PEC2

Presentación

La PEC2 contiene dos ejercicios sobre filtrado lineal óptimo y filtrado adaptativo. Su resolución implica resolver tanto cuestiones teóricas como prácticas. Representaciones gráficas y cálculos, que cuando se soliciten, deberán realizarse utilizando MATLAB. Los desarrollos matemáticos deben llevarse a cabo con un procesador de textos que permita escribir las ecuaciones.

Objectivos

Ofrecer una visión general sobre las técnicas de diseño de filtrado lineal óptimo y las ecuaciones del filtro de Wiener, así como la importancia del filtrado adaptativo que proporciona soluciones de gran interés práctico cuando no disponemos de estadísticas estacionarias. Conocer diferentes escenarios en los que utilizaremos el filtrado lineal óptimo y adaptativo

Descripción de la PEC (Práctica Evaluación Continua) a realizar

1) En los sistemas telefónicos de manos libres se produce acoplamiento acústico entre el altavoz y el micrófono. Imaginemos un usuario 1 que hace una llamada al usuario 2. La señal de voz del usuario 1 que suena al altavoz del teléfono del usuario 2 se propaga también hacia el micrófono del usuario 2 a través del aire, y por lo tanto se superpone con la propia señal de este usuario 2. Esta mezcla, por tanto, es la que será transmitida al usuario 1 generando un efecto de eco muy molesto. Una manera de evitar este problema es cortando la transmisión en un lado mientras el otro está hablando (comunicación semidúplex).

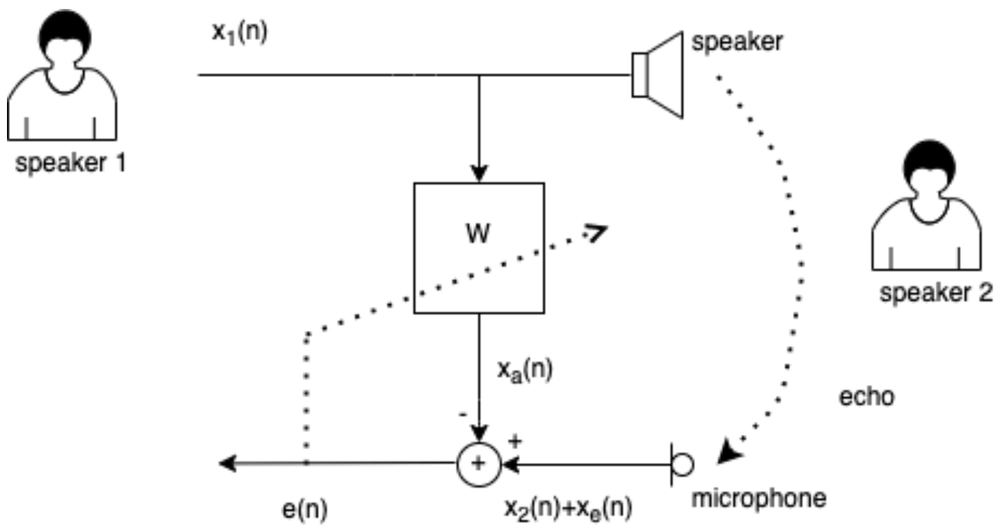


Figura 1: Esquema manos libres

Esta técnica, sin embargo, aunque reduce la presencia de eco, tiene el inconveniente de la pérdida de fluidez y naturalidad de la comunicación. La otra opción, y que estudiaremos en este ejercicio, es utilizar un dispositivo de cancelación del eco acústico mediante el uso de un filtro adaptativo, tal y como se muestra en la figura 1.
Si la autocorrelación de x_1 (locutor 1) es de la forma $R_{x_1x_1} = E[x_1(n)x_1(n-m)] = \alpha^{|m|}$ y la señal de eco es de la forma $x_e(n) = \beta x_1(n-1)$, se pide:

a) Explica cómo funciona el esquema para conseguir cancelar el eco al altavoz del locutor 1.
b) Razona claramente qué márgenes de valor tendrían sentido físico en este escenario, para los parámetros α y β. Elija un valor para cada uno de ellos.
c) Determine las ecuaciones necesarias para calcular los valores óptimos de los coeficientes del filtro adaptativo W para el caso general de P coeficientes, considerando que la correlación cruzada entre las señales originales de los dos locutores es nula.
d) Razonar cuál sería el mínimo número de coeficientes que debería tener el filtro para poder cancelar el eco en este escenario.
e) Considere ahora que el filtro tiene 4 coeficientes. Determine exactamente los valores de los coeficientes del filtro W, a partir del resultado de los apartados anteriores. Calcular la norma del error de los coeficientes respecto de los valores teóricos (use la función norm del Matlab). Comente el resultado.
f) Genere el código matlab necesario para obtener la solución exacta del filtro de Wiener utilizando como señal del locutor 1 y del locutor 2 los señal 'ManAudio.mat' y 'WomanAudio.mat', respectivamente, que ya utilizaste la PEC1. Utilice el valor del parámetros α y β que ha decidido en el apartado (b). Para el número de coeficientes del filtro, considere el valor P = 4. Dar los valores obtenidos para los coeficientes del filtro. Comente el resultado.
g) Detalle como quedaría, para el caso del apartado anterior, el filtro adaptativo entrenado con LMS. Dé las ecuaciones y el código Matlab que se necesitaría para implementarlo, utilizando como señal del locutor 1 y del locutor 2 los señal 'ManAudio.mat' y 'WomanAudio.mat', respectivamente, que ya utilizaste la PEC1 Dado que la frecuencia de muestreo de estas señales era muy elevada, aplique un una delmació por un factor de 10 y tome como entrada las señales en el rango de muestras 4000: 580000, tal y como se muestra a continuación:

```
% data
F = 10;
load ManAudio, x1 = decimate(data(4000:580000,1)',F);
load WomanAudio, x2 = decimate(data(4000:580000,1)',F);
fs = fs/F;
```

