UCLouvain
Faculté des sciences

Université catholique de Louvain
Louvain School of Statistics

LSTAT2170 - Time series

Final Project

Lionel Lamy
1294-1700

Florida Natural Gas Deliveries Analysis
May 9, 2021

Contents

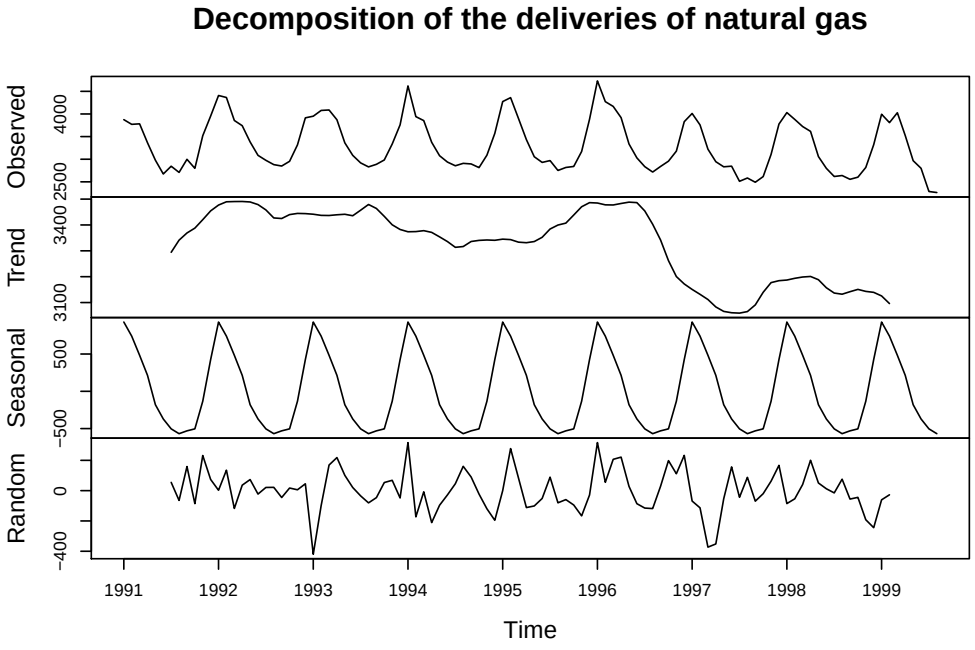
1 Introduction

In this project one will focus on the analysis of real data. These data are the U.S. Natural Gas State Data and concern the monthly quantity of natural gas delivered to residential and commercial consumers (excluding vehicle fuel) in Florida. Aggregated on a monthly basis, the data are presented in millions of cubic feet (MMcf[1]) and cover the period from January 1991 to August 1999.

The beginning of this report will start with a first visual discovery of the dataset. Then, a set of transformations will be applied to stabilize the variance, remove possible trends and seasonality. Next, one will analyze the autocorrelation and partial autocorrelation functions in order to have a first intuition on the type of model to fit. Following this, several methods such as a significance test of the coefficients, an analysis of the residuals by a Ljung-Box test or the evaluation of the predictive capacity "on sample" will allow to establish which model would be the most appropriate for the data. The final objective is to be able to give a prediction interval for future values over roughly one year.

2 Data discovery

Considering the data to be an additive model $Y_t = T_t + S_t + \epsilon_t$, one can have a first insight by decomposing the series into trend (T), seasonal (S) and random (ϵ) pieces. A plot containing the undecomposed data is available in the appendix.

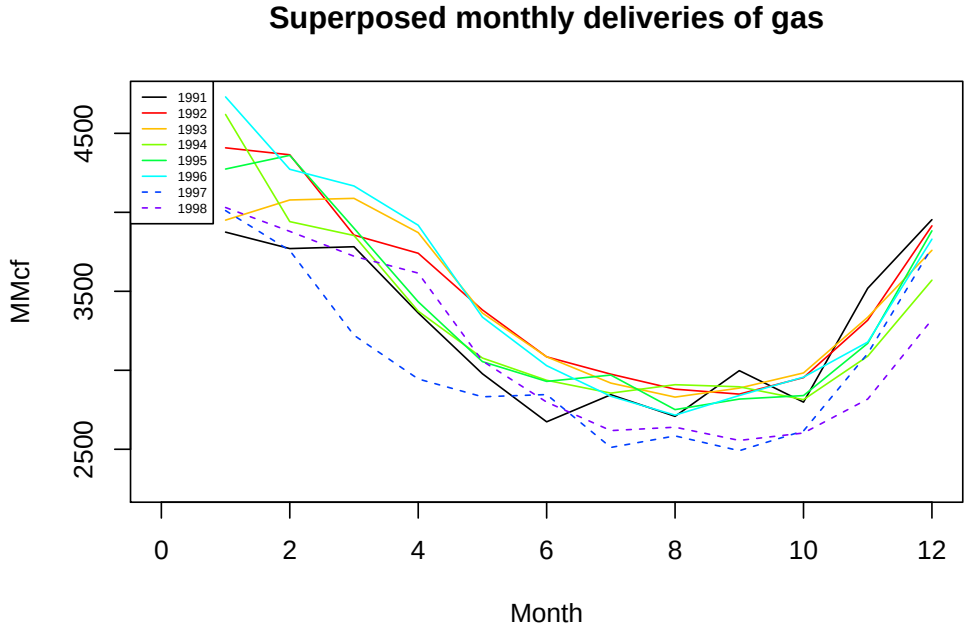


The first obvious observation one can make by looking at just the first line is one is dealing with seasonal data. Indeed, the third row shows an almost perfect seasonality with maximum values at the beginning of each year and minimum values towards the middle. These results are hardly surprising given the nature of the data. Indeed, it seems normal that a greater quantity of gas is used during winter and that gas consumption decreases during summer. Looking at the second line of the graph, one see that the data

[1] "Mcf" means 1,000 cubic feet of natural gas; "MMcf" means 1,000 Mcf.

do not really seem to vary except in 1996-1997 when a decrease is noticeable. A closer look at the first line shows this phenomenon.

Another interesting way to present the data is to display the evolution of the deliveries by stacking the years line by line. The outcome is fortunately the same. One notice a strong seasonality (since the lines all follow the same pattern) as well as a small decrease that seems to start from 1997 (colored dashed lines). Moreover, the lines seems to remain relatively close to each other and there is no drastic change in variance over time.



