

Práctica 4 - Programación en MatLab

1 Introducción

En esta práctica veremos la utilización de diversas órdenes de MatLab, así como el uso de bucles y la creación de funciones y ficheros para la ejecución automatizada de órdenes. También es nuestro objetivo la adquisición de cierta habilidad en el manejo de las matrices y los vectores y en el tratamiento de sus índices. Para ello se ha recurrido a la generación de muestras de procesos aleatorios estacionarios discretos en el tiempo y a su caracterización a través de su autocorrelación y su densidad espectral de potencia.

2 Generación de Variables Aleatorias

Es habitual en la práctica el uso de generadores de números aleatorios para simular el efecto de señales similares a ruido y otros fenómenos aleatorios que aparecen en el mundo físico. Dicho ruido está presente en los dispositivos electrónicos y en los sistemas y normalmente limita nuestra capacidad para comunicarnos a lo largo de grandes distancias o de detectar señales relativamente débiles. Generando este ruido en un ordenador, podemos estudiar sus efectos durante la simulación de sistemas de comunicación y comprobar las prestaciones de dichos sistemas en presencia de un ruido.

MatLab dispone de un generador de números aleatorios que permite obtener variables aleatorias con funciones de densidad de probabilidad de muchos tipos, incluidas las variables aleatorias uniformes y Gaussianas, de gran importancia.

Sabemos que los valores numéricos generados por un ordenador tienen una precisión limitada y, por lo tanto, es imposible representar la totalidad de los valores continuos que hay en un intervalo $a \leq X \leq b$. Sin embargo, podemos suponer que el ordenador representa cada valor con un gran número de bits tanto en punto fijo como en punto flotante. Así, con fines prácticos, el número de valores posibles generados en el intervalo $a \leq X \leq b$ es lo suficientemente grande para justificar la suposición de que es posible obtener cualquier valor en el intervalo deseado.

2.1 Generación de variables aleatorias en MatLab

La función de densidad de probabilidad que caracteriza a una variable aleatoria X se denomina $f(X)$. Para una variable aleatoria uniforme U en el intervalo $[0,1]$ esta función se ilustra en la Figura 1(a). La integral de la función de densidad de probabilidad, que representa el área bajo $f(X)$, se denomina *función de distribución de probabilidad* de la variable X y se define como:

$$F(X) = \int_{-\infty}^X f(x) dx$$

Para cualquier variable aleatoria, este área debe ser la unidad, que es el máximo valor que puede alcanzar una función de distribución de probabilidad. La función de distribución de probabilidad de U se ilustra en la Figura 1(b).

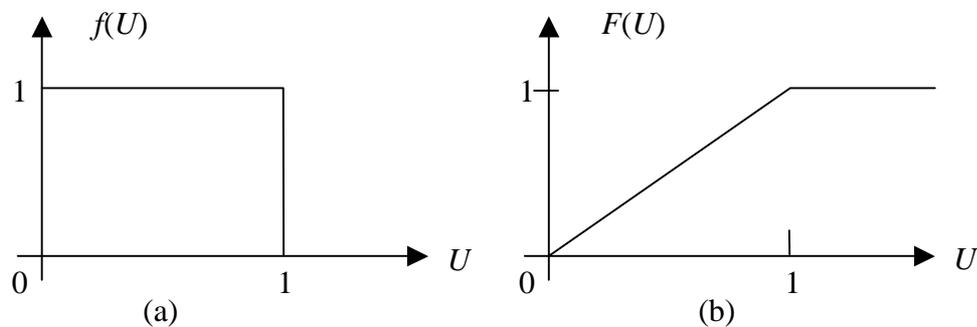


Figura 1 – Función de densidad de probabilidad $f(U)$ y función de distribución de probabilidad $F(U)$ de una variable aleatoria uniforme U .

Para generar una variable aleatoria uniforme discreta en MatLab disponemos de dos posibilidades.

- Las funciones `rand(N)` y `rand(M,N)` devuelven matrices de orden $N \times N$ y $M \times N$ respectivamente de valores aleatorios independientes uniformemente distribuidos entre 0 y 1. Si deseamos obtener un conjunto de variables aleatorias uniformemente distribuidas en el intervalo $[a, b]$ basta con hacer:
`>> u = a + (b - a)*rand(M,N);`
- También disponemos de la función `random` que permite obtener valores aleatorios con una cierta distribución de probabilidad. Para el caso una matriz de $M \times N$ valores aleatorios independientes pertenecientes a una distribución uniforme en el intervalo $[a, b]$ los argumentos que toma esta función son:
`>> u = random('unif', a, b, M, N);`

Cuestión teórica 1: Se desea obtener un proceso aleatorio con 1000 muestras en un vector columna distribuidas uniformemente entre -100 y 100 . ¿Cuáles son las dos posibles órdenes en MatLab con las que se puede obtener dicho proceso?

