5/25/2016

Forecast of Mobile Network Data Download Volumes (LTE)

Danielle Ormandy

Forecast of Mobile Network Data Download Volumes (LTE)

Introduction

Currently, there are as many mobile subscriptions as people in the world, and every second, 20 new mobile broadband subscriptions are activated. With the increase in the number of subscribers, comes an increase in the data downloaded through mobile telephony networks. Mobile data traffic has grown 4,000-fold over the past 10 years and almost 400-million-fold over the past 15 years.

A mobile network is a complex amalgamation of radio equipment, nodes and interconnecting transmission which must process and transport the data. Increases in data volumes must be catered for, and accurate forecasts of future traffic volumes are an important part of network planning.

This report examined the Mobile Data downloaded daily from one LTE (4G) mobile telephony network with the aim of:

- Forecasting future Mobile Data Download Volume
- Quantifying the contribution of Seasonality and Annual events to network traffic

Description of the Data Series

The data series used as input for this report consists of the Daily Download Volume (TB) from a LTE (4G) Mobile Telephone network from April 2015 until April 2016. Figure 1 shows a time plot of the series.

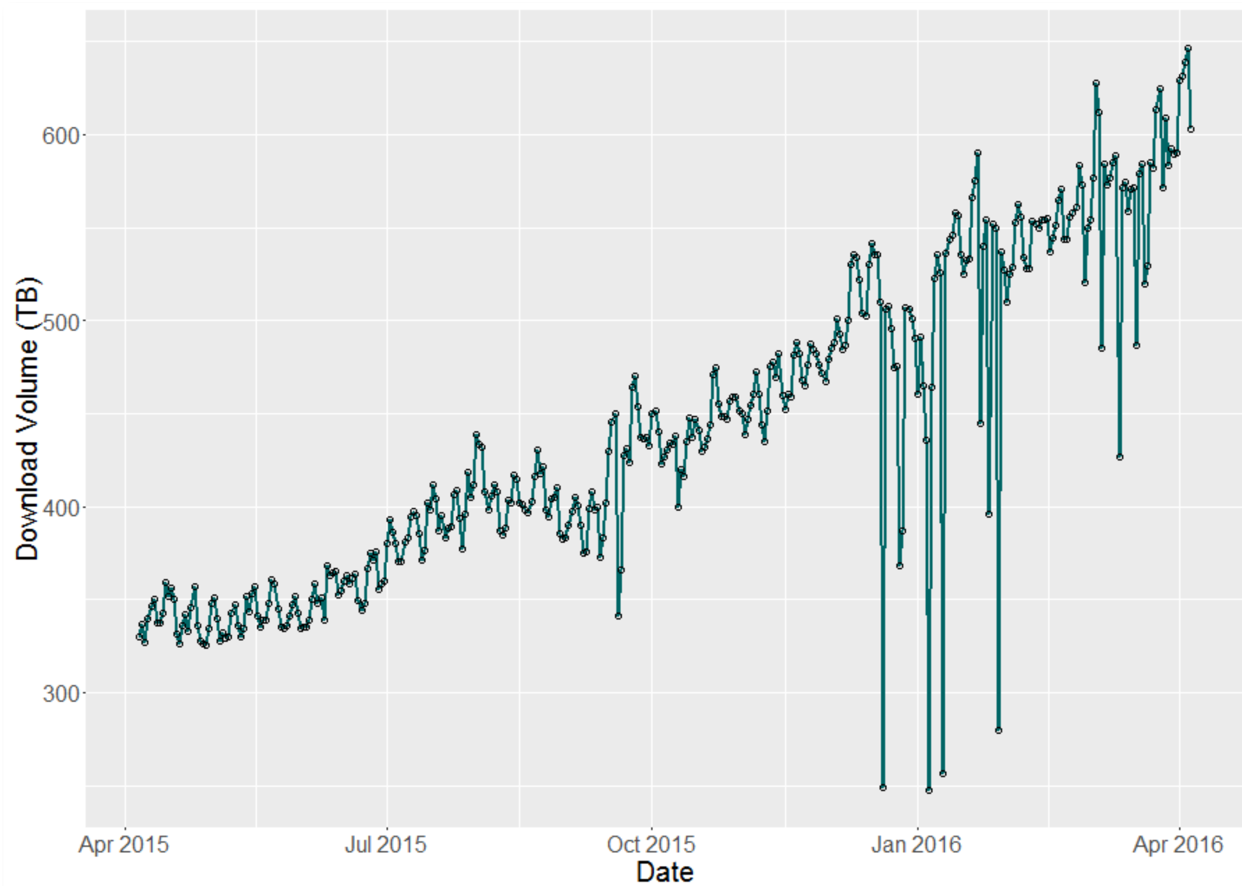

Figure 1 LTE Mobile Network Daily Download Volume (TB) – Original Data Series

It can be observed that the series has strong upward trend and seasonality that appears to be weekly. There are major downward disruptions to the data series in December, January and February. Other smaller irregularities exist in the time series in August 2015 and March 2016. Excluding the irregularities, the variance appears constant with time.

Analysis of the Data Series

Before analyzing the time series, the disruptions were first handled through by passing the data through a function to find and correct outliers. The resulting time series is shown in Figure 2. It can be observed the cleaned data series has the same strong upward trend and weekly (7 day) seasonality. The variance fluctuates across the data series but does not increase or decrease with time. The irregularities in August, December and February remain, but the variation of the irregularities is reduced.

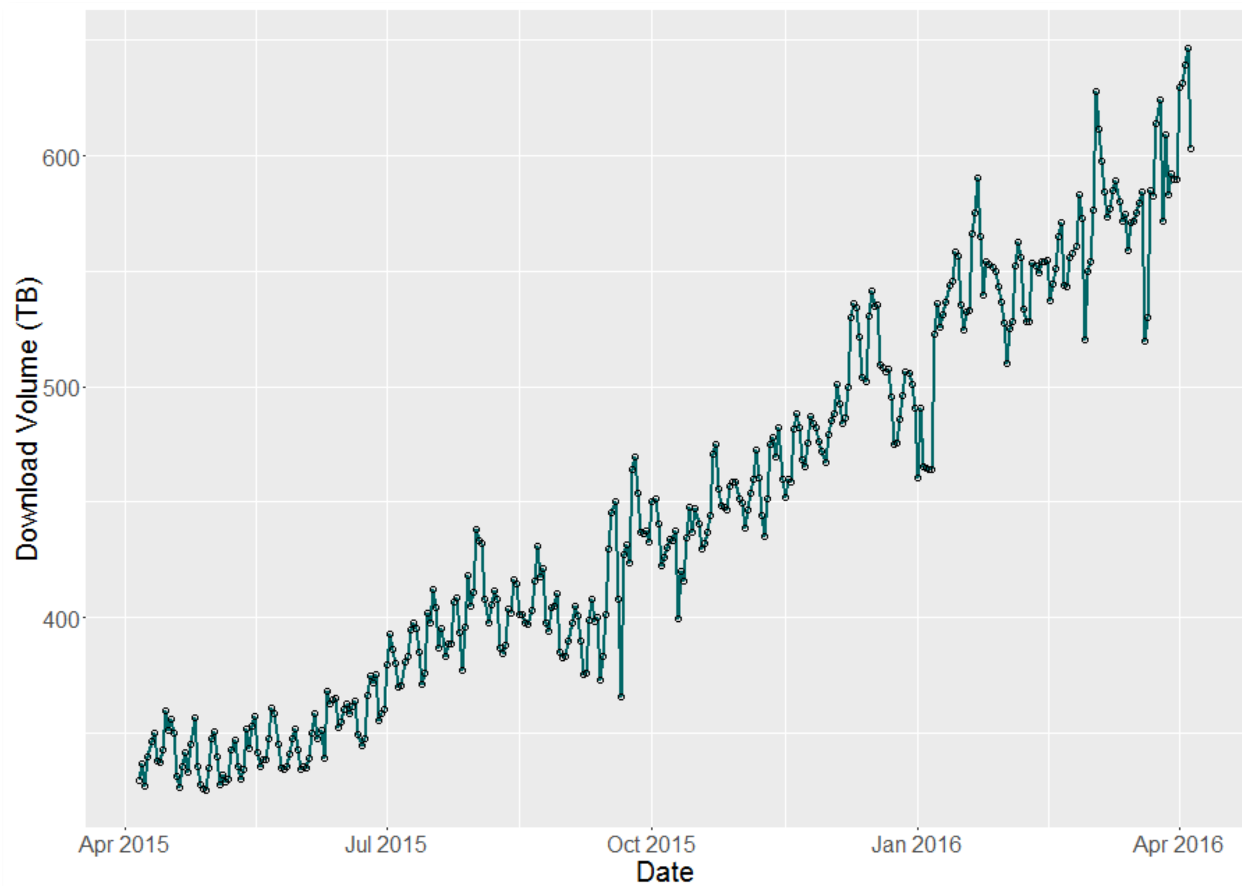

Figure 2 LTE Mobile Network Daily Download Volume (TB) – Cleaned Data Series

Decomposition of the Data Series

The decomposition of the cleaned data series shows there is a clear seasonal component, although it is the smallest component. The seasonal component, which is the weekly variation, can be seen to create a variation of +/- 10TB.

