MAS8911: Time Series Project

Cleo Bamber & Thomas Oman

May 3, 2020

1 Modelling

1.1 The data

We have a time series that represents the monthly total electricity consumption in a city over a ten-year period, from January 2006 to December 2015. The units are millions of kilowatt-hours (106 kWh). In order to identify an appropriate ARMA(p,q) model to the time series, we first need to remove any trend and/or seasonality in the data. The time series can initially be shown in a plot as below in Figure 1.

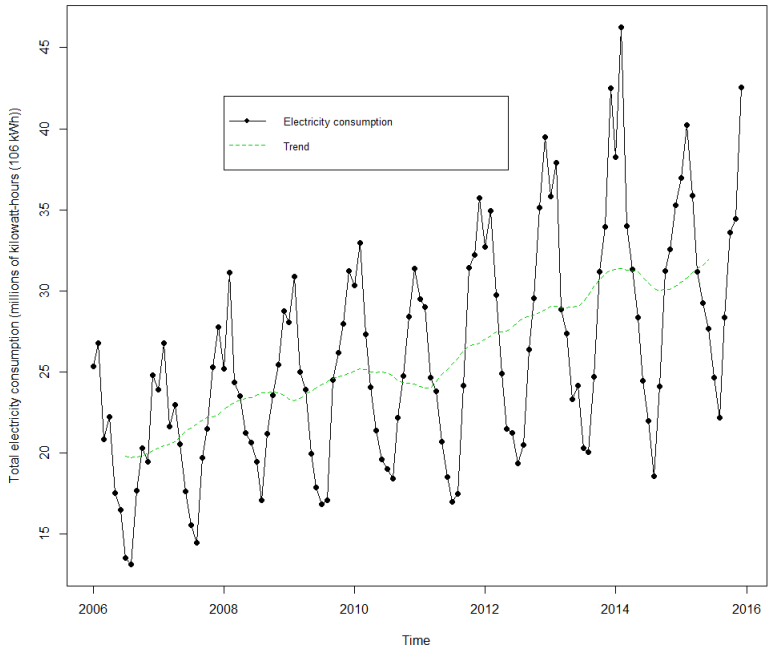

Figure 1: The monthly total electricity consumption in a city over a ten-year period, from January 2006 to December 2015.

There is a clear increasing trend and clear evidence of seasonality therefore we need to account for both. To do this, we can fit a smooth curve to the data. After fitting the curve to the data, it was found that a log transformed model fit the data better than the untransformed model. This is because we see a clear improvement in the R-squared and adjusted R-squared for the transformed model. See Appendix A.

1.2 Identification of an appropriate ARMA(p,q) model(s)

Having adjusted the data for trend and seasonality, we can then obtain the correlogram of the residual time series and we can use this to choose appropriate values of p and q for the required ARMA(p,q) model. Additionally, we can look at the partial autocorrelation coefficients to help choose an AR(p) process.

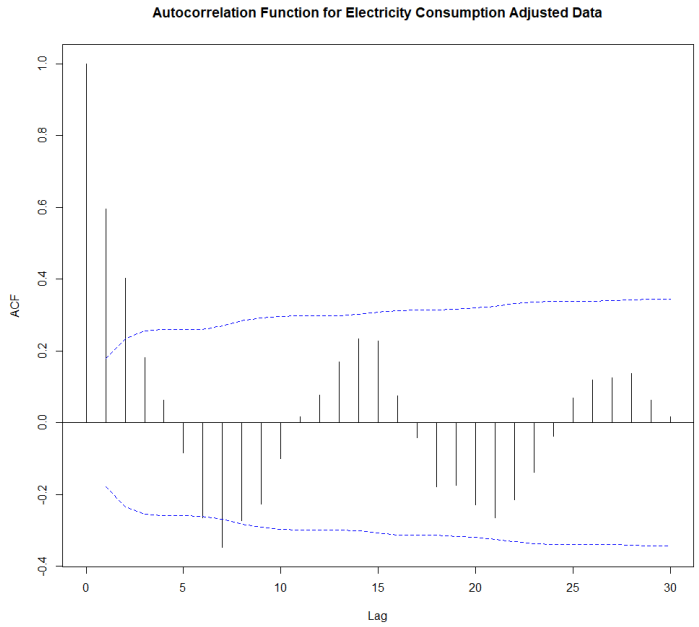

Figure 2: Autocorrelation Function for Electricity Consumption Adjusted Data for Seasonality and Trend.

