Lab: Convolución
Tratamiento de señales

Convolución lineal

- Borramos todas las variables y cerramos todas las figuras: `clear; close all;`
- Construimos la señal de entrada y la respuesta al impulso: `x=1:4;h=[2,4,5,3];`
- Dibujamos ambas señales: `subplot(2,1,1);stem(x);subplot(2,1,2);stem(h);`
- Calculamos la convolución lineal: `yl1=conv(x,h);`
- El producto de convolución es conmutativo, ¿puedes comprobarlo?
- ¿Qué dimensiones tienen las señales `x,h,l`?
- Calcula la convolución lineal de otra forma: `yl2=filter(x,1,h);`
- ¿Qué dimensión tiene `yl2`? ¿Coincide parcialmente con `yl1`?
- Prueba haciendo zero-padding: `yl2=filter(x,1,[h,0,0,0]);`
- Construye[1] la matriz de Toeplitz H_l de las dimensiones apropiadas y calcula la convolución lineal de una tercera manera multiplicando H_l por una versión rellenada con ceros de x.
- Haz una función `y=convlineal(x,h)` que utilice el método de la matriz de Toeplitz para calcular la convolución lineal de sus argumentos.

Convolución circular

- Calcula[2] la convolución circular: `yc1=cconv(x,h,4);`
- Comprueba que la convolución circular es simétrica.
- ¿Qué dimensiones tiene `yc1`?
- Calcula una segunda versión utilizando la DFT: `yc2=ifft(fft(x).*fft(h));`
- Construye la matriz circulante H_c de las dimensiones apropiadas y calcula la convolución circular: `yc3=Hc*x';`
- Compara los vectores obtenidos por los tres métodos.
- Haz una funcion `y=convcircular(x,h)` que utilice el método de la matriz circulante para calcular la convolución circular de sus argumentos.

Calculando una con la otra

- Calcula la convolución lineal de `x,h` utilizando la función `convcircular` y modificando apropiadamente los vectores de entrada y de salida (sin tocar la función por dentro).
- Calcula la convolución circular de `x,h` utilizando la función `convlineal` y modificando apropiadamente los vectores de entrada y de salida (sin tocar la función por dentro).

[1] Busca la ayuda de la orden `toeplitz`
[2] En algunas versiones de matlab será `yc1=conv(x,h,4);`