

Demanda de Cerveza en Australia

Scarlen Lobaton¹, Gustavo Jemio², Noelia Baina³, and Alejandra Rubín de Celis⁴

¹ Cossmil, Av. Camcho #1324, La Paz, Bolivia
calen1297@gmail.com

² Laboratorios Cofar S.A., Calle Victor Eduardo # 2293, La Paz, Bolivia
gus.jemio@gmail.com

³ Universidad Catolica Boliviana San Pablo, Av. 14 de septiembre # 4807, La Paz, Bolivia
noebel231196@gmail.com

⁴ Centro de distribución El Alto CBN, Av. 6 de marzo, El Alto, Bolivia
alerubindecelisr@gmail.com

Abstract. Arima is a very popular model for scenario simulation and forecasting for technology-based companies, it is flexible and ideal for working with models with seasonal variations based on historical data information that is not related to the innovative product, but is complementary or substitute. Especially to make predictions on future scenarios. In this article we perform the applicability of the ARIMA method to obtain a time series forecast based on beer demand datasets in Australia. Our results suggest that the ARIMA model parameters can be used as a predictive model to forecast future values of time series. By adjusting the ARIMA model, the study shows that the demand for beer shows continuous growth without finding negative deviations that could impact future forecasts.

Arima es modelo muy popular para simulado y pronóstico de escenarios para empresas de base tecnológica, es flexible e ideal para trabajar modelos con variaciones estacionales a partir de información histórica de datos que no son relativos al producto innovador, pero que sí son complementarios o sustitutos. Especialmente para hacer predicciones en los escenarios futuros. En este artículo realizamos la aplicabilidad del método ARIMA para obtener un pronóstico de series temporales en base a datasets de la demanda de cerveza en Australia. Nuestros resultados sugieren que los parámetros del modelo ARIMA se pueden usar como un modelo predictivo para hacer pronósticos de valores futuros de las series temporales. Al ajustar el modelo ARIMA, el estudio demuestra que la demanda de cerveza muestra un crecimiento continuo sin encontrar desviaciones negativas que puedan impactar los pronósticos futuros.

Keywords: beer supply chain datasets · Supply Chain · Datasets

1 Introducción

Se estudiará el caso de un producto con fuerte demanda estacional. En nuestro caso la demanda de cerveza en Australia.

Se comenzará modelando los datos de una serie temporal. Usando R, convertiremos los datos disponibles en formato de datos de serie temporal.

ARIMA es una técnica básica para el pronóstico de series de tiempo complejas, basado en inteligencia artificial. Si son series estacionales se usa S-ARIMA (Seasonal). Este artículo provee una descripción paso a paso para hacer el ajuste y parametrización de la biblioteca arima usando R.

1.1 Estado del arte

Realizamos una búsqueda en google académico con las siguientes palabras clave:

- beer supply chain datasets
- supply chain
- datasets

El resultado nos dio 24500 artículos

De la revisión de los más interesantes seleccionamos por la lectura del título solamente los siguientes artículos

Carbonneau, R., Laframboise, K., & Vahidov, R. (2008). Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, 184(3), 1140-1154.

Fu, G., Shen, S., Zhao, L., Liu, Q., & Li, W. (2013, December). Effective tracking and tracing in rfid-enabled supply chain networks. In *Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)* (pp. 2310-2315). IEEE.

Mircetic, D., Rostami-Tabar, B., Nikolicic, S., & Maslaric, M. (2022). Forecasting hierarchical time series in supply chains: an empirical investigation. *International Journal of Production Research*, 60(8), 2514-2533.

Luego de haber revisado estos artículos hemos incluido las siguientes palabras a las palabras claves:

- beer supply chain datasets
- supply chain
- datasets
- Neural net

Al revisar los artículos destacados podemos comentar lo siguiente de cada uno de ellos:

1.2 Application of machine learning techniques for supply chain demand forecasting

Este artículo hace referencia a la investigación de la aplicabilidad de la tecnología avanzada en técnicas de aprendizaje automático, incluidas redes neuronales, redes neuronales recurrentes y máquinas de vectores de soporte, para pronosticar una demanda distorsionada al final de una cadena de suministro (efecto látigo). Los hallazgos sugieren que, si bien las redes neuronales recurrentes y las máquinas de vectores de soporte muestran el mejor rendimiento, la precisión del pronóstico no fue estadísticamente significativamente mejor que la del modelo de regresión

1.3 Effective tracking and tracing in rfid-enabled supply chain networks

Este artículo hace referencia a que una red de cadena de suministro habilitada para RFID facilita el rastreo y seguimiento de etiquetas mediante el intercambio de datos en infraestructuras de servicios de información (IS) o servicios de descubrimiento (DS), a lo largo de todo el ciclo de vida de una etiqueta. EPCGlobal es un estándar ampliamente aplicado para una red de cadena de suministro habilitada para RFID, que define la gestión del ciclo de vida de una etiqueta EPC, al describir los problemas de seguimiento y rastreo en una red EPCGlobal. Sin embargo, los algoritmos de rastreo y rastreo existentes no consideran eventos complicados definidos en los estándares EPCGlobal, como agregación, desagregación, recuperación, comentario, etc. y rastreo en el contexto de eventos complicados.

1.4 Forecasting hierarchical time series in supply chains: an empirical investigation

En este artículo, se investiga el desempeño de los pronóstico de combinación óptima, ascendente y descendente en una cadena de suministro. Primero se evalúa su rendimiento mediante un estudio de simulación y una investigación empírica en una red de distribución. Luego, se examina la precisión del pronóstico de combinar pronósticos de estos enfoques. Los resultados revelan que las combinaciones de pronósticos producen pronósticos que son más precisos y menos sesgados que los enfoques individuales.

2 Justificación

Realizamos este trabajo para poder conocer más el método ARIMA, ya que es un modelo predictivo que permite hacer pronósticos de valores futuros de las series temporales, por esta razón, como la cerveza tiene etapas en las que su venta varía, se realizó el análisis y la implementación de este método.