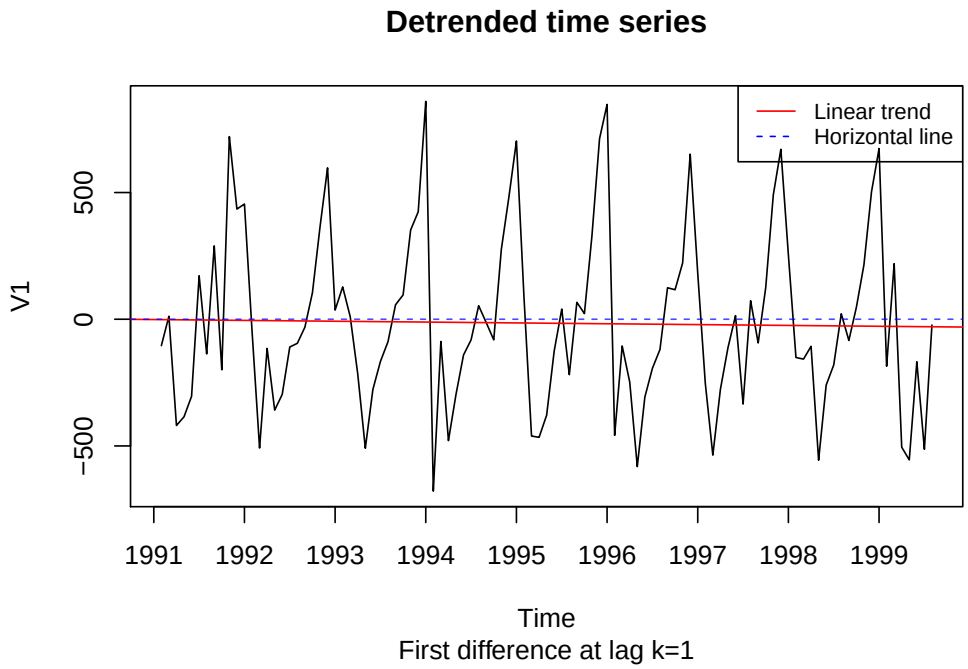
3 Box-Jenkins

The previous observations do not require stabilizing the variance by any transformation of the data. On the other hand, as one aim to examine the correlation structure of the residuals, (weak) stationarity need to achieved. Meaning that it is necessary to deseasonalize and delinearize the data in order to go further with the visual analysis.

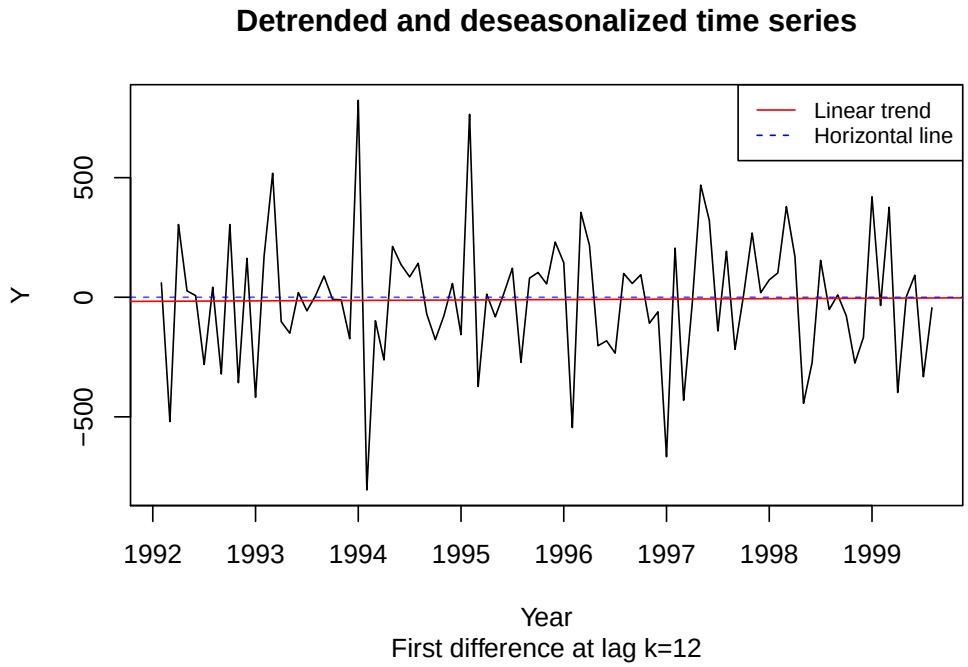
3.1 Deseasonalize and detrend

For this purpose, the method of (iterated) differences will be used. Considering that Y_t is the time series at period t, then the first difference at lag k is $\bigtriangledown_k Y_t := Y_t - Y_{t-k}$. As a first step, the latter operation is applied on the data at lag 1 to remove the linear trend. This process is next repeated at a lag equal to the periodicity, ie. the value 12 since the data is collected monthly and the seasonality is annual.

The resulting time series is therefore shifted by $k = 13$ periods.



One observe in the figure above that once the data is differentiated, the linear trend is much less pronounced but there still is a visible periodicity. This particularity seems to be corrected on the second graph below as the time series now appears to be random. Even if by plotting a linear regression (in red) one can see that the trend is not quite equal to zero, the slope is flat enough to consider the time series to be (weakly) stationary.



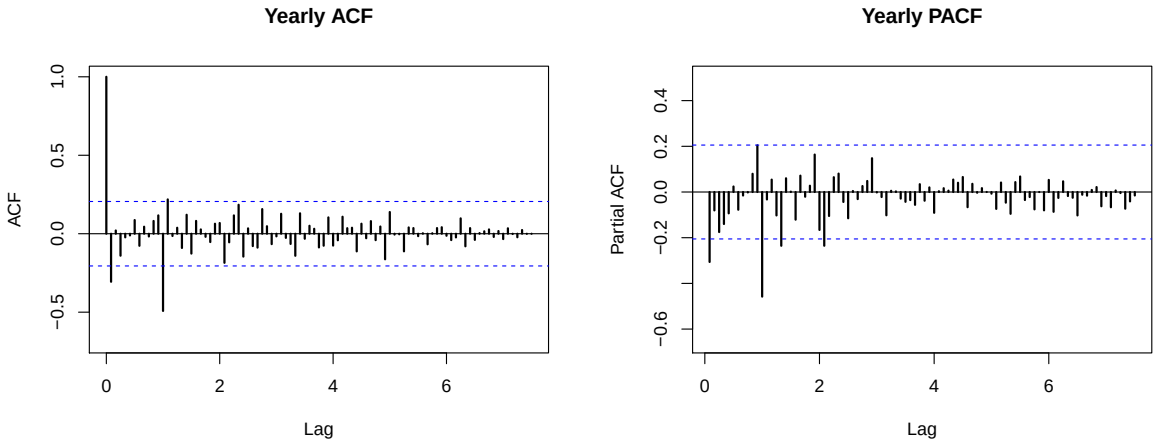
3.2 ACF and PACF

Now that the data is differenced and assumed to be stationary, one can examine the correlation structure and have an intuition of the values to feed into the model. The auto-correlation and partial auto-correlation functions will serve this purpose.

