



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ciencias Naturales y Matemáticas

"DISEÑO E IMPLEMENTACIÓN DE UN ASISTENTE
PARA EL ANÁLISIS DE SERIES TEMPORALES ESTACIONARIAS
A TRAVÉS DE PAQUETES DE CÓDIGO ABIERTO
PARA EL DESARROLLO DE APLICACIONES WEB REACTIVAS"

Informe de

MATERIA INTEGRADORA

Previa a la obtención del Título de

INGENIERO EN ESTADÍSTICA INFORMÁTICA

Presentado por

RÓMULO FERNANDO TENESACA TIMANÁ

Guayaquil – Ecuador

2016



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ciencias Naturales y Matemáticas

Ingeniería en Estadística Informática

INFORME DEL PROFESOR DE LA MATERIA INTEGRADORA

Habiendo sido nombrada PROFESOR DE LA MATERIA INTEGRADORA del señor,

RÓMULO FERNANDO TENESACA TIMANÁ

con el tema del proyecto integrador “DISEÑO E IMPLEMENTACIÓN DE UN ASISTENTE PARA EL ANÁLISIS DE SERIES TEMPORALES ESTACIONARIAS A TRAVÉS DE PAQUETES DE CÓDIGO ABIERTO PARA EL DESARROLLO DE APLICACIONES WEB REACTIVAS”, previa a la obtención del título de INGENIERO EN ESTADÍSTICA INFORMÁTICA, me permito informar que he leído el contenido del proyecto integrador y recibido los comentarios favorables del profesor colaborador, *Ph.D. Francisco Xavier Vera Alcívar*; luego de lo cual indico que estoy de acuerdo en que el mismo se lo ha desarrollado conforme a los lineamientos de la Unidad de Titulación Especial de la ESPOL.

M.Sc. Sonnia Paola Reyes Ramos
PROFESOR DE LA MATERIA INTEGRADORA

Guayaquil, 12 de febrero del 2016

AGRADECIMIENTO

A Dios.

A mi Familia,

por su invalorable apoyo y respaldo en todo momento.

Al Ph.D. Francisco Vera,

por la oportunidad de colaborar con su iniciativa académica.

A la M.Sc. Sonnia Reyes,

por su asesoría y gestión académica en la materia integradora.

A los Profesores, Amigos y Compañeros de la ESPOL,

con quienes compartí el camino.

DEDICATORIA

*Con cariño y aprecio a mi Padre,
por su confianza en permitirme elegir el camino, y
a la ESPOL,
por constituir el ambiente idóneo donde transitarlo.*

DECLARACION EXPRESA

“La responsabilidad del contenido de este trabajo de titulación, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral.”

(Reglamento de Graduación de la ESPOL)

Rómulo Fernando Tenesaca Timaná

RESUMEN

La Facultad de Ciencias Naturales y Matemáticas de la ESPOL a través de los proyectos de Materia Integradora tiene la iniciativa de desarrollar aplicaciones, como prototipos para diferentes módulos de un sistema integrado de análisis estadístico, con el objeto de plantear a las autoridades de la Institución la importancia de contar con soluciones de este nivel que contribuyan a fomentar la investigación y las actividades de docencia.

El presente proyecto integrador se desarrolló utilizando un lenguaje y entorno de programación orientados al análisis estadístico, junto con el soporte de paquetes de código abierto, con el propósito de diseñar e implementar AMIRA, una aplicación dirigida a la comunidad académica con formación estadística y conocimientos en el campo del análisis de series temporales para que a través de este asistente computacional se facilite la selección de modelos autorregresivos y de media móvil, abstrayendo los detalles estadísticos y de programación del análisis específico para pronosticar situaciones de incertidumbre. AMIRA, es un prototipo de aplicación web reactiva concebido bajo los criterios teóricos planteados por Box y Jenkins¹ como metodología para identificar, estimar y verificar modelos que se ajustan a las series temporales estacionarias.

En el informe de Materia Integradora, el primer capítulo describe el planteamiento del problema señalando los objetivos general y específicos; además, de la justificación del proyecto AMIRA. El segundo capítulo sintetiza el esquema de desarrollo y la aplicación práctica del conocimiento científico en el desarrollo de aplicaciones; así como el marco conceptual para establecer el alcance de la aplicación AMIRA. El tercer capítulo detalla el marco teórico relacionado con el análisis de series temporales estacionarias de acuerdo con el alcance establecido. El capítulo cuatro explica como el paradigma orientado al flujo de datos y a la propagación de cambios de los datos en dicho flujo facilita el diseño e implementación de aplicaciones web reactivas como AMIRA para el análisis estadístico.

El desarrollo de AMIRA, como proyecto de Materia Integradora, a través de lenguajes de programación como R⁵, entornos de desarrollo como Rstudio⁶ y complementos como Shiny⁷, hace posible que la comunidad académica se involucre en la ampliación del alcance original, modificando el código fuente, para soportar nuevos requerimientos funcionales e implementar nuevas características basados en un flujo de trabajo estructurado y en objetos reactivos para carga y manipulación de datos; además de cumplir con los principales resultados del programa de estudios de la carrera de Ingeniería en Estadística Informática.

CONTENIDO

PORTADA.....	i
INFORME DEL PROFESOR.....	ii
AGRADECIMIENTO	iii
DEDICATORIA	iv
DECLARACION EXPRESA	v
RESUMEN.....	vi
CONTENIDO	viii
ÍNDICE DE GRÁFICOS	x
ÍNDICE DE TABLAS	xi
ÍNDICE DE ILUSTRACIONES.....	xi
CAPÍTULO I.....	1
1. PLANTEAMIENTO DEL PROBLEMA	1
1.1. Definición del Problema	1
1.2. Objetivo General.....	1
1.3. Objetivos Específicos	2
1.4. Justificación	3
CAPÍTULO II.....	5
2. MARCO TEÓRICO.....	5
2.1. Estado del Arte.....	5
2.2. Marco Conceptual.....	8
2.2.1. Series Temporales	8
2.2.2. Modelos de Series Temporales.....	9
2.2.3. Series Temporales Univariadas	9
2.2.4. Series Temporales Univariadas Estacionarias.....	10
CAPÍTULO III.....	11
3. SERIES TEMPORALES ESTACIONARIAS	11
3.1. Proceso Estocástico.....	11
3.2. Proceso Estocástico Estacionario.....	12
3.3. Función de Autocovarianza	13
3.4. Función de Autocorrelación.....	14
3.5. Función de Autocorrelación Parcial	15
3.6. Prueba de Ljung-Box	16
3.7. Modelos Estacionarios	17
3.7.1. Modelos Autorregresivos	17
3.7.1.1. Modelo Autorregresivo de Primer Orden.....	18
3.7.1.2. Características del Modelo $AR(1)$	21
3.7.1.3. Modelo Autorregresivo de Segundo Orden.....	24
3.7.1.4. Características del Modelo $AR(2)$	25
3.7.2. Modelos de Medias Móviles	27
3.7.2.1. Modelo de Media Móvil de Primer Orden	28
3.7.2.2. Características del Modelo $MA(1)$	31

3.7.2.3. Modelo de Media Móvil de Segundo Orden	34
3.7.2.4. Características del Modelo $MA(2)$	34
3.7.3. Modelos Autorregresivos de Medias Móviles.....	36
3.7.3.1. Modelo Autorregresivo de Media Móvil de Primer Orden	37
3.7.3.2. Características del Modelo $ARMA(1,1)$	38
CAPÍTULO IV	41
4. AMIRA	41
4.1. Definición	41
4.1.1. Programación Reactiva.....	41
4.2. Diseño	42
4.2.1. Casos de Uso	43
4.2.2. Actividades	45
4.2.3. Artefactos	46
4.2.4. Componentes	47
4.2.5. Paquetes	48
4.2.6. Despliegue	49
4.3. Implementación	51
4.3.1. Plataforma como un Servicio	52
4.4. Ejemplos de Aplicación	54
4.4.1. Ejemplo 1	54
4.4.2. Ejemplo 2	63
CONCLUSIONES Y RECOMENDACIONES.....	72
BIBLIOGRAFÍA	75
ANEXOS	77
A. FUNCIÓN DE AUTOCORRELACIÓN PARCIAL ¹²	77
B. CASOS DE USO	79
C. PLATAFORMA COMO UN SERVICIO	83
D. CÓDIGO FUENTE.....	84

ÍNDICE DE GRÁFICOS

Gráfico 2.1: Serie estacionaria	10
Gráfico 3.1: Función de Autocorrelación	15
Gráfico 3.2: Función de Autocorrelación Parcial	16
Gráfico 3.3: Modelo $AR(1)$ ACF y PACF Teóricos	23
Gráfico 3.4: Modelo $AR(1)$ $\phi = +0.7$	23
Gráfico 3.5: Modelo $AR(1)$ $\phi = -0.7$	24
Gráfico 3.6: Modelo $AR(2)$ ACF y PACF Teóricos	26
Gráfico 3.7: Modelo $AR(2)$ $\phi_1 = +0.7$ $\phi_2 = +0.2$	26
Gráfico 3.8: Modelo $AR(2)$ $\phi_1 = -0.7$ $\phi_2 = +0.2$	27
Gráfico 3.9: Modelo $MA(1)$ ACF y PACF Teóricos	32
Gráfico 3.10: Modelo $MA(1)$ $\theta = +0.7$	33
Gráfico 3.11: Modelo $MA(1)$ $\theta = -0.7$	33
Gráfico 3.12: Modelo $MA(2)$ ACF y PACF Teóricos	35
Gráfico 3.13: Modelo $MA(2)$ $\theta_1 = +0.7$ $\theta_2 = +0.2$	35
Gráfico 3.14: Modelo $MA(2)$ $\theta_1 = -0.7$ $\theta_2 = +0.2$	36
Gráfico 3.15: Modelo $ARMA(1,1)$ ACF y PACF Teóricos	39
Gráfico 3.16: Modelo $ARMA(1,1)$ $\phi = +0.7$ $\theta = +0.2$	40
Gráfico 3.17: Modelo $ARMA(1,1)$ $\phi = -0.7$ $\theta = +0.2$	40
Gráfico 4.1: Ejemplo 1 Serie Temporal	55
Gráfico 4.2: Ejemplo 1 Función de Autocorrelación	56
Gráfico 4.3: Ejemplo 1 AMIRA Serie Temporal	57
Gráfico 4.4: Ejemplo 1 AMIRA ACF	58
Gráfico 4.5: Ejemplo 1 AMIRA PACF	58
Gráfico 4.6: Ejemplo 1 AMIRA Identificación	59
Gráfico 4.7: Ejemplo 1 AMIRA Estimación	60
Gráfico 4.8: Ejemplo 1 AMIRA Análisis	61
Gráfico 4.9: Ejemplo 1 AMIRA Predicción	62
Gráfico 4.10: Ejemplo 2 AMIRA Serie Temporal	63
Gráfico 4.11: Ejemplo 2 AMIRA ACF	64
Gráfico 4.12: Ejemplo 2 AMIRA Transformación	64
Gráfico 4.13: Ejemplo 2 AMIRA ACF [d=1]	65
Gráfico 4.14: Ejemplo 2 AMIRA PACF [d=1]	66
Gráfico 4.15: Ejemplo 12 AMIRA Identificación	66
Gráfico 4.16: Ejemplo 2 AMIRA Identificación	67
Gráfico 4.17: Ejemplo 2 AMIRA Identificación	68
Gráfico 4.18: Ejemplo 2 AMIRA Estimación	69
Gráfico 4.19: Ejemplo 2 AMIRA Análisis	70
Gráfico 4.20: Ejemplo 2 AMIRA Predicción	71

ÍNDICE DE TABLAS

Tabla 3.1: Prueba de Ljung-Box	16
Tabla 4.1: Casos de uso	43
Tabla 4.2: Casos de uso Ingresar a la aplicación	44
Tabla 4.3: Ejemplo 1 Serie Temporal	54
Tabla 4.4: Ejemplo 1 Raíces	55
Tabla 4.5: Ejemplo 1 Función de Autocorrelación [ACF & PACF]	56

ÍNDICE DE ILUSTRACIONES

Ilustración 3.1: Origen de una serie temporal	11
Ilustración 3.2: Esquema de correlación <i>AR</i>	18
Ilustración 3.3: Esquema de correlación <i>MA</i>	28
Ilustración 3.4: Esquema de correlación <i>ARMA</i>	37
Ilustración 4.1: Esquema de programación reactiva	42
Ilustración 4.2: Diseño UML Casos de uso	44
Ilustración 4.3: Diseño UML Actividades	45
Ilustración 4.4: Diseño UML Artefactos	46
Ilustración 4.5: Diseño UML Componentes	47
Ilustración 4.6: Diseño UML Paquetes	48
Ilustración 4.7: Diseño UML Despliegue Aplicación y Administración	49
Ilustración 4.8: Diseño UML Despliegue Servicio	49
Ilustración 4.9: Diseño UML Despliegue Cliente y Programador	50
Ilustración 4.10: Implementación	51
Ilustración 4.11: Implementación PaaS	52
Ilustración 4.12: Implementación PaaS Distribución	53

CAPÍTULO I

1. PLANTEAMIENTO DEL PROBLEMA

1.1. Definición del Problema

Utilizando un lenguaje y entorno de programación desarrollados para el análisis estadístico junto con el soporte de paquetes de código abierto se diseña e implementa una aplicación web reactiva dirigida a la comunidad académica del País y el mundo con formación estadística y conocimientos en el campo del análisis de series temporales para que a través de un asistente se facilite la selección de modelos autorregresivos y de media móvil que mejor se ajusten a los datos, abstrayendo los detalles estadísticos y de programación del análisis específico para pronosticar situaciones de incertidumbre.

1.2. Objetivo General

Proveer a la comunidad académica del País y al mundo con formación estadística y conocimientos en el campo del análisis de series temporales de una aplicación web reactiva que los asista en el proceso de selección de modelos autorregresivos y de media móvil.

1.3. Objetivos Específicos

- ✓ Diseñar un asistente para el análisis de series temporales estacionarias utilizando un lenguaje de programación y entorno de desarrollo de código abierto.
- ✓ Implementar una aplicación web reactiva para el tratamiento de series temporales estacionarias como prototipo para un sistema integrado de análisis estadístico.
- ✓ Aplicar los criterios desarrollados por Box y Jenkins¹ como metodología para identificar, estimar y verificar modelos que se ajustan a las series temporales estacionarias.
- ✓ Abstraer los detalles estadísticos y de programación del análisis específico para pronosticar situaciones de incertidumbre.
- ✓ Cumplir con los principales resultados del programa de estudios de la carrera de Ingeniería en Estadística Informática acorde con la normativa vigente.

1.4. Justificación

La Facultad de Ciencias Naturales y Matemáticas de la ESPOL provee servicios de consultoría para el sector público y privado, así como para consumo interno. A través de su planta docente y con la colaboración de sus estudiantes ha desarrollado diferentes proyectos en las áreas de conocimiento de su competencia. En el área de matemáticas, ofrece servicios de consultoría en: optimización, simulación, muestreo, administración de operaciones, control de la producción, análisis actuariales, series temporales, mejoramiento y control de la calidad, entre otros.

Para fomentar la investigación en esta línea la FCNM ha mantenido una constante supervisión de los trabajos que se realizan en todas las modalidades de graduación lo que ha permitido desarrollar iniciativas en el campo de la estadística e informática generando soluciones específicas para el análisis de datos.

En la actualidad la ESPOL, como centro de estudios superior enfocado en la investigación, debe contar con un sistema integrado de análisis estadístico que permita a la comunidad politécnica acceder a un conjunto de aplicaciones de código abierto que facilite esta labor. En virtud de lo expuesto, la FCNM a través del Departamento de Matemáticas tiene la iniciativa de desarrollar

aplicaciones como prototipos para diferentes módulos de un sistema integrado de análisis estadístico, con el objeto de plantear a las autoridades de la Institución la importancia de contar con soluciones de este nivel que contribuyan a fomentar la investigación y las actividades de docencia.

CAPÍTULO II

2. MARCO TEÓRICO

2.1. Estado del Arte

La arquitectura asociada con el diseño de software es uno de los conceptos en programación más importantes desde la década del setenta, periodo donde se destacan las observaciones de Edsger Dijkstra en las cuales se refuerza la importancia del establecimiento de una estructuración correcta de los sistemas antes de proceder a programar². Como parte de la ingeniería de software³, el definir un esquema de desarrollo acorde con los objetivos planteados da paso a la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora; así como, a la documentación asociada requerida para implementar, operar y mantenerlos.

Con la estandarización de la programación modular como una arquitectura que estructura a los programas en grupos funcionales muy acoplados, se establece un punto de partida para el perfeccionamiento de la metodología que permite identificar patrones y abstraer del usuario las especificaciones de desarrollo y con esto establecer de manera sólida la arquitectura de dos capas (cliente/servidor).

Cuando se desagrega la arquitectura de dos capas en sus niveles funcionales más representativos se da soporte a la arquitectura multinivel o de n-capas; siendo la más frecuente la arquitectura de 3 capas que permite modelar las capas de presentación, de negocio y de datos por separado a través de especificaciones de análisis y diseño más complejos para implementar diversas aplicaciones que requieren un nivel superior de abstracción y el desarrollo de interfaces de usuario cada vez más intuitivas.

El continuo crecimiento de las fuentes u orígenes de datos y la necesidad casi imperativa de información oportuna y precisa ha dado paso al planteamiento de nuevos y cada vez más evolucionados paradigmas de programación que contribuyen al análisis de grandes volúmenes de datos relevantes para el usuario y lo abstraen completamente de las especificaciones de desarrollo de los programas.

A través de un enfoque sistemático, disciplinado y cuantificable, la ingeniería de software permite la integración de conceptos y modelos matemáticos que se estudian y aplican en diversas ramas de las ciencias⁴. La Estadística, ciencia transversal por excelencia, se ha beneficiado con la evolución de la ingeniería de software y la arquitectura de programación.

El auge del software libre y de código abierto ha permitido que lenguajes de programación como R⁵, entornos de desarrollo como Rstudio⁶ y complementos como Shiny⁷ se posicionen a todo nivel al momento de desarrollar aplicaciones específicas para el análisis de datos.

Shiny es un paquete de código abierto para el lenguaje de programación estadística R que en conjunto establecen como base una arquitectura cliente/servidor soportada por componentes que se interpretan a nivel del cliente como websockets⁸ y Bootstrap⁹; así como, un conjunto de complementos que a partir del paradigma de programación reactiva se ejecutan a nivel del servidor en base al flujo y a los eventos que se definen sobre los datos.

La potencia para el análisis estadístico y la visualización de datos del lenguaje de programación R permite el despliegue de interfaces de usuario funcionales y reactivas utilizando RStudio como entorno de desarrollo y Shiny como complemento¹⁰. Shiny a través de R permite de manera rápida y sencilla compartir análisis y gráficos a través de la red con los cuales el usuario puede interactuar y tomar decisiones.

2.2. Marco Conceptual

Al considerar como objetivo del presente trabajo de titulación la abstracción de los detalles estadísticos y de programación para el usuario del *Asistente para el Análisis de Series Temporales Estacionarias a través de Paquetes de Código Abierto para el Desarrollo de Aplicaciones Web Reactivas*, es importante entonces establecer el alcance del diseño y la implementación del prototipo alrededor de un marco conceptual apropiado para la valoración del informe y la interpretación de los resultados.

2.2.1. Series Temporales

De forma general, el análisis de series temporales tiene como propósito realizar inferencias sobre el comportamiento de diferentes características a través del tiempo utilizando diversos criterios estadísticos para identificar, estimar y verificar modelos que se ajusten a los datos; por tanto, el orden de las observaciones que componen la serie temporal es importante y su modificación implica la ocurrencia de una serie temporal distinta.

2.2.2. Modelos de Series Temporales

Los modelos de series temporales se construyen sobre la premisa que la serie tiene una historia estadística recurrente y particular que puede ser modelada para pronosticar el comportamiento de variables asociadas a diferentes características. En este contexto, se identifican dos tipos de modelos: series temporales univariadas y series temporales multivariadas.

El presente trabajo de titulación se enfoca en el análisis de series temporales univariadas.

2.2.3. Series Temporales Univariadas

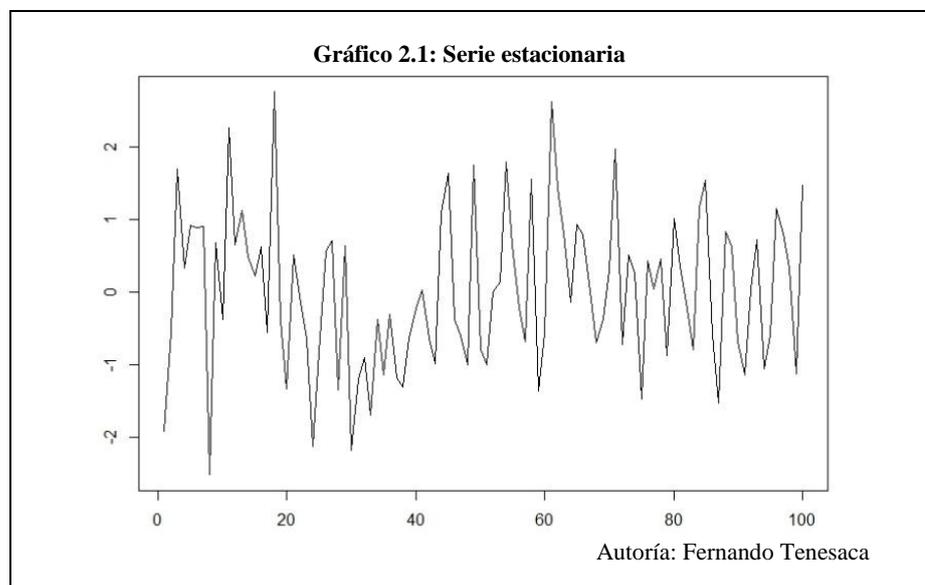
Una serie temporal univariada ($Y_t: t = 1, 2, 3, \dots, n$) es una secuencia de observaciones ordenadas y equidistantes cronológicamente que describen una sola característica que evoluciona a través del tiempo y que constituye la variable de interés; donde Y_t representa la medida en el tiempo t y n el número total de observaciones en la serie.

Las variaciones de una serie temporal univariada se modelan como una función de términos autorregresivos que comprenden los valores históricos de la variable y en términos de medias móviles que asocian innovaciones contemporáneas e históricas.

De manera específica, el presente trabajo de titulación se centra en el análisis de series temporales univariadas estacionarias.