3 Materiales y Métodos

3.1 Materiales

Se utilizó el dataset de un caso de productos con fuerte demanda estacional. Se analizará el caso la demanda de cerveza en Australia.

Para llevar a cabo este trabajo se utilizará el método ARIMA. ARIMA por sus siglas en inglés, acrónimo del inglés autoregressive integrated moving average, o Modelo autorregresivo integrado de media móvil es muy utilizado en estadística y econometría, en particular en series temporales, un modelo autorregresivo integrado de promedio móvil o simplemente un sistema que se comporta como ARIMA es un modelo estadístico que utiliza variaciones y regresiones de

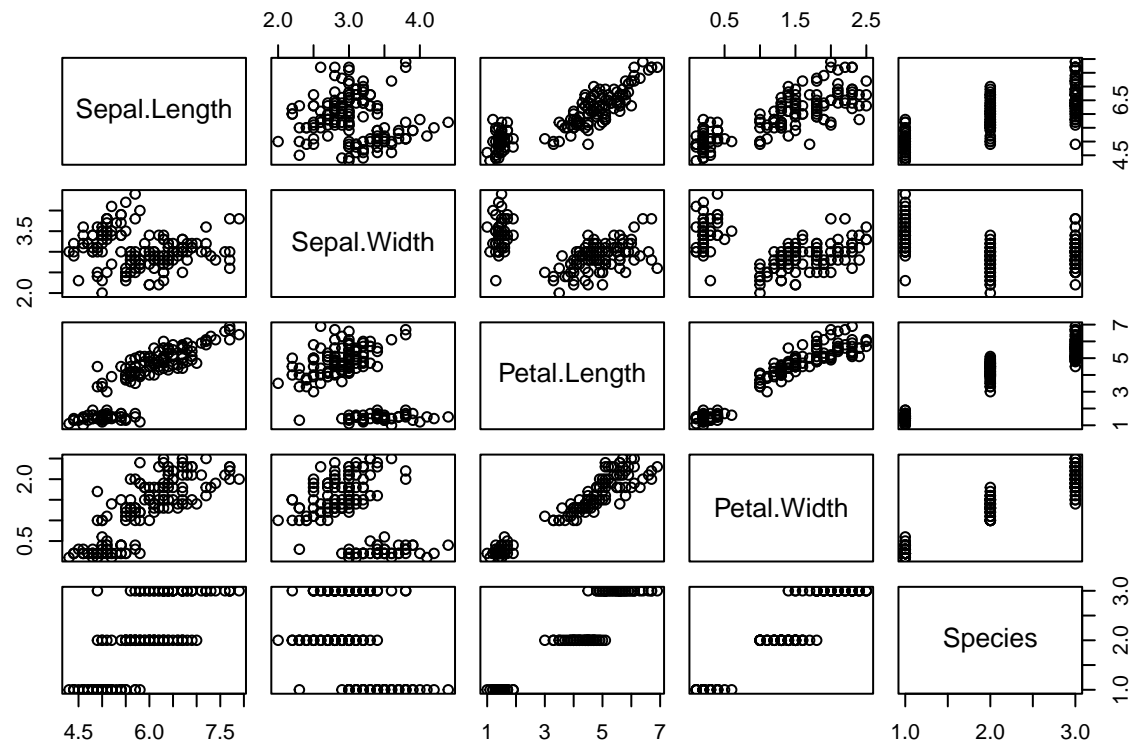
datos estadísticos con el fin de encontrar patrones para una predicción hacia el futuro.

Para llevar a cabo este trabajo, se utilizaron distintos comandos, por ejemplo los siguientes:

```
summary(cars)
```

```
##      speed      dist
##  Min.   : 4.0   Min.   : 2.00
## 1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##   Mean  :15.4   Mean   : 42.98
## 3rd Qu.:19.0   3rd Qu.: 56.00
##   Max.  :25.0   Max.   :120.00
```

```
plot(iris)
```



3.2 Métodos

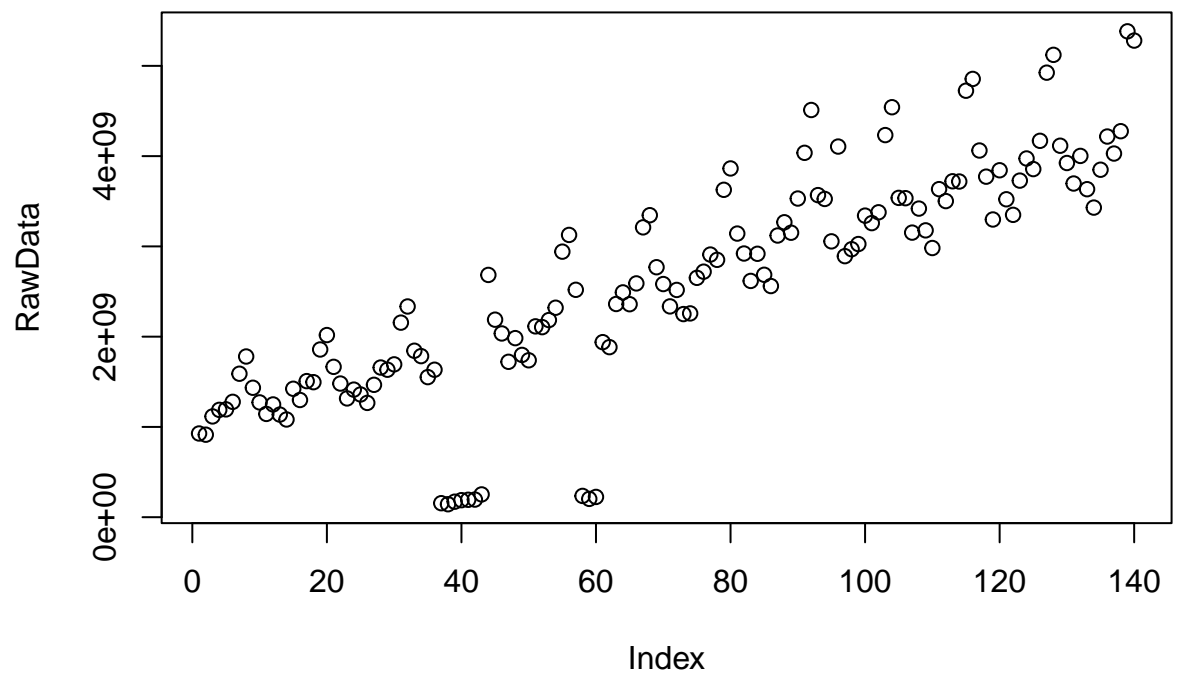
El dataset que utilizamos podría ser analizado con KNN o también con una red neuronal, se podrían clasificar los consumos por mes, etc.

