Trabajo autónomo I: Series Temporales

Máster DATCOM

Serie Temporales y Minería de Flujo de Datos

Autor
Alberto Armijo Ruiz
armijoalb@correo.ugr.es

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

—

23 de abril de 2019

Índice general

1. Parte Práctica **5**
1.1. Ejecución de los ficheros de código 5
1.2. Predicción diaria 5
1.3. Predicción mensual 16

2. Parte Teórica **22**
2.1. Preprocesamiento 22
2.2. Análisis de tendencia y estacionalidad 22
2.3. Estacionariedad 23
2.4. Modelado de series temporales 24

Índice de figuras

1.1. Transformación de datos del dataset 6
1.2. Imputación de valores en la serie 6
1.3. Serie temporal datos diarios 7
1.4. Comparación entre serie filtrada y serie normal 7
1.5. Descomposición de la serie 8
1.6. Separación en train y test 9
1.7. Serie sin estacionalidad 9
1.8. Test Dickey-Fuller y ACF de la serie 10
1.9. Test Dickey-Fuller y ACF de la serie diferenciada 10
1.10. Gráficas ACF y PACF 11
1.11. Resultados del modelo AR1 12
1.12. Resultados del modelo MA4 13
1.13. Resultados del modelo ARMA 14
1.14. Comparación modelos 15
1.15. Predicción temperatura máxima para la primera semana de marzo 16
1.16. Serie temperatura media 17
1.17. Descomposición serie media 17
1.18. Conjuntos de test y train 18
1.19. Serie sin estacionalidad 18
1.20. Estacionariedad y ACF serie 19
1.21. Gráficas ACF y PACF serie medias 19
1.22. Resultados modelo MA2 20
1.23. Predicción serie temperatura media 21

Índice de cuadros

Capítulo 1

Parte Práctica

En esta parte se describirá el trabajo realizado para predecir la temperatura máxima durante la próxima semana y durante los dos siguientes meses dependiendo del experimento. El dataset que se ha utilizado corresponde con la estación meteorológica de Reus, Tarragona.

1.1. Ejecución de los ficheros de código

Para la ejecución de los ficheros de código, basta con abrir dichos ficheros en RStudio y pulsar el botón source o la combinación de teclas Ctrl+Alt+R. Para ver los resultados de la ejecución, se debe mirar la terminal que contiene RStudio y el apartado *Plots* que hay a la derecha del IDE para poder mirar las gráficas.

1.2. Predicción diaria

Para este apartado se pide realizar una predicción de la siguiente semana después del último dato del dataset correspondiente. Lo primero que se va a hacer es cargar el dataset y transformar las variables a sus para que sean enteros, fechas, etc... Si nos fijamos en los datos de la serie, el primer dato que contiene es 7 de mayo de 2013 y el último del 28 de febrero de 2018; por lo que el dataset contiene casi 5 años de datos.

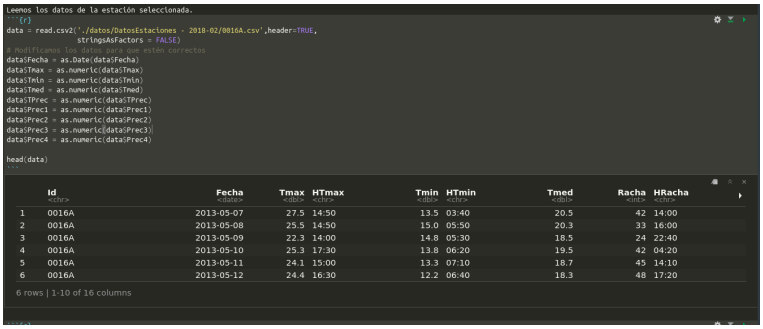


Figura 1.1: Transformación de datos del dataset

Una vez tenemos los datos cargados y transformados, se debe comprobar si hay datos pérdidos en la columna que vamos a estudiar. Para ello, se selecciona dicha columna junto con la fecha de la medición y se utilizará el método *nic* de la librería *mice* para comprobar si existen datos perdidos; si existen, se imputarán con el método *amelia* de la libería *Amelia*.

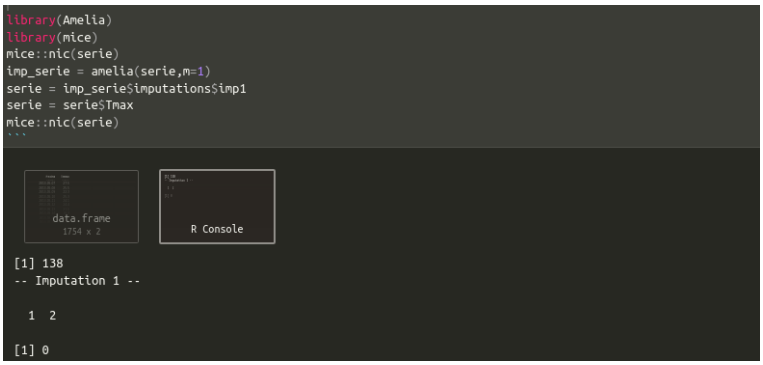


Figura 1.2: Imputación de valores en la serie

Ahora, podemos pasar a estudiar la serie al completo, para ello se debe crear un objeto de la clase *ts*; a este objeto se le debe pasar el conjunto de datos de la serie y la frecuencia. Para nuestro caso, la frecuencia se ha estimado como 365, ya que tenemos datos diarios de varios años. Lo siguiente será visualizar la serie temporal y poder ver que forma tiene.

