Multiple Linear Regression model with ARIMA errors applied to predict Suncor Energy Inc.'s stock price

Dung Le

University of Victoria, Victoria, British Columbia, Canada

1. Introduction:

Multiple linear regression (MLR) is a popular statistical method, which uses several independent (explanatory) variables to predict the outcome of a dependent (response) variable. MLR can be expressed as below [Shumway, R. H., & Stoffer, D. S., 2016, p. 145-147]:

$$y_t = \sum_{j=1}^{r} \beta_j z_{tj} + x_t$$

where: y_t is a dependent variable (response variable)

z_{tj} is an independent variable (explanatory variable)

β_j is regression coefficient of regressor z_j

x_t is a random error or noise process consisting of independent and identically distributed normal variables with mean zero and variance σ^2.

In the time series regression model, error x_t is normally autocorrelated. The violation of the independent error assumption can make the predicted values inaccurate. The autocorrelated issue can be resolved by using Autoregressive Integrated Moving Average (ARIMA) transformation. For example, ARIMA representation for x_t is AR(p):

$$\phi(B)y_t = \sum_{j=1}^{r} \beta_j \phi(B) z_{tj} + \phi(B)x_t$$

$$\phi(B)x_t = w_t$$

where $\phi(B) = 1 - \phi_1 B - \cdots - \phi_p B^p$ is linear transformation, when applied to the error process, produces the white noise w_t [Shumway, R. H., & Stoffer, D. S., 2016, p. 145-147].

In this paper, we will apply Multiple Linear Regression with ARIMA errors to predict the value of Suncor Energy's stock price (response variable) from WTI Crude Oil price and S&P/TSX Composite Index (explanatory variables). Suncor Energy is a leading Canadian integrated energy company specializing in producing synthetic crude oil and related products from oil sands. Suncor Energy is listed on Toronto Stock Exchange as SU.TO. This research uses publicly available data, sourcing from data files provided by Yahoo Finance. It includes daily price of all trading days from December 2014 to December 2019.

The research starts with having the first overview of the data and splitting three datasets into train/test subsets. Then, it will fit the train data into MLR model and check for multicollinearity problem. Since the regression model's residuals are autocorrelated, the research continues with applying ARMA model to make the error stationary. Finally, we will use the final model to forecast Suncor's stock price upon the test period, then compare the forecast result with the real data.

2. Initial Data Examination:

Figure 1 represents data of Suncor's stock price, WTI crude oil price, and S&P/TSX Composite index from December 2014 to December 2019. It is easily to spot that three graphs in Figure 1 have similar movements; especially during the bearish market periods: January 2016 (Oil price crash) and December 2018 (Cryptocurrency crash).

