FACULTAD
DE INGENIERIA
Universidad de Buenos Aires

86.09 - Procesos estocásticos

Segundo cuatrimestre 2023
Grupo 9

Nombres:	E-mail:	Padrón
Llambí Tomás Federico	tllambi@fi.uba.ar	102074
De lucia Lautaro	ldelucia@fi.uba.ar	100203
Curti Chavero María Paz	mcurti@fi.uba.ar	100745
Abbenda de Oto Franco Luciano	fabbenda@fi.uba.ar	101048

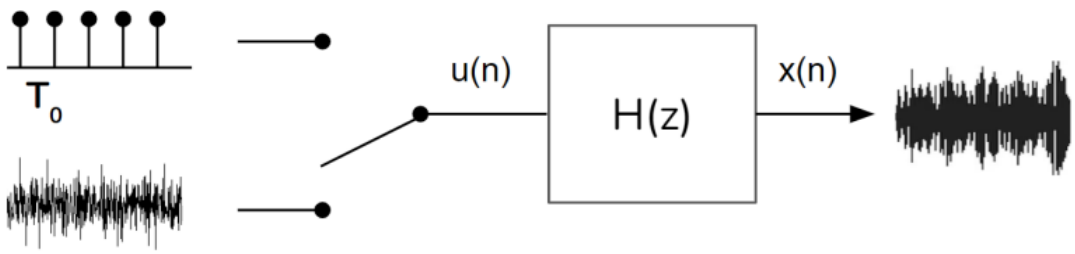
TP2 - LINEAR PREDICTIVE CODING

Explicación de la Consigna

- **Introducción**

LPC es una técnica que busca reproducir el habla humana. basandose en el modelado de el tracto vocal de la persona como un filtro $H(z)$ cuyos coeficientes se extraen de una muestra de audio de la voz a modelar.

- **Modelo**



Las vocales se modelan como un trén de impulsos y las consonantes como ruido blanco. El tren de impulsos representa la excitación periódica de las cuerdas vocales durante la producción de sonidos vocálicos como "a" o "e". Mientras que el ruido blanco representa el "ruido" de alta frecuencia que producimos al pronunciar una consonante como "f", "s". Por supuesto, el trén de impulsos no tiene por si mismo "sonido de vocal", sino que es el modelo de el tracto vocal $H(z)$ el que se encarga de, para cierto período T_0 de el tren de impulsos, producir una salida $x(n)$ que suene como una "a" o una "e".

Las ecuaciones de el modelo van a ser de la forma:

$$H(z) = \frac{X(z)}{U(z)} = \frac{G}{1 - \sum_{k=1}^{P} a_k z^{-1}}$$

$$x(n) = \sum_{k=1}^{P} a_k x(n-k) + G\,u(n)$$

Notamos que $x(n)$ resulta de una combinación lineal de las muestras anteriores de la señal (término autoregresivo) y la excitación $u(n)$ modulada por una ganancia G.

- **Estimación de G y $\vec{a}$**

Esta claro que lo que queremos conocer son los coeficientes a_k y la ganancia G. Solo con esto alcanza para generar a partir $u(n)$ tren de impulsos o ruido blanco la *voz* $x(n)$.

Como asumimos que $x(n)$ es un proceso auto-regresivo (AR), existe una dependencia temporal donde valores recientes tienen una fuerte influencia sobre los valores actuales. Luego, un valor actual de esta serie temporal puede ser aproximado por una combinación lineal de sus valores anteriores. En suma, es justo asumir que $\hat{x}(n) = \sum_{k=1}^{P} a_k x(n-k)$ es un buen estimador de $x(n)$. Relacionandolo con la definición de arriba, tenemos entonces que $e(n) = x(n) - \hat{x}(n) = \mathrm{Gu}(n)$ es el residuo de esta estimación. Aplicando el criterio de minimización de el MSE, que en este caso implica minimizar la potencia de $e(n)$, podemos llegar a ecuaciones normales de la forma:

$$r(k) = \sum_{i=1}^{P} a_i r(k-i) \; ; \quad k = 1, ..., P$$

$$r(0) = \sum_{i=1}^{P} a_i r(i) + G^2 \; ; \quad k = 0$$

$$R = \begin{bmatrix} r(0) & r(1) & \dots & r(P-1) \\ r(-1) & r(0) & \dots & r(P-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(-P+1) & r(-P+2) & \dots & r(0) \end{bmatrix}$$

$$\hat{\mathbf{a}} = R^{-1}\mathbf{r}$$

Siendo $r(k)$ la autocorrelación para $x(n)$. Podemos despejar G de la segunda ecuación y obtener valores estimados para los coeficientes a_k de la cuarta ecuación.

- **Estimación de el Pitch f_0**

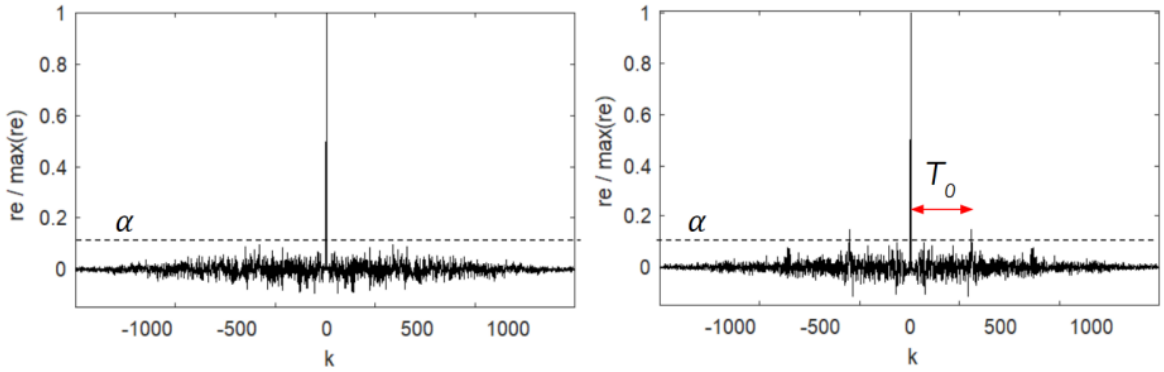
El objetivo de la detección de pitch o tono fundamental (representado como T_0 en el texto) es identificar la frecuencia a la que vibran las cuerdas vocales durante la producción de sonidos sonoros, particularmente durante las vocales en el canto.

Cuando hablamos de señales de voz o canto, hay segmentos en los cuales la voz tiene un tono definido, es decir, un patrón periódico, que en el contexto de procesamiento de señales se relaciona con la existencia de un pitch definido. Estos segmentos corresponden generalmente a las vocales al cantar, ya que las cuerdas vocales vibran de manera periódica durante su producción.

Para recrear la señal de voz con precisión en sistemas de codificación de voz o síntesis, es esencial identificar y mapear estos tonos. La idea es que, al reconstruir la voz, se utilice un tren de impulsos

$u(n)$ con un período T_0 para representar la frecuencia fundamental de la voz en esos segmentos sonoros.

No todos los sonidos tienen un tono. Por ejemplo, los susurros no tienen un tono claro. Así que es importante distinguir entre señales sonoras (con tono) y señales sordas (sin tono o más parecido al ruido). El gráfico muestra dos situaciones: en la señal sorda (izquierda), no vemos un pico prominente aparte del centro, lo que significa que no hay un tono claro. Por otro lado, en la señal sonora (derecha), hay un pico claro aparte del centro, indicando la presencia de un tono.



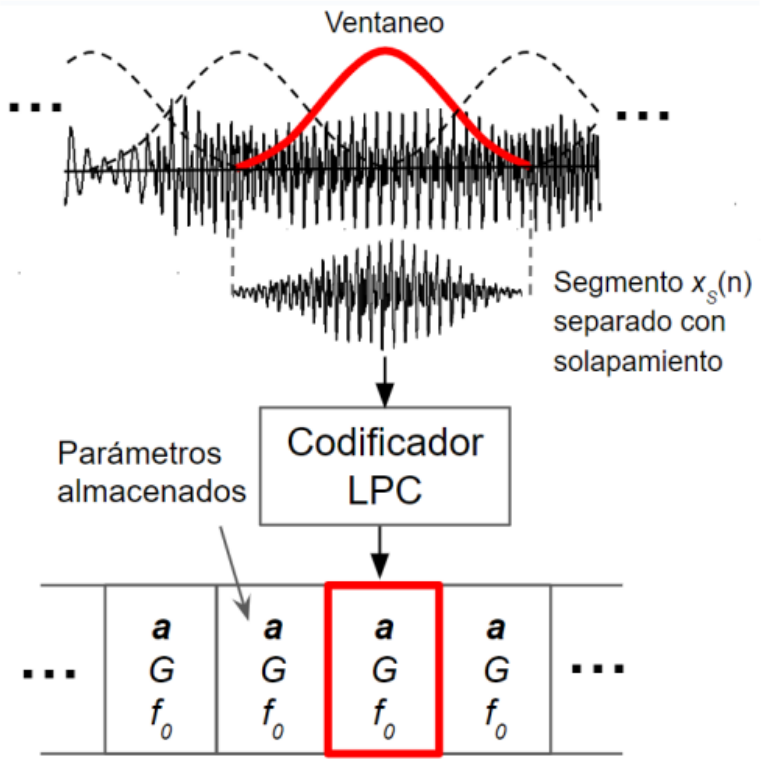
Para decidir si una señal tiene un tono o no, se utiliza un umbral. Si la autocorrelación normalizada excede este umbral en algún punto aparte del centro, se considera que esa señal tiene un tono. En caso contrario, se asume que es más parecido al ruido blanco. Este umbral es una herramienta importante y puede necesitar ajustes dependiendo del contexto.

