

GRADO EN INGENIERÍA EN ORGANIZACIÓN
INDUSTRIAL

TRABAJO FIN DE GRADO

*ANÁLISIS DEL COMPORTAMIENTO DEL
MERCADO BURSÁTIL MEDIANTE
MODELOS ARIMA*

Alumno: Pablo Nieves Moina

Director: Patxi Ruiz de Arbulo López

Curso: 2018/2019

Fecha: 22/07/2019

Quiero agradecer la ayuda prestada para la consecución de este trabajo, en primer lugar a mi madre, Susana, por su incansable apoyo durante todo el grado. A mi amigo Diego por su aportación y apoyo, ayudándome con sus profundos conocimientos del mercado de valores. A mi pareja, Eider, por sus continuas revisiones al trabajo, ayudándome con la estética y dándome innumerables consejos. A Verónica de Prado, antigua alumna de esta universidad, por su ayuda prestada a la hora de facilitarme la elección del software. Y por último, a mi director del trabajo y profesor, Patxi Ruiz de Arbulo, por su atención siempre que la he necesitado y su entusiasmo en la realización del éste.

RESUMEN

Este trabajo tiene el objetivo de analizar el comportamiento del precio de los activos bursátiles para examinar sus posibilidades de predicción mediante modelos matemáticos. En él, a través de modelos ARIMA aplicados por medio del software R, se va a tratar de comprobar o desmentir la hipótesis del mercado eficiente y el paseo aleatorio, y a la par que se va a evaluar la utilidad de ARIMA para la predicción. Para el estudio se analizarán dos índices: el estadounidense S&P 500 y el español IBEX 35.

Palabras Clave: *“Predicción”, “R”, “ARIMA”, “Series Temporales”, “Mercado Bursátil”, “Tendencia”, “Inversión”, “Aleatoriedad”.*

ABSTRACT

The main purpose of this work is to analyze the behaviour of stock assets prices, to check their possibilities of being predicted by mathematical models. In it, using ARIMA models in the R software, is going to be tried to prove or to reject the hypothesis of the market efficiency and the random walk, and, meanwhile, is going to be examined the capability of prediction of ARIMA. In the study two indexes are going to be analyzed: The american S&P 500 and the spanish IBEX 35

Key Words: *“Prediction”, “R”, “ARIMA”, “Time Series”, “Stock Market”, “Trend”, “Investment”, “Randomness”.*

LABURPENA

Proiektu honen helburua burtsa-aktiboen salneurrien ibilkera aztertzea da, haien iragarpen posibilitateak aztertzeko eredu matematikoen bitartez. Honetan, ARIMA eredu bidez R softwaren bitartez erabilitakoak, eraginkor merkatuaren eta zorizko pasatzearen hipotesiak baieztatzen edo ezeztatzen saiatuko da, eta era berean ARIMA-ren iragartzeko utilitatea ebaluatuko da. Azterlanarako bi indize aztertuko dira: estutubatuarra S&P 500 eta espaniarra IBEX 35.

Hitz Gakoak: *“Iragarpena”, “R”, “ARIMA”, “Demborazko Saila”, “Burtsa Merkatua”, “Joera”, “Ibertsio”, “Auzazkotasun”*

CONTENIDO

1.	INTRODUCCIÓN	9
2.	OBJETIVOS Y ALCANCE DEL TRABAJO	11
3.	ESTADO DEL ARTE. OTROS MÉTODOS DE PREDICCIÓN DEL MERCADO DE VALORES	12
3.1.	RAZÓN DE SER DEL SISTEMA FINANCIERO	12
3.2.	EL MERCADO DE VALORES. ¿QUÉ ES Y CÓMO FUNCIONA?	13
3.3.	ÍNDICES BURSÁTILES. S&P 500 E IBEX 35	14
3.4.	PREDICCIÓN DE LAS TENDENCIAS: LAS SUBIDAS, Y LAS BAJADAS	15
3.5.	MERCADOS EFICIENTES Y PASEOS ALEATORIOS	19
4.	DESARROLLO TEÓRICO DE LOS MODELOS E IMPLEMENTACIÓN EN R.	22
4.1.	ELECCIÓN DE SOFTWARE PARA LLEVAR A CABO LA PRÁCTICA.	22
4.2.	R. ¿QUÉ ES? ¿CÓMO FUNCIONA?	23
4.3.	SERIES TEMPORALES EN R	30
4.3.1.	Manejar una serie temporal en R	30
4.3.2.	Tendencia en series temporales	31
4.3.3.	Modelos básicos de series temporales	37
5.	METODOLOGÍA CON MODELOS ARIMA EN R	51
5.1.	DESARROLLO TEÓRICO DE ARMA	51
5.2.	IDENTIFICACIÓN DEL MODELO	52
5.3.	CRITERIOS <i>AIC</i> Y <i>BIC</i>	54
5.4.	ANÁLISIS DE RESIDUOS	56
5.5.	MODELOS ARIMA AL COMPLETO	58
5.5.1.	ARIMA	58
5.5.2.	SARIMA	59
6.	DESARROLLO PRÁCTICO	63
6.1.	EJEMPLO DE PREDICCIÓN DE UNA SERIE. PIB DE LA CAPV	63
6.2.	EJECUCIÓN DE PREDICCIONES S&P500 E IBEX 35	75
6.2.1.	S&P 500	76
6.2.2.	IBEX 35	89
7.	CONCLUSIONES	102
8.	BIBLIOGRAFÍA	105
9.	ANEXO I- Paquetes de funciones instalados y utilizados durante el trabajo.	108
10.	ANEXO II- Series utilizadas en los ejemplos del trabajo:	110

11. ANEXO III- Código utilizado en el cuarto y quinto apartado para explicar el funcionamiento de R y las bases teóricas: 113
12. ANEXO IV- Código utilizado en la parte práctica: 123

ÍNDICE DE FIGURAS

Figura 1. Componentes de un sistema financiero	13
Figura 2. Esquema del funcionamiento de una función.....	27
Figura 3. Paquetes más importantes usados durante el trabajo.....	30
Figura 4. Gráfico de la temperatura global media del planeta.	32
Figura 5. Gráfico de la temperatura global media del planeta diferenciada.	33
Figura 6. Gráfico de la tasa de desempleo estadounidense entre 2012 y 2017 con la tendencia lineal eliminada.....	35
Figura 7. Gráfico de la tasa de desempleo estadounidense entre 2012 y 2017 con tendencia lineal y estacional eliminada.	35
Figura 8. Gráfico del número de desempleados estadounidenses desde 1950 hasta 1980.	36
Figura 9. Gráfico del número de desempleados estadounidenses desde 1950 hasta 1980 con transformación logarítmica aplicada.	37
Figura 10. Gráfico de dispersión entre el índice alemán DAX y el francés CAC.....	39
Figura 11. Autocorrelograma de la serie de temperaturas globales.	41
Figura 12. Gráfico de una serie simulada de ruido blanco con 50 datos.	43
Figura 13. Esquema de la relación del modelo de ruido blanco con el del paseo aleatorio.....	44
Figura 14. Comparativa de gráficos de paseos aleatorios con distintas medias de ruido blanco.....	45
Figura 15. Comparativa de autocorrelogramas normales de series con comportamiento autorregresivo y series de comportamiento de media móvil.....	49
Figura 16. Tabla descriptiva de los comportamientos de los correlogramas en función al tipo de serie temporal.....	52
Figura 17. Comparativa de los comportamientos de los autocorrelogramas normal y parcial de los modelos AR(1), MA(1) y ARMA(1, 1).	54
Figura 18. Gráficos de evaluación de residuos para la serie del PIB del EEUU del paquete <i>astsa</i> aplicando un ARIMA(1, 0, 1).....	57
Figura 19. Autocorrelograma de la serie de temperaturas globales del paquete <i>astsa</i>	58
Figura 20. Gráfico de la función <i>birth</i> del paquete <i>astsa</i>	60
Figura 21. Comparativa de los autocorrelogramas de la serie <i>birth</i> del paquete <i>astsa</i> normal y diferenciada.....	61

Figura 22. Autocorrelograma de la serie <i>birth</i> del paquete <i>astsa</i> difeneciada de forma normal y estacional con <i>lag</i> =12.....	62
Figura 23. Vista preliminar de los datos del PIB de la CAPV en Excel previo a la importación a R.....	64
Figura 24. Ruta para importar datos de Excel a R.....	65
Figura 25. Gráfico de la evolución de los datos brutos del PIB de la CAPV desde 1995-1T hasta 2019-1T	67
Figura 26. Gráfico de la serie del PIB de la CAPV diferenciada lineal y estacionalmente.....	68
Figura 27. Autocorrelogramas normal y parcial de la serie del PIB de la CAPV diferenciada lineal y estacionalmente.....	69
Figura 28. Tabla descriptiva de los comportamientos de los correlogramas en función al tipo de serie temporal.....	70
Figura 29. Autocorrelogramas normal y parcial de la serie del PIB de la CAPV diferenciada lineal y estacionalmente destacando los <i>lag</i> estacionales.....	71
Figura 30. Gráficos de los residuos de la serie del PIB de la CAPV para un modelo ARIMA(0, 1, 2)xSARIMA(1, 1, 1).....	73
Figura 31. Gráfico de la serie del PIB de la CAPV con predicción hasta 2021-1T.....	75
Figura 32. Gráfico de los valores de cierre del primer día operativo de cada mes del S&P500 desde abril de 2011 hasta julio de 2019.	76
Figura 33. Autocorrelogramas normal y parcial de la serie mensual de los valores del S&P500 diferenciada.	77
Figura 34. Gráficos de los residuos de la serie mensual del S&P500 para un modelo ARIMA(0, 1, 0).	78
Figura 35. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie mensual del S&P500.	79
Figura 36. Gráfico de los valores de cierre de cada lunes del S&P500 desde el 4 de enero de 2016 hasta el 18 de diciembre de 2017.	80
Figura 37. Autocorrelogramas normal y parcial de la serie semanal de los valores del S&P500 diferenciada.	81
Figura 38. Gráficos de los residuos de la serie semanal del S&P500 para un modelo ARIMA(1, 1, 0).....	82
Figura 39. Gráficos de los residuos de la serie semanal del S&P500 para un modelo ARIMA(0, 1, 0).	82

Figura 40. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie semanal del S&P500.	84
Figura 41. Gráfico de los valores de cierre diario del S&P500 desde el 2 de enero de 2019 hasta el 24 de mayo de del mismo año.	85
Figura 42. Autocorrelogramas normal y parcial de la serie diaria de los valores del S&P500 diferenciada.	86
Figura 43. Gráficos de los residuos de la serie diaria del S&P500 para un modelo ARIMA(0, 1, 0).	87
Figura 44. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie diaria del S&P500.	88
Figura 45. Gráfico de los valores de cierre del primer día operativo de cada mes del IBEX 35 desde abril de 2011 hasta julio de 2019.	90
Figura 46. Autocorrelogramas normal y parcial de la serie mensual de los valores del IBEX 35 diferenciada.	90
Figura 47. Gráficos de los residuos de la serie mensual del IBEX 35 para un modelo ARIMA(0, 1, 0).	91
Figura 48. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie mensual del IBEX 35.	93
Figura 49.Figura 36. Gráfico de los valores de cierre de cada lunes del IBEX 35 desde el 4 de enero de 2016 hasta el 18 de diciembre de 2017.	94
Figura 50. Autocorrelogramas normal y parcial de la serie semanal de los valores del IBEX 35 diferenciada.	94
Figura 51. Gráficos de los residuos de la serie semanal del IBEX 35 para un modelo ARIMA(0, 1, 0).	95
Figura 52. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie semanal del IBEX 35.	97
Figura 53. Gráfico de los valores de cierre diario del IBEX 35 desde el 2 de enero de 2019 hasta el 24 de mayo de del mismo año.	98
Figura 54. Autocorrelogramas normal y parcial de la serie diaria de los valores del IBEX 35 diferenciada.	98
Figura 55. Gráficos de los residuos de la serie diaria del IBEX 35 para un modelo ARIMA(0, 1, 0).	99
Figura 56. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie diaria del IBEX 35.	101

Figura 57. Gráfico comparativo de la serie real del IBEX 35 en la etapa de la crisis de 2008 junto con una predicción a partir del momento más alto de la burbuja mediante modelos ARIMA de paseos aleatorios. 103

1. INTRODUCCIÓN

El sistema financiero global no ha parado de crecer desde los inicios del comercio. A lo largo de la historia se han ido sucediendo importantes saltos en lo que se refiere al desarrollo de éste, desde la creación del dinero con patrón oro, hasta el nacimiento del mercado bursátil y de los casi infinitos elementos que lo componen. Sin embargo, el verdadero incremento de su volumen e importancia a escala humana se ha dado en la etapa en la que ahora vivimos, la globalización. Con este fenómeno, el número de participantes, de entes y personas interesados en tomar parte en el mercado ha crecido exponencialmente.

Todo esto ha generado la necesidad, cada vez más notable, de descubrir métodos que predigan los valores futuros de los activos que participan en él. Con ello, primero el análisis de gráficos, más tarde modelos matemáticos, pasando por los populares métodos de análisis técnico y fundamental, se han hecho populares entre los distintos tipos de inversores alrededor del globo, deseosos de obtener una rentabilidad por medio de un método que sea capaz de reducir el riesgo de la inversión hasta un valor residual.

Bajo este marco, la gran diversidad de técnicas ha traído consigo el debate de qué clase de modelos es mejor para reducir riesgos. No son pocos los que afirman, que, por ejemplo, como defiende el análisis técnico, solo con la simple observación del gráfico de precio histórico de un activo se pueden fijar unos máximos y unos mínimos valores entre los que va a oscilar el precio, o determinar si el precio va a comenzar a subir o bajar observando patrones que ha tenido la trazada de éste en el pasado. Por otra parte, también existen teorías, como la de los mercados eficientes y los paseos aleatorios, que afirman que bajo unas determinadas condiciones, como pueden ser las del mercado bursátil, la valoración de activos o el análisis de gráficos no tienen ningún sentido, ya que toda la información existente sobre el activo en cuestión está reflejada en su precio y se actualiza de forma casi inmediata, eficiente, bajo perturbaciones que desde una perspectiva matemática pueden considerarse aleatorias.

Por esto, se puede afirmar que no existe para nada un consenso en la definición de la forma en la que se comporta el mercado, ni siquiera entre los profesionales dedicados a la inversión o al *trading*, lo que puede suponer un verdadero quebradero

de cabeza para el inversor individual que se está decidiendo a invertir entre diferentes fondos de inversión que utilizan diferentes metodologías.

Tratando de ofrecer una solución a este problema se han realizado numerosos estudios, algunos más concluyentes que otros, sobre la eficiencia de los mercados, y, a pesar de esto, tampoco se ha conseguido cerrar el debate, aunque todo parece indicar que las bolsas de los países occidentales sí que tienen un comportamiento bastante eficiente.

Es aquí donde se pueden integrar los modelos ARIMA para resolver esta incógnita. Los modelos ARIMA (*Autoregressive Integrated Moving Average- Autorregresivos Integrados de Media Móvil*) son modelos matemáticos sobre series temporales que son capaces de ofrecer predicciones, aislando la tendencia, la estacionalidad y el error de los datos. Además de esto, descubriendo el modelo que mejor se adapta a la serie, también se puede analizar la dependencia del precio en un determinado momento con sus valores y variaciones pasados, lo que servirá para concluir si las series temporales analizadas, en este caso series temporales de valores de índices, tienen un comportamiento acorde a mercados eficientes o si guardan una correlación de alguna forma con su pasado.

2. OBJETIVOS Y ALCANCE DEL TRABAJO

El desarrollo del trabajo va a consistir en un análisis del comportamiento de dos series temporales de índices del mercado bursátil, como son el S&P 500 (*Standard and Poors 500*) o el IBEX35 (*Índice Bursátil Español*). Para la realización de éste, se explicará el marco teórico en el que está encuadrado, tanto en lo referido al mercado bursátil como a los modelos matemáticos a utilizar, además de una explicación detallada del método de trabajo con el software elegido.

Siendo esto así, el método será llevar a cabo un análisis de ambos índices en tres resoluciones temporales distintas (mensual, semanal y diaria), con 90 datos cada una, y ver a qué tipo de modelo ARIMA se adaptan, para confirmar o desestimar en distintos grados la hipótesis del mercado eficiente y del paseo aleatorio. Paralelamente a esto, se realizarán predicciones de los 10 próximos datos para comprobar cómo funcionan este tipo de técnicas a la hora de predecir el futuro en diferentes situaciones, y se ofrecerá finalmente una conclusión de si son útiles para guiar una inversión o no.

De este modo, el objetivo principal del trabajo será constatar o desmentir la eficiencia de estos mercados mediante modelos ARIMA, mientras que se comprobará a su vez la capacidad analítica de los modelos matemáticos de este tipo para guiar inversiones reduciendo el riesgo.

Como objetivos secundarios destacan aprender a desarrollar e implementar estos modelos en un *software*, que habrá que aprender desde cero, además de profundizar en métodos estadísticos o econométricos, que pueden ser útiles para un futuro a corto plazo en el contexto del mercado laboral.

3. ESTADO DEL ARTE. OTROS MÉTODOS DE PREDICCIÓN DEL MERCADO DE VALORES

3.1. RAZÓN DE SER DEL SISTEMA FINANCIERO

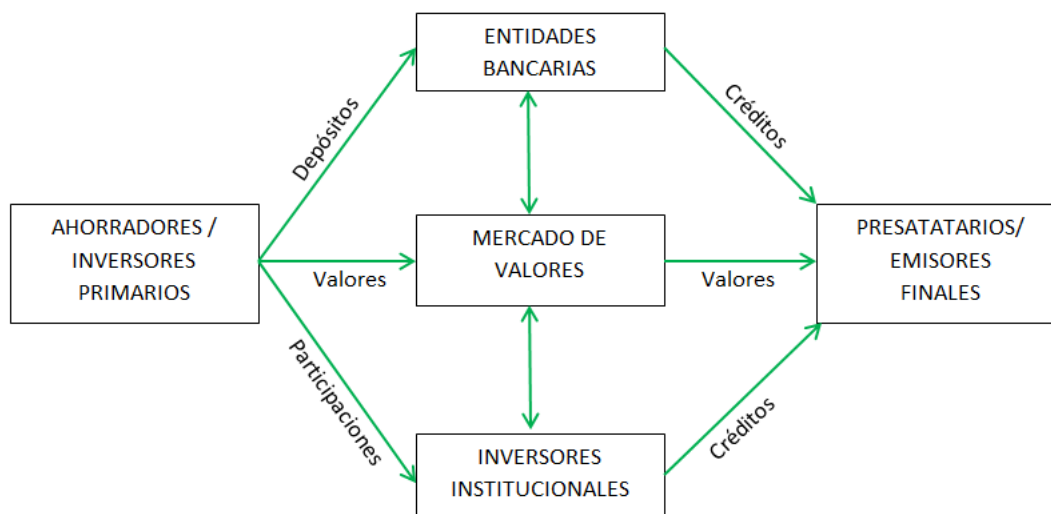
Antes de adentrarse en los métodos de predicción del mercado de valores, conviene aclarar en qué contexto se encuadra éste y en qué está basado su funcionamiento.

El mercado de valores no es más que una de las partes que ejerce de intermediario en el sistema financiero. Se define sistema financiero como “*Sistema que sirve para poner en contacto a los agentes económicos que quieren prestar o invertir, por tener fondos disponibles para ello, con los que necesitan dichos fondos para adquirir bienes, emprender nuevos negocios, o ampliar los ya existentes, para lo cual deben endeudarse o emitir valores*” (Zabala, 2019) . Dicho de otra forma, todo el sistema que permite que la financiación circule de manos de los que poseen el capital a manos de quien lo necesita es parte del sistema financiero, además de las partes emisora y receptora de éste.

Se pueden distinguir tres agentes principales que ejercen de intermediarios en un sistema financiero genérico: las entidades bancarias, los inversores institucionales y el mercado de valores.

- Entidades bancarias: Son aquellas entidades (bancos, cajas de ahorro, cooperativas de crédito...) que tienen la labor de almacenar el ahorro de cada uno de sus clientes, a cambio de una determinada rentabilidad, permitiendo el acceso a éste a prestatarios a cambio de un interés. En consecuencia, todo cliente de una entidad bancaria con un depósito abierto está contribuyendo al funcionamiento del sistema financiero, prestando su dinero, a veces inconscientemente, a muy diversos prestatarios.
- Inversores institucionales: Se trata de aquellas entidades que se dedican a reunir dinero para comprar valores, bienes inmuebles y otros activos de inversión, u originar préstamos. Entre muchos otros tipos destacan las Instituciones de Inversión Colectiva (Fondos de Inversión, Sociedades de Inversión, Exchange Traded Funds o ETFs...), fondos de pensiones, fondos de seguros, fondos de cobertura...

- Mercado de valores: El mercado de valores es la parte del sistema financiero en la que se negocian los valores emitidos por las partes que necesitan el capital. Estas partes ofrecen sus valores en dicho mercado, donde son colocados a inversionistas interesados en éstos, con el objetivo de obtener una rentabilidad.



Fuente: Elaboración propia.

Figura 1. Componentes de un sistema financiero

Como se puede ver en la figura 1, el funcionamiento del sistema financiero es tal como se ha descrito. Sin embargo, no se han comentado las flechas verticales con doble dirección entre los intermediarios del sistema. Y es que, cada uno de ellos puede acudir a los demás si la operación le conviene. De hecho, se lleva a cabo a diario, y ya no se entiende el sistema financiero sin la interacción también entre ellos.

3.2. EL MERCADO DE VALORES. ¿QUÉ ES Y CÓMO FUNCIONA?

El mercado de valores, como su propio nombre indica, es aquel en el que se negocian valores. Explicar qué es un valor es algo complejo, por la elevada diversidad de éstos. Sin embargo, se pueden dividir en dos tipos, y, explicando cada uno es posible formarse una idea simplificada de lo que son.

En primer lugar, existen los valores de renta fija, que, entre otros, suelen conocerse como bonos, obligaciones o letras. Éstos, en función, normalmente, a su demanda, suelen ofrecer un determinado tipo de interés para el inversor, que depende del plazo de finalización, terminan generando una determinada rentabilidad. Como bien indica su nombre, antes de la compra del valor se conoce la rentabilidad que se va a obtener, y ésta no está sujeta a variaciones de ningún tipo, aparte del mordisco que se pueda llevar la inflación (aunque también existen valores de renta fija con protección contra la inflación, evidentemente con un menor interés).

Sin embargo, éste trabajo se va a centrar en los valores de renta variable, y más concretamente en el mercado principal de éstos, comúnmente conocido como la bolsa. A la hora de adquirir uno de éstos, los cuales son conocidos como acciones, se puede esperar una rentabilidad determinada llamada dividendo, que es la parte del resultado que la empresa relacionada a esa acción reparte entre los accionistas. Éste valor suele fluctuar con el tiempo en función al resultado de la empresa en cuestión, siendo determinante la cantidad monetaria repartida a la hora de definir el precio de dicha acción. Y esto lleva directamente a explicar la parte más importante, y es que, estas acciones no tienen un precio fijo, si no que su precio se va determinando a diario en base a la oferta y la demanda. Los precios de éstas fluctúan a diario, por lo que se pueden obtener plusvalías en cortos periodos de tiempo, lo que también atrae, a parte de inversores, a especuladores que quieren llevarse su trozo del pastel. Esto convierte a la bolsa en un juego complejo, prácticamente de azar, en el que la comprensión del funcionamiento de ella es fundamental para sacar provecho de las posibilidades que ofrece. Ahí reside la motivación para la elaboración de este trabajo.

3.3 ÍNDICES BURSÁTILES. S&P 500 E IBEX 35

Dentro del mercado de valores, se va a proceder a realizar las predicciones sobre índices bursátiles, concretamente el americano *Standard and Poor's 500* (S&P500), y el español IBEX 35. Se ha decidido trabajar con índices por representar al mercado en su conjunto de forma más fiel, pudiendo aglutinar mayor diversificación dentro del valor de la acción en cada uno de éstos. Antes de pasar a explicar en qué consiste cada uno de ellos, se debe aclarar la idea de qué es un índice bursátil.

La diferencia entre un índice bursátil y una empresa cotizada es que el índice recoge la variación de un conjunto de activos de un determinado tipo. En el caso de la bolsa, los índices bursátiles aglutinan de una determinada manera la variación

conjunta de las acciones de un determinado grupo de empresas cotizadas. Así, por ejemplo, el S&P 500 concentra de forma ponderada la variación del precio de las 500 empresas con mayor capitalización de la bolsa de Nueva York y el NASDAQ (*National Association of Securities Dealers Automated Quotation*), mientras que el IBEX 35 hace lo propio con las 35 empresas de mayor liquidez de la bolsa española.

Cabe destacar que un índice en un principio no tiene un precio estipulado ni es algo de donde se pueda obtener una acción. Estos se evalúan con puntos en una determinada escala y tienen la principal función de ser indicativos. Si se quisiera invertir en índices, habría que acudir a algún ETF, que son fondos de inversión que se encargan en replicarlos. Se podría obtener una participación en este fondo y de esta forma ya se podría decir que se ha invertido en dicho índice, aunque estrictamente no sea así.

Por la gran importancia de ambos índices, el primero, el S&P 500, en el mercado mundial, y el segundo, el IBEX 35, en el español, se ha decidido ejecutar las predicciones sobre ellos para analizar la predictibilidad del mercado bursátil con ARIMA. Para ello, se realizarán tres predicciones a cada uno en la parte práctica, una sobre los puntos diarios, otra sobre los puntos semanales, y otra sobre los puntos mensuales, para analizar su comportamiento en distintas resoluciones temporales.

3.4. PREDICCIÓN DE LAS TENDENCIAS: LAS SUBIDAS, Y LAS BAJADAS

“Lo que hace fascinante a la inversión es que es una apuesta cuyo éxito depende de la capacidad de predecir el futuro” (Malkiel, 1973)

A lo largo de la historia han existido innumerables casos en los que un determinado mercado ha terminado explotando debido a la falta de conocimiento de los participantes y a las elevadas dosis de especulación. Existen ejemplos muy recurrentes, como el Crack de Wall Street en 1929 o la reciente crisis financiera de 2008, aunque tal vez el ejemplo más conocido sea el de la fiebre de los tulipanes en Holanda en el siglo XVII, cuando la gente terminó vendiendo sus propiedades por hacerse con un solo bulbo, terminando en la ruina de gran parte del país cuando la burbuja explotó y los precios volvieron a la realidad. Todos estos casos evidencian la conexión de los mercados financieros con la economía real, y, además, demuestran que es necesario tener un elevado conocimiento de éstos para que estas situaciones

no se vuelvan a repetir, al menos en los mismos órdenes de magnitud en los que se produjeron. De ahí la importancia de la capacidad de la previsión de éstos, no solo para el inversor individual con el objetivo de obtener rendimientos económicos, sino también para mantener el sistema económico estable.

En cuanto al inversor individual, si quiere obtener una rentabilidad en bolsa las instrucciones iniciales son muy sencillas: vender más caro de lo que se ha comprado, o comprar más barato de lo que se ha vendido. Estas estrategias se conocen como ponerse en largo y ponerse en corto, respectivamente. Sin embargo, pese a lo sencillo de este método, la dificultad reside en adelantarse a lo que vaya a hacer el mercado. Es evidente que antes de vender más caro habrá que haber invertido un dinero, y antes de comprar más barato habrá que haber vendido por una determinada cantidad, y nadie puede asegurar que en el periodo de tiempo transcurrido desde la primera operación hasta la segunda todo vaya a ir según lo previsto. Por ello, se dice que toda inversión tiene un riesgo, que normalmente se ve repercutido en la potencial rentabilidad de ésta.

Es aquí donde aparecen las técnicas de predicción de precios del mercado de valores, que tienen en cuenta factores desde la situación de la empresa emisora de las acciones, hasta el gráfico que representa los precios de las acciones a lo largo del tiempo. Algunas de estas técnicas gozan de reconocimiento entre los inversores profesionales, mientras que otras son consideradas absolutamente ineficientes por la mayoría. Sin embargo, todas tienen un mismo objetivo: obtener una rentabilidad por medio de plusvalías en las compras o ventas de acciones.



Fuente: Tripadvisor.

Ilustración 1. Bolsa de Frankfurt. El toro y el oso son los símbolos de las subidas y bajadas de los precios en bolsa, respectivamente. Por ello están presentes en muchas sedes de bolsas del mundo.

Primeras técnicas: base sólida y castillos en el aire

En el libro *A Random Walk Throgh Wall Street* de Burton G. Malkiel, se plantean dos teorías para valorar activos, acuñadas por el autor como la teoría de la base sólida y la teoría de los castillos en el aire, que representan una dicotomía entre las dos formas clásicas de inversión.

La teoría de la base sólida afirma que todas las acciones tienen un valor intrínseco. Con valor intrínseco se quiere expresar que las acciones tienen un valor real, calculable, en base a unas condiciones actuales y a las perspectivas de futuro. El método de inversión en base a esta teoría es el de aplicar unas determinadas fórmulas para calcular el valor intrínseco de éstas, muchas veces referidas al beneficio esperado por dividendos por su adquisición. En función al resultado obtenido, se comparará dicho valor con el precio real de la acción. Se esperará siempre que el precio real de la acción converja hacia el valor intrínseco calculado, y de esta forma se decidirá si ponerse en corto, ponerse en largo, o no hacer nada. Tal vez el caso más conocido de éxito de esta teoría sea el del famoso inversor Warren Buffett, que dice haberse convertido en un inversor de éxito siguiendo éste método.

A diferencia de la base sólida, la teoría de los castillos en el aire rechaza realizar cálculos complejos sobre el valor real de las acciones, considerando mucho

más importante analizar cómo es probable que se comporte la masa de inversores en el futuro, y, especialmente, cómo en periodos alcistas tienden a subirse al carro de las fiebres especulativas. Al fin y al cabo, como decía su máximo exponente, John Maynard Keynes: *“La mayoría de las personas no están básicamente preocupadas por hacer espectaculares predicciones a largo plazo sobre los posibles rendimientos de una inversión a lo largo de su vida, sino en prever los cambios en la base convencional de valoración con anterioridad al público general. No es razonable pagar 25 por una inversión en la que se considera que las perspectivas de rendimiento justifican un valor de 30, si a la vez se considera que el mercado la valorará en 20 de aquí a tres meses”* (Malkiel, 1973). Además de Keynes, la teoría de los castillos en el aire goza de muchos defensores, como puede ser Daniel Kahneman, que obtuvo el premio Nobel por su aportación al campo de las finanzas conductuales, ampliamente sustentado por este método de inversión.

Análisis Técnico y Análisis Fundamental

El análisis técnico y el análisis fundamental son las dos principales técnicas de predicción que se utilizan en el mercado de valores. Desde que la predicción de los precios de los activos en el mercado no es una ciencia exacta, ambas técnicas cuentan con sus defensores y sus detractores, y, a pesar de sus más que evidentes diferencias, ambas cuentan aun a día de hoy con una gran popularidad.

Siguiendo con las ya comentadas en la página anterior técnicas de la base sólida y de los castillos en el aire, se podría hacer un paralelismo entre esas dos y los análisis técnico y fundamental.

En primer lugar el análisis fundamental es el que toma la teoría de la base sólida como pilar principal. Este análisis produce predicciones del precio de las acciones basándose en factores económicos, políticos, sociales... que afectan a la oferta y la demanda de dicho activo. A diferencia de la base sólida, esta teoría busca también averiguar el valor intrínseco de las acciones, pero no solo basándose en las rentabilidades por dividendos, sino que teniendo en cuenta factores aún más complejos como informes financieros, económicos, noticias...

Por otro lado, el análisis técnico se basa en evaluaciones puramente estadísticas. De esta forma, analiza las gráficas para detectar tendencias y patrones que pueden repetirse en el futuro, lo que lo hace parecerse a los modelos ARIMA, como se verá más adelante. Éstos suponen que toda la información analizada en el

análisis fundamental ya está recogida en el precio de las acciones, y, de esta forma, rechaza, al igual que en la teoría de los castillos en el aire, el cálculo del valor intrínseco de dichas acciones, dando también gran importancia a la psicología de masas.

3.5. MERCADOS EFICIENTES Y PASEOS ALEATORIOS

Mercados Eficientes

Las teorías del paseo aleatorio y los mercados eficientes son de gran importancia para este trabajo, dado que el gran objetivo de éste es comprobar hasta donde sea posible su existencia mediante los modelos ARIMA.

La hipótesis de los mercados eficientes fue formulada por primera vez en 1970, por el profesor de la Universidad de Chicago Eugene Fama. Lo que defiende esta hipótesis, que muchos toman ya como teoría en determinados casos, es que cualquier suceso o noticia que haga que cambie el precio de un activo no puede aprovecharse para obtener una rentabilidad, ya que el mercado en su conjunto es lo suficientemente eficiente para ajustarse a un precio de una forma prácticamente instantánea. Por ello, considera una pérdida de tiempo métodos como la búsqueda de valores intrínsecos, ya que ningún activo estará infravalorado o sobrevalorado según esta hipótesis.

Evidentemente, esta teoría no vale para cualquier mercado, ya que algunos no son capaces de conseguir el dinamismo necesario para cambiar de precios tan velozmente. Es por ello que la teoría defiende que un mercado será eficiente si este cumple unas determinadas características:

- Debe tener el número de participantes (o volumen de negociación necesario) para garantizar la liquidez y la fluctuación rápida de precios. Cuanto mayor sea este número, mayor será la eficiencia. Un ejemplo de esta propiedad se puede ver en el mercado de divisas.
- La información no debe estar distribuida de forma asimétrica entre los participantes del mercado, si no que tienen que existir fuentes a las que todo el mundo tenga acceso, y el menor número de personas con información privilegiada.
- Los costes de información y transacción han de ser los mínimos posibles para que los inversores no eviten participar en dicho mercado.

Si un mercado logra reunir estas características podrá ser considerado eficiente. Es por esto que gran parte de los inversores consideran el mercado bursátil como uno de este tipo.

Además de las características de éstos, la hipótesis también puede ser clasificada de 3 formas, en función a la información que reflejan los precios de dichos mercados:

- Eficiencia débil: Los precios de los activos reflejan toda la información histórica, por lo que el análisis técnico no puede servir para predecir el valor de las acciones. Únicamente se podrá predecir mediante información fundamental.
- Eficiencia semi-fuerte: En este caso el análisis fundamental también resulta inútil, ya que la información útil en el análisis fundamental también se considera incluida en el precio.
- Eficiencia fuerte: Los precios reflejan toda la información disponible. En caso de existir algún tipo de información, aunque fuese privilegiada, el mercado se ajustará casi inmediatamente.

Paseos aleatorios

La teoría del paseo aleatorio procede de un modelo estadístico que defiende que las series estocásticas que se adapten a él se comportan de forma completamente aleatoria, siendo lo único conocido del siguiente incremento o decremento de valor su media y su varianza.

Esta teoría pasó a considerarse adaptable al mercado bursátil a partir de la teoría de los mercados eficientes, y, tal vez su más reconocido defensor sea Burton G. Malkiel, que en su conocido libro *Un Paseo Aleatorio Por Wall Street* la dio a conocer, libro que ha servido de inspiración para la realización de este trabajo.

En esta teoría, basada en un modelo matemático que se explica en este trabajo en el cuarto apartado, se considera cierta la hipótesis del mercado eficiente, dando por hecho que toda la información disponible está reflejada en el precio de las acciones, lo que implica que siempre existirá una asunción de riesgo bajo las condiciones dadas por los mercados. Es bastante acertada la analogía que se hace en el libro: *“Llevado a un extremo lógico, quiere decir que un chimpancé con los ojos vendados tirando dardos sobre páginas de valores bursátiles de un periódico podría seleccionar una cartera de valores tan buena como la seleccionada con el mayor cuidado por los expertos”* (Malkiel, 1973).

La idea principal de un paseo aleatorio es que en ningún caso se puede obtener información en función a los datos anteriores, y que las variaciones de precio a corto medio y largo plazo serán aleatorias desde una perspectiva matemática. Es por esto por lo que entra en conflicto con la utilización de ARIMA para estos cometidos, a pesar de estar incluido también dentro de este tipo de modelos cuando se introducen unos determinados parámetros. Por ello, se podrá comprobar analizando algunas series de precios de índices bursátiles si éstos se comportan como paseos aleatorios o no, y, de esta forma, desmentir o confirmar en la medida de lo posible la teoría de los mercados regidos por la aleatoriedad y la independencia respecto a sus valores anteriores.

4. DESARROLLO TEÓRICO DE LOS MODELOS E IMPLEMENTACIÓN EN R.

4.1. ELECCIÓN DE SOFTWARE PARA LLEVAR A CABO LA PRÁCTICA.

Previo a comenzar con el desarrollo y la ejecución de la aplicación de los modelos ARIMA será necesario definir las herramientas a utilizar durante todo el proceso. Debido a la complejidad matemática de este tipo de modelos, habrá que recurrir a un software con la capacidad de trabajar con ellos, tomando en cuenta que debe ser capaz de soportar la elevada carga computacional a la que ARIMA puede someter al sistema, teniendo que iterar entre muchos valores de la misma serie temporal muchas veces.

Para la elección se han plantearon varias alternativas. Entre ellas se encontraban MatLab, eViews, Mathematica y R.

En un corto periodo de tiempo, Mathematica quedó descartado, ya que, a pesar de la experiencia del autor en este programa, no existe bibliografía suficiente como para modelar series temporales adecuándose a los modelos ARIMA.

En cuanto a MatLab, resulta ser un programa muy versátil, y el autor constató que su capacidad y herramientas para trabajar con estos modelos eran bastante adecuadas. Sin embargo, al no haber trabajado nunca con este programa y tras haber consultado con otras personas experimentadas en este software, quedó descartado, ya que las otras dos alternativas se prometían más prácticas para el desarrollo del trabajo.

Tras estos dos descartes, los programas eViews y R quedaban como los dos posibles candidatos.

Poniendo los dos softwares en balance, eViews ofrecía un sistema de modelización de series ARIMA integrado en el código base del programa, lo que lo hacía mucho más especializado y sencillo para este tipo de tareas. R, sin embargo, suponía una carga de tiempo en cuanto a formación, ya que era necesario llegar a un nivel de conocimiento más profundo para llevar a cabo la práctica, y una búsqueda de paquetes de funciones que se adecuasen a las características de los modelos. Como contrapartida, R es un programa específicamente desarrollado para el análisis estadístico de datos, teniendo unas herramientas de gran potencial como pueden ser

la capacidad de desarrollar un código propio para el tratamiento más eficiente de la información, o una desatacada versatilidad a la hora de representar los resultados en diversos tipos de gráficos.

Tras investigar sobre el tiempo y el resultado que se podría obtener con cada uno de los programas, se decidió optar por R, ya que el proceso de aprendizaje de programación en éste desde 0, hasta los modelos ARIMA, serviría también para incrementar y reforzar la comprensión e interiorización del funcionamiento al detalle de éstos.

4.2.R. ¿QUÉ ES? ¿CÓMO FUNCIONA?