Del mismo modo, podemos generar también una variable aleatoria normal o Gaussiana de dos formas distintas en MatLab:

- Las funciones `randn(N)` y `randn(M,N)` devuelven matrices de orden $N \times N$ y $M \times N$ respectivamente de valores aleatorios independientes con una distribución normal de media 0 y varianza 1. Si deseamos obtener un conjunto de variables aleatorias normales con media m y desviación típica σ basta con hacer:
`>> n = m + sigma*randn(M,N);`

- Por otro lado, la función `random` permite obtener valores aleatorios independientes con una distribución de probabilidad normal o Gaussiana con media m y desviación típica σ de la manera siguiente:

```
>> n = random('norm',m,sigma,M,N);
```

Cuestión teórica 2: Se desea obtener un proceso aleatorio Gaussiano de media 5 y desviación típica 3 con 4096 muestras en un vector fila. ¿Cuáles son las dos posibles órdenes en MatLab con las que se puede obtener dicho proceso?

Cuestión teórica 3: ¿Cuál es la varianza del proceso Gaussiano anterior?

Una vez generado un vector de valores aleatorios es posible estimar su media y su varianza, entre otros muchos estadísticos, mediante las funciones `mean` y `var` respectivamente. Como sugerencia proponemos generar la señal aleatoria Gaussiana que se indica en la cuestión teórica y comprobar que tanto su media como su varianza son las deseadas.

Además de variables aleatorias uniformes y Gaussianas, mediante la función `random` es posible generar variables aleatorias con otras muchas funciones de distribución de probabilidad conocidas. Sólo es necesario indicar como argumentos de esta función el número de parámetros adecuado para cada tipo de distribución.

Cuestión práctica 1: Averigüe utilizando la ayuda proporcionada por MatLab acerca de la función `random` cuáles son las funciones de distribución de probabilidad con las que proporciona valores de salida el generador de números aleatorios.

2.2 Generación de una variable aleatoria con una función de distribución de probabilidad cualquiera

Aunque ya hemos visto que mediante la función `random` es posible generar números aleatorios para una gran cantidad de funciones de distribución de probabilidad conocidas, en un momento dado puede ser deseable una cierta función de distribución que no sea ninguna de las disponibles. Por eso, veremos a continuación un método para generar variables aleatorias con cualquier función de distribución de probabilidad a partir de una variable aleatoria uniforme en el intervalo $[0,1]$. Por ejemplo, supongamos que queremos generar una variable aleatoria C con una función de distribución $F(C)$ como la que se muestra en la Figura 2.

Podemos recordar la propiedad estadística que afirma que al aplicar la función de distribución sobre la propia variable aleatoria, el resultado es otra variable aleatoria con una distribución uniforme entre 0 y 1.

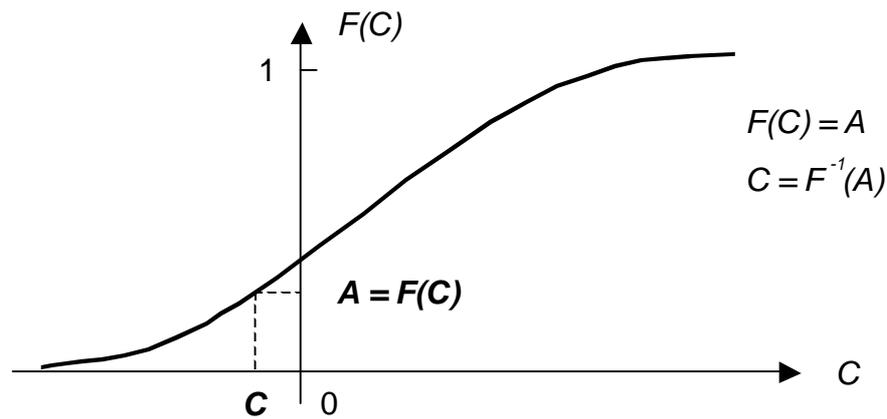


Figura 2 – Mapeo inverso de la variable aleatoria uniformemente distribuida A en la nueva variable aleatoria C .

Puesto que el rango de $F(C)$ es el intervalo $[0,1]$, podemos empezar generando una variable aleatoria A uniformemente distribuida entre 0 y 1. Si hacemos:

$$F(C) = A$$

entonces

$$C = F^{-1}(A)$$

Así, puesto que conocemos la función F deseada para la variable C , podemos resolver la primera ecuación para obtener el valor de C en cada caso para el que la función de distribución toma un valor A concreto.

Ahora vamos a poner en práctica este método mediante un ejemplo. Vamos a generar una variable aleatoria C que tenga la función de densidad de probabilidad siguiente:

$$f(C) = \begin{cases} \frac{1}{2}C^2, & 0 \leq C \leq 1 \\ \frac{1}{2}, & 1 \leq C \leq 2 \\ \frac{1}{6\sqrt{\frac{1}{3}C - \frac{5}{9}}}, & 2 \leq C \leq 3 \\ 0, & e.o.c. \end{cases}$$

Podemos comprobar que el área bajo esta función es la unidad. La función de distribución de probabilidad $F(C)$ correspondiente es la siguiente:

$$F(C) = \begin{cases} 0, & C < 0 \\ \frac{1}{6}C^3, & 0 \leq C \leq 1 \\ \frac{1}{2}C - \frac{1}{3}, & 1 \leq C \leq 2 \\ \sqrt{\frac{1}{3}C - \frac{5}{9}} + \frac{1}{3}, & 2 \leq C \leq 3 \\ 1, & C > 3 \end{cases}$$

