Time_Series_HW3

Patrick Kelly

Thursday, May 07, 2015

1. Use datasets from 1955 to 1968 to build an ARMA or ARIMA models for UN and GDP.

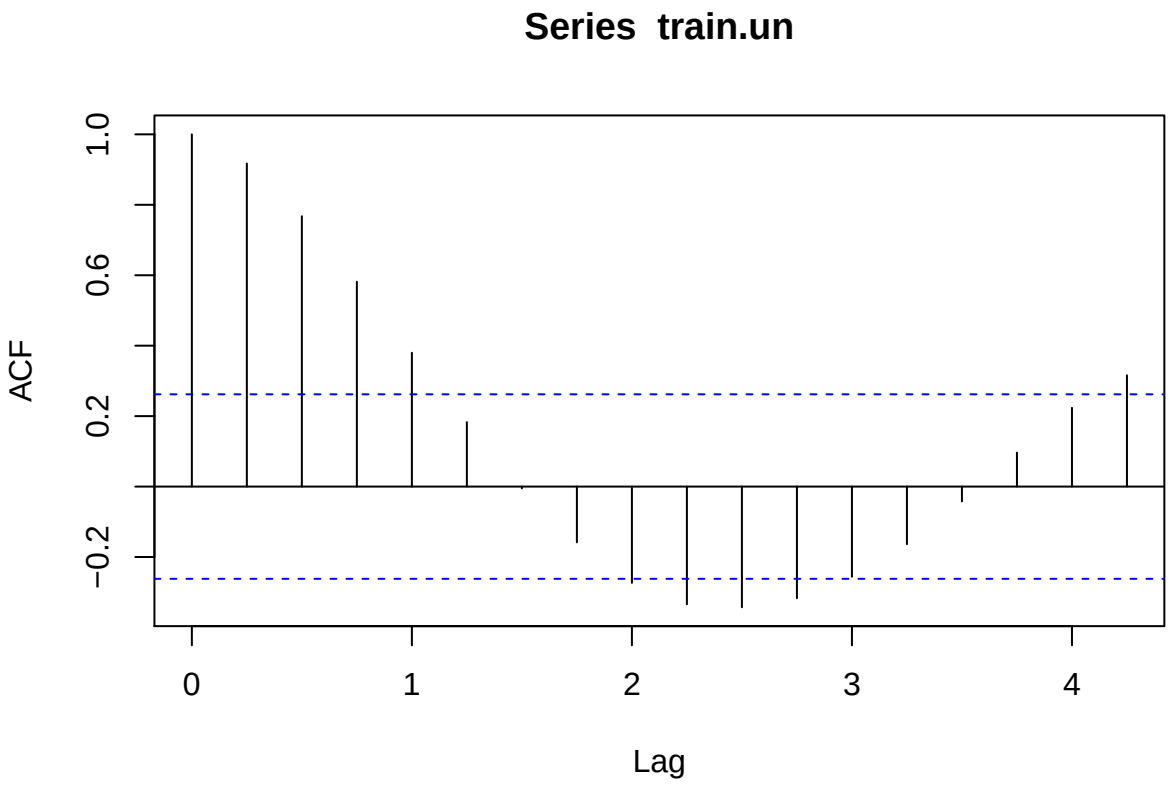
```
#install packages
library('tseries')
library('ggplot2')

#read in data
un.gdp.uk <- read.csv("~/Education/UChicago/Time_Series/Unemployment_GDP_UK.csv", header = TRUE)

un <- ts(data = un.gdp.uk$UN, frequency = 4, start = c(1955,1))
gdp <- ts(data = un.gdp.uk$GDP, frequency = 4, start = c(1955,1))

train.un <- window(x = un, end = c(1968,4))
train.gdp <- window(x = gdp, end = c(1968,4))

#check stationarity
adf.test(train.un)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  train.un
## Dickey-Fuller = -3.3336, Lag order = 3, p-value = 0.07538
## alternative hypothesis: stationary
```