R

El lenguaje de programación R nace en 1993, en forma de software libre y basado generalmente en un lenguaje anterior desarrollado en *Bell Laboratories*, llamado S. El enfoque desde un inicio fue el análisis estadístico, lo que lo ha hecho un lenguaje muy utilizado en los campos del Big Data y las matemáticas financieras.

Se trata de un lenguaje de programación orientado a objetos con su propio software descargable desde su página oficial, donde también se puede acceder a bases de datos con funciones ya predefinidas que serán recurridas durante el trabajo, como se verá más adelante.



Fuente: R-CRAN.

Ilustración 2. Logo de R.

Existe una versión de la interfaz más intuitiva del software, que tiene las mismas funciones y se puede descargar gratuitamente como complemento, llamado *R Studio*. Esta interfaz será también utilizada durante el estudio.

Asignación de variables

Dentro del entorno de R, se denominan objetos a las variables o grupos de caracteres que tienen un determinado valor asignado. Así, un objeto podría ser una variable *x* a la que se le asigna un valor cualquiera, por ejemplo, 100. Lo que se consigue de la siguiente forma:

```
> x<-100
> print(x)
[1] 100
```

Este tipo de objeto es llamado *numeric*, por guardar el valor de un número real.

Tipos de objetos en R

Entre los principales tipos de objetos se encuentran los *logical*, que tan solo guardan un valor binario, que puede ser o bien *TRUE* (1) o *FALSE* (0). Al realizar operaciones aritméticas con ellos, como podría ser una media, estos valores son tomados con su valor numérico:

```
> LOGICAL1<-TRUE
> LOGICAL2<-FALSE
> LOGICAL3<-TRUE
> MEDIA<-(LOGICAL1+LOGICAL2+LOGICAL3)/3
> print(MEDIA)
[1] 0.6666667
```

Se tiene también entre los objetos más elementales los ya mencionados *numeric*, que se corresponden con cualquier número real, es decir, números con decimales o sin ellos.

Como último tipo de objeto elemental están las cadenas de caracteres o *character*, que no son nada más que textos asignados a variables, de utilidad generalmente organizativa o estética dentro del desarrollo del código. Estos elementos se escriben siempre entre comillas a la hora de asignarlos a variables para no confundirlos con otros objetos.

```
> CHAR<-"Programar ARIMA con R"
> print(CHAR)
[1] "Programar ARIMA con R"
```

Entre las casi infinitas funciones de R, la función `class()` puede identificar de qué tipo es la variable con la que se está trabajando.

```
> class(x)
[1] "numeric"
> class(LOGICAL1)
[1] "logical"
> class(CHAR)
[1] "character"
```

Teniendo ya en cuenta los tipos básicos, se pueden describir ahora los tipos de grupos de objetos más utilizados en R, y que serán continuamente recurridos en el desarrollo del trabajo.

Array: Los *array* son grupos de objetos unidimensionales, también conocidos como vectores. Este grupo de objetos tiene que contener todos los objetos del mismo tipo, es decir, su composición ha de ser homogénea en cuanto a *numeric*, *logical*, *charecater*... Los *array* se generan con el siguiente comando (hay que destacar la necesidad de colocar la letra *c* antes de la serie de objetos):

```
> Mi.Array<-c(1,2,3,4,5)
> print(Mi.Array)
[1] 1 2 3 4 5
```

Matrices: Las matrices siguen el mismo principio que los *array*, aunque en vez de ser unidimensionales son multidimensionales, con *n* filas y *m* columnas. La función utilizada para su generación es `matrix()`.

Data Frame: Los *data frame* son como las matrices, pero tienen la capacidad de almacenar vectores de diferentes tipos de objetos. De esta forma se consiguen relacionar, por ejemplo, valores numéricos con cadenas de caracteres, para aportar mayor información expresándolos en forma de matriz. Se generan con la función `data.frame()`. Cabe destacar que estos son de gran utilidad porque pueden ser transformados en objetos de tipo *ts* (*time series*) con gran facilidad.

```
> CUERPO<-c("Ojos", "Nariz", "Boca", "Brazos")
> NUMERO<-c(2,1,1,2)
> DaFr<-data.frame(CUERPO, NUMERO)
> DaFr
  CUERPO NUMERO
1  Ojos      2
2  Nariz      1
3  Boca      1
4  Brazos     2
```

Listas: Sería posible definir las listas como vectores dentro de vectores, aunque también se podría decir que son vectores que pueden almacenar objetos de diferente tipo. Ambas cosas se pueden hacer con una lista, y evidentemente, la combinación de ambas propiedades también es posible. La función correspondiente es *list()*.

```
> list(c(1,2,3),c(TRUE,FALSE),c("ARIMA"))
[[1]]
[1] 1 2 3

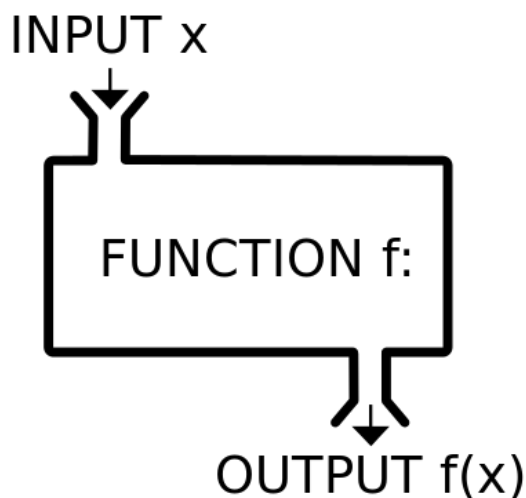
[[2]]
[1] TRUE FALSE

[[3]]
[1] "ARIMA"
```

Series temporales: Los grupos objetos de tipo *ts* o *time serie* son capaces de almacenar series temporales dentro de R. Su funcionamiento será explicado con mayor detalle más adelante debido a que son objetos de gran utilidad para el desarrollo de este trabajo. Se generan con la función *ts()*.

Funciones útiles para el desarrollo del trabajo:

En programación se considera función a cualquier tipo de comando que tome unos inputs, los procese de una determinada manera y genere los outputs deseados de acuerdo a la misión de dicha herramienta. De esta forma, una función puede ser vista como una caja opaca donde se introducen unos datos, estos se procesan de acuerdo a unas determinadas condiciones, sin que el usuario tenga que saber necesariamente cómo, y se obtiene la información deseada. Al aplicarlas hay que tener en cuenta que el lenguaje R es sensible a las mayúsculas y minúsculas.



Fuente: francescricart.com.

Figura 2. Esquema del funcionamiento de una función.

En R, las funciones tienen el siguiente formato: *función(argumento1, argumento2, argumento3,...)* Llamando argumentos a los inputs que son necesarios para que dicha función trabaje correctamente.

La forma de identificar los argumentos de las funciones que tiene R, es, o bien por nombre, escribiendo el nombre del argumento e introduciendo su valor con un "=", o por orden de escritura de los argumentos.

```
> round(7.89479,3)
[1] 7.895
> round(digits = 3, x=7.89479)
[1] 7.895
```

Cabe destacar que la mayoría de funciones de R tienen varios argumentos *default*, lo que quiere decir que sus valores vienen dados por defecto, por lo que, a no ser que se quiera realizar la aplicación de la función en condiciones diferentes, no es necesario introducir estos argumentos en el código.

El paquete base de R es muy completo en cuanto a funciones, conteniendo muchas y de gran variedad, orientadas en su mayoría al análisis estadístico de datos. Muchas de ellas serán utilizadas en el desarrollo práctico.

Algunos tipos de funciones no tienen la misma estructura que la descrita anteriormente. Sin embargo, se destacan varias de ellas, de gran utilidad a la hora de programar, que van a ser utilizadas también en la práctica.

Conditional Statement: También conocidas como condiciones de ejecución en castellano o *ifs* en el argot de la programación, son funciones que introducen una determinada condición sobre los datos, y dependiendo de si la cumplen o no, son tratados de una forma o de otra. La estructura de estas funciones es la siguiente:

```
> if( "Condición" ){  
  "tratamiento de datos 1"  
}else{  
  "tratamiento de datos 2"  
}
```

Loop: Los *loop* o bucles, en castellano, son herramientas que repiten una y otra vez una ejecución sobre los datos hasta que se cumple una determinada condición, o se terminan las iteraciones programadas, dependiendo del tipo de *loop*. Existen de dos clases: *while* y *for*.

- En cuanto a los ***while***, son bucles que especifican una determinada condición, y hasta que los inputs la cumplen, éstos son tratados por el código contenido dentro de él. Hay que tener especial cuidado con este tipo de funciones porque se corre el riesgo de caer en bucles infinitos si las iteraciones no convergen a la condición deseada, lo que puede estropear la ejecución de nuestro código.
- Los bucles ***for*** funcionan de forma diferente. Si los bucles *while* especificaban una determinada condición a cumplir por los datos, éstos especifican el número de iteraciones a realizar a los inputs.

```
> while ( "Condición a eliminar" ){  
  "Instrucciones para tratamiento de datos"  
}
```

```
> for (i in 1:"numero de iteraciones deseadas"){  
  "instrucciones para el tratamiento de datos"  
}
```

Funciones creadas por el usuario: Existe una herramienta cuyo cometido es que el propio usuario pueda crear sus propias funciones y guardarlas en el *Workspace*. De esta forma se puede hacer más eficiente la estructura de código guardando como funciones procesos que se vayan a repetir a lo largo de la práctica.

La estructura y el funcionamiento de esta herramienta se conforma de la siguiente manera:

```
> Calcular.Media<- function(x,y,z){  
  (x+y+z)/3  
}  
> Calcular.Media(18,16,5)  
[1] 13
```

R-Packages

Todas las funciones definidas en R están contenidas en diferentes paquetes o *packages*. Un paquete es un grupo de funciones con el código ya desarrollado que generalmente tienen una serie de características comunes, como puede ser el fin para el que fueron creadas o el ámbito en el que pueden ser aplicadas. El paquete de funciones de R más elemental es el *base*, que viene cargado e instalado cuando se descarga o se inicia R. Además de este conjunto de funciones elementales, existen gran cantidad de *packages* con el objetivo de cumplir diferentes cometidos. La utilidad de la existencia de los paquetes reside en el ahorro de tiempo que puede suponer disponer de funciones que eviten que se tengan que escribir múltiples líneas de código. Es por esto que, antes de iniciar el desarrollo de un código conviene mirar el catálogo de paquetes para ver si existen en ellos funciones que puedan ahorrar tiempo.

El comando para instalarlos es *install.packages("nombre del paquete")*, el cual trasladará al usuario directamente a CRAN (*Comprehensive R Archive Network*), donde hay miles de paquetes con funciones disponibles. Antes de poder usarlos, una vez descargados, habrá que cargarlos en la sesión que se esté usando. Para encontrarlo se puede usar la función *search()*, de este modo aparecerán todos los paquetes ya instalados. Para cargarlo simplemente hay que usar el comando *library()*. En el desarrollo de esta práctica así se ha hecho, y se han cargado los paquetes *xts*, *zoo* y *astsa*, entre otros, que tienen funciones muy útiles para el modelaje de series temporales. Todos los paquetes instalados para la realización del trabajo vienen descritos en el ANEXO I.

Files Plots Packages Help Viewer			
Install		Update	
Name	Description	Version	
User Library			
<input checked="" type="checkbox"/>	astsa	Applied Statistical Time Series Analysis	1.9
<input checked="" type="checkbox"/>	xts	eXtensible Time Series	0.11-2
<input checked="" type="checkbox"/>	zoo	S3 Infrastructure for Regular and Irregular Time Series (Z's Ordered Observations)	1.8-6
System Library			
<input checked="" type="checkbox"/>	base	The R Base Package	3.6.0

Fuente: Elaboración propia.

Figura 3. Paquetes más importantes usados durante el trabajo.

4.3. SERIES TEMPORALES EN R

4.3.1. Manejar una serie temporal en R

R destaca por la gran facilidad que da para trabajar con series temporales. Para comenzar a hablar de las series temporales en este lenguaje se puede empezar por explicar cómo crear una serie temporal. Para esto R tiene la función `ts()`, que genera un vector con los datos introducidos y los distribuye en intervalos de tiempo equitativos. Si no se especifica, el índice de tiempos será, por defecto: 1, 2, 3, 4,... De querer especificarse, como se ha explicado en el apartado de funciones de R, habrá que cambiar los valores *default* de la serie incluyendo más inputs. Un ejemplo para una serie temporal de 34 datos que tome 12 datos al año y comenzase en 1980 sería el siguiente:

```
> vector.de.datos<-c(60,45,354,45,35,354,35,35,45,34,345,3
23,23,3,44,554,564,34,534,545,764,34,23,45,67,54,34,56,34,
23,435,5,354,34)
> mi.serie<-ts(vector.de.datos,start=1980,frequency = 12)
> print(mi.serie)
      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
1980  60  45 354  45  35 354  35  35  45  34 345 323
1981  23   3  44 554 564  34 534 545 764  34  23  45
1982  67  54  34  56  34  23 435   5 354  34
```

Una vez se obtenga el objeto de tipo *ts* (*time series*) podría surgir la duda de si la transformación ha resultado adecuada. Una vez más, se puede aplicar la función *class()* sobre la serie temporal, como se ha explicado en el apartado de tipos de objetos.

```
> class(mi.serie)
[1] "ts"
```

Cuando ya se tiene definida la serie, puede interesar extraer información de ella, ya sea para comprobar que la transformación a objeto *ts* ha sido óptima, o para obtener una serie de datos concretos de ella. Las funciones más recurridas en estos casos son las siguientes:

- *print(miserie)*: Escribe todos los datos de la serie para que podamos examinarlos uno a uno.
- *length(miserie)*: Nos devuelve el número de datos que tiene la serie.
- *head(miserie, n=10)*: Nos dice los diez primeros datos de la serie.
- *tail(miserie, n=10)*: Nos dice los diez últimos datos de la serie.
- *plot(miserie)*: Nos devuelve un gráfico sencillo de la serie.
- *time(miserie)*: Calcula el vector de los índices de tiempo adecuándose a la premisa de que los intervalos son constantes.
- *frequency(miserie)*: Devuelve el número de datos por unidad de tiempo de la serie.

4.3.2. Tendencia en series temporales

Uno de los mayores problemas a los que hay que enfrentarse en el análisis de los datos es la tendencia de las series temporales. Por razones que se explicarán más adelante, es necesaria la eliminación de esta propiedad para su predicción, y, en consecuencia, también para la aplicación de modelos ARIMA.

Existen cuatro tipos de tendencia que puede presentar una serie temporal:

Tendencia lineal:

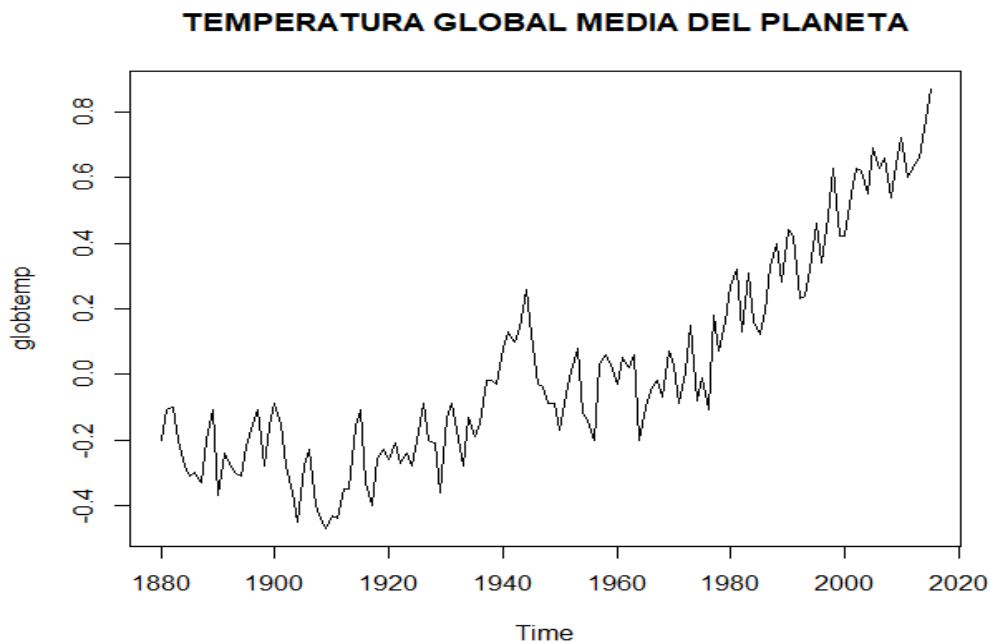
Ésta es una de las dos tendencias que se pueden catalogar como tendencia en media. Las series que siguen este tipo de comportamiento presentan incrementos o decrementos de su media siguiendo una determinada pendiente. Si se trazase una línea recta por el método de mínimos cuadrados sobre un gráfico de la serie, la tendencia de ésta se corresponde con la pendiente de dicha recta.

Esta tendencia tiene una eliminación sencilla, por medio de la diferenciación de la serie, que consiste en restar a cada parámetro su valor inmediatamente anterior. Con este método el resultado son las diferencias entre los valores.

$$\Delta\{Y_t\} = \{Y_t\} - \{Y_{t-1}\}$$

Esto se consigue en R de forma muy sencilla y rápida con la función *diff()*, en la que únicamente habrá que introducir como argumento la serie a modelar.

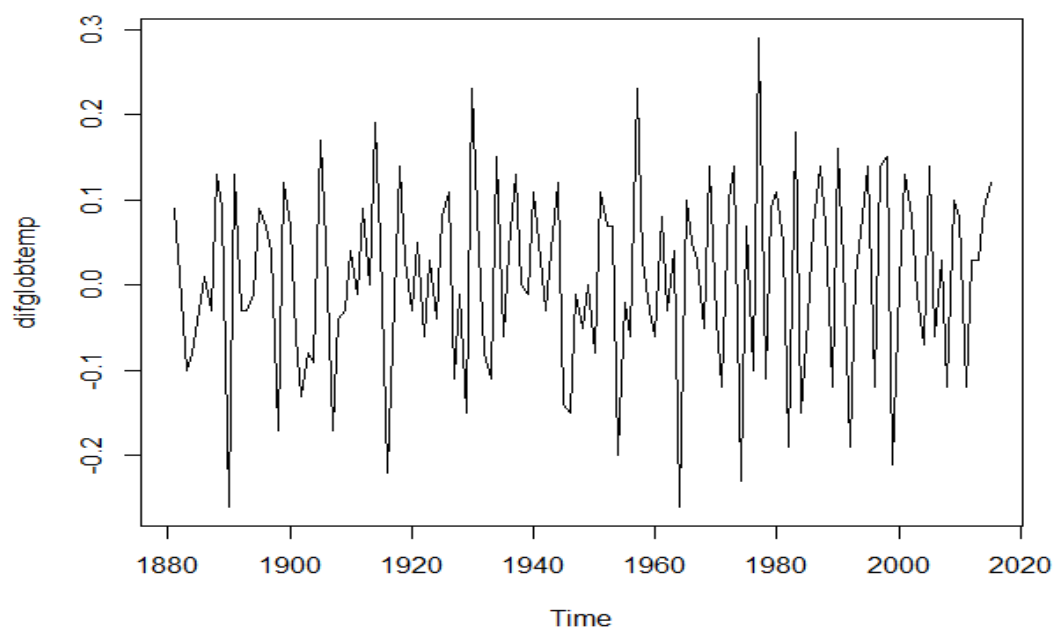
```
> difglobtemp<-diff(globtemp)
```



Fuente: Elaboración propia.

Figura 4. Gráfico de la temperatura global media del planeta.

TEMPERATURA GLOBAL MEDIA DEL PLANETA DIFERENCIADA



Fuente: Elaboración propia.

Figura 5. Gráfico de la temperatura global media del planeta diferenciada.

Tendencia de crecimiento (o decrecimiento) rápido:

Este tipo de tendencia también se corresponde con la clasificación de tendencias en media. La diferencia con la anterior es que, al trazar una curva sobre la serie, ésta no se corresponde con una ecuación lineal, sino que responde a curvas de órdenes superiores. Por ello, con una simple diferenciación no basta. La solución para este problema es diferenciar la función el mismo número de veces que al orden al que responde dicha curva. Por ejemplo, una serie con tendencia cuadrática:

$$\Delta^2\{Y_t\} = \Delta(\Delta\{Y_t\}) = (\{Y_t\} - \{Y_{t-1}\}) - (\{Y_{t-1}\} - \{Y_{t-2}\})$$

De la misma forma que en el caso anterior, esta diferenciación se conseguiría con el comando *diff()*, solo que habría que usarlo dos veces con la misma serie: *diff(diff())*. El procedimiento es igual para series de mayor grado.

$$\Delta^n \{Y_t\} = \Delta(\Delta^{n-1} \{Y_t\}) = (\{Y_t\} - \{Y_{t-1}\}) - (\{Y_{t-1}\} - \{Y_{t-2}\}) - \dots - (\{Y_{t-(n-1)}\} - \{Y_{t-n}\})$$

Tendencia periódica o sinusoidal (Estacionalidad)

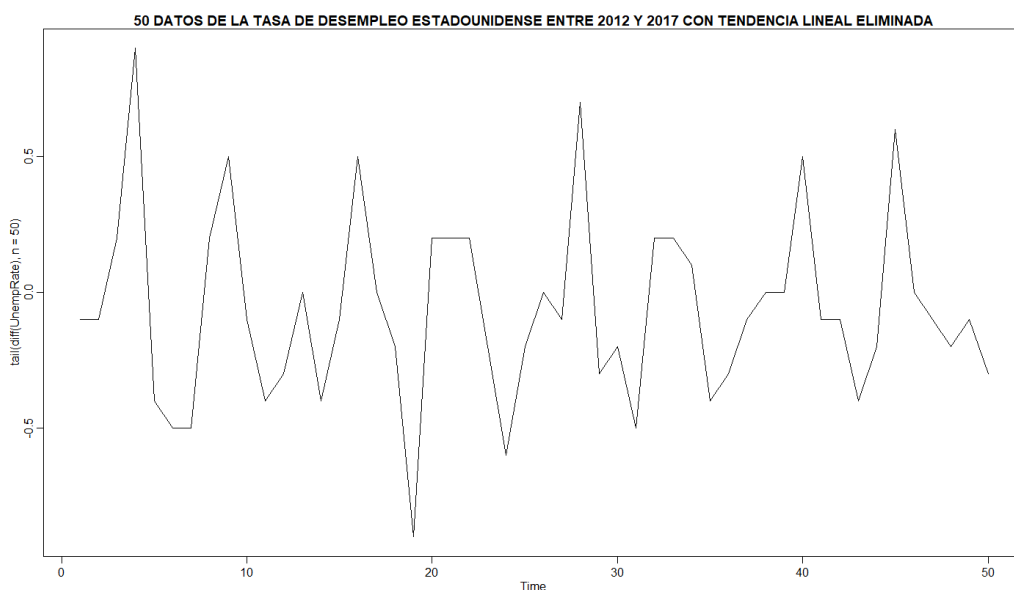
La tendencia periódica, más popularmente conocida como estacionalidad, responde a patrones de la serie en pequeños lapsos de tiempo, como puede ser un año. Son particularmente fáciles de detectar mediante el análisis visual del gráfico de la serie. Estos casos responden a fenómenos estacionales varios. Un ejemplo de serie estacional puede ser la del desempleo estadounidense, donde antes de empezar el verano se observa una pronunciada bajada y al finalizarlo se produce una abrupta subida. Este patrón se repite en todos los años.

Si se quiere eliminar este tipo de tendencia, se debe recurrir también a la diferenciación, pero, en vez de aplicar la resta mediante el valor directamente anterior, se debe detectar el ciclo del patrón (normalmente anual), y el número de datos que se tiene dentro del ciclo, número al que a partir de ahora se llamará s (normalmente $s=12$ para datos mensuales y $s=4$ para datos trimestrales). La diferenciación se hará mediante intervalos separados a una distancia de s datos, y, de esta forma, la tendencia estacional desaparece.

$$\Delta^n \{Y_t\}_s = \Delta(\Delta^{n-1} \{Y_t\}_s) = (\{Y_t\} - \{Y_{t-s}\}) - (\{Y_{t-s}\} - \{Y_{t-2s}\}) - \dots - (\{Y_{t-(ns-s)}\} - \{Y_{t-ns}\})$$

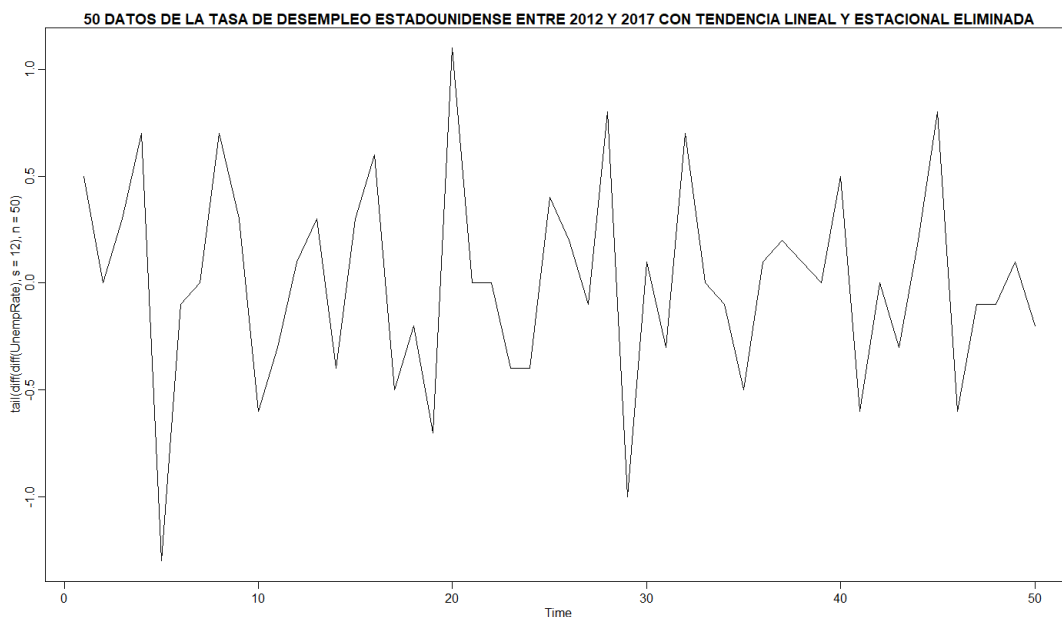
Una vez más, la función a utilizar en R vuelve a ser *diff()*, solo que en este caso se deberá introducir el parámetro s , que por defecto viene dado con valor 1 (el valor 1 de s es el utilizado para eliminar tendencias en la media)

```
> ts.plot(tail(diff(UnempRate), n=50), type="l", main= "50 DATOS DE LA TASA DE DESEMPLEO ESTADOUNIDENSE ENTRE 2012 Y 2017 CON TENDENCIA LINEAL ELIMINADA")
> ts.plot(tail(diff(diff(UnempRate),s=12), n=50), type="l", main= "50 DATOS DE LA TASA DE DESEMPLEO ESTADOUNIDENSE ENTRE 2012 Y 2017 CON TENDENCIA LINEAL Y ESTACIONAL ELIMINADA")
```



Fuente: Elaboración propia.

Figura 6. Gráfico de la tasa de desempleo estadounidense entre 2012 y 2017 con la tendencia lineal eliminada.



Fuente: Elaboración propia.

Figura 7. Gráfico de la tasa de desempleo estadounidense entre 2012 y 2017 con tendencia lineal y estacional eliminada.

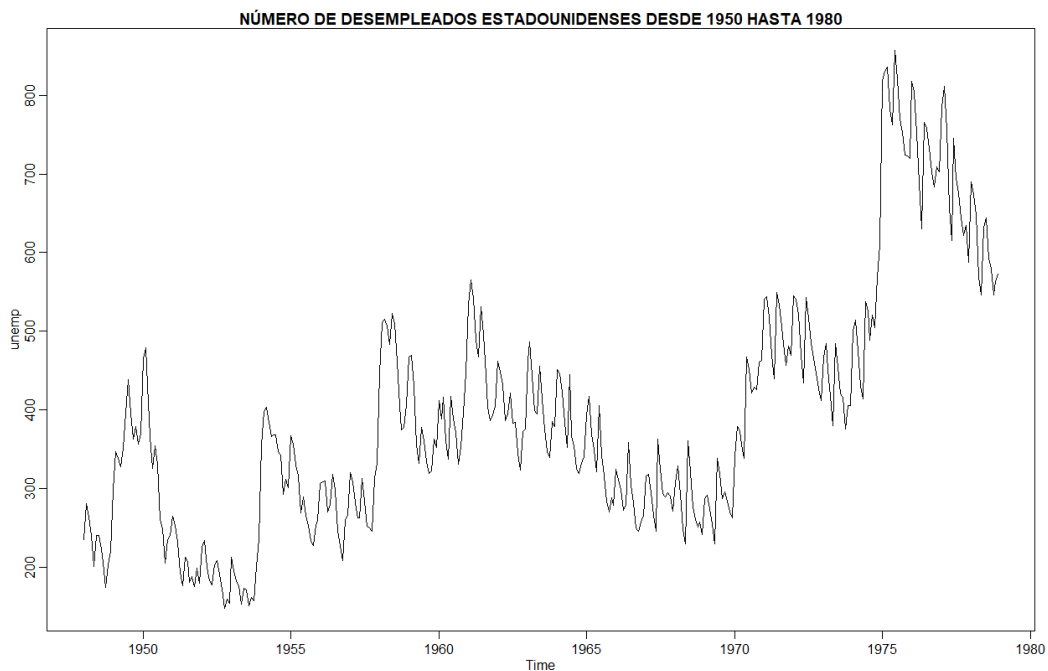
Como se puede ver en la figura 6 y en la figura 7, si se realiza un análisis de los dos gráficos, existe un ciclo entre los datos de intervalos separados por 12 datos, que se identifican como los datos tomados los mismos meses. Haciendo la

diferenciación, esta dependencia desaparece, eliminando de esta forma dichos patrones.

Tendencia en varianza o heterocedasticidad

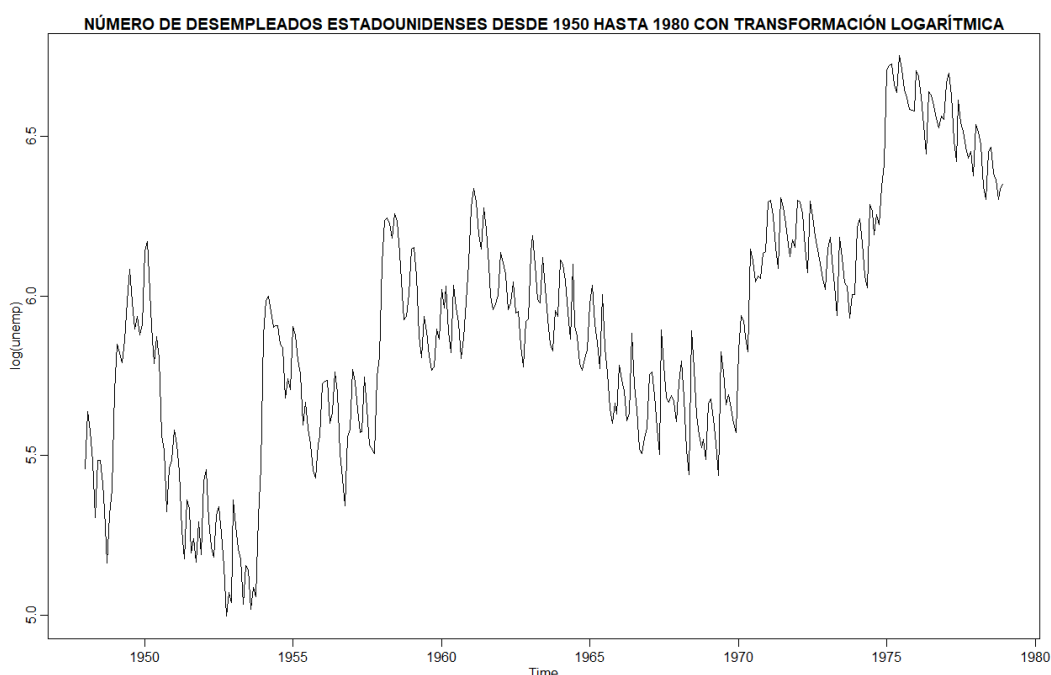
La heterocedasticidad es una propiedad muy común en las series temporales, generalmente en series que también tienen una tendencia en media. Hablar de heterocedasticidad es hablar de variaciones en la varianza. Generalmente, al aumentar de valor de la media, la serie aumenta su varianza de forma proporcional, y al disminuirlo, ésta también disminuye.

La forma más eficaz de eliminar la heterocedasticidad es aplicando una transformación logarítmica a la serie en cuestión. La función en R para la transformación logarítmica es $\log()$. Se puede comprobar cómo funciona en el siguiente ejemplo:



Fuente: Elaboración propia.

Figura 8. Gráfico del número de desempleados estadounidenses desde 1950 hasta 1980.



Fuente: Elaboración propia.

Figura 9. Gráfico del número de desempleados estadounidenses desde 1950 hasta 1980 con transformación logarítmica aplicada.

Como se puede ver en la figura 8, existe una leve heterocedasticidad. A medida que aumenta la población estadounidense, y en consecuencia el número de desempleados, las variaciones en las caídas y subidas del nivel de desempleados son mayores en los mismos periodos de tiempo. Como ya se ha comentado, y se puede apreciar en la figura 9, la transformación logarítmica transforma la serie heterocedástica en otra homocedástica, reduciendo significativamente la varianza, que es especialmente apreciable al final de la serie.

Hay que destacar que esta transformación solo se puede hacer en series de valores positivos, por lo que si se tienen datos negativos habrá que hacer las correspondientes transformaciones.

4.3.3. Modelos básicos de series temporales

4.3.3.1 Fundamentos teóricos

Previo a la explicación teórica y el implemento en R de los modelos que construyen ARIMA, se deben aclarar algunos conceptos y definir su implementación en R

Procesos estacionarios.

Los procesos estacionarios son un fundamento previo al análisis de series temporales. Esta condición es fundamental, ya que, si un proceso tiene las condiciones suficientes de estacionariedad, se convierte en un proceso estimable y modelable por ARIMA.

Pero, ¿Qué quiere decir que un proceso sea estacionario? Explicado a grosso modo, se define con esta cualidad a una serie que muestra una oscilación aleatoria, de carácter normal, sobre un nivel, fenómeno que es conocido como regresión a la media.

En términos técnicos, se dice que una serie temporal es estacionaria cuando:

- Media μ de Y_t es igual (constante) para todo t .
- La varianza σ^2 es igual (constante) para todo t (homocedasticidad).
- La covarianza entre datos separados las mismas distancias son constantes para todo t .

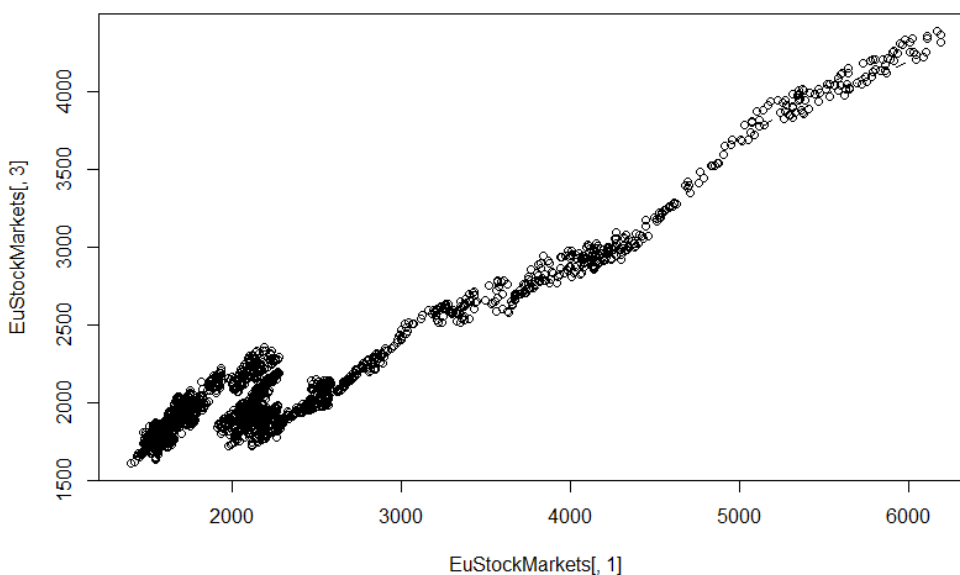
Es por esta propiedad por lo que son necesarias las transformaciones para eliminar las diferentes tendencias explicadas anteriormente.

Gráficos de dispersión

Los gráficos de dispersión son herramientas muy utilizadas en el campo de la estadística y el análisis de datos para comprobar visualmente la covarianza y la correlación (términos que se explicarán más adelante) de dos series distintas. Su análisis consiste en comprobar la dispersión de los datos alrededor de una recta aproximada por mínimos cuadrados. Cuanta menor dispersión se aprecie alrededor de la recta, mayor (en valor absoluto) será la covarianza y correlación de las dos series, obteniendo valores cercanos a 0 en los casos en los que exista mucha dispersión.

La forma más sencilla de realizar un gráfico de dispersión en R es con la función `plot()`, solo que en este caso en vez de introducir una serie como único argumento, se introducirán dos, separadas por una coma. El resultado de ejecutar esto se puede ver en la figura 10.

GRÁFICO DE DISPERSIÓN ENTRE EL ÍNDICE DAX Y EL ÍNDICE CAC



Fuente: Elaboración propia.

Figura 10. Gráfico de dispersión entre el índice alemán DAX y el francés CAC.

Covarianza y Correlación

La covarianza, a nivel cualitativo, es un término numérico que expresa la variación conjunta de dos conjuntos de datos. Se calcula por medio de la siguiente fórmula:

$$COV(X, Y) = (\Sigma(x_i - \underline{x})(y_i - \underline{y})) / n$$

Las conclusiones a las que se pueden llegar con estos datos son tres. Si la covarianza es positiva, se sabrá que la variación de las series es, de alguna forma, proporcional (una sube cuando la otra sube). Si el valor es nulo, o prácticamente nulo, la variación de las series no tendrá relación una con la otra, lo que se conoce como series no correlativas. Y, de ser negativa, esto indicaría que la variación de las series es inversamente proporcional (una baja cuando la otra sube).

Sin embargo, el valor de la covarianza, aparte de su signo, puede no indicar mucho, ya que la magnitud de ésta está dada en términos relativos a la serie. Por ello,

para conocer con precisión la “cantidad de relación” que existe entre las dos series, hay que utilizar el concepto de correlación.

La correlación puede comprender valores desde -1 a 1, siendo estos dos límites los casos en los que la serie tendría una relación perfecta (en el caso de ser 1), es decir, que varían conjuntamente de forma ideal, y una relación perfectamente inversa (en el caso de ser -1), caso en el que la variación positiva de una es igual a la variación negativa de la otra. La fórmula de la correlación es la siguiente:

$$\text{COR}(X, Y) = \text{COV}(X, Y) / (\sigma_x * \sigma_y)$$

Siendo σ_x y σ_y las desviaciones estándar de las series, es decir, las raíces cuadradas de sus varianzas σ^2 .

