

Guía Laboratorio 3

Procesamiento Digital de Señales

Paula Pérez, Alejandro Escobar, y Cristian Ríos

2024-1

- Descargar el informe del laboratorio con el siguiente nombre: *Lab3_PDS_Apellido_Nombre.ipynb*
- Enviar junto con el informe los archivos adicionales generados y descargados. Todo esto debe ir en un archivo comprimido con el siguiente nombre: *Lab3_PDS_Apellido_Nombre.zip*
- OJO! Recuerde tener cuidado con la indentación y caracteres como el guion bajo y las llaves cuando copie y pegue el código entregado en esta guía.
- Las preguntas deberán ser resueltas en el notebook, indicando sus respectivos numerales.

1. Introducción

En este laboratorio se observará el concepto de correlación, la cual es una medida de similitud entre dos series de tiempo en función del retraso de una respecto a la otra. Esta es definida para dos secuencias en tiempo discreto como:

$$R_{xy}[k] = \sum_{m=-\infty}^{\infty} x[m] y[m-k] \quad (1)$$

2. Análisis de pulsos de radar

Una de las aplicaciones clásicas de la correlación se centra en los sistemas de radar aeronáuticos. En estos sistemas, un transmisor envía un pulso electromagnético de forma conocida, el cual es reflejado por alguna aeronave y esta reflexión es captada por un receptor. La señal recibida es una copia retrasada y con ruido del pulso original. Para detectar el pulso original en esta señal, se calcula la correlación entre la señal recibida y el pulso original, allí donde se encuentre el máximo de la señal de correlación será la ubicación del pulso en la reflexión.

Otra aplicación de este mismo concepto es en las comunicaciones digitales, en este caso, un pulso que representa un símbolo (conjunto de bits) pasa por un canal que lo distorsiona. En el receptor se realiza la correlación con la forma original del pulso para encontrar el símbolo adecuado.

Para esta práctica se debe usar la función adjunta en el archivo *utils.py*, esta función nos permitirá simular la distorsión de la señal. No olvide incluirlo en la misma carpeta que tiene el jupyter-notebook de la práctica e importarla.

1. Genere un pulso rectangular de 35 muestras. Tip: Utilice la función de numpy *np.ones(35)*.
2. Calcule y grafique la función de autocorrelación del pulso generado.

```
Rxx=np.correlate(rect_pulse, rect_pulse, mode='full')
tau=np.arange(-len(rect_pulse)+1, len(rect_pulse), 1) # vector de retraso
```

rect_pulse es el vector con el pulso rectangular creado en 1, *Rxx* es la función de autocorrelación resultante y *tau* es el vector de retraso.

¿Qué puede concluir de la gráfica obtenida?

3. Utilice la función *delay_noise* incluida en la librería adjunta '*utils.py*' para simular el ruido introducido por el canal, y el desfase de la señal recibida respecto a la original. Grafique el resultado, ¿Es capaz de distinguir el pulso y decir cuanto es su retraso?

```
delayed_pulse=delay_noise(rect_pulse)
```

4. Utilice la función de numpy *np.correlate* para hallar la correlación entre la señal distorsionada y la señal original. Muestre el resultado gráficamente ¿Cuál es el retraso? ¿Qué se puede concluir?

Tip 1: Para utilizar la función *np.correlate* utilice el siguiente comando:

```
xcorr=np.correlate(delayed_pulse,rect_pulse)
```

Tip 2: Puede hallar el retraso con las funciones de numpy *np.where* o *np.argmax*.

3. Análisis de Señales de Marcha

El archivo *gaitSignal.txt* contiene señales de varios sensores: acelerómetros, giróscopos y magnetómetros, capturados durante la actividad física de caminar en una cinta caminadora. Cada sensor se encarga de capturar tres señales, una en cada eje: x,y,z. A los participantes se les pusieron 5 sensores, para un total de 45 señales simultáneas. La frecuencia de muestreo es de $25Hz$.

Más información: <https://archive.ics.uci.edu/ml/datasets/Daily+and+Sports+Activities>

Desarrolle un código que cargue la señal adjunta, y seleccione el canal correspondiente al último dígito de su cédula, y finalmente la grafique (no olvide las etiquetas de los ejes).

0. T_zgyro
1. T_zgyro
2. RA_zgyro
3. RA_zgyro
4. LA_zgyro
5. LA_zgyro
6. LL_zgyro
7. LL_zgyro
8. RL_zgyro
9. RL_zgyro

Para cargar este archivo a una matriz puede utilizar la librería *pandas* junto con las siguientes instrucciones:

```
import pandas as pd
csvfile='gaitSignal.txt' # Debe estar en la misma carpeta del .ipybn
f = pd.read_csv(csvfile)
ft=f['LL_xacc']
```

3.1. Procedimiento

En esta sección vamos a determinar el periodo cuando la persona realiza actividades físicas. Para ello se utilizará la autocorrelación de la señal. En la mayoría de señales reales el periodo no se puede determinar directamente de la señal; sin embargo, este se puede estimar analizando la autocorrelación de la señal hallando la diferencia entre picos consecutivos de la señal de auto-correlación.

1. Normalice la señal restándole la media y luego dividiendo por su máximo valor absoluto (Recuerde del laboratorio 2). Luego, grafique la señal asignada en función del tiempo y presente un breve análisis sobre la misma.
2. ¿Cuál es el periodo de muestreo de la señal?
3. Calcule la autocorrelación de la señal obtenida y grafíquela. ¿Qué se puede concluir?
4. Realice un zoom de la autocorrelación entre 0 y 3.

3.2. Medición del periodo de la señal

El periodo de la señal corresponde al tiempo entre 0 y la posición del primer pico de la función de autocorrelación. Ya que es donde se encuentra la máxima similitud de las señales cuando van ocurriendo los desplazamientos

1. Determine e indique el periodo de la señal a partir del tiempo para el primer máximo global de la autocorrelación después de cero.
2. Buscamos solo los valores de tiempo positivo de la autocorrelación (Recuerde que la función es par).
3. Buscamos la posición del primer máximo después del primer pico mínimo (para excluir el máximo en 0).
4. Finalmente encontramos la posición del pico máximo.

4. Conclusiones

Realice conclusiones generales sobre la práctica. Recuerde que las conclusiones son parte fundamental de su evaluación en el laboratorio, tómese el tiempo de pensar las conclusiones.