MARCH 31, 2024

TIME SERIES ANALYSIS

ON STOCK PRICES

DEEPTHI SURESH
S3991481

Table of Contents

Table of Figures

1. Introduction

Time Series Analysis is used in the stock markets to identify any patterns, so that an appropriate prediction can be made for future investment decisions. The dataset on which the Time Series analysis is conducted in this study, represents the return (in AUD 100) of a share market trader's investment portfolio. The dataset comprises of 179 observations out of a possible 252 trading days in a year. The aim of the analysis is to find a time series model that fits the dataset and create a prediction for the next 5 trading days.

2. Method

To identify the best fitting time series model, a series of descriptive analysis is conducted. A visual inspection of the TS plot helps to identify the Trend, Seasonality, Changing Variance, Behavior and Change points. A set of possible models are selected by using the model building strategy. A thorough diagnostic checking is conducted using Residual Analysis to select the best fitting model among quadratic, cosine, cyclical, seasonal or a combination trend model. The best fitting model is then used to develop a prediction for the next five trading days. R programming software is used for this study. Important snippets of the code is provided in the results section and the entire code used for this project can be found in the Appendix Section.

3. Results

3.1 Descriptive Analysis

The Summary of the raw data shows that there are 179 observations with a minimum stock value of -49.167 AUD and a maximum stock value of 214.611 AUD in a day. The average of the stock across 179 days is 57.043 AUD.

```
summary(stock)
```

```
     DayNo          returns
 Min.   :  1.0   Min.   :-49.167
 1st Qu.: 45.5   1st Qu.: -2.685
 Median : 90.0   Median : 51.105
 Mean   : 90.0   Mean   : 57.043
 3rd Qu.:134.5   3rd Qu.:117.781
 Max.   :179.0   Max.   :214.611
```

Figure 1: The Summary statistics of the raw data 'Stock'

The dataset is then converted into a TS Object called Stock_TS with a frequency of 1 considering it is a continuous data with consecutive trading days (weekends are disregarded) and each observation represents one trading day . Capturing of daily fluctuations and patterns are important in a stock market analysis, the frequency of 1 helps with that. Figure 2 shows the summary statistics of Stock_TS which is same as the dataset.

```
#Converting to a TS object
Stock_TS <- ts(as.vector(stock$returns),start = c(1,1),end =c(1,179), frequency = 1)
class(Stock_TS)
```

```
[1] "ts"
```

```
summary(Stock_TS)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-49.167  -2.685  51.105  57.043 117.781 214.611
```

Figure 2: Summary Statistics of the TS object called Stock_TS

In Figure 3, we observe a noticeable trend in the stock prices over the days from the time series plot with a U-shaped curved indicating a possible quadratic trend. Other characteristics of the plot are – some changing variance especially at the end of the data points (around 150), slight repetitive pattern indicating a possible seasonality, no change points are noted. Since there is a possibility of seasonality in the data, we cannot clearly determine the behavior.

```
#creating a TS plot
par(mfrow=c(1,2))
plot(Stock_TS,type='o',ylab='Return on portfolio(in AUD100)',
     main = " Time series plot of Return on investment portfolio")
# Adding weekdays label to check seasonality
stock <- rbind(stock, list('180', "0")) # creating an additional row to add label as 179 is not divisible by 5
stock<- stock %>% mutate(days = rep(c("M","T","W","Th","F"),times =36))
stock<- stock[-c(180),]# deleting the new row created earlier
plot(Stock_TS,type='l',ylab='Return on portfolio(in AUD100)',
     main = " Time series plot of Return on investment portfolio with labels")
```

```
points(y = Stock_TS, x = time(Stock_TS),
       pch = stock$days) #checking seasonality based on days label
```

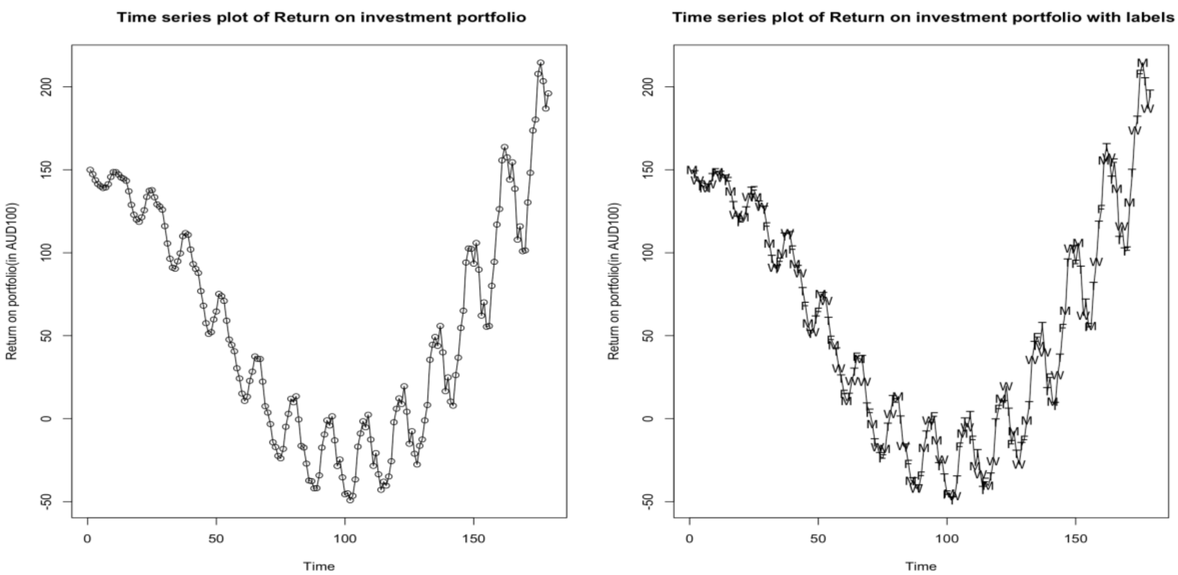

Figure 3: A side-by side of TS plot of the Returns on the investment portfolio and the same plot with day labels

For the analysis, it is important to check whether or not consecutive days are related in some way. Figure 4, shows the correlation between the previous day's return and the next day's returns. A correlation of 0.98 indicates a strong positive correlation. The same can be seen in the scatter plot on the left in Figure 6.

```
#Lag - checking the impact of previous day's returns on the next day's return
y = Stock_TS
x = zlag(Stock_TS) # generate the first lag of the returns time series
```

```
index = 2:length(x) # Create an index to get rid of the first NA value in x
cor(y[index],x[index])
```

```
[1] 0.9868369
```

Figure 4: Correlation of Lag 1

Figure 5 shows the correlation of lag 2. The correlation of 0.96 is lower than lag 1 correlation however as indicated by the scatter plot in Figure 6, it is still a strong positive correlation, suggesting a tendency for the series to follow its past values.

```
# looking at the second lag
x = zlag(zlag(Stock_TS))
index = 3:length(x)
cor(y[index],x[index])
```

```
[1] 0.963663
```

Figure 5: Correlation of lag 2

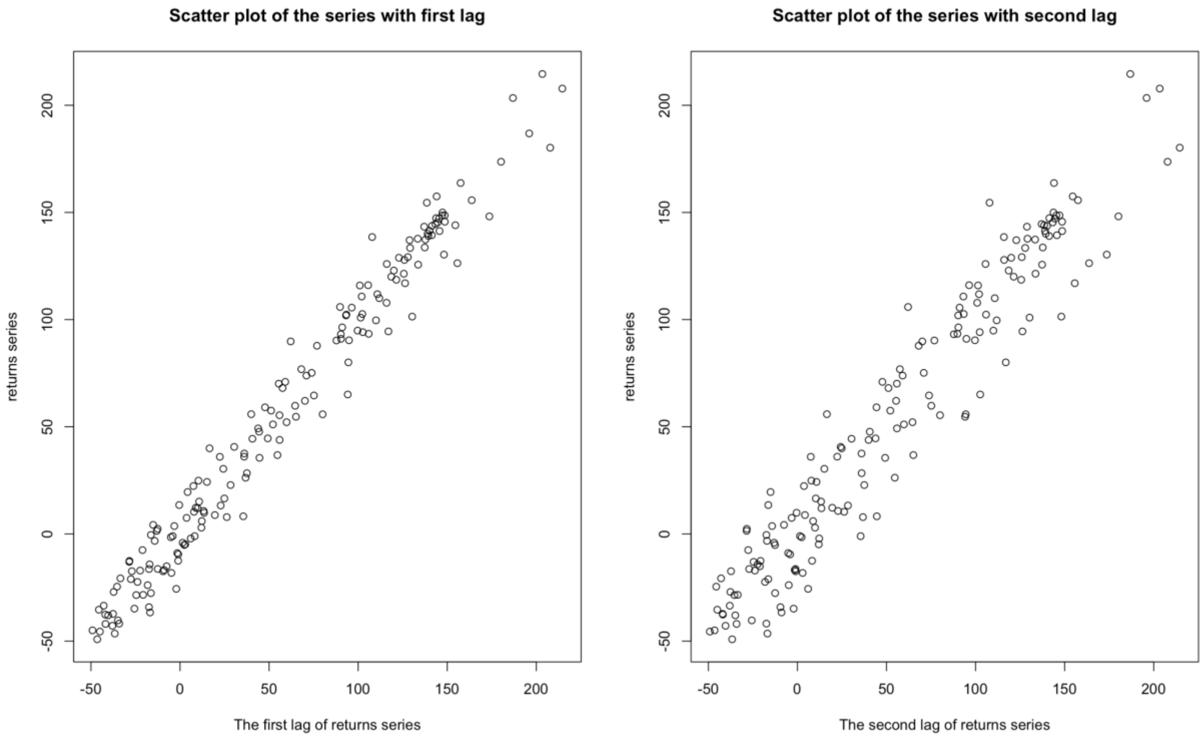

Figure 6: Side-by-side comparison of the Scatter plot with Lag 1 and Lag 2

```
#Displying the ACF
acf(Stock_TS, lag.max = 60,main = "ACF of solar return series")
```

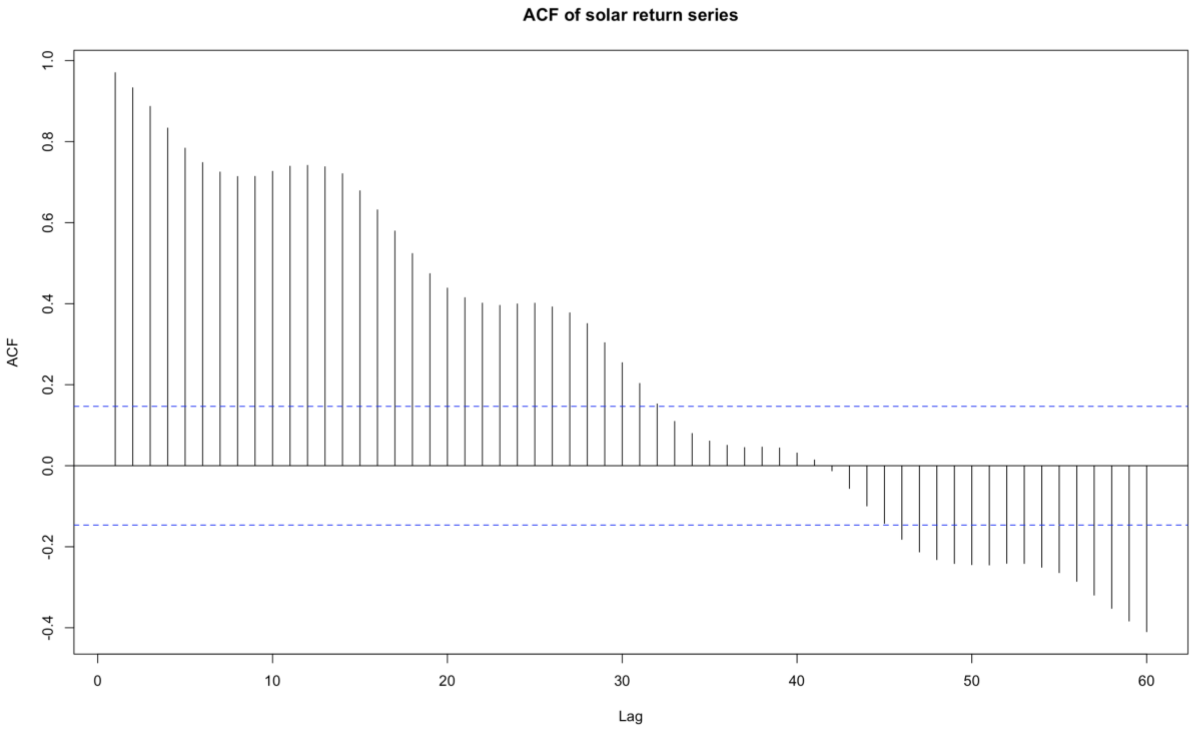

Figure 7: ACF plot of the return series

Figure 7 shows the ACF plot of the return series, which indicates a decaying pattern suggesting a possible trend and a possible seasonality with its wave-like pattern. There are significant autocorrelations as there are several lines above/below the interval.

