Group Project

MAS3911 – Time Series

NAME:	*STUDENT ID:*
Lingyu Tan	180518368
Thomas Bold	170203012
Bradley Nicholson	170126526

Submitted 06/05/20

Contents

1 Introduction

The team's goal of this project is to identify and fit a suitable time series model to a set of data. This data represents the monthly total electricity consumption in a city over a ten-year period, from Jan 2006 to Dec 2015. In addition to this, the team's model will then be used to create a forecast for the period Jan 2016 to June 2016.

2 Exploratory Data Analysis

The first thing the team did was plot the data and describe the main features of the time series. Figure 1 displays the time series plot of the electricity consumption data:

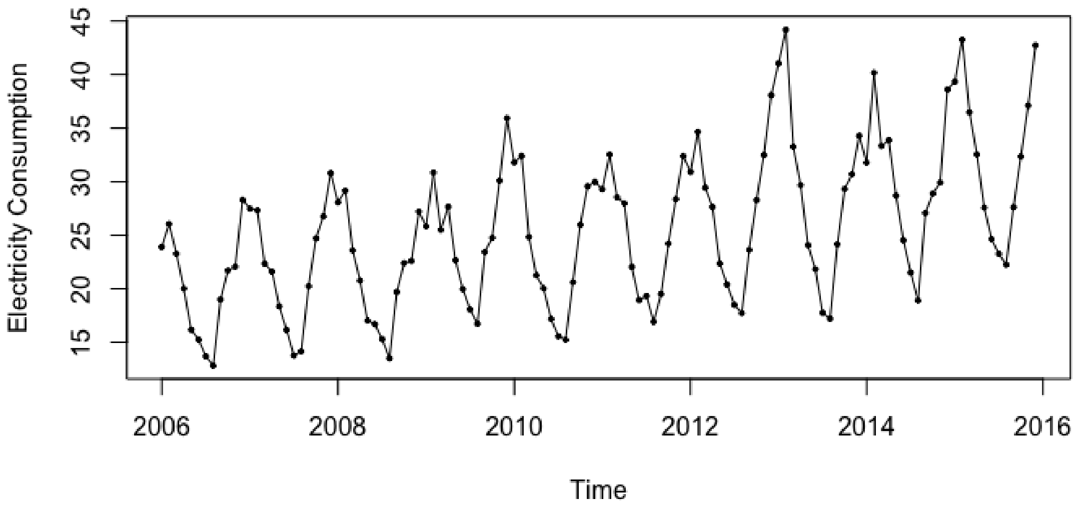

Figure 1: Time Series Plot of Electricity Consumption

From Figure 1, we have the following conclusions:

- There seems to be an upward linear trend in the long term.
- There is a fairly clear consistent seasonal effect with period $p = 12$, which seems to be more significant over time, thus a transformation might be suitable.
- As there exists trend and seasonality, outliers are not obvious in this plot.

3 Modelling

3.1 ARMA Models

3.1.1 Identification

As we find an increasing seasonality, a multiplicative model is appropriate, but we can convert to an additive model by taking logs. We can compare the transformed and untransformed model to see if R^2 changed. (See Appx.A, line.26-30) Here is the summary:

Trend model	R^2	R^2_{adj}
Untransformed model	0.9151	0.9056
Transformed model	0.9398	0.9331

Table 1: Comparison between untransformed and transformed model

We see a clear improvement in R^2 and R^2_{adj} using the transformed model. Hence, the log transformed model fits the data better.

Moreover, we can use the power transformations, which was introduced by Box and Cox (1964), to verify our choice [1]. For a given value of the parameter λ, the transformation is defined by

$$g(x) = \begin{cases} \dfrac{x^{\lambda} - 1}{\lambda} & \text{for } \lambda \neq 0 \\ log(x) & \text{for } \lambda = 0 \end{cases}$$

We can use *BoxCox.lambda*, in R package *forecast*, to evaluate the value of λ (A, ll.32-36). This gives $\lambda = 0.046$, which is quite close to 0. Therefore, a logarithmic transformation is strongly suggested.

We can again compute λ after transformation, obtaining $\lambda = 1.35$, which is close to 1 indicating we do not need to apply a transformation now.

The team now need to estimate the trend. As forecasting is needed in a later section, a linear filter is not recommended. From above we see the trend appears to be a simple linear form so we can use curve fitting to estimate it. The plot of transformed data and fitted trend is as below:

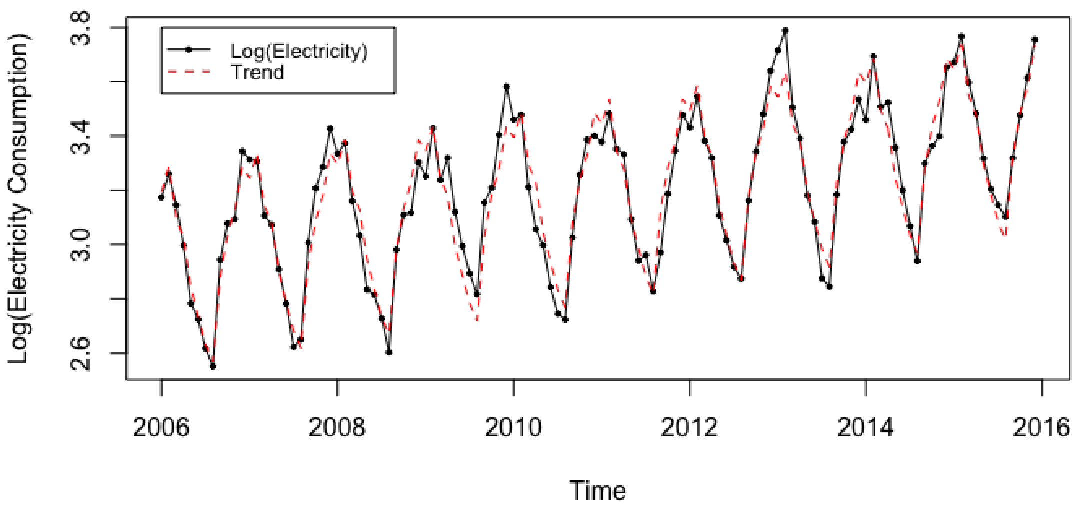

Figure 2: Time Series Plot of Transformed Data and Fitted Trend

From Figure 2 we see that the model seems to fit the data very well as exemplified by a large R^2 (0.9398) in Table 1. We have set up month as a factor and fit different trend models for different months separately, which means we have taken seasonality into consideration. The residuals of the fitted trend model should be free from trend and seasonality now. The plot of the residuals is shown below:

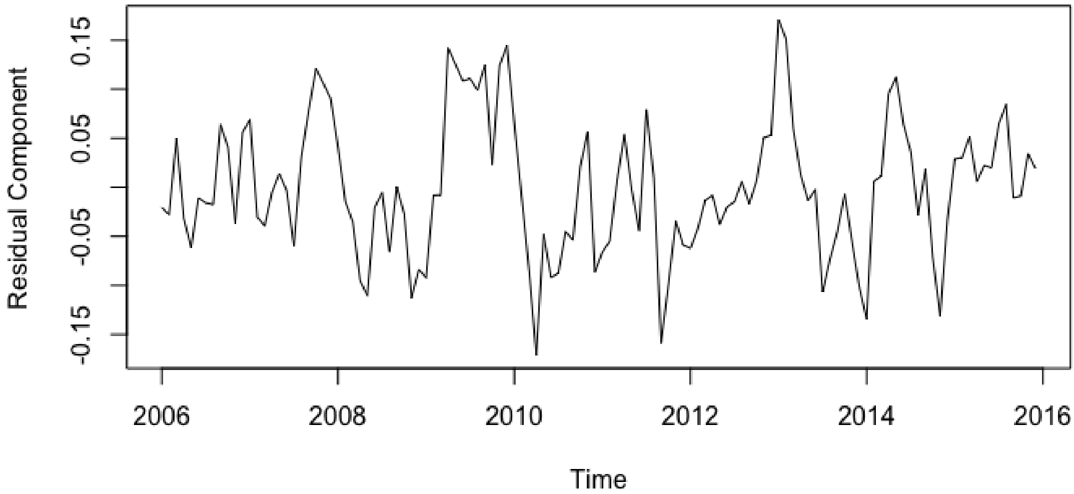

Figure 3: Time Series Plot of Residuals of Fitted Trend Model

The above residuals data seems stationary, and we can test this by doing an Augmented Dickey-Fuller test, using the command *adf.test()* in R package *tseries* [2]. A p-value greater than 0.05 indicates that the data is non-stationary. The residual data gave a $p < 0.01$, hence indicating that it is stationary (A, ll.53). It is obvious that the data is not independent identically distributed. We can investigate such non-randomness by interpreting the correlogram Figure 4:

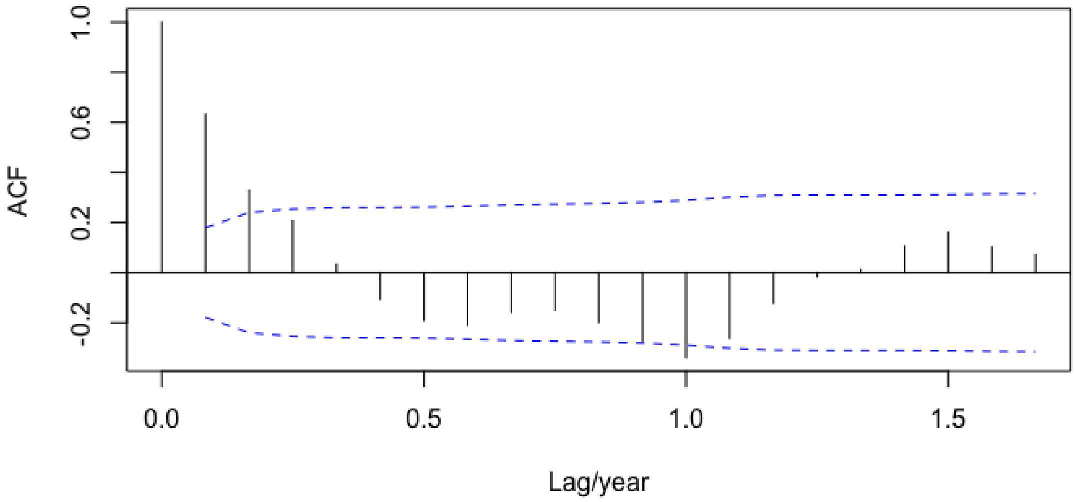

Figure 4: Correlogram for Residuals

We see there is significant short term positive autocorrelation. We can also use the Durbin-Watson Statistic and Peaks and Troughs test to test for the non-randomness in the residuals. (A, ll.59-64) The test results are given below:

Method	DW	p-value
Durbin-Watson	0.73469	1.473e-11

Method	nturns	E(p)	Var(p)	$\lvert Z \rvert$	p-value
Peaks and Troughs	60	78.66667	21.01111	4.07	4.7e-05

Table 2: Test Results for Non-Randomness

We see that both p-values are fairly small implying significant positive autocorrelation as we saw in Figure 4 so we reject randomness.

Now we are going to identify an appropriate ARMA(p, q) model for the residuals. We have obtained the ACF plot above and we also need the PACF plot to help us choose models:

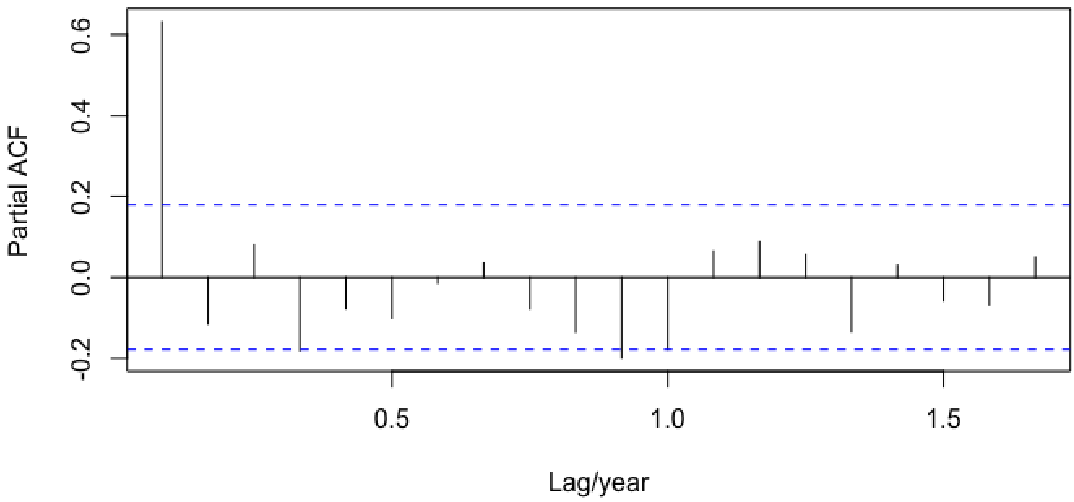

Figure 5: Partial autocorrelations for Residuals

From Figure 4 & 5, we have the following comments:

- The first two autocorrelations are significant and are gradually decreasing in size.
- The 1st partial autocorrelation is significant as is the 4th (only slightly!) and 11th.
- All the other autocorrelations, except from the 12th, and partial autocorrelations are non-significant.
- The cut off after the first partial autocorrelation, as well as the geometric decline in the autocorrelations, suggests AR(1).

- The later significant partial autocorrelation may well be due to chance and we can just ignore it as far as initial model choice is concerned.
- The two significant autocorrelations suggest MA(2).

3.1.2 Verification

As stated above, the figures suggest we consider AR(1), MA(2) and we will consider slightly more complex models including mixed models later on in our analysis (A, ll.69-118). Let us first consider AR(1):

Coefficients	ar1	intercept
estimates	0.6275	0.000
s.e.	0.0701	0.013

	$Sigma^2$	log-likelihood	AIC
Value	0.00288	180.48	-354.97

Table 3: Significance Test for AR(1)

The t-statistic of the AR(1) coefficient is given by:

$$t = \frac{estimate}{s.e.(estimate)} = \frac{0.6275}{0.0701} = 8.951$$

on (120-2) = 118 degrees of freedom.

We have pt(8.951, 118) = 1 implying p $<$ 0.0000001. Therefore, α_1 is significantly different from zero.

We now produce diagnostic plots, shown by Figure 6. We then plot the residuals against fitted values to check:

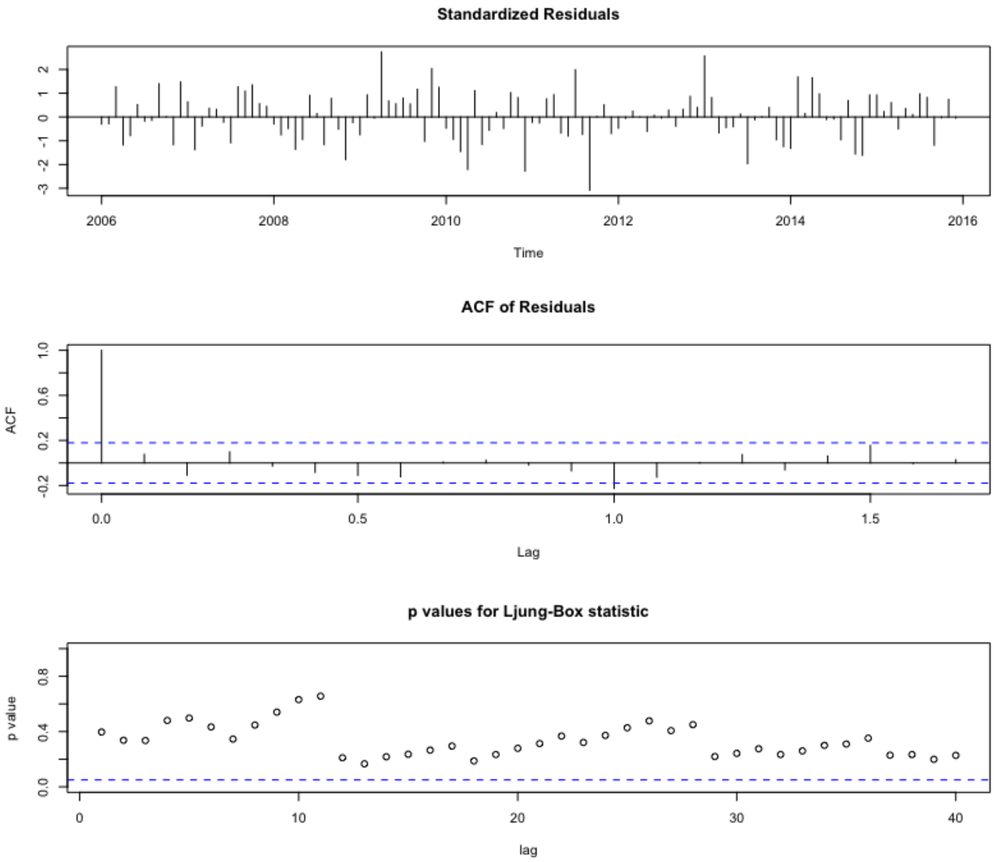

Figure 6: Diagnostic plots for AR(1)