Considerando la estructura del dataset que conseguimos hemos optamos por utilizar el método ARIMA.

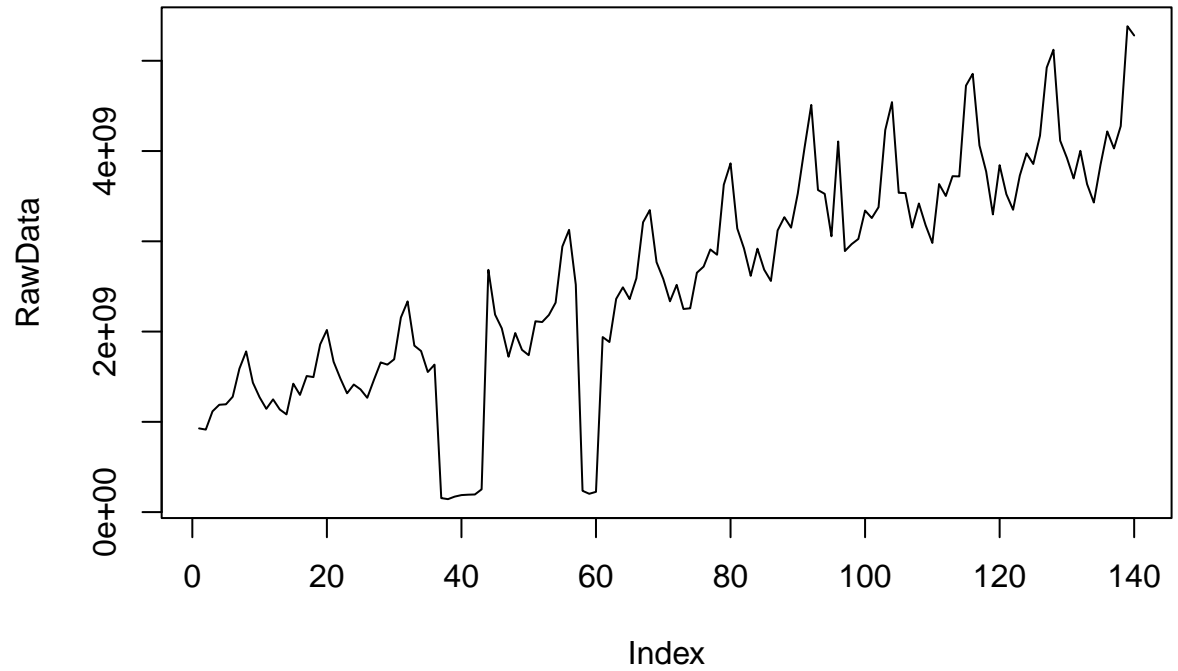
4 Consumo de cerveza

Primer paso recuperar los datos

```
RawData=scan('https://themys.sid.uncu.edu.ar/~rpalma/R-cran/Routers.txt')  
plot(RawData)
```



```
RawData=scan('https://themys.sid.uncu.edu.ar/~rpalma/R-cran/Routers.txt')  
plot(RawData, type = "l")
```

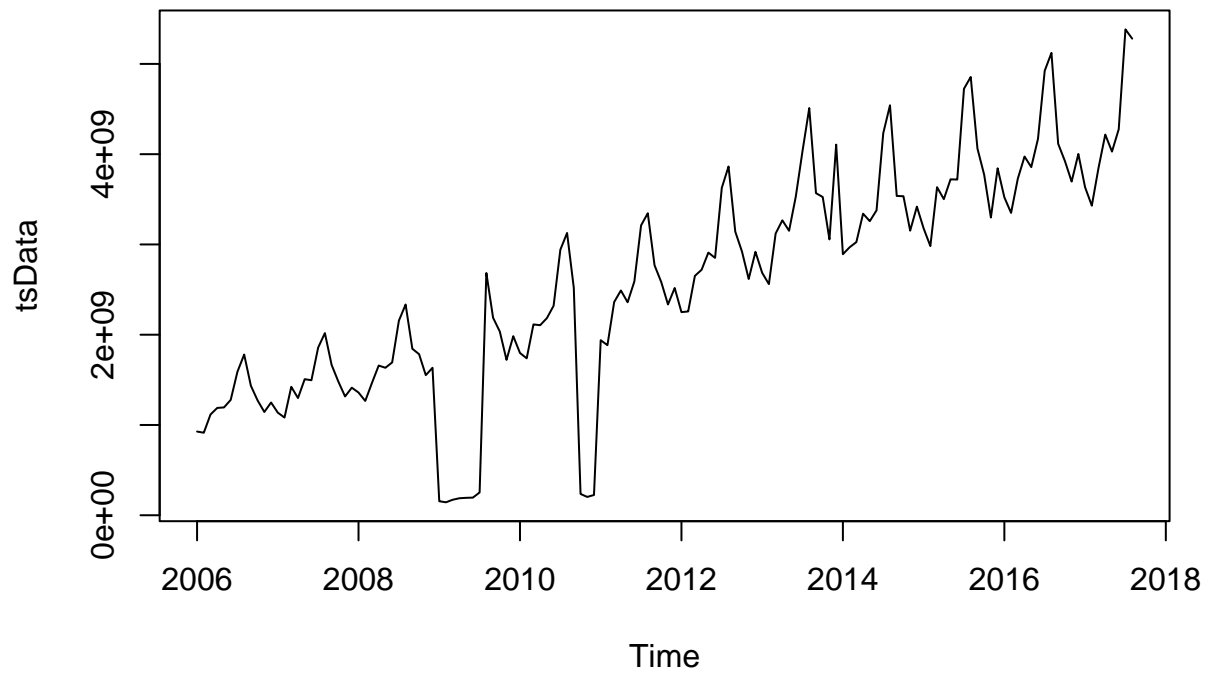


```
tsData = ts(RawData, start = c(2006,1), frequency = 12)
print(tsData)
```

```
##           Jan           Feb           Mar           Apr           May           Jun
## 2006  927180000  913800000 1116430000 1188880000 1194320000 1277960000
## 2007 1136610000 1082240000 1422560000 1298350000 1507350000 1495540000
## 2008 1359510000 1266150000 1466470000 1658220000 1633650000 1692940000
## 2009 154844000  143552000  171573000  188322000  192756000  195296000
## 2010 1797590000 1738210000 2113870000 2105510000 2183710000 2320570000
## 2011 1939160000 1883750000 2361870000 2490370000 2359570000 2589800000
## 2012 2250100000 2257420000 2651590000 2719860000 2909530000 2851080000
## 2013 2685780000 2560630000 3120410000 3267410000 3151570000 3530160000
## 2014 2891860000 2968810000 3025890000 3340910000 3257900000 3377820000
## 2015 3177600000 2981880000 3634290000 3502030000 3721490000 3718770000
## 2016 3522000000 3349380000 3728910000 3973880000 3856570000 4169610000
## 2017 3633670000 3429790000 3849360000 4217180000 4028770000 4276150000
##           Jul           Aug           Sep           Oct           Nov           Dec
```