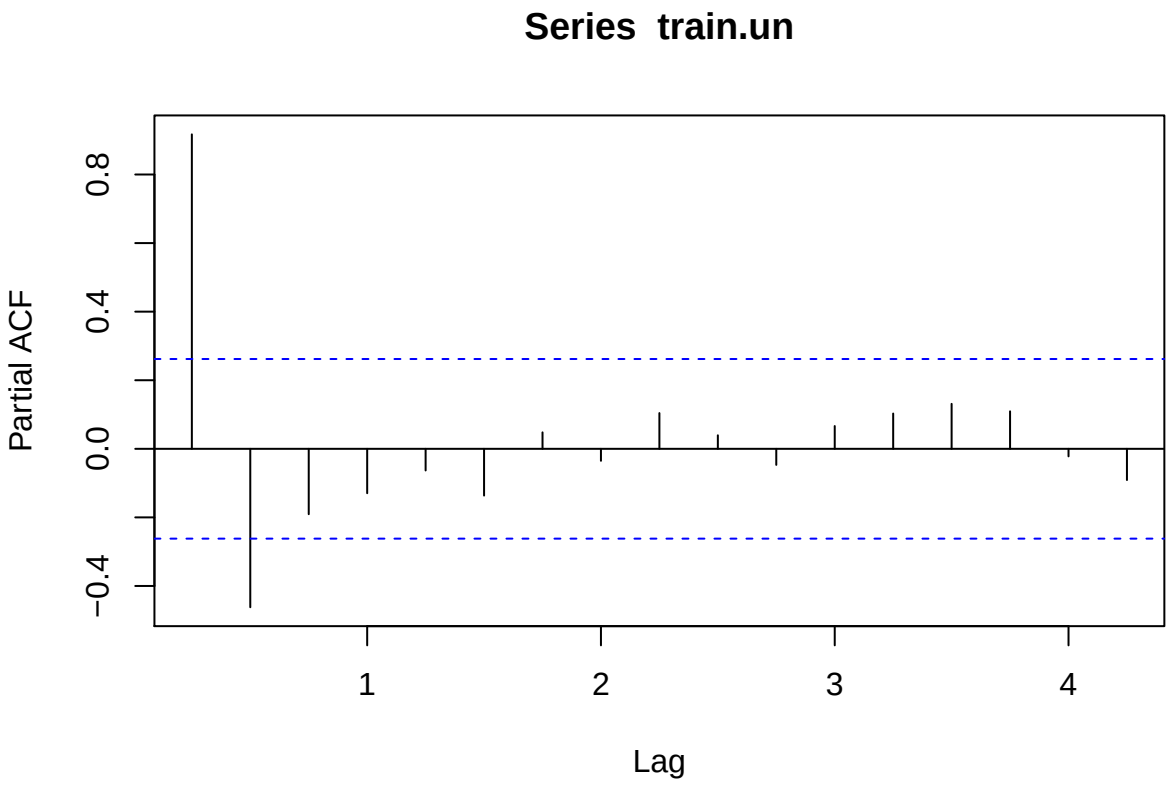
```
adf.test(train.gdp)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  train.gdp
## Dickey-Fuller = -2.9551, Lag order = 3, p-value = 0.1895
## alternative hypothesis: stationary
```

