Trabajo No 2 de Estadística III - 3009137
Análisis de Serie de Tiempo: Modelos de regresión con errores estructurales ARMA(p,q) estacionarios

1. Características del Trabajo

Este trabajo es una continuación del Trabajo No 1. Consiste en ajustar con validación cruzada uno de los modelos de regresión global para la serie, escogido entre los dos que fueron propuestos en el trabajo 1 (ver en la Tabla 1 cuál de los dos modelos deben considerar), luego, validar supuesto de ruido blanco para los errores estructurales y determinar en caso de no ser ruido blanco, si estos errores definen un proceso estacionario de media cero e identificar posibles modelos. De acuerdo a la asignación en Tabla 1, cada grupo deberá considerar cuatro modelos de regresión con errores estructurales ARMA (recuerde que el término ARMA también incluye a procesos AR y MA como casos particulares), manteniendo misma estructura de regresión en la tendencia y estacionalidad, con el fin de ajustarlos, validar sus supuestos y compararlos en ajuste y pronóstico. Finalmente, el mejor modelo de regresión con errores ARMA deberá ser comparado con el mejor de los dos modelos de ajuste local del trabajo1, al cual también deberá evaluar supuestos sobre sus errores de ajuste.

2. Puntos a Desarrollar

La presentación de la solución de los puntos a desarrollar y que se enuncian a continuación, se hará en modalidad de exposición oral del 18 al 22 de mayo de 2021. Sin embargo, independiente de la fecha y hora específica en la que cada grupo hará su exposición, todos los grupos deberán subir en la plataforma moodle la presentación en archivo pdf, entre las 00:00 horas del 14 de mayo hasta las 8:00 horas del 14 de mayo. Antes de la semana de exposiciones se les informará una agenda de horarios en los que se puede convenir las exposiciones durante la semana establecida, incluyendo horarios de asesorías. El orden de temas a presentar en la exposición es igual al de los puntos que se piden desarrollar. Cada grupo contará con un máximo de 45 minutos para su presentación. ***Recuerde que aun cuando el documento debe tener formato presentación, debe estar completo en contenido: Resultados gráficos, numeríos, ecuaciones, pruebas de hipótesis y conclusiones con cada resultado presentado en las diferentes secciones.***

1. Introducción: Deberá presentar nuevamente la definición DANE del índice asignado (ecuación debidamente explicada), dando además un ejemplo interpretativo de sus valores; también, la definición del dominio o clasificación industrial al que está relacionada la serie. Luego, deberá relatar brevemente los resultados alcanzados en el trabajo 1 (incorporando las correcciones realizadas a su trabajo 1): Cuáles modelos globales y locales fueron propuestos (debe dar las ecuaciones teóricas), cuál es el mejor modelo global, qué logró explicar este modelo en relación a los patrones que fueron observados sobre la serie, y si resultó mejor el ajuste y pronóstico global vs. lo local (cuál fue el mejor local entre Holt-Winters y combinación del filtro de la descomposición con loess) y por qué.

2. Análisis descriptivo de la serie, modelo global asignado y sus resultados:

    a) Presente y analice brevemente la gráfica de la serie (y su logaritmo natural si la serie es multiplicativa) indicando los patrones observables de tendencia, estacionalidad, varianza, ciclos. Grafique y analice además la ACF de la serie (para el caso multiplicativo sólo presente y analice la ACF del logaritmo natural) y concluya en términos de estacionariedad o no y por qué, contrastando con lo que a partir de la gráfica de la serie se concluye al respecto.

b) Para el modelo de regresión global señalado en la Tabla 1, reporte la ecuación teórica con sus supuestos y con la estrategia de validación cruzada usando la misma longitud $n = 216$ de ajuste del trabajo anterior, ajuste nuevamente este modelo y reporte los resultados de ajuste (solo la tabla de parámetros estimados, medidas de ajuste, gráfico del ajuste) y pronósticos (solo la tabla de pronósticos, medidas de cobertura, amplitud media de los I.P y medidas MAE, MAPE y RMSE). Dé una conclusión breve sobre la calidad del ajuste y de los pronósticos con este modelo. Use $\exp(C_n^*(p))$ para el cálculo de AIC y BIC.

3. Evaluación de supuesto de ruido blanco e identificación de procesos estocásticos sobre los errores estructurales del modelo global

a) Validación de supuestos: Guarde los residuos estructurales $\hat{E}_t$ en la escala en que ajustó la serie. Analice inicialmente las gráficas de estos residuales en términos de los supuestos sobre los errores E_t de media constante en cero, varianza constante y determine si hay ciclos evidentes no explicados o rachas en signos $\pm$, y qué concluye frente a la existencia de estos patrones. Realice las Pruebas de incorrelación con: Ljung-Box, Durbin-Watson, y gráficas de la ACF y PACF con bandas de Bartlett. Concluya sobre si los errores estructurales E_t son ruido blanco o no y en este último caso, evalúe si por lo menos son estacionarios con media cero. Use $m = 36$ en ACF, PACF y $m = 6, 12, 18, 24, 30$ y 36 en Ljung-Box.

b) Identificación de posibles modelos para los errores estructurales E_t: En caso de rechazar ruido blanco por cualquiera de los análisis anteriores, determine cuáles modelos ARMA provisionales (recuerde de nuevo que procesos AR y MA son casos particulares de los ARMA), podrían ser los adecuados para el error estructural, con base en **acf(), pacf(), auto.arima(), eacf(), SelectModel(), armasubsets(), para los métodos automáticos y la EACF debe presentar un print-screen de la consola R donde sea visible la línea de programa R ejecutada junto con la salida que aparece en la consola R**. En cada caso debe dar la ecuación teórica del modelo ARMA identificado para los errores estructurales y presentar la salida R original que arroje la función de identificación y gráficas dónde aplique.

- la ACF y la PACF.
- La EACF. Realizar la gráfica con orden máximo p, q de 36.
- `SelectModel()` de la librería FitAR, úsela sólo en caso de identificar un patrón tipo cola en ACF. Tenga en cuenta que esta función sólo identifica procesos AR(p). Use un p máximo de 36. Use la función variando el argumento `criterion=" AIC"` y con `criterion="BIC"`. Con cada criterio puede estar obteniendo modelos AR(p) distintos.
- `auto.arima()` de la librera forecast. Use esta función sobre el vector de residuales `residuals(modelo)`, y sobre el objeto de serie de tiempo creado con los residuales, con la función `ts()`, usando la misma frecuencia y fecha de inicio de la serie asignada. En cada caso, varíe también el argumento `ic=" aic"` y `ic="bic"`. De esta forma es posible obtener máximo cuatro modelos distintos.
- La función `armasubsets()` de la librería `TSA`, usando criterio BIC y argumento `ar.method="ml"`, sobre residuos estructurales, tomando el modelo en el renglón que se indica en la Tabla 1. Además, si en la Tabla 1 se indica agregar algún parámetro, complete la ecuación del modelo ingresando los términos correspondientes. Tenga en cuenga que en los modelos identificados con esta función algunos coeficientes deben ser fijados en cero. Verifique que la gráfica que obtiene con su programación R es igual a la que se exhibe en el Apéndice C de esta guía (identifique la respectiva gráfica en el Apéndice C con el nombre de la serie: Datos1, Datos2, etc.)

NOTA 1: Verifique que en todos los modelos que identifique la función auto.arima sean ARIMA(p, d, q) con $d = 0$ ó ARIMA$(p, d, q) \times (P, D, Q)_{[12]}$ con $d = 0$ y $D = 0$, lo cual indica un proceso estacionario. Por otra parte, como es posible que con la función auto.arima() se identifiquen procesos ARMA estacionales estacionarios o $\text{ARMA}(p, q) \times (P, Q)_{[12]}$, estos también deben ser considerados.

NOTA 2: Con métodos de identificación diferentes a ACF-PACF, recuerde que no se deben considerar como candidatos modelos que no son consistentes con lo que ya se ha identificado con ACF-PACF; por ejemplo, si es muy claro que la ACF es cola, entonces no son admisibles procesos MA(q), ni MA(Q)[12], y si además, la ACF es tipo cola exponencial-sinusoidal, sólo son admisibles procesos AR(p) y ARMA(p,q), con $p \geq 2$, y procesos ARMA(p,q)(P,Q)[12] tales que $(p + P \times 12) \geq 2$.

4. Modelos de regresión global con errores estructurales E_t ARMA: Considerando misma estructura de regresión del modelo global, pero errores estructurales según los modelos que se indican en la Tabla 1, dé las ecuaciones teóricas con sus supuestos y ajústelos con validación cruzada, usando de nuevo $n = 216$. Es posible que algunos o todos los modelos propuestos en la Tabla 1 para el error estructural no coincidan con los identificados en 3-b), sino que son una modificación obtenida tras probar con modelos preliminares y chequear que no hay validez de supuestos, por lo que resultó necesario hacer cambios en algunos de ellos. Tenga en cuenta además lo siguiente:

   a) En modelos de regresión lineal múltiple (sobre Y_t ó $\log(Y_t)$, según sea el caso) con errores estructurales ARMA, debe ajustar juntos tendencia, estacionalidad y errores E_t ARMA, usando la función `Arima()` de la librería `forecast`; para ello, en esta función se debe usar el argumento `xreg` igual a una matriz X cuyas columnas son los valores de las variables predictoras en el modelo global, matriz que debe crearse previamente; también en la función `Arima()` se deben definir los órdenes del ARMA para el error E_t, usando los argumentos correspondientes en esta función.

   b) Recuerde además que si ajusta sobre $\log(Y_t)$, debe calcular $\widehat{Y}_t = \exp\left(\widehat{\log(Y_t)}\right) \times \exp(\hat{\sigma}_a^2/2)$ y reportar el valor del factor de corrección por transformación lognormal, donde $\hat{\sigma}_a^2$ es la varianza estimada del error de ajuste a_t del modelo.

   c) Para la programación R estudie los ejemplos presentados en el Apéndice B de esta guía, Secciones I, II y III para modelos con ecuación de regresión lineal en los parámetros de tendencia y estacionalidad (modelos polinomiales estacionales y modelos log polinomiales estacionales, con error estructural ARMA) y Sección IV para modelos de regresión con estructura no lineal en los parámetros de tendencia y estacionalidad (modelos exponenciales polinomial estacional con error estructural ARMA).

   NOTA 3: Para todos los modelos presente las tablas de parámetros estimados, las gráficas de ajustes, tabla de medidas de ajuste, pero la ecuación ajustada solo para el modelo 3. Use $\exp(C_n^*(p))$ para AIC y BIC en todos los modelos ***(quienes tienen asignado modelos exponenciales con error ARMA, seguir expresamente lo que se ilustra en la Sección IV del Apéndice B de esta guía, para el cálculo de AIC y BIC).*** También se deben formular y realizar los tests de significancia de parámetros de interés en la tendencia y en la estacionalidad y de los parámetros en el modelo del error estructural, además, en modelos de regresión usando indicadoras en la componente estacional, debe interpretar y comparar las estimaciones de los parámetros estacionales como se hizo en el trabajo 1 del curso (consulte en asesoría cómo realizar las gráficas pertinentes).

5. Análisis de residuales y validación de supuestos: Realice sobre los residuos de ajuste $\widehat{a}_t$ en los modelos de regresión con errores E_t ARMA, los análisis de las respectivas gráficas, de manera comparativa, como también indicar si hay o no evidencia fuerte de carencia de ajuste, e indique cuáles modelos de los ajustados presentan mejor comportamiento en sus residuales de ajuste $\widehat{a}_t$. Recuerde que con estos residuos se desea evaluar para los errores de ajuste a_t si ¿media es cero, varianza constante y ausencia de patrones contrarios a la independencia? Haga la validación de supuestos sobre los errores de ajuste a_t: Tests ACF, PACF, Ljung-Box (con el mismo m usado en el punto 3-a) de esta guía), Test de normalidad y gráfico de normalidad (estos dos últimos sólo si no se rechaza supuesto de ruido blanco sobre el error de ajuste a_t). Concluya acerca de la validez del supuesto de que los

errores de ajuste a_t definen un proceso de ruido blanco con distribución normal y cómo afecta esto sobre las estimaciones y pronósticos.

6. Pronósticos para la validación cruzada: Para todos los modelos asignados presente, interprete, analice y compare los pronósticos puntuales y por intervalos del 95 %, y la gráfica comparativa de los valores reales y pronósticos puntuales, pero la ecuación de los pronósticos puntuales presente solo la del modelo 3. Tenga en cuenta que para modelos ajustados sobre logaritmo de la serie también es necesario aplicar el factor de corrección por transformación lognormal al traer valores pronosticados a la escala original, es decir, $\widehat{Y}_{216}(L) = \exp\left(\widehat{\log(Y_{216}(L))}\right) \times \exp(\hat{\sigma}_a^2/2)$ y de la misma forma proceder con los intervalos de predicción.

7. Comparación de modelos de regresión con errores ARMA vs. mejor local y conclusiones del trabajo: Para el mejor modelo local entre los dos considerados en el trabajo 1, deberá presentar:

   - Gráfica de su ajuste, y valores de los criterios de información AIC y BIC
   - Gráficos de sus residuales,
   - Medidas de precisión de los pronósticos puntuales (MAE, MAPE, RMSE) y por intervalos del 95 % de confianza (amplitud media y cobertura de los IP, si los hay)
   - Evaluación del supuesto de ruido blanco (tests ACF y PACF) para sus errores de ajuste E_t y test de normalidad (si no rechaza en tests ACF y PACF).
   - Elabore conclusiones: En esta sección debe presentar
     - Un resumen de los resultados encontrados en el respectivo trabajo,
     - Enunciar los problemas enfrentados en la modelación,
     - Postular cuál ha sido el mejor modelo en ajuste y pronóstico entre los tratados y comentar acerca de lo que usted crea que logró este mejor modelo: ¿capturó la dinámica de la serie? ¿Su tendencia, estacionalidad y sus variaciones cíclicas son bien ajustadas?, ¿Los pronósticos parecen realistas y confiables?, ¿Qué otras alternativas podrían haberse propuesto?,
     - ¿Críticas al mejor modelo que encontró en el trabajo actual? Exprese claramente qué recomienda para la serie en cuanto a ajustes globales con errores estructurales ARMA o locales según lo realizado hasta el momento, considerando sobre todo qué tan bien representa cada modelo los patrones observados en la serie.

3. Programación R

Ver el siguiente material para la programación R necesaria.

- Taller 7 de monitoría sobre validación de supuestos e identificación de procesos estacionarios, el Apéndice A de esta guía "Uso de métodos de Identificación automática de modelos ARMA en los errores estructurales de modelos de regresión en R".

- Para el ajuste, pronósticos y validación supuestos en modelos de regresión lineal sobre Y_t o sobre $\log(Y_t)$, con errores estructurales ARMA, remitirse a los ejemplos en el Apéndice B de esta guía, Secciones I a III y la Sección IV solo para quienes tienen modelos exponenciales polinomial estacional con errores estructurales ARMA. Recuerde que si está modelando a $\log(Y_t)$, deberá luego obtener ajustes y pronósticos en la escala original.

