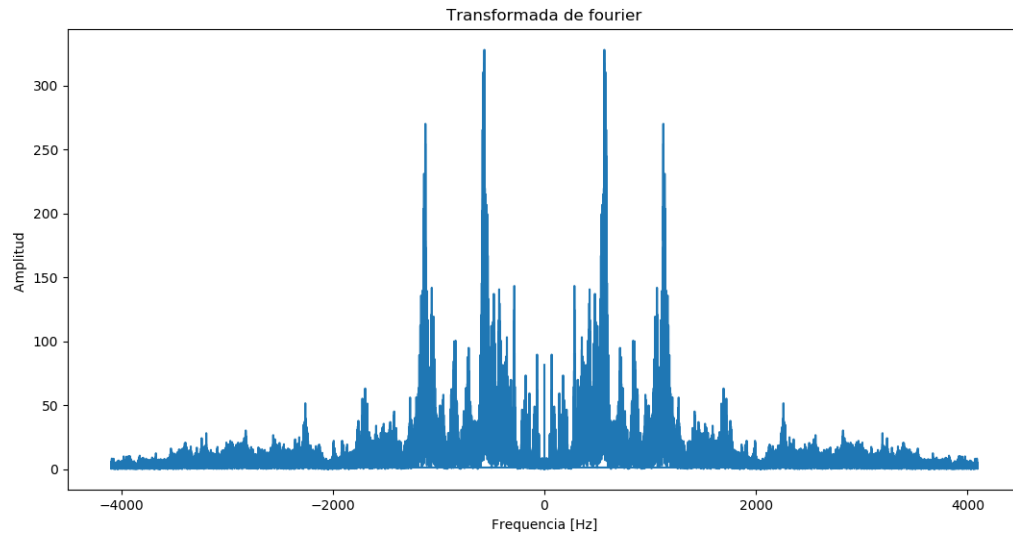


Figura 2: Transformada de Fourier del Audio original

Ahora si se obtiene el espectrograma de la seal original:

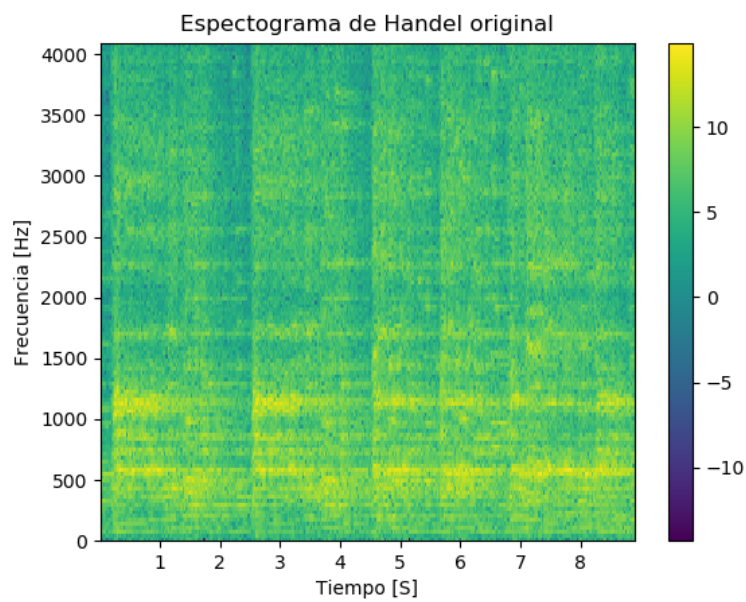


Figura 3: Espectrograma de la seal original

3.2. Obtención del espectro con un filtro Pasa Bajos

Como su nombre lo indica, este filtro hacer que pase las frecuencias bajas, y las altas las elimina o suaviza bajando la amplitud o intensidad. Aplicando el filtro y sacando la transformada de Fourier :