The trend appears linear and is the largest component of the data series with the variation introduced by the irregularities evident.

The remaining data is also a major component of the data series and captures some of the noise of the irregularities. These irregularities can be seen to have a negative impact on the data series in the order of 20 to 30 TB.

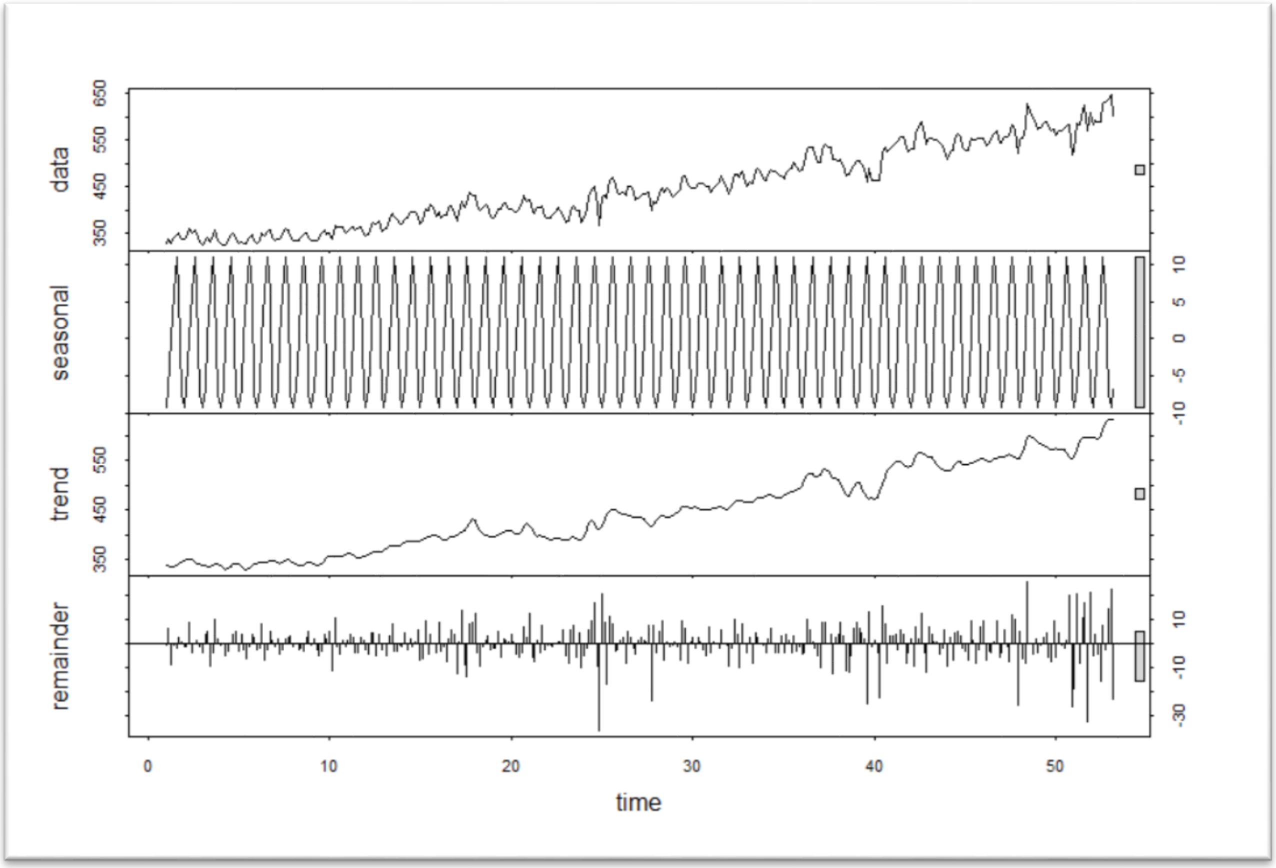

Figure 3 Time Series Decomposition

ACF and PACF Plots

The ACF and PACF plots of the time series, shown in Figure 4 indicate the non-stationary nature of the data series. The ACF plot shows significant autocorrelations at all lags. The ACF plot values decay slowly with slight peaks at lags 7, 14 and 21 . The PACF plot shows the partial autocorrelations are significant at lags 4, 5 and 8. There do not appear to be any regularity in the partial autocorrelations.

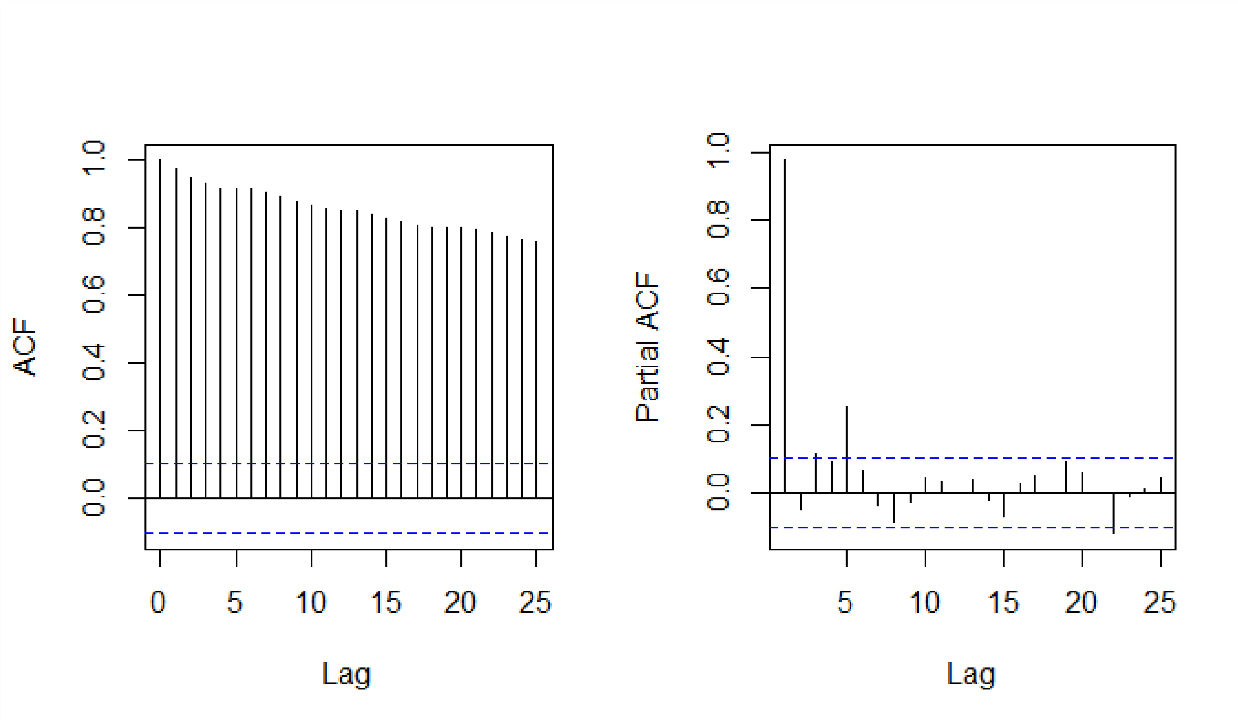

Figure 4 ACF and PACF Plots of the Cleaned Mobile Network Daily Download Volume (TB) data series

Transformation of the Data

The decaying ACF plot confirms that the time series is not stationary. The slight peaks at lags 7, 14 and 21 indicate that differencing at lag 7 is required to remove the weekly seasonality.

The time series plot of the data shows that the variance does not increase or decrease with time and no further transformation is required.

Time Series differenced at Lag 7

The time series, ACF and PACF plots of the data differenced at lag 7 are shown in Figure 5.

The time series plot appears to have a mean close to zero, which indicates that the trend has been removed successfully from the time series. The ACF plot displays a a pattern of decay with many values outside the significance bounds. The PACF shows significant values at lower lag values, becoming less significant after lag 20. The decrease of the ACF and PACF plots with increasing lags indicates the data series may need to be differenced again, this time at lag 1.