From Figure 2 above, we can see that the first and second autocorrelation coefficients are clearly significant. Autocorrelation coefficients 6 and 7 are also significant. All of the others are contained within the 95% intervals. Now, a plot of the partial autocorrelation function is obtained as demonstrated by Figure 3.

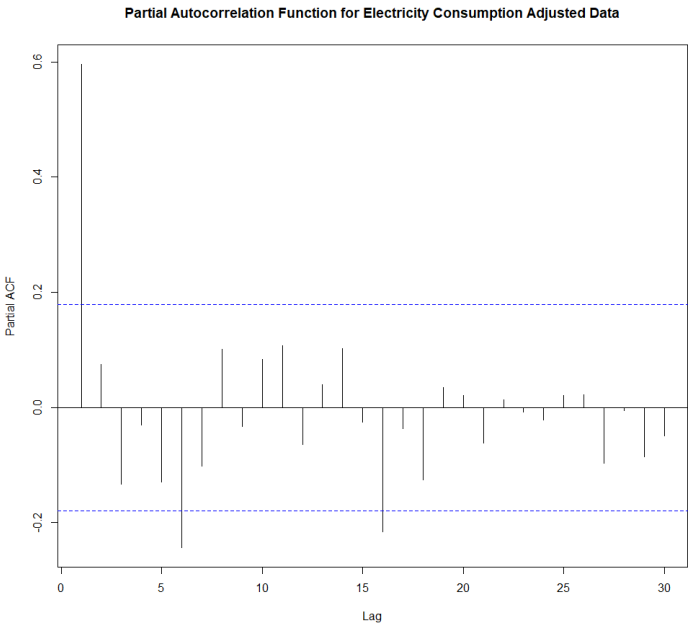

Figure 3: Partial Autocorrelation Function for Electricity Consumption Adjusted Data for Seasonality and Trend.

The first partial autocorrelation coefficient is clearly significant. Nearly all the others are non significant. $\hat{\pi_6}$ and $\hat{\pi_{16}}$ are significant but this is probably due to chance as we are doing a lot of tests and approximately 5% will be marginally significant by chance. We shall ignore these as far as model selection is concerned.

These plots suggest that we consider AR(1) and MA(2) initially. We will consider more complex models including mixed models later.

1.3 Fitting the models to the data and verifying if they fit the data

See Appendix C for all `R` code for this section.

1.3.1 AR(1) model

Let us first consider AR(1). We can fit this ARMA model to the data using `R` and check to see if the AR(1) coefficient is significant. The t-statistic is given by $t = 8.171939$ on $n-p-q-1 = 120-2 = 118$ degrees of freedom. `pt(t,118)` in `R` gives 1 implying $p < 0.000001$ thus α_1 is significantly different from zero. We can now produce diagnostic plots:

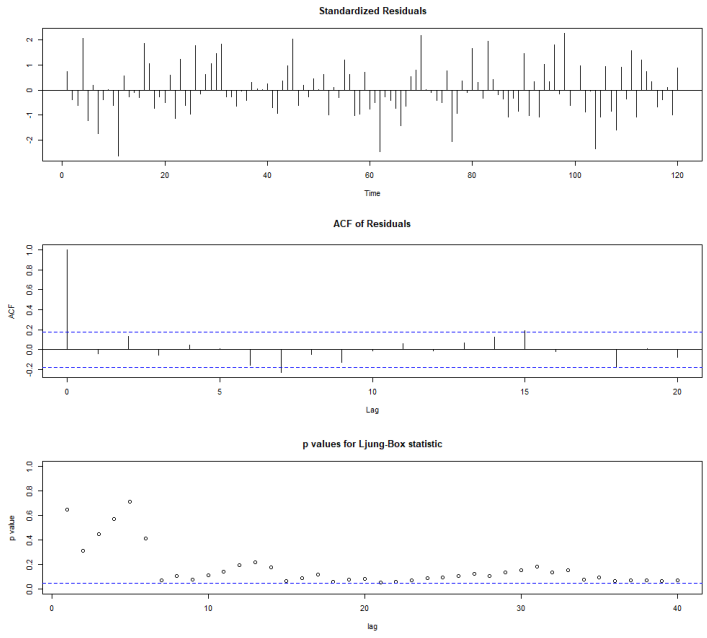

Figure 4: Diagnostic plots for up to lag 40 for AR(1) model.