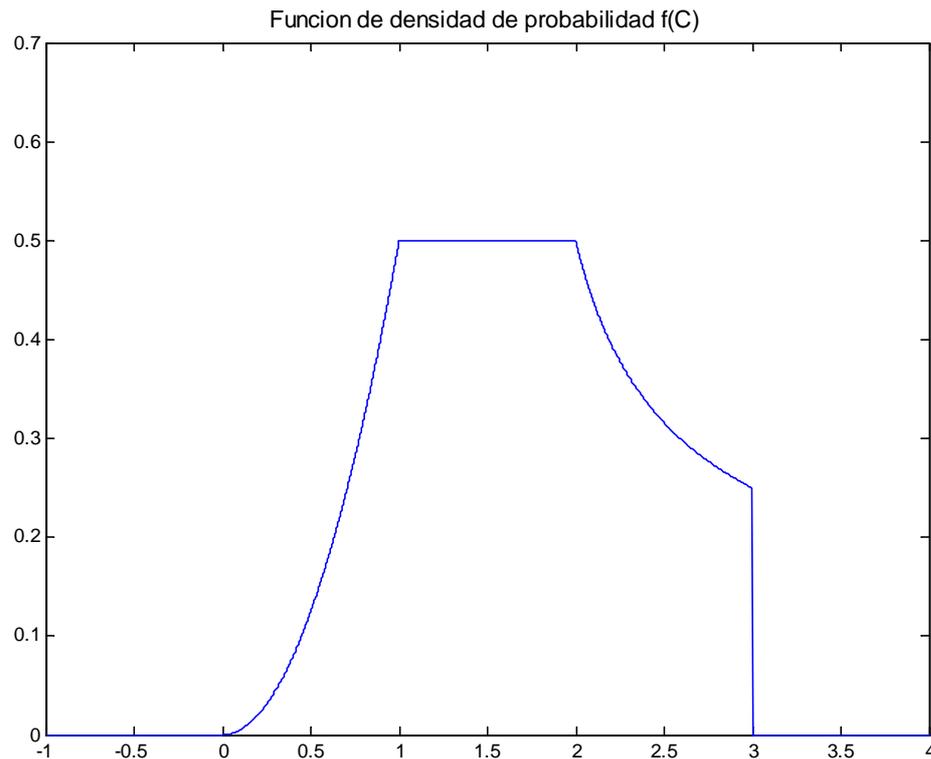


Figura 3 – Función de densidad de probabilidad de la nueva variable aleatoria C .

Cuestión teórica 4: Empleando la ecuación para despejar C en cada tramo, encuentre la expresión de la función $F^{-1}(A)$.

Cuestión práctica 2: Realice una función en MatLab llamada `F_inv` que implemente $F^{-1}(A)$, es decir, que tome como argumento de entrada un valor entre 0 y 1 y devuelva como argumento de salida el valor C generado por la inversa de la función de distribución. Utilice construcciones con el bucle `if` para generar la función definida a trozos.

Ahora que ya disponemos de esta función podemos generar fácilmente un conjunto de variables aleatorias independientes con la función de densidad de probabilidad deseada. Para eso, generamos un proceso aleatorio con 20000 muestras independientes con una distribución uniforme entre 0 y 1, y a continuación le aplicamos la función `F_inv` creada.

La orden de MatLab `hist(x,N)` proporciona el histograma de los valores contenidos en el vector x , representando el número de elementos que hay en cada uno de los N contenedores igualmente espaciados que genera. El histograma puede considerarse una aproximación de la función de densidad de probabilidad de las muestras del proceso aleatorio, aunque con los valores del eje vertical escalados. Emplee esta orden para comprobar que el resultado obtenido es coherente con lo deseado (tome un número elevado de contenedores, por ejemplo 100, para obtener una mejor visualización).

2.3 Autocorrelación y densidad espectral de potencia de procesos aleatorios

Un proceso aleatorio estacionario $X(t)$ está caracterizado en el dominio de la frecuencia por su densidad espectral de potencia (PSD), que es la transformada de Fourier de su función de autocorrelación, $R_x(\tau)$. Es decir,

$$S_X(F) = \int_{-\infty}^{\infty} R_X(\tau) e^{-j2\pi F\tau} d\tau$$

siendo

$$R_X(\tau) = E[X(t+\tau)X^*(t)]$$

Un proceso aleatorio $X(t)$ se dice que es blanco cuando su densidad espectral de potencia es constante para toda frecuencia. La importancia de este tipo de procesos reside en que el ruido térmico se puede modelar como un proceso aleatorio blanco, al menos para las frecuencias empleadas en las comunicaciones convencionales. En dichos sistemas la densidad espectral de potencia del ruido térmico se puede considerar $S_n(F) = N_0/2$, siendo $N_0 = kT$, donde T es la temperatura en grados kelvin y k la constante de Boltzmann (1.38×10^{-23} J/K).