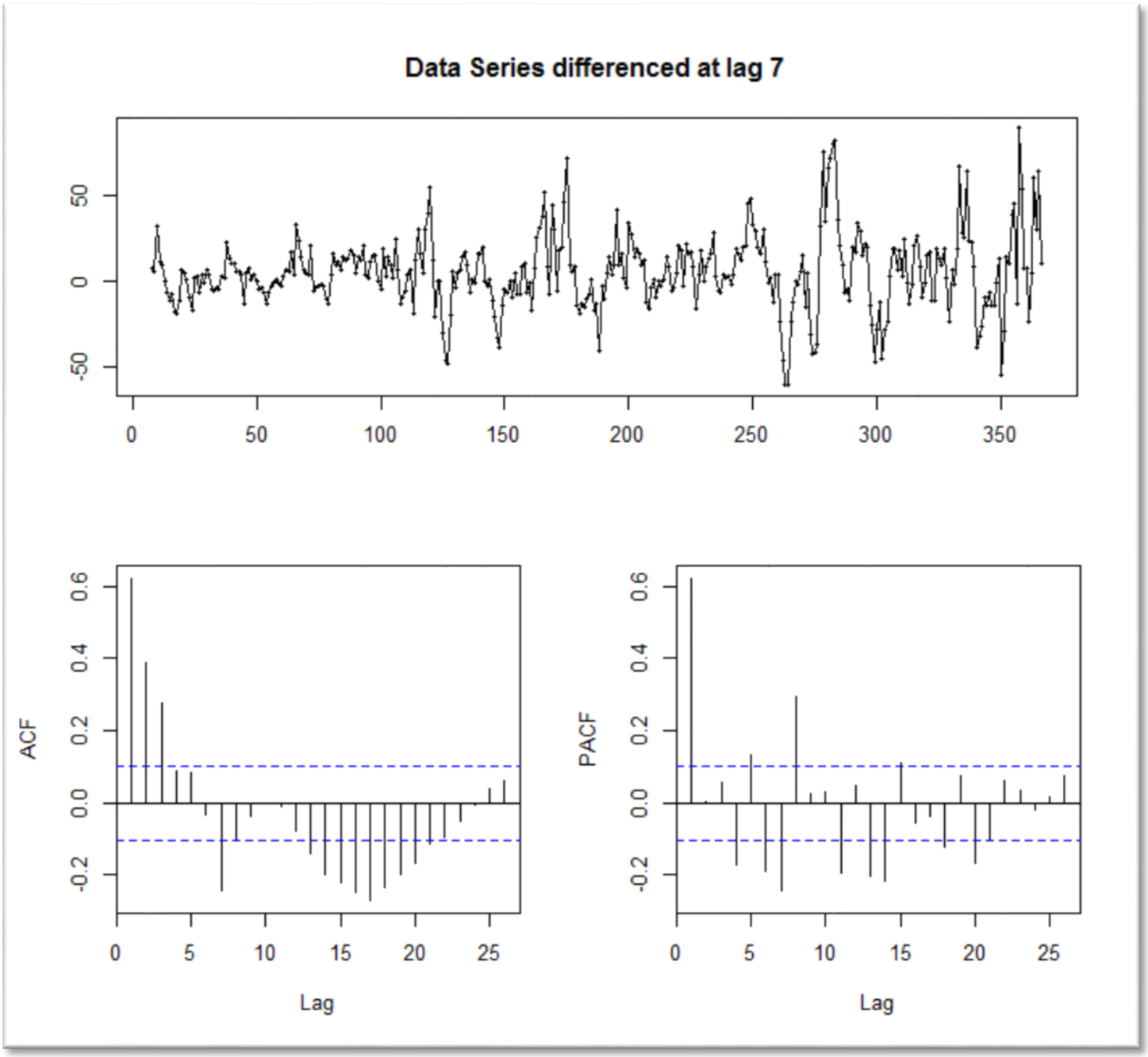

Figure 5 Time Series, ACF and PACF Plots of the Cleaned Data series Differenced at Lag 7

Time Series differenced at Lag 7 and Lag 1

The time series, ACF and PACF plots of the data differenced at lag 7 and then at lag 1 are shown in Figure 6. The time series of the twice differenced data oscillates more frequently around the mean. Both ACF and PACF plots have no discernable pattern in either, although both plots show significant values at various lag values.

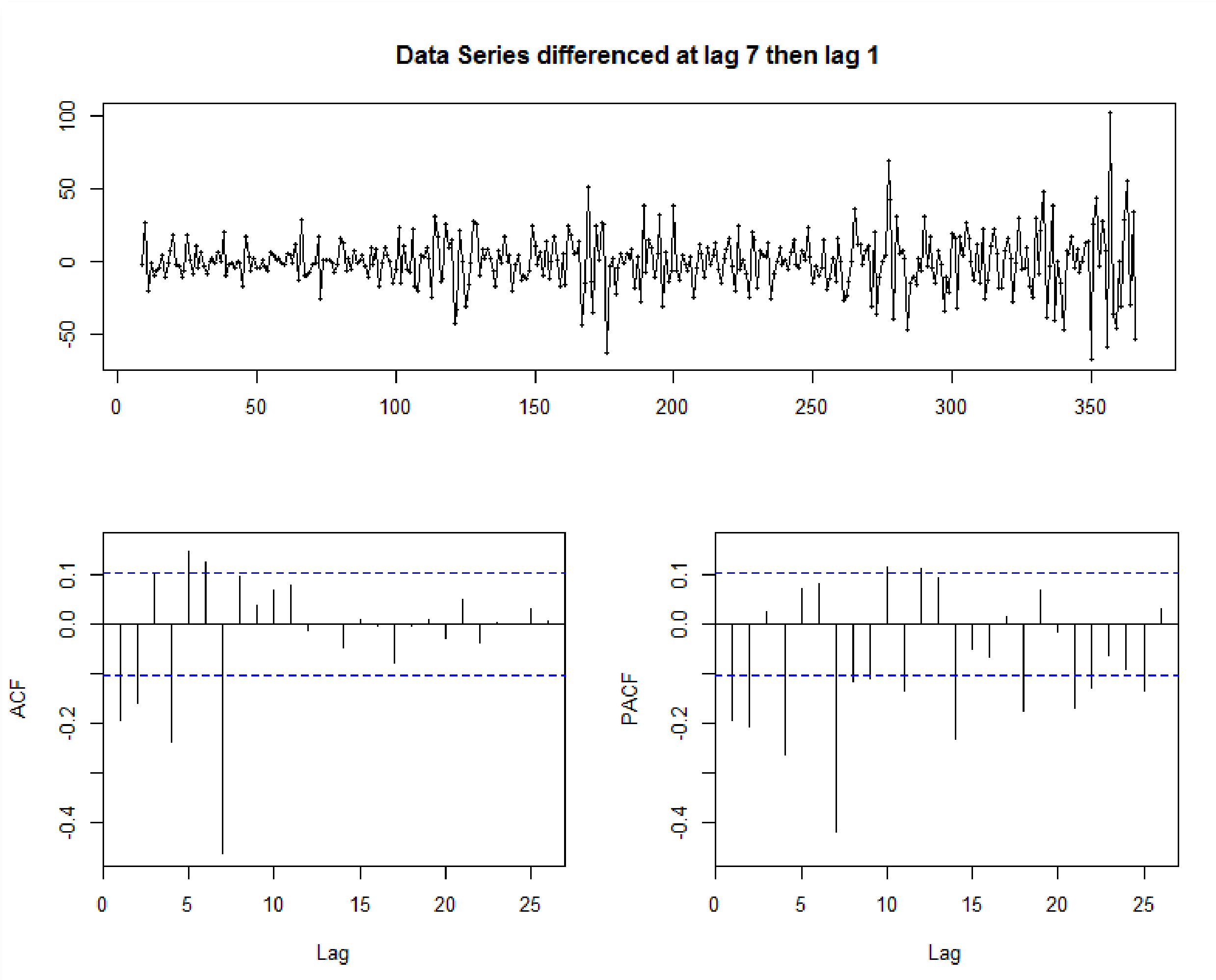

Figure 6 Time Series, ACF and PACF Plots of the Cleaned Data series Differenced at Lag 7 and Lag 1

Unit Root Tests

Statistical hypothesis tests of stationarity were performed on the data differenced at lags 7 and 1 to determine if stationarity had been achieved.

The results shown in Table 1 confirm the differenced data series passes the Unit Root tests for stationarity

Table 1 Unit Root Test Results

Test Type	Null Hypothesis	P-value	Result
Augmented Dickey-Fuller (ADF)	Data is non-stationary	0.01	Data is stationary within 95% CI
Kwiatkowski-Phillips-Schmidt-Shin (KPSS)	Data is stationary	0.1	Data is stationary within 95% CI

ARIMA Modelling

The ARIMA model suggested from the ACF and PACF charts of the differenced data is

- AR of order no more than 7
- MA model also no more than 7
  where the
- Trend difference is 1 and
- Seasonality difference is 7

Excluding the last 10 weeks from the data series, 42 weeks of data were used as the basis for the ARIMA modelling. Different models were trialed and compared on the basis of the lowest AICc value and residuals with the highest statistical indicators of randomness. Note: Some Order combinations did not resolve and are therefore not listed.