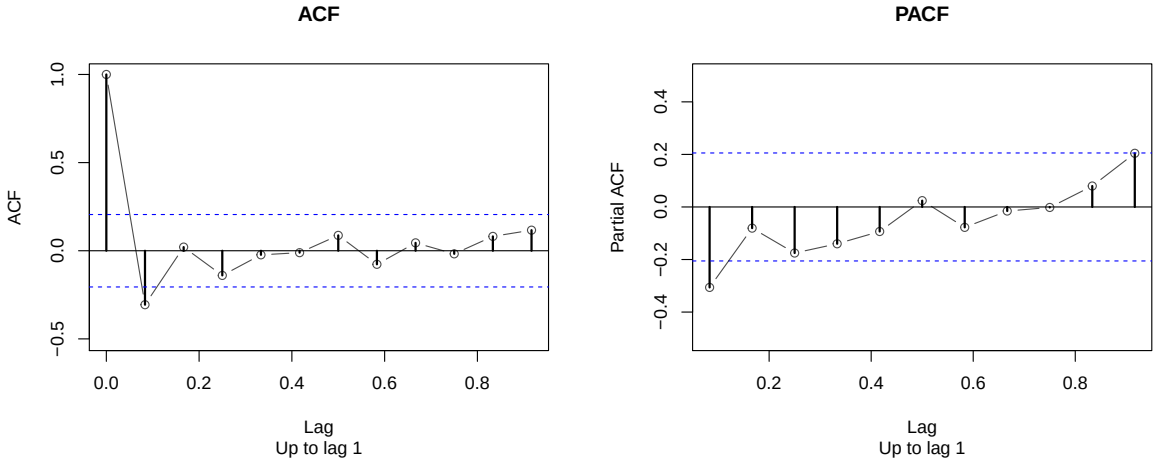
To recall, the former describes how well the present value is correlated with its past values while the latter does the same but removing the linear dependence at intermediate lags. Both assume the data to be stationary and the lagged values to act like white noise. Thus, confidence intervals are constructed assuming a normal distribution, with Z_α the normal quantile at level α and T the length of the data :

$$CI = \left[\frac{Z_{\alpha/2}}{\sqrt{T}} \; ; \; \frac{Z_{1-\alpha/2}}{\sqrt{T}} \right]$$

In the figures below, one notice that both ACF and PACF tend towards zero. The values fall below the confidence interval after lag 1 for the ACF and lag 2 (or 1 according to the degree of tolerance) for the PACF. The simultaneous look on the ACF and PACF might suggest an ARMA(0,1) or ARMA(1,1) on a yearly basis.



Although kind of predictable, the presence of significant correlations at lag 1 is nevertheless intriguing. The transformations performed have only removed a deterministic element of seasonality and a stochastic component could indeed remain. Analyzing the ACF and PACF on a monthly basis, i.e. between lag [0 and 1[also reveal a significant value at first lag. Note that in the case of the ACF, the value at lag 0 is always 1. On a monthly basis, one might thus move with an ARMA(1,1).



4 Model selection

In consideration of the aforementioned elements, it was decided to proceed with an SARIMA model. Before continuing, one should yet provide a small definition. An SARIMA model is an Autoregressive Integrated Moving Average model that supports the direct modeling of a Seasonal component (S). Meaning we have 4 more elements (P, D, Q) and s, where P is the seasonal AR order, D the seasonal difference order, Q the seasonal MA order and s the seasonal period.

Putting them together, one obtain a model called SARIMA$(p, d, q)\times(P, D, Q)_s$ if the time series X_t can be transformed into a stationary series without trend or seasonality.

$$Y_t = \bigtriangledown^d \bigtriangledown_s^D X_t = (1-B)^d \, (1-B^s)^D \, X_t$$

And where the latter can be modelised in the form of a stationary ARMA process

$$\phi(B) \; \Phi(B^s) \; Y_t = \theta(B) \; \Theta(B^s) \; \epsilon_t$$

where B is the backshift operator, $\bigtriangledown$ the difference operator and where $\phi(z)$, $\Phi(z)$, $\theta(z)$ and $\Theta(z)$ are the generating polynomial of respectively, an AR(p), AR(P), MA(q) and MA(Q) process.

4.1 Selection

Following the visual analysis of the previous section, it is expected to obtain SARIMA models of orders up to 1. However, to figure out which model would be the best one will iterate over all possible SARIMA models with parameters value up to 2.

To evaluate the effectiveness of the different models, the Akaike information criterion (AIC) will primarily be used. However, since the AIC tends to overestimate the number of parameters needed, the Bayesian information criterion (BIC), which penalizes a bit more the number of parameters, will provide a secondary information.

The results below describe the top 3 models found, from best to worst. First, one notice via the AICR[2] that the models have very close results. The winning model has an AIC of 1229, a BIC of 1241 and is composed of 4 parameters. The second one has a higher AIC of only 0.064 and a lower BIC of 2.447 (relative to the first one) but has only 3 parameters. The last model is somewhat less interesting since the increase in AIC is more important and it has as much parameters as the best one.

```
TOP 3 AIC || MODEL : (p,d,q)x(P,D,Q)[s]

(1,1,1)x(1,1,1)[12] || AIC: 1228.804 | AICR: 0.000 | BIC: 1241.359 || P: 4
(1,1,1)x(0,1,1)[12] || AIC: 1228.868 | AICR: 0.064 | BIC: 1238.912 || P: 3
(2,1,1)x(0,1,1)[12] || AIC: 1230.100 | AICR: 1.296 | BIC: 1242.654 || P: 4
```

As an additional information, the log-likelihood ($\mathcal{L}$) of the models are respectively : -609.402, -610.434 and -610.05 and the AIC is computed via $-2\mathcal{L} + 2\,(P+1)$.

[2] Akaike information criterion relative to the best model.

5 Models comparison and validation

Now that a selection of models is established, one need to check if they are valid and correctly fitted. To do so, one will first perform an univariate and two-sided significance test (based on normal approximations) of the coefficients. Then, evaluate the predictive ability "on sample" and finally, after the choice of the final model, do an analysis of the residuals by a Portmanteau (Ljung-box) test.