3.2 Model Building Strategy

For the analysis we will be considering linear, quadratic, seasonal, and cosine trends or a combination of the above.

3.2.1 Model 1 :Linear Model

Creating the model

Figure 8 shows the summary of Model 1. The p-value of the coefficient is 0.08, which is greater than 0.05 and is therefore insignificant. Th adjusted R-square indicated that only 1% of the variation in the returns series is explained by the linear model. The model as a whole is also insignificant with a p-value of 0.08.

```
#Linear trend Model
t <- time(Stock_TS) # Get a continuous time points in TS
model1<- lm(Stock_TS ~ t) #
summary(model1)
```

```
Call:
lm(formula = Stock_TS ~ t)

Residuals:
     Min       1Q   Median       3Q      Max 
-104.186  -58.689   -5.424   57.532  172.072 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 72.22086   10.20622   7.076 3.33e-11 ***
t           -0.16865    0.09835  -1.715   0.0881 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 67.99 on 177 degrees of freedom
Multiple R-squared:  0.01634,	Adjusted R-squared:  0.01079 
F-statistic: 2.941 on 1 and 177 DF,  p-value: 0.08812
```

Figure 8 Summary of Model 1- Linear Model

Model Fitting

Figure 9 shows that the trend line does not cover most of the datapoints and is not a good fit.

```
  #Fitting the Linear Model
fitted.model1 <- fitted(model1)
plot(Stock_TS,ylab='returns series',xlab='Days',type='o',
     main = "Time series plot of Return on investment portfolio")
abline(model1) # add the fitted linear trend line
```

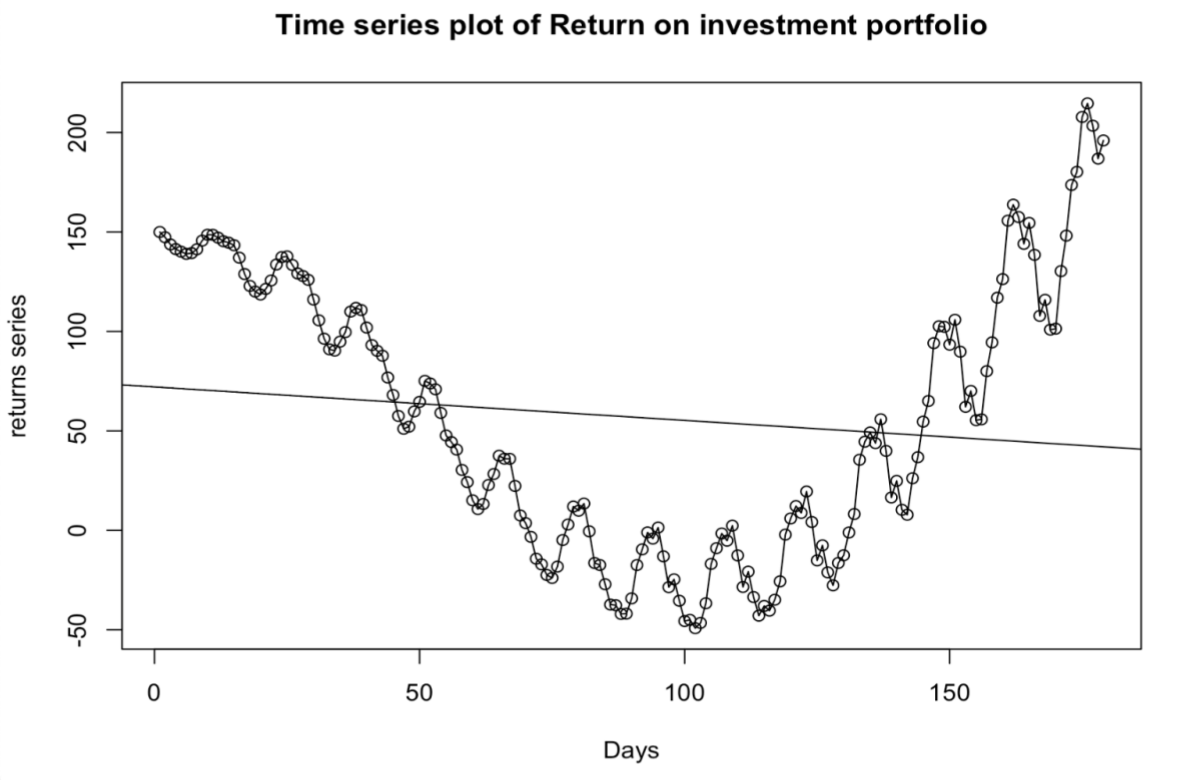

Figure 9 Fitting the linear trend model

Residual Analysis

Figure 10 shows a snapshot of the residual Analysis of the Linear Model. The Time series plot of the residuals (top left) indicates a clear trend, a possible seasonality and changing variance. The Histogram (top right) shows that there is no symmetry. The QQ plot (bottom left) has several data points away from the QQline indicating that that it is not a normal distribution. This is further confirmed in the Shapiro-Wilk normality test with a p value <0.05. The ACF plot (bottom right) shows several lines above the intervals indicating significant Autocorrelation.

Below analysis confirms the conclusions from the descriptive analysis conducted earlier showing a U-shaped curve that the Linear Model is not a good fit.

```
  #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model1),x=as.vector(time(Stock_TS)), xlab='Time',
     ylab='Standardized Residuals',type='l', main = "Standardised residuals from linear model")
#plotting a histogram
hist(rstudent(model1),xlab='Standardized Residuals', main = "Histogram of standardised residuals from linear mode
l")
```

```
# QQ plot
y = rstudent(model1)
qqnorm(y, main = "QQ plot of standardised residuals for the linear model")
qqline(y, col = 2, lwd = 1, lty = 2)
```

```
#Shapiro Wilk normality test
shapiro.test(y)
```

```
	Shapiro-Wilk normality test

data:  y
W = 0.9525, p-value = 1.022e-05
```

```
#ACF plot for the residuals
acf(rstudent(model1), main = "ACF of standardized residuals for the linear model")
```

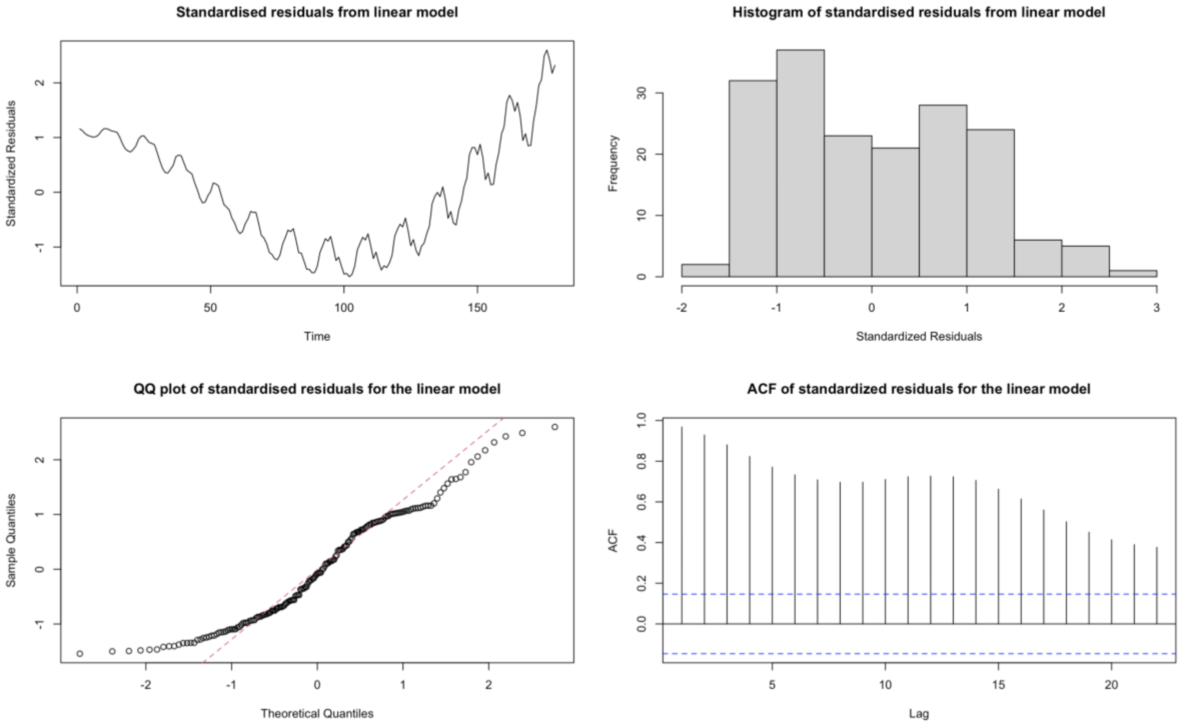

Figure 10 Residual Analysis of Linear Model

3.2.2 Model 2 :Quadratic Model

Creating the model

Figure 11 shows the summary of Model 2: Quadratic Model. The p-value of the coefficient t and t^2 i.e the linear and quadratic time effect is less than 0.05 and therefore significant. Th adjusted R2 indicated that around 85% of the variation in the returns series is explained by the Quadratic model, which is a good indication to the fit of the model. The model as a whole is also significant with a p-value less than 0.05.

```
#Quadratic Trend Model
t = time(Stock_TS)
t2 = t^2 # Create t^2
model2 = lm(Stock_TS ~ t + t2)
summary(model2)
```

```

Call:
lm(formula = Stock_TS ~ t + t2)

Residuals:
    Min      1Q  Median      3Q     Max 
-59.258 -19.321   0.571  19.992  53.165 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.141e+02  5.958e+00   35.93   <2e-16 ***
t           -4.871e+00  1.528e-01  -31.87   <2e-16 ***
t2           2.612e-02  8.224e-04   31.77   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 26.27 on 176 degrees of freedom
Multiple R-squared:  0.8539,	Adjusted R-squared:  0.8523 
F-statistic: 514.4 on 2 and 176 DF,  p-value: < 2.2e-16
```

Figure 11: Summary of Quadratic Trend Model

Model Fitting

Figure 12 shows that the trend line does not cover most of the datapoints in the beginning of the plot. It is not the best fit, but it could be a good model.

```
#Fitting Quadratic Trend Model
fitted.model2 <- fitted(model2)
plot(ts(fitted.model2), ylim = c(min(c(fitted(model2), as.vector(Stock_TS))), max(c(fitted(model2),as.vector(Stoc
k_TS)))),
     ylab='y' , main = "Fitted quadratic curve to return series", type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")
```