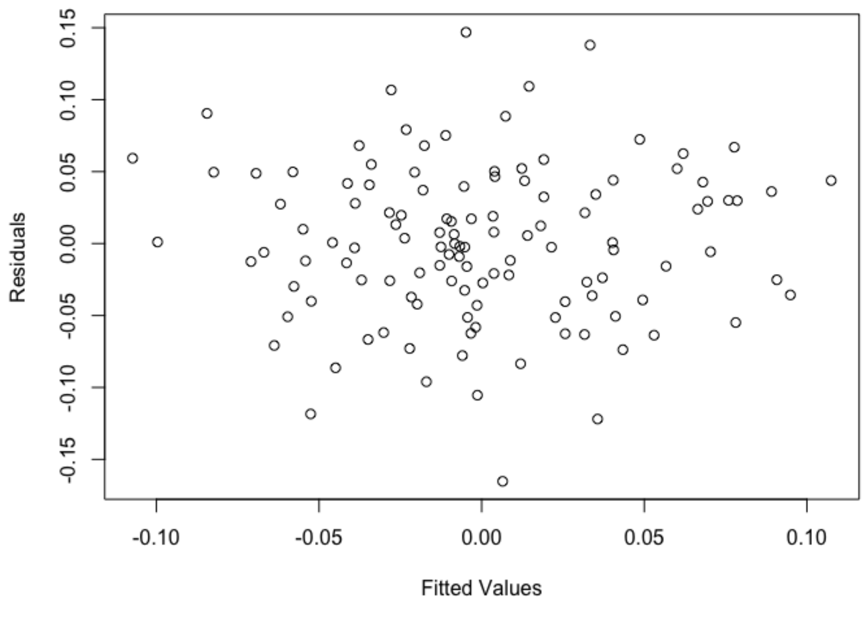

Figure 7: Residuals against fitted values for AR(1)

As we can see from above, the autocorrelations are now mainly non-significant (although r_{12} is close to significance, probably due to chance) and the Ljung-Box statistics are all above the threshold and are therefore non-significant. Also there is an approximately random scatter in Figure 7, implying the model fits the data well.

Now we consider our second chosen model MA(2):

Coefficients	ma1	ma2	intercept
estimates	0.7114	0.1767	0.0000
s.e.	0.0859	0.0778	0.0093

	$Sigma^2$	log-likelihood	AIC
Value	0.002912	179.8	-351.6

Table 4: Significance Test for MA(2)

The t-statistic for β_2 is given by:

$$t = \frac{estimate}{s.e.(estimate)} = \frac{0.1767}{0.0778} = 2.2712$$

on (120-3) = 117 degrees of freedom.

We have pt(2.2712, 117) = 0.98752 implying p = 0.02496 $<$ 0.05. Therefore, we can imply that β_2 is somewhat significantly different from zero and we need to retain all the terms.

We now produce diagnostic plots, shown by Figure 8. We then plot the residuals against fitted values to check:

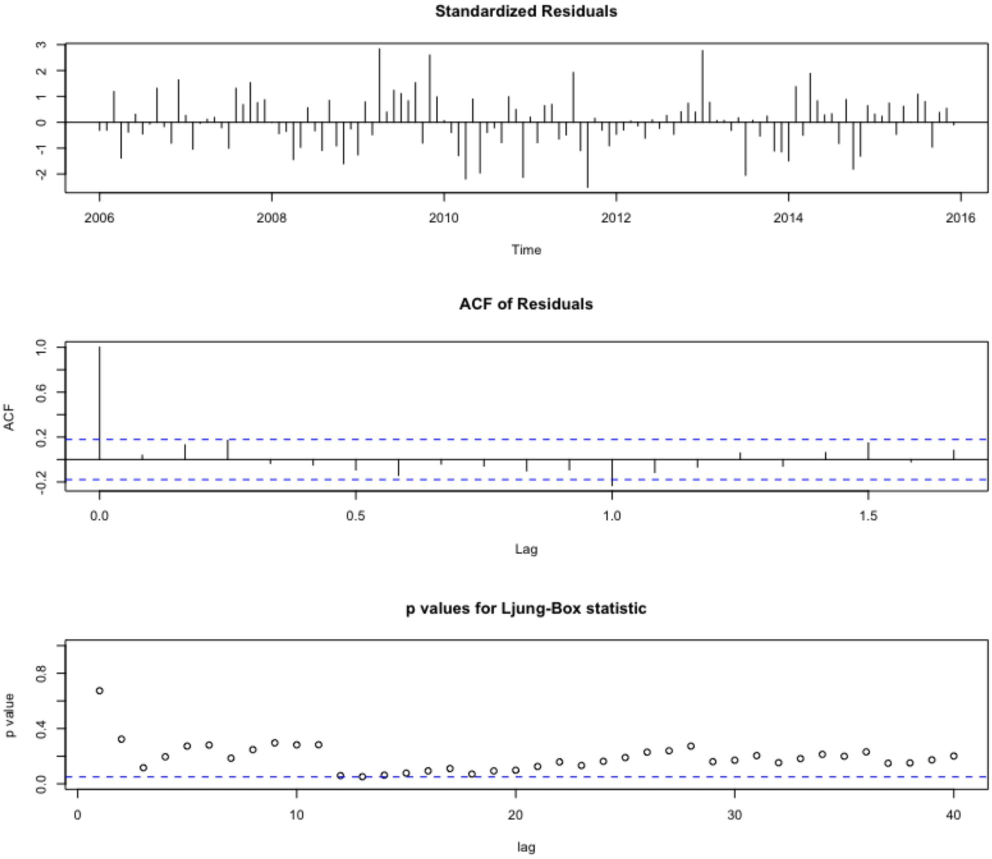

Figure 8: Diagnostic plots for MA(2)

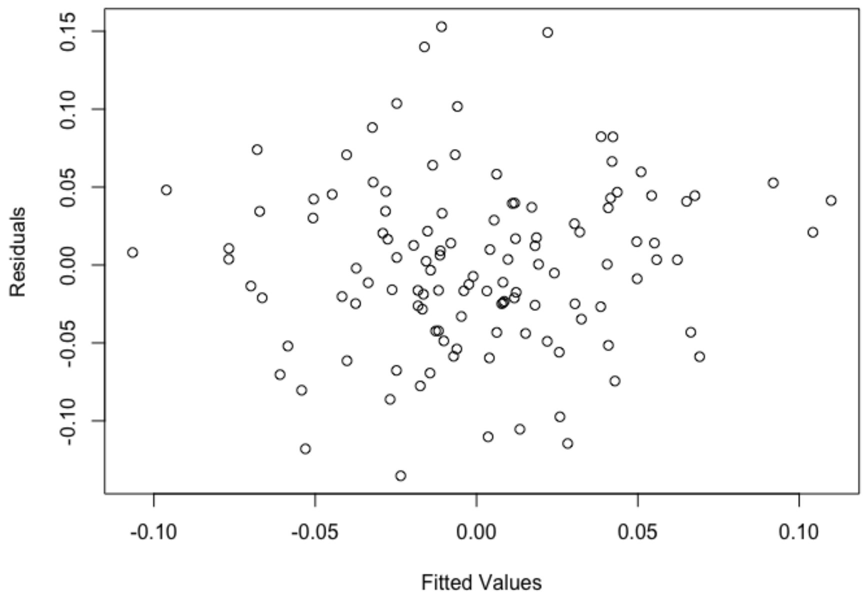

Figure 9: Residuals against fitted values for MA(2)

As we can see from above, the autocorrelations are now mainly non-significant (although r_{12} is close to significance (probably due to chance) and the Ljung-Box statistics are all on/above the threshold which implies that this model fits the data well. There is also an approximately random scatter in Figure 9 which implies a good model fit.

Now we consider an ARMA(1,1) model:

Coefficients	ar1	ma1	intercept
estimates	0.4658	0.2662	0.0000
s.e.	0.1447	0.1743	0.0114

	$Sigma^2$	log-likelihood	AIC
Value	0.002825	181.61	-355.22

Table 5: Significance Tests for ARMA(1,1)

The t-statistics for α_1 and β_1 are 3.219 and 1.527 respectively.

$$t1 = \frac{estimate}{s.e.(estimate)} = \frac{0.4658}{0.1447} = 3.219$$

$$t2 = \frac{estimate}{s.e.(estimate)} = \frac{0.2662}{0.1743} = 1.527$$

on (120-3) = 117 degrees of freedom.

We have pt(3.219, 117) = 0.9991675 implying $p < 0.005$. Therefore, α_1 is somewhat significantly different from zero.

We also have that pt(1.527, 117) = 0.93527. Therefore, we have a large p-value of 0.1294599 which implies that β_1 is not significantly different from zero. This result means that we should remove this term and return to an AR(1) model, we do not proceed any further with this model.

We may also consider an AR(2) model:

Coefficients	ar1	ar2	intercept
estimates	0.6990	-0.1125	0.0000
s.e.	0.0902	0.0901	0.0117

	$Sigma^2$	log-likelihood	AIC
Value	0.002842	181.21	-354.52

Table 6: Significance Tests for AR(2)

The t-statistic for α_2 is -1.249.

$$t = \frac{estimate}{s.e.(estimate)} = \frac{-0.1125}{0.0901} = -1.249$$

on (120-3) = 117 degrees of freedom.

