Predicción con series temporales

Ciencia de Datos en Negocio, Máster Ciencia de Datos

Kevin Craig Alisauskas

2019-04-19

Contents

1 Introducción y descripción.

1.1 Enunciado

En el archivo adjunto series.xlsx bajo tu nombre hay una serie medida mensualmente o trimestralmente para un determinado periodo temporal (la columna junto a los valores representa, en un formato no homogéneo, las fechas a las que corresponde cada uno de los valores). La mayoría de las series corresponden a la Comunidad Valenciana, hay unas pocas de España.

Además del archivo de series.xlsx puedes encontrar otros archivos xlsx, que contienen algunos regresores potenciales para las series, útiles si especificas un modelo regARIMA. En el nombre de cada uno de estos archivos se indica: * el año de inicio de la serie para los cuales los regresores del correspondiente archivo son de utilidad. * la frecuencia (mensual o trimestral) asociada a los regresores. * si deberían emplearse para series de España (ES) o de la Comunidad Valenciana.

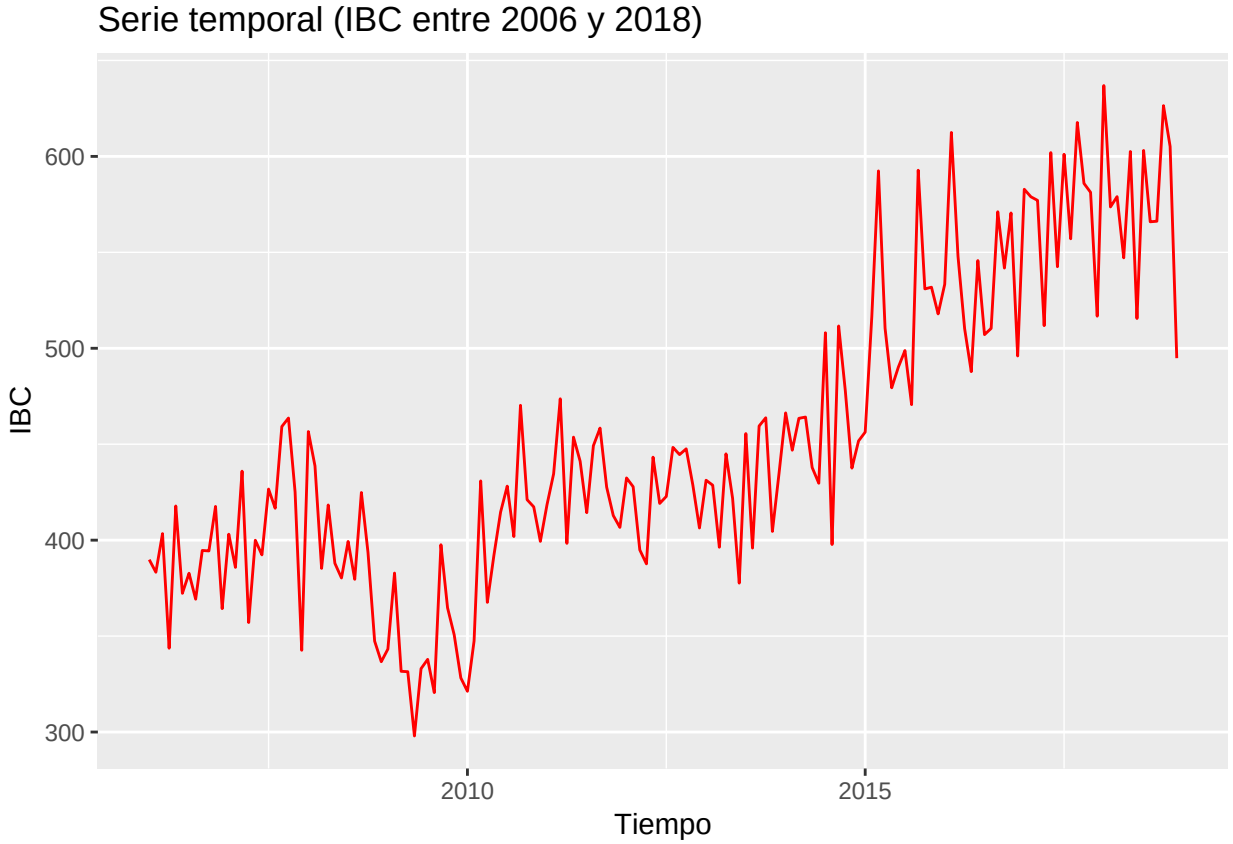
Tu objetivo en esta tarea consiste en predecir (sobre la serie original y para la serie desestacionalizada) los valores de tu serie para 2018 (con un horizonte de 1 mes/trimestre, 2 meses/trimestres, ..., 12 meses/4 trimestres) entre un conjunto de alternativas. Selecciona al menos un modelo de la clase ETS (librería forecast), un modelo de la clase regARIMA (librería seasonal) y un modelo de los considerados sencillos.