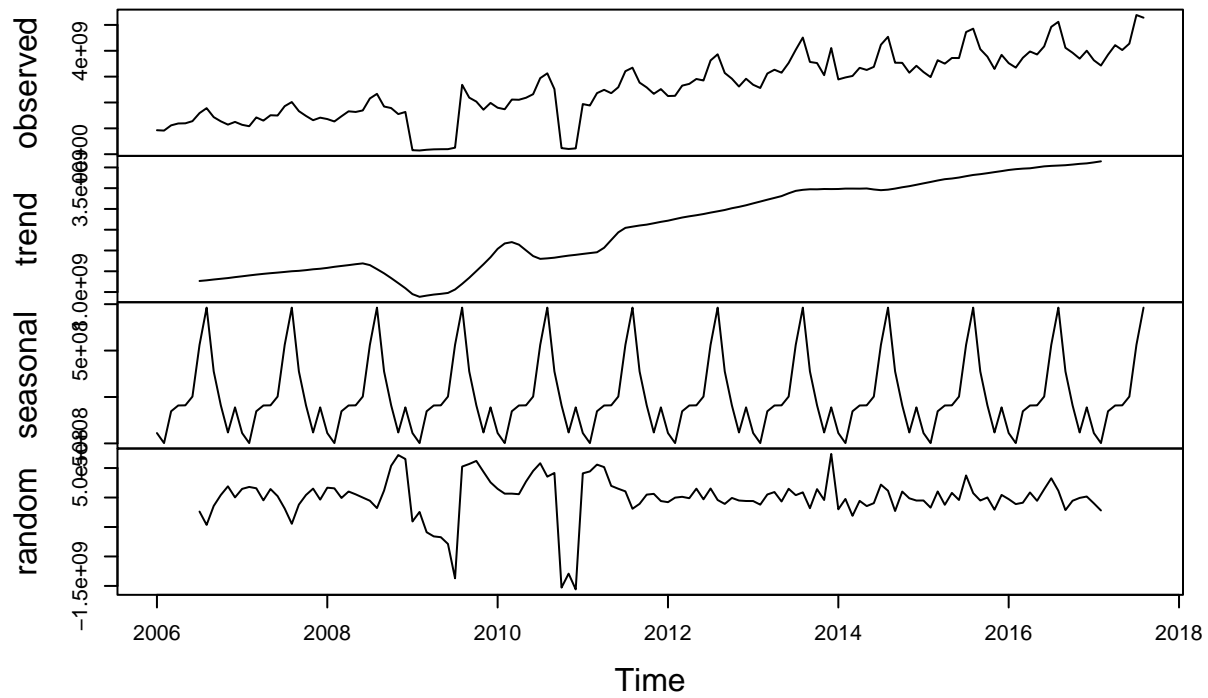
```
## 2006 1589430000 1780130000 1433850000 1271790000 1144030000 1249000000
## 2007 1857920000 2017580000 1665650000 1480480000 1315810000 1413150000
## 2008 2155380000 2334270000 1844020000 1784320000 1551790000 1633550000
## 2009 252288000 2683790000 2188100000 2035450000 1721480000 1983810000
## 2010 2941730000 3127000000 2518910000 235560000 202876000 224383000
## 2011 3210850000 3345620000 2769320000 2582690000 2335320000 2517550000
## 2012 3626870000 3863470000 3142050000 2921240000 2617400000 2918100000
## 2013 4036620000 4510980000 3568110000 3525660000 3055800000 4106140000
## 2014 4232970000 4541720000 3537270000 3534130000 3152720000 3419020000
## 2015 4724580000 4855170000 4062230000 3772620000 3297940000 3843500000
## 2016 4924800000 5122090000 4115140000 3923800000 3696710000 4002430000
## 2017 5382540000 5280070000
```

```
plot(tsData)
```



```
componentes.ts = decompose(tsData)
plot(componentes.ts)
```