Existen varios métodos para detectar el pitch, y uno de los más efectivos es el método de Autocorrelación. Básicamente, la autocorrelación mide cómo una señal se parece a sí misma en diferentes puntos en el tiempo. Se busca el patrón repetitivo que representa la frecuencia fundamental del sonido. La fórmula proporcionada muestra cómo calcular este valor utilizando la autocorrelación del residuo de la señal.

- **Codificación de la Señal**

1- La señal de habla original se divide en segmentos cortos (de decenas de milisegundos) mediante la multiplicación por una ventana (e.g ventana de Hamming).

2- Calculamos, usando la metodología que se ilustra más arriba, $\left\{\vec{a}, G, f_o\right\}$ para cada segmento.

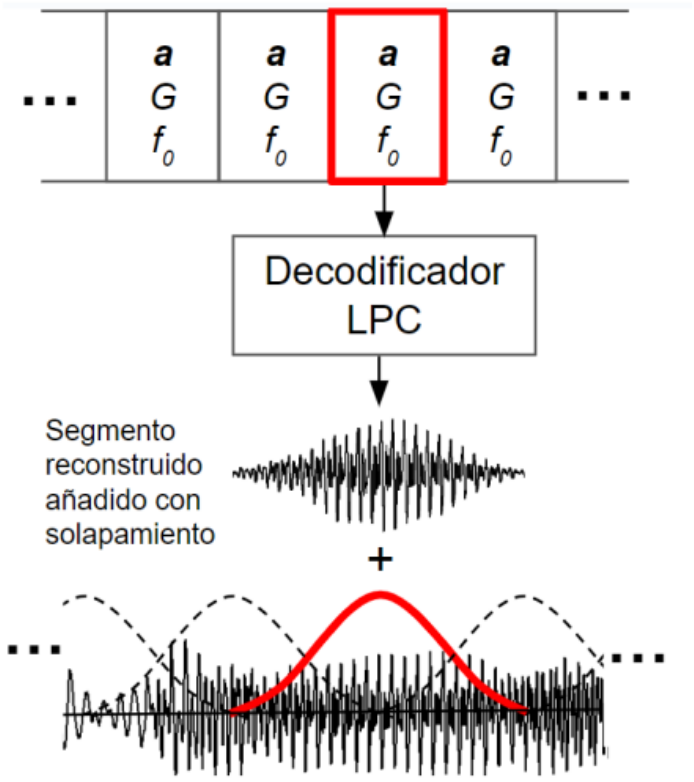


- **De-Codificación de la Señal**

1- Para cada segmento, tomamos $\left\{\vec{a}, G, f_o\right\}$ calculado.

2- Construimos el filtro $H(z)$ utilizando los coeficientes $\vec{a}$ y la ganancia G y construimos la excitación, que puede ser un tren de impulsos con frecuencia f_0 (vocales) o ruido blanco (consonantes).

3- Obtenemos los $x(n)$ para cada segmento y los agregamos (utilizando nuevamente ventaneo para prevenir discontinuidades).



2.1. Ejercicio 1

Siguiendo las definiciones realizadas en la introducción, demuestre las ecuaciones (3) y (4) suponiendo un proceso blanco de entrada $u(n) \sim N(0,1)$.

Por definición, la autocorrelación de un proceso $x(n)$ es de la forma:

$$r(k) = E\left[x(n)x(n-k)\right]$$

El estimador propuesto para $x(n)$ era de la forma:

$$\hat{x}(n) = \sum_{i=1}^{P} a_i x(n-i)$$

Luego, teníamos un residuo $e(n) = x(n) - \hat{x}(n)$, el cual se minimizaba cuando se verifica ortogonalidad:

$$E\left[x(n-k)e(n)\right] = 0$$

Reemplazando $e(n)$ en la ecuación:

$$E\left[x\left(n-k\right)\left(x(n)-\hat{x}(n)\right)\right] = E\left[x(n)x(n-k)\right] - E\left[x(n-k)\sum_{i=1}^{P} a_i x(n-i)\right]$$

$$= r(k) - \sum_{i=1}^{P} a_i E\left[x(n-i)x(n-k)\right] = r(k) - \sum_{i=1}^{P} a_i r(k-i) = 0$$

Finalmente, llegamos a que:

$$r(k) = \sum_{i=1}^{P} a_i r(k-i)$$

En cuanto a la ecuación (4), notamos que:

$$r(k) = E[x(n)x(n-k)] \rightarrow r(0) = E[x(n)x(n)] = E[x^2(n)]$$

Donde

$$x(n) = \sum_{k=1}^{P} a_k x(n-k) + \mathrm{Gu}(n) \rightarrow x^2(n)$$

$$= \left[\sum_{k=1}^{P} a_k x(n-k)\right]^2 + 2\mathrm{Gu}(n)\sum_{k=1}^{P} a_k x(n-k) + [\mathrm{Gu}(n)]^2$$

- Como $u(n)$ es independiente de $x(n-k)$, el término cruzado se anula
- Como $u(n) \sim N(0,1) \rightarrow E[(\mathrm{Gu}(n))^2] = G^2 E[u^2(n)] = \mathrm{GVar}(u(n)) = G^2$

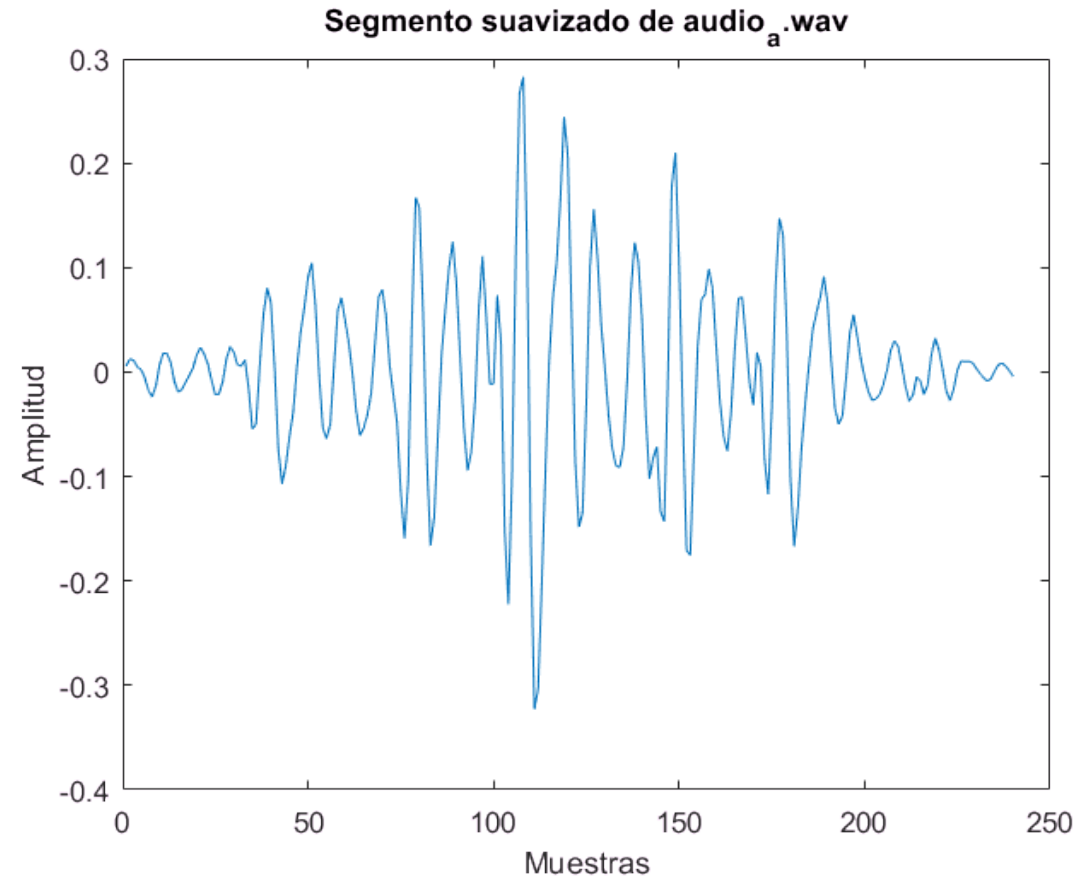
Tenemos entonces que $r(0) = E\left[\left[\sum_{k=1}^{P} a_k x(n-k)\right]^2\right] + G^2$

