Introduction

This report explores various implementations of Distributed Lag Models, or DLMs on a time series describing the monthly averages of ASX (Ords) Price Index. In addition to the Price Index, other predictor variables in the dataset are described below:

- **Gold price (AUD):** The average price of Gold in AUD for the particular month
- **Crude Oil (Brent, USD/bbl):** The price of crude oil in USD per barrel for the particular month
- **Copper (USD/tonne):** The price of Copper in USD per tonne for the particular month

Initially, the time series is explored and observations are made on its stationarity. Then, various transformations are applied with the intent of making the series stationary. The series is decomposed in order to identify its components, and attempts are made to remove seasonal components. Multiple DLMs are applied to the series, including the polynomial DLM, Koyck DLM, finite DLM, and Autoregressive DLM (ARDL) model. Where applicable, multiple models are created with varying values of p and/or q to identify the ideal lag parameters. Finally, the ideal models are discussed using their summaries and residuals in order to identify the model which best describes the ASX price index series.

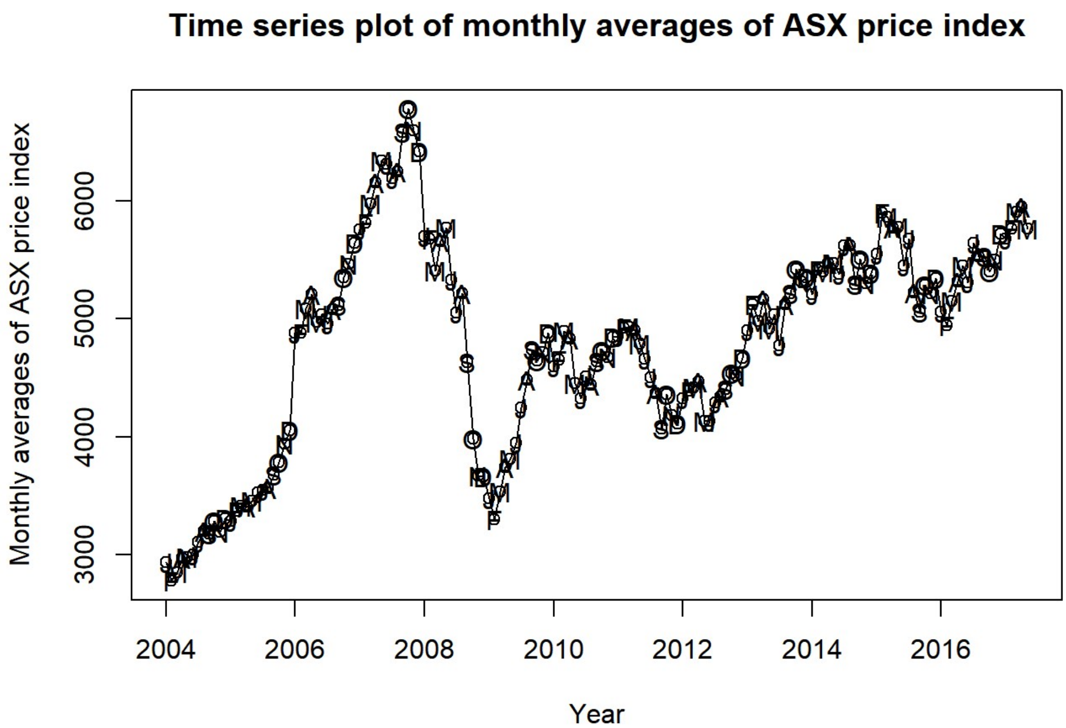
Reading Data and Pre-processing

The dataset is read in and the data types for Gold price and Copper price are changed to numeric since the existence of commas in those columns causes R to read them in as character columns initially. Then, each variable is converted to a time series object and the ASX price time series is plotted along with points denoting each month in order to identify any seasonality. However, the plot of ASX price does not suggest seasonality at the outset and it also does not seem stationary since the mean and variance of the plot seem to change over time.

```
setwd("C:/Work/Master in Analytics/Semester 2/Forecasting/Forecasting Assignment 1")
ASX_data <- read.csv("ASX_data.csv")
ASX_data$Gold.price <- as.numeric(gsub(",", "", ASX_data$Gold.price))
ASX_data$Copper_USD.tonne <- as.numeric(gsub(",", "", ASX_data$Copper_USD.tonne))
head(ASX_data)
```

```
##   ASX.price Gold.price Crude.Oil..Brent._USD.bbl Copper_USD.tonne
## 1    2935.4      611.9                     31.29             1650
## 2    2778.4      603.3                     32.65             1682
## 3    2848.6      565.7                     30.34             1656
## 4    2970.9      538.6                     25.02             1588
## 5    2979.8      549.4                     25.81             1651
## 6    2999.7      535.9                     27.55             1685
```

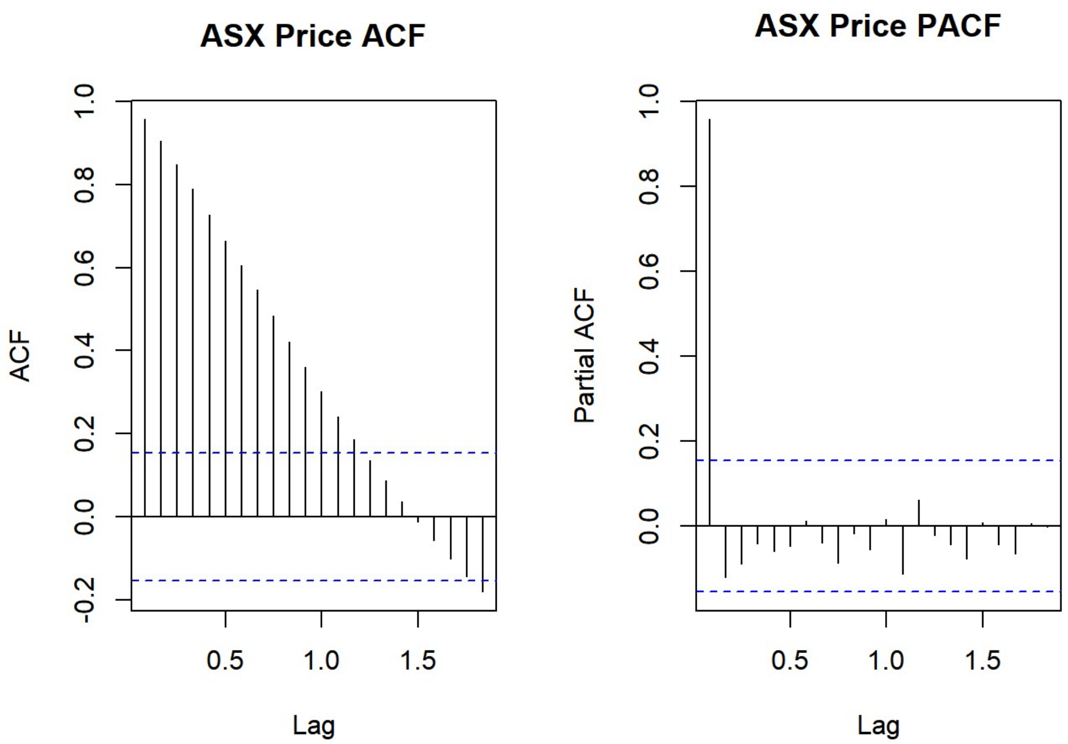
```
ASX_price.ts <- ts(ASX_data$ASX.price, start=c(2004,1), frequency=12)
Gold.ts = ts(ASX_data$Gold.price, start=c(2004,1), frequency=12)
Crude.Oil.ts = ts(ASX_data$Crude.Oil..Brent._USD.bbl, start=c(2004,1), frequency=12)
Copper.ts = ts(ASX_data$Copper_USD.tonne, start=c(2004,1), frequency=12)

plot(ASX_price.ts,ylab='Monthly averages of ASX price index',xlab='Year',type='o',
     main = "Time series plot of monthly averages of ASX price index")
points(y=ASX_price.ts,x=time(ASX_price.ts), pch=as.vector(season(ASX_price.ts)))
```



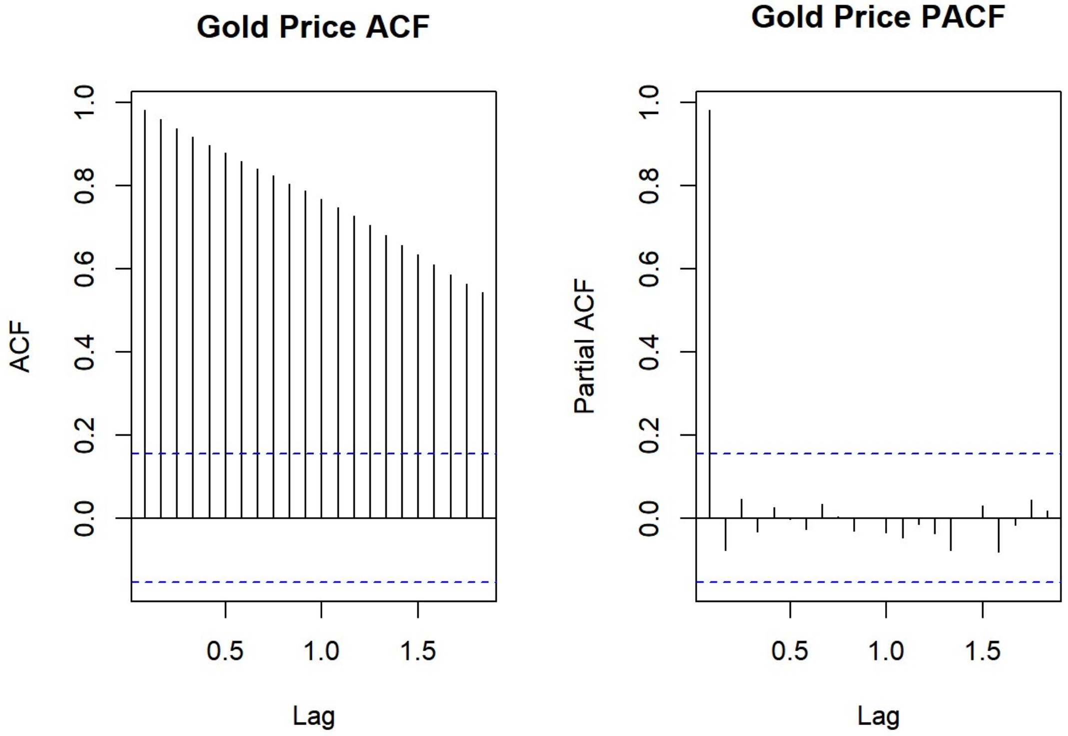
Exploring Stationarity of Variables

Plotting the ACF and PACF for all the variables demonstrates that they all produce similar results. All the variables demonstrate a gradually declining ACF with most peaks lying above the 95% confidence interval, and the PACF plot shows that the first lag is significant. This suggests that the series is non-stationary. The lack of peaks and troughs in the ACF and PACF plots demonstrates a lack of seasonality in the data.