From Figure 4, we can see that all of the p values are above the threshold except one which is marginally below the threshold. Also, the autocorrelations are all now non significant except for lags 7 and 15. These diagnostic plots imply the model is a good fit. We also need to look at a plot of residuals against fitted values:

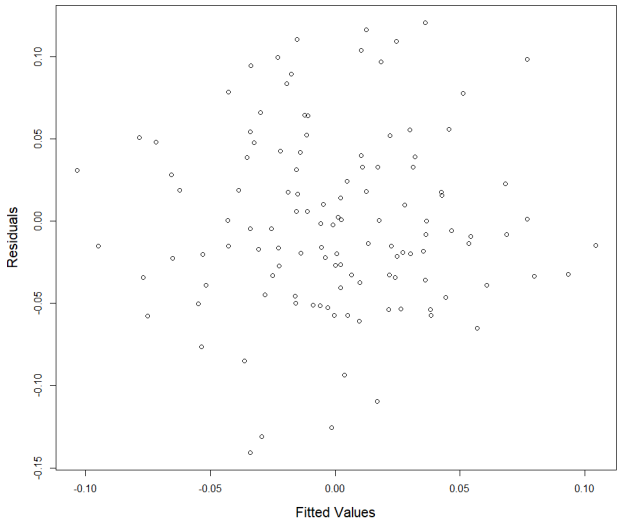

Figure 5: Residuals against fitted values for AR(1) model.

The plot shows a random scatter implying that the AR(1) model fits the data well.

1.3.2 MA(2) model

We now consider the second chosen model MA(2). As with polynomial regression, we need to assess the significance of the highest order term, β_2 in this case. The t-statistic is $t = 4.506792$ on $120 - 3 = 117$ degrees of freedom. `pt(t,117)` in `R` gives 0.9999921 implying $p < 0.00002$ thus the β_2 term is highly significant. We can now produce diagnostic plots:

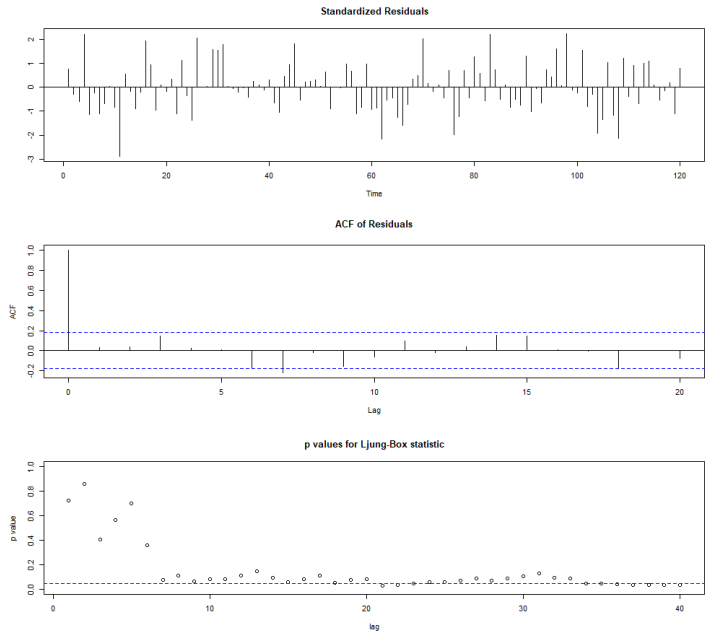

Figure 6: Diagnostic plots for up to lag 40 for MA(2) model.

