

## ***Proyecto Final: Procesamiento Digital de Señales.***

### ***1. Introducción al análisis de vibraciones.***

El análisis de vibraciones es una herramienta que permite la medición de los niveles y frecuencias de vibración de la maquinaria industrial, este análisis permite determinar el estado de una maquina en particular, es decir, se pueden obtener resultados que lleven a tomar decisiones acerca del funcionamiento de la maquinaria, dentro de las decisiones que se pueden tomar teniendo en cuenta un análisis de vibraciones están, si se le hace un mantenimiento o si se sigue monitoreando con más regularidad o si simplemente está dañada y hay que cambiarla por otra.

Las tareas principales del análisis de vibraciones son identificar, predecir y prevenir fallas en las maquinas. La implementación de esta tecnología mejora la fiabilidad en las maquinas, y se logra una mejor eficiencia y una reducción del tiempo de inactividad, eliminando fallas mecánicas o eléctricas. Este procedimiento es muy utilizado en el mundo industrial porque mejora la planeación de las reparaciones de la máquina y logra mantener en funcionamiento la maquina durante el mayor tiempo posible sin fallas inesperadas.

Cuando una maquina industrial está en funcionamiento, genera una vibración. Esta vibración puede ser medida, con la ayuda de un dispositivo llamado acelerómetro (dispositivo mostrado en la **figura 1**), este dispositivo genera una señal de voltaje, que es proporcional a la cantidad de vibración, así como a la frecuencia de vibración o el periodo de vibración.



***Figura 1. Acelerómetro para captar vibraciones.***

Las señales que genera el acelerómetro está conectada a un colector de datos, que registra la señal como una forma de onda en el tiempo (amplitud contra tiempo) o como una Transformada Rápida de Fourier (amplitud contra frecuencia), algunos registran ambos datos.

Estos datos son interpretados por un analista experto o simplemente por un algoritmo “inteligente” que analice las gráficas y sugiera decisiones con base en los datos obtenidos por el acelerómetro.

En esta práctica se hará uso de la Transformada Rápida de Fourier para realizar un análisis en frecuencia de tres diferentes tipos de audios de máquinas que se utilizan diariamente en el ambiente industrial, además se realizara un análisis en tiempo de cada uno de estos audios. Este proceso permite simular lo que se describía anteriormente, que los datos mostrados por un acelerómetro eran luego transferidos a un colector de datos que muestra la gráfica en el dominio del tiempo o en el dominio de la frecuencia o en ambos.

### ***1.1 Aplicación de la Transformada Rápida de Fourier a señales de audio industriales.***

En el curso se ha tratado la transformada de Fourier la cual permite un análisis de las señales en frecuencia, esta transformada permite analizar las diferentes componentes frecuenciales que tiene una señal.

Para este caso se estudiará los datos que puede aportar la Transformada de Fourier para audios de máquinas industriales.

#### ***Pregunta 1.***

***Luego de leer el numeral 1 y el numer 1.1 ¿Qué palabras claves puede resaltar? Escribalo en el Jupyter Notebook.***

1. Importe los audios para que puedan ser analizados en el Jupyter Notebook, en la **figura 2** se muestra un código que le puede ser útil.

```
file_audio_1=('prensa_industrial.wav') # Ruta del archivo con La senal
fs, y=read(file_audio_1) # Cargar el archivo
y = y / np.max(np.abs(y)) # escala la amplitud de la senal
t=np.arange(0, float(len(y))/fs, 1.0/fs) # Vector de tiempo
plt.plot(t,y) # Dibujar la grafica
plt.grid()

# Los siguientes dos comandos dibujan las etiquetas de los ejes
plt.title("Grafica en el tiempo del audio 1",fontsize = 18)
plt.xlabel('Tiempo (segundos)') # Etiqueta eje X
plt.ylabel('Amplitud') # Etiqueta eje Y
plt.show() # Mostrar la grafica
print("La frecuencia de este caso es: ",fs)
Audio(y, rate=fs) # para escuchar la senal
```

***Figura 2.*** Posible implementación para cargar y graficar las señales de audio.

#### ***Pregunta 2.***

***De la gráfica obtenida de cada audio en el dominio del tiempo. ¿Se puede decir que son señales periódicas? Y si son periódicas explique, con respecto a la gráfica, como se puede intuir que son periódicas.***

2. Para cada señal de audio, calcule la transformada rápida de Fourier, para esto puede implementar la función definida en la *figura 3*.

```
### Primero definamos la funcion
def dftmatrix(N, Nfft):
    #construct DFT matrix
    k= np.arange(Nfft)
    if N is None: N = Nfft
    n = np.arange(N)
    U = np.matrix(np.exp(1j* 2*np.pi/Nfft *k*n[:,None]))
    return U/np.sqrt(Nfft)
```

*Figura 3. Función sugerida para calcular la FFT.*

Note que la función recibe un N que representa el tamaño de la señal y un valor de NFFT, itere con valores diferentes de NFFT (al menos dos, uno mayor al otro).

### ***Pregunta 3.***

***Cuando aplico la NFFT con un número menor. ¿Qué noto de diferente entre las gráficas generadas?***

Como se explicó en la sesión de introducción, el colector de datos que se conecta al acelerómetro, muestra la grafica en el dominio del tiempo y el espectro en frecuencia de la señal, según el cambio del voltaje que muestra el acelerómetro debido a las vibraciones.

Para poder realizar una analogía lo que se planteó en esta práctica fue analizar con señales de audio, las cuales también capturan la perturbación en el aire debido a las vibraciones que generan las maquinas, y con base a estos audios generar las gráficas en el dominio del tiempo y en el dominio de la frecuencia, y analizar a partir de estas gráficas el funcionamiento de las maquinas.