4. Los Grupos para el Trabajo y Asignaciones

De acuerdo a los grupos y series definidas en el primer trabajo. El modelo de regresión global con error estructural RB y con errores ARMA que deberán trabajar, son los siguientes:

Tabla 1: Modelo global y modelos con error estructural ARMA

				Modelos de regresión con error estructural ARMA			
				modelo 1	modelo 2	modelo 3	modelo 4
serie	modelo global	Estacionalidad	polinomio	error	error	error	según armasubsets(1)
Datos1	polinomial estacional	Indicadoras de enero a noviembre	3	AR(14)	ARMA(2,3)	ARMA(3,0)(1,0)[12]	tablero 18x18 renglón 1 e incluir a ϕ_5
Datos3	log polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,3,4,5,6$	5	AR(13)	ARMA(6,6)**	ARMA(4,0)(2,0)[12]**	tablero 18x18 renglón 2
Datos4	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,3,4,5;$ F_6 =0.35	3	AR(12)	ARMA(4,4)	ARMA(1,2)(2,0)[12]	tablero 18x18 renglón 5 e incluir ϕ_2 y θ_1
Datos6	log polinomial estacional	Indicadoras de enero a noviembre	3	AR(15)	ARMA(7,7)	ARMA(5,2)(2,0)[12]	tablero 18x18 renglón 3 e incluir θ_2 y θ_3
Datos9+	exponencial polinomial estacional	Indicadoras de enero a noviembre	5	AR(13)	ARMA(5,12)	ARMA(3,0)(0,1)[12]	tablero 12x12 renglón 5
Datos10	polinomial estacional	Indicadoras de enero a noviembre	4	AR(13)	ARMA(3,12)	ARMA(3,0)(0,1)[12]	tablero 18x18 renglón 4
Datos11	log polinomial estacional	Indicadoras de enero a noviembre	3	AR(12)	ARMA(4,2)	ARMA(1,1)(1,0)[12]	tablero 12x12 renglón 5 e incluir ϕ_3, ϕ_4 y ϕ_5
Datos12+	exponencial polinomial estacional	Indicadoras de enero a noviembre	3	AR(12)	ARMA(8,7)	ARMA(3,1)(0,2)[12]	tablero 12x12 renglón 3 e incluir a ϕ_3, ϕ_4 y ϕ_5
Datos13	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,5$	4	AR(14)	ARMA(3,12)	ARMA(1,6)(1,0)[12]	tablero 12x12 renglón 1 e incluir ϕ_2, ϕ_3, y θ_{12}
Datos14	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,5,6;$ F_7 =0.35	5	AR(3)	ARMA(4,6)	ARMA(1,3)(1,0)[12]	tablero 12x12 renglón 5 e incluir a ϕ_2
Datos15	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,3,4,5$	4	AR(13)	ARMA(3,12)	ARMA(3,6)(1,0)[12]	tablero 12x12 renglón 1
Datos16	polinomial estacional	$Fj = j/12$, $j = 1,4,5$	3	AR(13)	ARMA(2,13)	ARMA(1,1)(1,2)[12]	tablero 18x18 renglón 2 e incluir ϕ_2, ϕ_5 y θ_{20}
Datos17	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,4,5$	6	AR(13)**	ARMA(3,13)**	ARMA(1,1)(1,2)[12]**	tablero 18x18 renglón 6 **
Datos18	log polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,3,4,5$ y 6	3	AR(13)	ARMA(3,15)	ARMA(3,2)(2,1)[12]	tablero 24x24 renglón 3
Datos19	polinomial estacional	Indicadoras de enero a noviembre	4	AR(18)	ARMA(3,15)	ARMA(3,2)(0,2)[12]	tablero 12x12 renglón 5 e incluir ϕ_{13}
Datos20	polinomial estacional	Indicadoras de enero a noviembre	2	AR(16)	ARMA(4,7)	ARMA(3,6)(1,1)[12]	tablero 24x24 renglón 1 e incluir a ϕ_1 y ϕ_2
Datos21	polinomial estacional	Indicadoras de enero a noviembre	4	AR(24)	ARMA(4,3)	ARMA(3,2)(2,1)[12]	tablero 18x18 renglón 4 e incluir a ϕ_2, ϕ_4, ϕ_8 y ϕ_{19}
Datos22	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,3,4$	2	AR(20)	ARMA(6,8)	ARMA(4,6)(1,1)[12]	tablero 18x18 renglón 3 e incluir ϕ_1, θ_8 y θ_{16} ***
Datos23	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,3,4$	4	AR(25)	ARMA(10,11)	ARMA(6,6)(1,2)[12]	tablero 24x24 renglón 3 e incluir ϕ_{15} y θ_8
Datos25	polinomial estacional	Trigon. $F_j = j/12$, $j = 1,2,5,6$	6	AR(8)**	ARMA(3,7)**	ARMA(3,2)(0,1)[12]**	tablero 12x12 renglón 1 e incluir a ϕ_3 **

Tabla 1 (Cont.) Modelo global y modelos con error estructural ARMA

				Modelos de regresión con error estructural ARMA			
				modelo 1	modelo 2	modelo 3	modelo 4
serie	modelo global	Estacionalidad	polinomio	error	error	error	según armasubsets$^{(1)}$
Datos28	log polinomial estacional	Indicadoras de enero a noviembre	4	AR(22)	ARMA(6,6)	ARMA(5,3)(2,0)[12]	tablero 24x24 renglón 2
Datos29	polinomial estacional	Indicadoras de enero a noviembre	4	AR(22)	ARMA(7,7)	ARMA(5,5)(1,0)[12]	tablero 18x18 renglón 1
Datos31	polinomial estacional	Trigon. $F_j = j/12$, $j = 1, 2, 3, 4, 5$; F_6 =0.35 y $F_7 = 113/240$	1	AR(15)	ARMA(7,7)	ARMA(2,2)(0,1)[12]	tablero 18x18 renglón 1 e incluir a ϕ_3
Datos32	log polinomial estacional	Indicadoras de enero a noviembre	5	AR(22)	ARMA(7,4)**	ARMA(3,1)(1,0)[12]	tablero 24x24 renglón 1 e incluir a θ_1

Observaciones:

(1) En función armasubsets debe usar argumento ar.method="ml", y cada grupo deberá comparar gráfica obtenida con la que se muestra en el Apéndice C de esta guía. Para tableros 12x12 se fijan argumentos nar=12 y nma=12; para tableros 18x18 se fijan argumentos nar=18 y nma=18 y para tableros 24x24 se fijan argumentos nar=24 y nma=24.

** En estos modelos, ajuste de manera aproximada fijando estimaciones de la parte estructural iguales a las del modelo global. Consultar en Asesoría cómo hacerlo.

+ Como la estructura de regresión no es lineal, todos los modelos para estas series se deben ajustar como explica el Apéndice B, Sección IV de esta guía.

$* * *$ En este modelo usar argumento optim.method="Nelder-Mead" en la función Arima.

Referencias

[1] Bowerman, B. L, O'Connell, R. T y Koehler, A. B. (2009) *Pronósticos, Series de Tiempo y Regresión. Un Enfoque Aplicado. 4 ed.* CENGAGE Learning

[2] Chatfield, C. (2019) *The Analysis of Time Series. An Introduction with R, Seventh edition*. CRC Press-USA.

[3] Diebold, F. (2001) *Elementos de Pronósticos*. International Thomson Editores, México.

[4] Cryer, J. D. and Chan, K-S. (2008) *Time Series Analysis With Applications in R*. Springer.

[5] González, N. G. (2013) *Notas de Clase Estadística III 3009137.* Escuela de Estadística, Universidad Nacional de Colombia Sede Medellín.

[6] Shumway, R. H. and Stoffer, D. S. (2017) *Time Series Analysis and Its Applications. With R Examples*. Fourth ed. Springer

APÉNDICE A. Uso de métodos de Identificación automática de modelos ARMA en los errores estructurales de modelos de regresión en R

Considere la serie mensual de muertes en accidente de tránsito en Ontario, enero 1960 a diciembre 1974 (Fuente: Ledolter, 1983) que se ilustra a continuación junto con su logaritmo natural. Para el logaritmo natural de la serie, $\log(Y_t)$, se ajustó un modelo de regresión lineal (MODELO 1) de tendencia cuadrática y estacionalidad con funciones trigonométricas en las frecuencias $F_j=j/12$, j=1,2,…,6, usando los primeros n=168 datos (de enero 1960 a diciembre 1973).

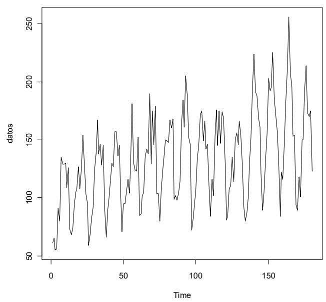


Fig. A1. Serie en escala original

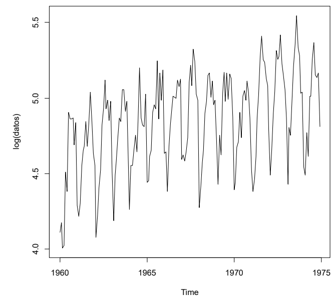


Fig. A2. Serie en escala log. natural

Modelo 1:

$\log(Y_t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \sum_{j=1}^{5}\left[\alpha_j \sin(\pi j t/6) + \gamma_j \cos(\pi j t/6)\right] + \gamma_6 \cos(\pi t) + E_t, \quad \{E_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma^2)$

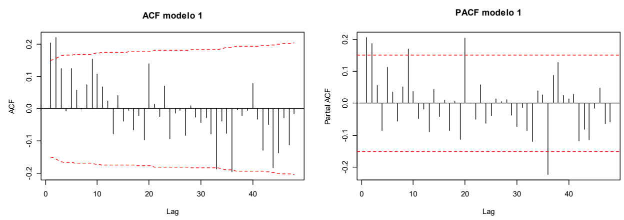


Fig. A3. ACF, PACF construidos con residuos del modelo 1

```
library(forecast);library(TSA);library(FitAR)
datos=scan(file.choose()) #leer abraham5.dat
datos=ts(datos,freq=12,start=c(1960,1))
n=length(datos)-12; t=1:n
yt=ts(datos[t],freq=12,start=c(1960,1))
sen1=sin(pi*t/6);cos1=cos(pi*t/6);sen2=sin(pi*t/3)
cos2=cos(pi*t/3);sen3=sin(pi*t/2);cos3=cos(pi*t/2)
sen4=sin(2*pi*t/3);cos4=cos(2*pi*t/3);sen5=sin(5*pi*t/6)
cos5=cos(5*pi*t/6);cos6=cos(pi*t)
mod1=lm(log(yt)~t+I(t^2)+sen1+cos1+sen2+cos2+sen3+cos3+sen4+cos4+sen5+cos5+cos6)
summary(mod1)
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.540e+00  3.011e-02 150.772  < 2e-16 ***
t            4.789e-03  8.222e-04   5.825 3.22e-08 ***
I(t^2)      -1.200e-05  4.712e-06  -2.548 0.011826 *
sen1        -3.160e-01  1.404e-02 -22.512  < 2e-16 ***
cos1        -7.059e-02  1.402e-02  -5.036 1.32e-06 ***
sen2        -5.587e-02  1.402e-02  -3.985 0.000104 ***
cos2         5.810e-02  1.402e-02   4.145 5.60e-05 ***
sen3        -2.431e-02  1.402e-02  -1.734 0.084887 .
cos3         4.632e-02  1.402e-02   3.305 0.001183 **
sen4        -1.081e-02  1.402e-02  -0.772 0.441535
cos4         2.229e-02  1.402e-02   1.590 0.113914
sen5         2.766e-02  1.402e-02   1.974 0.050222 .
cos5         3.783e-02  1.402e-02   2.699 0.007740 **
cos6         1.599e-02  9.911e-03   1.613 0.108822
```

- Print-Screen de la identificación con auto.arima sobre vector de residuales sin fechas, usando criterios AIC y BIC

```
> auto.arima(residuals(mod1),ic="aic")
Series: residuals(mod1)
ARIMA(4,0,1) with zero mean

Coefficients:
          ar1     ar2     ar3      ar4     ma1
      -0.2463  0.2611  0.1421  -0.0809  0.4128
s.e.   0.3998  0.0978  0.1026   0.0888  0.3958

sigma^2 estimated as 0.01375:  log likelihood=121.6
AIC=-231.19   AICc=-230.67   BIC=-212.45
> auto.arima(residuals(mod1),ic="bic")
Series: residuals(mod1)
ARIMA(2,0,0) with zero mean

Coefficients:
         ar1     ar2
      0.1667  0.1876
s.e.  0.0757  0.0758

sigma^2 estimated as 0.01397:  log likelihood=120.31
AIC=-234.62   AICc=-234.48   BIC=-225.25
>
```

Se identifican dos modelos:
Con AIC: E_t un ARIMA(4,0,1)=ARMA(4,1) estacionario y de media cero, entonces

$$E_t = \sum_{j=1}^{4} \phi_j E_{t-j} + a_t + \theta_1 a_{t-1}\,, \text{con } \{a_t\}_{t\in\mathbb{Z}^+} \text{ un } RB \sim N(0,\sigma_a^2)$$

Con BIC: E_t un ARIMA(2,0,0)=AR(2) estacionario de media cero, entonces

$$E_t = \phi_1 E_{t-1} + \phi_2 E_{t-2} + a_t, \text{con } \{a_t\}_{t\in\mathbb{Z}^+} \text{ un } RB \sim N(0,\sigma_a^2)$$

Los coeficientes en la salida R denominados ar1, ar2, ar3 y ar4 hacen referencia a ϕ_1, ϕ_2, ϕ_3 y ϕ_4 respectivamente, mientras que el coeficiente ma1 hace referencia a θ_1. La función da además los valores estimados de los parámetros y sus errores estándar, pero estos no son los que tomaremos en la estimación del modelo global con errores ARMA.

- Print-Screen de la identificación con auto.arima sobre vector de residuales con fechas, usando criterios AIC y BIC

```
> SerieEt=ts(residuals(mod1),freq=12,start=c(1960,1))
>
> auto.arima(SerieEt,ic="aic")
Series: SerieEt
ARIMA(2,0,0)(2,0,0)[12] with zero mean

Coefficients:
         ar1     ar2     sar1     sar2
      0.1776  0.1857  -0.0044  -0.1326
s.e.  0.0761  0.0769   0.0831   0.0838

sigma^2 estimated as 0.01373:  log likelihood=121.55
AIC=-233.1   AICc=-232.73   BIC=-217.48
> auto.arima(SerieEt,ic="bic")
Series: SerieEt
ARIMA(2,0,0)(2,0,0)[12] with zero mean

Coefficients:
         ar1     ar2     sar1     sar2
      0.1776  0.1857  -0.0044  -0.1326
s.e.  0.0761  0.0769   0.0831   0.0838

sigma^2 estimated as 0.01373:  log likelihood=121.55
AIC=-233.1   AICc=-232.73   BIC=-217.48
>
```

Aquí se identifican modelos ARMA estacionales estacionarios, así
Por AIC: E_t es un ARIMA(2,0,0)(2,0,0)[12]=AR(2)×AR(2)[12] de media cero y estacionario
Por BIC: Identifica el mismo modelo dado con AIC.