Para calcular las fórmulas de la covarianza y la correlación en R existen dos funciones: *cov()* y *cor()*, respectivamente.

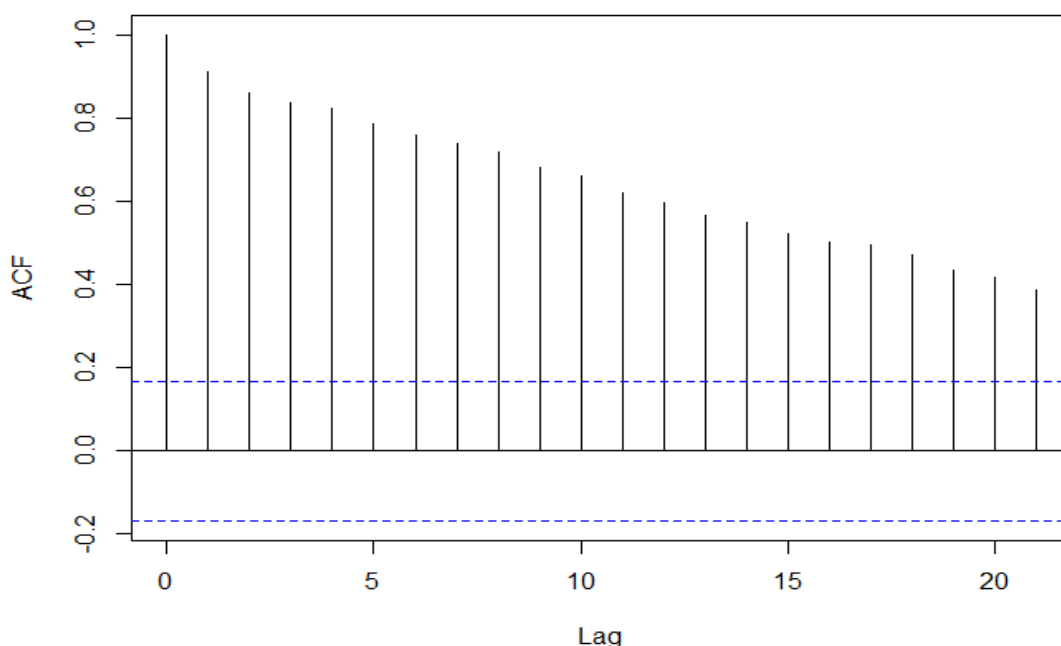
Autocorrelación

La autocorrelación no es más que una forma de adaptar la fórmula de la correlación a una sola serie. Autocorrelacionar la serie implica relacionar el pasado de la serie temporal con su valor actual. Esto se hace porque cuando una serie tiene correlación entre datos separados por los mismos intervalos, predecir los siguientes será mucho más sencillo. De hecho, este es el principio en el que se basan todos los modelos ARIMA.

Para obtener el valor de autocorrelación de la serie, no hay más que obtener la misma serie con un determinado desfase. Para ello, se debe crear una nueva serie eliminando el último parámetro de la original, a la que llamaremos $\{Y_{-1}\}$, y crear otra eliminando el primer parámetro $\{Y_{-n}\}$ (Este segundo paso solo se hace para que ambas series tengan el mismo número de datos). Una vez hecho esto, se podrá aplicar la fórmula de correlación a estas dos series para obtener una correlación de *lag*=1 (separadas por un intervalo): *cor*($\{Y_{-1}\}$, $\{Y_{-n}\}$). Para correlaciones entre mayores *lag* (mayores intervalos), habrá que repetir el mismo proceso, eliminando el mismo número de términos que *lag* que se deseen obtener.

Por suerte, R dispone de una función llamada *Auto Correlation Function*, que calcula la autocorrelación de todos los *lag* que se necesiten simultáneamente. Para ejecutarla, habrá que utilizar el comando *acf()*, y especificar tres argumentos: la serie de la que se desea obtener los datos, el número máximo de *lag* del que se querrá obtener la autocorrelación, y si será necesario que dibuje un gráfico con los valores (correlograma), o en su defecto, que devuelva el resultado numérico. En este trabajo siempre se optará por el correlograma, porque como se verá más adelante, será una herramienta de gran sencillez de comprensión y gran potencial de análisis. Un ejemplo de autocorrelograma se puede ver en la figura 11.

AUTOCORRELOGRAMA DE LA SERIE DE TEMPERATURAS GLOBALES



Fuente: Elaboración propia.

Figura 11. Autocorrelograma de la serie de temperaturas globales.

4.3.3.2 Modelos básicos

Antes de entrar en profundidad a explicar los modelos complejos, conviene que definir los modelos más sencillos de series temporales, que son los pilares fundamentales de los ARIMA.

Ruido Blanco

El ruido blanco (ε_t) es el proceso estacionario más simple que se puede encontrar dentro de estos modelos. Tiene las siguientes características:

- Media ajustada constante.
- Varianza ajustada constante.
- No tiene correlación a lo largo del tiempo.

En otras palabras, una serie de ruido blanco es una serie que toma valores aleatorios alrededor de una media, y no tiene ninguna dependencia de los valores que ha tomado la serie anteriormente. Por ello, una serie de ruido blanco se puede definir con dos únicos parámetros. Su media μ , y su varianza σ^2 .

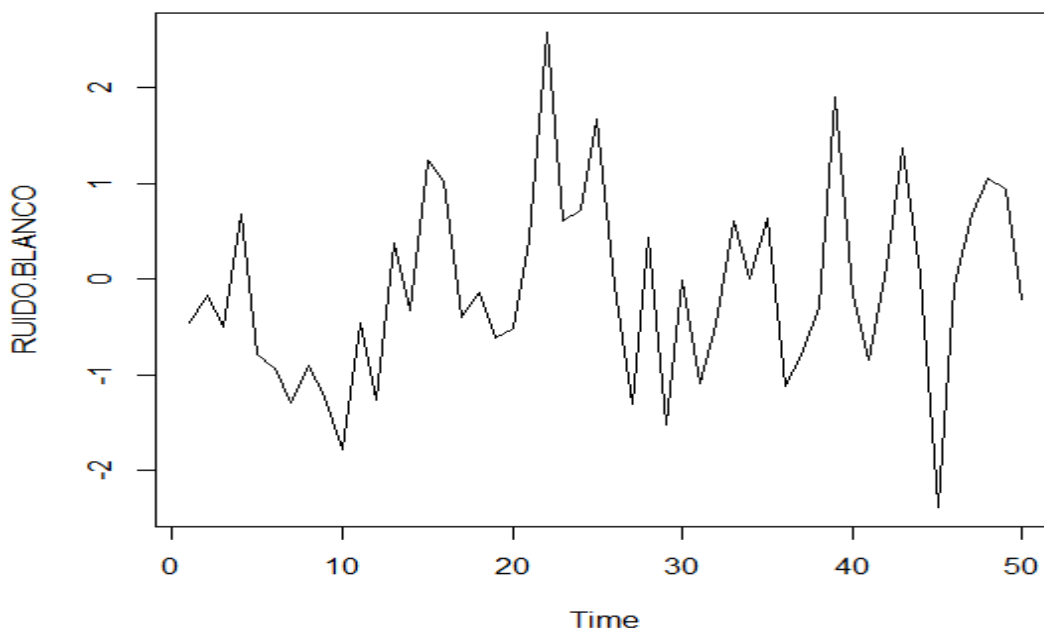
$$Y_t = \varepsilon_t$$

Un modelo de ruido blanco se corresponde con un ARIMA(0, 0, 0), algo que será explicado más adelante, pero que de momento es útil saber para introducir la función en R de simulación de diferentes modelos.

En caso de querer simular un modelo de ruido blanco en R, se puede acudir a la función *arima.sim()*, e introducir los parámetros mencionados en el párrafo anterior de acuerdo al modelo de ruido blanco.

```
> RUIDO.BLANCO<-arima.sim(model = list(order=c(0,0,0)),n=50)
> plot(RUIDO.BLANCO)
```

SERIE SIMULADA DE RUIDO BLANCO



Fuente: Elaboración propia.

Figura 12. Gráfico de una serie simulada de ruido blanco con 50 datos.

En la figura 12 se puede ver cómo la serie sigue un comportamiento normal alrededor de la media, siendo el cada valor de la serie independiente del anterior.

Paseo Aleatorio

Un modelo de paseo aleatorio no es más que un modelo integrado de ruido blanco. Por integrado se entiende que los valores de la serie son acumulativos, siguiendo la siguiente fórmula:

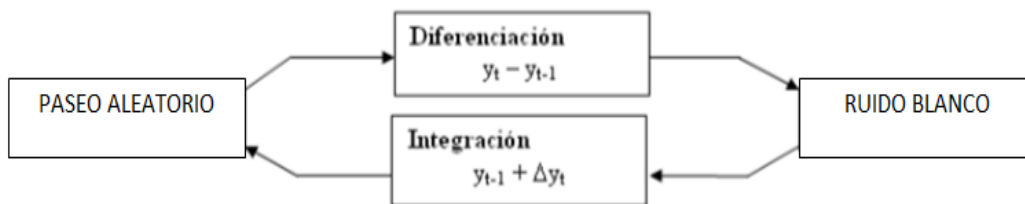
$$Y_t = Y_{t-1} + \epsilon_t$$

Los modelos de paseos aleatorios son el mejor ejemplo de procesos no estacionarios, y cumplen las siguientes características:

- No tiene media o varianza específica.
- Gran dependencia sobre el tiempo.
- La simulación requiere de un punto inicial (R toma 0 por defecto).

- Como ya se ha explicado, sus cambios o incrementos son ruido blanco.

Estas características hacen que una diferenciación de un modelo de paseo aleatorio dé como resultado un modelo de ruido blanco. Por esto mismo, y estando los modelos de ruido blanco y paseo aleatorio relacionados por la integración o la diferenciación, ambos dependen de los mismos parámetros, que como se ha visto visto en la descripción del caso anterior, son la media μ , y la varianza σ^2 .



Fuente: Elaboración propia.

Figura 13. Esquema de la relación del modelo de ruido blanco con el del paseo aleatorio.

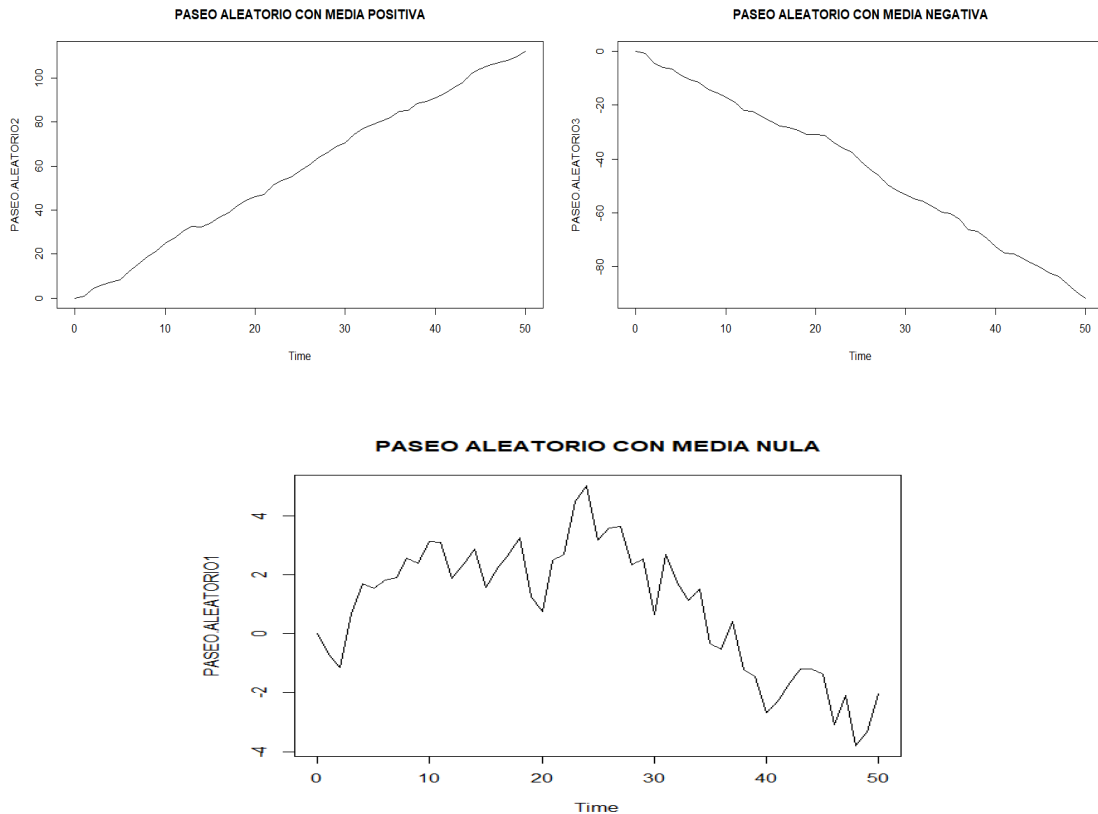
De esta relación, también se puede deducir que la tendencia en media de un paseo aleatorio depende de la media del ruido blanco, cumpliéndose las siguientes relaciones:

- Si $\mu > 0$, entonces la tendencia de la serie será positiva.
- Si $\mu = 0$, entonces la tendencia de la serie será nula.
- Si $\mu < 0$, entonces la tendencia de la serie será negativa.

Como se verá más adelante, un modelo que necesita de una diferenciación para ser estacionario tiene como parámetro $d=1$. Como en $ARIMA(p, d, q)$, la d corresponde al segundo parámetro, se deduce a partir del modelo $ARIMA$ del ruido blanco que el modelo para un paseo aleatorio es $ARIMA(0, 1, 0)$. Sabiendo esto, la simulación de un paseo aleatorio se realizará de la siguiente forma con el comando *arima.sim()*.

```

> PASEO.ALEATORIO1<-arima.sim(model = list(order=c(0,1,0)),n=50,mean=0)
> PASEO.ALEATORIO2<-arima.sim(model = list(order=c(0,1,0)),n=50,mean=2)
> PASEO.ALEATORIO3<-arima.sim(model = list(order=c(0,1,0)),n=50,mean=-2)
  
```



Fuente: Elaboración propia.

Figura 14. Comparativa de gráficos de paseos aleatorios con distintas medias de ruido blanco.

Modelo Autorregresivo AR(p)

Para simplificar y explicarlo de forma más clara, se comenzará con el modelo autorregresivo de primer orden o AR (1). Teniendo en cuenta una serie temporal con datos diarios, este modelo se puede definir como algo que se adapta a la siguiente fórmula:

$$\text{Hoy} = \text{Media} + \text{Pendiente} * \text{Ayer} + \text{Ruido Blanco}$$

Lo que en términos técnicos y ya generalizados a cualquier serie temporal se expresa como:

$$Y_t = \mu + \Phi_1 Y_{t-1} + \epsilon_t$$

Teniendo ya esto claro, cabe destacar que, si el parámetro Φ es 0, el modelo es idéntico al del ruido blanco. Lo que en el fondo quiere decir esto es que existe una dependencia entre el valor anterior de la serie y el actual, y dicha dependencia viene expresada por el parámetro Φ . El resto de variación se debe a perturbaciones aleatorias basadas en el modelo de ruido blanco. De este modo, se puede decir que la autocorrelación de esta serie viene definida en el parámetro.

En cuanto al modelo general $AR(p)$, este sigue el mismo principio, solo que la dependencia con los valores anteriores se extiende más allá del dato directamente anterior. De hecho, para conocer el orden del modelo, solo ha de analizarse la correlación parcial (en R se consigue con la función *pacf()*), que es lo mismo que la normal pero solo teniendo en cuenta la correlación directa entre dos factores, eliminando la correlación que pueda ser derivada por la dependencia entre la cadena de datos. Habrá que comprobar el número de correlaciones parciales relevantes inmediatamente anteriores para estimar el parámetro p . De este modo, la fórmula general del modelo queda de la siguiente forma:

$$Y_t = \mu + \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \dots + \Phi_n Y_{t-n} + \varepsilon_t$$

Sabiendo todo esto, y teniendo en cuenta los modelos anteriores, un modelo autorregresivo $AR(p)$ depende de sus parámetros $\Phi_1, \Phi_2, \Phi_3, \dots, \Phi_n$, de su media μ y de la varianza del ruido blanco σ^2 .

Para adaptar este modelo a series reales se puede utilizar la función de R *arima()*, sustituyendo el parámetro p por el orden del modelo AR que queremos utilizar, y dejando el resto de parámetros con valor 0. De esta forma se conseguirá que R encuentre el valor de los parámetros Φ que mejor se adapten a la serie. Una vez se tiene el valor de estos parámetros, se puede predecir el futuro de la serie aplicando la fórmula del modelo y tomando Y_t como el valor a predecir. Esto lo puede hacer R directamente con la función *predict()*.

```
> AR<-arima.sim(model = list(ar=0.4),n=50)
> AR_ARIMA<-arima(AR, order=c(1,0,0))
> predict(AR_ARIMA)
$pred
Time Series:
Start = 51
End = 51
Frequency = 1
[1] 1.507404

$se
Time Series:
Start = 51
End = 51
Frequency = 1
[1] 1.118383
```

Modelo de Media Móvil MA(q)

Al igual que se ha hecho con los modelos AR(p), en los modelos de media móvil o MA(q) se comenzará explicando el caso simple, de primer orden MA(1). Teniendo en cuenta series temporales de un día, este modelo se puede describir con la siguiente fórmula:

Hoy= media + pendiente*(ruido blanco de ayer) + ruido blanco

Que expresado en términos técnicos queda de la siguiente forma:

$$Y_t = \mu + \theta_1 \varepsilon_{t-1} + \varepsilon_t$$

Del mismo modo que en el caso anterior, si en el parámetro θ que define el modelo toma valor 0, una vez más éste vuelve a ser ruido blanco. Así, se puede deducir que el parámetro θ expresa la dependencia existente entre el ruido blanco del parámetro anterior y el actual. De este modo, existe una correlación entre los ruidos blancos, y por ello, el orden q del modelo se deduce viendo la dependencia que tienen los datos con los valores directamente anteriores, contando aquellos que guarden una correlación normal relevante. Por esta razón, se usa el correlograma normal para la identificación del valor parámetro q . En él habrá que ver el número de correlaciones relevantes inmediatamente anteriores para estimarlo

De este modo, el modelo MA(q) general se define con la siguiente fórmula:

$$Y_t = \mu + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_n \varepsilon_{t-n} + \varepsilon_t$$

Sabiendo todo esto, y teniendo en cuenta los modelos anteriores, un modelo autorregresivo MA(q) depende de sus parámetros $\theta_1, \theta_2, \theta_3, \dots, \theta_n$ de su media μ y de la varianza del ruido blanco σ^2 .

Al igual que con AR(p), este modelo a series reales se puede utilizar la función de R *arima()*, sustituyendo el parámetro q por el orden del modelo MA que queremos utilizar, y dejando el resto de parámetros con valor 0. De esta forma se consigue que R encuentre el valor de los parámetros θ que mejor se adapten a la serie. Una vez se obtiene el valor de estos parámetros, se predice el futuro de la serie aplicando la fórmula, de nuevo sustituyendo Y_t por el dato que se desea calcular. Esto lo puede hacer R, una vez más con la función *predict()*.

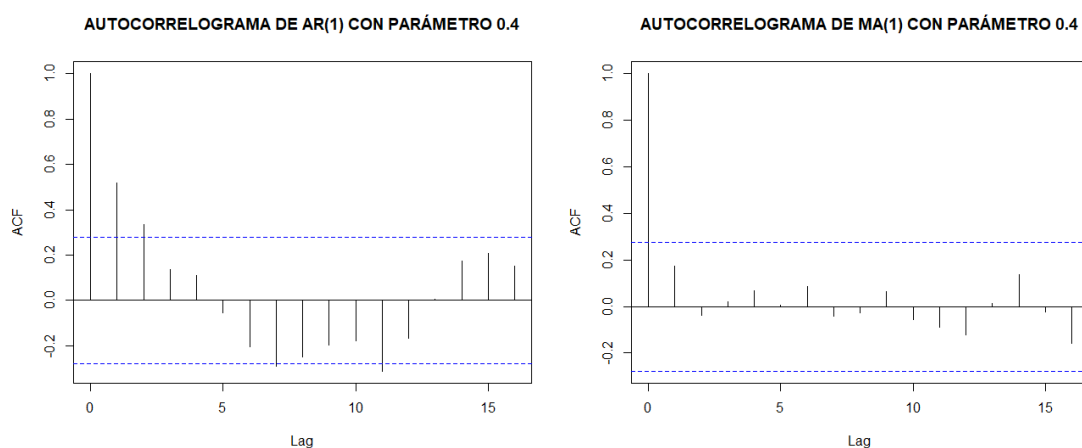
```
> MA<-arima.sim(model=list(ma=0.4),n=50)
> MA_ARIMA<-arima(MA, order = c(0,0,1))
> predict(MA_ARIMA)
$pred
Time Series:
Start = 51
End = 51
Frequency = 1
[1] 0.2434911

$se
Time Series:
Start = 51
End = 51
Frequency = 1
[1] 0.8823966
```

4.3.3.3 Comparación de modelos AR(p) y MA(q)

Llegado el momento de analizar una serie temporal real, habrá que enfrentarse a la tesitura de qué modelo elegir. Para esto es necesario conocer perfectamente los modelos posibles a ser elegidos. Con respecto a los modelos AR(p) y MA(q), la identificación de las series que pueden adaptarse a estos modelos puede servir de introducción para explicar los criterios que se utilizarán con ARIMA.

En primer lugar, los diagramas de autocorrelaciones son la principal herramienta para estimar un modelo que se pueda adecuar a la serie sometida a estudio. Tomando las series generadas anteriormente para ambos ejemplos se van a generar los correlogramas con los que profundizaremos en las propiedades de estas series.



Fuente: Elaboración propia.

Figura 15. Comparativa de autocorrelogramas normales de series con comportamiento autorregresivo y series de comportamiento de media móvil.

Como se puede ver en la figura 15, en el caso del modelo autorregresivo la correlación es decreciente pero todavía relevante con los valores anteriores, decayendo poco a poco. En el modelo de medias móviles el correlograma se corta en un valor determinado y el resto deja de tener relevancia. (La relevancia de la correlación es indicada por las bandas de significación, representadas con líneas azules en los correlogramas de R, que en el caso de este trabajo se van a considerar siempre constantes, con un valor de 0'2, aunque se pueden aplicar muy diversos criterios para esto). Esto es porque en los modelos AR, el correlograma siempre tenderá a caer poco a poco hasta perder la relevancia. En los modelos MA, el correlograma mostrará relevancia hasta exactamente el lag equivalente al orden del modelo, en este caso el primero.

A pesar de esto, el correlograma puede no ser suficiente para identificar el modelo. Para ello, existen dos parámetros que sirven para decidir entre uno y otro. Son el *AIC* (Akaike Information Criterion) y el *BIC* (Bayesian Information Criterion).

Más adelante se profundizará en la explicación del funcionamiento de estos criterios. Sin embargo, basta con saber que los modelos más adecuados serán los que tengan menores valores de *BIC* y de *AIC*.

5. METODOLOGÍA CON MODELOS ARIMA EN R

5.1. DESARROLLO TEÓRICO DE ARMA

Modelos ARMA

Herman Wold, miembro de la *Royal Swedish Academy of Sciences*, demostró que cualquier serie temporal puede expresarse como combinación lineal de todos los ruidos blancos anteriores.

Cualquier modelo ARMA (*Autoregressive Moving Average*) tiene estas características, lo que quiere decir que cualquier serie temporal tiene un correspondiente modelo ARIMA al que puede ser adaptado de forma perfecta. Evidentemente no se busca dicho modelo a la hora de aplicarlos, pero sí una aproximación de órdenes bajos que dé unos resultados con un error despreciable.

La gran característica de los modelos ARMA(p, q) es que integran los modelos autorregresivos con los de media móvil. Es decir, tienen en cuenta tanto la dependencia con los valores anteriores, como la dependencia con los ruidos blancos anteriores. Un modelo ARMA (1, 1) tiene la siguiente forma:

$$Y_t = \mu + \Phi_1 Y_{t-1} + \varepsilon_t + \theta_1 \varepsilon_{t-1}$$

De esta forma, se puede apreciar que al igual que en los modelos descritos anteriormente, el parámetro Φ tiene en cuenta la dependencia con el valor anterior (parte autorregresiva), mientras que el parámetro θ se encarga de tener en cuenta la dependencia con el ruido blanco generado anteriormente (parte de media móvil). Sintetizando, el orden AR acude directamente al valor anterior de la serie, mientras que el orden MA toma en cuenta la relación con la perturbación aleatoria generada justo antes.

Teniendo ya claro a que se refiere cada uno de los parámetros, la fórmula general de un modelo ARMA es la siguiente:

$$Y_t = \mu + \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \dots + \Phi_n Y_{t-n} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_n \varepsilon_{t-n}$$

5.2. IDENTIFICACIÓN DEL MODELO

Uso de los correlogramas

Como ya se ha comentado anteriormente, el correlograma será la herramienta inicial para identificar modelos con ARIMA. Para ello se utilizará en R la función *acf2()*, incluida en el paquete *astsa*, que genera automáticamente el correlograma con dos gráficos, uno para ver la autocorrelación pura o normal de la serie, y otra para ver la autocorrelación parcial. Será necesario especificar en un segundo argumento los *lag* necesarios a los que se querrá que esta función analice nuestra serie. Usualmente con unos 50 datos suele bastar para ver todas las propiedades, incluyendo las de los modelos ARIMAxSARIMA en su totalidad, con correcciones de estacionalidad incluidas, pero por ahora se explicará el análisis de los correlogramas para aplicar modelos ARMA.

En la figura 16 se recogen los métodos para diferenciar entre los modelos AR(*p*), MA(*q*) y ARMA (*p*, *q*):

	AR(<i>p</i>)	MA(<i>q</i>)	ARMA(<i>p</i> , <i>q</i>)
Correlaciones	Decaen de manera progresiva	Pierden relevancia abruptamente en el <i>lag</i> igual al orden	Decaen de manera progresiva
Correlaciones parciales	Pierden relevancia abruptamente en el <i>lag</i> igual al orden	Decaen de manera progresiva	Decaen de manera progresiva

Fuente: Elaboración propia.

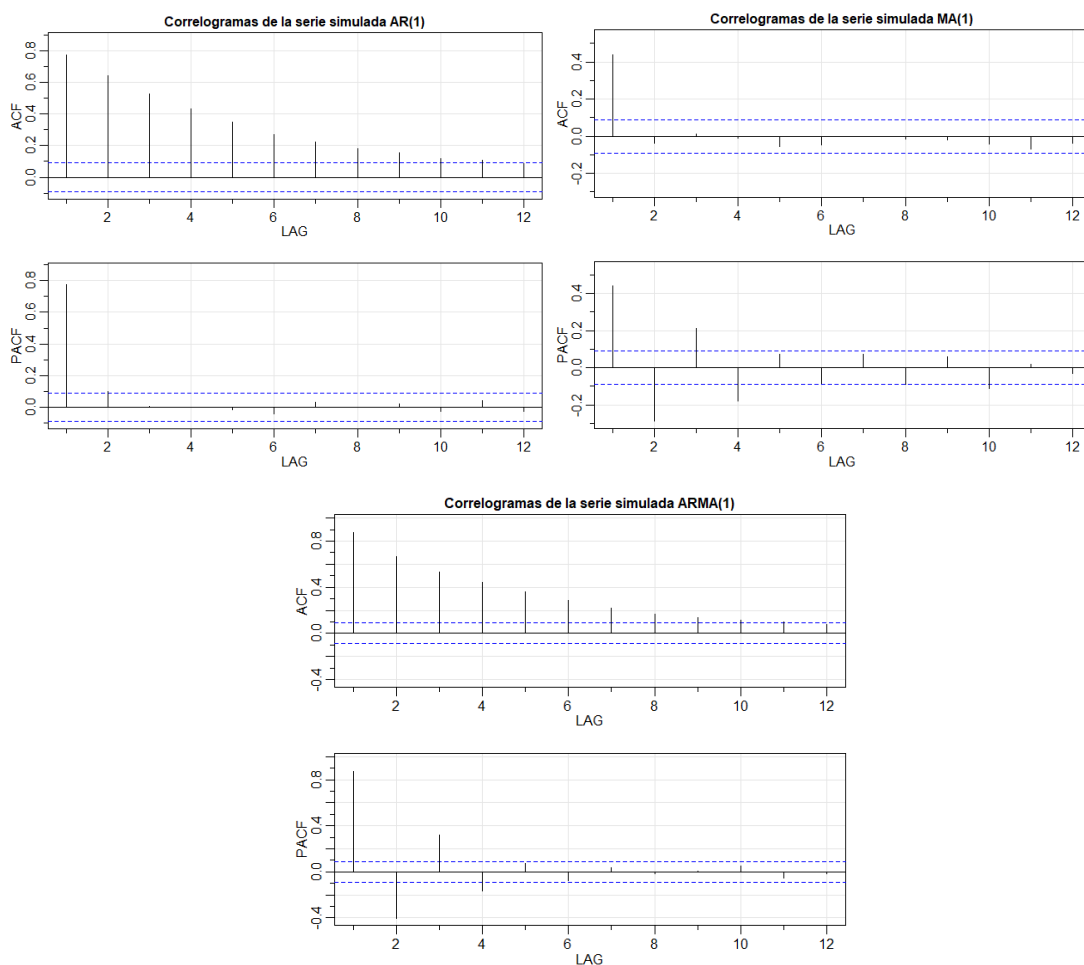
Figura 16. Tabla descriptiva de los comportamientos de los correlogramas en función al tipo de serie temporal.

Como se puede ver en la figura 16, los modelos $AR(p)$ presentan una autocorrelación relevante en los primeros *lag*, sin embargo, en la autocorrelación parcial ésta desaparece abruptamente en el *lag* igual al orden del modelo. En cuanto al modelo $MA(q)$, el resultado es justo inverso, siendo el diagrama de autocorrelación normal el que pierde la significación abruptamente en el *lag* de número igual al orden del modelo, mientras que la autocorrelación parcial es relevante en los *lag* inmediatamente anteriores, diluyéndose poco a poco. En cuanto a los modelos ARMA, ambos correlogramas presentan un decrecimiento progresivo según se avanza en el análisis de los *lag*.

En los figura 17 se tienen ejemplos de los correlogramas que representarían series $AR(1)$, $MA(1)$ Y $ARMA(1, 1)$. Para la generación de estos correlogramas se han utilizado las series $MA(1)$ y $AR(1)$ expuestas anteriormente en la explicación de estos modelos concretos. Para el ejemplo de ARMA se ha generado una nueva serie en R, de nuevo con la función *arima.sim()*, dando a ambos valores, p y q , valor de 1.

Para ARMA (1,1)

```
> ARMA<-arima.sim(model = list(ar=0.75,ma=0.75),n=500)
> acf2(ARMA, main = "Correlogramas de la serie simulada ARMA(1)",max.lag=12)
      ACF  PACF
[1,] 0.87  0.87
[2,] 0.67 -0.40
[3,] 0.53  0.32
[4,] 0.44 -0.16
[5,] 0.36  0.07
[6,] 0.28 -0.08
[7,] 0.22  0.04
[8,] 0.17 -0.01
[9,] 0.13  0.00
[10,] 0.12  0.05
[11,] 0.10 -0.05
[12,] 0.07 -0.02
```



Fuente: Elaboración propia.

Figura 17. Comparativa de los comportamientos de los autocorrelogramas normal y parcial de los modelos AR(1), MA(1) y ARMA(1, 1).

5.3. CRITERIOS *AIC* Y *BIC*

Una vez explicado el método para diferenciar entre estos tres tipos de modelos, podría surgir la pregunta de cómo podrían identificarse las series que se adecúen a modelos de órdenes superiores en ARMA. La respuesta es que no existe ningún método más innovador que el de iterar con cada modelo e ir analizando los resultados que se obtienen. Este análisis se lleva a cabo principalmente con los ya mencionados anteriormente números *AIC* (Akaike Information Criterion) y *BIC* (Bayesian Information Criterion). Además de tener en cuenta el error cometido por la estimación, estos números tienen en cuenta el número de parámetros de cada modelo, ponderando la elección de aquellos que tengan una menor cantidad. El objetivo será minimizar lo máximo posible el valor de ambos.

En cuanto a los errores, en el *argot* de series temporales son conocidos como residuos, o *residuals* en inglés. Estos errores son los achacados al ruido blanco, y en función al modelo adoptado la serie tendrá diferentes valores para éstos, siendo los de menor valor total generalmente los más óptimos.

El número de parámetros se tiene en cuenta en el cálculo de los números *BIC* y *AIC* multiplicándolo por un número *k*. La forma de calcular este número es lo que diferencia a ambos criterios, siendo el valor de $k = 2$ en el caso del criterio *AIC*, y $k = \log(n)$ en el caso del *BIC*. De esta forma, la fórmula general para calcular estos números es la siguiente:

$$\underbrace{\left(\frac{\text{Observado} - \text{Predicho}}{n} \right)^2}_{\text{Media de los residuos al cuadrado}} + \underbrace{k * (p + q)}_{K * \text{Parámetros}}$$

Media de los residuos al cuadrado

K*Parámetros

En cuanto a *AIC* y *BIC*, en caso de ser uno menor para un modelo y otro para otro, ambos resultarán igualmente óptimos según estos criterios. Siendo esto así, se procederá a un análisis de residuos, y, en caso de ser igualmente óptimos y no apreciar diferencias significativas, se escogerá el que menos términos tenga.

En el paquete *astsa* de R existe la función *sarima()*, una mejora de la ya mencionada función *arima()*, que otorga también el valor de estos dos criterios para el modelo deseado. Así, aplicando, por ejemplo, un modelo ARMA(1, 1) a la serie del PIB de EEUU *gnp* (con su correspondiente tendencia lineal eliminada), también contenida en el mismo paquete, se puede ver el resultado de ambos criterios:

```
> sarima(diff(gnp), p=1, d=0, q=1, P=0, D=0, Q=0)
```



```
$ttable
      Estimate      SE t.value p.value
ar1      0.7024 0.0953  7.3688  0.0000
ma1     -0.3705 0.1208 -3.0658  0.0024
xmean   36.0762 5.3911  6.6917  0.0000
```

```
$AIC
[1] 10.16457
```

```
$AICc
[1] 10.16506
```

```
$BIC
[1] 10.22588
```

La función *sarima()*, al igual que la función *arima()*, tienen el cometido principal de calcular el valor de los parámetros Φ_n y θ_n . La matemática detrás de este cálculo no será estudiada en este trabajo, ya que R es capaz de realizarla por sí mismo, aunque basta con saber que se hace por medio de iteraciones, haciendo que el error (residuos) se minimice lo máximo posible.

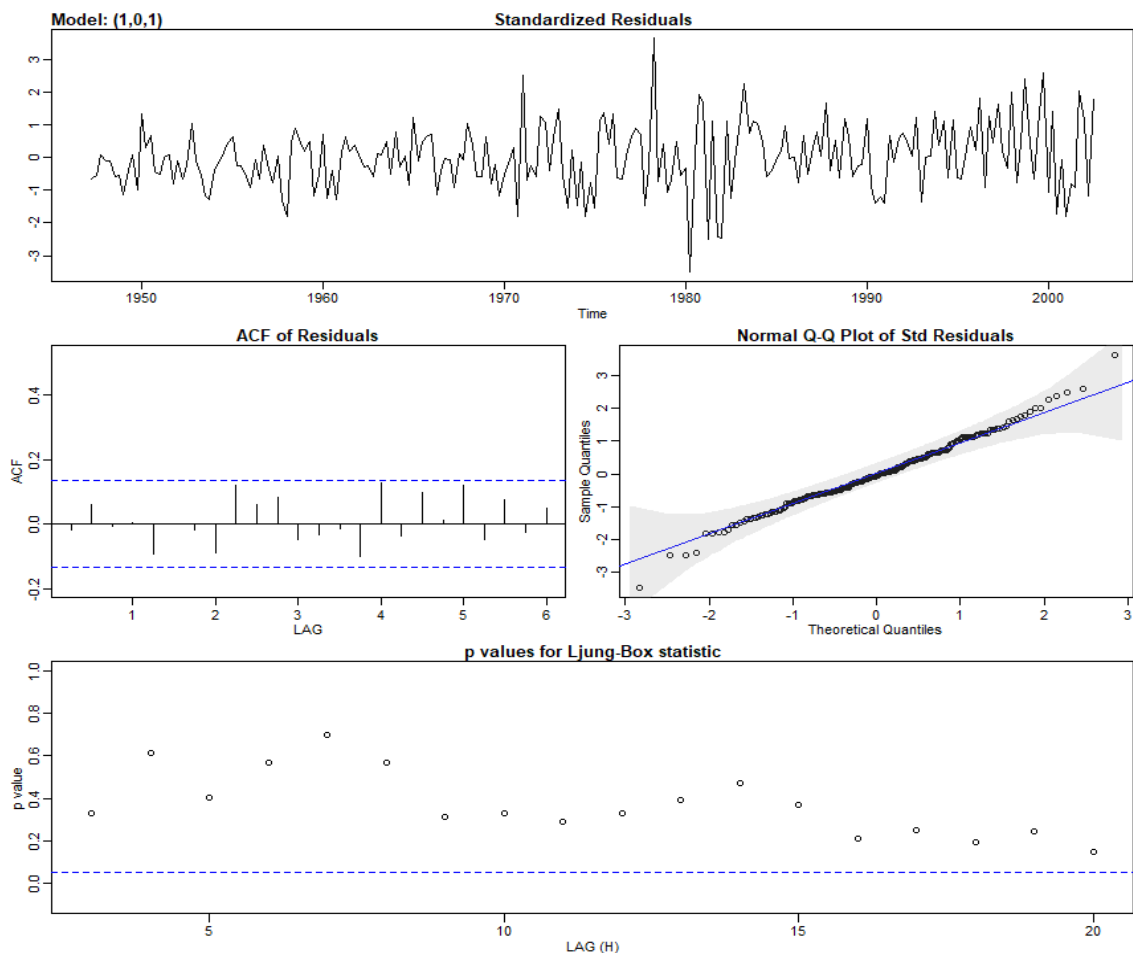
5.4. ANÁLISIS DE RESIDUOS

A pesar de todo esto, el análisis de estos resultados no resulta suficiente para saber si el modelo escogido es adecuado, ya que hay que asegurarse de que el comportamiento de los residuos sea válido, de tal forma que se comporte como ruido blanco. Para ello la función *sarima()* nos genera, además de muchos otros datos como los criterios *BIC* y *AIC*, cuatro gráficos para analizar si los residuos se comportan como ruido blanco.

El significado de estos gráficos es el siguiente:

- **Residuos estandarizados:** Este gráfico no es más que una representación del valor de los residuos. Deberá examinarse si hay periodos en él, además de que tenga media 0 y varianza constante.
- **Autocorrelograma de los residuos:** Para ver cómo de blancos son esos residuos. Al menos el 95% de los valores debería estar entre las bandas de significación para poder asegurar un comportamiento de ruido blanco.
- **Gráfico normal Q-Q:** Dirá si los residuos son normales. De ser así, se alinearán con la recta. Cabe destacar que lo más común es encontrar valores alejados de la línea en los extremos de este gráfico.