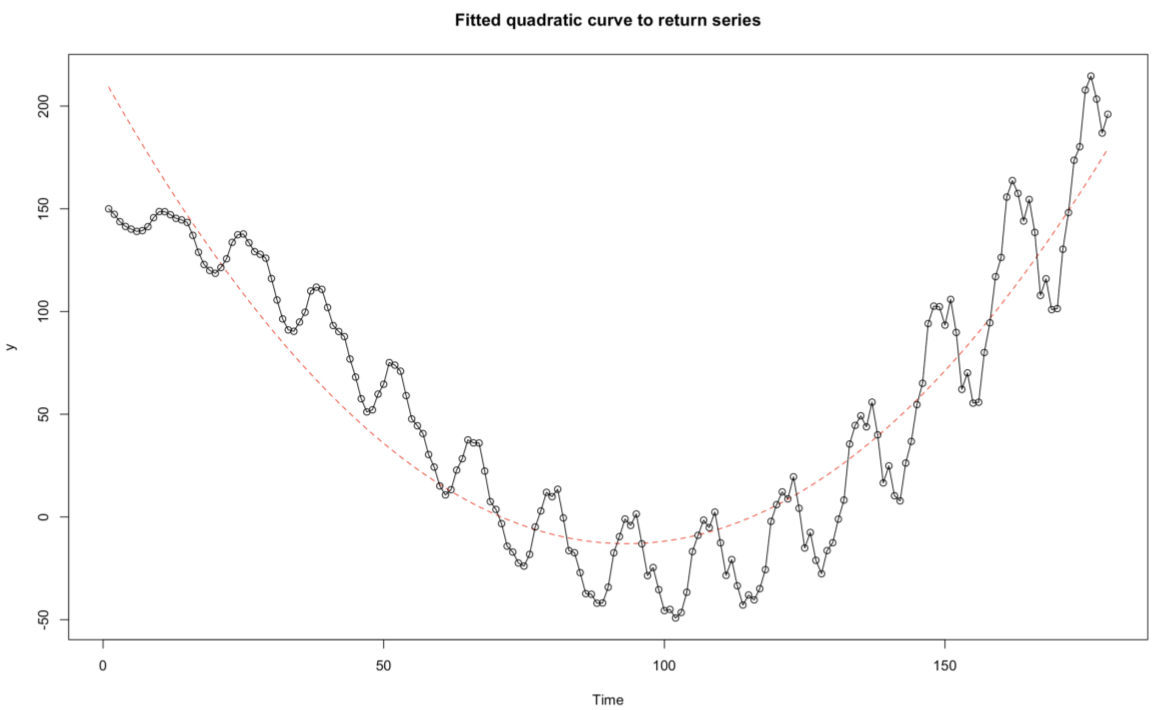

Figure 12: Fitting the Quadratic model

Residual Analysis

Figure 13 shows a snapshot of the residual Analysis of the Quadratic Model. The Time series plot of the residuals (top left) indicates a trend especially at the start of the plot (could be a cubic trend), a possible seasonality and changing variance. The Histogram (top right) shows that there is symmetry. The QQ plot (bottom left) has a few data points away from the QQline especially at the start and end of the data indicating that it could be normal distributed. This is further confirmed in the Shapiro-Wilk normality test with a p-value of 0.03 which is less than 0.05, however at an alpha of 1%, this model is normally distributed. The ACF plot (bottom right) shows several lines above/below the intervals indicating significant Autocorrelation. ACF is also showing a wave like pattern which could indicate seasonality.

Below analysis shows that a Quadratic model is a better fit than the linear model and considers the U-shaped curve that we found earlier in the descriptive analysis, but it is not a perfect fit.

```
#Fitting Quadratic Trend Model
par(mfrow=c(2,2))
plot(y=rstudent(model2),x=as.vector(time(Stock_TS)), xlab='Time',
     ylab='Standardized Residuals',type='l', main = "Standardised residuals from quadratic model.")
#plotting a histogram
hist(rstudent(model2),xlab='Standardized Residuals', main = "Histogram of standardised residuals for the quadrati
c model")
```

```
# plotting QQ plot
y = rstudent(model2)
qqnorm(y, main = "QQ plot of standardised residuals for the quadratic model
fitted to the return series.")
qqline(y, col = 2, lwd = 1, lty = 2)
```

```
#Shapiro Wilk normality test
y = rstudent(model2)
shapiro.test(y)
```

```

    Shapiro-Wilk normality test

data:  y
W = 0.98397, p-value = 0.03799
```

```
#ACF plot for the residuals
acf(rstudent(model2), main = "ACF of standardized residuals the quadratic model
fitted to the return series.")
par(mfrow=c(1,1))
```

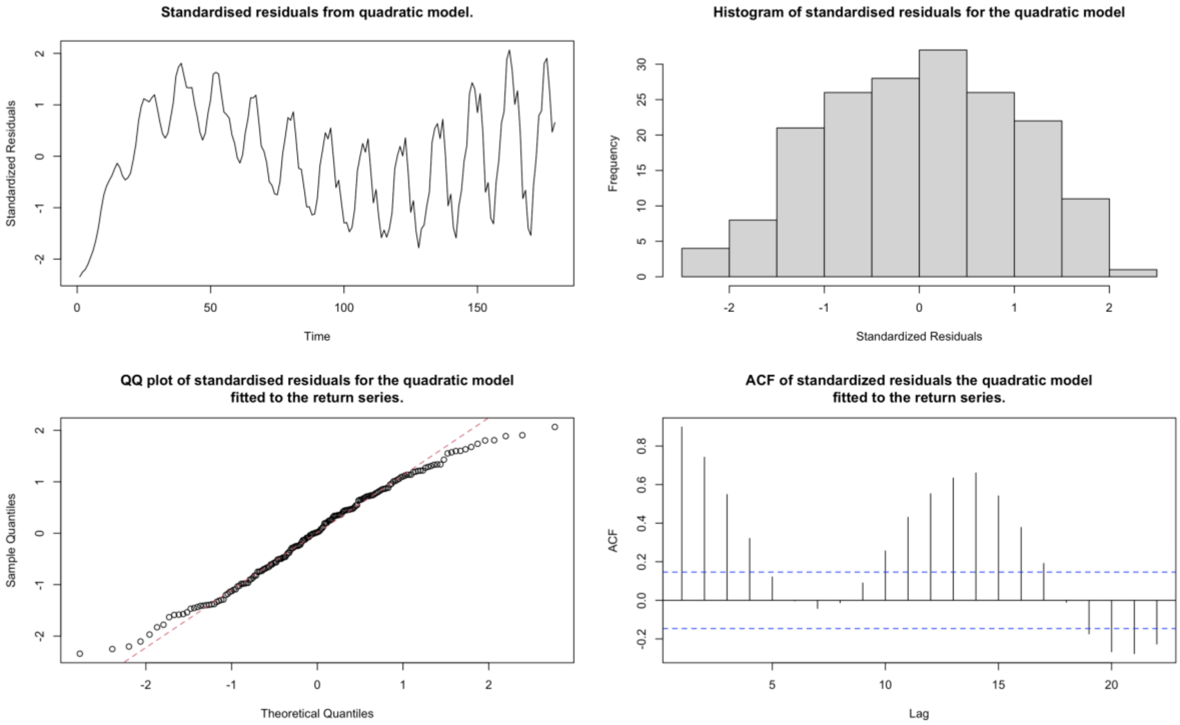

Figure 13: Residual Analysis of Quadratic Model

3.2.3 Model 3 :Seasonal Model

Creating the Model

Figure 14 shows the summary of Model 3: Seasonal Model. The p-value of the coefficient of all the weekdays are less than 0.05 and therefore significant. However, the adjusted R-squared for the model is only 0.39 indicating that only 39% of the variation is explained by the seasonal model. The model as a whole is significant with a p-value less than 0.05. Even with a TS object with increased frequency (considered frequency of 5 and 14(not included in the report)), the model did not have a good Adjusted R-squared value.

```
Weekday. <- factor(stock$days,levels = c( "M","T","W","Th","F"))
model3=lm(Stock_TS ~ Weekday.-1) # -1 removes the intercept term
summary(model3)
```

```
Call:
lm(formula = Stock_TS ~ Weekday. - 1)

Residuals:
     Min       1Q   Median       3Q      Max 
-106.785  -60.423   -5.695   62.046  156.912 

Coefficients:
           Estimate Std. Error t value Pr(>|t|)    
Weekday.M     57.70      11.52   5.008 1.34e-06 ***
Weekday.T     57.62      11.52   5.001 1.38e-06 ***
Weekday.W     57.83      11.52   5.020 1.27e-06 ***
Weekday.Th    57.78      11.52   5.015 1.30e-06 ***
Weekday.F     54.21      11.68   4.639 6.85e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 69.13 on 174 degrees of freedom
Multiple R-squared:  0.4121,	Adjusted R-squared:  0.3952 
F-statistic: 24.39 on 5 and 174 DF,  p-value: < 2.2e-16
```

Figure 14: Summary of Seasonal Model

Fitting the model

As indicated by the model summary, the plots indicated that there is no good fit of the seasonal model on the returns series.

```
  #Fitting the Model
plot(ts(fitted(model3)), ylab='returns series', main = "Fitted seasonal model to returns series.",
     ylim = c(-50,250), col = "red" )
lines(as.vector(Stock_TS),type="o")
```

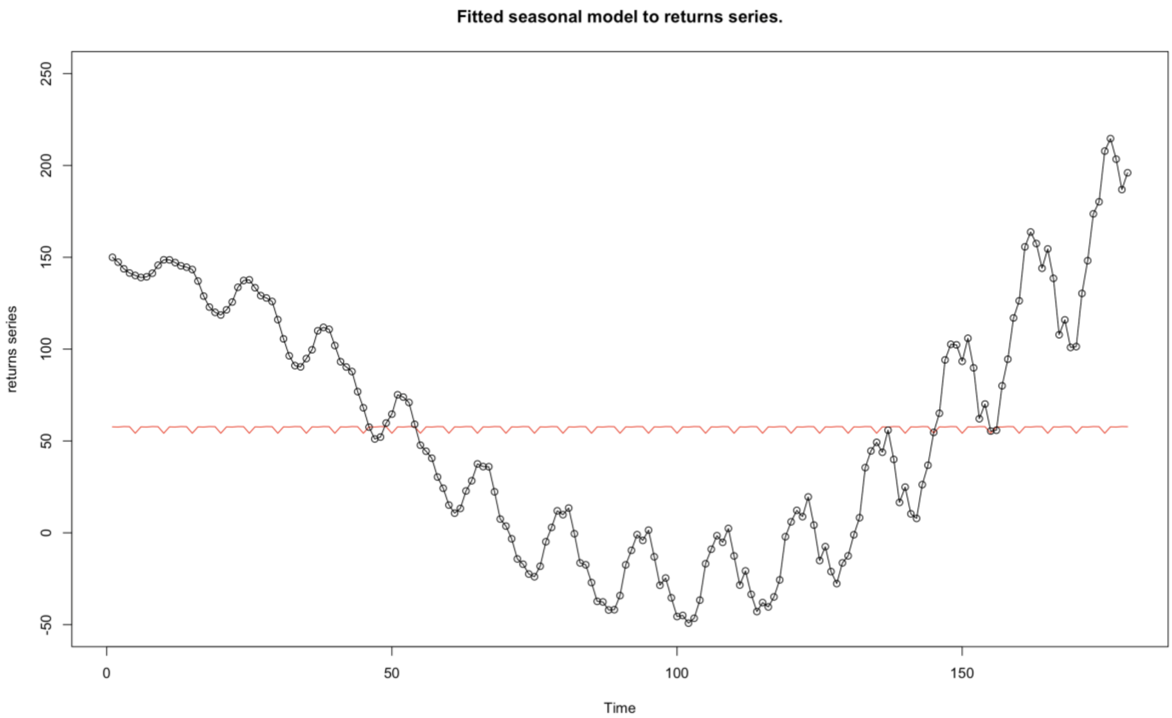

Figure 15: Fitting the Seasonal model

Residual Analysis

Figure 16 shows a snapshot of the residual Analysis of the Seasonal Model. The Time series plot of the residuals (top left) indicates a very similar pattern as the original time series indicating that the trend, a possible seasonality and changing variance seen in the original time series is not being captured by the model. The Histogram (top right) shows that there is no symmetry. The QQ plot (bottom left) has a lot of data points away from the QQline especially at the start and end of the data indicating that it is not normal distributed. The Shapiro-Wilk normality test has a p-value which is less than 0.05, confirming that the normality assumption does not hold. The ACF plot (bottom right) shows several lines above the intervals indicating significant Autocorrelation. ACF is also showing a wave like pattern which is very similar to the original time series which could indicate that the seasonality is not being captured by the model.