Utilice el valor del parámetro β que ha decidido en el apartado (b). Para el número de coeficientes del filtro, considere el valor P = 4. Representa la evolución temporal de los coeficientes del filtro W y dé su valor final. Calcular la norma del error de los coeficientes respecto de los valores teóricos. Comente el resultado, comparándolo con los resultados anteriores.
h) Repetir el apartado anterior para
   i. P igual al mínimo teórico obtenido en el apartado (d).
   ii. $P = 10$.

Para los dos valores de P considerados, representa la evolución temporal de los coeficientes del filtro W y dé su valor final. Calcular la norma del error de los coeficientes respecto de los valores teóricos. Comente el resultado, comparándolo con los resultados anteriores.

i) Escuchar (instrucción 'sound' del matlab) la señal que recibe el locutor 1 ($e(n)$), y compararla con la señal que recibiría sin la corrección que hace el filtro de Wiener ($x_2 + x_e$). Comentar el resultado.

j) ¿Cómo se podría mejorar el resultado del filtro adaptativo? Porque?

Solución

a) El locutor 1 genera la señal de voz x_1. El locutor 2 genera la señal de voz x_2. A la vez, la señal del altavoz que ha llegado al usuario 2 es captado por su micrófono (sistema de manos libres) y por tanto se superpone al propio señal de voz del usuario 2. Esta es la mezcla que llegaría al usuario 1. Por lo tanto, este usuario escucharía la voz del usuario 2 conjuntamente con su propia voz en forma de eco.

El filtro adaptativo W intenta replicar la señal de eco del usuario 1. Esta señal que genera el filtro la llamamos x_a. La señal que llega al locutor 1 será e $e = x_2 + x_e - x_a$, de modo que si el filtro adaptativo estimó bien la señal de eco x_e, es decir si $x_a \approx x_e$ la señal que sentirá el locutor 1 será esencialmente la señal del locutor 2.

b) Los parámetros α y β se refieren a la magnitud de la correlación de la señal del locutor 1 (parámetros α) y en la amplitud que tendrá la señal de eco (parámetro β). Como la señal del locutor 1 es una señal de voz, la autocorrelación será bastante grande y por lo tanto tiene sentido considerar que este parámetro será cercano a 1. Escogemos, por ejemplo, α = 0.9. Como el eco es una señal que tendrá menos amplitud que la señal original, tiene sentido considerar que será cercano a 0. Elegimos, por ejemplo, β = 0.2.

c) Si definimos $y = x_2 + x_e$, la salida del filtro intentará aproximar este señal: $\tilde{y} = x_a \approx y$. Los valores óptimos de los coeficientes del filtro se obtienen a partir de las ecuaciones de Wiener. Por lo tanto, la solución óptima viene de resolver la sigüente ecuación:

$$\mathbf{w}_{opt} = \mathbf{R}_{\mathbf{x_1}}{}^{-1}\mathbf{r}_{y\mathbf{x_1}}$$

con:

$$\mathbf{R}_{\mathbf{x_1}} = \begin{bmatrix} \mathrm{R}_{x_1x_1}(0) & \mathrm{R}_{x_1x_1}(1) & \cdots & \mathrm{R}_{x_1x_1}(P-1) \\ \mathrm{R}_{x_1x_1}(1) & \mathrm{R}_{x_1x_1}(0) & \cdots & \mathrm{R}_{x_1x_1}(P) \\ \cdots & \cdots & \cdots & \cdots \\ \mathrm{R}_{x_1x_1}(P-1) & \mathrm{R}_{x_1x_1}(P-2) & \cdots & \mathrm{R}_{x_1}x_1(0) \end{bmatrix} \text{ i } \mathbf{r}_{y\mathbf{x_1}} = \begin{bmatrix} \mathrm{R}_{yx_1}(0) \\ \mathrm{R}_{yx_1}(1) \\ \cdots \\ \mathrm{R}_{yx_1}(P-1) \end{bmatrix}$$

Concretando para las relaciones que tenemos entre las señales, tendremos:

$$y(n) = x_2(n) + x_e(n) = x_2(n) - 0.2x_1(n-1)$$

$$\mathrm{R}_{x_1x_1}(m) = 0.9^{|m|}$$

$$\begin{aligned}\mathrm{R}_{yx_1}(m) = E[y(n)x_1(n-m)] &= E\big[\big(x_2(n) - 0.2x_1(n-1)\big)x_1(n-m)\big] \\ &= E[x_2(n)x_1(n-m)] - 0.2E[x_1(n-1)x_1(n-m)] \\ &= 0 - 0.2E[x_1(n-1)x_1(n-m)] = -0.2\mathrm{R}_{x_1x_1}(m-1) = -0.2\big(0.9^{|m-1|}\big)\end{aligned}$$

d) Dado que el eco es la señal del locutor 1 con retraso de una sola muestra, el filtro de Wiener ideal se podría determinar con un mínimo de 2 coeficientes.

e) Si consideramos, como nos piden, que el filtro tiene 4 coeficientes, encontraremos sus valores exactos a partir de las ecuaciones del apartado (b) para $P = 4$.