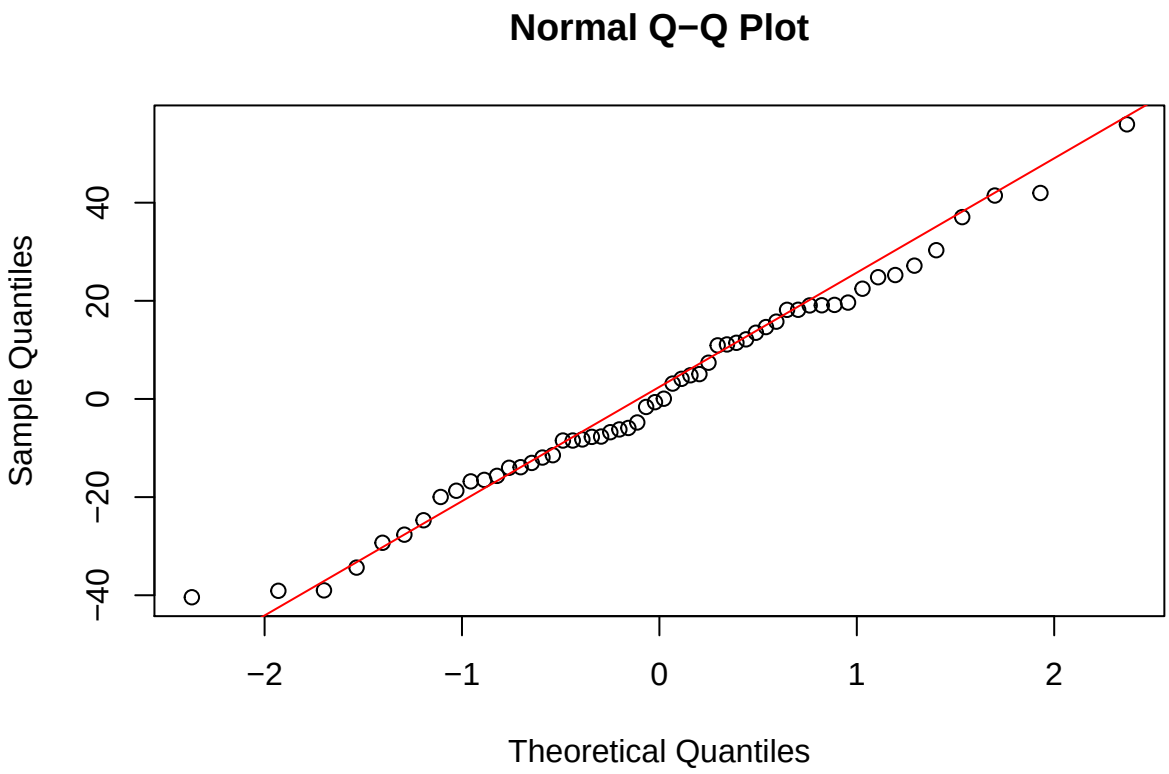
```
###use ARMA model for UN bc it appears to be stationary
#first, check the acf and pacf graphs
acf(train.un)  # this output indicates that q should equal 4 for MA(q)
```




```
pacf(train.un) # this output indicates that p should equal 1 for AR(p)
```




```
model.un <- arima(x = train.un, order = c(1,0,4))

#check normal distribution of residuals
qqnorm(model.un$residuals)
qqline(model.un$residuals,col=2)
```




```
#use ARIMA model for GDP bc it appears to NOT be stationary
#first, check the stationarity of the 1st diff and 2nd diff
train.gdp.d1 <- diff(train.gdp)
train.gdp.d2 <- diff(train.gdp.d1)

adf.test(train.gdp.d1) #not stationary
```

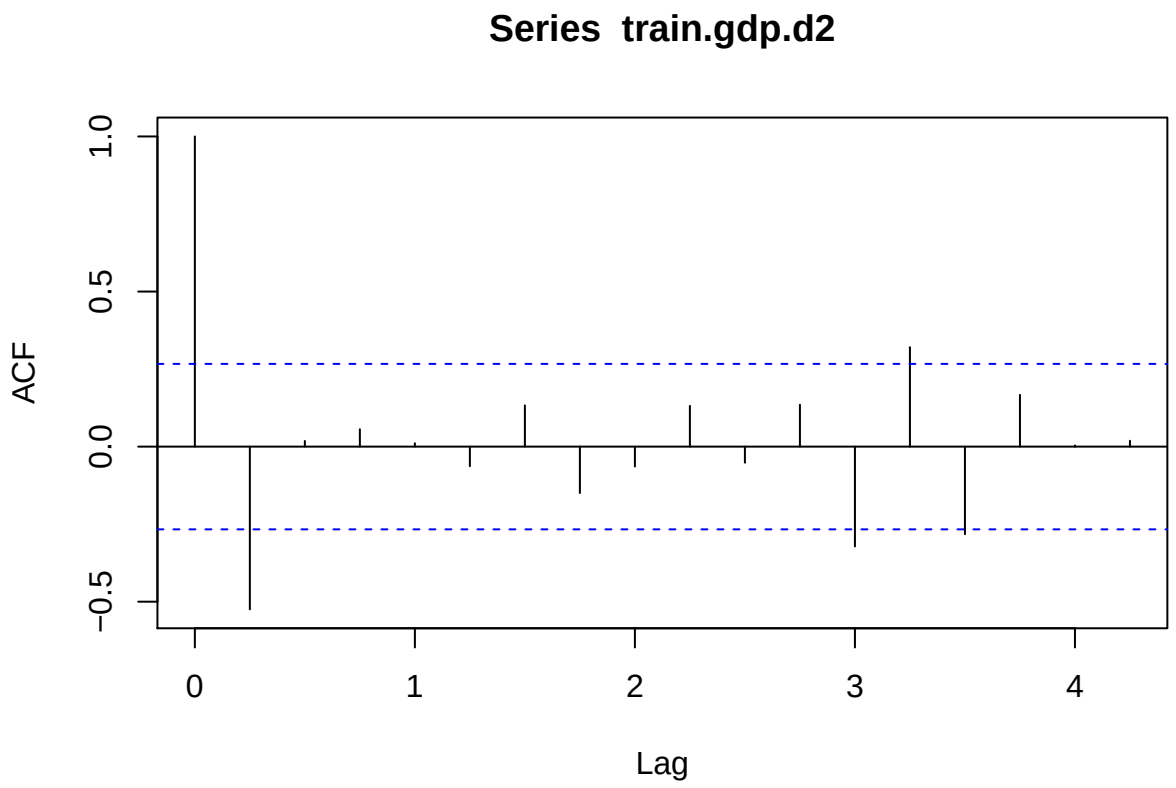
```
##
##  Augmented Dickey-Fuller Test
##
## data:  train.gdp.d1
## Dickey-Fuller = -2.8173, Lag order = 3, p-value = 0.2452
## alternative hypothesis: stationary
```

```
adf.test(train.gdp.d2) #use the 2nd difference
```