2.2.4. Series Temporales Univariadas Estacionarias

Una serie temporal univariada (de aquí en adelante “serie”) es estacionaria cuando la media y varianza del conjunto de observaciones son constantes en el tiempo. Esto se refleja gráficamente en que los valores de la serie tienden a oscilar alrededor de una media constante y la variabilidad con respecto a esta medida también permanece constante en el tiempo.



CAPÍTULO III

3. SERIES TEMPORALES ESTACIONARIAS

3.1. Proceso Estocástico

Un proceso estocástico¹¹ es una familia o colección de variables aleatorias $\{Y_t: t \in T\}$ indexadas por un parámetro t que toma valores de un conjunto de índices T ; por tanto, cuando se interpreta el parámetro t como *tiempo*, el proceso estocástico se denomina *serie temporal*.

Ilustración 3.1: Origen de una serie temporal

Proceso estocástico $\dots, Y_{-1}, Y_0, \underbrace{Y_1, Y_2, \dots, Y_n}_{Y_1, Y_2, \dots, Y_n}, Y_{n+1}, \dots$
Serie temporal

Autoría: Fernando Tenesaca

Una serie temporal (Y_t) representa una realización muestral de un proceso estocástico que se observa únicamente para un número finito de periodos $t = 1, 2, \dots, n$; y, describe como una secuencia de datos evolucionaria y están relacionadas entre sí siguiendo una función de distribución.

3.2. Proceso Estocástico Estacionario

Un proceso estocástico¹² es estacionario cuando las propiedades estadísticas de cualquier secuencia finita $Y_{t_1}, Y_{t_2}, \dots, Y_{t_n}$ ($n \geq 1$) de componentes (Y_t) son semejantes a las de la secuencia $Y_{t_1+k}, Y_{t_2+k}, \dots, Y_{t_n+k}$ para cualquier número entero $k = \pm 1, \pm 2, \dots$

El análisis de una serie temporal tiene como fin realizar inferencias sobre el comportamiento de una característica observable en el tiempo de forma que si se quiere obtener métodos de predicción consistentes se requiere que el proceso estocástico sea estacionario lo que significa que su estructura probabilística sea estable en un periodo específico.

Una serie temporal como ocurrencia muestral de un proceso estocástico estacionario no presenta tendencia y las observaciones que la componen se disponen alrededor de la media; por tanto, cualquier variación en el tiempo no presenta efectos a largo plazo.

El concepto de estacionariedad de la serie puede ser estricto (fuerte) o amplio (débil); sin embargo, el sentido estricto es demasiado riguroso para ser comprobado en la práctica lo que implica que la distribución de Y_t y todos los momentos sean independientes del tiempo. Al flexibilizar los supuestos, se

limita la independencia al primer y segundo momento; es decir, a las medias, varianzas y covarianzas, lo que establece las condiciones para el sentido amplio de estacionariedad.

3.3. Función de Autocovarianza

La función de autocovarianza de un proceso estocástico estacionario (Y_t) es una función de k que representa la covarianza entre cualquier par de componentes (Y_t) separados entre sí por un intervalo temporal $k > 0$. Se denota por

$$\gamma_k = Cov[Y_t, Y_{t+k}] = E[(Y_t - \mu)(Y_{t+k} - \mu)] \quad , \quad k = 0, 1, 2, 3, \dots$$

La función de autocovarianza tiene dos características fundamentales, por un lado incluye la varianza del proceso para $k = 0$ tal que

$$\gamma_0 = Cov[Y_t, Y_t] = Var[Y_t];$$

y, por otra parte es una función simétrica, esto implica que

$$\gamma_k = E[(Y_t - \mu)(Y_{t+k} - \mu)] = E[(Y_t - \mu)(Y_{t-k} - \mu)] = \gamma_{-k}.$$

Por tanto, la función de covarianza recoge toda la información sobre la estructura lineal del proceso pero depende de las unidades de medida de la variable, por lo que, en general, se suele utilizar la función de autocorrelación y sus factores a través de la función de autocorrelación parcial.

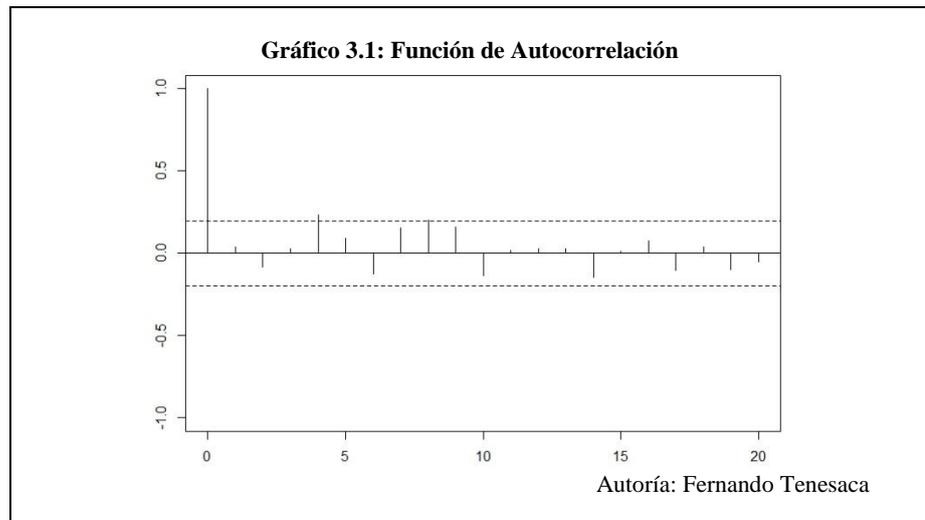
3.4. Función de Autocorrelación

La función de autocorrelación (ACF) de un proceso estocástico estacionario (Y_t) es una función de k que representa el coeficiente de correlación lineal entre cualquier par de componentes (Y_t) separados entre sí por un intervalo temporal $k > 0$. Se denota por

$$\rho_k = \text{Corr}[Y_t, Y_{t+k}] = \frac{\text{Cov}[Y_t, Y_{t+k}]}{\sqrt{\text{Var}(Y_t)\text{Var}(Y_{t+k})}} = \frac{\text{Cov}[Y_t, Y_{t+k}]}{\text{Var}(Y_t)} = \frac{\gamma_k}{\gamma_0}$$

$$\rho_k = \frac{\gamma_k}{\gamma_0}, \quad k = 0, 1, 2, 3, \dots; \quad \rho_0 = 1$$

La función de autocorrelación es una función simétrica ($\rho_k = \rho_{-k}$) y por ser un coeficiente de correlación no depende de las unidades de medida de la variable, es decir, que para todo k intervalo temporal $|\rho_k| \leq 1$.

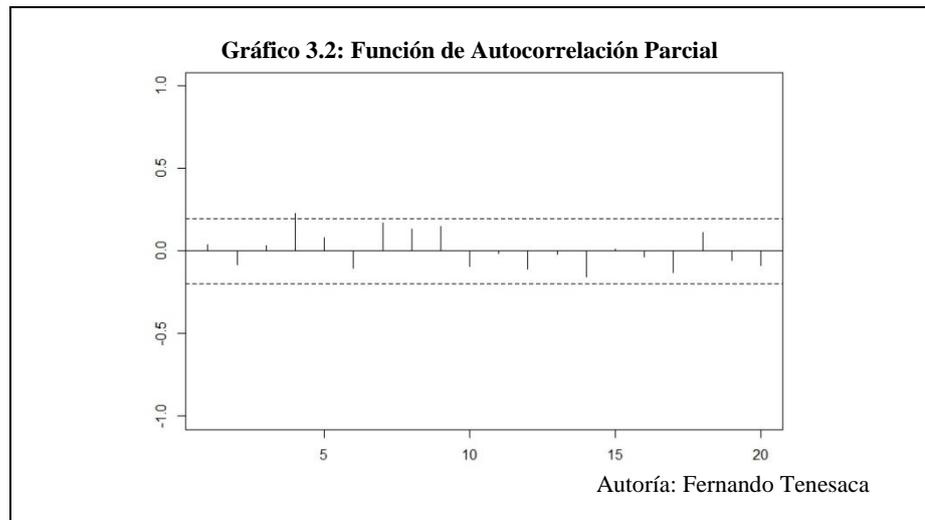


3.5. Función de Autocorrelación Parcial

La función de autocorrelación parcial (PACF) de un proceso estocástico estacionario (Y_t) es una función de k que representa el grado de asociación lineal entre un par de componentes (Y_t) separados entre sí por un intervalo temporal $k \geq 1$. Se denota por

$$\phi_{kk} = \frac{|A_k|}{|B_k|} , \quad k = 1, 2, 3, \dots ; \quad \phi_{11} = \rho_1$$

Los coeficientes de autocorrelación parcial son funciones de los coeficientes de autocorrelación de un proceso estrictamente estacionario. Una descripción detallada sobre la definición de la función de autocorrelación parcial se presenta adjunta en los anexos (ver Anexo A).



3.6. Prueba de Ljung-Box

La prueba de Ljung-Box¹³ permite probar de forma conjunta que todos los coeficientes de autocorrelación (ρ_k) de un proceso estocástico estacionario (Y_t) son simultáneamente iguales a cero. En general, esta prueba estadística plantea

Tabla 3.1: Prueba de Ljung-Box

ELEMENTOS	EXPRESIÓN
Hipótesis nula	$H_0: \rho_k = 0$
Hipótesis alterna	$H_1: \rho_k \neq 0$
Estadístico de prueba	$Q = n(n + 2) \sum_{k=1}^m \left(\frac{r_k^2}{n - k} \right)$
Región crítica	$Q > \chi_{1-\alpha, h}^2$

Autoría: Fernando Tenesaca

En la validación del modelo asociado a una serie temporal estacionaria, el estadístico de prueba Ljung-Box, conocido también como estadístico Q , sirve para evaluar los supuestos después de ajustar un modelo a una serie temporal estacionaria para asegurar que los residuos en un periodo de tiempo específico sean aleatorios e independientes.

3.7. Modelos Estacionarios

3.7.1. Modelos Autorregresivos

Un proceso estocástico estacionario (Y_t) sigue un modelo autorregresivo de orden (p) , o $AR(p)$, si y solo si

$$Y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t \quad , \quad \varepsilon_t \sim NIID(0, \sigma^2)$$

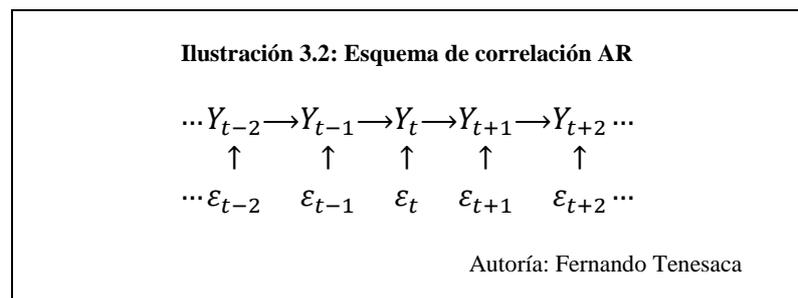
Este modelo establece que la serie estacionaria (Y_t) puede explicarse en función de p valores históricos donde éste parámetro determina el número de rezagos necesarios para pronosticar un valor actual.

3.7.1.1. Modelo Autorregresivo de Primer Orden

El modelo $AR(1)$ define que la variable Y_t esta determinada únicamente por su valor pasado, Y_{t-1} , y la perturbación contemporánea, ε_t . Se denota por

$$Y_t = \phi_1 y_{t-1} + \varepsilon_t \quad , \quad t = 1, 2, \dots \quad , \quad \varepsilon_t \sim NIID(0, \sigma^2)$$

El esquema de correlación en el tiempo del modelo autorregresivo se ilustra de la siguiente forma



Por consiguiente, la perturbación ε_t condiciona al sistema en el tiempo t influyendo en Y_t y en las observaciones futuras, pero no en el registro histórico.

Comprobar que el modelo $AR(1)$ es estacionario para todo valor ϕ requiere verificar las condiciones de estacionariedad.

Modelo estacionario en media

Para que el modelo $AR(1)$ sea estacionario, la media debe ser constante y finita, así se tiene

$$\begin{aligned}
 E[Y_t] &= E[\phi Y_{t-1} + \varepsilon_t] \\
 E[Y_t] &= E[\phi Y_{t-1}] + E[\varepsilon_t] \\
 E[Y_t] &= \phi E[Y_{t-1}] + E[\varepsilon_t] \\
 E[Y_t] &= \phi E[Y_{t-1}] + (0) \\
 E[Y_t] - \phi E[Y_{t-1}] &= 0 \\
 E[Y_t] - \phi E[Y_t] &= 0 \\
 (1 - \phi)E[Y_t] &= 0 \\
 E[Y_t] &= 0/(1 - \phi) \\
 E[Y_t] &= 0
 \end{aligned}$$

Por consiguiente, para que el modelo sea estacionario en media se requiere que $\phi \neq 1$.

Modelo estacionario en varianza

Para que el modelo $AR(1)$ sea estacionario, la varianza debe ser constante y finita, así se tiene

$$\begin{aligned}
 V[Y_t] &= E[(Y_t - E[Y_t])^2] \\
 &= E[(\phi Y_{t-1} + \varepsilon_t - 0)^2] \\
 &= E[(\phi Y_{t-1} + \varepsilon_t)^2] \\
 &= E[\phi^2 Y_{t-1}^2 + 2\phi Y_{t-1} \varepsilon_t + \varepsilon_t^2] \\
 &= \phi^2 E[Y_{t-1}^2] + 2\phi E[Y_{t-1} \varepsilon_t] + E[\varepsilon_t^2] \\
 &= \phi^2 E[Y_{t-1}]^2 + 2\phi E[Y_{t-1} \varepsilon_t] + E[\varepsilon_t]^2
 \end{aligned}$$

Dada la estructura de autocorrelación del proceso estocástico estacionario

$$\begin{aligned} E[Y_{t-1}\varepsilon_t] &= E[(Y_{t-1} - E[Y_{t-1}])(\varepsilon_t - E[\varepsilon_t])] \\ E[Y_{t-1}\varepsilon_t] &= E[(Y_{t-1} - 0)(\varepsilon_t - 0)] \\ E[Y_{t-1}\varepsilon_t] &= Cov[Y_{t-1}\varepsilon_t] \\ E[Y_{t-1}\varepsilon_t] &= 0 \end{aligned}$$

Por el supuesto que el proceso estocástico es estacionario

$$E[Y_{t-1}]^2 = Var[Y_{t-1}] = Var[Y_t] = \gamma_0$$

De donde

$$\begin{aligned} Var[Y_t] &= \phi^2 E[Y_{t-1}]^2 + 2\phi E[Y_{t-1}\varepsilon_t] + E[\varepsilon_t]^2 \\ Var[Y_t] &= \phi^2 Var[Y_t] + 2\phi(0) + \sigma_\varepsilon^2 \\ \gamma_0 &= \phi^2 \gamma_0 + \sigma_\varepsilon^2 \\ \gamma_0 - \phi^2 \gamma_0 &= \sigma_\varepsilon^2 \\ (1 - \phi^2) \gamma_0 &= \sigma_\varepsilon^2 \\ \gamma_0 &= \sigma_\varepsilon^2 / (1 - \phi^2) \end{aligned}$$

Por consiguiente, para que el modelo sea estacionario en varianza se requiere que $|\phi| < 1$.

Acerca de la función de autocovarianza, la autocovarianza de orden k se definen como

$$\begin{aligned} \gamma_k &= Cov[Y_t, Y_{t-k}] \\ &= E[(Y_t - E[Y_t])(Y_{t-k} - E[Y_{t-k}])] \\ &= E[Y_t Y_{t-k}] \\ &= E[(\phi Y_{t-1} + \varepsilon_t) Y_{t-k}] \\ &= E[\phi Y_{t-1} Y_{t-k} + \varepsilon_t Y_{t-k}] \\ &= E[\phi Y_{t-1} Y_{t-k}] + E[\varepsilon_t Y_{t-k}] \\ &= E[\phi Y_{t-1} Y_{t-k}] + (0) \\ &= \phi E[Y_{t-1} Y_{t-k}] \\ &= \phi \gamma_{k-1} \end{aligned}$$

Por tanto

$$\begin{aligned}\gamma_1 &= \phi\gamma_0 \\ \gamma_2 &= \phi\gamma_1 \\ \gamma_3 &= \phi\gamma_2 \\ &\vdots\end{aligned}$$

Por lo antes citado, si el parámetro es $|\phi| < 1$, la varianza es finita y el resto de las autocorrelaciones lo son; además éstas dependen únicamente de los periodos de separación entre las variables y no del tiempo. Se puede concluir entonces que el modelo $AR(1)$ es estacionario.

3.7.1.2. Características del Modelo $AR(1)$

La función de autocovarianza de un modelo $AR(1)$ es

$$\gamma_k = \begin{cases} \frac{\sigma_\varepsilon^2}{1 - \phi^2}, & k = 0 \\ \phi\gamma_{k-1}, & k > 0 \end{cases}$$

Los coeficientes de autocorrelación de un modelo $AR(1)$ son

$$\rho_k = \frac{\gamma_k}{\gamma_0} = \frac{\phi\gamma_{k-1}}{\gamma_0} = \phi\rho_{k-1}$$

La función de autocorrelación de un modelo $AR(1)$ es

$$\rho_k = \begin{cases} 1, & k = 0 \\ \phi\rho_{k-1}, & k > 0 \end{cases}$$

La autocorrelación de un modelo $AR(1)$ es una función exponencial, tal que

$$\begin{aligned}\rho_1 &= \phi \rho_0 = \phi \\ \rho_2 &= \phi \rho_1 = \phi^2 \\ \rho_3 &= \phi \rho_2 = \phi^3 \\ &\vdots\end{aligned}$$

Por tanto

$$\rho_k = \phi \rho_{k-1} = \phi^k, \quad k = 1, 2, 3, \dots$$

Como en el modelo $AR(1)$ el parámetro ϕ en valor absoluto es menor que la unidad, la función de autocorrelación decrece exponencialmente con k , es decir, se va aproximando a cero conforme $k \rightarrow \infty$ pero no llega nunca a valer cero.

La función de autocorrelación parcial de un modelo $AR(1)$ es

$$\phi_{kk} = \begin{cases} \phi, & k = 1 \\ 0, & k > 1 \end{cases}$$

Gráfico 3.3: Modelo AR(1)

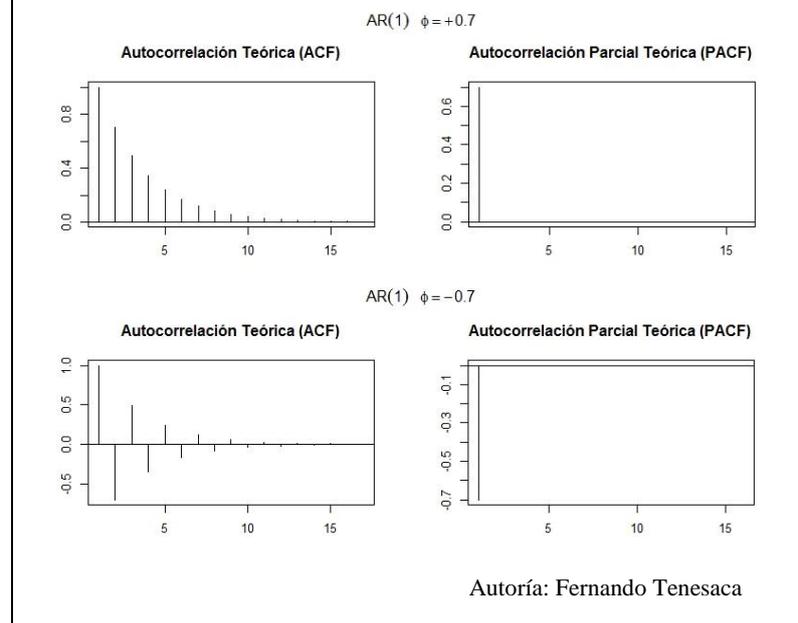
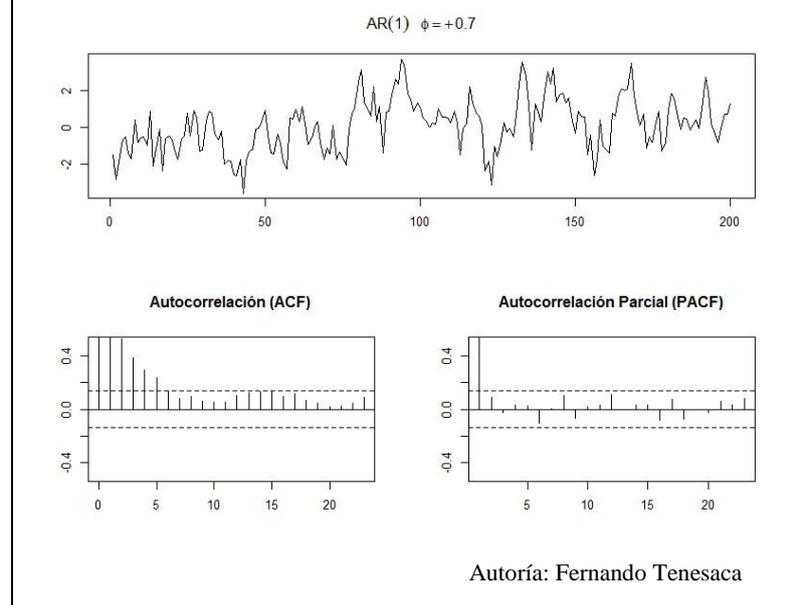
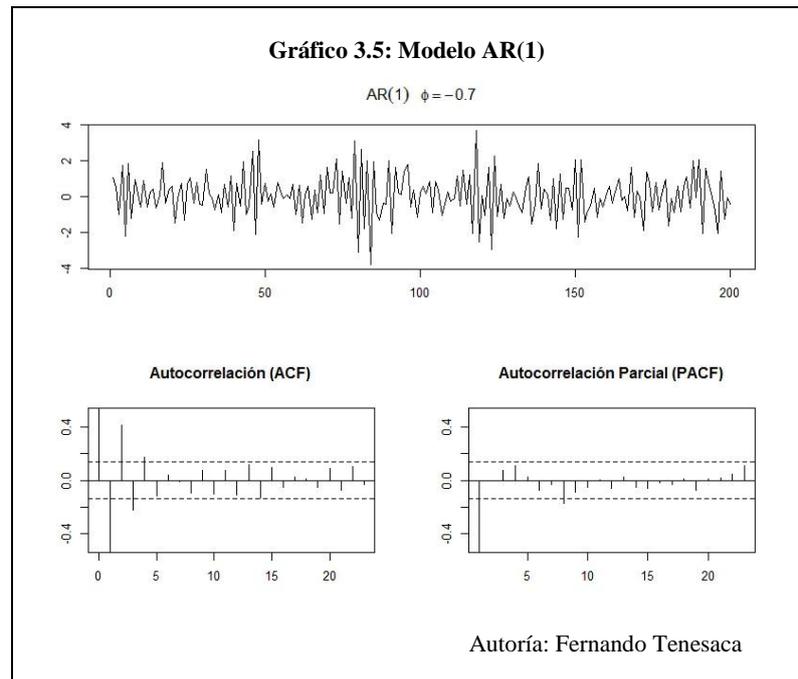


