STA2202 - Time Series Analysis - Assignment 2 - PRACTICE

Luis Correia - Student No. 1006508566

May 27th 2020

Submission instructions: Submit *three separate files* to A2 on Quercus - the deadline is 11:59PM on Tuesday, June 2.
- A PDF file with your Theory part answers.
- A PDF file with your Practice part report.
- A CSV file with your Practice part forecasts.

Practice

Description

It is your first day on the job and your boss, who graduated from the UofT Statistics program in 2013 has given you the task of forecasting a time series. Your forecasts will serve as an input to the firm's budget, so it is critical that they are accurate. Your boss would like you to provide them with forecasts, as well as a description of how you came up with them.

Assignment Structure

You will be given one time series and must produce a forecast for the next twelve observations. You can find your time series in the *Student Data* sub-folder of the R Studio Cloud project; the name of your data file is your student number, and the name of the series you are forecasting is on the top row of the file. Your submission will include two files:

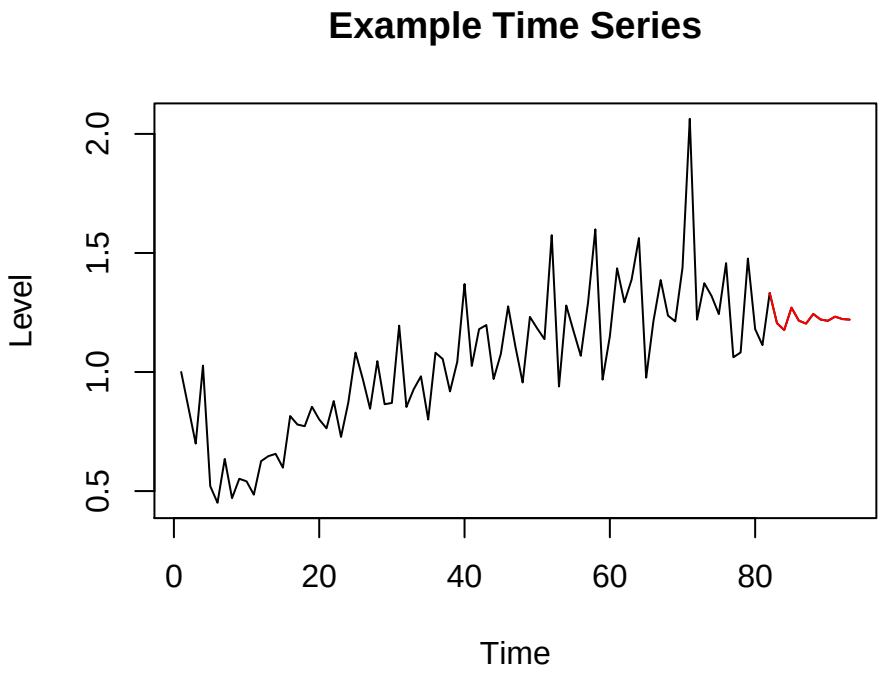
1. A 500-word written report in PDF format, with all your code in the Appendix.
2. A CSV file named `XXXXXXXXXX.csv`, where `XXXXXXXXXX` is your student number. This CSV should include your forecasts for the next twelve values of your series; the first entry should be your one-step-ahead forecast, and the twelfth entry should be your 12-step-ahead forecast (see also the sample file `123456789.csv` in the project's *Examples* sub-folder.)

Written Report

Your written report should be able to be understood by your boss, someone who remembers the main ideas from a time series course several years ago, but not the finer details. Be sure to clearly explain what you have done, and if you are using any advanced concepts a sentence or two to refresh your boss on what they are is a good idea. Your written report must include the following:

1. A discussion of the characteristics of the time series (e.g. trend, seasonality, stationarity)
2. An explanation of any data preprocessing you had to do.
3. The model which you used.
4. A graph of the time series, with your forecasts in a different colour (see graph below for an example)

5. A discussion of your model's fit (diagnostics) and limitations.
   The list above is what your written report must contain, but not an exhaustive list of all that it can contain. If there are any other topics that are worth discussing related to how you forecasted the data, please include them.



Tips

- This is a report to your boss. Concise & clear is better. They do not want to see single spaced size 6 font with expanded margins. They want to see all important and relevant information neatly organized.
- If you are going to include a code snippet in your written report (this is not required), make sure it is important enough to warrant your boss' attention.
- Make sure the model you choose, and how you fit it, makes sense. The data you are working with may violate some basic time series assumptions.

Assessment (15pts total)

- 1pt Your written report has a clean layout, and includes the requested graph.
- 1pt The text of your report is easy to follow, and conveys ideas effectively.
- 1pt Your CSV file with your predictions is properly formatted.
- 2pt Time Series Characteristics
  - 1/2 Some mention of the important time series characteristics.
  - 2/2 A clear identification of all important time series characteristics.
- 2pt Data preprocessing
  - 1/2 Some vague explanation of how the data has been preprocessed is provided.
  - 2/2 A clear explanation of how the data was preprocessed and the justification for why it was done.
- 2pt Model Explanation
  - 1/2 You have included a model description, but little in the way of explanation.
  - 2/2 You have concisely and clearly explained your model.
- 2pt Model Fit and Limitations

  - 1/2 Give vague description of the model's fit and limitations.
  - 2/2 Give clear and accurate description of the model's fit and limitations.
- 4pt Forecast Accuracy
  - 1pt Your method beats the naive forecast (the entire forecast is equal to the last data-point)
  - 1pt Your forecast beats the forecast produced by the R code `ts_arima_model = auto.arima(x); forecast(ts_arima_model, h = 12)`
  - 1pt Your forecast beats the forecast produced by the R code `ts_ets_model = ets(x); forecast(ts_ets_model, h = 12)`
  - 1pt Your method beats all of the naive, `auto.arima()`, and `ets()` methods.

The way your forecasts will be judged is via Mean Absolute Percentage Error (MAPE) on the actual subsequent 12 values (not given to you, but known to us). Defining A_t as the actual value at time t, and F_t as your corresponding forecasted values in the submitted csv file, the MAPE for your forecasts will be calculated as

$$MAPE = \frac{1}{12}\sum_{t=1}^{12}\left|\frac{A_t - F_t}{A_t}\right|$$

Your forecast *beats* another forecast if your MAPE is lower.

Technical Report

Preliminary Analysis

The series of interest is composed by a sequence of data related to **Italian Banana Demand**. It covers 306 observations plotted in the graph below.