5.1 Coefficients

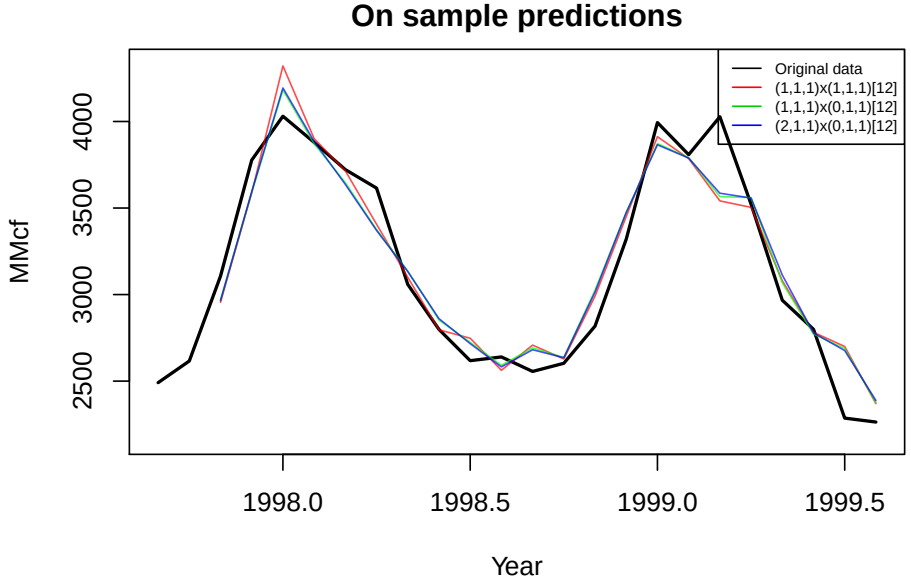
In the same order as displayed in the previous section:

```
          ar1           ma1           sar1          sma1
"(**) 0.0146" "(***) 0.0000"   "(x) 0.1531"  "(**) 0.0018"

          ar1           ma1           sma1
"(**) 0.0333" "(***) 0.0000"  "(***) 2e-04"

          ar1           ar2           ma1           sma1
"(***) 6e-04"   "(x) 0.3609" "(***) 0.0000"  "(***) 1e-04"
```

Thanks to the indicators in parentheses, it is easy to see which coefficients are significant and to what degree. First, for the model with the lowest AIC (1) one cannot reject the hypothesis that the coefficient sar1 (Φ_1) is statistically different from zero, even with an $\alpha = 0.10$. The same is observed for the third model with ar2 (ϕ_2).

5.2 Predictive power

Lets now continue with a comparison of the predictive power of each model. To this end, each model will be fed with $Y_{1:t}$ where $Y_{1:t}$ is the set of data up to t. At each step, one step ahead will be predicted (ie. $\hat{Y}_{t+1}$) and the operation repeated with t starting at 80% of the series and until the end is reached. In other terms, $t = 0.8T$ to $T-1$. Finally, one can calculate the MSE of each models by comparing with the last 20% observed data.



Although not perfect, the models manage to follow the global trend pretty well but models 2 (green) and 3 (blue) seem to perform slightly better than the first one (red). In fact, their MSE are lower with respectively 30142 and 29607, what corresponds to a decrease of more than 8% compared to 32830.

Model 1	Model 2	Model 3
32 830	30 142	29 607

Those results lead to choose the second model. This one being the most parsimonious, it only loses 0.064 points of AIC, has only 1.8% more MSE than the third model and in view of the significance test seems to be the most relevant. The latter can mathematically be written as :

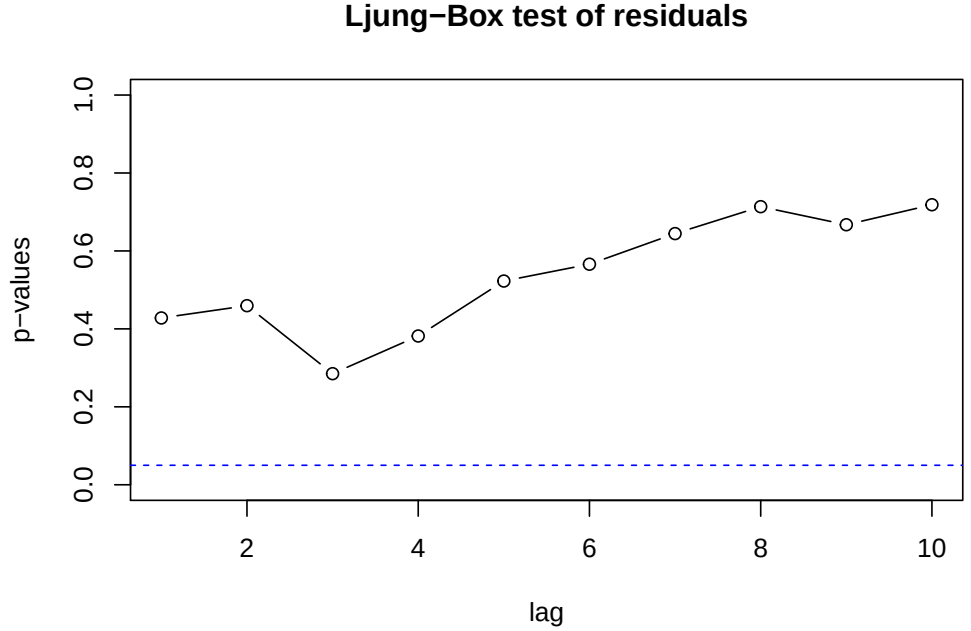
$$\text{S-ARIMA}(1,1,1)\times(0,1,1)_{12}$$
$$(1-\phi_1 B)(1-B)^1(1-B^{12})^1\ Y_t = (1+\theta_1 B)(1+\Theta_1 B^{12})\ \epsilon_t$$

with coefficients, $\phi_1 = 0.453$, $\theta_1 = -0.832$, and $\Theta_1 = -1.000$.

5.3 Ljung-Box

As a last test, one will check the autocorrelation of the residuals of the model via a portmanteau test, more specifically a Ljung-box test. One test the null hypothesis that the residuals are not different from white noise, up to lag $K = \sqrt{T}$.

$$H_0 : \rho_\epsilon(1) = ... = \rho_\epsilon(K) = 0$$
$$H_1 : \rho_\epsilon(1) = ... = \rho_\epsilon(K) \neq 0$$



As the p-values are all well above the threshold, one cannot reject the null hypothesis. This is convenient because it means that there is no correlation in the residuals. The same test was performed for the residuals squared and the result is similar. Also, a more complete figure which includes the ACF of the residuals is available in the appendix.

