Justin Sogoni - Final Project - Spring 2019 - Big Data and Economic Forecasting

Saturday, April 27, 2019 8:17 PM

Q1: Histogram and Density (10 points)

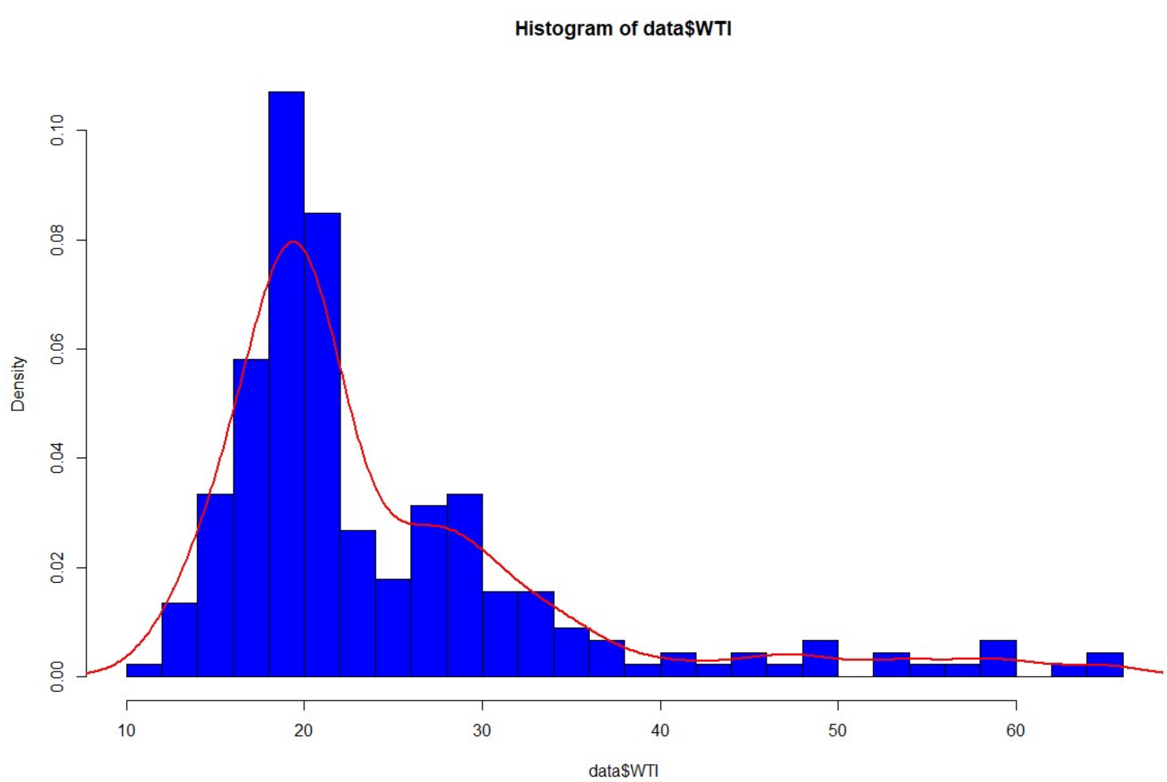
Draw a figure which contains both the histogram (set bin number to be 20) and the smoothed density of WTI.

In Assignment 2, you have done similar task. The answer code in that case was

```
# If you want to have histogram and density on the same figure
hist(return, breaks=100, freq=FALSE, col='blue')
lines(density(return), col="red", lwd=2)
```

Make changes accordingly in this case.