NOTA: E_t es un ARMA(p,q)(P,Q)[s] corresponde a un proceso ARMA donde hay la interacción de dos estructuras de dependencia:

a) Una estructura ARMA(p,q) que explica la dependencia entre los valores sucesivos de E_t, es decir, entre $\dots, E_{t-3}, E_{t-2}, E_{t-1}, E_t, E_{t+1}, E_{t+2}, E_{t+3}, \dots$. Esta estructura tiene un polinomio AR(p) dado por $\phi_p(B) = 1 - \sum_{j=1}^{p} \phi_j B^j$ y un polinomio MA(q) dado por $\theta_q(B) = 1 + \sum_{i=1}^{q} \theta_i B^i$

b) Una estructura ARMA(P,Q)[s] que explica la dependencia entre valores de E_t separados s períodos en el tiempo, es decir, entre $\dots, E_{t-3s}, E_{t-2s}, E_{t-s}, E_t, E_{t+s}, E_{t+2s}, E_{t+3s}, \dots$. Esta estructura tiene un polinomio AR(P) de período s dado por $\Phi_P(B^s) = 1 - \sum_{k=1}^{P} \Phi_k B^{ks}$ y un polinomio MA(Q) de período s dado por $\Theta_Q(B^s) = 1 + \sum_{l=1}^{Q} \Theta_l B^{ls}$

c) El modelo general tiene la siguiente ecuación $\phi_p(B)\Phi_P(B^s)E_t = \theta_q(B)\Theta_Q(B^s)a_t$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma_a^2)$

Luego, para el modelo identificado se tiene, $\phi_2(B) = 1 - \phi_1 B - \phi_2 B^2$, $\Phi_2(B^{12}) = 1 - \Phi_1 B^{12} - \Phi_2 B^{24}$; no hay polinomios MA y por tanto, $\phi_2(B)\Phi_2(B^{12})E_t = a_t$ de donde

$$(1 - \phi_1 B - \phi_2 B^2)(1 - \Phi_1 B^{12} - \Phi_2 B^{24})E_t = a_t$$
$$\Rightarrow (1 - \phi_1 B - \phi_2 B^2 - \Phi_1 B^{12} + \phi_1\Phi_1 B^{13} + \phi_2\Phi_1 B^{14} - \Phi_2 B^{24} + \phi_1\Phi_2 B^{25} + \phi_2\Phi_2 B^{26})E_t = a_t$$

Despejando E_t, obtenemos,

$$E_t = \phi_1 E_{t-1} + \phi_2 E_{t-2} + \Phi_1 E_{t-12} - \phi_1\Phi_1 E_{t-13} - \phi_2\Phi_1 E_{t-14} + \Phi_2 E_{t-24} - \phi_1\Phi_2 E_{t-25} - \phi_2\Phi_2 E_{t-26} + a_t,$$
$$\text{con } \{a_t\}_{t\in\mathbb{Z}^+} \text{ un } RB \sim N(0,\sigma_a^2)$$

Los coeficientes en la salida R denominados ar1, ar2, hacen referencia a ϕ_1 y ϕ_2 de la estructura ARMA(2,0) mientras que los coeficientes denominados sar1 y sar2 corresponden a Φ_1 y Φ_2 de la estructura ARMA(2,0)[12]. Si hubiera parámetros θ_i aparecerían con nombre mai mientras que parámetros Θ_k aparecerían con el nombre smak, siendo las letras i y k los respectivos subíndices.

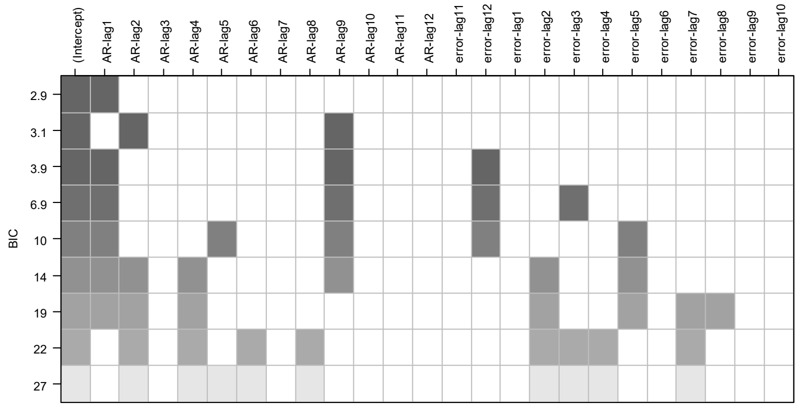
- Print-Screen de la identificación de modelos AR(p) con SelectModel sobre vector de residuales, usando criterios AIC y BIC y orden p máximo de 36.

```
> SelectModel(residuals(mod1), lag.max = 36, Criterion="AIC",ARModel = "AR")
  p AIC-Exact AIC-Approx
1 2 -711.3872  -5.882468
2 3 -709.9035  -5.115701
3 4 -709.1081  -5.333515
> SelectModel(residuals(mod1), lag.max = 36, Criterion="BIC",ARModel = "AR")
  p BIC-Exact BIC-Approx
1 2 -702.0153   6.613388
2 1 -701.1310   2.014936
3 0 -699.0217   2.971439
>
```

Se identifican los siguientes modelos para E_t:
Por AIC: el mejor modelo es un AR(2): $E_t = \phi_1 E_{t-1} + \phi_2 E_{t-2} + a_t$, el segundo mejor un AR(3): $E_t = \phi_1 E_{t-1} + \phi_2 E_{t-2} + \phi_3 E_{t-3} + a_t$ y el tercer mejor es un AR(4): $E_t = \phi_1 E_{t-1} + \phi_2 E_{t-2} + \phi_3 E_{t-3} + \phi_4 E_{t-4} + a_t$, y en todos los casos bajo el supuesto de$\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma_a^2)$.
Por BIC: el mejor modelo es un AR(2): $E_t = \phi_1 E_{t-1} + \phi_2 E_{t-2} + a_t$, el segundo mejor es un AR(1): $E_t = \phi_1 E_{t-1} + a_t$ y en ambos casos bajo el supuesto de $\{a_t\}_{t\in\mathbb{Z}^+}\ un\ RB \sim N(0,\sigma_a^2)$. El tercero dice que p=0 pero indicaría que no hay estructura AR(p) (no necesariamente que haya un ruido blanco!!)

- Identificación con armasubsets, fijando un valor máximo para p y q en 12

plot(armasubsets(residuals(mod1),nar=12,nma=12,y.name='AR',ar.method='ml'))



En cada renglón se identifica un posible ARMA, de modo que en el renglón superior está el modelo ARMA de menor BIC y hacia abajo va aumentando el BIC. Esta función internamente ajusta de modo aproximado modelos arma para la serie sobre la que se aplica, a través de una regresión con intercepto, donde las variables explicatorias son rezagos de esa serie con coeficientes de regresión iguales a los ϕ_j y rezagos de un proceso del cual se aproximan los valores de los a_{t-i}, cuyos coeficientes de regresión son los θ_i; por tanto, en los modelos que se muestran en esta gráfica aparece siempre un intercepto, pero sólo interesan los coeficientes denominados AR-lagj, que representan a los ϕ_j y los coeficientes denominados error-lagi que representan a los θ_i, respectivamente. Para cada renglón, las casillas sombreadas indican cuáles coeficientes deben ir en el modelo mientras que las casillas en blanco indican cuáles de los coeficientes no van en el modelo, es decir, valen cero. Entonces para los tres primeros modelos (renglón 1, 2 y 3 de arriba hacia abajo), esta función nos dice que

Renglón 1: $E_t = \phi_1 E_{t-1} + a_t$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0, \sigma_a^2)$, es decir, E_t es un AR(1)

Renglón 2: $E_t = \phi_2 E_{t-2} + \phi_9 E_{t-9} + a_t$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0, \sigma_a^2)$, es decir, E_t es un AR(9) pero sólo ϕ_2 y ϕ_9 van en el modelo.

Renglón 3: $E_t = \phi_1 E_{t-1} + \phi_9 E_{t-9} + a_t + \theta_{12} a_{t-12}$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0, \sigma_a^2)$, es decir, E_t es un ARMA(9,12) pero sólo ϕ_1, ϕ_9 y θ_{12} van en el modelo.

- Print-screen de la identificación con EACF fijando un valor máximo de 36 para los órdenes p y q

```
> eacf(residuals(mod1),ar.max=36,ma.max=36)
AR/MA
   0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
0  x x o o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  x  o  o  x  o
1  x o o o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  x  o  o  x  o
2  x x o o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  x  o  o  x  o
3  x o x o o o o o o o o  o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  x  o
4  x o x o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
5  x x x o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
6  x o x o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
7  x o x o o x o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
8  x x x o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
9  x x o x o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
10 x x o x o o o x x o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
11 x x o x x o o o x o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
12 x x x x o o o o o x o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
13 x x o x o o o o o x o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
14 x x o x o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
15 x x o x o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
16 o x x o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
17 o x x o o o o o o o x  x  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
18 o x x o o o o o o x x  x  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
19 x x x x x x x o o o x  x  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
20 o o x x o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
21 o o o x o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
22 x o o x o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
23 x x x x x o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
24 x x o o o o o o o o o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
25 x o o o o o o o o o o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
26 o o o o o o o o o o o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
27 x o o o o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
28 o o o o o o o o o o o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
29 x o o o o o o o o o o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
30 x x x x x o o o o o o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
31 x x x x o o o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
32 x o x o o x o o o o o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
33 x x x o o o o o o o o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
34 x o x o o o o o o o o  x  o  o  o  o  o  o  o  o  o  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
35 o x x o o o o o o o x  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
36 x x x x x o o o o o x  x  x  x  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
```

¿Qué modelo se identifica?

APÉNDICE B. Sobre ajustes, validación supuestos y pronósticos en R de modelos de regresión con errores ARMA

Los ejemplos en las Secciones I, II y III de este Apéndice, consideran una estructura de regresión lineal en los parámetros de tendencia y estacionalidad, es decir, modelos aditivos polinomiales estacionales y modelos log polinomiales estacionales, con errores estructurales E_t correlacionados según algún modelo estocástico estacionario de media cero. ***Para los casos de modelos de regresión no lineal en los parámetros de tendencia y estacionalidad (modelos exponenciales), vaya a la Sección IV página 18.***

Pasos iniciales: Cargar librerías y definir funciones de usuario necesarias:

```
library(forecast);library(lmtest)
#Creando función usuario crit.inf.resid() para calcular C*n(p)
crit.inf.resid=function(residuales,n.par,AIC="TRUE"){
if(AIC=="TRUE"){
#Calcula AIC
CI=log(mean(residuales^2))+2*n.par/length(residuales)
}
if(AIC=="FALSE"){
#Calcula BIC
CI=log(mean(residuales^2))+n.par*log(length(residuales))/length(residuales)
}
CI
}

#DEFINIENDO FUNCIÓN USUARIO PARA TEST LJUNG-BOX
BP.LB.test=function(serie,maxlag,type="Box"){
aux=floor(maxlag/6); X.squared=c(rep(NA,aux))
df=c(rep(NA,aux)); p.value=c(rep(NA,aux))
for(i in 1:aux){
test=Box.test(serie,lag=(6*i),type=type)
X.squared[i]=test[[1]]; df[i]=test[[2]]
p.value[i]=test[[3]]
}
lag=6*c(1:aux)
teste=as.data.frame(cbind(X.squared,df,p.value))
rownames(teste)=lag; teste
}

#Función para calcular la amplitud de los I.P
amplitud=function(LIP,LSP){
a=LSP-LIP
am=mean(a)
am
}

#Función para calcular la cobertura de los I.P
cobertura=function(real,LIP,LSP){
I=ifelse(real>=LIP & real<=LSP,1,0)
p=mean(I)
p
}
```

Nota: En los siguientes ejemplos, el paso 1) solo debe realizarlo una vez, pues en todos los modelos que ud. ajustará en este segundo trabajo, se mantiene igual la estructura de tendencia y estacionalidad. En los pasos 2) a 7), tenga en cuenta que, por cada modelo distinto a considerar en los errores, debe crear con diferentes nombres los objetos que guarden valores ajustados, residuos, pronósticos, etc.

I. Modelos lineales en los parámetros de tendencia y estacionalidad, y errores ARMA(p,q) con todos los parámetros. Por ejemplo, considere los siguientes dos modelos de regresión lineal

a. $Y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \sum_{i=1}^{11} \delta_i I_{i_t} + E_t$, donde $E_t = \sum_{j=1}^{p} \phi_j E_{t-j} + a_t + \sum_{i=1}^{q} a_{t-i}$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma_a^2)$,

b. $\log(Y_t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \sum_{j=1}^{5}\{\alpha_j \sin(\pi j t/6) + \gamma_j \cos(\pi j t/6)\} + \gamma_6 \cos(\pi t) + E_t$, donde $E_t = \sum_{j=1}^{p} \phi_j E_{t-j} + a_t + \sum_{i=1}^{q} \theta_i a_{t-i}$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma_a^2)$.

Veamos cada caso:

a. Para modelo $Y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \sum_{i=1}^{11} \delta_i I_{i_t} + E_t$, donde $E_t = \sum_{j=1}^{p} \phi_j E_{t-j} + a_t + \sum_{i=1}^{q} \theta_i a_{t-i}$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma_a^2)$, se procede de la siguiente manera:

1) Lea datos y defina variables necesarias para el ajuste y el pronóstico (en este ejemplo se asumen 12 pronósticos ex – post):

```
datos=read.table(file.choose(),....) #leer conforme a la base de datos asignada
datos=ts(datos,freq=12,start=c(1960,1)) #convertir en serie según frecuencia y fecha inicial
n=length(datos)-12 #Defina longitud de los datos para ajuste con validación cruzada. Aquí se recorta en 12 datos
#Defina el índice de tiempo y sus potencias según polinomio planteado para Tt
t=1:n; t2=t^2
yt=ts(datos[t],freq=12,start=c(1960,1)) #serie recortada
```

```
#Defina la matriz de variables indicadoras
Indicadoras=seasonaldummy(yt)

#Defina la matriz de variables predictoras de tendencia y estacionalidad:
#van juntas todas las variables de la tendencia y la estacionalidad en ese orden
X=cbind(t,t2,Indicadoras)

#Definiendo variables para pronósticos
tnuevo=(n+1):length(datos)
t2nuevo=tnuevo^2
Indicadorasnuevo=seasonaldummy(yt,h=12) #Horizonte de pronóstico h igual a longitud de recorte de la serie
Xnuevo=cbind(t=tnuevo,t2=t2nuevo,Indicadoras=Indicadorasnuevo) #matriz predictores en el pronóstico
ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según su caso
```

2) Ajuste el modelo con función Arima de librería forecast

```
#indique el valor correspondiente a p y q en la función Arima, así como la matriz de predictores X en xreg=X
modelo=Arima(yt,order=c(p,0,q),xreg=X,method="ML")
```

3) Construya tabla completa de parámetros estimados, valores P corresponden a $P(|t_{n-k}| > |T_0|)$, con k la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA.

```
#Calcule grados de libertad del MSE del modelo
k=length(coef(modelo)[coef(modelo)!=0]);k #número de parámetros del modelo
dfmodelo=n-k
#Construya tabla de parámetros estimados con estadísticos T0 y valores P para cada parámetro
coeftest(modelo,df=dfmodelo)
```