6 Predictions

[TODO : Introduce the section and the exponential smoothing]

Exponential smoothing methods produce forecasts where recent observations have an exponentially greater weight than past ones. In other words, the more recent the observation, the higher the associated weight. Moreover, they can be used in a wide range of time series without any particular conditions. Therefore, it would be interesting to compare one of these methods before concluding with a SARIMA model. In this case, as the data contains both trend and seasonal components, the Holt (1957) and Winters (1960) method will be used. For an additive model, with p the seasonality:

$$\widehat{Y}_{t+k} = a_t + kb_t + s_{[t-p+1+(k-1) \mod p]}$$

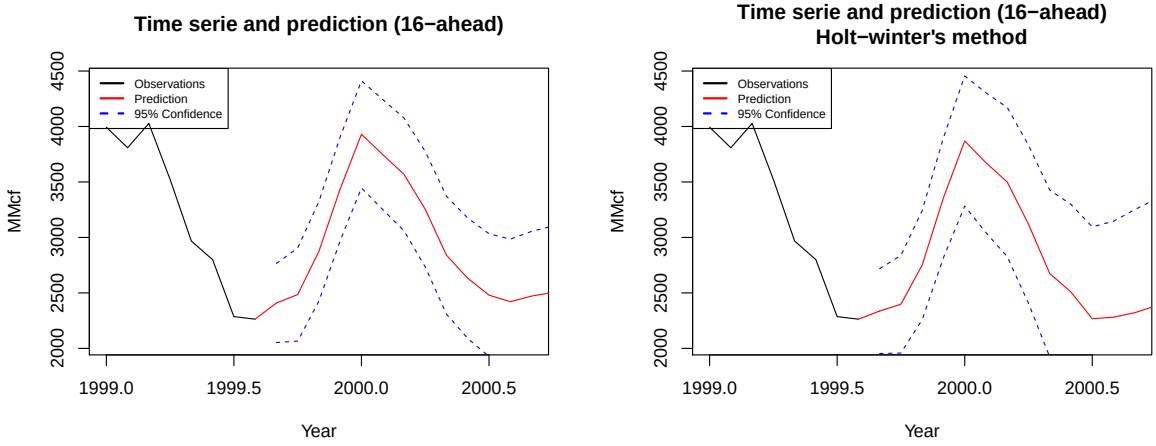
where a_t, b_t and s_t are given by:

$$\begin{aligned} a_t &= \alpha(Y_t - s_{t-p}) + (1-\alpha)(a_{t-1} + b_{t-1}) \\ b_t &= \beta(a_t - a_{t-1}) + (1-\beta)\ b_{t-1} \\ s_t &= \gamma(Y_t - a_t) + (1-\gamma)\ s_{t-p} \end{aligned}$$

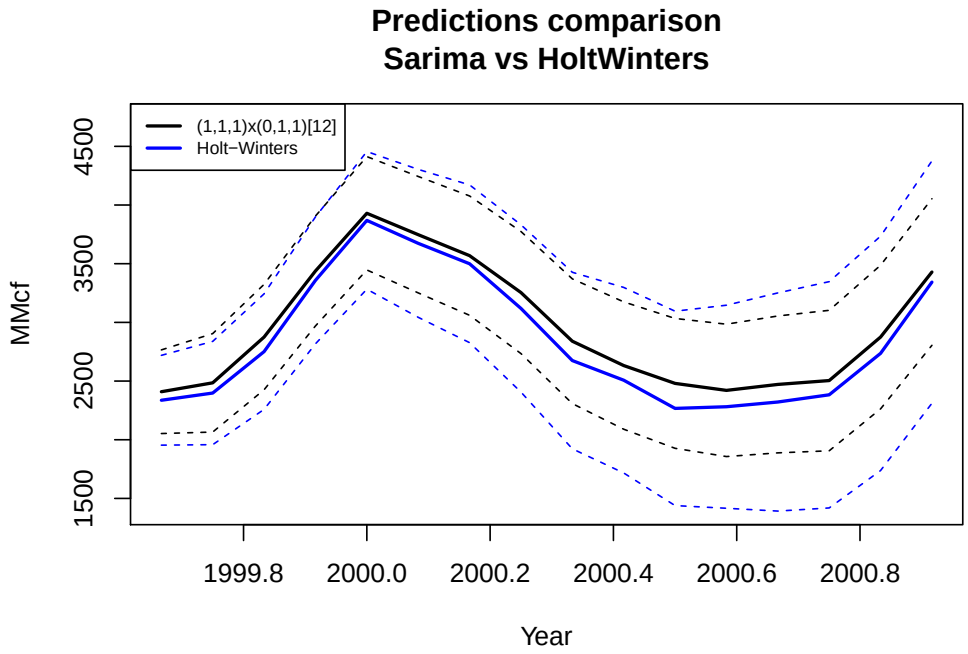
and correspond to respectively the level, the trend and the seasonal component with their smoothing parameters α, β and γ. Those parameters are determined by minimizing the squared prediction error. The full output is available in the appendix. With the gas deliveries time series, the latter are equal to:

α	β	γ
0.5598	0.0153	0.5127

[TODO : - Introduce the plots - Explain how confidence interval are constructed]



[TODO]



7 Conclusion

[TODO]

