Week 5 Assignment

Matthew Dunne

February 9, 2019

Week 5: Homework Assignment

This assignment helps understanding stationarity and seasonality of linear models for time series

Complete reading Chapter 2, pages 117-125 Analysis of Moody's Bond Yields

Consider the monthly yields of Moody's AAA and BAA bonds from exercises 4-6 on page 126. The data are in the file MYieldsData.csv. Analyze possible types of relationships between the two yield variables using regression model with stationary residuals and cointegration.

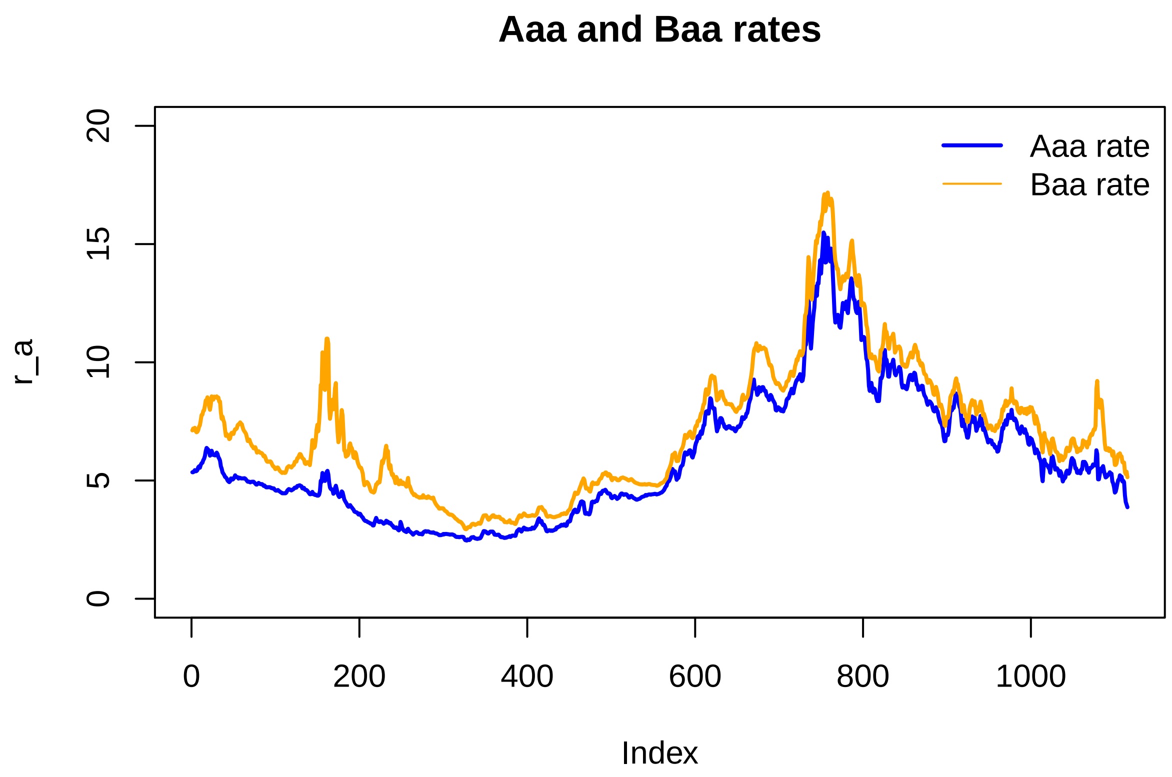
```
datapath<-"C:/Users/mjdun/Desktop/Financial Analytics/Week 5"
data<-read.csv(file=paste(datapath,"MYieldsData.csv",sep="/"),header=T)
head(data)
```

```
##     Date AAAyield BAAyield
## 1 Jan-19     5.35     7.12
## 2 Feb-19     5.35     7.20
## 3 Mar-19     5.39     7.15
## 4 Apr-19     5.44     7.23
## 5 May-19     5.39     7.09
## 6 Jun-19     5.40     7.04
```

Regression Model

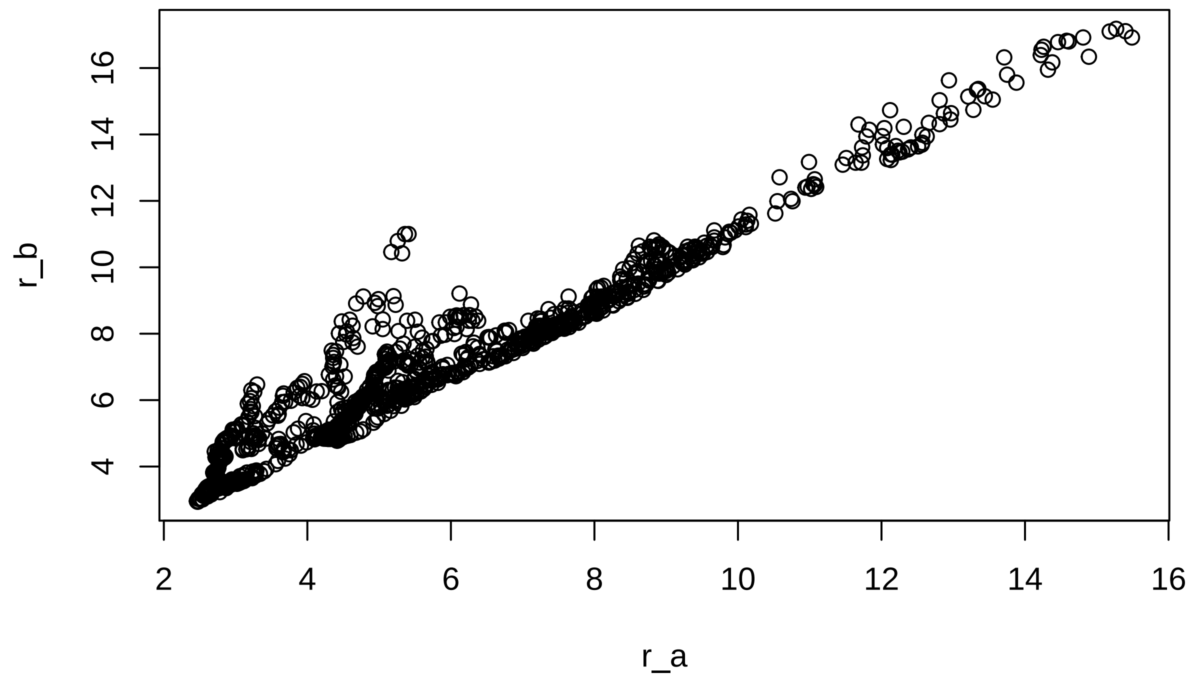
Do a simple plot of the raw data.

```
r_a<-data[,2]
r_b<-data[,3]
plot(r_a,col ="blue",type="l",lwd=2,main = "Aaa and Baa rates", ylim=c(0,20))
lines(r_b,col="orange",lwd=2)
legend("topright", c("Aaa rate","Baa rate"), lwd=c(2,1), col = c("blue","orange"), bty="n")
```



Would a regression model fit well on this data? Plot the relationship of Baa to Aaa.