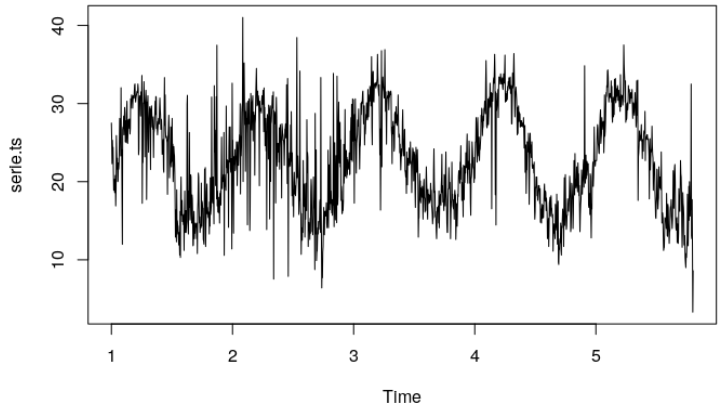
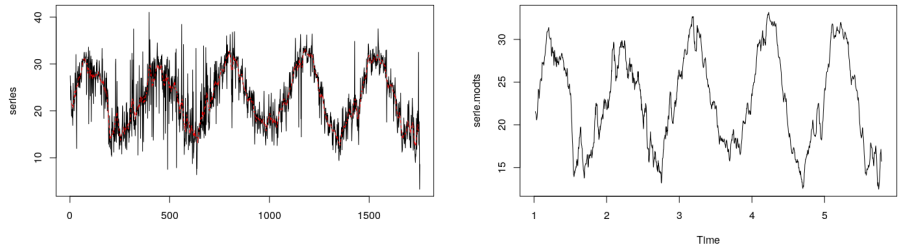


Figura 1.3: Serie temporal datos diarios

Como se puede ver, la serie tiene mucha variación entre unos días y otros; esto puede hacer que el estudio de modelos de la serie se vea afectado por esta gran variación. Por ello, se va a filtrar la serie haciendo por cada día la media de los quince días de alrededor (la temperatura del día, la de los siete días anteriores y la de los siete siguientes); de esta forma, se debería conseguir una serie mucho más suave y sin tantos picos. Para ello se utilizará un filtro de convolución sobre la serie. Los resultados son los siguientes.



(a) Serie y filtrado (b) Serie filtrada

Figura 1.4: Comparación entre serie filtrada y serie normal

Como se puede ver, esta serie tiene muchos menos picos que la serie normal y será mucho más fácil de analizar. Al haber hecho el filtro sobre la serie, se han perdido algunos valores al principio y al final de la serie, por ello, cuando se vaya a hacer la predicción de la primera semana de marzo, se deberá también predecir estos datos perdidos.

Lo siguiente que se va hacer es realizar un estudio sobre las componentes de la serie, para ello se va a utilizar una descomposición de la serie en sus diferentes componentes y se analizará que tipo de componentes están presentes en la serie y se eliminarán para convertir la serie en estacionaria. La descomposición de la serie es la siguiente.

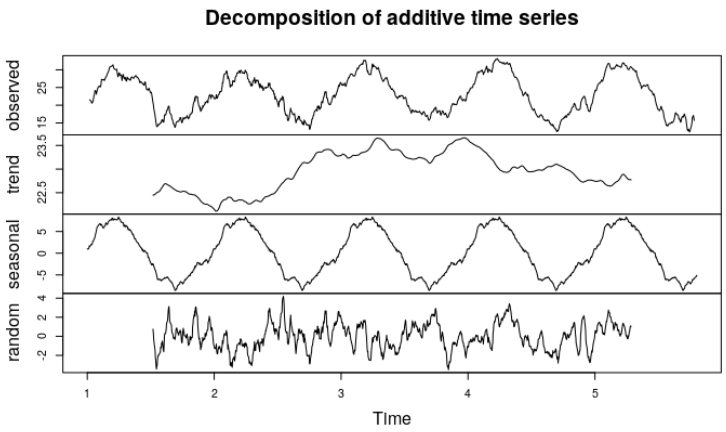


Figura 1.5: Descomposición de la serie

Por lo que se puede ver en la descomposición de la serie, hay una componente estacional clara dentro de los datos; esto es normal en una serie donde se mide la temperatura ya que hay que tener en cuenta las estaciones del año. En la componente de tendencia puede parecer a simple vista que existe una tendencia positiva, pero si nos fijamos en el rango de esta, se puede ver que la serie solamente varia en un grado y medio durante toda la serie; además a partir de la mitad de la serie parece que vuelve a bajar; por lo tanto, se considerará que realmente no tiene una componente de tendencia y no será necesario estudiarla. Lo siguiente que debemos hacer es separar los datos de la serie en un conjunto de validación y en otro de entrenamiento; utilizaremos el conjunto de entrenamiento para modelar la estacionalidad y eliminarla del conjunto de datos y para crear un modelo de predicción, con el conjunto de validación comprobaremos la calidad del modelo de predicción. Para realizar la partición entre train y test, se va a utilizar un conjunto de 4 años para train y el resto de la serie para test. La separación entre train y test quedaría de la siguiente forma.

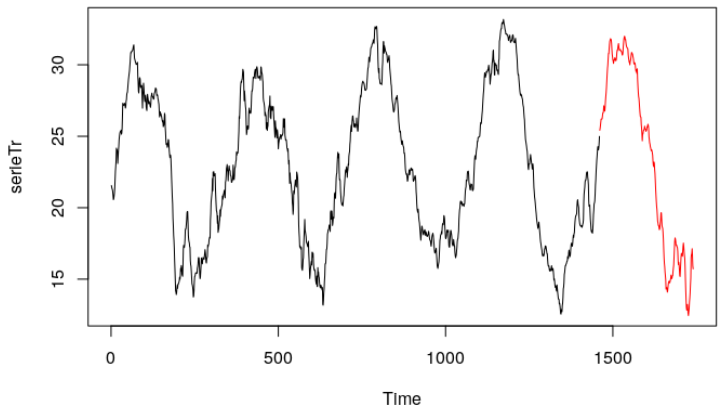


Figura 1.6: Separación en train y test

Lo siguiente que vamos a hacer es eliminar la componente estacional de la serie, para ello, obtendremos la componente estacional de la serie mediante la función *decompose*; de los datos que nos devuelve esta función se pueden utilizarán los de un año, para los datos se test se replicará para adaptarse al tamaño de train y para los datos de test se utilizarán solamente los necesarios para el tamaño de test; tras esto se le restará la componente a cada uno de los conjuntos. El resultado es el siguiente.