Gráfico 3.4: Modelo AR(1)





3.7.1.3. Modelo Autorregresivo de Segundo Orden

Con el razonamiento del modelo $AR(1)$, considere el modelo autorregresivo de segundo orden tal que

$$Y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \varepsilon_t \quad , \quad t = 1, 2, \dots$$

$$\varepsilon_t \sim NIID(0, \sigma^2)$$

3.7.1.4. Características del Modelo $AR(2)$

Como la serie es estacionaria la media del modelo es constante

$$(1 - \phi_1 - \phi_2)E[Y_t] = 0 \Rightarrow E[Y_t] = 0$$

La función de autocovarianza de un modelo $AR(2)$ es

$$\gamma_k = \begin{cases} \gamma_0, & k = 0 \\ \gamma_1, & k = 1 \\ \phi_1\gamma_{k-1} + \phi_2\gamma_{k-2}, & k > 1 \end{cases}$$

Los coeficientes de autocorrelación de un modelo $AR(2)$ son

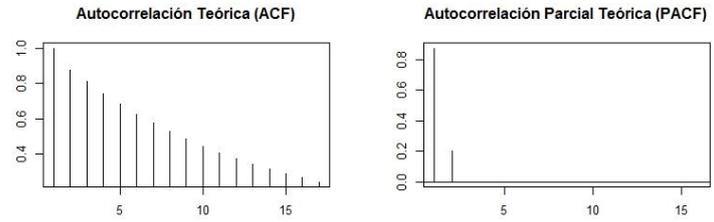
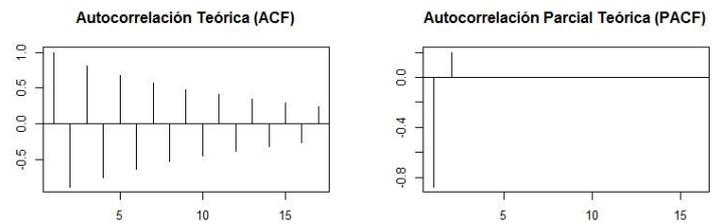
$$\rho_k = \frac{\gamma_k}{\gamma_0} = \frac{\phi_1\gamma_{k-1} + \phi_2\gamma_{k-2}}{\gamma_0} = \phi_1\rho_{k-1} + \phi_2\rho_{k-2}$$

La función de autocorrelación de un modelo $AR(2)$ es

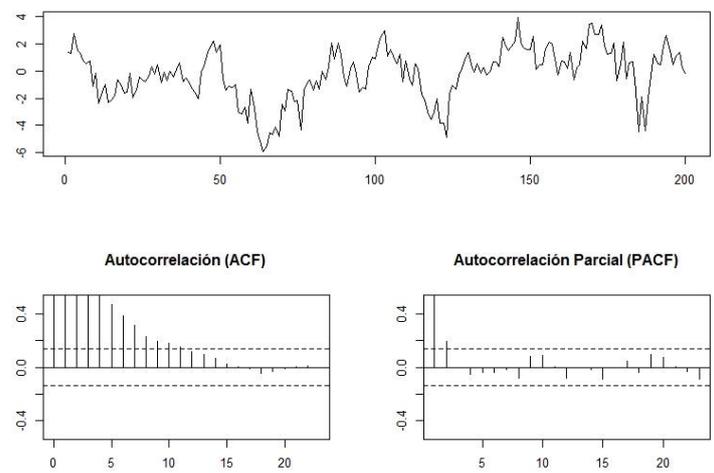
$$\rho_k = \begin{cases} 1, & k = 0 \\ \phi_1\rho_{k-1} + \phi_2\rho_{k-2}, & k > 0 \end{cases}$$

La función de autocorrelación parcial de un modelo $AR(2)$ es

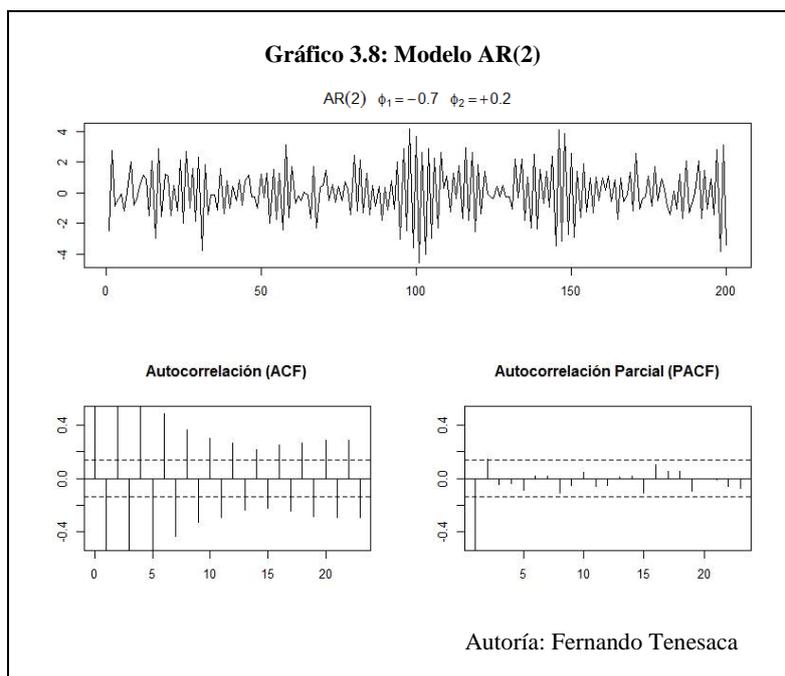
$$\phi_{kk} = \begin{cases} \phi_1/(1 - \phi_2), & k = 1 \\ \phi_2, & k = 2 \\ 0, & k > 2 \end{cases}$$

Gráfico 3.6: Modelo AR(2)AR(2) $\phi_1 = +0.7$ $\phi_2 = +0.2$ AR(2) $\phi_1 = -0.7$ $\phi_2 = +0.2$ 

Autoría: Fernando Tenesaca

Gráfico 3.7: Modelo AR(2)AR(2) $\phi_1 = +0.7$ $\phi_2 = +0.2$ 

Autoría: Fernando Tenesaca



3.7.2. Modelos de Medias Móviles

Un proceso estocástico estacionario (Y_t) sigue un modelo de media móvil de orden (q), o $MA(q)$, si y solo si

$$Y_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \dots - \theta_q \varepsilon_{t-q} \quad , \quad \varepsilon_t \sim NIID(0, \sigma^2)$$

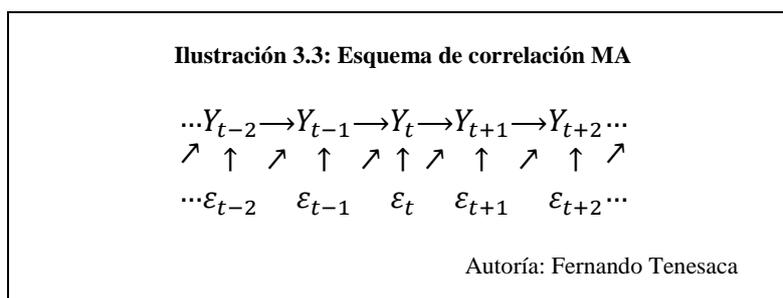
Este modelo establece que la serie estacionaria (Y_t) puede explicarse en función de q innovaciones contemporáneas e históricas donde éste parámetro determina el número de retardos necesarios para pronosticar un valor actual.

3.7.2.1. Modelo de Media Móvil de Primer Orden

El modelo $MA(1)$ define que la variable Y_t esta determinada únicamente por su valor contemporáneo de innovación, ε_t , y su primer rezago, ε_{t-1} . Se denota por

$$Y_t = \varepsilon_t - \theta\varepsilon_{t-1} \quad , \quad t = 1, 2, \dots \quad , \quad \varepsilon_t \sim NIID(0, \sigma^2)$$

El esquema de correlación en el tiempo del modelo de media móvil se ilustra de la siguiente forma



Por consiguiente, la perturbación ε_t condiciona al sistema en el tiempo t influyendo únicamente en Y_t e Y_{t+1} por éste comportamiento se lo denomina proceso de memoria corta; a diferencia del modelo autorregresivo que se lo denomina proceso de memoria larga por su esquema de correlación.

Comprobar que el modelo $MA(1)$ es estacionario para todo valor θ requiere verificar las condiciones de estacionariedad.

Modelo estacionario en media

Para que el modelo $MA(1)$ sea estacionario, la media debe ser constante y finita, así se tiene

$$\begin{aligned} E[Y_t] &= E[\varepsilon_t - \theta\varepsilon_{t-1}] \\ E[Y_t] &= E[\varepsilon_t] + E[-\theta\varepsilon_{t-1}] \\ E[Y_t] &= (0) - \theta E[\varepsilon_{t-1}] \\ E[Y_t] &= (0) - \theta(0) \\ E[Y_t] &= 0 \end{aligned}$$

Por consiguiente, la media del modelo $MA(1)$ cumple la condición para cualquier valor de θ .

Modelo estacionario en varianza

Para que el modelo $MA(1)$ sea estacionario, la varianza debe ser constante y finita, así se tiene

$$\begin{aligned} V[Y_t] &= E[(Y_t - E[Y_t])^2] \\ &= E[(\varepsilon_t - \theta\varepsilon_{t-1} - 0)^2] \\ &= E[(\varepsilon_t - \theta\varepsilon_{t-1})^2] \\ &= E[\theta^2\varepsilon_{t-1}^2 - 2\theta\varepsilon_{t-1}\varepsilon_t + \varepsilon_t^2] \\ &= \theta^2 E[\varepsilon_{t-1}^2] - 2\theta E[\varepsilon_{t-1}\varepsilon_t] + E[\varepsilon_t^2] \\ &= \theta^2 E[\varepsilon_{t-1}]^2 - 2\theta E[\varepsilon_{t-1}\varepsilon_t] + E[\varepsilon_t]^2 \end{aligned}$$

Dada la estructura de autocorrelación del proceso estocástico estacionario

$$\begin{aligned} E[\varepsilon_{t-1}\varepsilon_t] &= E[(\varepsilon_{t-1} - E[\varepsilon_{t-1}])(\varepsilon_t - E[\varepsilon_t])] \\ E[\varepsilon_{t-1}\varepsilon_t] &= E[(\varepsilon_{t-1} - 0)(\varepsilon_t - 0)] \\ E[\varepsilon_{t-1}\varepsilon_t] &= Cov[\varepsilon_{t-1}\varepsilon_t] \\ E[\varepsilon_{t-1}\varepsilon_t] &= 0 \end{aligned}$$

Por el supuesto que el proceso estocástico es estacionario

$$E[\varepsilon_{t-1}]^2 = Var[\varepsilon_{t-1}] = Var[Y_t] = \sigma_\varepsilon^2 = \gamma_0$$

De donde

$$\begin{aligned} Var[Y_t] &= \theta^2 E[\varepsilon_{t-1}]^2 - 2\theta E[\varepsilon_{t-1}\varepsilon_t] + E[\varepsilon_t]^2 \\ Var[Y_t] &= \theta^2 \sigma_\varepsilon^2 - 2\phi(0) + \sigma_\varepsilon^2 \\ \gamma_0 &= \theta^2 \sigma_\varepsilon^2 + \sigma_\varepsilon^2 \\ \gamma_0 &= (\theta^2 + 1)\sigma_\varepsilon^2 \end{aligned}$$

Por consiguiente, la varianza del modelo $MA(1)$ cumple la condición para cualquier valor de θ .

Acerca de la función de autocovarianza, γ_k , se tiene

$$\begin{aligned} \gamma_1 &= Cov[Y_t, Y_{t-1}] \\ &= E[(Y_t - E[Y_t])(Y_{t-1} - E[Y_{t-1}])] \\ &= E[Y_t Y_{t-1}] \\ &= E[(\varepsilon_t - \theta\varepsilon_{t-1})(\varepsilon_{t-1} - \theta\varepsilon_{t-2})] \\ &= E[\varepsilon_t \varepsilon_{t-1} - \theta\varepsilon_t \varepsilon_{t-2} - \theta\varepsilon_{t-1} \varepsilon_{t-1} + \theta^2 \varepsilon_{t-1} \varepsilon_{t-2}] \\ &= E[\varepsilon_t \varepsilon_{t-1}] - \theta E[\varepsilon_t \varepsilon_{t-2}] - \theta E[\varepsilon_{t-1}^2] + \theta^2 E[\varepsilon_{t-1} \varepsilon_{t-2}] \\ &= (0) - (0) - \theta E[\varepsilon_{t-1}^2] + (0) \\ &= -\theta \sigma_\varepsilon^2 \end{aligned}$$

$$\begin{aligned}
\gamma_2 &= \text{Cov}[Y_t, Y_{t-2}] \\
&= E[(Y_t - E[Y_t])(Y_{t-2} - E[Y_{t-2}])] \\
&= E[Y_t Y_{t-2}] \\
&= E[(\varepsilon_t - \theta \varepsilon_{t-1})(\varepsilon_{t-2} - \theta \varepsilon_{t-3})] \\
&= E[\varepsilon_t \varepsilon_{t-2} - \theta \varepsilon_t \varepsilon_{t-3} - \theta \varepsilon_{t-1} \varepsilon_{t-2} + \theta^2 \varepsilon_{t-1} \varepsilon_{t-3}] \\
&= E[\varepsilon_t \varepsilon_{t-2}] - \theta E[\varepsilon_t \varepsilon_{t-3}] - \theta E[\varepsilon_{t-1} \varepsilon_{t-2}] + \theta^2 E[\varepsilon_{t-1} \varepsilon_{t-3}] \\
&= 0
\end{aligned}$$

Por lo antes citado, la varianza es finita y el resto de las autocorrelaciones lo son; además éstas dependen únicamente de los periodos de separación entre las variables y no del tiempo. Se puede concluir que no es necesario poner restricciones al valor de θ para que el modelo $MA(1)$ sea estacionario.

3.7.2.2. Características del Modelo $MA(1)$

La función de autocovarianza de un modelo $MA(1)$ es

$$\gamma_k = \begin{cases} (\theta^2 + 1)\sigma_\varepsilon^2, & k = 0 \\ -\theta\sigma_\varepsilon^2, & k = 1 \\ 0, & k > 1 \end{cases}$$

La función de autocorrelación de un modelo $MA(1)$ es

$$\rho_k = \begin{cases} 1, & k = 0 \\ -\frac{\theta}{\theta^2 + 1}, & k = 1 \\ 0, & k > 1 \end{cases}$$

La autocorrelación de un modelo $MA(1)$ se caracteriza por ser una función truncada tomando el valor de cero a partir del rezago 1.

La función de autocorrelación parcial de un modelo $MA(1)$ es

$$\phi_{kk} = - \left[\frac{1}{\sum_{i=0}^k \theta^{2i}} \right] \theta^k = - \left[\frac{1 - \theta^2}{1 - \theta^{2(k+1)}} \right] \theta^k, \quad k \geq 1$$

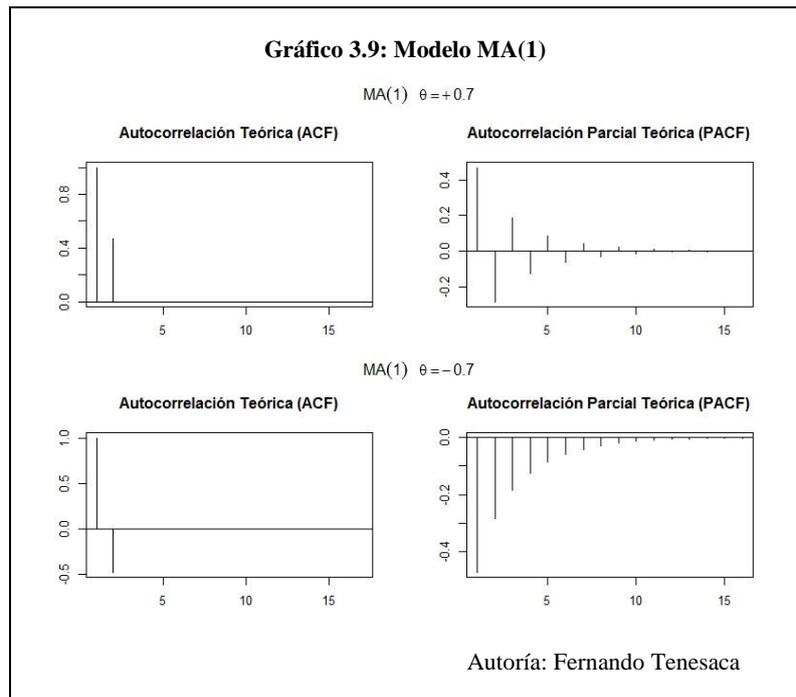
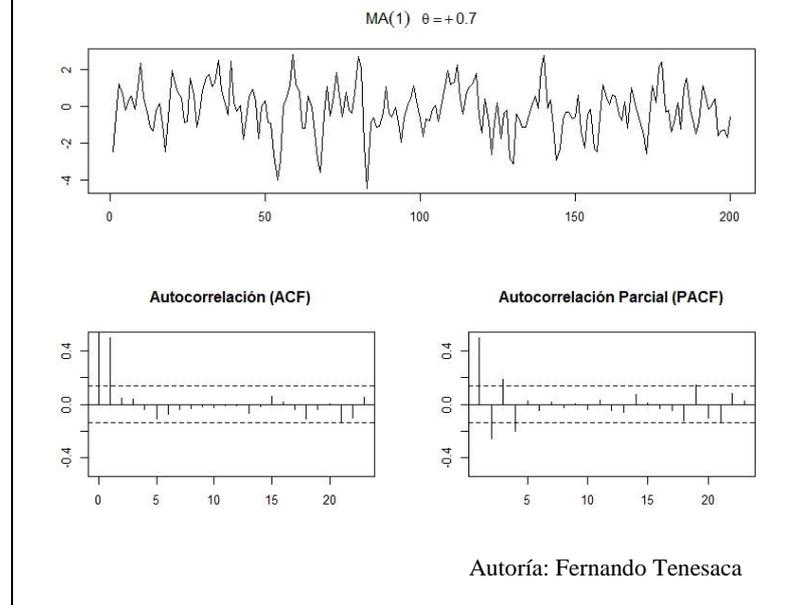
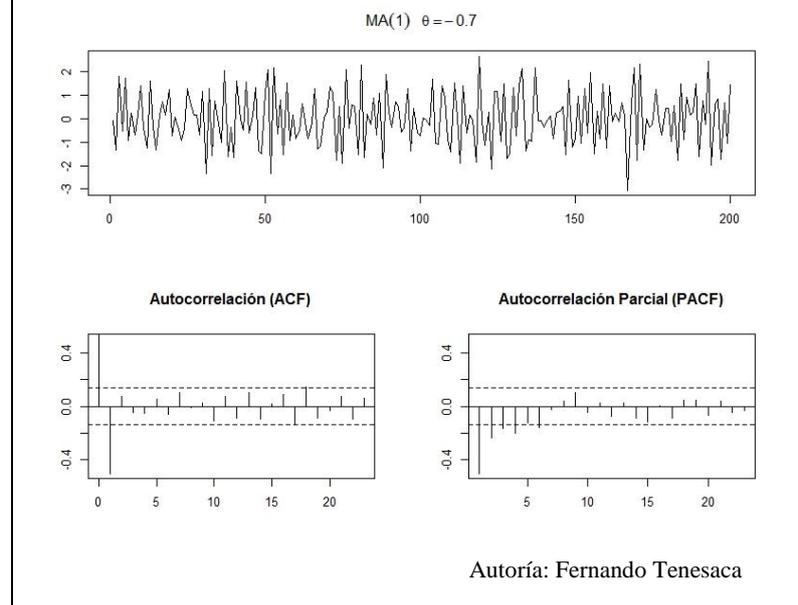


Gráfico 3.10: Modelo MA(1)**Gráfico 3.11: Modelo MA(1)**

3.7.2.3. Modelo de Media Móvil de Segundo Orden

Con el razonamiento del modelo $MA(1)$, considere el modelo de media móvil de segundo orden tal que

$$Y_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} \quad , \quad t = 1, 2, \dots$$

$$\varepsilon_t \sim NIID(0, \sigma^2)$$

3.7.2.4. Características del Modelo $MA(2)$

Como la serie es estacionaria la media del modelo es constante

$$E[Y_t] = E[\varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2}] \Rightarrow E[Y_t] = 0$$

La función de autocovarianza de un modelo $MA(2)$ es

$$\gamma_k = \begin{cases} (1 + \theta_1^2 + \theta_2^2)\sigma_\varepsilon^2, & k = 0 \\ (-1 + \theta_1\theta_2)\sigma_\varepsilon^2, & k = 1 \\ -\theta_2\sigma_\varepsilon^2, & k = 2 \\ 0, & k > 2 \end{cases}$$

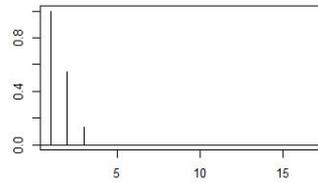
La función de autocorrelación de un modelo $MA(2)$ es

$$\rho_k = \begin{cases} \frac{-1 + \theta_1\theta_2}{1 + \theta_1^2 + \theta_2^2}, & k = 1 \\ \frac{-\theta_2}{1 + \theta_1^2 + \theta_2^2}, & k = 2 \\ 0, & k > 2 \end{cases}$$

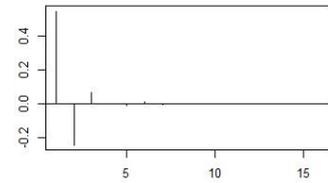
Gráfico 3.12: Modelo MA(2)

$$MA(2) \quad \theta_1 = +0.7 \quad \theta_2 = +0.2$$

Autocorrelación Teórica (ACF)

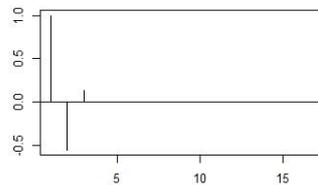


Autocorrelación Parcial Teórica (PACF)

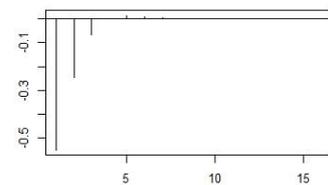


$$MA(2) \quad \theta_1 = -0.7 \quad \theta_2 = +0.2$$

Autocorrelación Teórica (ACF)