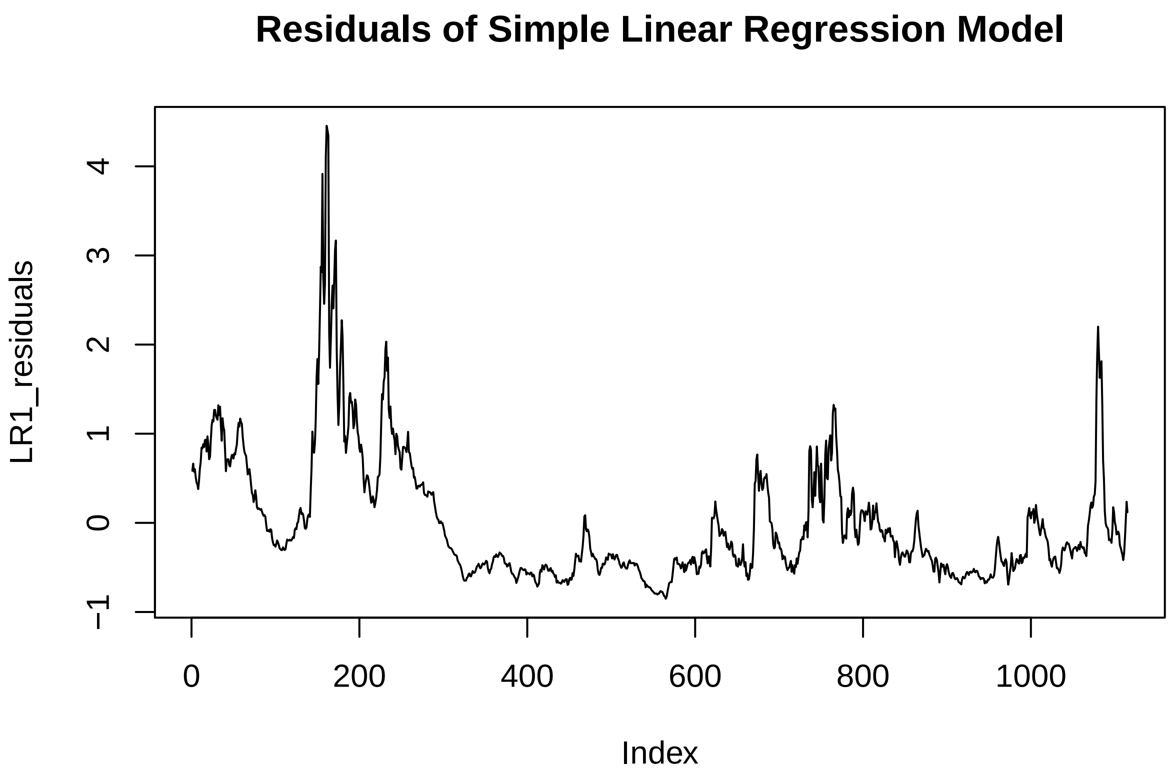
```
r_a<-as.vector(r_a)
r_b<-as.vector(r_b)
plot(r_a, r_b)
```



We see that a linear trend will fit but also that the noise around that trend is not random.

We need a linear regression model with stationary residuals. Do we have that at present?

```
linreg1<-lm(r_b~r_a)
LR1_residuals<-linreg1$residuals
plot(LR1_residuals, type = "l", main="Residuals of Simple Linear Regression Model")
```



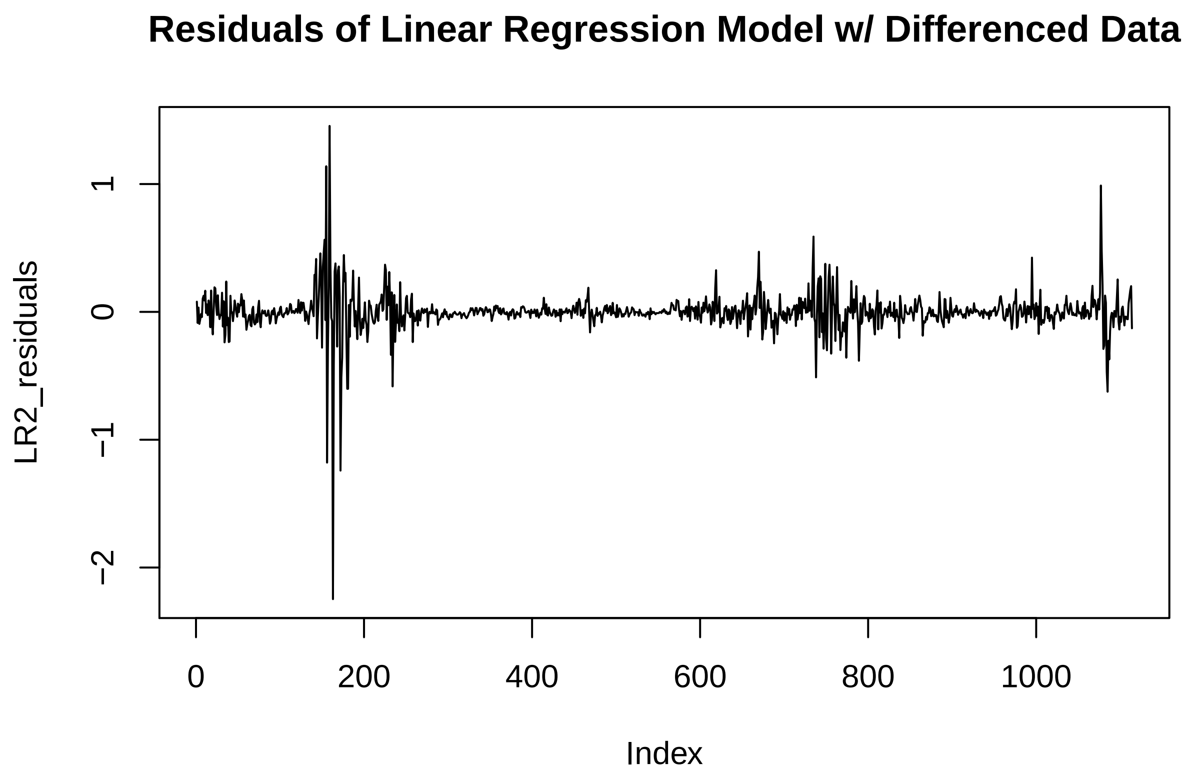
These residuals do not appear to be constant with respect to either mean or variance, i.e. they are not stationary. The Box test confirms it with a p-value of 0 - reject the null hypothesis of stationarity.

```
Box.test.LR1_residuals<-Box.test(LR1_residuals,lag=10,type='Ljung')
Box.test.LR1_residuals
```

```
##
##  Box-Ljung test
##
## data:  LR1_residuals
## X-squared = 8534.3, df = 10, p-value < 2.2e-16
```