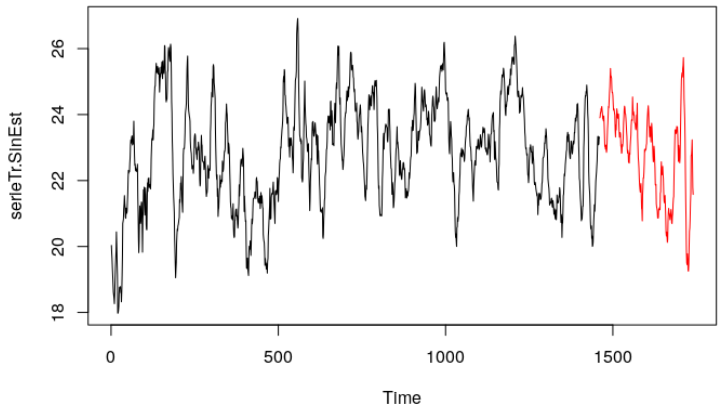
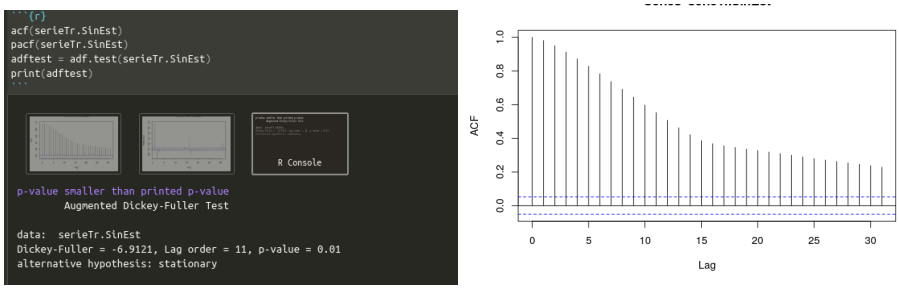


Figura 1.7: Serie sin estacionalidad

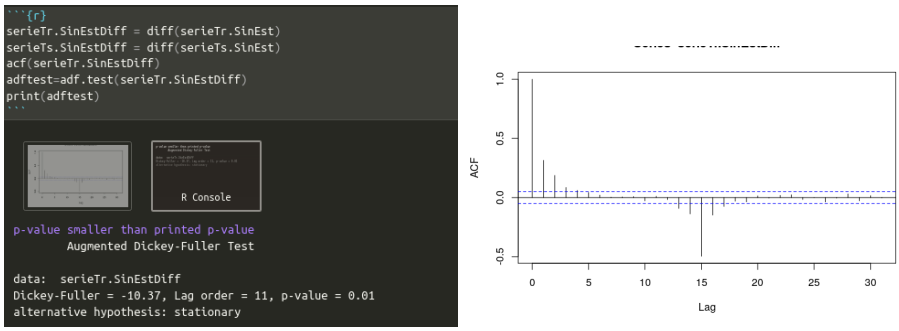
A continuación, tenemos que comprobar si la serie es estacionaria, para ello, se utilizará el test de *Dickey-Fuller* y la gráfica *ACF* de la serie. Si la serie es estacionaria, debe de pasar el test (obtener un p-valor menor que 0.05) y la gráfica debería descender rápida a 0. Los resultados son los siguientes.



(a) Resultados test (b) Gráfica ACF

Figura 1.8: Test Dickey-Fuller y ACF de la serie

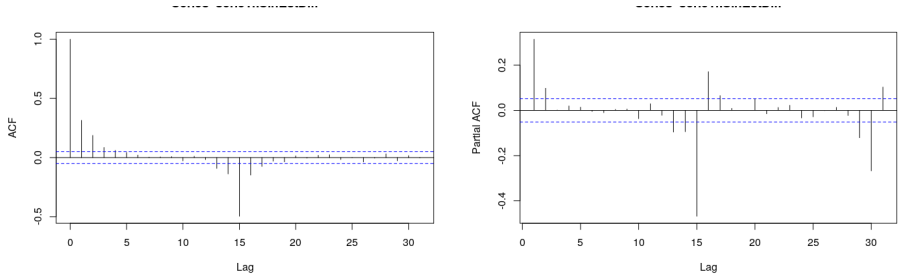
Como se puede ver, la serie pasa el test, sin embargo, la gráfica ACF no desciende a 0, por lo tanto no es estacionaria. Para conseguir que sea estacionaria, se diferenciará la seria el número de veces que sea necesario hasta que sea estacionaria, para nuestra serie, con una diferenciación ya ha sido suficiente; estos son los resultados.



(a) Resultados test (b) Gráfica ACF

Figura 1.9: Test Dickey-Fuller y ACF de la serie diferenciada

Ahora, la gráfica sí desciende rápidamente a 0 y pasa el test, por lo que estacionaria. Lo siguiente que vamos a hacer es analizar la gráfica ACF y la gráfica $PACF$ para saber que tipo de modelo se puede utilizar.

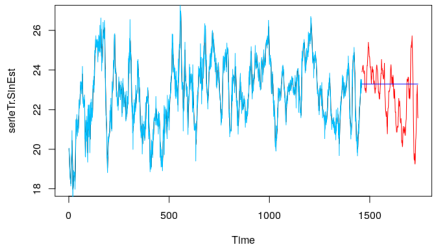


(a) Gráfica ACF (b) Gráfica PACF

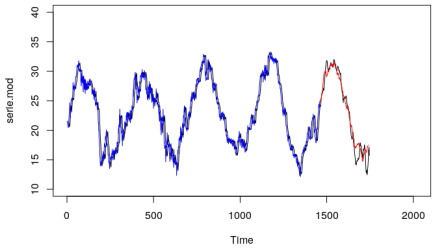
Figura 1.10: Gráficas ACF y PACF

Por lo que se puede ver en las gráficas, podría ser un modelo AR(1), un modelo MA(4), o una combinación de ambos. Se crearán los tres modelos y se compararán para ver cuál de ellos es mejor. Para crear los modelos, se utilizará el modelo *ARIMA* indicándole el tipo de modelo; tras esto, se hará una gráfica sobre el ajuste que tiene sobre los datos y la predicción sobre el conjunto de validación. También se utilizarán diferentes test para saber que los modelos obtenidos son correctos.