Below analysis shows that a Seasonal model is not a good fit, even though the descriptive analysis showed indications of having seasonality.

```
  #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model3),x=as.vector(time(Stock_TS)), xlab='Time',
     ylab='Standardized Residuals',type='l', main = "Standardised residuals from seasonal model")
points(y=rstudent(model3),x=as.vector(time(Stock_TS)),
       pch=as.vector(stock$days))
```

```
#plotting a histogram
hist(rstudent(model3),xlab='Standardized Residuals', main = "Histogram of standardised residuals for the seasonal
model")
#plotting qq plot
y = rstudent(model3)
qqnorm(y, main = "QQ plot of standardised residuals for the seasonal model
fitted to the returns series.")
```

```
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
y = rstudent(model3)
shapiro.test(y)
```

```
	Shapiro-Wilk normality test

data:  y
W = 0.94757, p-value = 3.631e-06
```

```
#ACF plot for the residuals
acf(rstudent(model3), main = "ACF of standardized residuals the seasonal model
fitted to the return series.")
```

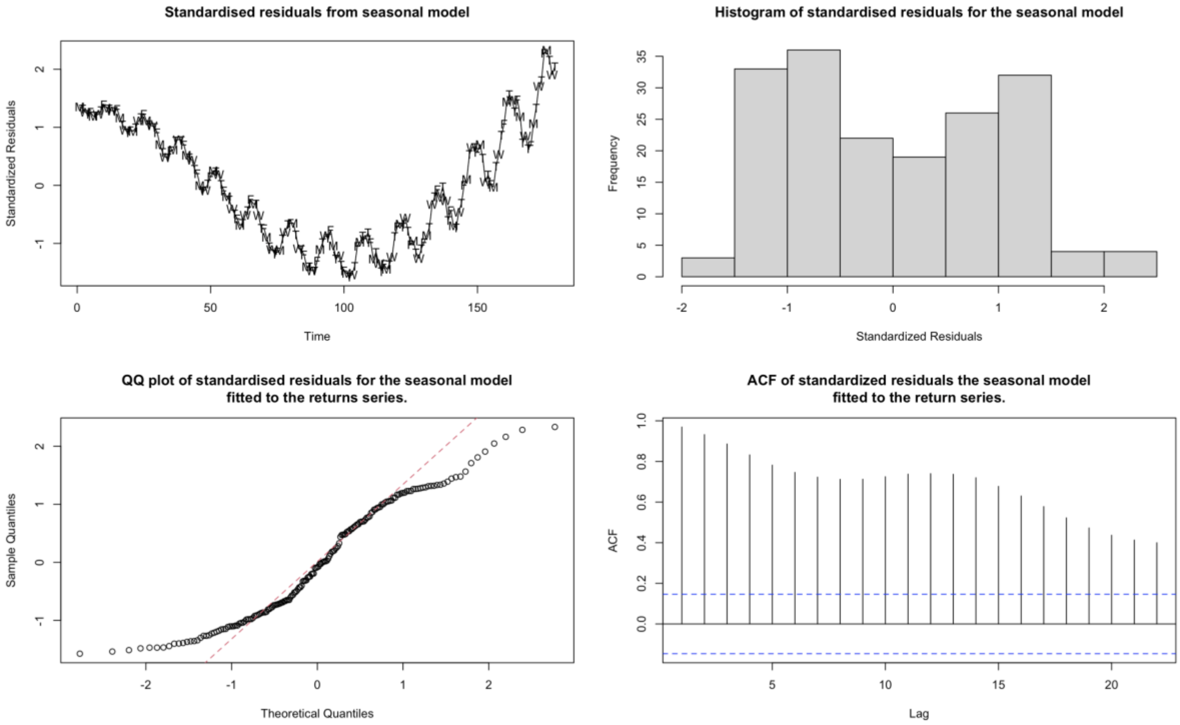

Figure 16: Residual Analysis of Seasonal Model

3.2.4 Model 4 :Cosine Model

Creating the Model

Figure 17 shows the summary of Model 4: Cosine or Harmonic Model. A new TS object with frequency of 5 (for 5 trading days in a week) has been created as a cosine model can be created only on a TS object with a frequency of >1.Here the frequency is considered as 5 assuming that the pattern we see in the plot happens on a weekly basis. The p-value of both cos and sine coefficient are more than 0.05 and therefore insignificant. The adjusted R-squared for the model is -1% indicating that the model is not a good fit for the timeseries. The model as a whole is also insignificant with a p-value less than 0.98. Even with a TS object with increased frequency (considered frequency of 5 and 14(not included in the report)), model was not significant and did not have a good Adjusted R-squared value.

```
#3.2.3 Model 4 - Cosine model
#creating a new TS object as harmonic model requires a frequency of more than 1
Stock_TS_5 <- ts(as.vector(stock$returns),start = c(1,1),end =c(1,179), frequency = 5)
har. <- harmonic(Stock_TS_5, 1) # calculate cos(2*pi*t) and sin(2*pi*t)
data <- data.frame(Stock_TS_5,har.)
model4 <- lm(Stock_TS_5 ~ cos.2.pi.t. + sin.2.pi.t. , data = data)
summary(model4)
```

```
Call:
lm(formula = Stock_TS_5 ~ cos.2.pi.t. + sin.2.pi.t., data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-107.278  -59.716   -6.408   61.674  158.081

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  57.0348     5.1379  11.101   <2e-16 ***
cos.2.pi.t.  -0.5054     7.2496  -0.070    0.945
sin.2.pi.t.   1.2955     7.2826   0.178    0.859
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 68.74 on 176 degrees of freedom
Multiple R-squared:  0.0002069, Adjusted R-squared:  -0.01115
F-statistic: 0.01821 on 2 and 176 DF,  p-value: 0.982
```

Figure 17: Summary of Cosine Model

Fitting the Model

As indicated by the model summary, the plots indicated that it is not a good fit. The fitted line doesn't consider majority of the data point.

```
 #Fitting the model
par(mfrow=c(1,1))
plot(ts(fitted(model4)), ylab='returns series', main = "Fitted cosine wave to returns series.",
     ylim = c(-50,250), col = "green" )
lines(as.vector(Stock_TS_5),type="o")
```

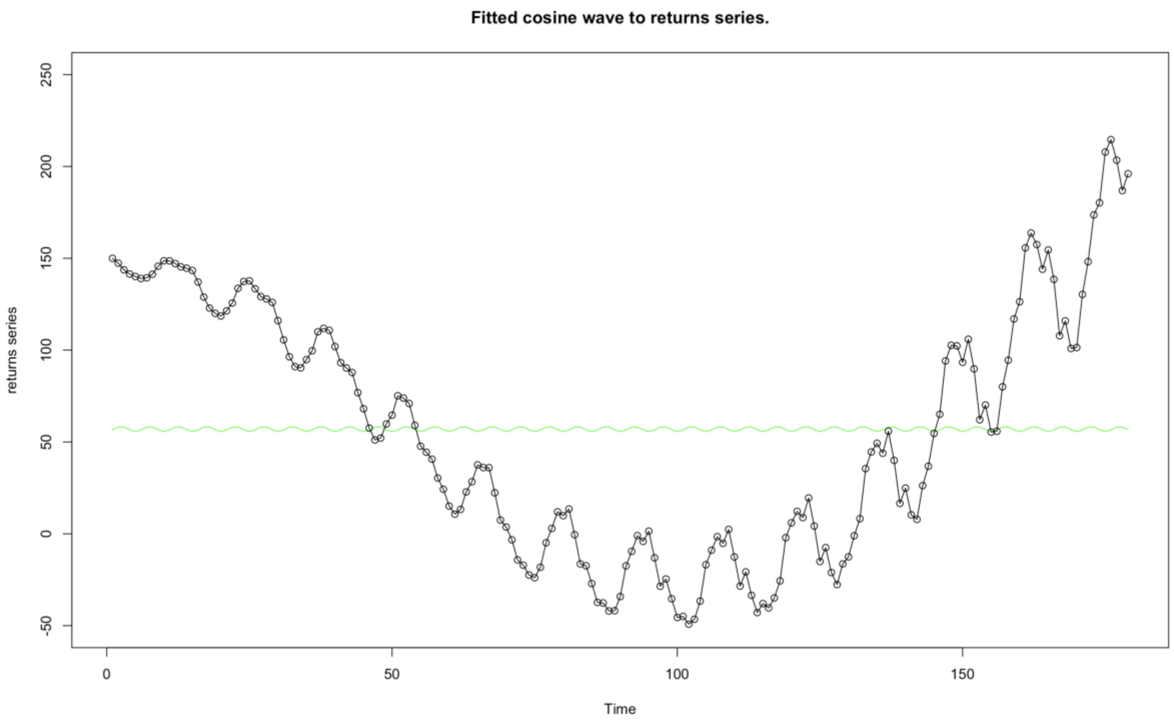

Figure 18: Fitting the Cosine Model

Residual Analysis
Figure 19 shows a snapshot of the residual Analysis of the Cosine Model. The Time series plot of the residuals (top left) indicates that the residuals have trend, a possible seasonality and changing variance like the one seen in the original time series. The Histogram (top right) shows a longer right tail indicating that there is no symmetry. The QQ plot (bottom left) has a lot of data points away from the QQline especially at the start and end of the data indicating that it is not normal distributed. The Shapiro-Wilk normality test has a p-value which is less than 0.05, confirming that it is not a normal distribution. The ACF plot (bottom right) shows several lines above the confidence intervals indicating significant Autocorrelation.

Below analysis shows that a Cosine model is not a good fit.

```
 #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model4),x=as.vector(time(Stock_TS_5)), xlab='Time',
     ylab='Standardized Residuals',type='l', main = "Time series plot of standardised residuals for the cosine wa
ve")
#Plotting histogram
hist(rstudent(model4),xlab='Standardized Residuals', main = "Histogram of standardised residuals for the cosine w
ave")
```

```
#QQ plot
y = rstudent(model4)
qqnorm(y, main = "QQ plot of standardised residuals for the cosine wave")
qqline(y, col = 2, lwd = 1, lty = 2)
```

```
#Shapiro Wilk normality test
y = rstudent(model4)
shapiro.test(y)
```

```
	Shapiro-Wilk normality test

data:  y
W = 0.94756, p-value = 3.628e-06
```

```
#ACF plot for the residuals
acf(rstudent(model4), main = "ACF of standardized residuals for the cosine model")
par(mfrow=c(1,1))
```