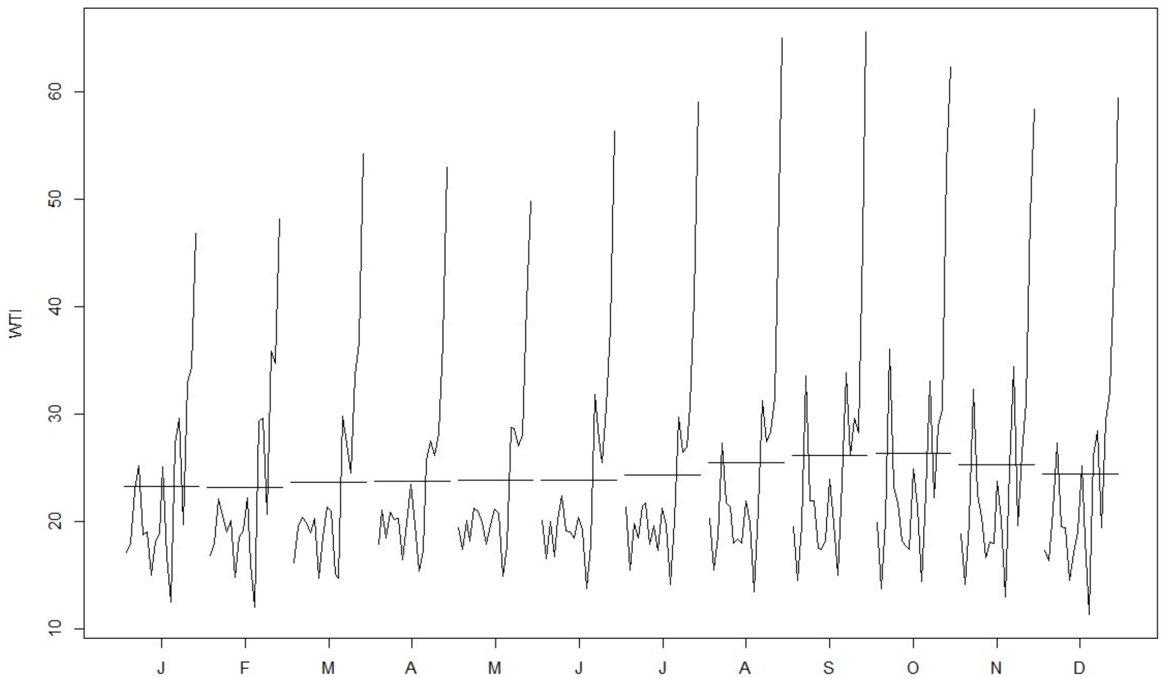
```
setwd("C:/Users/JSogoni/Google Drive/Rutgers/Big Data & Economic Forecasting/Final Project")
data <- read.csv("Oil.csv")
library(fpp)

#Q1
hist (data$WTI, breaks=20, freq=FALSE, col="blue")
lines (density(data$WTI) , col="red", lwd=2)
```