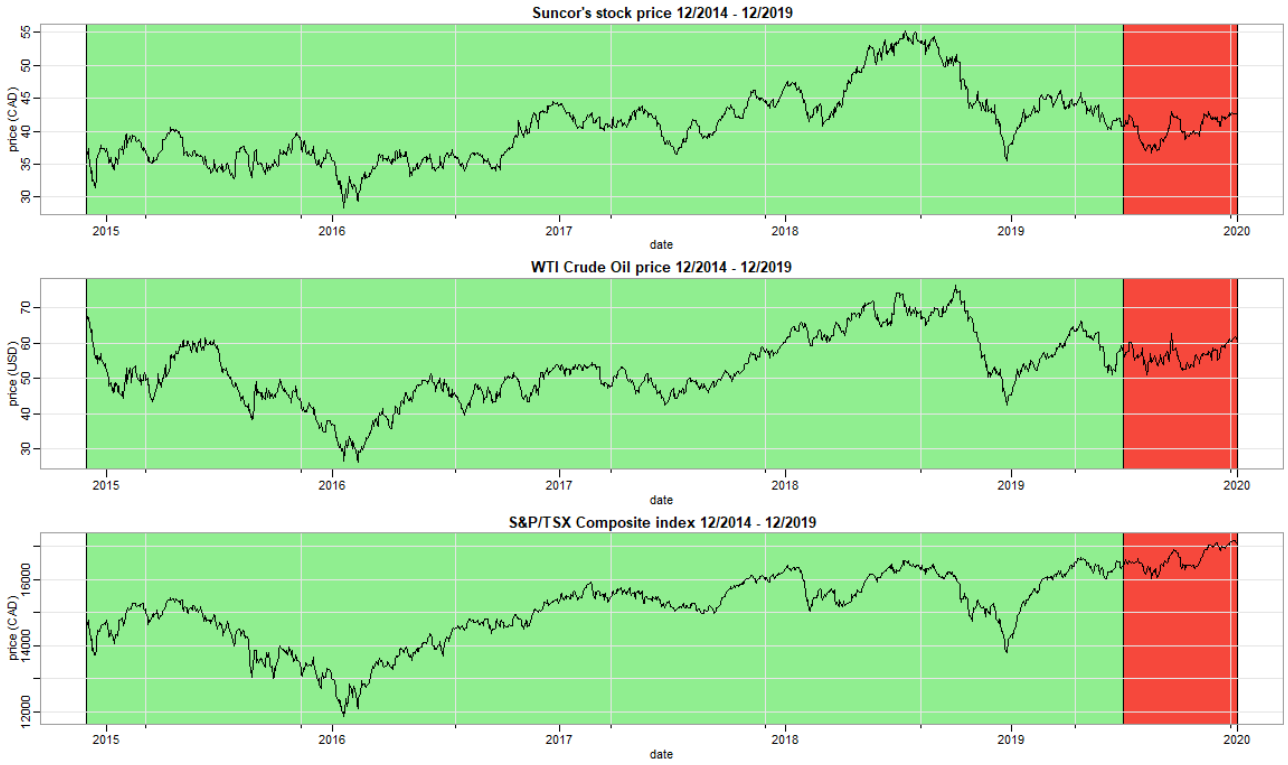

Figure 1: Suncor, WTI Crude Oil, S&P/TSX index trends (12/2014 - 12/2019)

As expected, in the pairwise scatterplots (Figure 2), there are clear positive relationships between Suncor's stock price and S&P/TSX Composite index (corr = 0.7207), and between Suncor's stock price and WTI Crude Oil price (corr = 0.7985). The model might have multicollinearity if both WTI Crude Oil price and S&P/TSX Composite index are included in the regression model given that they are highly correlated (corr = 0.7601)

Each data set is split into two subsets:

- Train data: 12/01/2014 to 07/01/2019 (green area in Figure 1)
- Test data: 07/01/2019 to 12/31/2019 (red area in Figure 1)