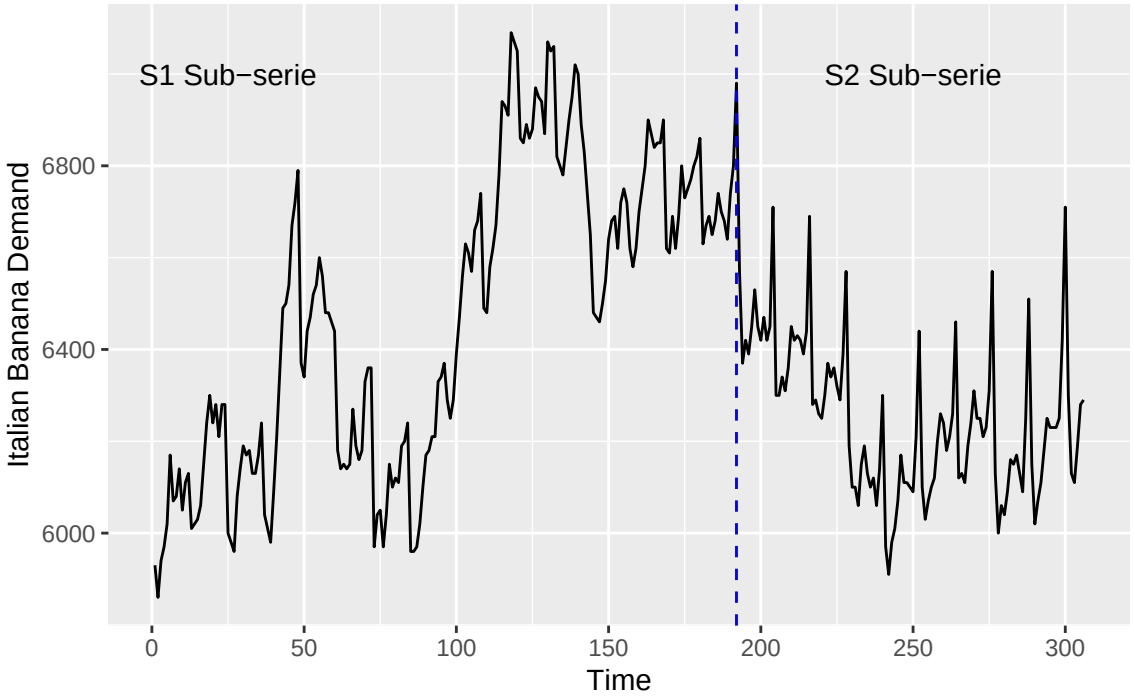


Figure 1: Observed - Italian Banana Demand

As we can see, there are different patterns of this time-series which might suggest the series is not stationary. From observation $No.1$ to 50 the approximate demand level of the product is around level $6,300$, jumping to $6,700$ in the next 100 observations.

Furthermore, the series shows *two different behaviors* affecting its trend, seasonality, as follows:

- The first pattern emerged approximately after 2/3 of the series - which we will call *S1* - and the second in the remaining third of the series, which we will name it *S2* from now on;
- The **trend** of the series is clearly different in S1 and S2: a) S1 has a *random-walk*-like pattern with the demand increasing and decreasing randomly, with some spikes that could suggest a presence of hidden **seasonality**; and b) S2 with a more clear 12-month seasonality, besides of a negative trend in the first 60-observations which might reflect retraction of demand of the product, followed by a positive trend after it;
- This alternation of behavior between S1 and S2 lead us to suspect that possibly a change in the market might have occurred from in between years #16 to #17 that created some external influence which affected the way the demand for the product behaves in these two periods.

