Hoja de ejercicios 4

Corredor Vegas, Guillermo
Martín Gallego, Eugenio
Montoya Manosalvas, Michell

June 1, 2018

Ejercicio 1. Operadores Booleanos en Matlab

Para una simulación de 10 pasos con 10 trayectorias independientes, con límites superior $B_{up} = B_0 + \mu + \sigma$ e inferior $B_{down} = B_0 - \mu - \sigma$

El objetivo de este ejercicio es explorar el uso de operadores Booleanos en Matlab

```
%Definicin de las condiciones iniciales
t0 = 2.0; T = 3.0; N = 10; M = 100;
mu = 0; sigma = 1; B0 = 100;
B = ArithmeticBrownian(t0,T,M,N,mu,sigma,B0);
Bup = B0+mu+sigma; Blow = B0-mu-sigma;
out = zeros(M,5);
for i = 1:M    %comprueba las condiciones especificadas (estan en orden)
    out(i,1) = any(B(i,:)>Bup);
    out(i,2) = any(B(i,:)>Bup) || any(B(i,:)<Blow);
    out(i,3) = (any(B(i,:)>Bup)) && (any(B(i,:)<Blow));
    out(i,4) = all(B(i,:)<Bup);
    out(i,5) = (all(B(i,:)<Bup)) || (all(B(i,:)>Blow));
end
```

(i) En algún momento han estado por encima de B_{up}.

```
>> out(1:10,1)'

ans =

1×10 logical array

1   1   0   1   0   1   0   0   1   1
```

(ii) En algún momento han estado por encima de B_{up} o por debajo de B_{low}.

```
>> out(1:10,2)'

ans =

1×10 logical array

1   1   1   1   0   1   0   0   1   1
```

(iii) En algún momento han estado por encima de B_{up} y en algún otro por debajo de B_{low}.

```
>> out(1:10,3)'

ans =

1×10 logical array

0   0   0   0   0   0   0   0   0   0
```

(iv) Nunca han estado por encima de B_{up}.

```
>> out(1:10,4)'

ans =

1×10 logical array

0   0   1   0   1   0   1   1   0   0
```

(v) Nunca han estado por encima de B_{up} o por debajo de B_{low}.

```
>> out(1:10,5)'

ans =

1×10 logical array

1   1   1   1   1   1   1   1   1   1
```

Ejercicio 2. Valoración por MC de una call europea sobre una cesta sobre varios subyacentes con correlaciones.

(i) Encontrar la descomposición de Cholesky para la matriz de correlaciones. Es decir, encontrar la matriz triangular inferior L, tal que $\rho = L' \cdot L$

$$u_t = X_t + \phi_0$$

$$\rho = L' \cdot L \begin{pmatrix} 1 & \rho_{12} & \rho_{13} \\ \rho_{12} & 1 & \rho_{23} \\ \rho_{13} & \rho_{23} & 1 \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & 0 \\ L_{12} & L_{22} & 0 \\ L_{13} & L_{23} & L_{33} \end{pmatrix} \cdot \begin{pmatrix} L_{11} & L_{12} & L_{13} \\ 0 & L_{22} & L_{23} \\ 0 & 0 & L_{33} \end{pmatrix} =$$

$$\begin{pmatrix} L_{11}^2 & L_{11} \cdot L_{12} & L_{11} \cdot L_{13} \\ L_{12} \cdot L_{11} & L_{12}^2 \cdot L_{22}^2 & L_{11} \cdot L_{13} + L_{22} \cdot L_{23} \\ L_{13} \cdot L_{11} & L_{13} \cdot L_{12} + L_{23} \cdot L_{22} & L_{13}^2 L_{23}^2 L_{33}^2 \end{pmatrix}$$

