Adjuntar respuestas, comentarios, gráficas y código pedido en el fichero de respuestas. Hacedlo por GRUPOS, pero entregadlo en una SOLA CUENTA.

Ejercicio 1) Una operación muy usada en Procesado de Señales es la correlación entre dos secuencias x1 y x2 (si las dos secuencias son la misma es lo que se denomina autocorrelación). En esta implementación asumiremos que las dos secuencias son de la misma longitud. La correlación entre x1 y x2 se define como:

$$C[n] = \sum_{k} x_1[k] \cdot x_2[k-n]$$

La correlación mide el parecido entre dos señales x_1 y x_2 para un desplazamiento n. Si n=0 estamos comparando las señales sin desplazar y tendríamos:

$$C[0] = \sum_{k=1}^{N} x_1[k] \cdot x_2[k]$$

La idea es que si x1 y x2 son similares (p.e. ambas positivas o negativas al mismo tiempo) los productos del sumatorio tenderán a ser positivos, se acumularán, y la suma terminará con un valor alto. Por el contrario, si x1 y x2 no están relacionadas, unas veces coincidirán en signo y otras no: el signo de los productos irá cambiando y la suma tenderá a cancelarse, terminando con un valor bajo.

Si n≠0 se compara x1 con una versión de x2 desplazada n muestras para detectar si x2 es una versión desfasada de x1. En ese caso su correlación sin desplazar (n=0) será baja, pero tendremos una alta correlación para n o -n muestras (dependiendo si x2 está adelantada o retrasada). La correlación en n y -n se define como:

$$C[n] = \sum_{k=n+1}^{N} x_1[k] \cdot x_2[k-n] \qquad C[-n] = \sum_{k=1}^{N-n} x_1[k] \cdot x_2[k+n]$$

Fijaros que al desplazar las señales el sumatorio ya no es sobre todas las muestras para no "salirnos" del rango conocido (de 1 a N).

Completar la función correlación dada: `function [C,m]=correlacion(x1,x2,M)`
La función debe calcular la correlación de las secuencias x1 y x2 (de longitud N) para los desplazamientos (desde n=-M a n=M) indicados por el 3er argumento.

La función devuelve un vector C (tamaño 2M+1) con las correlaciones encontradas para los desplazamientos en -M, -M+1, ..., -1, 0, 1, ... M-1, M (el 2º argumento de salida m ya calculado en el código anterior devuelve el vector de desplazamientos). Fijaros que la correlación para n=0 (C[0] en las fórmulas anteriores) se guardará en C(M+1), la de n=-1, C[-1], en C(M), C[1] en C(M+2), etc.

Para probad vuestra función, cread un vector x con 1000 números aleatorios con **`x=randn(1,1000)`** y calculad su autocorrelación haciendo **`[C,m]=correlacion(x,x,3);`** Verificad vuestros resultados con la función de MATLAB **`[C,m]=xcorr(x,x,3)`**. Ambos resultados deberían dar lo mismo. Una vez que os funcione adjuntad el código de vuestra función.

Calculad ahora la autocorrelación de x para M=±200 muestras y mostradla en una gráfica haciendo plot(m,C). Adjuntad gráfica obtenida.

El máximo (indicando alto parecido) en n=0 aparece siempre en una autocorrelación ya que cualquier señal sin desplazar es obviamente idéntica a sí misma. También se verifica que C[n]=C[-n] (la autocorrelación es simétrica). Lo que es peculiar de la autocorrelación de esta señal en que cuanto nos desplazamos un poco (n≠0) el resultado es mucho más bajo. Esto se debe a que a poco que se desplace la señal ya no se parece en nada a la original (por definición, en un ruido aleatorio, cada muestra no está relacionada para nada con sus vecinas).