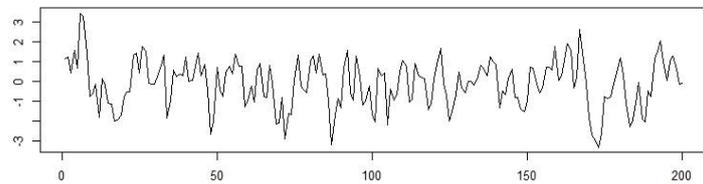
Autocorrelación Parcial Teórica (PACF)



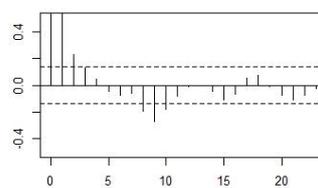
Autoría: Fernando Tenesaca

Gráfico 3.13: Modelo MA(2)

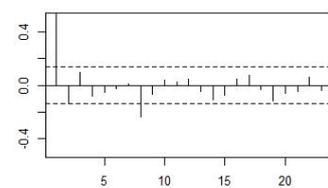
$$MA(2) \quad \theta_1 = +0.7 \quad \theta_2 = +0.2$$



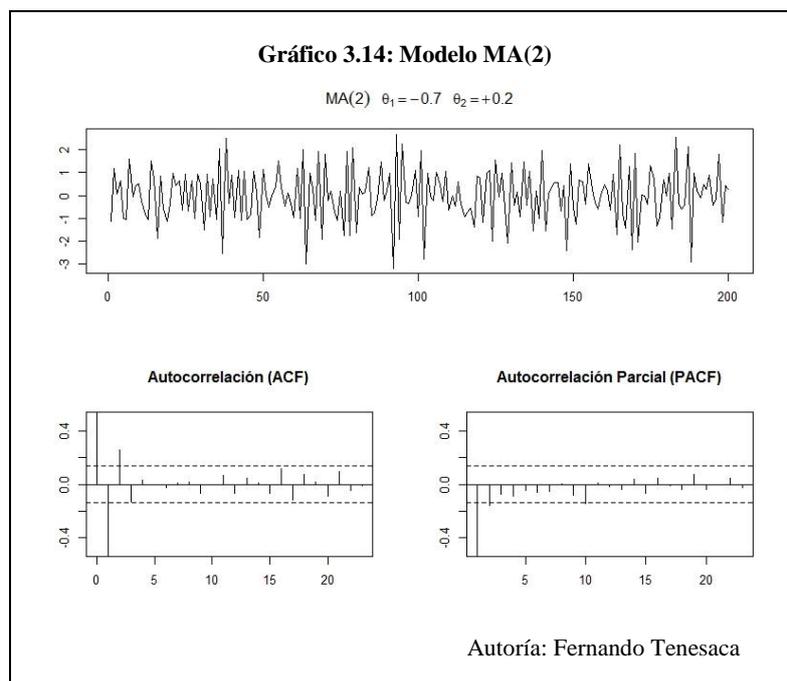
Autocorrelación (ACF)



Autocorrelación Parcial (PACF)



Autoría: Fernando Tenesaca



3.7.3. Modelos Autorregresivos de Medias Móviles

Un proceso estocástico estacionario (Y_t) sigue un modelo autorregresivo de media móvil de orden (p, q) , o $ARMA(p, q)$, si y solo si

$$Y_t = \phi_1 y_{t-1} + \dots + \phi_p y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-p}$$

$$\varepsilon_t \sim NIID(0, \sigma^2)$$

Este modelo establece que la serie estacionaria (Y_t) puede explicarse en función de p valores históricos y de q innovaciones contemporáneas y previas donde estos parámetros determinan el número de rezagos necesarios para pronosticar un valor actual.

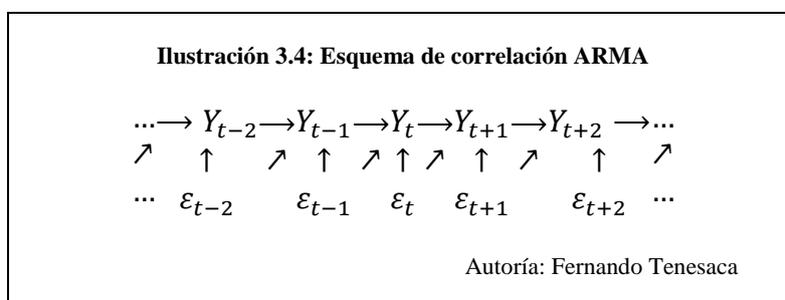
3.7.3.1. Modelo Autorregresivo de Media Móvil de Primer Orden

El modelo $ARMA(1,1)$ define que la variable Y_t está determinada por su valor pasado, Y_{t-1} , la perturbación contemporánea, ε_t , y su primer rezago, ε_{t-1} . Se denota por

$$Y_t = \phi y_{t-1} + \varepsilon_t - \theta \varepsilon_{t-1} \quad , \quad t = 1, 2, \dots$$

$$\varepsilon_t \sim NIID(0, \sigma^2)$$

El esquema de correlación en el tiempo del modelo autorregresivo de media móvil se ilustra de la siguiente forma



Por consiguiente, la perturbación ε_t condiciona al sistema en el tiempo t influyendo directamente en Y_t e Y_{t+1} ; y, de forma indirecta en las observaciones futuras; por este comportamiento se lo denomina proceso de memoria larga.

3.7.3.2. Características del Modelo $ARMA(1, 1)$

La función de autocovarianza de un modelo $ARMA(1,1)$ es

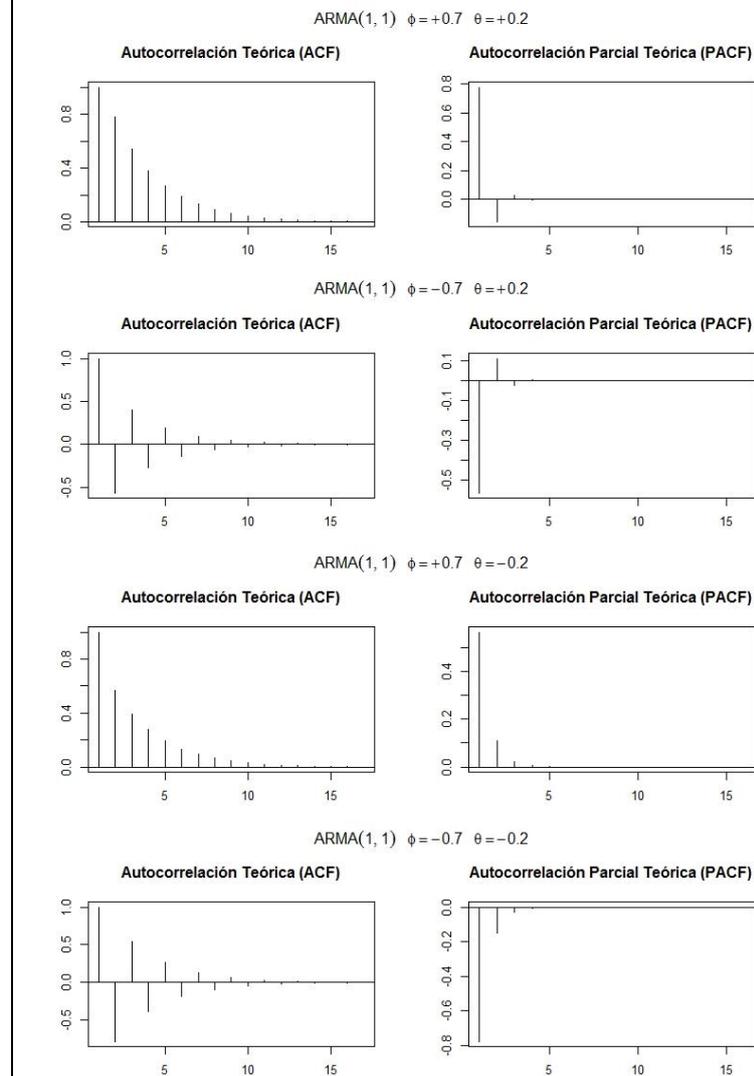
$$\gamma_k = \begin{cases} (1 + \theta^2 - 2\phi\theta)\sigma_\varepsilon^2, & k = 0 \\ \phi\gamma_0 - \theta\sigma_\varepsilon^2, & k = 1 \\ \phi\gamma_{k-1}, & k > 1 \end{cases}$$

La función de autocorrelación de un modelo $ARMA(1,1)$ es

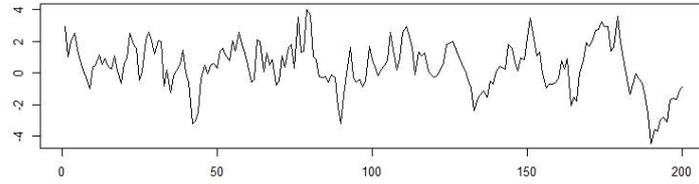
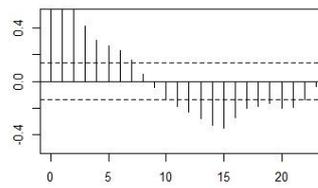
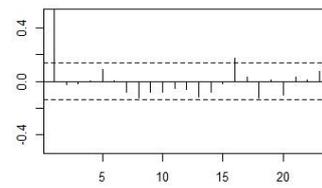
$$\rho_k = \begin{cases} \phi - \frac{\theta\sigma_\varepsilon^2}{\gamma_0}, & k = 1 \\ \phi\rho_{k-1}, & k > 1 \end{cases}$$

La autocorrelación de un modelo $ARMA(1,1)$ se caracteriza por que el primer coeficiente depende de la condición autorregresiva y de media móvil del modelo; mientras que a partir del segundo rezago decrece exponencialmente por la condición autorregresiva de primer orden.

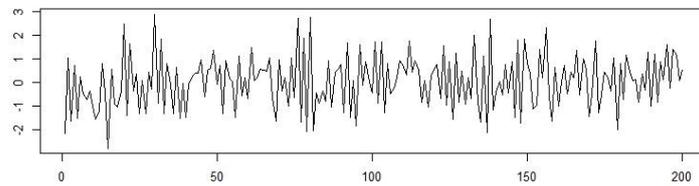
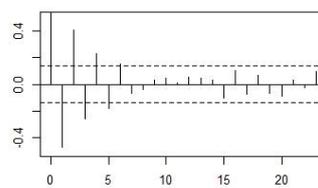
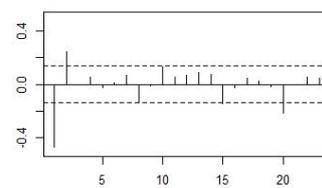
Gráfico 3.15: Modelo ARMA(1,1)



Autoría: Fernando Tenesaca

Gráfico 3.16: Modelo ARMA(1,1)ARMA(1,1) $\phi=+0.7$ $\theta=+0.2$ **Autocorrelación (ACF)****Autocorrelación Parcial (PACF)**

Autoría: Fernando Tenesaca

Gráfico 3.17: Modelo ARMA(1,1)ARMA(1,1) $\phi=-0.7$ $\theta=+0.2$ **Autocorrelación (ACF)****Autocorrelación Parcial (PACF)**

Autoría: Fernando Tenesaca

CAPÍTULO IV

4. AMIRA

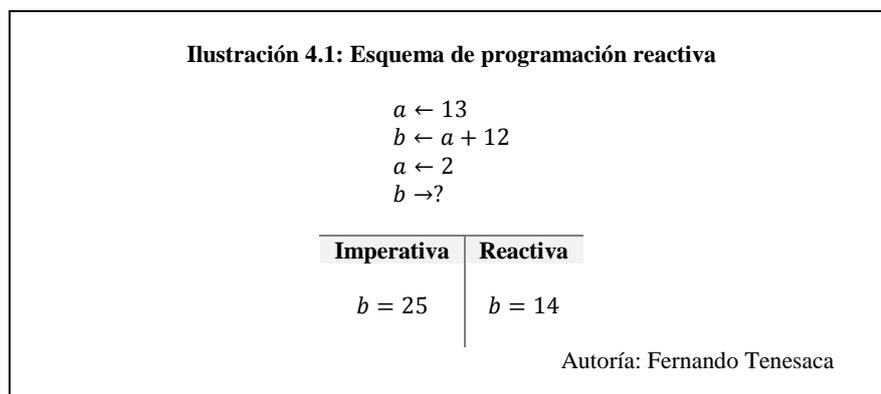
4.1. Definición

AMIRA es una aplicación web reactiva de código abierto diseñada y desarrollada como prototipo de asistente para el análisis de series temporales estacionarias.

A través de un esquema de paneles y pestañas se facilita a los usuarios, con formación estadística y conocimientos en el análisis de series de tiempo, la selección de modelos autorregresivos o de media móvil que mejor se ajusten a los datos; abstrayendo los detalles estadísticos y de programación del análisis específico para pronosticar situaciones de incertidumbre.

4.1.1. Programación Reactiva

La programación reactiva es un paradigma orientado al flujo de datos y a la propagación de cambios de los datos en dicho flujo. El término “reactiva” hace referencia al comportamiento de una aplicación en función del valor de un dato en un instante de tiempo; y, a partir de los cambios realizados en este dato, cambia también el comportamiento.



El objetivo de la programación reactiva¹⁴ es poder, de manera sencilla en cualquier lenguaje, expresar flujos de datos ya sean estáticos o dinámicos que reaccionen a cambios en ciertas condiciones y propaguen estos cambios a todos los elementos que componen una aplicación.

4.2. Diseño

En la etapa de diseño se utilizan diagramas UML como lenguaje de modelado para visualizar, especificar, construir y documentar el desarrollo de la aplicación web desde diferentes perspectivas lo que permite obtener una representación simplificada de su funcionalidad y alcance.

4.2.1. Casos de Uso

Es un diagrama UML que permite visualizar el conjunto de escenarios que describen el uso particular de un aplicación desde la óptica del actor o rol específico que desencadena una funcionalidad del mismo como respuesta a su estímulo o interacción.

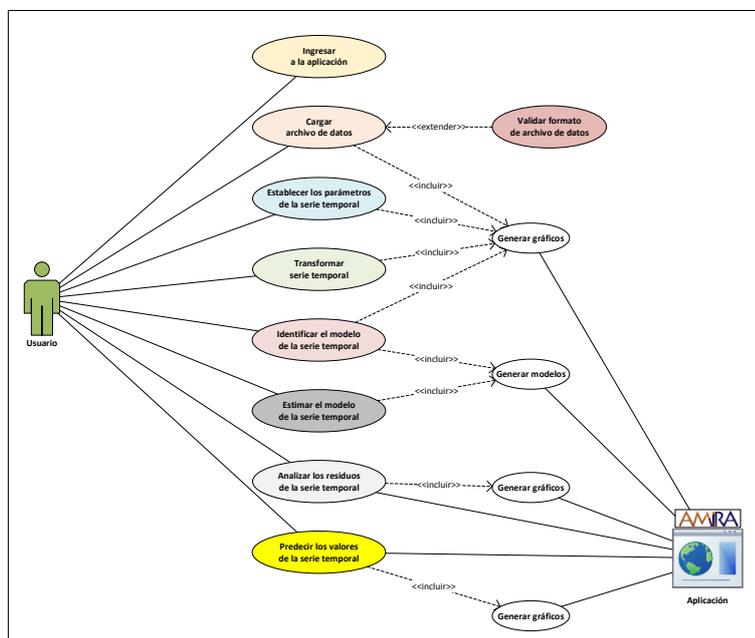
Los casos de uso principales que se derivan de los requerimientos funcionales se presentan en la tabla a continuación.

Tabla 4.1: Casos de uso

Orden	Descripción
1	Ingresar a la aplicación
2	Cargar archivo de datos
3	Establecer los parámetros de la serie temporal
4	Transformar la serie temporal
5	Establecer los parámetros de la serie temporal
6	Transformar la serie temporal
7	Identificar el modelo de la serie temporal
8	Estimar los coeficientes del modelo
9	Analizar los residuos del modelo
10	Predecir los valores futuros de la serie temporal

Autoría: Fernando Tenesaca

Ilustración 4.2: Diseño | UML | Casos de uso



Autoría: Fernando Tenesaca

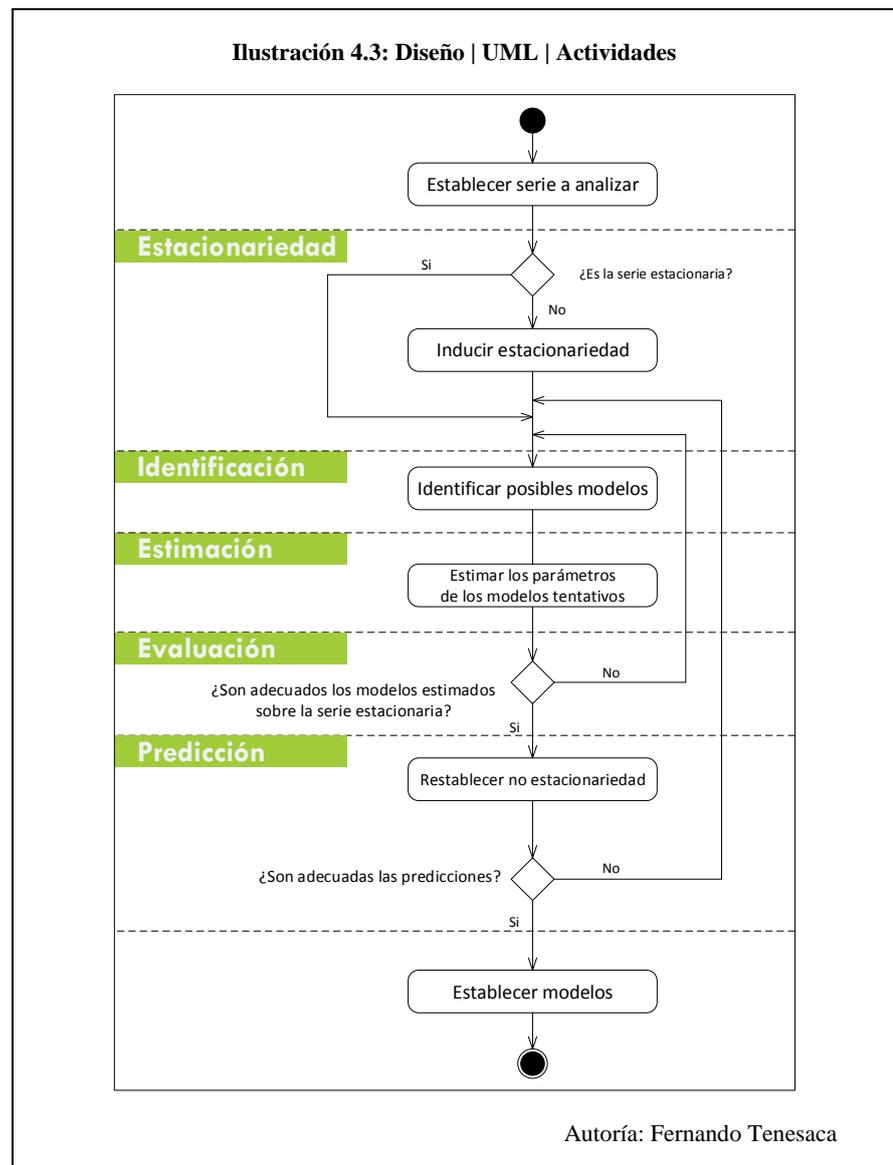
Tabla 4.2: Casos de uso | Ingresar a la aplicación

Nombre:	Ingresar a la aplicación
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe ingresar a la aplicación a través de un navegador.
Actores:	Usuario
Precondiciones:	El usuario debe disponer de un equipo portátil o de escritorio con acceso a internet o en su defecto instalado el lenguaje de programación, <i>R</i> ; el entorno de desarrollo integrado, <i>RStudio</i> ; el entorno de desarrollo, <i>Shiny</i> ; y, los complementos necesarios para su ejecución de forma local.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario abre una ventana del navegador. 2. El usuario en la barra de dirección escribe el URL http://espol.shinyapps.io/amira y presiona la tecla <i>Enter</i>.
Flujo Alternativo:	<ol style="list-style-type: none"> 1A. El usuario carga el entorno de desarrollo integrado, <i>RStudio</i>. 2A. El usuario ejecuta la aplicación <i>AMIRA</i> a través del comando <code>runApp("AMIRA")</code>.
Poscondiciones:	El usuario ha ingresado a la aplicación y se encuentra activo en la sección <i>Datos</i> en el panel derecho (main panel).

Autoría: Fernando Tenesaca

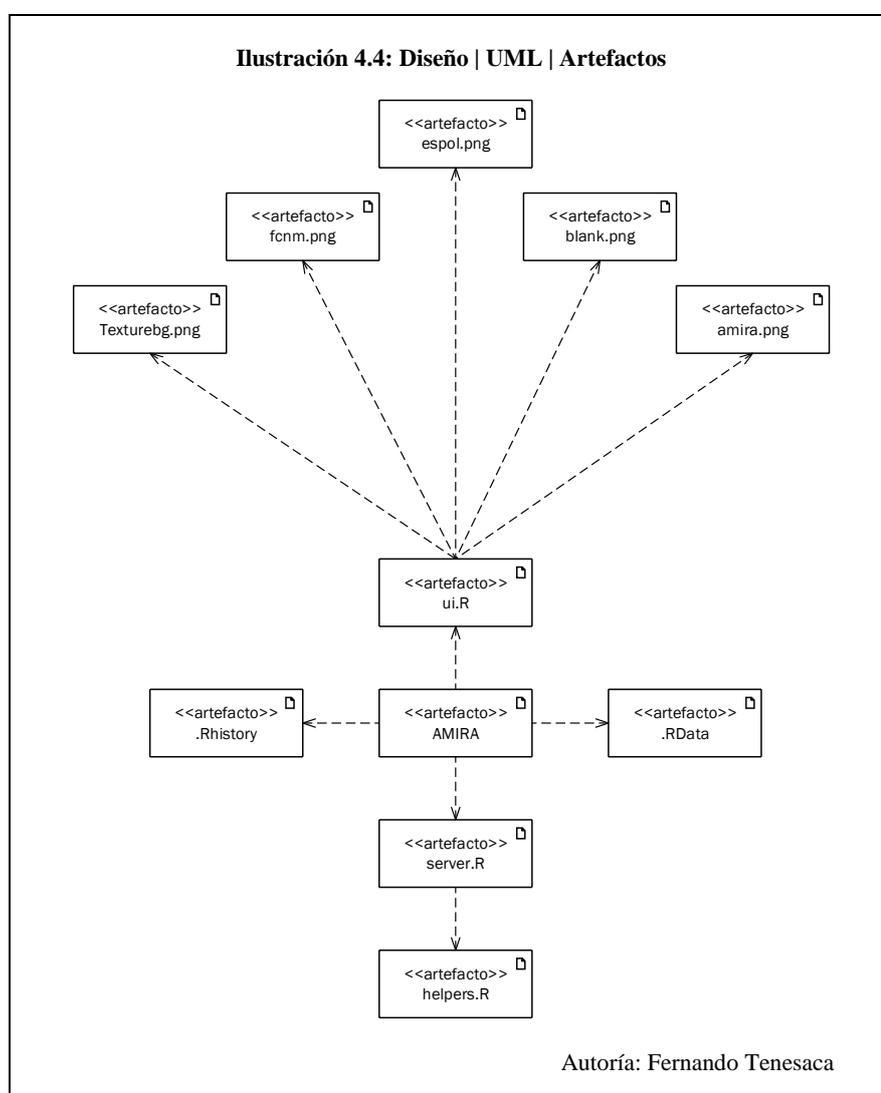
4.2.2. Actividades

Es un diagrama UML que describe el comportamiento del sistema a través de una serie de acciones o tareas que se ejecutan en cierto orden y que estructuran un flujo de trabajo.