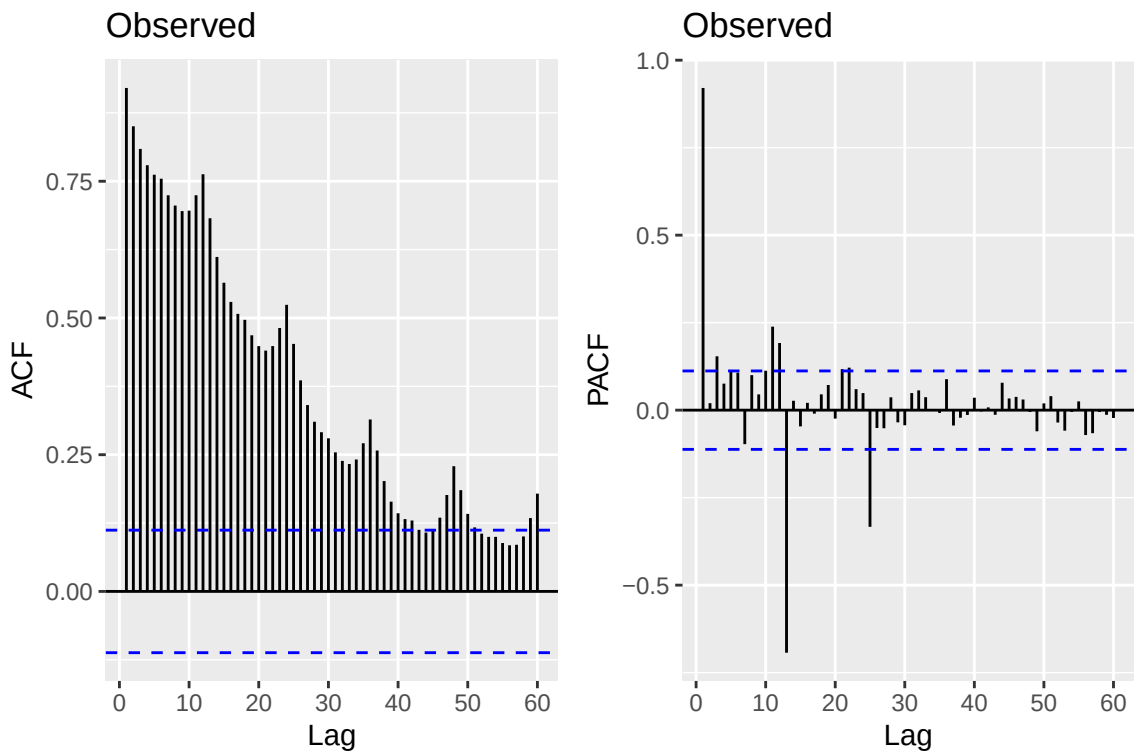


Figure 2: ACF/PACF - Italian Banana Demand

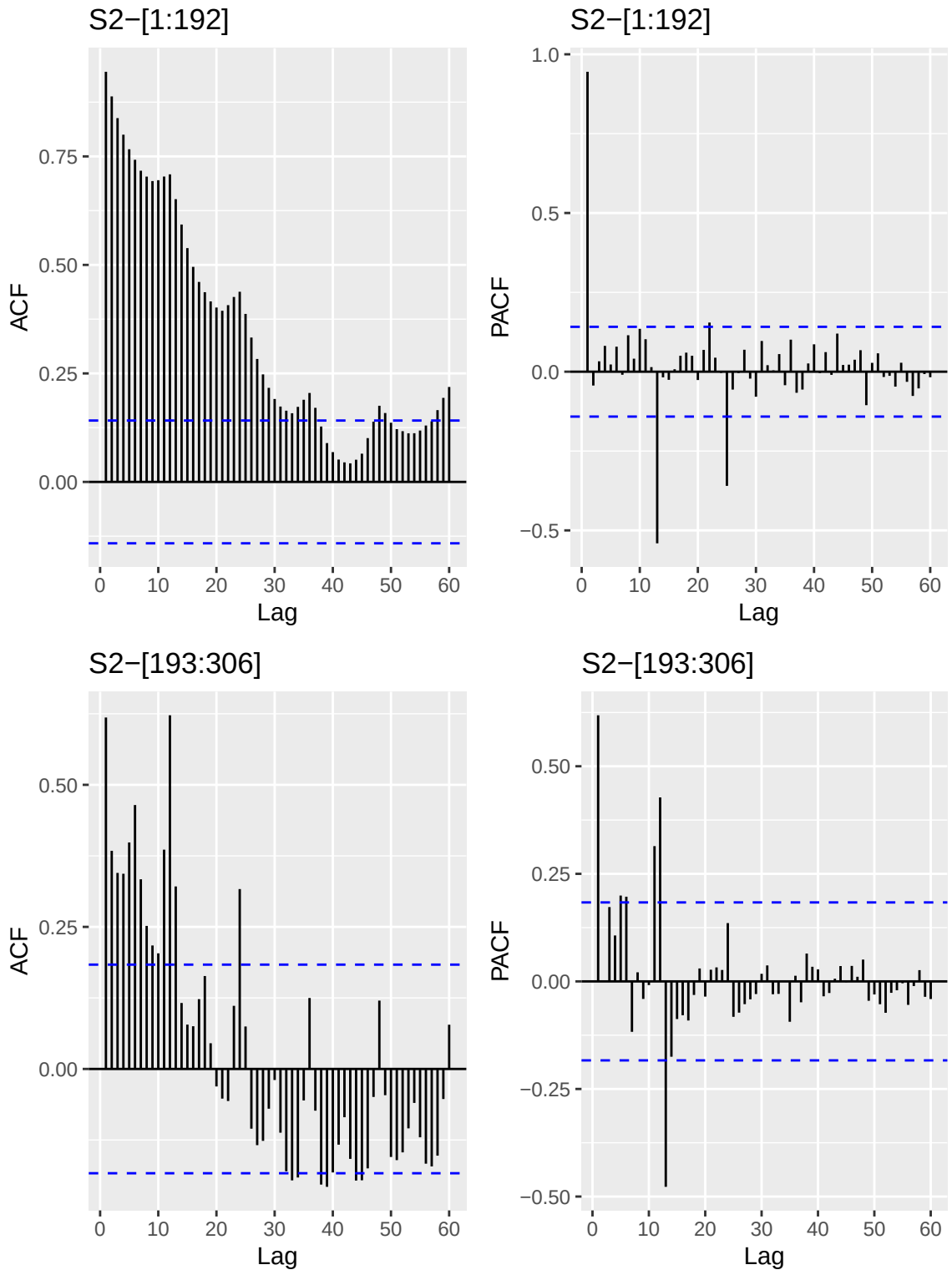
Analyzing the *ACF-Auto Correlation Function* we can see decreasing correlation as the lags grow while the *PACF-Partial Auto-Correlation Function* cuts-off after $lag = 24$ confirms our initial impressions and suggests we can have a seasonal component that must be addressed in order to fit a proper model.

Methodological Approach

In order to address the issues raised in the *Preliminary Analysis*, we will treat the data to minimize the effect of lack of stationarity and presence of seasonal component.

We will focus our efforts on S2 series, which counts with more recent data and can provide better and more accurate estimates for our predictions, since it encapsulates the pattern we want to capture in our model to produce the desired forecast.

Aligned with this strategy, as already mentioned, we will start by splitting the original series into S1/S2 sub-series using a Cut-off at Observation No. 204, which is multiple of 12 and appropriately represent the change of behavior mentioned in *Preliminary Analysis*.



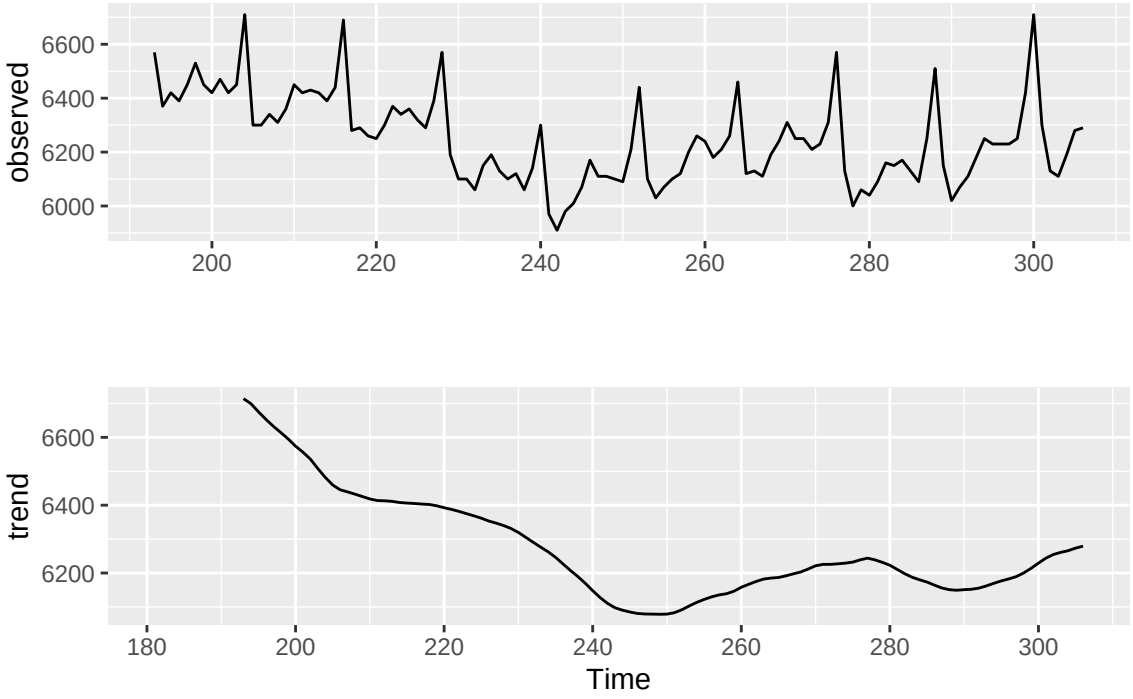
We can see through the ACF/PACF plots that S1 has a different shape, with S2 with a pure auto-regressive behavior with ACF tailing off and PACF cutting off after some lags. This turns S2 into a more promising series to be modeled.