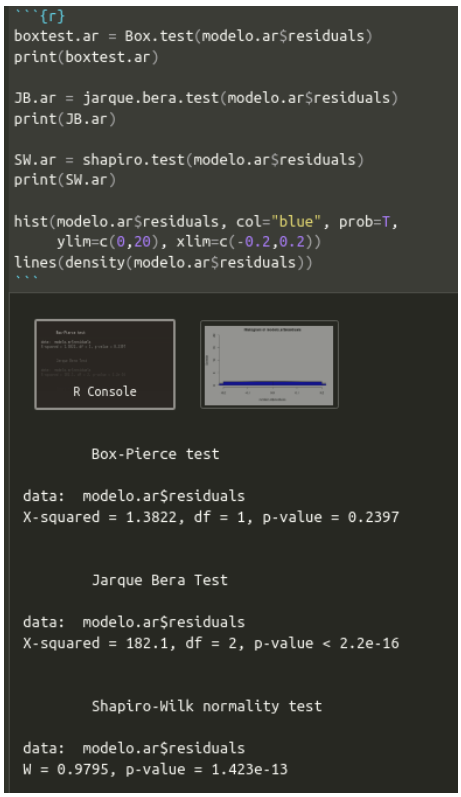
Para el primer modelo, estos son los resultados obtenidos por los test y el ajuste que tiene con los datos.



(a) Ajuste sin estacionalidad



(b) Ajuste con estacionalidad

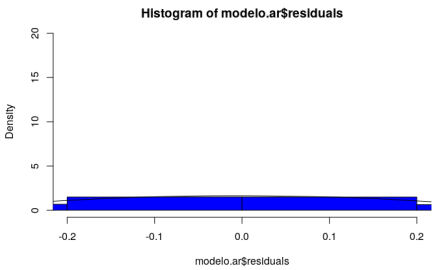


```r
boxtest.ar = Box.test(modelo.ar$residuals)
print(boxtest.ar)

JB.ar = jarque.bera.test(modelo.ar$residuals)
print(JB.ar)

SW.ar = shapiro.test(modelo.ar$residuals)
print(SW.ar)

hist(modelo.ar$residuals, col="blue", prob=T,
     ylim=c(0,20), xlim=c(-0.2,0.2))
lines(density(modelo.ar$residuals))
```

R Console

```
        Box-Pierce test

data:  modelo.ar$residuals
X-squared = 1.3822, df = 1, p-value = 0.2397


        Jarque Bera Test

data:  modelo.ar$residuals
X-squared = 182.1, df = 2, p-value < 2.2e-16


        Shapiro-Wilk normality test

data:  modelo.ar$residuals
W = 0.9795, p-value = 1.423e-13
```

(c) Tests AR1

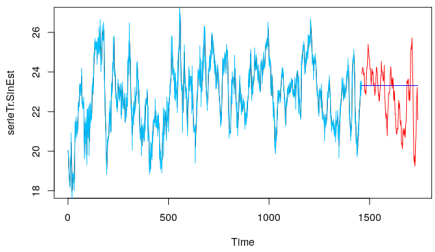


(d) Histograma residuos AR1

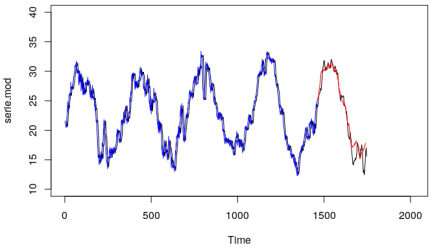
Figura 1.11: Resultados del modelo AR1

Por lo que se puede ver, el modelo generado con AR(1) obtiene un buen ajuste. Los test *Jarque-Bera* y *Shapiro-Wilk* sirven para medir si los residuos del modelo tienen una distribución normal, como el p-valor en ambos es menor que 0.05, pasa el test. El otro test, el test de *Box-Pierce* sirve para saber si los residuos del modelo son aleatorios o no; en este caso es necesario que el p-valor sea mayor que 0.05, ya que sino estaríamos comprobando que los residuos siguen una distribución no aleatoria, y eso significaría que el modelo está sobreajustándose a los datos, cosa que no queremos. Veamos ahora los

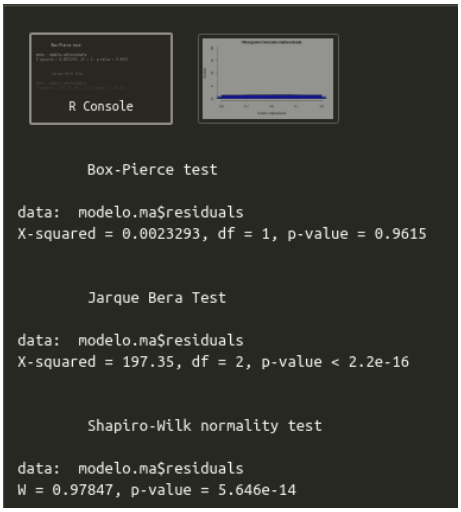
resultados obtenidos por los otros dos modelos.