En un proceso aleatorio blanco la función de autocorrelación es

$$R_X(\tau) = \int_{-\infty}^{\infty} S_X(F) e^{j2\pi F\tau} dF = \frac{N_0}{2} \delta(\tau)$$

Esto significa que si muestreamos un proceso aleatorio blanco en dos puntos diferentes t_1 y t_2 , las variables aleatorias resultantes serán incorreladas.

2.3.1 Procesos aleatorios discretos en el tiempo

Un proceso aleatorio discreto en el tiempo se puede ver como una secuencia de variables aleatorias, X_n . Su autocorrelación es

$$R_X(m) = E[X_{n+m}X_n^*]$$

Su densidad espectral de potencia es la transformada de Fourier en tiempo discreto de su autocorrelación.

$$S_X(f) = \sum_{m=-\infty}^{\infty} R_X(m) e^{-j2\pi fm}$$

Supongamos que el proceso discreto en el tiempo X_n procede del muestreo de un proceso continuo en el tiempo $X(t)$. Si la período de muestreo es T_s y llamamos $R_X^C(\tau)$ a la función de autocorrelación y $S_X^C(F)$ a la PSD del proceso continuo entonces se cumple

$$R_X(m) = R_X^C(mT_s)$$

$$S_X(f) = \frac{1}{T} \sum_{l=-\infty}^{\infty} S_X^C\left(\frac{f+l}{T}\right)$$

Si el proceso está limitado en banda y muestreamos adecuadamente es posible recuperar la PSD del proceso continuo a partir del discreto en el tiempo.

En un ordenador trabajamos con secuencias de muestras de los procesos aleatorios. Por lo tanto el cálculo de las esperanzas hay que realizarlo suponiendo que los procesos son ergódicos. Es decir, que sus promedios temporales coinciden con sus esperanzas. En ese caso, la función de autocorrelación se puede estimar como

$$\hat{R}'_X(m+M+1) = \begin{cases} \frac{1}{N-m} \sum_{n=1}^{N-m} X_{n+m} X_n^* & m = 0, 1, \dots, M \\ \frac{1}{N+m} \sum_{n=-m+1}^N X_{n+m} X_n^* & m = -1, -2, \dots, -M \end{cases}$$

siendo N el número de muestras de la secuencia y M el número de desplazamientos positivos y negativos para el cálculo de la autocorrelación. Se ha tenido en cuenta que MatLab considera los índices empezando por el 1 (no existen índices negativos ni 0).

Podemos ver que

$$\hat{R}'_X(m+M+1) = \begin{cases} \frac{1}{N-|m|} \sum_{n=1}^{N-|m|} X_{n+|m|} X_n^* & m = 0, 1, \dots, M \\ \left(\frac{1}{N-|m|} \sum_{n=1}^{N-|m|} X_{n+|m|} X_n^* \right)^* & m = -1, -2, \dots, -M \end{cases}$$

Es decir, se cumple la propiedad de simetría $\hat{R}'_X(-m+M+1) = \hat{R}'_X(m+M+1)^*$ para $m = 0, 1, \dots, M$.

La secuencia $\hat{R}'_X(m+M+1)$ es una estimación de la autocorrelación desplazada (ya que en la muestra 1 aparecerá la correspondiente a $m = -50$). Para evitar ese

$$\hat{S}_x(k+1) = \sum_{m=-M}^M \hat{R}_x(m+M+1)e^{-j\frac{2\pi}{2M+1}k(m+M)} \quad k = 0, \dots, 2M$$

Las frecuencias para las que se realiza la estima serán, en rad, $\omega = 2\pi k/(2M+1)$ y en ciclos $f = k/(2M+1)$.

En MatLab este cálculo se realiza con la función `fft`, que implementa el algoritmo FFT, que estima la Transformada de Fourier de una secuencia de datos. La orden `fft` permite realizar esta estimación utilizando el mismo número de puntos que tiene la secuencia discreta o bien utilizando un número distinto. Si se utilizan más puntos se rellena con ceros. Si se utilizan menos se trunca la secuencia. Nosotros utilizaremos los mismos puntos (vea la ayuda de MatLab).

Aunque la PSD de un proceso tiene fase igual a 0 para todas las frecuencias, debido a la precisión del ordenador aparece una fase muy pequeña (despreciable), pero que nos impide representar la PSD en dos dimensiones frente a la frecuencia. Para evitar esto tendremos que solicitar a MatLab de forma explícita que represente el módulo de la PSD. El cálculo del módulo de un número complejo se realiza mediante la orden `abs`, mientras que para calcular la fase se utiliza `angle`.

Tal como hemos definido el cálculo de la PSD (es decir, tal como lo hace MatLab con la orden `fft`) las frecuencias están definidas de 0 a $2M/(2M+1)$. Para que en la representación de la PSD las frecuencias se encuentren en el intervalo $[-M/(2M+1), M/(2M+1)]$ es necesario utilizar la orden `fftshift`. Si tenemos una secuencia de un número par de muestras $2L$, la orden `fftshift` realiza el cambio observado en la Figura 5.