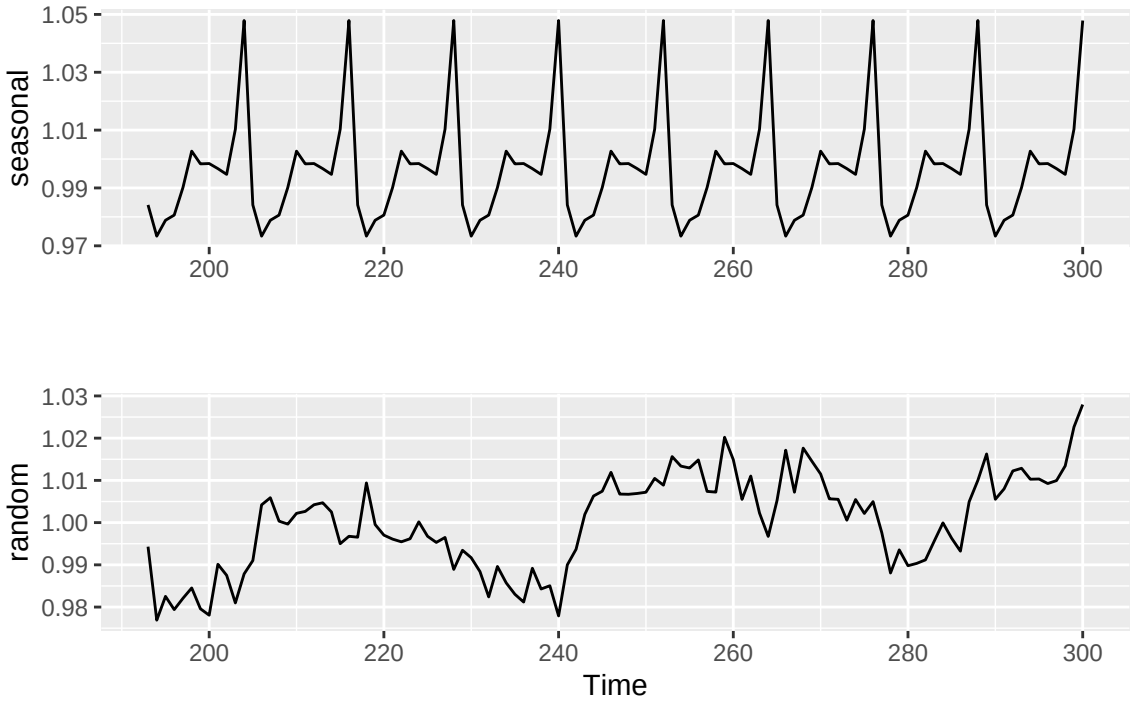
Data Transformation & Processing

The steps followed to stabilize the S2-series will be as follows:

1. Decompose the series in 03 parts: a) Trend; b) Seasonality and c) Remainder using the *multiplicative method*;
2. Remove the 12-period seasonality observed in the original series;
3. Analyse the ACF/PACF plots of the transformed data in order to identify the appropriate order/type of model to be used, including additional first order of difference that might be necessary;
4. Select AR/MA/ARMA/ARIMA or SARIMA Model, according with the previous steps.

The decomposition of the S2-Series seems to confirm our initial perception about *trend* and *seasonality* as we can see in the graphs below.





In steps *No.*1 and *No.*2 we obtained the *de-seasonal S2- Series*, which is plotted in the next graph. We can see that this plot shows an auto-regressive pattern that we will confirm during the analysis.