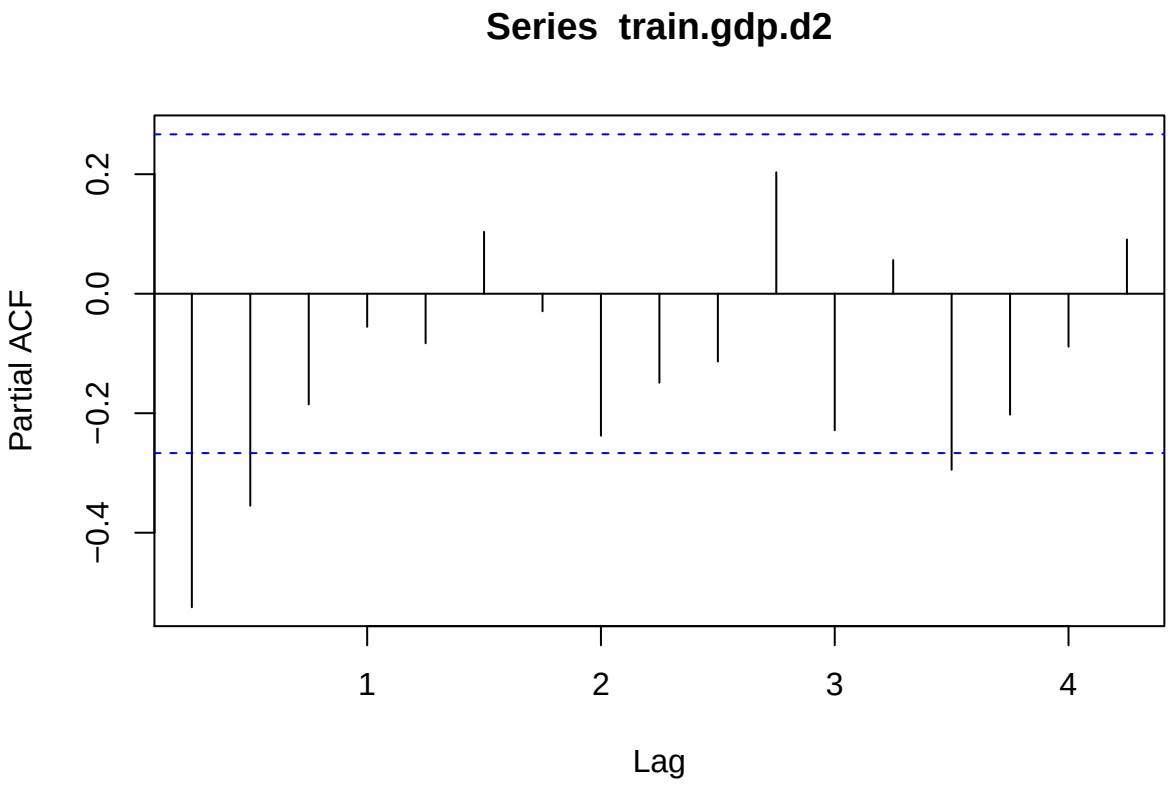
```
## Warning in adf.test(train.gdp.d2): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  train.gdp.d2
## Dickey-Fuller = -4.9224, Lag order = 3, p-value = 0.01
## alternative hypothesis: stationary
```