Para seleccionar modelos adecuados dentro de cada clase, utiliza los valores de la serie hasta 2017 (prediciendo secuencialmente, y para los diferentes horizontes, los valores de 2016 y 2017, utilizando exclusivamente los valores previos de la serie) y genera las predicciones para 2018 con todos ellos. Valora la calidad predictiva de las diferentes especificaciones en el conjunto de comprobación para selección y en el conjunto de test para cada uno de los horizontes. En las especificaciones regARIMA es necesario considerar, además de los regresores incluidos en los archivos suministrados, los regresores incluidos en el propio programa como la pascua móvil (Easter) o el ciclo semanal (Trading day). Refleja todos los resultados alcanzados, con una justificación adecuada de los mismos, en un informe.

2 Carga de Datos y análisis introductorio.

Nuestra serie muestra la evolución del IBC (Importaciones en Bienes de Consumo en millones de euros) entre 2006 y 2018, de forma mensual, es decir, que la periodicidad esperada será anual.

Comenzamos cargando los datos, almacenándolos como una serie temporal y realizando un plot.

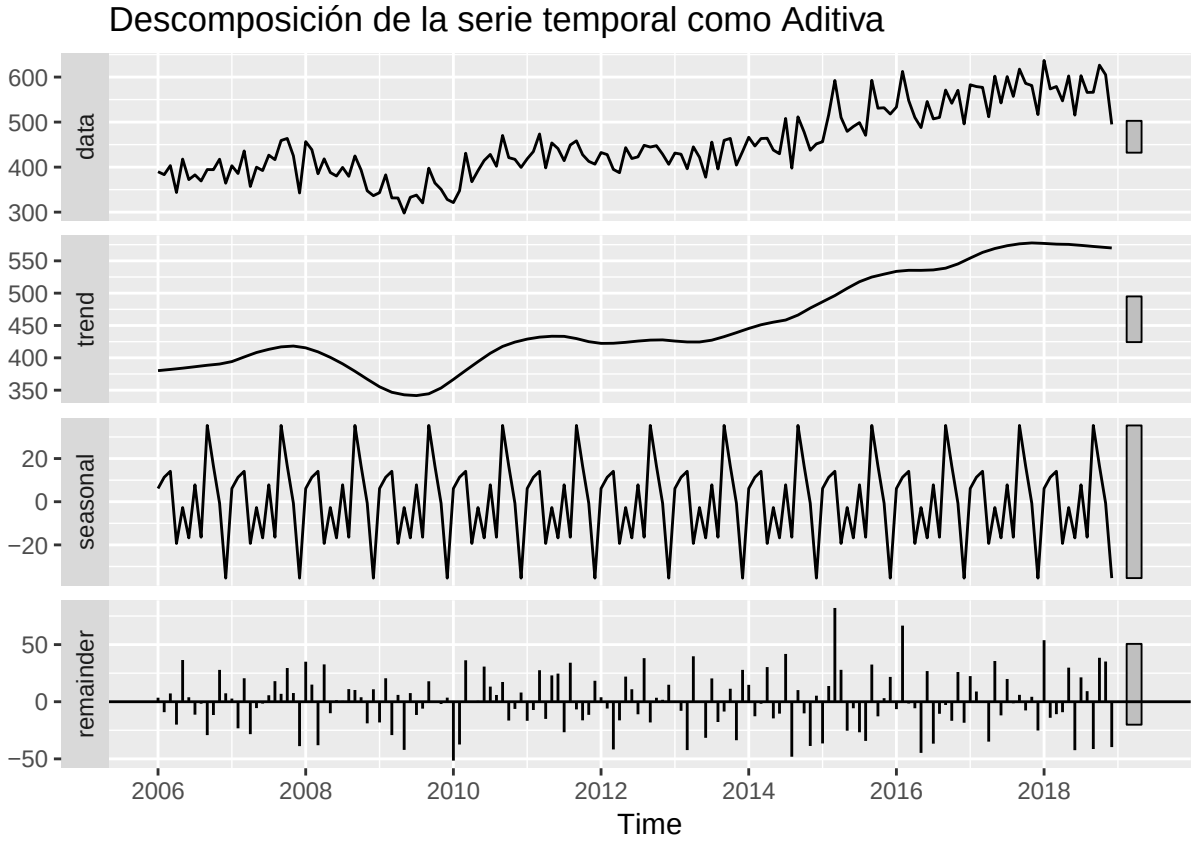


Observamos la tendencia lineal ascendente (sobretodo a partir de 2010) y una cierta estacionalidad. La varianza parece constante (no tenemos heterocedasticidad), por lo que trataremos con un esquema de tipo aditivo

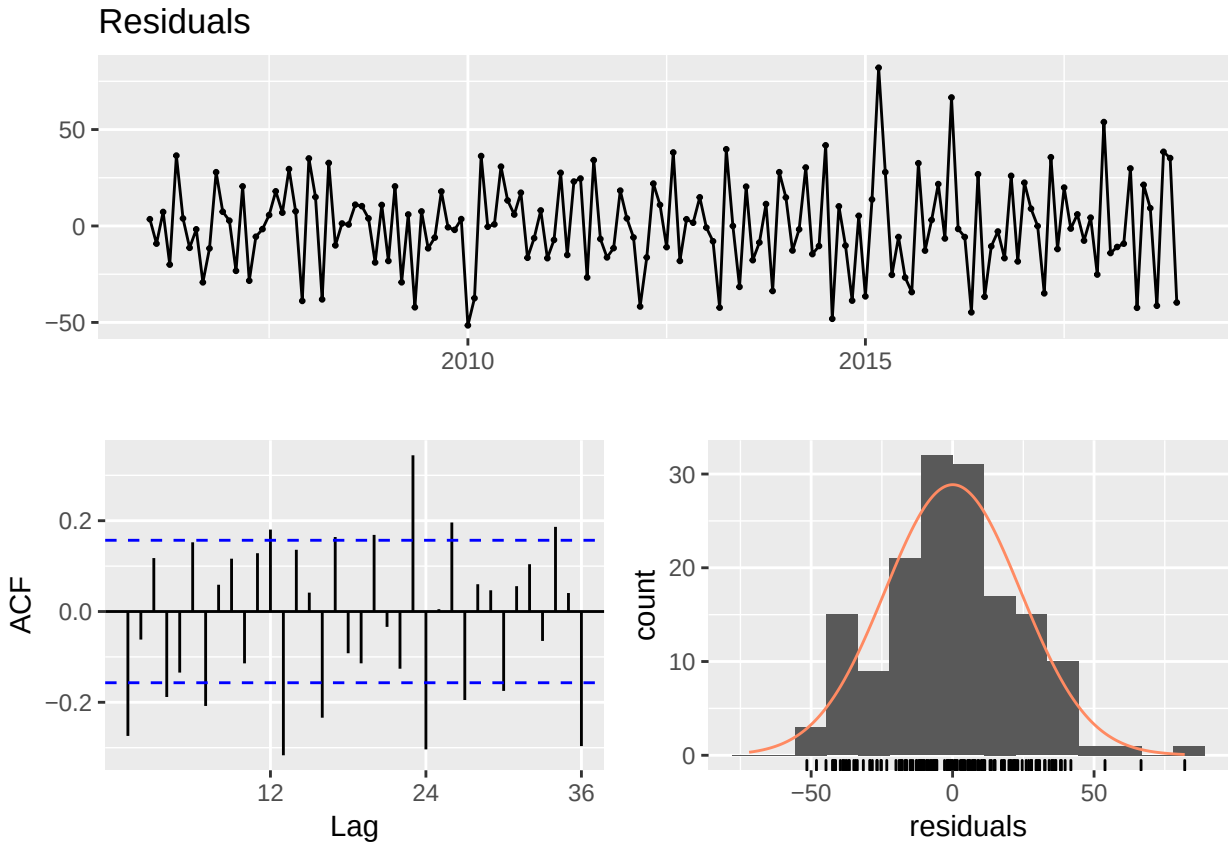
$$X_t \sim T_t + S_t + E_t$$

.