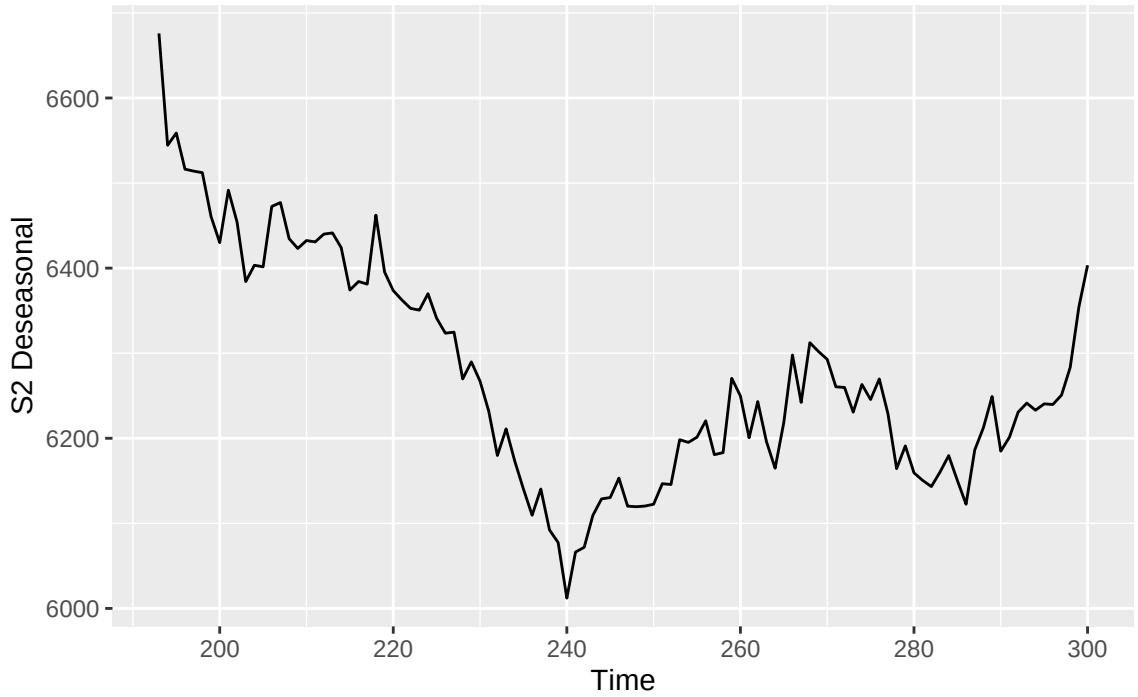


Figure 3: Deseasonal - Italian Banana Demand

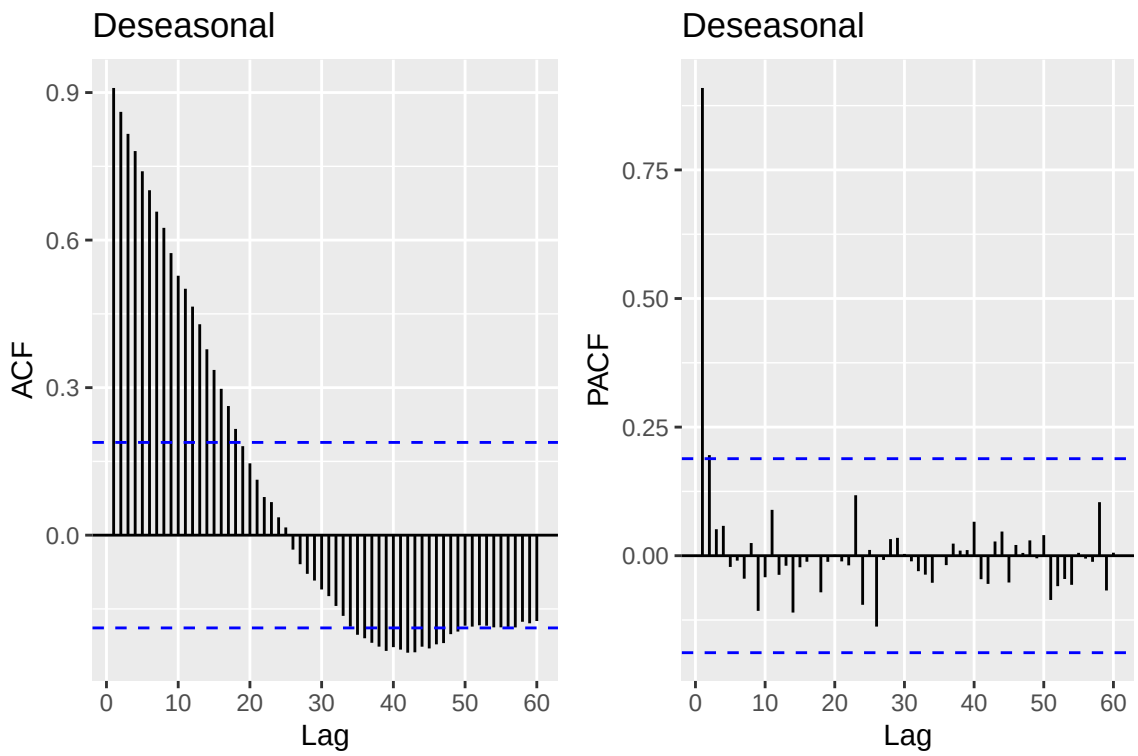


Figure 4: Deseasonal ACF/PACF - Italian Banana Demand

The ACF/PACF plots above have confirmed our initial perception and indicate a possible model an auto-regressive model of $order = 1$, integrated of $order = 1$.

```
auto.arima(S2)
```

```
## Series: S2
## ARIMA(0,1,2)
##
## Coefficients:
##          ma1      ma2
##       -0.420  -0.3652
## s.e.   0.085   0.0809
##
## sigma^2 estimated as 14942:  log likelihood=-702.83
## AIC=1411.67   AICc=1411.89   BIC=1419.85
```

```
arima(S2, order = c(0,1,2))
```

```
##
## Call:
## arima(x = S2, order = c(0, 1, 2))
##
## Coefficients:
##          ma1      ma2
##       -0.420  -0.3652
## s.e.   0.085   0.0809
##
## sigma^2 estimated as 14677:  log likelihood = -702.83,  aic = 1411.67
```

```
arima(S2, order = c(1, 2, 0), seasonal = list(order = c(0, 1, 0), period = 12))
```

```
##
## Call:
## arima(x = S2, order = c(1, 2, 0), seasonal = list(order = c(0, 1, 0), period = 12))
##
## Coefficients:
##           ar1
##       -0.5701
## s.e.   0.0866
##
## sigma^2 estimated as 3980:  log likelihood = -559.42,  aic = 1122.83
```

```
arima(diff(S2), order = c(1, 2, 0), seasonal = list(order = c(0, 1, 0), period = 12))
```

```
##
## Call:
## arima(x = diff(S2), order = c(1, 2, 0), seasonal = list(order = c(0, 1, 0),
##     period = 12))
##
## Coefficients:
##           ar1
##       -0.7031
## s.e.   0.0719
##
## sigma^2 estimated as 8710:  log likelihood = -592.76,  aic = 1189.53
```

The Model

The model adjusted is an *Integrated Auto-Regressive model* of first order, with presence of seasonality of first order, with period 12.