- **Estadístico Q, valores p:** Es un test estadístico que mide la pureza de los residuos. Si los puntos (o la mayoría) están por encima de la línea azul estos tendrán el comportamiento adecuado.



Fuente: Elaboración propia.

Figura 18. Gráficos de evaluación de residuos para la serie del PIB del EEUU del paquete *astsa* aplicando un ARIMA(1, 0, 1).

En la figura 18 se comprueba que en el caso anterior, aplicar un ARMA(1,1) a la serie diferenciada del PIB de EEUU es apropiado, dado que los residuos se comportan como ruido blanco. Sería conveniente tratar de buscar otros modelos con un *BIC* o *AIC* más bajo, pero aun así, este modelo podría dar predicciones bastante adecuadas.

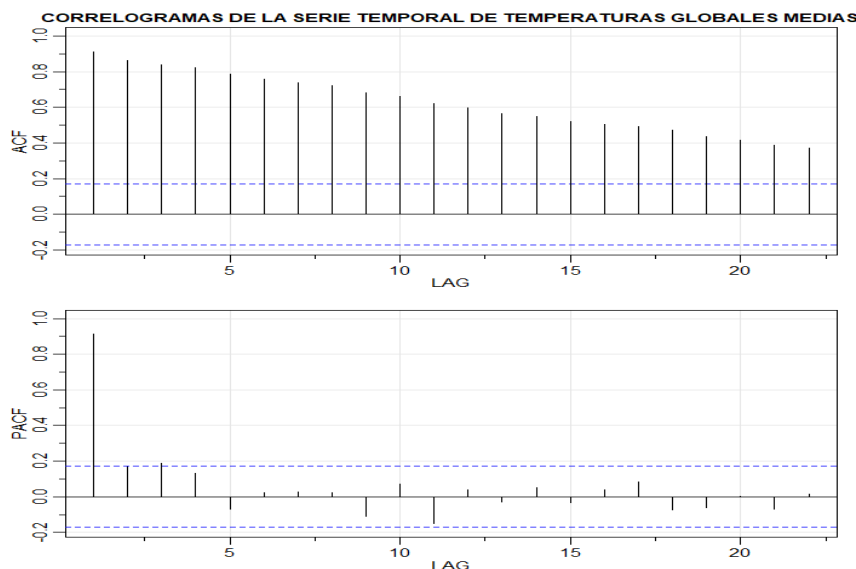
5.5. MODELOS ARIMA AL COMPLETO

5.5.1. ARIMA

La diferencia entre los modelos ARIMA (*Autoregressive Integrated Moving Average*) y los ARMA es que estos últimos no incluyen la diferenciación en su desarrollo. Como ya se ha explicado, la tendencia en media es eliminada por medio de diferenciaciones, dependiendo del orden al que se adapte la curva de aproximación de la serie temporal a analizar. El parámetro d de los modelos ARIMA hace referencia al número de diferenciaciones necesarias.

En caso de dudar de la función gráficamente o de no poder apreciar si esta cumple la condición de estacionariedad en media, se puede recurrir al autocorrelograma de la serie, aplicando, una vez más, la función $acf2()$.

Un correlograma típico de una serie que necesita diferenciación es el siguiente:



Fuente: Elaboración propia.

Figura 19. Autocorrelograma de la serie de temperaturas globales del paquete *astsa*.

Observando la figura 19, en una serie con tendencia en media ya representada anteriormente, como es la de la temperatura global media de la tierra, la autocorrelación entre distintos *lag* anteriores es muy elevada y decae muy lentamente. Este comportamiento se repite en todas las series con tendencia lineal.

5.5.2. SARIMA

Estacionalidad pura. SARIMA (P, D, Q)

Es muy común que al recoger datos se encuentren patrones estacionales, siendo los más habituales los patrones que se dan anualmente en series de forma trimestral o anual. Existen unos modelos particulares de ARIMA para modelar las series de forma estacional. Son los conocidos como SARIMA(P, D, Q) (*Stational Autoregressive Integrated Moving Average*). La única diferencia entre ARIMA y SARIMA es que este último se aplica de igual manera, pero en intervalos de tamaño igual a s (hay que recordar que s representa el número de datos que hay en un ciclo, por ejemplo, meses en un año). Es decir, la diferenciación se hace entre intervalos de s (como ya se había comentado en el apartado de tendencia estacional), los parámetros Φ son igual a 0 en *lag* no estacionales, y los parámetros θ , también.

Siendo todo esto así, el correlograma con el que se identificaría una serie temporal estacional pura (quiere decir que no existe ningún tipo de dependencia entre valores que no correspondan al mismo periodo en los datos) sigue el mismo principio que el de los modelos ARIMA, solo que con los valores de los *lag* no correspondientes a los múltiplos de s no deberán tener relevancia.

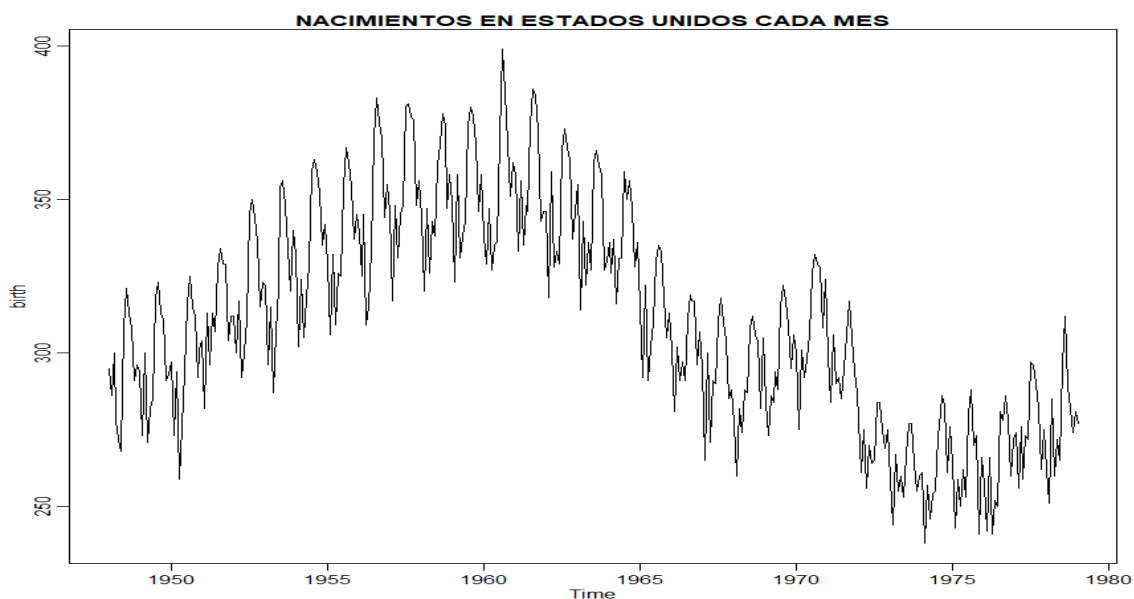
Estacionalidad Mixta ARIMA(p, d, q)xSARIMA(P, D, Q)

La estacionalidad mixta es el caso más frecuente que se puede encontrar en análisis de series temporales. Todas las series tienen comportamiento adaptable a ARIMA, de mejor o peor manera, pero de la misma forma, muchas de las series temporales del mundo real tienen algún tipo de carácter estacional. Es por ello que la mayoría de las veces es necesario aplicar los modelos ARIMA y SARIMA juntos, lo que se consigue analizando la serie temporal como se ha explicado a lo largo de este trabajo, tanto de forma natural analizando todos los *lag*, como de forma estacional, viendo si existe algún patrón de correlaciones entre los *lag* separados más de un intervalo, ignorando todos los valores que no se encuentren en múltiplos de s .

En función a todo esto, con el correlograma se estimaría el modelo ARIMA(p, d, q)xSARIMA(P, D, Q). Una vez hecho esto, hay que iterar y elegir el modelo iterando entre los valores que mejor se aproximen, tomando ayuda de los criterios *AIC* y *BIC*,

además de los gráficos de análisis de residuos de los que nos provee la función *sarima()*.

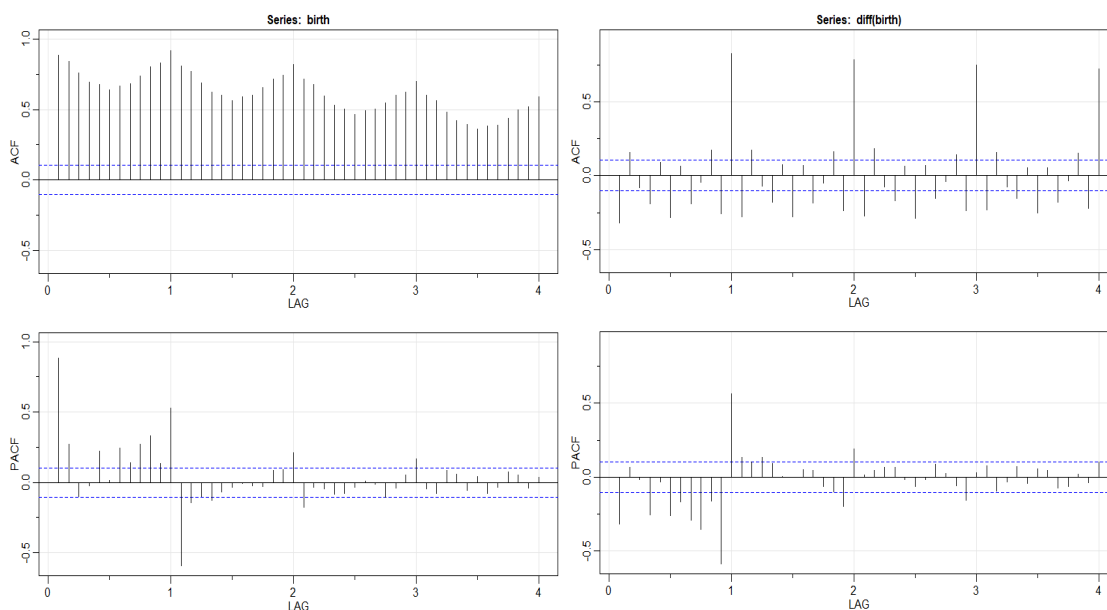
A continuación, se va a poner un ejemplo para ver cómo se identifica un modelo con componentes tanto ARIMA como SARIMA. Se ha elegido la función *birth*, del paquete *astsa*, que ofrece datos del número de la natalidad en EEUU de forma mensual.



Fuente: Elaboración propia.

Figura 20. Gráfico de la función *birth* del paquete *astsa*.

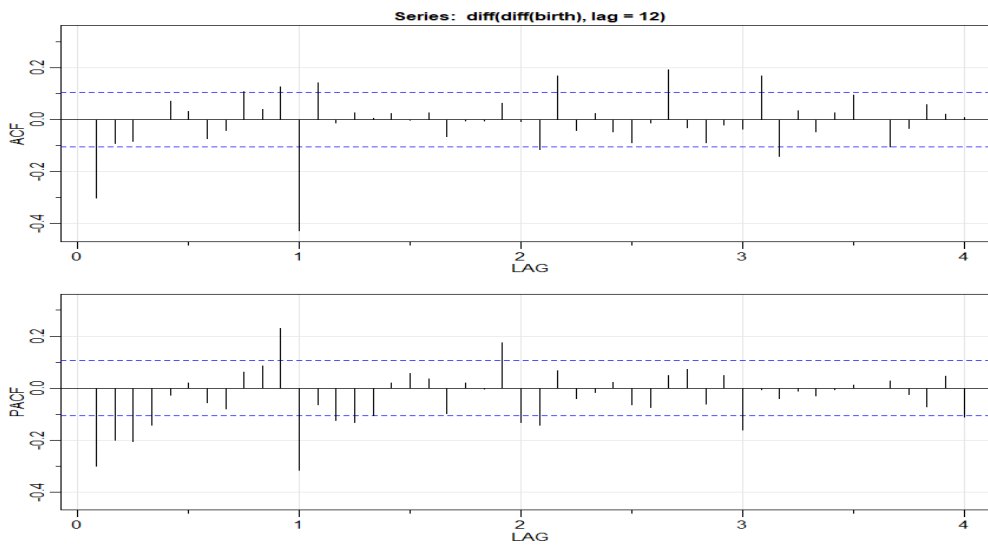
En la figura 20 se comprueba que la serie tiene una tendencia tanto en media como estacional. El correlograma es el típico de tendencias en media lineales, por lo que lo primero que habrá que hacer para proceder a su análisis es diferenciarlo una vez. De este modo el parámetro *d* tendrá valor 1 para la serie original.



Fuente: Elaboración propia.

Figura 21. Comparativa de los autocorrelogramas de la serie *birth* del paquete *astsa* normal y diferenciada.

Una vez aplicada la diferenciación (viendo el segundo gráfico de la figura 21 puede ser que hiciese falta alguna más, pero al ser un ejemplo no se va a proceder a la solución óptima) se ven las ya comentadas autocorrelaciones relevantes en los *lag* estacionales, por lo que habría que diferenciar con $s=12$, pasando el parámetro D a valer 1 para la serie original. Se dispondría de la serie ya estacionaria, con todo preparado para probar los modelos probando con diferentes parámetros interpretando los correlogramas. Por un lado, todos los *lag*, y por otro, únicamente los *lag* estacionales.



Fuente: Elaboración propia.

Figura 22. Autocorrelograma de la serie *birth* del paquete *astsa* difeneciada de forma normal y estacional con *lag=12*.

Interpretándolo con todo lo que se ha visto, a priori, un modelo que podría ser adecuado sería un $ARIMA(1,1,1) \times SARIMA(0,1,1)$ con $s=12$.

6. DESARROLLO PRÁCTICO

6.1. EJEMPLO DE PREDICCIÓN DE UNA SERIE. PIB DE LA CAPV

Para ilustrar el proceso que se va a llevar a cabo con cada una de las series analizadas en el trabajo, se va a tomar la serie del PIB de la Comunidad Autónoma del País Vasco.

Se ha escogido esta serie porque a pesar de no tener una relación directa con el objetivo, la idea de realizar el TFG sobre predicción con modelos ARIMA fue en base a una proposición inicial para intentar adecuar un modelo a la serie de PIB de la CAPV y realizar proyecciones sobre ésta. Además, la serie sirve como ejemplo para integrar todas las características que se han explicado previamente (estacionalidad, tendencia en media...), pudiendo ser una transición adecuada para demostrar lo que se ha explicado teóricamente en cuanto a modelos ARIMA, y las predicciones que se van a llevar a cabo de los índices. Por ello, se va a detallar paso a paso lo que será repetido en cada uno de los análisis de la parte práctica para que no sea necesario aclararlo en cada uno de ellos.

1. Introducción de la serie en R y transformación a objeto de tipo *ts*.

Lo primero que hay que hacer es obtener los datos sobre los que se va a proceder al análisis. En este caso, se ha accedido a la página web de Eustat, donde es posible descargar una hoja de cálculo con los datos del PIB de la CAPV desde el primer trimestre de 1995 hasta el primer trimestre de 2019, expresados en forma trimestral, es decir, con una frecuencia anual de 4 trimestres. Los datos de la hoja de Excel extraídos están expresados como representa la figura 23.

		1995-1	1995-2	1995-3	1995-4	1996-1	1996-2	1996-3	1996-4	1997-1	1997-2
Datos brutos	Precios corrientes (miles euros)	7206534	7185481	7154999	7503393	7618383	7716368	7670349	7981771	8021794	8163334
	Índice de volumen encadenado	63,2	62,4	61,3	63,9	64,6	64,2	63,3	66	66,1	67
Datos corregidos de efectos estacionales y de	Precios corrientes (miles euros)	7119681	7193140	7302548	7435037	7579188	7710355	7793763	7903565	8009048	8197044
	Índice de volumen encadenado	62,1	62,4	62,7	63,6	64	64,2	64,7	65,2	66	67,2
Ciclo-tendencia	Precios corrientes (miles euros)	7191298	7191298	7305057	7435881	7579789	7703504	7800317	7896271	8018959	8197434
	Índice de volumen encadenado	61,9	62,3	62,9	63,5	63,9	64,3	64,7	65,2	66	67,1
Datos brutos	Precios corrientes (miles euros)	-	-	-	-	5,7	7,4	7,2	6,4	5,3	5,8
	Índice de volumen encadenado	-	-	-	-	2,2	2,9	3,3	3,3	2,4	4,4
Datos corregidos de efectos estacionales y de	Precios corrientes (miles euros)	-	-	-	-	6,5	7,2	6,7	6,3	5,7	6,3
	Índice de volumen encadenado	-	-	-	-	3	2,9	3,2	2,6	3,2	4,6
Ciclo-tendencia	Precios corrientes (miles euros)	-	-	-	-	5,4	7,1	6,8	6,2	5,8	6,4
	Índice de volumen encadenado	-	-	-	-	3,3	3,1	2,9	2,8	3,3	4,5
Datos brutos	Precios corrientes (miles euros)	-	-0,3	-0,4	4,9	1,5	1,3	-0,6	4,1	0,5	1,8
	Índice de volumen encadenado	-	-1,2	-1,8	4,2	1	-0,6	-1,4	4,2	0,2	1,4
Datos corregidos de efectos estacionales y de	Precios corrientes (miles euros)	-	1	1,5	1,8	1,9	1,7	1,1	1,4	1,3	2,3
	Índice de volumen encadenado	-	0,4	0,6	1,3	0,6	0,4	0,9	0,7	1,2	1,8
Ciclo-tendencia	Precios corrientes (miles euros)	-	0	1,6	1,8	1,9	1,6	1,3	1,2	1,6	2,2
	Índice de volumen encadenado	-	0,8	0,9	1	0,7	0,5	0,6	0,9	1,3	1,6

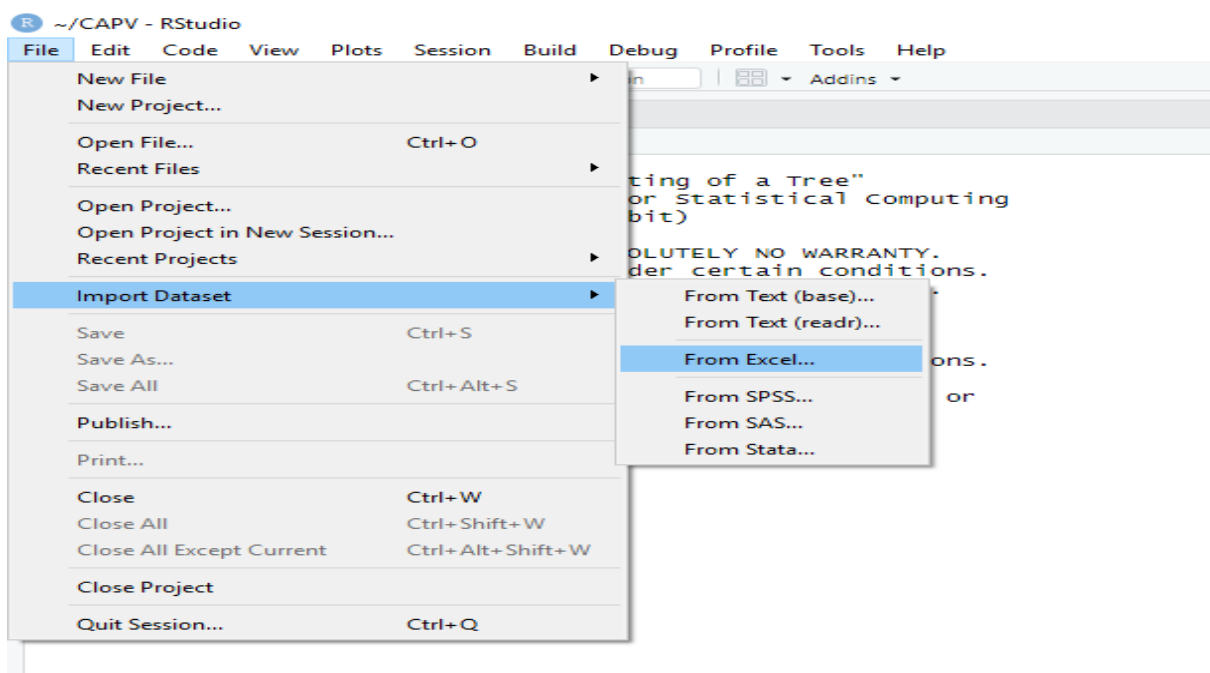
Fuente: Elaboración propia.

Figura 23. Vista preliminar de los datos del PIB de la CAPV en Excel previo a la importación a R.

En el caso a analizar en este trabajo, solo interesan los datos brutos de los precios corrientes, es decir, la primera línea, por lo que únicamente se importarán éstos a R.

Por suerte, las nuevas versiones de *R Studio* cuentan con herramientas para pasar datos desde hojas de cálculo al *Workspace* sin que sea necesario escribir más código, guardando los datos como un objeto de tipo *data frame*, por lo que una vez se tengan los datos en R la transformación a objeto de tipo *ts* es muy sencilla:

Primero se accede a la barra de tareas para importar datos desde Excel.



Fuente: Elaboración propia.

Figura 24. Ruta para importar datos de Excel a R.

El programa escribe automáticamente el código correspondiente para importar los datos en formato *data frame*.

```
> library(readxl)
> PIB <- read_excel("F:/Escritorio/PIB.xlsx",
+   sheet = "cet04tb", col_names = FALSE,
+   col_types = c("numeric", "numeric", "numeric",
+     "numeric", "numeric", "numeric",
+     "numeric", "numeric", "numeric",
+     "numeric", "numeric", "numeric",
+     "numeric", "numeric", "numeric",
+     "numeric", "numeric", "numeric",
```

Será necesario extraer los datos que necesitamos de todo el *data frame*. Para ello se escribe el siguiente código, y se tienen entonces los datos en forma de objeto *list*. Después, será necesario transformarlo a objeto *array*.

```
> VECTORDEPRECIOS<-c(PIB[,1:97])
> str(VECTORDEPRECIOS)
List of 97
 $ ...1 : num 7206534
 $ ...2 : num 7185481
 $ ...3 : num 7154999
 $ ...4 : num 7503393
 $ ...5 : num 7618383
 $ ...6 : num 7716368
 $ ...7 : num 7670349
```

```
> mivector<-unlist(VECTORDEPRECIOS)
```

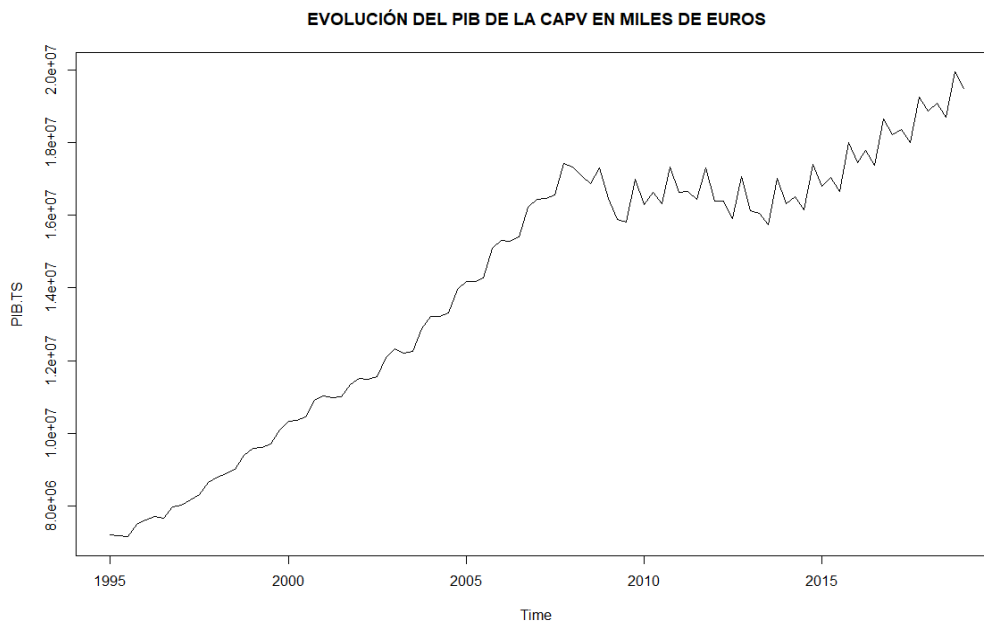
Una vez conseguido, se transforman los datos a tipo *time series* (*ts* en R), y se realiza la comprobación de que se ha realizado adecuadamente. De este modo, ya está todo preparado para aplicar toda la teoría explicada en los apartados anteriores.

```
> PIB.TS<-ts(mivector,start = 1995,frequency = 4)
> class(PIB.TS)
[1] "ts"
> print(PIB.TS)
      Qtr1    Qtr2    Qtr3    Qtr4
1995  7206534  7185481  7154999  7503393
1996  7618383  7716368  7670349  7981771
1997  8021794  8163334  8307376  8644291
1998  8790156  8885439  9001973  9391666
1999  9576483  9617400  9697120 10101923
2000 10337363 10367889 10441891 10906529
2001 11031896 10973401 10996389 11336800
2002 11506794 11484375 11568657 12098806
2003 12328444 12209303 12267045 12873075
2004 13214234 13228181 13316551 13969764
2005 14171200 14186180 14282499 15093135
2006 15317505 15279146 15394484 16214062
2007 16427243 16461505 16566745 17426435
2008 17318511 17083343 16876347 17296932
2009 16450770 15883060 15821242 16988255
2010 16302105 16623895 16321209 17322163
2011 16620036 16652751 16440276 17309257
2012 16386232 16397515 15907677 17071778
2013 16124373 16063047 15729922 17005667
2014 16313852 16515198 16147075 17402110
2015 16801159 17045315 16646826 18003277
2016 17440360 17779022 17382736 18637173
2017 18219145 18367716 17996858 19237459
2018 18862595 19080608 18705533 19956097
2019 19492407
```

2. Análisis gráfico de la serie

Una vez accesible la serie temporal, se debe proceder a la primera exploración del gráfico, observando posibles tendencias, heterocedasticidad o estacionalidad. El gráfico de la serie, ya dentro de R, se representa en la figura 25.

```
> plot(PIB.TS,main="EVOLUCIÓN DEL PIB DE LA CAPV EN MILES DE EUROS")
```



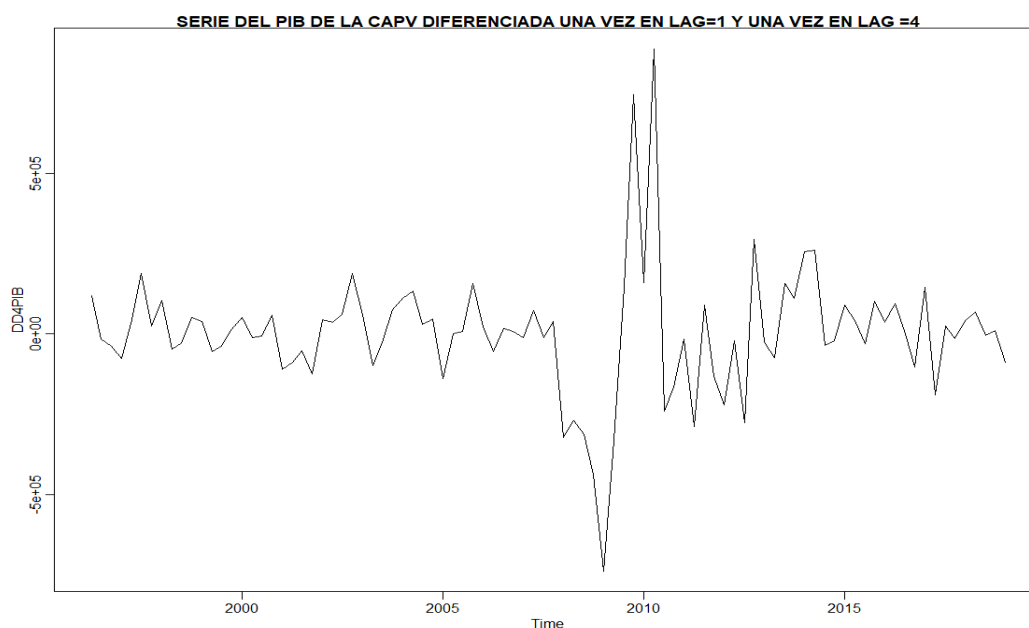
Fuente: Elaboración propia.

Figura 25. Gráfico de la evolución de los datos brutos del PIB de la CAPV desde 1995-1T hasta 2019-1T

Observando la figura 25 la serie parece tener tendencia en media, generalmente creciente, aunque no demasiado lineal, por lo que habrá que realizar al menos una diferenciación para eliminarla. Además, la serie parece tener un carácter prácticamente homocedástico, por lo que no será necesaria, en principio, una transformación logarítmica. Por último, con respecto a la estacionalidad, al ser datos cuatrimestrales se puede ver una tendencia estacional con $s=4$ (cuatrimestral) bastante pronunciada, especialmente en los últimos años, por lo que también parece evidente que habrá que diferenciar con $lag=4$ al menos una vez.

A continuación, se realizarán dichas transformaciones y se comprobará gráficamente si la serie cumple las condiciones de estacionariedad débil:

```
> DPIB<-diff(PIB.TS)
> DD4PIB<-diff(DPIB,4)
> plot(DD4PIB, main="SERIE DEL PIB DE LA CAPV DIFERENCIADA UNA VEZ EN LAG=1 Y UNA VEZ EN LAG =4")
```



Fuente: Elaboración propia.

Figura 26. Gráfico de la serie del PIB de la CAPV diferenciada lineal y estacionalmente.

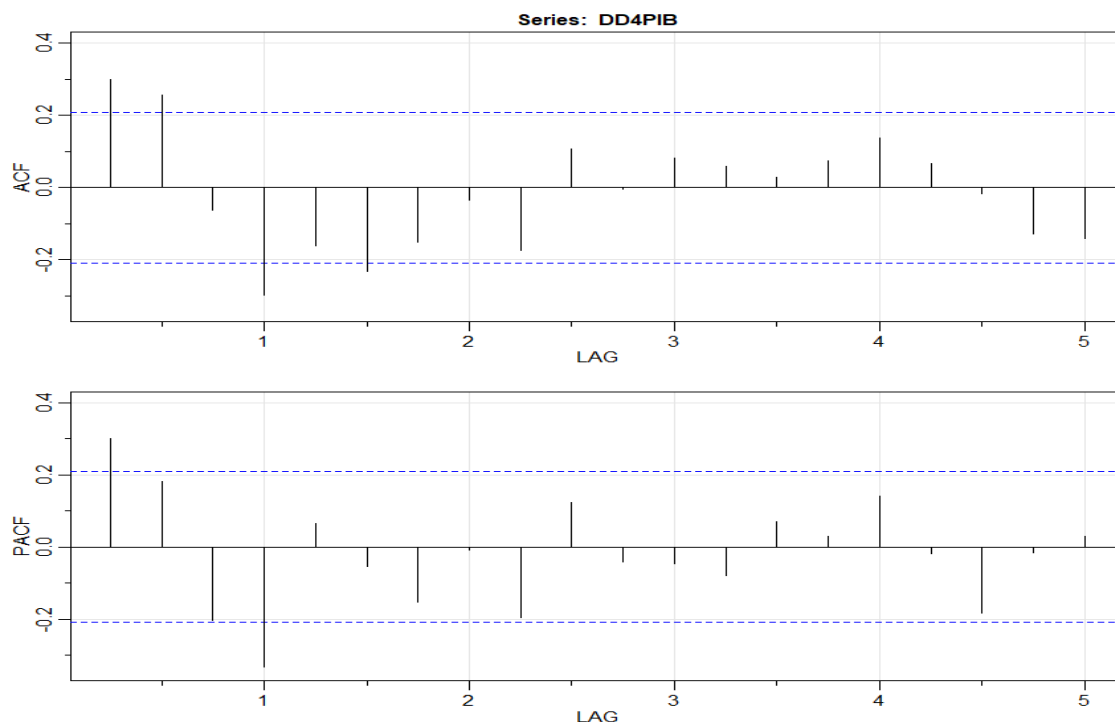
Observando la figura 26, tras las transformaciones la serie transformada parece cumplir las condiciones de estacionariedad salvo en un periodo, el de la crisis del 2008. Llegados a este punto, en estos casos se puede escoger entre tres alternativas en función a la magnitud de la perturbación:

- Tomar los últimos datos no perturbados para aplicar ARIMA. En este caso la opción queda descartada porque quedarían menos de 30 datos en la serie, lo que no es adecuado para estos modelos.
- Eliminar los datos de la perturbación. Si los datos tienen una incidencia notable a la hora de aplicar modelos, lo mejor será ignorarlos y eliminarlos de la serie.
- Proseguir y dejarlo tal cual está. Si la perturbación no es muy prolongada en el tiempo, no va a incidir demasiado en la estimación de parámetros ni en los autocorrelogramas, ya que se diluye la perturbación entre la magnitud del resto de datos. De esta forma se conserva el carácter original de la serie.

En este caso se va a optar por proseguir con el análisis, sin eliminar ningún dato, lo que no acabará teniendo repercusión en la aplicación de los modelos, como se comprobará más adelante.

3. Interpretación de los correlogramas

El siguiente paso lógico a dar es un análisis de los autocorrelogramas de la serie. Se utilizará, como ya se ha explicado anteriormente, la función $acf2()$ del paquete *astsa*. Los autocorrelogramas de la serie diferenciada son los siguientes:



Fuente: Elaboración propia.

Figura 27. Autocorrelogramas normal y parcial de la serie del PIB de la CAPV diferenciada lineal y estacionalmente.

Como se ha explicado en la parte teórica, a la hora de interpretar los autocorrelogramas, se debe interpretar primero la autocorrelación con todos los datos, para estimar de esta forma los parámetros p y q . En cuanto a los parámetros P y Q , hay que ignorar los datos de aquellos lag no estacionales. Es decir, en este caso solo se tienen en cuenta los múltiplos de 4.

Primero se comienza por el análisis no estacional. Hay que recordar que para la serie original PIB, el parámetro d ya está definido, ya que solo se ha diferenciado una vez para llegar a las condiciones de estacionariedad. Hay que recordar la tabla para estimar los parámetros p y q con los autocorrelogramas una vez se consiguen las condiciones de estacionariedad, en la figura 28.

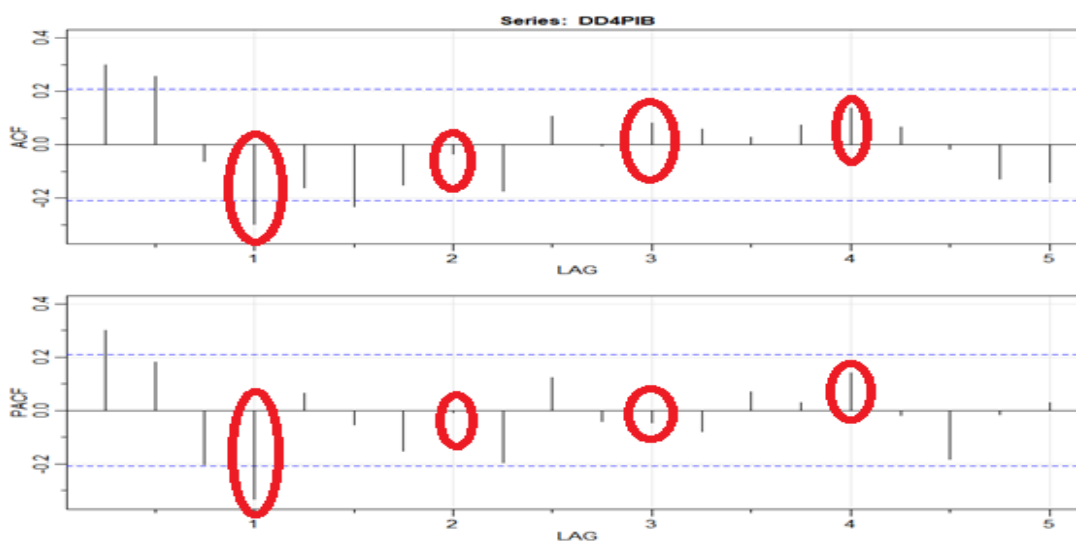
	AR(p)	MA(q)	ARMA(p, q)
Correlaciones	Decaen de manera progresiva	Pierden relevancia abruptamente en el <i>lag</i> igual al orden	Decaen de manera progresiva
Correlaciones parciales	Pierden relevancia abruptamente en el <i>lag</i> igual al orden	Decaen de manera progresiva	Decaen de manera progresiva

Fuente: Elaboración propia.

Figura 28. Tabla descriptiva de los comportamientos de los correlogramas en función al tipo de serie temporal.

Poniendo ya la vista con intención analítica en el correlograma de la serie (figura 27), se ve que los primeros *lag* tanto de la autocorrelación normal como de la autocorrelación parcial tienen relevancia. Sin embargo, tampoco parece que ninguno de ellos decaiga de manera progresiva. Por tener un *lag* más alejado fuera de las bandas de significación, se interpreta inicialmente que el autocorrelograma parcial es el que decae progresivamente, por lo que se interpreta un modelo ARIMA(0, 1, 2), por tener dos *lag* de relevancia en el autocorrelograma normal y después caer abruptamente.

En cuanto a la parte estacional, hay que tener en cuenta, como ya se ha comentado, solo los *lag* múltiplos de 4, como viene señalado en la figura 29.



Fuente: Elaboración propia.

Figura 29. Autocorrelogramas normal y parcial de la serie del PIB de la CAPV diferenciada lineal y estacionalmente destacando los lag estacionales.

Se puede ver que en este caso solo se tiene relevancia en el primer lag de cada autocorrelograma. La única interpretación posible es que los dos decaen progresivamente, por lo que la interpretación inicial es un SARIMA (1, 1, 1), quedando el modelo inicial completo estimado en base a los correlogramas como ARIMA(0, 1, 2)xSARIMA(1, 1, 1).

4. Elección del mejor modelo por medio de BIC y AIC

El siguiente paso a dar es probar los modelos diferentes aumentando o reduciendo el valor de los parámetros p , d , q , P , D y Q , y, observando que ocurre con los criterios AIC y BIC. Se aumentarán y reducirán las cifras de los parámetros, y, cuando se consigan menores magnitudes de los criterios se le dará este nuevo valor a éstos. Se lleva a cabo el proceso como en el siguiente ejemplo:

ARIMA(0, 1, 2) contra ARIMA (1, 1, 2):

- ARIMA(0,1,2):

```
> sarima(DD4PIB,0,0,2,1,0,1,4)$AIC [1] 27.05735
```

```
> sarima(DD4PIB,0,0,2,1,0,1,4)$BIC [1] 27.22181
```

- ARIMA(1,1,2):


```
> sarima(DD4PIB,1,0,2,1,0,1,4)$AIC [1] 27.07895
```

```
> sarima(DD4PIB,1,0,2,1,0,1,4)$BIC [1] 27.27083
```

En este caso ambos criterios dan un mejor resultado para el modelo estimado en base a los correlogramas, del cual se partía. De este modo, se escogerá el valor original, 0, para el parámetro p .