Basado en lo leído en la introducción y en el párrafo anterior, analice y responda en el Notebook las siguientes preguntas:

### ***Pregunta 4.***

***Si usted fuera analista de las gráficas de tiempo y de frecuencia, ¿Cómo intuiría que alguna maquina tiene alguna anomalía?***

### ***Pregunta 5.***

***¿Cree usted que todas las maquinas industriales tienen un comportamiento periódico?***

### ***Pregunta 6.***

***Si usted fuera analista de graficas en el dominio de la frecuencia ¿Consideraría correcto tener un rango de tolerancia de error en las frecuencias para analizar un espectro de una maquina determinada?***

## 2. Introducción a filtros FIR en la música.

En la música existen muchas maneras de hacer que un sonido llegue a un tono requerido para lograr transmitir a los oyentes una sensación en específico, el sonido es una señal que viaja a través del espacio y la captamos con nuestros oídos, esta señal tiene propiedades como frecuencia, periodo, fase, entre otras propiedades que son características de las señales.

Para lograr que un sonido tome la forma que el productor requiera, la manera más común es implementar un ecualizador (dispositivo físico mostrado en la **figura 4**) el cual es un dispositivo que va filtrando las frecuencias de ese sonido, lo cual hace que el sonido tome una forma distinta a la original, el ecualizador puede verse no como un solo filtro, sino como un dispositivo que es capaz de aplicar varios filtros al mismo tiempo, lo que hace mucho más versátil el cambio de forma del sonido, los ecualizadores pueden ser físicos o también digitales (interfaz digital de un ecualizador mostrado en la **figura 5**), los ecualizadores digitales se pueden encontrar dentro de los programas para hacer música, por ejemplo FL Studio o Protools, entre otros.



**Figura 4.** Ecualizador físico o analógico.



**Figura 5.** Interfaz de un ecualizador digital.

## 2.1 Análisis en el dominio del tiempo y de la frecuencia de los audios musicales.

1. Importe los audios y gráfíquelos en el dominio del tiempo, debe tener en cuenta escuchar los audios y tener muy claro los sonidos que escucha, para graficar y escuchar los audios puede utilizar el código mostrado en la **figura 2**.

### **Pregunta 7.**

**¿Qué sonidos le parecen interesantes en cada audio?**

2. Luego de hacer esto, realice la Transformada de Fourier para cada uno de los audios anteriormente cargados, puede tomar como guía el siguiente trozo de código, mostrado en la **figura 6**.

```
# Obtener la transformada de Fourier de los datos de audio
transformada_fourier = np.fft.fft(b)

# Calcular las frecuencias correspondientes a los coeficientes de la transformada
frecuencias = np.fft.fftfreq(len(b), 1 / fs)

# Obtener el espectro de amplitud (magnitud) normalizado
espectro_amplitud = np.abs(transformada_fourier) / len(b)

# Crear una figura para mostrar el espectro de frecuencia
plt.figure(figsize=(10, 6))
plt.plot(frecuencias[:len(frecuencias)], espectro_amplitud[:len(frecuencias)])
plt.title('Espectro de frecuencia del audio musical 1', fontsize = 18)
plt.xlabel('Frecuencia (Hz)')
plt.ylabel('Amplitud')
plt.grid()
plt.show()
```

**Figura 6.** Trozo de código para la Transformada de Fourier.

3. Tenga en cuenta las figuras generadas en el numeral anterior, y considerando el espectro de frecuencia de cada audio, diseñe un filtro FIR para cada uno, el diseño del filtro es libre, la idea es que sea un filtro haga las veces de un ecualizador y logre modificar el audio resaltando los sonidos que describió en la **Pregunta 7**. Haga una gráfica de la respuesta en frecuencia del filtro diseñado, puede guiarse de las líneas de código mostrada en la **figura 7**.

```
# Frecuencia de corte del filtro pasa altas (en Hz)
frecuencia_corte = 5000

# Orden del filtro FIR
orden_filtro = 101

# Crear la respuesta de impulso del filtro FIR (pasa altas)
respuesta_impulso = signal.firwin(orden_filtro, frecuencia_corte, fs = 44100, pass_zero=False)

# Calcular la respuesta en frecuencia del filtro FIR
frecuencias, respuesta_frecuencia = signal.freqz(respuesta_impulso, 1, fs = 44100)

# Graficar la respuesta en frecuencia
plt.figure(figsize=(10, 6))
plt.plot(frecuencias, 20 * np.log10(np.abs(respuesta_frecuencia)))

plt.title('Respuesta en frecuencia del filtro FIR', fontsize = 18)
plt.xlabel('Frecuencia (Hz)')
plt.ylabel('Amplitud (dB)')
plt.grid()
plt.show()
```

**Figura 7.** Código de ayuda para generar la respuesta en frecuencia del filtro.

***Pregunta 8.***

***Describa brevemente como diseño el filtro a partir del espectro de frecuencia.***

***Pregunta 9.***

***¿Qué puede decir acerca de la gráfica de diseño del filtro? ¿Quedo con lóbulos en los extremos?***

4. Luego de filtrar cada uno de los audios, escúchelos, para esto puede implementar las líneas de código mostrada en la ***figura 8***.

```
# Aplicar el filtro FIR al audio
audio_filtrado = signal.lfilter(respuesta_impulso, 1, b)

Audio(audio_filtrado, rate=fs) # para escuchar la señal
```

***Figura 8.*** Código de ayuda para filtrar el audio y escuchar el audio filtrado.

***Pregunta 10.***

***Luego de escuchar el audio filtrado ¿Fue posible hacer los cambios que deseaba en cada audio?***

***3. Conclusiones.***

Agregue una sesión de conclusiones generales en el Notebook.