

Caso de estudio: modelos autoregresivos en IBEX35 (1993-2013)

(c) Antonio Sala Piqueras, 2019. Todos los derechos reservados.

Presentación en vídeo: <http://personales.upv.es/asala/YTV/ibexabs.html> , <http://personales.upv.es/asala/YTV/ibexinc.html> .

Este código funcionó correctamente en Matlab R2019a

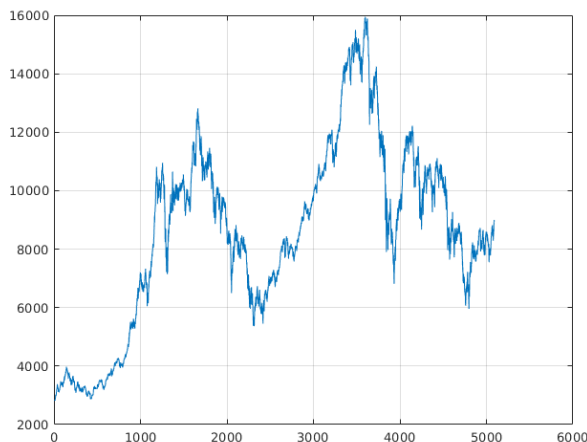
Motivación: predecir los índices bursátiles podría generar beneficio económico. Usemos la predicción de series temporales (modelos autoregresivos) de la System ID Toolbox de Matlab para intentarlo.

Tabla de Contenidos

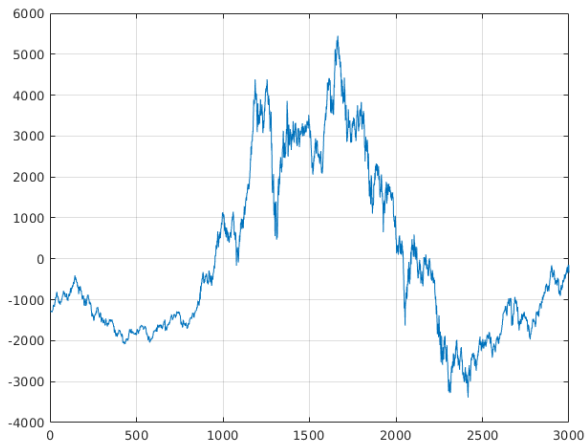
Carga y preprocesado de datos.....	1
Identificación serie temporal.....	2
Análisis del resultado: ¿Nos vamos a hacer ricos?.....	3
Intervalos de confianza.....	5
peso en frecuencia: pasar a incrementos.....	6
Conclusión.....	8

Carga y preprocesado de datos

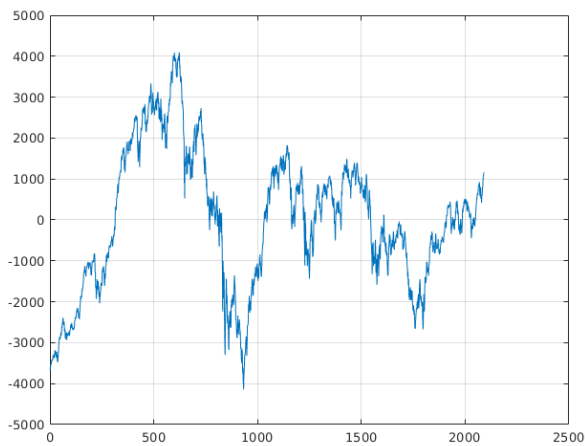
```
load ibex2.mat
plot(ibex2), grid on
```



```
ibex2id=detrend(ibex2(1:3000));
%quitar "linea" largo plazo ascenso. Objetivo del modelo "decidir invertir a corto plazo"
plot(ibex2id),grid on
```



```
ibex2val=detrend(ibex2(3001:end)); %datos de validación
plot(ibex2val), grid on
```



Identificación serie temporal

Identifiquemos un modelo de serie temporal (sin entradas):

```
IDD=iddata(ibex2id,[],1);
IDDv=iddata(ibex2val,[],1);
th=arx(IDD,5)
```

th =

Discrete-time AR model: $A(z)y(t) = e(t)$

$A(z) = 1 - 1.029 z^{-1} + 0.08011 z^{-2} - 0.03508 z^{-3} - 0.03031 z^{-4} + 0.01562 z^{-5}$

Sample time: 1 seconds

Parameterization:

Polynomial orders: na=5

Number of free coefficients: 5

Use "polydata", "getpvec", "getcov" for parameters and their uncertainties.

