

Guía Laboratorio 4

Procesamiento Digital de Señales

Paula Pérez, Alejandro Escobar y Cristian Ríos

2024-1

NOTAS:

- Enviar el informe del laboratorio con el siguiente nombre: *Lab4_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: *Lab4_PDS_Apellido_Nombre.zip*

1. Introducción

En este laboratorio se pretende afianzar los conceptos relacionados con el análisis de las series de tiempo, filtro de media móvil y filtro de primera diferencia.

2. Análisis de series de tiempo

2.1. Procedimiento

1. La señal corresponde a datos de la tasa de cambio representativa del mercado (TRM), la cual mide la cantidad de pesos colombianos por un dólar de los Estados Unidos. El archivo *historical_TRM.csv* contiene 6 columnas con la siguiente información:
 - Fecha
 - Precio
 - Apertura
 - Máximo
 - Mínimo
 - Cambio

Para cargar este archivo utilice la librería pandas. Puede ayudarse del siguiente comando:

```
def extract_currency(x):
    if isinstance(x, str):
        return(x.replace('%', '').replace(', ', ''))
    return(x)

df = pd.read_csv('historical\_-TRM.csv', delimiter=',')
#Para analizar su contenido
print(df.head())
senal=np.array(df['Precio']).apply(extract_currency).astype('float')
```

Asegúrese que los datos en el tiempo están organizados de forma ascendente.

2. Utilice la columna *Fecha* y calcule el periodo de muestreo para esta serie de tiempo. Expréselo en días, semanas y años.
3. Grafique la señal *Precio* en función del tiempo y presente un análisis sobre la misma.

4. Grafique la señal *Cambio* en función del tiempo y presente un análisis sobre la misma.
5. Calcule la autocorrelación de la señal *Precio* y la correlación con la señal *Cambio*. ¿Que puede concluir del resultado?
6. Investigue para que se utiliza el test *Dickey-Fuller*, aplíquelo a sus dos señales y concluya. Esto también le servirá de referente para elaborar su conclusión en los próximos numerales. Basease Puede instalar la librería en el notebook con el siguiente comando¹:

```
!pip install statsmodels
```

Tip: La prueba ADF (Dickey-Fuller aumentada) es una prueba de significación estadística, lo que significa que la prueba dará resultados en pruebas de hipótesis con hipótesis nulas y alternativas, se querrá comprobar una hipótesis de no-estacionariedad. Como resultado, tendremos un valor p a partir del cual necesitaremos hacer inferencias sobre la serie temporal, sea estacionaria o no. si el valor p (p-value) es menor a 0.05 existirá estacionariedad, lo que significará que se rechazara la hipótesis.

3. Filtro de media móvil

En muchos casos es útil eliminar ruido aleatorio de alta frecuencia para suavizar la señal de interés y observar el comportamiento más general de la señal. El filtro de media móvil (Moving Average, *MA*) es un filtro que permite realizar este proceso. En este caso se utilizará para suavizar las señales.

Este filtro tiene la siguiente ecuación en diferencias:

$$y[n] = \frac{1}{L} \sum_{i=0}^{L-1} x[n-i] \quad (1)$$

donde L es el número de puntos usados (orden del filtro), $x[n]$ es la señal de entrada y $y[n]$ es la señal de salida. Para generar los coeficientes de un filtro de media móvil con L puntos, se puede utilizar el siguiente código:

```
coefs=np.ones(L)/float(L)
```

Este filtro tiene función de transferencia:

$$H(z) = \frac{1}{L} \sum_{i=0}^{L-1} z^{-i} = \frac{1}{L} \cdot \frac{1 - z^{-L}}{1 - z^{-1}} = \frac{1}{L} \cdot \frac{z^L - 1}{z^{L-1}(z - 1)} \quad (2)$$

Para representar el denominador, el numerador y la respuesta natural de esta función de transferencia en Python se puede utilizar el siguiente código:

```
num=np.zeros(L)
num[0]=1.0/L
num[-1]=-1.0/L
den=np.zeros(L)
den[0]=1
den[1]=-1
```

Los ceros están determinados por las raíces de la ecuación $z^L - 1 = 0$, es decir, $z^L = 1$. Las raíces se pueden encontrar con base en que $e^{j2\pi k} = 1$ para cualquier k entero. Las raíces serán:

$$z_k = e^{j2\pi k/L} \quad (3)$$

¹<https://www.statsmodels.org/dev/generated/statsmodels.tsa.stattools.adfuller.html>

3.1. Procedimiento

1. Diseñe un filtro de media móvil con 3 diferentes valores de L en el rango de 5 a 70, calcule la respuesta de la señal *Maximo* (obtenida en el punto 2.1.6) ante dicho filtro y grafique la señal resultante usando subplots (se puede utilizar el vector de tiempos generado anteriormente). ¿Con cual considera que se visualiza y se podría analizar mejor la información?

Tip: Para calcular la salida se usa la convolución:

```
senal_smooth=np.convolve(senal, coefs, mode='same')
```

Donde *senal* es el vector con la señal. ¿Por qué se debe usar `mode='same'`?

2. Importe las funciones del archivo *ztrans.py* incluido con esta guía (asegúrese de ubicarlo en la misma carpeta del Notebook), las cuales calculan la respuesta en frecuencia y fase, respuesta al escalón, y al impulso

Tip: Para hacerlo utilice el siguiente comando:

```
from ztrans import *
```

3. Utilice la función *impz* para graficar y analizar la respuesta al impulso y al escalón del filtro de media móvil. ¿Qué me indica la respuesta al impulso? Compare los resultados con $L = 6$ y $L = 25$. Tip: para usarla utilice el siguiente comando:

```
impz(num, den)
```

4. Utilice la función *mfreqz* para graficar y analizar la respuesta en frecuencia del filtro de media móvil. ¿Qué tipo de filtro es? ¿Cómo es su respuesta en fase? Compare los resultados con $L = 6$ y $L = 25$.

Tip: para usarla utilice el siguiente comando:

```
mfreqz(num, den)
```

4. Filtro de primera diferencia

En muchos casos es útil analizar la tasa a la que cambia alguna variable. Para una señal continua se utiliza la derivada. En el caso de señales en tiempo discreto se utiliza el filtro de primera diferencia, el cual tiene como ecuación en diferencias:

$$y[n] = x[n] - x[n - 1] \quad (4)$$

Los coeficientes de este filtro se pueden generar con la siguiente instrucción:

```
difcoefs=[1, -1]
```

Y su función de transferencia es:

$$H(z) = 1 - z^{-1} = \frac{z - 1}{z} \quad (5)$$

Para representar el denominador y el numerador de esta función de transferencia en Python se puede utilizar el siguiente código:

```
num=[1, -1]
den=[1, 0]
```

1. Mediante la función de convolución utilizada anteriormente, calcule la derivada de la señal con el filtro de primera diferencia, grafique y analice brevemente la señal resultante. ¿Qué me indica esta señal?

2. Utilice la función *impz* para graficar y analizar la respuesta al impulso y al escalón del filtro de primera diferencia. ¿Qué me indica la respuesta al impulso?
3. Utilice la función *mfreqz* para graficar y analizar la respuesta en frecuencia del filtro de primera diferencia. ¿Qué tipo de filtro es? ¿Cómo es la respuesta de fase?

5. Conclusiones

Realice conclusiones generales sobre la práctica.