Let us use differenced data as input for the Linear Regression model.

```
r_a_1<-diff(r_a)
r_b_1<-diff(r_b)
#make sure to take out the intercept
linreg2<-lm(r_b_1 ~ r_a_1-1)
LR2_residuals<-linreg2$residuals
plot(LR2_residuals, type="l", main="Residuals of Linear Regression Model w/ Differenced Data")
```



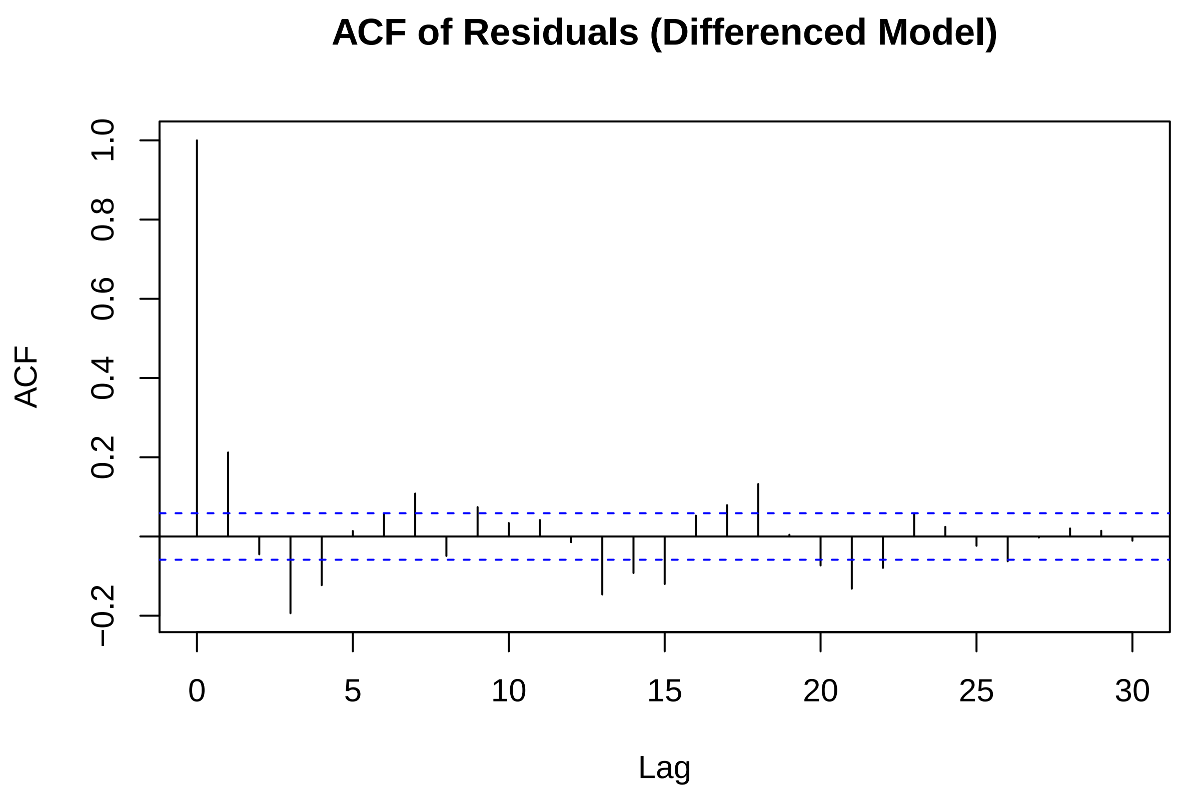
```
Box.test.LR2_residuals<-Box.test(LR2_residuals,lag=10,type='Ljung')
Box.test.LR2_residuals
```

```
##
##  Box-Ljung test
##
## data:  LR2_residuals
## X-squared = 139, df = 10, p-value < 2.2e-16
```

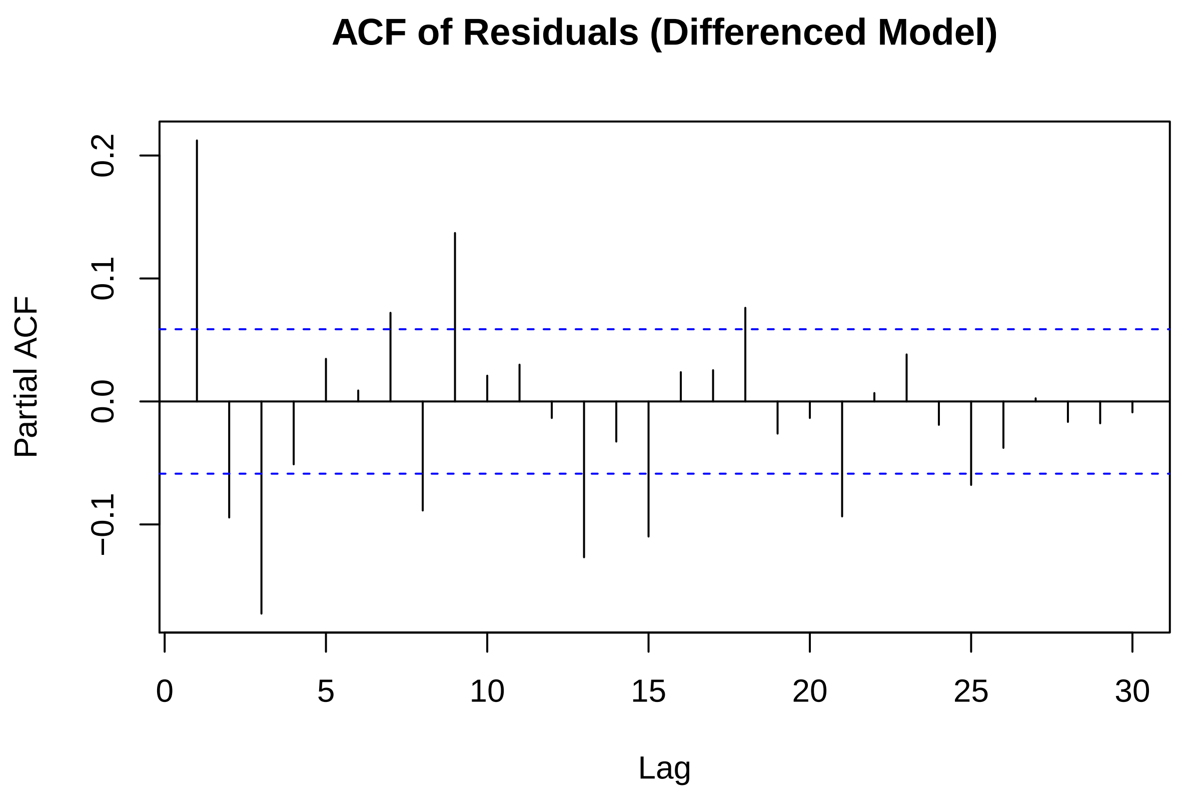
The residuals look constant with respect to mean but not variance. The Box test confirms non-stationarity of residuals.