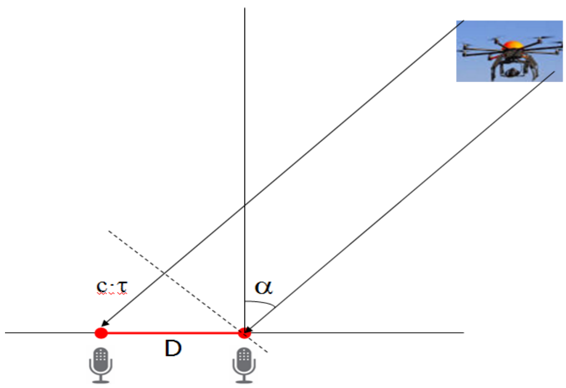
Haced ahora `x=audioread('ecg.wav',[1 1000]);` para cargar en x las 1000 primeras muestras de la señal de un electrocardiograma. Calcular su autocorrelación (de nuevo en ±200 muestras) y volver a hacer su gráfica. Adjuntad gráfica resultante.

Observaréis el mismo máximo en 0 (cualquier señal se parece mucho a si misma), pero a diferencia de antes ahora, para ciertos desfases, la autocorrelación vuelve a mostrar otros máximos (aunque no tan altos como el de n=0). ¿Cada cuántas muestras de desfase se repiten estos nuevos máximos? ¿Por qué suceden? Deducid el correspondiente ritmo cardiaco (en pulsaciones por minuto).

Aplicación: En el 1[er] LAB vimos un ejercicio donde creábamos una señal estéreo poniendo en cada canal una versión adelantada o retrasada de una señal mono. El oído interpretaba la diferencia en el tiempo de llegada de la señal a ambos oídos como una pista sobre el origen de la señal. Si la señal llegaba antes al oído derecho es porque el emisor estaba a nuestra derecha y viceversa.

Cargar los datos contenidos en el fichero 'drone.wav', que correspondonden a un fragmento de audio sacado de un video de YouTube. Escuchando la señal con sound (mejor usando cascos), apreciaréis que la fuente de sonido parece moverse. Al ser una señal estéreo, el desfase de la señal entre los dos micrófonos nos da una pista (sin ver el video) de si el drone está a nuestra derecha o izquierda.

Si somos capaces de estimar el desfase entre ambos canales podremos estimar el ángulo de azimuth formado entre la posición del drone y la dirección perpendicular a la línea que une los micrófonos. Asumiendo que la distancia del drone a los micrófonos es bastante mayor que la separación entre ellos, es fácil demostrar (ver gráfica adjunta) que el ángulo de azimuth cumple la siguiente relación:



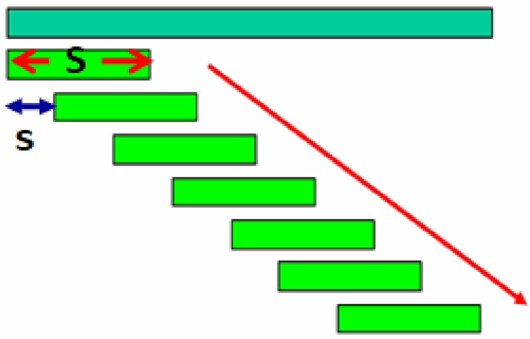
$$\sin(\alpha) = \frac{c \cdot \tau}{D}$$

En esta relación D es la distancia entre micrófonos (en este caso D=0.65 metros), c la velocidad del sonido (340 m/s) y τ el retraso o adelanto de la señal entre ambos canales (en segundos).

En esta tarea usaremos la función de correlación que hemos escrito para determinar el desfase relativo τ de la señal entre ambos canales. Para ello usaremos como señales x1 y x2, los canales derecho e izquierdo de la señal estéreo. La posición del máximo de correlación nos indica cuántas muestras hay que desplazar (en uno u otro sentido) el segundo canal para que se alinee lo mejor posible con el primero. Recordad que usando la frecuencia de muestreo podemos luego pasar de muestras a tiempos (en segundos).