Ahora mostramos la serie descompuesta con el esquema Aditivo, mostramos la serie, la tendencia, la estacionalidad (El resultado de sustraer la tendencia a la serie original) y los residuos (resultado de sustraer la tendencia y la componente estacional).



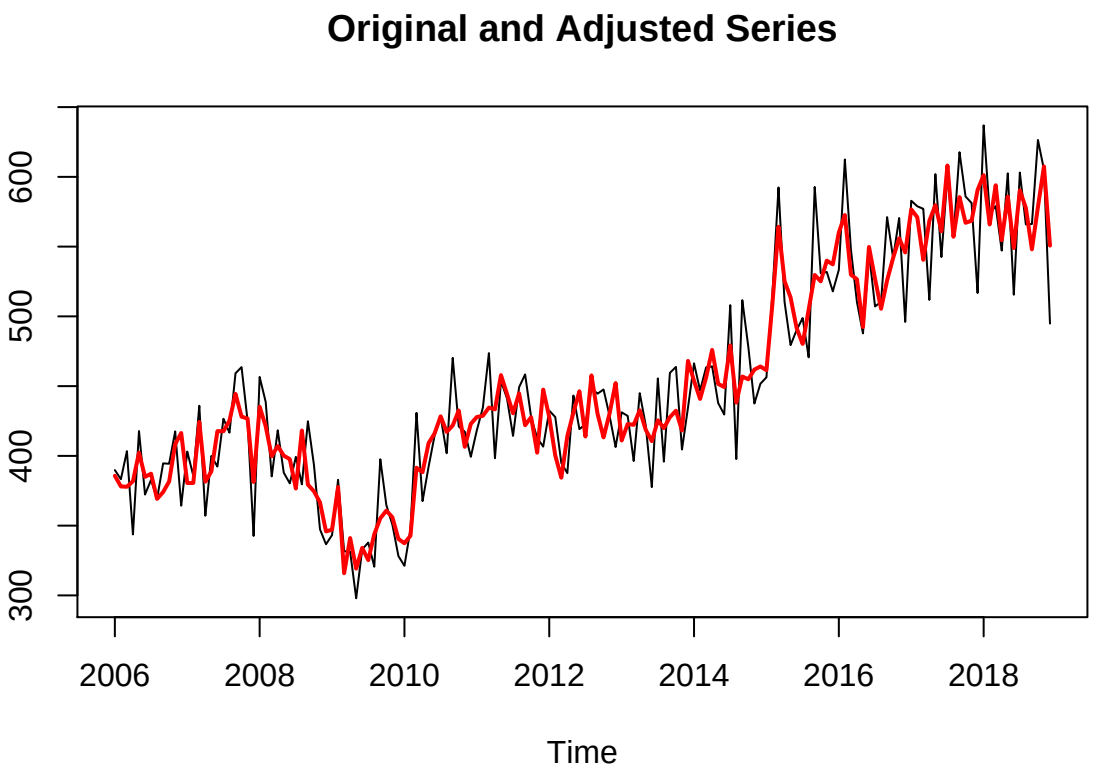
Como vemos, la tendencia tiene un marcado carácter lineal, con algunas componentes que seguramente sean sucesos particulares (especialmente la bajada en 2009, en la que se registra el nivel más bajo del PIB). La componente estacional se muestra claramente, teniendo ciclos cada 2 años (al ser tener frecuencia anual...) muy marcados. Por último, los residuos parecen aleatorios, pero vamos a realizar un análisis de ellos para verlo claramente.



Observamos en la función de autocorrelación algunos "palos significativos", es posible que corresponndan a estacionalidad sistemática (semana santa...). Por otra parte, la distribución de residuos se asemeja a una gaussiana, indicando una descomposición satisfactoria.

3 Análisis y predicción.

A la hora de analizar la serie o modelizarla, tomaremos tanto la serie original como la serie desestacionalizada. Recordamos que la serie original era *series_ts*, a la desestacionalizada la llamaremos *seas_ts*. Ahora la crearemos con el paquete *seasonal* y las compararemos.



Comprobamos que la nueva serie desestacionalizada resulta mucho más suave al quitar tanto la estacionalidad como los efectos de calendario.