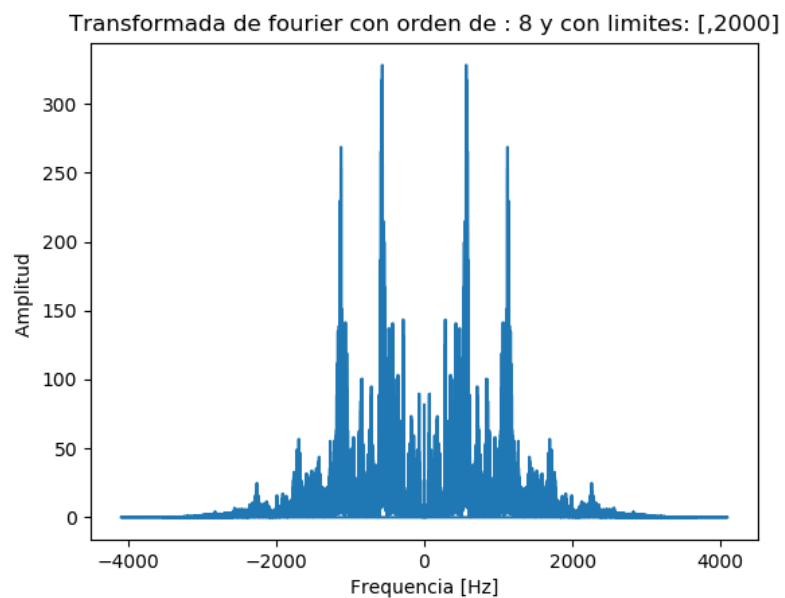


Figura 4: Transformada de Fourier para el filtro pasa bajos.

Y con su espectrograma:

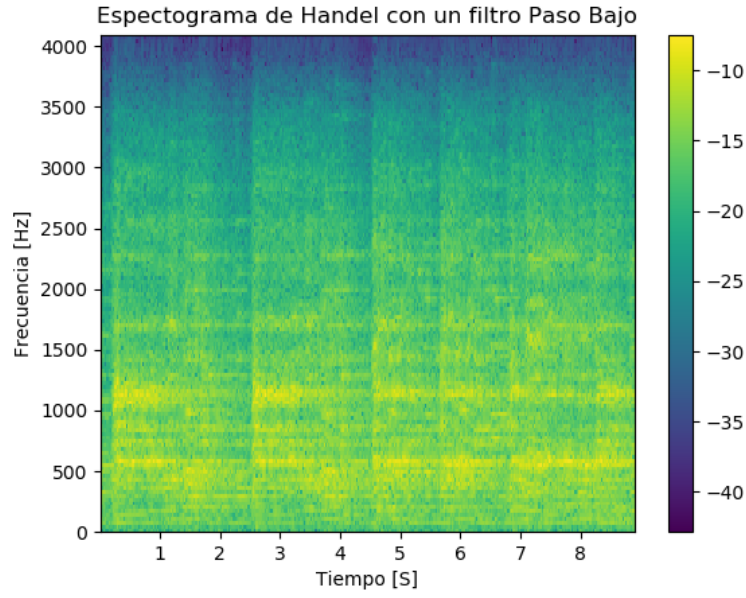


Figura 5: Espectrograma para un filtro de paso Bajo

3.3. Obtención del espectro con un filtro Pasa Altos

Como su nombre lo indica, este filtro hacer que pase las frecuencias altas, y las bajas las elimina o suaviza bajando la amplitud o intensidad. Aplicando el filtro y sacando la transformada de Fourier :

Y con su espectrograma:

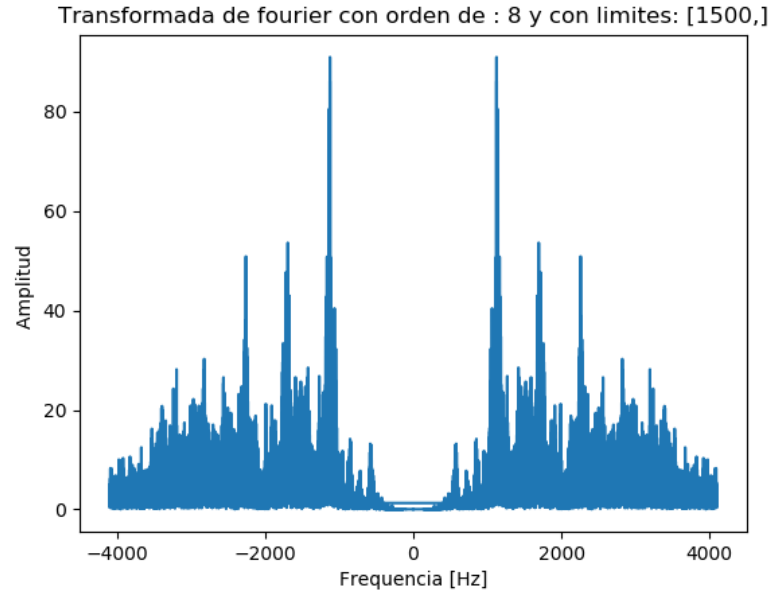


Figura 6: Transformada de Fourier para el filtro pasa Altos.