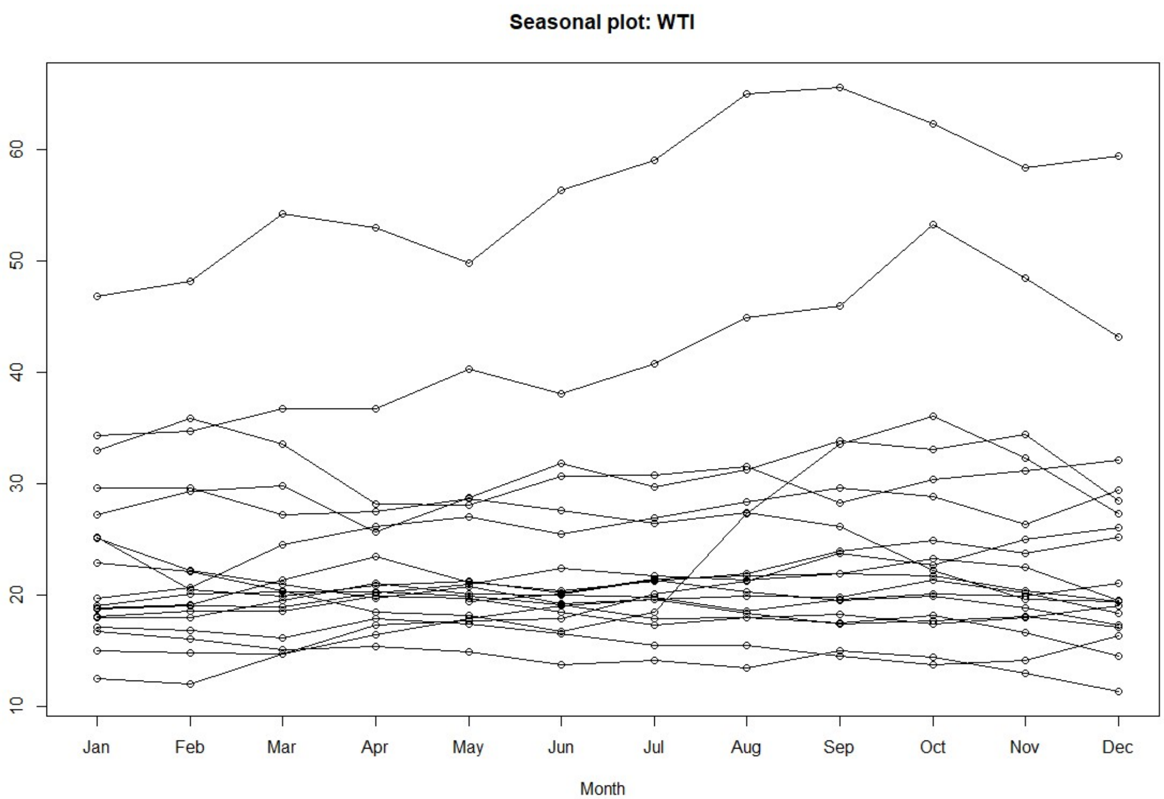


Q2: Check Seasonality (10 points)

Now turn the WTI column in `data` into a time series object: `WTI <- ts(data$WTI, frequency=12, start=c(1987,5))`. Draw a monthplot and a seasonplot of WTI. Do you find a strong seasonal pattern?

```
#Q2
WTI <- ts(data$WTI,frequency=12, start=c(1987,5))
monthplot(WTI)
seasonplot(WTI)
```





Based on the graphs, there does not appear to be a strong seasonal pattern.

Q3: Check Stationarity (30 points)

Split the time series WTI into training sample and test sample in the following way:

```
t0 = c(1987,5)
t1 = c(2004,12)
t2 = c(2005,1)
WTI_train <- window(WTI, start=t0, end=t1)
WTI_test <- window(WTI, start=t2)
```

Now perform the ADF test on the training set with different options: none, drift and trend.

```
WTI_ADF_none <- ur.df(WTI_train, type="none", lags=12, selectlags = "AIC")
summary(WTI_ADF_none)
WTI_ADF_drift <- ur.df(WTI_train, type="drift", lags=12, selectlags =
"AIC")
summary(WTI_ADF_drift)
WTI_ADF_trend <- ur.df(WTI_train, type="trend", lags=12, selectlags =
"AIC")
summary(WTI_ADF_trend)
```

Q3.1. (10 points) Discuss whether you reject the null hypothesis in each case. Is WTI a stationary process?

Q3.2. (5 points) In the output from `summary(WTI_ADF_drift)`, you will see the following in the end

```
Critical values for test statistics:
      1pct  5pct 10pct
tau2 -3.46 -2.88 -2.57
phi1  6.52  4.63  3.81
```

The critical values in the row of "tau2" is for the null hypothesis $\rho = 0$ (hence unit root). Those in the row of "phi1" is for the joint null $\rho = \beta_0 = 0$ (unit root and no drift). See slide 31 in Lecture 13 for notation. Compare the second statistic you get with the critical values in the row of "phi1", do you reject this joint null $\rho = \beta_0 = 0$? Based on your answer to this question, which specification do you think is more appropriate for the ADF test, with or without constant(drift)? Or put it in other words, do you need to include constant(drift) in the ADF test?