From Figure 6, we can see that all of the autocorrelations are now insignificant implying a good fit. However, some of the p values for the Ljung-Box statistic are significant (marginally below the threshold). We stay wary of this fact and go on to look at the plot of residuals against the fitted values:

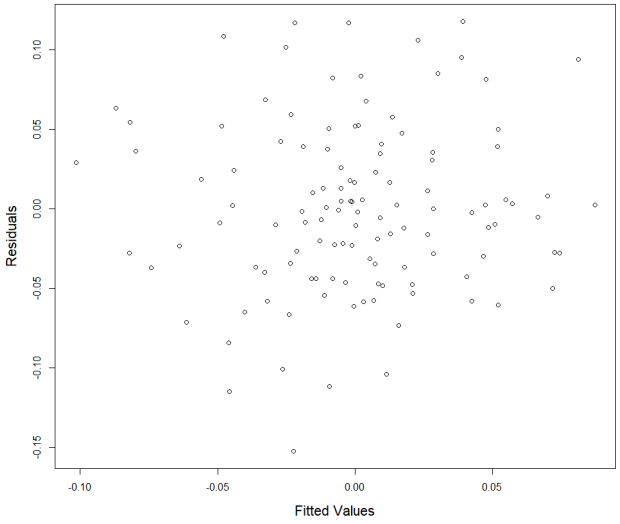

Figure 7: Residuals against fitted values for MA(2) model.

The plot shows approximately random scatter implying that the MA(2) model fits the data well.

1.3.3 ARMA(1,1) model

We now try a mixed model. α_1 is clearly highly significant with $t = 6.333986$. However, for β_1, the t-statistic is given by $t = -0.6540628$ on $120 - 3 = 117$ degrees of freedom. `pt(t,117)` in `R` implies $p > 0.05$ so this term is not significant and should be removed. Thus, we return to AR(1) and we do not need to look at the diagnostics.

1.3.4 AR(2) and MA(3)

We should now check more complex models to see if they fit the data any better. Firstly, we consider AR(2). The t-statistic for α_2 is $t = 0.7881449$ on 117 degrees of freedom and therefore `pt(t,117)` in `R` gives 0.7838971 thus, $p = 0.4322$. The p value is therefore not significant suggesting that we do not need an α_2 term. Consequently, we stick with AR(1) and there is no need to look at any diagnostics.

Next, if we fit MA(3), the t-statistic for β_3 is $t = 0.8405627$ on 116 degrees of freedom and therefore `pt(t,116)` in `R` gives 0.7988389 thus, $p = 0.4023222$. The p value is therefore not significant suggesting that we do not need an β_3 term. Consequently, we stick with MA(2) and there is no need to look at any diagnostics.

Thus, we have 2 models which seem to fit well; AR(1) and MA(2).

1.4 Choosing the model that fits the best

We have 2 models which seem to fit well; AR(1) and MA(2). We can determine which model fits the best by inspecting two goodness of fit measures, namely Akaike's Information Criterion (AIC) and the log of the Likelihood.

Model	AIC	Log Likelihood	No. of fitted parameters
AR(1)	-357.32	181.66	2
MA(2)	-356.47	182.24	3

MA(2) has the largest log Likelihood but AR(1) has the smaller AIC and has fewer parameters, thus AR(1) is to be preferred but there is relatively little to choose between them.

2 Forecasting

In this section we aim to produce forecasts of monthly electricity consumption for the period between January 2016 - June 2016 using both the AR(1) and the MA(2) models from the previous section.

2.1 AR(1)

In the AR(1) model we first removed any trend and seasonality to the data and used the log of the data, so in order to generate accurate forecasts we first had to predict the trend and seasonality of the period between January and June of 2016. We can then generate forecasts using the AR(1) model and add the predicted seasonality and trend to those forecasts to get accurate predictions throughout the 6 month period.

AR(1)			
Month (2016)	Forecast	Lower 95% confidence limit	Upper 95% confidence limit
January	40.18	36.13	44.69
February	43.98	38.87	49.77
March	35.48	31.18	40.38
April	33.47	29.36	38.16
May	29.18	25.58	33.29
June	27.10	23.75	30.93