Recordemos que $\mathrm{R}_{x_1x_1}(m) = 0.9^{|m|}$, y por lo tanto, los valores que necesitamos serán:

$$\begin{aligned}\mathrm{R}_{x_1x_1}(0) &= 0.9^{|0|} = 1 \\ \mathrm{R}_{x_1x_1}(1) &= 0.9^{|1|} = 0.9 \\ \mathrm{R}_{x_1x_1}(2) &= 0.9^{|2|} = 0.81 \\ \mathrm{R}_{x_1x_1}(3) &= 0.9^{|3|} = 0.729\end{aligned}$$

Para el vector de correlaciones cruzadas, partimos de $\mathrm{R}_{yx_1}(m) = -0.2\big(0.9^{|m-1|}\big)$, y por lo tanto tendremos:

$$\begin{aligned}\mathrm{R}_{yx_1}(0) &= -0.2\big(0.9^{|0-1|}\big) = -0.18 \\ \mathrm{R}_{yx_1}(1) &= -0.2\big(0.9^{|1-1|}\big) = -0.2 \\ \mathrm{R}_{yx_1}(2) &= -0.2\big(0.9^{|2-1|}\big) = -0.18 \\ \mathrm{R}_{yx_1}(3) &= -0.2\big(0.9^{|3-1|}\big) = -0.162\end{aligned}$$

La ecuación quedarà como:

$$\mathbf{w}_{opt} = \mathbf{R}_{\mathbf{x_1}}{}^{-1}\mathbf{r}_{y\mathbf{x_1}} = \begin{bmatrix} \mathrm{R}_{x_1x_1}(0) & \mathrm{R}_{x_1x_1}(1) & \mathrm{R}_{x_1x_1}(2) & \mathrm{R}_{x_1x_1}(3) \\ \mathrm{R}_{x_1x_1}(1) & \mathrm{R}_{x_1x_1}(0) & \mathrm{R}_{x_1x_1}(1) & \mathrm{R}_{x_1x_1}(2) \\ \mathrm{R}_{x_1x_1}(2) & \mathrm{R}_{x_1x_1}(1) & \mathrm{R}_{x_1x_1}(0) & \mathrm{R}_{x_1x_1}(1) \\ \mathrm{R}_{x_1x_1}(3) & \mathrm{R}_{x_1x_1}(2) & \mathrm{R}_{x_1x_1}(1) & \mathrm{R}_{x_1}x_1(0) \end{bmatrix}^{\mathbf{-1}} \begin{bmatrix} \mathrm{R}_{yx_1}(0) \\ \mathrm{R}_{yx_1}(1) \\ \mathrm{R}_{yx_1}(2) \\ \mathrm{R}_{yx_1}(3) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0.9 & 0.81 & 0.729 \\ 0.9 & 1 & 0.9 & 0.81 \\ 0.81 & 0.9 & 1 & 0.9 \\ 0.729 & 0.81 & 0.9 & 1 \end{bmatrix}^{\mathbf{-1}} \begin{bmatrix} -0.18 \\ -0.2 \\ -0.18 \\ -0.162 \end{bmatrix}$$

$$= \begin{bmatrix} 5.2632 & -4.7368 & 0 & 0 \\ -4.7368 & 9.5263 & -4.7368 & 0 \\ 0 & -4.7368 & 9.5263 & -4.7368 \\ 0 & 0 & -4.7368 & 5.2632 \end{bmatrix} \begin{bmatrix} -0.18 \\ -0.2 \\ -0.18 \\ -0.162 \end{bmatrix} = \begin{bmatrix} 0 \\ -0.2 \\ 0 \\ 0 \end{bmatrix}$$

Fijémonos que el filtro óptimo modela exactamente la señal de eco, por lo que la señal de error $y = x_2 + x_e = x_2$. Además, podemos ver como este resultado se puede conseguir con sólo dos coeficientes, tal y como esperábamos y habíamos avanzado en el apartado anterior. En este caso, pues, la norma del error será cero.

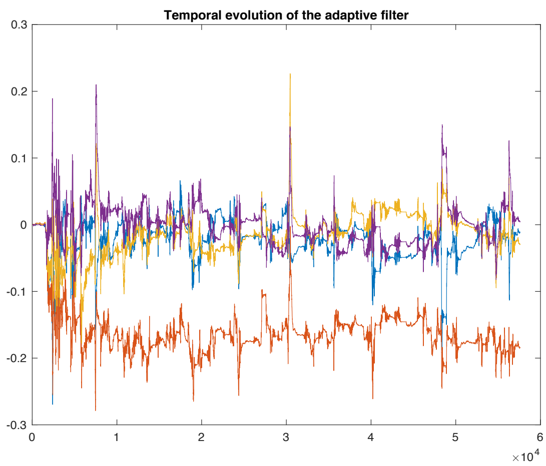
f) Código

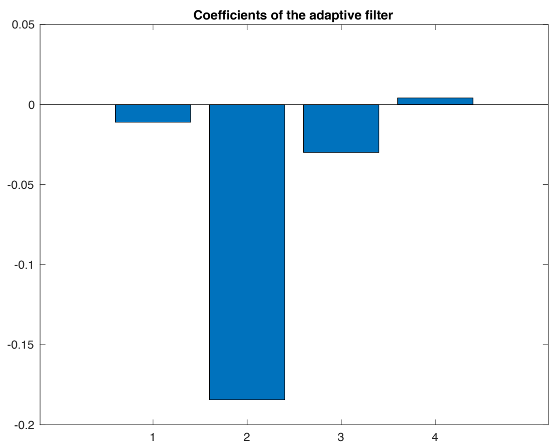
```
clear all
close all
clc
% data
load ManAudio
x1 = data(4000:580000,1)';
load WomanAudio
x2 = data(4000:580000,1)';
clear data
beta = 0.2;
xe = [0 beta*x1(1,1:end-1)];
y = x2-xe; % observed signal
N = numel(x1); % # of samples of the signal
P = 2;  % # of coefficients
%%  Wiener
% correlation values and autocorrelation matrix, for x1
rx = xcorr(x1,'biased');
Rxx = toeplitz(rx(N:N+P-1)); % autocorrelation matrix
% correlation between observed signal and x1
ryx = xcorr(y,x1,'biased');
ryx1 = ryx(1,N:N+P-1);
% optimal coefficients
w = inv(Rxx)*ryx1'
```