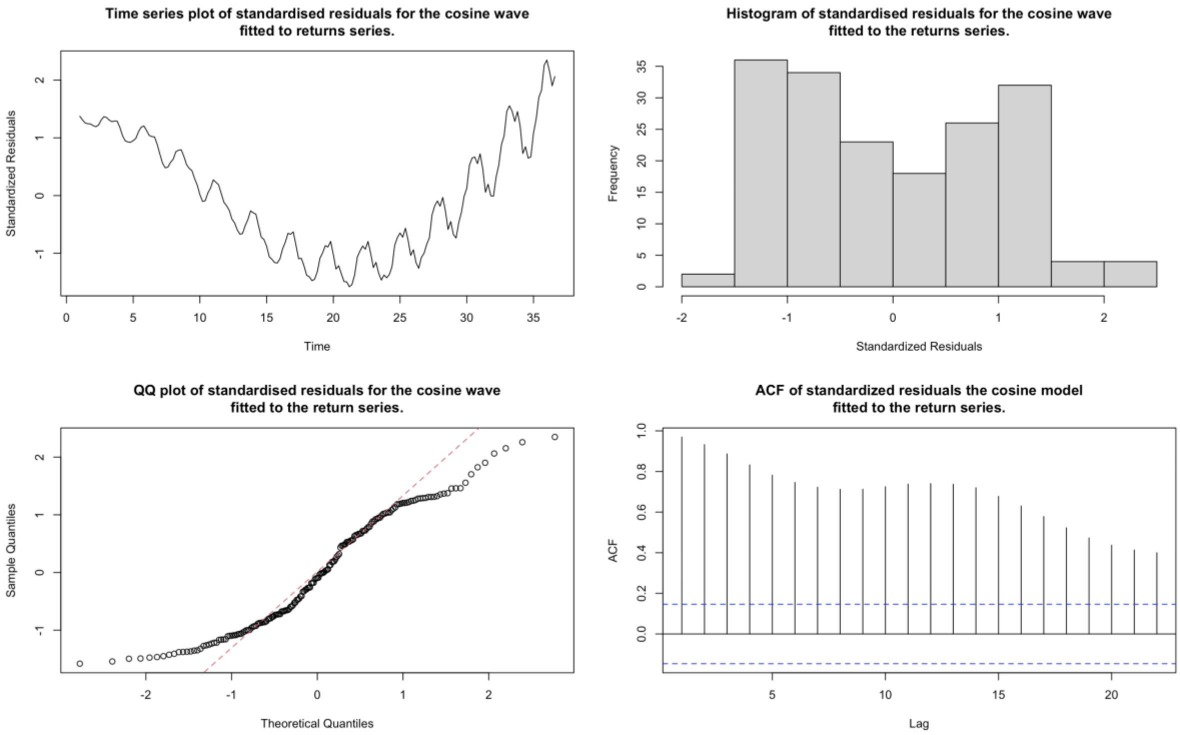

Figure 19: Residual Analysis of Cosine Model

3.2.4 Model 5 :Cubic Model

Creating the Model

Figure 20 shows the summary of Model 5: Cosine or Harmonic Model. The p-value of both quadratic and cubic coefficient are less than 0.05 and therefore significant. The adjusted R-squared for the model is 90% indicating that the model can be a good fit for the timeseries. The model is also significant with a p-value less than 0.05.

```
# #3.2.5 Model 5 Cubic model
t = time(Stock_TS)
t2 = t^2 # Create t^2
t3 = t^3
model5 = lm(Stock_TS ~ t + t2+t3)
summary(model5)
```

```
Call:
lm(formula = Stock_TS ~ t + t2 + t3)

Residuals:
    Min      1Q  Median      3Q     Max 
-58.720 -16.044  -0.263  15.724  49.764 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.696e+02  6.288e+00  26.973  < 2e-16 ***
t           -1.947e+00  3.017e-01  -6.455 1.03e-09 ***
t2          -1.437e-02  3.889e-03  -3.695 0.000294 ***
t3           1.500e-04  1.420e-05  10.558  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 20.59 on 175 degrees of freedom
Multiple R-squared:  0.9108,	Adjusted R-squared:  0.9092 
F-statistic: 595.4 on 3 and 175 DF,  p-value: < 2.2e-16
```

Figure 20:Summary of the cubic Model

Fitting the Model

Figure 21 shows that the model takes into account a lot of the data points in the start and end of the report that was not taken into account in a lot of the other models.

```
  #Fitting the model
fitted.model5 <- fitted(model5)
plot(ts(fitted.model5), ylim = c(min(c(fitted(model5), as.vector(Stock_TS))), max(c(fitted(model5),as.vector(Stoc
k_TS)))),
     ylab='y' , main = "Fitted cubic curve to return series", type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")
```

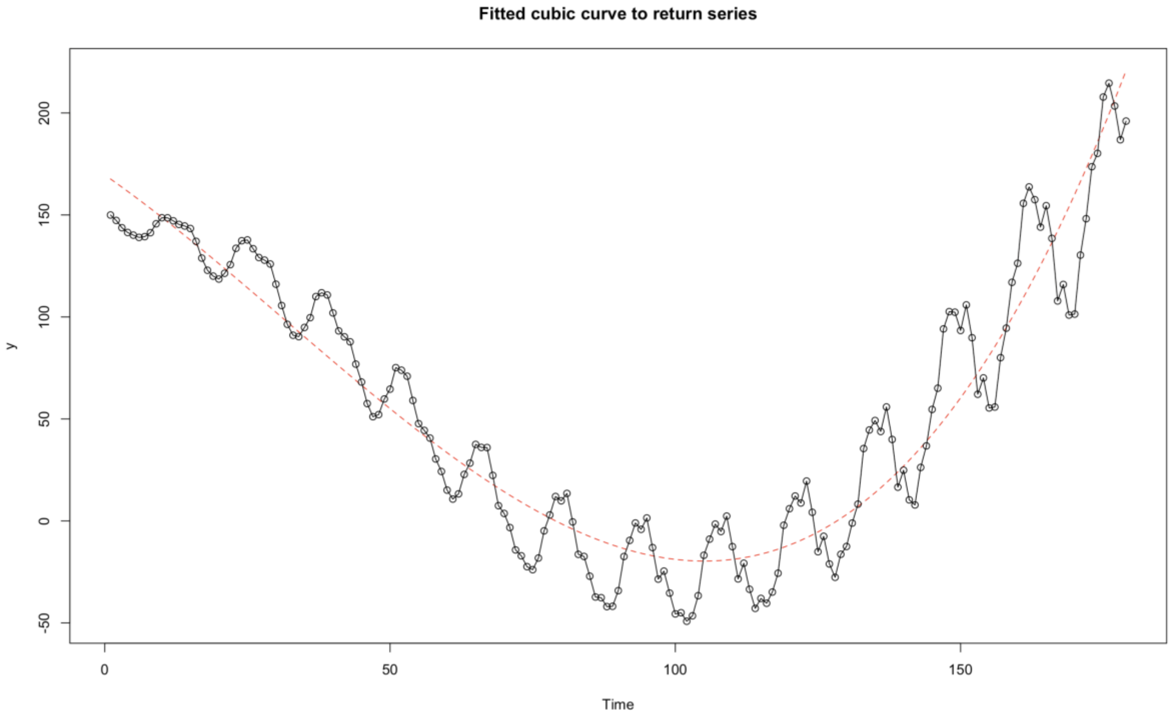

Figure 21:Fitting the Cubic Model

Residual Analysis

Figure 22 shows a snapshot of the residual Analysis of the Cubic Model. The timeseries plot (top left) of the residuals in cubic model looks very similar to the quadratic model, however here the trend is captured better especially in the trend that could be seen in the start of the curve. However, there is a possible seasonality and changing variance especially in the last quarter of the plot. The Histogram (top right) shows that there is symmetry, however there is an outlier around -3. The QQ plot (bottom left) has a few data points away from the QQline especially at the start, but most of it is aligned to the QQ line indicating that it could be normal distributed. This is further confirmed in the Shapiro-Wilk normality test with a p-value of 0.14 which is greater than 0.05 and this model is normally distributed. The ACF plot (bottom right) shows several lines above/below the intervals indicating significant Autocorrelation. ACF is also showing a wave like pattern which could indicate seasonality.

Below analysis shows that a Cubic model is a better fit than all the models tested so far indicating that the trend is being captured, but it is not a perfect fit.

```
 #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model5),x=as.vector(time(Stock_TS)), xlab='Time',
     ylab='Standardized Residuals',type='l', main = "Standardised residuals from cubic model.")
#plotting a histogram
hist(rstudent(model5),xlab='Standardized Residuals', main = "Histogram of standardised residuals for the cubic mo
del")
```

```
# plotting QQ plot
y = rstudent(model5)
qqnorm(y, main = "QQ plot of standardised residuals for the quadratic model
fitted to the return series.")
qqline(y, col = 2, lwd = 1, lty = 2)
```

```
#Shapiro Wilk normality test
y = rstudent(model5)
shapiro.test(y)
```

```

    Shapiro-Wilk normality test

data:  y
W = 0.98835, p-value = 0.1482
```

```
#ACF plot for the residuals
acf(rstudent(model5), main = "ACF of standardized residuals the cubic model
fitted to the return series.")
```

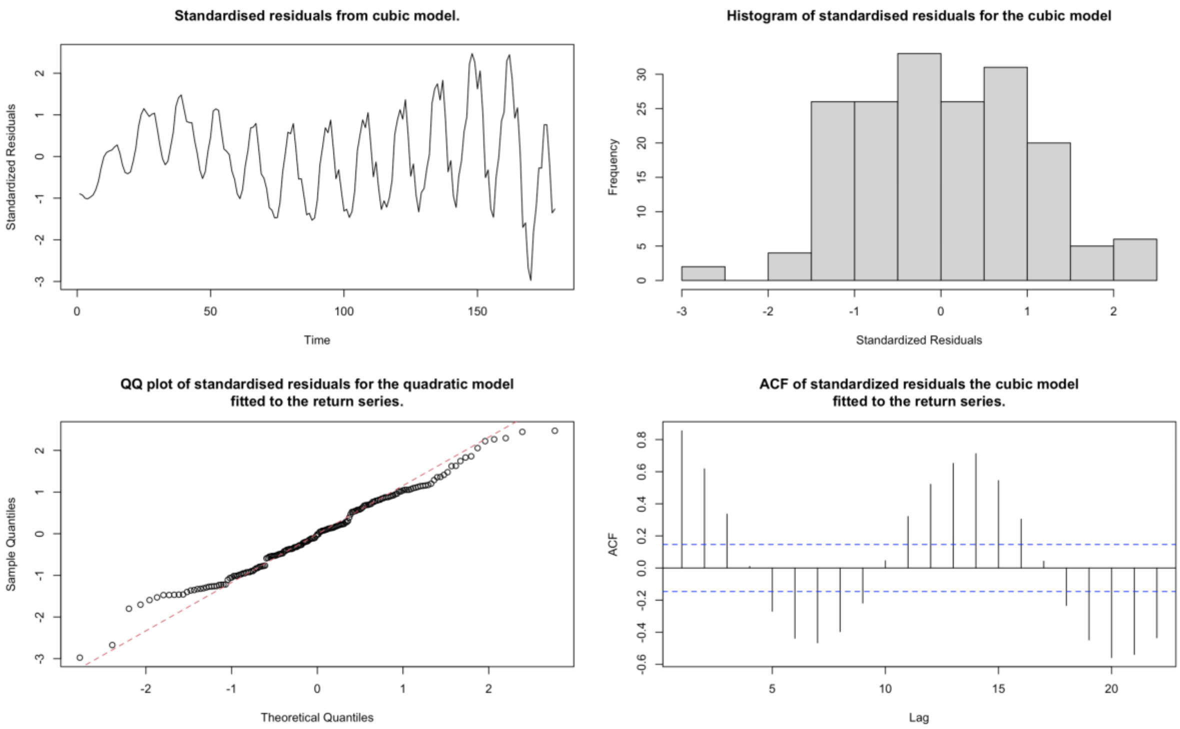

Figure 22:Residual Analysis of Cubic Model

3.2.6 Model 6 :Quartic + Seasonal Model

Creating the Model

Figure 20 shows the summary of Model 6: Quartic + Seasonal Model. The p-value of both seasonal and quartic coefficients are less than 0.05 and therefore significant. The adjusted R-squared for the model is 95% indicating that the model is a very good fit for the timeseries. The model is also significant with a p-value less than 0.05.

```
#3.2.6 Model 6 :seasonal plus quartic time trend model
t = time(Stock_TS)
t2 = t^2 # Create t^2
t3 = t^3
t4 = t^4 # takes into account the curve
model6 = lm(Stock_TS~ Weekday. + t + t2+t3+t4 -1)
summary(model6)
```

```
Call:
lm(formula = Stock_TS ~ Weekday. + t + t2 + t3 + t4 - 1)

Residuals:
    Min      1Q  Median      3Q     Max 
-53.610 -14.119   0.059  13.085  42.800 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
Weekday.M   1.405e+02  7.633e+00  18.403  < 2e-16 ***
Weekday.T   1.401e+02  7.666e+00  18.274  < 2e-16 ***
Weekday.W   1.400e+02  7.695e+00  18.190  < 2e-16 ***
Weekday.Th  1.396e+02  7.721e+00  18.077  < 2e-16 ***
Weekday.F   1.397e+02  7.796e+00  17.913  < 2e-16 ***
t           1.264e+00  5.497e-01   2.300   0.0226 *  
t2         -9.412e-02  1.237e-02  -7.609 1.80e-12 ***
t3          8.378e-04  1.031e-04   8.125 8.78e-14 ***
t4         -1.910e-06  2.842e-07  -6.722 2.60e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.57 on 170 degrees of freedom
Multiple R-squared:  0.9585,	Adjusted R-squared:  0.9564 
F-statistic: 436.8 on 9 and 170 DF,  p-value: < 2.2e-16
```