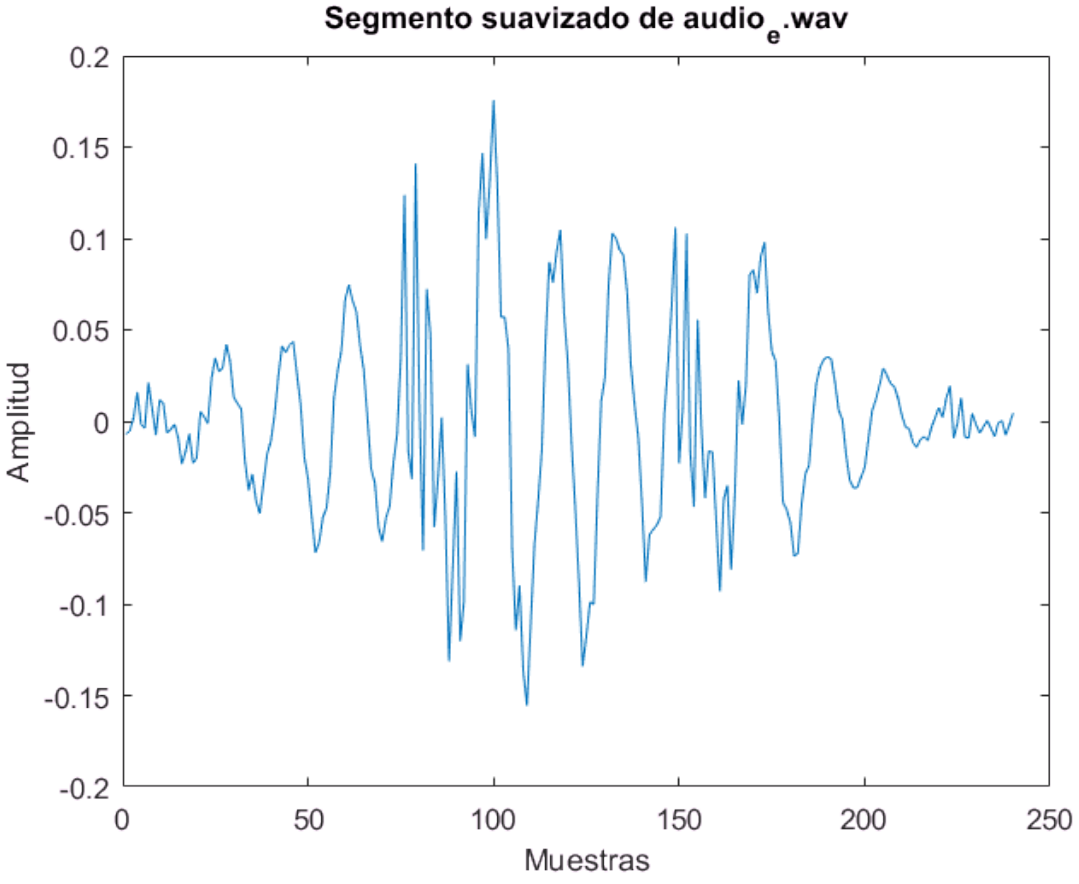
Queda demostrar que $E\left[\left[\sum_{k=1}^{P} a_k x(n-k)\right]^2\right] = \sum_{i=1}^{P} a_i r(i)$

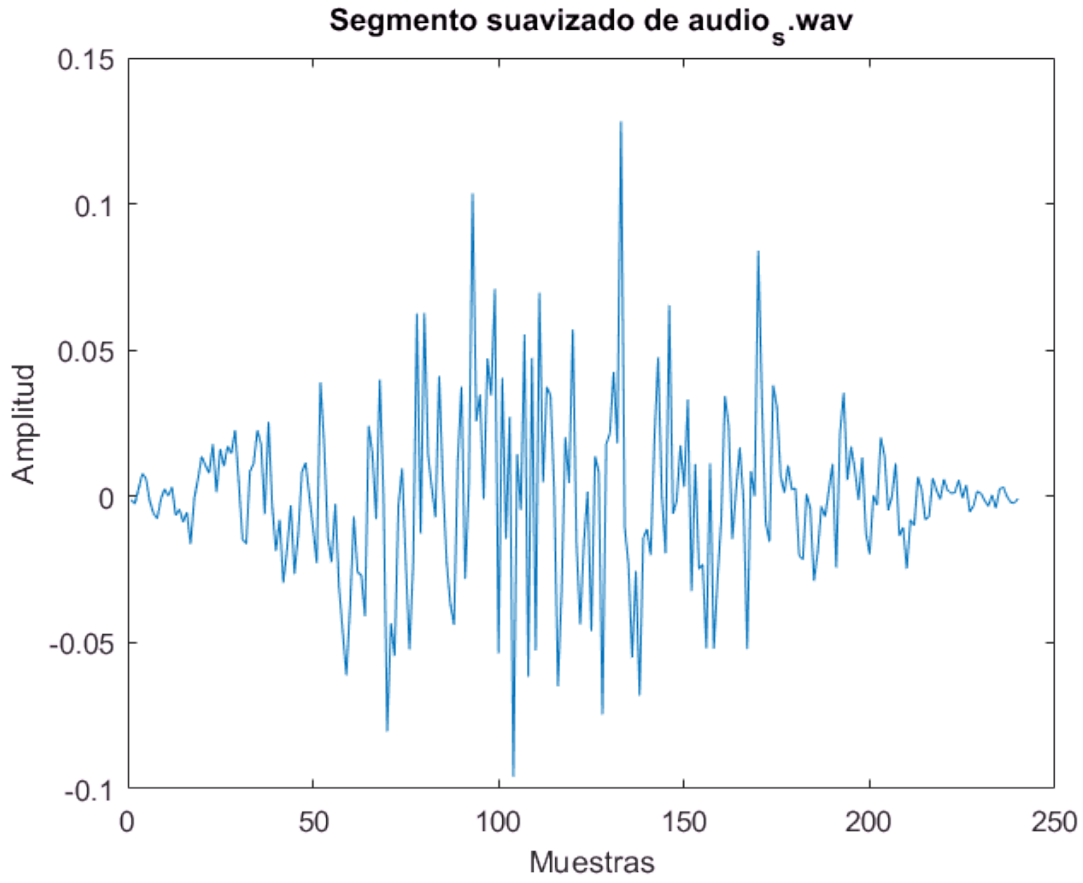
2.2. Ejercicio 2

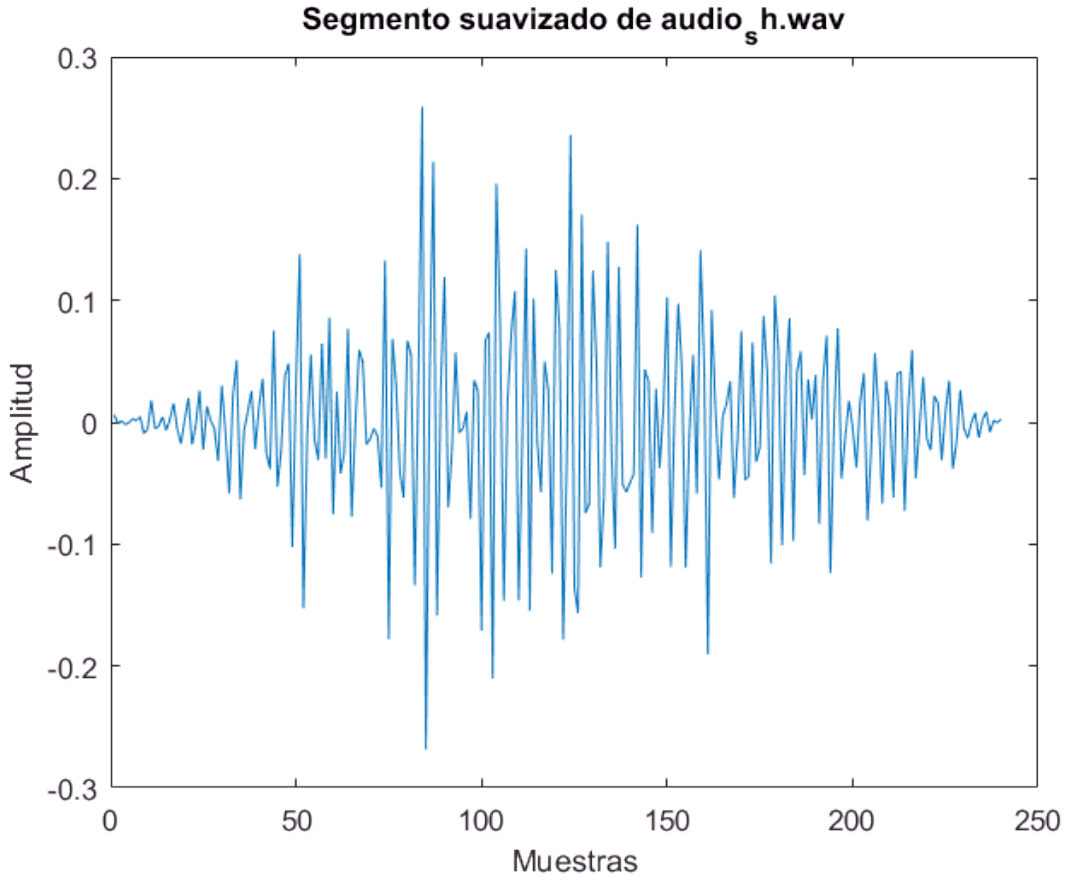
Para cada uno de los archivos de audio "`audio_a.wav`", "`audio_e.wav`", "`audio_s.wav`", "`audio_sh.wav`" (disponibles en el campus), considere un único segmento de 30 ms de duración centrado en la mitad de la señal y suavizado por una ventana de *Hamming*.