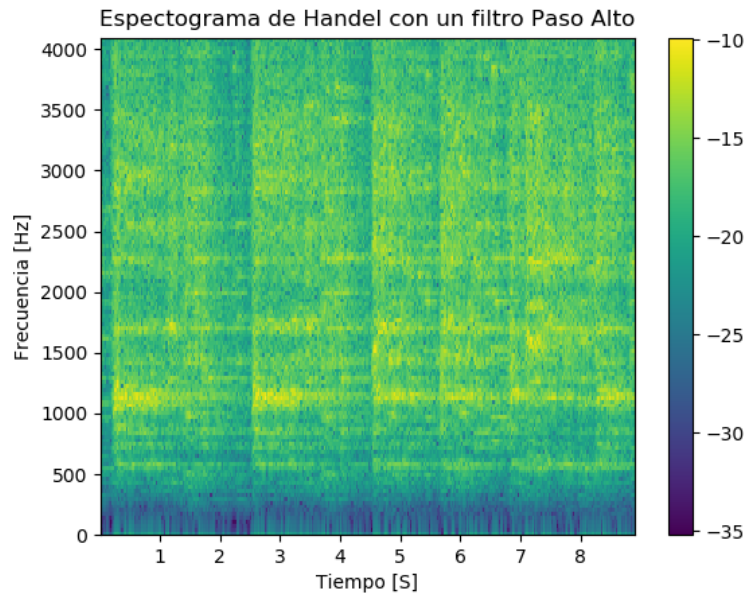


Figura 7: Espectrograma para un filtro de pasa Altos

3.4. Obtención del espectro con un filtro Pasa Banda

Este filtro hacer que las frecuencias por debajo del limite inferior y por encima del limite superior se eliminen o se suavicen. Aplicando el filtro y sacando la transformada

de Fourier :

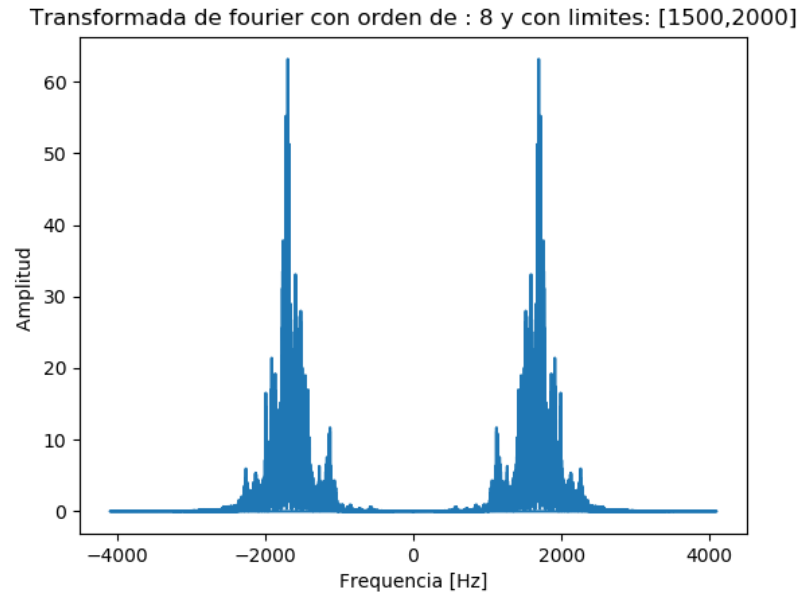


Figura 8: Transformada de Fourier para el filtro pasa banda.