Figure 23: Summary of Quartic + Seasonal Model

Fitting the Model

Figure 24 shows that the model follows the curve of the time series plot and considers the curve in the first quarter of the plot which was a shortcoming in the models tested above. It was clear that a seasonal component was missed in the earlier models and therefore taking a combination of both Quartic and Seasonal aspects should help solve the issue.

```
  #Fitting the model
fitted.model6 <- fitted(model6)
plot(ts(fitted(model6)),ylim = c(min(c(fitted(model6), as.vector(Stock_TS))), max(c(fitted(model6),as.vector(Stoc
k_TS)))),
     ylab='y' , main = "Fitted seasonal plus quadratic curve to return series", type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")
```

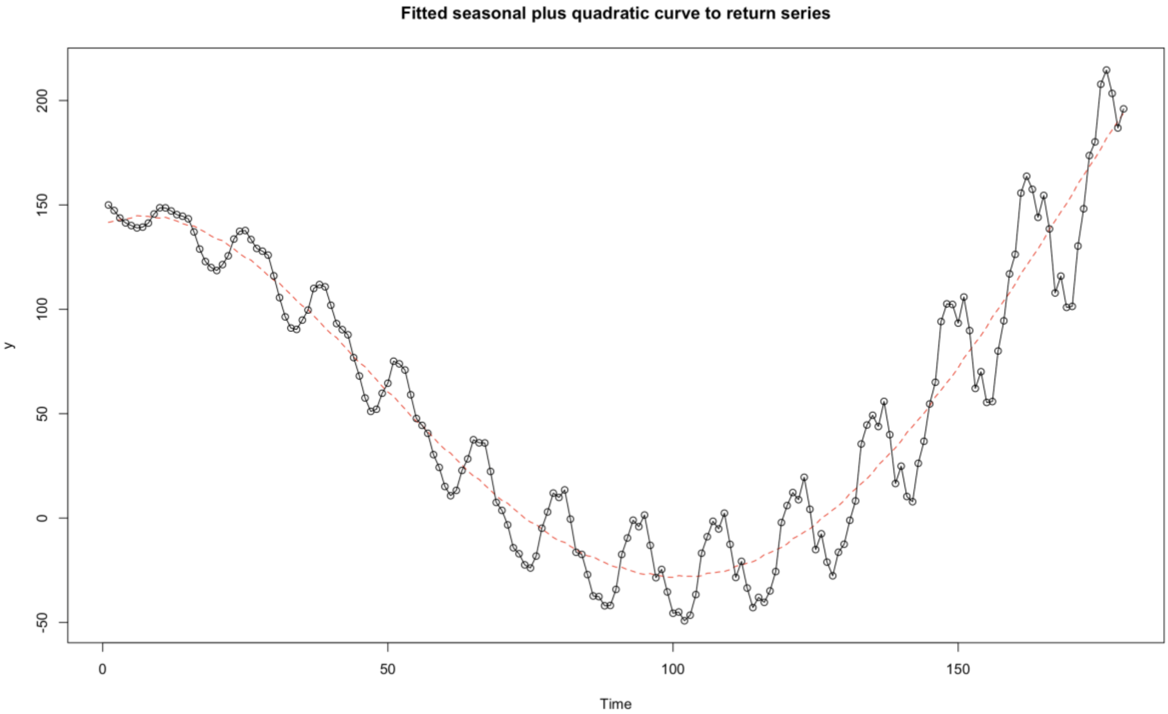

Figure 24: Fitting the Quartic + Seasonal Model

Residual Analysis

Figure 25 shows a snapshot of the residual Analysis of the Quartic and Seasonal Model. The timeseries plot (top left) of the residuals in Quartic and Seasonal Model looks like it has captured all the trend, however the there is still changing variance which is not being captured by the model. The Histogram (top right) looks like there is symmetry. The QQ plot (bottom left) has a most data points aligned with the QQline indicating that it could be normal distributed. This is further confirmed in the Shapiro-Wilk normality test with a p-value of 0.39 which is greater than 0.05 and this model is normally distributed. The ACF plot (bottom right) however shows several lines above/below the intervals indicating significant Autocorrelation. ACF is still showing a wave like pattern which could indicate that the model is unable to capture seasonality.

Below analysis shows that a Quartic and Seasonal model is a better fit than the others and could be the best that a deterministic model could provide.

```
par(mfrow=c(2,2))
plot(y=rstudent(model6),x=as.vector(time(Stock_TS)), xlab='Time',
     ylab='Standardized Residuals',type='l', main = "Standardised residuals from linear model")
#plotting a histogram
hist(rstudent(model6),xlab='Standardized Residuals', main = "Histogram of standardised residuals from linear mode
l")
```

```
# QQ plot
y = rstudent(model6)
qqnorm(y, main = "QQ plot of standardised residuals for the linear model")
qqline(y, col = 2, lwd = 1, lty = 2)
```

```
#Shapiro Wilk normality test
y = rstudent(model6)
shapiro.test(y)
```

```
	Shapiro-Wilk normality test

data:  y
W = 0.99169, p-value = 0.3933
```

```
#ACF plot for the residuals
acf(rstudent(model6), main = "ACF of standardized residuals for the linear model")
par(mfrow=c(1,1))
```

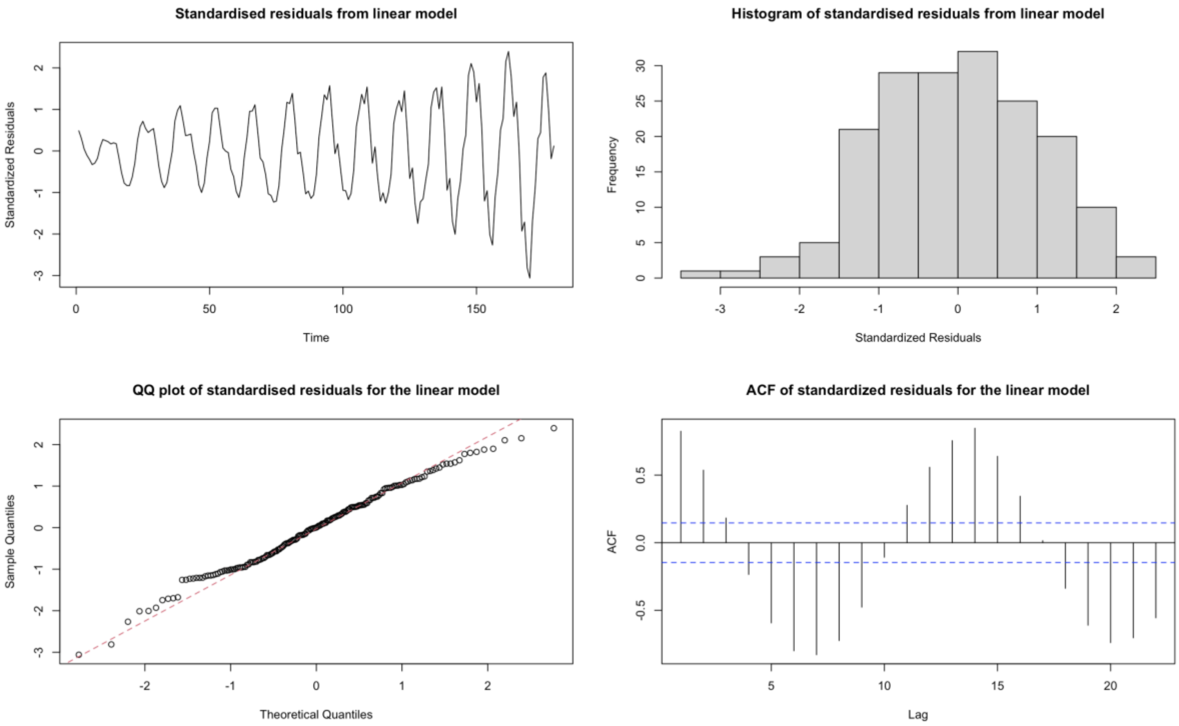

Figure 25: Residual Analysis of Quartic + Seasonal Model

3.2.7 Model 7: Quartic Model

Creating the Model

Figure 26 shows the summary of Model 7: Quartic Model. The p-value of all the time coefficients is less than 0.05 and therefore significant. The adjusted R-squared for the model is 92% indicating that the model is a very good fit for the timeseries. The model is also significant with a p-value less than 0.05. Since the seasonal component was not accounted for in the Quartic +Seasonal model, Quartic model might be a better fit to adhere to the principal of parsimony.

```r
# #3.2.7 model 7 Quartic Model
t = time(Stock_TS)
t2 = t^2 # Create t^2
t3 = t^3
t4 = t^4
model7 = lm(Stock_TS ~ t + t2+t3+t4)
summary(model7)
```

```

Call:
lm(formula = Stock_TS ~ t + t2 + t3 + t4)

Residuals:
    Min      1Q  Median      3Q     Max 
-53.904 -14.159   0.545  13.220  42.940 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.400e+02  7.097e+00  19.728  < 2e-16 ***
t            1.261e+00  5.431e-01   2.322   0.0214 *  
t2          -9.406e-02  1.222e-02  -7.696 1.00e-12 ***
t3           8.374e-04  1.019e-04   8.219 4.54e-14 ***
t4          -1.909e-06  2.808e-07  -6.800 1.61e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.36 on 174 degrees of freedom
Multiple R-squared:  0.9295,	Adjusted R-squared:  0.9279 
F-statistic: 573.5 on 4 and 174 DF,  p-value: < 2.2e-16
```

Figure 26: Summary of Quartic Model

Fitting the model

Figure 27 shows that the model follows the curve of the time series plot and similar to the quartic +seasonal model, considers the curve in the first quarter of the plot which was a shortcoming in the other models.

```r
  #Fitting the model
fitted.model7 <- fitted(model7)
plot(ts(fitted.model7), ylim = c(min(c(fitted(model7), as.vector(Stock_TS))), max(c(fitted(model7),as.vector(Stoc
k_TS)))),
     ylab='y' , main = "Fitted Quartic curve to return series", type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")
```

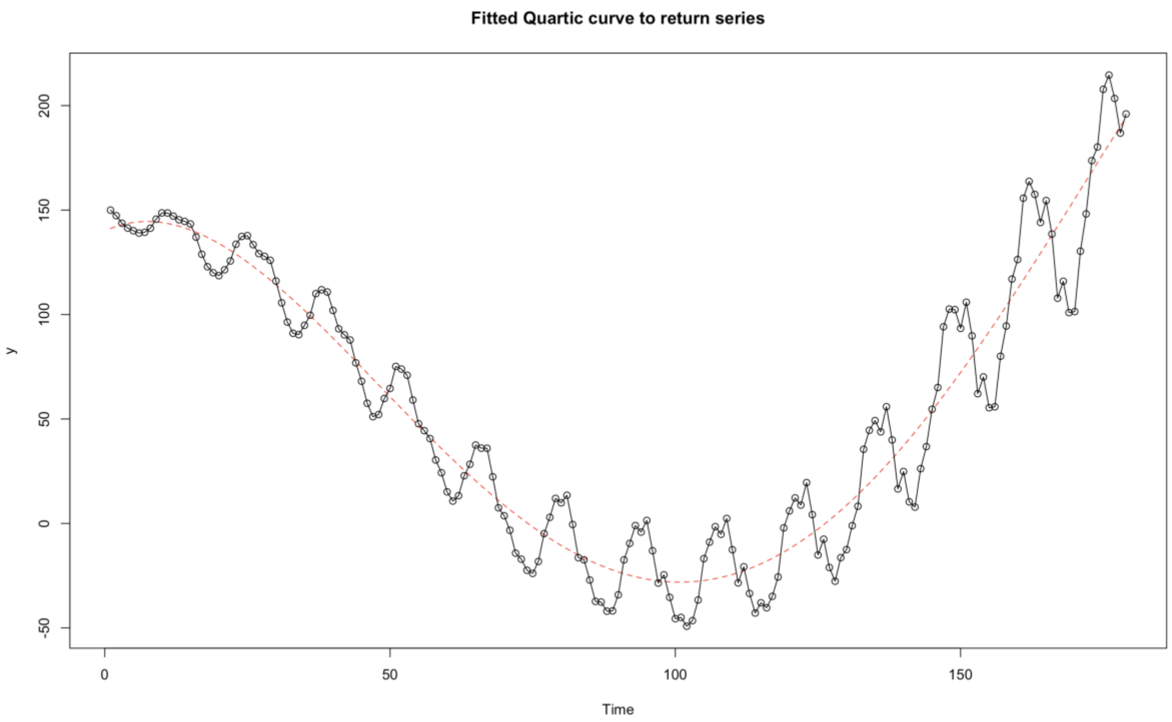

Figure 27:Fitting the Quartic Model

Residual Analysis

Figure 28 shows a snapshot of the residual Analysis of the Quartic Model. The timeseries plot (top left) of the residuals in Quartic Model looks like it has captured all the trend and there no noticeable intervention point, however the there is still changing variance which is not being captured by the model. As expected, the seasonality is also not being captures by the model. The Histogram (top right) looks almost symmetry with a possible left tail. The QQ plot (bottom left) has a most data points aligned with the QQline indicating that it could be normal distributed. This is further confirmed in the Shapiro-Wilk normality test with a p-value of 0.39 which is greater than 0.05 , therefore a normally distributed model. The ACF plot (bottom right) however shows several lines above/below the intervals indicating significant Autocorrelation. As expected from model 6, ACF is still showing a wave like pattern which could indicate that the model is unable to capture seasonality.