Nuestra función, además de las señales x1, x2 recibía el máximo desplazamiento M para el cual se quería calcular la correlación (desde -M a M muestras). Esto es muy adecuado en este problema porque es posible deducir apriori el máximo desfase que puede existir entre ambos canales para cualquier posición del drone. En este caso el máximo desfase posible es de ~ 85 muestras. Justificad cual es la razón e indicad a qué posición del drone correspondería dicho desfase máximo.

Como el drone se está moviendo, el desfase entre los dos canales irá cambiando, y se trata de estudiar su evolución. La idea es sencilla (ver figura adjunta). En vez de analizar la señal al completo la vamos dividiendo en trozos (de S muestras) y para cada fragmento calculamos el desfase detectando el máximo de la correlación entre los canales. Es habitual usar un cierto solapamiento entre fragmentos, por lo que el salto s entre sucesivos fragmentos será menor que el tamaño del fragmento (s<S).



En este caso trabajaremos con fragmentos de señal de S=8820 muestras y un salto entre ellos de s=S/4=2205. ¿A cuánto tiempo corresponde dicho desplazamiento s entre un fragmento y el siguiente?

Calculad la correlación entre los dos canales al inicio de la señal (con las primeras S muestras de cada canal) y al final (con las últimas S muestras). En ambos casos limitad el cálculo a un desfase máximo de M=85 como se ha indicado. Superponed en la misma gráfica ambas correlaciones (azul para la inicial, roja para la final) en función del desfase m. Adjuntad la figura resultante y justificad si el drone empieza y acaba el video en el mismo lado o en lados distintos del observador.

Para estudiar la evolución del drone a lo largo del tiempo simplemente repetiremos el proceso anterior moviendo el fragmento analizado:

1. Crear un vector rg=(1:S) con el rango de los índices del 1er fragmento.

2. Mientras no nos salgamos de la señal repetir el siguiente bucle:

   a) Extraed en xx el rango de muestras de la señal x indicado por rg y hallar la correlación C entre los dos canales de xx usando un desfase máximo M de 85 muestras.

b) Usando la función max() determinar el valor y la posición del máximo de C. A partir de la posición del máximo de la correlación determinar el desfase (en muestras) donde se ha encontrado el máximo de correlación. Guardar el desfase hallado en un vector de resultados.
c) Sumar al rango rg el salto s para que en la próxima iteración se analice el siguiente fragmento de señal (s muestras más adelante).

3. Tras terminar el bucle haced una gráfica con el desfase en muestras (eje Y) frente al tiempo (eje X) donde suceden. Para los tiempos podéis usar el centro de cada trozo analizado. Adjuntad la gráfica.

4. Convertir el desfase a segundos y usad la relación anterior para calcular los correspondientes ángulos de azimuth (en º). Repetir la gráfica anterior pero ahora con el ángulo de azimuth en el eje Y frente al tiempo en el eje X. Adjuntad la gráfica resultante. ¿Cuál es el máximo ángulo observado en uno u otro sentido? ¿En qué instantes de tiempo el drone cambia de lado respecto al observador?

En 'mosquito.wav' tenéis una grabación (también en estéreo) de un mosquito. Cargad el audio y usar las técnicas anteriores para hacer una gráfica con la evolución del desfase entre ambos canales (en MILISEGUNDOS) frente al tiempo. Adjuntad la gráfica obtenida, indicando el máximo desfase observado (en muestras y en milisegundos). En este caso no sería tan sencillo convertir dicho desfase a un ángulo de azimuth porque seguramente no se verifica que la distancia del mosquito a los micrófonos sea mucho mayor que la distancia entre micrófonos.

Ejercicio 2: Vamos a estudiar el fenómeno que ocurre al combinar dos señales sinusoidales de una frecuencia similar (podéis encontrar algunas demostraciones en https://demos.smu.ca/index.php/demos/waves/99-beat-frequency).En primer lugar veremos el efecto con señales sencillas y luego aplicaremos estos conocimientos a algunos datos reales.