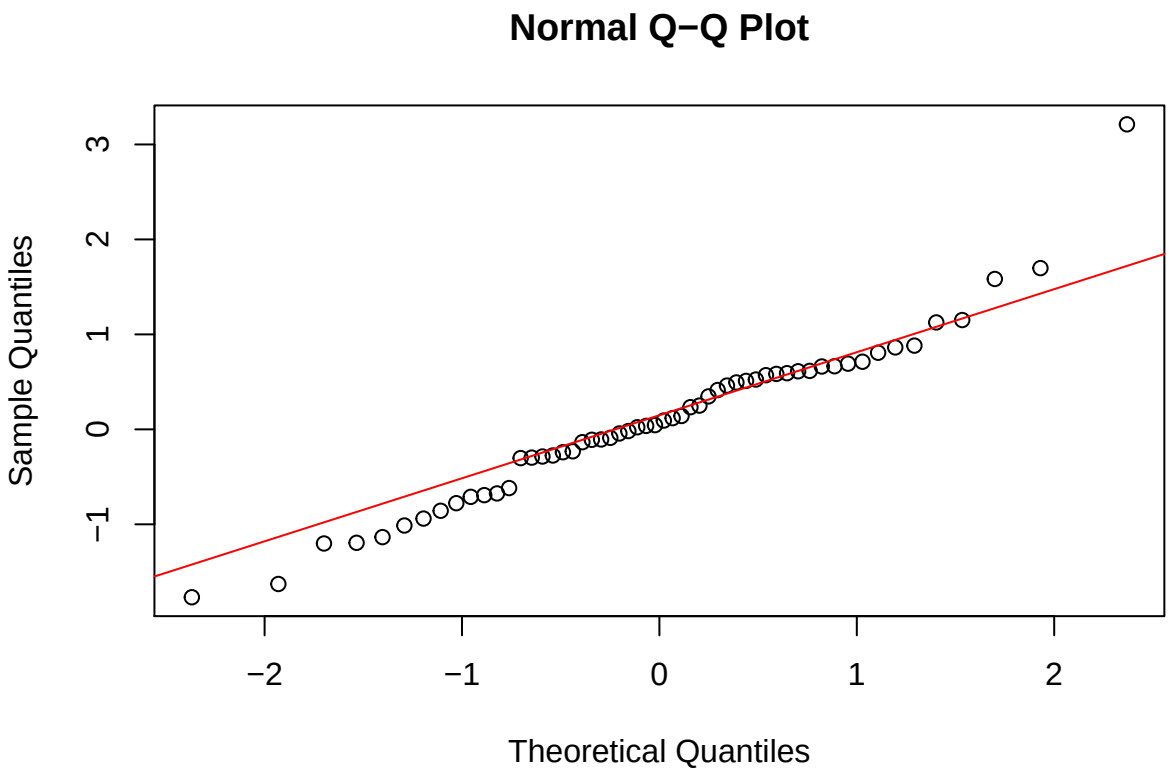
```r
#next, check the acf and pacf graphs
acf(train.gdp.d2)  # this output indicates that q should equal 1 for MA(q)
```




```r
pacf(train.gdp.d2) # this output indicates that p should equal 1 for AR(p)
```




```
model.gdp <- arima(x = train.gdp, order = c(1,2,1))

#check normal distribution of residuals
qqnorm(model.gdp$residuals)
qqline(model.gdp$residuals,col=2)
```



2. Justify why you chose (ARMA or ARIMA) one over the other. Note there will be 2 models, one for UN and another for GDP

As is mentioned in the notes above, it is necessary to test the stationarity of each time series to determine which methodology is appropriate. The UN data proved to be stationary (per the adf test) so an ARMA model is appropriate and was used above. However, we were not able to confirm that the GDP data was stationary (per the adf test) so an ARIMA model is appropriate - checking the stationarity of the 1st and 2nd degree differences then indicated that 2 is the appropriate `d` value within the ARIMA(p,d,q) model.

