**Trabajo Práctico 8**

**Identificación de sistemas**

**EJERCICIO 1:**

***Considere el sistema y[n] = 0,3y[n-1] – 0,4y[n-2] + 0,2y[n-3] + x[n] y genere una secuencia de salida a una entrada de tipo aleatoria con distribución uniforme y valor medio cero.***

Coef = [1 -0.3 0.4 -0.2]; %coeficientes Ak del sistema (acompañan a Y)

Entrada=rand(1,1000)-0.5; %entrada aleatoria con media aproximada cero

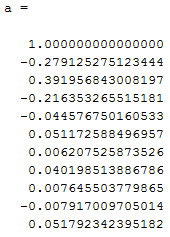
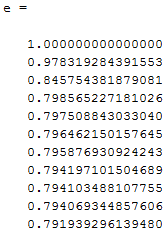
%obtengo la salida del sistema con coeficientes Bk = 1 y Ak = coef

y = filter(1,coef,entrada);

***Utilizando esta señal de salida implemente el método de predicción lineal y verifique el comportamiento de los criterios para estimación del orden.***

r = autocorr(y); %obtengo el vector de autocorrelación de la salida

[a,g,e] = LevDur(r,10); %aplico algoritmo de Levinson-Durbin de orden 10 y obtengo los coeficientes “a” del sistema AR, así como los errores “e” y la ganancia “g” del sistema



Ahora observemos los criterios de estimación de orden:

function[Vp] = prediccionFinal(ep,r0,orden)

for i=1:orden

Vp(i) = (ep(i)/r0); % Vp = ep / e0

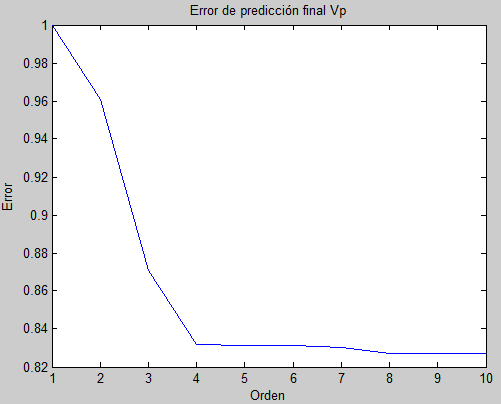
end

end

Vp = prediccionFinal(e,r(1),10); %error de predicción final hasta orden 10

plot(Vp);

En la gráfica podemos observar como los valores del error Vp decrecen acorde se incrementa el orden, pero es monótonamente decreciente. Por lo que el criterio para determinar un orden es algo arbitrario aunque parece indicar que orden 4 sería suficiente, esto es debido a que, entre el orden 4 y el 5 el error se disminuye en una cantidad pequeña (menor a un umbral de error aceptado, el cual también es arbitrario).



Criterio de Akaike:

function[Ip] = akaike(ep,ne,orden) %ep es el error y ne es el número efectivo de muestras en la señal (N si uso ventana cuadrada)

for i=1:orden

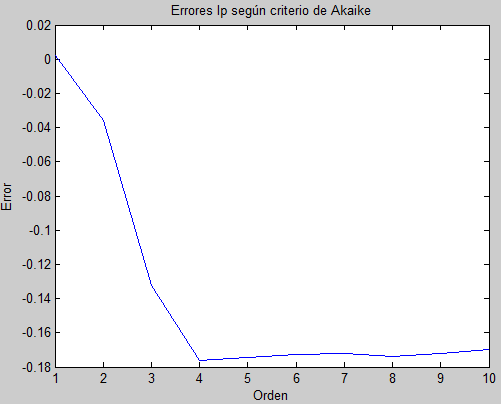
Ip(i) = log(ep(i)) + 2\*i/(ne);

end

end

Ip = akaike(e,1000,10); %aplico criterio de Akaike

plot(Ip);



En esta gráfica, a diferencia de la de Vp observamos que no es monótonamente decreciente, esto es debido a que el criterio de Akaike contrapone la disminución del error al aumentar el orden, con la carga computacional que acarrea dicho procesamiento extra. Es por esto que se observa en la gráfica un punto mínimo que indica el orden óptimo. En este caso orden p=4.

[a,g,e] = LevDur(r,4); %el sistema de orden optimo 4

**EJERCICIO 2:**

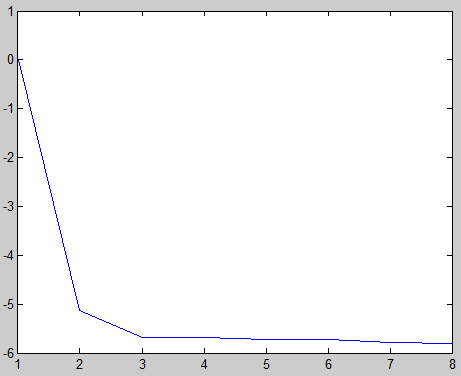
***La señal de electroencefalograma se puede modelar mediante un sistema AR de orden cuatro a ocho. Identifique el sistema que generó la señal almacenada en el archivo eeg.txt.***

s = load('eeg.txt');

r = autocorr(s);

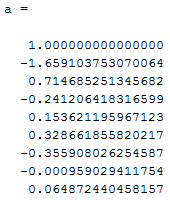
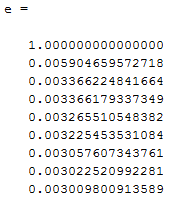
[a,g,e] = LevDur(r,8); %8 es orden suficiente dice el enunciado

Si aplicamos el criterio de Akaike descripto en el ejercicio anterior observamos:



Como muestra la gráfica no se aprecia un mínimo (es aún decreciente), por lo que nos quedamos con la información dada en el enunciado y tomamos al orden p=8 como suficiente.

El sistema AR que modela este electroencefalograma es el determinado por los coeficientes a, ganancia g y error e que se ilustran a continuación:

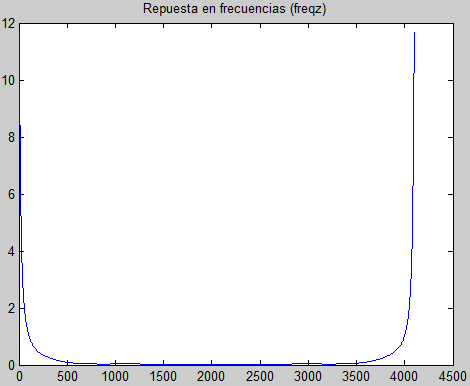




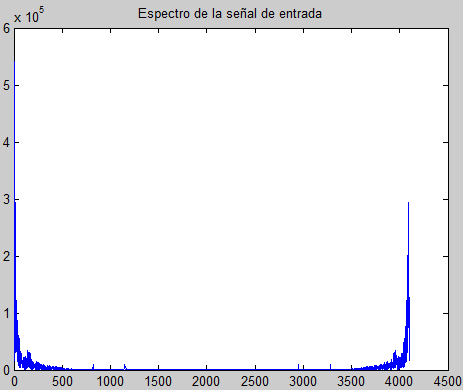
***Compare la respuesta en frecuencia de este sistema con el espectro de la señal.***

[H,W] = freqz(g,a,length(s),'whole');

plot(abs(H)); %respuesta en frecuencias del sistema



plot(abs(fft(s))); %espectro de la señal de entrada



La respuesta en frecuencias es similar al espectro de la señal, buena aproximación. Solo difieren en magnitud y se observa un rizado en el espectro que puede ser creado por el ventaneo o algún ruido.