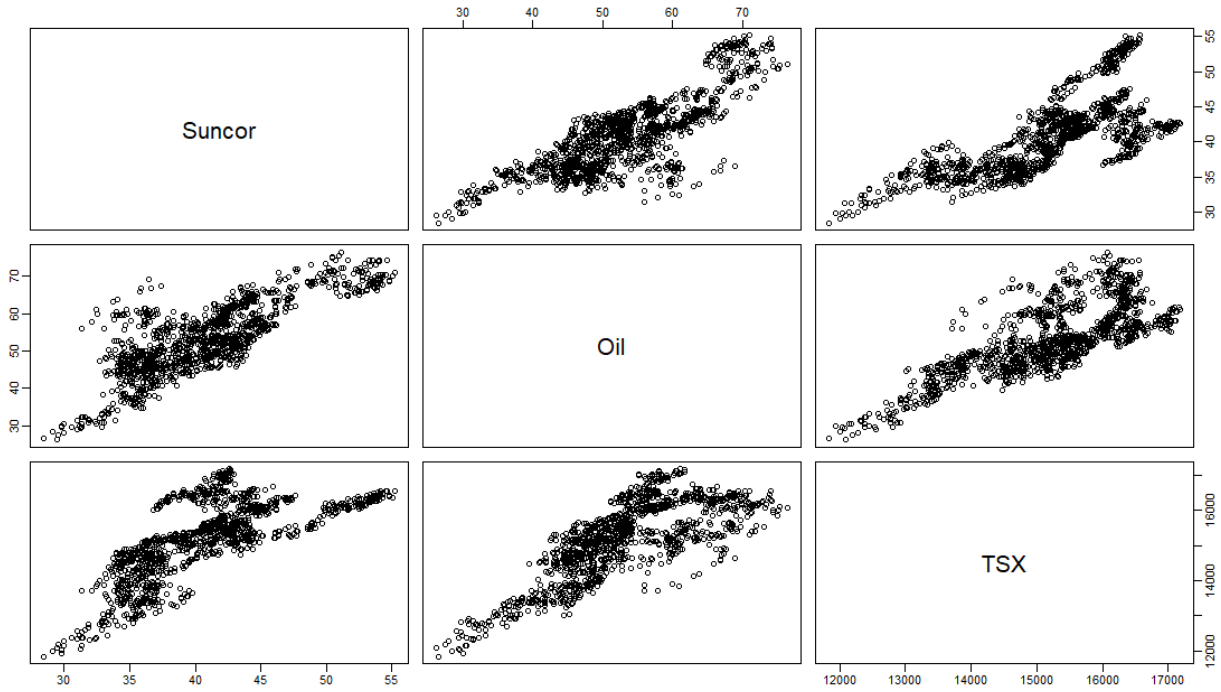

Figure 2: Pairwise scatterplots of Suncor's stock price, WTI Crude oil price, and S&P/TSX composite index

3. Fitted Multiple Linear Regression model:

a) Fitted multiple linear regression model:

After fitting some multiple linear regression models, the best model is:

$$\text{Suncor} = -36.710_{(5.807)} + 1.862_{(0.351)}*\text{Oil} - 0.050_{(0.007)}*\text{Oil}^2 + 4.154*10^{-4}{}_{(0.000)}*\text{Oil}^3 + 0.004_{(0.000)}*\text{TSX} + \varepsilon$$

- Residuals:

Min	1Q	Median	3Q	Max
-7.5340	-1.5009	0.0197	1.6591	5.8662

- Coefficients:

	Estimate	SE	t-value	Pr(>|t|)
(Intercept)	-3.671e+01	5.807e+00	-6.321	3.71e-10
Oil	1.862e+00	3.506e-01	5.309	1.32e-07
Oil^2	-5.028e-02	6.941e-03	-7.244	7.98e-13
Oil^3	4.154e-04	4.481e-05	9.270	< 2e-16
TSX	3.774e-03	1.243e-04	30.372	< 2e-16

- Residual standard error: 2.312 on 1144 degrees of freedom
- Multiple R-squared: 0.8214, Adjusted R-squared: 0.8207
- F-statistic: 1315 on 4 and 1144 DF, p-value: < 2.2e-16
- AIC = 5193.893

With adjusted R-squared equals to 0.8207, 82.07% of the data fit the regression model.

b) Multicollinearity diagnostics:

As discussed in the previous section, adding both WTI crude oil price and S&P/TSX Composite index to the regression model can introduce multicollinearity problem. Therefore, we need to run the Variance Inflation Factors (VIF) test to verify if the regression model (a) has the multicollinearity problem. According to Frost, J. [2017] in the article “Multicollinearity in Regression Analysis: Problems, Detection, and Solutions”, VIF identifies correlation between independent variables and the strength of that correlation. With any variables having VIF greater than fourteen, they need to be either excluded or transformed. In Table 1, VIF of S&P/TSX index is 3.387524, so we can include both WTI Crude Oil price and S&P/TSX index in the model. There are

structural multicollinearity in the model among Oil variables, which can be fixed by standardizing each variable (subtract each variable by its mean)

Oil	Oil^2	Oil^3	TSX
2481.433553	11153.630689	3248.820306	3.387524

Table 1: Variance Inflation Factors (VIF) test for regression model before standardizing

Oil	Oil^2	Oil^3	TSX
6.195372	1.355803	3.702495	3.387524

Table 2: Variance Inflation Factors (VIF) test for regression model after standardizing

After standardizing each WTI Crude Oil price variable, the structural multicollinearity problem is resolved (Table 2).

4. Transformation

In the Figure 3 – residuals plot, it is clear that the mean of residuals is not zero and variance is not constant. Because ACF decreases very slowly, residuals are correlated. As a result, residuals are not stationary, so we need to do a transformation.