4) Obtenga gráfica del ajuste de la serie y de los residuales de ajuste $\hat{a}_t$

```
#Gráfico de la serie y su ajuste
ythat=modelo$fitted #Este objeto ya tiene fechas
plot(datos)
lines(ythat,col=2)
legend("topleft",legend=c("datos","ajuste modelo"),lty=1,col=1:2)
#Gráficos de residuales
plot(residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
plot(as.numeric(modelo$fitted),residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
```

5) Cálculo de AIC y BIC, versión $\exp(C_n^*(p))$, con el número de parámetros k siendo la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA, previamente calculado en 3).

```
AICmodelo=exp(crit.inf.resid(residuales=residuals(modelo),n.par=k))
BICmodelo=exp(crit.inf.resid(residuales=residuals(modelo),n.par=k,AIC="FALSE"))
```

6) Valide supuestos sobre el error de ajuste a_t

```
#ACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
acf(as.numeric(residuals(modelo)),ci.type="ma",lag.max=m,main="ACF modelo",ci.col=2)

#PACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
pacf(as.numeric(residuals(modelo)),lag.max=m,main="PACF modelo",ci.col=2)

BP.LB.test(residuals(modelo),maxlag=m,type="Ljung") #test Ljung-Box use máximo m igual al de ACF y PACF

#Normalidad sobre residuales de ajuste en el modelo. Sólo si no se rechaza supuesto de ruido blanco
shapiro.test(residuals(modelo))

win.graph()
qqnorm(residuals(modelo),main="Gráfico de normalidad residuos de ajuste modelo")
qqline(residuals(modelo),col=2)
```

7) Pronósticos para la validación cruzada (en este ejemplo se asume 12 pronósticos ex – post)

```
predmodelo=ts(as.data.frame(forecast(modelo,xreg=Xnuevo,level=95)),freq=12, start=c(1974,1))
predmodelo
ytpronmodelo=predmodelo[,1] #Tomando el pronóstico puntual. Este objeto tiene fechas

#Medidas precisión pronósticos
accuracy(ytpronmodelo,ytf)
amplitud(LIP=predmodelo[,2],LSP=predmodelo[,3])
cobertura(real=ytf,LIP=predmodelo[,2],LSP=predmodelo[,3])
```

b. Modelo $\log(Y_t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \sum_{j=1}^{5}\{\alpha_j \sin(\pi j t/6) + \gamma_j \cos(\pi j t/6)\} + \gamma_6 \cos(\pi t) + E_t$, donde $E_t = \sum_{j=1}^{p} \phi_j E_{t-j} + a_t + \sum_{i=1}^{q} \theta_i a_{t-i}$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0, \sigma_a^2)$

1) Lea datos y defina variables necesarias para el ajuste y el pronóstico (en este ejemplo se asume 12 pronósticos ex – post)

```
datos=read.table(file.choose(),....) #leer conforme a la base de datos asignada
datos=ts(datos,freq=12,start=c(1960,1)) #convertir en serie según frecuencia y fecha inicial
n=length(datos)-12 #Defina longitud de los datos para ajuste con validación cruzada, según su caso
#Defina el índice de tiempo y sus potencias según polinomio
t=1:n; t2=t^2; t3=t^3
yt=ts(datos[t],freq=12,start=c(1960,1)) #serie recortada

#Defina las variables trigonométricas necesarias según modelo postulado
sen1=sin(pi*t/6)
cos1=cos(pi*t/6)
sen2=sin(pi*t/3)
cos2=cos(pi*t/3)
sen3=sin(pi*t/2)
cos3=cos(pi*t/2)
sen4=sin(2*pi*t/3)
cos4=cos(2*pi*t/3)
sen5=sin(5*pi*t/6)
cos5=cos(5*pi*t/6)
cos6=cos(pi*t)
#Defina la matriz de variables predictoras de tendencia y estacionalidad:
#van juntas todas las variables de la tendencia y de la estacionalidad en ese orden
X=cbind(t,t2,t3,sen1,cos1,sen2,cos2,sen3,cos3,sen4,cos4,sen5,cos5,cos6)
#Definiendo variables para pronósticos
tnuevo=(n+1):length(datos)
t2nuevo=tnuevo^2
t3nuevo=tnuevo^3
sen1n=sin(pi*tnuevo/6)
cos1n=cos(pi*tnuevo/6)
sen2n=sin(pi*tnuevo/3)
cos2n=cos(pi*tnuevo/3)
sen3n=sin(pi*tnuevo/2)
cos3n=cos(pi*tnuevo/2)
sen4n=sin(2*pi*tnuevo/3)
cos4n=cos(2*pi*tnuevo/3)
sen5n=sin(5*pi*tnuevo/6)
cos5n=cos(5*pi*tnuevo/6)
cos6n=cos(pi*tnuevo)

Xnuevo=cbind(t=tnuevo,t2=t2nuevo,t3=t3nuevo,sen1=sen1n,cos1=cos1n,sen2=sen2n,cos2=cos2n,sen3=sen3n,cos3=cos3n,
sen4=sen4n,cos4=cos4n,sen5=sen5n,cos5=cos5n,cos6=cos6n) #matriz de predictores en el pronóstico
ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según datos
```

2) Ajuste el modelo con función Arima de librería forecast

```
#indique el valor correspondiente a p y q en la función Arima, así como la matriz de predictores X en xreg=X
modelo=Arima(log(yt),order=c(p,0,q),xreg=X,method="ML")
```

3) Construya tabla completa de parámetros estimados, valores P corresponden a $P(|t_{n-k}| > |T_0|)$, con k la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA.

```
#Calcule grados de libertad del MSE del modelo
k=length(coef(modelo)[coef(modelo)!=0]);k #número de parámetros del modelo
dfmodelo=n-k

#Construya tabla de parámetros estimados con estadísticos T0 y valores P para cada parámetro
coeftest(modelo,df=dfmodelo)
```

4) Obtenga gráfica del ajuste de la serie en escala original y de los residuales de ajuste $\hat{a}_t$. Observe que en el factor de corrección por transformación lognormal, se invoca a modelo$sigma2 el cual es igual a $\hat{\sigma}_a^2$

```
ythat=exp(modelo$fitted)*exp(modelo$sigma2/2) #este objeto ya queda con las fechas de la serie
#Gráfico de la serie y su ajuste
plot(datos)
lines(ythat,col=2)
legend("topleft",legend=c("datos","ajuste modelo"),lty=1,col=1:2)

#Gráficos de residuales
plot(residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
plot(as.numeric(modelo$fitted),residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
```

5) Cálculo de AIC y BIC, versión $\exp(C_n^*(p))$, con el número de parámetros k siendo la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA, previamente calculado en 3).

```
Res.orig=yt-ythat #Cálculo de pseudo residuales

AICmodelo=exp(crit.inf.resid(residuales= Res.orig,n.par=k))
BICmodelo=exp(crit.inf.resid(residuales= Res.orig,n.par=k,AIC="FALSE"))
```

6) Valide supuestos sobre el error de ajuste a_t

```
#ACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
acf(as.numeric(residuals(modelo)),ci.type="ma",lag.max=m,main="ACF modelo",ci.col=2)

#PACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
pacf(as.numeric(residuals(modelo)),lag.max=m,main="PACF modelo",ci.col=2)
BP.LB.test(residuals(modelo),maxlag=m,type="Ljung") #test Ljung-Box use máximo m igual al de ACF y PACF
#Normalidad sobre residuales de ajuste en el modelo. Sólo si no se rechaza supuesto de ruido blanco
shapiro.test(residuals(modelo))

win.graph()
qqnorm(residuals(modelo),main="Gráfico de normalidad residuos de ajuste modelo")
qqline(residuals(modelo),col=2)
```

7) Pronósticos para la validación cruzada. Observe de nuevo que en el factor de corrección por transformación lognormal, se invoca a modelo$sigma2 el cual es igual a $\hat{\sigma}_a^2$ (en este ejemplo se asume 12 pronósticos ex – post)

```
#Cálculo del pronóstico con I.P del 95%, en escala original
predmodelo=exp(as.data.frame(forecast(modelo,xreg=Xnuevo,level=95)))*exp(modelo$sigma2/2)
predmodelo=ts(predmodelo,freq=12,start=c(1974,1)) #Fechas y frecuencia según datos en pronósticos
predmodelo
ytpronmodelo=predmodelo[,1] #Tomando el pronóstico puntual

#Medidas precisión pronósticos
accuracy(ytpronmodelo,ytf)
amplitud(LIP=predmodelo[,2],LSP=predmodelo[,3])
cobertura(real=ytf,LIP=predmodelo[,2],LSP=predmodelo[,3])
```

II. Modelos lineales en los parámetros de tendencia y estacionalidad, y errores ARMA(p,q) con sólo alguno de los parámetros de esta estructura ARMA. Por ejemplo,
$\log(Y_t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \sum_{j=1}^{5}\{\alpha_j \sin(\pi j t/6) + \gamma_j \cos(\pi j t/6)\} + \gamma_6 \cos(\pi t) + E_t$, donde $E_t = \phi_1 E_{t-1} + \phi_9 E_{t-9} + a_t + \theta_{12} a_{t-12}$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0, \sigma_a^2)$. Se procede de la siguiente manera:

1) Lea datos y defina variables necesarias para el ajuste y el pronóstico (en este ejemplo se asume 12 pronósticos ex – post)

```
datos=read.table(file.choose(),....) #leer conforme a la base de datos asignada
datos=ts(datos,freq=12,start=c(1960,1)) #convertir en serie según frecuencia y fecha inicial
n=length(datos)-12 #Defina longitud de los datos para ajuste con validación cruzada, según su caso
#Defina el índice de tiempo y sus potencias según polinomio
t=1:n; t2=t^2; t3=t^3
yt=ts(datos[t],freq=12,start=c(1960,1)) #serie recortada

#Defina las variables trigonométricas necesarias según modelo postulado
sen1=sin(pi*t/6)
cos1=cos(pi*t/6)
sen2=sin(pi*t/3)
cos2=cos(pi*t/3)
sen3=sin(pi*t/2)
cos3=cos(pi*t/2)
sen4=sin(2*pi*t/3)
cos4=cos(2*pi*t/3)
sen5=sin(5*pi*t/6)
cos5=cos(5*pi*t/6)
cos6=cos(pi*t)

#Defina la matriz de variables predictoras de tendencia y estacionalidad:
#van juntas todas las variables de la tendencia y de la estacionalidad en ese orden
X=cbind(t,t2,t3,sen1,cos1,sen2,cos2,sen3,cos3,sen4,cos4,sen5,cos5,cos6)

#Definiendo variables para pronósticos
tnuevo=(n+1):length(datos)
t2nuevo=tnuevo^2
t3nuevo=tnuevo^3
sen1n=sin(pi*tnuevo/6)
```

```
cos1n=cos(pi*tnuevo/6)
sen2n=sin(pi*tnuevo/3)
cos2n=cos(pi*tnuevo/3)
sen3n=sin(pi*tnuevo/2)
cos3n=cos(pi*tnuevo/2)
sen4n=sin(2*pi*tnuevo/3)
cos4n=cos(2*pi*tnuevo/3)
sen5n=sin(5*pi*tnuevo/6)
cos5n=cos(5*pi*tnuevo/6)
cos6n=cos(pi*tnuevo)
Xnuevo=cbind(t=tnuevo,t2=t2nuevo,t3=t3nuevo,sen1=sen1n,cos1=cos1n,sen2=sen2n,cos2=cos2n,sen3=sen3n,cos3=cos3n,
sen4=sen4n,cos4=cos4n,sen5=sen5n,cos5=cos5n,cos6=cos6n) #matriz de predictores en el pronóstico
ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según su caso
```

2) Ajuste el modelo con función Arima de librería forecast. Dado que no se usan todos los parámetros del modelo ARMA(p,q) en los errores estructurales E_t, en la función Arima debe especificarse cuáles parámetros van en el modelo y cuáles se fijan en cero. Para ello, se usa el argumento fixed igual a un vector de longitud (p+q)+(No. Parámetros tendencia)+(No. Parámetros estacionalidad). Las entradas o valores de este vector se ordenan de la siguiente manera: Los primeros p valores son relativos a los coeficientes ϕ_j, los siguientes q valores hacen referencia a los coeficientes θ_i, los siguientes r valores (r= No. Parámetros tendencia+ No. Parámetros estacionalidad) hacen referencia a los parámetros de la tendencia y la componente estacional de acuerdo a la estructura de regresión postulada. En cada entrada del vector se coloca un 0 si el parámetro no va en el modelo y un NA, si el respectivo parámetro debe ir en el modelo. Por tanto, para los últimos r valores del vector se debe asignar un NA, mientras que para los primeros (p+q) valores correspondientes a los coeficientes del ARMA, se coloca 0 ó NA, según la ecuación específica del ARMA. Para el ejemplo, como el error estructural es un ARMA(9,12) pero sólo los coeficientes ϕ_1, ϕ_9 y θ_{12} van en la estructura ARMA, mientras que en la parte de tendencia y estacionalidad se tienen 15 parámetros, el argumento fixed en la función Arima debe ser especificado así (OJO: rep(a,b) repite b veces el argumento a),
fixed=c(NA,rep(0,7),NA,rep(0,11),NA,rep(NA,15))
que es equivalente a escribir de forma extendida lo siguiente,
fixed=c(NA,0,0,0,0,0,0,0,NA,0,0,0,0,0,0,0,0,0,0,0,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA)
donde el color del carácter indica

- Las entradas para los parámetros parte AR(9)
- Las entradas para los parámetros parte MA(12)
- Las entradas para los parámetros de tendencia y estacionalidad

```
#indique el valor correspondiente a p y q en la función Arima, así como la matriz de predictores X en xreg=X y
#el vector para argumento fixed
modelo=Arima(log(yt),order=c(9,0,12),xreg=X,fixed= c(NA,rep(0,7),NA,rep(0,11),NA,rep(NA,15)),method="ML")
```

3) Construya tabla completa de parámetros estimados, valores P corresponden a $P(|t_{n-k}| > |T_0|)$, con k la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA.

```
#Calcule grados de libertad del MSE del modelo
k=length(coef(modelo)[coef(modelo)!=0]);k #número de parámetros del modelo
dfmodelo=n-k
#Construya tabla de parámetros estimados con estadísticos T0 y valores P para cada parámetro
coeftest(modelo,df=dfmodelo)
```

4) Obtenga gráfica del ajuste de la serie en escala original y de los residuales de ajuste $\hat{a}_t$. Observe que en el factor de corrección por transformación lognormal, se invoca a modelo$sigma2 el cual es igual a $\hat{\sigma}_a^2$

```
ythat=exp(modelo$fitted)*exp(modelo$sigma2/2) #este objeto ya queda con las fechas de la serie
#Gráfico de la serie y su ajuste
plot(datos)
lines(ythat,col=2)
legend("topleft",legend=c("datos","ajuste modelo"),lty=1,col=1:2)

#Gráficos de residuales
plot(residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
plot(as.numeric(modelo$fitted),residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
```

5) Cálculo de AIC y BIC, versión $\exp(C_n^*(p))$, con el número de parámetros k siendo la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA, previamente calculado en 3).

```
Res.orig=yt-ythat #Cálculo de pseudo residuales

AICmodelo=exp(crit.inf.resid(residuales=Res.orig,n.par=k))
BICmodelo=exp(crit.inf.resid(residuales=Res.orig,n.par=k,AIC="FALSE"))
```