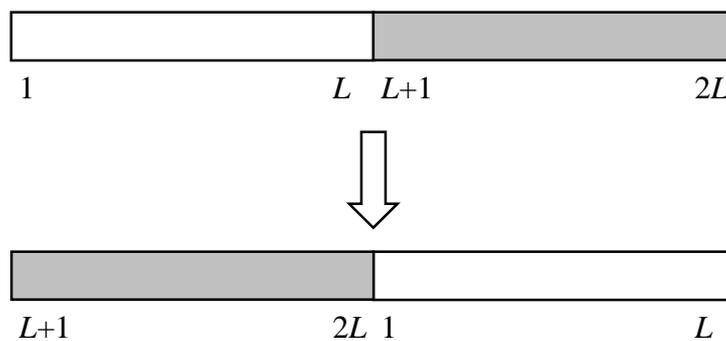


Figura 5 – Efecto de la orden `fftshift` en secuencias con un número par de puntos

Para secuencias con un número impar de puntos $2L+1$ el cambio realizado por `fftshift` es el que se aprecia en la Figura 6.

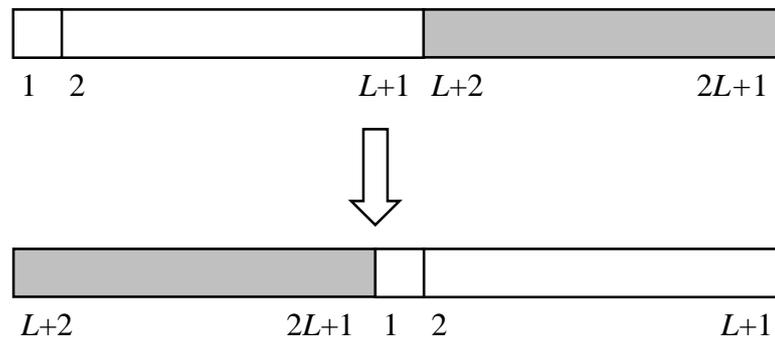


Figura 6 – Efecto de la orden `fftshift` en secuencias con un número impar de puntos

Si hacemos `fftshift` a una PSD (o a cualquier secuencia obtenida de una orden `fft`) con un número impar de puntos las frecuencias asociadas a sus muestras serán $k/(2L+1)$ para $k = -L, \dots, L$. Si se trata de un número par de puntos las frecuencias serán $k/(2L)$ para $k = -L, \dots, L-1$ (recordemos que la Transformada de Fourier en Tiempo Discreto es periódica con período 2π rad o 1 ciclo).

Cuestión práctica 4: Estimar la densidad espectral de potencia del proceso generado en la cuestión previa. Compruebe la diferencia entre la fase de la `fft` realizada a \hat{R}'_x y la realizada a \hat{R}_x . Diseñar un fichero que realice los pasos anteriores (generación de la secuencia y cálculos de autocorrelaciones y densidades espectrales) 100 veces, promediando los resultados obtenidos para la autocorrelación y para la densidad espectral de potencia. Dibujar los resultados obtenidos para la autocorrelación y para la densidad espectral de potencia. Utilizando la orden `axis`, defina en las gráficas de las PSD el eje vertical de 0 a 0.14 y deje el horizontal de -0.5 a 0.5 para visualizar mejor el aspecto plano de las PSD.

A continuación se muestran las gráficas de la estimación de la autocorrelación desplazada (\hat{R}'_x) y la estimación de la autocorrelación (\hat{R}_x), así como la gráfica del módulo de la PSD del proceso. Compruebe que los resultados que obtiene son similares.

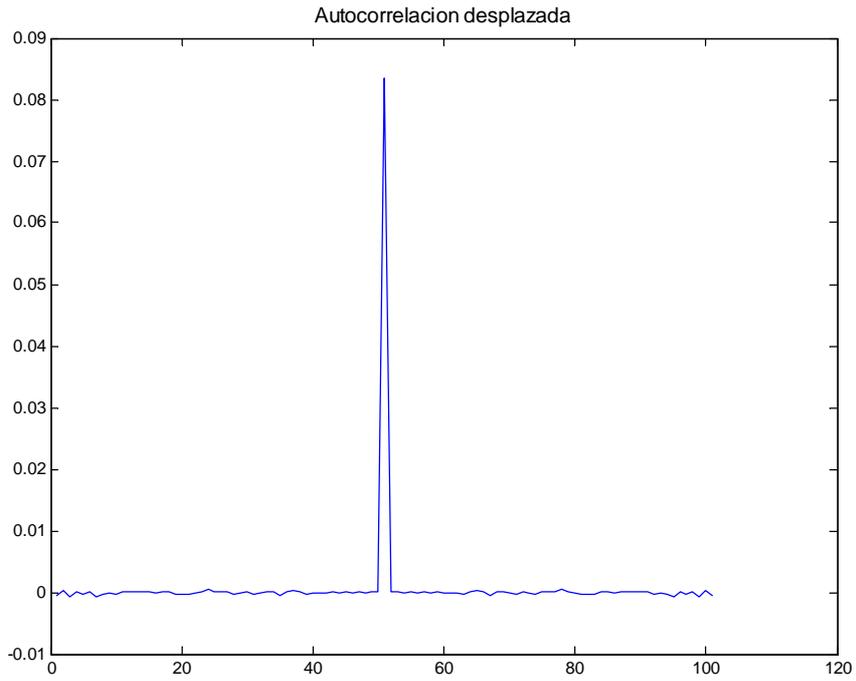


Figura 7 – Autocorrelación desplazada.