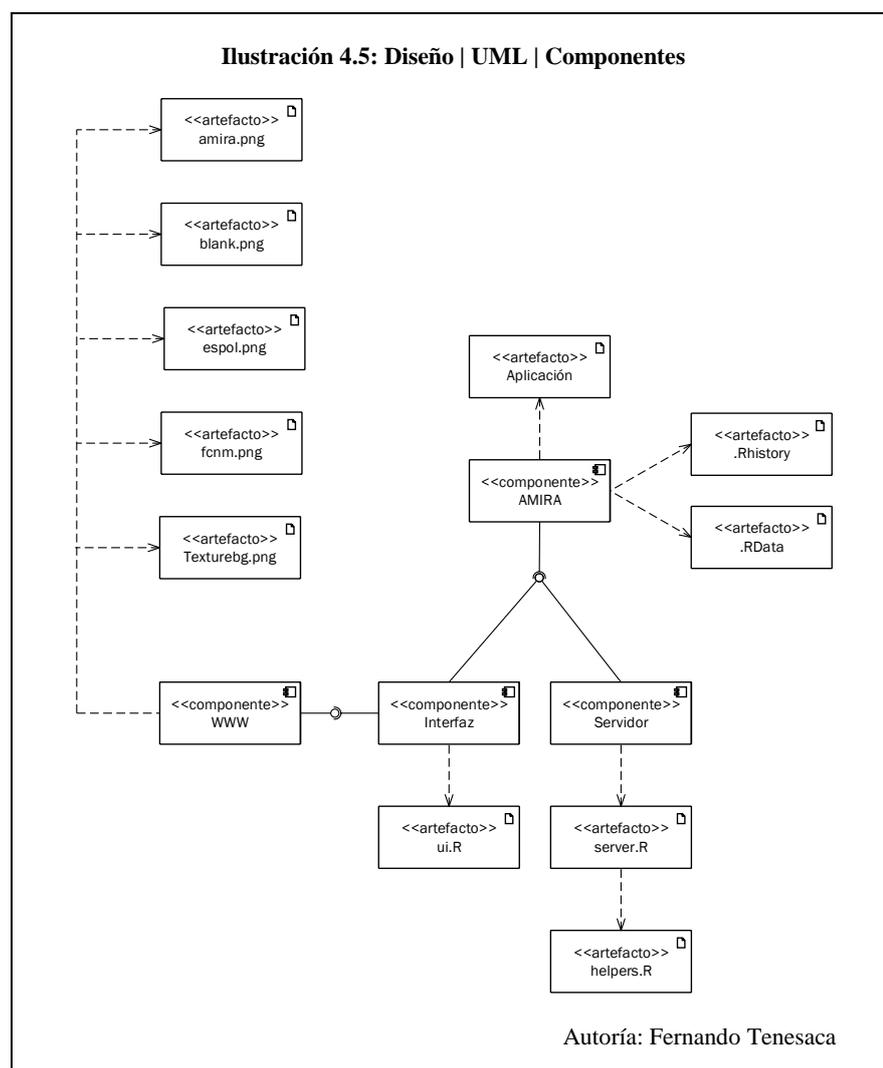
4.2.3. Artefactos

Es un diagrama UML que permite visualizar los elementos físicos de información que se producen como parte del proceso de desarrollo de software; y representar su especificación en un escenario de implementación concreto real.



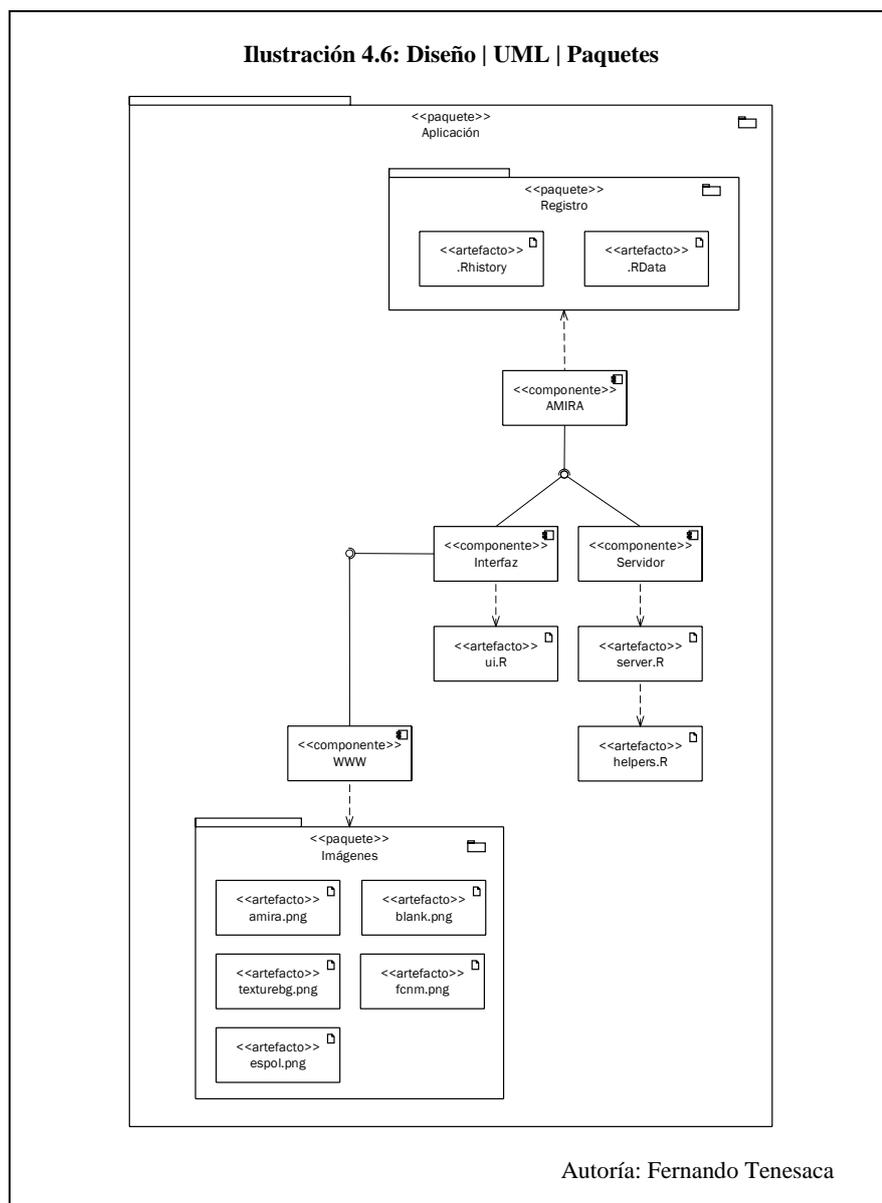
4.2.4. Componentes

Es un diagrama UML que permite representar las dependencias entre componentes software, incluyendo componentes de código fuente, componentes de código binario y componentes ejecutables.



4.2.5. Paquetes

Es un diagrama UML que permite denotar como una aplicación se encuentra desagregada en agrupaciones lógicas con un nivel jerárquico y dependencias entre sí.



4.2.6. Despliegue

Es un diagrama UML de tipo estructurado que muestra la arquitectura del sistema desde el punto de vista de la distribución de los artefactos de software a lo largo de la arquitectura de hardware.

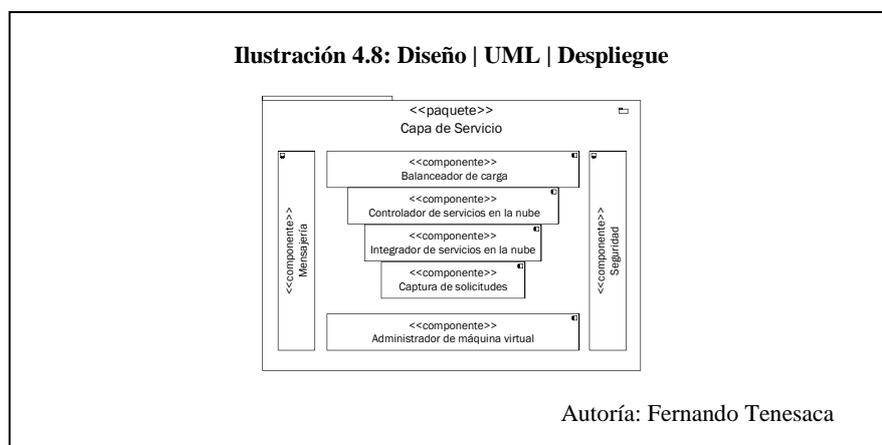
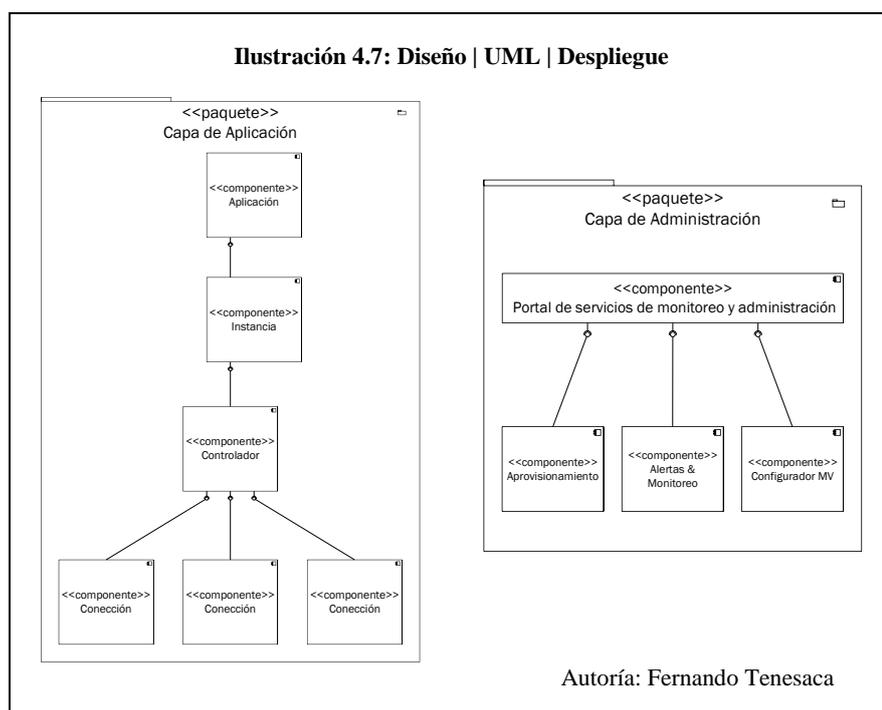
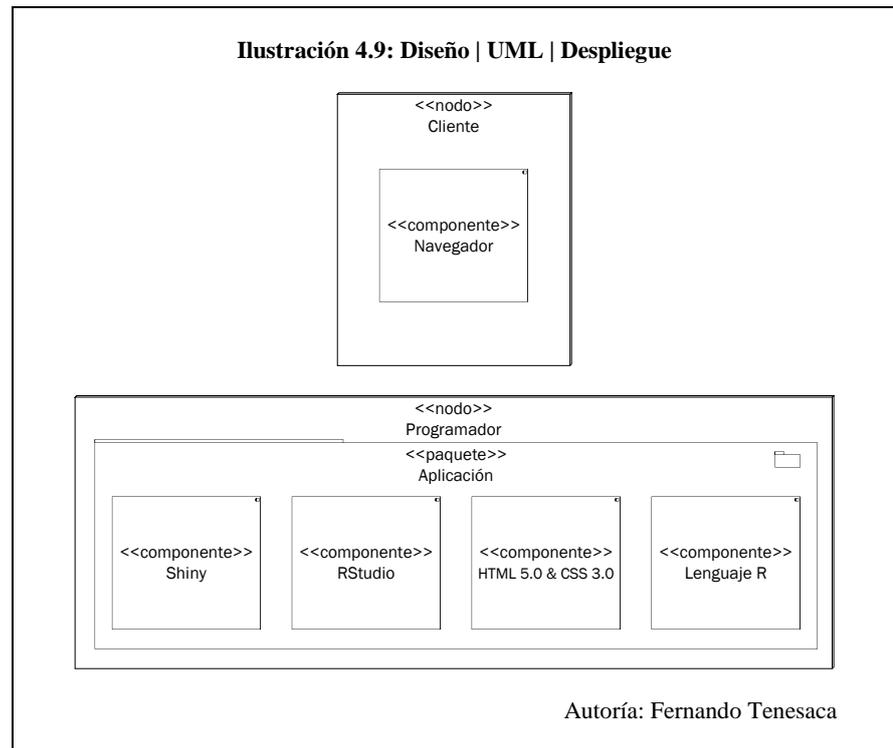


Ilustración 4.9: Diseño | UML | Despliegue

Autoría: Fernando Tenesaca

4.3. Implementación

Para implementar el asistente para el análisis de series temporales estacionarias se debe tener presente los siguientes conceptos:

- La solución de software (AMIRA) esta soportada sobre una plataforma de servicios (PaaS) para proveer acceso a aplicaciones web reactivas basadas en componentes de desarrollo de código abierto.
- El componente de análisis estadístico y gráfico esta soportado por el lenguaje de programación R.
- Las capacidades de visualización e interactividad las proveen las funcionalidades de HTML 5.0 junto con CSS 3.0.
- El entorno de desarrollo lo integran RStudio y el componente de desarrollo Shiny (R package).

Ilustración 4.10: Implementación



Autoría: Garrett Grolemond

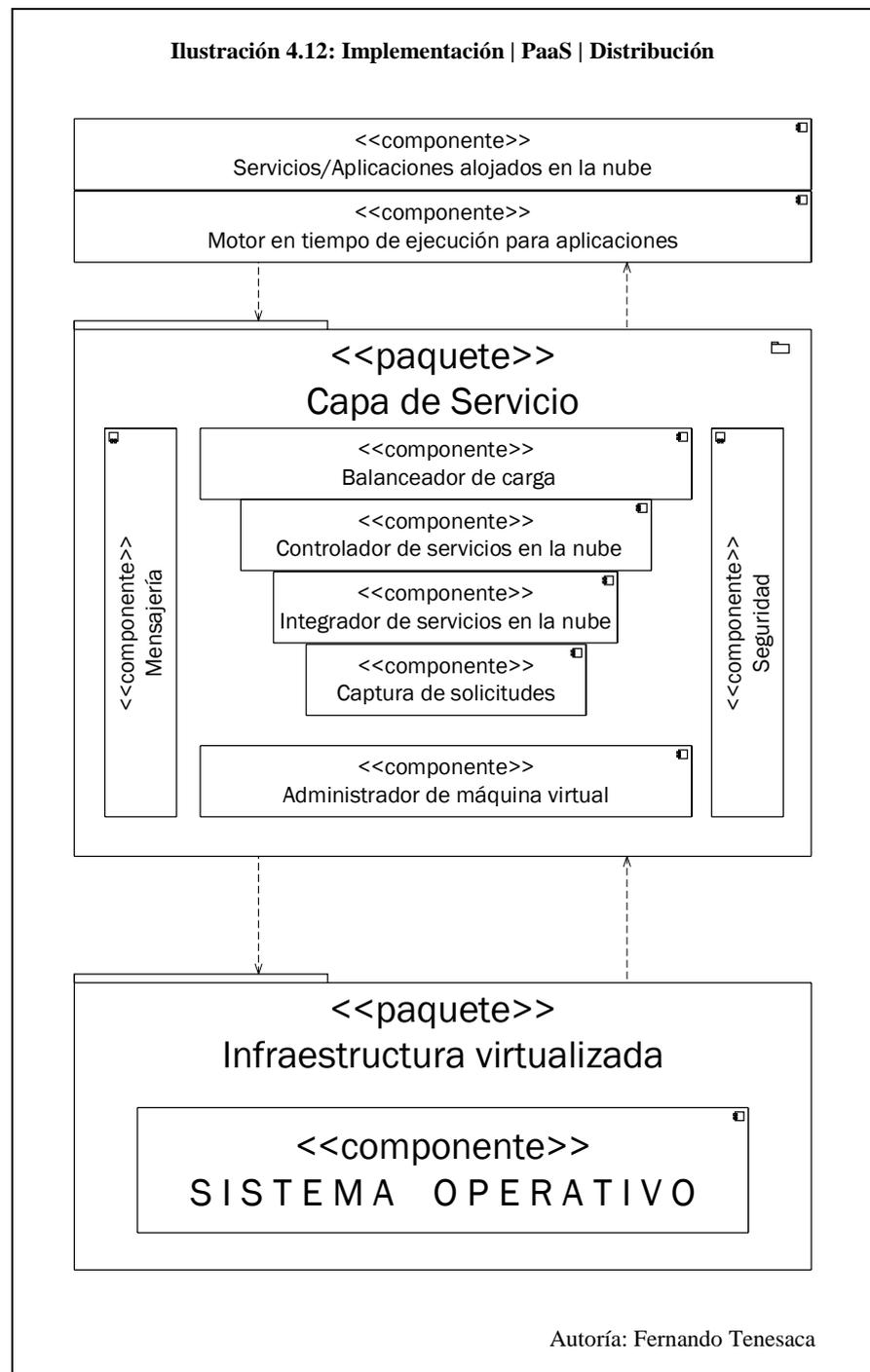
4.3.1. Plataforma como un Servicio

Es una categoría de servicios de computación en la nube que provee a los usuarios de una plataforma para el desarrollo, ejecución y administración de aplicaciones web sin la complejidad de implementar y mantener la infraestructura típica asociada a su operación.

La plataforma como un servicio (PaaS, Platform as a Service) utilizada para implementar la solución de software esencialmente es una caja negra.



Ilustración 4.12: Implementación | PaaS | Distribución



4.4. Ejemplos de Aplicación

4.4.1. Ejemplo 1

En el presente ejemplo de aplicación se considera la siguiente ecuación:

$$Y_t = 0.5Y_{t-1} - 0.4Y_{t-2} + 0.6Y_{t-3}$$

Nótese que en el contexto de este informe, la estructura de la ecuación está asociada con una serie de tiempo que puede ser descrita por un modelo autorregresivo de tercer orden, $AR(3)$, con sus coeficientes:

$$\phi_1 = +0.5$$

$$\phi_2 = -0.4$$

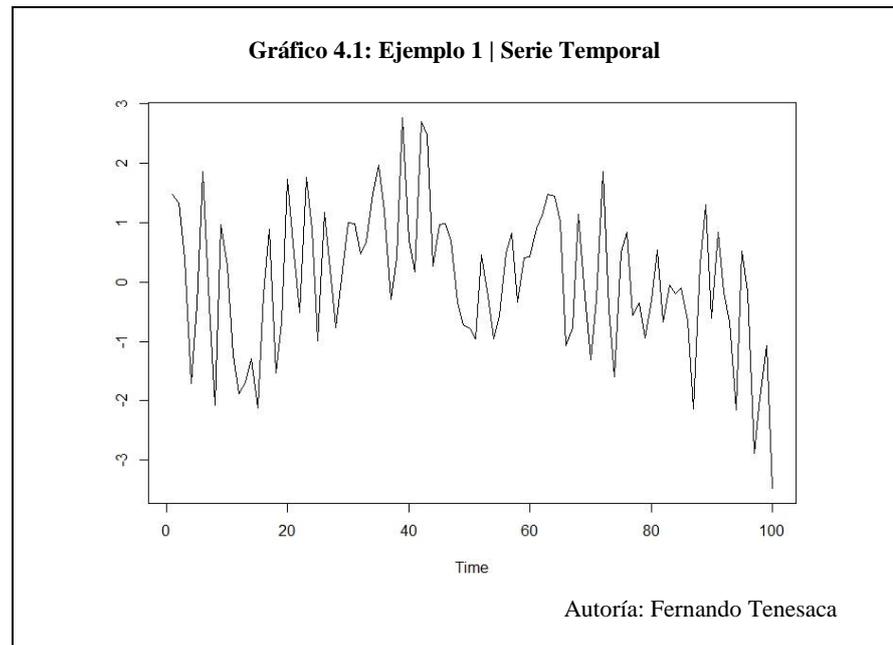
$$\phi_3 = +0.6$$

A partir de la ecuación, se construye la serie relacionada para analizar sus características, de la siguiente forma:

Tabla 4.3: Ejemplo 1 | Serie Temporal

```
set.seed(3)
ar.coef <- c(0.5, -0.4, 0.6)
ts.sim <- arima.sim(list(ar = ar.coef), n = 100)
```

Autoría: Fernando Tenesaca



Al observar el Gráfico 4.1, se puede considerar la serie temporal como estacionaria; sin embargo, para demostrar teóricamente esta condición, se puede calcular las raíces del polinomio autorregresivo.

Tabla 4.4: Ejemplo 1 | Raíces

```
abs(polyroot(c(1, - ar.coef)))
[1] 1.170785 1.193125 1.193125
```

Autoría: Fernando Tenesaca

Del resultado en la Tabla 4.4 se verifica que todas las raíces se encuentran fuera del círculo unitario, por tanto la serie temporal se puede considerar como estacionaria.