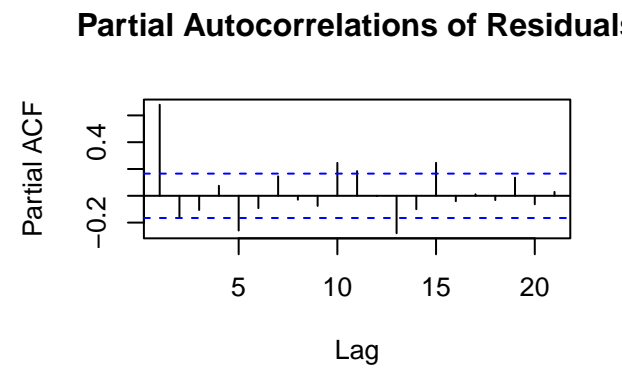
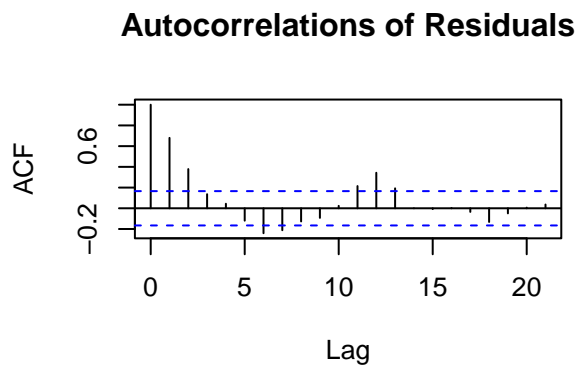
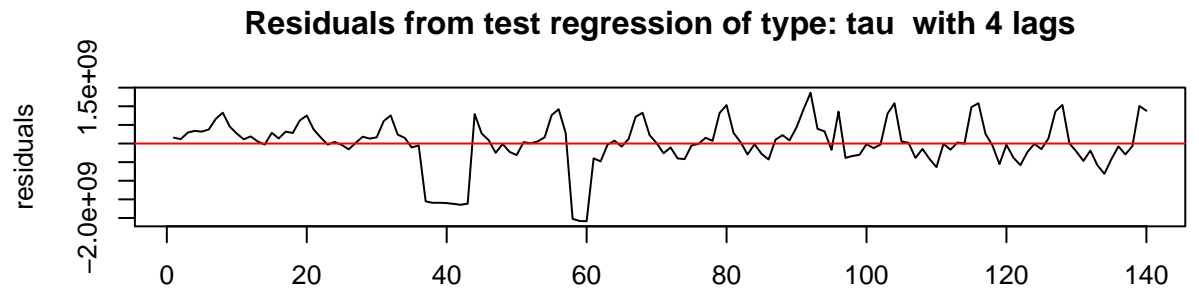
Decomposition of additive time series



```
library("fUnitRoots")
```

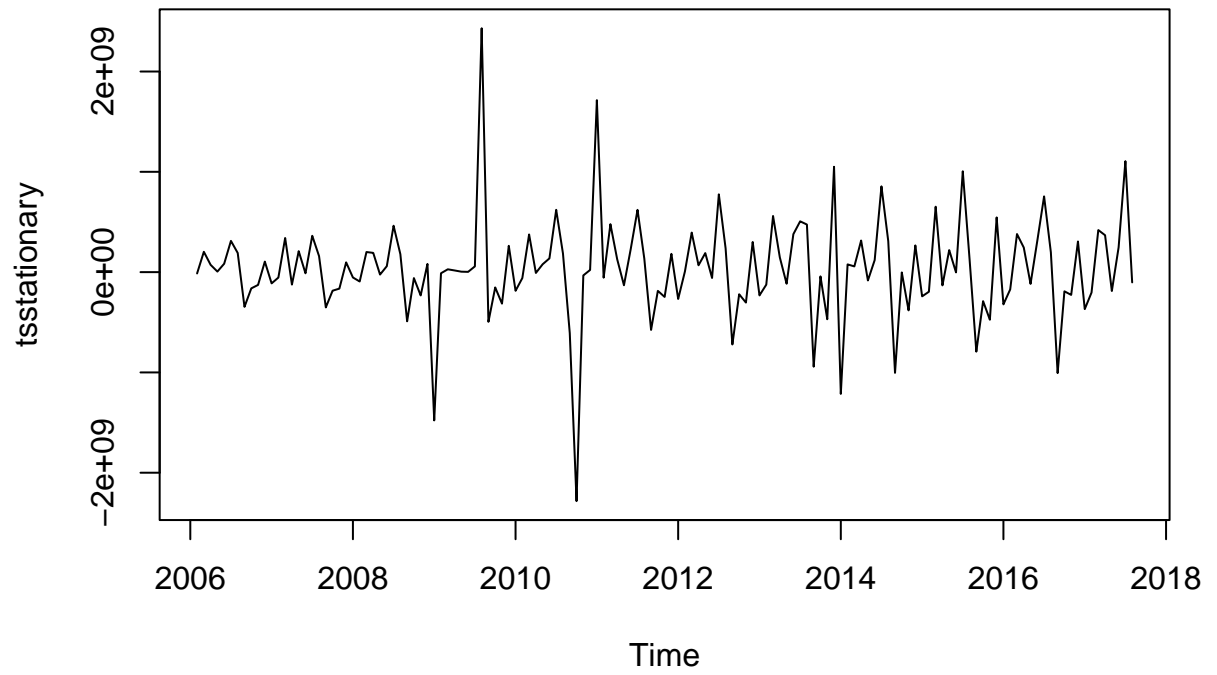
```
## Warning: package 'fUnitRoots' was built under R version 4.3.1
```

```
urkpssTest(tsData, type = c("tau"), lags = c("short"), use.lag = NULL, doplot = TRUE)
```