A continuación, se presentan los segmentos suavizados mediante la ventana de Hamming. Es notable que, en el caso de los fonemas 'a' y 'e', la señal muestra una similitud visual más cercana a una onda senoidal o cosenoidal con frecuencia constante. En contraste, para los fonemas 's' y 'sh', la apariencia del ruido se asemeja a la de ruido blanco.


Segmento suavizado de audio_a.wav
Amplitud
Muestras
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
0
50
100
150
200
250


Segmento suavizado de audio_e.wav
Amplitud
Muestras
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
0
50
100
150
200
250


Segmento suavizado de audio_s.wav
0.15
0.1
0.05
0
-0.05
-0.1
Amplitud
0
50
100
150
200
250
Muestras


Segmento suavizado de audio_sh.wav
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
Amplitud
0
50
100
150
200
250
Muestras

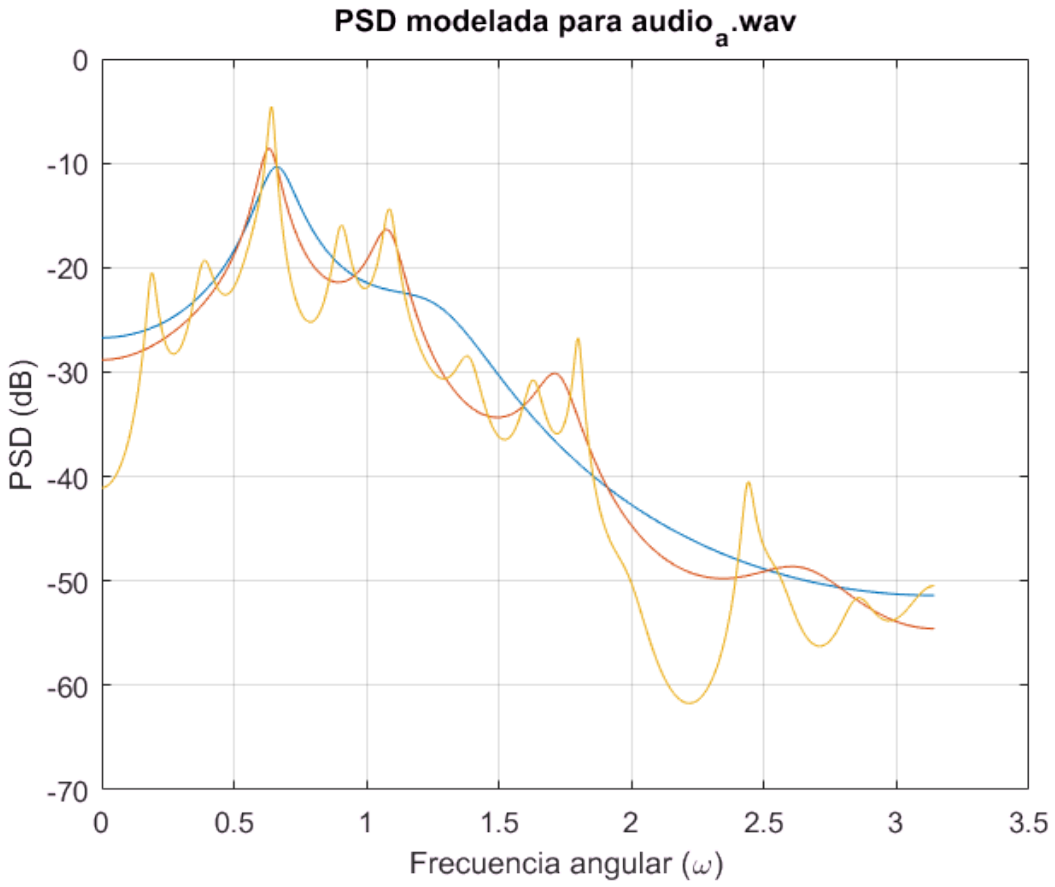
Defina una función con prototipo `param_lpc(xs, P)` donde `xs` es el segmento de señal y `P` el orden del modelo. La función debe retornar los coeficientes LPC y la ganancia G. Para cada audio, estime todos los parámetros LPC suponiendo órdenes $P = \{5, 10, 30\}$.

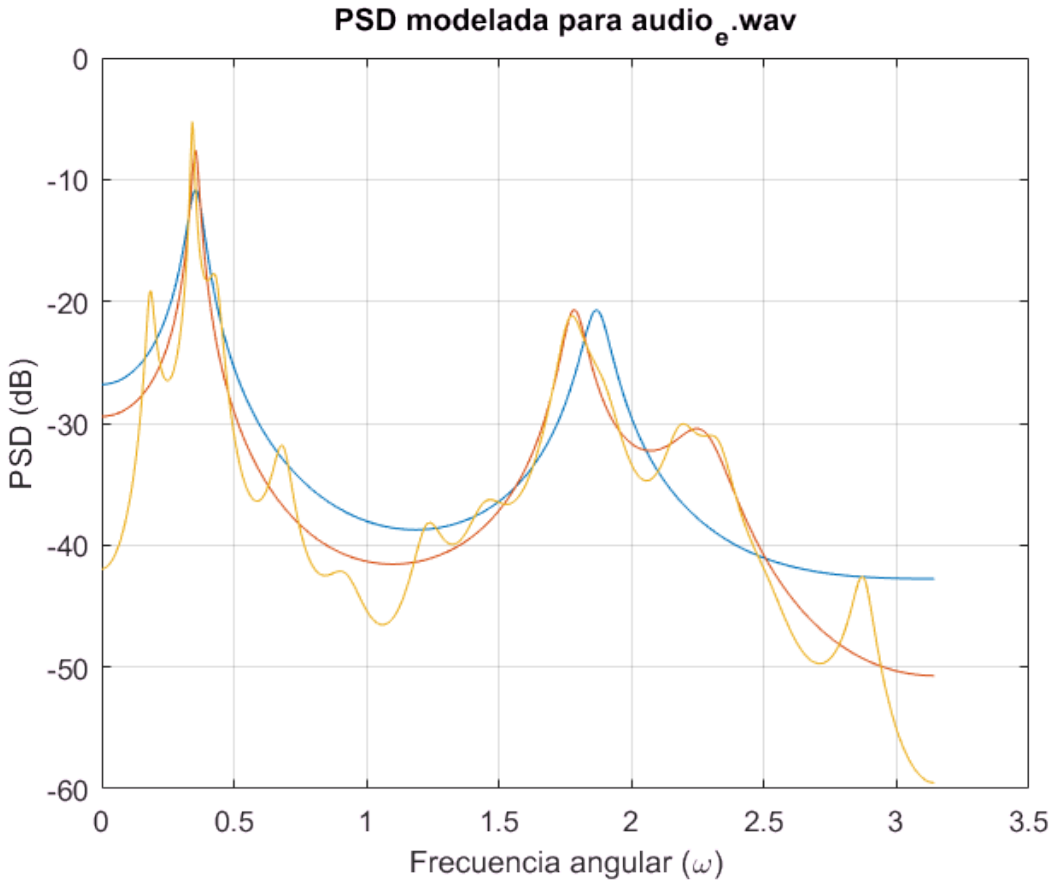
Luego, determine la PSD modelada con los parámetros estimados de acuerdo a la siguiente ecuación (suponer $\omega \in [0, \pi)$):