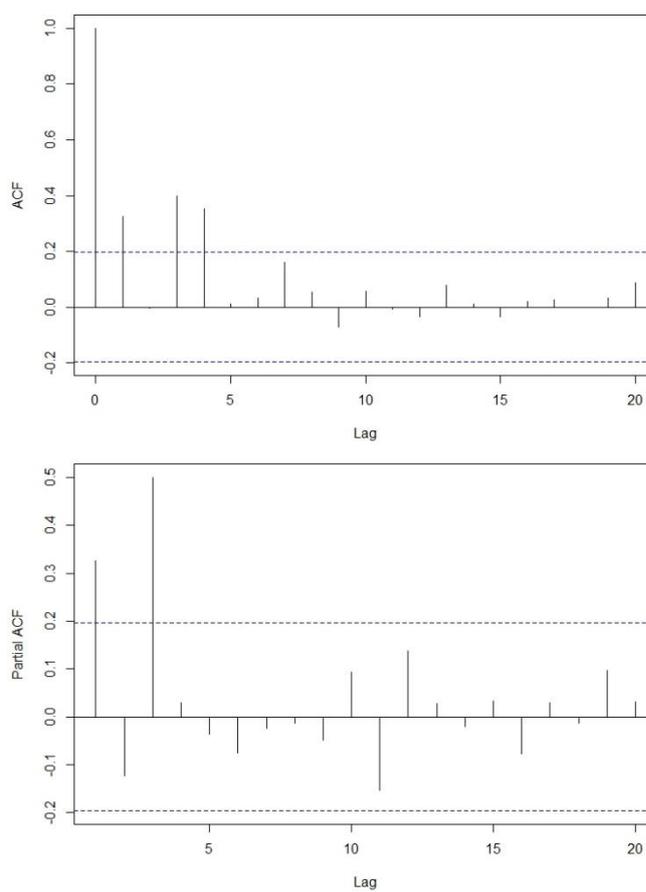
A continuación, se generan las funciones de autocorrelación simple y parcial para comprobar el orden del modelo autorregresivo.

Tabla 4.5: Ejemplo1 | Función de Autocorrelación [ACF & PACF]

```
acf(ts.sim, plot=TRUE)
acf(ts.sim, type="partial", plot=TRUE)
```

Autoría: Fernando Tenesaca

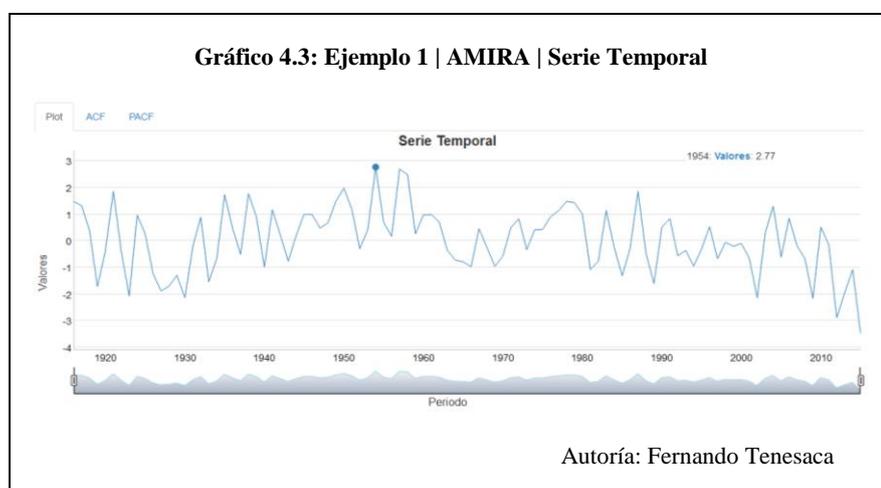
Gráfico 4.2: Ejemplo 1 | Función de Autocorrelación



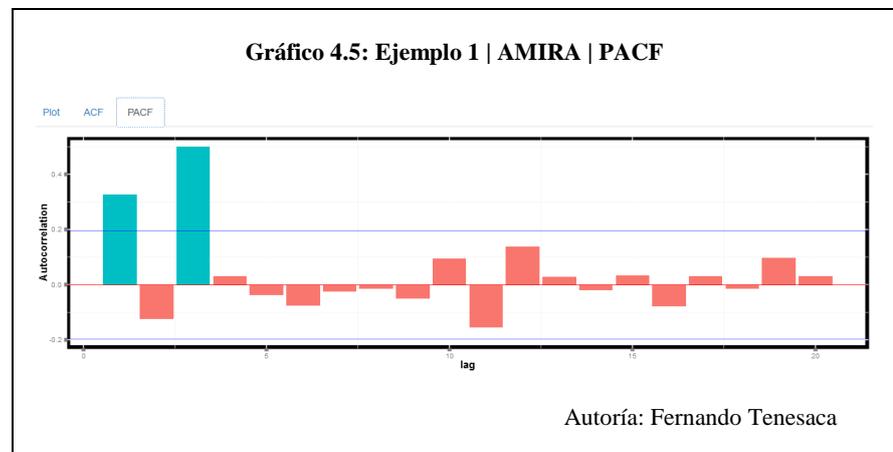
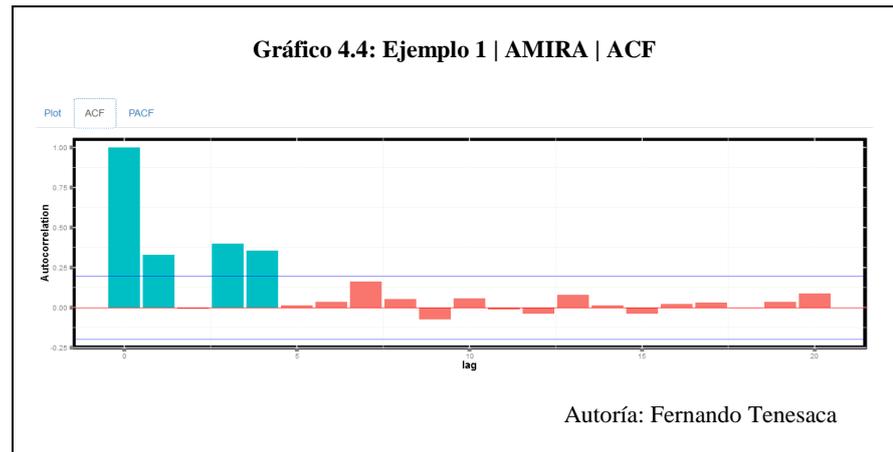
Autoría: Fernando Tenesaca

Al observar el Gráfico 4.2, se evidencia que con excepción del rezago 7, las autocorrelaciones (ACF) decaen rápidamente. Las autocorrelaciones parciales son significativas entre el rezago 1 y 3, lo que indica que el modelo autorregresivo asociado a la serie es de tercer orden, $AR(3)$; los rezagos subsiguientes se consideran no significativos, independiente de los rezagos del 10 al 12.

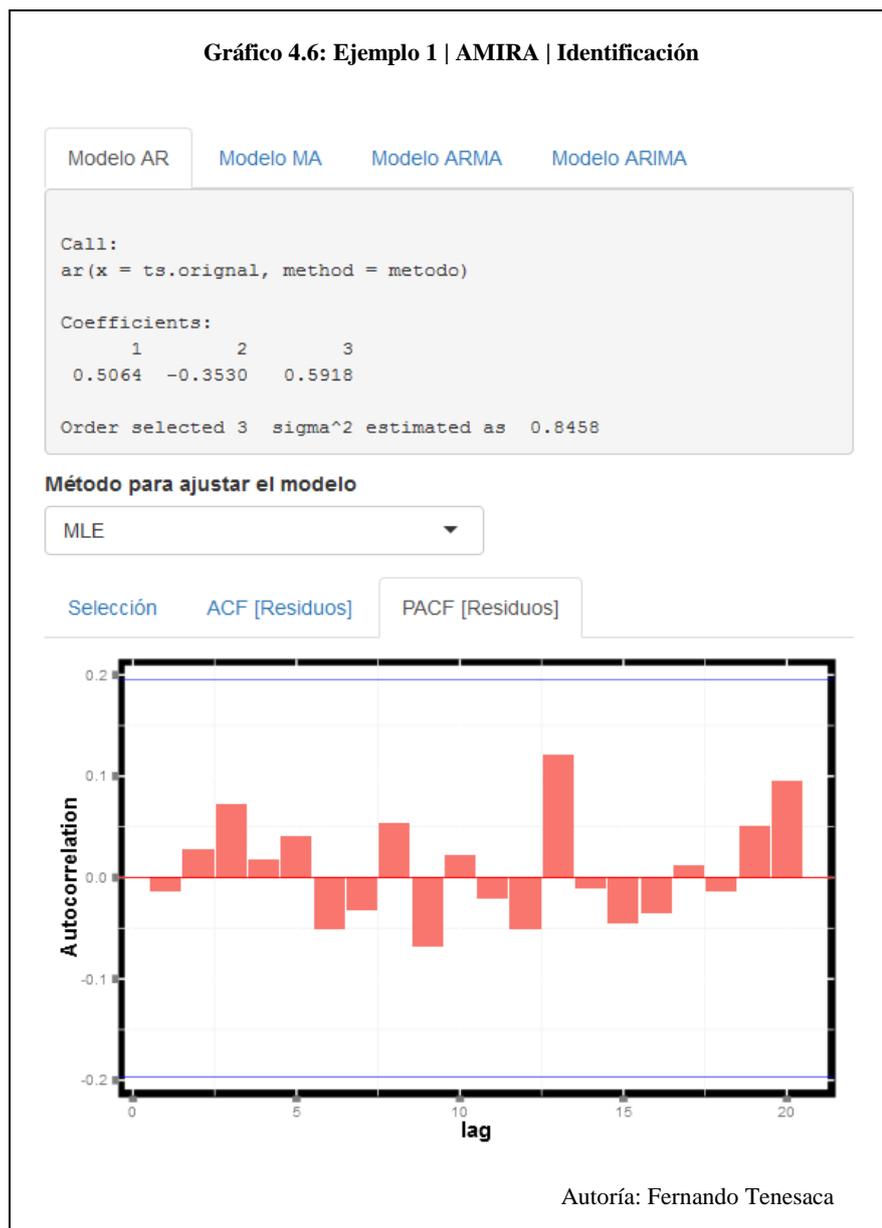
Al cargar los datos en la aplicación, AMIRA, el usuario constata la funcionalidad reactiva de la misma, lo que facilita la navegación y visualización del comportamiento de los datos cuando el tamaño de la serie temporal es lo suficientemente grande.



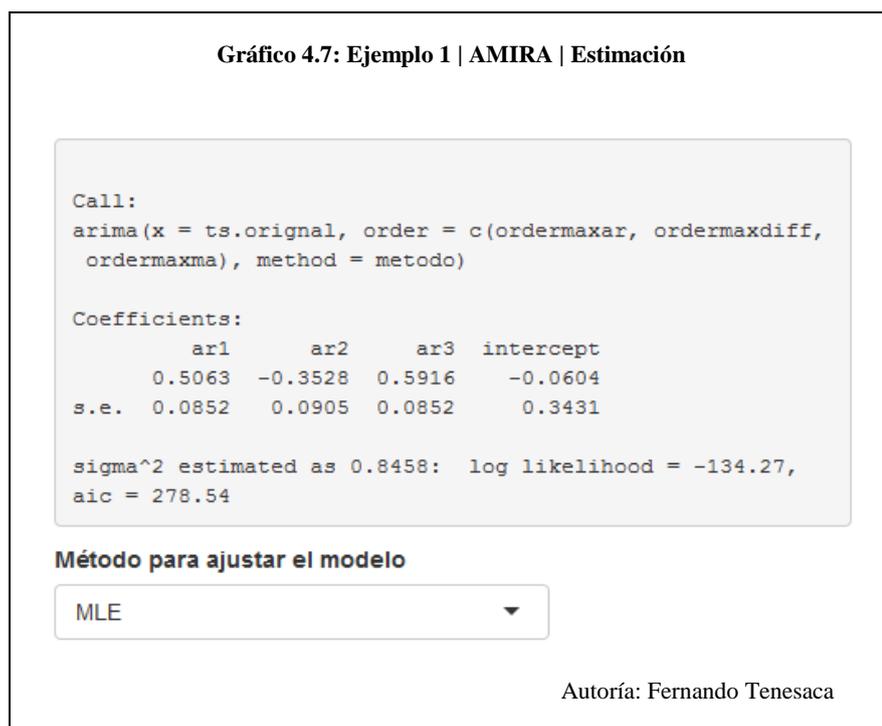
A continuación, AMIRA grafica las funciones de autocorrelación simple y parcial, que de manera preliminar establecen que el modelo autorregresivo es $AR(3)$.



AMIRA identifica el modelo asociado a la serie temporal y también permite que el usuario establezca diferentes modelos para contrastar con el sugerido por la aplicación.



Una vez identificado y seleccionado el modelo, AMIRA, procede a estimar los coeficientes para la ecuación que mejor se ajusta a los datos de la serie temporal.



Es importante la validación de resultados, por tanto, AMIRA analiza o diagnostica los residuos de la serie estimada en función del modelo seleccionado a través de la prueba de aleatoriedad o prueba de Ljung-Box, la misma comprueba si los residuos se distribuyen de forma independiente (hipótesis nula).

Gráfico 4.8: Ejemplo 1 | AMIRA | Análisis

Prueba de Aleatoriedad de los Residuos

Ljung-Box

Hipótesis:**H0:** Los datos SI se distribuyen de forma independiente.**H1:** Los datos NO se distribuyen de forma independiente.**Cálculo:**

El estadístico de prueba es:

[1] 0.01221125

El valor-p de la prueba es:

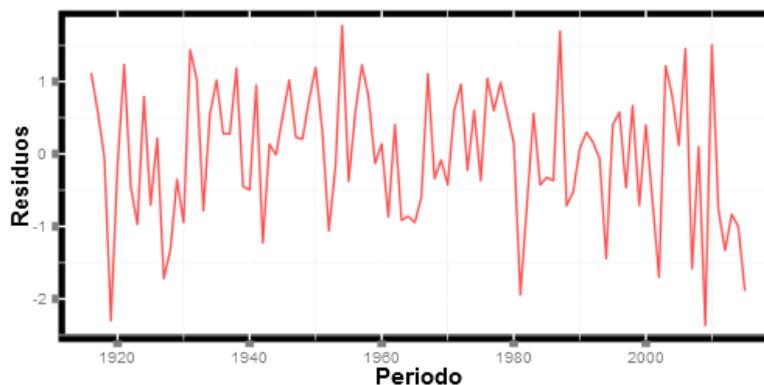
[1] 0.9120092

Decisión:

[1] "El valor-p de la prueba es mayor que 0.05 por tanto NO se rechaza la hipótesis nula H0"

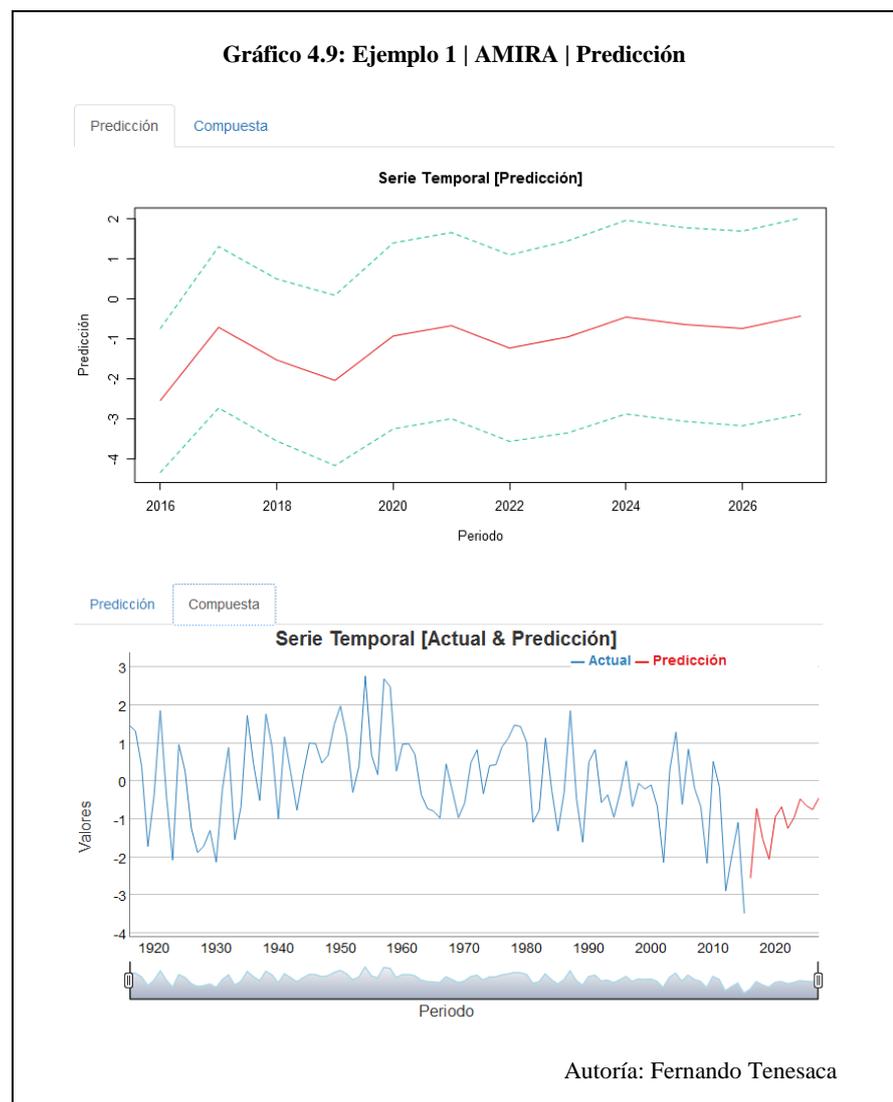
Conclusión:

[1] "Por consiguiente, NO hay suficiente evidencia estadística para rechazar la hipótesis nula H0; es decir, los residuos del modelo presentan aleatoriedad."

Residuos de la Serie Estimada

Autoría: Fernando Tenesaca

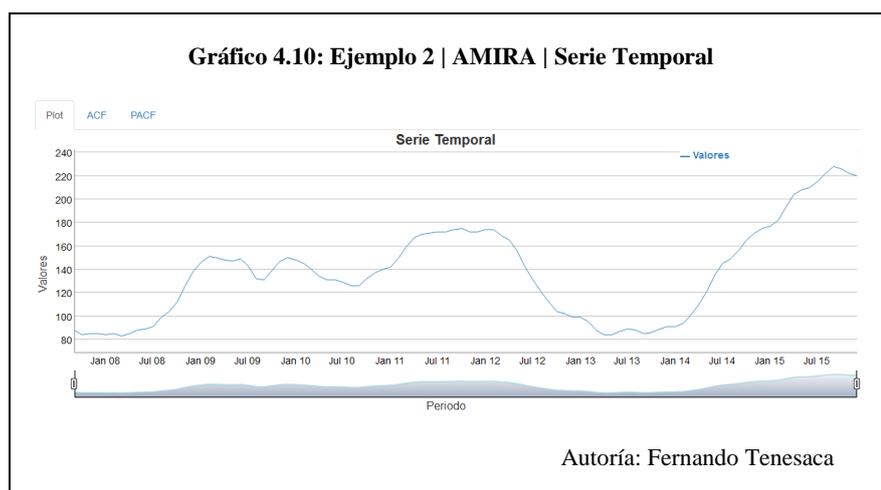
Al no existir suficiente evidencia estadística para rechazar la hipótesis nula, AMIRA predice y grafica los valores asociados a la serie estimada de acuerdo al modelo seleccionado como parte del proceso final del análisis de la serie temporal.



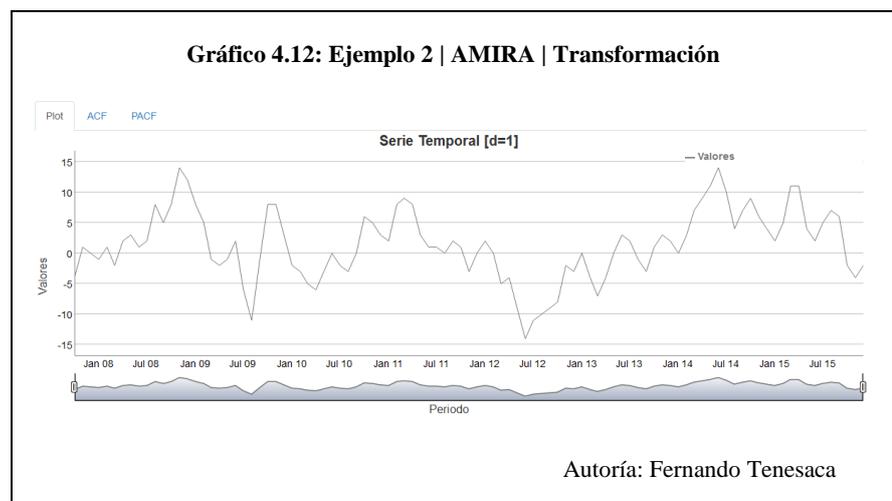
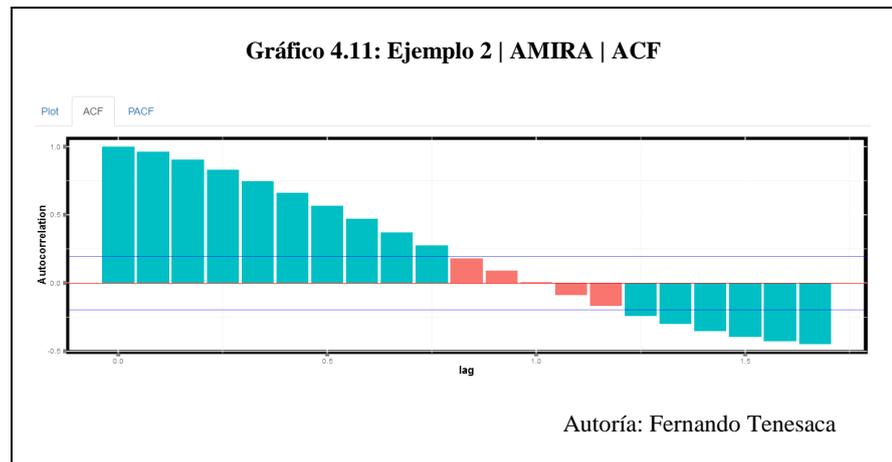
4.4.2. Ejemplo 2

Una empresa de ventas por internet en su servidor para comercio electrónico registra mensualmente el número de usuarios nuevos que completan correctamente su perfil y efectúan alguna compra con éxito durante un periodo de 100 meses, desde septiembre del 2007. El gerente de la empresa requiere predecir el número de usuarios registrados bajo esas características para el primer trimestre en el año 2016.

Al cargar los datos en la aplicación, AMIRA, el usuario constata la funcionalidad reactiva de la misma, lo que facilita la navegación y visualización del comportamiento de los datos cuando el tamaño de la serie temporal es lo suficientemente grande.