6) Valide supuestos sobre el error de ajuste a_t

```
#ACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
acf(as.numeric(residuals(modelo)),ci.type="ma",lag.max=m,main="ACF modelo",ci.col=2)

#PACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
pacf(as.numeric(residuals(modelo)),lag.max=m,main="PACF modelo",ci.col=2)

BP.LB.test(residuals(modelo),maxlag=m,type="Ljung") #test Ljung-Box use máximo m igual al de ACF y PACF
#Normalidad sobre residuales de ajuste en el modelo. Sólo si no se rechaza supuesto de ruido blanco
shapiro.test(residuals(modelo))
win.graph()
qqnorm(residuals(modelo),main="Gráfico de normalidad residuos de ajuste modelo")
qqline(residuals(modelo),col=2)
```

7) Pronósticos para la validación cruzada. Observe de nuevo que en el factor de corrección por transformación lognormal, se invoca a modelo$sigma2 el cual es igual a $\hat{\sigma}_a^2$

```
#Cálculo del pronóstico con I.P del 95%, en escala original
predmodelo=exp(as.data.frame(forecast(modelo,xreg=Xnuevo,level=95)))*exp(modelo$sigma2/2)
predmodelo=ts(predmodelo,freq=12,start=c(1974,1))
predmodelo
ytpronmodelo=predmodelo[,1] #Tomando el pronóstico puntual

#Medidas precisión pronósticos
accuracy(ytpronmodelo,ytf)
amplitud(LIP=predmodelo[,2],LSP=predmodelo[,3])
cobertura(real=ytf,LIP=predmodelo[,2],LSP=predmodelo[,3])
```

III. Modelos lineales en los parámetros de tendencia y estacionalidad, y errores ARMA(p,q)(P,Q)[s]

Por ejemplo,

$Y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \sum_{i=1}^{11} \delta_i I_{i_t} + E_t$, donde E_t es un ARMA(p,q)(P,Q)[12], es decir $\phi_p(B)\Phi_P(B^{12})E_t = \theta_q(B)\Theta_Q(B^{12})a_t$, con $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma_a^2)$. Entonces, el modelo queda como $Y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \sum_{i=1}^{11} \delta_i I_{i_t} + E_t$, con

$$E_t = \sum_{j=1}^{p} \phi_j E_{t-j} + \sum_{k=1}^{P} \Phi_k\, E_{t-12*k} - \sum_{k=1}^{P} \Phi_k \sum_{j=1}^{p} \phi_j E_{t-j-12*k} + a_t + \sum_{i=1}^{q} \theta_i a_{t-i} + \sum_{l=1}^{Q} \Theta_l a_{t-12*l} + \sum_{l=1}^{Q} \Theta_l \sum_{i=1}^{q} \theta_i a_{t-i-12*l}$$

y $\{a_t\}_{t\in\mathbb{Z}^+}$ un $RB \sim N(0,\sigma_a^2)$

1) Lea datos y defina variables necesarias para el ajuste y el pronóstico (en este ejemplo se asume 12 pronósticos ex – post)

```
datos=read.table(file.choose(),....) #leer conforme a la base de datos asignada
datos=ts(datos,freq=12,start=c(1960,1)) #convertir en serie según frecuencia y fecha inicial
n=length(datos)-12 #Defina longitud de los datos para ajuste con validación cruzada, según su caso
#Defina el índice de tiempo y sus potencias según polinomio
t=1:n; t2=t^2; t3=t^3
yt=ts(datos[t],freq=12,start=c(1960,1)) #serie recortada

#Defina la matriz de variables indicadoras
Indicadoras=seasonaldummy(yt)

#Defina la matriz de variables predictoras de tendencia y estacionalidad:
#van juntas todas las variables de la tendencia y de la estacionalidad en ese orden
X=cbind(t,t2,t3,Indicadoras)

#Definiendo variables para pronósticos
tnuevo=(n+1):length(datos)
t2nuevo=tnuevo^2
t3nuevo=tnuevo^3
Indicadorasnuevo=seasonaldummy(yt,h=12) #Horizonte de pronóstico h igual a longitud de recorte de la serie
Xnuevo=cbind(t=tnuevo,t2=t2nuevo,t3=t3nuevo,Indicadoras=Indicadorasnuevo) #matriz predictores en el pronóstico
ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según su caso
```

2) Ajuste el modelo con función Arima de librería forecast. El orden p, q se especifica con order=c(p,0,q), mientras que el orden P, Q se especifica con seasonal=list(order=c(P,0,Q))

```
#indique valor correspondiente a p, q, P y Q en la función Arima, así como la matriz de predictores X en xreg=X
modelo=Arima(yt,order=c(p,0,q),seasonal=list(order=c(P,0,Q)),xreg=X,method="ML")
```

3) Construya tabla completa de parámetros estimados, valores P corresponden a $P(|t_{n-k}| > |T_0|)$, con k la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA.

```
#Calcule grados de libertad del MSE del modelo,
k=length(coef(modelo)[coef(modelo)!=0]);k #número de parámetros del modelo
dfmodelo=n-k

#Construya tabla de parámetros estimados con estadísticos T0 y valores P para cada parámetro
coeftest(modelo,df=dfmodelo)
```

4) Obtenga gráfica del ajuste de la serie en escala original y de los residuales de ajuste $\hat{a}_t$.

```
ythat=modelo$fitted #este objeto ya queda con las fechas de la serie
#Gráfico de la serie y su ajuste
plot(datos)
lines(ythat,col=2)
legend("topleft",legend=c("datos","ajuste modelo"),lty=1,col=1:2)

#Gráficos de residuales
plot(residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
plot(as.numeric(modelo$fitted),residuals(modelo));abline(h=0)
abline(h=c(-2*sqrt(modelo$sigma2),2*sqrt(modelo$sigma2)),lty=2)
```

5) Cálculo de AIC y BIC, versión $\exp(C_n^*(p))$, con el número de parámetros k siendo la suma del número de parámetros en las estructuras de tendencia, estacionalidad y errores ARMA, previamente calculado en 3).

```
AICmodelo=exp(crit.inf.resid(residuales=residuals(modelo),n.par=k))
BICmodelo=exp(crit.inf.resid(residuales= residuals(modelo),n.par=k,AIC="FALSE"))
```

6) Valide supuestos sobre el error de ajuste a_t

```
#ACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
acf(as.numeric(residuals(modelo)),ci.type="ma",lag.max=m,main="ACF modelo",ci.col=2)

#PACF sobre residuales de ajuste en el modelo. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
pacf(as.numeric(residuals(modelo)),lag.max=m,main="PACF modelo",ci.col=2)

BP.LB.test(residuals(modelo),maxlag=m,type="Ljung") #test Ljung-Box use máximo m igual al de ACF y PACF
#Normalidad sobre residuales de ajuste en el modelo. Sólo si no se rechaza supuesto de ruido blanco
shapiro.test(residuals(modelo))

win.graph()
qqnorm(residuals(modelo),main="Gráfico de normalidad residuos de ajuste modelo")
qqline(residuals(modelo),col=2)
```

7) Pronósticos para la validación cruzada (en este ejemplo se supone que son 12 los pronósticos)

```
#Cálculo del pronóstico con I.P del 95%, en escala original
predmodelo=ts(as.data.frame(forecast(modelo,xreg=Xnuevo,level=95)),freq=12, start=c(1974,1))
predmodelo
ytpronmodelo=predmodelo[,1] #Tomando el pronóstico puntual. Este objeto tiene fechas

#Medidas precisión pronósticos
accuracy(ytpronmodelo,ytf)
amplitud(LIP=predmodelo[,2],LSP=predmodelo[,3])
cobertura(real=ytf,LIP=predmodelo[,2],LSP=predmodelo[,3])
```

IV. Modelos exponenciales con errores ARMA.

Ningún modelo donde la estructura de regresión o ecuación estructural, es no lineal en los parámetros de tendencia y estacionalidad, podrá ajustarse con la función Arima(). Por otro lado la función disponible en R para modelos de regresión no lineal con errores ARMA, gnls() de la librería nlme, no logra correr para muchos casos y es difícil ajustar sus argumentos de control para lograr que estime. La solución aproximada para este inconveniente es ajustar y pronosticar separadamente la estructura de regresión no lineal (el modelo exponencial), luego usar los residuales del modelo exponencial para ajustar y pronosticar la estructura ARMA y finalmente, juntar los valores ajustados y pronosticados de estas dos partes, así:

1. Ajustar y pronosticar la estructura exponencial: el ajuste con la función nls() sobre la serie de datos asignados (serie recortada para el ajuste con validación cruzada) y el pronóstico de la estructura exponencial con la función predict() sobre el objeto que guarde el ajuste exponencial;
2. Ajustar y pronosticar la estructura ARMA: el ajuste con la función Arima() sobre los residuos estructurales que genera el ajuste exponencial y el pronóstico con la función forecast() sobre el objeto que guarde el ajuste ARMA;
3. Finalmente procedemos a construir una estimación y pronóstico total para la serie, sumando los respectivos ajustes y pronósticos de la regresión no lineal con los correspondientes ajustes y pronósticos del ARMA: es decir, $\hat{Y}_t \approx \widehat{\mathrm{EXP}}_t + \widehat{\hat{\mathrm{E}}}_t$, $\hat{Y}_n(L) \approx \widehat{\mathrm{EXP}}_n(L) + \hat{\mathrm{E}}_n(L)$, donde $\widehat{\mathrm{EXP}}_t$ y $\widehat{\mathrm{EXP}}_n(L)$ son respectivamente, el ajuste y el pronóstico de la estructura de regresión exponencial, y $\hat{\hat{E}}_t$ y $\hat{\mathrm{E}}_n(L)$ son respectivamente, el ajuste y el pronóstico de la estructura ARMA de media cero y estacionaria usando sólo los residuos del ajuste del modelo exponencial del trabajo 1.

Vea en los siguientes ejemplos cómo se implementa lo anterior.

Pasos iniciales: Cargar librerías y definir funciones de usuario necesarias

```
library(forecast);library(lmtest)

#Funciones para el cálculo aproximado de AIC y BIC para ajuste de estructura exponencial con error ARMA
AIC.aprox=function(modeloarma,n1=n,npar.estruct){
k=length(coef(modeloarma)[coef(modeloarma)!=0])+npar.estruct
aic=log(mean(residuals(modeloarma)^2))+2*k/n1
aic
}
BIC.aprox=function(modeloarma,n1,npar.estruct){
k=length(coef(modeloarma)[coef(modeloarma)!=0])+npar.estruct
bic=log(mean(residuals(modeloarma)^2))+k*log(n1)/n1
bic
}

#DEFINIENDO FUNCIÓN USUARIO PARA TEST LJUNG-BOX
BP.LB.test=function(serie,maxlag,type="Box"){
aux=floor(maxlag/6); X.squared=c(rep(NA,aux))
df=c(rep(NA,aux)); p.value=c(rep(NA,aux))
for(i in 1:aux){
test=Box.test(serie,lag=(6*i),type=type)
X.squared[i]=test[[1]]; df[i]=test[[2]]
p.value[i]=test[[3]]
}
lag=6*c(1:aux)
teste=as.data.frame(cbind(X.squared,df,p.value))
rownames(teste)=lag; teste
}
```

Nota: En los siguientes ejemplos, el paso 1) solo se realiza una vez, no es necesario correrlo con cada modelo distinto en los errores. En los pasos 2) a 7), tenga en cuenta que, por cada modelo distinto a considerar en los errores, debe crear con diferentes nombres los objetos que guarden valores ajustados, residuos, pronósticos, etc.

a. Modelo de regresión exponencial con errores ARMA(p,q) con todos los parámetros. Considere el siguiente ejemplo,
$Y_t = \exp\left(\beta_0 + \beta_1 t + \beta_2 t^2 + \sum_{i=1}^{11} \delta_i I_{i_t}\right) + E_t$, donde $E_t = \sum_{j=1}^{p} \phi_j E_{t-j} + a_t + \sum_{i=1}^{q} \theta_i a_{t-i}$, con a_t un $RB \sim N(0, \sigma_a^2)$,

1) Lectura datos, ajuste y pronóstico sobre la estructura exponencial (en este ejemplo se asume 12 pronósticos ex – post).

```
datos=read.table(file.choose(),....) #leer conforme a la base de datos asignada
datos=ts(datos,freq=12,start=c(1960,1)) #convertir en serie según frecuencia y fecha inicial
n=length(datos)-12 #Defina longitud de los datos para ajuste con validación cruzada. Aquí se recorta en 12 datos
#Defina el índice de tiempo y sus potencias según polinomio planteado para Tt
t=1:n; t2=t^2
yt=ts(datos[t],freq=12,start=c(1960,1)) #serie recortada
#Defina la matriz de variables indicadoras y separe las indicadoras
Indicadoras=seasonaldummy(yt)
I1=Indicadoras[,1]
I2=Indicadoras[,2]
I3=Indicadoras[,3]
```

```
I4=Indicadoras[,4]
I5=Indicadoras[,5]
I6=Indicadoras[,6]
I7=Indicadoras[,7]
I8=Indicadoras[,8]
I9=Indicadoras[,9]
I10=Indicadoras[,10]
I11=Indicadoras[,11]

Definiendo variables para pronósticos de la estructura exponencial
tnuevo=(n+1):length(datos)
t2nuevo=tnuevo^2
Indicadorasnuevo=seasonaldummy(yt,h=12) #Horizonte de pronóstico h igual a longitud de recorte de la serie
I1n=Indicadorasnuevo[,1]
I2n=Indicadorasnuevo[,2]
I3n=Indicadorasnuevo[,3]
I4n=Indicadorasnuevo[,4]
I5n=Indicadorasnuevo[,5]
I6n=Indicadorasnuevo[,6]
I7n=Indicadorasnuevo[,7]
I8n=Indicadorasnuevo[,8]
I9n=Indicadorasnuevo[,9]
I10n=Indicadorasnuevo[,10]
I11n=Indicadorasnuevo[,11]

ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según su caso

Aux=lm(log(yt)~t+t2+ I1+I2+I3+I4+I5+I6+I7+I8+I9+I10+I11) #modelo auxiliar para valores de inicio de parámetros

#Guardando coeficientes estimados para inicializar parámetros del modelo que sigue
bin=coef(Aux)

#Modelo exponencial
modexpo=nls(yt~exp(b0+b1*t+b2*I(t^2)+d1*I1+d2*I2+d3*I3+d4*I4+d5*I5+d6*I6+d7*I7+d8*I8+d9*I9+d10*I10+d11*I11),
start=list(b0=bin[1],b1=bin[2],b2=bin[3],d1=bin[4],d2=bin[5],d3=bin[6],d4=bin[7],d5=bin[8],d6=bin[9],d7=bin[10],
d8=bin[11],d9=bin[12],d10=bin[13],d11=bin[14]))
summary(modexpo)

#Estimación de la estructura exponencial
hatestructura=ts(fitted(modexpo),freq=12,start=c(1960,1))

#Pronósticos estructura exponencial
pronexpo=predict(modexpo,newdata=data.frame(t=tnuevo,t2=t2nuevo,I1=I1n,I2=I2n,I3=I3n,I4=I4n,I5=I5n,I6=I6n,
I7=I7n,I8=I8n,I9=I9n,I10=I10n,I11=I11n))
pronexpo=ts(pronexpo,freq=12,start=c(1974,1))

#Guarde el número de parámetros de la estructura exponencial
pestruct=length(coef(modexpo))
```