Se lleva a cabo a continuación otro ejemplo, esta vez con la estimación del parámetro q :

ARIMA(0,1,2) contra ARIMA(0,1,1) y contra ARIMA(0,1,3):

- ARIMA(0,1,2):

```
> sarima(DD4PIB,0,0,2,1,0,1,4)$AIC [1] 27.05735
```

```
> sarima(DD4PIB,0,0,2,1,0,1,4)$BIC [1] 27.22181
```

- ARIMA(0,1,3):

```
> sarima(DD4PIB,0,0,3,1,0,1,4)$AIC [1] 27.07902
```

```
> sarima(DD4PIB,0,0,3,1,0,1,4)$BIC [1] 27.27089
```

- ARIMA(0,1,1):

```
> sarima(DD4PIB,0,0,1,1,0,1,4)$AIC [1] 27.16127
```

```
> sarima(DD4PIB,0,0,1,1,0,1,4)$BIC [1] 27.29832
```

Una vez más se ve que el mejor parámetro d para el modelo es el que se había estimado inicialmente desde los correlogramas, cosa que no es necesariamente habitual. En caso de que no hubiese sido así se hubiera tenido que repetir el proceso con el nuevo valor del parámetro, viendo si sumando o restando un valor los criterios seguirían decreciendo.

Cabe destacar también, que en el código de R escrito, los parámetros d y D tienen valor 0 porque se están aplicando los modelos sobre la serie ya diferenciada.

Siguiendo con este procedimiento, el modelo que menor valor ha obtenido en los criterios, y en consecuencia el mejor, es un ARIMA(0, 1, 2)xSARIMA(1, 1, 0), siendo el único cambio sobre la estimación en base a los correlogramas el del parámetro Q , que ha pasado de un valor de 1 a 0. De este modo, el valor final de los criterios queda de la siguiente forma:

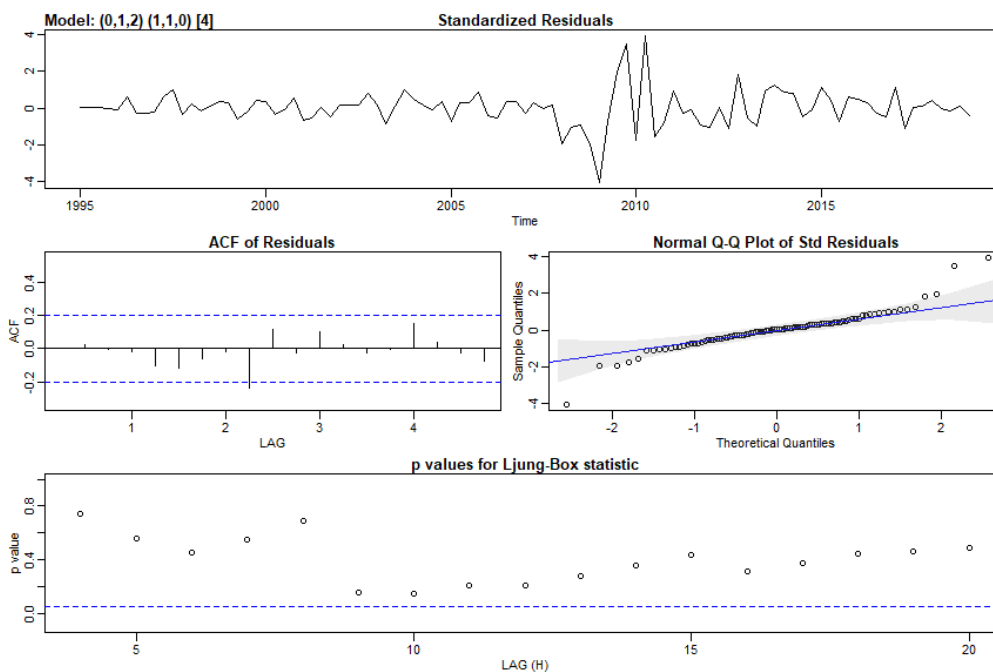
\$AIC
[1] 26.16182

\$BIC
[1] 26.268

Una ayuda que será recurrida para buscar los modelos óptimos en función a los criterios *AIC* y *BIC* será la función *auto.arima()* del paquete *forecasting*, que busca el mejor modelo en función a estos parámetros, pero no siempre consigue el óptimo. Por ello, esta función servirá únicamente de orientación inicial y para corroborar lo analizado en los correlogramas, y nunca como herramienta definitiva.

5. Análisis de los residuos para el modelo escogido

Una vez escogido el modelo habrá que hacer una última comprobación para verificar que éste es válido. Se trata del análisis de residuos. La función *sarima()* del paquete *astsa* provee al usuario, además de entre otras muchas cosas, de 4 gráficos para comprobar si los residuos se comportan como ruido blanco. La razón de ser de estos gráficos ya ha sido explicada en el apartado 5 del trabajo, así que se va a proceder a analizarlos con la serie del PIB de la CAPV:



Fuente: Elaboración propia.

Figura 30. Gráficos de los residuos de la serie del PIB de la CAPV para un modelo ARIMA(0, 1, 2)xSARIMA(1, 1, 1)

Como se aprecia en la figura 30, los gráficos son bastante adecuados si se tiene en cuenta lo explicado en la teoría. Solo hay que destacar, en el gráfico superior, el de los residuos puros, que hay una parte donde estos no se comportan como ruido blanco. Esto es así porque inicialmente no se ha eliminado la anomalía de la crisis de 2008. Salvo esto, se puede afirmar que los residuos son bastante adecuados para considerarlos como ruido blanco, por lo que se sigue adelante con el modelo.

En caso de no haber sido así, habría que haber escogido el siguiente modelo con mejor resultado en los criterios *AIC* y *BIC*, y haber continuado hasta que en alguno los residuos se comportasen adecuadamente.

6. Predicción con el modelo seleccionado y análisis de resultado

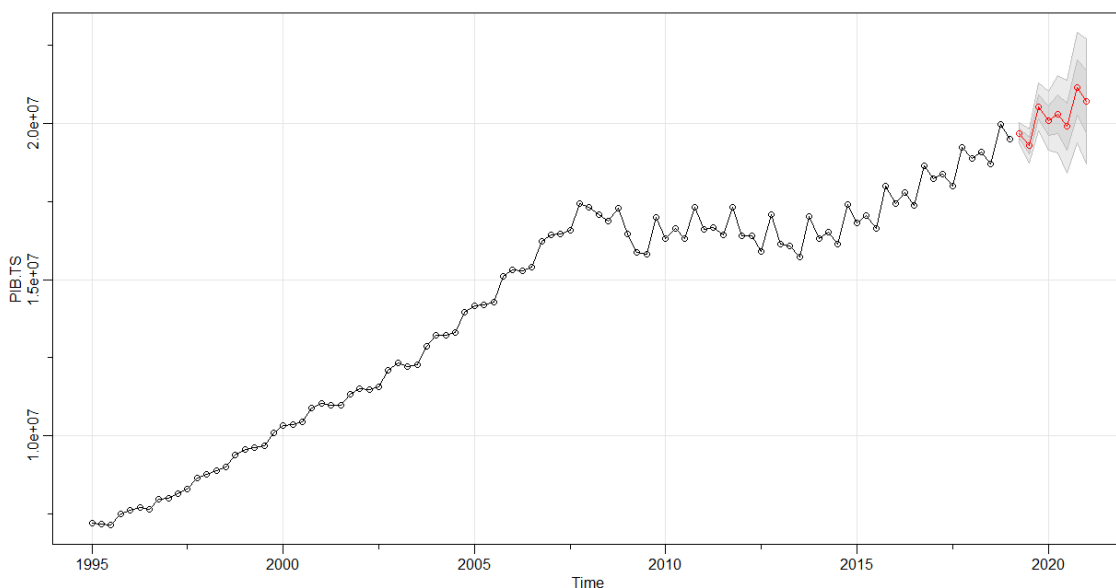
Una vez seleccionado el modelo y habiendo comprobado que es adecuado, la predicción es la parte más sencilla. Lo único que hay que hacer es sustituir en las fórmulas de los modelos los valores de los parámetros Φ_n y θ_n , tal y como están escritos en las fórmulas generales de los modelos. Por suerte, en el paquete *astsa* hay una función, *sarima.for()*, que hace todo esto por nosotros, y además nos ofrece un gráfico con las predicciones y una representación de su error estándar.

Habrá que estipular el número de predicciones deseadas, en este caso 8, que se corresponden a dos años en esta serie:

```
> sarima.for(PIB.TS,n.ahead = 8,0,1,2,1,1,0,4)
$pred
      Qtr1      Qtr2      Qtr3      Qtr4
2019      19678897 19279462 20527402
2020 20087105 20281897 19888877 21137508
2021 20691050

$se
      Qtr1      Qtr2      Qtr3      Qtr4
2019      169641.4 271066.6 382820.7
2020 468643.1 613090.1 746599.7 881929.4
2021 999093.4
```

De este modo, el gráfico con la predicción correspondiente se representa en la figura 31.



Fuente: Elaboración propia.

Figura 31. Gráfico de la serie del PIB de la CAPV con predicción hasta 2021-1T.

En él, la parte roja representa la predicción obtenida, mientras que la parte sombreada son los límites del error, siendo la más oscura la que comprende el 90% de los casos y la más clara la que comprende un 95%.

Se puede concluir la predicción apreciando que se ha conseguido mantener la misma tendencia y los mismos patrones estacionales, pudiendo afirmar que está bien ejecutada tan solo mirando el gráfico. Es decir, según estos modelos se prevé que el crecimiento del PIB siga, continuando la tendencia de los últimos años, en casi la totalidad de las posibilidades.

La metodología descrita para este ejemplo del PIB de la CAPV será la que se seguirá a lo largo de todas las siguientes predicciones, no volviendo a escribirse otra vez lo mismo en cada una de ellas, ya que no se considera necesario y se cree que no aportaría mayor valor al trabajo.

6.2. EJECUCIÓN DE PREDICCIONES S&P500 E IBEX 35

Como ya se ha comentado en la parte teórica, se van a realizar tres predicciones sobre los índices en distintas resoluciones temporales. Se va a procurar que los periodos en los que se realizan sean estables y no fuera de lo común, para que los resultados no correspondan a etapas económicas anómalas en los que

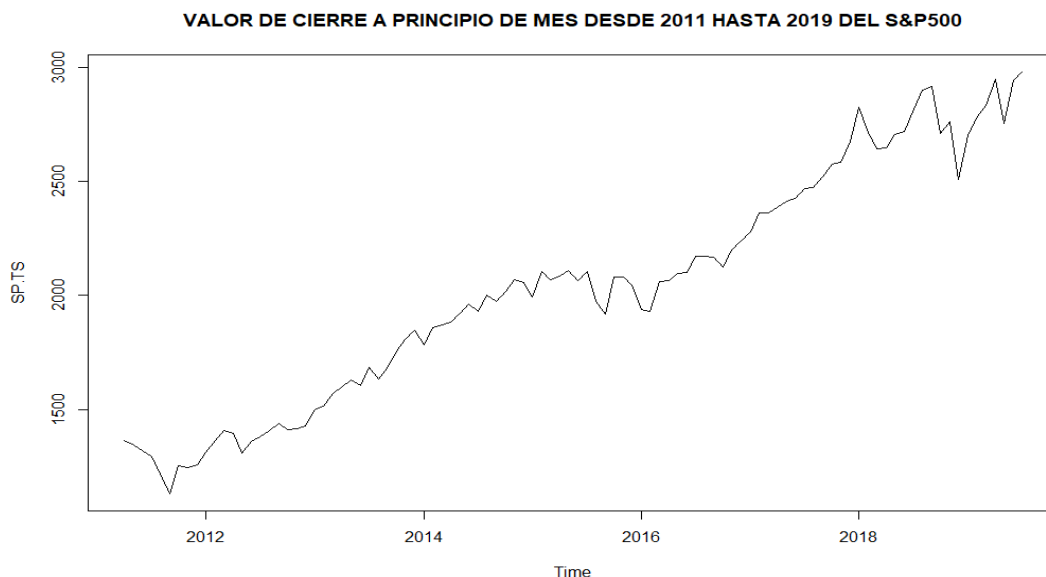
podrían verse alterados. Siendo esto así, se van a realizar las predicciones sobre 90 datos, siempre en la etapa comprendida entre 2010 y 2019, y se predecirán 10 nuevos datos que serán comparados con los valores reales. No se llegará a predecir en ningún momento valores posteriores a los de la fecha de realización de este trabajo, ya que lo que interesa es analizar la calidad de la predicción obtenida. En los datos tampoco se consideran los índices con reinversión de dividendos (*Total Return*).

6.2.1. S&P 500

S&P 500 mensual

Para el caso mensual se toman los datos desde marzo de 2011 hasta el primer día de mes actual. El gráfico y los datos del S&P 500 se representan en la figura 32:

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2011				1363.61	1345.20	1320.64	1292.28	1218.89	1131.42	1253.30	1246.96	1257.60
2012	1312.41	1365.68	1408.47	1397.91	1310.33	1362.16	1379.32	1406.58	1440.67	1412.16	1416.18	1426.19
2013	1498.11	1514.68	1569.19	1597.57	1630.74	1606.28	1685.73	1632.97	1681.55	1756.54	1805.81	1848.36
2014	1782.59	1859.45	1872.34	1883.95	1923.57	1960.23	1930.67	2003.37	1972.29	2018.05	2067.56	2058.90
2015	1994.99	2104.50	2067.89	2085.51	2107.39	2063.11	2103.84	1972.18	1920.03	2079.36	2080.41	2043.94
2016	1940.24	1932.23	2059.74	2065.30	2096.95	2098.86	2173.60	2170.95	2168.27	2126.15	2198.81	2238.83
2017	2278.87	2363.64	2362.72	2384.20	2411.80	2423.41	2470.30	2471.65	2519.36	2575.26	2584.84	2673.61
2018	2823.81	2713.83	2640.87	2648.05	2705.27	2718.37	2816.29	2901.52	2913.98	2711.74	2760.17	2506.85
2019	2704.10	2784.49	2834.40	2945.83	2752.06	2941.76	2979.63					

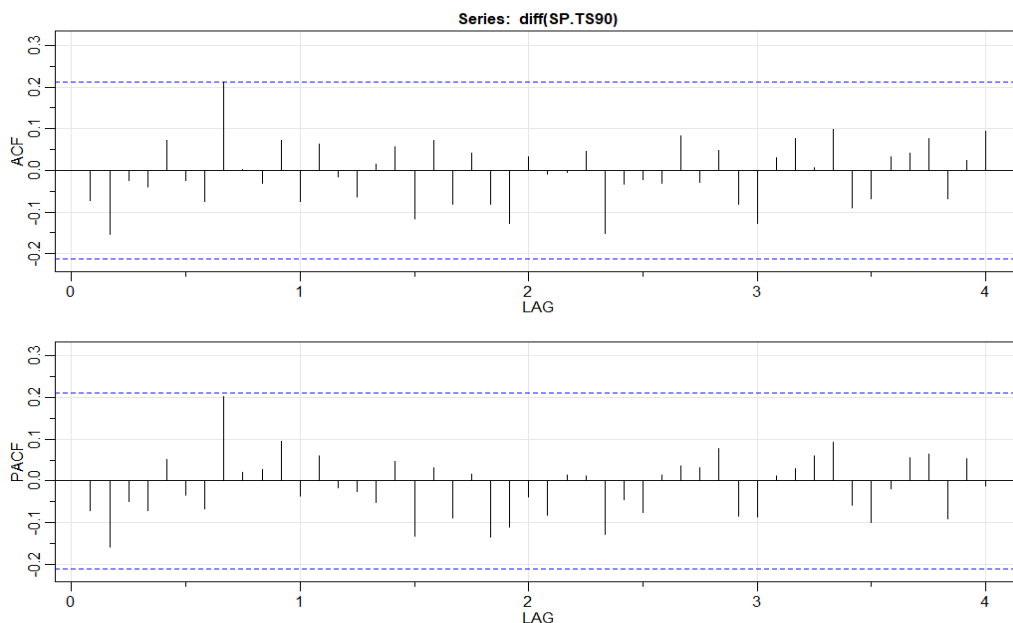


Fuente: Elaboración propia.

Figura 32. Gráfico de los valores de cierre del primer día operativo de cada mes del S&P500 desde abril de 2011 hasta julio de 2019.

Para la predicción se van a eliminar los 10 últimos datos y sobre los 90 primeros se escogerá el modelo ARIMA más adecuado. Como se puede apreciar en la

figura 32 se va a tratar de predecir una zona con mucha perturbación, por lo que es esperable desde un primer momento que no sea demasiado precisa.

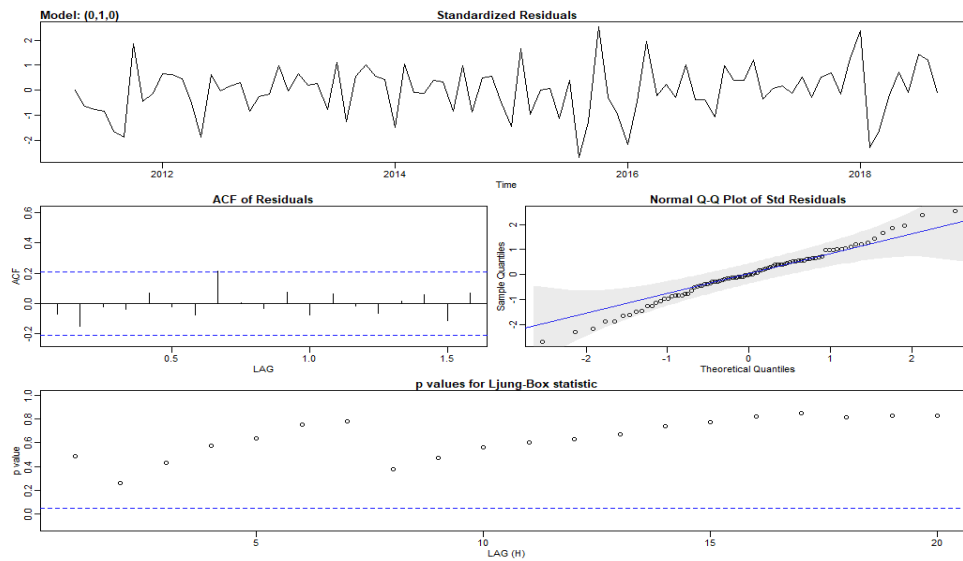


Fuente: Elaboración propia.

Figura 33. Autocorrelogramas normal y parcial de la serie mensual de los valores del S&P500 diferenciada.

El autocorrelograma de la serie diferenciada (figura 33) resulta ser el típico del ruido blanco, sin autocorrelaciones entre sí. En este caso no hay duda, se estima de un paseo aleatorio. Los parámetros correspondientes al paseo aleatorio son $(p, d, q) = (0, 1, 0)$. La aplicación de los criterios *AIC* y *BIC* lo confirma.

El análisis de residuos de la figura 34 confirma que la serie diferenciada tiene el comportamiento de ruido blanco:



Fuente: Elaboración propia.

Figura 34. Gráficos de los residuos de la serie mensual del S&P500 para un modelo ARIMA(0, 1, 0).

El valor de la media, de la desviación estándar, y de los criterios BIC y AIC vienen dados por la función *sarima()*:

```

Coefficients:
  constant
  17.4199
s. e.      5.9312

sigma^2 estimated as 3131:  log likelihood = -484.47,  aic = 972.94

$degrees_of_freedom
[1] 88

$ttable
      Estimate      SE t.value p.value
constant 17.4199  5.9312   2.937  0.0042

$AIC
[1] 10.93192

$AICC
[1] 10.93244

$BIC
[1] 10.98785
    
```

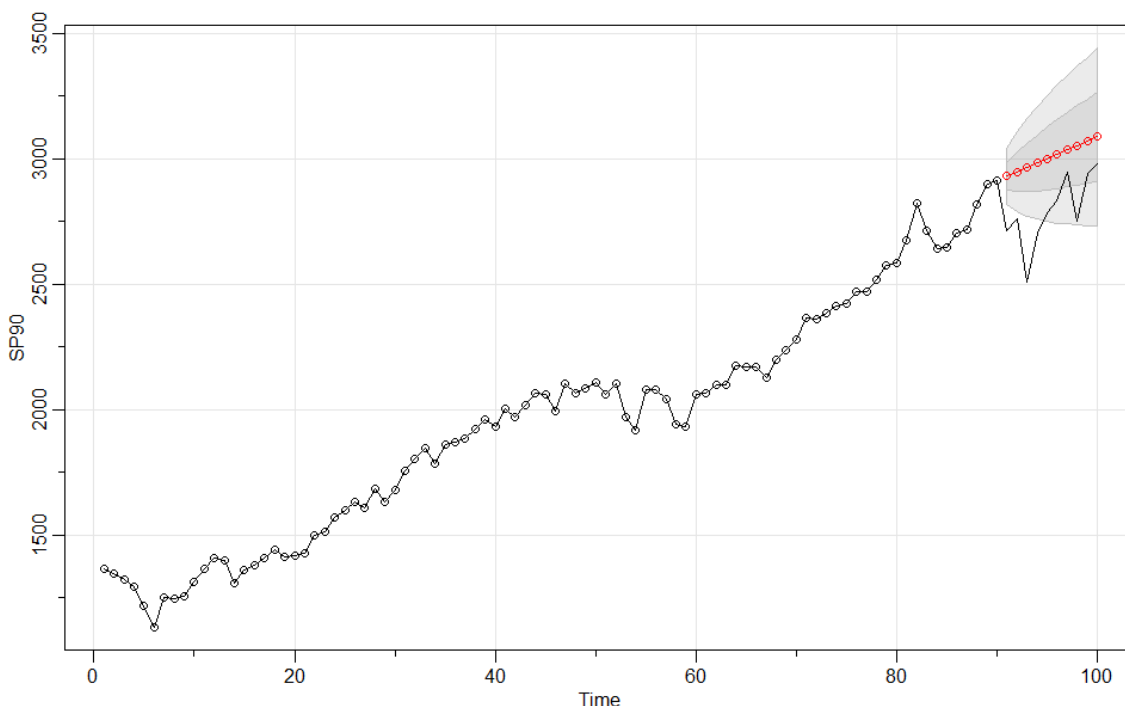
De este modo, se obtiene la siguiente predicción para los siguientes 10 valores:

```

$pred
Time Series:
Start = 91
End = 100
Frequency = 1
[1] 2931.400 2948.820 2966.240 2983.660 3001.079 3018.499 3035.919 3053.339 3070.759 3088.179

$se
Time Series:
Start = 91
End = 100
Frequency = 1
[1] 55.95523 79.13265 96.91730 111.91046 125.11970 137.06176 148.04363 158.26529 167.86569 176.94598
    
```

Que como se puede ver en la figura 35, no se trata de una predicción muy acertada.



Fuente: Elaboración propia.

Figura 35. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie mensual del S&P500.

S&P 500 semanal

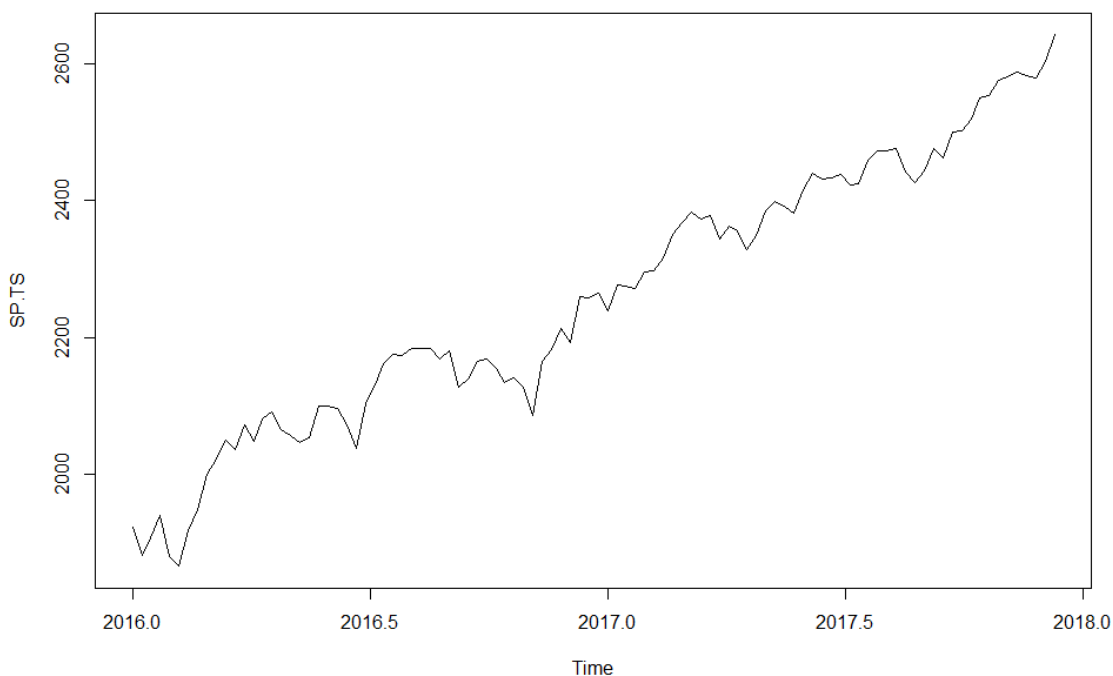
En el caso de la serie semanal se han tomado datos desde la primera semana de 2016, ya que se ha visto en el gráfico que es la etapa con menor magnitud de perturbaciones aleatorias que podría deberse a los ruidos blancos y también es interesante probar los modelos ARIMA en momentos muy estables.

Los datos de la serie temporal están representados en la figura 36.


```

start = c(2016, 1)
End = c(2017, 49)
Frequency = 51
...1100 ...199 ...198 ...197 ...196 ...195 ...194 ...193 ...192 ...191 ...190 ...189 ...188
1922.03 1880.33 1906.90 1940.24 1880.05 1864.78 1917.78 1948.05 1999.99 2022.19 2049.58 2035.94 2072.78
...187 ...186 ...185 ...184 ...183 ...182 ...181 ...180 ...179 ...178 ...177 ...176 ...175
2047.60 2080.73 2091.58 2065.30 2057.14 2046.61 2052.32 2099.06 2099.13 2096.07 2071.22 2037.41 2102.95
...174 ...173 ...172 ...171 ...170 ...169 ...168 ...167 ...166 ...165 ...164 ...163 ...162
2129.90 2161.74 2175.03 2173.60 2182.87 2184.05 2183.87 2169.04 2179.98 2127.81 2139.16 2164.69 2168.27
...161 ...160 ...159 ...158 ...157 ...156 ...155 ...154 ...153 ...152 ...151 ...150 ...149
2153.74 2132.98 2141.16 2126.41 2085.18 2164.45 2181.90 2213.35 2191.95 2259.53 2258.07 2263.79 2238.83
...148 ...147 ...146 ...145 ...144 ...143 ...142 ...141 ...140 ...139 ...138 ...137 ...136
2276.98 2274.64 2271.31 2294.69 2297.42 2316.10 2351.16 2367.34 2383.12 2372.60 2378.25 2343.98 2362.72
...135 ...134 ...133 ...132 ...131 ...130 ...129 ...128 ...127 ...126 ...125 ...124 ...123
2355.54 2328.95 2348.69 2384.20 2399.29 2390.90 2381.73 2415.82 2439.07 2431.77 2433.15 2438.30 2423.41
...122 ...121 ...120 ...119 ...118 ...117 ...116 ...115 ...114 ...113 ...112 ...111 ...110
2425.18 2459.27 2472.54 2472.10 2476.83 2441.32 2425.55 2443.05 2476.55 2461.43 2500.23 2502.22 2519.36
...19 ...18 ...17 ...16 ...15 ...14 ...13 ...12 ...11
2549.33 2553.17 2575.21 2581.07 2587.84 2582.30 2578.85 2602.42 2642.22
    
```

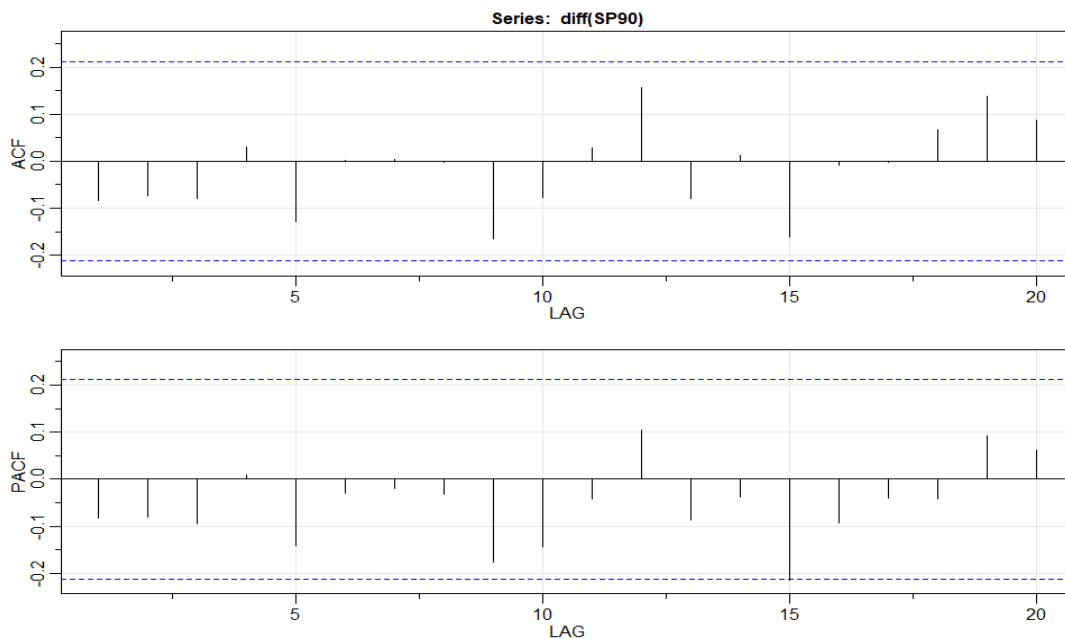
VALOR DE CIERRE DE PRINCIPIO DE SEMANA DESDE 2016 HASTA 2018 DEL S&P500



Fuente: Elaboración propia.

Figura 36. Gráfico de los valores de cierre de cada lunes del S&P500 desde el 4 de enero de 2016 hasta el 18 de diciembre de 2017.

Se toman los primeros 90 datos, se diferencian, y se examina el correlograma de la figura 37.

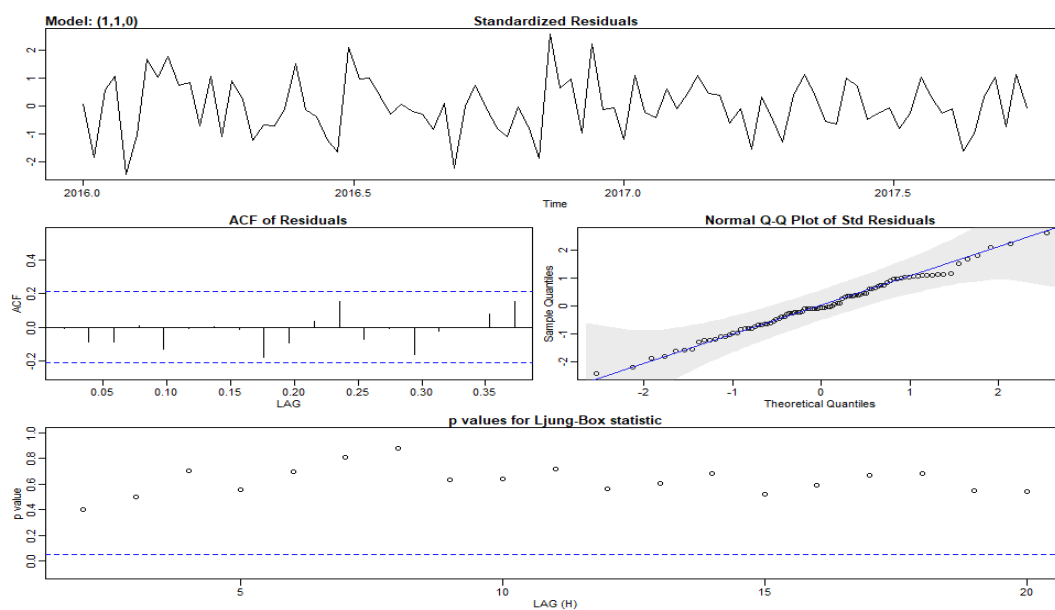


Fuente: Elaboración propia.

Figura 37. Autocorrelogramas normal y parcial de la serie semanal de los valores del S&P500 diferenciada.

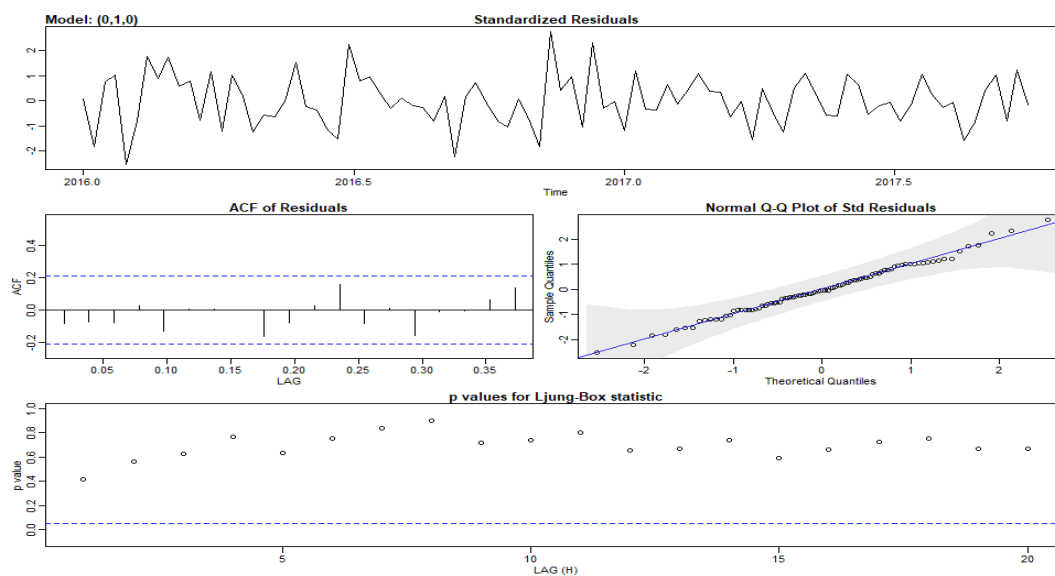
De nuevo el autocorrelograma indica un paseo aleatorio. Parece que el carácter de la serie persiste a menor plazo.

Esta vez, los criterios *AIC* y *BIC* indican que el mejor modelo es un $ARIMA(1, 1, 0)$. Sin embargo, al estimar el parámetro Φ del modelo se obtiene un número muy cercano a 0 (-0.082), lo que quiere decir que la parte autorregresiva de éste, a pesar de mejorar el tamaño de los residuos, es prácticamente insignificante. Siendo esto así, se procede a comparar los residuos de ambas en las figuras 38 y 39.



Fuente: Elaboración propia.

Figura 38. Gráficos de los residuos de la serie semanal del S&P500 para un modelo ARIMA(1, 1, 0).



Fuente: Elaboración propia.

Figura 39. Gráficos de los residuos de la serie semanal del S&P500 para un modelo ARIMA(0, 1, 0).

En el caso del paseo aleatorio, los residuos parecen comportarse de forma más blanca. En primer lugar, el gráfico de los residuos estandarizados de la figura 39 parece algo más hocedástico que el de la figura 38, y además, el gráfico normal Q-Q tiene bastante más alineados los puntos con su línea recta, lo que indica que el

comportamiento de los residuos es mejor para el paseo aleatorio, por lo que se vuelve a considerar como modelo para adaptar la serie.

Se obtienen los siguientes valores para la predicción:

```
Coefficients:
      constant
      6.5190
s.e.      2.8024

sigma^2 estimated as 699:  log likelihood = -417.74,  aic = 839.49

$degrees_of_freedom
[1] 88

$ttable
      Estimate      SE t.value p.value
constant    6.519 2.8024  2.3262  0.0223

$AIC
[1] 9.432424

$AICc
[1] 9.432941

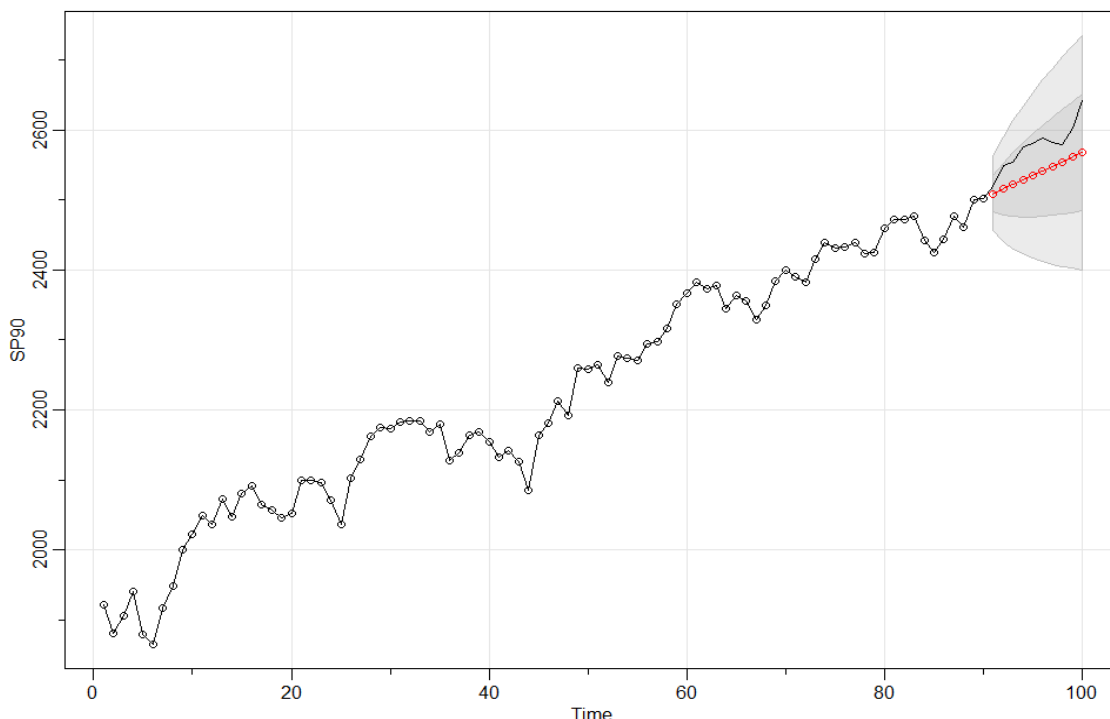
$BIC
[1] 9.488349
```

Por lo que la predicción para los siguientes 10 calculada por medio de *sarima.for()* es:

```
$pred
Time Series:
Start = c(2017, 40)
End = c(2017, 49)
Frequency = 51
[1] 2508.739 2515.258 2521.777 2528.296 2534.815 2541.334 2547.853 2554.372 2560.891 2567.410

$se
Time Series:
Start = c(2017, 40)
End = c(2017, 49)
Frequency = 51
[1] 26.43798 37.38895 45.79193 52.87597 59.11713 64.75957 69.94833 74.77791 79.31395 83.60424
```

Se compara con los datos reales en la figura 40.