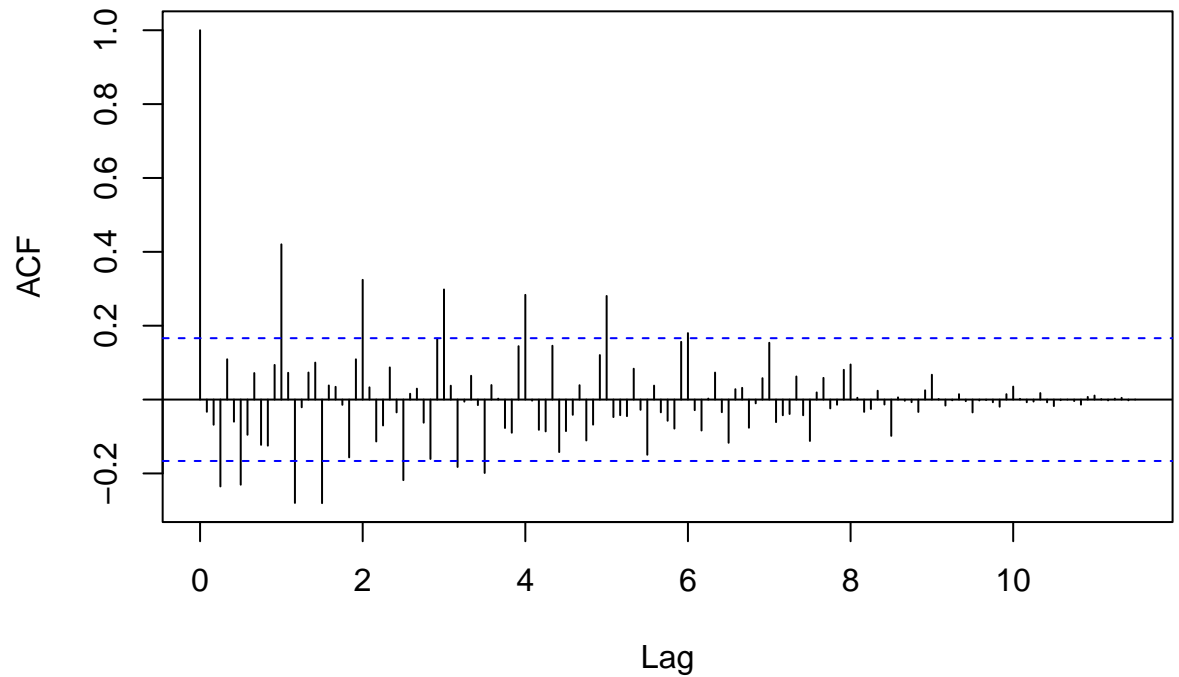
```
##
## Title:
## KPSS Unit Root Test
##
## Test Results:
## NA
##
## Description:
## Mon Sep 18 01:15:46 2023 by user: DELL
```

```
tsstationary = diff(tsData, differences=1)
plot(tsstationary)
```



```
acf(tsstationary, lag.max=140)
```

Series tsstationary



```
fitARIMA <- arima(tsData, order=c(1,1,1),seasonal =list(order = c(1,0,0), period = 12),metho
#install.packages("lmtest")
library(lmtest)
```

```
## Warning: package 'lmtest' was built under R version 4.3.1
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.3.1
```

```
##
```

```
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      as.Date, as.Date.numeric
```

```
coeftest(fitARIMA)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ar1  -0.795253   0.108606 -7.3224 2.436e-13 ***
## ma1   0.648117   0.129663  4.9985 5.778e-07 ***
## sar1  0.484253   0.074262  6.5209 6.991e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
fitARIMA <- arima(tsData, order=c(1,1,1), seasonal =list(order = c(1,0,0), period = 12),met
```

```
confint(fitARIMA)
```

```
##           2.5 %      97.5 %
## ar1  -1.0081176 -0.5823892
## ma1   0.3939830  0.9022507
## sar1  0.3387015  0.6298038
```

```
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.3.1
```

```
## Registered S3 method overwritten by 'quantmod':
```

```
##   method          from
```

```
##   as.zoo.data.frame zoo
```

```
auto.arima(tsData, trace=TRUE)
```

```
##
## ARIMA(2,1,2)(1,0,1)[12] with drift          : Inf
## ARIMA(0,1,0)                with drift          : 5971.076
## ARIMA(1,1,0)(1,0,0)[12] with drift          : Inf
## ARIMA(0,1,1)(0,0,1)[12] with drift          : Inf
## ARIMA(0,1,0)                : 5969.545
## ARIMA(0,1,0)(1,0,0)[12] with drift          : Inf
## ARIMA(0,1,0)(0,0,1)[12] with drift          : Inf
## ARIMA(0,1,0)(1,0,1)[12] with drift          : Inf
## ARIMA(1,1,0) with drift          : Inf
## ARIMA(0,1,1) with drift          : Inf
## ARIMA(1,1,1) with drift          : Inf
##
## Best model: ARIMA(0,1,0)
```

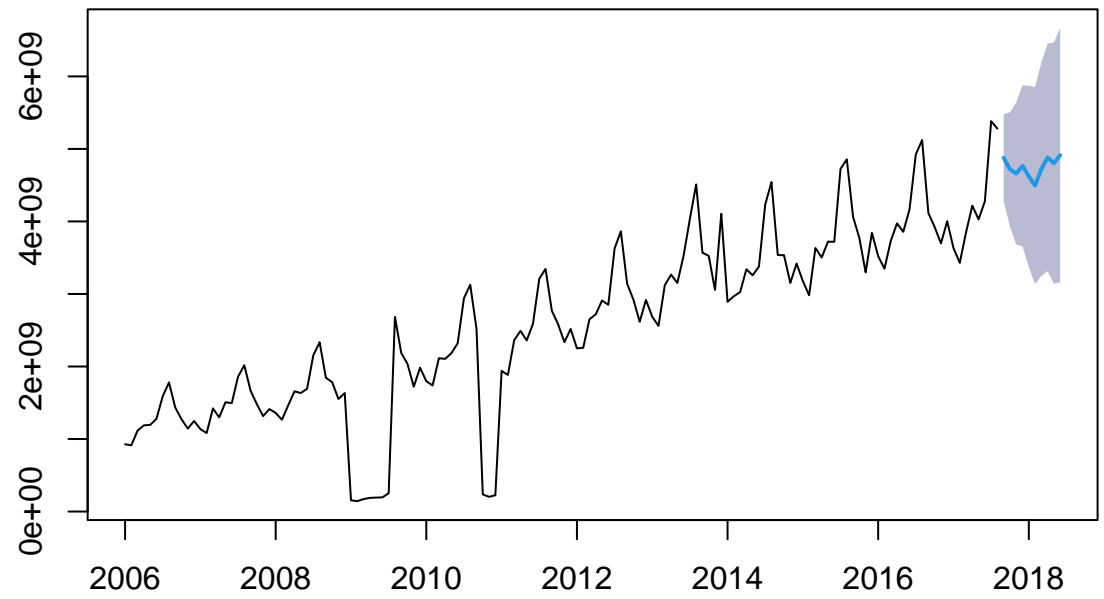
```
## Series: tsData
## ARIMA(0,1,0)
##
## sigma^2 = 2.586e+17: log likelihood = -2983.76
## AIC=5969.52 AICc=5969.55 BIC=5972.45
```

```
predict(fitARIMA,n.ahead = 25)
```

```
## $pred
##           Jan           Feb           Mar           Apr           May           Jun
## 2017
## 2018 4621589493 4495874101 4720512664 4881563800 4803898089 4912899068
## 2019 5085922077 5023317415 5133472360 5210369818 5173628394 5225721807
##           Jul           Aug           Sep           Oct           Nov           Dec
## 2017
## 2018 5457254885 5400807440 5211190384 5129331127 5105494727 5153792809
## 2019 5489876759 5462105179 5370629947
##
## $se
##           Jan           Feb           Mar           Apr           May           Jun
## 2017
## 2018 921409969 1002298813 1084880952 1155826322 1226992814 1290977709
## 2019 1957329034 2046140741 2134457227 2216808215 2298088884 2375143098
##           Jul           Aug           Sep           Oct           Nov           Dec
## 2017
## 2018 1354418526 1413126017 1544150766 1651889332 1762718042 1859667291
## 2019 2450890512 2523504029 2620829284
```

```
futurVal <- forecast(fitARIMA,h=10, level=c(82.5))
plot(futurVal)
```