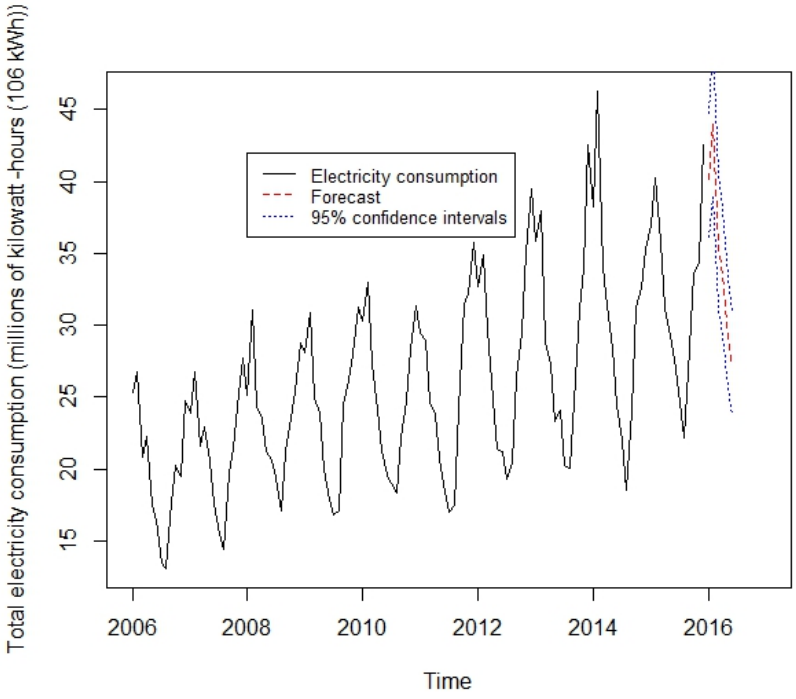

Figure 8: Time series plot of the forecasted values between January 2016 - June 2016

In order to check the reliability of this model, we fit the model to the first 9 years of data and forecast the last year between 2015 and 2016. Then we are able to compare the true value against the forecasted values and their 95% confidence limits. As shown in the table below we can see that each true value is within the 95% confidence intervals for the forecasted values and so we can say this model is reliable in its ability to generate forecasts.

AR(1)				
Month (2015)	True value	Forecast	Lower 95% confidence limit	Upper 95% confidence limit
January	36.98	35.16	31.58	39.13
February	40.23	39.72	39.72	45.06
March	35.85	32.30	28.28	36.90
April	31.16	31.12	27.17	35.63
May	29.22	27.11	23.65	31.07
June	27.64	25.24	22.02	28.95
July	24.62	22.82	19.90	26.17
August	22.18	22.01	19.19	25.24
September	28.37	28.80	25.12	33.03
October	33.59	33.59	29.29	38.53
November	34.44	36.43	31.76	41.72
December	42.53	41.60	36.28	47.72

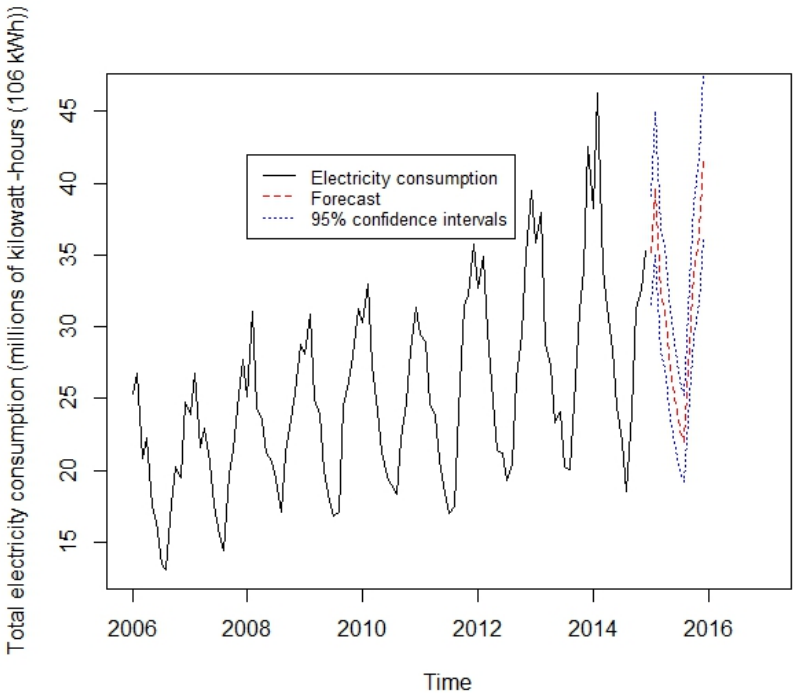

Figure 9: Time series plot of the forecasted values between January 2015 - December 2015

In Figure 9 we show the time series plot of the forecasted values for the year of 2015. Comparing this to Figure 1, we can see that the forecasts follow the same trend as the true values and that the model is able to forecast reliably within this period.