Q3.3. (5 points) In the output from `summary(WTI_ADF_trend)`, you will see the following in the end

```
Critical values for test statistics:
      1pct  5pct 10pct
tau3 -3.99 -3.43 -3.13
phi2  6.22  4.75  4.07
phi3  8.43  6.49  5.47
```

The critical values in the row of "tau2" is for the null hypothesis $\rho = 0$ (hence unit root). Those in the row of "phi2" is for the joint null $\rho = \beta_0 = 0$ (unit root and no drift). Those in the row of "phi3" is for the joint null $\rho = \beta_0 = \delta = 0$ (unit root, no drift and no trend). Discuss whether you need to include constant(drift) and trend in the ADF test?

Q3.4. (10 points) Apply the ADF test on the first difference sequence of WTI: `DWTI <- diff(WTI_train)`. Again, set largest lag order to be 12 and use AIC to automatically determine the lag order. Given your answers to Q3.2 and Q3.3, what specification you think is the best for the first difference sequence? Is this first difference sequence stationary? What is the integrated order of WTI?

Q3.2 and Q3.3 are essentially logic questions.

```
library(urca)
t0 = c(1987,5)
t1 = c(2004,12)
```

```
t2 = c(2005,1)
WTI_train <- window(WTI, start=t0, end=t1)
WTI_test <- window(WTI, start=t2)
WTI_ADF_none <- ur.df(WTI_train, type="none", lags=12, selectlags = "AIC")
summary(WTI_ADF_none)
```

```
Value of test-statistic is: 0.8438

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62
```

3.1) To check for stationarity, we must interpret the results from the unit root test. If the time series contains a unit root, it is not stationary. We see that the test statistic = 0.8438 > -2.58, -1.95 and -1.62. This indicates that we cannot reject the null hypothesis at any level of significance, indicating that the time series has a unit-root and is not stationary.

```
WTI_ADF_drift <- ur.df(WTI_train, type="drift", lags=12, selectlags = "AIC")
summary(WTI_ADF_drift)
```

```
Value of test-statistic is: -0.6627 0.7752

Critical values for test statistics:
      1pct  5pct 10pct
tau2 -3.46 -2.88 -2.57
phi1  6.52  4.63  3.81
```

```
WTI_ADF_trend <- ur.df(WTI_train, type="trend", lags=12, selectlags = "AIC")
summary(WTI_ADF_trend)
```

3.2) The test statistic is 0.7752 < 6.52, 4.63 and 3.82 which are the critical values. This indicates that we reject the joint null hypothesis of a unit root and no drift at all levels. I think it is more appropriate to include a constant(drift) in the ADF test

```
Value of test-statistic is: -2.8868 3.0806 4.3791

Critical values for test statistics:
      1pct  5pct 10pct
tau3 -3.99 -3.43 -3.13
phi2  6.22  4.75  4.07
phi3  8.43  6.49  5.47
```

3.3) The test statistic = 4.791 < 8.43, 6.49 and 5.47, indicating we reject the null hypothesis at any level. We reject the joint null of a unit root, no drift and no trend. This indicates we need to include the constant(drift) and trend in the ADF test.

3.4)

```
DWTI <-diff(WTI_train)
DWTI_ADF_none <- ur.df(DWTI, type="none", lags=12, selectlags = "AIC")
summary(DWTI_ADF_none)
```

```
Value of test-statistic is: -8.5049

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62
```

Based on the ADF test on unit root, we see that the test statistic is -8.5 < -2.58, -1.95, -1.62, indicating we reject the null hypothesis of a unit root at any level of significance. Thus, the first difference sequence is stationary.

```
DWTI_ADF_drift <- ur.df(DWTI, type="drift", lags=12, selectlags = "AIC")
```

summary(DWTI_ADF_drift)

```
Value of test-statistic is: -8.5757 36.7836

Critical values for test statistics:
      1pct  5pct 10pct
tau2 -3.46 -2.88 -2.57
phi1  6.52  4.63  3.81
```

Based on the test statistic for drift, we find that 36.7836 > 6.52, 4.63 and 3.83, indicating we cannot reject the null hypothesis. Thus, the join null hypothesis with no drift remains.