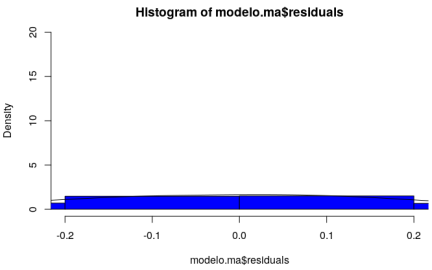
(a) Ajuste sin estacionalidad



(b) Ajuste con estacionalidad

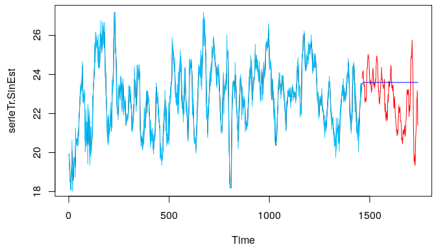


(c) Tests MA4

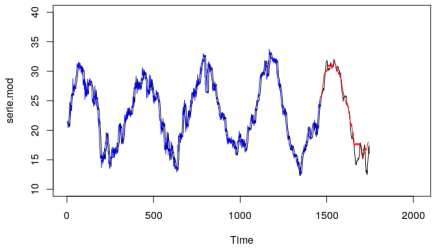


(d) Histograma residuos MA4

Figura 1.12: Resultados del modelo MA4



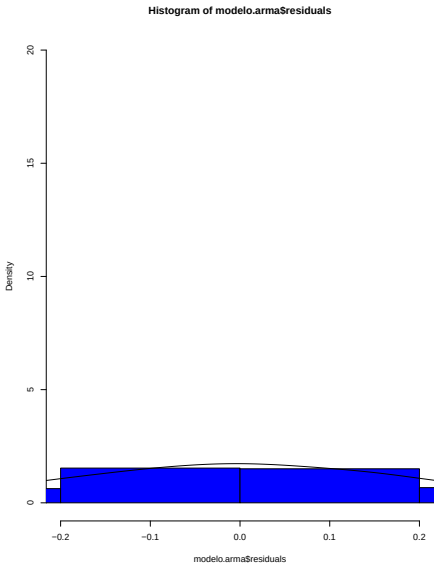
(a) Ajuste sin estacionalidad



(b) Ajuste con estacionalidad

```
        Box-Pierce test

data:  modelo.arma$residuals
X-squared = 0.00021592, df = 1, p-value = 0.9883


        Jarque Bera Test

data:  modelo.arma$residuals
X-squared = 184.01, df = 2, p-value < 2.2e-16


        Shapiro-Wilk normality test

data:  modelo.arma$residuals
W = 0.97997, p-value = 2.197e-13
```

(c) Tests ARMA



(d) Histograma residuos ARMA

Figura 1.13: Resultados del modelo ARMA

Como se puede ver, los tres modelos pasan los test y los ajustes son muy parecidos; para elegir a uno de los tres para realizar la predicción, utilizaremos el criterio de AIC y el error producido en test de cada uno de los algoritmos. El error y el resultado del criterio de AIC son los siguientes.

(a) Resultados AIC (b) Error en test

Figura 1.14: Comparación modelos

Por lo que se vé en la salida del criterio de AIC, los modelos son muy parecidos, el que peores resultados obtiene es el modelo AR(1), aunque este es el más sencillo; los modelos MA(4) y ARMA(1,4) son muy parecidos. Si nos fijamos en el error producido por cada uno de ellos, el que produce el error más pequeño es el modelo AR(1), mientras que los otros modelos obtiene errores mayores conforme se aumenta la complejidad. Por ello, elegiremos el modelo AR(1) para realizar la predicción.

Para realizar la predicción de la siguiente semana, se utilizará la serie entera (con el filtrado). Lo primero será eliminar la estacionalidad, una vez eliminada se construye un modelo AR(1) con una diferenciación. Tras esto se obtiene los valores ajustados y la predicción (recordar, hay que predecir 15 días por el filtrado utilizado). Una vez obtenidos la predicción y el ajuste, se vuelve a añadir la estacionalidad y se muestra la predicción obtenida por el modelo. La predicción es la siguiente.

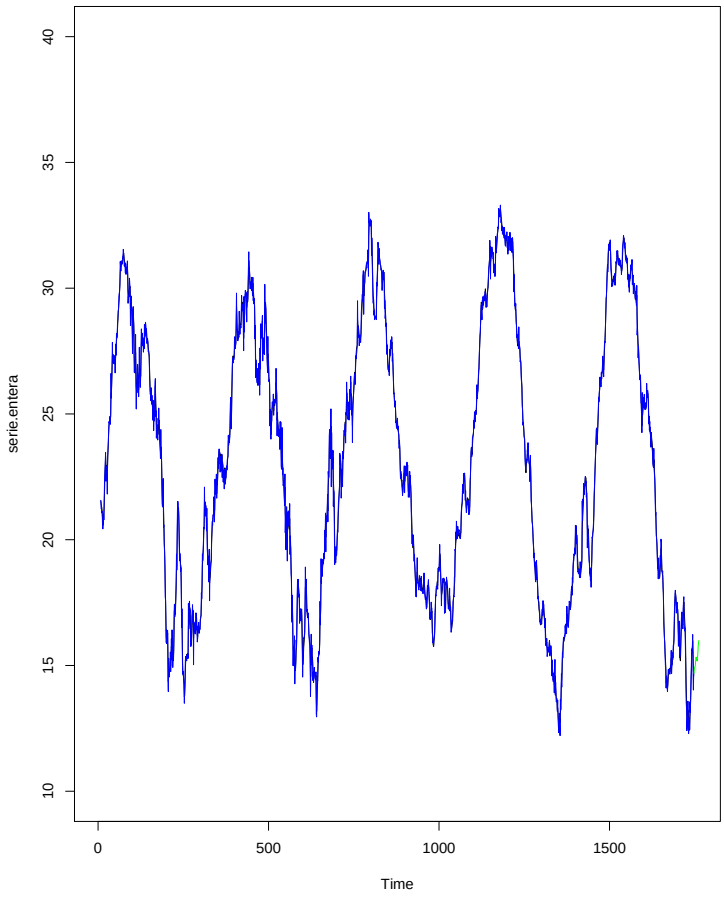


Figura 1.15: Predicción temperatura máxima para la primera semana de marzo

En la esquina derecha de la imagen, se puede ver en verde la predicción del modelo; la cual parece acertada, ya que normalmente a partir de marzo las temperaturas empiezan a incrementar.