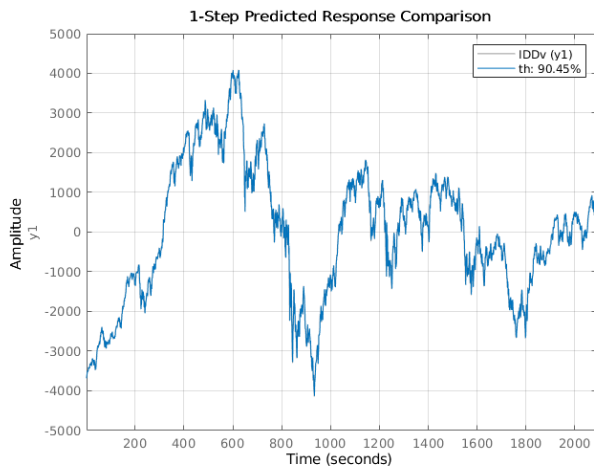
Status:

Estimated using ARX on time domain data "IDD".

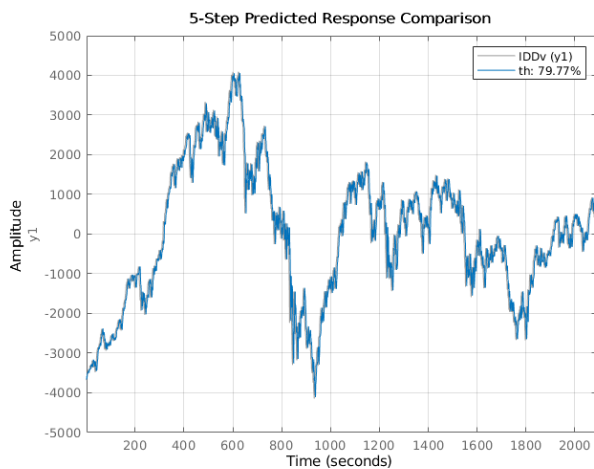
Fit to estimation data: 94.8% (prediction focus)
FPE: 1.144e+04, MSE: 1.137e+04

Comparemos los datos con la predicción del modelo a un cierto horizonte (usando datos de validación)

```
compare(IDDv,th,1), grid on
```



```
compare(IDDv,th,5), grid on
```



Análisis del resultado: ¿Nos vamos a hacer ricos?

Comparemos el modelo identificado con un modelo "trivial":

Predicción 1: mañana estará como predice mi modelo *autoregresivo*, aparte de la tendencia "a 10 años vista" que quitó `detrend`.

Predicción 2: mañana estará "más o menos igual que hoy"... aparte de la tendencia "a 10 años vista" que quitó `detrend`

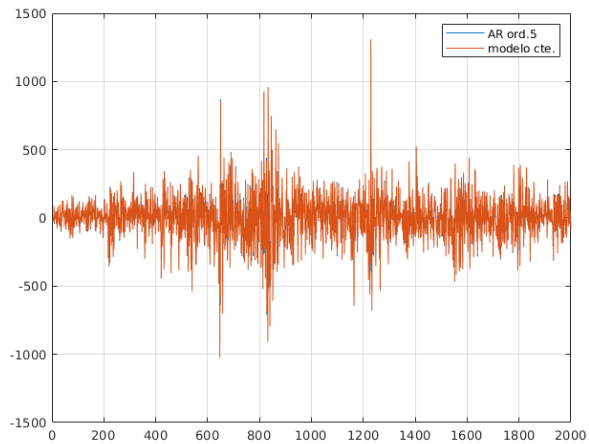
```
orden=5;errpre=zeros(1,2000);  
errpre2=zeros(1,2000);  
predix=zeros(1,2000);
```

```

for i=(orden+1):2000
    predix(i)=-th.A(2:end)*ibex2val((i-1):-1:(i-orden));
    errpre(i)=ibex2val(i)-predix(i);
    errpre2(i)=ibex2val(i)-ibex2val(i-1);
end

plot([errpre; errpre2]'), grid on, legend('AR ord.5','modelo cte.')