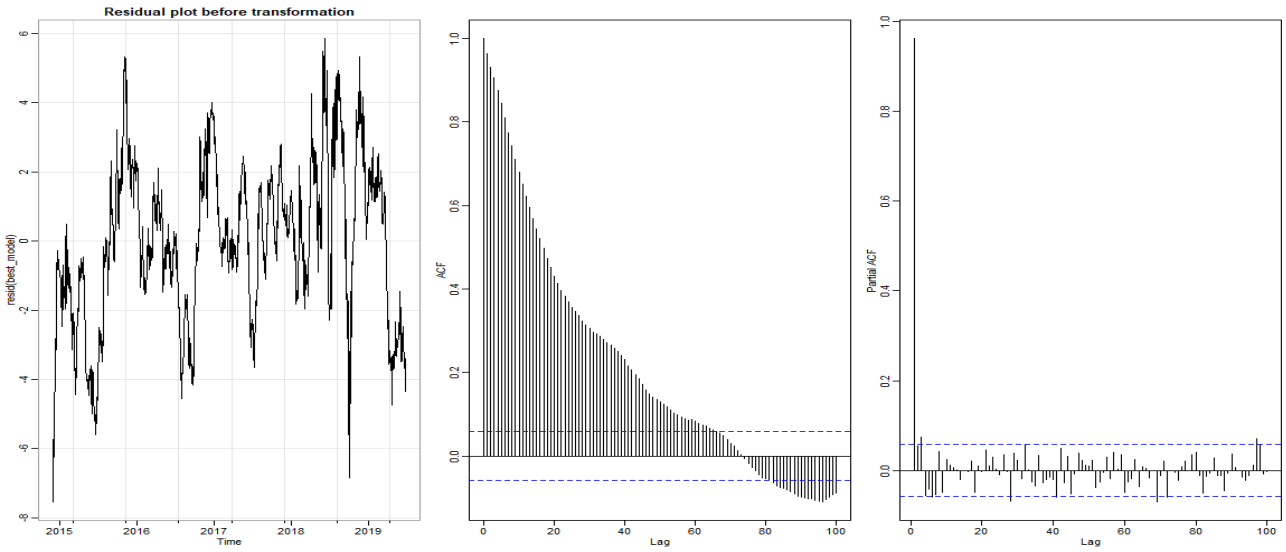

Figure 3: Residuals plot before transformation

After the first difference transformation, residuals seem to be stationary (Figure 4). Their mean roughly equals zero, and ACF decreases exponentially fast. However, there are some huge spikes in the time series plot in December 2018, which are related to the cryptocurrency market crash in that year. The crash blew away 19.73% of S&P 500 index and 18.78% of DJIA index. Other than that outlier, variance of residuals is approximately constant.

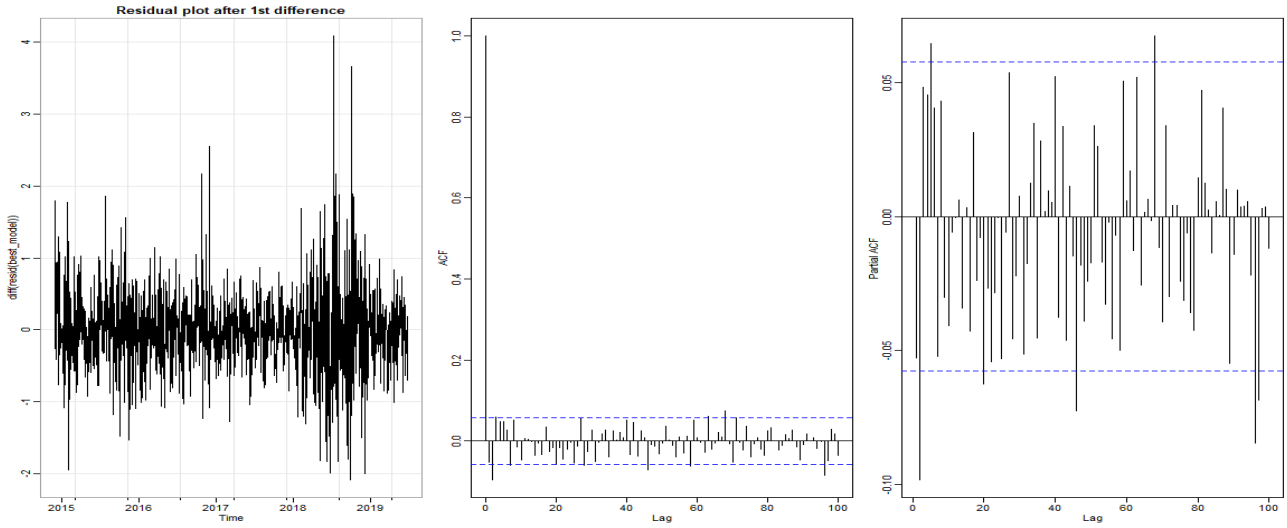

Figure 4: Residuals plot after transformation

From Figure 4, ACF cuts off after lag 2, and PACF cuts off after lag 2, ARIMA(2,2) could be a good ARMA model for this data.