g) Código

```
%%  LMS
clear all
close all
clc
% data
F = 10;
load ManAudio
x1 = decimate(data(4000:580000,1)',F);
load WomanAudio
x2 = decimate(data(4000:580000,1)',F);
fs = fs/F;
clear data
beta = 0.2;
xe = [0 beta*x1(1,1:end-1)];
y = x2-xe; % observed signal
P = 4;
N = numel(x1);
mu = 0.5;
% initializations
w = zeros(P,1);         % initializing the filter coefficients
x_vect = zeros(P,1);    % initializing the buffer of the signal
track_w = [];           % initializing the matrix to keep the evolution of
the filter coefficients
% loop
for n = 1:N
```

```
        x_vect = [x1(n); x_vect(1:P-1)];
        est_y(n)= w'*x_vect;
        e(n) = y(n)-est_y(n);
        w = w+mu*x_vect*e(n); % LMS
        track_w = [track_w w];
    end
    figure(2)
    plot(track_w')
    title('Temporal evolution of the adaptive filter')
    figure(3)
    bar(w)
    title('Coefficients of the adaptive filter')
    wop = zeros(P,1); wop(2) = -0.2;
    ew = norm(w-wop)
```

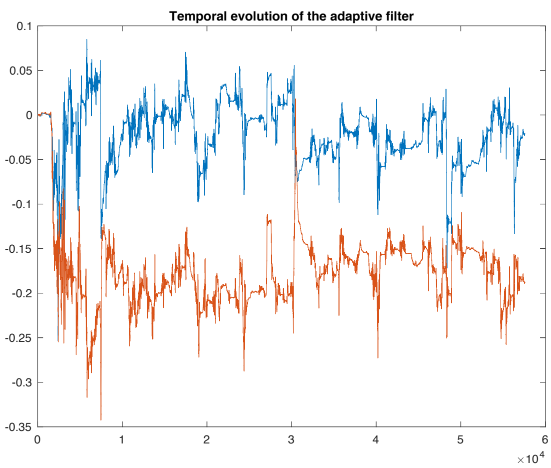


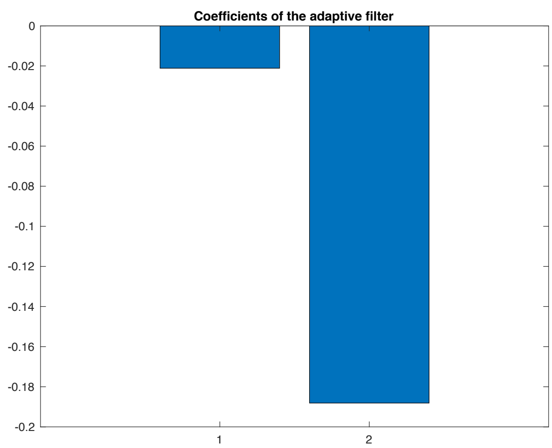


Los valores de los coeficientes son $W = (-0.0110, -0.1844, -0.0298, 0.0042)$. La norma del error será $e_w = 0.0357$.

h) El mismo código pero con $P = 2$.

Las gráficas quedarán como:

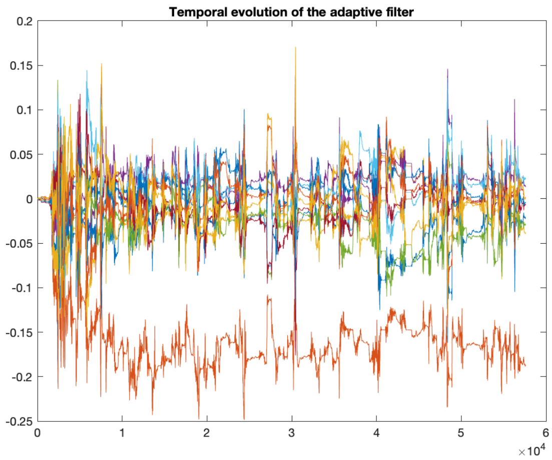


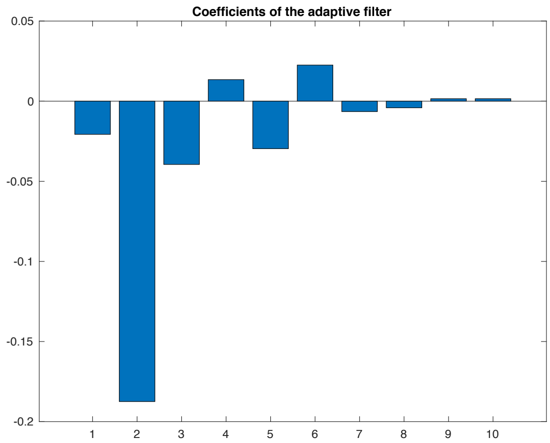


Los valores de los coeficientes son $W = (-0.0211, -0.1881)$.