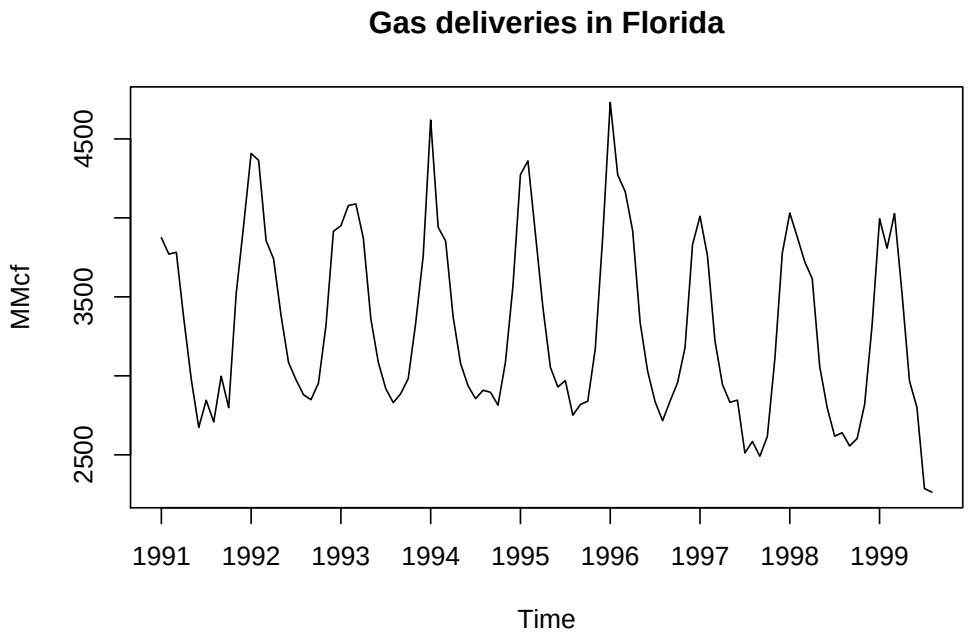
A Appendix

Note

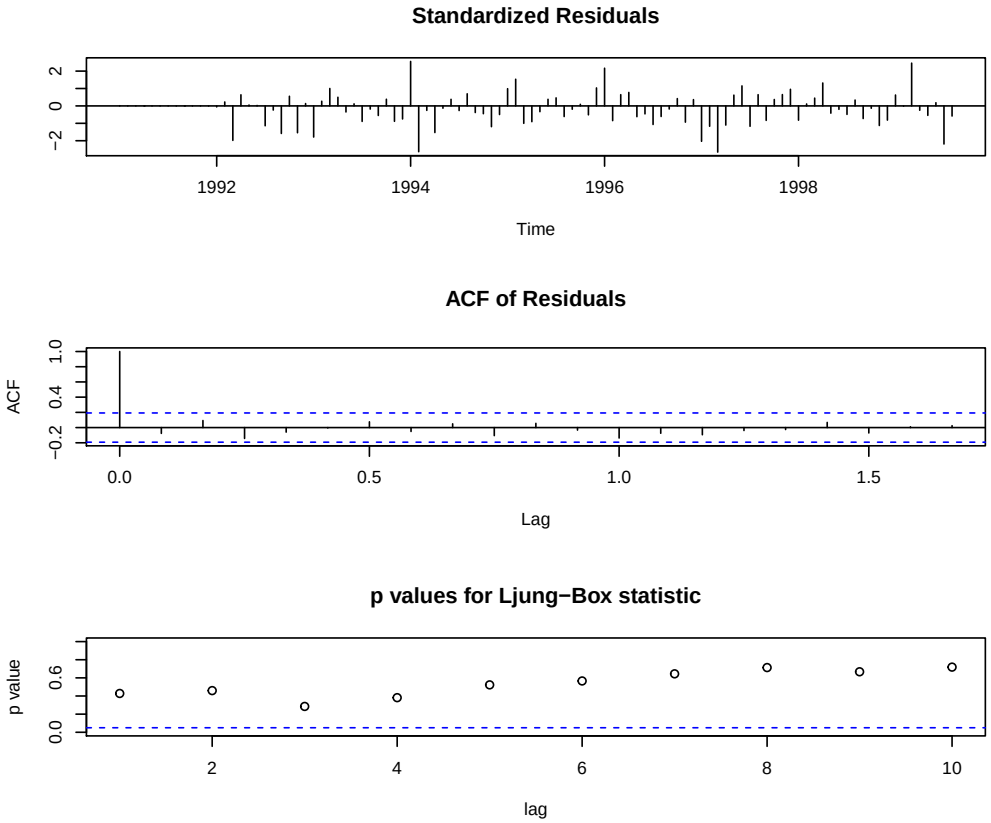
For reproducibility purposes, the complete project containing the source code, the full sized figures and the results is available on github.com/lamylio/LSTAT2170-Project.

A.1 Figures

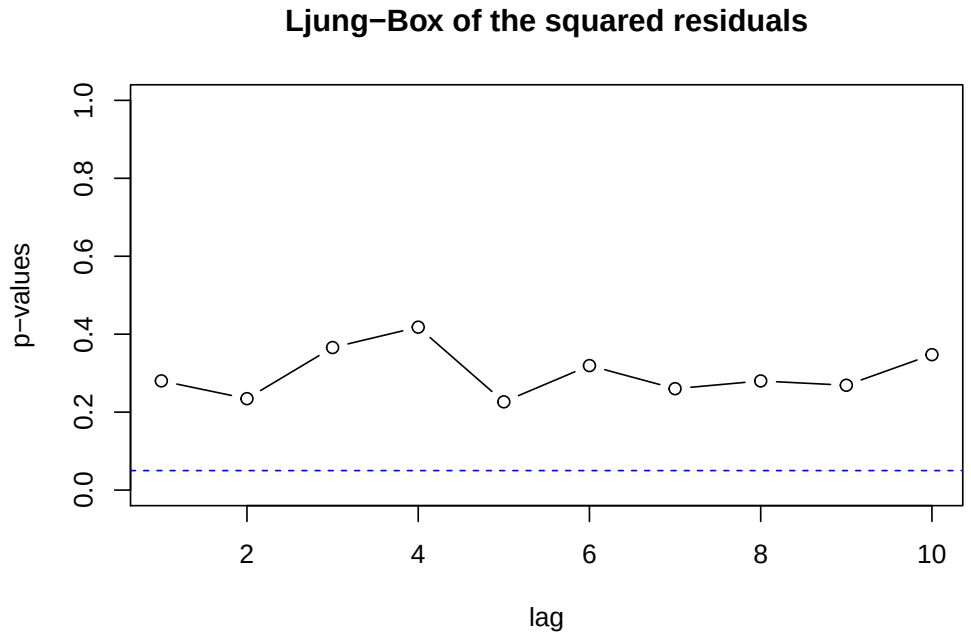
A.1.1 Plot of the data



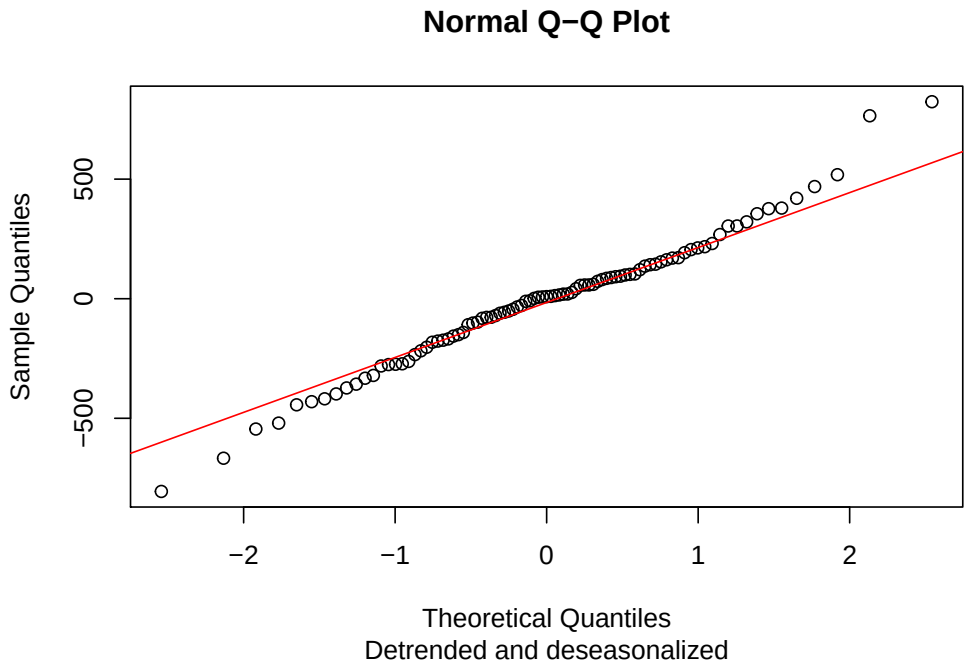
A.1.2 TSDiag (Portemanteau test)