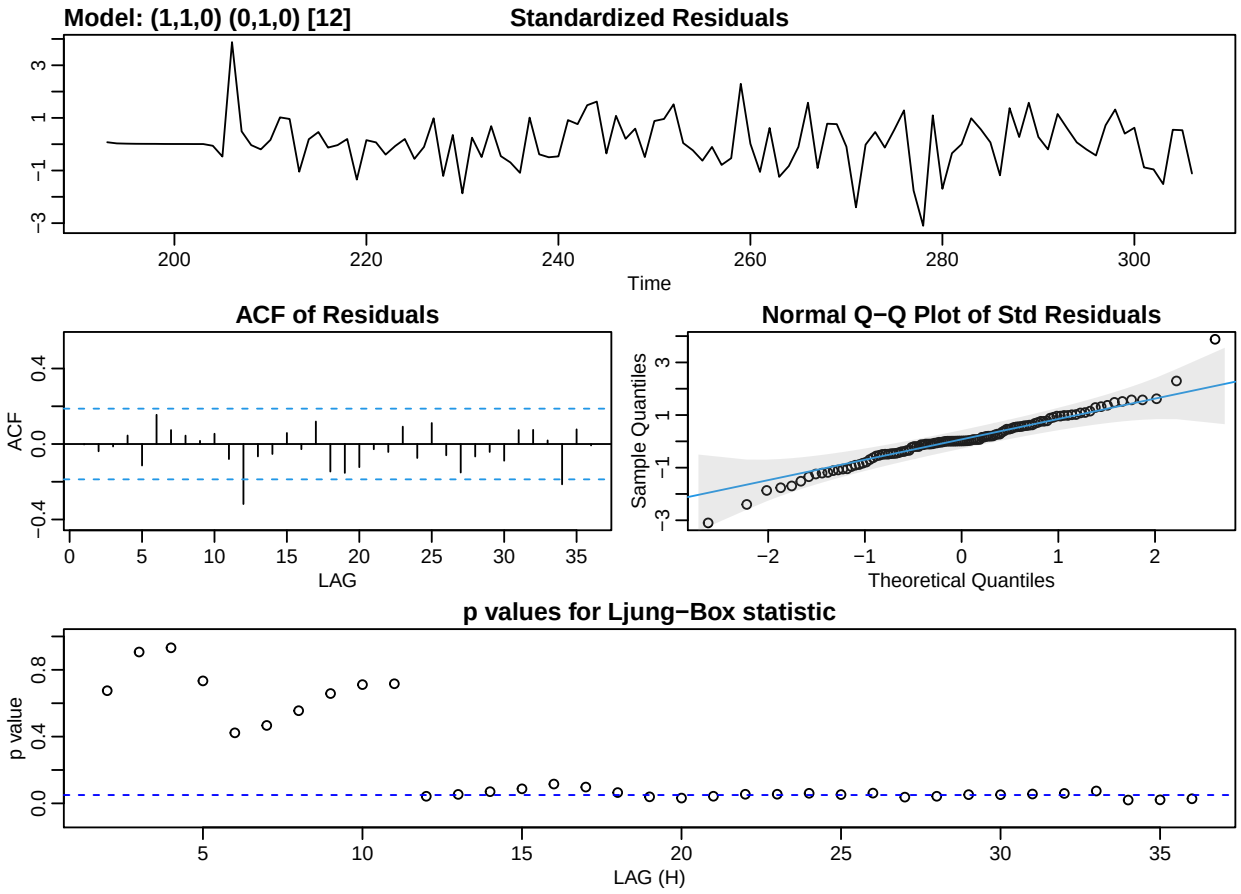
In other words, the best fit for the data to provide the desired forecast is $SARIMA[1,1,0] \times [0,1,0]_{12}$, integrated with ordinary differencing of $order = 1$ and seasonal differencing of $order = 1$, with $period = 12$.

```
##
## Call:
## arima(x = diff(S2, 12), order = c(1, 1, 0))
##
## Coefficients:
##           ar1
##       -0.1741
## s.e.   0.1067
##
## sigma^2 estimated as 2578:  log likelihood = -539.99,  aic = 1083.98
##
## Training set error measures:
##                     ME     RMSE     MAE      MPE     MAPE      MASE       ACF1
## Training set 3.638034 50.52291 37.8038 1.007311 84.74378 0.9815383 0.01406783
```

```
##
## AR1: -0.1740608  and Sigma^2: 2577.836
```

The diagnostics of the adjusted model shows residuals approximate independently distributed and the ACF of residuals doesn't show significant correlation. The normality of standard residuals is approximately preserved in the *Q-Q Plot* and Ljung-Box statistic confirms the residuals for lags 1-12 are independent which fits our needs in this study.

```
## initial  value 3.864428
## iter   2 value 3.851649
## iter   3 value 3.851308
## iter   4 value 3.851308
## iter   4 value 3.851308
## final  value 3.851308
## converged
## initial  value 3.927760
## iter   2 value 3.927502
## iter   3 value 3.927494
## iter   3 value 3.927494
## iter   3 value 3.927494
## final  value 3.927494
## converged
```



Our model can then be written as:

$$(1 + 0.1740608B)\nabla_{12}^{1}\nabla^{1}X_t = W_t$$

with $W_t \sim WN(0, \sigma_W^2)$ and $\hat{\sigma}_W^2 = 2577.84$.

Forecasts

The model in the previous section produced the following forecast for the next 12 observations with the confidence interval around it.

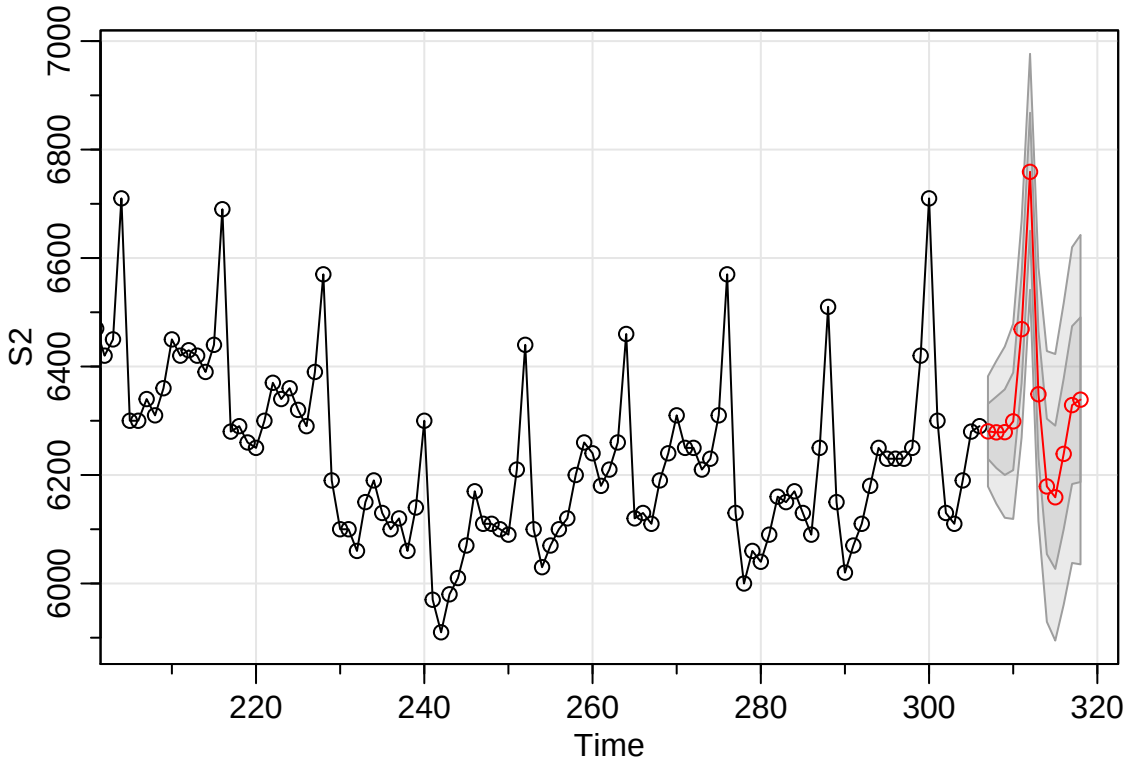


Figure 5: Forecast with 95% C.I. - Italian Banana Demand

In the following table we have the forecast obtained from this simulation.