So we need to fit an additional model on the residuals, but to decide which we must look at the ACF and PACF plots of the residuals.

```
acf(LR2_residuals, main="ACF of Residuals (Differenced Model)")
```



```r
pacf(LR2_residuals, main="ACF of Residuals (Differenced Model)")
```

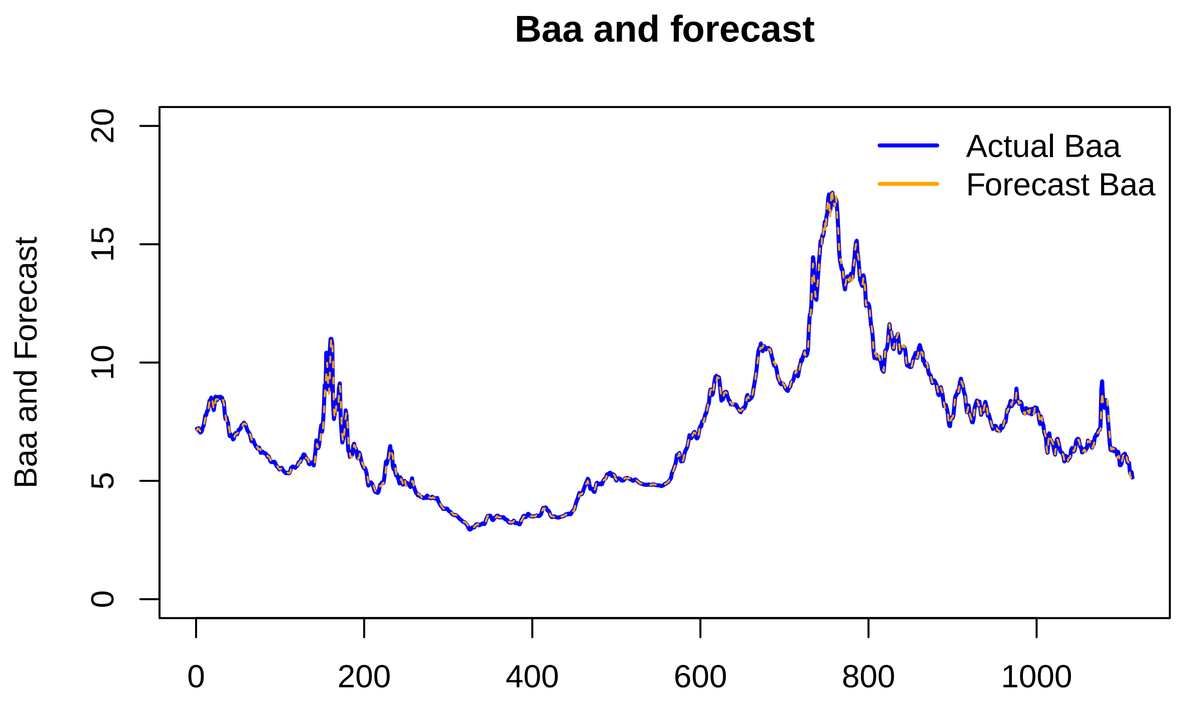


Let us try a MA(1) model on the residuals.

```
ma1<-arima(LR2_residuals, order=c(0,0,1))
#get the relevant coefficient
theta<-ma1$coef[1]
#get the residuals of the model
ma1_res<-residuals(ma1)
#get the values for the errors in the equation
a_t<-theta*ma1_res
```

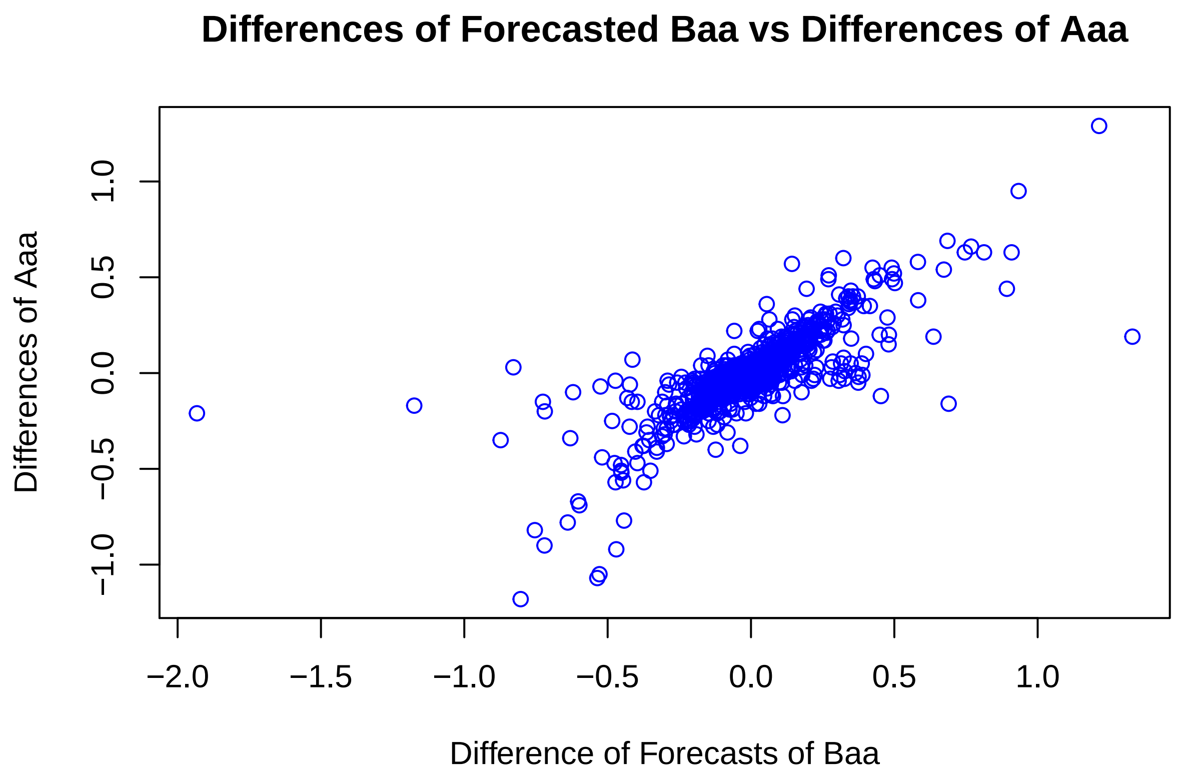
Then look at a one-step forecast of the Baa yields.

```
#take off last observation from original non-differenced Baa
x_1<-r_b[-length(r_b)]
#original Baa + LR model coeff. * Aaa differenced + the MA model on the errors
forec<-x_1 + linreg2$coefficients*r_a_1 + a_t
matplot(cbind(r_b[-1],forec),type = "l",col = c("blue","orange"),lwd=c(2,1),main= "Baa and forecast",yla
legend("topright", c("Actual Baa","Forecast Baa"), lwd=2,col = c("blue","orange"), bty="n")
```