For α_2, we have that pt(1.249,117) = 0.8929. This gives p = 0.2142. Clearly there is insufficient evidence to suggest that α_2 is significantly different from zero - we can remove this term and return to an AR(1) model. We do not need to proceed any further with this model.

Considering an MA(3) model:

Coefficients	ma1	ma2	ma3	intercept
estimates	0.7021	0.2784	0.1816	0.0001
s.e.	0.0912	0.0951	0.0869	0.0104

	$Sigma^2$	log-likelihood	AIC
Value	0.002813	181.86	-353.73

Table 7: Significance Tests for MA(3)

The t-statistic for β_3 is 2.0898.

$$t = \frac{estimate}{s.e.(estimate)} = \frac{0.1816}{0.0869} = 2.0898$$

on (120-4) = 116 degrees of freedom.

Testing the highest order term, β_3, we have that pt(2.0898,116) = 0.98059, giving a p-value of 0.03882. This implies that β_3 is different from zero at the 5% level.

Figure 10 displays the diagnostic plots for this model. All of the autocorrelations apart from one are non-significant, and all p-values for the Ljung-Box statistic are above the threshold level. Thus, this seems to be a fairly good model.

Figure 11 is a plot of the new residuals against the fitted values. There is approximately random scatter, suggesting that this model fits well.

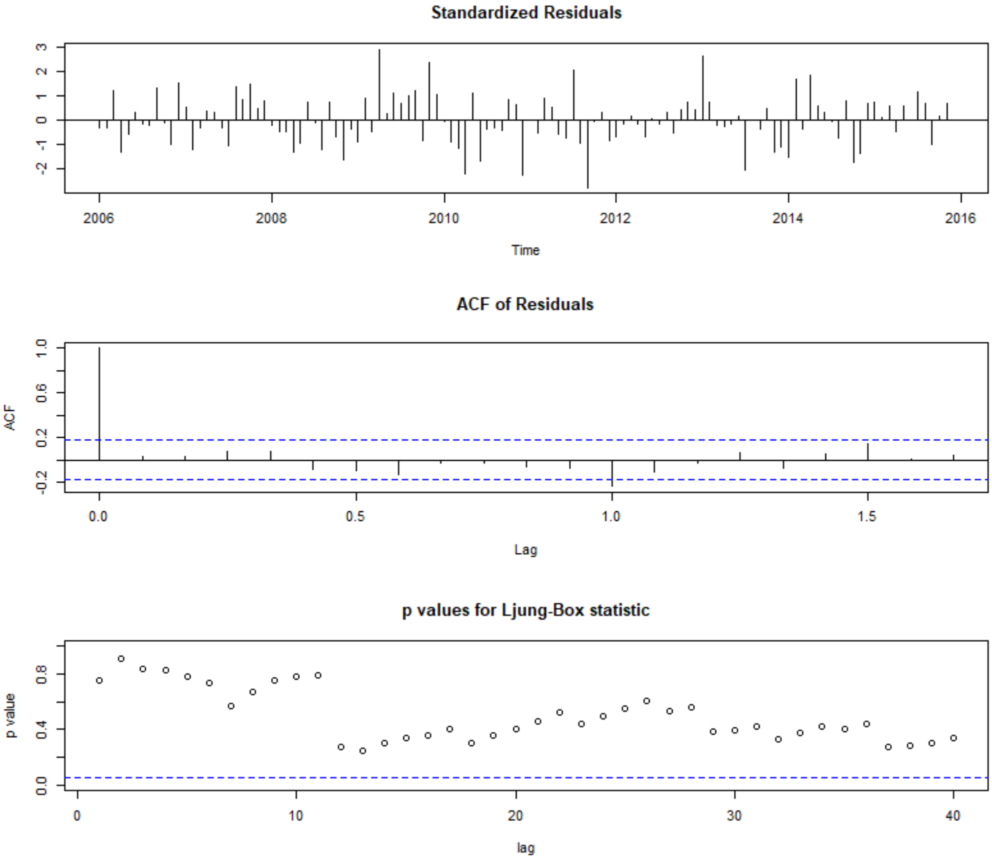

Figure 10: Diagnostic plots for MA(3)

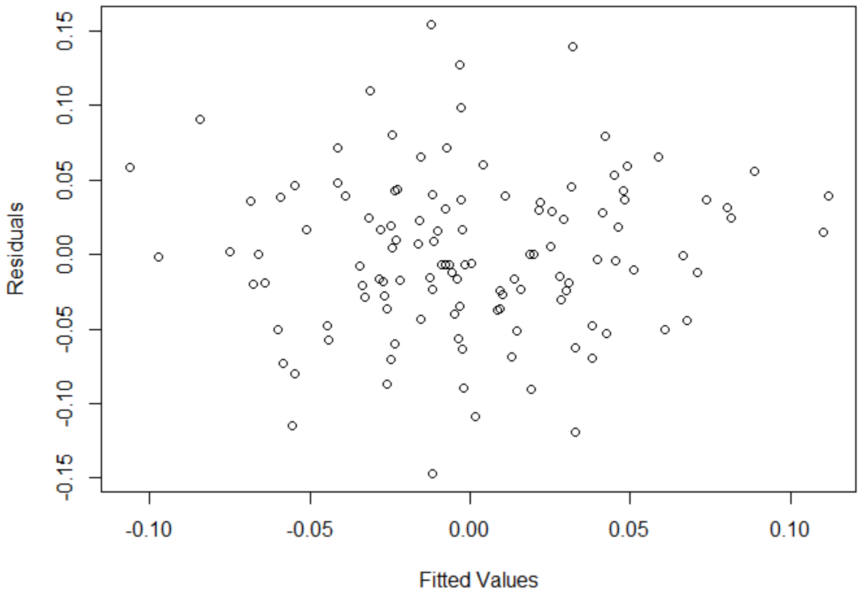

Figure 11: Residuals against fitted values for MA(3)

Since MA(3) fits well, we also want to consider an MA(4) model, as seen below.

Coefficients	ma1	ma2	ma3	ma4	intercept
estimates	0.7019	0.2887	0.3000	0.1867	0.0002
s.e.	0.0914	0.1001	0.1082	0.1192	0.0117

	$Sigma^2$	log-likelihood	AIC
Value	0.0028748	183.2	-354.4

Table 8: Significance Tests for MA(4)

We want to test the highest order term, β_4. The t-statistic for β_4 is 1.56628

$$t = \frac{estimate}{s.e.(estimate)} = \frac{0.1867}{0.1192} = 1.56628$$

on (120-4) = 115 degrees of freedom.

We have that pt(1.56628,115) = 0.93998, giving a p-value of 0.12004. Hence there is insufficient evidence to suggest that β_4 differs from zero at the 5% level. Therefore we can remove this term, return to MA(3) and not continue with this model.

Overall, we have 3 models which seem to fit well.

Model	Log-likelihood	AIC	No. of fitted parameters
AR(1)	180.48	-354.97	2
MA(2)	179.8	-351.6	3
MA(3)	181.86	-353.73	4

Table 9: Comparison amongst 3 fitted models

In summary, MA(3) has the largest log-likelihood but AR(1) has has the smallest AIC and has fewer parameters, thus AR(1) is to be preferred.

3.2 Seasonal ARIMA Model

So far, we have used curve fitting to remove the trend thus making the time series stationary. As we are not interested in the trend per se and our aim is to forecast, we can remove the trend by differencing, that is, using a ARIMA(p, d, q) model with $d \neq 0$. Upon adding an additional seasonal term we have a seasonal ARIMA model, which is written as follows [3]:

$$ARIMA \quad \underbrace{(p, d, q)}_{\text{Non-seasonal part}} \quad \underbrace{(P, D, Q)_m}_{\text{Seasonal part}}$$

3.2.1 Identification

First of all, we take logarithms to stabilise the variance as before. From Figure 1, we already knew the data is strongly seasonal with a 12-month period and clearly non-stationary, so we take a seasonal difference (A, ll.125). The seasonally differenced data are shown in Figure 12.