Table 1: Forecast for next 12-observations

Obs	Fcst
1	6280.4
2	6278.6
3	6278.9
4	6298.9
5	6468.9
6	6758.9
7	6348.9
8	6178.9
9	6158.9
10	6238.9
11	6328.9
12	6338.9

As requested, the graphical representation of the forecast.

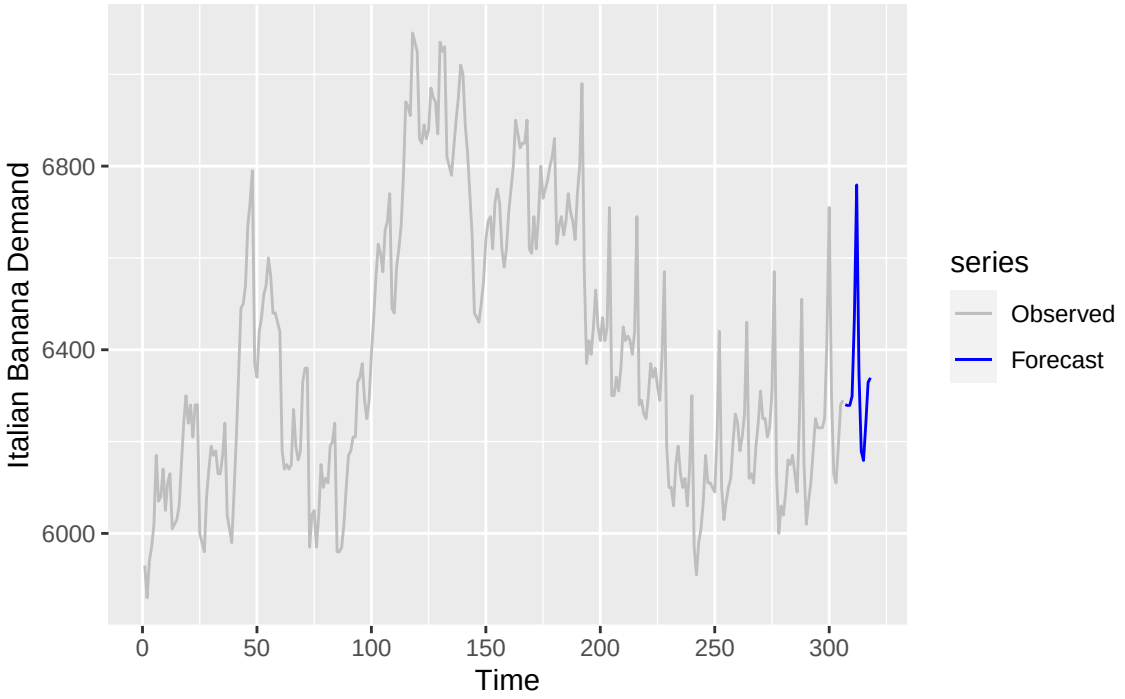


Figure 6: Forecast - Italian Banana Demand

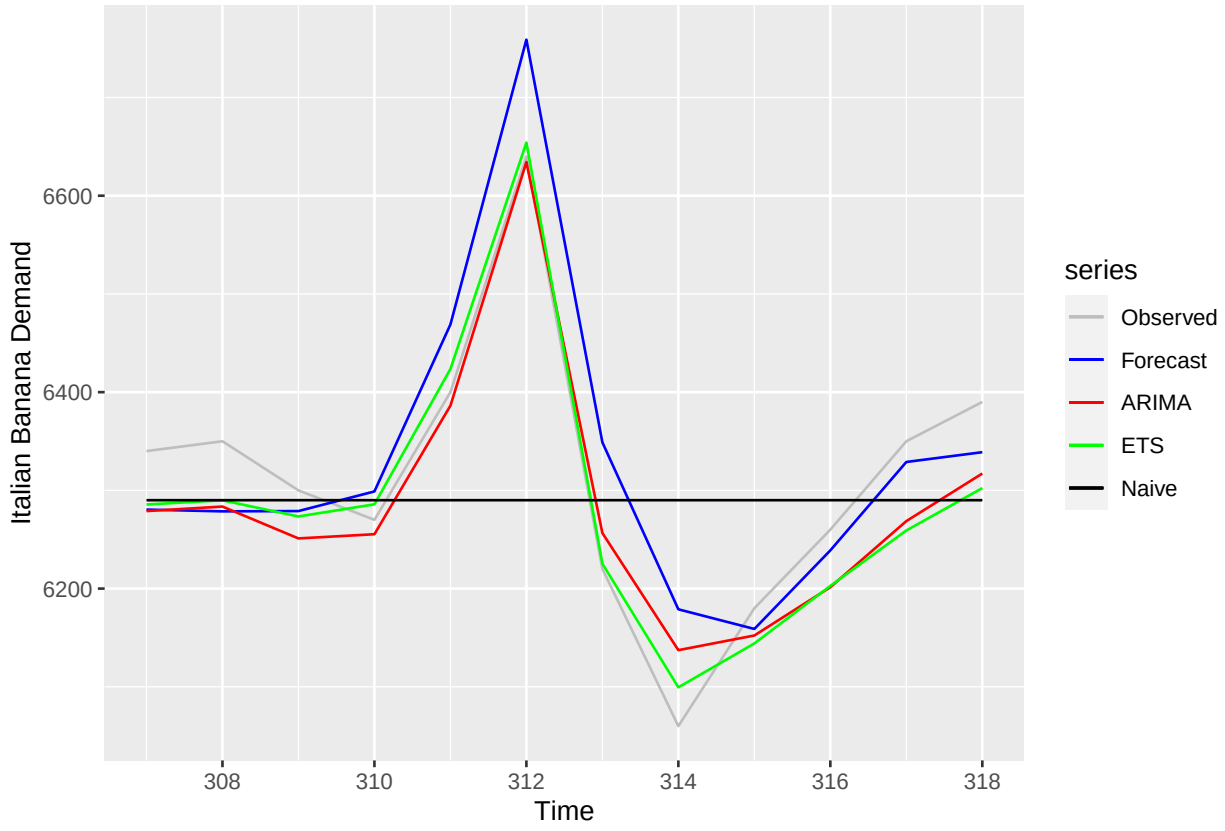


Figure 7: FCST vs Actuals - Italian Banana Demand

Final Considerations

Despite of the fact that the present forecast has been obtained with the appropriate techniques of Time Series Analysis and all efforts were done to provide the best estimate possible, every forecast is subject to imprecision originated from diverse sources such as goodness of fit of the model used to produce them, error of measure, hidden variability not captured by the model, among others.