What this shows is that we can predict *today's* Baa level based in part on *today's* Aaa level.

```
difforec <- diff(forec)
cr<-cbind(difforec,r_a_1[-1])
plot (cr[,1],cr[,2],col = "blue",main = "Differences of Forecasted Baa vs Differences of Aaa",
      xlab="Difference of Forecasts of Baa",ylab="Differences of Aaa")
```



We see that the regression model with ARIMA residuals more or less preserves the "short term" dependence of the variables. Although this relationship looks like it breaks down at the extremes in the differenced data.

Co-Integration

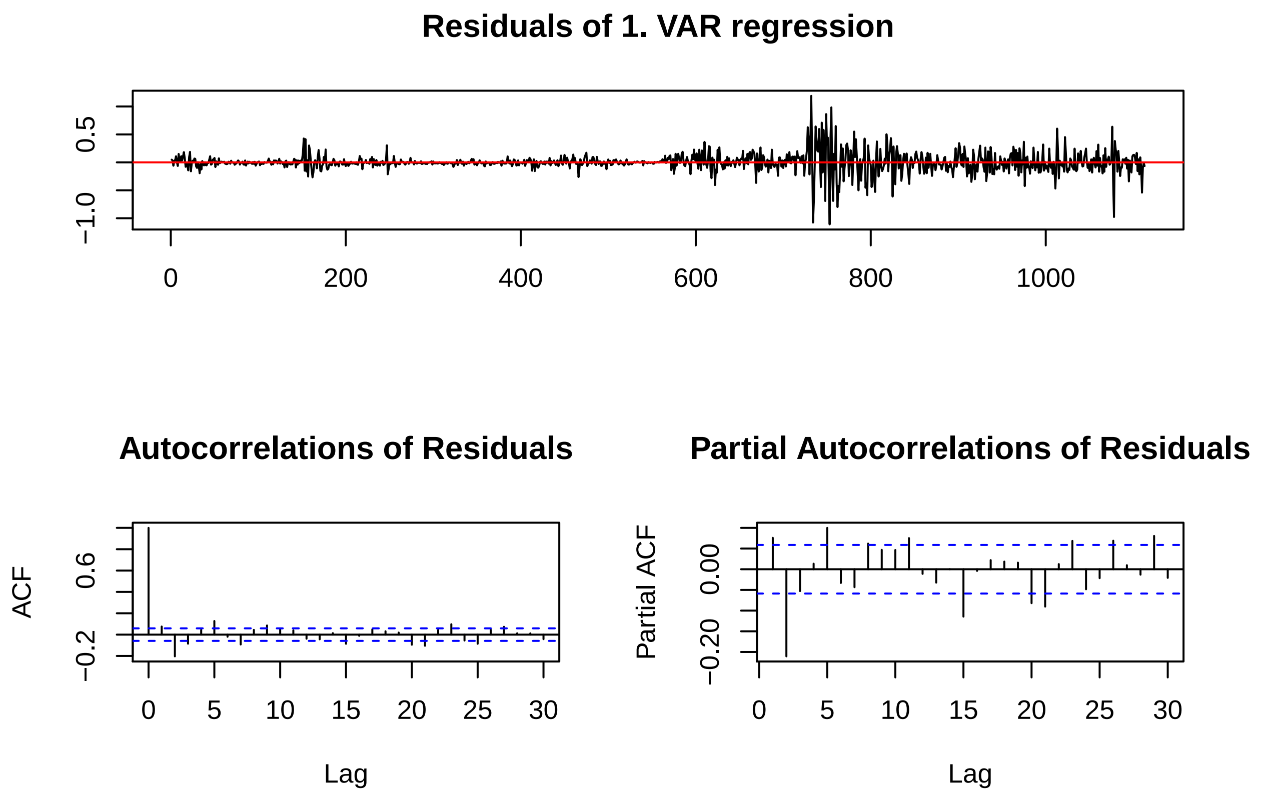
What if we use co-integration instead?

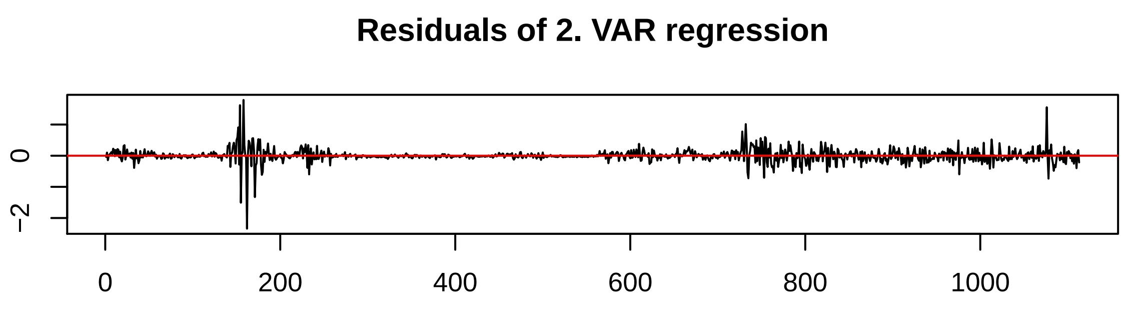
```
suppressWarnings(library(urca))
data<-cbind(r_a, r_b)
cajo <- ca.jo(data, ecdet = "none", type="eigen", K=2, spec="longrun")
summary(cajo)
```