5. Fitted ARIMA model with diagnostics

a) Fitted ARIMA model:

In the fitted ARIMA(2,1,4) model, Oil^2 and Oil^3 variables are dropped from the regression model in part 3, because they are insignificant. Table 3 contains the estimation, standard error, and p-value of each coefficient in ARMA and regression model.

	Estimate	SE	t-value	p-value
AR1	-0.4063	0.2534	-1.6034	0.1091
AR2	0.5760	0.2474	2.3286	0.0201
MA1	0.3369	0.2533	1.3298	0.1838
MA2	-0.7045	0.2281	-3.0882	0.0021
MA3	0.0675	0.0424	1.5919	0.1117
MA4	0.1128	0.0317	3.5655	0.0004
Oil	0.1452	0.0125	11.6638	0.0000
TSX	0.0034	0.0001	24.0205	0.0000

Table 3: ARIMA(2,1,4) with regressors summary

- AIC = 1.182117
- AICc = 1.182227
- BIC = 1.221674

b) Residuals diagnostics:

From Figure 5 – residuals diagnostics plot for above ARIMA model:

- Mean is approximately 0
- There are some spikes in the plot, but other than that, variance seems to be constant.
- ACF decreases very fast, and all lines stay within two dashed lines
- Ljung-box statistics: all points are above the dashed line (p-value > 0.05)

  ➔ Residuals are uncorrelated
- Normal Q-Q plot: there are some outliers in both left and right tails. Residuals are not normally distributed.

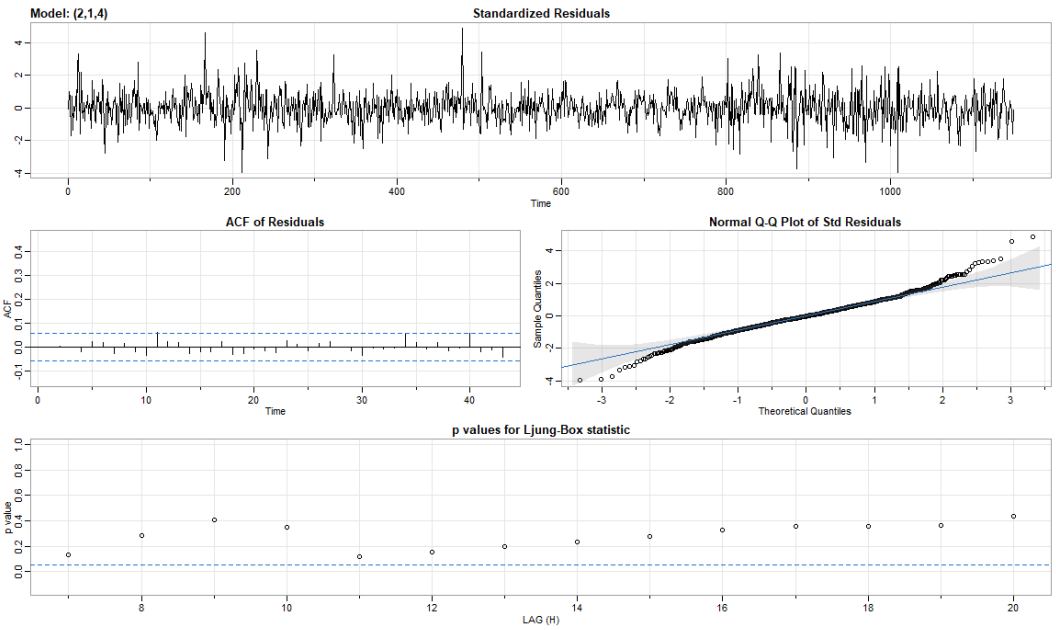

Figure 5: ARIMA Residuals diagnostics

6. Prediction

From the ARIMA(2,1,4) model in previous section, we run the prediction on the test data subsets (July 2019 to December 2019) – Figure 6.