$L_{11}^2 = 1 \Rightarrow L_{11} = 1$
$L_{12} \cdot L_{11} = \rho_{12} \Rightarrow L_{12} = \rho_{12}/L_{11}$
$L_{13} \cdot L_{11} = \rho_{13} \Rightarrow L_{13} = \rho_{13}/L_{11}$
$L_{12}^2 \cdot L_{22}^2 = 1 \Rightarrow L_{22} = \sqrt{1 - L_{12}^2}$
$L_{13} \cdot L_{12} + L_{23} \cdot L_{22} = \rho_{23} \Rightarrow L_{23} = \frac{\rho_{23} - L_{13} \cdot L_{12}}{L_{22}}$
$L_{13}^2 + L_{23}^2 + L_{33}^2 = 1 \Rightarrow L_{33} = \sqrt{1 - L_{13}^2 - L_{23}^2}$

$$L = \begin{pmatrix} 1 & 0 & 0 \\ \rho_{12} & \sqrt{1 - L_{12}^2} & 0 \\ \rho_{13} & \frac{\rho_{23} - L_{13} \cdot L_{12}}{L_{22}} & \sqrt{1 - L_{13}^2 - L_{23}^2} \end{pmatrix} \quad L' = \begin{pmatrix} 1 & \rho_{12} & \rho_{13} \\ 0 & \sqrt{1 - L_{12}^2} & \frac{\rho_{23} - L_{13} \cdot L_{12}}{L_{22}} \\ 0 & 0 & \sqrt{1 - L_{13}^2 - L_{23}^2} \end{pmatrix}$$

(ii) Utilizando la función de Matlab *chol*, escribid una función en Matlab que estime el precio de una cesta sobre varios subyacentes y su error mediante Montecarlo.

```
function [precio,error] = precioCesta_MC(M,S0,K,r,T,sigma,Rho)
%precioCesta_MC: Estimacin del precio de una cesta por Monte Carlo
%
%% SINTAXIS: [precio,error] = precioCesta_MC(M,S0,r,Rho,sigma,K,T)
%
%%  INPUT:
%    precio: estimacin MC del precio de una call sobre cesta de
%            subyacentes (con los mismos pesos)
%    error : estimacin del error MC
%
%%  OUTPUT:
%        M : Nmero de trayectorias a simular
%       S0 : Precios iniciales del subyacente (nSubyacente x 1)% K : Precio
    de ejercicio
%        r : Tipo de  inters libre de riesgo
```

```
%        T : Vencimiento
%    sigma : Volatilidades (nSubyacente x 1)
%      Rho : Matriz de correlaciones (nSubyacente x nSubyacente)
%
% Ejemplo 1:
% S0 = [100 100 100]; sigma = [0.3 0.3 0.3];
% r = 0.05; T = 2; K = 100;
% Rho = ones(3)%+eps*eye(3);
% % ¿Qu ocurre si quitamos a Rho el trmino proporcional a eps?
% M = 5e6;
% [precio, error] = precioCesta_MC(M,S0,K,r,T,sigma,Rho)
% blsprice(S0(1),K,r,T,sigma(1)) % ¿Por qu?
%
% Ejemplo 2:
% S0 = [100 120 80]; sigma = [0.3 0.2 0.4];
% r = 0.05; T = 2; K = 100;
% Rho = [1 0.3 -0.5; 0.3 1 0.4; -0.5 0.4 1];
% M = 1e6;
% [precio, error] = precioCesta_MC(M,S0,K,r,T,sigma,Rho)
%
nSubyacentes = length(S0); % Nmero de subyacentes en la cesta
L = chol(Rho,'lower'); % Para n Subyacentes

%% generate M samples for N underlying assets from N(0,1)
X = randn(nSubyacentes,M);
%% simulates M trajectories for N underlying assets
ST = S0'.*exp((r-0.5*sigma'.^2)*T+sigma'.*L*X*sqrt(T));
%% define payoff
payoff = max(mean(ST,1)-K,0);
%% MC estimate
discountFactor = exp(-r*T);

precio = discountFactor*mean(payoff);
error = sqrt(sum((payoff-precio).^2)./(M-1))./sqrt(M);
```

Ejecutando el ejemplo 1 se obtienen los resultados:

```
precio = 21.2033
error = 0.0173
```

Si se quita el término proporcional a *eps* entonces al realizar la descomposición de Cholesky va a intentar descomponer una matriz no invertible (formada por vectores linealmente dependientes). Se necesita una matriz definida positiva para realizar esta descomposición, y lo anteriormente expuesto muestra que no lo es.