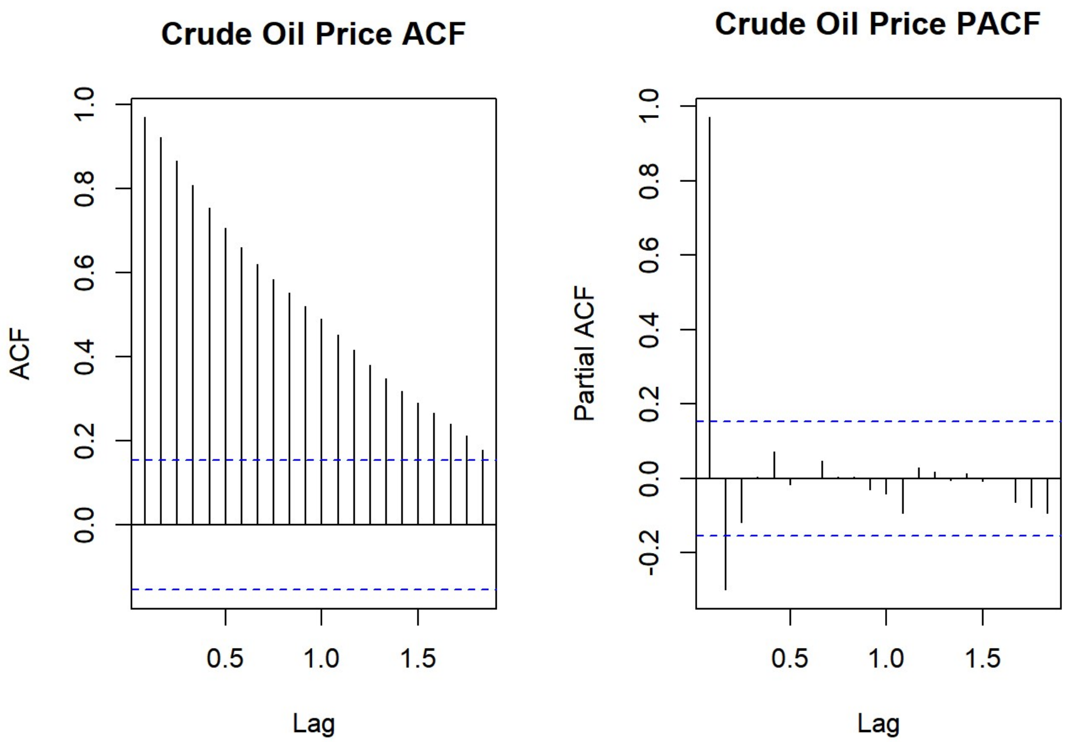
```
par(mfrow=c(1,2))
acf(ASX_price.ts, max.lag = 48, main="ASX Price ACF")
pacf(ASX_price.ts, max.lag = 48, main = "ASX Price PACF")
```



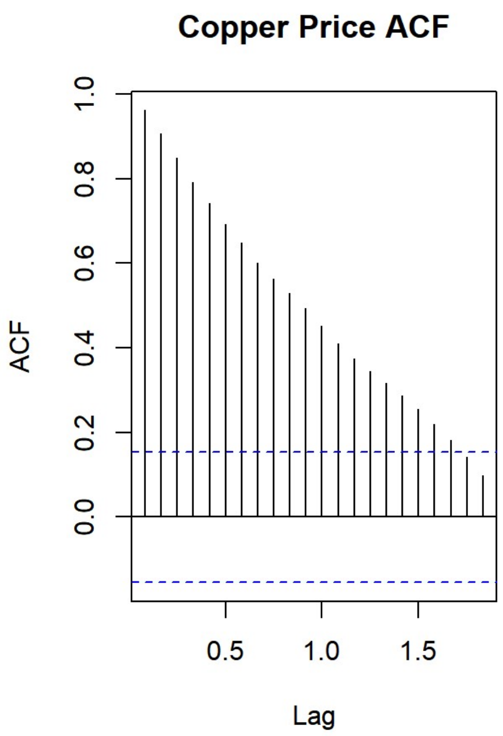

```
par(mfrow=c(1,2))
acf(Gold.ts, max.lag = 48, main="Gold Price ACF")
pacf(Gold.ts, max.lag = 48, main = "Gold Price PACF")
```

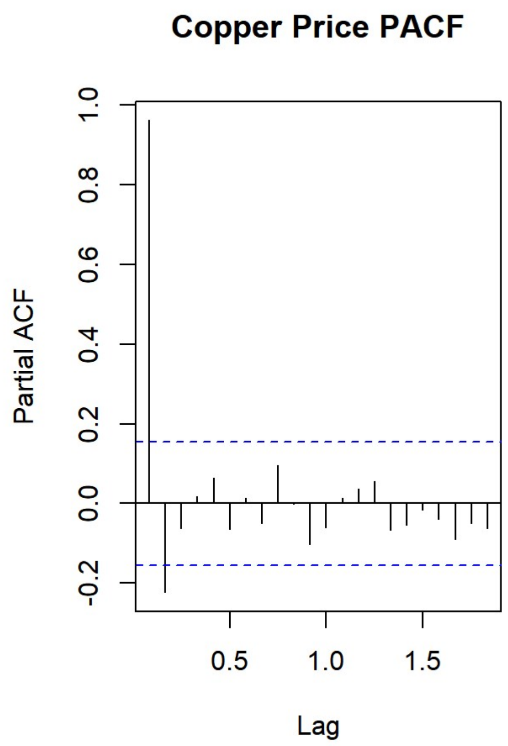



```
par(mfrow=c(1,2))
acf(Crude.Oil.ts, max.lag = 48, main="Crude Oil Price ACF")
pacf(Crude.Oil.ts, max.lag = 48, main = "Crude Oil Price PACF")
```




```
par(mfrow=c(1,2))
acf(Copper.ts, max.lag = 48, main="Copper Price ACF")
pacf(Copper.ts, max.lag = 48, main = "Copper Price PACF")
```





For further confirmation of non-stationarity in the series, an Augmented Dicky-Fuller test is performed on the ASX price series. The absolute value of the test statistic is lower than the absolute of the critical values at 1%, 5% and 10% significance levels. The p-value is also quite high (0.4286), all suggesting that the series is non-stationary due to a failure to reject the null hypothesis.

To account for the case of heteroskedasticity in the data, it could be useful to confirm non-stationarity using a Phillips-Perron unit root test in addtion to the ADF test. The Phillips-Perron test produces a Z-tau test statistic of -2.2025 which is less extreme than the critical values at 1%, 5%, and 10% levels. This implies a failure to reject the null hypothesis once again, and suggests that the series is indeed non-stationary.

```
adf.ASX = ur.df(ASX_price.ts, type = "none", lags = 1, selectlags = "AIC")
summary(adf.ASX)
```

```
## 
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
## 
## Test regression none 
## 
## 
## Call:
## lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -722.83 -103.12   36.63  138.79  809.17 
## 
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## z.lag.1    0.002107   0.003309   0.637    0.525
## z.diff.lag 0.085167   0.079885   1.066    0.288
## 
## Residual standard error: 203.3 on 157 degrees of freedom
## Multiple R-squared:  0.01074,    Adjusted R-squared:  -0.001866 
## F-statistic: 0.8519 on 2 and 157 DF,  p-value: 0.4286
## 
## 
## Value of test-statistic is: 0.6366 
## 
## Critical values for test statistics: 
##       1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
```

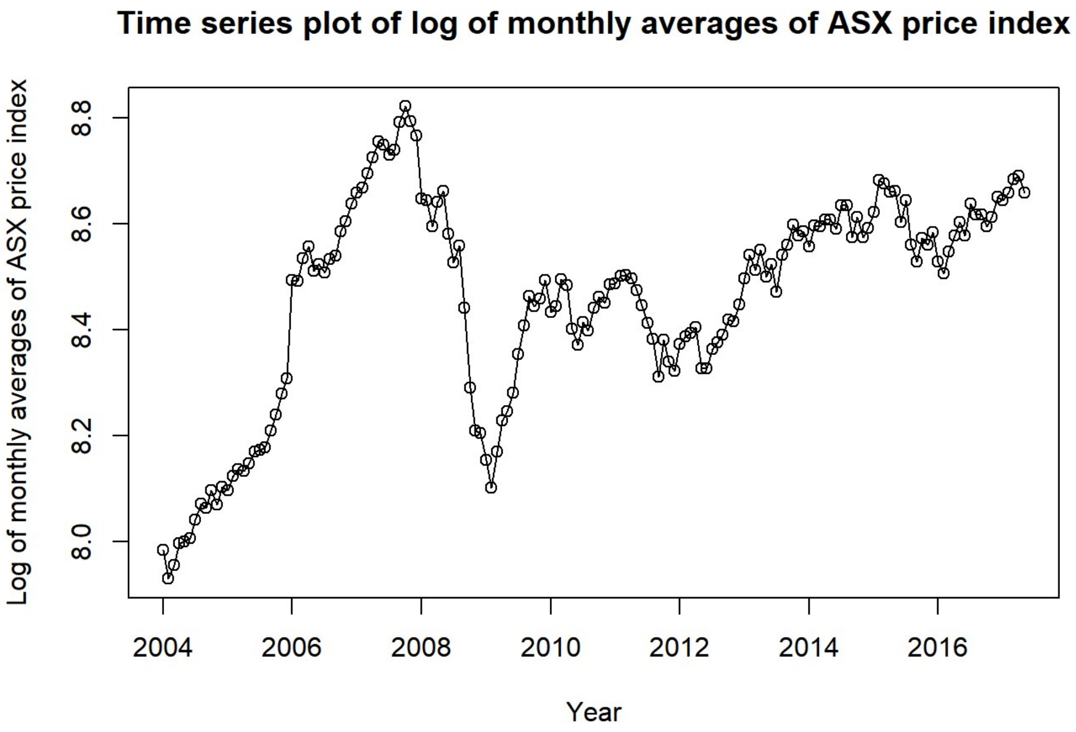
```
pp.ASX = ur.pp(ASX_price.ts, type = "Z-tau", lags = "short")
summary(pp.ASX)
```

```
##
## ###################################
## # Phillips-Perron Unit Root Test #
## ###################################
##
## Test regression with intercept
##
##
## Call:
## lm(formula = y ~ y.l1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -683.35 -104.89   15.86  135.94  782.18
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 191.38104   87.09761   2.197   0.0295 *
## y.l1          0.96386    0.01781  54.107   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 200.7 on 158 degrees of freedom
## Multiple R-squared:  0.9488, Adjusted R-squared:  0.9485
## F-statistic:  2928 on 1 and 158 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic, type: Z-tau  is: -2.2025
##
##          aux. Z statistics
## Z-tau-mu            2.3366
##
## Critical values for Z statistics:
##                      1pct      5pct     10pct
## critical values -3.472136 -2.879539 -2.576262
```

Applying Transformations to ASX Price Index

In order to make the series stationary, a log transformation is applied to the ASX price series and the transformed series is plotted. However, the transformed series looks quite similar to the original series, and this is confirmed by the subsequent ADF test which produces a p-value of 0.2168. Hence, the log transform fails to reject the null hypothesis that the series is non-stationary.

```
# Trying a log transformation to make the series stationary
log_ASX = log(ASX_price.ts)
plot(log_ASX,ylab='Log of monthly averages of ASX price index',xlab='Year',type='o',
     main = "Time series plot of log of monthly averages of ASX price index")
```