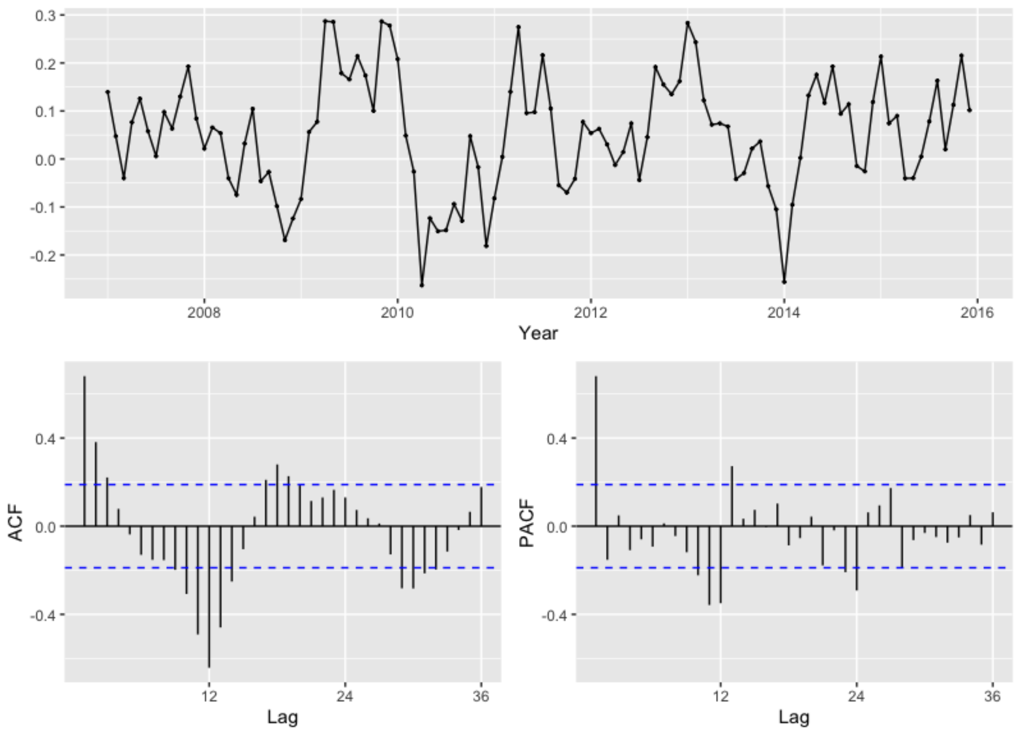

Figure 12: Seasonally differenced electricity consumption

The data appears to be stationary so we do not need to take a further difference, so currently we have the model ARIMA$(0, 0, 0)(0, 1, 0)_{12}$. We are going to find an appropriate ARIMA model based on the ACF and PACF shown in Figure 12. The significance spikes up to lag 3 in the ACF suggests a non-seasonal MA(3) component, and the significant spikes at lag 12 in the ACF suggests a seasonal MA(1) component. Consequently, we have an ARIMA$(0, 0, 3)(0, 1, 1)_{12}$ model. By similar strategy applied to the PACF, we could also have an initial model ARIMA$(1, 0, 0)(2, 1, 0)_{12}$.

3.2.2 Verification

We have obtained two initial models from above:

$$ARIMA(0, 0, 3)(0, 1, 1)_{12} \qquad \& \qquad ARIMA(1, 0, 0)(2, 1, 0)_{12}.$$

To simplify the verification procedures, we first fit these two models along with some variations, and compute the AICc values shown in Table 10 below (A, ll.130-140).

Model	AICc
ARIMA$(0,0,3)(0,1,1)_{12}$	-232.0243
ARIMA$(1,0,3)(0,1,1)_{12}$	-265.3321
ARIMA$(0,0,4)(0,1,1)_{12}$	-235.2611
ARIMA$(0,0,3)(1,1,1)_{12}$	-233.0145
ARIMA$(0,0,3)(0,1,2)_{12}$	-233.9923
ARIMA$(1,0,0)(2,1,0)_{12}$	-255.0984
ARIMA$(2,0,0)(2,1,0)_{12}$	-254.0728
ARIMA$(1,0,1)(2,1,0)_{12}$	-255.8913
ARIMA$(1,0,0)(3,1,0)_{12}$	-252.9311
ARIMA$(1,0,0)(2,1,1)_{12}$	-255.3253

Table 10: Comparison amongst 10 ARIMA models

Of these above models, the best is ARIMA$(1,0,3)(0,1,1)_{12}$ with the smallest AICc value. We can do the significance tests as follows:

Coefficients	ar1	ma1	ma2	ma3	sma1
estimates	1	-0.2749	-0.5287	-0.1798	-0.9781
s.e.	0	0.0958	0.0977	0.0818	0.0966

Table 11: Significance Tests for ARIMA$(1,0,3)(0,1,1)_{12}$

Obviously ar1 and sma1 are significantly different from zero. The only coefficient we need to check is ma3. The t-statistic of ma3 is given by:

$$t = \frac{estimate}{s.e.(estimate)} = \frac{-0.1798}{0.0818} = -2.198$$

on (120-5) = 115 degrees of freedom.

We have pt(2.198, 115) = 0.985 implying $p = 0.030 < 0.05$. Therefore, ma3 is significantly different from zero. We can look at two diagnostic figures; Figure 13 & 14 below:

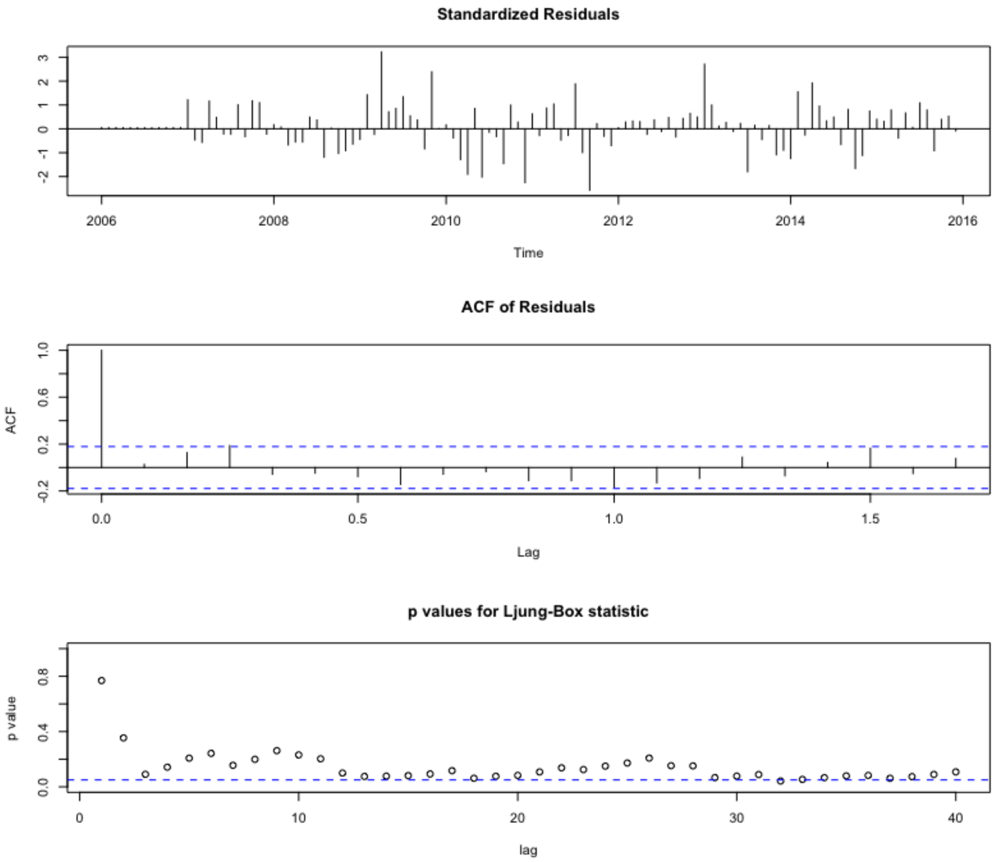

Figure 13: Diagnostic plots for ARIMA$(1,0,3)(0,1,1)_{12}$

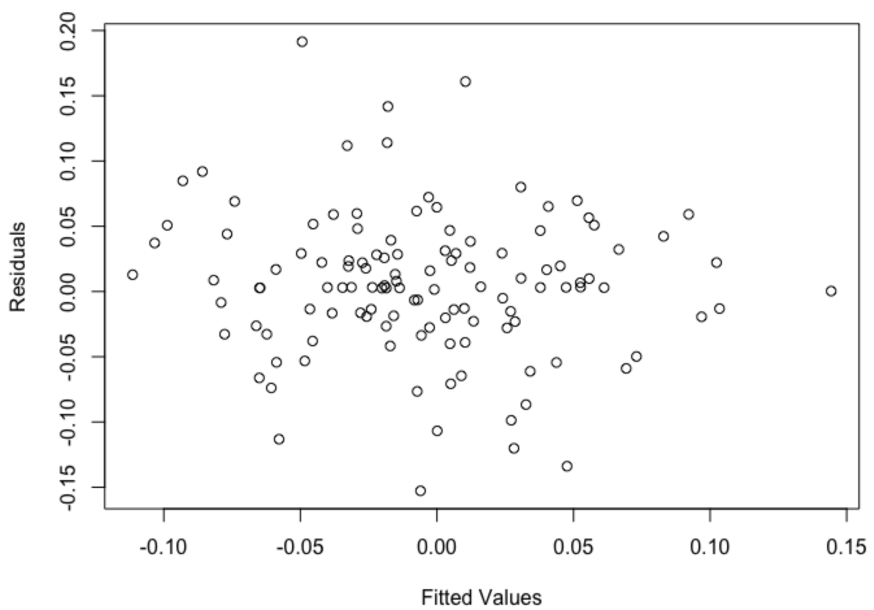

Figure 14: Residuals against fitted values for ARIMA$(1,0,3)(0,1,1)_{12}$

From Figure 13, we see that all residuals are either non-significant or very marginally significant, indicating a goodness of fit. Furthermore, an overwhelming majority of the p-values for the Ljung-Box statistic lie above or on the threshold line. The residual plot in Figure 14 displays approximately random scatter - indicating again a good fit.