Este precio coincide con el obtenido para un subyacente de precio inicial S_0 y σ igual al de los tres subyacentes de la cesta con correlación uno entre todos los elementos. Esto se debe a que la función del precio es lineal con los subyacentes, y es equivalente una suma ponderada de los subyacentes a tener solo el subyacente.

Ejercicio 3. Ajustes por máxima verosimilitud.

Demostrar que los parámetros ϕ_0 , ϕ_1 que se obtienen maximizando la verosimilitud coinciden con los que se obtienen minimizando el error cuadrático medio para un ajuste a los datos $\{X_t; t = 0, 1, 2, ..., T\}$ a un modelo lineal.

$$X_t = \phi_0 + \phi_1 X_{t-1} + u_t \;\Rightarrow\; u_t = X_t - \phi_1 X_{t-1} - \phi_0$$

$$normpdf(X_t - \phi_1 X_{t-1} - \phi_0 \ ; \ 0 \ ; \ \sigma) \ = \ \frac{1}{\sqrt{2\pi\sigma^2}} \, exp \left\{ -\frac{(X_t - \phi_1 X_{t-1} - \phi_0)^2}{2\sigma^2} \right\}$$

$$L(\phi_0; \phi_1; \{X_t\}_{t=0}^{T}) = \prod_{t=1}^{T} \ normpdf(X_t - \phi_1 X_{t-1} - \phi_0 \ ; \ 0 \ ; \ \sigma) \ = \ \frac{1}{\sqrt{2\pi\sigma^2}} exp \left\{ -\frac{(X_t - \phi_1 X_{t-1} - \phi_0)^2}{2\sigma^2} \right\} =$$

$$logL(\phi_0 \, ; \phi_1; \{X_t\}_{t=0}^{T}) = -T log(\sigma\sqrt{2\pi}) - \sum_{t=1}^{T} \frac{(X_t - \phi_1 X_{t-1} - \phi_0)^2}{2\sigma^2}$$

Al realizar la derivada de la función de verosimilitud e igularla a cero se obtiene los mismos parámetros que minimizando el error cuadrático medio, ya que las constantes que hacen que se diferencien ambos se pueden eliminar. Los parámetros obtenidos son:

$$\hat{\phi_0} = \frac{1}{T} \left(\sum_{t=1}^{T} X_t - \sum_{t=1}^{T} X_{t-1} \right)$$

$$\hat{\phi_1} = \frac{\sum_{t=1}^{T} X_{t-1} \, (X_{t-1} - \phi_0)}{\sum_{t=1}^{T} X_{t-1}^2}$$

¿Cómo se obtendría una estimación del valor de σ?

Una estimación de σ se obtendría a través de su estimador máximo verosímil, este se obtiene maximizando la función de verosimilitud.

$$normpdf(u \ ; \ 0 \ ; \ \sigma) \ = \ \frac{1}{\sqrt{2\pi\sigma^2}} \, exp \left\{ -\frac{u^2}{2\sigma^2} \right\}$$

$$L(\phi_0 \, ; \phi_1; \{X_t\}_{t=0}^{T}) = \prod_{t=1}^{T} \ normpdf(u_t \ ; \ 0 \ ; \ \sigma) \ = \ \frac{1}{(2\pi\sigma^2)^{\frac{T}{2}}} \, exp \left\{ -\frac{\sum_{t=1}^{T} u_t^2}{2\sigma^2} \right\}$$

$$logL(\phi_0 \, ; \phi_1; \{X_t\}_{t=0}^{T}) = -\frac{T}{2} \, log \, (\sigma^2) \ - \frac{T}{2} log(\, 2\pi) \ - \frac{1}{2\sigma^2} \sum_{t=1}^{T} u_t^2$$

$$\frac{\partial L}{\partial \sigma^2} = \frac{-T}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{t=1}^{T} u_t^2 = 0$$

$$\hat{\sigma} = \sqrt{\frac{\sum_{t=1}^{T} \hat{u^2}}{T}} = \sqrt{\frac{\sum_{t=1}^{T} (X_t - \hat{\phi_1} X_{t-1} - \hat{\phi_0})^2}{T}}$$