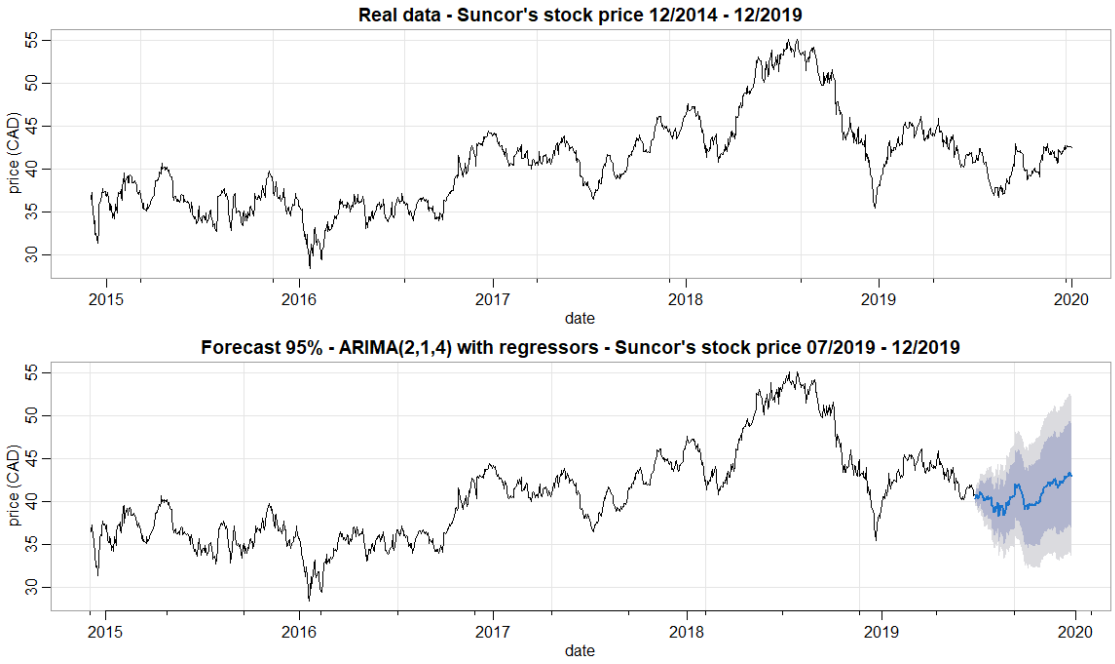

Figure 6: Suncor's stock price prediction using ARIMA(2,1,4) model (with 95% interval)

7. Conclusion

In Figure 6, the forecast result and the real data have very similar pattern, and all values lies within 95% interval prediction. Thus, we can use WTI Crude Oil price and S&P/TSX Composite index to predict Suncor Energy's stock price.

There is one problem in this research is residuals of the final ARIMA model are not normally distributed. As a result, we will need to apply some other transformations to improve the model.

8. List of references

Frost, J. (2017). Multicollinearity in Regression Analysis: Problems, Detection, and Solutions. Statistics By Jim. Retrieved April 12, 2021, from https://statisticsbyjim.com/regression/multicollinearity-in-regression-analysis/.

Shumway, R. H., & Stoffer, D. S. (2016). Time Series Analysis and Its Applications. Springer. doi:10.1007/0-387-36276-2

Yahoo Finance. Crude Oil May 21 (CL=F). Retrieved March 25, 2021, from https://ca.finance.yahoo.com/quote/CL%3DF?p=CL%3DF

Yahoo Finance. Suncor Energy Inc. (SU.TO). Retrieved March 25, 2021, from https://ca.finance.yahoo.com/quote/su.to/

Yahoo Finance. S&P/TSX Composite index (^GSPTSE). Retrieved March 25, 2021, from https://ca.finance.yahoo.com/quote/%5EGSPTSE?p=%5EGSPTSE