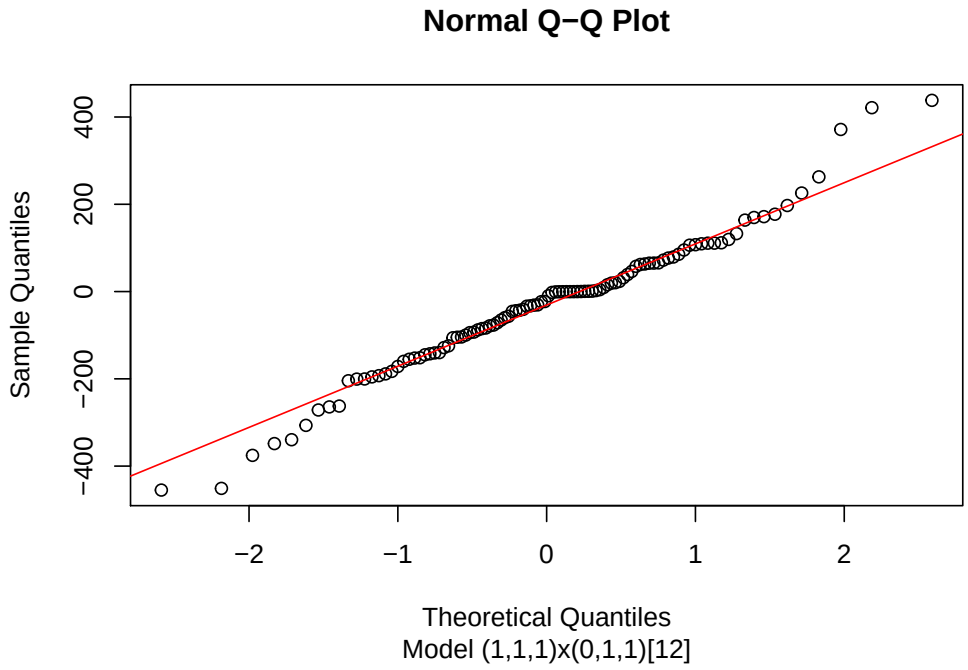
A.1.3 Ljung-Box test of the squared residuals



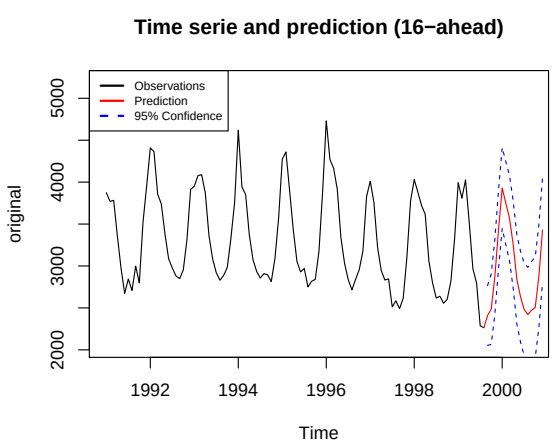
A.1.4 QQ-Plot of the differenciated time series

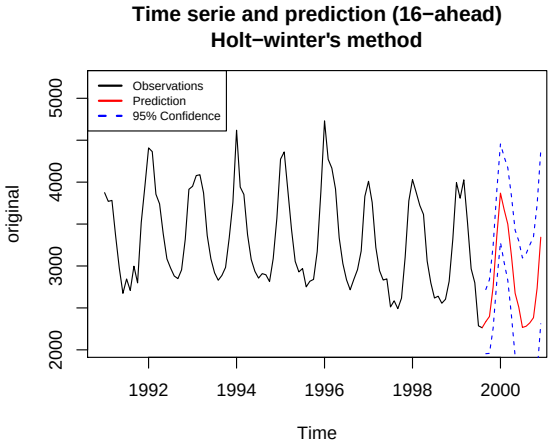


A.1.5 QQ-Plot of the residuals of model 2



A.1.6 Predictions not zoomed





A.2 Output

A.2.1 Holt-Winters

```
Holt-Winters exponential smoothing with trend and additive seasonal component.

Call:
HoltWinters(x = gas)

Smoothing parameters:
 alpha: 0.5598047
 beta : 0.01525426
 gamma: 0.5127244

Coefficients:
           [,1]
a   2943.578118
b     -1.242818
s1  -605.801246
s2  -543.146617
s3  -188.827427
s4   419.468257
s5   931.720992
s6   737.254591
s7   564.547248
s8   185.449827
s9  -257.634483
s10 -423.794430
s11 -662.768518
s12 -647.345427
```