3.1 Split en train, validacion y test.

Antes de empezar a predecir necesitamos dividir en conjuntos de train, validación y test. Usaremos la serie hasta 2016 como conjunto train, los años 2016 y 2017 como validación (secuencial, se irán agregando datos a train para generar predicciones y seleccionar los mejores modelos) y el año 2018 como test. Todo esto lo haremos tanto para la serie original como desestacionalizada.

Serie desestacionalizada:

```
## [1] "La longitud de la serie desestacionalizada es igual a la suma de las series train, validación y
```

Serie original:

```
## [1] "La longitud de la serie original es igual a la suma de las series train, validación y test: TRU
```

Nota: Para valorar la precisión de los métodos realizaremos una predicción secuencial con el conjunto de validación con cada modelo, realizando un total de 24 modelos (2 años) y obteniendo su suma de error cuadrático (RSS, residual sum of squares), esta técnica es llamada Hold-Out Cross Validation, en concreto Leave-One-Out, porque realizamos la predicción para un único item de validación (mes) cada vez.

3.2 Modelos sencillos.

Primero vamos crear una función para realizar el cálculo del error de Cross Validation con las funciones sencillas:

Vamos a empezar a predecir para modelos sencillos. Usaremos las dos series y los modelos naïve estacional y de deriva (drift), que son simples sin llegar a ser totalmente absurdos.

```
##   Error.snaive.desestacionalizada Error.snaive.original
## 1                         27827.8              71800.95
```

```
##   Error.deriva.desestacionlizada Error.deriva.original
## 1                       18640.93               66631.5
```

Al analizar los RSS de cada método sobre cada una de las dos series, observamos que los errores en la serie desestacoinlizada son mucho menores, lógicamente, ya que es una serie mucho mas suave y sencilla de predecir. En cuanto a la precisión, notamos que el error tanto en la serie original como en la desestacionlizada del método de la deriva es menor, por lo que nos quedaremos con ese método de cara a la comparación final.

3.3 Modelos de clase ETS.

Vamos crear una función para realizar el cálculo con la función ets:

Ahora vamos a entrar en métodos de alisado exponencial (ETS = Error, Trend, Seasonal), una forma de media móvil que tiene en cuenta cada uno de los 3 componentes en los que consideramos, se descompone una serie temporal.

A partir del análisis previo de la serie, hemos considerado que tratamos un esquema Aditivo tanto en Tendencia como en Estacionalidad, por lo que probaremos variando la clase de Error entre Aditivo y Multiplicativo, además, aunque en la práctica no tiene sentido, ya que el modelo va variando según agregamos los datos de validación, realizaremos un ajuste automático comparativo.

```
##   Error.ets.desestacionalizada Error.ets.original
## 1                     13327.29           40483.59
##   Error.ets.desestacionlizada.mult Error.ets.original.mult
## 1                         13288.38               38399.11
##   Error.ets.desestacionalizada.automatico Error.ets.original.automatico
## 1                                13575.33                      38443.26
```

Observamos los errores en la serie desestacionalizada mucho más reducidos, como en los casos sencillos. En cuanto a la serie original, observamos cierta mejora al usar un esquema multiplicativo tanto para el error como para la tendencia. También cabe notar que el ajuste automático para cada iteración no introduce ninguna mejora respecto al MMN. Así pues, nos quedamos con un esquema Multiplicativo-Multiplicativo-Dependiente (el término dependiente corresponde a Nulo en el caso de validación, ya que perdemos estacionalizad al introducir términos no múltiplos de la estacionalizad original, 12, y a Aditivo en el caso de test, ya que no tenemos heterocedasticidad).

3.4 Modelos de clase regARIMA.

Ahora pasaremos a estudiar modelos de clase regARIMA, un método autoregresivo integrado de medias móviles, en particular una de sus variantes, modelo de regresión lineal con errores ARIMA.

Vamos crear una función para realizar el cálculo con la función ets:

En esta ocasión nos limitaremos a comparar entre un modelo con regresores y el ajuste por el "trading day" y otro igual pero incluyendo el ajuste por "easter". Cabe mencionar que no podemos realizar un regArima sobre una serie ya desestacionalizada, por lo que nos limitaremos a la serie original. Como regresores utilizaremos los disponibles a partir del año 2006, fecha de inicio de nuestra serie temporal.

```
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
```

```
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
## specs have been added to the model: forecast
```

```
##   Error.regArima.original Error.regArima.original.easter
## 1                25538.99                       24930.74
```

Como vemos, resultado teniendo en cuenta la pascua (easter) es levemente mejor.