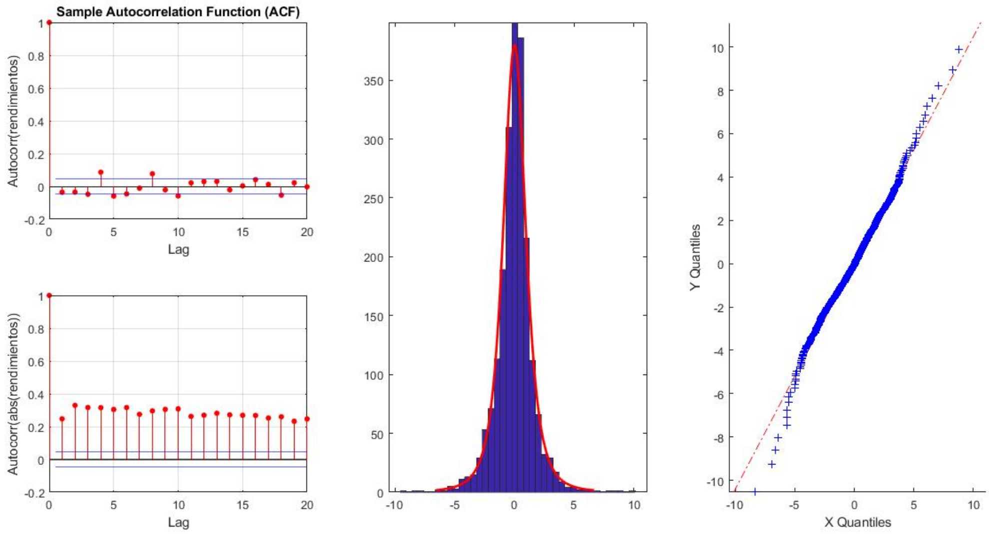
Ejercicio 4. Modelos no normales para los rendimientos del IBEX.

Basándote en el código proporcionado,

(i) Estima los parámetros de una distribución t de Student por máxima verosimilitud. Los parámetros son μ, σ y υ. Haz una gráfica similar a "Figure 1" para comprobar si se trata de un buen modelo y comenta los resultados.

Los parámetros estimados son:

```
mu = 0.0425
sigma = 0.9448
nu = 2.9083
```



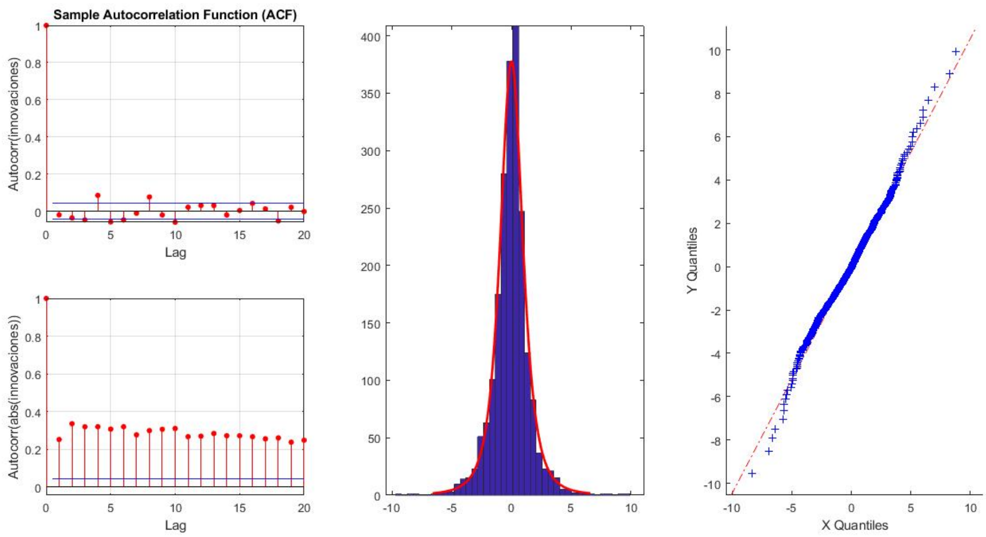
Se corresponden con una t-student de 2.9 grados de libertad. Las correlaciones entre los rendimientos son bajas, por lo tanto no hay estructura autorregresiva en la serie. Se observa que existen autocorrelaciones entre los valores absolutos de los rendimientos, por lo que la serie es heterocedástica, lo que provoca una pérdida de eficiencia en la estimación de los parámetros de la distribución. En el histograma de los rendimientos se observa un mejor ajuste a esta distribución que a la distribución normal utilizando los tres modelos anteriores. Esto se debe a que la t-student refleja mejor las colas pesadas que la distribución normal, las cuales son característcas de los rendimientos en los mercados financieros.
El gráfico Q-Q es utilizado para comprobar si la distribución de los rendimientos ha sido correctamente especificada. En ella se representa en cada eje los cuantiles de la distribución correspondiente. Consecuentemente si ambas distribuciones son iguales tendrán el mismo peso en el cuantil, y los puntos se situarán sobre la recta. En nuestro caso, tipificaremos la distribución de los parámetros estimados, de tal forma que si está correctamente especificada entonces tendrá la misma distribución que una normal de media 0 y varianza 1. En este caso se observa que define correctamente la mayor parte de la distribución, excepto las colas, donde no está correctamente definida sus distribuciones (son más pesadas que las estimadas).