1.3. Predicción mensual

Para la predicción mensual se seguirá los mismos pasos que para el estudio de la serie con datos diarios. Para obtener la temperatura media por mes, se utilizará la función *group_by*, *summarise* y pipelines de la librería *dplyr*, antes se leeran los datos y se imputarán. La serie resultante es la siguiente. Una vez se va a crear la serie, se debe especificar la frecuencia como 12, ya que estamos tratando con meses.

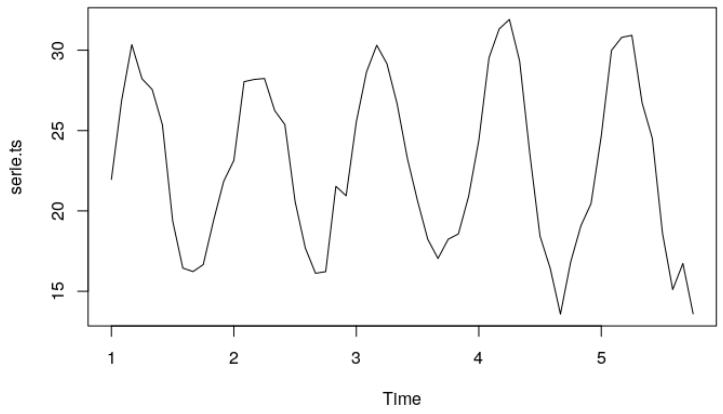


Figura 1.16: Serie temperatura media

La descomposición de la serie es la siguiente.

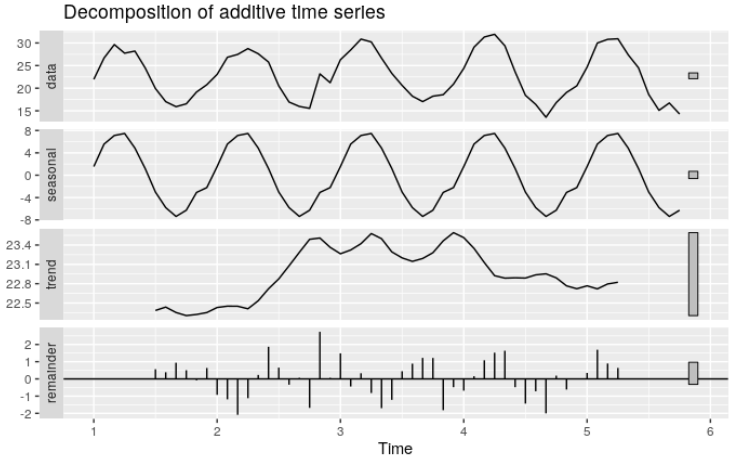


Figura 1.17: Descomposición serie media

Al igual que antes, no hay una componente estacional, y sí que hay una componente estacional, la cual tendremos que eliminar para poder comprobar si la serie es estacionaria. Antes de eliminar la estacionalidad, se va a separar la serie en train y test; para train se utilizarán 4 años y el resto para test. La separación en train y test es la que se muestra a continuación.

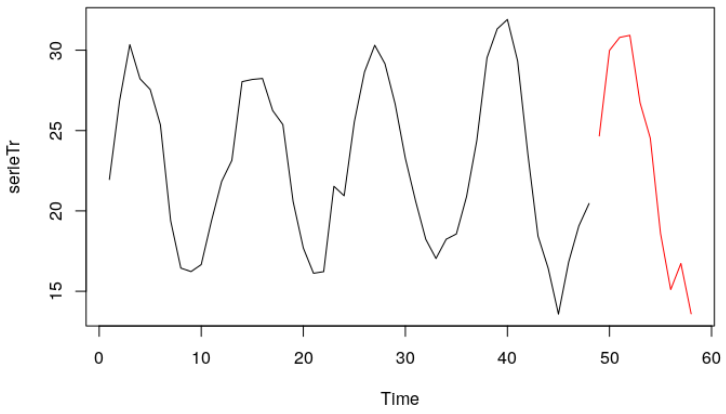


Figura 1.18: Conjuntos de test y train

Para eliminar la estacionalidad, se debe hacer lo mismo que en el ejercicio anterior; obtener la componente estacional y restar a cada conjunto dicha componente. La serie sin estacionalidad sería la siguiente.

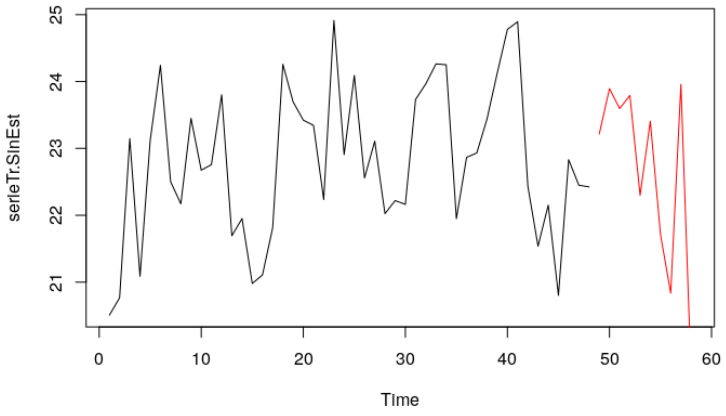
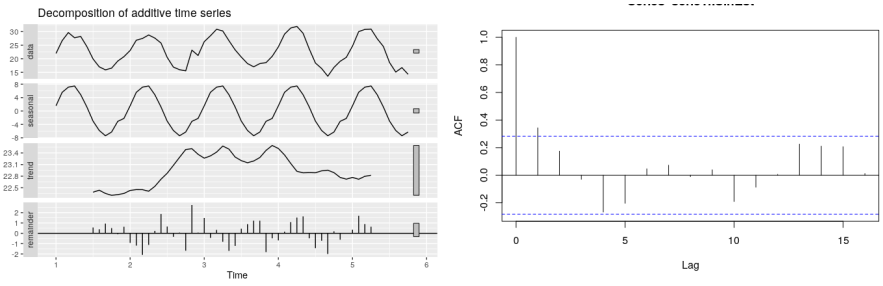