3.5 Comparación y conclusiones.

Ahora procederemos a evaluar los resultados globales de los 3 métodos seleccionados.

```
##   Error.método.deriva Error.ets.Multiplicativo Error.regArima.Easter
## 1             66631.5                 38399.11              24930.74
```

Como vemos, el error del regArima en el conjunto de validación es mucho menor que el de los otros dos. A continuación realizaremos la misma comparación con el conjunto test.

```
## specs have been added to the model: forecast
```

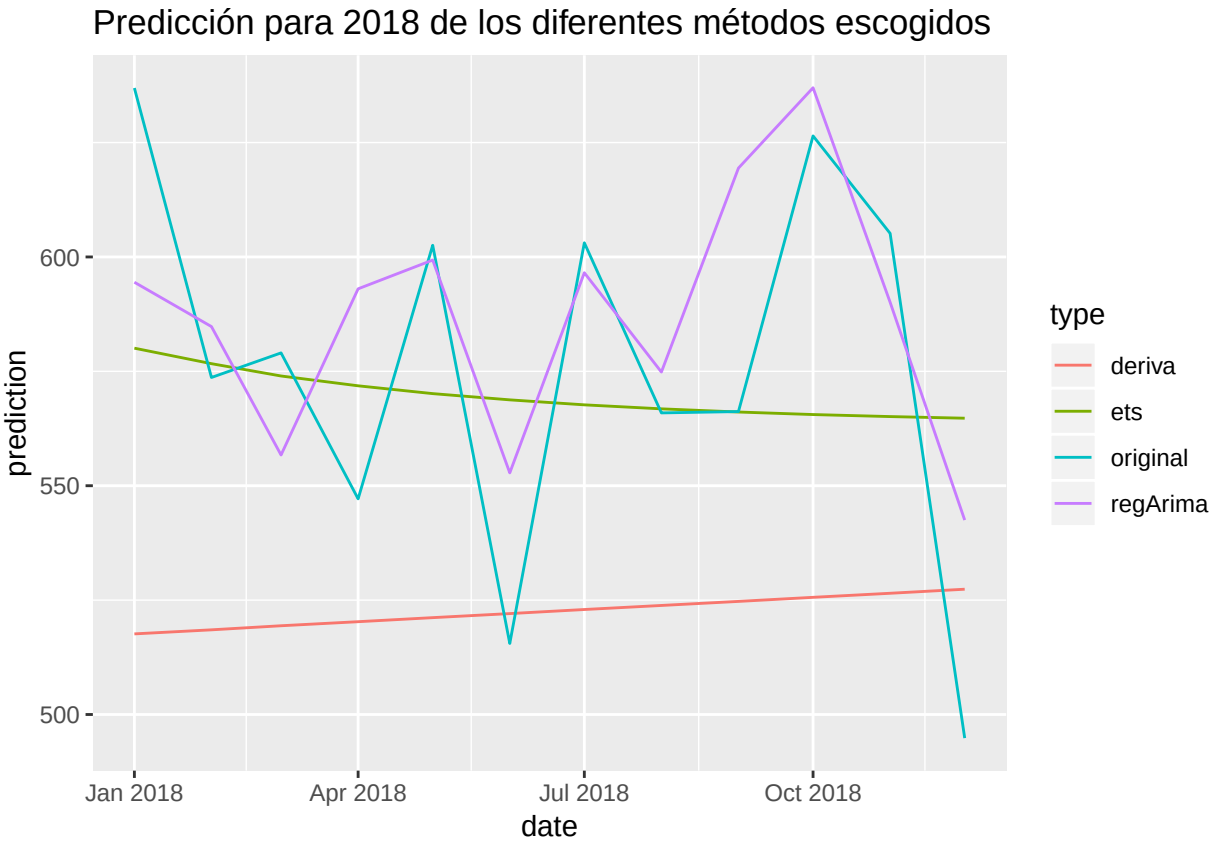
```
##   Error.método.deriva Error.ets.Multiplicativo Error.regArima.Easter
```

```
## 1            62514.69                 38399.11                 11492.37
```

Observamos que el modelo regArima sigue siendo el de menor error, menor incluso que con el conjunto de validación por Cross Validation.