3. Use the chosen UN and GDP models to forecast the UN and the GDP for 1969

```
#predict the next 4 quarters (1969) for both models
un.forecast.1969 <- predict(object = model.un, n.ahead = 4)
un.forecast.1969$pred
```

```
##          Qtr1     Qtr2     Qtr3     Qtr4
## 1969 511.3963 488.6112 466.8458 453.4144
```

```
gdp.forecast.1969 <- predict(object = model.gdp, n.ahead = 4)
gdp.forecast.1969$pred
```

```
##          Qtr1     Qtr2     Qtr3     Qtr4
## 1969 118.5719 119.3098 120.0458 120.7817
```

4. Compare your forecasts with the actual values using error = actual - estimate and plot the errors

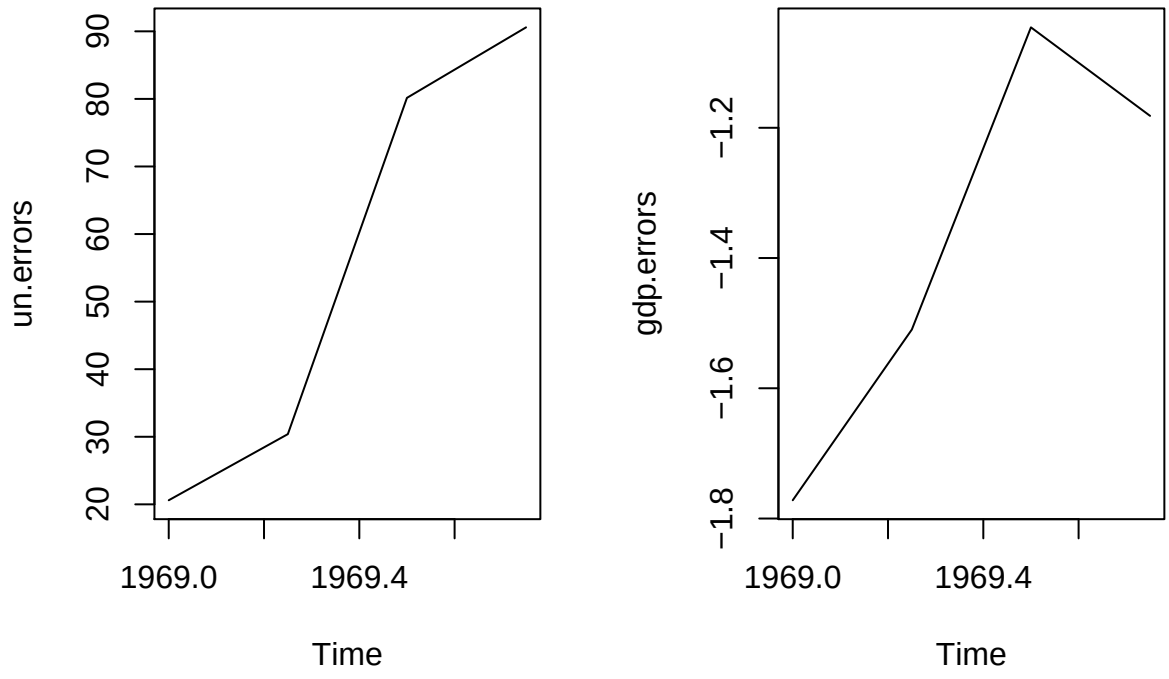
```
#compare predicted values to actuals
(un.errors <- window(x = un, start = c(1969,1), end = c(1969,4)) - un.forecast.1969$pred)
```

```
##          Qtr1     Qtr2     Qtr3     Qtr4
## 1969 20.60371 30.38881 80.15416 90.58556
```

```
(gdp.errors <- window(x = gdp, start = c(1969,1), end = c(1969,4)) - gdp.forecast.1969$pred)
```

```
##           Qtr1      Qtr2      Qtr3      Qtr4
## 1969 -1.771947 -1.509802 -1.045807 -1.181712
```

```
par(mfrow = c(1,2))
plot(un.errors)
plot(gdp.errors)
```



5. Calculate the Sum of squared(error) for each UN and GDP models

```
# UN model sum of squared erros
(un.sse <- sum(sapply(un.errors, function(x) x**2)))
```

```
## [1] 15978.43
```

```r
# GDP model sum of squared erros
(gdp.sse <- sum(sapply(gdp.errors, function(x) x**2)))
```

```
## [1] 7.909455
```