La norma del error será $e_w = 0.0243$. Observemos que ahora es menor respecto del caso anterior ya que el filtro utiliza menos coeficientes (el número óptimo) y por lo tanto se ajusta mejor.

I ahora con $P = 10$:





Los valores de los coeficientes son
$W =$
$(-0.0207, -0.1874, -0.0395, 0.0135, -0.0296, 0.0225, -0.0065, -0.0041, 0.0015, 0.0015)$
La norma del error será $e_w = 0.0614$. Observemos que ahora es la mayor de todas ya que el filtro utiliza muchos más coeficientes de los necesarios y por lo tanto el algoritmo no llega a ajustar bien los valores. No obstante, se observa claramente que el algoritmo intenta poner todos los coeficientes a cero excepto el segundo que debería ser -0.2.

i) Si escuchamos las dos señales, e(n) e y(n), podemos sentir perfectamente como el filtro de Wiener es capaz de cancelar muy bien la señal del propio locutor 1 en la señal que éste recibe, por lo que esencialmente se siente sólo la señal del locutor 2. En cambio, en la señal y (n) se escucha tanto el locutor 2 como el locutor 1, aunque este está atenuar (factor β).

   Si la señal de entrada tiene una correlación elevada, el algoritmo LMS tendrá una convergencia lenta. Esto se podría solucionar aplicando una etapa de decorrelación de la señal (con un procesador de Gram-Schmidt o con una estructura Lattice) antes del filtro LMS, o bien cambiando el algoritmo de adaptación LMS por un algoritmo RLS, que tiene una convergencia óptima .

2) Los métodos clásicos de predicción lineal están basados en la minimización del error cuadrático, que es la diferencia entre la señal de entrada y su predicción, tal y como se ve en la figura 1:

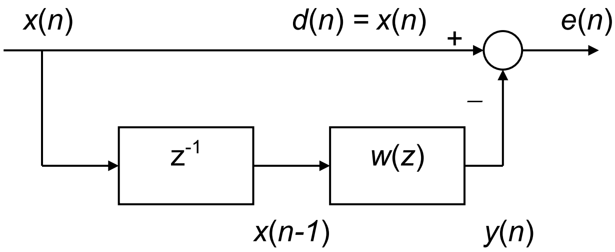


Figura 2: Esquema clásico de predicción lineal

La solución óptima es la que ya conocemos y se obtiene a partir de les ecuaciones de Wiener.

a) Escribe la expresión de la señal y (n) en forma vectorial, definiendo claramente cada uno de los términos en función de las variables de la figura 2. Considera que el filtro W es de 3 coeficientes.
b) Escribe la solución del filtro óptimo para el caso anterior, definiendo claramente todos los términos a partir de las variables de la figura 2.

Si la distribución de la señal de error no es Gaussiana, esta solución no es óptima. En este caso, el error mínimo se obtindría según el esquema que es propone (figura 3):

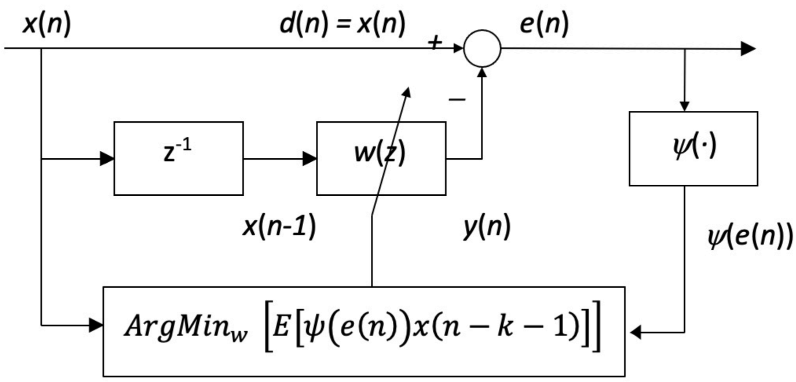


Figura 3: Esquema propuesto

Fijémonos en que la estructura es esencialmente la misma pero ahora los coeficientes del filtro W se adaptan de forma que se minimiza la función $E\left[\psi\left(e(n)\right)x(n-k-1)\right] = 0$, $k = 0, \dots, L-1$, donde L es el número de coeficientes del filtro y $\psi(x)$ es la función score, definida como:

$$\psi(e) = \frac{\partial log\left(p(e)\right)}{\partial e} = \frac{p'(e)}{p(e)}$$

donde $p(e)$ es la función de densidad de probabilidad (*fdp*) de la señal e.

c) Considera e una señal con una *fdp* Gaussiana, de media nula y desviación estándar 1, es decir $\mu = 0$ y $\sigma = 1$. Determinar la función *score* $\psi(e)$ para este caso.

d) Utilitzando ahora la ecuación $E\left[\psi\big(e(n)\big)x(n-k-1)\right] = 0,\ \ k = 0, \dots, L-1$ con la función *score* del apartado anterior, determinar el conjunto de ecuaciones que se obtienene interpretarlas.