(ii) Estima los parámetros de un modelo AR(1) suponiendo que las innovaciones son independientes siguen una distribución t de Student.Haz una gráfica similar a "Figure 1"para las innovaciones con el fin de comprobar si se trata de un buen modelo y comenta los resultados.

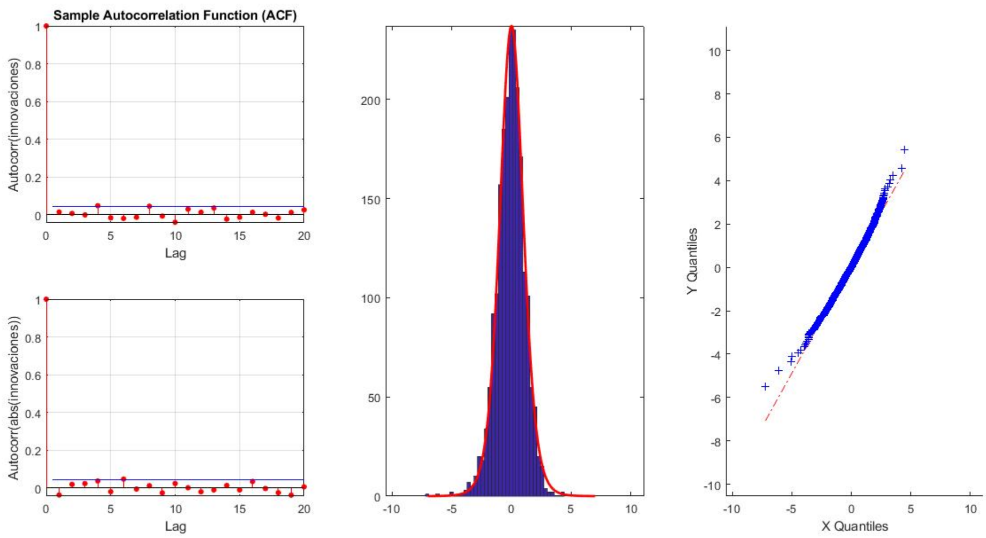
Los parámetros estimados son:

```
phi0 = 0.0441
phi1 = -0.0170
sigma = 0.9438
nu =  2.9019
```

Realizando la estimación del modelo AR(1) se han obtenido estos datos. El análisis es equivalente al caso anterior, aunque parece que define ligeramente mejor las colas pesadas de la distribución (qqplot). Este modelo define la media condicional del proceso, por lo tanto no se elimina la heterocedasticidad del modelo.



(iii) Estima los parámetros de un modelo $AR(1)+GARCH(1,1)$ suponiendo que las innovaciones son independientes siguen una distribución t de Student.Haz una gráfica similar a "Figure 1" para $\epsilon_t$con el fin de comprobar si se trata de un buen modelo. Comentad los resultados.