Figura 1.19: Serie sin estacionalidad

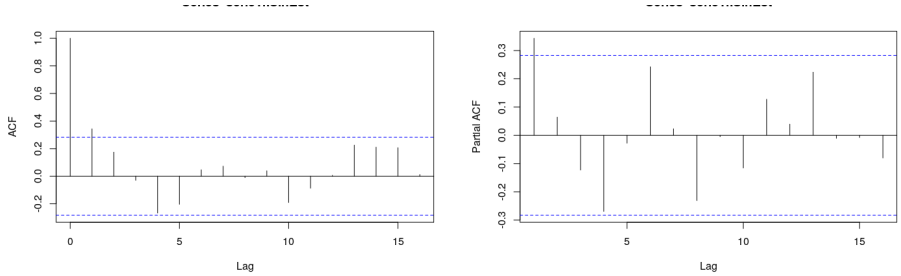
Ahora comprobaremos si la serie es estacionaria, para ello se utilizará el test de *Dickey-Fuller* y la gráfica *ACF*. Si pasa el test y la gráfica desciende rápidamente a 0 sabemos que la serie es estacionaria.



(a) Test estacionaria (b) Gráfica ACF

Figura 1.20: Estacionariedad y ACF serie

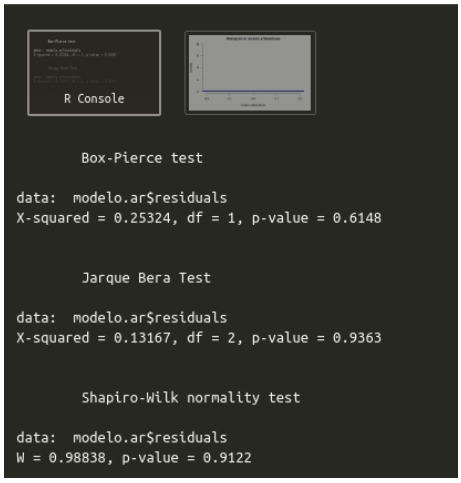
Como se puede ver, pasa el test y la gráfica desciende rápidamente a 0, por lo que es estacionaria. Lo siguiente será decidir que modelo utilizar, para ello se utilizará tanto la gráfica *ACF* como la gráfica *PACF*.



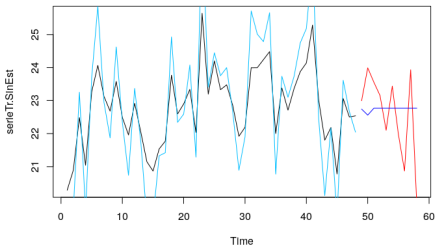
(a) Gráfica ACF (b) Gráfica PACF

Figura 1.21: Gráficas ACF y PACF serie medias

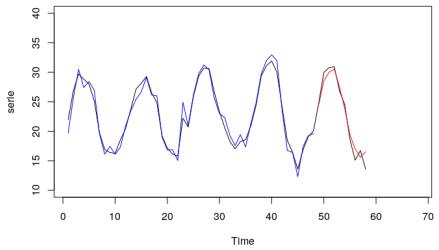
Por lo que se puede ver en las gráfica, parece que claramente se debe utilizar un modelo MA(2). Para crear el modelo se utilizará *ARIMA*, tras esto, se comprobará el ajuste del modelo con los datos de train y test y con los test de *Shapiro*, *Jarque-Bera* y *Box-Pierce*.



(a) Test MA2



(b) Ajuste MA2



(c) Ajuste MA2 con estacionalidad

Figura 1.22: Resultados modelo MA2

Por lo que se puede ver, el ajuste parece bueno. Aunque no pase los test de normalidad, sí que pasa el test de *Box-Pierce*. Posiblemente no pase los test de normalidad porque tenemos bastantes pocos datos para train y este tipo de test suelen necesitar bastantes datos para ser fiables. Por ello, utilizaremos este modelo para realizar la predicción de los dos siguientes meses. Al igual que en el ejercicio anterior, ahora se utilizará la serie entera, se eliminará la estacionalidad y se obtendrá un modelo MA(2) entrenados con esos datos, una vez obtenidos las predicciones se volverá a añadir la estacionalidad y se comentarán los resultados.

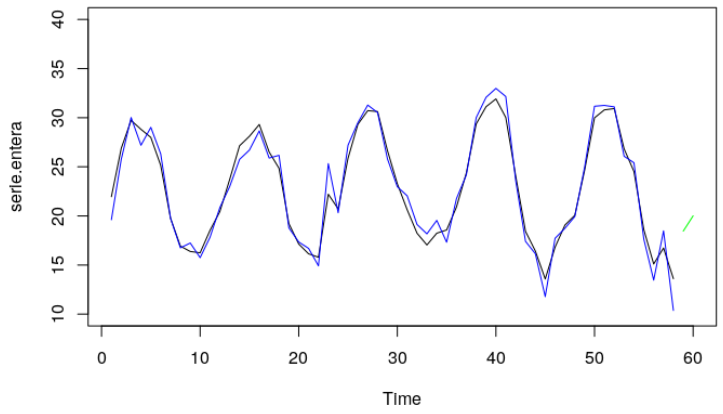


Figura 1.23: Predicción serie temperatura media

Los resultados obtenidos en la predicción tienen bastante sentido, ya que es normal que la temperatura media en estos meses continúe aumentando, además si nos fijamos en meses iguales de otros años, los valores son muy parecidos a los obtenidos por la predicción; por lo que podemos concluir que el modelo obtenido consigue un buen ajuste para predecir la temperatura.

Capítulo 2

Parte Teórica

En este apartado se describirán los conceptos teóricos utilizados en la práctica.