Below analysis shows that a Quartic model is a good deterministic model that would also adhere to the principal of minimum coefficients.

```
#Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model7),x=as.vector(time(Stock_TS)), xlab='Time',
     ylab='Standardized Residuals',type='l', main = "Standardised residuals from Quartic model.")
#plotting a histogram
hist(rstudent(model7),xlab='Standardized Residuals', main = "Histogram of standardised residuals for the Quartic
model")
```

```
# plotting QQ plot
y = rstudent(model7)
qqnorm(y, main = "QQ plot of standardised residuals for the Quartic model")
qqline(y, col = 2, lwd = 1, lty = 2)
```

```
#Shapiro Wilk normality test
y = rstudent(model7)
shapiro.test(y)
```

```

	Shapiro-Wilk normality test

data:  y
W = 0.99171, p-value = 0.3954
```

```
#ACF plot for the residuals
acf(rstudent(model7), main = "ACF of standardized residuals the Quartic model
fitted to the return series.")
```

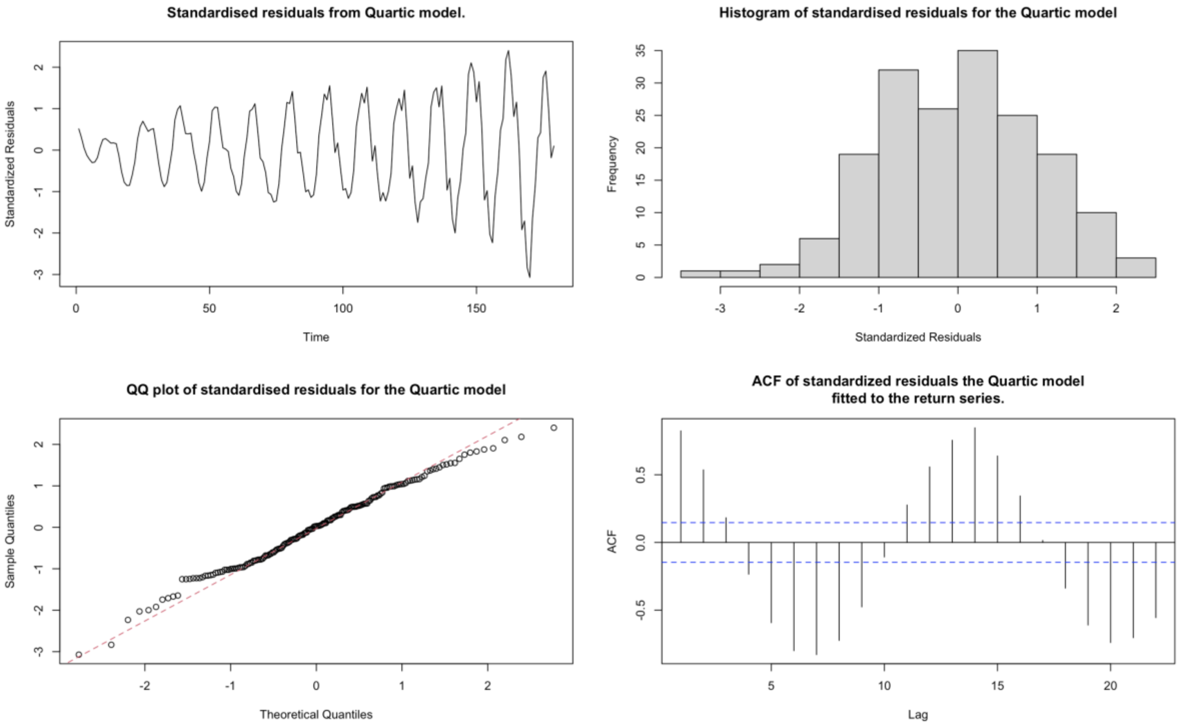

Figure 28:Residual Analysis of Quartic Model

3.3 Forecasting with Regression Model

From the above analysis, Model 6 – Seasonal + Quartic has the best fit among others, however as noted in the residual analysis the model cannot capture the seasonality and therefore to adhere to the principle of parsimony, Model 7 Quartic Model is a better fit. The quartic model as a whole and its coefficient are significant. It also has an adjusted R-square of 0.92 indicating that it explains 92% of the variation in the time series, making it a good model. Other models that can be considered are Model 2 - Quadratic and Model 5- Cubic Models.

Model 7: Quartic Model

Figure 29 shows the forecasts for the next 5 days using a quartic model. The forecasts are following the fitted line, and the forecast interval seems realistic with smaller gap between the lower and upper bounds, however it is not following the curves/patterns of the Time series plot. There are also some data points in the last quarter of the plot that are not within the forecast intervals.

```
h <- 5 # 5 steps ahead forecasts
lastTimePoint <- t[length(t)]
aheadTimes <- data.frame(t = seq(lastTimePoint+(1), lastTimePoint+h*(1), 1),
                         t2 =  seq(lastTimePoint+(1), lastTimePoint+h*(1), 1)^2,
                         t3 =  seq(lastTimePoint+(1), lastTimePoint+h*(1), 1)^3,
                         t4 = seq(lastTimePoint+(1), lastTimePoint+h*(1), 1)^4
)
frcModel7 <- predict(model7, newdata = aheadTimes, interval = "prediction")
frcModel7
```

```
       fit      lwr      upr
1 198.5896 159.7438 237.4355
2 202.8253 163.6214 242.0292
3 207.0315 167.4266 246.6364
4 211.2050 171.1529 251.2571
5 215.3425 174.7938 255.8913
```

```
plot(Stock_TS, xlim= c(t[1],aheadTimes$t[nrow(aheadTimes)]), ylim = c(-150,300), ylab = "return series",
     main = "Forecasts from the Quartic model fitted to the return series.")
lines(ts(fitted.model7,start = t[1],frequency = 1))
```

```
lines(ts(as.vector(frcModel7[,3]), start = aheadTimes$t[1],frequency = 1), col="blue", type="l")
lines(ts(as.vector(frcModel7[,1]), start = aheadTimes$t[1],frequency = 1), col="red", type="l")
```

```
lines(ts(as.vector(frcModel7[,2]), start = aheadTimes$t[1],frequency = 1), col="blue", type="l")
legend("topleft", lty=1, pch=1, col=c("black","blue","red"),
       c("Data","5% forecast limits", "Forecasts"))
```

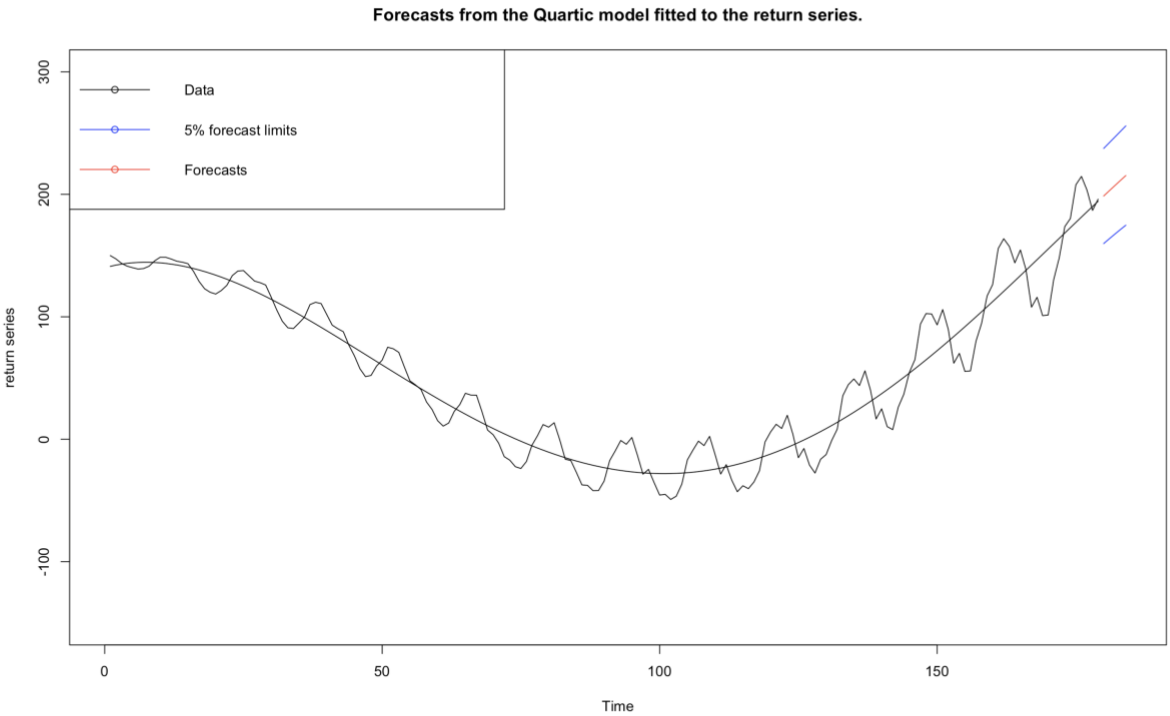

Figure 29: Forecasts using Quartic Model

Conclusion

From the above Model fitting and residual Analysis, it is quite evident that none of the models can provide a perfect fit. The residual analysis shows a significant seasonality and changing variance which is not being captured by the models above. ACF plots shows significant autocorrelation in the lags despite combining a seasonal model with the quartic model. The leftover auto correlations in the ACF could also indicate a stochastic trend and therefore would require a stochastic trend model in order to find randomness in the residual analysis and a perfectly fitting model for the given dataset. For the purpose of this exercise, forecasting was done with the Quartic model in order to follow the principle of parsimony.