DWTI_ADF_trend <- ur.df(DWTI, type="trend", lags=12, selectlags = "AIC")
summary(DWTI_ADF_trend)

```
Value of test-statistic is: -8.6288 24.8783 37.3048

Critical values for test statistics:
      1pct  5pct 10pct
tau3 -3.99 -3.43 -3.13
phi2  6.22  4.75  4.07
phi3  8.43  6.49  5.47
```

Based on the test statistic, 37.3048 > 8.43, 6.49, 5.47, indicating we cannot reject the joint null hypothesis of a unit root, no drift and no trend.

This indicates that the best specification is the first model which does not have a unit root since we could not reject the null hypothesis for drift, drift and trend, but we reject it for unit root. Thus, the integrated order is 1.

Q4: ARMA Model Selection (10 points)

Select the best ARMA (or ARIMA with $d = 0$) model for the first difference sequence of WTI. Set `max.p=5, max.q=5, max.order=10, seasonal=FALSE, stepwise=FALSE.`

Q4.1. (5 points) Use AIC (`ic='aic'` in `auto.arima`). Report the estimation results, including the best ARMA model and the estimated coefficients and standard errors.

Q4.2. (5 points) Use BIC (`ic='bic'` in `auto.arima`). Report the estimation results, including the best ARMA model and the estimated coefficients and standard errors.

4.1)

auto.arima(DWTI, max.p=5, max.q=5,ic="aic", max.order=10, seasonal=FALSE, stepwise=FALSE)

```
Series: DWTI
ARIMA(0,0,4) with zero mean

Coefficients:
         ma1      ma2      ma3      ma4
      0.2376  -0.0652  -0.0582  -0.1933
s.e.  0.0681   0.0702   0.0728   0.0649

sigma^2 estimated as 3.482:  log likelihood=-429.13
AIC=868.26   AICc=868.55   BIC=885.02
```

Based on AIC, ARMA(0,4) is the best model with the coefficients and standard error listed above. ARIMA(0,0,4) = ARMA(0,4)

4.2)

auto.arima(DWTI, max.p=5, max.q=5,ic="bic", max.order=10, seasonal=FALSE, stepwise=FALSE)

```
Series: DWTI
ARIMA(0,0,1) with zero mean

Coefficients:
         ma1
      0.2602
s.e.  0.0707

sigma^2 estimated as 3.581:  log likelihood=-433.52
AIC=871.05   AICc=871.1   BIC=877.75
```

Based on BIC, ARMA(0,1) is the best model with the coefficients and standard error listed above. ARIMA(0,0,1) = ARMA(0,1)

Q5: Cointegration Test and VECM (20 points)

Now lets move to the bivariate case. Defined `Brent_train` and `Brent_test` accordingly. Combine `WTI_train` and `Brent_train` in to a data frame `df_level <- data.frame(WTI_train, Brent_train)` to be used in the cointegration test.

Q5.1. (10 points) Report the cointegration test results and discuss how many cointegration relation(s) you find. You can use the `ca.jo` function with the following options: `type='eigen', ecdet="const", K=2, spec="transitory"`.

Q5.2. (5 points) You can obtain the error correction term by using the following command (assume you save the output from the `ca.jo` function in `vecm`)

```
vecm.rls <- cajorls(vecm, r = 1)
error <- vecm.rls$rlm$model['ect1']
```

Note: you need the `vars` package to use the function `cajorls`.

Apply the ADF test on this error correction term (with options `type="none", lags=12, selectlags = "AIC"`) and report the test statistic and whether it is stationary or not. Plot `error$ect1` and its ACF and PACF. Does it behave like a white noise process?