As shown in Table 2 the model with the lowest AICc value is AR Order 5 and MA Order 6 with differencing at lag 1 and Seasonality of 7. The p-value from the Ljung-Box test (h=14) of the residuals is greater than 0.05, which mean the residuals from this model pass this test of randomness.

Table 2 AICs and Ljung-Box Test results for different orders of ARIMA Models

ARIMA Order	AICc	Ljung-Box Test p-value (h=14)
$(2,1,2)(0,1,0)_7$	2310.28	0.0035
$(2,1,3)(0,1,0)_7$	2342.62	0.0000
$(2,1,4)(0,1,0)_7$	2294.66	0.1564
$(2,1,5)(0,1,0)_7$	2296.64	0.1175
$(2,1,6)(0,1,0)_7$	2293.45	0.4430
$(2,1,7)(0,1,0)_7$	2295.28	0.3172
$(3,1,2)(0,1,0)_7$	2308.93	0.0037
$(3,1,3)(0,1,0)_7$	2311.01	0.0036
$(5,1,4)(0,1,0)_7$	2302.24	0.1189
$(5,1,6)(0,1,0)_7$	**2292.82**	**0.0611**
$(6,1,2)(0,1,0)_7$	2311.31	0.6019
$(6,1,6)(0,1,0)_7$	2299.62	0.0575
$(7,1,2)(0,1,0)_7$	2312.79	0.6193

Residual Diagnosis

The residuals of the ARIMA Model (5,1,6) $(0,1,0)_7$ are shown in Figure 6. The residual analysis from the ARIMA model had the following results:

- The residuals are largely uncorrelated
- The residuals have a mean of 1.375

The time plot of the residuals shows that the variation of the residuals stays the same for most of the time across the historical data, so the residual variance can be treated as constant.

A non-zero mean indicates a bias that needs to be corrected by adding the value 1.375 to all forecasts. The residuals have no correlations of significance excepting the small spike at lag 16.

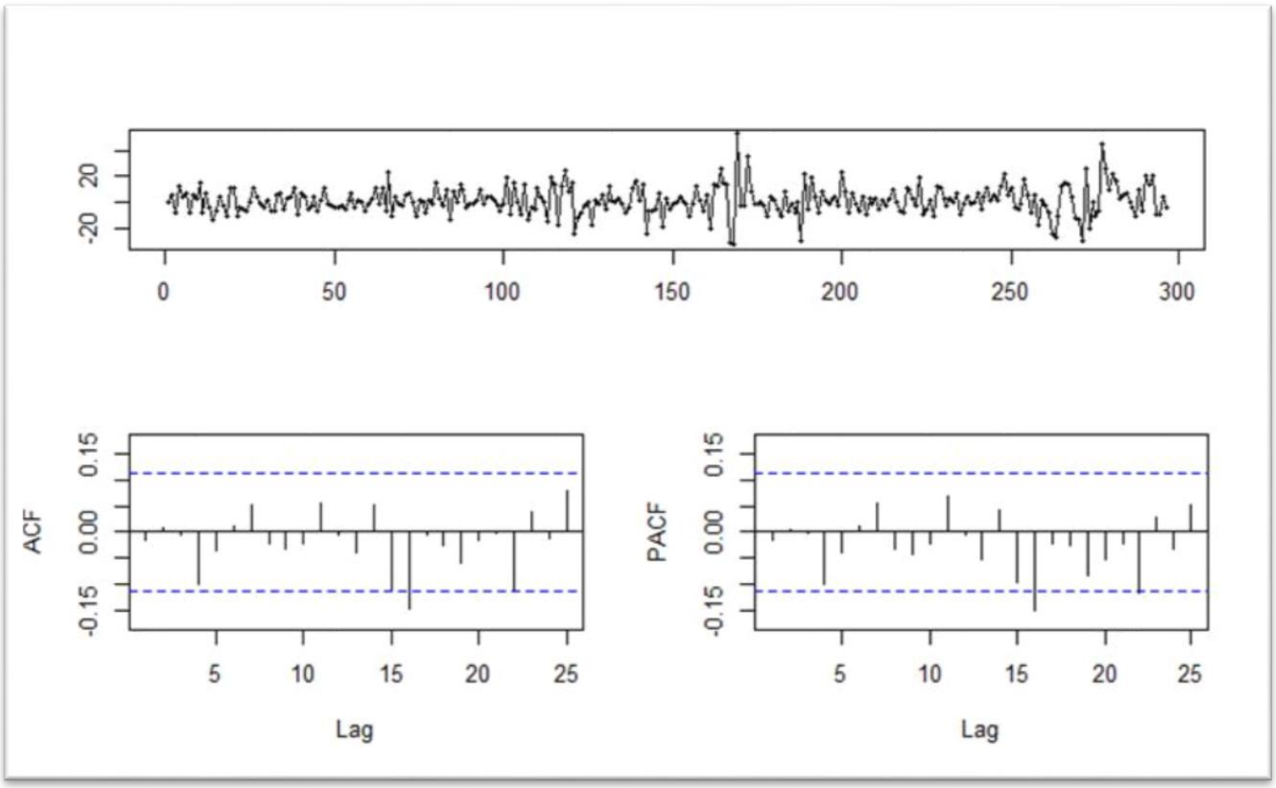

Figure 6 Residuals Analysis Plots of ARIMA Model (5,1,6) $(0,1,0)_7$

The histogram plot and cumulative periodogram of the residuals is shown in Figure 7. The histogram shows the residuals mostly fit a normal distribution, with slight positive skewness. The cumulative periodogram of the residuals follows the 45° line and is within the confidence bands.

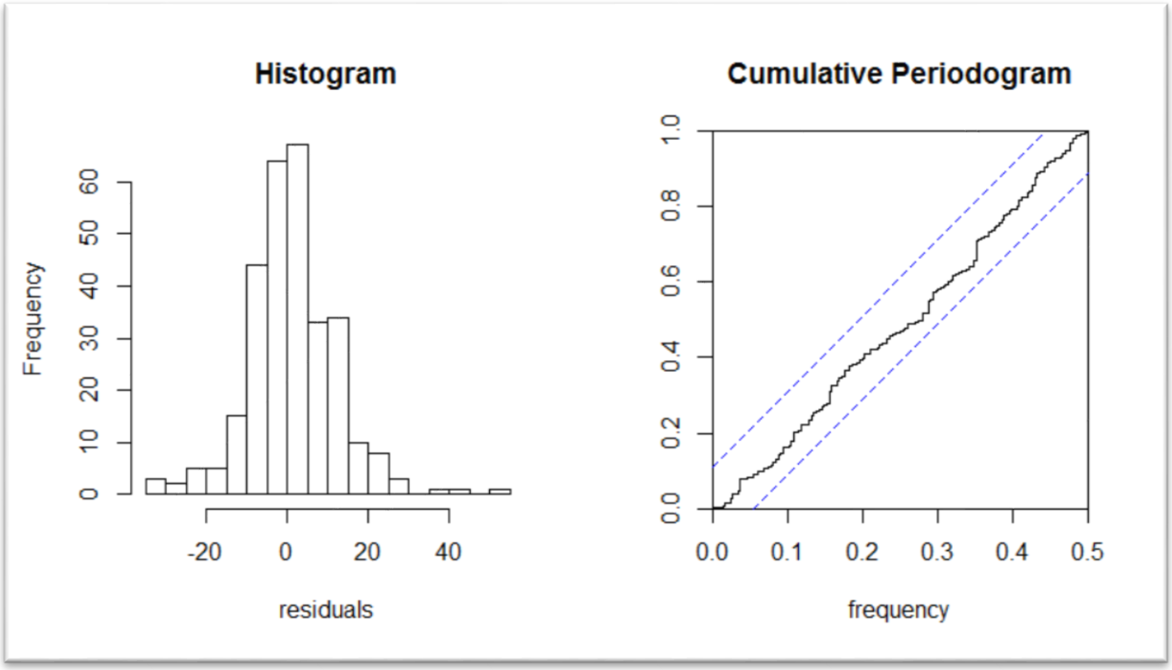

Figure 7 Histogram and Cumulative Periodogram Plot of residuals

On the basis that the residuals pass the criteria for randomness, the ARIMA model $(5,1,6)(0,1,0)_7$ appears to account for all available information and is determined to be suitable to be used for forecasting of this dataset.