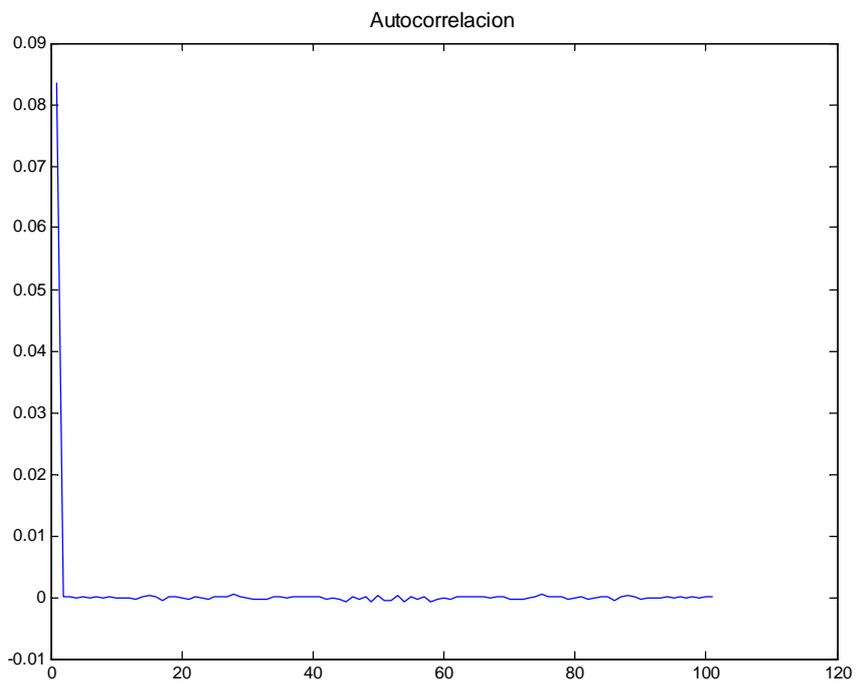


Figura 8 – Autocorrelación.

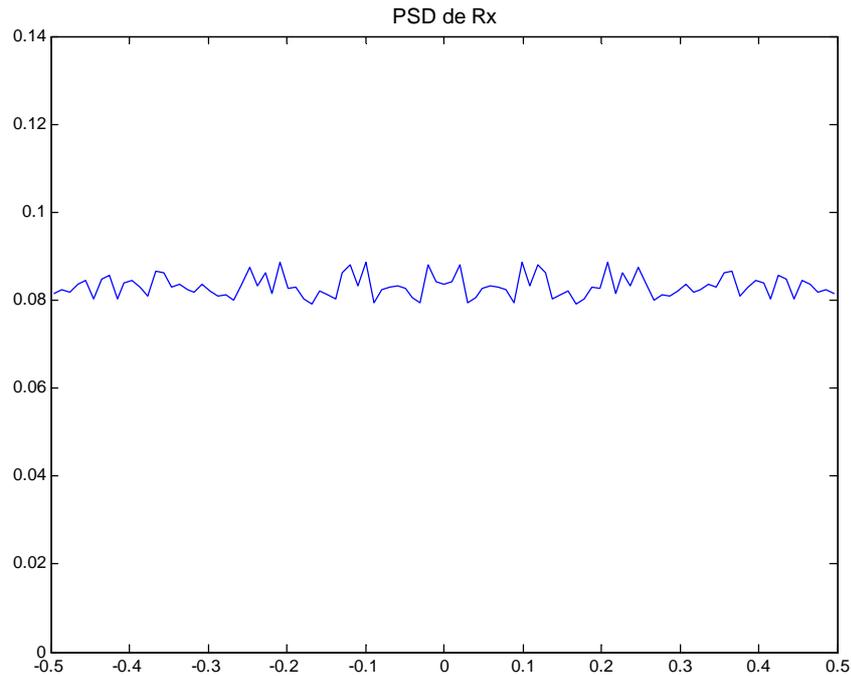


Figura 9 – Densidad espectral de potencia.