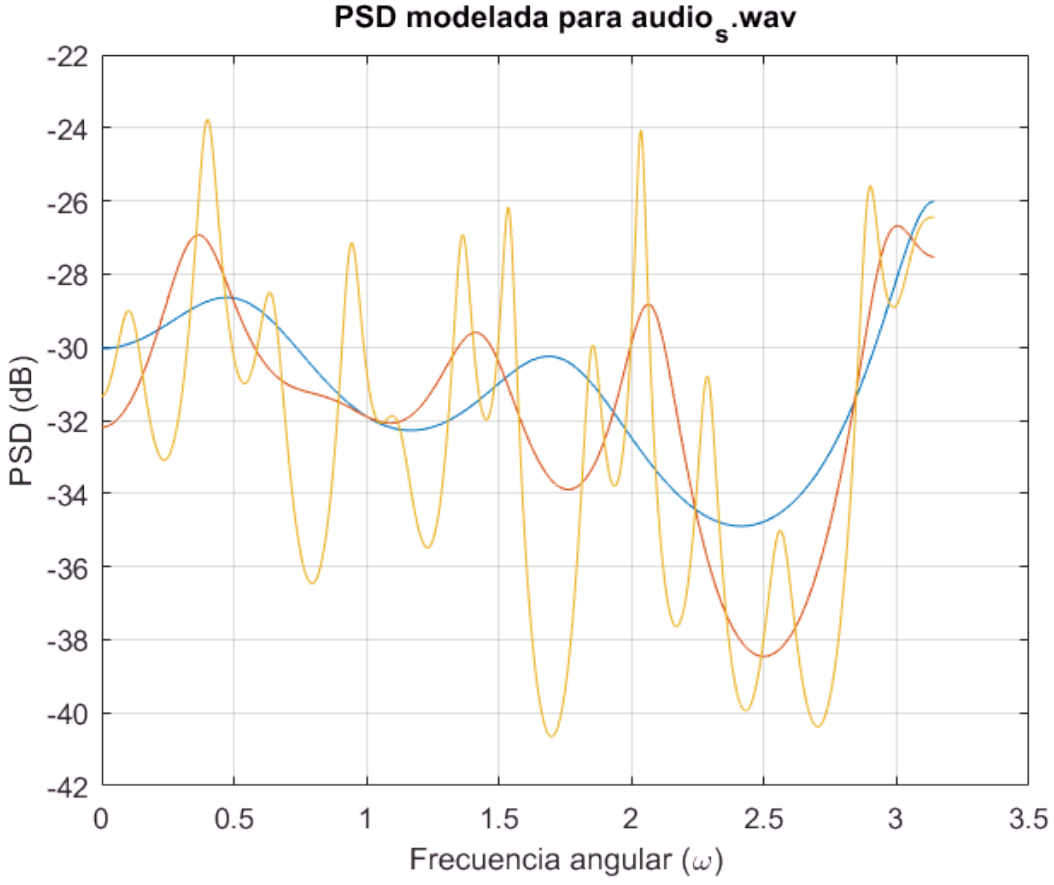
$$S_x(\omega) = \frac{G^2}{|1 - \sum_{k=1}^{P} a_k e^{-j\omega k}|^2} \tag{8}$$

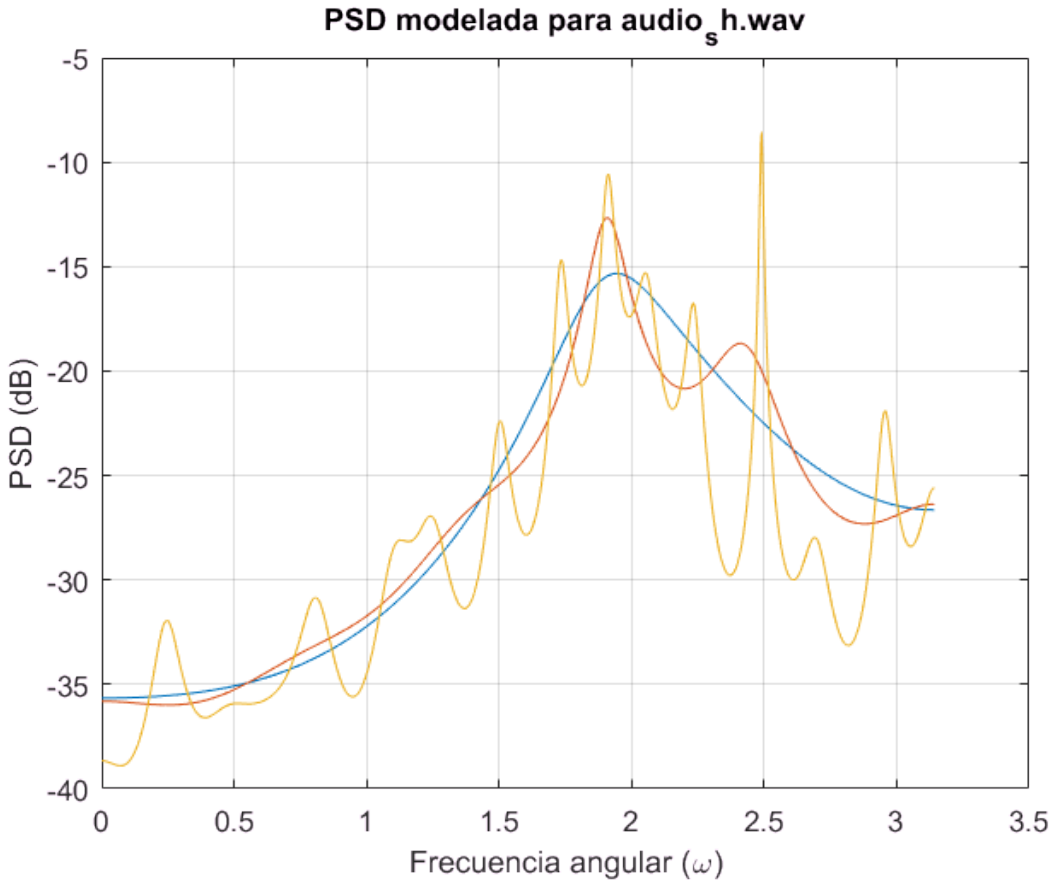
A continuación, se presentan los espectros de densidad espectral de potencia (PSD) de los cuatro fonemas. En el gráfico correspondiente a la vocal 'a', se destaca una predominancia de frecuencias en el rango de 0.5 a 1 rad/segundo. En el caso de la vocal 'e', se observa una situación similar, con dos picos notables alrededor de 0.4 y 1.7 rad/segundo. Estos patrones coinciden con las características esperadas para estos fonemas, que exhiben frecuencias constantes en el tiempo.

Por otro lado, al analizar el PSD del fonema 's', se aprecia que los valores son relativamente uniformes en prácticamente todas las frecuencias ω, lo que se asemeja a la respuesta de una señal de ruido blanco. En contraste, el fonema 'sh' muestra similitudes con 's', pero con una mayor atenuación en las frecuencias más bajas.




PSD modelada para audio$_e$.wav
PSD (dB)
0
-10
-20
-30
-40
-50
-60
0
0.5
1
1.5
2
2.5
3
3.5
Frecuencia angular (ω)