2) Ajuste el modelo ARMA sobre los residuos de ajuste del modelo exponencial, con función Arima de librería forecast. Observe que se pide que la media sea cero y no se usa una matriz de regresión X.

```
#indique el valor correspondiente a p y q en la función Arima y una media de cero
ciclos=Arima(residuals(modexpo),order=c(p,0,q),include.mean=F, method="ML")
```

3) Construya tabla de parámetros estimados para el modelo ARMA de los errores estructurales. ***Valores P corresponden a $P(|Z| > |Z_0|)$, es decir, bajo aproximación N(0,1).***

```
#Construya tabla de parámetros estimados con estadísticos Z0 y valores P para cada parámetro
coeftest(ciclos)
```

4) Obtenga valor ajustado y gráfica del ajuste de la serie en escala original y de los residuales de ajuste $\hat{a}_t$.

```
hatciclos=ts(ciclos$fitted,freq=12,start=c(1960,1)) #Ciclos ajustados. Fechas y frecuencia según serie
ythat=hatestructura+hatciclos #Ajuste total de la serie
#Gráfico de la serie y su ajuste
plot(datos)
lines(ythat,col=2)
legend("topleft",legend=c("datos","ajuste modelo"),lty=1,col=1:2)

#Gráficos de residuales del ajuste
plot(residuals(ciclos));abline(h=0)
abline(h=c(-2*sqrt(ciclos$sigma2),2*sqrt(ciclos$sigma2)),lty=2)
plot(as.numeric(ciclos$fitted),residuals(ciclos));abline(h=0)
abline(h=c(-2*sqrt(ciclos$sigma2),2*sqrt(ciclos$sigma2)),lty=2)
```

5) Cálculo de AIC y BIC, versión $\exp(C_n^*(p))$. Debe usar las funciones usuario AIC.aprox y BIC.aprox. NOTA: El objeto *pestruct* es el que se creó al final del paso 1)

```
#Cálculo aproximado de AIC y BIC para ajuste de estructura exponencial más ciclos arma
AICajuste=exp(AIC.aprox(modeloarma=ciclos,n1=n,npar.estruct=pestruct))
BICajuste=exp(BIC.aprox(modeloarma=ciclos,n1=n,npar.estruct=pestruct))
```

6) Valide supuestos sobre el error de ajuste a_t

```
#ACF sobre residuales de ajuste. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
acf(as.numeric(residuals(ciclos)),ci.type="ma",lag.max=m,main="ACF modelo",ci.col=2)
#PACF sobre residuales de ajuste. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
pacf(as.numeric(residuals(ciclos)),lag.max=m,main="PACF modelo",ci.col=2)
BP.LB.test(residuals(ciclos),maxlag=m,type="Ljung") #test Ljung-Box use máximo m igual al de ACF y PACF
#Normalidad sobre residuales de ajuste en el modelo. Sólo si no se rechaza supuesto de ruido blanco
shapiro.test(residuals(ciclos))

qqnorm(residuals(ciclos),main="Gráfico de normalidad residuos de ajuste modelo")
qqline(residuals(ciclos),col=2)
```

7) Pronósticos para la validación cruzada. No es posible obtener pronósticos por I.P para toda la serie. Tampoco interesan los I.P para los errores.

```
#Cálculo del pronóstico de los ciclos arma. Horizonte de pronósticos h según cantidad de datos recortados
pronciclos=ts(forecast(ciclos,h=12)$mean,freq=12, start=c(1974,1)) #Frecuencia y fechas según los datos

#Construya pronóstico total de la serie
ytpron=pronexpo+pronciclos;ytpron

#Medidas precisión pronósticos puntuales
accuracy(ytpron,ytf)
```

b. Modelo de regresión exponencial con errores ARMA(p,q) con algunos de los parámetros fijos en cero. Considere el siguiente ejemplo,
$Y_t = \exp\left(\beta_0 + \beta_1 t + \beta_2 t^2 + \sum_{i=1}^{11} \delta_i I_{i_t}\right) + E_t$, donde $E_t = \phi_1 E_{t-1} + \phi_9 E_{t-9} + a_t + \theta_{12} a_{t-12}$, con a_t un $RB \sim N(0, \sigma_a^2)$. Se procede de la siguiente manera:

1) Lectura datos, ajuste y pronóstico sobre la estructura exponencial (en este ejemplo se asume 12 pronósticos ex – post).

```
datos=read.table(file.choose(),....) #leer conforme a la base de datos asignada
datos=ts(datos,freq=12,start=c(1960,1)) #convertir en serie según frecuencia y fecha inicial
n=length(datos)-12 #Defina longitud de los datos para ajuste con validación cruzada. Aquí se recorta en 12 datos
#Defina el índice de tiempo y sus potencias según polinomio planteado para Tt
t=1:n; t2=t^2
yt=ts(datos[t],freq=12,start=c(1960,1)) #serie recortada
#Defina la matriz de variables indicadoras y separe las indicadoras
Indicadoras=seasonaldummy(yt)
I1=Indicadoras[,1]
I2=Indicadoras[,2]
I3=Indicadoras[,3]
I4=Indicadoras[,4]
I5=Indicadoras[,5]
I6=Indicadoras[,6]
I7=Indicadoras[,7]
I8=Indicadoras[,8]
I9=Indicadoras[,9]
I10=Indicadoras[,10]
I11=Indicadoras[,11]

Definiendo variables para pronósticos de la estructura exponencial
tnuevo=(n+1):length(datos)
t2nuevo=tnuevo^2
Indicadorasnuevo=seasonaldummy(yt,h=12) #Horizonte de pronóstico h igual a longitud de recorte de la serie
I1n=Indicadorasnuevo[,1]
I2n=Indicadorasnuevo[,2]
I3n=Indicadorasnuevo[,3]
I4n=Indicadorasnuevo[,4]
I5n=Indicadorasnuevo[,5]
I6n=Indicadorasnuevo[,6]
I7n=Indicadorasnuevo[,7]
I8n=Indicadorasnuevo[,8]
I9n=Indicadorasnuevo[,9]
I10n=Indicadorasnuevo[,10]
```

```
I11n=Indicadorasnuevo[,11]

ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según su caso

Aux=lm(log(yt)~t+t2+ I1+I2+I3+I4+I5+I6+I7+I8+I9+I10+I11) #modelo auxiliar para valores de inicio de parámetros

#Guardando coeficientes estimados para inicializar parámetros del modelo que sigue
bin=coef(Aux)

#Modelo exponencial
modexpo=nls(yt~exp(b0+b1*t+b2*I(t^2)+d1*I1+d2*I2+d3*I3+d4*I4+d5*I5+d6*I6+d7*I7+d8*I8+d9*I9+d10*I10+d11*I11),
start=list(b0=bin[1],b1=bin[2],b2=bin[3],d1=bin[4],d2=bin[5],d3=bin[6],d4=bin[7],d5=bin[8],d6=bin[9],d7=bin[10],
d8=bin[11],d9=bin[12],d10=bin[13],d11=bin[14]))
summary(modexpo)

#Estimación de la estructura exponencial
hatestructura=ts(fitted(modexpo),freq=12,start=c(1960,1))

#Pronósticos estructura exponencial
pronexpo=predict(modexpo,newdata=data.frame(t=tnuevo,t2=t2nuevo,I1=I1n,I2=I2n,I3=I3n,I4=I4n,I5=I5n,I6=I6n,
I7=I7n,I8=I8n,I9=I9n,I10=I10n,I11=I11n))
pronexpo=ts(pronexpo,freq=12,start=c(1974,1))

#Guarde el número de parámetros de la estructura exponencial
pestruct=length(coef(modexpo))
```

2) Ajuste modelo ARMA sobre los residuales del modelo exponencial. Dado que no se usan todos los parámetros del modelo ARMA(p,q) en los errores estructurales E_t, en la función Arima debe especificarse cuáles parámetros van en el modelo y cuáles se fijan en cero. Para ello, se usa el argumento fixed igual a un vector de longitud (p+q). Las entradas o valores de este vector se ordenan de la siguiente manera: Los primeros p valores son relativos a los coeficientes ϕ_j, los siguientes q valores hacen referencia a los coeficientes θ_i. En cada entrada del vector se coloca un 0 si el parámetro no va en el modelo y un NA, si el respectivo parámetro debe ir en el modelo. Para el ejemplo, como el error estructural es un ARMA(9,12) pero sólo los coeficientes ϕ_1, ϕ_9 y θ_{12} van en la estructura ARMA, el argumento fixed en la función Arima debe ser especificado así (OJO: rep(a,b) repite b veces el argumento a),

fixed=c(NA,rep(0,7),NA,rep(0,11),NA)
que es equivalente a escribir de forma extendida lo siguiente,
fixed=c(NA,0,0,0,0,0,0,0,NA,0,0,0,0,0,0,0,0,0,0,0,NA)

donde el color del caracter indica
Las entradas para los parámetros parte AR(9)
Las entradas para los parámetros parte MA(12)

```
#indique el valor correspondiente a p y q en la función Arima, no hay matriz de predictores,
#en cambio fije media en cero y especifique el vector para argumento fixed
ciclos=Arima(residuals(modexpo),order=c(9,0,12),include.mean=F,fixed=c(NA,rep(0,7),NA,rep(0,11),NA),method="ML")
```

3) Construya tabla de parámetros estimados para el modelo ARMA de los errores estructurales. ***Valores P corresponden a $P(|Z| > |Z_0|)$, es decir, bajo aproximación N(0,1).***

```
#Construya tabla de parámetros estimados con estadísticos Z0 y valores P para cada parámetro
coeftest(ciclos)
```

4) Obtenga valor ajustado y gráfica del ajuste de la serie en escala original y de los residuales de ajuste $\hat{a}_t$.

```
hatciclos=ts(ciclos$fitted,freq=12,start=c(1960,1)) #Fechas y frecuencia según serie
ythat=hatestructura+hatciclos #Ajuste total de la serie

#Gráfico de la serie y su ajuste
plot(datos)
lines(ythat,col=2)
legend("topleft",legend=c("datos","ajuste modelo"),lty=1,col=1:2)

#Gráficos de residuales del ajuste
plot(residuals(ciclos));abline(h=0)
abline(h=c(-2*sqrt(ciclos$sigma2),2*sqrt(ciclos$sigma2)),lty=2)
plot(as.numeric(ciclos$fitted),residuals(ciclos));abline(h=0)
abline(h=c(-2*sqrt(ciclos$sigma2),2*sqrt(ciclos$sigma2)),lty=2)
```

5) Cálculo de AIC y BIC, versión $\exp(C_n^*(p))$. Debe usar las funciones usuario AIC.aprox y BIC.aprox. NOTA: El objeto *pestruct* es el que se creó al final del paso 1)

```
AICajuste=exp(AIC.aprox(modeloarma=ciclos,n1=n,npar.estruct=pestruct))
BICajuste=exp(BIC.aprox(modeloarma=ciclos,n1=n,npar.estruct=pestruct))
```

6) Valide supuestos sobre el error de ajuste a_t

```
#ACF sobre residuales de ajuste. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
acf(as.numeric(residuals(ciclos)),ci.type="ma",lag.max=m,main="ACF modelo",ci.col=2)

#PACF sobre residuales de ajuste. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
pacf(as.numeric(residuals(ciclos)),lag.max=m,main="PACF modelo",ci.col=2)

BP.LB.test(residuals(ciclos),maxlag=m,type="Ljung") #test Ljung-Box use máximo m igual al de ACF y PACF
#Normalidad sobre residuales de ajuste en el modelo. Sólo si no se rechaza supuesto de ruido blanco
shapiro.test(residuals(ciclos))
win.graph()
qqnorm(residuals(ciclos),main="Gráfico de normalidad residuos de ajuste modelo")
qqline(residuals(ciclos),col=2)
```

7) Pronósticos para la validación cruzada. No es posible obtener pronósticos por I.P para toda la serie. Tampoco interesan los I.P para los errores.

```
#Cálculo del pronóstico de los ciclos arma. Horizonte de pronósticos h según cantidad de datos recortados
pronciclos=ts(forecast(ciclos,h=12)$mean,freq=12, start=c(1974,1)) #Frecuencia y fechas según los datos

#Construya pronóstico total de la serie
ytpron=pronexpo+pronciclos
ytpron

#Medidas precisión pronósticos puntuales
accuracy(ytpron,ytf)
```

c. Modelo de regresión exponencial con errores ARMA(p,q)(P,Q)[s]. Considere el siguiente ejemplo,
$Y_t = \exp\left(\beta_0 + \beta_1 t + \beta_2 t^2 + \sum_{i=1}^{11} \delta_i I_{i_t}\right) + E_t$, donde E_t es un ARMA(p,q)(P,Q)[12], es decir
$\phi_p(B)\Phi_P(B^{12})E_t = \theta_q(B)\Theta_Q(B^{12})a_t$, con a_t un $RB \sim N(0,\sigma_a^2)$. Entonces, el modelo queda como

$Y_t = \exp\left(\beta_0 + \beta_1 t + \beta_2 t^2 + \sum_{i=1}^{11} \delta_i I_{i_t}\right) + E_t$, con

$$E_t = \sum_{j=1}^{p} \phi_j E_{t-j} + \sum_{k=1}^{P} \Phi_k\, E_{t-12*k} - \sum_{k=1}^{P} \Phi_k \sum_{j=1}^{p} \phi_j E_{t-j-12*k} + a_t + \sum_{i=1}^{q} \theta_i a_{t-i} + \sum_{l=1}^{Q} \Theta_l a_{t-12*l} + \sum_{l=1}^{Q} \Theta_l \sum_{i=1}^{q} \theta_i a_{t-i-12*l}$$

y a_t un $RB \sim N(0,\sigma_a^2)$

1) Lectura datos, ajuste y pronóstico sobre la estructura exponencial (en este ejemplo se asume 12 pronósticos ex – post).

```
datos=read.table(file.choose(),....) #leer conforme a la base de datos asignada
datos=ts(datos,freq=12,start=c(1960,1)) #convertir en serie según frecuencia y fecha inicial
n=length(datos)-12 #Defina longitud de los datos para ajuste con validación cruzada. Aquí se recorta en 12 datos
#Defina el índice de tiempo y sus potencias según polinomio planteado para Tt
t=1:n; t2=t^2
yt=ts(datos[t],freq=12,start=c(1960,1)) #serie recortada
#Defina la matriz de variables indicadoras y separe las indicadoras
Indicadoras=seasonaldummy(yt)
I1=Indicadoras[,1]
I2=Indicadoras[,2]
I3=Indicadoras[,3]
I4=Indicadoras[,4]
I5=Indicadoras[,5]
I6=Indicadoras[,6]
I7=Indicadoras[,7]
I8=Indicadoras[,8]
I9=Indicadoras[,9]
I10=Indicadoras[,10]
I11=Indicadoras[,11]

Definiendo variables para pronósticos de la estructura exponencial
tnuevo=(n+1):length(datos)
t2nuevo=tnuevo^2
Indicadorasnuevo=seasonaldummy(yt,h=12) #Horizonte de pronóstico h igual a longitud de recorte de la serie
I1n=Indicadorasnuevo[,1]
I2n=Indicadorasnuevo[,2]
```

```
I3n=Indicadorasnuevo[,3]
I4n=Indicadorasnuevo[,4]
I5n=Indicadorasnuevo[,5]
I6n=Indicadorasnuevo[,6]
I7n=Indicadorasnuevo[,7]
I8n=Indicadorasnuevo[,8]
I9n=Indicadorasnuevo[,9]
I10n=Indicadorasnuevo[,10]
I11n=Indicadorasnuevo[,11]

ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según su caso

Aux=lm(log(yt)~t+t2+ I1+I2+I3+I4+I5+I6+I7+I8+I9+I10+I11) #modelo auxiliar para valores de inicio de parámetros

#Guardando coeficientes estimados para inicializar parámetros del modelo que sigue
bin=coef(Aux)

#Modelo exponencial
modexpo=nls(yt~exp(b0+b1*t+b2*I(t^2)+d1*I1+d2*I2+d3*I3+d4*I4+d5*I5+d6*I6+d7*I7+d8*I8+d9*I9+d10*I10+d11*I11),
start=list(b0=bin[1],b1=bin[2],b2=bin[3],d1=bin[4],d2=bin[5],d3=bin[6],d4=bin[7],d5=bin[8],d6=bin[9],d7=bin[10],
d8=bin[11],d9=bin[12],d10=bin[13],d11=bin[14]))
summary(modexpo)

#Estimación de la estructura exponencial
hatestructura=ts(fitted(modexpo),freq=12,start=c(1960,1))

#Pronósticos estructura exponencial
pronexpo=predict(modexpo,newdata=data.frame(t=tnuevo,t2=t2nuevo,I1=I1n,I2=I2n,I3=I3n,I4=I4n,I5=I5n,I6=I6n,
I7=I7n,I8=I8n,I9=I9n,I10=I10n,I11=I11n))
pronexpo=ts(pronexpo,freq=12,start=c(1974,1))

#Guarde el número de parámetros en la estructura exponencial
pestruct=length(coef(modexpo))
```