3.3 Model Determination

During the last two subsections, we obtained the following two "best" models by different methods:

$$AR(1) \qquad \& \qquad ARIMA(1,0,3)(0,1,1)_{12}.$$

To assess the forecasting performance of our models and also to avoid over-fitting, we need to fit two models again using the first nine years of data and then forecast the next twelve months comparing the forecast with the actual values. We also need to compute RMSE for our chosen models to help us make a decision (A, ll.236-281). Here are the "best" fitted models using nine years of data by ARMA and seasonal ARIMA method respectively:

$$AR(1) \qquad \& \qquad ARIMA(1,0,3)(0,1,1)_{12}.$$

To our surprise, we get exactly the same models using only first 9 years of data. Now we use Box-Jenkins methods to forecast. The predicted values and forecast intervals for each model are shown in Table 12 (A, ll.256-264).

		AR(1)		ARMA$(1,0,3)(0,1,1)_{12}$	
Month	Real	Fitted	95% CI	Fitted	95% CI
Jan 2015	39.34	37.13	$(33.26, 41.46)$	38.37	$(33.75, 43.62)$
Feb 2015	43.27	40.97	$(35.96, 46.68)$	41.67	$(35.52, 48.88)$
March 2015	36.48	33.84	$(29.49, 38.83$	34.02	$(28.93, 40.01)$
April 2015	32.55	31.95	$(27.17, 37.58)$	31.95	$(27.17, 37.58)$
May 2015	27.58	26.51	$(23.01, 30.54)$	26.57	$(22.59, 31.25)$
June 2015	24.65	23.78	$(20.64, 27.41)$	23.82	$(20.25, 28.01)$
July 2015	23.26	21.36	$(18.54, 24.62)$	21.38	$(18.18, 25.14)$
August 2015	22.26	20.00	$(17.35, 23.05)$	20.01	$(17.01, 23.53)$
September 2015	27.62	27.59	$(23.94, 31.80)$	27.59	$(23.45, 32.45)$
October 2015	32.35	32.25	$(27.98, 37.17)$	32.24	$(27.41, 37.92)$
November 2015	37.11	35.27	$(30.60, 40.66)$	35.27	$(29.98, 41.49)$
December 2015	42.74	41.32	$(35.85, 47.63)$	41.29	$(35.09, 48.57)$

Table 12: Forecasting values and 95% CI for AR(1) and ARIMA$(1,0,3)(0,1,1)_{12}$

We can easily see that all real values lie in the 95% confidence intervals of both models. Then we use Root Mean Squared Error, RMSE, to evaluate the prediction performance of these two models. The definition of RMSE is as follows[3]:

$$RMSE = \sqrt{mean(e_t^2)}$$

where e_t is the forecast errors. These can be computed in R, giving the RMSEs of 1.673 for AR(1) and 1.465 for ARIMA$(1,0,3)(0,1,1)_{12}$.

Since the RMSE for the ARIMA$(1,0,3)(0,1,1)_{12}$ is smaller, we expect this model to fit the data slightly better.

Figures 15 and 16 display plots of the original data along with the predicted data. The models appear to predict the data very similarly, with both being fairly accurate. However the first quarter of 2015 seems to be predicted better by the ARIMA$(1,0,3)(0,1,1)_{12}$ model.

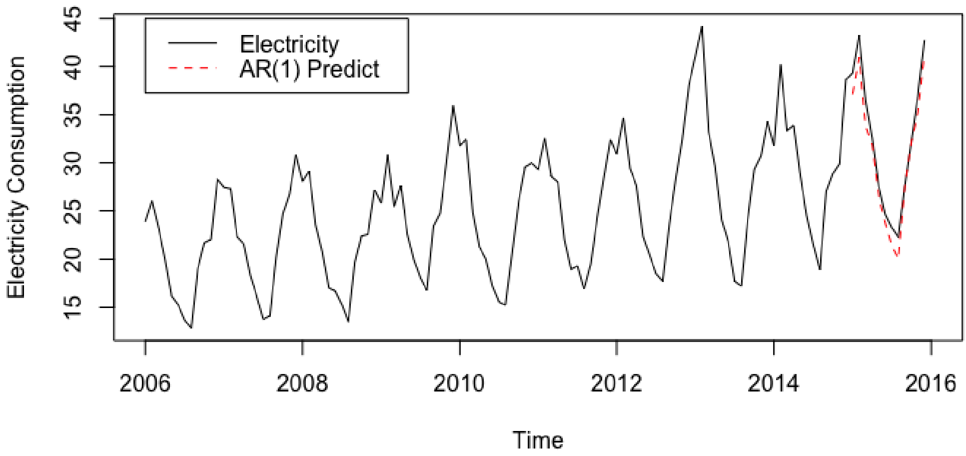

Figure 15: Plot of Original Data and Predicted values of AR(1)

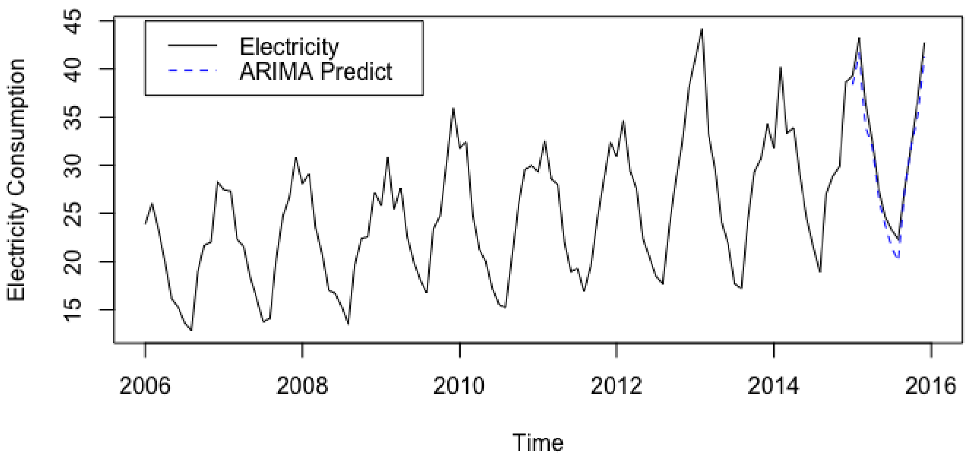

Figure 16: Plot of Original Data and Predicted values of ARIMA$(1,0,3)(0,1,1)_{12}$

Although the ARIMA$(1,0,3)(0,1,1)_{12}$ model seems to fit the data better, this difference is only slight. Given that the ARIMA$(1,0,3)(0,1,1)_{12}$ model fits 6 variables, while the AR(1) model fits only 2, the team chooses to continue with AR(1).

4 Forecasting

Using the final model choice of AR(1), the team intends to predict the electricity consumption for the next 6 months (i.e. Jan 2016 - June 2016. A, ll.286-311).