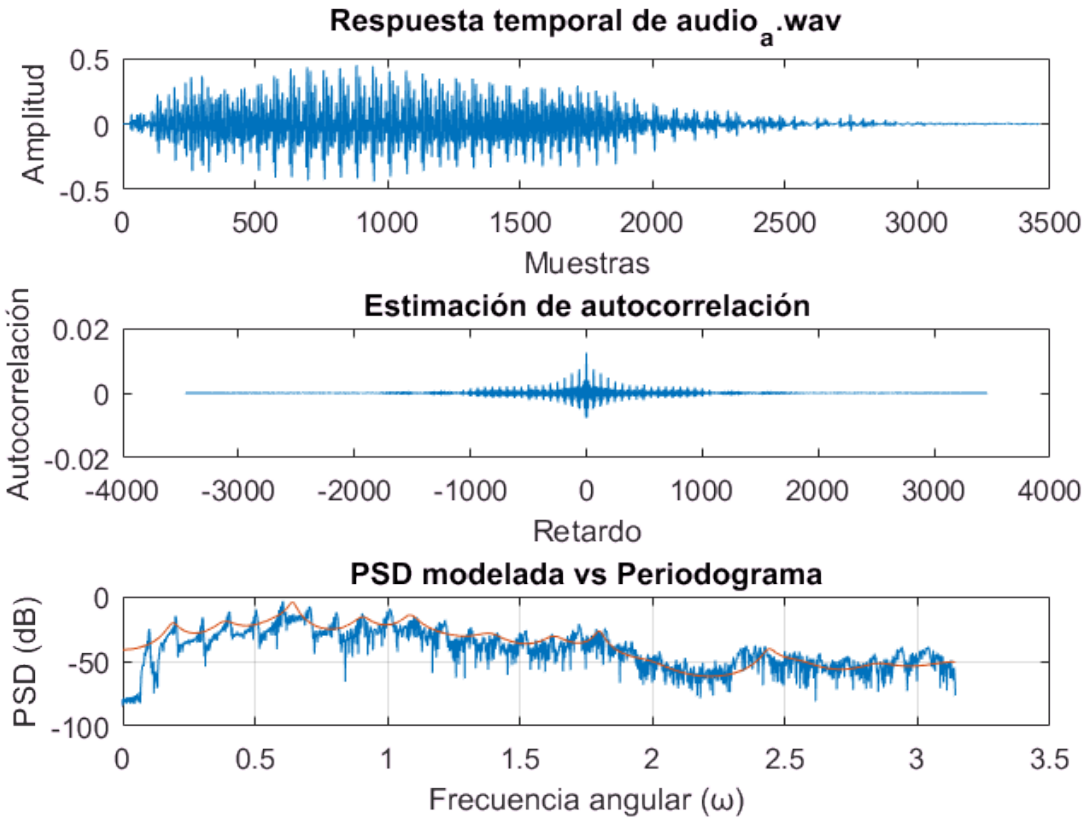

PSD modelada para audio$_s$.wav
PSD (dB)
-22
-24
-26
-28
-30
-32
-34
-36
-38
-40
-42
0
0.5
1
1.5
2
2.5
3
3.5
Frecuencia angular (ω)

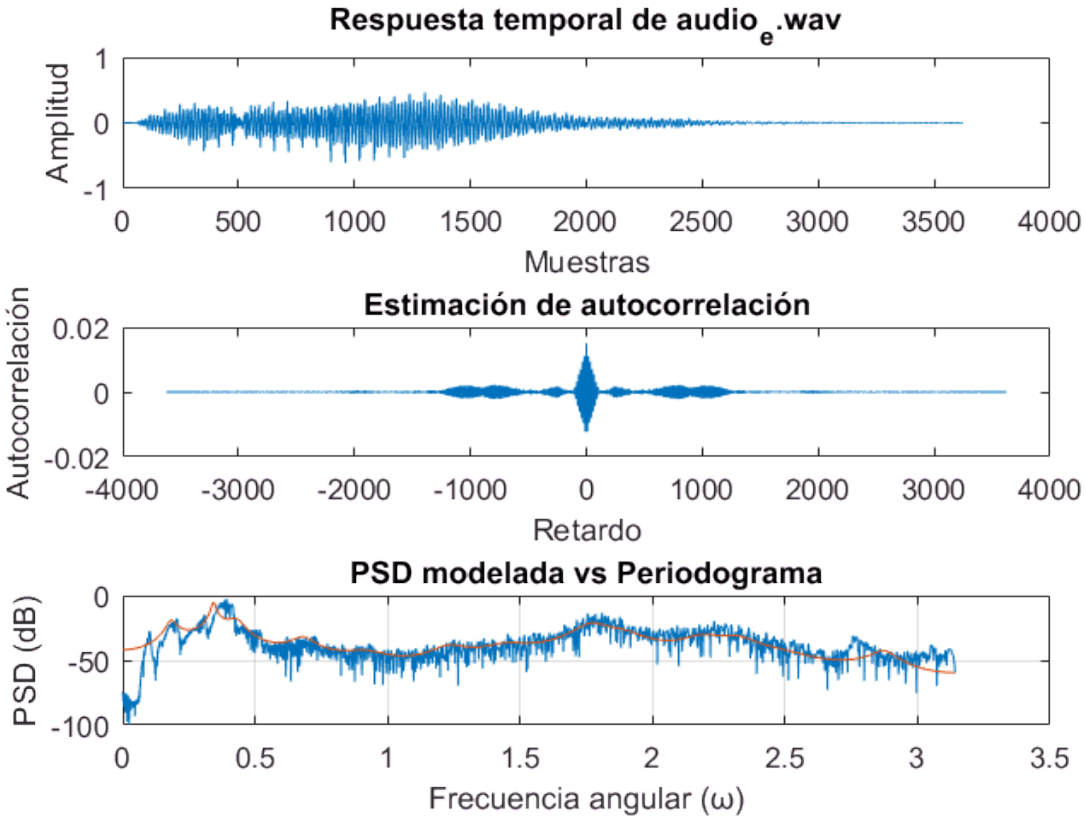


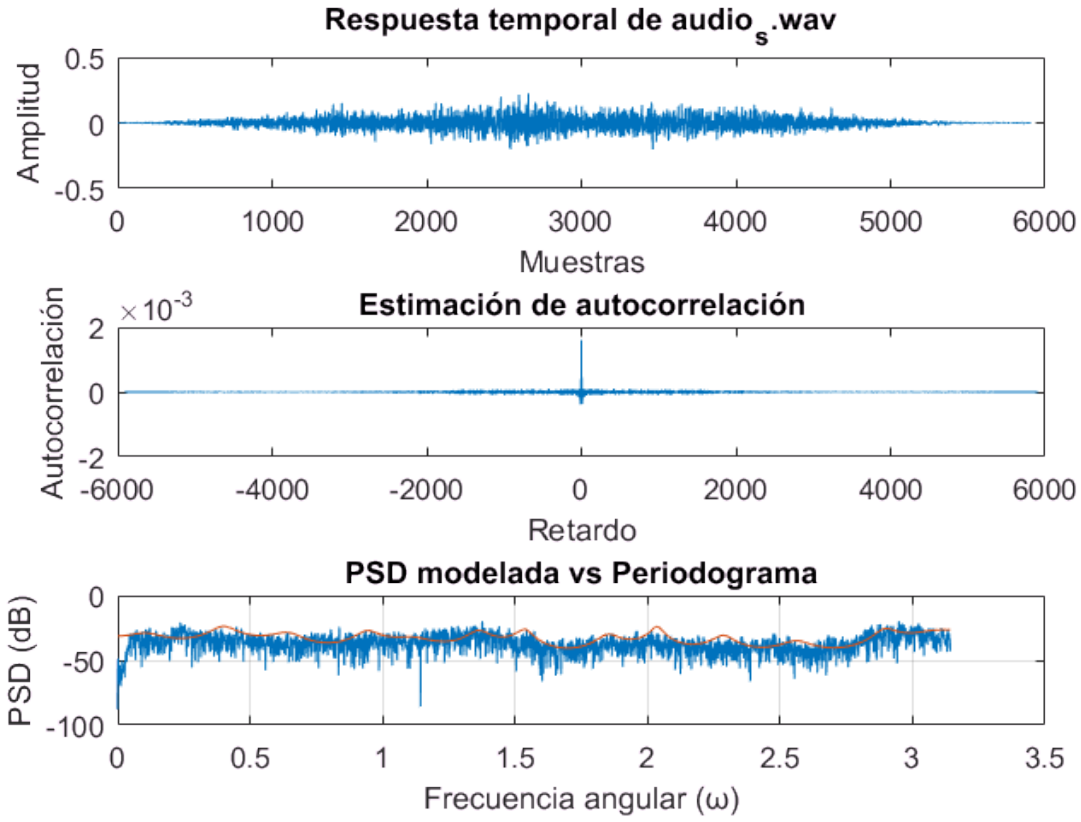
Grafique la respuesta temporal de los audios utilizados, la estimación de su autocorrelación y su PSD (dB) modelada superpuesta al periodograma.