The parameters from this ARIMA Model are shown in Table 2 below.

For each coefficient, z = estimated coeff. / std. error of coeff. is calculated. If $|z| > 1.96$, the estimated coefficient is significantly different from 0.

It can be seen that the coefficients for AR1, AR5, MA1 and MA5 could be excluded from the model without having much significance on the modelling result.

Table 2 Coefficient parameters for ARMA (5,1,6,D=7) estimation

	ar1	ar2	ar3	ar4	ar5	ma1	ma2	ma3	ma4	ma5	ma6
Coeff	-0.226	-0.105	-0.277	-0.976	0.041	0.097	-0.201	0.143	0.918	-0.217	-0.314
s.e	0.253	0.067	0.034	0.070	0.250	0.249	0.059	0.033	0.054	0.234	0.079
|z|	0.892	1.564	8.198	14.006	0.165	0.391	3.393	4.382	17.071	0.926	3.958

The formula for this model, in terms of the backshift operator is:

$$\boxed{\begin{aligned}(1 + 0.226B + 0.105B^2 + 0.277B^3 + 0.976B^4 - 0.041B^5)(1 - B)(1 - B^7)Y_t \\ = (1 - 0.097B + 0.201B^2 - 0.143B^3 - 0.918B^4 + 0.217B^5 + 0.314B^6)e_t\end{aligned}}$$

where Y_t is the time series at period t
and e_t is the random shock (noise) occurring at time t.

Forecasts

ARIMA Model Forecast

The ARIMA Model $(5,1,6)(0,1,0)_7$ was used to forecast a further 10 weeks of data. The forecast result, with 80% and 95% confidence intervals and the actual data is shown in Figure 8. Note that the x-axis represents the number of data points in the series (366) and not the date. The plot shows seasonality is well represented in the forecast, but the upward trend is not captured adequately. Actual values are within 80% confidence intervals

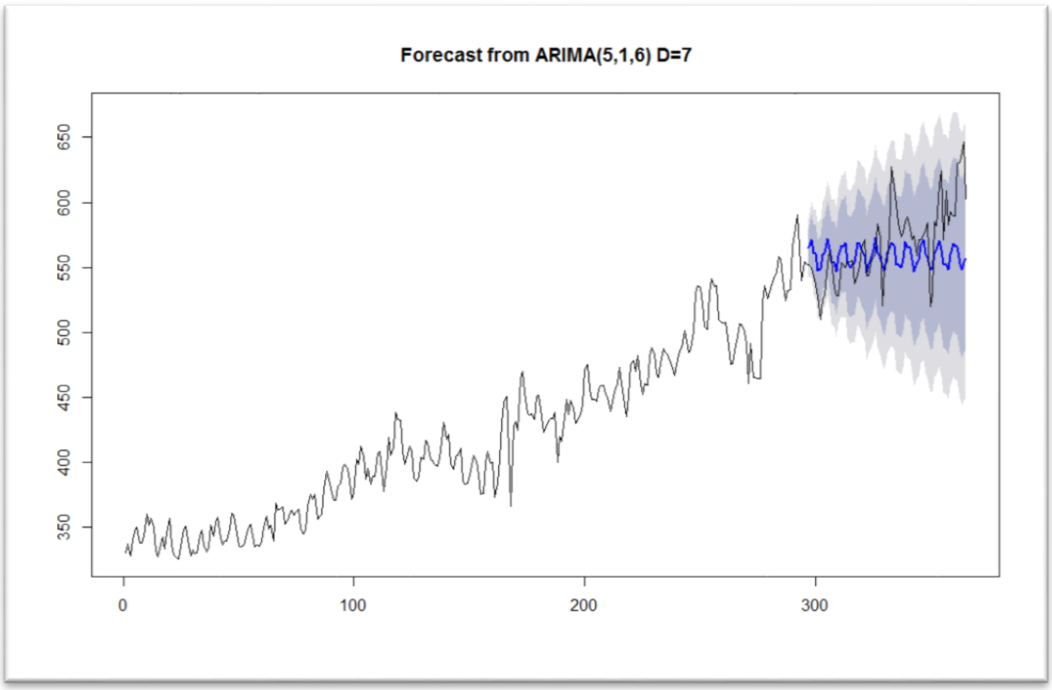

Figure 8 Forecast result from ARIMA $(5,1,6)(0,1,0)_7$ Model

Naive Forecast

Two methods of naïve forecast: Naïve and Drift method were used to produce a forecast for 10 weeks. A plot of the results is shown in Figure 9. Note that the x-axis represents the number of data points in the series (366) and not the date. The Naïve method, using the last known value of the trial series as the forecast result, does not provide a good match for the trend of this data series. However, the Drift method is able to capture the upwards trend quite well.

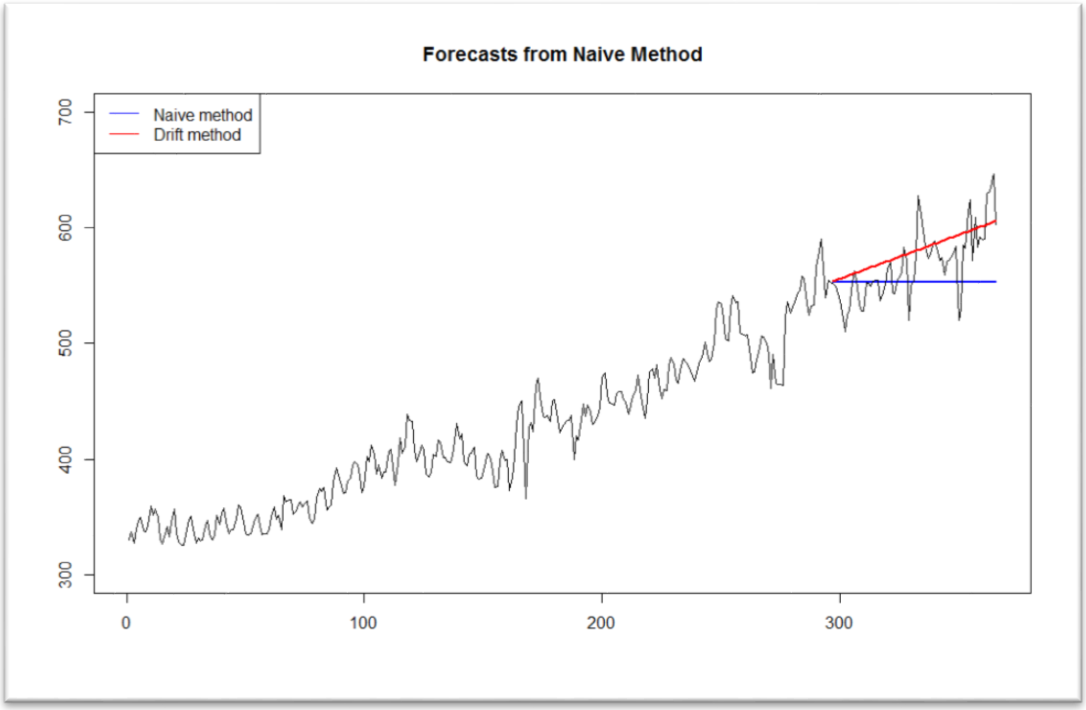

Figure 9 Forecast Results from Naïve Methods

Exponential Smoothing Forecast

The different exponential smoothing forecast methods were used to produce a forecast for 10 weeks. A plot of the results is shown in Figure 10. Note that the x-axis represents the number of data points in the series (366) and not the date. R Programming functions were used to find the optimal values of alpha and beta for the Exponential smoothing models. The functions were not able to determine the seasonality of the data, and a values of alpha, beta and gamma which minimized the MAPE were found instead by trial and error.

The values of the parameters are shown in Table 3.

Table 3 Exponential Smoothing Parameters

	α	β	γ	Initial State(s)
Exp. Smoothing	0.99	n/a	n/a	l = 340.18
Holt's method	0.99	1e-04	n/a	l = 330.73 b = 0.77
Holt-Winter's method	0.99	0.01	0.09	l = 329.68 b = 7.04

The Exponential Smoothing method is very similar to the Naïve forecast, the only difference being the initial value chosen for the forecasts. The Holt's Method forecast appears similar to the Drift Method result capturing the required upward trend and the Holt-Winter's forecast appears to have too great a trend and does not reflect the required seasonality.