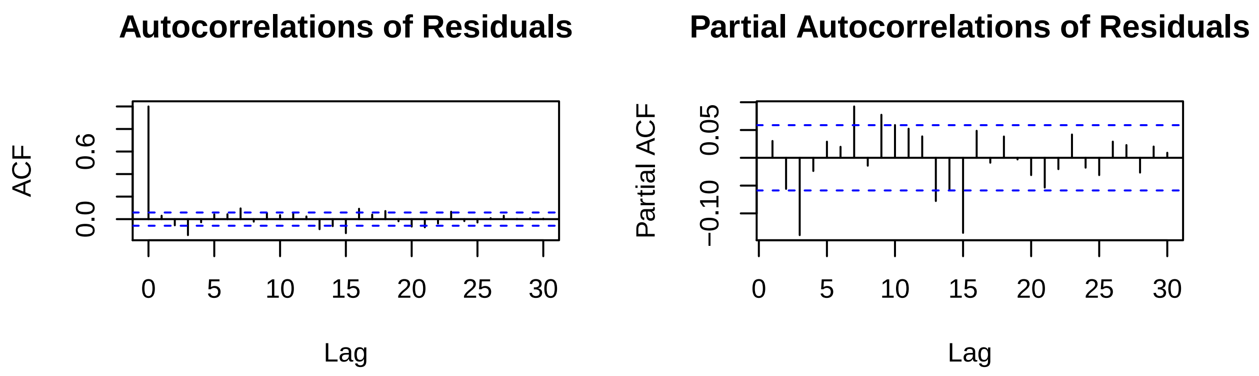
```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: maximal eigenvalue statistic (lambda max) , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.01938740 0.00201877
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 1 |  2.25  6.50  8.18 11.65
## r = 0  | 21.79 12.91 14.90 19.19
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##            r_a.l2    r_b.l2
## r_a.l2  1.0000000 1.0000000
## r_b.l2 -0.9689945 0.1320439
##
## Weights W:
## (This is the loading matrix)
##
##            r_a.l2       r_b.l2
## r_a.d 0.008113374 -0.002364087
## r_b.d 0.036577081 -0.001594315
```

Look at the residuals.

```r
plotres(cajo)
```







And the relevant statistic, critical values

```
cajo@teststat
```

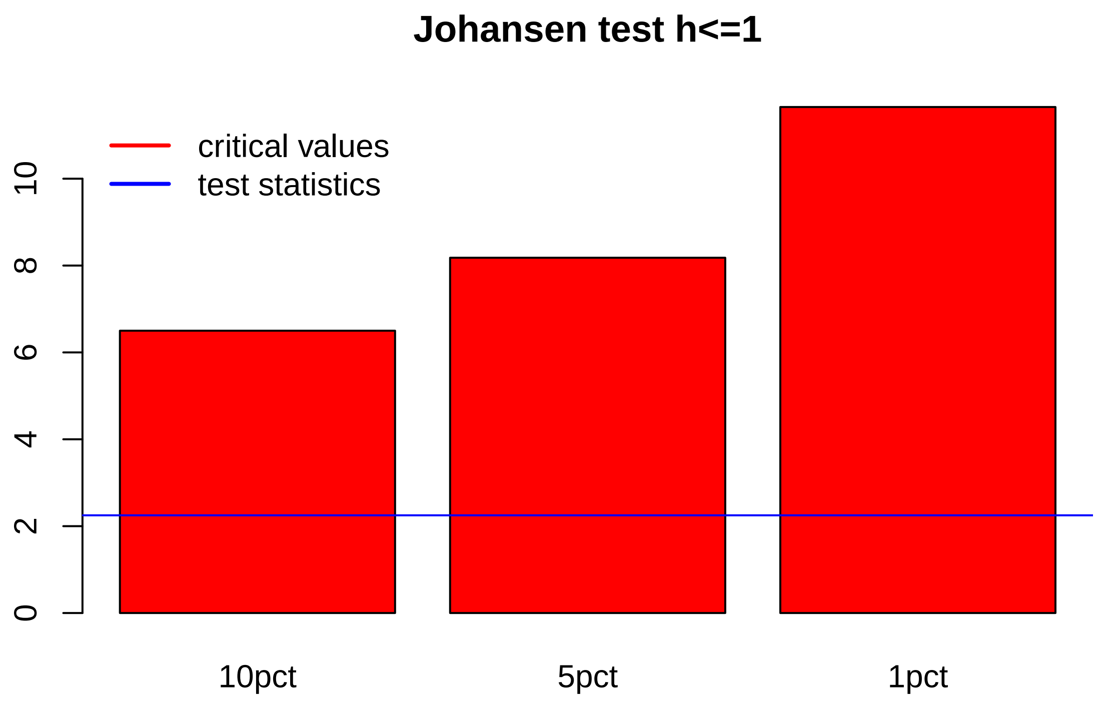
```
## [1]  2.249163 21.790090
```

```
cajo@cval
```

```
##           10pct  5pct  1pct
## r <= 1 |  6.50  8.18 11.65
## r = 0  | 12.91 14.90 19.19
```

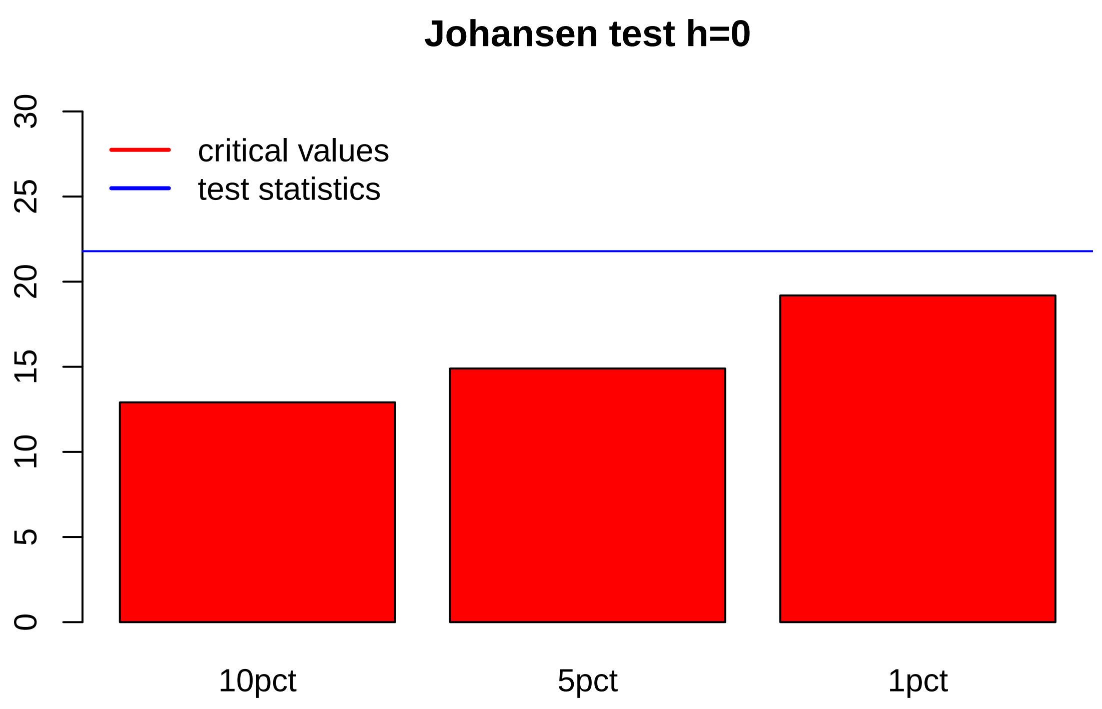
For the Johansen Test: h<=1

```
barplot(cajo@cval[1,],main = "Johansen test h<=1",col = "red")
abline(h=cajo@teststat[1], col="blue")
legend("topleft", c("critical values","test statistics"), lwd=2,col = c("red","blue"), bty="n")
```