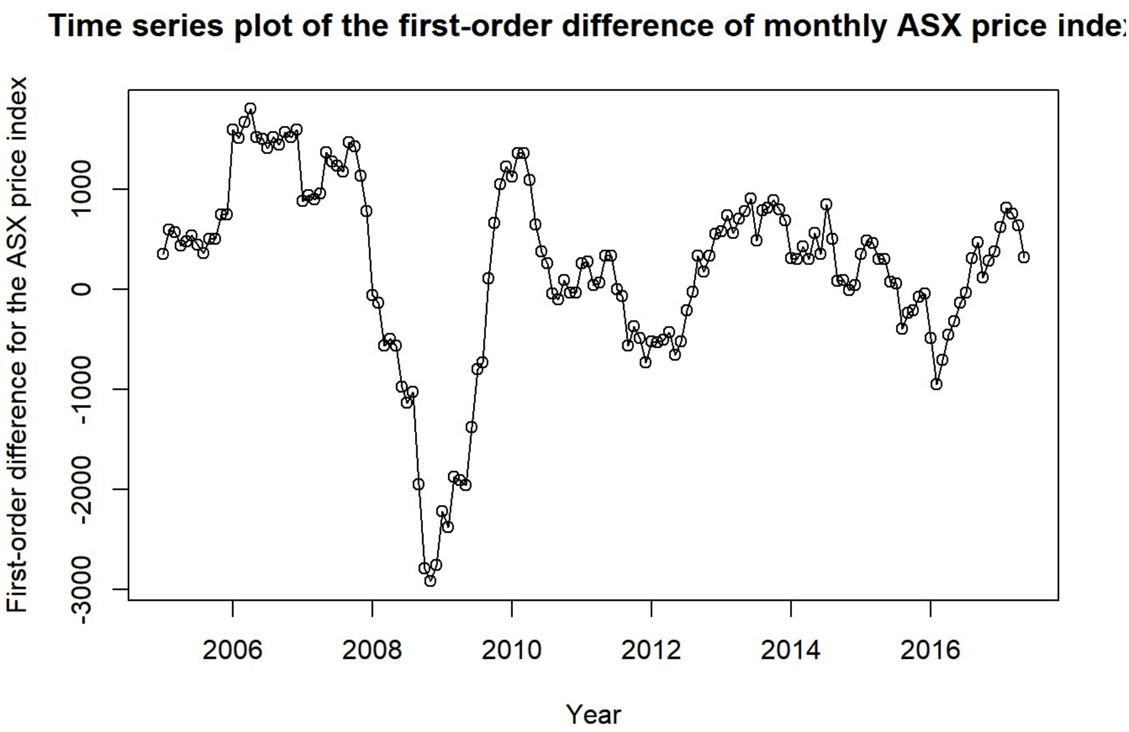

```
adf.test(log_ASX)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  log_ASX
## Dickey-Fuller = -2.8619, Lag order = 5, p-value = 0.2168
## alternative hypothesis: stationary
```

Another avenue to make the series stationary is through differencing. The first and second-order differences of the ASX price index are obtained and plotted. Both differences seem to even out the series a bit compared to the original, and this is further confirmed by performing ADF tests for both differenced series. The first-order differenced series produces a p-value of 0.09518, which is lower than the p-value for the original series, but still not enough to reject the null hypothesis of non-stationarity. However, the second-order differenced series produces a p-value of 0.01945 for the ADF test, effectively rejecting the null hypothesis for non-stationarity. Therefore, it is possible to suggest that the second-order differenced series for the ASX price index is stationary.

However, plotting the ACF and PACF plots for the second-order differenced series shows most of the peaks lying beyond the 95% confidence interval for the ACF plot and a few significant peaks for the PACF plot. This implies that the series is not truly stationary after all. This is further corroborated by performing a Phillips-Perron test on the second-order differenced series, which produces a Z-tau of -0.1205. Since this value is less extreme than the critical values at 1%, 5%, and 10% significant levels, the null hypothesis cannot be rejected, and the series is still assumed to be non-stationary.

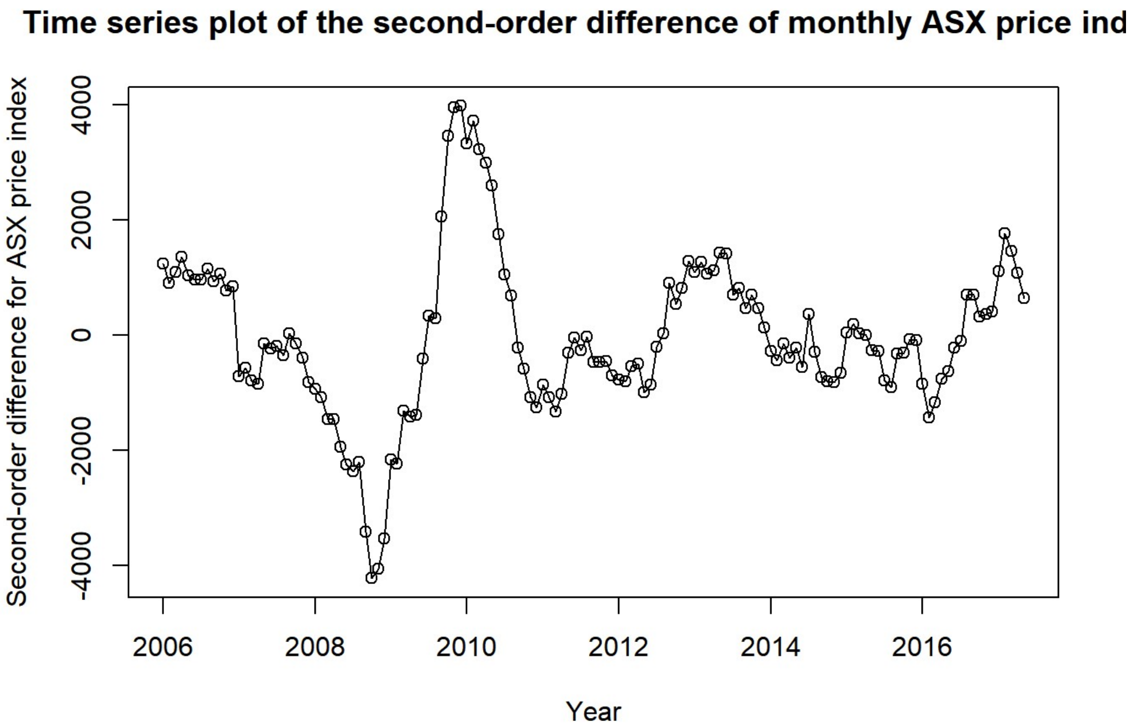
```
# Trying a first-order differencing transformation to make the series stationary
diff1_ASX = diff(ASX_price.ts, differences = 1, lag = 12)
plot(diff1_ASX,ylab='First-order difference for the ASX price index',xlab='Year',type='o',
     main = "Time series plot of the first-order difference of monthly ASX price index")
```




```
adf.test(diff1_ASX)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff1_ASX
## Dickey-Fuller = -3.1725, Lag order = 5, p-value = 0.09518
## alternative hypothesis: stationary
```

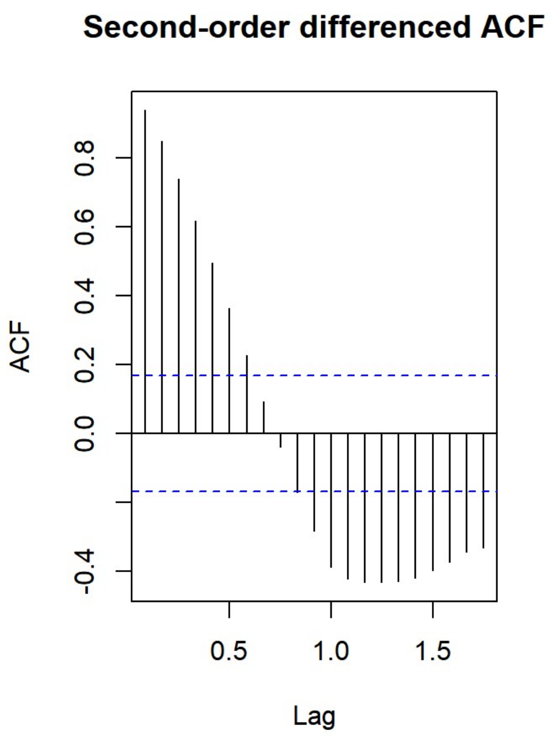
```
# Trying a second-order differencing transformation to make the series stationary
diff2_ASX = diff(ASX_price.ts, differences = 2, lag = 12)
plot(diff2_ASX,ylab='Second-order difference for ASX price index',xlab='Year',type='o',
     main = "Time series plot of the second-order difference of monthly ASX price index")
```

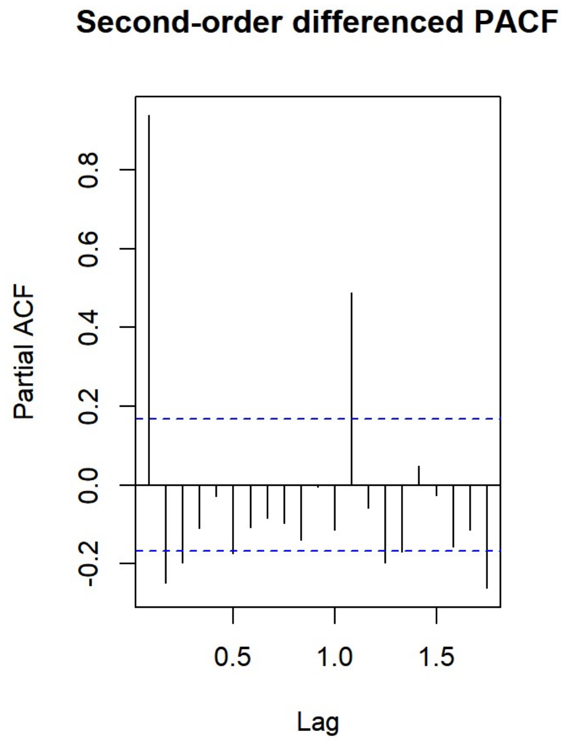



```
adf.test(diff2_ASX)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff2_ASX
## Dickey-Fuller = -3.8343, Lag order = 5, p-value = 0.01945
## alternative hypothesis: stationary
```

```
par(mfrow=c(1,2))
acf(diff2_ASX, max.lag = 48, main="Second-order differenced ACF")
pacf(diff2_ASX, max.lag = 48, main = "Second-order differenced PACF")
```






```
pp.ASX = ur.pp(diff2_ASX, type = "Z-tau", lags = "short")
summary(pp.ASX)
```