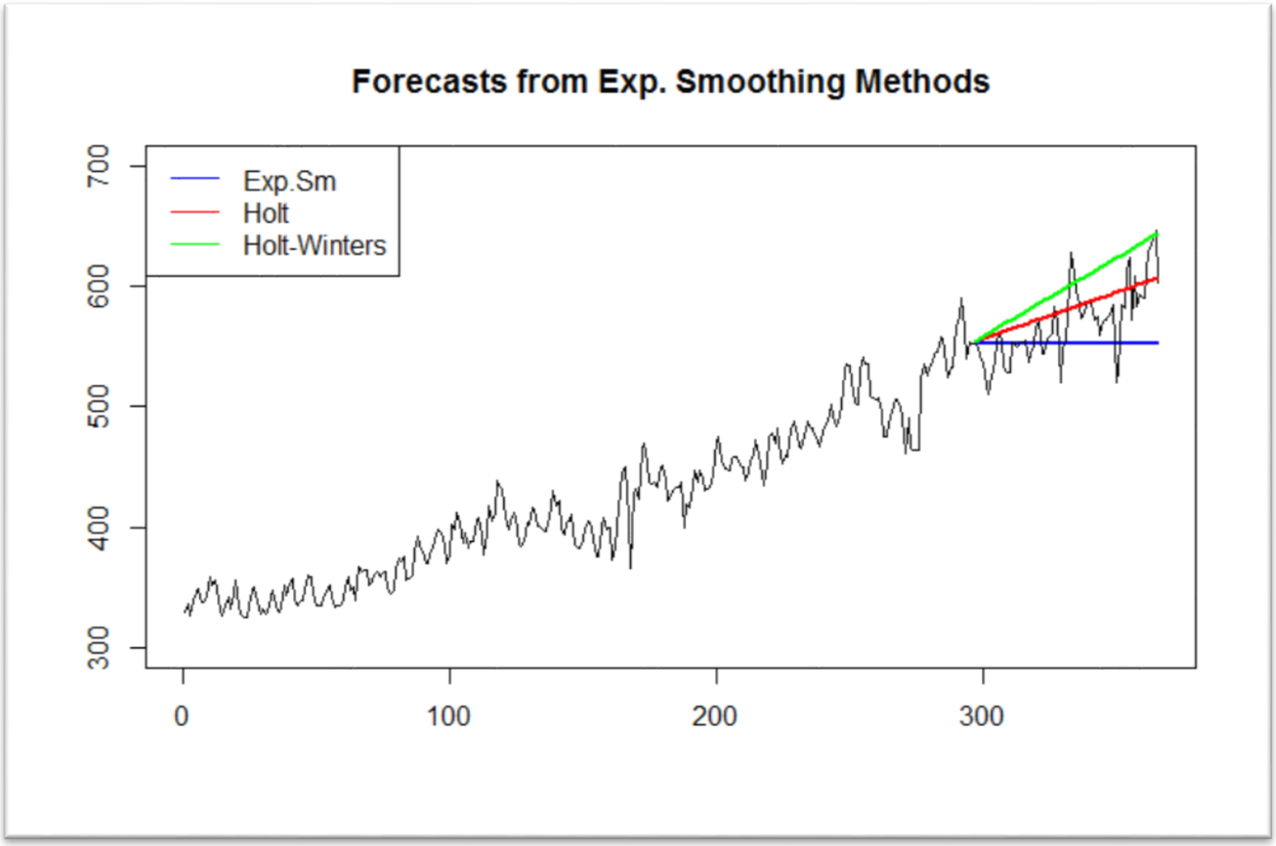

Figure 10 Forecast Results from Exponential Smoothing Methods

Parzan's ARAR Forecast

ITSM was used to produce a forecast for 10 weeks of data using Parzan's ARAR Model. The result is shown in Figure 11. Note that the x-axis represents the number of data points in the series (366) and not the date. The model provides a fairly accurate forecast for the first weeks of the forecast, however the trend prediction appears to sit too high above the data series.

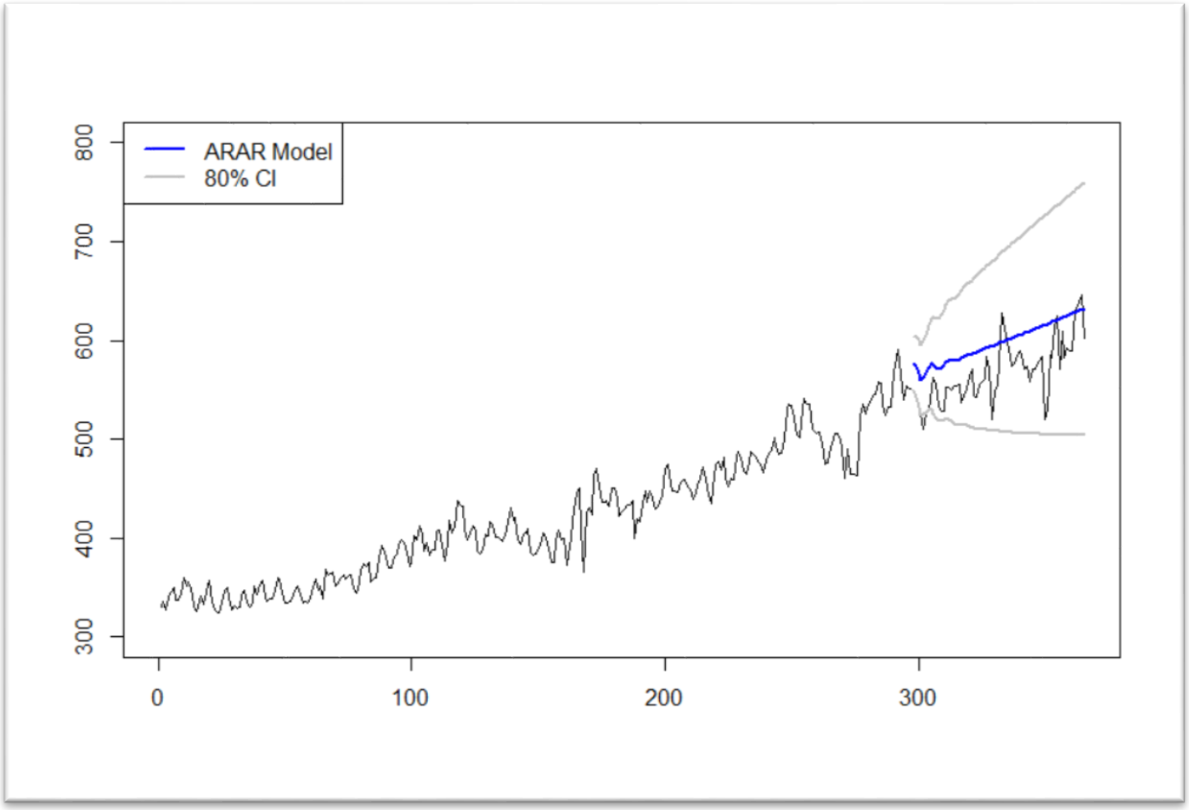

Figure 11 Forecast Results from Parzan's ARAR Method

Forecast Accuracy Comparisons

The accuracy of the different forecast methods is presented in Table 4. It can be seen that the Drift method results in a forecast with the best accuracy result by all measures except Mean Error (ME), where the ARIMA model returns the lowest ME result. Holt's Method also produced good accuracy results in comparison to the other forecast methods.

Table 4 Forecast Accuracy comparisons of Different Forecast Methods

Forecast Method	ME	RMSE	MAE	MPE	MAPE
ARIMA $(5,1,6)(0,1,0)_7$	9.53	31.36	24.20	1.42	4.16
Naive	15.38	34.05	25.77	2.43	4.40
Drift Method	-11.50	24.71	19.68	-2.21	3.51
Exp. Smoothing	15.38	34.05	25.77	2.43	4.40
Holt's method	-11.92	24.84	19.87	-2.28	3.55
Holt-Winter's method	-30.74	37.32	31.87	-5.53	5.71

Parzan's ARAR method	-30.00	36.54	31.90	-5.44	5.74

On the basis of these results the Drift Method is used to produce a forecast for the next 10 weeks of data based on the original cleaned data set.

Forecast using Drift Method

The forecast for 10 weeks to the 14th June using the Drift Method on the original cleaned data set, together with 80% and 95% Confidence intervals is shown in Figure 12. The forecast can be seen to follow the general trend of the original data set and results in a forecast of 655.36 TB within 95% CI [384.97, 925.75].

This forecast indicates that Downlink Throughput Volumes will increase by 52.41 TB in 10 weeks.