```



```
sum(abs(errpre)/4000) %error AR orden 5 (mínimos cuadrados)
```

```
ans = 57.9332
```

```
sum(abs(errpre2)/4000) %error modelo constante
```

```
ans = 58.0133
```

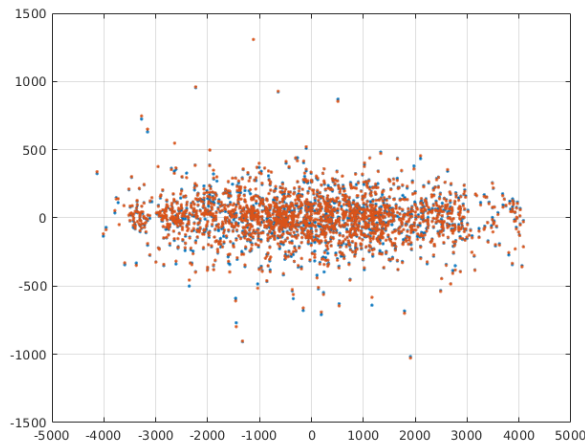
```
sum(abs(errpre.^2)/4000) %error AR orden 5 (mínimos cuadrados)
```

```
ans = 1.3337e+04
```

```
sum(abs(errpre2.^2)/4000) %error modelo constante
```

```
ans = 1.3379e+04
```

```
plot(predix,[errpre; errpre2]','.'), grid on
```



No detectamos ninguna "no-linealidad" en este plotresiduals.

Intervalos de confianza

```
desvtip_cadaparam=sqrt(diag(getcov(th)))'
```

```
desvtip_cadaparam = 1×5
    0.0183    0.0262    0.0263    0.0262    0.0183
```

```
th.A-[0 desvtip_cadaparam*2] %limite superior
```

```
ans = 1×6
    1.0000   -1.0655    0.0276   -0.0876   -0.0828   -0.0210
```

```
th.A+[0 desvtip_cadaparam*2] %limite inferior
```

```
ans = 1×6
    1.0000   -0.9924    0.1326    0.0175    0.0222    0.0522
```

El cero está incluido en 3 de los 5 intervalos conv. de parametros (cf. 95%).

Probamos orden 2:

```
th2=armax(IDD,[2 0]);
roots(th2.A)
```

```
ans = 2×1
    0.9985
    0.0295
```

```
desvtip_cadaparam2=sqrt(diag(getcov(th2)))';
th2.A-[0 desvtip_cadaparam2*2]
```

```
ans = 1×3
    1.0000   -1.0654   -0.0080
```

```
th2.A+[0 desvtip_cadaparam2*2]
```

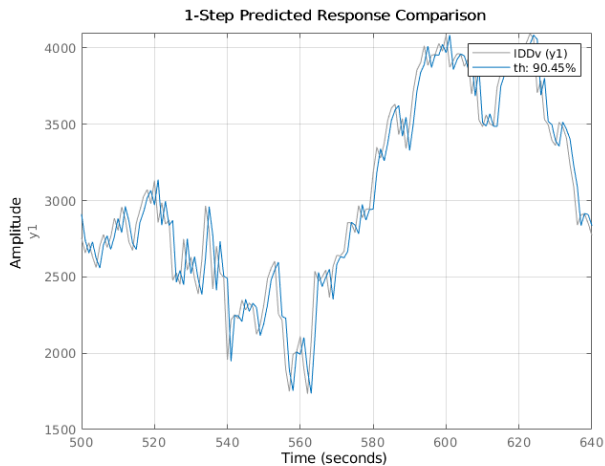
```
ans = 1×3
```

1.0000 -0.9905 0.0668

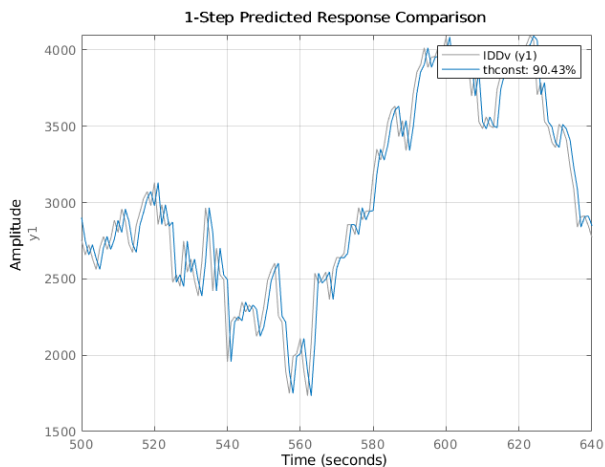
El cero está en el intervalo del segundo parámetro... y el uno en el primero... podría ser $y_k = y_{k-1}$...

En efecto, haciendo zoom en los "compare" tampoco parece que se haga mucho más que ese modelo "constante":

```
compare(IDDv,th,1), grid on, axis([500 640 1500 4100])
```



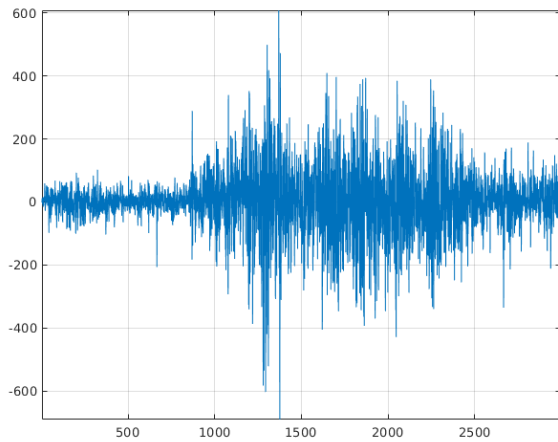
```
thconst=th;thconst.A=[1 -1 0 0 0 0];  
compare(IDDv,thconst,1), grid on, axis([500 640 1500 4100])
```



peso en frecuencia: pasar a incrementos.

Como nos interesa modelar el ibex en incrementos, repetimos experimento pero con salidas incrementales.

```
inc_ibexID=detrend(diff(ibex2id),'constant');  
%quitamos incrementos "constantes" (pendiente recta en coord. orig).  
plot(inc_ibexID), axis tight, grid on
```



```
IDDincrem=iddata(inc_ibexID,[],1);
thincr=arx(IDDincrem,3)
```

```
thincr =
Discrete-time AR model:  $A(z)y(t) = e(t)$ 
 $A(z) = 1 - 0.02943 z^{-1} + 0.05124 z^{-2} + 0.01488 z^{-3}$ 
```

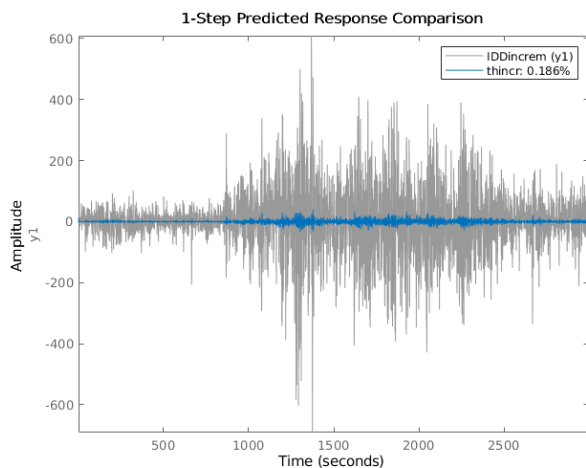
Sample time: 1 seconds

Parameterization:
 Polynomial orders: na=3
 Number of free coefficients: 3
 Use "polydata", "getpvec", "getcov" for parameters and their uncertainties.

Status:
 Estimated using ARX on time domain data "IDDincrem".
 Fit to estimation data: 0.186% (prediction focus)
 FPE: 1.143e+04, MSE: 1.138e+04

Comparemos los datos con la predicción del modelo a un cierto horizonte

```
compare(IDDincrem,thincr,1)
```



Vemos que el "fit" en variables absolutas de >90% se convierte en un fit del 0.1% HORRIBLE cuando queremos modelar los incrementos.

Los intervalos de confianza del modelo "incremental" resultan:

```
desvtip_cadaparami=sqrt(diag(getcov(thincr)))';  
thincr.A-[0 desvtip_cadaparami*2]
```

```
ans = 1×4  
1.0000 -0.0660 0.0147 -0.0217
```

```
thincr.A+[0 desvtip_cadaparami*2]
```

```
ans = 1×4  
1.0000 0.0071 0.0878 0.0515
```

Conclusión

Pese a un buen "fit" en coordenadas absolutas, el resultado no era muy diferente de un modelo "constante"; en coordenadas incrementales no tiene mucha capacidad de predicción, y no podemos desmentir la "hipótesis nula" de que los datos han sido generados por $\Delta x_k = 0$. Los métodos de mínimos cuadrados hasta orden 5 sobre modelos lineales no parecen muy útiles en este problema.