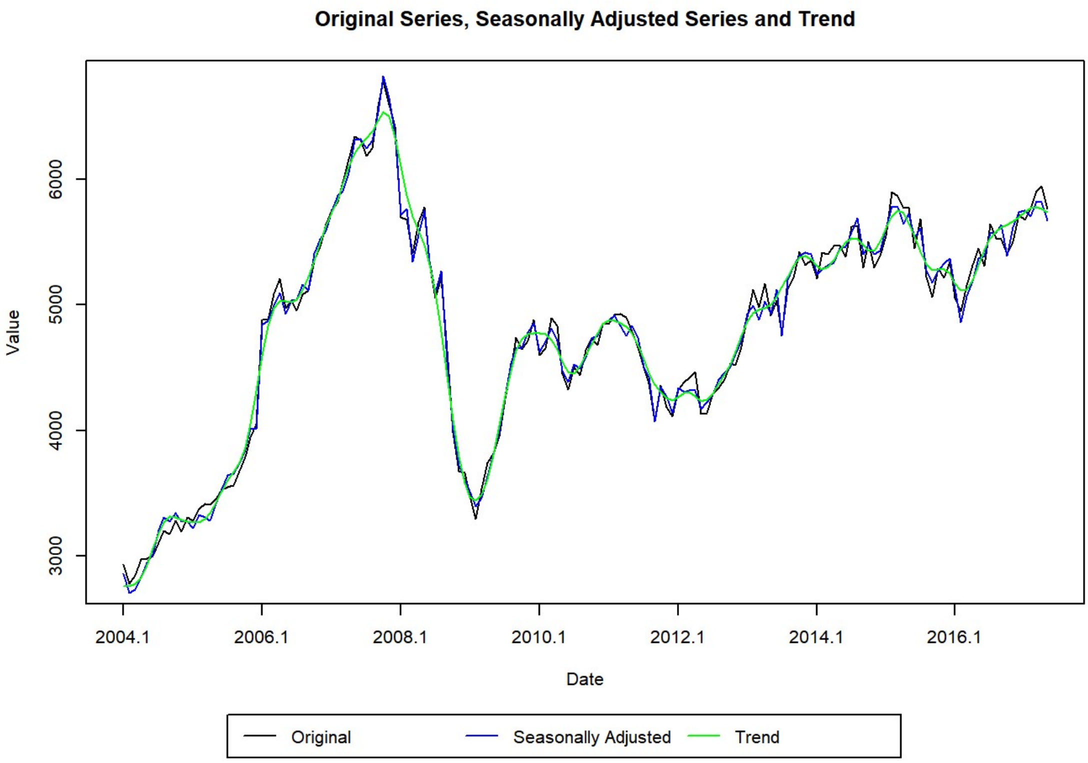
```
##
## ###################################
## # Phillips-Perron Unit Root Test #
## ###################################
##
## Test regression with intercept
##
##
## Call:
## lm(formula = y ~ y.l1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1506.30  -261.65   -44.85   236.67  1780.08
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.54002   40.77979  -0.136    0.892
## y.l1         0.93916    0.02912  32.256   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 475.5 on 134 degrees of freedom
## Multiple R-squared:  0.8859, Adjusted R-squared:  0.8851
## F-statistic:  1040 on 1 and 134 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic, type: Z-tau  is: -2.6823
##
##          aux. Z statistics
## Z-tau-mu           -0.1205
##
## Critical values for Z statistics:
##                      1pct      5pct     10pct
## critical values -3.479192 -2.882684 -2.577916
```

Verifying Seasonality

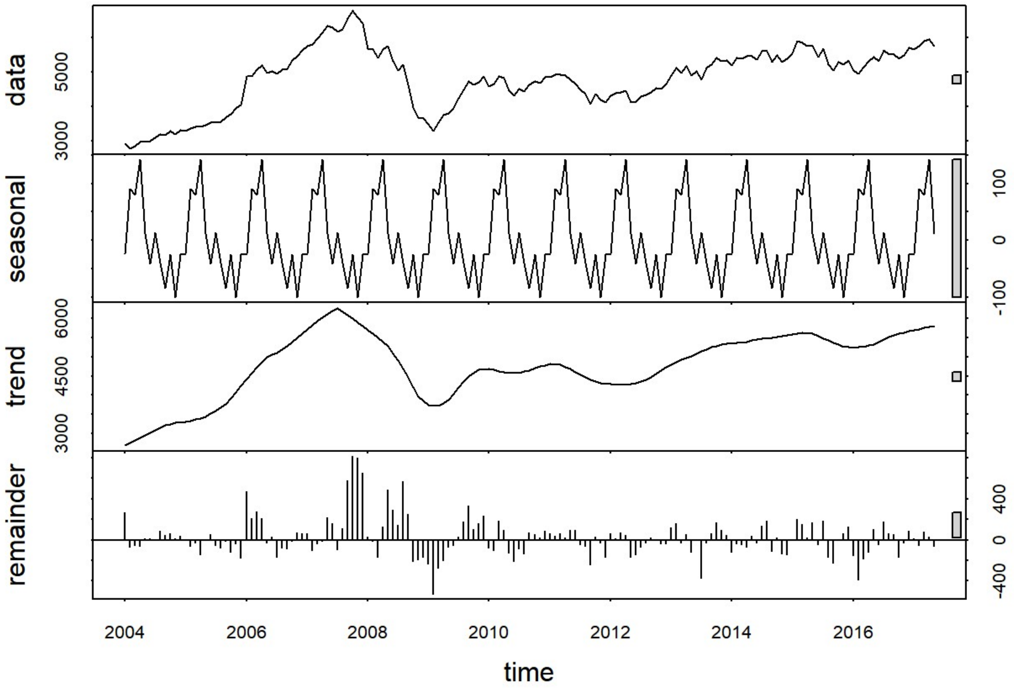
The seasonal components of the ASX price index series are explored using the X-12 ARIMA and stl functions. The seasonally adjusted series obtained from the X-12 function is almost identical to the original series, suggesting that there is not much of a seasonal component to the series. This is further corroborated by the results of the STL function, which does manage to isolate a seasonal component. However, the relative magnitude of the seasonal component compared to the trend and data is much smaller when comparing the plots using the range bars on the right-hand side. This suggests that only a very small component of the data can be explained by seasonal effects.

This is further confirmed by using the seasadj() function to obtain a seasonally adjusted series for ASX price, which looks almost identical to the original series. Plotting the ACF and PACF functions for the seasonally adjusted series produces significant peaks with a gradual decline, suggesting that the seasonal adjustments did not affect the non-stationarity of the series.

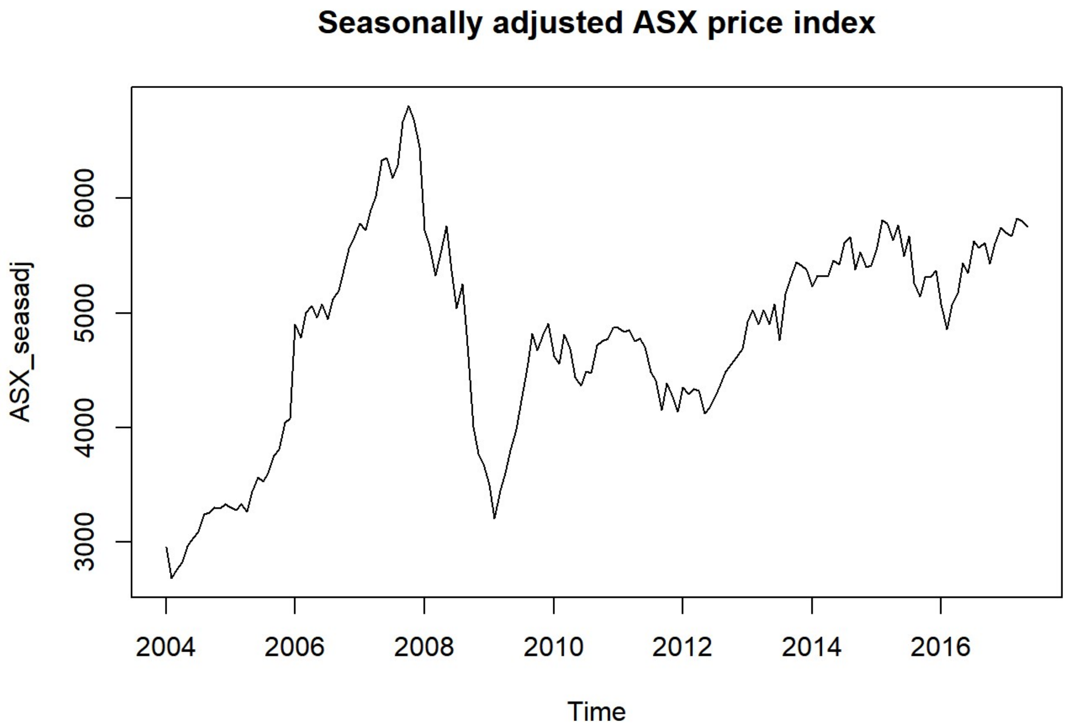
```
# Decomposing and adjusting seasonality

decomp_ASX_x12 = x12(ASX_price.ts)
plot(decomp_ASX_x12, sa=TRUE, trend=TRUE)
```



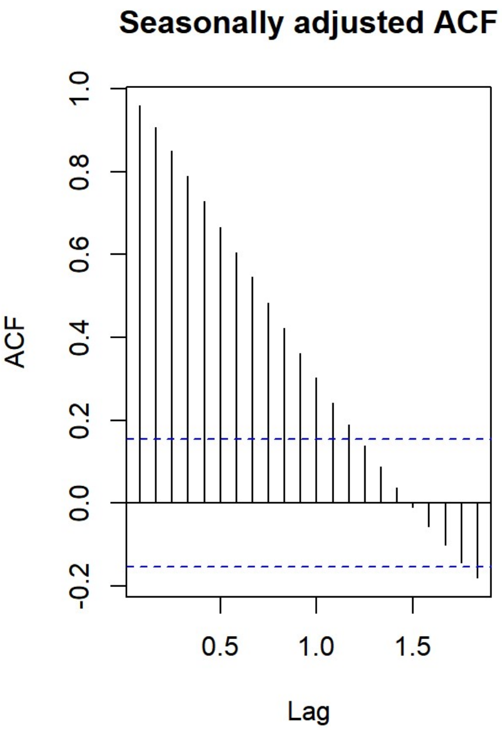

```
decomp_ASX = stl(ASX_price.ts, t.window = 12, s.window = "periodic", robust=TRUE)
plot(decomp_ASX)
```

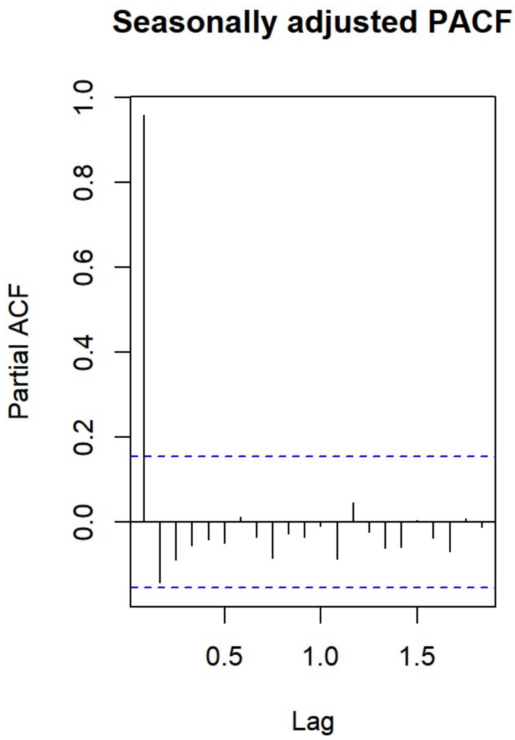



```
ASX_seasadj = seasadj(decomp_ASX)
plot(ASX_seasadj, main = "Seasonally adjusted ASX price index")
```




```
par(mfrow=c(1,2))
acf(ASX_seasadj, max.lag = 48, main="Seasonally adjusted ACF")
pacf(ASX_seasadj, max.lag = 48, main = "Seasonally adjusted PACF")
```


Seasonally adjusted ACF
ACF
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
0.5
1.0
1.5
Lag