Los parámetros estimados son:

```
phi0 = 0.0826
phi1 = -0.0287
kappa = 0.0075
alpha = 0.0660
beta = 0.9120
nu = 8.4041
```

Aplicando el modelo AR(1)+GARCH(1,1) se mantiene la no existencia de estructura autorregresiva en los datos, y la no existencia de autocorrelaciones entre los valores absolutos de los rendimientos implica que hay homocedasticidad en las varianzas de los rendimientos, dando una mejor estimación de los parámetros. Esto se puede observar en el ajuste de la función de densidad, que es más preciso con el

histograma que los modelos anteriores. Han aumentado el número de grados de libertad de 2.9 a 8.4, es decir, ha aumentado el número de parámetros en el cálculo que pueden variar.
Los coeficientes del término autorregresivo (ϕ_0, ϕ_1) son el doble que en el modelo AR(1), y su bajo valor indica que pueden no ser relevantes en la simulación de los rendimientos (el presente no se puede decir a partir del pasado). El alto valor del término β respecto al resto implica una mayor importancia del término h, el término que modela la varianza de la volatilidad. La volatilidad media de este modelo es $\sigma = 11749$, mayor que la volatilidad estimada del modelo AR, $\sigma_{AR} = 0.94$. Al introducir la variabilidad de la varianza en el tiempo en función del elemento pasado esta ha aumentado.

Ejercicio 5. Modelo de Nelson-Siegel.

Estimad los parámetros del modelo de Nelson-Siegel (b_0, b_1, b_2, τ) que minimizan elerror cuadrático medio (ECM). Compara en una gráfica el modelo estimado con los datos.Haced la gráfica del tipo de interés spot $r_{NS}(t)$ en función del tiempo.

```
function [r,precioBono] = NelSieg(t,b0,b1,b2,tau,B)
% NelSieg: Modelo TI Nelson y Siegel
%
% SINTAXIS:
% [r,precioBono] = NelSieg(t,b0,b1,b2,tau,B)
%
% r : Vector de tipos de inters futuros calculados
% por NS
% precioBono : Vector de precios de bonos que paga B
% calculados por NS.
% B : Pago del bono a vencimiento
% t : Vector de tiempos
% parameters : Vector de parmetros del modelo Nelson-Siegel
% [b0, b1, b2, tau]
% Se supone que b0,b1,b2 estn en % y tau en aos
%
% EJEMPLO 1:
% t = [1/12 3/12 6/12 1 3 5 10] ;
% b0 = 4; b1 = -2; b2 = -5; tau = 0.5; B = 100;
% [r,precioBonoNS] = NelSieg(t,b0,b1,b2,tau,B);
%
% for n = 1: length(t)
% precioBonoNS_quadl(n) = ...
% B*exp(-0.01*quadl(@(t)NelSieg(t,b0,b1,b2,tau,B),0,t(n)));
% end
% precioBonoNS_quadl
% precioBonoNS % Deberia dar el mismo valor
%
r = b0+b1*exp(-t/tau)+b2*t.*exp(-t/tau)/tau;
integral = b0*t+b1*tau*(1-exp(-t/tau))+b2*tau-b2*(t+tau).*exp(-t/tau);
precioBono = B*exp(-integral/100);
```

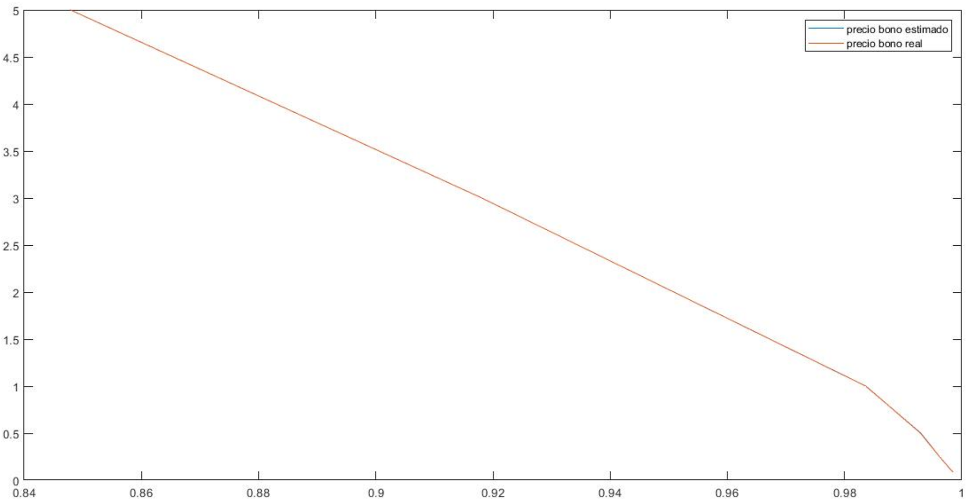
```
function ECM = ECM_BonoNS(parameters,t,precioBono,B)
% ECM_BonoNS
%
% parameters : Vector de parmetros del modelo Nelson-Siegel
% [b0, b1, b2, tau]
% t : Vector de tiempos
% precioBono : Vector de precios de bonos.
```