Appendix

```
library("TSA") # calling the Time Series Package
library("readr")
library("dplyr")
#x11()

#Section 3.1 Descriptive Analysis

#Data Preprocessing
stock <- read_csv("~/OneDrive - RMIT University/Time Series Analysis/Assignment
1/assignment1Data2024.csv")
class(stock)
colnames(stock)[1:2]<- c("DayNo","returns") # converting the column name

summary(stock) # creating Summary Statistics for the raw data

#Converting to a TS object
Stock_TS <- ts(as.vector(stock$returns),start = c(1,1),end =c(1,179), frequency = 1)
class(Stock_TS)
summary(Stock_TS)

#creating a TS plot
par(mfrow=c(1,2))
plot(Stock_TS,type='o',ylab='Return on portfolio(in AUD100)',
   main = " Time series plot of Return on investment portfolio")
# Adding weekdays label to check seasonality
stock <- rbind(stock, list('180', "0")) # creating an additional row to add label as 179 is
not divisible by 5
stock<- stock %>% mutate(days = rep(c("M","T","W","Th","F"),times =36))
stock<- stock[-c(180),]# deleting the new row created earlier
plot(Stock_TS,type='l',ylab='Return on portfolio(in AUD100)',
   main = " Time series plot of Return on investment portfolio with labels")
points(y = Stock_TS, x = time(Stock_TS),
     pch = stock$days) #checking seasonality based on days label

#Lag - checking the impact of previous day's returns on the next day's return
```

```
y = Stock_TS
x = zlag(Stock_TS) # generate the first lag of the returns time series
head(y)
head(x)
index = 2:length(x) # Create an index to get rid of the first NA value in x
cor(y[index],x[index])
plot(y[index],x[index],ylab='returns series', xlab='The first lag of returns series',
   main = "Scatter plot of the series with first lag")

# looking at the second lag
x = zlag(zlag(Stock_TS))
index = 3:length(x)
cor(y[index],x[index])

plot(y[index],x[index],ylab='returns series', xlab='The second lag of returns series'
   ,main = "Scatter plot of the series with second lag")

#Displying the ACF
acf(Stock_TS, lag.max = 60,main = "ACF of solar return series")
#----------------------------------------------------------------
#Section 3.2 Model building Strategy

# 3.2.1 Linear trend Model
t <- time(Stock_TS) # Get a continuous time points in TS
model1<- lm(Stock_TS ~ t) #
summary(model1)

 #Fitting the Model
fitted.model1 <- fitted(model1) #alternate method
plot(Stock_TS,ylab='returns series',xlab='Days',type='o',
   main = "Time series plot of Return on investment portfolio")
abline(model1) # add the fitted linear trend line

 #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model1),x=as.vector(time(Stock_TS)), xlab='Time',
   ylab='Standardized Residuals',type='l', main = "Standardised residuals from linear
model")
#plotting a histogram
hist(rstudent(model1),xlab='Standardized Residuals', main = "Histogram of
standardised residuals from linear model")
# QQ plot
y = rstudent(model1)
qqnorm(y, main = "QQ plot of standardised residuals for the linear model")
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
shapiro.test(y)
```

```
#ACF plot for the residuals
acf(rstudent(model1), main = "ACF of standardized residuals for the linear model")
par(mfrow=c(1,1))



#3.2.2 Model 2 - quadratic model
t = time(Stock_TS)
t2 = t^2 # Create t^2
model2 = lm(Stock_TS ~ t + t2)
summary(model2)

  #Fitting the model
fitted.model2 <- fitted(model2)
plot(ts(fitted.model2), ylim = c(min(c(fitted(model2), as.vector(Stock_TS))),
max(c(fitted(model2),as.vector(Stock_TS)))),
    ylab='y' , main = "Fitted quadratic curve to return series", type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")

  #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model2),x=as.vector(time(Stock_TS)), xlab='Time',
    ylab='Standardized Residuals',type='l', main = "Standardised residuals from quadratic
model.")
#plotting a histogram
hist(rstudent(model2),xlab='Standardized Residuals', main = "Histogram of
standardised residuals for the quadratic model")
# plotting QQ plot
y = rstudent(model2)
qqnorm(y, main = "QQ plot of standardised residuals for the quadratic model
fitted to the return series.")
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
y = rstudent(model2)
shapiro.test(y)
#ACF plot for the residuals
acf(rstudent(model2), main = "ACF of standardized residuals the quadratic model
fitted to the return series.")
par(mfrow=c(1,1))

#3.2.3 Model 3 - Seasonal model
Weekday. <- factor(stock$days,levels = c( "M","T","W","Th","F"))
model3=lm(Stock_TS ~ Weekday.-1) # -1 removes the intercept term
summary(model3)

  #Fitting the Model
```

```
plot(ts(fitted(model3)), ylab='returns series', main = "Fitted seasonal model to returns
series.",
    ylim = c(-50,250), col = "red" )
lines(as.vector(Stock_TS),type="o")

  #Residual Analysis
#Model 3- seasonal model
par(mfrow=c(2,2))
plot(y=rstudent(model3),x=as.vector(time(Stock_TS)), xlab='Time',
    ylab='Standardized Residuals',type='l', main = "Standardised residuals from seasonal
model")
points(y=rstudent(model3),x=as.vector(time(Stock_TS)),
      pch=as.vector(stock$days))
#plotting a histogram
hist(rstudent(model3),xlab='Standardized Residuals', main = "Histogram of
standardised residuals for the seasonal model")
#plotting qq plot
y = rstudent(model3)
qqnorm(y, main = "QQ plot of standardised residuals for the seasonal model
fitted to the returns series.")
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
y = rstudent(model3)
shapiro.test(y)
#ACF plot for the residuals
acf(rstudent(model3), main = "ACF of standardized residuals the seasonal model
fitted to the return series.")
par(mfrow=c(1,1))


#3.2.4 Model 4 - Cosine model
#creating a new TS object as harmonic model requires a frequency of more than 1
Stock_TS_5 <- ts(as.vector(stock$returns),start = c(1,1),end =c(1,179), frequency = 5)
har. <- harmonic(Stock_TS_5, 1) # calculate cos(2*pi*t) and sin(2*pi*t)
data <- data.frame(Stock_TS_5,har.)
model4 <- lm(Stock_TS_5 ~ cos.2.pi.t. + sin.2.pi.t. , data = data)
summary(model4)
  #Fitting the model
par(mfrow=c(1,1))
plot(ts(fitted(model4)), ylab='returns series', main = "Fitted cosine wave to returns
series.",
    ylim = c(-50,250), col = "green" )
lines(as.vector(Stock_TS_5),type="o")
  #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model4),x=as.vector(time(Stock_TS_5)), xlab='Time',
```

```
    ylab='Standardized Residuals',type='l', main = "Time series plot of standardised residuals for the cosine wave")
#Plotting histogram
hist(rstudent(model4),xlab='Standardized Residuals', main = "Histogram of standardised residuals for the cosine wave")
#QQ plot
y = rstudent(model4)
qqnorm(y, main = "QQ plot of standardised residuals for the cosine wave")
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
y = rstudent(model4)
shapiro.test(y)
#ACF plot for the residuals
acf(rstudent(model4), main = "ACF of standardized residuals for the cosine model")
par(mfrow=c(1,1))

# #3.2.5 Model 5 Cubic model
t = time(Stock_TS)
t2 = t^2 # Create t^2
t3 = t^3
model5 = lm(Stock_TS ~ t + t2+t3)
summary(model5)

  #Fitting the model
fitted.model5 <- fitted(model5)
plot(ts(fitted.model5), ylim = c(min(c(fitted(model5), as.vector(Stock_TS))),
max(c(fitted(model5),as.vector(Stock_TS)))),
    ylab='y' , main = "Fitted cubic curve to return series", type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")

  #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model5),x=as.vector(time(Stock_TS)), xlab='Time',
    ylab='Standardized Residuals',type='l', main = "Standardised residuals from cubic model.")
#plotting a histogram
hist(rstudent(model5),xlab='Standardized Residuals', main = "Histogram of standardised residuals for the cubic model")
# plotting QQ plot
y = rstudent(model5)
qqnorm(y, main = "QQ plot of standardised residuals for the quadratic model fitted to the return series.")
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
y = rstudent(model5)
shapiro.test(y)
#ACF plot for the residuals
```

```
acf(rstudent(model5), main = "ACF of standardized residuals the cubic model
fitted to the return series.")

#3.2.6 Model 6 :seasonal plus quartic time trend model
t = time(Stock_TS)
t2 = t^2 # Create t^2
t3 = t^3
t4 = t^4 # takes into account the curve
model6 = lm(Stock_TS~ Weekday. + t + t2+t3+t4 -1)
summary(model6)

  #Fitting the model
fitted.model6 <- fitted(model6)
plot(ts(fitted(model6)),ylim = c(min(c(fitted(model6), as.vector(Stock_TS))),
max(c(fitted(model6),as.vector(Stock_TS)))),
    ylab='y' , main = "Fitted seasonal plus quadratic curve to return series",
type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")

  #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model6),x=as.vector(time(Stock_TS)), xlab='Time',
    ylab='Standardized Residuals',type='l', main = "Standardised residuals from linear
model")
#plotting a histogram
hist(rstudent(model6),xlab='Standardized Residuals', main = "Histogram of
standardised residuals from linear model")
# QQ plot
y = rstudent(model6)
qqnorm(y, main = "QQ plot of standardised residuals for the linear model")
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
y = rstudent(model6)
shapiro.test(y)
#ACF plot for the residuals
acf(rstudent(model6), main = "ACF of standardized residuals for the linear model")
par(mfrow=c(1,1))

# #3.2.7 model 7 Quartic Model
t = time(Stock_TS)
t2 = t^2 # Create t^2
t3 = t^3
t4 = t^4
model7 = lm(Stock_TS ~ t + t2+t3+t4)
summary(model7)

  #Fitting the model
```

```
fitted.model7 <- fitted(model7)
plot(ts(fitted.model7), ylim = c(min(c(fitted(model7), as.vector(Stock_TS))),
max(c(fitted(model7),as.vector(Stock_TS)))),
   ylab='y' , main = "Fitted Quartic curve to return series", type="l",lty=2,col="red")
lines(as.vector(Stock_TS),type="o")

 #Residual Analysis
par(mfrow=c(2,2))
plot(y=rstudent(model7),x=as.vector(time(Stock_TS)), xlab='Time',
   ylab='Standardized Residuals',type='l', main = "Standardised residuals from Quartic
model.")
#plotting a histogram
hist(rstudent(model7),xlab='Standardized Residuals', main = "Histogram of
standardised residuals for the Quartic model")
# plotting QQ plot
y = rstudent(model7)
qqnorm(y, main = "QQ plot of standardised residuals for the Quartic model")
qqline(y, col = 2, lwd = 1, lty = 2)
#Shapiro Wilk normality test
y = rstudent(model7)
shapiro.test(y)
#ACF plot for the residuals
acf(rstudent(model7), main = "ACF of standardized residuals the Quartic model
fitted to the return series.")
par(mfrow=c(1,1))
#------------------------------------------

#3.3    Forecasting with Regression Model

 #Forecasting using Quartic Model
h <- 5 # 5 steps ahead forecasts
lastTimePoint <- t[length(t)]
aheadTimes <- data.frame(t = seq(lastTimePoint+(1), lastTimePoint+h*(1), 1),
               t2 =  seq(lastTimePoint+(1), lastTimePoint+h*(1), 1)^2,
               t3 =  seq(lastTimePoint+(1), lastTimePoint+h*(1), 1)^3,
               t4 = seq(lastTimePoint+(1), lastTimePoint+h*(1), 1)^4
)
frcModel7 <- predict(model7, newdata = aheadTimes, interval = "prediction")
frcModel7
plot(Stock_TS, xlim= c(t[1],aheadTimes$t[nrow(aheadTimes)]), ylim = c(-150,300), ylab
= "return series",
   main = "Forecasts from the Quartic model fitted to the return series.")
lines(ts(fitted.model7,start = t[1],frequency = 1))
lines(ts(as.vector(frcModel7[,3]), start = aheadTimes$t[1],frequency = 1), col="blue",
type="l")
lines(ts(as.vector(frcModel7[,1]), start = aheadTimes$t[1],frequency = 1), col="red",
type="l")
```

```
lines(ts(as.vector(frcModel7[,2]), start = aheadTimes$t[1],frequency = 1), col="blue",
type="l")
legend("topleft", lty=1, pch=1, col=c("black","blue","red"),
    c("Data","5% forecast limits", "Forecasts"))
```

References

- Dr. Demirhan (2024) ‘Analysis of Trends' [PowerPoint slides, MATH1318], RMIT University, Melbourne.
- Jonathan D. Cryer and Kung-Sik Chan (2008) Time Series Analysis: With Applications in R, Springer New York, NY