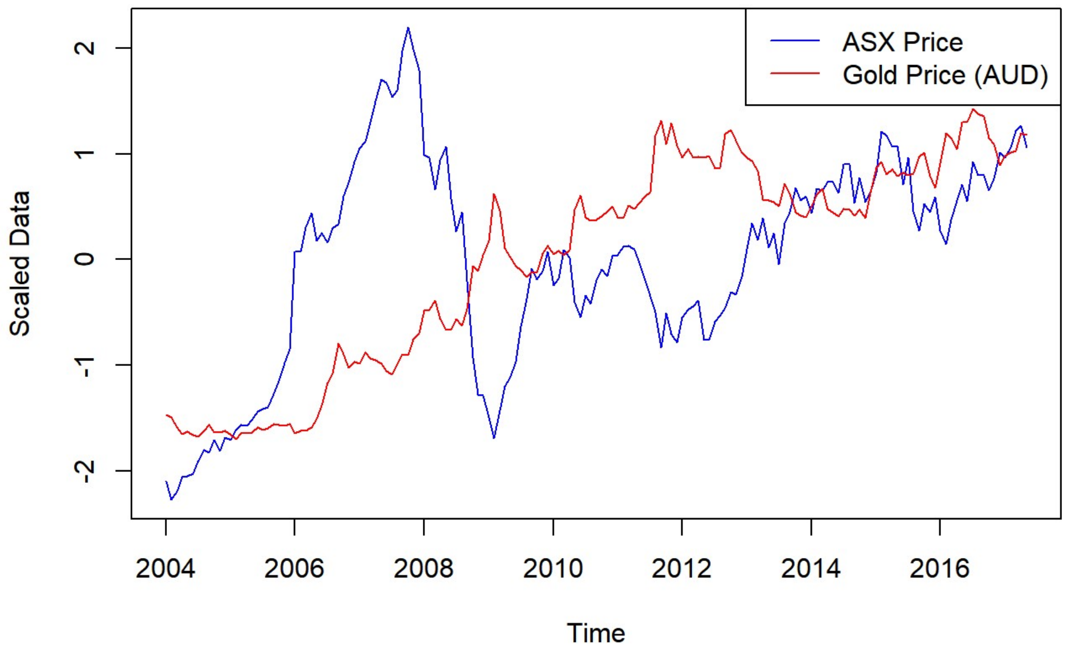

Seasonally adjusted PACF
Partial ACF
1.0
0.8
0.6
0.4
0.2
0.0
0.5
1.0
1.5
Lag

Correlating Predictor Variables with Dependent Variable

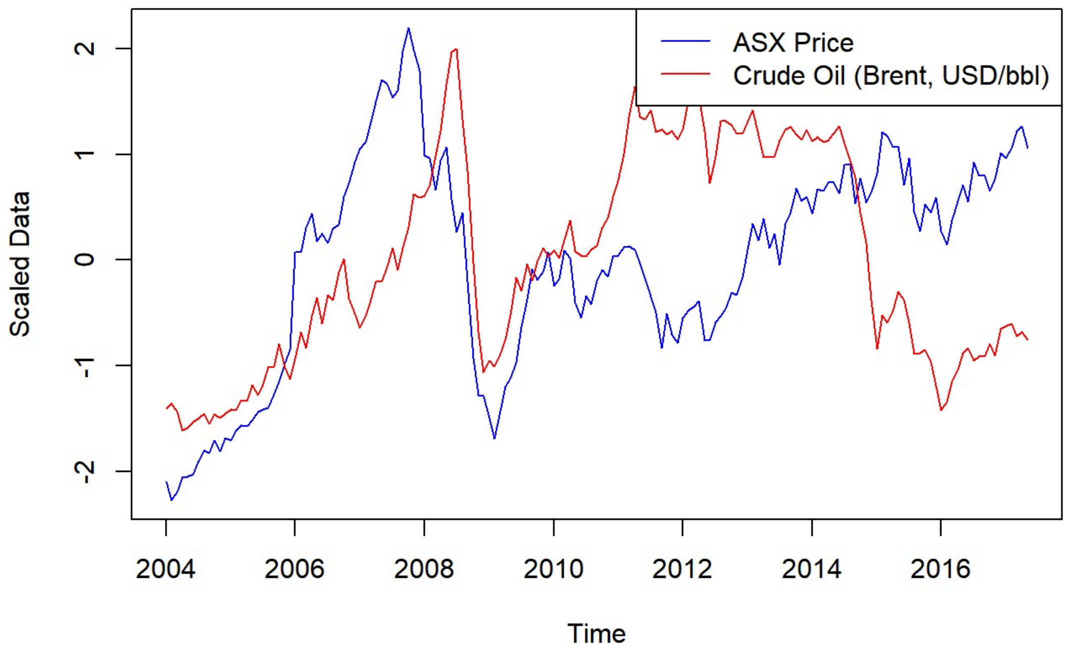
Certain Distributed Lag Models allow for modelling with multiple predictor variables while others only allow a singe predictor variable to be used. The models which allow multiple predictors are finite DLMs and ARDLs, while polynomial DLMs and Koyck DLMs only support a single predictor variable. For the latter case, it is important to choose a suitable predictor variable when modelling with only a single predictor variable. This should ideally be the predictor variable that is most closely correlated with the dependent variable. To obtain an idea of the relationship between predictor variables and the independent variable, the ASX price index is scaled and plotted along with each of the scaled predictor variables.

Gold price seems to follow the overall trend of the series but does not follow the peaks and troughs. Crude oil prices seem to follow the time series better than Gold price, and Copper price seems to follow the ASX price index even more closely than Crude oil price. However, none of the variables seem to explain the relative drop in ASX price between 2010 and 2014.

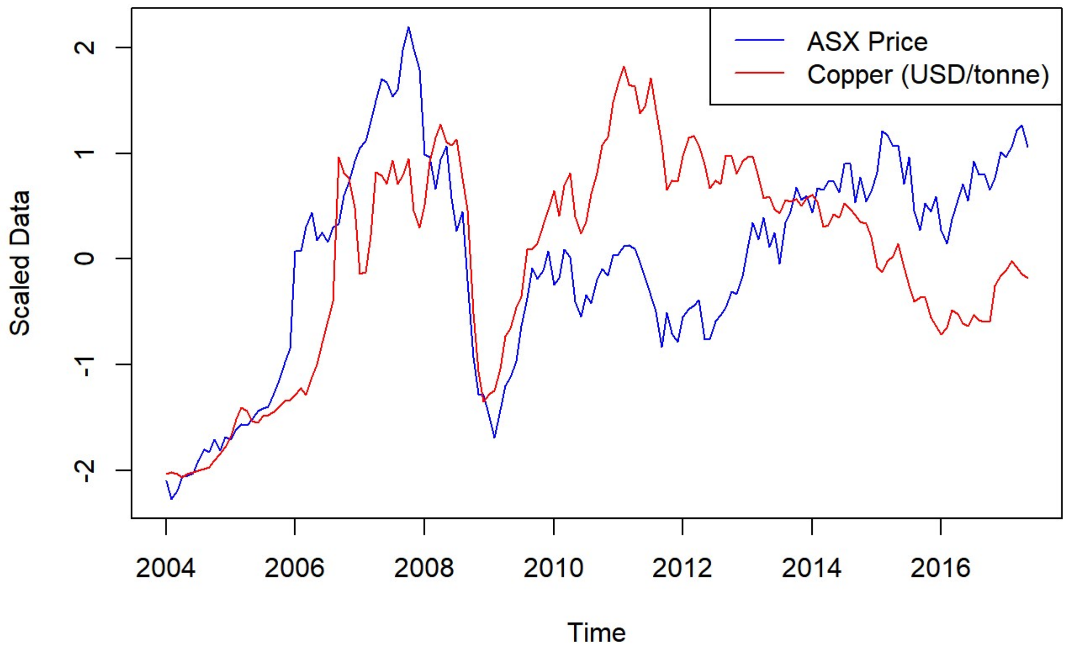
```
ASX_data.ts <- ts(ASX_data[,1:2], start=c(2004,1), frequency=12)
ASX_data_scaled = scale(ASX_data.ts)
plot(ASX_data_scaled, plot.type="s", col=c("blue", "red"), ylab="Scaled Data")
legend("topright", lty=1, col=c("blue", "red"), c("ASX Price", "Gold Price (AUD)"))
```




```
ASX_data.ts <- ts(ASX_data[,c(1,3)], start=c(2004,1), frequency=12)
ASX_data_scaled = scale(ASX_data.ts)
plot(ASX_data_scaled, plot.type="s", col=c("blue", "red"), ylab="Scaled Data")
legend("topright", lty=1, col=c("blue", "red"), c("ASX Price", "Crude Oil (Brent, USD/bbl)"))
```




```
ASX_data.ts <- ts(ASX_data[,c(1,4)], start=c(2004,1), frequency=12)
ASX_data_scaled = scale(ASX_data.ts)
plot(ASX_data_scaled, plot.type="s", col=c("blue", "red"), ylab="Scaled Data")
legend("topright", lty=1, col=c("blue", "red"), c("ASX Price", "Copper (USD/tonne)"))
```



Since a visual confirmation of the correlation between predictor and dependent variables is insufficient, the variables can be correlated using the cor() function in order to obtain a numerical representation of correlation. Obtaining the correlation values shows that Copper price is the most closely correlated with ASX price, with a value of 0.56, Crude oil and Gold price have correlation values of 0.33 and 0.34, respectively. Even though a correlation of 0.56 is not particularly high, Copper price is used as the predictor variable in the Koyck and polynomial DLM implementations since Copper price is the most highly correlated variable with ASX price.

```
ASX_data.ts = ts(ASX_data, start=c(2004,1), frequency=12)

#correlate variables
cor(ASX_data.ts)
```

```
##                           ASX.price Gold.price Crude.Oil..Brent._USD.bbl
## ASX.price                 1.0000000  0.3431908                 0.3290338
## Gold.price                0.3431908  1.0000000                 0.4366382
## Crude.Oil..Brent._USD.bbl 0.3290338  0.4366382                 1.0000000
## Copper_USD.tonne          0.5617864  0.5364213                 0.8664296
##                           Copper_USD.tonne
## ASX.price                        0.5617864
## Gold.price                       0.5364213
## Crude.Oil..Brent._USD.bbl        0.8664296
## Copper_USD.tonne                 1.0000000
```

Fitting Distributed Lag Models

Polynomial DLM

A polynomial DLM is first fitted using the polyDlm() function, using Copper price as a predictor variable to predict ASX price. Prior to fitting, the ideal lag value is found using the finiteDLMauto() function, which is made to generate multiple models with q ranging from 1 to 10 and uses AIC error to determine the suitability of each model. The results of finiteDLMauto() suggest that the ideal value for q is 10, and a polynomial order of 2.

```
#PolyDLM (Copper Price)
finiteDLMauto(x = as.vector(Copper.ts), y = as.vector(ASX_price.ts), q.min = 1, q.max = 10,
              model.type = "poly", error.type = "AIC", trace = TRUE)
```

```
##     q - k    MASE      AIC      BIC   GMRAE    MBRAE R.Adj.Sq Ljung-Box
## 10 10 - 2 3.97169 2420.626 2435.712 5.07815  0.15607  0.17411         0
## 9   9 - 2 4.00458 2437.682 2452.802 5.16393  1.44810  0.18623         0
## 8   8 - 2 4.03469 2455.235 2470.387 5.19326 -6.04331  0.19765         0
## 7   7 - 2 4.07139 2472.607 2487.791 5.09229 -0.06077  0.20824         0
## 6   6 - 2 4.09411 2490.095 2505.312 5.16972  2.76020  0.21979         0
## 5   5 - 2 4.11345 2507.644 2522.893 5.07058  0.76669  0.23276         0
## 4   4 - 2 4.15194 2525.094 2540.375 5.12005 -0.35391  0.24552         0
## 3   3 - 2 4.19231 2542.208 2557.521 5.19875  0.46247  0.25893         0
## 2   2 - 2 4.21688 2559.356 2574.700 5.34802  0.67093  0.27395         0
## 1   1 - 2 4.24262 2574.488 2586.789 5.38993 -0.49453  0.29391         0
```