Dado un vector de observaciones de T muestras, $\mathbf{x} = \{x(1), x(2), \dots, x(T)\}$, el algoritmo que permite determinar los coeficientes del filtro en el caso de una distribución general (no forzosamente Gaussiana),está codificado en matlab en el fichero PAC2_2.m. Veréis que para la parte de la estimación de la función score hay varias alternativas (líneas 63-70 del código). La alternativa 1 es la que estima la *fdp* real que tenemos.

e) Ejecutar el código PAC2_2.m con la alternativa 1 para el caso de W = 20. Tenga en cuenta que quizás se estará valores de parámetros para que el sistema converja correctamente. Indique, en su caso, los nuevos valores de los parámetros ajustados. Muestre los valores finales de los coeficientes del filtro W y su evolución a lo largo de las iteraciones del algoritmo.

f) Repetir el apartado anterior con la alternativa 2 y la alternativa 3. Indicar, en su caso, los nuevos valores de los parámetros ajustados en cada caso.

g) Comparar cómo es la distribución de la señal de error en todos los casos y razone, a partir de ahí, de las gráficas que ha generado y del valor de ganancia que devuelve el algoritmo, los resultados obtenidos.

h) Explorar el efecto de la función no-lineal en la alternativa 3. Puedes intentar otras funciones como x^3, $sign(\cdot)$, etc., y puedes aplicar funciones no-lineales a ambas variables en el cálculo de la función `xcorr`. ¿Por qué puede ser interesante utilizar el algoritmo de la alternativa 3 en vez de los otros dos?

Solución

a) Les ecuaciones serán:
Simplemente propagando las señales :

$$e(n) = x(n) - y(n) = x(n) - \mathbf{w}^T\mathbf{x}_{n-1}$$

donde $\mathbf{w}^T = [w_0 \quad w_1 \quad w_2\,]$, $y(n) = \sum_{k=0}^{L-1} w(k)x(n-k-1)$ y $\mathbf{x}_{n-1} = \begin{bmatrix} x(n-1) \\ x(n-2) \\ x(n-3) \end{bmatrix}$

Por lo tanto:

$$e(n) = x(n) - \mathbf{w}^T\mathbf{x}_{n-1} = x(n) - w_0x(n-1) - w_1x(n-2) - w_2x(n-3)$$

b) La solución del filtro óptimo será:

$$\mathbf{w}_{opt} = \mathbf{R_x}^{-1}\mathbf{r}_{y\mathbf{x}}$$

donde:

$$\mathbf{R}_x = E[\mathbf{x}_{\mathrm{n-1}}\mathbf{x}_{\mathrm{n-1}}^T] = E\left[\begin{bmatrix} x(n-1) \\ x(n-2) \\ x(n-3) \end{bmatrix} [x(n-1) \quad x(n-2) \quad x(n-3)]\right] =$$

$$= \begin{bmatrix} E[x(n-1)x(n-1)] & E[x(n-1)x(n-2)] & E[x(n-1)x(n-3)] \\ E[x(n-2)x(n-1)] & E[x(n-2)x(n-2)] & E[x(n-2)x(n-3)] \\ E[x(n-3)x(n-1)] & E[x(n-3)x(n-2)] & E[x(n-3)x(n-3)] \end{bmatrix} =$$

$$= \begin{bmatrix} r(0) & r(1) & r(2) \\ r(1) & r(0) & r(1) \\ r(2) & r(1) & r(0) \end{bmatrix}$$

$$\mathbf{r}_{y\mathbf{x}=}E\left[x(n+1)\begin{bmatrix} x(n) \\ x(n-1) \\ x(n-2) \end{bmatrix}\right] = \begin{bmatrix} E[x(n+1)x(n)] \\ E[x(n+1)x(n-1)] \\ E[x(n+1)x(n-2)] \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ r(3) \end{bmatrix}$$

c) Si particularizamos la función de densidad de probabilidad (cogemos x como variable para evitar confusiones con la función exponencial) que nos dan, tenemos:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(x-\mu)^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}$$

Luego, la función *score* será:

$$\psi(x) = \frac{\partial log\big(p(x)\big)}{\partial x} = \frac{p'(x)}{p(x)} = \frac{\frac{-x}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}}{\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}} = -x$$

d) A partir de la ecuación $E\big[\psi\big(e(n)\big)x(n-k-1)\big] = 0,\ \ k = 0, \ldots, L-1$, y con la función *score* que hemos determinado en el apartado anterior, tenemos:

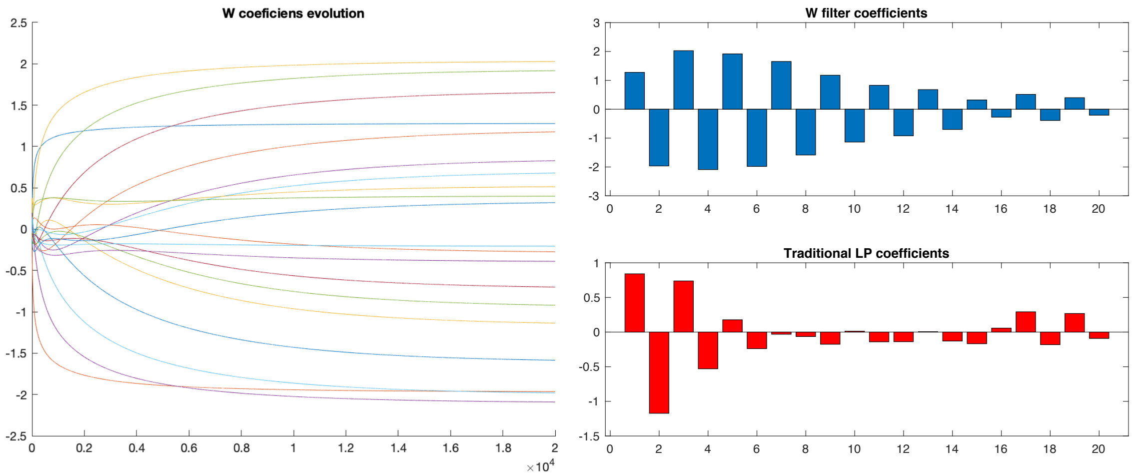
$$E[-e(n)x(n-k-1)] = 0,\ \ k = 0, \ldots, L-1$$