Y con su espectrograma:

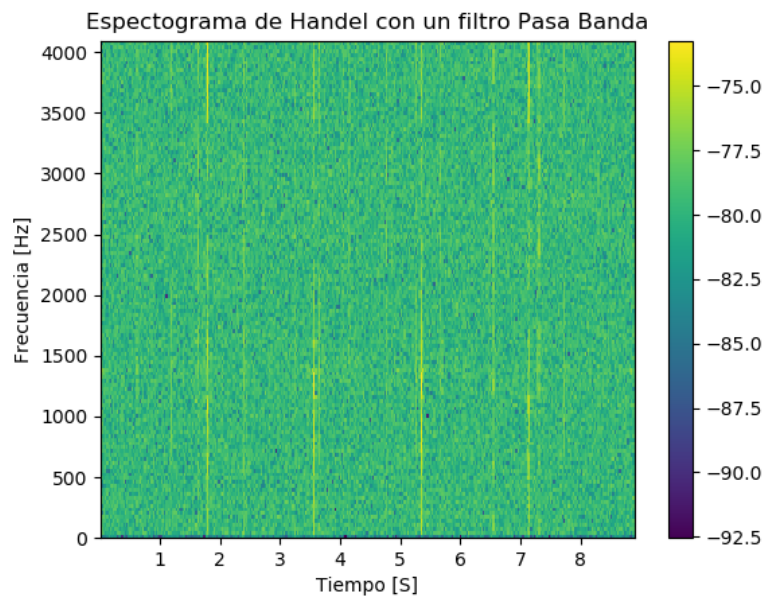


Figura 9: Espectrograma para un filtro de pasa banda

3.5. Obtención del nuevo archivo .wav

Como última parte es necesario explicar el cómo se vuelve a obtener un archivo de audio .wav ya que parte importar es experimentar como resulta la señal al volverla audio y escucharla. Para hacer esto se necesita ocupar la función *write()* de la *libreriascipy.io.wavfile* a la que se le debe entregar el nombre del archivo resultante con ".wav" incluido, la tasa de muestras (número de muestras obtenidas por segundo del .wav original) y la data que se desea pasar a audio. Es importante destacar que los datos que se le entregan a la función deben ser datos en el dominio del tiempo, es decir, no se le debe entregar los datos resultantes de la función *fft*. Con el uso de la función *write*, se tendrá el audio a escuchar finalmente.

4. Análisis de los resultados

4.1. Espectro amplitud versus tiempo

Analizando la Transformada de Fourier:

La señal obtenida en el dominio de la frecuencia (Hz) entrega información importante sobre cuales frecuencias son relevantes, en este caso las frecuencias importantes se encuentran en las bajas frecuencias, esto se debe a que ahí podemos notar los picos mas altos. Cabe mencionar que esta señal esta normalizada, es decir, que su amplitud esta a una escala mas pequeña. Respecto a los picos mas altos, indica que la señal es mas lenta.

Analizando el Espectrograma:

Viendo el espectrograma se puede dar cuenta que hay dos partes amarillas bien pronunciadas (Sabiendo que el color amarillo hace referencia a la presencia una mayor intensidad), este concuerda con los picos de amplitud que se pueden ver en la transformada de Fourier. Estas partes amarillas se encuentran en frecuencias muy bajas comparadas con la frecuencia de 4000 Hz. A partir de estas frecuencias no es posible observar claramente el color amarillo después lo que quiere decir que no hay presencia de intensidades muy altas.

4.2. Espectro de la señal Con un filtro Pasa Bajos

Analizando la Transformada de Fourier:

La transformada de la señal filtra con un filtro en la parte superior de 2000 Hz, dice que todas las frecuencias por encima de los 2000 Hz se irán haciendo 0, por lo que pasaron las frecuencias bajas. Como el filtro superior es demasiado alto, pasaron los dos picos de mayor intensidad por lo que la señal hay conserva sus características mas notables.

Analizando el Espectrograma:

Viendo el espectrograma se puede dar cuenta que las dos partes amarillas se conservan pero las frecuencias altas cambian su color tomando un azul oscuro dando la información de que no hay presencia de frecuencias altas.

Si se escucha el audio con la señal filtrada por pasa Bajos, se puede dar cuenta que la señal se limpió un tanto del ruido de fondo (ya no existe eco), sin embargo ha bajado

su volumen, por lo que se puede escuchar el ALELUYA pero como si estuviera lejos.