Q5.3. (5 points) Report the estimation results of VECM.

```
cajorls(vecm)
coef(summary(cajorls(vecm)$rlm))
```

Discuss which sequence(s) significantly respond to the error correction term (at 5% significance level)? That is, in which equation(s) does the error correction term have significant (at 5% significance level) coefficient(s)?

```
Brent_train <- data$Brent[0:212]
Brent_train
Brent_test <- data$Brent[213:224]
Brent_test
df_level <- data.frame(WTI_train, Brent_train)
df_level
vecm <-ca.jo(df_level, type="eigen", ecdet="const", K=2, spec="transitory")
summary(vecm)
```

```
######################
# Johansen-Procedure #
######################

Test type: maximal eigenvalue statistic (lambda max) , without linear trend and constant in cointegration

Eigenvalues (lambda):
[1]  1.584015e-01  1.341374e-02 -2.034015e-17

Values of teststatistic and critical values of test:

          test 10pct  5pct  1pct
r <= 1 |  2.84  7.52  9.24 12.97
r = 0  | 36.21 13.75 15.67 20.20

Eigenvectors, normalised to first column:
(These are the cointegration relations)

               WTI_train.l1 Brent_train.l1   constant
WTI_train.l1      1.0000000       1.000000  1.0000000
Brent_train.l1   -1.0672877      -0.697941 -0.7664606
constant         -0.1080767      -8.888882 -2.5618600

Weights W:
(This is the loading matrix)

              WTI_train.l1 Brent_train.l1      constant
WTI_train.d      0.2167937    -0.08647796 -3.867927e-16
Brent_train.d    0.5577903    -0.08353248 -1.237581e-15
```

Based on the Johansen-Procedure, we see that for r=0, we reject the null hypothesis since 36.21 > 13.75, 15.67, 20.20, indicating evidence of at least 1 cointegration relation. For r <=1, we cannot reject the null hypothesis since 2.84 < 7.52, 9.24, 12.97, indicating that there is only 1 cointegrating relationship

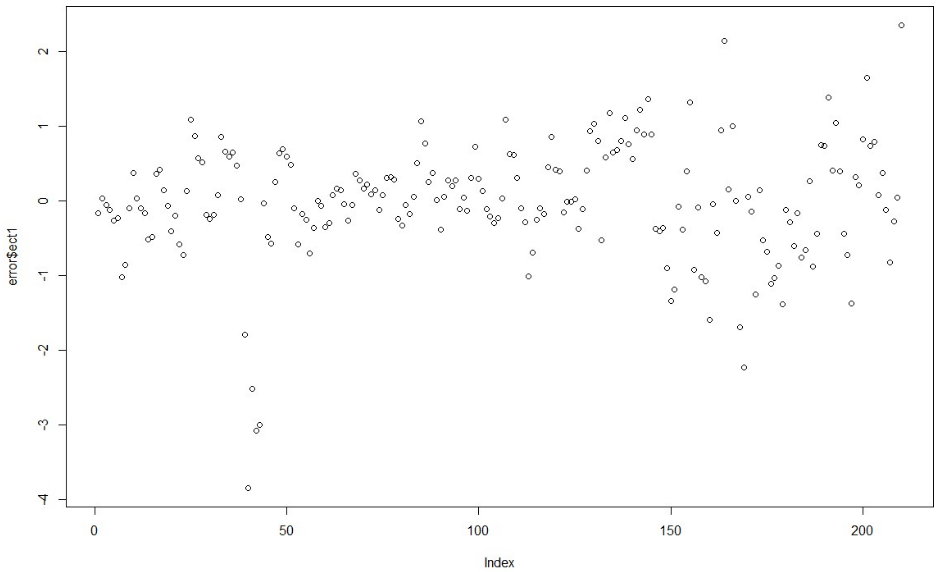
$Z_t = WTI_t - 1.0672877 Brent_t - 0.1080767$

```
vecm.rls <- cajorls(vecm, r = 1)
error <- vecm.rls$rlm$model["ect1"]
error.ADF <- ur.df(error$ect1, type="none",lags=12, selectlags = "AIC")
summary(error.ADF)
```