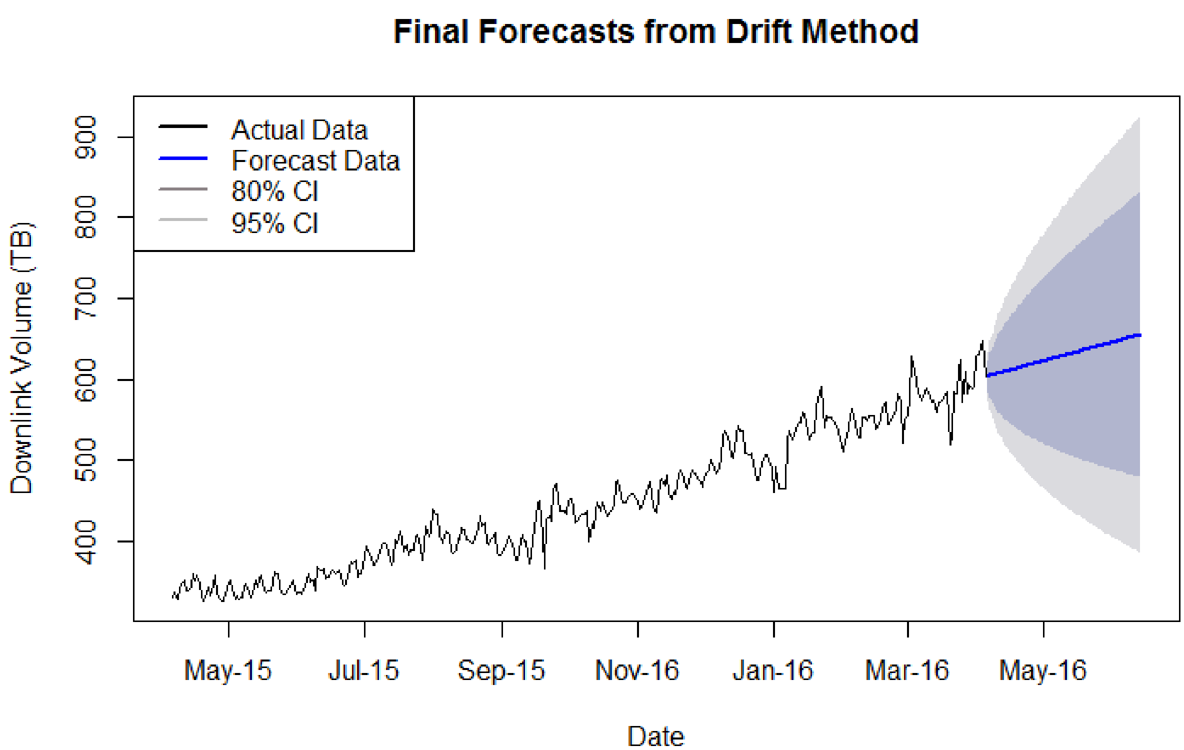

Figure 12 Final Forecast Results using Drift Method

Conclusion

The result of the forecast analysis of Mobile Network Downlink Data Volumes is that it was difficult to find a model that could accurately fit the fluctuating variance of the data series. The best forecast models were those that most accurately matched the trend of the data which were the Drift Method followed by Holt's method.

One of the aims of this forecasting analysis was to quantify the contribution of seasonality and annual events to network traffic. The decomposition of the cleaned data series showed the seasonal component, which is the weekly variation, is a variation of +/- 10TB on the data series. The impact of annual events such as Christmas and major holidays is to disrupt the data in a negative direction in the order 20 to 30TB.

The forecast result from the Drift Method for 10 weeks of data predicts that Mobile Network Downlink Data volumes will increase to 655.36 TB within 95% CI [384.97, 925.75] by the 14th June, an increase of 52.41 TB.