Fuente: Elaboración propia.

Figura 40. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie semanal del S&P500.

En este caso la predicción del paseo aleatorio parece, por lo menos, no estar fuera de los límites del 90%, por lo que se considera bastante aceptable. Se concluye que se debe a la estabilidad de la etapa predicha.

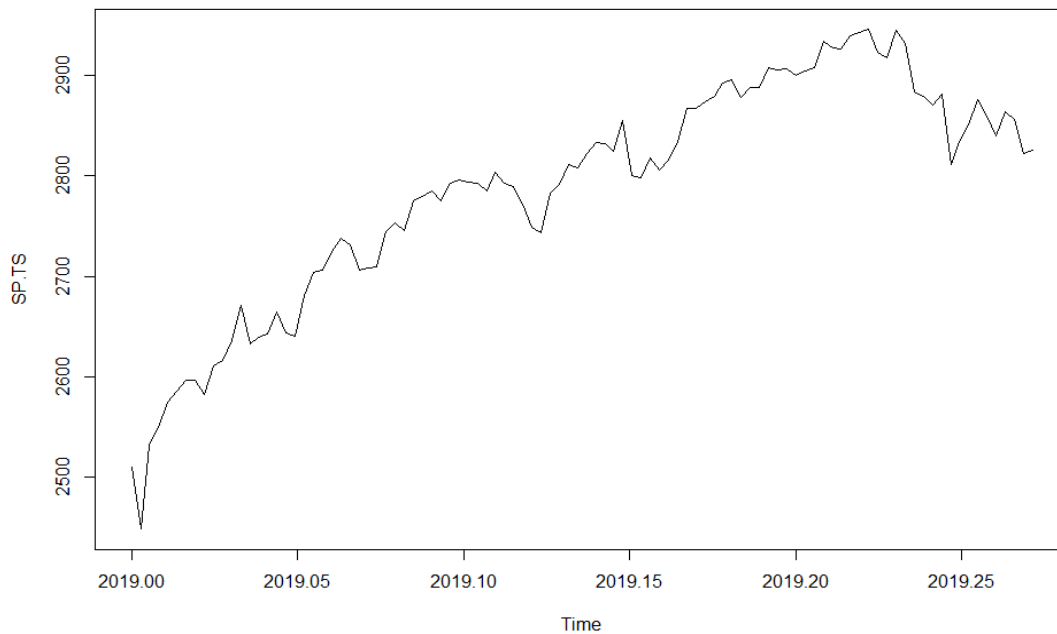
S&P 500 diario

Para el análisis diario de S&P500 se toman datos desde el 2 de enero de 2019 hasta completar 100. Se representan en la figura 41.

```

start = c(2019, 1)
End = c(2019, 100)
Frequency = 365
...1100 ...199 ...198 ...197 ...196 ...195 ...194 ...193 ...192 ...191 ...190 ...189 ...188
2510.03 2447.89 2531.94 2549.69 2574.41 2584.96 2596.64 2596.26 2582.61 2610.30 2616.10 2635.96 2670.71
...187 ...186 ...185 ...184 ...183 ...182 ...181 ...180 ...179 ...178 ...177 ...176 ...175
2632.90 2638.70 2642.33 2664.76 2643.85 2640.00 2681.05 2704.10 2706.53 2724.87 2737.70 2731.61 2706.05
...174 ...173 ...172 ...171 ...170 ...169 ...168 ...167 ...166 ...165 ...164 ...163 ...162
2707.88 2709.80 2744.73 2753.03 2745.73 2775.60 2779.76 2784.70 2774.88 2792.67 2796.11 2793.90 2792.38
...161 ...160 ...159 ...158 ...157 ...156 ...155 ...154 ...153 ...152 ...151 ...150 ...149
2784.49 2803.69 2792.81 2789.65 2771.45 2748.93 2743.07 2783.30 2791.52 2810.92 2808.48 2822.48 2832.94
...148 ...147 ...146 ...145 ...144 ...143 ...142 ...141 ...140 ...139 ...138 ...137 ...136
2832.57 2824.23 2854.88 2800.71 2798.36 2818.46 2805.37 2815.44 2834.40 2867.19 2867.24 2873.40 2879.39
...135 ...134 ...133 ...132 ...131 ...130 ...129 ...128 ...127 ...126 ...125 ...124 ...123
2892.74 2895.77 2878.20 2888.21 2888.32 2907.41 2905.58 2907.06 2900.45 2905.03 2907.97 2933.68 2927.25
...122 ...121 ...120 ...119 ...118 ...117 ...116 ...115 ...114 ...113 ...112 ...111 ...110
2926.17 2939.88 2943.03 2945.83 2923.73 2917.52 2945.64 2932.47 2884.05 2879.42 2870.72 2881.40 2811.87
...19 ...18 ...17 ...16 ...15 ...14 ...13 ...12 ...11
2834.41 2850.96 2876.32 2859.53 2840.23 2864.36 2856.27 2822.24 2826.06
    
```

VALOR DE CIERRE DIARIO DESDE EL 1 DE ENERO HASTA FINALES DE MAYO DE 2019 DEL S&P500

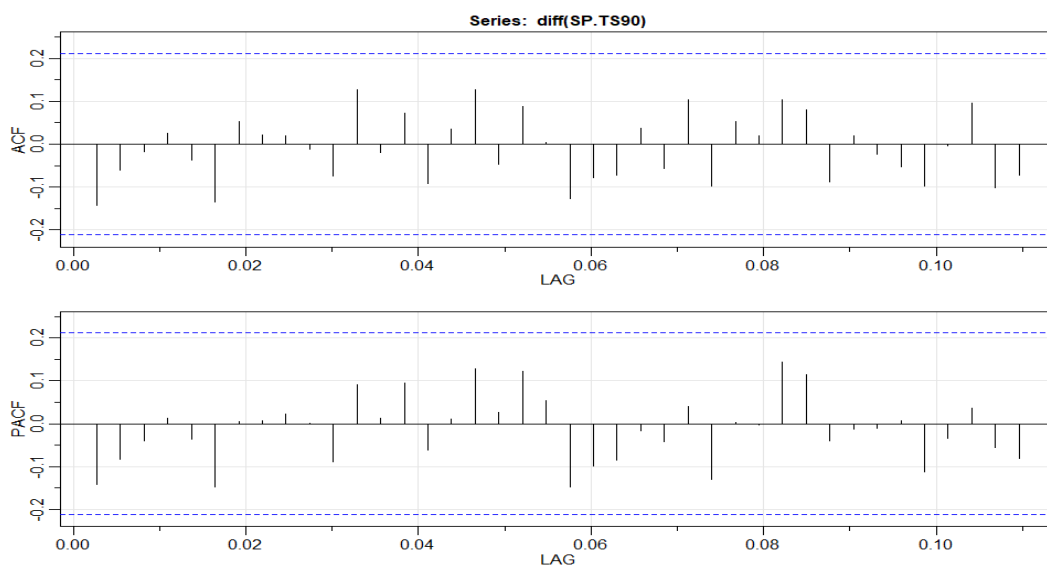


Fuente: Elaboración propia.

Figura 41. Gráfico de los valores de cierre diario del S&P500 desde el 2 de enero de 2019 hasta el 24 de mayo de del mismo año.

Como se ve en la figura 41 la zona a predecir va a ser justo después de un máximo anual, por lo que se podrá ver cómo se comportan los modelos ante estos cambios.

Los correlogramas de los 90 primeros datos diferenciados se examinan en la figura 42.

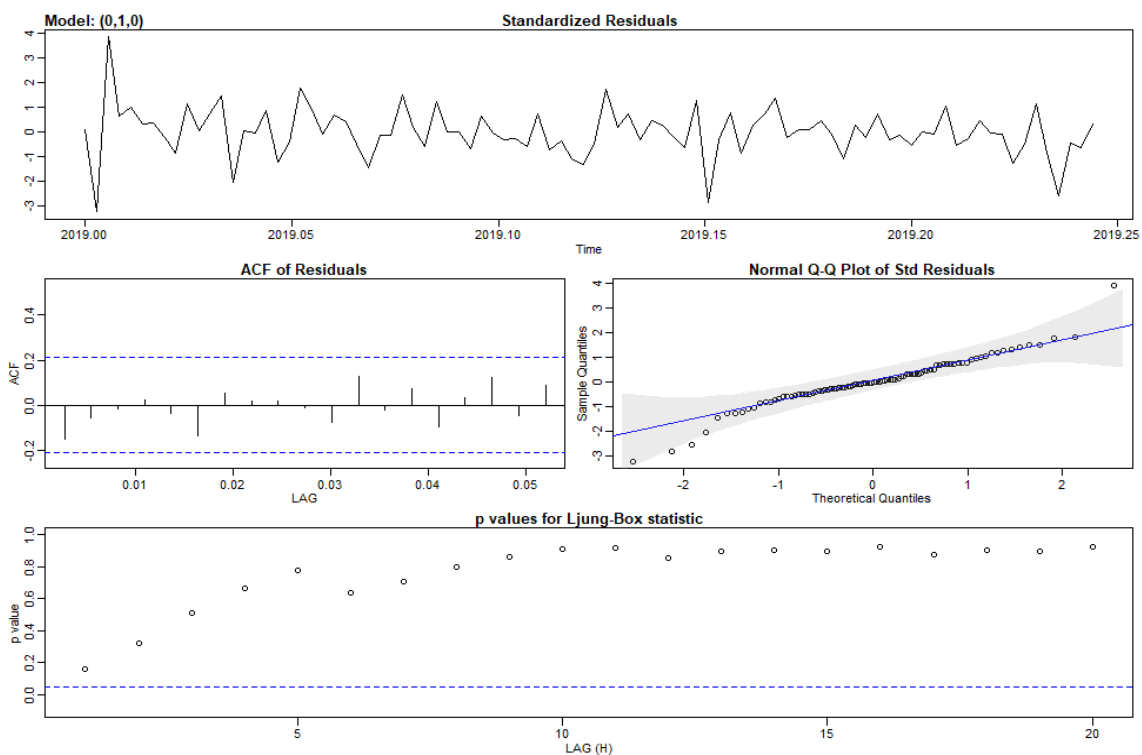


Fuente: Elaboración propia.

Figura 42. Autocorrelogramas normal y parcial de la serie diaria de los valores del S&P500 diferenciada.

De nuevo se vuelve a ver un gráfico típico de paseos aleatorios en la figura 42, con autorrrleaciones sin relevancia en la mayoría de los datos. Podría ser que en este caso si las bandas de significación no se considerasen constantes para distintos números de muestras la conclusión fuese diferente, sin embargo, la correlación seguiría siendo de muy bajo valor a lo largo de todo el gráfico.

Lo que provoca que se considere de nuevo por un paseo aleatorio, y los criterios *BIC* y *AIC* y el análisis de residuos así lo confirman, tal como muestra la figura 43.



Fuente: Elaboración propia.

Figura 43. Gráficos de los residuos de la serie diaria del S&P500 para un modelo ARIMA(0, 1, 0).

Los datos de la serie con el paseo aleatorio, es decir, la media y el error estándar, son los siguientes:

```
Coefficients:
      constant
      4.1727
s.e.      2.1797
```

```
sigma^2 estimated as 422.9: log likelihood = -395.38, aic = 794.76
```

```
$degrees_of_freedom
[1] 88
```

```
$table
      Estimate      SE t.value p.value
constant  4.1727  2.1797  1.9143  0.0588
```

```
$AIC
[1] 8.929874
```

```
$AICC
[1] 8.930391
```

```
$BIC
[1] 8.985798
```

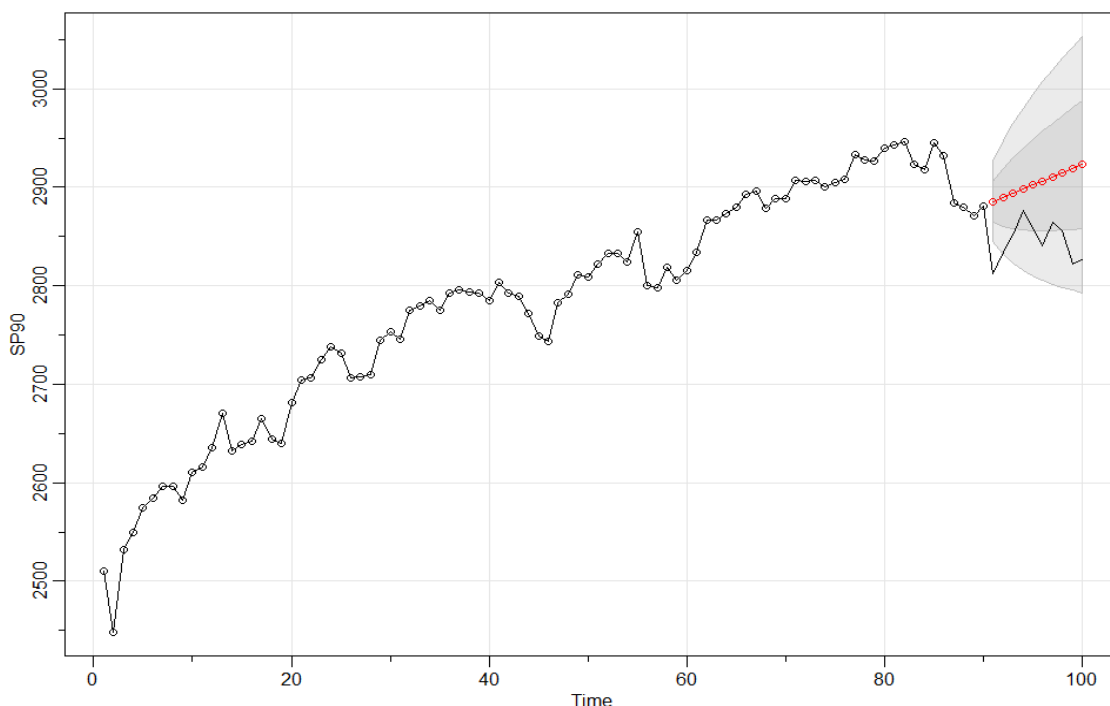

Y la predicción queda de la siguiente forma:

```

$pred
Time Series:
Start = c(2019, 91)
End = c(2019, 100)
Frequency = 365
[1] 2885.573 2889.745 2893.918 2898.091 2902.263 2906.436 2910.609 2914.782 2918.954 2923.127

$se
Time Series:
Start = c(2019, 91)
End = c(2019, 100)
Frequency = 365
[1] 20.56368 29.08144 35.61734 41.12737 45.98179 50.37053 54.40639 58.16288 61.69105 65.02808
    
```

Que, comparándolo con los datos reales en el gráfico enseña una notable diferencia entre los resultados y la realidad, como se ve en la figura 44.



Fuente: Elaboración propia.

Figura 44. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie diaria del S&P500.

La predicción, como se podía prever viendo con detalle los datos que iban a ser predichos, es bastante defectuosa. ARIMA no funciona bien prediciendo en momentos de cambio brusco o si la serie pierde la tendencia. En cuanto al modelo, parece que el S&P 500 confirma en todas sus resoluciones la hipótesis del paseo aleatorio.

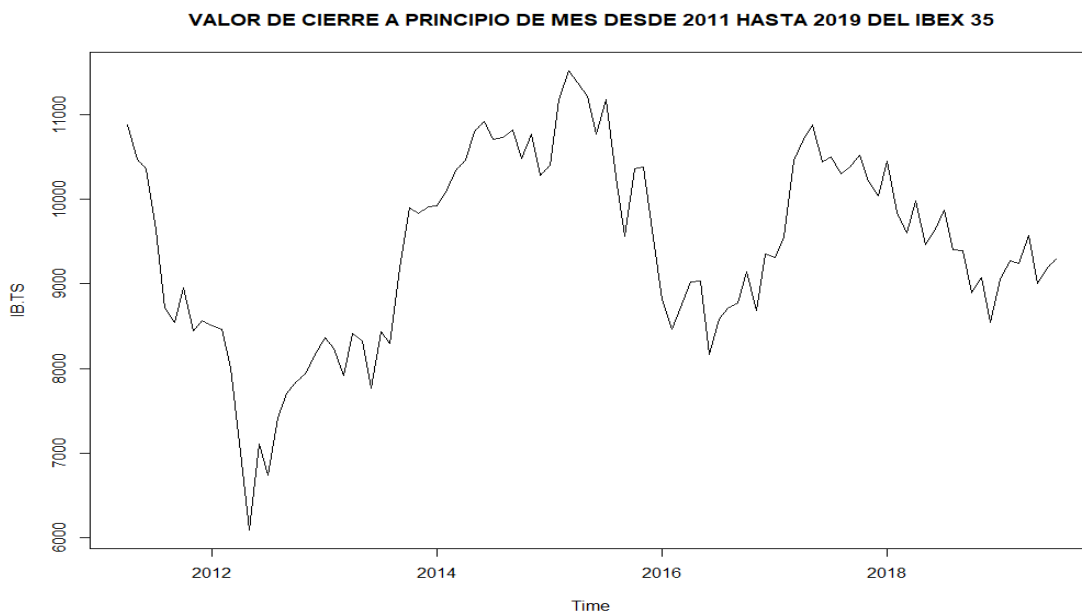
6.2.2. IBEX 35

En cuanto al IBEX 35, se van a tratar de repetir las mismas predicciones que con el S&P 500. Sin embargo, se espera, previo a la ejecución de las predicciones, que estas tengan más facilidad para adaptarse a modelos ARIMA distintos a paseos aleatorios, ya que al tratarse de un mercado menos globalizado (aunque lo suficiente como para seguir considerándose muy internacionalizado) que el del S&P 500, y además participando menos empresas en su cotización, sería natural que éste tenga más probabilidades de exhibir un comportamiento menos eficiente que el del índice estadounidense, aunque según la teoría del paseo aleatorio no debería notarse.

IBEX 35 mensual

Como se ha dicho, se van a tomar los mismos periodos que para el S&P 500, por lo que de nuevo se toman 100 datos desde abril de 2011 hasta julio de 2019, tal como representa la figura 45.

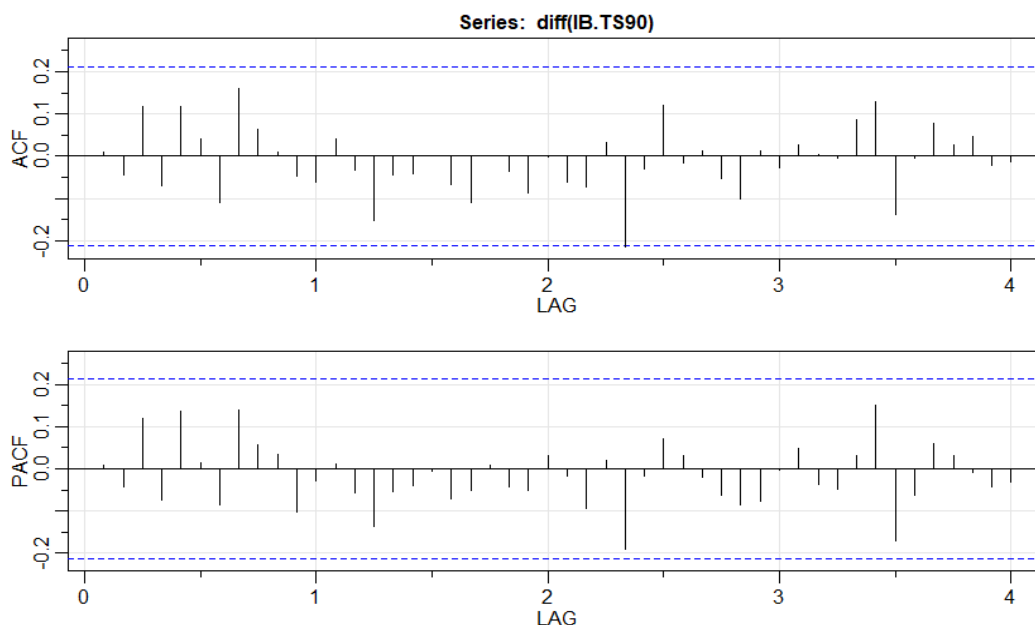
	Jan	Feb	Mar	Apr	May	Jun	Ju1	Aug	Sep	Oct	Nov	Dec
2011				10878.89	10475.99	10359.89	9630.69	8718.59	8546.59	8954.89	8449.49	8566.29
2012	8509.19	8465.89	8007.99	7010.99	6089.79	7102.19	6738.09	7420.49	7708.49	7842.89	7934.59	8167.49
2013	8362.29	8230.29	7919.99	8418.99	8320.59	7762.69	8433.39	8290.49	9186.09	9907.89	9837.59	9916.69
2014	9920.19	10114.19	10340.49	10458.99	10798.69	10923.49	10707.19	10728.79	10825.49	10477.79	10770.69	10279.49
2015	10403.29	11178.29	11521.09	11384.99	11217.59	10769.49	11180.69	10258.99	9559.89	10360.69	10386.89	9544.19
2016	8815.79	8461.39	8723.09	9025.69	9033.99	8163.29	8587.20	8716.80	8779.40	9143.30	8688.20	9352.10
2017	9315.20	9555.50	10462.90	10715.80	10880.00	10444.50	10502.20	10299.50	10381.50	10523.50	10211.00	10043.90
2018	10451.50	9840.30	9600.40	9980.60	9465.50	9622.70	9870.70	9399.10	9389.20	8893.50	9077.20	8539.90
2019	9056.70	9277.70	9240.30	9570.60	9004.20	9198.80	9293.20					



Fuente: Elaboración propia.

Figura 45. Gráfico de los valores de cierre del primer día operativo de cada mes del IBEX 35 desde abril de 2011 hasta julio de 2019.

Repitiendo lo hecho con las series del S&P500, se toman los 90 primeros datos y se hace una diferenciación y se examinan los autocorrelogramas en la figura 46.

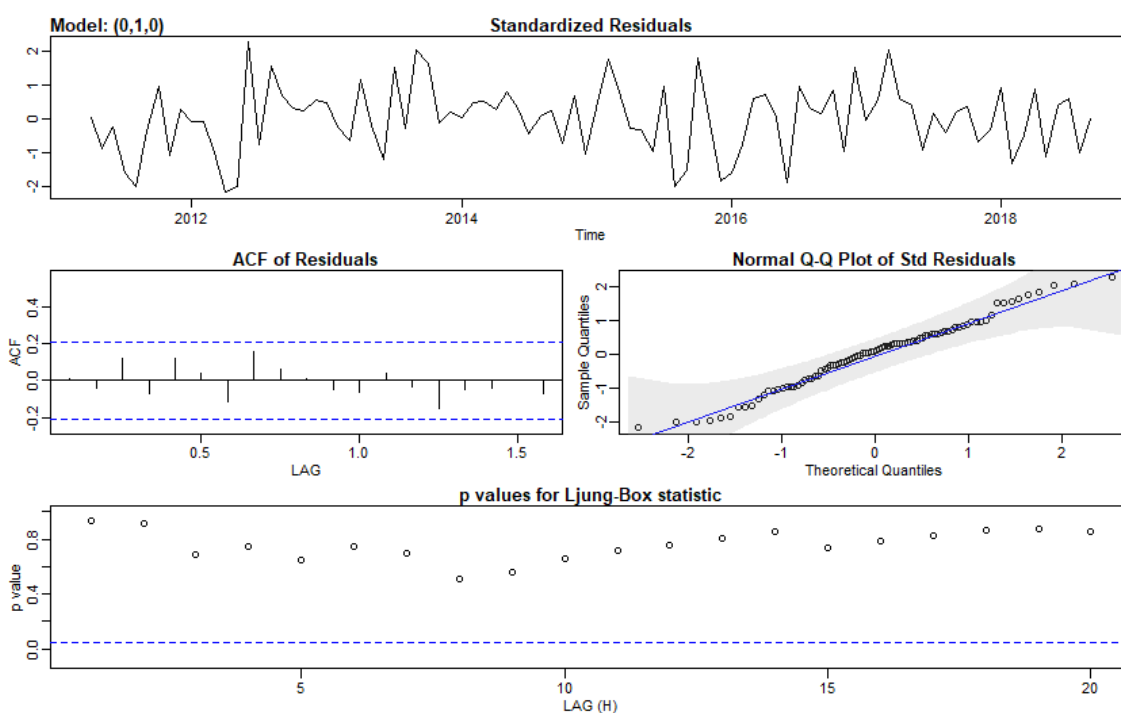


Fuente: Elaboración propia.

Figura 46. Autocorrelogramas normal y parcial de la serie mensual de los valores del IBEX 35 diferenciada.

De nuevo, parece que las características de paseo aleatorio se repiten en los correlogramas de la figura 46, aunque dependiendo de donde se colocasen las bandas de significación podrían encontrarse patrones estacionales. Como se ha optado por mantenerlas constantes en valor 0.2 en todos los análisis, se toma la hipótesis inicial de paseo aleatorio. Comprobando los criterios *AIC* y *BIC* se confirma que de nuevo, el modelo óptimo es el paseo aleatorio.

La figura 47 confirma que no hay problemas con los residuos.



Fuente: Elaboración propia.

Figura 47. Gráficos de los residuos de la serie mensual del IBEX 35 para un modelo ARIMA(0, 1, 0).

Los parámetros que definen al modelo son:

Coefficients:

constant
-16.7381
s.e. 47.8815

sigma^2 estimated as 204044: log likelihood = -670.35, aic = 1344.69

\$degrees_of_freedom

[1] 88

\$ttable

	Estimate	SE	t.value	p.value
constant	-16.7381	47.8815	-0.3496	0.7275

\$AIC

[1] 15.10891

\$AICC

[1] 15.10943

\$BIC

[1] 15.16484

Con estos datos se ejecuta la predicción:

\$pred

Time Series:

Start = 91

End = 100

Frequency = 1

[1] 9372.462 9355.724 9338.986 9322.248 9305.510 9288.771 9272.033 9255.295 9238.557 9221.819

\$se

Time Series:

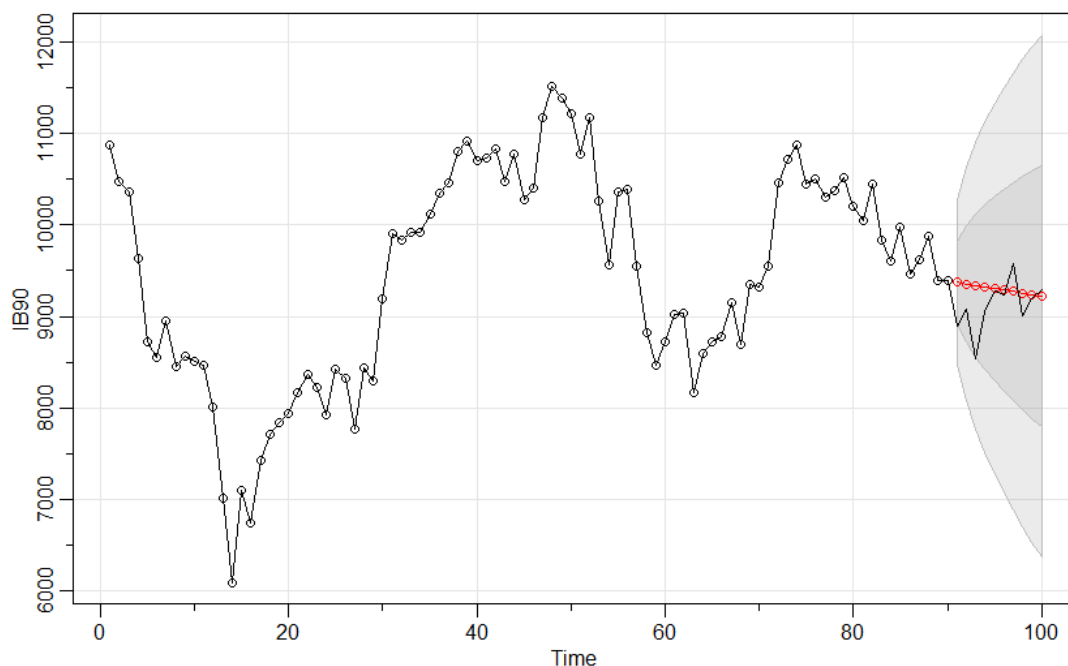
Start = 91

End = 100

Frequency = 1

[1] 451.7128 638.8183 782.3894 903.4255 1010.0604 1106.4658 1195.1196 1277.6366 1355.1383 1428.4412

Y se comparan gráficamente con el resultado real en la figura 48.



Fuente: Elaboración propia.

Figura 48. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie mensual del IBEX 35.

Como se ve, la predicción es bastante adecuada y los valores reales se corresponden bastante bien con los márgenes de error. Sin embargo, esto se debe a que en este caso, al ser un paseo aleatorio sin apenas tendencia lineal, los errores estándar previstos son de gran magnitud, lo que hace a la predicción sea poco precisa pero con mayor probabilidad de acierto.

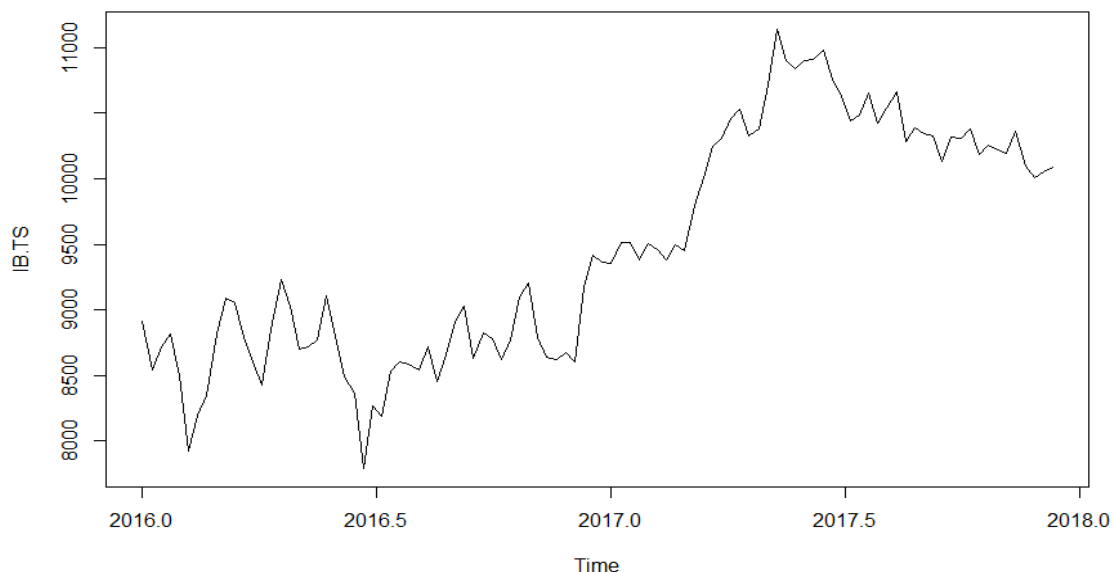
IBEX 35 semanal

Los 100 datos se vuelven a tomar desde la primera semana de 2016 y se representan en la figura 49.

```

Start = c(2016, 1)
End = c(2017, 49)
Frequency = 51
...1100 ...199 ...198 ...197 ...196 ...195 ...194 ...193 ...192 ...191 ...190 ...189 ...188
8909.19 8543.59 8722.89 8815.79 8499.49 7920.79 8194.19 8349.19 8811.59 9090.59 9051.09 8789.79 8602.29
...187 ...186 ...185 ...184 ...183 ...182 ...181 ...180 ...179 ...178 ...177 ...176 ...175
8427.59 8850.89 9232.79 9025.69 8702.09 8721.49 8771.19 9107.29 8801.59 8490.49 8361.99 7787.69 8268.90
...174 ...173 ...172 ...171 ...170 ...169 ...168 ...167 ...166 ...165 ...164 ...163 ...162
8185.90 8531.00 8599.90 8587.20 8539.40 8716.40 8450.60 8659.50 8908.90 9025.50 8633.40 8823.60 8779.40
...161 ...160 ...159 ...158 ...157 ...156 ...155 ...154 ...153 ...152 ...151 ...150 ...149
8624.30 8767.90 9100.40 9201.30 8791.60 8639.20 8622.90 8674.40 8607.10 9169.60 9412.80 9367.70 9352.10
...148 ...147 ...146 ...145 ...144 ...143 ...142 ...141 ...140 ...139 ...138 ...137 ...136
9515.90 9511.60 9380.10 9504.10 9462.70 9378.10 9500.30 9453.50 9798.50 10006.40 10245.80 10309.40 10462.90
...135 ...134 ...133 ...132 ...131 ...130 ...129 ...128 ...127 ...126 ...125 ...124 ...123
10529.00 10326.10 10377.00 10715.80 11135.40 10897.00 10835.40 10904.20 10905.90 10978.30 10759.40 10630.80 10444.50
...122 ...121 ...120 ...119 ...118 ...117 ...116 ...115 ...114 ...113 ...112 ...111 ...110
10488.80 10655.10 10426.60 10536.10 10658.40 10282.90 10385.70 10345.30 10325.50 10129.60 10317.40 10305.00 10381.50
...19 ...18 ...17 ...16 ...15 ...14 ...13 ...12 ...11
10185.50 10258.00 10222.70 10197.50 10357.80 10092.70 10010.40 10053.50 10085.00
    
```

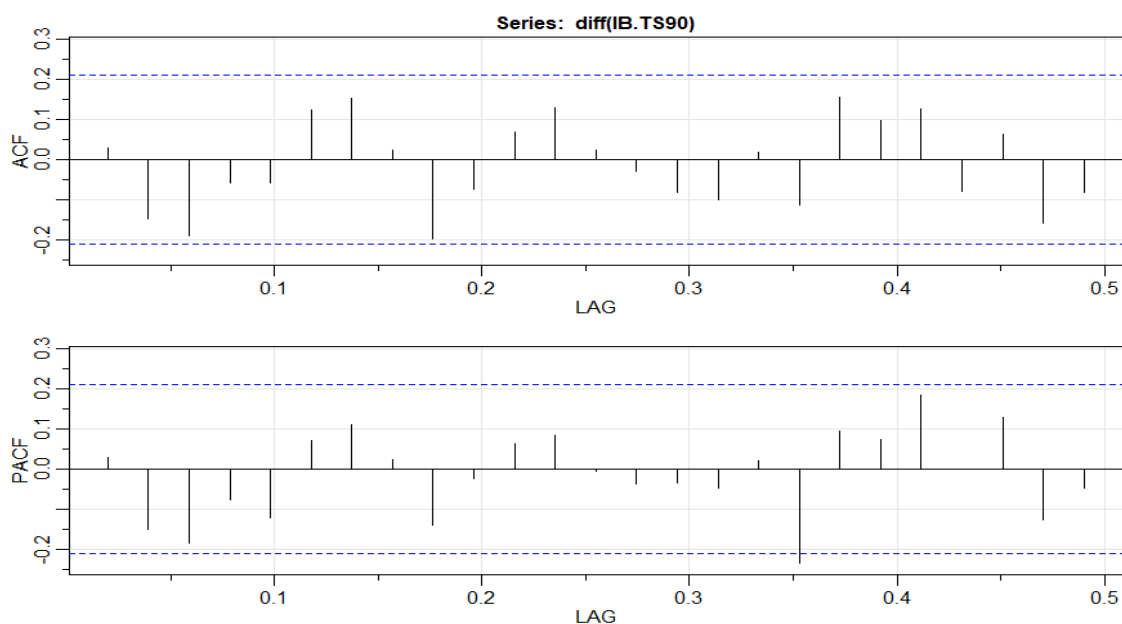
VALOR DE CIERRE A PRINCIPIO DE SEMANA DESDE 2016 HASTA 2019 DEL IBEX 35



Fuente: Elaboración propia.

Figura 49.Figura 36. Gráfico de los valores de cierre de cada lunes del IBEX 35 desde el 4 de enero de 2016 hasta el 18 de diciembre de 2017.

El autocorrelograma de los 90 primeros datos diferenciados de esta serie se da en la figura 50.

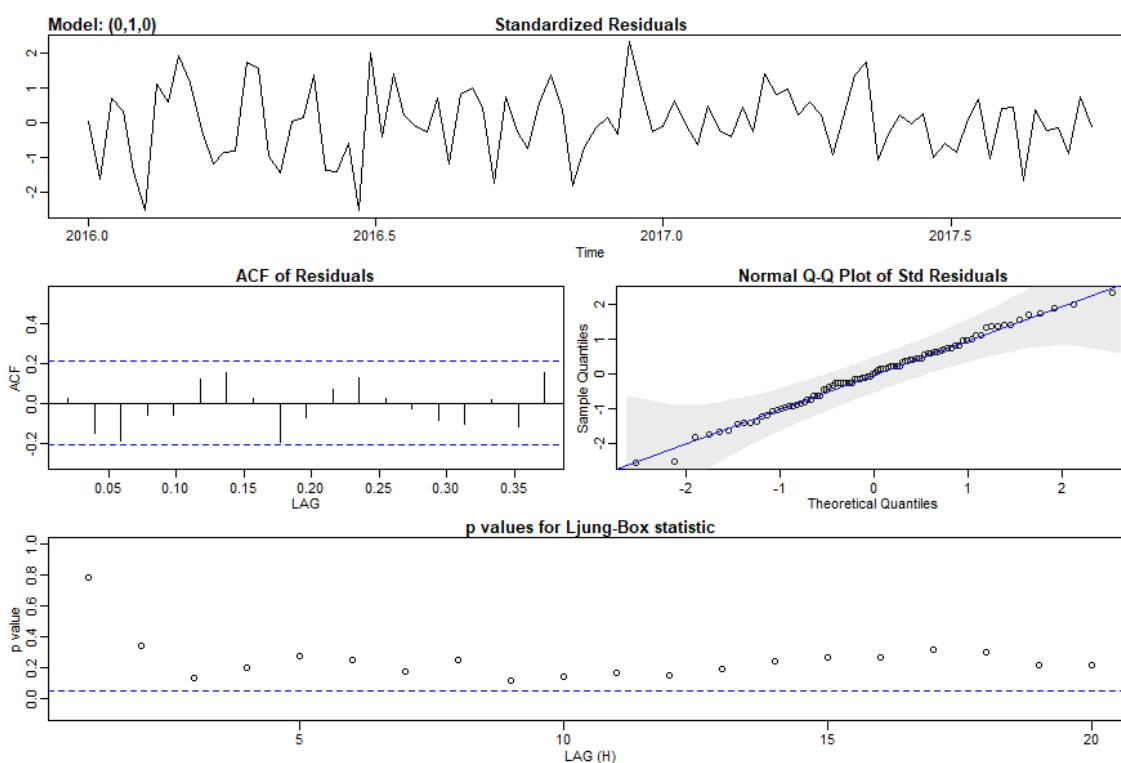


Fuente: Elaboración propia.

Figura 50. Autocorrelogramas normal y parcial de la serie semanal de los valores del IBEX 35 diferenciada.

El autocorrelograma de la serie semanal del IBEX 35 tampoco parece ser una excepción en lo que se refiere a la hipótesis del paseo aleatorio, observando la figura 50. Una vez más no se encuentran correlaciones relevantes ni en el autocorrelograma normal, ni en el parcial.

Se analizan los criterios *AIC* y *BIC* de nuevo y se concluye que la estimación de paseo aleatorio es correcta. Los gráficos para comprobar los resultados de los residuos se representan en la figura 51.