In this sense, it is strongly recommended to make use of such predictions with parsimony and considering its 95% Confidence Interval in order to mitigate eventual unpleasant results, specially those with financial impacts.

Future enhancements can be done in this model as more data is available to refine and fine-tune the predictions produced.

Appendix - R-Code

```
library(tidyverse)
library(astsa)
library(forecast)
library(grid)

my_data = read_csv("1006508566-A2.csv")

SLabel <- colnames(my_data)[2]

colnames(my_data) <- c("Time", "Series")

S <- ts(my_data$Series)

# Define and Cuts the series into 02 parts
SCut <- 192
N <- length(S)

autoplot(S, ylab=SLabel)+
  geom_vline(xintercept=SCut, color="blue", linetype = 2)+
  annotate("text", x = 25, y = 7000, label = "S1 Sub-serie")+
  annotate("text", x = 250, y = 7000, label = "S2 Sub-serie")

# Define function to plot ACF/PACF of a given series
pltACF_PACF <- function (S, Desc = "Series", MLag = 20 ) {
  p1 <- ggAcf(S, lag.max = MLag)+
    ggtitle(Desc)
  p2 <- ggPacf(S, lag.max = MLag)+
    ggtitle(Desc)

  gridExtra::grid.arrange(p1, p2, nrow = 1)
}

pltACF_PACF(S, "Observed", 60)

S1 <- window(S, start = 1, end = SCut)
S2 <- window(S, start = SCut + 1, end = N)

pltACF_PACF(S1, paste0("S2-[1:",SCut,"]"), 60)
pltACF_PACF(S2, paste0("S2-[",SCut+1,":",N,"]"), 60)

## SERIES DECOMPOSITION - Multiplicative

# Step 1 - Trend treatment
m <- 12
S2Enl <- window(S, start = SCut-11, end = N)
THat <- S2Enl %>%
  stats::filter(c(.5, rep(1,(m-1)), .5)/m, method = "convolution", sides = 1)

# Calculating the Detrended Series
Det_S <- S2/THat

# Step 2 - Seasonality treatment
```

```r
# Set initial values to calculate seasonality
n <- length(S2)
nper <- n/m

# Vector containing the indexes to be used to calculate averages of each period
v <- array(dim=c(m, nper))
MPer <- vector()

# Calculates the indexes for whole series
for (i in 1:m)
  v[i,] <- (0:(nper-1)*m)+1+(i-1)

# Calculates the seasonality
for (i in 1:m)
  MPer[i] <- mean(Det_S[v[i,1:nper]], na.rm = TRUE)

# Replicate to all series
SHat <- ts(rep(MPer, nper), start=head(time(S2), 1), frequency=frequency(S2))

# Calculates the Deseasonall Series
Des_S <- S2/SHat

# Calculates the Detrended + DeSeasonal series (= Remainder)
Dets_S <- Det_S/SHat

# Step 3 Calculate the Remainder of the series
RHat <- S2/(SHat*THat)

# Plot the composed series obtained
p1 <- autoplot(S2, ylab="observed", xlab=NULL)
p2 <- autoplot(THat, ylab="trend")
p3 <- autoplot(SHat, ylab="seasonal", xlab=NULL)
p4 <- autoplot(RHat, ylab="random")

grid.newpage()
grid.draw(rbind(ggplotGrob(p1), ggplotGrob(p2), size = "last"))
grid.newpage()
grid.draw(rbind(ggplotGrob(p3), ggplotGrob(p4), size = "last"))

autoplot(Des_S, ylab="S2 Deseasonal")

pltACF_PACF(Des_S, "Deseasonal", 60)

# Define a structure for 95% C.I. for each coefficient
CICoef <- data.frame (
  Coef1 = data.frame (
    upper = 0, lower = 0
  ),
  Coef2 = data.frame (
    upper = 0, lower = 0
  ),
  Coef3 = data.frame (
    upper = 0, lower = 0
```

```
  )
)

ml.fit  <-  arima(diff(S2, 12), order=c(1,1,0))
summary(ml.fit)
cat("\nAR1:", ml.fit$coef, " and Sigma^2:", ml.fit$sigma2)

SECoef <- sqrt(diag(ml.fit$var.coef))  # SE's of coefficients

CICoef$Coef1.upper <- ml.fit$coef[1]+1.96*SECoef[1]
CICoef$Coef1.lower <- ml.fit$coef[1]-1.96*SECoef[1]

# Fit Model SARIMA[1,1,0]x[0,1,0]_12

M4 <- sarima(S2, 1, 1, 0, P=0, D=1, Q=0, S=12, no.constant=FALSE)

# Calculates the prediction Values
for.ml <- predict(ml.fit, n.ahead = 12)

# Creates predictions for another 12 observations
V <- window(S2, start=N-11, end = N)
NewS2 <- ts(c(S2,for.ml$pred+as.vector(V)), start=start(S2), frequency=frequency(S2))

# Forecast 12-observation with the chosen model
fore4 <- sarima.for(S2, n.ahead = 12, p = 1, d = 1, q = 0,
                    P=0, D=1, Q=0, S=12, no.constant=FALSE)

# Stores the Best Fit, including Forecast which matches with ML Prediction
BestFit <- fore4

prtTable <- data.frame (
  Obs = 1:12,
  Fcst = format(BestFit$pred, digits=5, nsmall = 1)
)

# Format Intervals for printing
kableExtra::kable(prtTable, "latex", booktabs = TRUE, caption = "Forecast for next 12-observations")

# Save CSV Data with Forecast
write(format(BestFit$pred, justify="left", digits=5, nsmall = 1), file = "1006508566.csv")

# Plot Requested
autoplot(S, series="Observed")+
  autolayer(window(NewS2, start=307, end=318), series="Forecast") +
  xlab("Time") + ylab(SLabel) +
  scale_colour_manual(values=c("gray","blue"),
             breaks=c("Observed","Forecast"))
```