Regression - build regression models that use:

1. UN as the independent variable and GDP as the dependent variable - use data from 1955 to 1968 to build the model. Forecast for 1969 and plot the errors and calculate the sum of squared(error) as previously

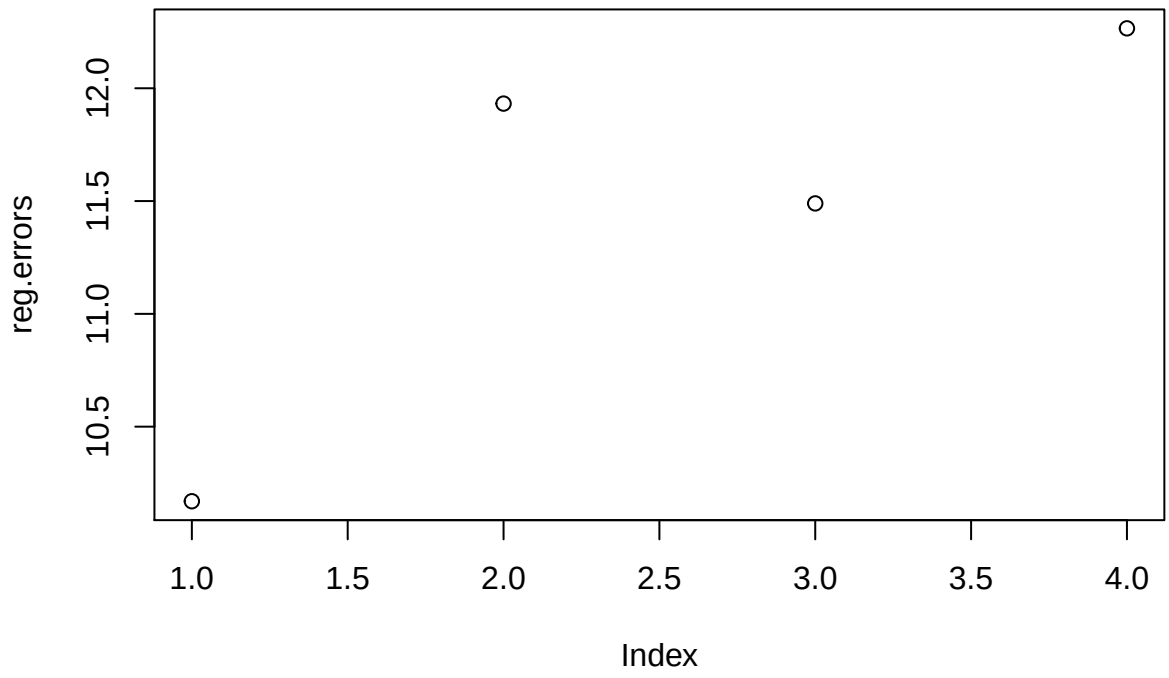
```r
#create training and new data by seperating out Year 1969
reg.train.data <- un.gdp.uk[un.gdp.uk$Year != 1969,]
reg.pred.data <- un.gdp.uk[un.gdp.uk$Year == 1969,]

#create lm() using GDP and UN data
reg.gdp.un <- lm(GDP ~ UN, data = reg.train.data)

#forecast for 1969
reg.forecast <- predict.lm(object = reg.gdp.un, newdata = reg.pred.data, type = "response")
reg.forecast
```

```
##       57       58       59       60
## 106.6305 105.8679 107.5105 107.3345
```

```r
#plot errors
(reg.errors <- reg.pred.data$GDP - reg.forecast)
```

```
##       57       58       59       60
## 10.16947 11.93207 11.48954 12.26553
```

```r
par(mfrow=c(1,1))
plot(reg.errors)
```




```
#calculate the sum of squared errors
(reg.sse <- sum(sapply(reg.errors, function(x) x**2)))
```

```
## [1] 528.2449
```

2. GDP as the independent variable and UN as the dependent variable - use data from 1955 to 1968 to build the model. Forecast for 1969 and plot the errors and calculate the sum of squared(error) as previously