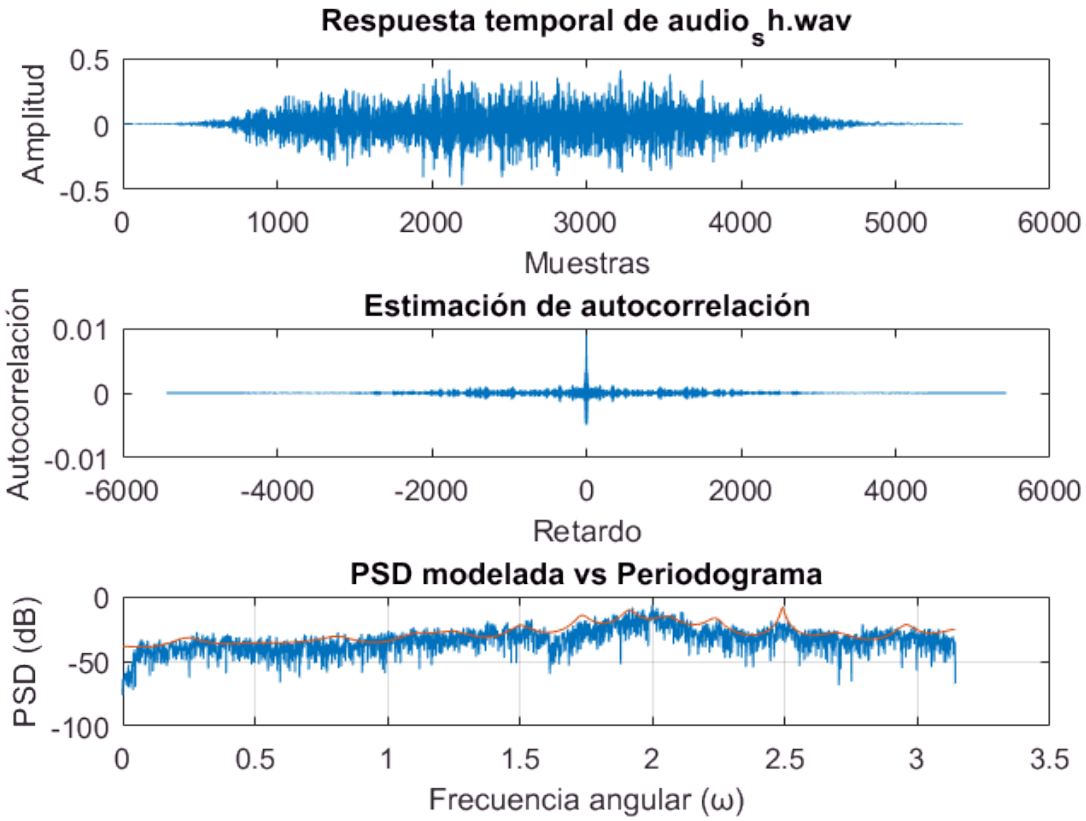
Con la diferencia de tiempo entre el máximo pico de potencia y el segundo pico de mayor potencia podemos determinar el periodo fundamental T0, que está relacionado con la frecuencia fundamental (f0) de la señal de habla. De esta manera podemos identificar y caracterizar las frecuencias fundamentales en los fonemas 'a' y 'e', que tienden a mostrar una apariencia más cercana a una onda senoidal o cosenoidal con frecuencia constante. Para el caso de 'a' obtenemos un T0 de aproximadamente 63 muestras, para 'e' obtenemos 17.

Por otro lado, los fonemas 's' y 'sh' exhiben características que se asemejan a ruido blanco en sus PSD, ya que presentan una distribución uniforme de potencia. Para distinguir entre señales sonoras y sordas, se aplica un método de normalización y se compara el segundo pico más alto con un umbral. Si el segundo pico supera el umbral, se asume una excitación periódica, lo que es especialmente relevante para los fonemas 's' y 'sh', donde la presencia de ruido blanco es más evidente.


Respuesta temporal de audio_a.wav
Amplitud
0.5
0
-0.5
0
500
1000
1500
2000
2500
3000
3500
Muestras
Estimación de autocorrelación
Autocorrelación
0.02
0
-0.02
-4000
-3000
-2000
-1000
0
1000
2000
3000
4000
Retardo
PSD modelada vs Periodograma
PSD (dB)
0
-50
-100
0
0.5
1
1.5
2
2.5
3
3.5
Frecuencia angular (ω)


Respuesta temporal de audio_e.wav
Amplitud
1
0
-1
0
500
1000
1500
2000
2500
3000
3500
4000
Muestras
Estimación de autocorrelación
Autocorrelación
0.02
0
-0.02
-4000
-3000
-2000
-1000
0
1000
2000
3000
4000
Retardo
PSD modelada vs Periodograma
PSD (dB)
0
-50
-100
0
0.5
1
1.5
2
2.5
3
3.5
Frecuencia angular (ω)

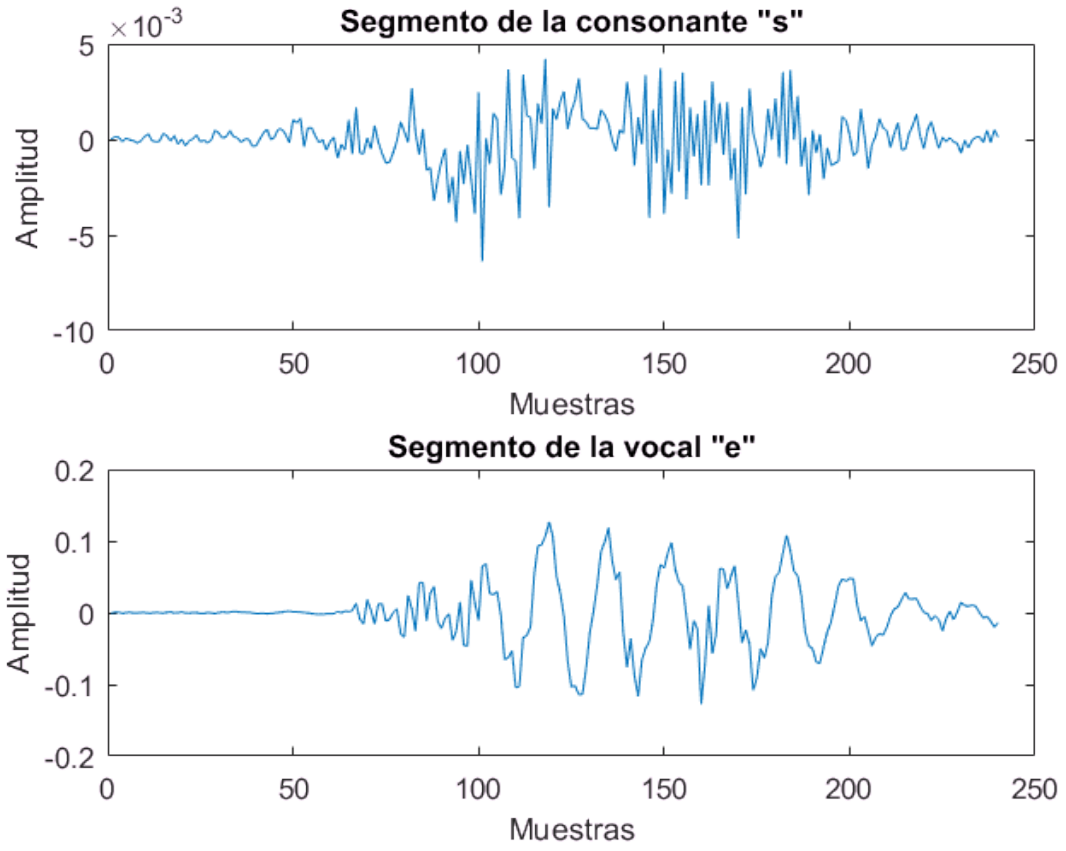

Respuesta temporal de audio_s.wav
Amplitud
0.5
0
-0.5
0
1000
2000
3000
4000
5000
6000
Muestras
Estimación de autocorrelación
Autocorrelación
×10^-3
2
0
-2
-6000
-4000
-2000
0
2000
4000
6000
Retardo
PSD modelada vs Periodograma
PSD (dB)
0
-50
-100
0
0.5
1
1.5
2
2.5
3
3.5
Frecuencia angular (ω)


Respuesta temporal de audio_sh.wav
Amplitud
0.5
0
-0.5
0
1000
2000
3000
4000
5000
6000
Muestras
Estimación de autocorrelación
Autocorrelación
0.01
0
-0.01
-6000
-4000
-2000
0
2000
4000
6000
Retardo
PSD modelada vs Periodograma
PSD (dB)
0
-50
-100
0
0.5
1
1.5
2
2.5
3
3.5
Frecuencia angular (ω)

2.3. Ejercicio 3

En este ejemplo se busca realizar la segmentación de una señal de audio completa, la estimación de los parámetros LPC en cada segmento y la reconstrucción. Tenga en cuenta que puede ajustar los parámetros (orden P y ventana) para lograr una reproducción óptima. Considere como primer prueba los audios del ejercicio anterior.