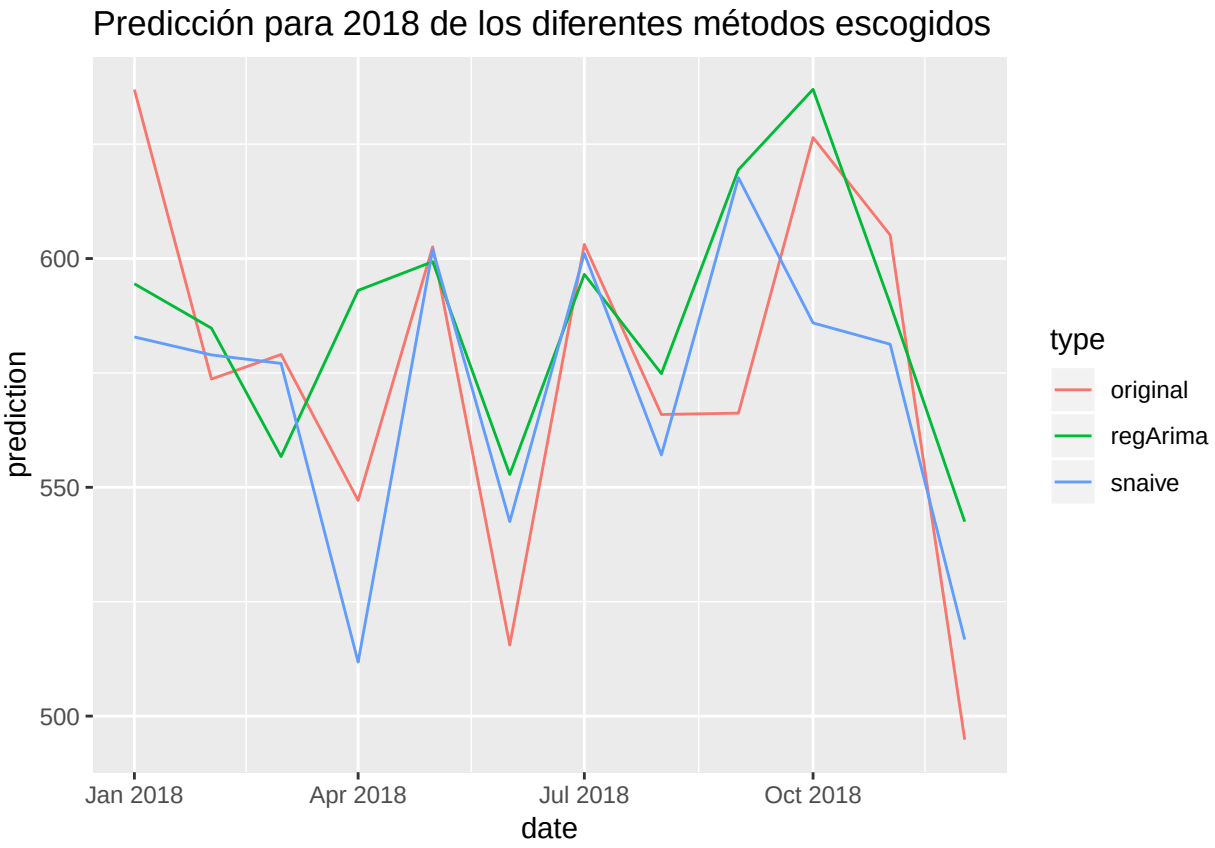
Si realizamos un plot comparando las predicciones de los 3 podremos verlo de forma más clara.

```
## specs have been added to the model: forecast
```



Observamos de forma más clara como el método regArima es el más efectivo a la hora de predecir el año 2018, sin embargo, hemos notado que en esta ocasión (aunque el error en validación es mayor que el del método de la deriva) el método snaive (seasonal naive) puede explicar muy bien los datos del 2018, mostrémoslo con su error test y un plot comparativo.

```
## [1] "El error de test del método snaive es: 10351.65"
```



Observamos como el snaive se ajusta todavía mejor que el regArima a la predicción de 2018. Con esto concluímos que en predicciones tan estacionales como estas (desde el análisis inicial observamos un carácter estacional muy marcado) no siempre conviene buscar métodos muy complejos para predecir adecuadamente los datos, si no que métodos más simples pueden dar resultados satisfactorios. Esto puede ser de utilidad cuando el calculo sea muy costoso especialmente.