Fuente: Elaboración propia.

Figura 51. Gráficos de los residuos de la serie semanal del IBEX 35 para un modelo ARIMA(0, 1, 0).

Y los parámetros son los siguientes:


```
Coefficients:  
      constant  
      15.6833  
s.e.      24.9627
```

```
sigma^2 estimated as 55459: log likelihood = -612.38, aic = 1228.75
```

```
$degrees_of_freedom  
[1] 88
```

```
$ttable  
      Estimate      SE t.value p.value  
constant 15.6833 24.9627  0.6283  0.5315
```

```
$AIC  
[1] 13.80622
```

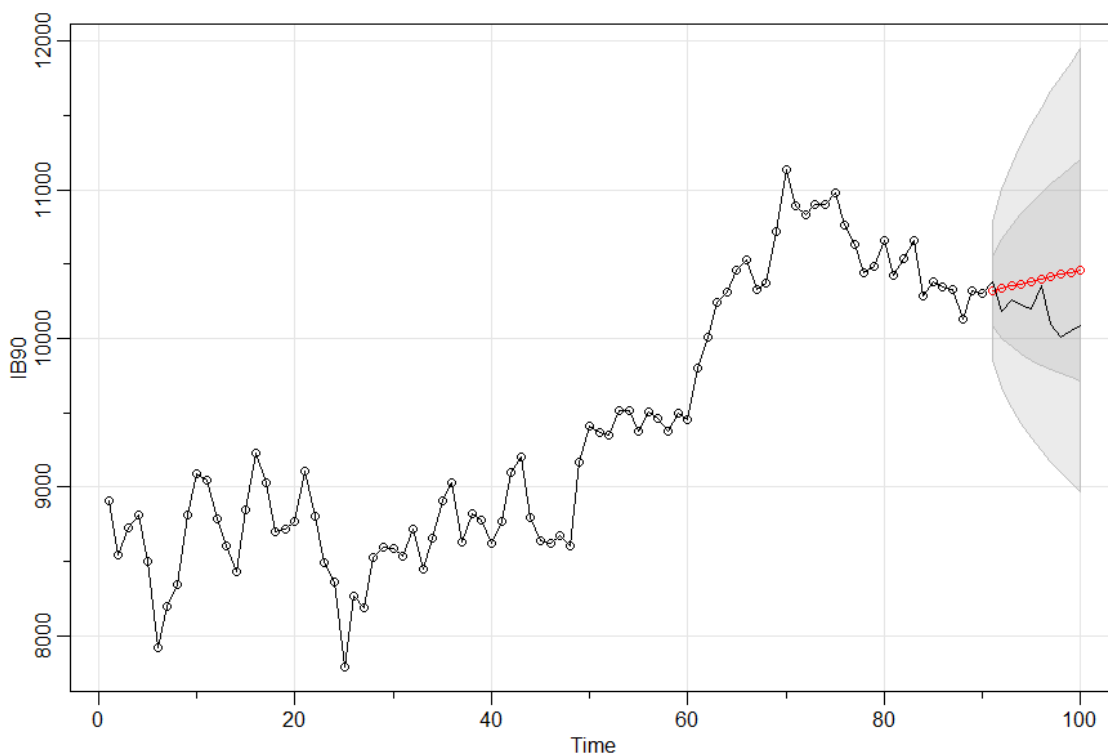
```
$AICc  
[1] 13.80673
```

```
$BIC  
[1] 13.86214
```

De este modo, la predicción y la comparación con los datos reales dan el siguiente resultado:

```
$pred  
Time Series:  
start = 91  
end = 100  
frequency = 1  
[1] 10320.68 10336.37 10352.05 10367.73 10383.42 10399.10 10414.78 10430.47 10446.15 10461.83
```

```
$se  
Time Series:  
start = 91  
end = 100  
frequency = 1  
[1] 235.4971 333.0432 407.8929 470.9942 526.5875 576.8477 623.0668 666.0864 706.4913 744.7072
```



Fuente: Elaboración propia.

Figura 52. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie semanal del IBEX 35.

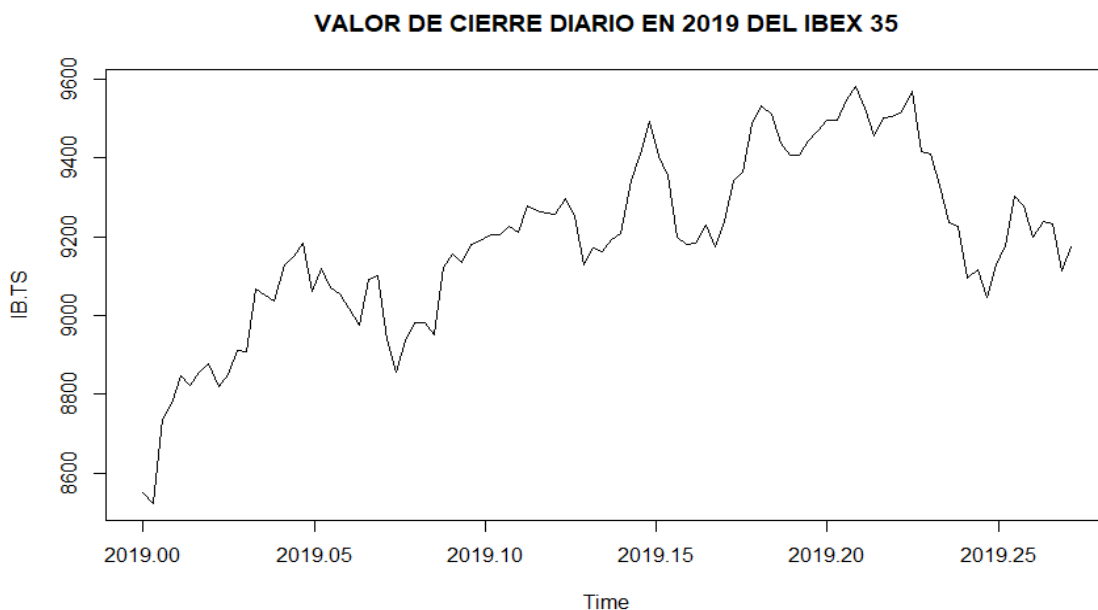
Del mismo modo que con el caso mensual, esta predicción del IBEX vuelve a mostrar un acierto considerable, como se ve en la figura 52, sin embargo, esto es debido al error predicho, por lo que la predicción en sí, aun siendo bastante acertada, tampoco tiene demasiado valor.

IBEX 35 diario

En el caso diario se vuelven a tomar 100 datos desde el 2 de enero de 2019 hasta completar 100.

```

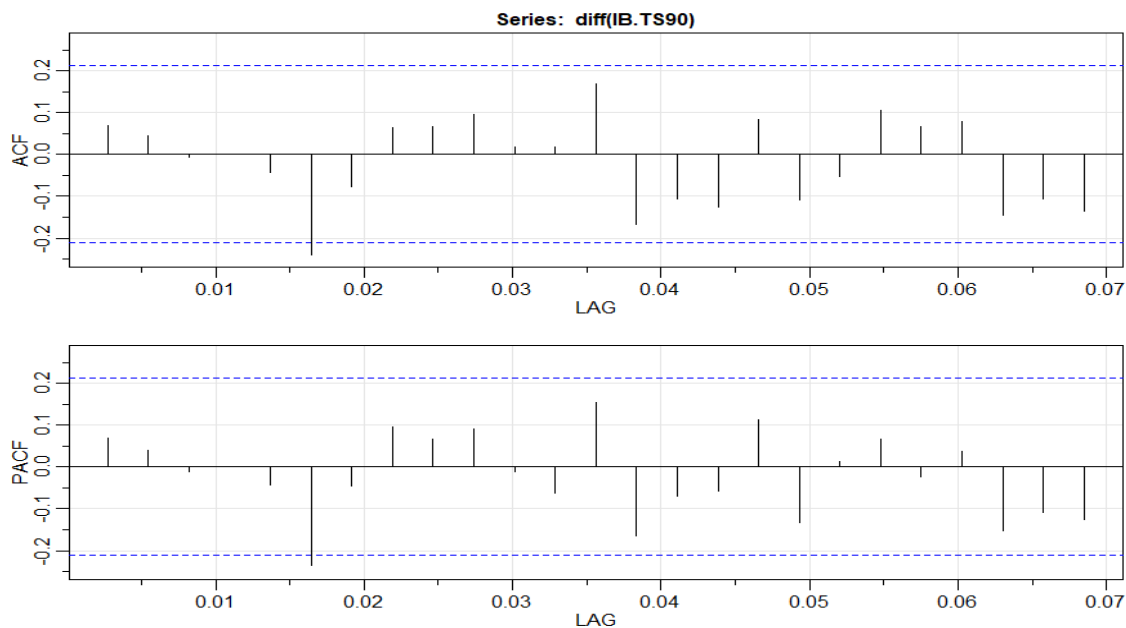
Start = c(2019, 1)
End = c(2019, 100)
Frequency = 365
...1100 ...199 ...198 ...197 ...196 ...195 ...194 ...193 ...192 ...191 ...190 ...189 ...188 ...187
8550.0 8523.3 8737.8 8776.3 8847.3 8823.6 8856.8 8877.1 8818.6 8850.1 8912.7 8908.6 9069.1 9053.8
...186 ...185 ...184 ...183 ...182 ...181 ...180 ...179 ...178 ...177 ...176 ...175 ...174 ...173
9037.5 9128.8 9150.0 9185.2 9062.4 9119.1 9071.5 9056.7 9019.4 8975.2 9092.0 9100.9 8938.3 8856.8
...172 ...171 ...170 ...169 ...168 ...167 ...166 ...165 ...164 ...163 ...162 ...161 ...160 ...159
8936.4 8983.1 8982.4 8952.5 9123.2 9155.5 9136.4 9181.1 9191.2 9204.6 9204.3 9227.2 9211.7 9277.7
...158 ...157 ...156 ...155 ...154 ...153 ...152 ...151 ...150 ...149 ...148 ...147 ...146 ...145
9267.7 9259.8 9258.2 9296.7 9249.9 9129.3 9171.9 9161.7 9192.7 9209.8 9342.2 9409.1 9492.3 9405.6
...144 ...143 ...142 ...141 ...140 ...139 ...138 ...137 ...136 ...135 ...134 ...133 ...132 ...131
9355.6 9199.4 9179.9 9183.1 9229.9 9174.7 9240.3 9341.7 9363.5 9487.8 9534.1 9510.3 9437.7 9407.8
...130 ...129 ...128 ...127 ...126 ...125 ...124 ...123 ...122 ...121 ...120 ...119 ...118 ...117
9406.5 9445.4 9468.5 9497.1 9497.3 9549.8 9581.9 9527.2 9456.4 9501.2 9506.0 9517.2 9570.6 9418.2
...116 ...115 ...114 ...113 ...112 ...111 ...110 ...109 ...108 ...107 ...106 ...105 ...104 ...103
9409.6 9331.0 9235.1 9227.0 9095.2 9117.5 9046.8 9127.6 9177.1 9304.3 9280.1 9199.7 9239.1 9232.2
...12 ...11
9114.0 9174.6
    
```



Fuente: Elaboración propia.

Figura 53. Gráfico de los valores de cierre diario del IBEX 35 desde el 2 de enero de 2019 hasta el 24 de mayo de del mismo año.

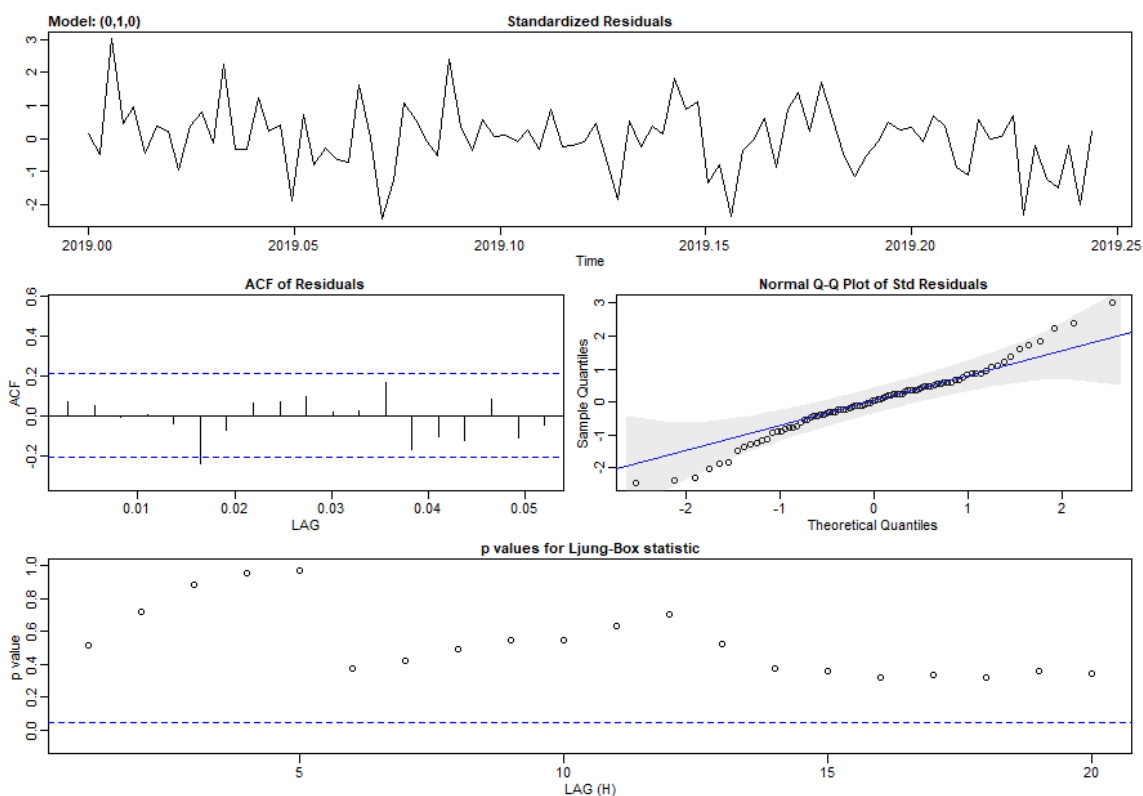
Diferenciando la serie con los 90 primeros datos se obtiene el siguiente correlograma, en la figura 54.



Fuente: Elaboración propia.

Figura 54. Autocorrelogramas normal y parcial de la serie diaria de los valores del IBEX 35 diferenciada.

El sexto correlograma de series del mercado de valores analizadas en este trabajo también representa un paseo aleatorio, confirmando como último resultado lo esperado por la hipótesis formulada en el tercer apartado. Analizando por última vez los criterios *AIC* y *BIC* se obtiene la conclusión definitiva de que el IBEX35 a escala diaria también se comporta como un paseo aleatorio, quedando los residuos de la serie bajo dicho modelo representados en la figura 55.



Fuente: Elaboración propia.

Figura 55. Gráficos de los residuos de la serie diaria del IBEX 35 para un modelo ARIMA(0, 1, 0).

Que como se ve una vez más, son adecuados para el modelo, así que de este modo los parámetros esta vez son los siguientes:

Coefficients:

```
constant
  6.3764
s.e.    7.2887
```

sigma^2 estimated as 4728: log likelihood = -502.81, aic = 1009.63

```
$degrees_of_freedom
[1] 88
```

```
$ttable
      Estimate      SE t.value p.value
constant  6.3764  7.2887  0.8748  0.384
```

```
$AIC
[1] 11.34412
```

```
$AICC
[1] 11.34463
```

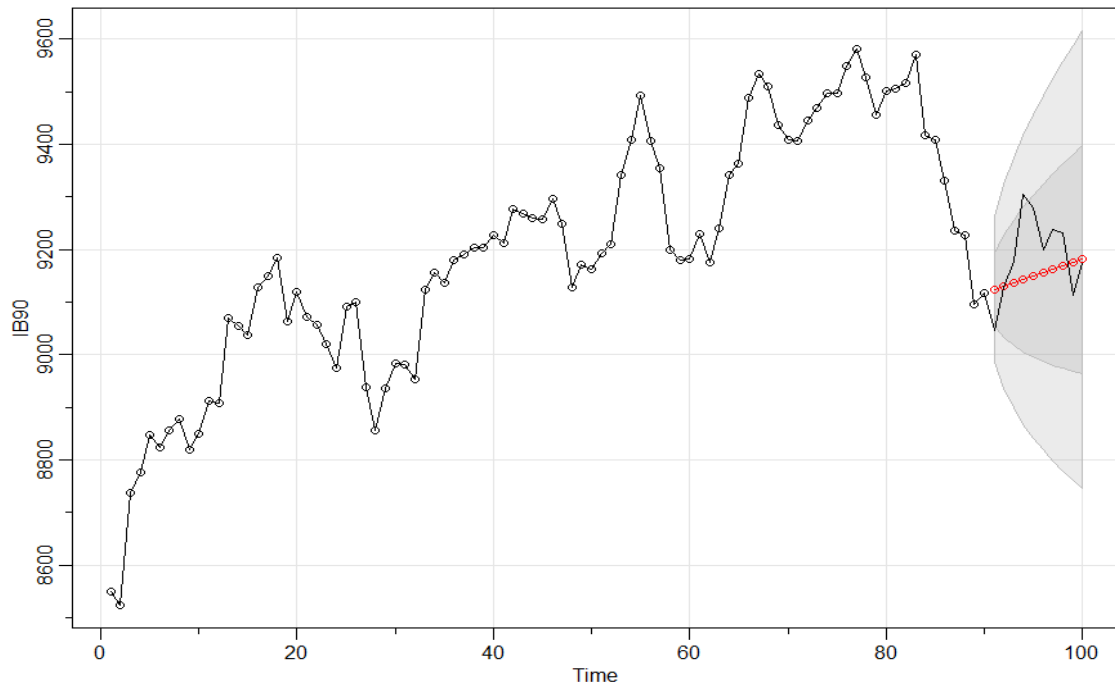
```
$BIC
[1] 11.40004
```

La predicción llevada a cabo otorga los siguientes resultados:

```
$pred
Time Series:
Start = 91
End = 100
Frequency = 1
[1] 9123.876 9130.253 9136.629 9143.006 9149.382 9155.758 9162.135 9168.511 9174.888 9181.264
```

```
$se
Time Series:
Start = 91
End = 100
Frequency = 1
[1] 68.76175 97.24380 119.09884 137.52350 153.75594 168.43120 181.92648 194.48759 206.28524 217.44374
```

Que introduciéndolos en la gráfica y comparándolos con los valores reales tienen el siguiente aspecto, en la figura 56.



Fuente: Elaboración propia.

Figura 56. Gráfico comparativo con la predicción y los 10 datos previamente eliminados de la serie diaria del IBEX 35.

Se puede ver que a pequeña escala prevalece el carácter de los errores de gran magnitud, dado que la serie del IBEX 35 es más oscilante y menos estable que la del índice americano. La predicción, una vez más, vuelve a ser óptima pero de menor valor debido al gran rango de posibilidades que ésta ofrece.

7. CONCLUSIONES

Pese a que al comienzo de la realización del trabajo se pensaba que se iba a refutar parcialmente, más que confirmar, la hipótesis del mercado eficiente y el paseo aleatorio, el resultado ha sido completamente el opuesto. Se esperaba encontrar signos de eficiencia del mercado en las series, pero no encontrarlo en todas y cada una de ellas de forma tan clara. De este modo, la principal conclusión es que, al menos en las series analizadas, el comportamiento aleatorio del mercado bajo una perspectiva matemática es más que una evidencia, es un hecho.

Tan solo una de las series resultaba tener un comportamiento mejor con un modelo distinto al paseo aleatorio, pero sus residuos, al adoptar éste, dejaban de tener un comportamiento tan perfectamente normal, lo que quiere decir que su ruido blanco perdía pureza. Además, su parámetro, autorregresivo en ese caso, era prácticamente insignificante, por lo que resultaba ser más molesto que productivo en la predicción, es decir, dicha serie también es un paseo aleatorio prácticamente puro, y así se ha modelizado.

Esta perspectiva arroja implicaciones más allá de la difícil modelización del mercado bursátil mediante modelos matemáticos de predicción. También ofrece una evidencia demoledora de la ineffectividad de herramientas como el análisis técnico en estas escalas de estudio.

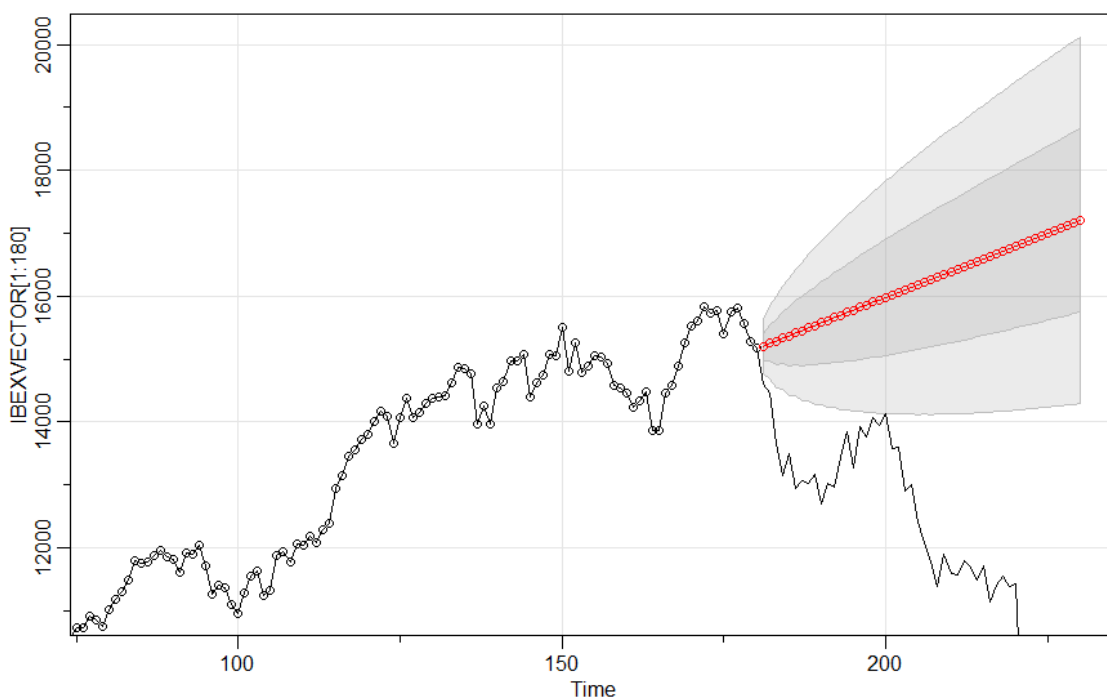
Existen múltiples fondos de inversión y asesores financieros que se jactan de la gran utilidad del análisis técnico y su habilidad con éste para obtener plusvalías, pero lo cierto es que no se conoce ningún inversor individual ni institución de inversión colectiva que haya conseguido acreditar sus beneficios con estas técnicas de forma prolongada en el tiempo. Este trabajo pasa a ser una evidencia más de su ineficacia en este aspecto, si bien es cierto que no ha de ser rechazado del todo, ya que en ocasiones puntuales y desde la perspectiva de las finanzas conductuales existen evidencias de su utilidad para predecir el comportamiento psicológico del mercado, especialmente en el corto plazo. Esto tiene utilidad práctica, por ejemplo, para el trading de alta frecuencia, cosa que está lejos de los inversores estándar y que no hace más que volver el mercado aún más eficiente para éstos.

Suponiendo que el mercado analizado es eficiente en la versión fuerte o semi-fuerte de la hipótesis, que es lo que parecen demostrar los cálculos de la parte

práctica de éste trabajo, se podría afirmar, al menos, que el estudio o análisis de valor de las acciones del mercado bursátil para calcular su precio futuro tampoco es tan útil como los defensores del análisis fundamental, por ejemplo, plantean.

En cuanto a la capacidad de predicción de ARIMA en el mercado bursátil, se concluye que es una herramienta útil para detectar tendencias y calcular probabilidades de los siguientes movimientos en situaciones estables, pero viendo los resultados no se considera en absoluto precisa ni útil para utilizarla como herramienta de inversión, sobre todo a largo plazo.

ARIMA obtiene mayor éxito en situaciones en la que la tendencia no cambia, es decir, los modelos recomiendan ponerse en largo siempre que el precio va al alza, y ponerse en corto siempre que el mercado está a la baja. Si se plantease calcular, a través de modelos ARIMA, el precio de acciones en un momento como puede ser una burbuja, siempre nos recomendará comprar, y está claro que no se tratará de una buena predicción. Esto se puede ver si, por ejemplo, se hace una predicción de qué hubiera pasado con el IBEX 35 en 2007, justo antes de la crisis:



Fuente: Elaboración propia.

Figura 57. Gráfico comparativo de la serie real del IBEX 35 en la etapa de la crisis de 2008 junto con una predicción a partir del momento más alto de la burbuja mediante modelos ARIMA de paseos aleatorios.

De este modo se concluye que ARIMA, y más en general, todos los modelos matemáticos que traten de predecir el precio de las acciones en base a los valores pasados, son inútiles para el inversor individual, más allá de ser una herramienta para intuir hacia dónde se dirige el mercado, pero en ningún caso determinante.

Por último, como posibles líneas futuras, cabe destacar que podría ser interesante evaluar estas propiedades en mercados con menor liquidez, y en consecuencia, presumiblemente menos eficientes, para ver la utilidad de los modelos matemáticos predictivos como ARIMA en éstos. Esto se podría llevar a cabo en mercados con bolsas de menor tamaño, como las de algún país en vías de desarrollo, o en mercados de activos inmovilizados, como el inmobiliario.

8. BIBLIOGRAFÍA

- Abellán, J. L. (s.f.). *Hipótesis del mercado eficiente*. Recuperado el 07 de 07 de 2019, de <https://economipedia.com/definiciones/hipotesis-del-mercado-eficiente.html>
- Acuñas, V. d. (2011). *Modelización de la Previsión del Mercado del Consumo Doméstico de Gas Natural*. Bilbao: UPV/EHU.
- Carpintero, Á. T., Nieto Salazar, S., & Carvajal Olaya, P. (2003). *Modelo de Predicción del Precio de la Acción Ordinaria Cementos Argos*. Pereira: Universidad Tecnológica de Pereira.
- Casimiro, M. P. (2009). *Análisis de Series Temporales: Modelos ARIMA*. Bilbao: UPV/EHU.
- Cornelissen, J. (s.f.). *Introduction to R*. Recuperado el 05 de 06 de 2019, de Datacamp: <https://www.datacamp.com/courses/free-introduction-to-r>
- CRAN. (20 de 05 de 2019). Obtenido de r-project: <https://cran.r-project.org/>
- Cryer, J., & Chan, K.-s. (2008). *Time Series Analysis: With Applications in R*. Springer.
- Estadística, I. V. (2019). *Valores históricos del PIB de la CAPV*. Recuperado el 28 de 06 de 2019, de Eustat: eustat.eus/bankupx/sq/52a852be-9f4d-4965-a94a-d5de605b8fd1
- Galán, J. S. (s.f.). *Teoría del paseo aleatorio*. Recuperado el 07 de 07 de 2019, de <https://economipedia.com/definiciones/teoria-paseo-aleatorio.html>
- Hernández, A. S. (06 de 2014). *Objetos en R: Variables numéricas*. Recuperado el 08 de 06 de 2019, de http://www.dma.ulpgc.es/profesores/personal/stat/cursosR4ULPGC/6b-VARIABLES_Numericas.html
- Lodeiro, F. J. (03 de 2015). *Índices bursátiles: Qué son, para qué sirven, tipos y ejemplos*. Recuperado el 07 de 07 de 2019, de <https://www.academiadeinversion.com/indices-bursatiles-que-son-para-que-sirven-tipos-y-ejemplos/>
- López, A. H. (2015). *Modelos Predictivos para el mercado FOREX*. Murcia: Universidad de Murcia.
- Malkiel, B. G. (1973). *A Random Walk Through Wall Street*. Estados Unidos: W. W. Norton & Company.
- Matteson, D. S. (s.f.). *Introduction to Time Series Analysis*. Recuperado el 07 de 06 de 2019, de Datacamp: <https://www.datacamp.com/courses/introduction-to-time-series-analysis>
- Schouwenaars, F. (s.f.). *Intermediate R*. Recuperado el 06 de 06 de 2019, de Datacamp: <https://www.datacamp.com/courses/intermediate-r>
- Stoffer, D. (s.f.). *ARIMA Modeling with R*. Recuperado el 08 de 06 de 2019, de Datacamp: <https://www.datacamp.com/courses/arima-modeling-with-r>
- Tornero, J. (27 de 04 de 2017). *Introducción al Forecasting con R Statistics*. Recuperado el 05 de 07 de 2019, de https://www.doctormetrics.com/2017/04/27/introduccion-al-forecasting-con-r-statistics/#.XRuP04_tbc

- Yahoo. (10 de 07 de 2019). *Datos Históricos del IBEX 35*. Recuperado el 10 de 07 de 2019, de <https://es.finance.yahoo.com/quote/%5EIBEX/history?p=%5EIBEX>
- Yahoo. (10 de 07 de 2019). *Datos históricos del S&P 500*. Recuperado el 10 de 07 de 2019, de <https://es.finance.yahoo.com/quote/%5EGSPC/history?p=%5EGSPC&.tsrc=fin-srch>
- Yrigoyen, C. C. (2012). *Manual de econometría de la empresa*. Madrid: Universidad Autónoma de Madrid.
- Zabala, A. G. (2019). *Legislación Financiera/ Dirección financiera*. Bilbao: UPV/EHU.

ANEXOS

9. ANEXO I- Paquetes de funciones instalados y utilizados durante el trabajo.

Se recogen todos los paquetes que se han instalado y se han utilizado para desarrollar el trabajo o durante el proceso de aprendizaje.

Name	Description	Version
User Library		
<input type="checkbox"/> assertthat	Easy Pre and Post Assertions	0.2.1
<input type="checkbox"/> asts	Applied Statistical Time Series Analysis	1.9
<input type="checkbox"/> backports	Reimplementations of Functions Introduced Since R-3.0.0	1.1.4
<input type="checkbox"/> cellranger	Translate Spreadsheet Cell Ranges to Rows and Columns	1.1.0
<input type="checkbox"/> cli	Helpers for Developing Command Line Interfaces	1.1.0
<input type="checkbox"/> colorspace	A Toolbox for Manipulating and Assessing Colors and Palettes	1.4-1
<input type="checkbox"/> crayon	Colored Terminal Output	1.3.4
<input type="checkbox"/> curl	A Modern and Flexible Web Client for R	3.3
<input type="checkbox"/> digest	Create Compact Hash Digests of R Objects	0.6.19
<input type="checkbox"/> fansi	ANSI Control Sequence Aware String Functions	0.4.0
<input type="checkbox"/> forecast	Forecasting Functions for Time Series and Linear Models	8.7
<input type="checkbox"/> fracdiff	Fractionally differenced ARIMA aka ARFIMA(p,d,q) models	1.4-2
<input type="checkbox"/> ggplot2	Create Elegant Data Visualisations Using the Grammar of Graphics	3.2.0
<input type="checkbox"/> glue	Interpreted String Literals	1.3.1
<input type="checkbox"/> gtable	Arrange 'Grob's' in Tables	0.3.0
<input type="checkbox"/> hms	Pretty Time of Day	0.4.2
<input type="checkbox"/> labeling	Axis Labeling	0.3
<input type="checkbox"/> lazyeval	Lazy (Non-Standard) Evaluation	0.2.2
<input type="checkbox"/> lmtest	Testing Linear Regression Models	0.9-37
<input type="checkbox"/> magrittr	A Forward-Pipe Operator for R	1.5
<input type="checkbox"/> munsell	Utilities for Using Munsell Colours	0.5.0
<input type="checkbox"/> pillar	Coloured Formatting for Columns	1.4.2
<input type="checkbox"/> pkgconfig	Private Configuration for 'R' Packages	2.0.2
<input type="checkbox"/> plyr	Tools for Splitting, Applying and Combining Data	1.8.4
<input type="checkbox"/> prettyunits	Pretty, Human Readable Formatting of Quantities	1.0.2
<input type="checkbox"/> progress	Terminal Progress Bars	1.2.2
<input type="checkbox"/> quadprog	Functions to Solve Quadratic Programming Problems	1.5-7
<input type="checkbox"/> quantmod	Quantitative Financial Modelling Framework	0.4-15
<input type="checkbox"/> R6	Encapsulated Classes with Reference Semantics	2.4.0
<input type="checkbox"/> RColorBrewer	ColorBrewer Palettes	1.1-2
<input type="checkbox"/> Rcpp	Seamless R and C++ Integration	1.0.1
<input type="checkbox"/> RcppArmadillo	'Rcpp' Integration for the 'Armadillo' Templated Linear Algebra Library	0.9.500.2.0
<input type="checkbox"/> readxl	Read Excel Files	1.3.1
<input type="checkbox"/> rematch	Match Regular Expressions with a Nicer 'API'	1.0.1
<input type="checkbox"/> reshape2	Flexibly Reshape Data: A Reboot of the Reshape Package	1.4.3
<input type="checkbox"/> rlang	Functions for Base Types and Core R and 'Tidyverse' Features	0.4.0
<input type="checkbox"/> scales	Scale Functions for Visualization	1.0.0
<input type="checkbox"/> stringi	Character String Processing Facilities	1.4.3
<input type="checkbox"/> stringr	Simple, Consistent Wrappers for Common String Operations	1.4.0

<input type="checkbox"/>	stringr	Simple, Consistent Wrappers for Common String Operations	1.4.0
<input type="checkbox"/>	tibble	Simple Data Frames	2.1.3
<input type="checkbox"/>	timeDate	Rmetrics - Chronological and Calendar Objects	3043.102
<input type="checkbox"/>	tseries	Time Series Analysis and Computational Finance	0.10-47
<input type="checkbox"/>	TTR	Technical Trading Rules	0.23-4
<input type="checkbox"/>	urca	Unit Root and Cointegration Tests for Time Series Data	1.3-0
<input type="checkbox"/>	utf8	Unicode Text Processing	1.1.4
<input type="checkbox"/>	vctrs	Vector Helpers	0.1.0
<input type="checkbox"/>	viridisLite	Default Color Maps from 'matplotlib' (Lite Version)	0.3.0
<input type="checkbox"/>	withr	Run Code 'With' Temporarily Modified Global State	2.1.2
<input type="checkbox"/>	xts	eXtensible Time Series	0.11-2
<input type="checkbox"/>	zeallot	Multiple, Unpacking, and Destructuring Assignment	0.1.0
<input type="checkbox"/>	zoo	S3 Infrastructure for Regular and Irregular Time Series (Z's Ordered Observations)	1.8-6
System Library			
<input checked="" type="checkbox"/>	base	The R Base Package	3.6.0
<input type="checkbox"/>	boot	Bootstrap Functions (Originally by Angelo Canty for S)	1.3-22
<input type="checkbox"/>	class	Functions for Classification	7.3-15
<input type="checkbox"/>	cluster	"Finding Groups in Data": Cluster Analysis Extended Rousseeuw et al.	2.0.8
<input type="checkbox"/>	codetools	Code Analysis Tools for R	0.2-16
<input type="checkbox"/>	compiler	The R Compiler Package	3.6.0
<input checked="" type="checkbox"/>	datasets	The R Datasets Package	3.6.0
<input type="checkbox"/>	foreign	Read Data Stored by 'Minitab', 'S', 'SAS', 'SPSS', 'Stata', 'Systat', 'Weka', 'dBase', ...	0.8-71
<input checked="" type="checkbox"/>	graphics	The R Graphics Package	3.6.0
<input checked="" type="checkbox"/>	grDevices	The R Graphics Devices and Support for Colours and Fonts	3.6.0
<input type="checkbox"/>	grid	The Grid Graphics Package	3.6.0
<input type="checkbox"/>	KernSmooth	Functions for Kernel Smoothing Supporting Wand & Jones (1995)	2.23-15
<input type="checkbox"/>	lattice	Trellis Graphics for R	0.20-38
<input type="checkbox"/>	MASS	Support Functions and Datasets for Venables and Ripley's MASS	7.3-51.4
<input type="checkbox"/>	Matrix	Sparse and Dense Matrix Classes and Methods	1.2-17
<input checked="" type="checkbox"/>	methods	Formal Methods and Classes	3.6.0
<input type="checkbox"/>	mgcv	Mixed GAM Computation Vehicle with Automatic Smoothness Estimation	1.8-28
<input type="checkbox"/>	nlme	Linear and Nonlinear Mixed Effects Models	3.1-139
<input type="checkbox"/>	nnet	Feed-Forward Neural Networks and Multinomial Log-Linear Models	7.3-12
<input type="checkbox"/>	parallel	Support for Parallel computation in R	3.6.0
<input type="checkbox"/>	rpart	Recursive Partitioning and Regression Trees	4.1-15
<input type="checkbox"/>	spatial	Functions for Kriging and Point Pattern Analysis	7.3-11
<input type="checkbox"/>	splines	Regression Spline Functions and Classes	3.6.0
<input type="checkbox"/>	stats4	Statistical Functions using S4 Classes	3.6.0
<input type="checkbox"/>	survival	Survival Analysis	2.44-1.1
<input type="checkbox"/>	tcltk	Tcl/Tk Interface	3.6.0
<input type="checkbox"/>	tools	Tools for Package Development	3.6.0
<input type="checkbox"/>	translations	The R Translations Package	3.6.0
<input checked="" type="checkbox"/>	utils	The R Utils Package	3.6.0

10. ANEXO II- Series utilizadas en los ejemplos del trabajo:

-U.S. Unemployment (unemp; *astsa*)

Description

Monthly U.S. Unemployment series (1948-1978, n = 372)

Usage

data (unemp)

Format

The format is: Time-Series [1:372] from 1948 to 1979: 235 281 265 241 201 ...

References

<http://www.stat.pitt.edu/stoffer/tsa4/>

-U.S. Monthly Live Births (births; *astsa*)

Description

Monthly live births (adjusted) in thousands for the United States, 1948-1979.

Format

The format is: Time-Series [1:373] from 1948 to 1979: 295 286 300 278 272 268 308 321 313 308 ...

References

<http://www.stat.pitt.edu/stoffer/tsa4/> and <http://www.stat.pitt.edu/stoffer/tsda/>

-Global mean land-ocean temperature deviations to 2015 (globtemp; astsa)

Description

Global mean land-ocean temperature deviations (from 1951-1980 average), measured in degrees centigrade, for the years 1880-2015. This was an update of `gtemp`, but `gtemp_land` and `gtemp_ocean` are the most recent updates.

Format

The format is: Time-Series [1:136] from 1880 to 2015: -0.2 -0.11 -0.1 -0.2 -0.28 -0.31 -0.3 -0.33 -0.2 -0.11 ...

Details

The data were changed after 2011, so there are discrepancies between this data set and `gtemp`. The differences are explained here: <http://www1.ncdc.noaa.gov/pub/data/ghcn/v3/GHCNM-v3.2.0-FAQ.pdf>.