4.3. Espectro de la señal Con un filtro Pasa Altos

Analizando la Transformada de Fourier:

La transformada de la señal filtra con un filtro en la parte inferior de 1500 Hz, dice que todas las frecuencias por debajo de los 1500 Hz se irán haciendo 0, por lo que pasaron las frecuencias altas. En este caso la transformada da la información que existe un pico de amplitud, sin embargo el gráfico es engañoso ya que se encuentra a una escala inferior a la transformada original. Por lo que efectivamente pasaron las frecuencias altas y los picos de intensidad desaparecieron y solo quedó uno suavizado.

Analizando el Espectrograma:

Viendo el espectrograma se puede dar cuenta las dos grandes partes amarillas no se encuentran, y en su haber existe un vacío con un color azul. Sin embargo existen partes amarillas aun en la parte superior del gráfico diciendo que han sobrevivido las frecuencias altas.

Si se escucha el audio con la señal filtrada por pasa Altos, se puede dar cuenta que se escucha el ALELUYA pero demasiado lejos pero de forma mas aguda que la original. El audio se escucha como si a lo que le faltó al audio de del filtro de paso bajo, se encontrara en este.

4.4. Espectro de la señal Con un filtro Pasa Banda

Analizando la Transformada de Fourier:

La transformada de la señal filtra con un filtro en la parte inferior de 1500 Hz y en la parte superior de 2000 Hz, dice que todas las frecuencias por debajo de las de 1500 se anulan y de que las por encima de los 2000 Hz se irán haciendo 0, por lo que pasaron las frecuencias que estaban entre esos rangos. Como el filtro superior es demasiado acotado, pasaron frecuencias o tan representativas ya que si bien se ve un pico de intensidad, este se encuentra a una escala menor.

Analizando el Espectrograma:

Viendo el espectrograma se puede dar cuenta de una homogeneidad en el color, predominando el verde. Aun que el gráfico es engañoso ya que no se encuentra a la misma escala que los anteriores. Este gráfico da la información que no hay mucha intensidad tanto ni en altas ni en bajas frecuencias. Esto se debe a que el intervalo tomado fue un intervalo bastante pequeño además de que el suavizado fue bastante abrupto por el orden 8.

Si se escucha el audio con la señal filtrada por pasa Bajos, se puede dar cuenta que la señal desaparece y solo se percibe un POP en un instante de segundo. Esto se debe a que la mayor parte de la señal fue borrada y no existe manera de que con solo 500 frecuencias se construya un ALELUYA claro.

5. Conclusiones

En esta experiencia se logró implementar lo solicitado, es decir, filtrar señales. Lo más difícil de haber realizado este trabajo fue la interpretación de los resultados. Al realizar los experimentos se dio cuenta que los intervalos tomados fueron unos intervalos muy malos ya que eran demasiado cercanos y críticos a la hora de filtrar señales.

Gráficamente ayudo bastante el espectrograma ya que con este la visualización de la información corroboraba con lo que se estaba pensando y con lo que se decía de la transformada de Fourier. Los filtros implementados fueron bastante eficaces logrado eliminar las frecuencias que no se querían.

Como se mencionó anteriormente la implementación no fue difícil, de todas formas, surgieron algunos problemas con la versión de Scipy, ya que la antigua no soportaba el parámetro F_s = tasa de muestreo, por lo que se perdió valioso tiempo en descubrir el error de la función butter. Respecto a la escala de los gráficos se tuvo problemas al interpretarlos ya que no saltaba a simple vista que lo rango de los valores no eran correctos por lo que provocaba confusión y no se podían sacar conclusiones adecuadas.

En síntesis, a pesar de todos los problemas surgidos, se pudo hacer la experiencia de forma correcta y con un análisis critico de que pasaba y lo más importante, el por que pasaba. El desarrollo de la experiencia y el análisis de resultados fueron hechos a partir de una base teórica.

Bibliografía

- (2015). Filtros electrónicos. [Online] <http://mer1516bonilla.blogspot.com/2015/10/filtros-electronicos.html>.
- (2019). Enunciado laboratorio 2. [Online] http://www.udesantiagoovirtual.cl/moodle2/pluginfile.php?file=%2F250970%2Fmod_resource%2Fcontent%2F1%2FLaboratorio%201%20Enunciado.pdf.
- (2019). Funciones scipy butter. [Online] <https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.butter.html/>.