Observamos que lo que estamos pidiendo es que el error sea ortogonal a los datos, a la señal x(n), ya que queremos que la correlación a diferentes desplazamientos temporales (de 0 a L-1) sea siempre cero. Es exactamente el principio de ortogonalidad del filtro lineal optimo. Por lo tanto, confrmamos que si la fdp del error es Gaussiana, las dos soluciones son equivalentes.

e) Para la alternativa 1, con los parámetros `mu = 0.4; itmax = 20000;` obtenemos

```
Gw = 4.9906;
Gm = 4.3994
```

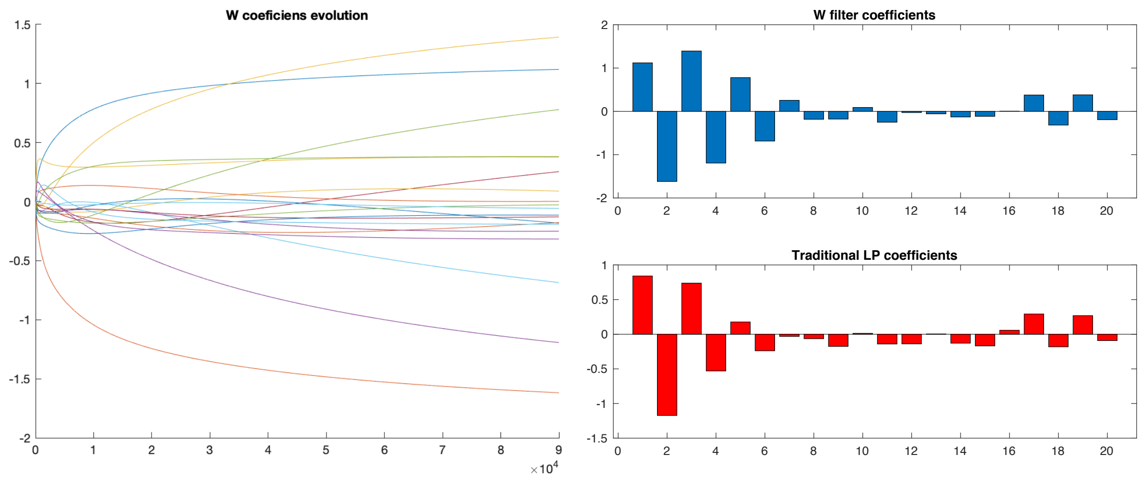
```
The iterative algorithm is better than the classical for
0.59119(dB)
```



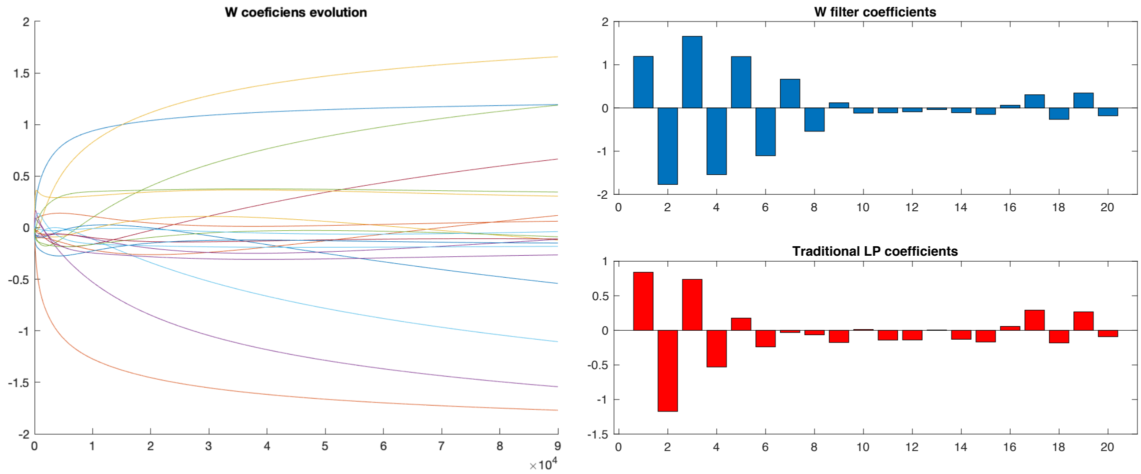
Observamos que el sistema operativo que estima la *fdp* real obtiene mejor resultado que el método clásico.

f) En este caso tenemos:

```
Gw = 4.7864
Gm = 4.3994
The iterative algorithm is better than the classical for
0.38695(dB)
mu = 0.4; itmax = 90000;
Cex = xcorr(e,x,'unbiased');
```

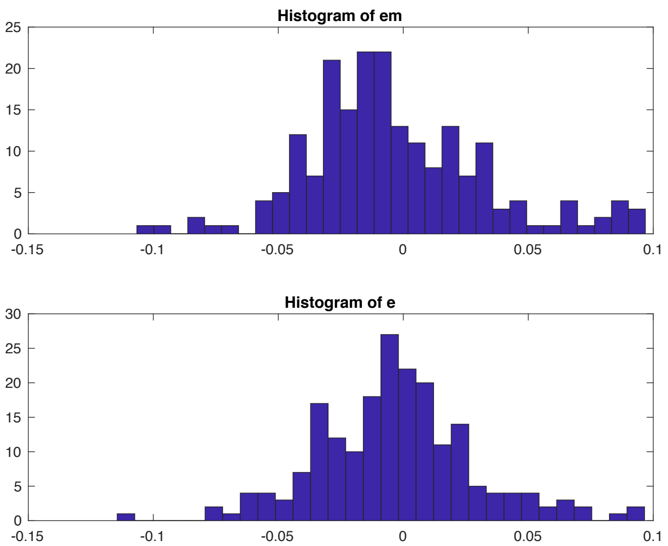


```
Gw = 4.8857
Gm =4.3994
The  iterative  algorithm  is  better  than  the  classical  for
0.48627(dB)
mu = 0.9; itmax = 90000;
Cex = xcorr(e,tanh(x),'unbiased');
```