	Fitted Value	95% CI
Jan 2016	40.65	(36.51,45.26)
Feb 2016	44.45	(39.16,50.45)
Mar 2016	36.59	(32.01,41.82)
Apr 2016	34.12	(29.78,39.10)
May 2016	28.40	(24.76,32.57)
Jun 2016	25.43	(22.17,29.19)

Table 13: Predicted Values and 95% CI of Jan-Jun 2016 for AR(1)

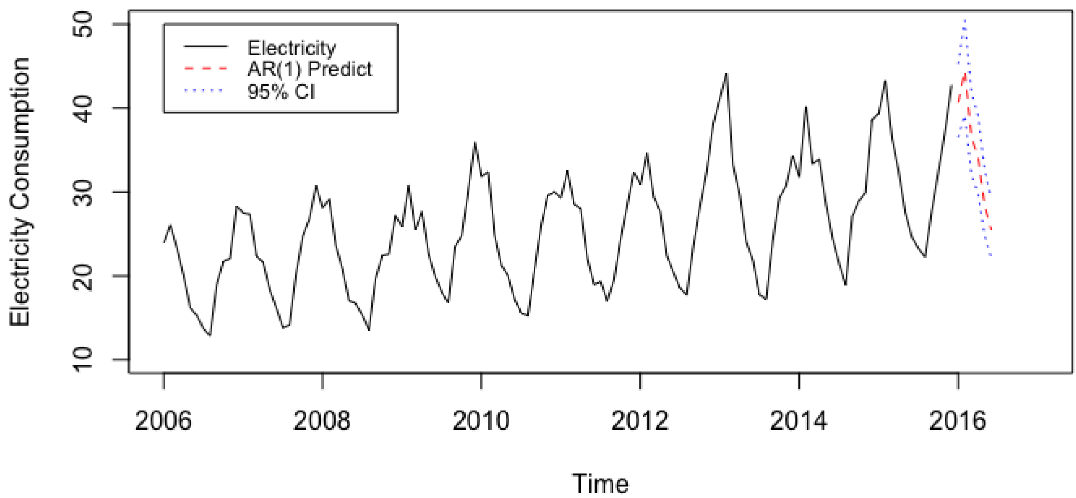

Figure 17: Prediction Plot of Jan-Jun 2016 for AR(1)

All predicted values are within their respective 95% intervals. The predicted data seems to follow a very similar pattern to previous data, including the seasonality and upward trend.

5 Conclusion

In comparing numerous models using the Box-Jenkins approach to model building, the team isolated two potential models - AR(1) and ARIMA$(1, 0, 3)(0, 1, 1)_{12}$. Through further analysis, a decision was made to continue with an AR(1) model due to less fitted parameters.

Using this model, the team were able to forecast the next 6 months of electricity consumption. As seen previously in Figure 17, the team have concluded that electricity consumption has increased periodically, and will continue increasing over time.

References

[1] Cryer, J., & Chan, K. (2008). *Time series analysis : With applications in R* (2nd ed., Springer texts in statistics). New York, N.Y.: Springer.

[2] Traplett, A., & Hornik, K., & LeBaron, B. (2019) *Time series analysis and compuational finance*

[3] Hyndman, R., & Athanasopoulos, G. (2018). *Forecasting : Principles and practice* (2nd ed.). Heathmont, Vic.]: OTexts. Available at https://otexts.com/fpp2/