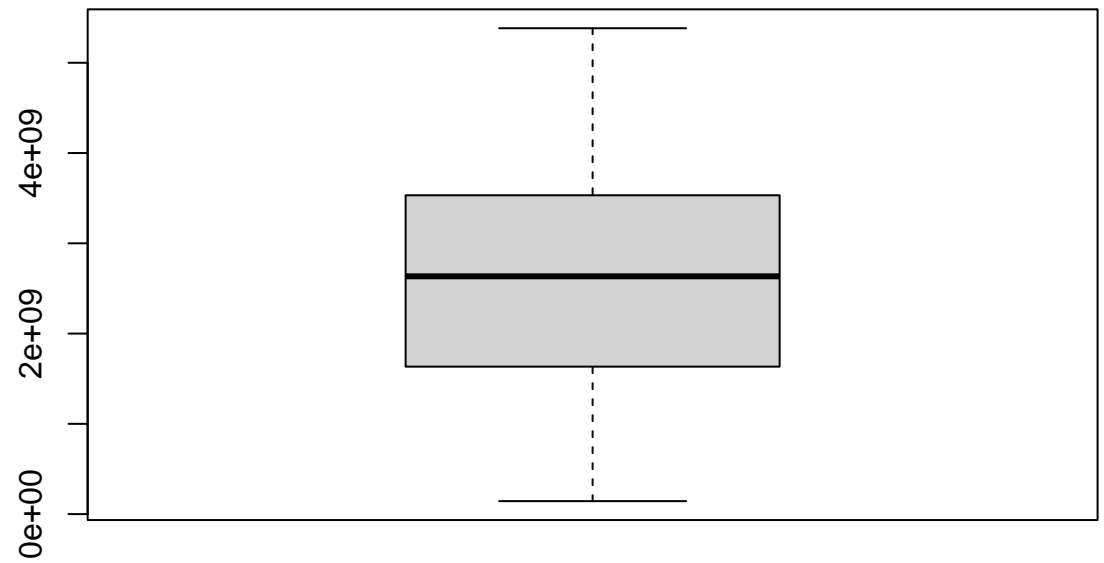
Forecasts from ARIMA(1,1,1)(1,0,0)[12]



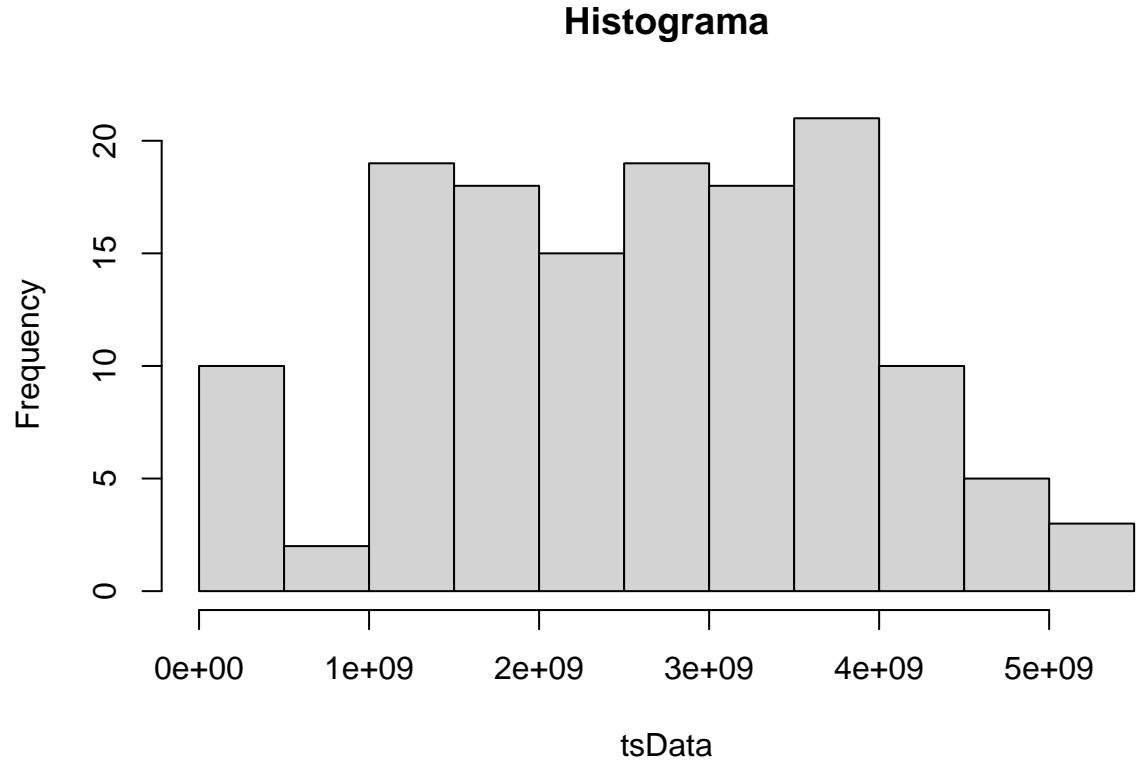
```
summary(tsData)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## 1.436e+08 1.634e+09 2.634e+09 2.585e+09 3.531e+09 5.383e+09
```

```
boxplot(tsData)
```



```
hist(tsData, main = "Histograma")
```



5 Interpretación de resultados y conclusiones

Según los resultados obtenidos, se utilizó R para convertir los datos en formato de datos de serie temporal, para poder tener una mejor visualización de los datos, se ajustó el primer gráfico (RawData).