Crear un **vector columna** con una base de tiempos desde 0 a 1 segundo usando una frecuencia de muestreo fs=44100. A partir de ella crear dos sinusoides s1 y s2 usando sin() con frecuencias de f=73 y 77 Hz respectivamente.

Con un **`subplot(211);plot(t,s1,t,s2)`** visualizar estas señales en la parte superior de la figura. Luego, usando **`subplot(211);`** haced un plot de la SUMA de s1 y s2 en función del tiempo en la parte inferior. Adjuntad la figura resultante. ¿A qué son debidos los cambios de amplitud observados en la señal suma? ¿Cada cuánto se repiten los máximos (o los mínimos) de amplitud?

Justificad por qué la distancia entre máximos de amplitud (el periodo de la "beat frequency") corresponde al inverso de la diferencia de las frecuencias involucradas $T=1/(f_1-f_0)$. Escribid vuestra demostración en un papel, y adjuntad una foto.

--------------- Aplicación -----------------

Esta oscilación de volumen que se aprecia al sumar dos frecuencias similares se puede usar para afinar un instrumento. Al escuchar una nota, poca gente es capaz de decir que es incorrecta porque su frecuencia es de 222 Hz en vez de los 220 Hz esperados. Por el contrario, si la escuchamos al mismo tiempo que una frecuencia de referencia, es fácil apreciar la oscilación que nos indica que el instrumento no están vibrando exactamente a la misma frecuencia que el patrón de referencia.

El fichero 'diapason.wav' contiene una grabación de 8 segundos de un diapasón que vibra a 440 Hz (frecuencia standard para afinar instrumentos musicales). Cargar los datos en una señal x usando audioread. Escuchar el sonido con sound(x,fs)

Crear una base de tiempos t (como un **vector columna**) con la misma frecuencia de muestreo fs y la misma longitud que x. A partir de esa base de tiempos crear una oscilación (sin) con una frecuencia de 440 Hz (frecuencia nominal del diapasón) y una amplitud A=0.1. Para escuchar ambas señales (original y "sintética") de forma simultánea, sumadlas y escuchar el resultado con sound(). Debéis apreciar una oscilación del volumen de la señal que indica que no son exactamente iguales. Haced un plot de la suma de las señales en función del tiempo t. Adjuntar la gráfica.

Medir la distancia T (en segundos) entre dos mínimos de amplitud de la señal suma (hacia el centro de la señal). Como se indicó antes, esta distancia T está relacionada con la diferencia de las frecuencias ($\Delta f=1/T$). Usando esta relación, determinar la discrepancia entre la frecuencia del diapasón y la de vuestra sinusoide.

Una vez conocida la discrepancia, podemos modificar la frecuencia de vuestra sinusoide para intentar eliminar la diferencia y repetir el "experimento". ¿Qué frecuencia habéis tenido que usar para que deje de escucharse la oscilación? Adjuntar la gráfica de la suma una vez modificada la frecuencia de la sinusoide.

Vamos a determinar la frecuencia del diapasón usando el análisis de Fourier. Extraed un fragmento de S=8820 muestras de la señal del diapasón alrededor del centro de la señal. Aplicad a dicho fragmento la función serie_fourier() que vimos en el LAB anterior para calcular los primeros 200 coeficientes de Fourier $\{a_k\}$ y $\{b_k\}$ (además del coeficiente a0 o media de la señal que debe tener un valor ~0).

¿Cuál es la duración (en segundos) del fragmento de señal analizado?

Haced un plot de (a^2+b^2) usando el modificador 'ro:'. Adjuntad la gráfica. ¿Para qué índice k aparece un claro máximo en la gráfica anterior? ¿Cuál es la frecuencia f_k del armónico asociado a ese índice?

Repetir la gráfica anterior pero ahora usando las frecuencias de los armónicos (f_k) en vez de los índices k en el eje de las X's. Haced un zoom horizontal de la gráfica alrededor del pico y adjuntad una captura. En base a esta gráfica, justificad cual podría ser la frecuencia real del diapasón.