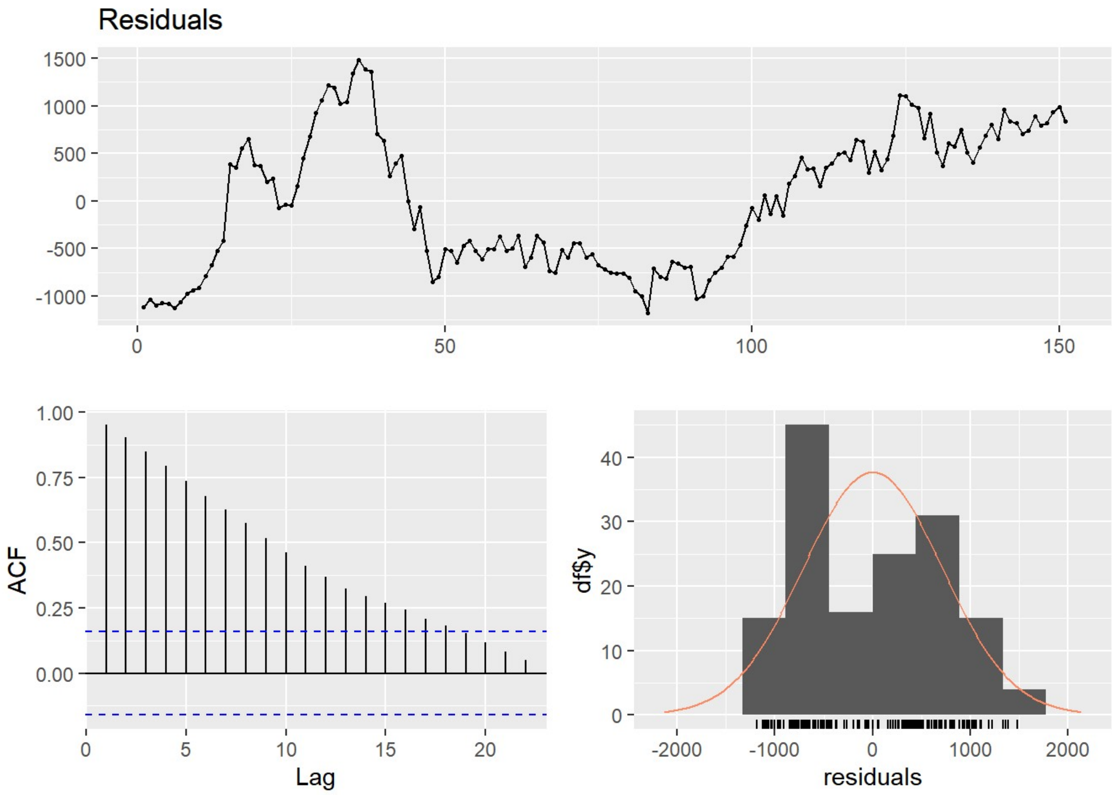
```
model.poly = polyDlm(x = as.vector(Copper.ts), y = as.vector(ASX_price.ts), q = 10, k = 2, show.beta = TRUE)
```

```
## Estimates and t-tests for beta coefficients:
##         Estimate Std. Error t value  P(>|t|)
## beta.0   0.09760    0.03120   3.130 2.14e-03
## beta.1   0.07150    0.01650   4.330 2.84e-05
## beta.2   0.04850    0.00916   5.300 4.44e-07
## beta.3   0.02860    0.01180   2.420 1.69e-02
## beta.4   0.01170    0.01580   0.741 4.60e-01
## beta.5  -0.00205    0.01730  -0.118 9.06e-01
## beta.6  -0.01270    0.01580  -0.806 4.22e-01
## beta.7  -0.02040    0.01170  -1.750 8.31e-02
## beta.8  -0.02490    0.00830  -3.000 3.18e-03
## beta.9  -0.02640    0.01530  -1.720 8.72e-02
## beta.10 -0.02480    0.02990  -0.829 4.09e-01
```

```
summary(model.poly)
```

```
##
## Call:
## "Y ~ (Intercept) + X.t"
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1180.62  -642.64   -40.32   616.63  1484.83
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.997e+03  2.088e+02  19.144  < 2e-16 ***
## z.t0         9.756e-02  3.119e-02   3.128  0.00212 **
## z.t1        -2.761e-02  1.802e-02  -1.532  0.12761
## z.t2         1.537e-03  1.772e-03   0.868  0.38706
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 718.2 on 147 degrees of freedom
## Multiple R-squared:  0.1906, Adjusted R-squared:  0.1741
## F-statistic: 11.54 on 3 and 147 DF,  p-value: 7.739e-07
```

```
checkresiduals(model.poly$model)
```




```
##
##  Breusch-Godfrey test for serial correlation of order up to 10
##
## data:  Residuals
## LM test = 138.14, df = 10, p-value < 2.2e-16
```

The summary of the polynomial DLM model shows a low p-value of 7.739e-07, which suggests that the model is significant. However, a low R-squared value of 0.1741 means that the model does not explain the data well. Additionally, only the z value at t is significant, while the first and second lags are not significant and do not act as suitable predictors for ASX price.

The Breusch-Godfrey test for serial correlation produces a low p-value of 2.2e-16, which means that there is evidence of serial correlation in the model, making it less suitable to model ASX price.

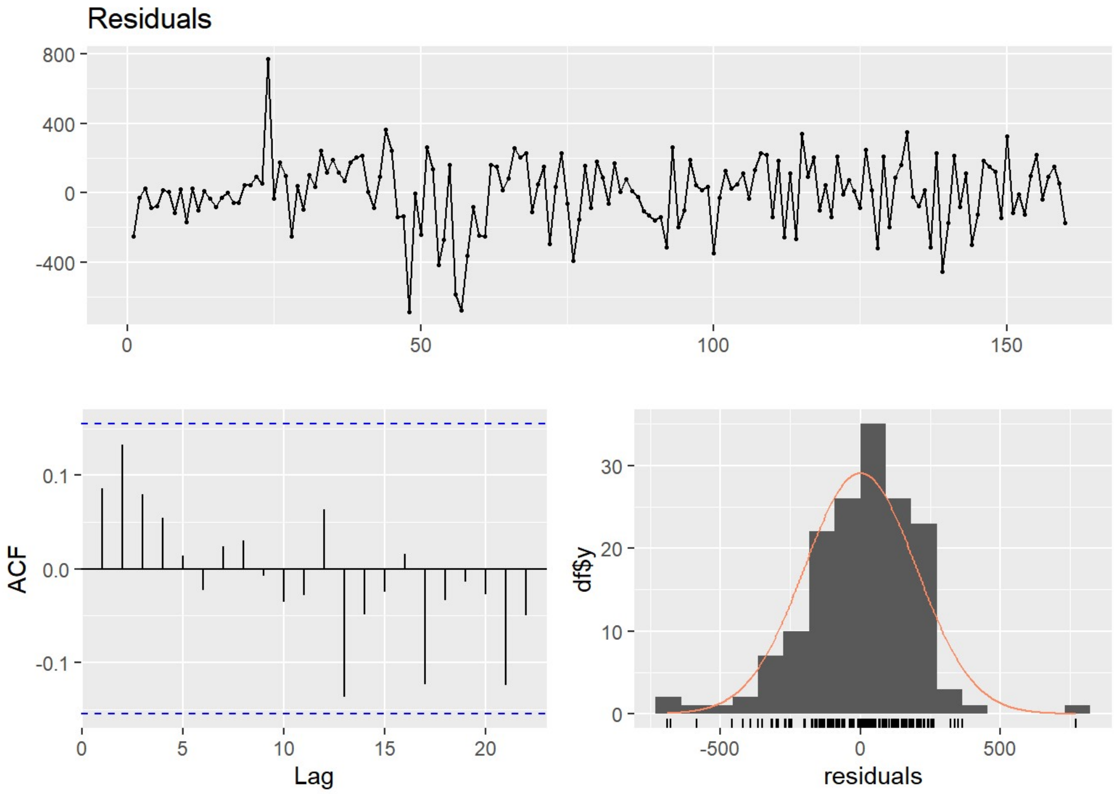
Analysing the residuals for the polynomial model shows that the residuals are relatively large and are not normally distributed, suggesting that the model does not account for the data well and the errors in the model are not random. The ACF plot also shows peaks above the 95% confidence interval, suggesting that this is an inadequate model to describe ASX price.

Koyck (Geometric) DLM

```
# Kocyk (Copper Price)
model.Koyck = koyckDlm(x = as.vector(Copper.ts), y = as.vector(ASX_price.ts), intercept = TRUE)
summary(model.Koyck)
```

```
##
## Call:
## "Y ~ (Intercept) + Y.1 + X.t"
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -689.64 -108.62   12.78  140.20  771.79
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 189.368812  87.644648   2.161   0.0322 *
## Y.1           0.971621   0.021895  44.376   <2e-16 ***
## X.t          -0.005864   0.009517  -0.616   0.5387
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 201.9 on 157 degrees of freedom
## Multiple R-Squared: 0.9485,  Adjusted R-squared: 0.9479
## Wald test:  1448 on 2 and 157 DF,  p-value: < 2.2e-16
##
## Diagnostic tests:
## NULL
##
##                              alpha         beta       phi
## Geometric coefficients:  6672.885 -0.005863623 0.9716211
```

```
checkresiduals(model.Koyck$model)
```




```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 6.2327, df = 10, p-value = 0.7953
##
## Model df: 0.   Total lags used: 10
```

A Koyck DLM is fitted using the koyckDlm() function, using Copper price as a predictor variable to predict ASX price. The summary of the Koyck model shows a very low p-value of 2.2e-16, which suggests that the model is significant. A high R-squared value of 0.9479 reveals that the model explains most of the data well.

The first lag of the Y value is significant but the X value at t is not significant, meaning that the X value at t is not a suitable predictor for ASX price.

The Ljung-Box test produces a high p-value of 0.7953, failing to reject the null hypothesis of independently distributed data. Therefore, it is possible to conclude that there is no serial correlation in the model.

Analysing the residuals for the Koyck model shows that the residuals are relatively small compared to the polynomial model and are roughly normally distributed, suggesting that the model accounts for the data well and the errors in the model are random. The ACF plot shows that the peaks are all within the 95% confidence interval, suggesting that this is an adequate model to describe ASX price.