2.4 Filtrado lineal de procesos aleatorios

Supongamos un proceso aleatorio discreto en el tiempo X_n que procede del muestreo de un proceso aleatorio estacionario $X(t)$. Si ese proceso pasa a través de un filtro $h(n)$ la secuencia resultante Y_n tendrá media

$$m_Y = m_X H(0)$$

función de autocorrelación

$$R_Y(m) = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} h(k)h(l)R_X(m-l+k)$$

y PSD

$$S_Y(f) = S_X(f)|H(f)|^2$$

siendo $H(f) = \sum_{n=-\infty}^{\infty} h(n)e^{-j2\pi fn}$

Supongamos que un proceso blanco pasa a través de un filtro lineal

$$h(n) = \begin{cases} 0.9^n & n \geq 0 \\ 0 & n < 0 \end{cases}$$

Cuestión teórica 5: Determinar el espectro de densidad de potencia del proceso resultante de pasar un proceso blanco por el filtro que se acaba de describir en función del espectro de densidad de potencia del proceso de entrada. Dibujarlo para $-1/2 \leq f \leq 1/2$ con saltos de 0.01 suponiendo que la PSD del proceso de entrada es 1. Compare con la Figura 10 el resultado obtenido.

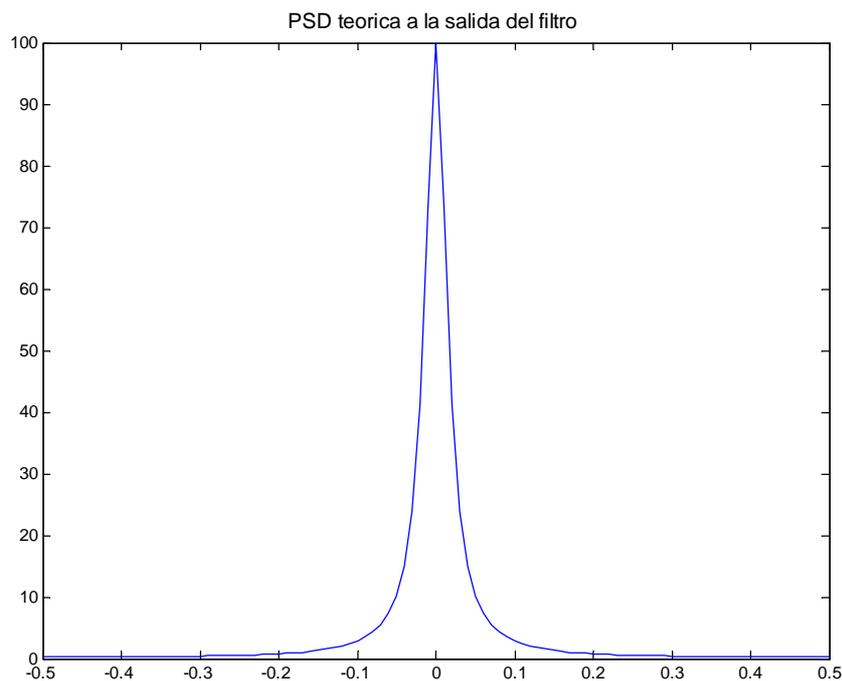


Figura 10 – Densidad espectral de potencia teórica a la salida del filtro.

El filtro $h(n)$ Está caracterizado por la ecuación en diferencias

$$y_n = 0.9y_{n-1} + x_n \quad n \geq 0, \quad y_{-1} = 0$$

Cuestión teórica 6: Utilizando la ayuda de MatLab acerca de la función `filter` indicar cómo realizar en MatLab el filtrado del proceso de una secuencia x obteniendo como salida la secuencia y .

Cuestión práctica 5: Generar una secuencia de 1000 muestras de un proceso aleatorio blanco independientes e idénticamente distribuidas de función de densidad de probabilidad uniforme en $[-1/2, 1/2]$. Hacer pasar dicha secuencia por el filtro antes descrito.

2.5 Procesos paso bajo y paso banda

Se denomina proceso aleatorio paso bajo aquél que tiene su PSD despreciable en altas frecuencias. Es decir, tiene su potencia concentrada en las bajas frecuencias (las cercanas a 0).

Cuestión práctica 6: Para la secuencia generada en la cuestión anterior y para el filtro utilizado calcular las autocorrelaciones mediante la función antes diseñada `aucor` y las PSD de la entrada y de la salida. Dibujar los resultados.

Cuestión práctica 7: Diseñar un fichero llamado `p_bajo` que realice estas operaciones (generar la entrada, pasar por el filtro y calcular autocorrelaciones y PSD) 100 veces y promedie los resultados. Dibujar los resultados promediados, comprobando que se trata de un proceso paso bajo.

Un proceso aleatorio se denomina paso banda cuando su PSD es grande en una banda de frecuencias próxima a una frecuencia central $\pm f_0$ y despreciable fuera de esa banda. Si el ancho de la banda es B se denomina proceso de banda estrecha a un proceso paso banda para el que $B \ll f_0$.