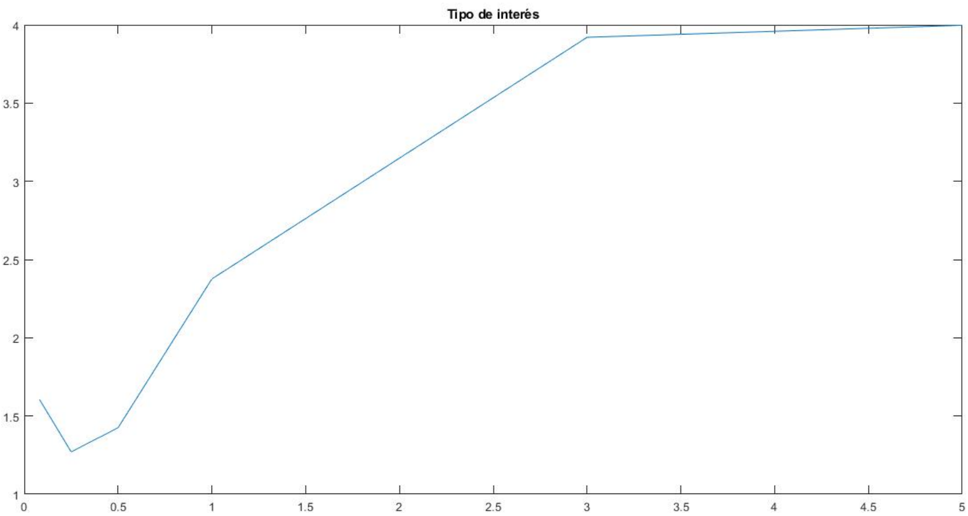
```
%
% EJEMPLO:
%
% t = [1/12 1/4 1/2 1 3 5 ]
% precioBono = [0.9985 0.9962 0.9930 0.9836 0.9181 0.8479]
% B = 1;
% b0 = 4 ; b1 = -2; b2 = -5; tau = 0.5;
% parameters = [b0 b1 b2 tau];
% ECM = ECM_BonoNS(parameters,t,precioBono,B)
%%
if exist('options','var') == 0 | isempty(options)
    options = optimset('fmincon');
    options = optimset(options , 'Diagnostics' , 'on');
    options = optimset(options , 'Display'     , 'iter');
    options = optimset(options , 'LargeScale'  , 'off');
end
[~,precioBonoNS] = NelSieg(t,parameters(1),parameters(2),parameters(3),
    parameters(4),B);

ECM = mean((precioBono-precioBonoNS).^2);
```

```
t = [1/12 1/4 1/2 1 3 5 ]
precioBono = [0.9985 0.9962 0.9930 0.9836 0.9181 0.8479]
B = 1;
b0 = 4 ; b1 = -2; b2 = -5; tau = 0.5;
parameters = [b0 b1 b2 tau];
x0 = parameters;
fun = @(parameters)ECM_BonoNS(parameters,t,precioBono,B);
[parameters,ECM] = fminunc(fun,x0);
parameters
[r,precioBonoNS] = NelSieg(t,parameters(1),parameters(2),parameters(3),
    parameters(4),B)
figure(1);plot(t,r);
figure(2);plot(precioBono,t,precioBonoNS,t)
```

Los parámetros estimados coinciden prácticamente con las condiciones iniciales, ya que el error cuadrático medio es muy pequeño para estos coeficientes $ECM = 6.4736e-10$. La representación del modelo estimado con los datos y del tipo de interés del modelo Nelson-Siegel calculado:


precio bono estimado
precio bono real


Tipo de interés