Finite DLM

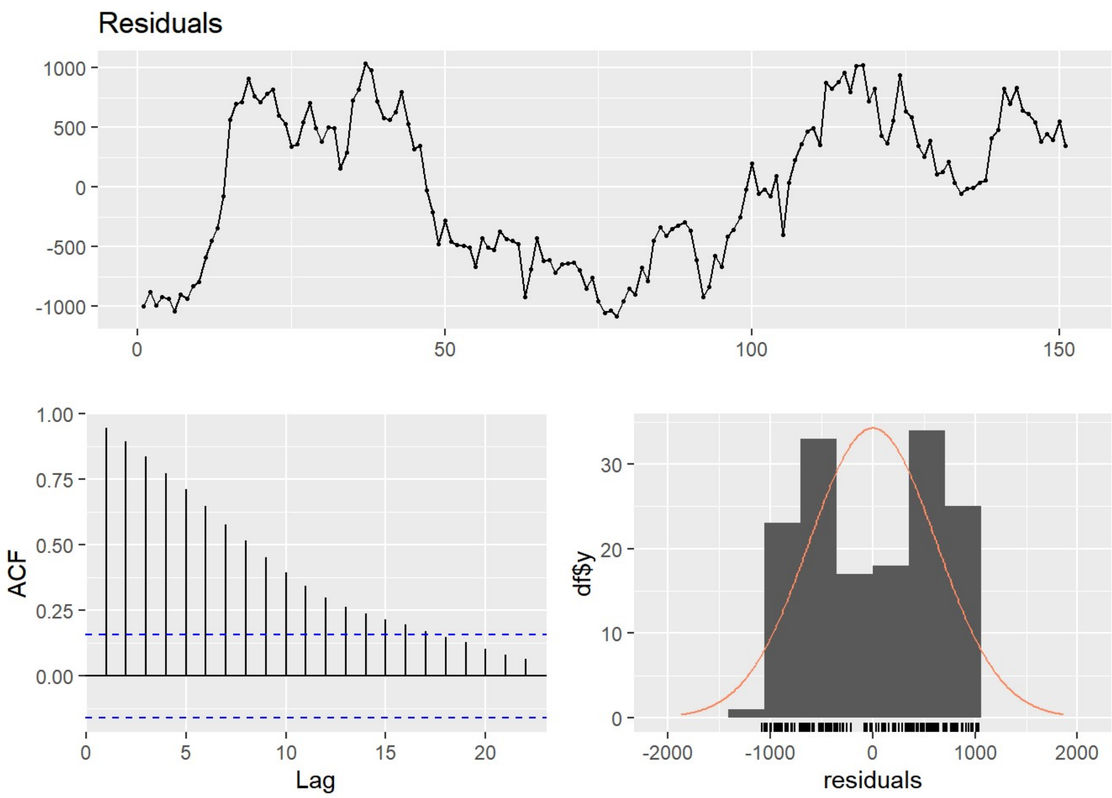
```
# Finite DLM
finiteDLMauto(formula = ASX.price ~ Gold.price + Crude.Oil..Brent._USD.bbl, data = ASX_data, q.min = 1, q.max
= 10,
              model.type = "dlm", error.type = "AIC", trace = TRUE)
```

```
##    q - k    MASE      AIC      BIC   GMRAE    MBRAE R.Adj.Sq Ljung-Box
## 10    10 3.85968 2472.752 2545.167 4.51817  0.80771 -0.04150         0
## 9      9 3.91880 2487.041 2553.566 4.59091 14.71560 -0.01713         0
## 8      8 3.97758 2501.647 2562.255 4.72289  1.69754  0.00681         0
## 7      7 4.03980 2516.044 2570.710 4.88556  0.30391  0.02917         0
## 6      6 4.08550 2531.088 2579.782 4.79178  1.42973  0.04879         0
## 5      5 4.13881 2546.306 2589.004 4.91945 -0.75355  0.06897         0
## 4      4 4.20872 2561.334 2598.008 5.09480  0.77841  0.08902         0
## 3      3 4.28929 2577.210 2607.836 5.00974 -6.04016  0.10274         0
## 2      2 4.35728 2593.043 2617.594 5.24889  1.33144  0.11879         0
## 1      1 4.42238 2609.286 2627.737 5.26525  1.44174  0.13299         0
```

```
model.dlm = dlm(formula = ASX.price ~ Gold.price + Crude.Oil..Brent._USD.bbl + Copper_USD.tonne, data = ASX_d
ata , q = 10)
summary(model.dlm)
```

```
##
## Call:
## lm(formula = as.formula(model.formula), data = design)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1083.44  -514.44    36.68   548.05  1036.19
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   3.959e+03  2.423e+02  16.343   <2e-16 ***
## Gold.price.t                 -1.535e+00  1.263e+00  -1.215    0.227
## Gold.price.1                  1.308e-01  1.810e+00   0.072    0.943
## Gold.price.2                 -1.574e-02  1.856e+00  -0.008    0.993
## Gold.price.3                 -8.298e-02  1.883e+00  -0.044    0.965
## Gold.price.4                 -5.540e-01  1.862e+00  -0.297    0.767
## Gold.price.5                  5.063e-01  1.856e+00   0.273    0.785
## Gold.price.6                 -4.526e-01  1.851e+00  -0.245    0.807
## Gold.price.7                  6.973e-01  1.817e+00   0.384    0.702
## Gold.price.8                 -2.325e-03  1.805e+00  -0.001    0.999
## Gold.price.9                 -1.831e-01  1.784e+00  -0.103    0.918
## Gold.price.10                 1.347e+00  1.206e+00   1.117    0.266
## Crude.Oil..Brent._USD.bbl.t  -9.613e+00  1.227e+01  -0.783    0.435
## Crude.Oil..Brent._USD.bbl.1   2.697e+00  1.908e+01   0.141    0.888
## Crude.Oil..Brent._USD.bbl.2  -6.970e+00  1.930e+01  -0.361    0.719
## Crude.Oil..Brent._USD.bbl.3  -3.224e+00  1.957e+01  -0.165    0.869
## Crude.Oil..Brent._USD.bbl.4  -1.204e+01  1.987e+01  -0.606    0.546
## Crude.Oil..Brent._USD.bbl.5   2.133e+00  1.997e+01   0.107    0.915
## Crude.Oil..Brent._USD.bbl.6  -3.001e+00  2.010e+01  -0.149    0.882
## Crude.Oil..Brent._USD.bbl.7  -3.793e+00  2.072e+01  -0.183    0.855
## Crude.Oil..Brent._USD.bbl.8   7.489e+00  2.086e+01   0.359    0.720
## Crude.Oil..Brent._USD.bbl.9  -8.148e+00  2.058e+01  -0.396    0.693
## Crude.Oil..Brent._USD.bbl.10  7.417e+00  1.278e+01   0.581    0.563
## Copper_USD.tonne.t            1.593e-01  1.563e-01   1.019    0.310
## Copper_USD.tonne.1            5.231e-02  2.382e-01   0.220    0.827
## Copper_USD.tonne.2            1.273e-01  2.428e-01   0.524    0.601
## Copper_USD.tonne.3            3.773e-02  2.415e-01   0.156    0.876
## Copper_USD.tonne.4            9.031e-02  2.407e-01   0.375    0.708
## Copper_USD.tonne.5            1.110e-02  2.437e-01   0.046    0.964
## Copper_USD.tonne.6            4.145e-02  2.424e-01   0.171    0.865
## Copper_USD.tonne.7            8.862e-02  2.463e-01   0.360    0.720
## Copper_USD.tonne.8           -7.441e-02  2.483e-01  -0.300    0.765
## Copper_USD.tonne.9           -1.465e-02  2.413e-01  -0.061    0.952
## Copper_USD.tonne.10           1.484e-02  1.605e-01   0.092    0.926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 703.9 on 117 degrees of freedom
## Multiple R-squared:  0.3811, Adjusted R-squared:  0.2065
## F-statistic: 2.183 on 33 and 117 DF,  p-value: 0.001225
##
## AIC and BIC values for the model:
##        AIC      BIC
## 1 2440.112 2545.716
```

```
checkresiduals(model.dlm$model)
```




```
##
##  Breusch-Godfrey test for serial correlation of order up to 37
##
## data:  Residuals
## LM test = 139.95, df = 37, p-value = 6.964e-14
```

A finite DLM is fitted using the dlm() function. Prior to fitting, the ideal lag value is found using the finiteDLMauto() function, which is made to generate multiple models with q ranging from 1 to 10 and uses AIC error to determine the suitability of each model. The results of finiteDLMauto() suggest that the ideal value for q is 10.

The summary of the finite DLM shows a low p-value of 0.001225, which suggests that the model is significant. However, a low R-squared value of 0.2065 means that the model does not explain the data well. None of the predictor variables or their lags are significant, meaning that none of the lag components are satisfactory predictors of ASX price.

The Breusch-Godfrey test for serial correlation produces a low p-value of 6.964e-14, which means that there is evidence of serial correlation in the model, making it less suitable to model ASX price.

Analysing the residuals for the finite DLM shows that the residuals are relatively large and are not normally distributed, suggesting that the model does not account for the data well and the errors in the model are not random. The ACF plot also shows peaks above the 95% confidence interval, suggesting that this is an inadequate model to describe ASX price.

ARDL Model

```
#ARDL
columns = c("p","q","AIC","BIC")
df = data.frame(matrix(nrow = 0, ncol = length(columns)))
colnames(df) = columns

for(i in 1:10){
  for(j in 1:10){
    model.ARDL = ardlDlm(formula = ASX.price ~ Gold.price + Crude.Oil..Brent._USD.bbl + Copper_USD.tonne, dat
a = ASX_data, p = i, q = j)
    new_row = data.frame(p = i, q=j, AIC=AIC(model.ARDL$model), BIC=BIC(model.ARDL$model))
    df = bind_rows(df, new_row)
  }
}

df[order(df$AIC),]
```