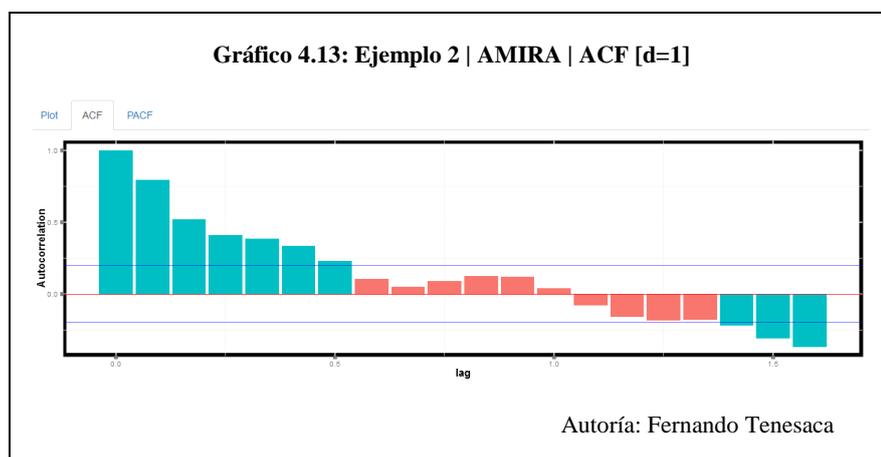


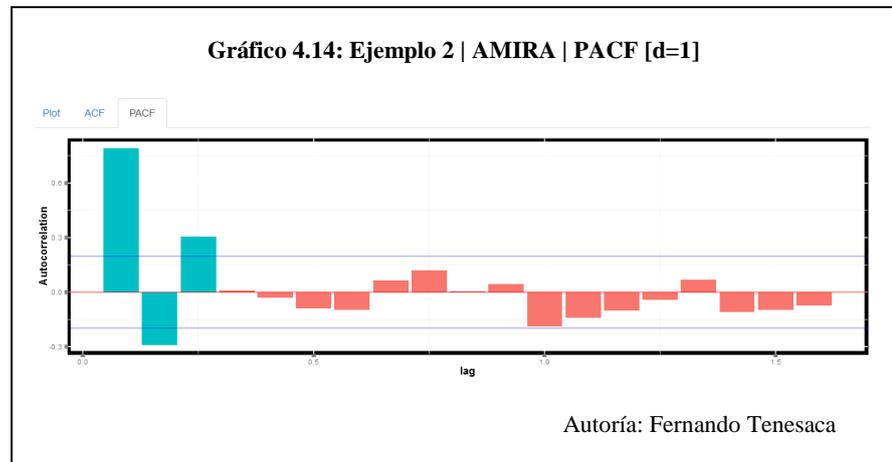
En el Gráfico 4.10, se puede observar cierto nivel de tendencia a medida que evoluciona la serie, lo que sugiere la condición de *no estacionariedad*; por tanto, es necesario aplicar transformaciones a la serie original para luego identificar un modelo que se ajuste a los datos. Este escenario se puede comprobar con un suave descenso de los valores en la función de autocorrelación (ACF) que genera AMIRA, como se visualiza en el Gráfico 4.11.



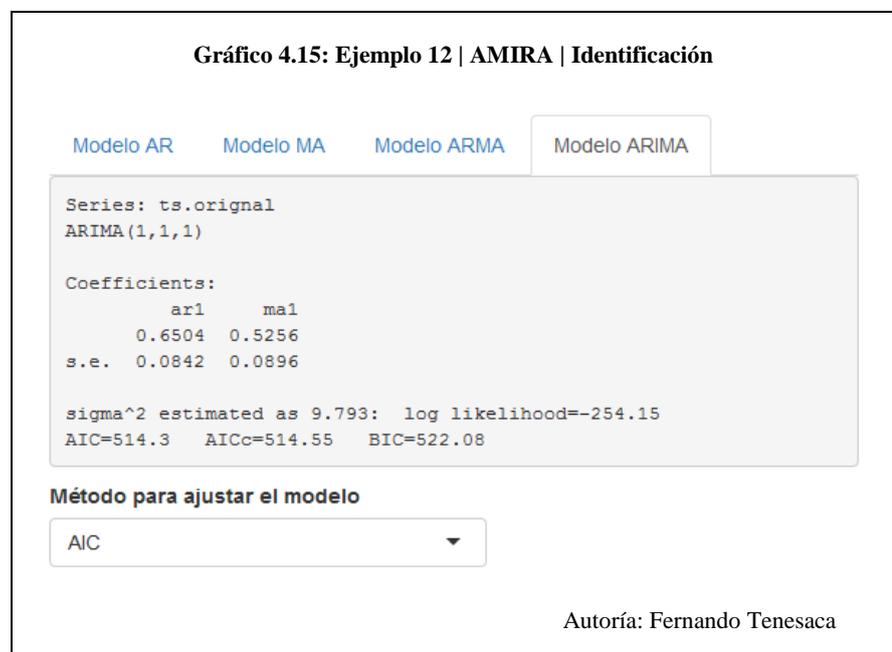
La opción correspondiente en AMIRA modifica la serie original y genera una secuencia de datos estable en media y varianza, para ser analizada como una serie temporal estacionaria. En síntesis, se calculan las diferencias existentes entre cada valor y el anterior de la serie original, a excepción del valor más antiguo. Por tanto, la serie transformada de esta forma es una *serie diferenciada* y tendrá un valor menor que la serie original.

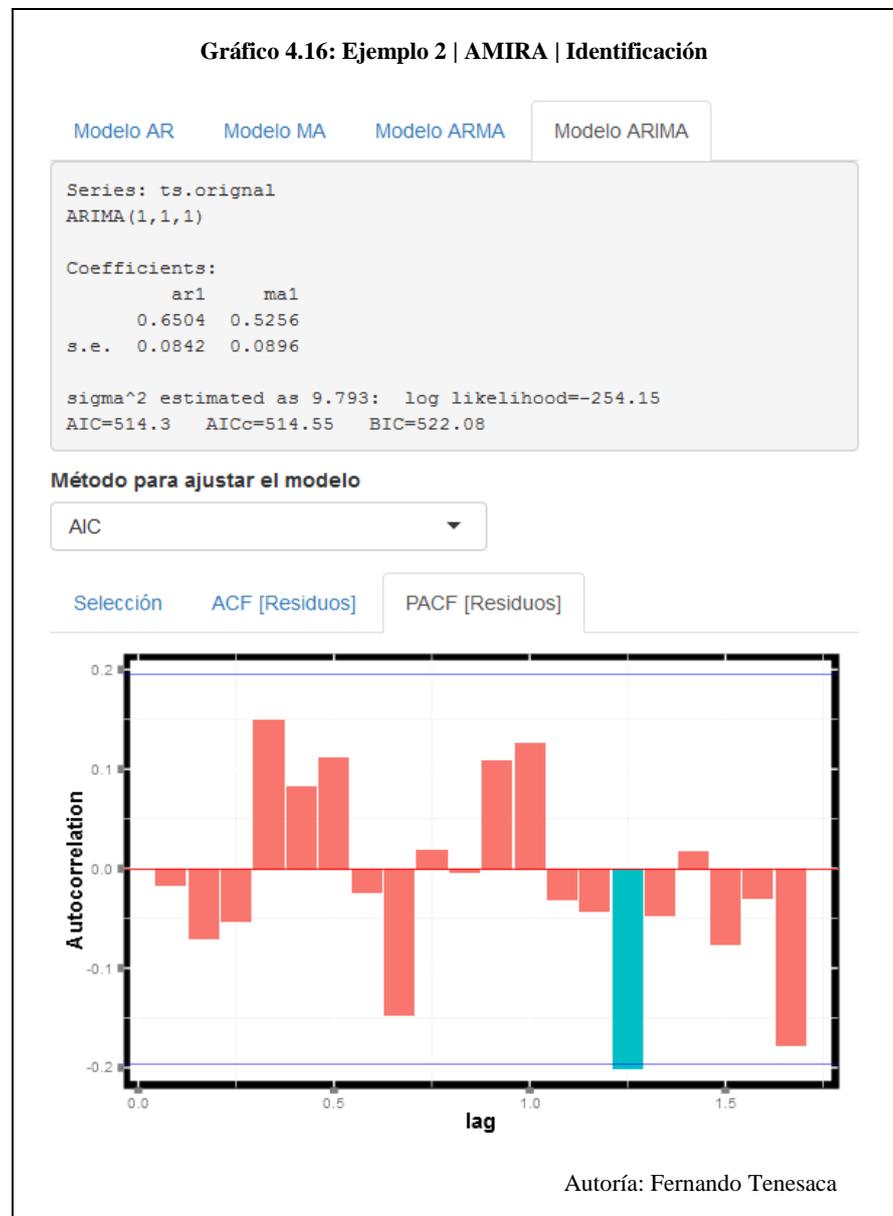
A continuación, AMIRA grafica la función de autocorrelación (ACF) en la que se puede apreciar un decrecimiento a cero y la función de autocorrelación parcial (PACF) que de manera preliminar sugiere un modelo autorregresivo de tercer orden para la serie.





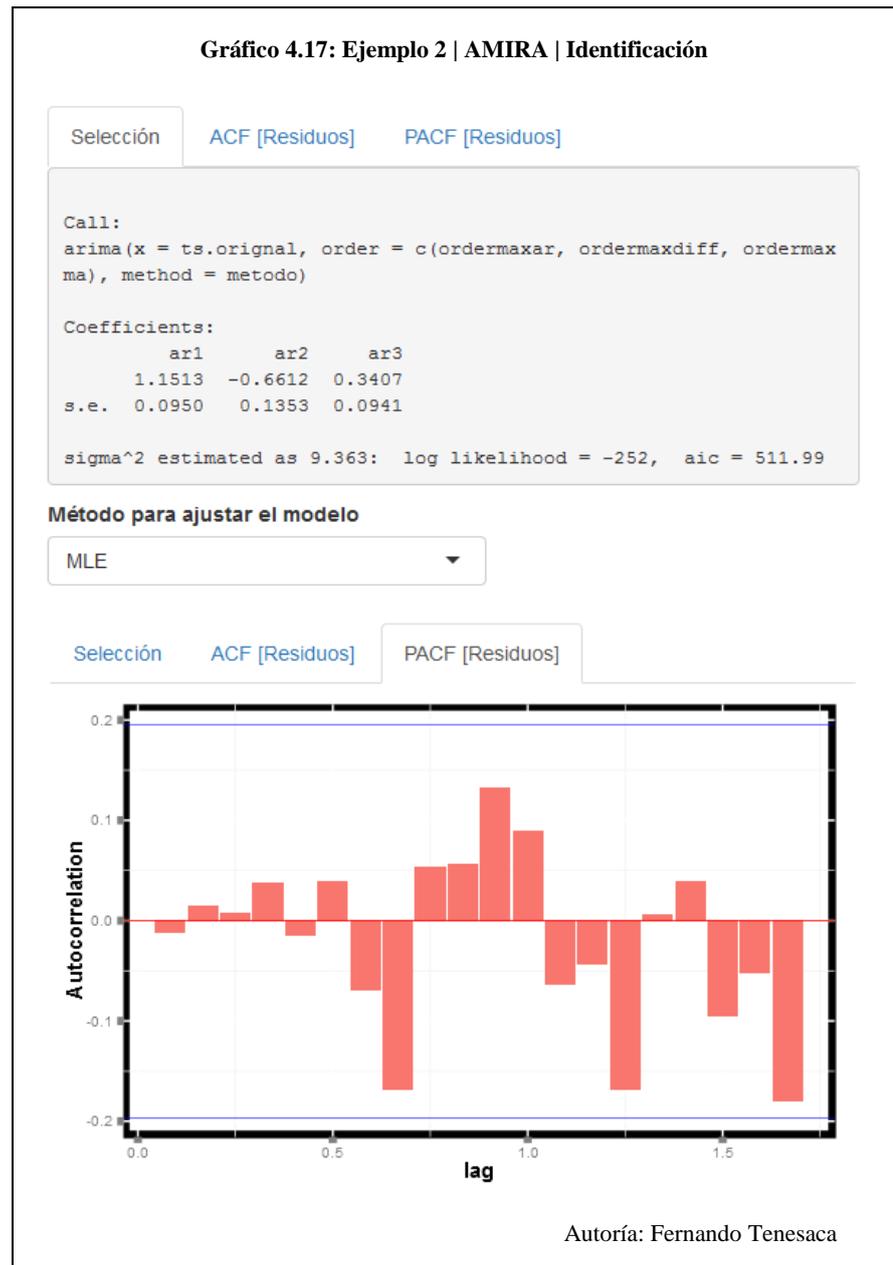
AMIRA identifica el modelo asociado a la serie temporal y también permite que el usuario pueda establecer diferentes modelos para contrastar con el sugerido por la aplicación.



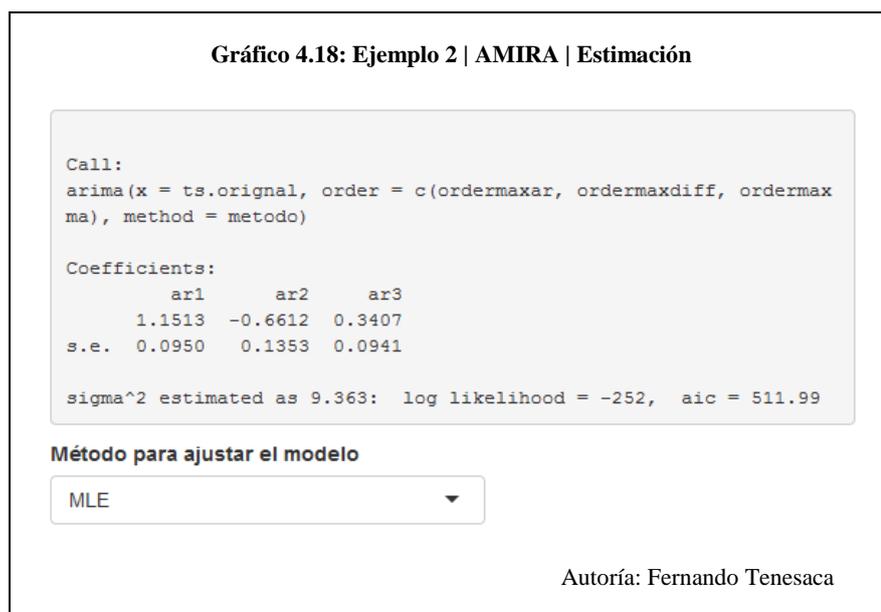


El modelo sugerido no siempre es el ideal y en este punto es importante el conocimiento del tema y la experiencia alrededor del origen de la fuente de datos por parte del usuario de la aplicación.

Como se mencionó previamente AMIRA permite que el usuario establezca diferentes modelos para contrastar con el sugerido por la aplicación.



Una vez identificado y seleccionado el modelo, AMIRA, procede a estimar los coeficientes para la ecuación que mejor se ajusta a los datos de la serie temporal.



Es importante la validación de resultados, por tanto, AMIRA analiza o diagnostica los residuos de la serie estimada en función del modelo seleccionado a través de la prueba de aleatoriedad o prueba de Ljung-Box, la misma comprueba si los residuos se distribuyen de forma independiente (hipótesis nula).

Gráfico 4.19: Ejemplo 2 | AMIRA | Análisis

Prueba de Aleatoriedad de los Residuos

Ljung-Box

Hipótesis:**H0:** Los datos SI se distribuyen de forma independiente.**H1:** Los datos NO se distribuyen de forma independiente.**Cálculo:**

El estadístico de prueba es:

[1] 0.0009887803

El valor-p de la prueba es:

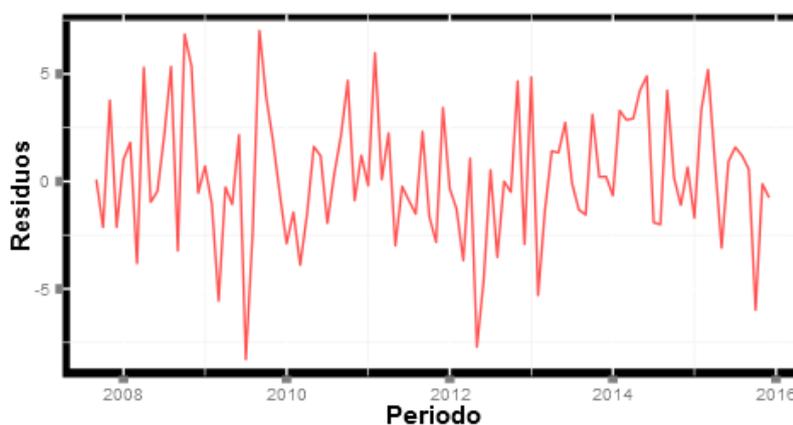
[1] 0.9749148

Decisión:

[1] "El valor-p de la prueba es mayor que 0.05 por tanto NO se rechaza la hipótesis nula H0"

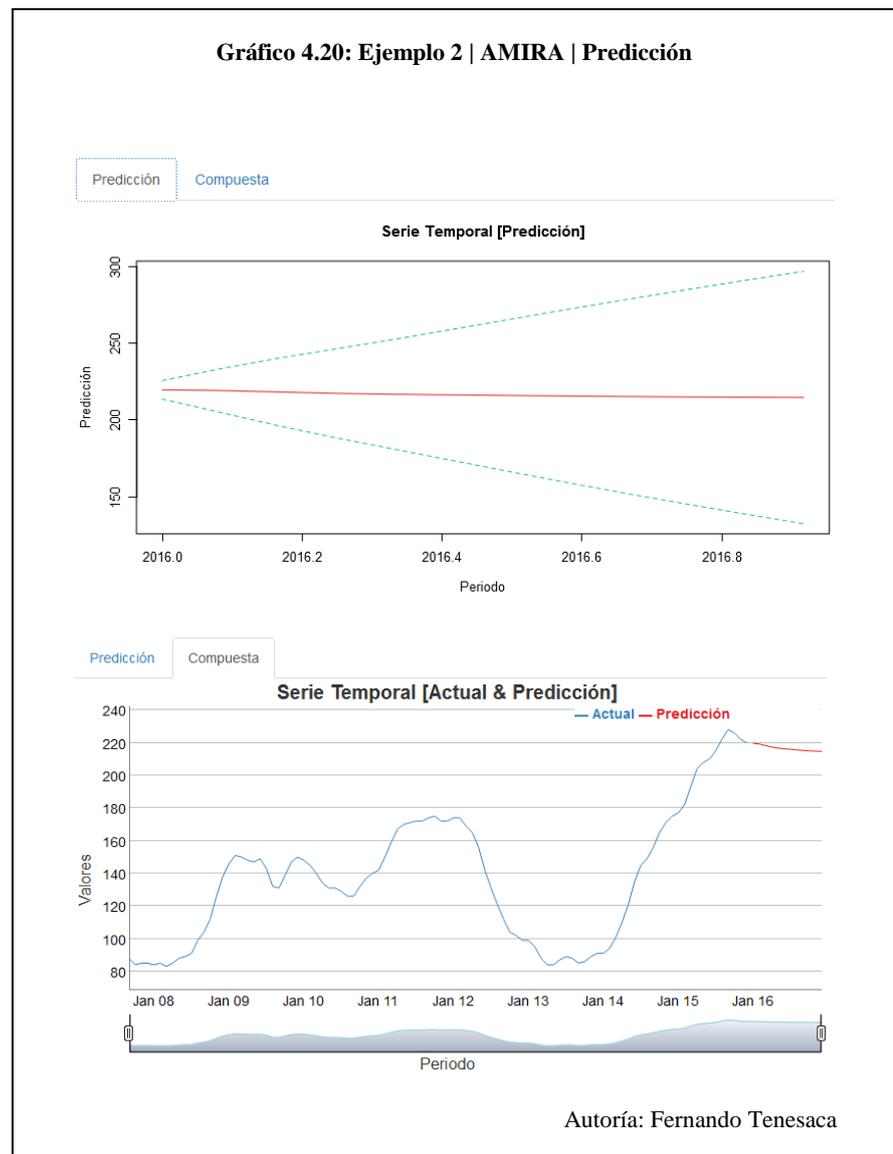
Conclusión:

[1] "Por consiguiente, NO hay suficiente evidencia estadística para rechazar la hipótesis nula H0; es decir, los residuos del modelo presentan aleatoriedad."

Residuos de la Serie Estimada

Autoría: Fernando Tenesaca

Al no existir suficiente evidencia estadística para rechazar la hipótesis nula, AMIRA predice y grafica los valores asociados a la serie estimada de acuerdo al modelo seleccionado como parte del proceso final del análisis de la serie temporal.



CONCLUSIONES Y RECOMENDACIONES

El desarrollo del presente proyecto de Materia Integradora, previo a la obtención del título de Ingeniero en Estadística Informática, ha permitido obtener las siguientes conclusiones y recomendaciones:

CONCLUSIONES

- ✓ El diseño de un asistente utilizando un lenguaje de programación y entorno de desarrollo de código abierto para el análisis de series temporales estacionarias hace posible que la comunidad académica se involucre en el desarrollo de AMIRA, para la ampliación del alcance original e implementación de nuevas características que mejoren la eficiencia y funcionalidad de la aplicación.
- ✓ La implementación de una aplicación web reactiva para el tratamiento de series temporales estacionarias como prototipo para un sistema integrado de análisis estadístico, a través de lenguajes de programación como R⁵, entornos de desarrollo como Rstudio⁶ y complementos como Shiny⁷, hace factible la modificación del código fuente para adaptar a AMIRA a nuevos requerimientos funcionales al centrar la potencial inversión que se gestione en capacitación sobre estas tecnologías.

- ✓ La aplicación de los criterios desarrollados por Box y Jenkins¹ como metodología de desarrollo permiten describir el comportamiento de AMIRA, a través de una serie de acciones o tareas que se ejecutan en cierto orden y que estructuran un flujo de trabajo eficaz para identificar, estimar y verificar modelos que se ajustan a las series temporales estacionarias.

- ✓ La abstracción de los detalles estadísticos y de programación permite a los usuarios, con formación estadística y conocimientos en el análisis de series de tiempo, a través de los objetos reactivos para carga y manipulación de datos que provee AMIRA enfocarse en el análisis específico para pronosticar situaciones de incertidumbre.

- ✓ El cumplimiento de los principales resultados del programa de estudios de la carrera de Ingeniería en Estadística Informática acorde con la normativa vigente permiten:
 - Aplicar Matemática, Estadística e Informática para resolver problemas relacionados con la profesión.
 - Aplicar técnicas y métodos estadísticos al tratamiento de pequeños y grandes volúmenes de datos para tomar decisiones, efectuar pronósticos, describir situaciones o procesos, etc.
 - Desarrollar “software” utilizando modelos matemáticos para su aplicación en el campo estadístico, financiero, actuarial, demográfico, optimización, etc.