2.2 MA(2)

As before in the AR(1) model, we first needed to predict the trend and seasonality for the period between January 2016 and June 2016. We are then able to generate forecasts for the MA(2) model in the same way as the AR(1) model and adding on the predicted seasonality and trend to get accurate forecasts throughout the 6 month period.

AR(1)			
Month (2016)	Forecast	Lower 95% confidence limit	Upper 95% confidence limit
January	39.6	35.7	44.19
February	44.56	39.50	50.27
March	35.37	31.10	40.24
April	33.41	29.37	38.00
May	29.15	25.63	33.16
June	27.09	23.81	30.81

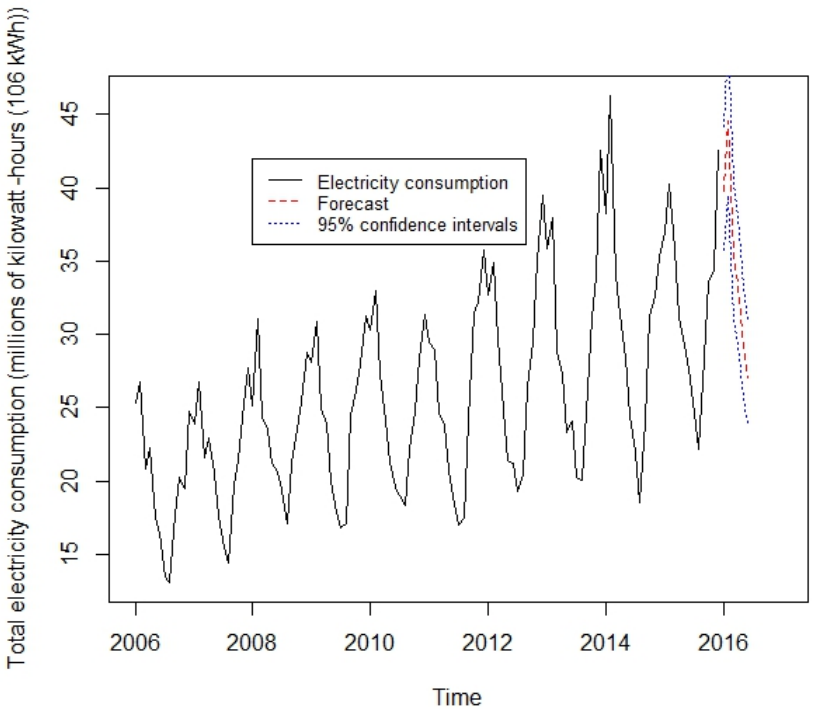

Figure 10: Time series plot of the forecasted values between January 2016 - June 2016

In Figure 10 we show the time series plot with the forecasts for the first 6 months of 2016 with their 95% confidence limits. Next we wish to check the reliability of this model to produce forecasts, so we fit the model to the first 9 years of the data and forecast the the year of 2015. In the table below we show the true value of the data in 2015 alongside with forecasts and the

95% confidence intervals. Each true value is found in within the 95% confidence intervals and so we can say this model is reliable at producing forecasts for this period of time.

AR(1)				
Month (2015)	True value	Forecast	Lower 95% confidence limit	Upper 95% confidence limit
January	36.98	34.65	31.11	38.59
February	40.23	39.82	35.1	45.09
March	35.85	33.24	29.12	37.93
April	31.16	31.68	27.76	36.16
May	29.22	27.42	24.03	31.30
June	27.64	25.43	22.28	29.02
July	24.62	22.92	20.09	26.17
August	22.18	22.08	19.34	25.19
September	28.37	28.87	25.29	32.94
October	33.59	33.64	29.48	38.40
November	34.44	36.47	31.95	41.62
December	42.53	41.64	36.49	47.53