2) Ajuste los ciclos ARMA con función Arima de librería forecast. Observe que se aplica sólo a los residuales del modelo exponencial convertidos en serie de tiempo, se pide además que la media sea cero y no se usa una matriz de regresión X. El orden p, q se especifica con order=c(p,0,q), mientras que el orden P, Q se especifica con seasonal=list(order=c(P,0,Q))

```
#Crear serie de tiempo para residuales del modelo exponencial
serieEt=ts(residuals(modexpo),freq=12,start=c(1960,1)) #Frecuencia y fechas según los datos

#indique el valor correspondiente a p, q, P y Q en la función Arima, y que la media es cero
ciclos=Arima(serieEt,order=c(p,0,q),seasonal=list(order=c(P,0,Q)),include.mean=F, method="ML")
```

3) Construya tabla de parámetros estimados para el modelo ARMA de los errores estructurales. ***Valores P corresponden a $P(|Z| > |Z_0|)$, es decir, bajo aproximación N(0,1).***

```
#Construya tabla de parámetros estimados con estadísticos Z0 y valores P para cada parámetro
coeftest(ciclos)
```

4) Obtenga valor ajustado y gráfica del ajuste de la serie en escala original y de los residuales de ajuste $\hat{a}_t$.

```
hatciclos=ts(ciclos$fitted,freq=12,start=c(1960,1)) #Ciclos ajustados. Fechas y frecuencia según serie
ythat=hatestructura+hatciclos #Ajuste total de la serie
#Gráfico de la serie y su ajuste
plot(datos)
lines(ythat,col=2)
legend("topleft",legend=c("datos","ajuste modelo"),lty=1,col=1:2)

#Gráficos de residuales del ajuste
plot(residuals(ciclos));abline(h=0)
abline(h=c(-2*sqrt(ciclos$sigma2),2*sqrt(ciclos$sigma2)),lty=2)
plot(as.numeric(ciclos$fitted),residuals(ciclos));abline(h=0)
abline(h=c(-2*sqrt(ciclos$sigma2),2*sqrt(ciclos$sigma2)),lty=2)
```

5) Cálculo de AIC y BIC, versión $\exp(C_n^*(p))$. Debe usar las funciones usuario AIC.aprox y BIC.aprox. NOTA: El objeto *pestruct* es el que se creó al final del paso 1)

```
AICajuste=exp(AIC.aprox(modeloarma=ciclos,n1=n,npar.estruct=pestruct))
BICajuste=exp(BIC.aprox(modeloarma=ciclos,n1=n,npar.estruct=pestruct))
```

6) Valide supuestos sobre el error de ajuste a_t

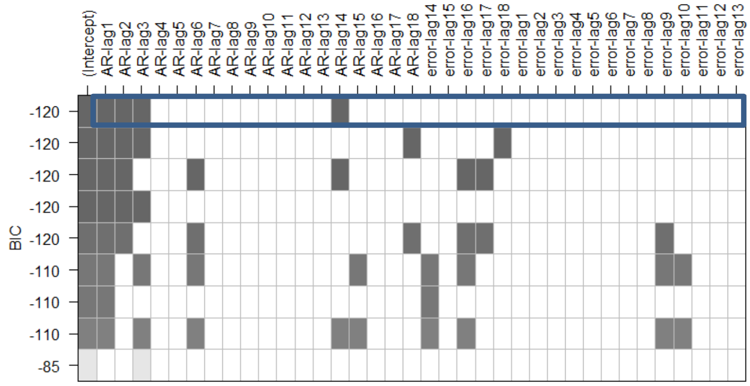
```
#ACF sobre residuales de ajuste. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
acf(as.numeric(residuals(ciclos)),ci.type="ma",lag.max=m,main="ACF modelo",ci.col=2)

#PACF sobre residuales de ajuste. Use valor para m el que se indica en la guía del trabajo
win.graph(width=4.875,height=3.5,pointsize=8)
pacf(as.numeric(residuals(ciclos)),lag.max=m,main="PACF modelo",ci.col=2)

BP.LB.test(residuals(ciclos),maxlag=m,type="Ljung") #test Ljung-Box use máximo m igual al de ACF y PACF
#Normalidad sobre residuales de ajuste en el modelo. Sólo si no se rechaza supuesto de ruido blanco
shapiro.test(residuals(ciclos))

win.graph()
qqnorm(residuals(ciclos),main="Gráfico de normalidad residuos de ajuste modelo")
qqline(residuals(ciclos),col=2)
```

7) Pronósticos para la validación cruzada. No es posible obtener pronósticos por I.P para toda la serie. Tampoco interesan los I.P para los errores.

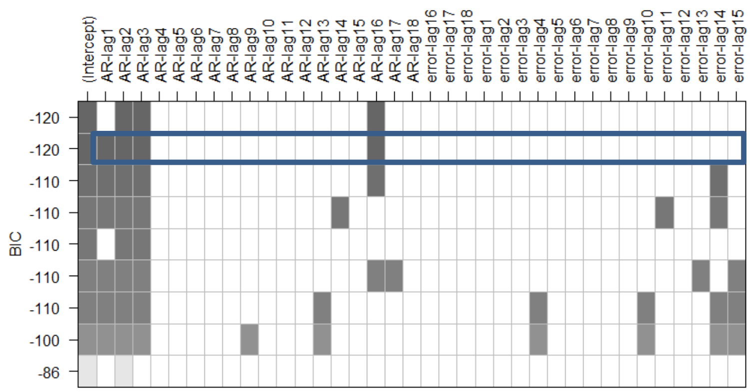
```
#Cálculo del pronóstico de los ciclos arma. Horizonte de pronósticos h según cantidad de datos recortados
pronciclos=ts(forecast(ciclos,h=12)$mean,freq=12, start=c(1974,1)) #Frecuencia y fechas según los datos

#Construya pronóstico total de la serie
ytpron=pronexpo+pronciclos
ytpron

#Medidas precisión pronósticos puntuales
ytf=ts(datos[tnuevo],freq=12,start=c(1974,1)) #los datos no usados en ajuste. Fechas y frecuencias según su caso
accuracy(ytpron,ytf)
```

APÉNDICE C. Gráficos generados con función armasubsets sobre residuos de ajuste del modelo global listado en la Tabla 1.

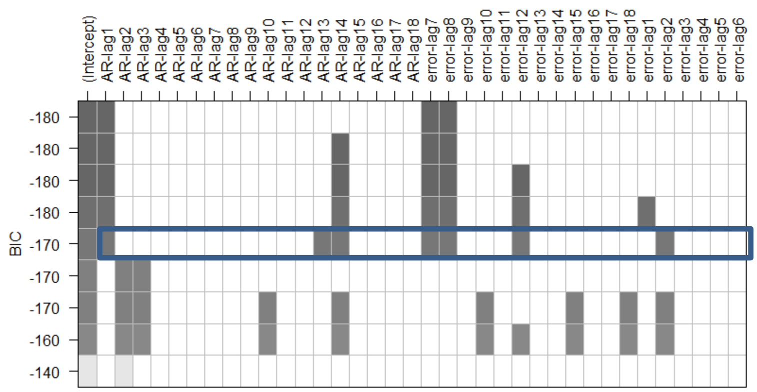
Datos1: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico, e incluir ϕ_5



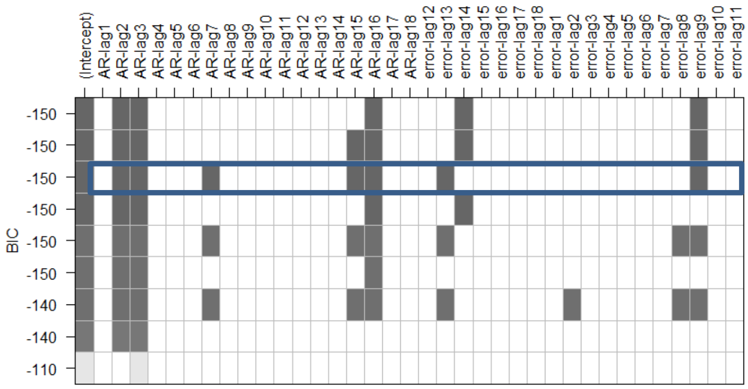
Datos3: Usar el renglón 2 señalado con rectángulo azul sobre el gráfico



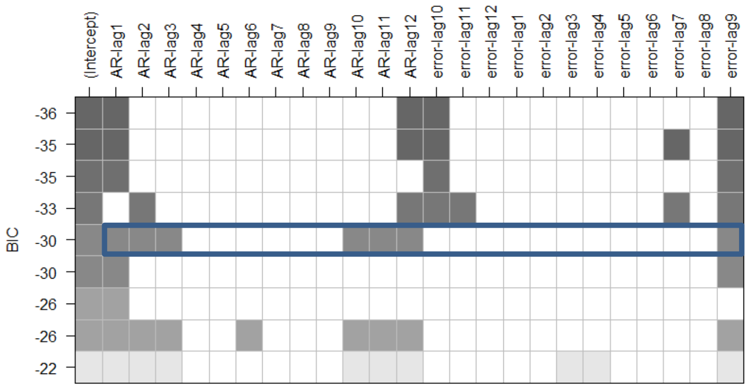
Datos4: Usar el renglón 5 señalado con rectángulo azul sobre el gráfico, e incluir ϕ_2, θ_1



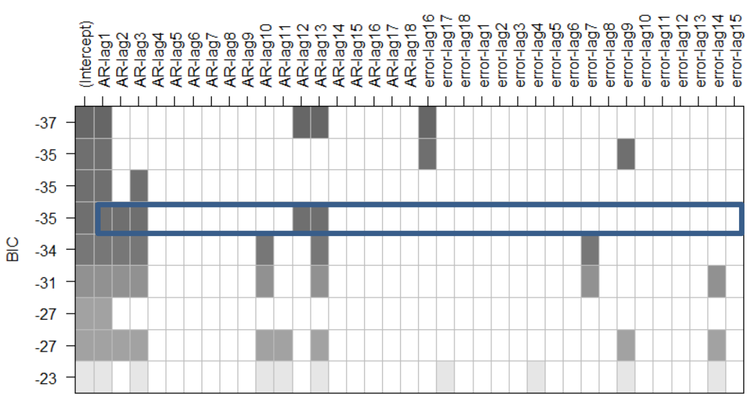
Datos6: Usar el renglón 3 señalado con rectángulo azul sobre el gráfico, e incluir θ_2, θ_3



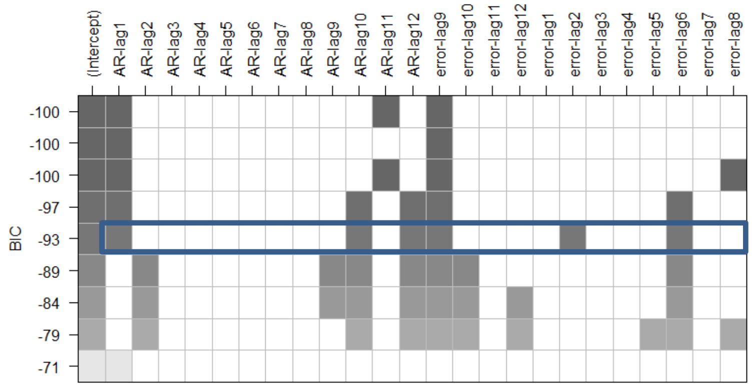
Datos9: Usar el renglón 5 señalado con rectángulo azul sobre el gráfico



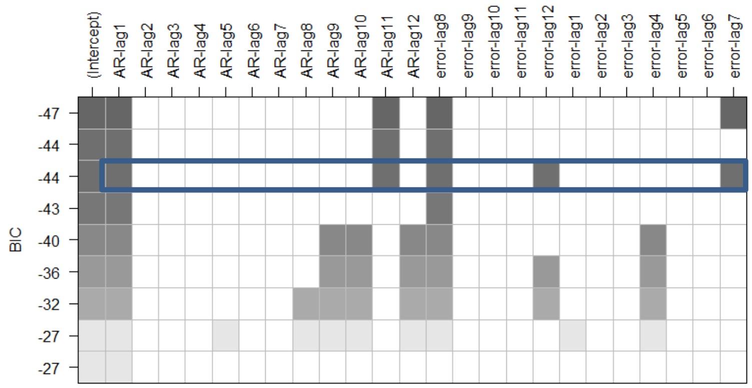
Datos10: Usar el renglón 4 señalado con rectángulo azul sobre el gráfico



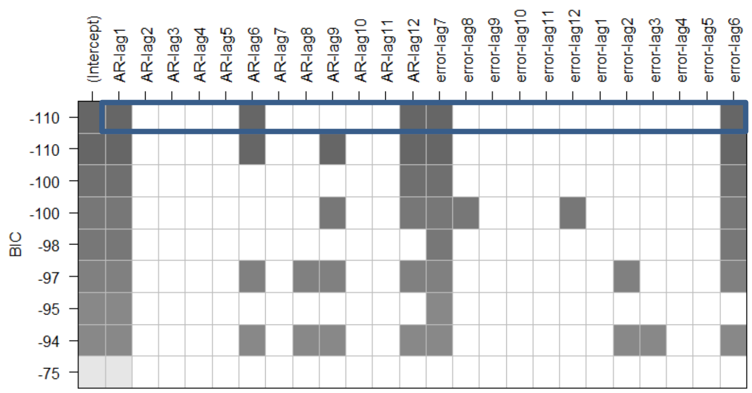
Datos11: Usar el renglón 5 señalado con rectángulo azul sobre el gráfico e incluir ϕ_3, ϕ_4, ϕ_5