RECOMENDACIONES

- ✓ Ampliar el alcance original del proyecto integrador e implementar nuevas características que mejoren la eficiencia y funcionalidad de la aplicación aprovechando las ventajas del lenguaje de programación y entorno de desarrollo de código abierto que hacen factible la modificación del código fuente para adaptar a AMIRA a nuevos requerimientos funcionales.
- ✓ Reforzar los programas de estudio de las materias de formación profesional para incluir el desarrollo de proyectos que utilicen componentes de programación reactiva en los cuales se apliquen los criterios teóricos y metodológicos impartidos en clases.
- ✓ Fomentar la transversalidad educativa para la consecución de este tipo de proyectos, lo que contribuye a los aprendizajes significativos de los estudiantes desde la conexión de los conocimientos disciplinares con los temas y contextos sociales, culturales y éticos presentes en su entorno.
- ✓ Generar un proyecto de investigación a nivel institucional para desarrollar un sistema integrado de análisis estadístico en el cual AMIRA y así como otros emprendimientos en el área de estudio puedan publicarse para beneficio de la comunidad académica.

BIBLIOGRAFÍA

- [1] G. Box and G. Jenkins, *Time Series Analysis: Forecasting and Control*, San Francisco: Holden-Day, 1970.
- [2] E. Dijkstra, *The Humble Programmer*, Communications of the ACM, vol. 15, pp. 859-866, Oct. 1972.
- [3] P. Naur and B. Randell, *Software Engineering: Report of a conference*, NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968, Brussels, Scientific Affairs Division, 1969.
- [4] J. Radatz, IEEE Standard Glossary of Software Engineering Terminology. IEEE std 610.12-1990, 1990.
- [5] Core Team, *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria, <http://www.R-project.org>, 2016.
- [6] RStudio Team, *RStudio: Integrated Development for R*. RStudio, Inc., Boston, MA, <http://www.rstudio.com>, 2016.
- [7] W. Chang, J. Cheng, J. Allaire, Y. Xie and J. McPherson, *shiny: Web Application Framework for R*, R package version 0.13.0., <http://CRAN.R-project.org/package=shiny>, 2016.
- [8] B. W. Lewis, *The websockets Package*, 1 Nov. 2001.
- [9] M. Otto, *Bootstrap*, <http://getbootstrap.com>, 12 Feb. 2016.
- [10] C. Beeley, *Web Application Development with R Using Shiny*, Reino Unido: Packt Publishing, 2013.
- [11] M. P. González, *Análisis de Series Temporales: Modelos ARIMA*, SARRIKOON, <http://hdl.handle.net/10810/12492>, 2009.
- [12] J. A. Mauricio, *Introducción al Análisis de Series Temporales*, España: UCM, <https://www.ucm.es/data/cont/docs/518-2013-11-11-JAM-IASST-Libro.pdf>, 12 Feb. 2016.
- [13] Ljung and G. Box, *On a measure of lack of fit in time series models*, Biometrika, vol. 65, pp. 297-303, Feb. 1978.

- [14] J. Moreno, *Programación Reactiva*, <http://www.ceiba.com.co>, 12 Feb. 2016.
- [15] Al-Ahmadgaid Asaad, R: *Explore ARIMA(2,2,2) subclass family on Shiny*, Analysis with Programming, <http://alstatr.blogspot.com>, 12 Feb. 2016.
- [16] F. Ruiz, *Arquitectura Física del Sistema*, España: Universidad de Cantabria, 2010.
- [17] J. Chicano y A. Ruiz, *Aplicación Web para Predicción de Series Temporales*, España: Universidad de Málaga, 2010
- [18] S. Jaiswar, *Platform as a Service (PaaS) Demystified*, SYNTEL, 2012.
- [19] G. Golemund. *Interactive Data Display*, RStudio, 2014
- [20] A. Kipp, *Getting started with shinyapps.io*, RStudio, <http://shiny.rstudio.com>, 12 Feb. 2016.
- [21] K. Fakhroutdinov, *Web Application Clusters*, <http://www.uml-diagrams.org>, 12 Feb. 2016.
- [22] M. Niño, *El entorno RStudio y el framework Shiny para el desarrollo de aplicaciones en R*, <http://www.mikelnino.com>, 12 Feb. 2016.
- [23] R. Alabern, *Introducción a Shiny*, Rstudio, http://rstudio-pubs-static.s3.amazonaws.com/21373_8af3d3634b97461089c8a76659982915.html#reactive-isolate, 12 Feb. 2016.
- [24] Rstudio, *Scaling and Performance Tuning with shinyapps.io*, Rstudio, <http://shiny.rstudio.com/articles/scaling-and-tuning.html>, 12 Feb. 2016.
- [25] M. Dettling, *Applied Time Series Analysis*, Switzerland: ETH Zurich, <http://stat.ethz.ch/education/semesters/ss2014/atsa> , 12 Feb. 2016

ANEXOS

A. FUNCIÓN DE AUTOCORRELACIÓN PARCIAL¹²

La autocorrelación parcial de orden k ($k > 0$) de un proceso estocástico (Y_t) estacionario se representa con el símbolo ϕ_{kk} y se define como el parámetro ϕ_{kk} en la regresión

$$\tilde{Y}_t = \phi_{k1}\tilde{Y}_{t-1} + \phi_{k2}\tilde{Y}_{t-2} + \cdots + \phi_{kk}\tilde{Y}_{t-k} + U_t$$

donde $\tilde{Y}_{t-k} \equiv Y_{t-i} - \mu_Y$ $i = 0, 1, \dots, k$ y U_t es independiente de Y_{t-i} para $i \geq 1$.

La regresión anterior se puede escribir como $Y_t = \phi_{k0} + \phi_{k1}\tilde{Y}_{t-1} + \phi_{k2}\tilde{Y}_{t-2} + \cdots + \phi_{kk}\tilde{Y}_{t-k} + U_t$, donde $\phi_{k0} \equiv (1 - \phi_{k1} - \phi_{k2})\mu_Z$, lo que implica que $\Delta Y_t = \phi_{kk} \times \Delta Y_{t-k}$ cuando $\Delta Y_{t-1} = \Delta Y_{t-2} = \cdots = \Delta Y_{t-k+1} = \Delta U_t = 0$; es decir, ϕ_{kk} representa el efecto parcial de Y_{t-k} sobre Y_t . Por lo tanto, ϕ_{kk} es una medida del grado de asociación lineal entre dos componentes cualesquiera de Y_t separados entre sí por un retardo $k \geq 1$ dado (como Y_t e Y_{t-k}) que no se debe a la posible correlación entre cada uno de ellos y todos los componentes de Y_t que se encuentran entre ambos ($Y_{t-1}, Y_{t-2}, \dots, Y_{t-k+1}$).

A partir de la regresión anterior, se puede comprobar que $\phi_{kk} = |A_k|/|B_k|$, donde

$$A_k = \begin{bmatrix} 1 & \rho_1 & \cdots & \rho_{k-2} & \rho_1 \\ \rho_1 & 1 & \cdots & \rho_{k-3} & \rho_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \cdots & \rho_1 & \rho_k \end{bmatrix}, \quad B_k = \begin{bmatrix} 1 & \rho_1 & \cdots & \rho_{k-2} & \rho_{k-1} \\ \rho_1 & 1 & \cdots & \rho_{k-3} & \rho_{k-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \cdots & \rho_1 & 1 \end{bmatrix}$$

La matriz B_k es una matriz de autocorrelaciones. La matriz A_k es idéntica a B_k excepto porque la última columna de B_k se reemplaza por el vector columna $[\rho_1, \rho_2, \dots, \rho_k]^T$. En particular, para $k = 1, 2, 3$ resulta que

$$\phi_{11} = \rho_1, \quad \phi_{22} = \frac{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & \rho_2 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2}, \quad \phi_{33} = \frac{\begin{vmatrix} 1 & \rho_1 & \rho_1 \\ \rho_1 & 1 & \rho_2 \\ \rho_2 & \rho_1 & \rho_3 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_1 \\ \rho_2 & \rho_1 & 1 \end{vmatrix}},$$

de manera que los coeficientes de autocorrelación parcial son funciones de los coeficientes de autocorrelación simple de un proceso estocástico estrictamente estacionario.

Considerada como una función del rezago k , la secuencia $(\phi_{kk}; k = 1, 2, \dots)$ se denomina la función de autocorrelación parcial (PACF, del inglés *Partial AutoCorrelation Function*) del proceso Y_t . Se denota por

$$\phi_{kk} = \frac{|A_k|}{|B_k|}, \quad k = 1, 2, 3, \dots; \quad \phi_{11} = \rho_1$$

B. CASOS DE USO

Nombre:	Ingresar a la aplicación
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe ingresar a la aplicación a través de un navegador.
Actores:	Usuario
Precondiciones:	El usuario debe disponer de un equipo portátil o de escritorio con acceso a internet o en su defecto instalado el lenguaje de programación, <i>R</i> ; el entorno de desarrollo integrado, <i>RStudio</i> ; el entorno de desarrollo, <i>Shiny</i> ; y, los complementos necesarios para su ejecución de forma local.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario abre una ventana del navegador. 2. El usuario en la barra de dirección escribe el URL <i>http://espol.shinyapps.io/amira</i> y presiona la tecla <i>Enter</i>.
Flujo Alternativo:	<ol style="list-style-type: none"> 1A. El usuario carga el entorno de desarrollo integrado, <i>RStudio</i>. 2A. El usuario ejecuta la aplicación <i>AMIRA</i> a través del comando <i>runApp("AMIRA")</i>.
Poscondiciones:	El usuario ha ingresado a la aplicación y se encuentra activo en la sección <i>Datos</i> en el panel derecho (main panel).

Nombre:	Cargar archivo de datos
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe cargar el archivo de datos de la serie temporal.
Actores:	Usuario/Aplicación
Precondiciones:	El usuario debe disponer de un archivo en formato texto (<i>text/csv</i>) con los datos de la serie temporal estacionaria listados por filas.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario presiona el botón <i>Examinar</i> dispuesto en la barra lateral izquierda (sidebar panel) 2. El usuario selecciona la ruta de almacenamiento. 3. El usuario selecciona el archivo en formato texto (<i>text/csv</i>). 4. El usuario presiona el botón <i>Abrir</i> para cargar el archivo de datos. 5. El sistema procesa el archivo de datos y lo presenta en el panel derecho (main panel).
Flujo Alternativo:	3A. El usuario selecciona el archivo en formato texto (<i>text/csv</i>) y hace " <i>doble click</i> " con el botón izquierdo del mouse para cargar el archivo de datos.
Poscondiciones:	El usuario ha cargado el archivo de datos y se encuentra activo en la sección <i>Datos</i> en el panel derecho (main panel).

Nombre:	Establecer los parámetros de la serie temporal
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe establecer los parámetros asociados con la frecuencia y el año de inicio de la serie temporal.
Actores:	Usuario/Aplicación
Precondiciones:	El usuario debe conocer que característica representa la variable que se analiza.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario hace “click” con el botón izquierdo del mouse para seleccionar y cambiar a la sección <i>Serie</i> en el panel derecho (main panel). 2. La aplicación grafica en el panel derecho (main panel) la serie temporal y los componentes de control con los parámetros por defecto. 3. El usuario selecciona la frecuencia en el objeto dispuesto en la barra lateral izquierda (sidebar panel). 4. La aplicación actualiza y grafica en el panel derecho (main panel) la serie temporal junto con los componentes de control con el nuevo parámetro. 5. El usuario ingresa el año en el objeto dispuesto en la barra lateral izquierda (sidebar panel). 6. La aplicación actualiza y grafica en el panel derecho (main panel) la serie temporal junto con los componentes de control con el nuevo parámetro.
Flujo Alternativo:	3A. El usuario selecciona la frecuencia en el objeto dispuesto en la barra lateral izquierda (sidebar panel); si la frecuencia es <i>Mensual</i> debe seleccionar adicionalmente el mes o, si la frecuencia es <i>Trimestral</i> debe seleccionar adicionalmente el trimestre.
Poscondiciones:	El usuario ha establecido los parámetros de la serie temporal y se encuentra activo en la sección <i>Serie</i> en el panel derecho (main panel).

Nombre:	Transformar la serie temporal
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe aplicar las transformaciones necesarias cuando la serie temporal no presenta las características requeridas para el análisis.
Actores:	Usuario/Aplicación
Precondiciones:	El usuario debe conocer la teoría asociada con el análisis de series temporales estacionarias.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario hace “click” con el botón izquierdo del mouse en el objeto dispuesto en la barra lateral izquierda (sidebar panel) para activar la opción de transformación de la serie temporal y decide combinar los tipos de transformaciones en caso de requerirse. <ol style="list-style-type: none"> 1.1. El usuario hace “click” con el botón izquierdo del mouse en el objeto dispuesto en la barra lateral izquierda (sidebar panel) para activar la transformación logarítmica de la serie temporal. 1.2. La aplicación actualiza y grafica en el panel derecho (main panel) la serie temporal junto con los componentes de control con la transformación escogida. 1.3. El usuario selecciona el orden de diferenciación ordinaria en el objeto dispuesto en la barra lateral izquierda (sidebar panel). 1.4. La aplicación actualiza y grafica en el panel derecho (main panel) la serie temporal junto con los componentes de control con la transformación escogida.
Flujo Alternativo:	No está definido
Poscondiciones:	El usuario ha aplicado en caso de requerirse las transformaciones a la serie temporal y se encuentra activo en la sección <i>Serie</i> en el panel derecho (main panel).

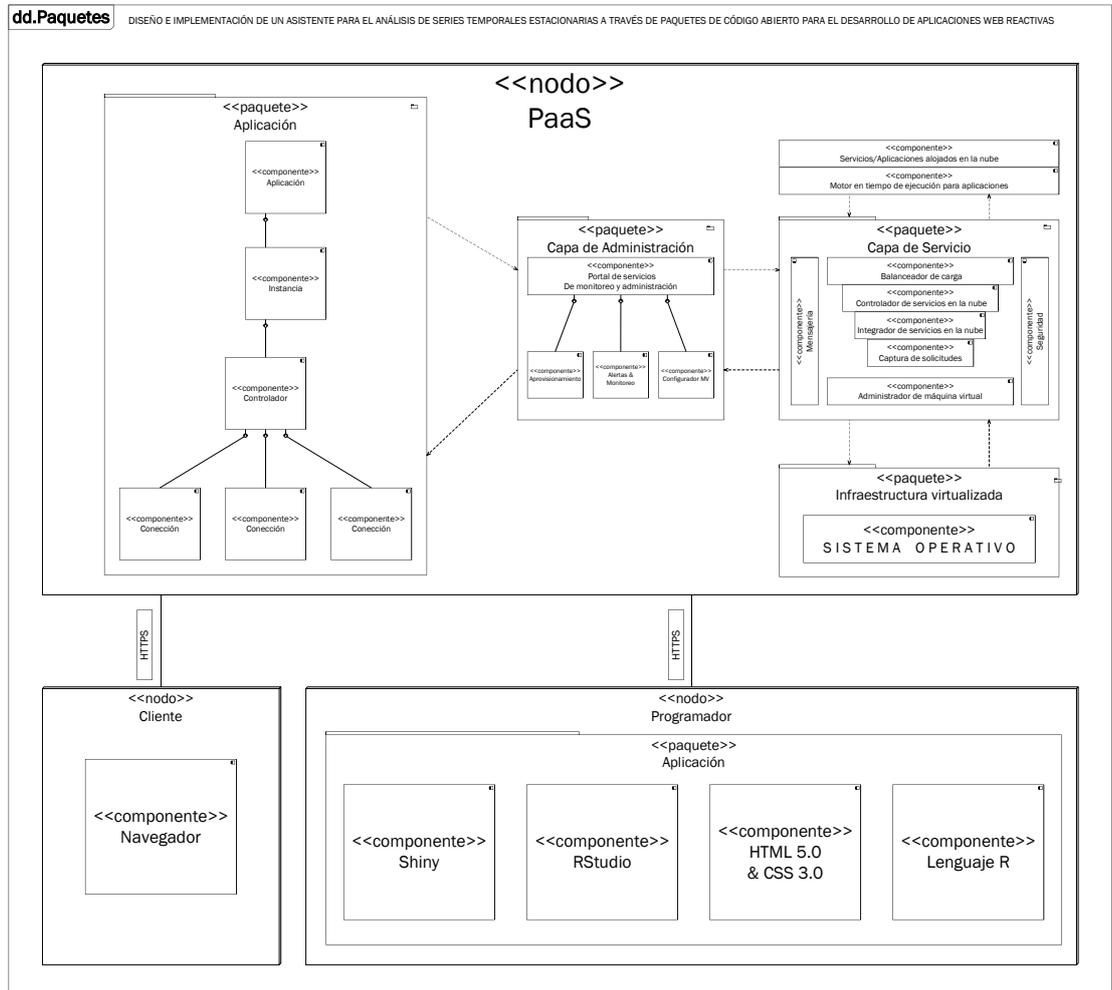
Nombre:	Identificar el modelo de la serie temporal
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe identificar el modelo de la serie temporal.
Actores:	Usuario/Aplicación
Precondiciones:	El usuario debe conocer la teoría asociada con el análisis de series temporales estacionarias.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario hace “click” con el botón izquierdo del mouse para seleccionar y cambiar a la sección <i>Modelos</i> en el panel derecho (main panel). 2. La aplicación presenta la información en el panel derecho (main panel) junto con los componentes de control asociados al orden de los modelos por defecto. 3. El usuario hace “click” con el botón izquierdo del mouse en los objetos dispuestos en la barra lateral izquierda (sidebar panel) para seleccionar el orden de los modelos y activar la transformación logarítmica en caso de requerirse. 4. La aplicación actualiza y presenta en el panel derecho (main panel) la información junto con los componentes de control asociados al orden de los modelos seleccionados.
Flujo Alternativo:	3A. El usuario hace “click” con el botón izquierdo del mouse en el objeto dispuesto en el panel derecho (main panel) y selecciona el método para ajustar el modelo en caso de requerirse.
Poscondiciones:	El usuario ha identificado el modelo de la serie temporal y se encuentra activo en la sección <i>Modelos</i> en el panel derecho (main panel).

Nombre:	Estimar los coeficientes del modelo
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe estimar los coeficientes del modelo escogido previamente en función del orden del modelo que mejor describe a la serie temporal.
Actores:	Usuario/Aplicación
Precondiciones:	El usuario debe conocer la teoría asociada con el análisis de series temporales estacionarias.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario hace “click” con el botón izquierdo del mouse para seleccionar y cambiar a la sección <i>Estimación</i> en el panel derecho (main panel). 2. La aplicación presenta la información en el panel derecho (main panel) asociada al orden del modelo por defecto. 3. El usuario hace “click” con el botón izquierdo del mouse en los objetos dispuestos en la barra lateral izquierda (sidebar panel) para seleccionar el orden del modelo escogido previamente y activar la transformación logarítmica en caso de requerirse. 4. La aplicación actualiza y presenta la información en el panel derecho (main panel) asociada al orden del modelo seleccionado.
Flujo Alternativo:	5. El usuario hace “click” con el botón izquierdo del mouse en el objeto dispuesto en el panel derecho (main panel) y selecciona el método para ajustar el modelo en caso de requerirse.
Poscondiciones:	El usuario ha estimado los coeficientes del modelo escogido previamente y se encuentra activo en la sección <i>Estimación</i> en el panel derecho (main panel).

Nombre:	Analizar los residuos del modelo
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe analizar los residuos del modelo escogido previamente y verificar si son independientes entre sí; es decir, si los residuos son completamente aleatorios.
Actores:	Usuario/Aplicación
Precondiciones:	El usuario debe conocer la teoría asociada con el análisis de series temporales estacionarias.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario hace “click” con el botón izquierdo del mouse para seleccionar y cambiar a la sección <i>Análisis</i> en el panel derecho (main panel). 2. La aplicación presenta y grafica en el panel derecho (main panel) la información asociada con el análisis de los residuos del modelo escogido previamente. 3. El usuario hace “click” con el botón izquierdo del mouse en los objetos dispuestos en el panel derecho (main panel) para seleccionar y cambiar la prueba de aleatoriedad de los residuos. 4. La aplicación actualiza, presenta y grafica en el panel derecho (main panel) la información asociada con el análisis de los residuos del modelo escogido previamente.
Flujo Alternativo:	No está definido
Poscondiciones:	El usuario ha analizado los residuos del modelo escogido previamente y se encuentra activo en la sección <i>Análisis</i> en el panel derecho (main panel).

Nombre:	Predecir los valores futuros de la serie temporal
Autor:	Fernando Tenesaca
Fecha:	2016-02-12
Descripción:	El usuario debe predecir los valores futuros de la serie temporal de acuerdo a los coeficientes del modelo escogido previamente.
Actores:	Usuario/Aplicación
Precondiciones:	El usuario debe conocer la teoría asociada con el análisis de series temporales estacionarias.
Flujo Normal:	<ol style="list-style-type: none"> 1. El usuario hace “click” con el botón izquierdo del mouse para seleccionar y cambiar a la sección <i>Predicción</i> en el panel derecho (main panel). 2. La aplicación presenta y grafica en el panel derecho (main panel) la información asociada con la predicción de los valores futuros de la serie temporal. 3. El usuario hace “click” con el botón izquierdo del mouse en los objetos dispuestos en el panel derecho (main panel) para seleccionar y cambiar la vista del gráfico. 4. La aplicación actualiza, presenta y grafica en el panel derecho (main panel) la información asociada con la predicción de los valores futuros de la serie temporal. 5. El usuario hace “click” con el botón izquierdo del mouse en los objetos dispuestos en la barra lateral izquierda (sidebar panel) para seleccionar el número de periodos para predecir. 6. La aplicación actualiza, presenta y grafica en el panel derecho (main panel) la información asociada con la predicción de los valores futuros de la serie temporal.
Flujo Alternativo:	No está definido
Poscondiciones:	El usuario ha predicho los valores futuros de la serie temporal y se encuentra activo en la sección <i>Predicción</i> en el panel derecho (main panel).

C. PLATAFORMA COMO UN SERVICIO



D. CÓDIGO FUENTE

ui.R

Archivo que describe la interfaz gráfica, que en este caso comprende una página web que funciona como cliente de la aplicación.

Repositorio GitHub: <https://github.com/fthenesaca/amira>

server.R

Archivo que describe las instrucciones para el flujo de datos y control de cambios de la aplicación desde y hacia la interfaz gráfica.

Repositorio GitHub: <https://github.com/fthenesaca/amira>