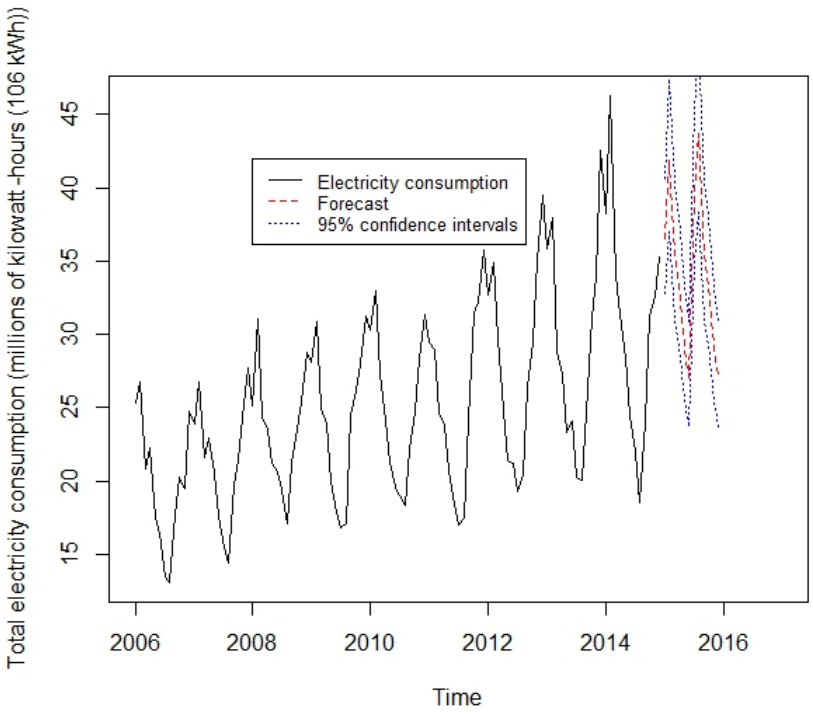

Figure 11: Time series plot of the forecasted values between January 2015 - December 2015

In Figure 11, we see the forecasts for the year of 2015 and when comparing this time series plot to Figure 1 we see that they follow the same shape showing the accuracy of this models forecasts during this time period.

A Section 1.1

```
data<-read.table("projectdata.txt")
y<-data[,51]
y$data=ts(y,start=c(2006,1),frequency=12)

plot(y$data,ylab="Total electricity consumption (millions of kilowatt-hours (10^6 kWh))")
points(y$data,pch=21,bg=1)

y$trend=filter(y$data,c(1/24,1/12,1/12,1/12,1/12,1/12,1/12,1/12,1/12,1/12,1/12,1/24))
y$trend=ts(y$trend,start=c(2006,1),frequency=12)
lines(y$trend,col=3,lty=2)
legend(2008,42,c("Electricity consumption","Trend"),col=c(1,3), pch=c(21,NA),pt.bg=c(1,NA),lty=c(1,2),cex=0.8)

y$mon=c(rep(1:12,10))  #10 years going from 1 to 12
ymonf=factor(y$mon)
time=(1:120)
fit1=lm(y$data~time+ymonf)
fit2=lm(log(y$data)~time+ymonf)
summary(fit1) #Multiple R-squared:  0.9103, Adjusted R-squared:  0.9002
summary(fit2) #Multiple R-squared:  0.9346, Adjusted R-squared:  0.9272
```

B Section 1.2

```
fit2$acf=acf(fit2$residuals,lag.max=30,ci.type="ma",main="Autocorrelation Function for Electricity Consumption Adjusted Data")
fit2$pacf=pacf(fit2$residuals,lag.max=30,main="Partial Autocorrelation Function for Electricity Consumption Adjusted Data")
```

C Section 1.3

C.1 Section 1.3.1

```
ar1=arima(fit2$residuals,order=c(1,0,0))
t=0.5941/0.0727
df=120-1-0-1
pt(t,df)
p=2*(1-pt(t,df))
tsdiag(ar1,gof.lag = 40)
plot.default(fit2$residuals-ar1$residuals,ar1$residuals,xlab="Fitted Values",ylab="Residuals",cex.lab=1.4)
```

C.2 Section 1.3.2

```
ma2=arima(fit2$residuals,order=c(0,0,2))
# We need to assess the signficance of the highest order term:
t=0.4313 /0.0957 #4.506792
df=120-0-2-1
pt(t,df)
tsdiag(ma2,gof.lag = 40)
plot.default(fit2$residuals-ma2$residuals,ma2$residuals,xlab="Fitted Values",ylab="Residuals",cex.lab=1.4)
```

C.3 Section 1.3.3

```
ar1ma1=arima(fit2$residuals,order=c(1,0,1))
t=-0.0813 /0.1243
df=120-1-1-1
pt(t,df)
p=2*(1- pt(t,df))
```