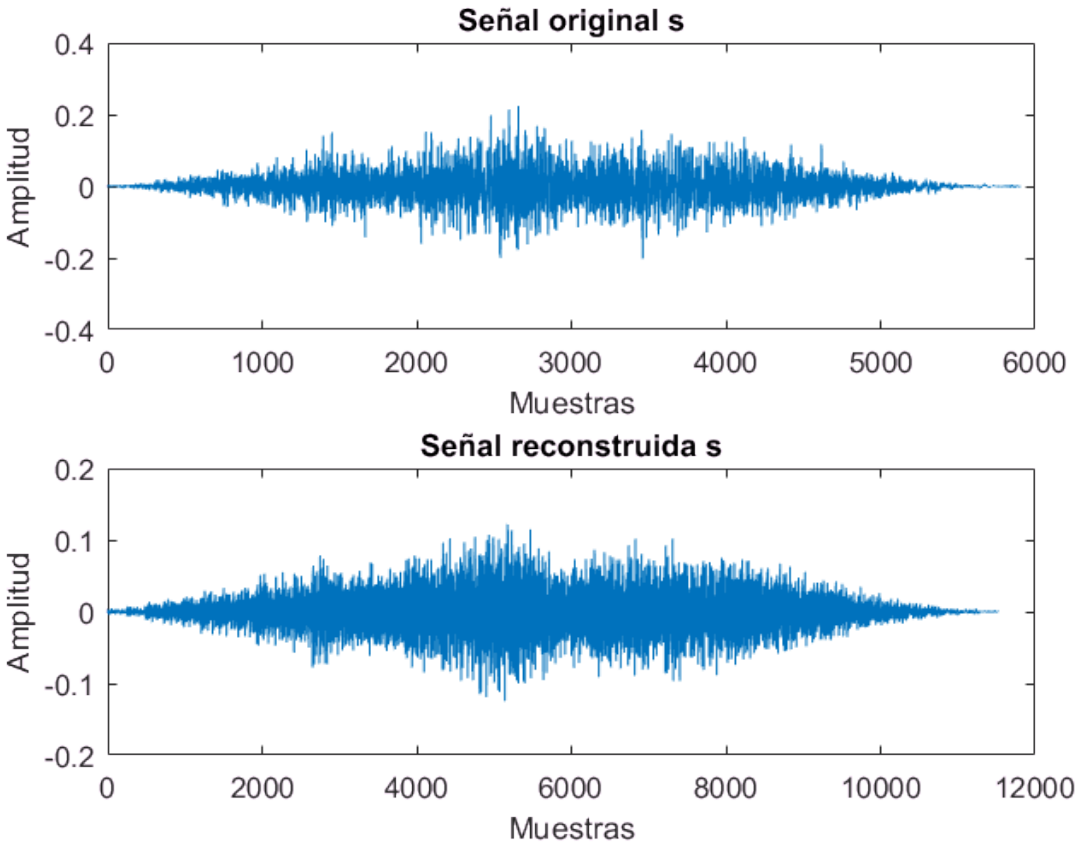
(a) Segmentación: Divida la señal con ventanas de 30 ms de duración suavizadas con una ventana de hamming. Considere un solapamiento del 50 % entre segmentos.

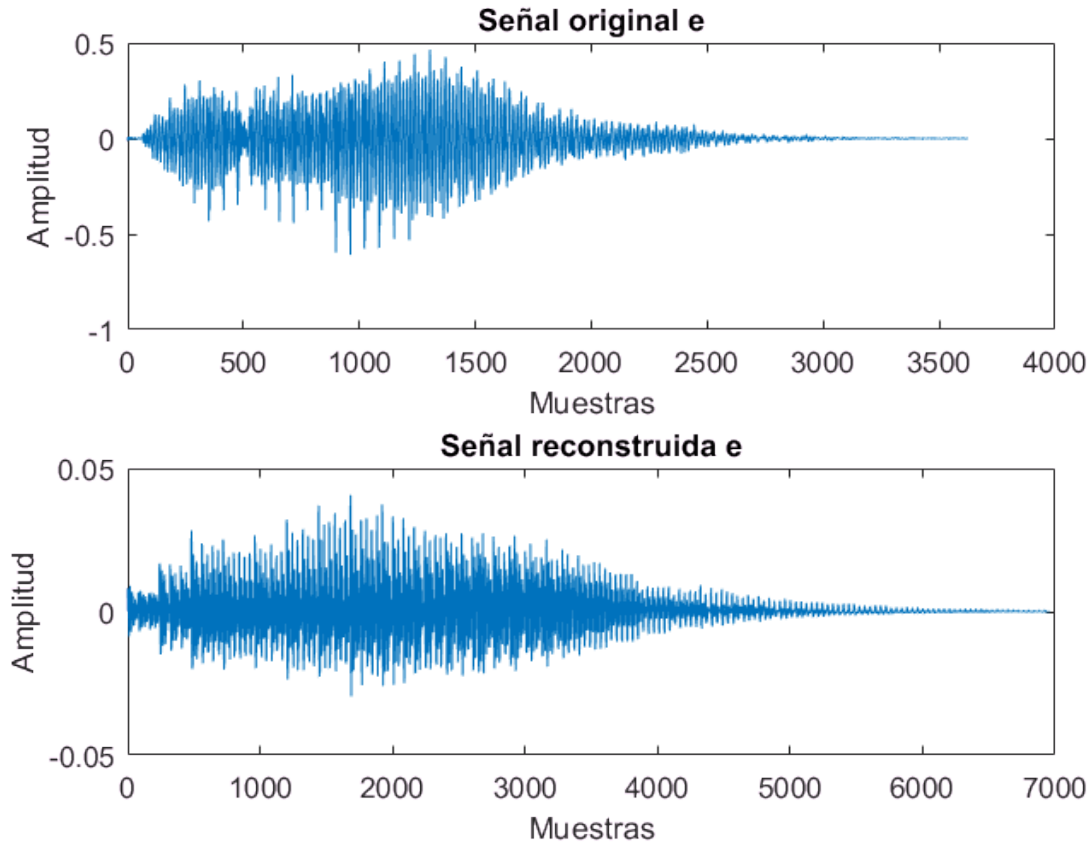


(b) Parámetros LPC: Encuentre los coeficientes y la ganancia del modelo para cada segmento.

(c) Reconstrucción: Utilice los parámetros hallados para regenerar cada segmento y reconstruir la señal. Como entrada considere un proceso blanco gaussiano $u(n) \sim N(0, 1)$ para los archivos de consonantes (`s` y `sh`) y un tren de impulsos periódico $u(n) = \sum_k \delta(n - kN_0)$ de frecuencia 200 Hz para los archivos de vocales (`a` y `e`).

Observamos que la reconstrucion es bastante buena aunque no identica, esto se vera reflejado claramente en el audio aunque es lo suficientemente bueno para que se pueda entender aun siendo un fonema aislado.


Señal original s
Amplitud
0.4
0.2
0
-0.2
-0.4
0
1000
2000
3000
4000
5000
6000
Muestras
Señal reconstruida s
Amplitud
0.2
0.1
0
-0.1
-0.2
0
2000
4000
6000
8000
10000
12000
Muestras



(d) Reproduzca las señales para escuchar las diferencias entre las originales y las reconstruidas. Sugerencia: atenúe la salida de ser necesario para adaptarla a niveles adecuados que impidan la saturación en la salida de audio (por ejemplo `y = y/(10*rms(y))`).

Al escuchar los audios se puede entender cual es el contenido de los mismos aunque la voz se escucha bastante robotica, para mejorar esto se aumento el orden del estimador lo maximo posible obteniendo resultados bastante buenos. Es notable que a ordenes de P bajos la inteligibilidad se mantiene pero se aleja mucho de el audio original en cuanto al timbre de la voz, en ordenes más grandes se mejora un poco este aspecto. En nuestro caso con P = 200 obtuvimos muy buenos resultados.

2.4. Ejercicio 4

Se buscará en este ejercicio incluir la estimación del pitch para cada segmento. En todos los casos se deberán ajustar los parámetros (orden P, ventana y umbral de pitch) para lograr una reproducción óptima. Utilice para las pruebas los audios "`audio_01.wav`", "`audio_02.wav`", "`audio_03.wav`" y "`audio_04.wav`". Considere inicialmente ventanas de 50 ms.

(a) Implemente el algoritmo de detección de la frecuencia fundamental y defina la función `pitch_lpc(xs, a, alpha, fs)` que recibe un segmento de señal `xs`, los coeficientes `a`, el umbral `alpha` y frecuencia de muestreo `fs`. En caso que no se supere el umbral, la función debe retornar una frecuencia nula.

Finalmente, encontramos que para una reproducion optima los parametros deben ser los siguientes:

- Duracion de segmentos: 35ms

- Solapamiento: 100ms
- Orden del estimador: 200
- Alpha: 0.9

Con estos parametros se reconstruyen audios inteligibles, aunque también tiene ciertos ruidos no deseados estan relacinados con el ancho del segmento y su solapamiento. Concluimos que aunque el metodo es muy interesante aun quedan muchos ajustes por realizar para obtener resultados de mejor calidad.

Conclucion:

Pudimos realizar la reconstruccion de la voz de manera optima dentro de los limites del modelo, alcanzando una buena inteligibilidad de la palabra y un timbre de voz que se asemeja al original. Es posible que con más ajustes se puedan conseguir mejores resultados aunque posiblemente metodos mas avanzados den mejores resultados.