And for the Johansen Test: h=0

```r
barplot(cajo@cval[2,],main = "Johansen test h=0",col = "red", ylim=c(0,30))
abline(h=cajo@teststat[2], col="blue")
legend("topleft", c("critical values","test statistics"), lwd=2,col = c("red","blue"), bty="n")
```



For the first chart the test statistic is below the critical values for all levels (1%, 5%, 10%) meaning the null hypothesis (that the order of co-integration <=1) cannot be rejected. For the second chart the test statistic is above the critical velues at all levels, meaning the null hypothesis (that the order of co-integration = 0) is rejected.

Conclusion: the order of co-integration = 1

Make a forecast with the co-integration model. Our prediction equation can be written as:

$$\Delta X_t = \Gamma \Delta X_{t-1} + \Pi_1 X_{t-2} + \mu + \epsilon_t$$

```
#mu
(mu <-cajo@GAMMA[,1])
```

```
##      r_a.d      r_b.d
## 0.02314231 0.04538451
```

```
#Pi_1 (as opposed to just Pi)
(PI<-cajo@PI)
```

```
##            r_a.l2       r_b.l2
## r_a.d 0.005749287 -0.008173978
## r_b.d 0.034982766 -0.035653512
```

```
#Gamma
(Gamma<-cajo@GAMMA[,2:3])
```

```
##         r_a.dl1     r_b.dl1
## r_a.d 0.3799650 -0.05291384
## r_b.d 0.2610594  0.14777997
```

```
#delta Xt-1
dX_1 <- cajo@Z0
#matrix Xt-2
X_2 <- cajo@ZK
```

We call time t the forecast origin and 1 is the forecast horizon. Denote the forecast itself as: $E[\Delta\hat{X}_t(1)|t]$ Conditional expectation of the forecast

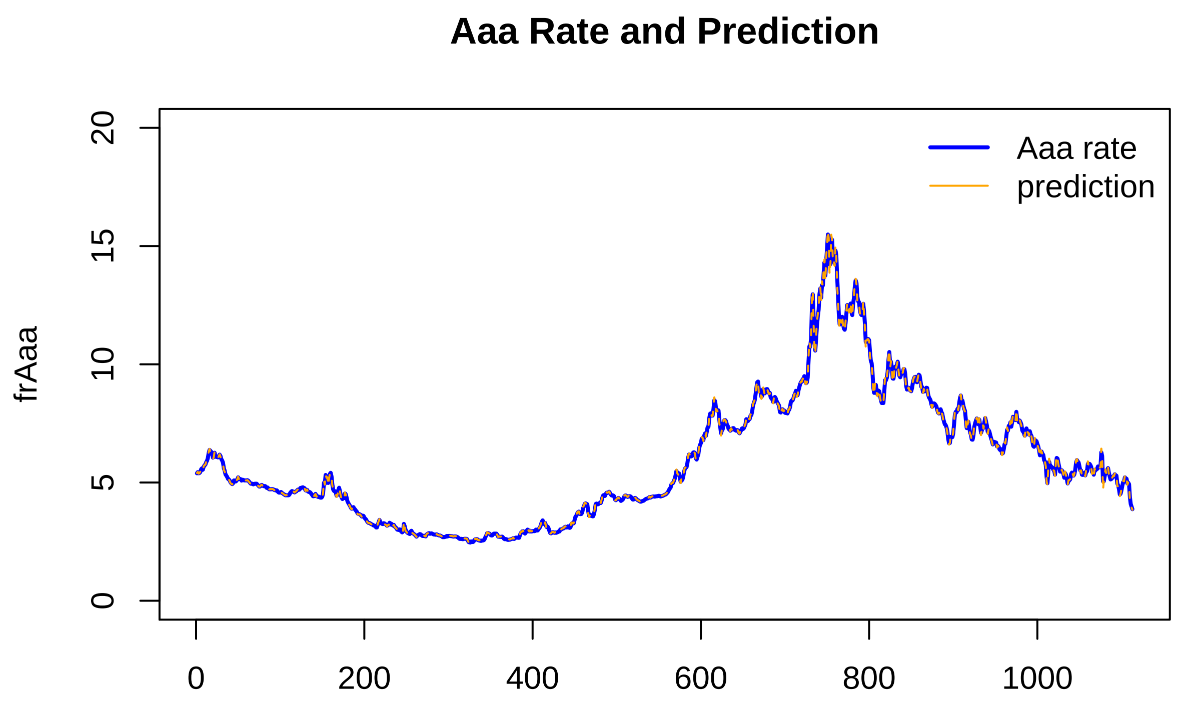
$$E[\Delta\hat{X}_t(1)|t] = \Gamma\Delta X_t + \Pi_1 X_{t-1} + \mu$$

Make the forecast.