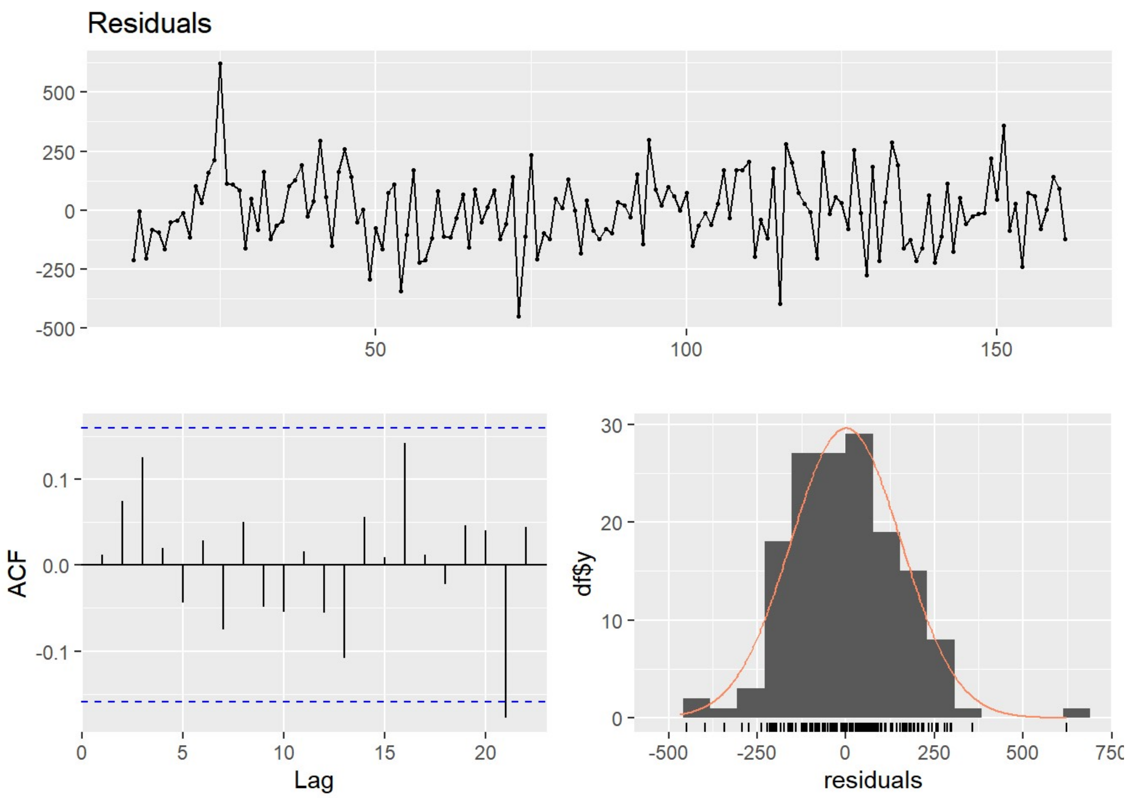
```
##       p  q      AIC      BIC
## 92  10  2 2026.553 2138.192
## 93  10  3 2027.304 2141.961
## 91  10  1 2027.601 2136.223
## 94  10  4 2027.757 2145.431
## 10   1 10 2029.308 2083.619
## 95  10  5 2029.715 2150.407
## 96  10  6 2031.704 2155.413
## 50   5 10 2032.187 2122.706
## 20   2 10 2032.850 2096.213
## 97  10  7 2033.601 2160.327
## 99  10  9 2035.383 2168.144
## 98  10  8 2035.459 2165.202
## 30   3 10 2035.704 2108.119
## 60   6 10 2036.376 2135.946
## 80   8 10 2037.029 2154.702
## 100 10 10 2037.257 2173.034
## 82   9  2 2039.257 2142.069
## 90   9 10 2039.272 2165.998
## 40   4 10 2039.470 2120.936
## 9    1  9 2039.671 2091.077
## 70   7 10 2039.867 2148.489
## 81   9  1 2040.311 2140.099
## 83   9  3 2040.431 2146.267
## 84   9  4 2041.448 2150.308
## 49   5  9 2042.439 2130.132
## 85   9  5 2043.065 2154.949
## 19   2  9 2043.305 2103.783
## 86   9  6 2045.059 2159.966
## 29   3  9 2046.031 2115.581
## 59   6  9 2046.656 2143.421
## 87   9  7 2047.009 2164.940
## 79   8  9 2047.061 2161.969
## 88   9  8 2048.906 2169.861
## 72   8  2 2049.283 2143.227
## 39   4  9 2049.757 2128.378
## 89   9  9 2049.912 2173.891
## 69   7  9 2050.140 2155.976
## 73   8  3 2050.619 2147.593
## 8    1  8 2050.875 2099.362
## 74   8  4 2051.792 2151.796
## 71   8  1 2052.054 2142.967
## 48   5  8 2052.834 2137.686
## 75   8  5 2053.371 2156.406
## 18   2  8 2054.680 2112.258
## 76   8  6 2055.319 2161.385
## 58   6  8 2056.997 2150.940
## 28   3  8 2057.111 2123.781
## 77   8  7 2057.315 2166.411
## 78   8  8 2059.286 2171.412
## 38   4  8 2060.635 2136.396
## 68   7  8 2060.662 2163.697
## 7    1  7 2061.407 2106.961
## 47   5  7 2062.884 2144.882
## 62   7  2 2064.074 2149.109
## 63   7  3 2065.037 2153.109
## 17   2  7 2065.343 2120.008
## 64   7  4 2065.548 2156.656
## 61   7  1 2066.062 2148.060
## 57   6  7 2067.038 2158.146
## 65   7  5 2067.071 2161.216
## 27   3  7 2067.461 2131.237
## 66   7  6 2068.859 2166.041
## 67   7  7 2070.765 2170.985
## 37   4  7 2070.902 2143.789
```

```
## 6    1  6 2071.964 2114.572
## 46   5  6 2073.546 2152.675
## 52   6  2 2074.419 2150.504
## 54   6  4 2074.458 2156.630
## 53   6  3 2074.498 2153.627
## 16   2  6 2075.737 2127.475
## 51   6  1 2075.916 2148.958
## 55   6  5 2076.102 2161.318
## 56   6  6 2077.643 2165.903
## 26   3  6 2077.790 2138.658
## 36   4  6 2081.189 2151.187
## 42   5  2 2082.796 2149.892
## 5    1  5 2082.841 2122.489
## 44   5  4 2082.852 2156.048
## 43   5  3 2082.928 2153.074
## 41   5  1 2084.699 2148.746
## 45   5  5 2084.746 2160.992
## 15   2  5 2086.518 2135.316
## 25   3  5 2088.589 2146.536
## 35   4  5 2092.135 2159.232
## 4    1  4 2093.199 2129.874
## 14   2  4 2096.840 2142.683
## 24   3  4 2098.840 2153.852
## 33   4  3 2102.338 2163.463
## 34   4  4 2102.448 2166.629
## 32   4  2 2103.338 2161.406
## 31   4  1 2103.367 2158.379
## 3    1  3 2104.541 2138.230
## 13   2  3 2108.425 2151.301
## 23   3  3 2111.997 2164.061
## 21   3  1 2112.595 2158.534
## 22   3  2 2113.144 2162.145
## 2    1  2 2119.288 2149.977
## 11   2  1 2120.906 2157.733
## 12   2  2 2121.905 2161.801
## 1    1  1 2131.866 2159.542
```

```
model.ARDL <- ardlDlm(formula = ASX.price ~ Gold.price + Crude.Oil..Brent._USD.bbl + Copper_USD.tonne, data =
ASX_data, p = 10, q = 2)
summary(model.ARDL)
```

```
##
## Time series regression with "ts" data:
## Start = 11, End = 161
##
## Call:
## dynlm(formula = as.formula(model.text), data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -449.76 -111.17   -5.21   95.23  622.81
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  215.97367  109.35514   1.975  0.05067 .
## Gold.price.t                  -1.11706    0.32100  -3.480  0.00071 ***
## Gold.price.1                   0.76491    0.46241   1.654  0.10082
## Gold.price.2                   0.51842    0.47779   1.085  0.28017
## Gold.price.3                  -0.28183    0.47680  -0.591  0.55563
## Gold.price.4                  -0.60517    0.47144  -1.284  0.20184
## Gold.price.5                   1.11018    0.47311   2.347  0.02066 *
## Gold.price.6                  -0.90747    0.48144  -1.885  0.06197 .
## Gold.price.7                   0.95674    0.46585   2.054  0.04227 *
## Gold.price.8                  -0.90465    0.46281  -1.955  0.05305 .
## Gold.price.9                   0.33776    0.45731   0.739  0.46166
## Gold.price.10                  0.13066    0.30780   0.425  0.67199
## Crude.Oil..Brent._USD.bbl.t    3.02290    3.27088   0.924  0.35732
## Crude.Oil..Brent._USD.bbl.1   -3.76335    4.94085  -0.762  0.44781
## Crude.Oil..Brent._USD.bbl.2   -4.17895    4.88776  -0.855  0.39434
## Crude.Oil..Brent._USD.bbl.3    1.71802    4.97967   0.345  0.73072
## Crude.Oil..Brent._USD.bbl.4   -8.08519    5.05130  -1.601  0.11221
## Crude.Oil..Brent._USD.bbl.5   12.32018    5.14541   2.394  0.01826 *
## Crude.Oil..Brent._USD.bbl.6   -3.22037    5.25780  -0.612  0.54142
## Crude.Oil..Brent._USD.bbl.7   -3.91734    5.25324  -0.746  0.45737
## Crude.Oil..Brent._USD.bbl.8   10.83063    5.29370   2.046  0.04304 *
## Crude.Oil..Brent._USD.bbl.9   -9.25451    5.26686  -1.757  0.08156 .
## Crude.Oil..Brent._USD.bbl.10   4.15391    3.24978   1.278  0.20375
## Copper_USD.tonne.t             0.06648    0.03975   1.672  0.09716 .
## Copper_USD.tonne.1            -0.02098    0.06087  -0.345  0.73095
## Copper_USD.tonne.2             0.07309    0.06153   1.188  0.23729
## Copper_USD.tonne.3            -0.08342    0.06134  -1.360  0.17651
## Copper_USD.tonne.4             0.05219    0.06120   0.853  0.39557
## Copper_USD.tonne.5            -0.07186    0.06180  -1.163  0.24738
## Copper_USD.tonne.6             0.04153    0.06163   0.674  0.50175
## Copper_USD.tonne.7             0.01515    0.06241   0.243  0.80859
## Copper_USD.tonne.8            -0.09426    0.06289  -1.499  0.13667
## Copper_USD.tonne.9            -0.01957    0.06203  -0.315  0.75296
## Copper_USD.tonne.10            0.04298    0.04079   1.054  0.29415
## ASX.price.1                    0.81879    0.09538   8.585 5.04e-14 ***
## ASX.price.2                    0.14682    0.09587   1.531  0.12841
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 178.2 on 115 degrees of freedom
## Multiple R-squared:  0.961,  Adjusted R-squared:  0.9492
## F-statistic: 81.04 on 35 and 115 DF,  p-value: < 2.2e-16
```

```
checkresiduals(model.ARDL$model)
```




```
##
##  Breusch-Godfrey test for serial correlation of order up to 39
##
## data:  Residuals
## LM test = 47.338, df = 39, p-value = 0.1689
```

An ARDL model is fitted using the ardlDlm() function. Prior to fitting, the ideal lag value for the independent series (p) and the ideal lag value for the dependent series (q) are found by looping through multiple iterations, creating a new model with unique p and q values each time. The AIC and BIC errors are calculated for each model and used to determine the suitability of each model. The results of this procedure suggest that the ideal value for p is 10 and the ideal value for q is 2.

The summary of the ARDL model shows a very low p-value of 2.2e-16, which suggests that the model is significant. A high R-squared value of 0.9492 makes it marginally better than the Koyck model and reveals that the model explains most of the data well.

The Gold price at t, as well as its 5th and 7th lags are significant. The 5th and 8th lags of crude oil price are also significant. Furthermore, none of the Copper price lags are significant.

The Breusch-Godfrey test produces a high p-value of 0.1689, failing to reject the null hypothesis of independently distributed data. Therefore, it is possible to conclude that there is no serial correlation in the model.

Analysing the residuals for the ARDL model shows that the residuals are relatively small compared to the even the Koyck model and are roughly normally distributed, suggesting that the model accounts for the data well and the errors in the model are random. The ACF plot shows that the peaks are mostly within the 95% confidence interval, suggesting that this is an adequate model to describe ASX price.

Conclusion

The methods explored in this report explored the characteristics of the ASX price index time series data as well as its predictor variables, which included Gold price, Crude oil price, and Copper price. It was discovered that the ASX price index was non-stationary in nature, and multiple attempts to transform the series using log transformations, differencing, and seasonal adjustments failed to effectively make the series stationary. Further exploration of the seasonal components of the ASX price index revealed that the seasonal component of the series is negligible compared to its trend and remainder components. Finally, various Distributed Lag Models, including the polynomial DLM, Koyck DLM, finite DLM, and ARDL model, were fitted to the time series and their summaries and residuals were interpreted. Analysing the diagnostics from each model revealed that, for the ASX price index series, the ARDL model was the best overall due to its goodness of fit and the lack of serial correlation in the model. The Koyck model was a close second for the same reasons but had a slightly less significant p-value and slightly higher residuals compared to the ARDL model. Both the polynomial and finite DLM models proved inadequate to describe the series due to low goodness of fit, the existence of serial correlation in the model, high residuals, and non-normal residual distributions.