Se puede observar que a inicios de la gestión 2009 y a finales de la gestión 2010 la demanda de la cerveza en Australia sufrió dos bajones en su consumo.

La demanda de la cerveza en Australia comienza con mucha variabilidad y no presenta un patrón al inicio de los datos obtenidos, posteriormente, al realizar los ajustes correspondientes se puede llegar a observar la estacionalidad de los datos, que se van ajustando.

Por otra parte, al realizar el ajuste del modelo se puede observar que el crecimiento de la demanda de cerveza va incrementando y no se genera ninguna eventualidad negativa, que pueda afectar a la demanda en el pronóstico realizado.

Los pronósticos se muestran con una línea azul, con los intervalos de predicción del 82.5% como un área sombreada oscura, y los intervalos de predicción del 95% como un área sombreada clara.

5.1 A Subsection Sample

Please note that the first paragraph of a section or subsection is not indented. The first paragraph that follows a table, figure, equation etc. does not need an indent, either.

Subsequent paragraphs, however, are indented.

Sample Heading (Third Level) Only two levels of headings should be numbered. Lower level headings remain unnumbered; they are formatted as run-in headings.

Sample Heading (Fourth Level) The contribution should contain no more than four levels of headings. Table 1 gives a summary of all heading levels.

```
knitr::kable(head(pressure),
              format = "latex",
              caption = "Table captions should be placed above the tables.")
```

Table 1. Table captions should be placed above the tables.

temperature	pressure
0	0.0002
20	0.0012
40	0.0060
60	0.0300
80	0.0900
100	0.2700

Displayed equations are centered and set on a separate line.

$$x + y = z \tag{1}$$

Please try to avoid rasterized images for line-art diagrams and schemas. Whenever possible, use vector graphics instead (see Fig. ??).

```
plot(pressure)
```

Theorem 1. *This is a sample theorem. The run-in heading is set in bold, while the following text appears in italics. Definitions, lemmas, propositions, and corollaries are styled the same way.*

The environments ‘definition’, ‘lemma’, ‘proposition’, ‘corollary’, ‘remark’, and ‘example’ are defined in the LLNCS documentclass as well.

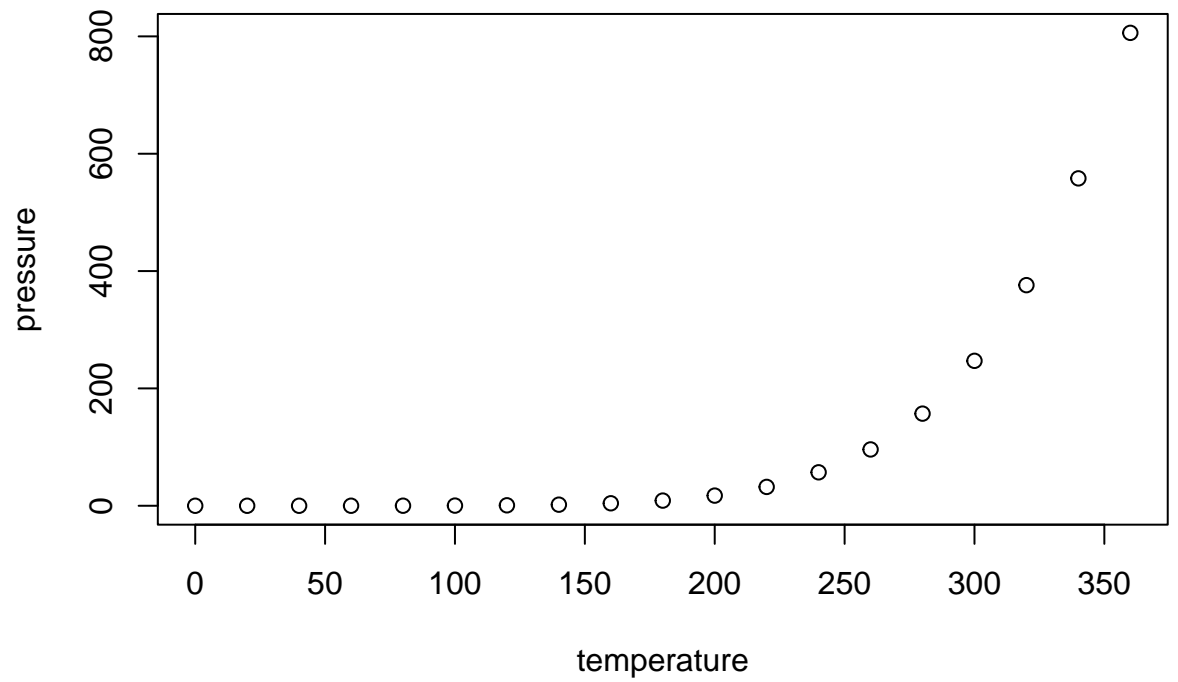


Fig. 1. A figure caption is always placed below the illustration. Please note that short captions are centered, while long ones are justified by the macro package automatically.

Proof. Proofs, examples, and remarks have the initial word in italics, while the following text appears in normal font.

For citations of references, use [?]. Multiple citations are grouped [?,?].

Acknowledgements Please place your acknowledgments at the end of the paper, preceded by an unnumbered run-in heading (i.e. 3rd-level heading).