Appendix

R Program Commands used for Forecast Analysis

```
#Load the Libraries we need
library(forecast)
library(xts)
library(ggplot2)
library(zoo)
library(ggfortify)
library(fpp)
library(MASS)

#Set the working directory
setwd("F:/RMIT_MasterofAnalytics/Forecasting/Report")

# Read Dataset from csv file
data <-
read.csv(file="F:/RMIT_MasterofAnalytics/Forecasting/Report/MobileNetwork_DownloadVol
ume_dates.csv",header=TRUE,sep=',')
data$time <- as.Date(as.character(data$time),format="%d/%m/%Y")
head(data)

#Plot the data
ggplot(data, aes(time, DL_Vol)) + geom_line(color = "black") +
geom_line(color="#006666",size=1) + geom_point(shape=1, size=2) + xlab("Date") +
ylab("Download Volume (TB)") + theme(text = element_text(size=20))

#Convert DL_Vol to time series
DLV.ts <- xts(data$DL_Vol,as.Date(data$time))

#Clean the data, reformat and export for ITSM functions
DLV_cl.ts <- tsclean(as.ts(DLV.ts))
write.table(DLV_cl.ts,file="F:/RMIT_MasterofAnalytics/Forecasting/Datasets/MDVclean.c
sv",row.names=FALSE)
DLV_cl <- data.frame(time=data$time, DL_Vol=as.matrix(DLV_cl.ts)) #Will need in this
format for some parts

#Plot the cleaned data
ggplot(DLV_cl, aes(time, DL_Vol)) + geom_line(color = "black") +
geom_line(color="#006666",size=1) + geom_point(shape=1, size=2) + xlab("Date") +
ylab("Download Volume (TB)") + theme(text = element_text(size=20))

#Plot the Sample ACF and PACF
par(mfrow=c(1,2))
acf(DLV_cl$DL_Vol,main="")
pacf(DLV_cl$DL_Vol,main="")

#Plot different components
#Convert DL_Vol to time series
data_DLV <- xts(DLV_cl$DL_Vol,as.Date(data$time))
#Set the frequency to 7 days (weekly)
attr(data_DLV, 'frequency') <- 7
```

```
DLV_decom <- stl(as.ts(data_DLV), s.window="periodic", t.window = 7)
plot(DLV_decom)

#Display time series differenced with ACF and PACF
tsdisplay(diff(DLV_cl.ts,7),main="Data Series differenced at lag 7")   # Same as
diff=1,lag=7
tsdisplay(diff(diff(DLV_cl.ts,7),1),main="Data Series differenced at lag 7 then lag
1")   # Same as diff=1,lag=7

#Create stationary Data set
DL_Vol_cldiff <- diff(diff(DLV_cl.ts,7),1)

#Tests for Stationarity
adftest <- adf.test(DL_Vol_cldiff, alternative = "stationary")
kpsstest <- kpss.test(DL_Vol_cldiff)

adftest$p.value
kpsstest$p.value

#Create Subset, omiting the last 70 values (10 weeks)
DLV_cl_sbset.ts <- window(DLV_cl.ts,start=c(1,1),end=c(1,296))

#ARMA Estimation using derived coefficients
#AR=2 MA=2
fit_DLV212 <- Arima(DLV_cl_sbset.ts, order=c(2,1,2), seasonal=c(7))
summary(fit_DLV212)
Box.test(residuals(fit_DLV212), lag=24, fitdf=4, type="Ljung")

#AR=2 MA=3
fit_DLV213 <- Arima(DLV_cl_sbset.ts, order=c(2,1,3), seasonal=c(7))
summary(fit_DLV213)
Box.test(residuals(fit_DLV213), lag=24, fitdf=4, type="Ljung")

#AR=2 MA=4
fit_DLV214 <- Arima(DLV_cl_sbset.ts, order=c(2,1,4), seasonal=c(7))
summary(fit_DLV214)
Box.test(residuals(fit_DLV214), lag=24, fitdf=4, type="Ljung")

#AR=2 MA=5
fit_DLV215 <- Arima(DLV_cl_sbset.ts, order=c(2,1,5), seasonal=c(7))
summary(fit_DLV215)
Box.test(residuals(fit_DLV215), lag=24, fitdf=4, type="Ljung")

#AR=2 MA=6
fit_DLV216 <- Arima(DLV_cl_sbset.ts, order=c(2,1,6), seasonal=c(7))
summary(fit_DLV216)
Box.test(residuals(fit_DLV216), lag=24, fitdf=4, type="Ljung")

#AR=2 MA=7
fit_DLV217 <- Arima(DLV_cl_sbset.ts, order=c(2,1,7), seasonal=c(7))
summary(fit_DLV217)
Box.test(residuals(fit_DLV217), lag=24, fitdf=4, type="Ljung")

#AR=3 MA=2
```

```
fit_DLV312 <- Arima(DLV_cl_sbset.ts, order=c(3,1,2), seasonal=c(7))
summary(fit_DLV312)
Box.test(residuals(fit_DLV312), lag=24, fitdf=4, type="Ljung")

#AR=3 MA=3
fit_DLV313 <- Arima(DLV_cl_sbset.ts, order=c(3,1,3), seasonal=c(7))
summary(fit_DLV313)
Box.test(residuals(fit_DLV313), lag=24, fitdf=4, type="Ljung")

#AR=5 MA=4
fit_DLV514 <- Arima(DLV_cl_sbset.ts, order=c(5,1,4), seasonal=c(7))
summary(fit_DLV514)
Box.test(residuals(fit_DLV514), lag=24, fitdf=4, type="Ljung")

#AR=5 MA=6
fit_DLV516 <- Arima(DLV_cl_sbset.ts, order=c(5,1,6), seasonal=c(7))
summary(fit_DLV516)
Box.test(residuals(fit_DLV516), lag=24, fitdf=4, type="Ljung")

#AR=6 MA=2
fit_DLV612 <- Arima(DLV_cl_sbset.ts, order=c(6,1,2), seasonal=c(7))
summary(fit_DLV612)
Box.test(residuals(fit_DLV612), lag=24, fitdf=4, type="Ljung")

#AR=6 MA=6
fit_DLV616 <- Arima(DLV_cl_sbset.ts, order=c(6,1,6), seasonal=c(7))
summary(fit_DLV616)
Box.test(residuals(fit_DLV616), lag=24, fitdf=4, type="Ljung")

#AR=7 MA=2
fit_DLV712 <- Arima(DLV_cl_sbset.ts, order=c(7,1,2), seasonal=c(7))
summary(fit_DLV712)
Box.test(residuals(fit_DLV712), lag=24, fitdf=4, type="Ljung")

#Check results
fit_DLV212$aicc
fit_DLV213$aicc
fit_DLV214$aicc
fit_DLV215$aicc
fit_DLV216$aicc
fit_DLV217$aicc
fit_DLV312$aicc
fit_DLV313$aicc
fit_DLV514$aicc
fit_DLV516$aicc
fit_DLV612$aicc
fit_DLV616$aicc
fit_DLV712$aicc

Box.test(residuals(fit_DLV212), lag=14, fitdf=4, type="Ljung")$p.value
Box.test(residuals(fit_DLV213), lag=14, fitdf=5, type="Ljung")$p.value
Box.test(residuals(fit_DLV214), lag=14, fitdf=6, type="Ljung")$p.value
Box.test(residuals(fit_DLV215), lag=14, fitdf=7, type="Ljung")$p.value
Box.test(residuals(fit_DLV216), lag=14, fitdf=8, type="Ljung")$p.value
```

```
Box.test(residuals(fit_DLV217), lag=14, fitdf=9, type="Ljung")$p.value
Box.test(residuals(fit_DLV312), lag=14, fitdf=5, type="Ljung")$p.value
Box.test(residuals(fit_DLV313), lag=14, fitdf=6, type="Ljung")$p.value
Box.test(residuals(fit_DLV514), lag=14, fitdf=9, type="Ljung")$p.value
Box.test(residuals(fit_DLV516), lag=14, fitdf=11, type="Ljung")$p.value
Box.test(residuals(fit_DLV612), lag=14, fitdf=8, type="Ljung")$p.value
Box.test(residuals(fit_DLV616), lag=14, fitdf=12, type="Ljung")$p.value
Box.test(residuals(fit_DLV712), lag=14, fitdf=9, type="Ljung")$p.value

#Best AICs result from AR=5 MA=6
fit_DLV516 <- Arima(DLV_cl_sbset.ts, order=c(5,1,6), seasonal=c(7))
summary(fit_DLV516)

par(mfrow=c(1,2))
hist(residuals(fit_DLV516),main="Histogram",nclass="FD")
cpgram(residuals(fit_DLV516),main="Cumulative Periodogram")
par(mfrow=c(1,1))

#Export Residuals for ITSM analysis
write.table(residuals(fit_DLV516),file="F:/RMIT_MasterofAnalytics/Forecasting/Dataset
s/AR516_residuals.csv",row.names=FALSE)

#Forecasting
#Forecast using ARIMA Model 5,1,6 with Seasonality D=7
fcast516 <- forecast(fit_DLV516, h=70)
fcast516$main = fcast516$main + 1.375
plot(fcast516,main="Forecast from ARIMA")
lines(DLV_cl$DL_Vol)
acc_arima516 <- accuracy(fcast516,DLV_cl$DL_Vol[297:366])
fcast516$mean

#Naive Forecast Method
fcast_naive <- naive(DLV_cl_sbset.ts, h=70)
plot(fcast_naive,plot.conf=FALSE)
lines(DLV_cl$data)
acc_naive <- accuracy(fcast_naive,DLV_cl$DL_Vol[297:366])
fitted(fcast_naive)
fcast_naive$mean

#Naive with Drift
fcast_naive_drift <- rwf(DLV_cl_sbset.ts, h=70, drift=TRUE)
plot(fcast_naive_drift)
lines(DLV_cl$DL_Vol)
acc_naive_drift <- accuracy(fcast_naive_drift,DLV_cl$DL_Vol[297:366])

#Simple Exponential Smoothing Forecast Method
fcast_ses <- ses(DLV_cl_sbset.ts, h=70)
plot(fcast_ses,plot.conf=FALSE)
lines(DLV_cl$DL_Vol)
acc_ses <- accuracy(fcast_ses,DLV_cl$DL_Vol[297:366])
fcast_ses$model

#Holt's Forecast Method
fcast_holt <- holt(DLV_cl_sbset.ts, exponential=FALSE, h=70)
```

```
plot(fcast_holt, plot.conf=FALSE)
lines(DLV_cl$DL_Vol)
acc_holt <- accuracy(fcast_holt,DLV_cl$DL_Vol[297:366])
fcast_holt$model

#Holt-Winters Forecast Method
fcast_hw_add <- hw(DLV_cl_sbset.ts, alpha=0.99, beta=0.01, gamma=0.9,
seasonal="additive", initial="simple", h=70)
plot(fcast_hw_add, plot.conf=FALSE)
lines(DLV_cl$DL_Vol)
acc_hw_add <- accuracy(fcast_hw_add,DLV_cl$DL_Vol[297:366])
fcast_hw_add$model

#ARAR From ITSM
# Read Dataset from csv file
fcast_arar <-
read.csv(file="F:/RMIT_MasterofAnalytics/Forecasting/Report/ARAR_forecast.csv",header
=TRUE,sep=',')
colnames(fcast_arar) = c("fcast","lower","upper")
fcast_arar.ts <- xts(fcast_arar$fcast, as.Date(data$time[298:366]))
acc_arar <- accuracy(as.ts(fcast_arar.ts),DLV_cl$DL_Vol[297:366])

#Comparison
acc_arima516
acc_naive
acc_naive_drift
acc_ses
acc_holt
acc_hw_add
acc_arar

#Plot Naive Forecasts
par(mfrow=c(1,1))
plot(fcast_naive,plot.conf=FALSE,ylim=c(300,700),main="Forecasts from Naive Method")
lines(DLV_cl$DL_Vol)
lines(fcast_naive_drift$mean,col="red",lwd=2)
legend("topleft", lty=1, col=c("blue","red"),legend=c("Naive method","Drift method"))

#Plot exp smoothing, Holt and Holt-Winters
plot(fcast_ses,plot.conf=FALSE,ylim=c(300,700),main="Forecasts from Exp. Smoothing
Methods")
lines(DLV_cl$DL_Vol)
lines(fcast_holt$mean, col="red",lwd=2)
lines(fcast_hw_add$mean,col="green",lwd=2)
legend("topleft", lty=2, col=c("blue","red","green"),legend=c("Exp.Sm","Holt","Holt-
Winters"))

#Plot Parzan's ARAR from ITSM
plot(DLV_cl$DL_Vol,type="l",ylim=c(300,800),ylab="",xlab="")
lines(c(298:366),fcast_arar$fcast,col="blue",lwd=2)
lines(c(298:366),fcast_arar$lower,col="grey",lwd=2)
lines(c(298:366),fcast_arar$upper,col="grey",lwd=2)
legend("topleft", lty=1, lwd=2, col=c("blue","grey"),legend=c("ARAR Model","80% CI"))
```

```
#Final Forecast using Drift Method
fcast_naive_drift_final <- rwf(DLV_cl.ts, h=70, drift=TRUE)
par(mfrow=c(1,1))
plot(finaldata$time,fcast_naive_drift_final,main="Final Forecasts from Drift Method")
fcast_naive_drift_final$mean[70]
fcast_naive_drift_final$upper[70,2]
fcast_naive_drift_final$lower[70,2]
DLV_cl$DL_Vol[366]
length(finaldata$time)
length(fcast_naive_drift_final)

#Plot with time x-axis
plot(xaxt = "n",ylab="Downlink Volume
(TB)",xlab="Date",fcast_naive_drift_final,main="Final Forecasts from Drift Method")
axis(1, at=c(26,87,149,210,271,331,392), labels=c("May-15","Jul-15","Sep-15","Nov-
16","Jan-16","Mar-16","May-16"))
legend("topleft", lwd=2,lty=1,
col=c("black","blue","lavenderblush4","grey"),legend=c("Actual Data","Forecast
Data","80% CI","95% CI"))
```