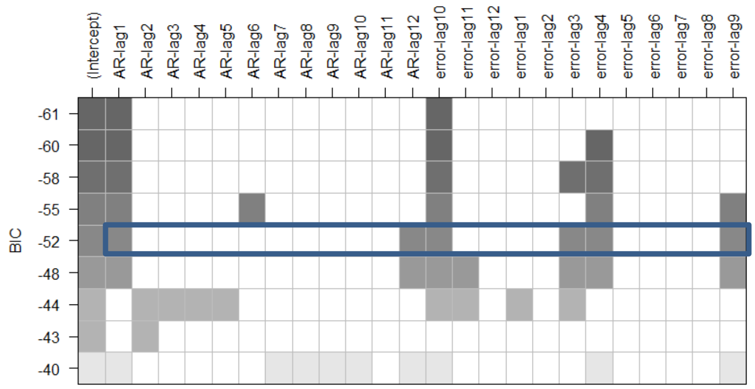
Datos12: Usar el renglón 3 señalado con rectángulo azul sobre el gráfico e incluir ϕ_3, ϕ_4, ϕ_5



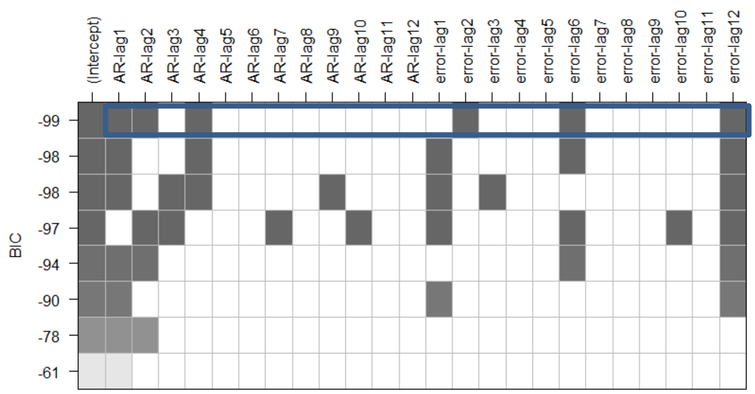
Datos13: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico e incluir $\phi_2, \phi_3, \theta_{12}$



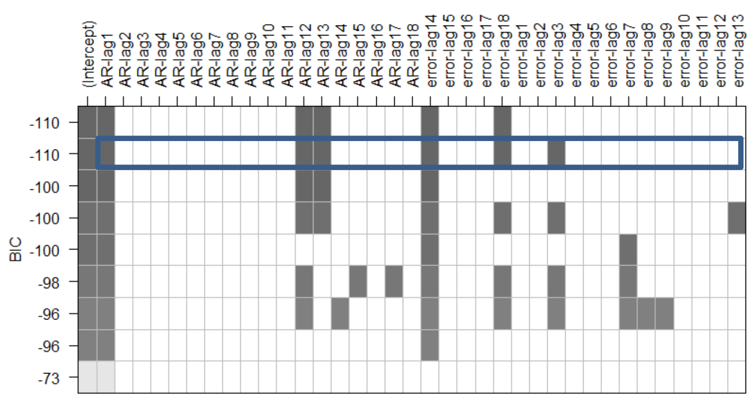
Datos14: Usar el renglón 5 señalado con rectángulo azul sobre el gráfico e incluir a ϕ_2



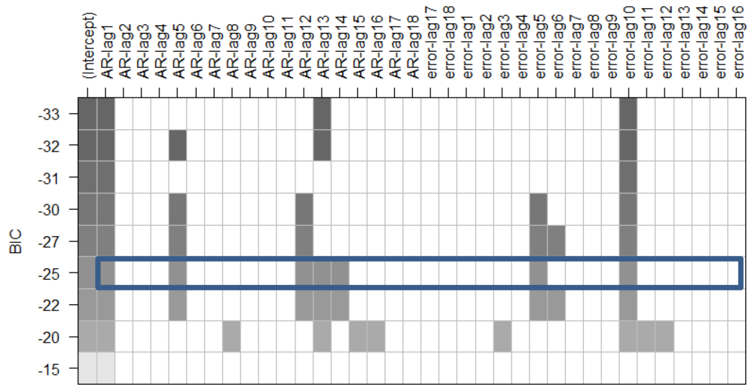
Datos15: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico



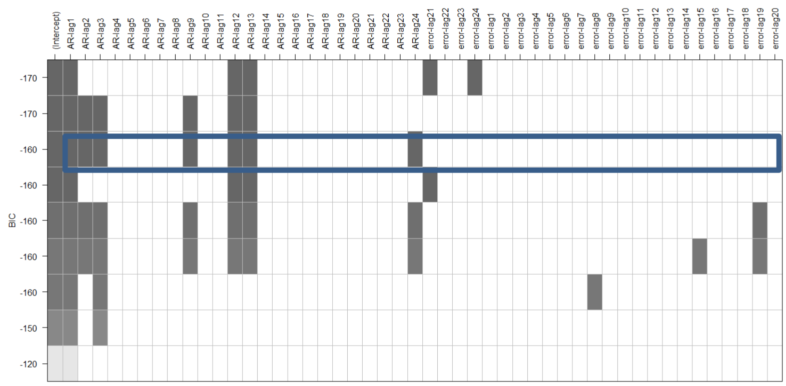
Datos16: Usar el renglón 2 señalado con rectángulo azul sobre el gráfico e incluir $\phi_2, \phi_5, \theta_{20}$



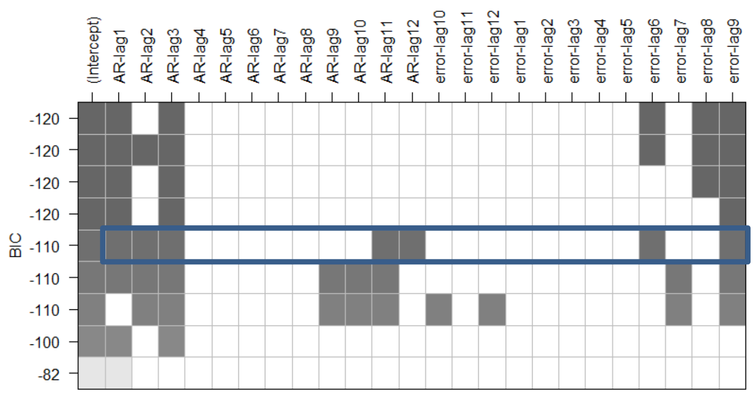
Datos17: Usar el renglón 6 señalado con rectángulo azul sobre el gráfico



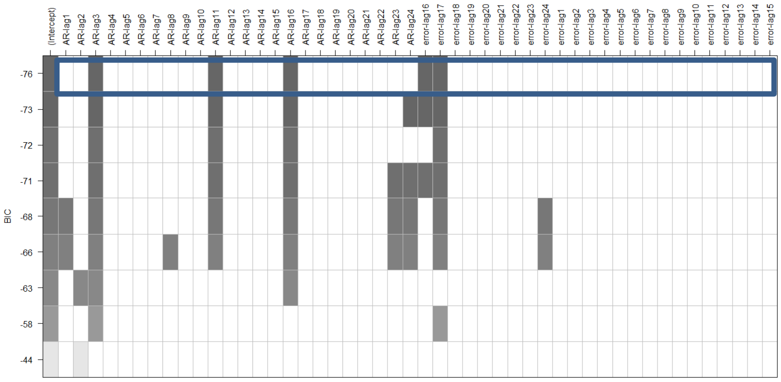
Datos18: Usar el renglón 3 señalado con rectángulo azul sobre el gráfico



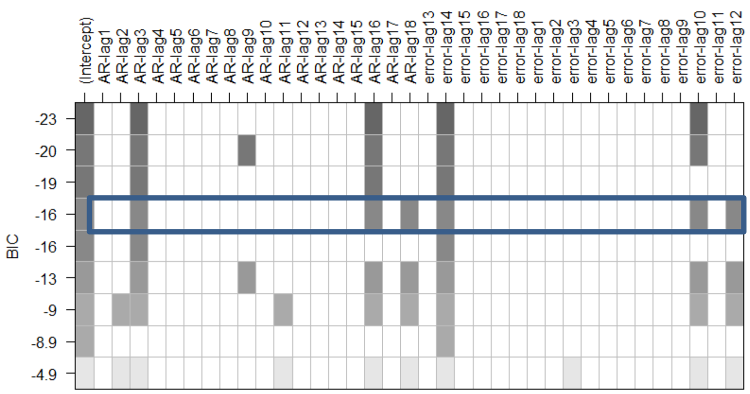
Datos19: Usar el renglón 5 señalado con rectángulo azul sobre el gráfico e incluir ϕ_{13}



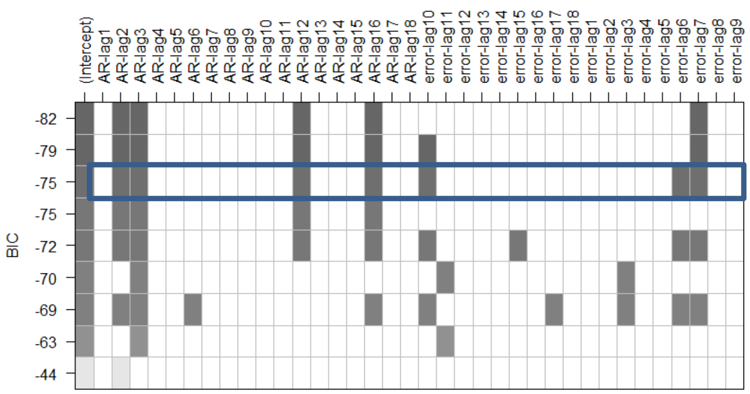
Datos20: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico e incluir ϕ_1, ϕ_2



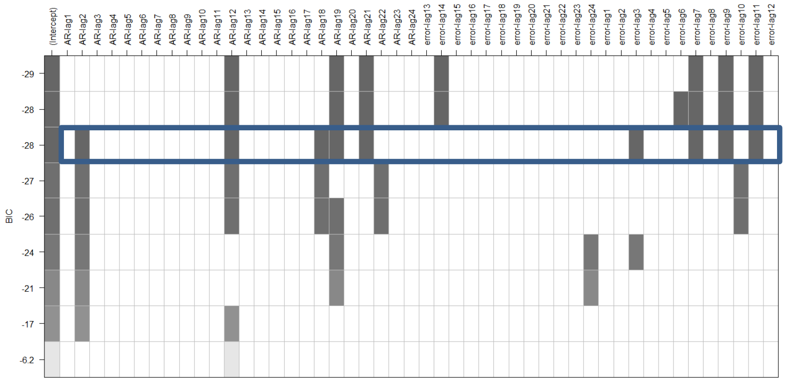
Datos21: Usar el renglón 4 señalado con rectángulo azul sobre el gráfico e incluir $\phi_2, \phi_4, \phi_8, \phi_{19}$



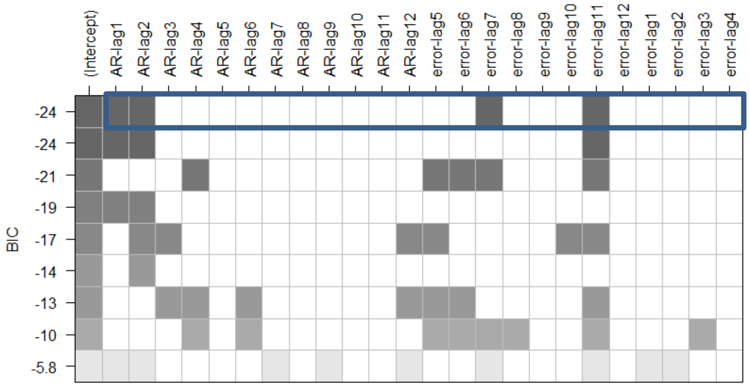
Datos22: Usar el renglón 3 señalado con rectángulo azul sobre el gráfico e incluir $\phi_1, \theta_8, \theta_{16}$



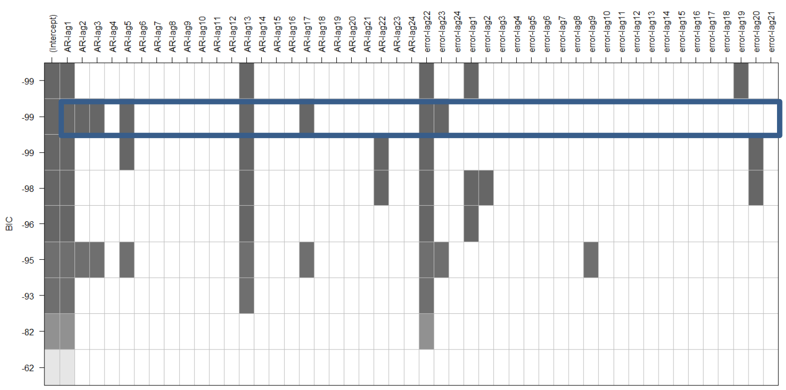
Datos23: Usar el renglón 3 señalado con rectángulo azul sobre el gráfico e incluir ϕ_{15}, θ_8



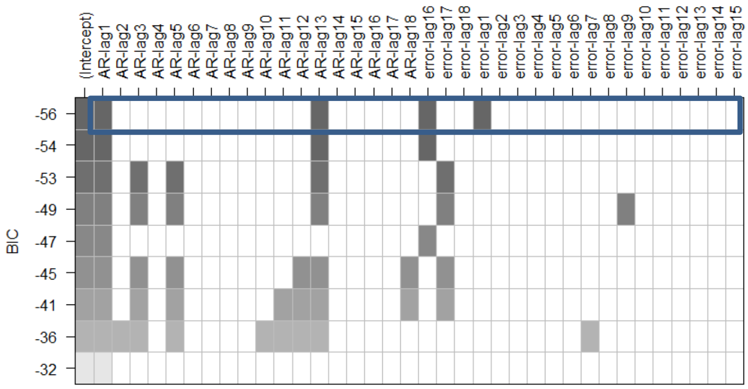
Datos25: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico e incluir ϕ_3



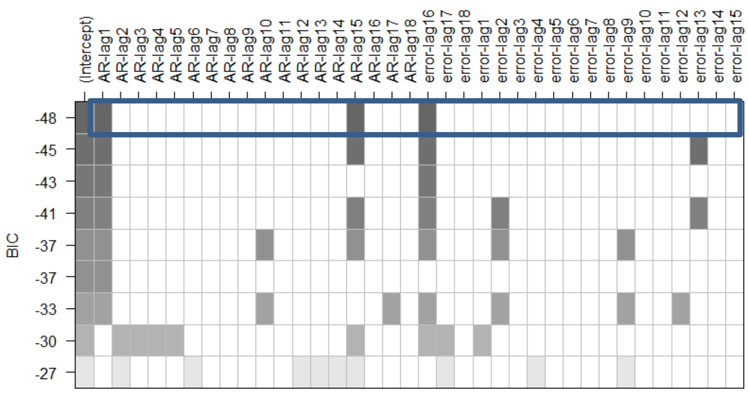
Datos28: Usar el renglón 2 señalado con rectángulo azul sobre el gráfico



Datos29: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico



Datos31: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico e incluir ϕ_3



Datos32: Usar el renglón 1 señalado con rectángulo azul sobre el gráfico e incluir θ_1