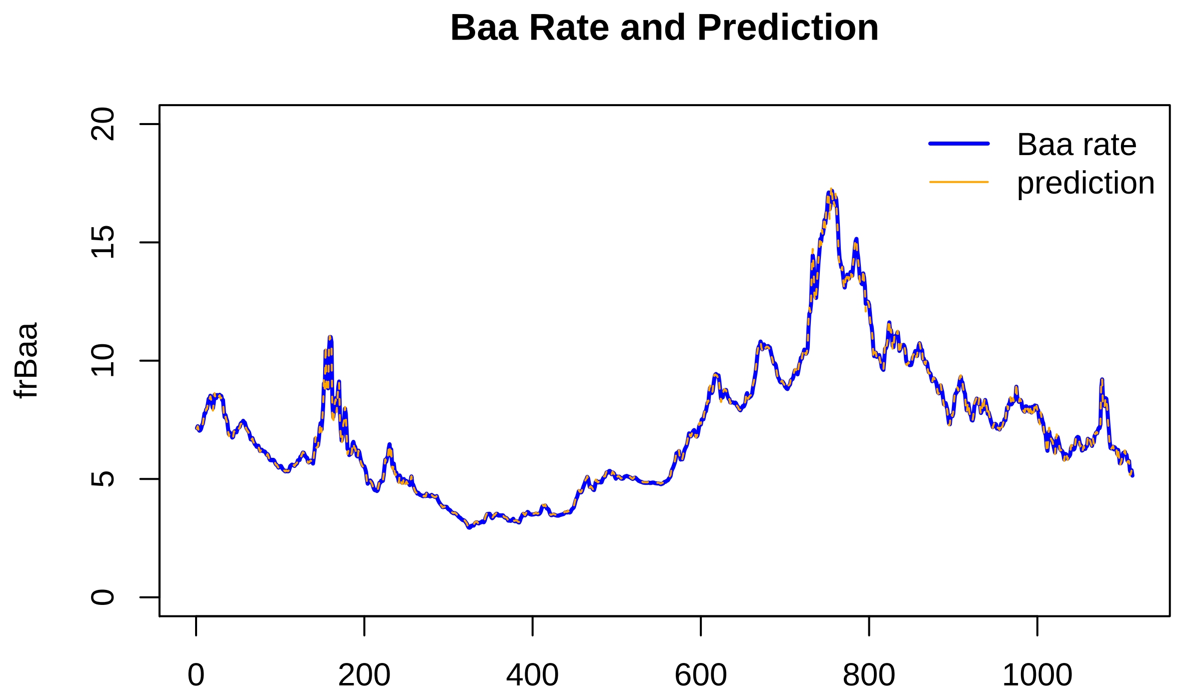
```
deltaX_t_1 <- Gamma %*% t(dX_1) + PI %*%t(X_2)
deltaX_t_1<-apply(deltaX_t_1,2,"+",mu)
nrowsdata <- dim(data)[1]
data_t_2 = data[3:nrowsdata,]
deltaX_t_1 <- t(deltaX_t_1)
forecX <- data_t_2+deltaX_t_1
```

How well does it work?

```
#for the Aaa rate
frAaa = cbind(r_a[3:length(r_a)],forecX[,1])
matplot(frAaa,col =c("blue","orange"),type="l",lwd=c(2,1), ylim=c(0,20), main = "Aaa Rate and Predictio
legend("topright", c("Aaa rate","prediction"), lwd=c(2,1), col = c("blue","orange"), bty="n")
```



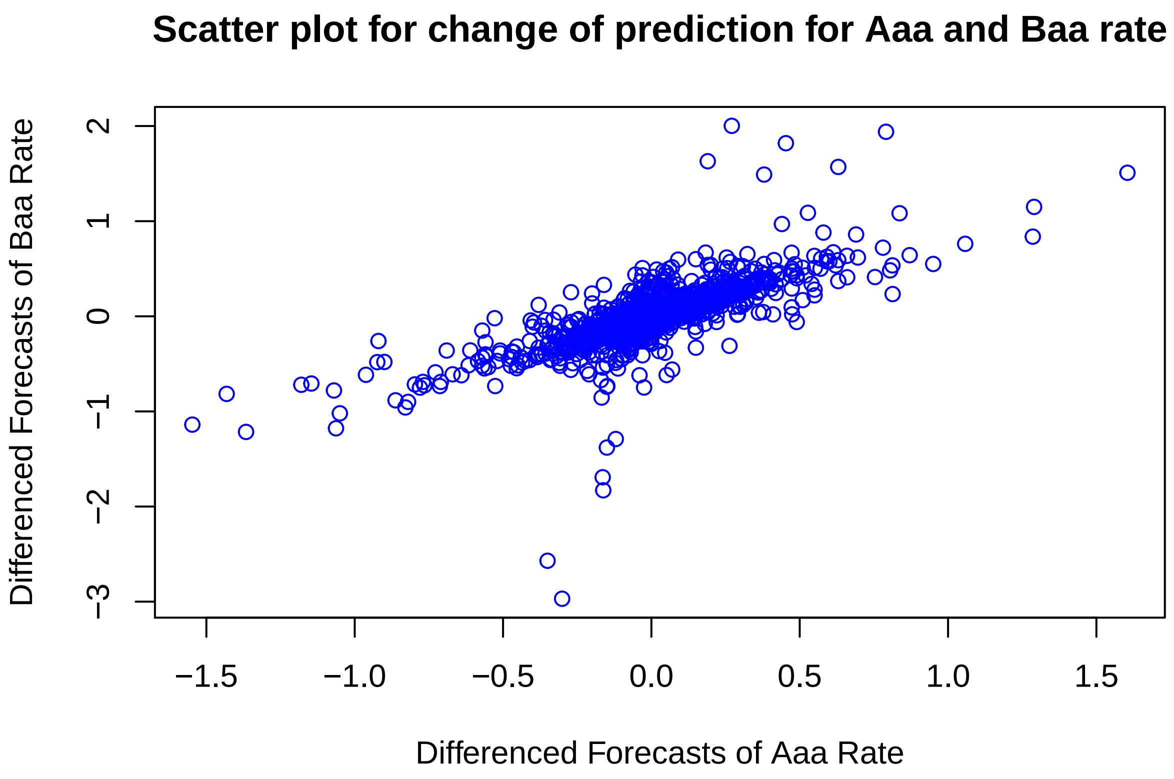
```r
#for the Baa rate
frBaa = cbind(r_b[3:length(r_b)],forecX[,2])
matplot(frBaa,col =c("blue","orange"),type="l",lwd=c(2,1), ylim=c(0,20), main = "Baa Rate and Predictio
legend("topright", c("Baa rate","prediction"), lwd=c(2,1), col = c("blue","orange"), bty="n")
```



Based on the above charts, we can say that our co-integration model preserves the long term dependence of the Aaa and Baa rates.

Does it preserve the short term dependence of the variables?

```r
df_frAaa<-diff(frAaa)
df_frBaa<-diff(frBaa)
plot(df_frAaa,df_frBaa,col ="blue",
     main = "Scatter plot for change of prediction for Aaa and Baa rate",
     xlab="Differenced Forecasts of Aaa Rate",ylab="Differenced Forecasts of Baa Rate")
```



Again this shows that the model more or less captures the short term dependence of the two variables, with some extreme values as exceptions.