2.1. Preprocesamiento

El preprocesamiento utilizado en la práctica ha sido la imputación de valores; este tipo de preprocesamiento se utiliza en todo tipo de problemas de minería de datos. La imputación de valores consiste en reemplazar valores pérdidos en los datos por datos coherentes.

Otro tipo de preprocesamiento utilizado en la primera parte de la práctica ha sido el filtrado de series temporales; para ello se ha utilizado un filtro de convolución que realiza la media de los quince días más cercanos a cada día; de esta forma se puede conseguir una serie más limpia sin picos que puedan ser considerados ruidosos. El filtrado en series temporales también se puede utilizar para detectar ciclos o tendencias dentro de series con estacionalidad.

2.2. Análisis de tendencia y estacionalidad

Para el análisis de tendencia y estacionalidad se ha utilizado descomposición de series temporales. Existen varios métodos de descomposición de series temporales; como por ejemplo la descomposición mediante *Medias Móviles* (*Moving Averages* en inglés, *MA*) o descomposición *STL*.

La descomposición de series temporales consiste en la separación de los datos de una serie temporal en tres componentes: tendencia, estacionalidad y residuos. La tendencia expresa incrementos o decrementos durante largos periodos de tiempos, estos incrementos/decrementos no tienen porque se de tipo lineal. La componente estacional muestra cambios en los datos afectados por un patrón estacional como por ejemplo las estaciones del año. La componente residual expresa el resto de la serie temporal al eliminar los valores de las dos componentes anteriores. Existen dos tipos de descomposiciones: aditiva

y multiplicativa.

Para realizar una descomposición mediante *Medias Móviles* se debe seguir el siguiente proceso: utilizar el método de *Medias Móviles* para calcular la componentes de tendencia y calcular la serie sin tendencia restando/dividiendo el resultado del algoritmo de *Medias Móviles*. Tras esto, se calcula la componente estacional como la media de los valores de un ciclo dado y después ajustada a 0. Por último, se calcula la componente residual como la resta/división de la suma/multiplicación de las otras dos componentes.

Aparte de la descomposición de series temporales, existen otras formas de estudiar la tendencia de una serie; como por ejemplo el filtrado (mediante *Medias Móviles* por ejemplo) o la estimación funcional, es decir, estimar la tendencia mediante una función, bien lineal o no lineal.

2.3. Estacionariedad

La estacionariedad es una característica de algunas series temporales para las cuales sus propiedades no dependen del momento en el que se observa, esto significa que la varianza en los datos es constante. Para detectar la estacionariedad en las series se suele utilizar el gráfico ACF; ya que las series estacionarias suelen tener gráficos ACF que descienden rápidamente a 0.

La gráfica ACF es un gráfico que representa la autocorrelación entre los distintos estados de la serie, es decir, muestra la importación de los estados anteriores de la serie con el estado actual y nos indica hasta que estado tiene importancia los valores de la serie para predecir estados siguientes.

Con la gráfica ACF no es suficiente para determinar si una serie es estacionaria, ya que series con estacionalidad pueden mostrar también gráficas ACF que descienden rápidamente a 0. Por ello, es necesario también utilizar un test que nos indique si la serie es estacionaria, o necesita ser diferenciada. La diferenciación entre series temporales se trata del cálculo entre observaciones sucesivas. El test que se debe utilizar para esto es el test de *Dickey-Fuller Ampliado*. Si los datos de la serie pasan este test, se sabe que la serie es estacionaria; en caso contrario, se sabe que se debe diferenciar la serie para conseguir estacionariedad.

La estacionariedad en series no estacionarias se puede obtener mediante la eliminación de sus componentes de tendencia y estacionalidad (si es que tienen) y la diferenciación de la serie.

2.4. Modelado de series temporales

Para el modelado de series temporales existen diferentes metodologías; una de la más comunes es el uso de modelos *ARIMA* con series estacionarias.

Los modelos *ARIMA* son modelos formados por la combinación de diferenciación en la serie, *modelos autoregresivos* y *modelos de medias móviles* de diferentes grados. Los *modelos autoregresivos* se calculan como una combinación lineal de p estados anteriores de la serie y un error; donde p representa el grado del modelo. Los *modelos de medias móviles* se calculan como la combinación lineal de q coeficientes en estados anteriores y sus errores asociados, donde q representa el grado del modelo. Por lo tanto, los modelos *ARIMA* pueden ser una combinación de un *modelo autoregresivo* de grado p, un *modelo de medias móviles* de grado q sobre una serie diferenciada d veces.

Para saber cuál es el grado de cada una de estas componentes es necesario utilizar los gráficos *ACF* y *PACF* para determinar el grado de los modelos y el test de *Dickey-Fuller Aumentado* para saber cuantas veces es necesario diferenciar la serie. El gráfico *PACF*, al igual que el gráfico *ACF*, muestra la correlación de los diferentes estados de una serie, pero esta vez tiene en cuenta la diferencia de espacio entre los diferentes estados.

Los modelos *AR* (autoregression), suelen tener gráficas *ACF* que descienden rápida a 0 y gráficas *PACF* que tardan más descender a 0. Los modelos *MA* (Moving Averages), tienen gráficas *PACF* que descienden rápida a 0 y gráficas *ACF* que tardan más en descender. Para algunos casos, es posible que no haya diferencias tan significativas entre ambas gráficas; en esos casos se debe probar también la combinación de ambos (modelo *ARMA*).

Una vez se ha obtenido un modelo, o modelos, se necesitan medidas para elegir el mejor modelo posible. Existen diferentes medidas, como por ejemplo el error cuadrático medio (*MSE*) o el error absoluto medio (*MAE*); el problema de estas medidas que no consideran la complejidad de los modelos, solamente consideran el error cometido por estos. Por ello, también es necesario utilizar medidas como el criterio de Akaike (*AIC*) que también tienen en cuenta la complejidad del modelo y por lo tanto nos puede mostrar cual de los modelos entrenados puede ser más interesante.