A R Scripts

```r
# MAS3911 Time Series Project

library(forecast)
library(lmtest)
library(timeSeries)
library(fpp2)
library(tseries)

setwd("~/Desktop/MAS3911")
data<-read.table("projectdata.txt")
y<-data[, 57]

# >>>>>>>>>> Section 2 <<<<<<<<<< #
elec <- list()
elec$data <- ts(y, start = c(2006, 1), frequency = 12)
elec$logdata <- ts(log(y), start = c(2006, 1), frequency = 12)

plot(elec$data, ylab = "Electricity Consumption") # Fig.1
points(elec$data, pch = 21, cex = 0.4, bg = 1)

# >>>>>>>>>> Section 3.1.1 <<<<<<<<<< #
elec$season = gl(12, 1, 120)
# Use curve fitting to estimate trend
elec$time = c(1 : 120)

# Compare trans and untrans
fit1 <- lm(elec$data ~ elec$time + elec$season)
fit2 <- lm(elec$logdata ~ elec$time + elec$season)
c(summary(fit1)$r.squared, summary(fit1)$adj.r.squared)
c(summary(fit2)$r.squared, summary(fit2)$adj.r.squared)

# BoxCox parameter test
lambda <- BoxCox.lambda(elec$data)
print(lambda)
lambda <- BoxCox.lambda(elec$logdata)
print(lambda)
# elec$tdata <- BoxCox(elec$data, lambda)

# We use log trans - fit2, use elec$logdata throughout the project
elec$trend <- ts(fitted.values(fit2), start = c(2006, 1), frequency = 12)

# Plot of logdata and trend
plot(elec$logdata, ylab = "Log(Electricity Consumption)") # Fig.2
points(elec$logdata, pch = 21, cex = 0.4, bg = 1)
lines(elec$trend, col = 2, lty = 2)
legend(2006, 3.8, legend = c("Log(Electricity)", "Trend"), col = c(1, 2), lty = c(1, 2), pch = c(21, NA), pt.bg = c(1, NA), pt.cex = c(0.4, NA), cex = 0.8)

# Now we are focusing elec$resid, the residuals of the fitted trend model
elec$resid <- ts(residuals(fit2), start = c(2006, 1), frequency = 12)
plot(elec$resid, ylab = "Residual Component") # Fig.3
```

```

# ADF Test
adf.test(elec$resid)

acf(elec$resid, ci.type = "ma", xlab = "Lag/year", main = "") # Fig.4
# acf(elec$logdata, lag.max = 120, ci.type = "ma", xlab = "Lag/year")

# Durbin-Watson Statistic
dwtest(fit2)

# Peaks and Troughs test
residt = timeSeries(residuals(fit2), elec$time)
turnsStats(residt)
# |Z| = 4.07, p-value = 2*(1-pnorm(4.07))


pacf(elec$resid, xlab = "Lag/year", main = "") # Fig.5

# >>>>>>>>>> Section 3.1.2 <<<<<<<<<< #

# AR(1)
ar1 <- arima(elec$resid, order = c(1, 0, 0))
ar1
pt(8.951, 118)

tsdiag(ar1, gof.lag = 40) # Fig.6, 800*700

plot.default(elec$resid-ar1$residuals, ar1$residuals, xlab = "Fitted
    Values", ylab = "Residuals") # Fig.7, 700*550


# MA(2)
ma2 <- arima(elec$resid, order = c(0, 0, 2))
ma2
pt(2.2712, 117) # (120 - 2 - 1) = 117 degrees of freedom
2 * (1 - pt(2.2712, 117)) # < 0.05, all terms needed
tsdiag(ma2, gof.lag = 40) # Fig.8
plot.default(elec$resid-ma2$residuals, ma2$residuals, xlab = "Fitted
    Values", ylab = "Residuals") # Fig.9


# ARMA(1, 1)
arma11 = arima(elec$resid, order = c(1, 0, 1))
arma11
pt(3.219, 117)
2 * (1 -  pt(3.219, 117))
pt(1.527, 117)
2 * (1 - pt(1.527, 117))
# No need!

# AR(2)
ar2 <- arima(elec$resid, order = c(2, 0, 0))
ar2
pt(1.249, 117)
# No need!
```

```
# MA(3)
ma3 <- arima(elec$resid, order = c(0, 0, 3))
ma3
pt(2.0898, 116)
2 * (1 - pt(2.0898, 116)) # < 0.05, all terms needed
tsdiag(ma3, gof.lag = 40) # Fig.10
plot.default(elec$resid-ma3$residuals, ma3$residuals, xlab = "Fitted
    Values", ylab = "Residuals") # Fig.11

# MA(4)
ma4 <- arima(elec$resid, order = c(0, 0, 4))
ma4
pt(1.566, 115)
2 * (1 - pt(1.566, 115))
# No need!


# >>>>>>>>>> Section 3.2.1 <<<<<<<<<< #

# ARIMA
# Seasonally differenced transformed electricity consumption
elec$logdata %>% diff(lag=12) %>% ggtsdisplay(xlab="Year", main="") #
    Fig.12


# >>>>>>>>>> Section 3.2.2 <<<<<<<<<< #

# Check AICc for 10 models near 2 chosen models
(fit_i1 <- Arima(elec$data, order=c(0,0,3), seasonal=c(0,1,1), lambda =
    0))$aicc
(fit_i2 <- Arima(elec$data, order=c(1,0,3), seasonal=c(0,1,1), lambda =
    0))$aicc # Best
(fit_i3 <- Arima(elec$data, order=c(0,0,4), seasonal=c(0,1,1), lambda =
    0))$aicc
(fit_i4 <- Arima(elec$data, order=c(0,0,3), seasonal=c(1,1,1), lambda =
    0))$aicc
(fit_i5 <- Arima(elec$data, order=c(0,0,3), seasonal=c(0,1,2), lambda =
    0))$aicc
(fit_i6 <- Arima(elec$data, order=c(1,0,0), seasonal=c(2,1,0), lambda =
    0))$aicc
(fit_i7 <- Arima(elec$data, order=c(2,0,0), seasonal=c(2,1,0), lambda =
    0))$aicc
(fit_i8 <- Arima(elec$data, order=c(1,0,1), seasonal=c(2,1,0), lambda =
    0))$aicc
(fit_i9 <- Arima(elec$data, order=c(1,0,0), seasonal=c(3,1,0), lambda =
    0))$aicc
(fit_i10 <- Arima(elec$data, order=c(1,0,0), seasonal=c(2,1,1), lambda
    = 0))$aicc


fit_i2 <- Arima(elec$data, order=c(1,0,3), seasonal=c(0,1,1), lambda =
    0)

# checkresiduals(fit_i2)
```

```
tsdiag(fit_i2, gof.lag = 40) # Fig.13
plot.default(elec$resid-fit_i2$residuals, fit_i2$residuals, xlab = "
    Fitted Values", ylab = "Residuals") # Fig.14


# >>>>>>>>>> Section 3.3 <<<<<<<<<< #

# Repeat the procedures above to determine "best" models using only
    nine years of data

####################### Repeating Process #######################

elec$train <- window(elec$data, start = c(2006, 1), end = c(2014, 12))
elec$test <- window(elec$data, start = c(2015, 1))
elec$logtrain <- window(elec$logdata, start = c(2006, 1), end = c(2014,
    12))
elec$logtest <- window(elec$logdata, start = c(2015, 1))

elec$ftime = c(1 : 108)
elec$fseason = gl(12, 1, 108)
fit_f1 <- lm(elec$logtrain ~ elec$ftime + elec$fseason)
elec$ftrend <- ts(fitted.values(fit_f1), start = c(2006, 1), frequency
    = 12)

elec$fresid <- ts(residuals(fit_f1), start = c(2006, 1), frequency =
    12)
plot(elec$fresid, ylab = "Residual Component")

acf(elec$fresid, ci.type = "ma", xlab = "Lag/year", main = "") #
    suggests MA(2)

dwtest(fit_f1) # Reject Randomness

pacf(elec$fresid, xlab = "Lag/year", main = "") # suggests AR(1)

# so the results are the same as using 10 years, then keep checking...

# AR(1) - Yes!!!
far1 <- arima(elec$fresid, order = c(1, 0, 0))
far1
tsdiag(far1, gof.lag = 40) # Yes!
plot.default(elec$fresid-far1$residuals, far1$residuals, xlab = "Fitted
    Values", ylab = "Residuals") # Yes!

# MA(2) - Yes!!!
fma2 <- arima(elec$fresid, order = c(0, 0, 2))
fma2
2 * (1 - pt(2.2712, 117)) # Yes!
tsdiag(fma2, gof.lag = 40) # Yes!
plot.default(elec$fresid-fma2$residuals, fma2$residuals, xlab = "Fitted
    Values", ylab = "Residuals") # Yes!

# ARMA(1, 1) - No!!!
farma11 = arima(elec$fresid, order = c(1, 0, 1))
farma11
```

```
2 * (1 - pt(1.274, 117)) # No!

# AR(2) - No!!!
far2 <- arima(elec$fresid, order = c(2, 0, 0))
far2

# MA(3) - Yes!!!
fma3 <- arima(elec$fresid, order = c(0, 0, 3))
fma3
2 * (1 - pt(2.104, 116)) # Yes!
tsdiag(fma3, gof.lag = 40) # Yes!
plot.default(elec$fresid-fma3$residuals, fma3$residuals, xlab = "Fitted Values", ylab = "Residuals")  # Yes!

# MA(4) - No!!!
fma4 <- arima(elec$fresid, order = c(0, 0, 4))
fma4
2 * (1 - pt(1.639, 115)) # No!

far1
fma2
fma3

# Check above three models and choose AR(1) again.

elec$logtrain %>% diff(lag=12) %>% ggtsdisplay(xlab="Year", main="")
# same as above

(fit_fi1 <- Arima(elec$train, order=c(0,0,3), seasonal=c(0,1,1), lambda = 0))$aicc
(fit_fi2 <- Arima(elec$train, order=c(1,0,3), seasonal=c(0,1,1), lambda = 0))$aicc # best
(fit_fi3 <- Arima(elec$train, order=c(0,0,4), seasonal=c(0,1,1), lambda = 0))$aicc
(fit_fi4 <- Arima(elec$train, order=c(0,0,3), seasonal=c(1,1,1), lambda = 0))$aicc
(fit_fi5 <- Arima(elec$train, order=c(0,0,3), seasonal=c(0,1,2), lambda = 0))$aicc
(fit_fi6 <- Arima(elec$train, order=c(1,0,0), seasonal=c(2,1,0), lambda = 0))$aicc
(fit_fi7 <- Arima(elec$train, order=c(2,0,0), seasonal=c(2,1,0), lambda = 0))$aicc
(fit_fi8 <- Arima(elec$train, order=c(1,0,1), seasonal=c(2,1,0), lambda = 0))$aicc
(fit_fi9 <- Arima(elec$train, order=c(1,0,0), seasonal=c(3,1,0), lambda = 0))$aicc
(fit_fi10 <- Arima(elec$train, order=c(1,0,0), seasonal=c(2,1,1), lambda = 0))$aicc

########################## End Repeating ##########################


# Predict two models using first nine years data

# AR(1)
```

```r
predictedtrend <- fit_f1$coef[1] + fit_f1$coef[2]*(109:120)
season <- c(0, fit_f1$coef[3:13])
predtrendseas <- predictedtrend + season
predtrendseas <- ts(predtrendseas, start = c(2015, 1), frequency = 12)

plot(elec$logdata, xlim = c(2006, 2016), ylim = c(2.5, 4.0))
points(predtrendseas, col = 2, cex = 0.6)
lines(predtrendseas, col = 2, lty = 2)
legend(2006, 3.8, legend = c("Log(Electricity)", "Predict"), col = c(1, 2), lty = c(1, 2), pch = c(NA, 21), pt.cex = c(NA, 0.6), cex = 0.8)

far1 <- arima(elec$fresid, order = c(1, 0, 0))
far1P <- predict(far1, n.ahead = 12)
far1P$pred <- ts(far1P$pred, start = c(2015, 1), frequency = 12)
far1PT <- far1P$pred + predtrendseas
far1P$se <- ts(far1P$se, start = c(2015, 1), frequency = 12)
far1PTU <- far1PT + 2*far1P$se
far1PTL <- far1PT - 2*far1P$se


# Predicted values and 95% CI for AR(1)
round(cbind(exp(far1PT), exp(far1PTL), exp(far1PTU)), 2)


# Predicted values and 95% CI for ARIMA
a <- elec$train %>%
      Arima(order=c(1,0,3), seasonal=c(0,1,1), lambda=0) %>%
      forecast(h = 12, level = 95)
round(cbind(a$mean, a$lower, a$upper), 2)

# Compute RMSE and plot
#AR(1)
sqrt(mean((exp(far1PT) - elec$test)^2)) # 1.673

#ARIMA
sqrt(mean((a$mean - elec$test)^2)) # 1.465


plot(elec$data, xlim = c(2006, 2016), ylab = "Electricity Consumption") # Fig.15
lines(exp(far1PT), col = 2, lty = 2)
legend(2006, 45, legend = c("Electricity", "AR(1) Predict"), col = c(1, 2), lty = c(1, 2))

ts_arima <- ts(a$mean, start = c(2015, 1), frequency = 12)
plot(elec$data, xlim = c(2006, 2016), ylab = "Electricity Consumption") # Fig.16
lines(ts_arima, col = "blue", lty = 2)
legend(2006, 45, legend = c("Electricity", "ARIMA Predict"), col = c(1, "blue"), lty = c(1, 2))


# >>>>>>>>>> Section 4 <<<<<<<<<< #

```

```r
# Predict 2016 for AR(1)
# We already fit a model using 10 years before: fit2

predictedtrend2 <- fit2$coef[1] + fit2$coef[2]*(121:126)
season2 <- c(0, fit2$coef[3:7])
predtrendseas2 <- predictedtrend2 + season2
predtrendseas2 <- ts(predtrendseas2, start = c(2016, 1), frequency =
    12)

ar1 <- arima(elec$resid, order = c(1, 0, 0))
ar1P <- predict(ar1, n.ahead = 12)
ar1P$pred <- ts(ar1P$pred, start = c(2016, 1), frequency = 12)
ar1PT <- ar1P$pred + predtrendseas2
ar1P$se <- ts(ar1P$se, start = c(2016, 1), frequency = 12)
ar1PTU <- ar1PT + 2*ar1P$se
ar1PTL <- ar1PT - 2*ar1P$se

# Plot of original data, predicted values for 2016 and 95% CI
plot(elec$data, xlim = c(2006, 2017), ylab = "Electricity Consumption",
     ylim = c(10, 50)) # Fig.17
lines(exp(ar1PT), col = 2, lty = 2)
lines(exp(ar1PTU), col = 4, lty = 3)
lines(exp(ar1PTL), col = 4, lty = 3)
legend(2006, 50, legend = c("Electricity", "AR(1) Predict", "95% CI"),
    col = c(1, 2, 4), lty = c(1, 2, 3), cex = 0.8)


# Predicted values and 95% CI for AR(1) 2016
round(cbind(exp(ar1PT), exp(ar1PTL), exp(ar1PTU)), 2)
```