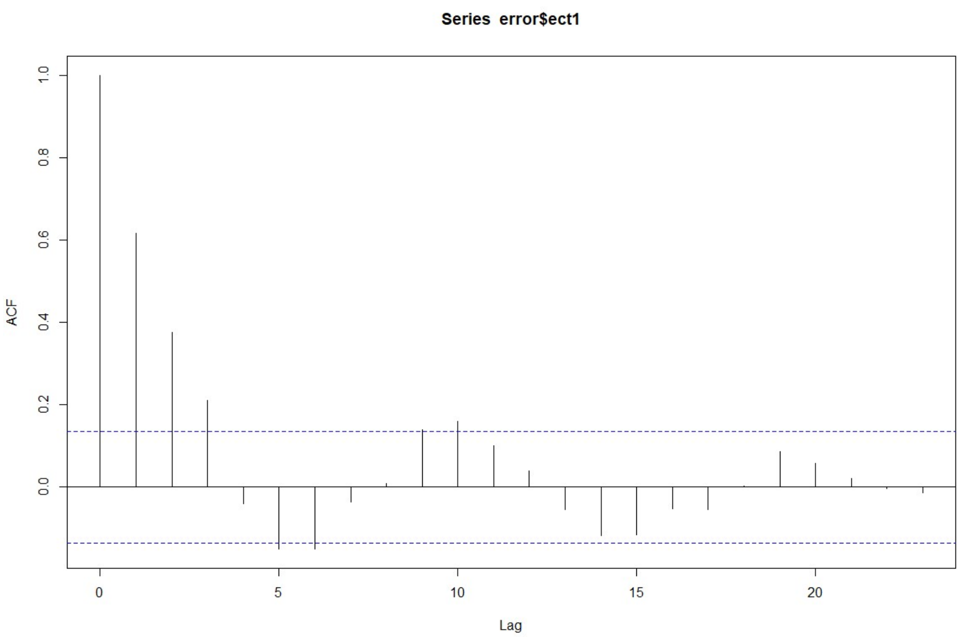
```
Value of test-statistic is: -3.1835

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62
```

The results of the ADF test on the error correction term, indicates that -3.18 < -2.58, -1.95 and -1.62. This indicates that we reject the null hypothesis of a unit root and that there is stationarity at any level of significance.