Fijémonos en que en ambos casos hemos incrementado el número de iteraciones para asegurar que el algoritmo convergía (lo podemos observar en la evolución de los valores de los coeficientes del filtro W). Además, en el segundo caso hemos incrementado el valor del parámetro mu para converger más rápido.

g) Si representamos el histograma de la señal error generado con el método clásico (*em*) veremos que no es Gaussiano. Representamos también el histograma del error (*e*) obtenido con el método iterativo (alternativa 1) para que podamos observar que son diferentes:



Por tanto, tal y como se explica en la teoría, la minimización del error cuadrático no es la solución óptima. Es en estos casos donde el algoritmo propuesto que estima la *fdp* real de la señal de error da mejores resultados (siempre y cuando la estimación de esta *fdp* se haga de forma correcta).

Observamos que la ganancia de cada uno de los tres casos en comparación con el algoritmo clásico, es:

```
Alternativa 1: The iterative algorithm is better than the classical
for 0.59119(dB)
Alternativa 2: The iterative algorithm is better than the classical
for 0.38695(dB)
Alternativa 3: The iterative algorithm is better than the classical
for 0.48627(dB)
```

Es decir, si estimamos correctamente la *fdp* (alternativa 1), el resultado `0.59119 (dB)` es mejor que el obtenido si consideramos simplemente la correlación del error y con la señal x (por tanto, equivalente a presuponer que la distribución del error es Gaussiana, que es lo que se hace en la alternativa 2), `0.38695 (dB)`. A la vez, introducir no-linealidades en la correlación hace que se cumpla mejor la ecuación $E[\psi(e(n))x(n-k-1)] = 0,\ k = 0, \dots, L-1$y por lo tanto el resultado obtenido ahora (alternativa 3), `0.48627 (dB)`, es superior al de la alternativa 2 pero inferior a la alternativa 1. En definitiva, como era de esperar, tenemos:

$$Alternativa\ 1\ >\ Alternativa\ 3\ >\ Alternativa\ 2$$

g) Si se prueban diferentes funciones, los resultados pueden dar mejor o peor, dependiendo del efecto de las no linealidades. Idealmente, queremos generar muchos efectos no-lineales porque así llegaremos a cumplir mejor la ecuación $E[\psi(e(n))x(n-k-1)] = 0,\ k = 0, \dots, L-1$. Este algoritmo alternativa 3 es interesante porque la solución óptima (alternativa 1) puede ser muy lenta en la estimación de la *fdp*. La introducción de funciones no lineales nos permite mejorar la alternativa 2 sin tener que emplear el tiempo que requiere la alternativa 1 si las señales son muy largas.

Recursos

MATLAB,

Módulo 2. Filtrado lineal óptimo

Módulo 3. Filtros adaptativos

Criterios de valoración

Cada ejercicio se puntua sobre 10. Los dos ejercicos valen lo mismo.

Formato y fecha de entrega

Se debe entregar la solución en un solo archivo PDF. El documento debe estar realizado con un procesador de textos que disponga de un editor de ecuaciones. No se admitirán ficheros con ecuaciones escritas a mano. No se admitirán archivos que no sean en formato PDF.

El nombre de este archivo debe tener el siguiente formato:

[Nom_usuari_uoc] _PEC [numero_PEC] .pdf

La fecha límite de entrega es el lunes 26 de abril de 2021 (antes de las 12 de la noche).

Respecto a los ejercicios:

1. Utilice un editor de ecuaciones cuando tenga que poner expresiones matemáticas (el propio editor de Office le sirve).
2. Explique las cosas de forma clara y concreta.
3. Para todos los apartados resueltos con Matlab, agregue el código para reproducir el ejercicio y las figuras que decide generar, de manera que yo pueda copiar y pegar el código en Matlab (por lo tanto, no una imagen del código sino el texto del código)

Nota: Propiedad intelectual

A menudo es inevitable, al producir una obra multimedia, hacer uso de recursos creados por terceras personas. Es por lo tanto comprensible hacerlo en el marco de una práctica de los estudios del Máster, siempre que esto se documente claramente y no suponga plagio en la práctica.

Por lo tanto, al presentar una práctica que haga uso de recursos ajenos se presentará junto con ella un documento en el que se detallen todos ellos, especificando el nombre de cada recurso, su autor, el lugar donde se obtuvo y el su estatus legal: si la obra está protegida por copyright o se acoge a alguna otra licencia de uso (Creative Commons, GNU, GPL). El estudiante deberá asegurarse que la licencia no impida específicamente su uso en el marco de la práctica. En caso de no encontrar la información correspondiente deberá asumir que la obra está protegida por copyright.

Deberán, además, adjuntar los archivos originales cuando las obras utilizadas sean digitales, y su código fuente si corresponde.

Otro punto a considerar es que cualquier práctica que haga uso de recursos protegidos por copyright no podrá en ningún caso publicarse en Mosaic, la revista del Graduado en Multimedia en la UOC, a no ser que los propietarios de los derechos intelectuales den su autorización explícita.