Source

<http://data.giss.nasa.gov/gistemp/graphs/>

References

<http://www.stat.pitt.edu/stoffer/tsa4/>

See Also

[gtemp_land](#), [gtemp_ocean](#), [globtemp1](#), [gtemp](#), [gtemp2](#)

-Quarterly U.S. GNP (gnp; astsa)

Description

Seasonally adjusted quarterly U.S. GNP from 1947(1) to 2002(3).

Format

The format is: Time-Series [1:223] from 1947 to 2002: 1489 1497 1500 1524 1547 ...

Source

<https://research.stlouisfed.org/>

References

<http://www.stat.pitt.edu/stoffer/tsa4/> and <http://www.stat.pitt.edu/stoffer/tsda/>

See Also

[gdp](#)

-U.S. Unemployment Rate (UnempRate; astsa)

Description

Monthly U.S. unemployment rate in percent unemployed (Jan, 1948 - Nov, 2016, n = 827)

Format

The format is: Time-Series [1:827] from 1948 to 2017: 4 4.7 4.5 4 3.4 3.9 3.9 3.6 3.4 2.9 ...

Source

<https://data.bls.gov/timeseries/LNU04000000/>

References

<http://www.stat.pitt.edu/stoffer/tsa4/> and <http://www.stat.pitt.edu/stoffer/tsda/>

11. ANEXO III- Código utilizado en el cuarto y quinto apartado para explicar el funcionamiento de R y las bases teóricas:

*Los códigos de este anexo han sido ejecutados directamente en la consola en su mayoría. Se ha intrducido el código original, por lo que hay pruebas y código que no da ningún resultado, es erróneo, o es de consulta.

```
ra<-arima.sim(model=list(0,0,0), n=100)

plot(ra)

arima(ra, order=c(0,0,0))

rw<-arima.sim(model=list(0,1,0),n=10000)

plot(rw)

?plot

plot(rw[1:500])

plot(rw[1:500], type="l")

plot(rw[1:50], type="l")

rw1<-diff(rw)

plot(rw1[1:50])

plot(rw1[1:50], type="l")

library(boot, lib.loc = "C:/Program Files/R/R-3.6.0/library")

detach("package:boot", unload = TRUE)

rw<-arima.sim(list(order=c(0,1,0)),mean=5,n=100)

plot(rw)

rw<-arima.sim(list(order=c(0,1,0)),mean=1,n=50)

plot(rw)

rw<-arima.sim(list(order=c(0,1,0)),mean=0,n=50)

plot(rw)

rw<-arima.sim(list(order=c(0,1,0)),mean=0,n=50)

> plot(rw)

rw<-arima.sim(list(order=c(0,1,0)),mean=0,n=50)
```

```
plot(rw)
rw<-arima.sim(list(order=c(0,1,0)),mean=0,n=50)plot(rw)
rw<-arima.sim(list(order=c(0,1,0)),mean=0,n=50)
> plot(rw)
rw<-arima.sim(list(order=c(0,1,0)),mean=0,n=50)
plot(rw)
rw<-arima.sim(list(order=c(0,1,0)),mean=0,n=50)
plot(rw, main= "Paseo Aleatorio")
X<-100
x<-100
print(x)
x<-100
print(x)
x<-100
print(x)
LOGICAL1<-TRUE
LOGICAL2<-FALSE
LOGICAL3<-TRUE
MEDIA<-(LOGICAL1+LOGICAL2+LOGICAL3)/3
print(MEDIA)
CHAR<-"Programar ARIMA con R"
print(CHAR)
class(x)
class(LOGICAL1)
class(CHAR)
Mi.Array<-c(1,2,3,4,5)
print(Mi.Array)
data.frame(c("Ojos","Nariz","Boca","Brazos"),c(2,1,1,2))
DF<-data.frame(c("Ojos","Nariz","Boca","Brazos"),c(2,1,1,2))
print(DF)
```

```

DF
CUERPO<-c("Ojos","Nariz","Boca","Brazos")
NUMERO<-c(2,1,1,2)
DaFr<-data.frame(CUERPO,NUMERO)
DaFr
list(c(1,2,3),c(TRUE,FALSE),c("ARIMA"))
LISTA<-list(c(1,2,3),c(TRUE,FALSE),c("ARIMA"))
print(LISTA)
if( "Condición" ){
if( "Condición" ){                "tratamiento de datos 1"           }else{
"tratamiento de datos 2"          }
?mean
?round
round(7,98989,3)
round(7.89479,3)
round(digits = 3, x=7.89479)
if( "Condición requerida"){      "instrucciones para tratar datos 1"
}else{                             "instrucciones para tratar datos 2"
}
install.packages("astsa")
library(astsa)
if( "Condición" ){                "tratamiento de datos 1"           }else{
"tratamiento de datos 2"          }
while ( "Condición a eliminar" ){ "Instrucciones para tratamiento de datos"
}
for (i in 1:"numero de iteraciones deseadas"){ "instrucciones para el tratamiento de
datos"      }
Calcular.Media<- function(x,y,z){    (x+y+z)/3      }
Calcular.Media(18,16,5)
install.packages("zoo")
library(zoo)
install.packages("xts")

```

```
library(xts)

vector.de.datos<-
c(60,45,354,45,35,354,35,35,45,34,345,323,23,3,44,554,564,34,534,545,764,34,23,45,67,54,34
,56,34,23,435,5,354,34)

mi.serie<-ts(vector.de.datos,start=1980,frequency = 12)

print(mi.serie)

class(mi.serie)

plot(inflation)

plot(UnempRate)

plot(globtemp)

difglobtemp<-diff(globtemp)

plot(difglobtemp)

plot(globtemp, main= "TEMPERATURA GLOBAL MEDIA DEL PLANETA")

plot(difglobtemp, main= "TEMPERATURA GLOBAL MEDIA DEL PLANETA DIFERENCIADA")

plot(unemp)

plot(tail(unemp,n=72),main="Desempleo en EEUU")

ts.plot(tail(unemp,n=36),main="Desempleo en EEUU", type = "l")

ts.plot(tail(unemp,n=72),main="Desempleo en EEUU", type = "l")

print(tail(unemp))

plot(UnempRate[1990:2016])

plot(tail(UnempRate, n=72),type="l",main="Desempleo en EEUU")

des<-(tail(UnempRate, n=72))

plot(des)

des<-(tail(UnempRate, n=72))

deesz<-ts(des,start=2011,frequency = 12)

plot(deesz,main="TASA DE DESEMPLEO DE EEUU",type="l")

dif<-diff(deesz,12)

plot(dif,main="TASA DE DESEMPLEO DE EEUU SIN ESTACIONALIDAD",type="l")

dif2<-diff(deesz,s=12)

plot(dif2,main="TASA DE DESEMPLEO DE EEUU SIN ESTACIONALIDAD",type="l")

plot(salmon)
```

```
plot(chicken)
plot(oil)
plot(head(prodn, n=100))
plot((des))
plot(UnempRate)
?plot
plot(unemp, 1980,1990)
plot(gas)
plot(gtemp_land,gtemp_ocean)
plot(gtemp_land,gtemp_ocean, type="l")
?plot
EuStockMarkets
plot(EuStockMarkets$DAX,EuStockMarkets$CAC)
EuStockMarkets$DAX
class(EuStockMarkets)
EuStockMarkets[,2]
plot(EuStockMarkets[,1],EuStockMarkets[,3], type="l",main("GRÁFICO DE DISPERSIÓN ENTRE EL ÍNDICE DAX Y EL ÍNDICE CAC"))
plot(EuStockMarkets[,1],EuStockMarkets[,3], type="l",main="GRÁFICO DE DISPERSIÓN ENTRE EL ÍNDICE DAX Y EL ÍNDICE CAC")
plot(EuStockMarkets[,1],EuStockMarkets[,3], type="b",main="GRÁFICO DE DISPERSIÓN ENTRE EL ÍNDICE DAX Y EL ÍNDICE CAC")
acf(inflation)
acf(globtemp)
acf(globtemp, main="CORRELOGRAMA DE LA SERIE DE TEMPERATURAS GLOBALES")
RUIDO.BLANCO<-arima.sim(model = list(order=c(0,0,0)),n=50)
plot(RUIDO.BLANCO)
plot(RUIDO.BLANCO, main="SERIE SIMULADA DE RUIDO BLANCO")
acf(globtemp, main="AUTOCORRELOGRAMA DE LA SERIE DE TEMPERATURAS GLOBALES")
PASEO.ALEATORIO1<-arima.sim(model = list(order=c(0,1,0)),n=50,MAIN=0)
PASEO.ALEATORIO1<-arima.sim(model = list(order=c(0,1,0)),n=50,mean=0)
```

```
PASEO.ALEATORIO2<-arima.sim(model = list(order=c(0,1,0)),n=50,mean=2)
PASEO.ALEATORIO3<-arima.sim(model = list(order=c(0,1,0)),n=50,mean=-2)
plot(PASEO.ALEATORIO2,main="PASEO ALEATORIO CON MEDIA POSITIVA")
plot(PASEO.ALEATORIO2,main="PASEO ALEATORIO CON MEDIA NULA")
plot(PASEO.ALEATORIO1,main="PASEO ALEATORIO CON MEDIA NULA")
plot(PASEO.ALEATORIO3,main="PASEO ALEATORIO CON MEDIA NEGATIVA")
AR<-arima.sim(model = list(order=c(1,0,0)),n=50)
AR<-arima.sim(model = list(AR=0.4),n=50)
predict(AR,n.ahead=4)
sarima.for(AR,n.ahead=4)
?predict
AR<-arima.sim(model = list(AR=0.4),n=50)
AR_ARIMA<-arima(AR, order=c(1,0,0))
predict(AR_ARIMA)
MA<-arima.sim(model = list(MA=0.4),n=50)
MA_ARIMA<-arima(AR, order=c(0,0,1))
predict(MA_ARIMA)
acf(AR)
acf(MA)
acf(AR, main="AUTOCORRELOGRAMA DE AR(1) CON PARÁMETRO 0.4")
acf(MA, main="AUTOCORRELOGRAMA DE MA(1) CON PARÁMETRO 0.4")
AR<-arima.sim(model = list(AR=0.8),n=50)
acf(AR, main="AUTOCORRELOGRAMA DE AR(1) CON PARÁMETRO 0.8")
acf2(AR)
?arima
?arima.sim
AR<-arima.sim(model = list(ar=0.4),n=50)
AR_ARIMA<-arima(AR, order=c(1,0,0))
predict(AR_ARIMA)
MA<-arima.sim(model = list(ma=0.4),n=50)
```

```
MA_ARIMA<-arima(AR, order=c(0,0,1))
predict(MA_ARIMA)
acf(AR, main="AUTOCORRELOGRAMA DE AR(1) CON PARÁMETRO 0.4")
acf(MA, main="AUTOCORRELOGRAMA DE MA(1) CON PARÁMETRO 0.4")
plot(PASEO.ALEATORIO1)
sarima.for(PASEO.ALEATORIO1,0,1,2,2,1,0,4)
library(astsa)
library(xts)
detach("package:zoo", unload = TRUE)
sarima.for(PASEO.ALEATORIO1,0,1,2,2,1,0,4)
?sarima.for
sarima.for(PASEO.ALEATORIO1,6,0,1,2,2,1,0,4)
sarima.for(gnp,6,0,1,2)
sarima.for(gnp,20,0,1,2)
gnpult<-sarima.for(gnp,20,0,1,2)
plot(tail(gnpult,40))
ts.plot(tail(gnpult,40))
print(gnpult)
sarima(gnp,0,1,2)
library(astsa)
library(xts)
detach("package:zoo", unload = TRUE)
library(zoo)
acf2(AR)
acf2(AR, main = "Correlogramas de la serie simulada AR(1)", lag.max=12)
acf2(AR, main = "Correlogramas de la serie simulada AR(1)")
acf2(AR, main = "Correlogramas de la serie simulada AR(1)",max.lag=12)
acf2(MA, main = "Correlogramas de la serie simulada MA(1)",max.lag=12)
MA<-arima.sim(model = list(ma=0.75),n=50)
acf2(MA, main = "Correlogramas de la serie simulada MA(1)",max.lag=12)
```



```
MA<-arima.sim(model = list(ma=0.75),n=500)
acf2(MA, main = "Correlogramas de la serie simulada MA(1)",max.lag=12)
AR<-arima.sim(model = list(ar=0.75),n=500)
acf2(AR, main = "Correlogramas de la serie simulada AR(1)",max.lag=12)
ARMA<-arima.sim(model = list(ar=0.75,ma=0.75),n=500)
acf2(ARMA, main = "Correlogramas de la serie simulada ARMA(1)",max.lag=12)
View(MA_ARIMA)
pi()
x<-pi
MA<-arima.sim((model=list(ma=0.4),n=50))
MA<-arima.sim(model=list(ma=0.4),n=50)
MA_ARIMA<-arima(MA, order = c(0,0,1))
predict(MA_ARIMA)
library(xts)
detach("package:zoo", unload = TRUE)
library(astsa)
library(zoo)
detach("package:zoo", unload = TRUE)
plot(unemp)
plot(tail(unemp, n=50))
plot(tail(unemp, n=50), type="l")
plot(tail(diff(unemp), n=50), type="l")
class(unemp)
tsplot(tail(diff(unemp), n=50), type="l")
?unemp
library(zoo)
tsplot(tail(diff(diff(unemp),s=12), n=50), type="l")
tsplot(tail(diff(diff(unemp),s=11), n=50), type="l")
?UnempRate
tsplot(tail(diff(UnempRate), n=50), type="l")
```

```
ts.plot(tail(unemp, n=50))
ts.plot(tail(diff(UnempRate), n=50), type="l")
ts.plot(tail(diff(diff(UnempRate),s=12), n=50), type="l")
ts.plot(tail(diff(UnempRate), n=50), type="l", main= "50 DATOS DE LA TASA DE DESEMPLEO ESTADOUNIDENSE ENTRE 2012 Y 2017, CON TENDENCIA LINEAL ELIMINADA")
ts.plot(tail(diff(UnempRate), n=50), type="l", main= "50 DATOS DE LA TASA DE DESEMPLEO ESTADOUNIDENSE ENTRE 2012 Y 2017 CON TENDENCIA LINEAL ELIMINADA")
ts.plot(tail(diff(diff(UnempRate),s=12), n=50), type="l", main= "50 DATOS DE LA TASA DE DESEMPLEO ESTADOUNIDENSE ENTRE 2012 Y 2017 CON TENDENCIA LINEAL Y ESTACIONAL ELIMINADA")
plot(globtemp)
plot(globtempl)
plot(unemp)
plot(log(unemp))
plot(unemp, main="NÚMERO DE DESEMPLEADOS ESTADOUNIDENSES DESDE 1950 HASTA 1980")
plot(log(unemp), main="NÚMERO DE DESEMPLEADOS ESTADOUNIDENSES DESDE 1950 HASTA 1980")
plot(log(unemp), main="NÚMERO DE DESEMPLEADOS ESTADOUNIDENSES DESDE 1950 HASTA 1980 CON TRANSFORMACIÓN LOGARÍTMICA")
sarima(gnp,1,0,1)
sarima(gnp,p=1,d=0,q=1,P=0,D=0,Q=0)
sarima(diff(gnp),p=1,d=0,q=1,P=0,D=0,Q=0)
acf2((globtemp))
acf2(globtemp, main="CORRELOGRAMAS DE LA SERIE DE TEMPERATURAS GLOBALES")
acf2(globtemp, main="CORRELOGRAMAS DE LA SERIE TEMPORAL DE TEMPERATURAS GLOBALES MEDIAS")
?sarima.sim
ESTACIONAL.PURA<-arima.sim(model=list(order=c(0,0,0,1,0,1,12)),n=500)
plot(birth)
plot(birth, main="NACIIMIENTOS EN ESTADOS UNIDOS CADA MES")
plot(birth, main="NACIIMIENTOS EN ESTADOS UNIDOS CADA MES")
acf2(birth)
```

acf2(birth)

acf2(diff(birth))

acf2(diff(diff(birth),s=12))

acf2(diff(diff(birth),lag=12))


```
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric", "numeric", "numeric",  
"numeric"))  
View(PIB)  
View(PIB)  
PIB[1,]  
PIB[,1]  
PIB.TS<-ts(PIB)  
View(PIB.TS)  
View(PIB.TS)  
View(PIB.TS)  
plot(ts)  
plot(PIB.TS)  
vectortiempos<-c(PIB[,1:97])  
View(vectortiempos)  
str(vectortiempos)  
mivector<-unlist(vectortiempos)  
print(mivector)  
plot(mivector)  
PIB.TS<-ts(mivector,start = 1995,frequency = 4)
```

```
plot(PIB.TS,main="EVOLUCIÓN DEL PIB DE LA CAPV")
plot(PIB.TS,main="EVOLUCIÓN DEL PIB DE LA CAPV EN MILES DE EUROS")
VECTORDEPRECIOS<-c(PIB[,1:97])
str(VECTORDEPRECIOS)
mivector<-unlist(VECTORDEPRECIOS)
PIB.TS<-ts(mivector,start = 1995,frequency = 4)
class(PIB.TS)
print(PIB.TS)
plot(PIB.TS,main="EVOLUCIÓN DEL PIB DE LA CAPV EN MILES DE EUROS")
install.packages("forecasting")
install.packages("forecast")
library(assertthat)
library(astsa)
library(backports)
library(cellranger)
library(cli)
library(colorspace)
library(forecast)
detach("package:cellranger", unload = TRUE)
detach("package:assertthat", unload = TRUE)
detach("package:backports", unload = TRUE)
detach("package:cli", unload = TRUE)
detach("package:colorspace", unload = TRUE)
library(xts)
auto.arima(PIB.TS)
sarima(PIB.TS,0,1,2,1,1,0,4)
sarima.for(PIB.TS,10,0,1,2,1,1,0,4)
DPIB<-diff(PIB.TS)
plot(DPIB, main="SERIE DEL PIB DE LA CAPV DIFERENCIADA")
LPIB<-log(PIB.TS)
```

```
DLPIB<-diff(LPIB)
plot(DLPIB, main="SERIE DEL PIB DE LA CAPV CON TRANSFORMACIÓN LOGARÍTMICA
DIFERENCIADA")
DD4PIB<-diff(DPIB,s=4)
plot(DD4PIB, main="SERIE DEL PIB DE LA CAPV DIFERENCIADA")
DD4PIB<-diff(DPIB,4)
plot(DD4PIB, main="SERIE DEL PIB DE LA CAPV DIFERENCIADA")
DPIB<-diff(PIB.TS)
DD4PIB<-diff(DPIB,s=4)
plot(DD4PIB, main="SERIE DEL PIB DE LA CAPV DIFERENCIADA UNA VEZ EN LAG=1 Y UNA VEZ
EN LAG =4")
DPIB<-diff(PIB.TS)
DD4PIB<-diff(DPIB,4)
plot(DD4PIB, main="SERIE DEL PIB DE LA CAPV DIFERENCIADA UNA VEZ EN LAG=1 Y UNA VEZ
EN LAG =4")
acf2(DD4PIB)
library(astsa)
library(forecast)
library(zoo)
library(xts)
auto.arima(DD4PIB)
sarima(DD4PIB,0,0,2,1,0,1,4)
sarima(DD4PIB,0,0,2,1,0,1,4)
sarima(DD4PIB,1,0,2,1,0,1,4)
sarima(DD4PIB,0,0,2,1,0,1,4)$AIC
sarima(DD4PIB,0,0,2,1,0,1,4)$BIC
sarima(DD4PIB,1,0,2,1,0,1,4)$AIC
sarima(DD4PIB,1,0,2,1,0,1,4)$BIC
sarima(DD4PIB,0,0,3,1,0,1,4)$AIC
sarima(DD4PIB,0,0,3,1,0,1,4)$BIC
sarima(DD4PIB,0,0,1,1,0,1,4)$AIC
```

```
sarima(DD4PIB,0,0,1,1,0,1,4)$BIC
```

```
sarima(DD4PIB,0,0,2,1,0,0,4)$AIC
```

```
sarima(DD4PIB,0,0,2,1,0,0,4)$BIC
```

```
sarima(PIB.TS,0,1,2,1,1,0,4)
```

```
sarima.for(PIB.TS,n.ahead = 8,0,1,2,1,1,0,4)
```

```
?sarima.for
```

```
sarima.for(PIB.TS,n.ahead = 8,0,1,2,1,1,0,4)
```

```
sarima.for(PIB.TS,n.ahead = 8,0,1,2,1,1,0,4,plot.all = FALSE)
```

- **S&P 500 MENSUAL:**

```
library(readxl)
```

```
SP500MENS <- read_excel("F:/Escritorio/SP500MENS.xlsx",
```

```
col_names = FALSE)
```

```
View(SP500MENS)
```

```
str(SP500MENS)
```

```
vectorSP500<-c(SP500MENS[1:100,])
```

```
View(vectorSP500)
```

```
SP500<-unlist(vectorSP500)
```

```
print(SP500)
```

```
SP500.TS<-ts(SP500, start= 03/2011,end=07/2019, frequency = 12)
```

```
?ts
```

```
vnc
```

```
SP500S<-c(1,2,SP500[1:100])
```

```
SP500.TS<-ts(SP500S,start = 2010,frequency=12)
```

```
print(SP500.TS)
```

```
SP500.TS<-SP500.TS[3:102]
```

```
print(SP500.TS)
```

```
SP500.TS<-ts(SP500S,start = 2010,frequency=12)
```



```
SP500.TS<-tail(SP500.TS, n=100)
print(SP500.TS)
sp500.ts<-ts(SP500, start = c(2011,3), frequency = 12)
print(sp500.ts)
load("~/SP500MENS/SP500MENS.Rproj")
load("~/SP500MENS/.Rhistory")
sp500mes
SP.TS
library(readxl)
SP500MENS <- read_excel("F:/Escritorio/SP500MENS.xlsx",
col_names = FALSE)
SP500MENS <- read_excel("F:/Escritorio/SP500MENS.xlsx",
col_names = FALSE)
library(readxl)
SP500MENS <- read_excel("F:/Escritorio/SP500MENS.xlsx",
col_names = FALSE)
View(SP500MENS)
str(SP500MENS)
vectorSP500<-c(SP500MENS[1:100,])
View(vectorSP500)
SP500<-unlist(vectorSP500)
print(SP500)
SP500.TS<-ts(SP500, start= 03/2011,end=07/2019, frequency = 12)
?ts
vnc
SP500S<-c(1,2,SP500[1:100])
SP500.TS<-ts(SP500S,start = 2010,frequency=12)
print(SP500.TS)
SP500.TS<-SP500.TS[3:102]
print(SP500.TS)
```

```
SP500.TS<-ts(SP500S,start = 2010,frequency=12)
SP500.TS<-tail(SP500.TS, n=100)
print(SP500.TS)
sp500.ts<-ts(SP500, start = c(2011,3), frequency = 12)
print(sp500.ts)
SP500[1:100]<-SP500[100:1]
SP.TS<-ts(SP500, start = c(2011,3), frequency = 12)
SP90<-SP500[1:90]
SP.TS90<-ts(SP90, start = c(2011,3), frequency = 12)
auto.arima(SP90)
library(astsa)
library(zoo)
library(xts)
library(forecast)
auto.arima(SP90)
sarima(SP90,0,1,0)
sarima(SP90,0,1,0)$AIC
sarima(SP90,0,1,0)$BIC
sarima(SP90,1,1,0)$AIC
sarima(SP90,1,1,0)$BIC
sarima(SP90,0,2,0)$AIC
sarima(SP90,0,2,0)$BIC
sarima(SP90,0,1,1)$AIC
sarima(SP90,0,1,1)$BIC
save.image("~/SP500MENS/SP500MENS.Rproj.RData")
?lines
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10) lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)+ lines(SP.TS)
```

```
lines(SP.TS)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
source('~/.active-rstudio-document')
sarima.for(SP90,0,1,0,n.ahead = 10)
sarima.for(SP90,0,1,0,n.ahead = 10)
sarima.for(SP90,0,1,0,n.ahead = 10)
sarima.for(SP90,0,1,0,n.ahead = 10)
PRED<-sarima.for(SP90,0,1,0,n.ahead = 10)
View(PRED)
plot(PRED)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
sarima.for(SP90,0,1,0,n.ahead = 10)
lines(SP.TS)
?sarima.for
?par
sarima.for(SP90,0,1,0,n.ahead = 10) + lines(SP.TS)
```

```
fun1<-function(){sarima.for(SP90,0,1,0,n.ahead = 10)}
```

```
fun2<-function(){lines(SP.TS)}
```

```
wrap_function <- function() {
```

```
  fun1()
```

```
  fun2()
```

```
}
```

```
wrap_function()
```

```
wrap_function()
```

```
View(wrap_function)
```

```
View(wrap_function)
```

```
View(wrap_function)
```

```
View(wrap_function)
```

```
?sarima.for
```

```
sarima.for(SP90,0,1,0,n.ahead = 10)
```

```
lines(SP.TS)
```

```
sarima.for(SP90,0,1,0,n.ahead = 10)
```

```
lines(SP500)
```

```
sarima.for(SP90,0,1,0,n.ahead = 10)
```

```
lines(SP500, col="blu")
```

```
?lines
```

```
sarima.for(SP90,0,1,0,n.ahead = 10)
```

```
lines(SP500, col.lab="blu")
```

```
sarima.for(SP90,0,1,0,n.ahead = 10)
```

```
lines(SP500, col.lab="red")
```

- **S&P 500 Semanal:**

```
library(readxl)
```

```
SP500SEM <- read_excel("F:/Escritorio/SP500SEM.xlsx",
```

```
col_names = FALSE)
```

```
View(SP500SEM)
LISTSP<-SP500SEM[1:100,]
SP500<-unlist(SP500SEM)
SP500<-SP500[100:1]
SP.TS<-ts(SP500, start=2016, frequency=51)
print(SP.TS)
plot(SP.TS, main= "VALOR DE CIERRE DE PRINCIPIO DE SEMANA DESDE 2016 HASTA 2018 DEL
S&P500")
SP90<-SP500[1:90]
SP.TS90<-ts(SP90, start=2016, frequency=51)
acf2(diff(SP90))
library(astsa)
library(xts)
library(forecast)
acf2(diff(SP90))
sarima(SP.TS90,0,1,0)
auto.aima(SP.TS90)
auto.aRima(SP.TS90)
auto.arima(SP.TS90)
sarima(SP.TS90,1,1,0)
sarima(SP.TS90,0,1,0)
sarima(SP.TS90,0,1,0)$AIC
sarima(SP.TS90,0,1,0)$BIC
sarima(SP.TS90,1,1,0)$AIC
sarima(SP.TS90,1,1,0)$BIC
sarima(SP.TS90,0,2,0)$AIC
sarima(SP.TS90,0,2,0)$BIC
sarima(SP.TS90,0,1,1)$AIC
sarima(SP.TS90,0,1,1)$BIC
sarima.for(SP.TS90, n.ahead=10,0,1,0)
```

```
sarima.for(SP.TS90, n.ahead=10,0,1,0)
```

```
lines(SP500)
```

```
sarima.for(SP.TS90, n.ahead=10,0,1,0)
```

```
lines(SP500)
```

```
sarima.for(SP.90, n.ahead=10,0,1,0)
```

```
lines(SP500)
```

```
sarima.for(SP90, n.ahead=10,0,1,0)
```

```
lines(SP500)
```

```
sarima.for(SP90, n.ahead=10,0,1,0)
```

```
lines(SP500)
```

- **S&P 500 DIARIO:**

```
library(readxl)
```

```
SP500DIAR <- read_excel("F:/Escritorio/SP500DIAR.xlsx",
```

```
col_names = FALSE)
```

```
View(SP500DIAR)
```

```
LISTSP<-SP500DIAR[1:100,]
```

```
SP500<-unlist(LISTSP)
```

```
SP500<-SP500[100:1]
```

```
SP.TS<-ts(SP500,start = 2019,frequency = 365)
```

```
print(SP.TS)
```

```
plot(SP.TS,main="VALOR DE CIERRE DIARIO DESDE EL 1 DE ENERO HASTA FINALES DE MAYO  
DEL S&P500")
```

```
plot(SP.TS,main="VALOR DE CIERRE DIARIO DESDE EL 1 DE ENERO HASTA FINALES DE MAYO  
DE 2019 DEL S&P500")
```

```
SP90<-SP500[1:100]
```

```
SP90<-SP500[1:90]
```

```
SP.TS90<-ts(SP90,start = 2019,frequency=365)
```

```
library(astsa)
```

```
library(forecast)
```

```
library(zoo)
library(xts)
acf2(SP.TS90)
acf2(SP.TS90, max.lag = 40)
acf2(diff(SP.TS90))
acf2(diff(SP.TS90),max.lag = 40)
auto.arima(SP.TS90)
sarima(SP.TS90,2,1,0)
sarima(SP.TS90,0,1,0)
sarima(SP.TS90,0,1,0)$AIC
sarima(SP90,0,1,0)$BIC
sarima(SP90,1,1,0)$AIC
sarima(SP90,1,1,0)$BIC
sarima(SP90,0,2,0)$AIC
sarima(SP,TS90,0,2,0)$BIC
sarima(SP.TS90,0,1,1)$AIC
sarima(SP.TS90,0,1,1)$BIC
sarima.for(SP.TS90,n.ahead = 10,0,1,0)
sarima.for(SP90,n.ahead = 10,0,1,0)
lines(SP500)
```

▪ **IBEX 35 MENSUAL:**

```
library(astsa)
library(xts)
detach("package:zoo", unload = TRUE)
library(zoo)
library(forecast)
library(readxl)
IBEXMENS <- read_excel("F:/Escritorio/IBEXMENS.xlsx",
```

```
col_names = FALSE)
View(IBEXMENS)
IBEX<-IBEXMENS[1:100,]
IBEXVECTOR<-unlist(IBEX)
IBEXVECTOR<-IBEXVECTOR[100:1]
IB.TS<- ts(IBEXVECTOR, start = c(2011,3), frequency = 12)
str(IB.TS)
print(IB.TS)
IB.TS<- ts(IBEXVECTOR, start = c(2011,4), frequency = 12)
print((IB.TS))
print((IB.TS))
plot(IB.TS, main="VALOR DE CIERRE A PRINCIPIO DE MES DESDE 2011 HASTA 2019 DEL IBEX
35")
IB90<-IBEXVECTOR[1:90]
IB.TS90<-ts(IB90, start = c(2011,4), frequency=12)
acf2(IB.TS90)
acf2(diff((IB.TS90))
acf2(diff(IB.TS90))
auto.arima(IB.TS90)
sarima(IB.TS90,0,1,0)
sarima(IB.TS90,0,1,0)$AIC
sarima(IB.TS90,0,1,0)$BIC
sarima(IB.TS90,1,1,0)$AIC
sarima(IB.TS90,1,1,0)$BIC
sarima(IB.TS90,0,2,0)$AIC
sarima(IB.TS90,0,2,0)$BIC
sarima(IB.TS90,0,1,1)$AIC
sarima(IB.TS90,0,1,1)$BIC
sarima.for(IB90,n.ahead = 10,0,1,0)
lines(IBEXVECTOR)
```



```
save.image("~/IBEXMENS/ibexmenss.RData")
load("~/IBEXMENS/ibexmenss.RData")
library(astsa)
library(xts)
detach("package:zoo", unload = TRUE)
library(zoo)
library(forecast)
library(readxl)
IBEXMENS <- read_excel("F:/Escritorio/IBEXMENS.xlsx",
col_names = FALSE)
View(IBEXMENS)
IBEX<-IBEXMENS[1:100,]
IBEXVECTOR<-unlist(IBEX)
IBEXVECTOR<-IBEXVECTOR[100:1]
IB.TS<- ts(IBEXVECTOR, start = c(2011,3), frequency = 12)
str(IB.TS)
print(IB.TS)
IB.TS<- ts(IBEXVECTOR, start = c(2011,4), frequency = 12)
print((IB.TS))
print((IB.TS))
plot(IB.TS, main="VALOR DE CIERRE A PRINCIPIO DE MES DESDE 2011 HASTA 2019 DEL IBEX
35")
IB90<-IBEXVECTOR[1:90]
IB.TS90<-ts(IB90, start = c(2011,4), frequency=12)
acf2(IB.TS90)
acf2(diff((IB.TS90))
acf2(diff(IB.TS90))
auto.arima(IB.TS90)
sarima(IB.TS90,0,1,0)
sarima(IB.TS90,0,1,0)$AIC
```

```
sarima(IB.TS90,0,1,0)$BIC
sarima(IB.TS90,1,1,0)$AIC
sarima(IB.TS90,1,1,0)$BIC
sarima(IB.TS90,0,2,0)$AIC
sarima(IB.TS90,0,2,0)$BIC
sarima(IB.TS90,0,1,1)$AIC
sarima(IB.TS90,0,1,1)$BIC
sarima.for(IB90,n.ahead = 10,0,1,0)
lines(IBEXVECTOR)
```

▪ **IBEX SEMANAL:**

```
library(astsa)
library(zoo)
library(xts)
library(forecast)
library(readxl)
IBEXSEM <- read_excel("F:/Escritorio/IBEXSEM.xlsx",
col_names = FALSE)
View(IBEXSEM)
View(IBEXSEM)
IBEX<-IBEXSEM[1:100,]
IBEXVECTOR<-unlist(IBEX)
IBEXVECTOR<-IBEXVECTOR[100:1]
IB.TS<- ts(IBEXVECTOR, start = c(2016), frequency = 51)
str(IB.TS)
print(IB.TS)
IB.TS<- ts(IBEXVECTOR, start = c(2016), frequency = 51)
print((IB.TS))
print((IB.TS))
```

```
plot(IB.TS, main="VALOR DE CIERRE A PRINCIPIO DE SEMANA DESDE 2016 HASTA 2019 DEL
IBEX 35")
IB90<-IBEXVECTOR[1:90]
IB.TS90<-ts(IB90, start = c(2016), frequency=51)
acf2(IB.TS90)
acf2(diff((IB.TS90))
p
acf2(diff(IB.TS90))
acf2(diff(IB.TS90),max.lag = 25)
auto.arima(IB.TS90)
sarima(IB.TS90,0,1,0)
sarima(IB.TS90,0,1,0)$AIC
sarima(IB.TS90,0,1,0)$BIC
sarima(IB.TS90,1,1,0)$AIC
sarima(IB.TS90,1,1,0)$BIC
sarima(IB.TS90,0,2,0)$AIC
sarima(IB.TS90,0,2,0)$BIC
sarima(IB.TS90,0,1,1)$AIC
sarima(IB.TS90,0,1,1)$BIC
sarima.for(IB90,n.ahead = 10,0,1,0)
lines(IBEXVECTOR)
?auto.arima
load("~/IBEXSEM/.RData")
library(astsa)
library(zoo)
library(xts)
library(forecast)
library(readxl)
IBEXSEM <- read_excel("F:/Escritorio/IBEXSEM.xlsx",
col_names = FALSE)
```

```
View(IBEXSEM)
View(IBEXSEM)
IBEX<-IBEXSEM[1:100,]
IBEXVECTOR<-unlist(IBEX)
IBEXVECTOR<-IBEXVECTOR[100:1]
IB.TS<- ts(IBEXVECTOR, start = c(2016), frequency = 51)
str(IB.TS)
print(IB.TS)
IB.TS<- ts(IBEXVECTOR, start = c(2016), frequency = 51)
print((IB.TS))
print((IB.TS))
plot(IB.TS, main="VALOR DE CIERRE A PRINCIPIO DE SEMANA DESDE 2016 HASTA 2019 DEL
IBEX 35")
IB90<-IBEXVECTOR[1:90]
IB.TS90<-ts(IB90, start = c(2016), frequency=51)
acf2(IB.TS90)
acf2(diff((IB.TS90))
p
acf2(diff(IB.TS90))
acf2(diff(IB.TS90),max.lag = 25)
auto.arima(IB.TS90)
sarima(IB.TS90,0,1,0)
sarima.for(IB90,n.ahead = 10,0,1,0)
lines(IBEXVECTOR)
?auto.arima
  ■ IBEX DIARIO:
library(readxl)
IBEXSEM <- read_excel("F:/Escritorio/IBEXSEM.xlsx",
col_names = FALSE)
View(IBEXSEM)
```

```
library(readxl)
IBEXDIAR <- read_excel("F:/Escritorio/IBEXDIAR.xlsx",
col_names = FALSE)
View(IBEXDIAR)
View(IBEXSEM)
View(IBEXSEM)
View(IBEXSEM)
View(IBEXSEM)
library(readxl)
IBEXDIAR <- read_excel("F:/Escritorio/IBEXDIAR.xlsx",
col_names = FALSE)
View(IBEXDIAR)
IBEX<-IBEXDIAR[1:100,]
IBEXVECTOR<-unlist(IBEX)
IBEXVECTOR<-IBEXVECTOR[100:1]
IB.TS<- ts(IBEXVECTOR, start = c(2019), frequency = 365)
str(IB.TS)
print(IB.TS)
IB.TS<- ts(IBEXVECTOR, start = c(2016), frequency = 365)
print((IB.TS))
print((IB.TS))
plot(IB.TS, main="VALOR DE CIERRE DIARIO EN 2019 DEL IBEX 35")
IB90<-IBEXVECTOR[1:90]
IB.TS90<-ts(IB90, start = c(2019), frequency=365)
IBEX<-IBEXDIAR[1:100,]
IBEXVECTOR<-unlist(IBEX)
IBEXVECTOR<-IBEXVECTOR[100:1]
IB.TS<- ts(IBEXVECTOR, start = c(2019), frequency = 365)
str(IB.TS)
print(IB.TS)
IB.TS<- ts(IBEXVECTOR, start = c(2019), frequency = 365)
```

```
print((IB.TS))
print((IB.TS))
plot(IB.TS, main="VALOR DE CIERRE DIARIO EN 2019 DEL IBEX 35")
IB90<-IBEXVECTOR[1:90]
IB.TS90<-ts(IB90, start = c(2019), frequency=365)
acf2(diff(IB.TS90),max.lag = 25)
library(astsa)
library(forecast)
library(xts)
detach("package:zoo", unload = TRUE)
library(zoo)
acf2(diff(IB.TS90),max.lag = 25)
print(IB.TS90)
plot(IB.TS90)
plot(diff(IB.TS90))
auto.arima(IB.TS90)
acf2(diff(IB.TS90),max.lag = 25)
sarima(IB.TS90,0,1,0)
sarima(IB.TS90,0,1,0)$AIC
sarima(IB.TS90,0,1,0)$BIC
sarima(IB.TS90,1,1,0)$AIC
sarima(IB.TS90,1,1,0)$BIC
sarima(IB.TS90,0,2,0)$AIC
sarima(IB.TS90,0,2,0)$BIC
sarima(IB.TS90,0,1,1)$AIC
sarima(IB.TS90,0,1,1)$BIC
sarima.for(IB90,n.ahead = 10,0,1,0)
lines(IBEXVECTOR)
sarima.for(IB90,n.ahead = 10,0,1,0)
lines(IBEXVECTOR)
```

sarima.for(IB90,n.ahead = 10,0,1,0)

lines(IBEXVECTOR)

sarima.for(IB90,n.ahead = 10,0,1,0)

lines(IBEXVECTOR)

sarima.for(IB90,n.ahead = 10,0,1,0)

lines(IBEXVECTOR)