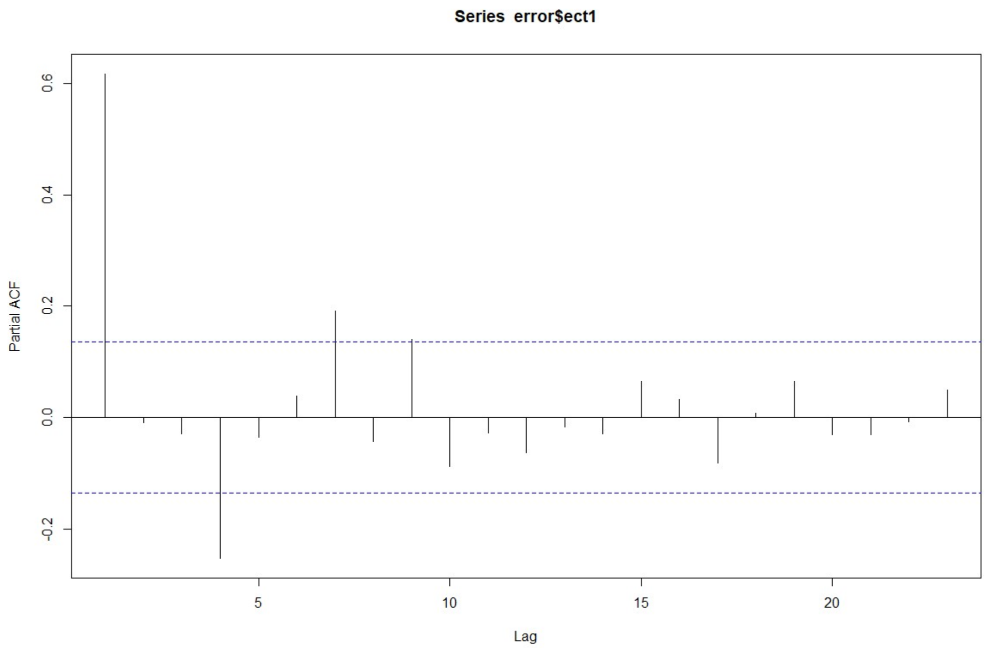
```
plot(error$ect1)
```



acf(error$ect1)



pacf(error$ect1)



The ACF plot indicates that the error correction term does not behave like white noise since there is serial dependence and there are autocorrelation lags greater than 0 as seen with spikes 0,1,2 and 3 and PACF plot showed spikes past the specified range.

cajorls(vecm)

```
Call:
lm(formula = substitute(form1), data = data.mat)

Coefficients:
                 WTI_train.d  Brent_train.d
ect1              0.2168       0.5578
WTI_train.dl1    -0.2347      -0.2406
Brent_train.dl1   0.4679       0.4620


$beta
                   ect1
WTI_train.l1    1.0000000
Brent_train.l1 -1.0672877
constant       -0.1080767
```

coef(summary(cajorls(vecm)$rlm))

```
Response WTI_train.d :
                  Estimate Std. Error    t value   Pr(>|t|)
ect1             0.2167937  0.1847225  1.1736181 0.24189662
WTI_train.dl1   -0.2347383  0.2448248 -0.9588013 0.33877739
Brent_train.dl1  0.4679492  0.2350365  1.9909637 0.04780003

Response Brent_train.d :
                  Estimate Std. Error    t value    Pr(>|t|)
ect1             0.5577903  0.1967175  2.8354890 0.005029451
WTI_train.dl1   -0.2406352  0.2607224 -0.9229555 0.357104961
Brent_train.dl1  0.4619795  0.2502986  1.8457131 0.066362008
```

From the WTI training set, Brent_training are statistically significant. From the Brent training set, only ect1 was statistically significant. Thus, the Brent equation had the statistically significant ect1 coefficient at a 5% level since p-value = 0.005 < 0.05.

Q6: Forecast (20 points)

Forecast with vecm first requires to transform the model from the first difference form to the level form using `vec2var`. Use the `predict` function to conduct forecast in the next 12 months.

```
var.model = vec2var(vecm)
H = 12
fc <- predict(var.model, n.ahead=H)
```

You can transform the forecast values into time series format by the following commands

```
WTI_forecast <- ts(fc$fcst$WTI_train[1:H,1], frequency=12, start=t2)
Brent_forecast <- ts(fc$fcst$Brent_train[1:H,1], frequency=12, start=t2)
```

Report the RMSE (root mean square error) and the MAE (mean absolute error) for the forecasts of these two sequences.

```
var.model = vec2var(vecm)
H = 12
fc <- predict(var.model, n.ahead=H)
WTI_forecast <- ts(fc$fcst$WTI_train[1:H,1], frequency=12, start=t2)
Brent_forecast <- ts(fc$fcst$Brent_train[1:H,1], frequency=12, start=t2)
accuracy(WTI_forecast,WTI_test)
```

	ME	RMSE	MAE	MPE	MAPE	ACF1	Theil's U
Test set	13.04817	14.28923	13.04817	22.23909	22.23909	0.6911544	3.512515

For the WTI forecast errors,

RMSE = 14.28923

MAE = 13.04817

```
accuracy(Brent_forecast,Brent_test)
```

	ME	RMSE	MAE	MPE	MAPE
Test set	13.90371	15.00149	13.90371	24.70016	24.70016

For the Brent forecast errors,

RMSE = 15.00149

MAE = 13.90371