C.4 Section 1.3.4

```
ar2=arima(fit2$residuals,order=c(2,0,0))
t=0.0718 /0.0911
df=120-2-0-1
pt(t,df)
2*(1-  pt(t,df))

ma3=arima(fit2$residuals,order=c(0,0,3))
t=0.0717 /0.0853
df=120-0-3-1
pt(t,df)
2*(1- pt(t,df))
```

D Section 2.1

```
#########AR1 2016 predictions##############
predicted.trend = fit2$coefficients[1] + fit2$coefficients[2]*(121:126)
season=c(0,fit2$coef[3],fit2$coef[4],fit2$coef[5],fit2$coef[6],fit2$coef[7])
predicted.trend.season = predicted.trend+season
ar1F<-predict(ar1,n.ahead=12)
ar1F$pred=ts(ar1F$pred,start=c(2016,1),frequency=12)
ar1FT<-ar1F$pred+predicted.trend.season
ar1F$se=ts(ar1F$se,start=c(2016,1),frequency=12)
ar1FTU=ar1FT+2*ar1F$se
ar1FTL=ar1FT-2*ar1F$se

#############AR1 2015 predictions###############
y=data[1:108,51]
y$data=ts(y,start=c(2006,1),frequency=12)
y$mon = c(rep(1:12,9))
ymonf=factor(y$mon)
time=(1:108)
fit2=lm(log(y$data)~time+ymonf)
ar1=arima(fit2$residuals ,order=c(1,0,0))
predicted.trend = fit2$coefficients[1] + fit2$coefficients[2]*(109:120)
season=c(0,fit2$coef[3],fit2$coef[4],fit2$coef[5],fit2$coef[6],fit2$coef[7],
         fit2$coefficients[8],fit2$coefficients[9],fit2$coefficients[10],fit2$
    coefficients[11],fit2$coefficients[12],fit2$coefficients[13])
predicted.trend.season = predicted.trend+season
ar1F<-predict(ar1,n.ahead=12)
ar1F$pred=ts(ar1F$pred,start=c(2015,1),frequency=12)
ar1FT<-ar1F$pred+predicted.trend.season
ar1F$se=ts(ar1F$se,start=c(2015,1),frequency=12)
ar1FTU=ar1FT+2*ar1F$se
ar1FTL=ar1FT-2*ar1F$se
```

E Section 2.2

```
################MA2 2016 predictions################
predicted.trend = fit2$coefficients[1] + fit2$coefficients[2]*(121:126)
season=c(0,fit2$coef[3],fit2$coef[4],fit2$coef[5],fit2$coef[6],fit2$coef[7])
predicted.trend.season = predicted.trend+season
ma2=arima(fit2$residuals ,order=c(0,0,2))
ma2F<-predict(ma2,n.ahead=6)
ma2F$pred=ts(ma2F$pred,start=c(2016,1),frequency=12)
ma2FT<-ma2F$pred+predicted.trend.season
ma2F$se=ts(ma2F$se,start=c(2016,1),frequency=12)
ma2FTU=ma2FT+2*ma2F$se
ma2FTL=ma2FT-2*ma2F$se

#################MA2 2015 predictions#############
y=data[1:108,51]
y$data=ts(y,start=c(2006,1),frequency=12)
y$mon = c(rep(1:12,9))
ymonf=factor(y$mon)
time=(1:108)
fit2=lm(log(y$data)~time+ymonf)
ma2=arima(fit2$residuals ,order=c(0,0,2))
predicted.trend = fit2$coefficients[1] + fit2$coefficients[2]*(109:120)
season=c(0,fit2$coef[3],fit2$coef[4],fit2$coef[5],fit2$coef[6],fit2$coef[7],
         fit2$coefficients[8],fit2$coefficients[9],fit2$coefficients[10],fit2$
    coefficients[11],fit2$coefficients[12],fit2$coefficients[13])
ma2F<-predict(ma2,n.ahead=12)
ma2F$pred=ts(ma2F$pred,start=c(2015,1),frequency=12)
ma2FT<-ma2F$pred+predicted.trend.season
ma2F$se=ts(ma2F$se,start=c(2015,1),frequency=12)
ma2FTU=ma2FT+2*ma2F$se
ma2FTL=ma2FT-2*ma2F$se
```