A.3 Code

The below section is automatically generated and tidied. Visit the repository for more readability.

```r
#' Import facility functions in a separate attached environment
#' To keep our global clean

facilities <- new.env()
source("./resources/scripts/fonctionsSeriesChrono.R", local = facilities)

# Please check the github repository
source("./resources/scripts/sarima_model_selection.R", local = facilities)
source("./resources/scripts/ts_custom_plots.R", local = facilities)
source("./resources/scripts/ts_significance_test.R", local = facilities)
source("./resources/scripts/ts_on_sample_prediction.R", local = facilities)

attach(facilities, name = "facilities")

# Import the gas dataset
gas <- read.table("./resources/data/gasflorida.txt", header = F)
gas <- ts(gas, start = 1991, frequency = 12)

# Define some useful variables
gas.start <- tsp(gas)[1]
gas.end <- tsp(gas)[2]
gas.freq <- tsp(gas)[3]
gas.t <- seq(gas.start, to = gas.end, length = length(gas))
gas.xaxp <- c(floor(gas.start), floor(gas.end), floor(gas.end - gas.start))

# Plot the decomposition of the data (see ts_custom_plot.R)
plot.decompose.ts(gas, main = "Decomposition of the deliveries of natural gas",
    cex.lab = 0.9, cex.axis = 0.95, xaxp = gas.xaxp)

# Plot the superposed view of the data (see ts_custom_plot.R)
plot.superposed.ts(gas, title = "Superposed monthly deliveries of gas",
    xlab = "Month", ylab = "MMcf", dashed_thick_from = c(6, 8), xlim = c(0,
        12))

# Remove global trend
gas.1 <- diff(gas, lag = 1, differences = 1)
# Remove the seasonality using lag 12 as we have monthly data
gas.2 <- diff(gas.1, lag = 12, differences = 1)

# Removed trend
plot(gas.1, main = "Detrended time series", sub = "First difference at lag k=1",
    xaxp = gas.xaxp)
abline(reg = lm(gas.1 ~ tail(gas.t, -1)), col = "red", lty = 1, xlab = "Year",
    ylab = "Y")
abline(h = 0, col = rgb(0, 0, 1, 0.7), lty = 2)
legend("topright", legend = c("Linear trend", "Horizontal line"),
    col = c("red", "blue"), cex = 0.8, lty = c(1, 2))
```

```r
# Removed trend and seasonality
plot(gas.2, main = "Detrended and deseasonalized time series", xaxp = gas.xaxp,
    sub = "First difference at lag k=12", xlab = "Year", ylab = "Y")
abline(reg = lm(gas.2 ~ tail(gas.t, -12 - 1)), col = "red", lty = 1)
abline(h = 0, col = rgb(0, 0, 1, 0.7), lty = 2)
legend("topright", legend = c("Linear trend", "Horizontal line"),
    col = c("red", "blue"), cex = 0.8, lty = c(1, 2))

# Plot the ACF and PACF, yearly basis
plot.acf.pacf(gas.2, lag.max = length(gas.2), simplify = F, linked_by_line = F,
    titles = c("Yearly ACF", "Yearly PACF"))

# Plot the ACF and PACF, monthly basis
plot.acf.pacf(gas.2, lag.max = gas.freq - 1, simplify = F, linked_by_line = T,
    titles = c("ACF", "PACF"), sub = "Up to lag 1")

# Model comparison via AIC. (see sarima_model_selection.R)
model.1 = sarima.model.selection(gas, max.pq = c(2, 2), max.PQ = c(1,
    1), d = 1, D = 1, top = 3, return.best = T)

# Second best model (best bic)
model.2 = arima(gas, order = c(1, 1, 1), seasonal = list(order = c(0,
    1, 1), period = gas.freq))
model.3 = arima(gas, order = c(2, 1, 1), seasonal = list(order = c(0,
    1, 1), period = gas.freq))

# Test the coefficients of each model (see ts_significance_test.R)
significance.test(model.1, T)
significance.test(model.2, T)
significance.test(model.3, T)

#' MSE and predictions for the last 20% (see ts_on_sample_prediction.R)
model.1.osp = on.sample.prediction(gas, order = c(1, 1, 1), seasonal = list(order = c(1,
    1, 1), period = gas.freq))
model.2.osp = on.sample.prediction(gas, order = c(1, 1, 1), seasonal = list(order = c(0,
    1, 1), period = gas.freq))
model.3.osp = on.sample.prediction(gas, order = c(2, 1, 1), seasonal = list(order = c(0,
    1, 1), period = gas.freq))


# Adapt the predictions to the plot of the time series
models.osp.ts = function(values) ts(values, end = gas.end, frequency = gas.freq)

par(mar = c(4, 4, 2.2, 4))
# Plot the predictions
plot(models.osp.ts(tail(gas, 24)), lwd = 2, main = "On sample predictions",
    ylab = "MMcf", xlab = "Year", ylim = c(min(tail(gas, 24)) - 100,
        max(tail(gas, 24)) + 300))
lines(models.osp.ts(model.1.osp$pred), col = rgb(1, 0, 0, 0.7))
lines(models.osp.ts(model.2.osp$pred), col = rgb(0, 1, 0, 0.7))
```

```
lines(models.osp.ts(model.3.osp$pred), col = rgb(0, 0, 1, 0.7))
legend("topright", legend = c("Original data", "(1,1,1)x(1,1,1)[12]",
    "(1,1,1)x(0,1,1)[12]", "(2,1,1)x(0,1,1)[12]"), col = 1:4, lty = 1,
    cex = 0.6)

# Box.test of the residuals
plot.ljungbox(resid(model.2), floor(sqrt(length(gas))))

# Predictions with model 2 and Holt-winters, zoomed
md = plot.n.ahead.predictions(gas, model.2, n = gas.freq + 4, before = 8,
    xlab = "Year", ylab = "MMcf")
hw = plot.n.ahead.predictions(gas, model.2, n = gas.freq + 4, before = 8,
    holtwinters = T, xlab = "Year", ylab = "MMcf")

(function() {
    plot(md$pred, lwd = 2, ylim = c(min(md$pred - 1000), max(md$pred +
        800)), main = "Predictions comparison\nSarima vs HoltWinters",
        col = 1, xlab = "Year", ylab = "MMcf")
    lines(md$pred + 1.96 * md$se, lty = "dashed", col = 1)
    lines(md$pred - 1.96 * md$se, lty = "dashed", col = 1)
    lines(hw[, "fit"], col = 4, lwd = 2)
    lines(hw[, "upr"], lty = "dashed", col = 4)
    lines(hw[, "lwr"], lty = "dashed", col = 4)
    legend("topleft", legend = c("(1,1,1)x(0,1,1)[12]", "Holt-Winters"),
        col = c(1, 4), lty = c(1, 1), lwd = c(2, 2), cex = 0.7)
})()
#' ========================================

# Plot the complete dataset non decomposed
plot(gas, main = "Gas deliveries in Florida", xaxp = gas.xaxp, ylab = "MMcf")

# TSDiag : Ljung-Box test with ACF
tsdiag(model.2, gof.lag = floor(sqrt(length(gas))))

# Ljung-Box test of the squared residuals
plot.ljungbox(resid(model.2)^2, floor(sqrt(length(gas))), title = "Ljung-Box of the squared re

# QQPlot of the residuals of model 2
qqnorm(gas.2, sub = "Detrended and deseasonalized")
qqline(gas.2, col = "red")

# QQPlot of the residuals of model 2
qqnorm(resid(model.2), sub = "Model (1,1,1)x(0,1,1)[12]")
qqline(resid(model.2), col = "red")

# Predictions with model 2 and Holt-winters, zoomed
plot.n.ahead.predictions(gas, model.2, n = gas.freq + 4, before = length(gas))
plot.n.ahead.predictions(gas, model.2, n = gas.freq + 4, before = length(gas),
    holtwinters = T)

# HoltWinters coefficients and parameters
```

```
HoltWinters(gas)
```