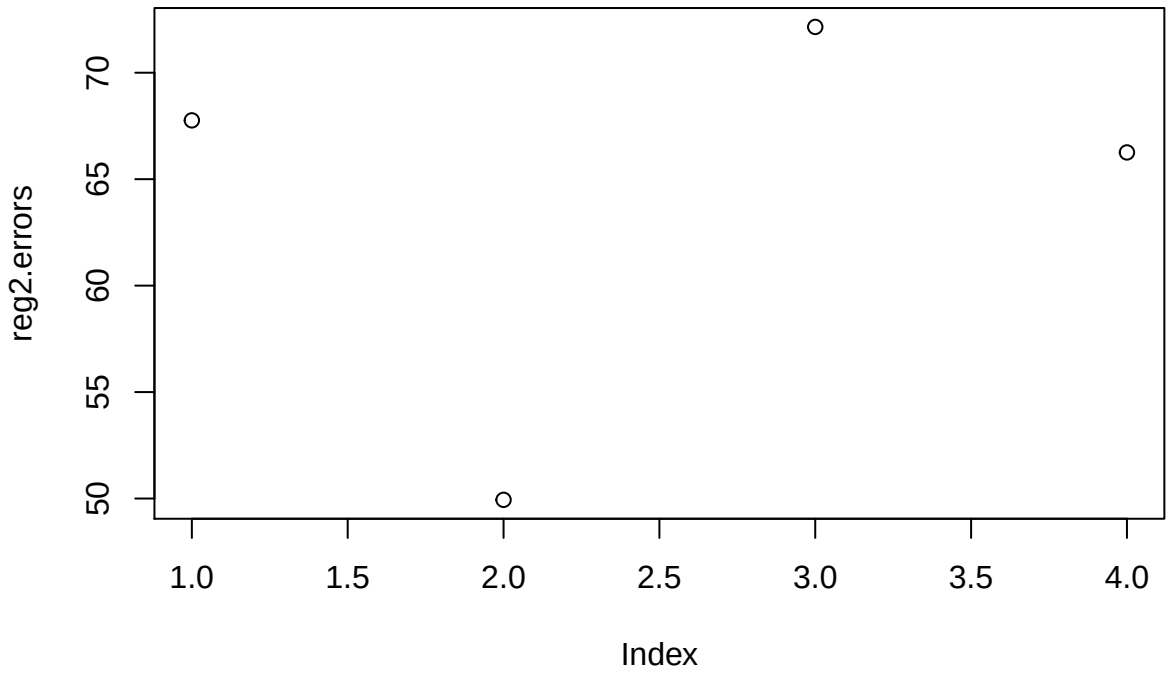
```
#create training and new data by seperating out Year 1969
reg2.train.data <- un.gdp.uk[un.gdp.uk$Year != 1969,]
reg2.pred.data <- un.gdp.uk[un.gdp.uk$Year == 1969,]

#create lm() using GDP and UN data
reg2.un.gdp <- lm(UN ~ GDP, data = reg2.train.data)

#forecast for 1969
reg2.forecast <- predict.lm(object = reg2.un.gdp, newdata = reg2.pred.data, type = "response")
reg2.forecast
```

```
##       57       58       59       60
## 464.2380 469.0615 474.8497 477.7438
```

```
#plot errors
(reg2.errors <- reg2.pred.data$UN - reg2.forecast)
```

```
##       57       58       59       60
## 67.76199 49.93849 72.15029 66.25619
```

```
par(mfrow=c(1,1))
plot(reg2.errors)
```




```
#calculate the sum of squared errors
(reg2.sse <- sum(sapply(reg2.errors, function(x) x**2)))
```

```
## [1] 16681.09
```

3. Compare the 2 models - any reason to believe which should be the independent and the dependent variables

In order to compare the two models, I will calculate the out of sample mean absolute percentage error and then compare these values to determine the model will the least relative error.

```
#mean absolute percentage error of the first model GDP ~ UN
(reg.mape <- mean(abs(reg.pred.data$GDP - reg.forecast)/reg.pred.data$GDP))
```

```
## [1] 0.09686589
```

```r
#mean absolute percentage error of the second model UN ~ GDP
(reg2.mape <- mean((reg2.pred.data$UN - reg2.forecast)/reg2.pred.data$UN))
```

```
## [1] 0.1193223
```

Based on the results above, in which the GDP on Unemployement model has a lower mean absolute percentage error out of sample (indicating that this model is a better fit) I conclude that GDP should be the dependent variable with Unemployement as the independent variable.