9. Appendix: R code:

```
library(xts)
library(astsa)
library(zoo)
library(forecast)
library(car)



#=====================================================================
=======================================
# Support Function:
# a) Time conversion function:
date_converter <- function(d) {
  temp_d <- as.Date(paste0(d,"-01-01"))
  return(temp_d)
}



#=====================================================================
=======================================
# Step 1: Loading data files:
suncor_file <- read.csv("./Suncor-2014-2019.csv",na.strings = "")
WTI_oil_file <- read.csv("./Crude_Oil_WTI_2014-2019.csv",na.strings = "")
SP_TSX_file <- read.csv("./SPTSX_2014-2019.csv",na.strings = "")
```

```
#=================================================================
=====================================
# Step 2: Convert data into time series data
# Note: Cannot use ts() function because it cannot convert original data to time series
#       daily data excluding weekends and holidays correctly
suncor <- xts(suncor_file[,2], order.by=date_converter(suncor_file[,1]))
WTI_oil <- xts(WTI_oil_file[,2], order.by=date_converter(WTI_oil_file[,1]))
SP_TSX <- xts(SP_TSX_file[,2], order.by=date_converter(SP_TSX_file[,1]))


#=================================================================
=====================================
# Step 3: Initial examination of data
# Time series plot overview
time_check <- c(date_converter("2016-01-24"),date_converter("2018-12-24"))
par(mfrow=c(3,1))
tsplot(time(suncor),suncor,main="Suncor's stock price 12/2014 - 12/2019",ylab="price
(CAD)",xlab="date")
abline(v=time_check,col = "blue")
tsplot(time(WTI_oil),WTI_oil,main="WTI Crude Oil price 12/2014 - 12/2019",ylab="price
(USD)",xlab="date")
abline(v=time_check,col = "blue")
```

```
tsplot(time(SP_TSX),SP_TSX,main="S&P/TSX Composite index 12/2014 -
12/2019",ylab="price (CAD)",xlab="date")
abline(v=time_check,col = "blue")

#-----------------------------
# Check data correlation
# - There is a positive relationship between TSX index and Suncor stock price
# - There is a positive relationship between Suncor stock price and WTI oil price
# - TSX index seems to be correlated with Oil price. We will need to check for
Multicollinearity after fitting model.
cor(suncor,SP_TSX)
cor(suncor,WTI_oil)
cor(WTI_oil,SP_TSX)
pairs(cbind(Suncor=suncor_file[,2], Oil=WTI_oil_file[,2], TSX=SP_TSX_file[,2]))


#=================================================================
======================================
# Step 4: Split dataset into interested (train/test) time interval
# Set up train/test intervals
suncor_train <- window(suncor, start=date_converter("2014-12-01"),
end=date_converter("2019-07-01"))
oil_train <- window(WTI_oil, start=date_converter("2014-12-01"),
end=date_converter("2019-07-01"))
```

```
TSX_train <- window(SP_TSX, start=date_converter("2014-12-01"),
end=date_converter("2019-07-01"))

suncor_test <- window(suncor, start=date_converter("2019-07-01"),
end=date_converter("2019-12-31"))
oil_test <- window(WTI_oil, start=date_converter("2019-07-01"),
end=date_converter("2019-12-31"))
TSX_test <- window(SP_TSX, start=date_converter("2019-07-01"),
end=date_converter("2019-12-31"))

# ------------------------------
# Visualize train/test intervals
par(mfrow=c(3,1))
tsplot(time(suncor),suncor,
      panel.first = c(rect(date_converter("2014-12-01"), -1e6,
                           date_converter("2019-07-01"), 1e6, col='lightgreen', border=TRUE),
                      rect(date_converter("2019-07-01"), -1e6,
                           date_converter("2019-12-31"), 1e6, col=2, border=TRUE)),
      main="Suncor's stock price 12/2014 - 12/2019",ylab="price (CAD)",xlab="date")
tsplot(time(WTI_oil),WTI_oil,
      panel.first = c(rect(date_converter("2014-12-01"), -1e6,
                           date_converter("2019-07-01"), 1e6, col='lightgreen', border=TRUE),
                      rect(date_converter("2019-07-01"), -1e6,
                           date_converter("2019-12-31"), 1e6, col=2, border=TRUE)),
```

```
      main="WTI Crude Oil price 12/2014 - 12/2019",ylab="price (USD)",xlab="date")
tsplot(time(SP_TSX),SP_TSX,
      panel.first = c(rect(date_converter("2014-12-01"), -1e6,
                           date_converter("2019-07-01"), 1e6, col='lightgreen', border=TRUE),
                      rect(date_converter("2019-07-01"), -1e6,
                           date_converter("2019-12-31"), 1e6, col=2, border=TRUE)),
      main="S&P/TSX Composite index 12/2014 - 12/2019",ylab="price (CAD)",xlab="date")



#=====================================================================