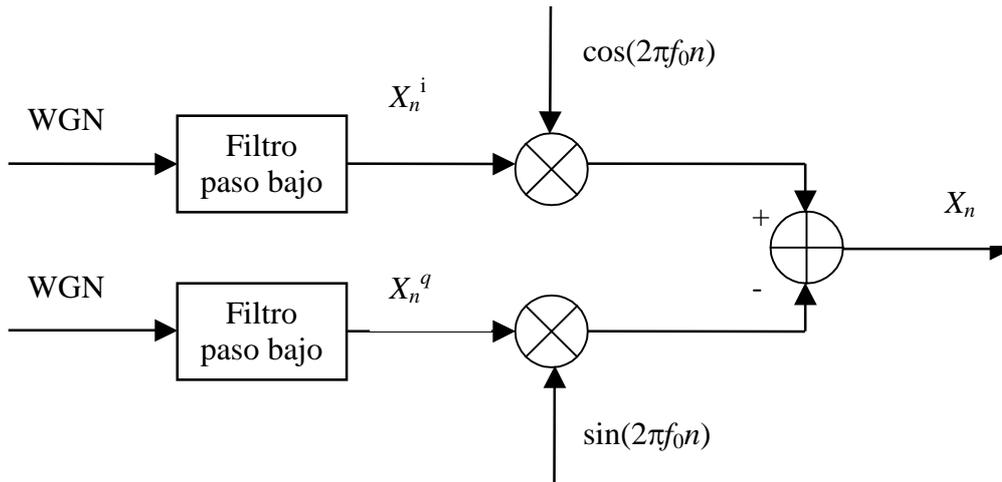
En comunicaciones la señal de información suele ser un proceso aleatorio paso bajo que modula una portadora para su transmisión por un canal de comunicación paso banda (generalmente de banda estrecha).

El proceso paso banda puede representarse como

$$X(t) = X_i(t) \cos(2\pi f_0 t) - X_q(t) \sin(2\pi f_0 t)$$

donde $X_i(t)$ y $X_q(t)$ son las componentes en fase y cuadratura de $X(t)$. Dichas componentes son procesos paso bajo.

Generaremos ahora un proceso paso banda discreto a partir de dos procesos paso bajo discretos. Partimos de dos procesos Gaussianos blancos. El esquema de generación de X_n (el proceso paso banda) es el siguiente:



Cuestión práctica 8: Generar 1000 muestras de un proceso paso banda con frecuencia central f_0 de 0.3 ciclos a partir de dos procesos Gaussianos blancos (de media 0 y varianza 1) tal como se indica en el esquema. Dibujar los espectros de densidad de potencia del proceso paso banda así como los de las componentes en fase y en cuadratura. (Utilice el filtro paso bajo del apartado 2.4 y realice promedios de 100 veces).

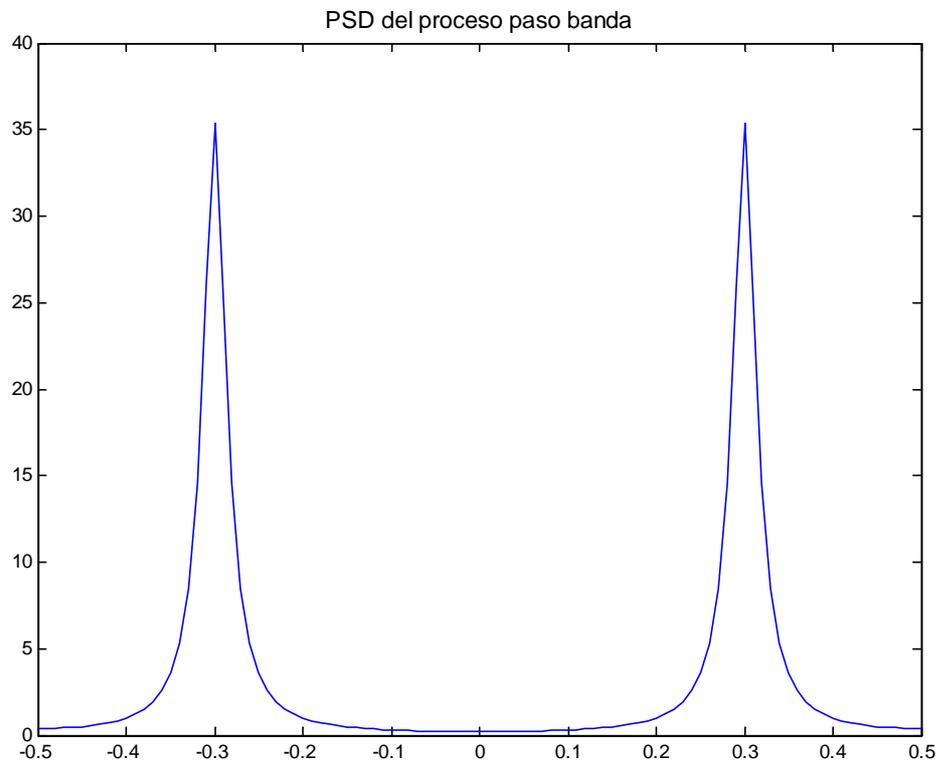


Figura 11 – Densidad espectral de potencia teórica del proceso paso banda.

3 Hoja de resultados. NOMBRE: _____

Cuestión teórica 1:

Cuestión teórica 2:

Cuestión teórica 3:

Cuestión práctica 1:

Cuestión teórica 4:

Cuestión práctica 2:

Cuestión práctica 3:

Cuestión práctica 4:

Cuestión teórica 5:

Cuestión teórica 6:

Cuestión práctica 5:

Cuestión práctica 6:

Cuestión práctica 7:

Cuestión práctica 8: