

Lección 5.C - Simulación de procesos AR(1) y exploración de sus correlogramas

Marcos Bujosa

1. Objetivo de la práctica

Guión: [P-L05-C-simulacion-procesos-AR.inp](#)

Objetivo

1. Observar la ACF y PACF de distintos modelos AR(1).

- Observar que el signo y la magnitud de θ_1 afecta al signo y magnitud de la autocorrelación de orden 1.
- Observar que el signo y la magnitud de θ_1 afecta al comportamiento de la PACF y su velocidad de decaimiento.

Requerimientos previos

Programe o recupere de una práctica anterior una función que simule procesos AR(q)

```
function series SimuladorAR(matrix phi)
    # SimuladorAR(phi) simula un proceso AR(p),
    # donde phi es el polinomio AR y p es su grado.
    scalar p = cols(phi)
    series U = normal(0,1)
    series Y = 0
    setinfo Y --description="Serie simulada"
    loop i = (p+1)..$nobs
        scalar comb_pasado_Yt = 0
        scalar perturbacion = U[i]
        loop j = 2..p
            comb_pasado_Yt += -phi[1,j] * Y[i-j+1]  # expresión abreviada
        endloop
        Y[i] = comb_pasado_Yt + perturbacion
    endloop
    return Y
end function
```

Para que se observe bien la estructura de las ACF y PACF estimadas, establezca un tamaño de muestra suficientemente grande.

```
# establecemos la muestra
nulldata 1500
setobs 12 1900:01 --time-series
```

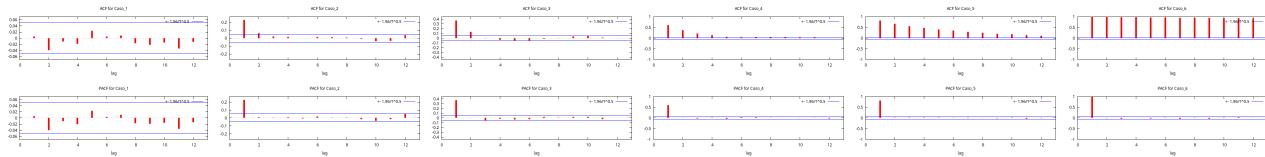
2. Actividad 2 - Modelos AR(1) parámetro positivo

Pruebe a simular modelos AR(1)

$$(1 - \theta B) * \mathbf{X} = \mathbf{U}$$

donde $\mathbf{U} \sim WN(0, 1)$, con valores paramétricos: entre 0 y 1; y estime los correlogramas.

- Para $\theta_1 \approx 0, 0.2, 0.4, 0.6, 0.8, 1$.



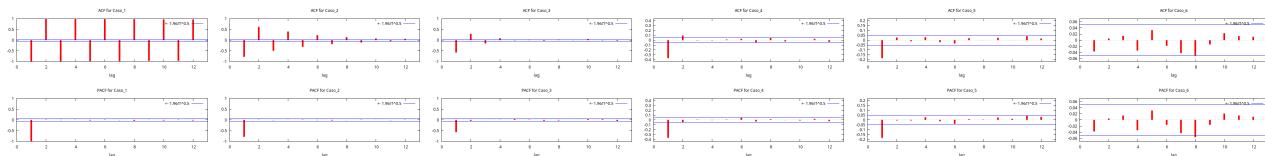
3. Actividad 1 - Modelos AR(1) parámetro negativo

Pruebe a simular modelos AR(1)

$$(1 - \theta B) * \mathbf{X} = \mathbf{U}$$

donde $\mathbf{U} \sim WN(0, 1)$, con valores paramétricos: entre -1 y 0; y estime los correlogramas.

- Para $\theta_1 \approx -1, -0.8, -0.6, -0.4, -0.2, 0$.



4. Código completo de la práctica

P-L05-C-simulacion-procesos-AR.inp

```
# _____
# Copyright (C) 2025 Marcos Bujosa
# Licencia: GNU General Public License v3.0 o posterior
# Este código es software libre y puede ser redistribuido y/o modificado bajo los términos de la GPL.
# Ver el archivo LICENSE del repositorio para más detalles.
# _____

# Los dos primeros comandos son necesarios para que Gretl guarde los resultados de la práctica en el directorio de trabajo
# al ejecutar lo siguiente desde un terminal (use los nombres y ruta que correspondan)
#
# DIRECTORIO="Nombre_Directorio_trabajo" gretlcli -b ruta/nombre_fichero_de_la_practica.inp
#
# Si esto no le funciona en su sistema, comente las siguientes dos líneas y sítuese en el directorio de trabajo de gretl
# que corresponda (configure dicho directorio de trabajo desde la ventana principal de Gretl).

string directory = getenv("DIRECTORIO")
set workdir "@directory"

function series SimuladorAR(matrix phi)
    # SimuladorAR(phi) simula un proceso AR(p),
    # donde phi es el polinomio AR y p es su grado.
    scalar p = cols(phi)
    series U = normal(0,1)
    series Y = 0
    setinfo Y --description="Serie simulada"
    loop i = (p+1)..$nobs
        scalar comb_pasado_Yt = 0
        scalar perturbacion = U[i]
        loop j = 2..p
            comb_pasado_Yt += -phi[1,j] * Y[i-j+1]  # expresión abreviada
        endloop
        Y[i] = comb_pasado_Yt + perturbacion
    endloop
    return Y
end function

# establecemos la muestra
nulldata 1500
setobs 12 1900:01 --time-series

scalar n = 0
loop for (r=0; r<=1.1; r+=0.2)
    n += 1
    scalar theta1 = r
    nombre = sprintf("PhiPositivo-%d(%1.1f)", n, theta1)
    sname = sprintf("Caso_%d", n)
    series @sname = SimuladorAR( {1, -theta1} )
    corrgm @sname 12 --plot=@nombre.png
endloop

scalar n = 0
loop for (r=-1; r<=0.1; r+=0.2)
    n += 1
    scalar theta1 = r
    nombre = sprintf("PhiNegativo-%d(%1.1f)", n, theta1)
    sname = sprintf("Caso_%d", n)
    series @sname = SimuladorAR( {1, -theta1} )
    corrgm @sname 12 --plot=@nombre.png
endloop
```