=======================================
# Step 4: Fit multiple linear regression model
oil_train_2 <- oil_train^2
oil_train_3 <- oil_train^3
TSX_train_2 <- TSX_train^2
TSX_train_3 <- TSX_train^3
oil_TSX <- oil_train*TSX_train


model <- lm(suncor_train ~ oil_train + TSX_train, na.action=NULL)
model_1 <- lm(suncor_train ~ oil_train + oil_train_2
                                    + TSX_train, na.action=NULL)
model_2 <- lm(suncor_train ~ oil_train + TSX_train
                                    + TSX_train_2, na.action=NULL)
model_3 <- lm(suncor_train ~ oil_train + oil_train_2
```

```
                                    + TSX_train + TSX_train_2, na.action=NULL)
model_4 <- lm(suncor_train ~ oil_train + oil_train_2
                                    + oil_train_3 + TSX_train, na.action=NULL)
model_5 <- lm(suncor_train ~ oil_train + oil_TSX + TSX_train, na.action=NULL)

# Variance Inflation Factor is small << 10
# -> we can add TSX index to model with no multicollinearity issue
car::vif(model)

# Model selection:
# > After checking model summary, anova table, AIC:
#   >> Best model = model_4:
#      suncor_train ~ oil_train + oil_train^2 + oil_train^3 + TSX_train
summary(model_4)       # regression results
summary(aov(model_4))  # ANOVA table
AIC(model,model_1,model_2,model_3,model_4,model_5)

# > Variance Inflation Factor of Oil variables are huge >> 14 -> structural multicollinearity
issue
#   >> We can try standardizing Oil variables by subtracting the mean
car::vif(model_4)
oil_train_S <- oil_train - mean(oil_train)
oil_train_2S <- oil_train_S^2
oil_train_3S <- oil_train_S^3
```

```
model_4S <- lm(suncor_train ~ oil_train_S + oil_train_2S
                    + oil_train_3S + TSX_train, na.action=NULL)


# > After standardization, multicollinearity issue is fixed
summary(model_4S)       # regression results
summary(aov(model_4S))  # ANOVA table
car::vif(model_4S)


# > Best Model:
best_model = model_4S
fitted <- suncor_train - resid(best_model)
par(mfrow=c(1,1))
plot.zoo(cbind(suncor_train, fitted),
        plot.type = "single",
        col = c("black", "blue"),lty = c(1,2),
        main="Fitted regression model of Suncor's stock price 2015 - 2019",
        ylab="price (CAD)",xlab="date",xy.label=TRUE)



#=================================================================
=====================================
# Step 5: Verify/Transform stationary data
# - There is clear trend in time series plot, and residuals are correlated
#   > We need to do a transformation
```

```
#   > After the first difference transformation, residuals are uncorrelated
#   > ACF cuts off after lag 2, and PACF cuts off after lag 2.
#     >> we can try model ARIMA(2,2): p=2, q=2
par(mfrow=c(1,3))
tsplot(time(resid(best_model)),resid(best_model),main="Residual plot before
transformation")
acf(resid(best_model),100)
pacf(resid(best_model),100)
tsplot(time(resid(best_model)),diff(resid(best_model)),main="Residual plot after 1st
difference")
acf(diff(resid(best_model)),100,na.action=na.pass)
pacf(diff(resid(best_model)),100,na.action=na.pass)



#================================================================
=======================================
# Step 6: Adding ARIMA model to Multiple Linear Regression model
# - Model selection:
#   > Drop two variables oil_train_S^2 and oil_train_S^3, because they are insignificant in
new ARIMA model
#   > After checking p-value of coefficients, comparing residuals plot and AIC:
#     >> Best model = arima(suncor_train,2,1,4,xreg=cbind(oil_train_S,TSX_train))
#   > In all models, residuals are not normally distributed.
oil_test_S <- oil_test - mean(oil_test)
```

```
oil_test_2S <- oil_test_S^2

oil_test_3S <- oil_test_S^3


#sarima(suncor_train,2,1,2,xreg=cbind(oil_train_S,TSX_train))

sarima(suncor_train,2,1,4,xreg=cbind(oil_train_S,TSX_train))

suncor_model <- Arima(suncor_train,order=c(2,1,4),xreg=cbind(oil_train_S,TSX_train))

suncor_model

AIC(suncor_model)


#=====================================================================

=======================================

# Step 7: Prediction using above model (95% interval)

par(mfrow=c(2,1))

tsplot(time(suncor),suncor,main="Real data - Suncor's stock price 12/2014 - 12/2019",

      ylab="price (CAD)",xlab="date")

suncor_forecast <- forecast(suncor_model,xreg=cbind(oil_test_S,TSX_test))

tsplot(suncor_forecast,xaxt = 'n',xlab='date',ylab="price (CAD)",

      xm.grid=FALSE,main="Forecast 95% - ARIMA(2,1,4) with regressors - Suncor's stock

price 07/2019 - 12/2019")

axis(1, at=seq(20, 1325, by=252), labels = seq(2015, 2020, by=1))
```