GARCH Model GBP And CAD

Jane

28/04/2021

Forcasting Exchange Rate Using GARCH Model for Bristish Pound And canadian Dollar

Reading GBP and CAD Currency into r

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
GBPCADGARCH <-  read.csv ("GBPCAD_Candlestick_1_D_BID_01.01.2000-31.12.2020.csv")%>%
  select('GMT.TIME', CLOSE)%>%
  rename(Date = ('GMT.TIME'), RateGBPCAD = ("CLOSE"))


head(GBPCADGARCH)
```

```
##         Date RateGBPCAD
## 1 2000-01-03     2.3675
## 2 2000-01-04     2.3778
## 3 2000-01-05     2.3822
## 4 2000-01-06     2.4037
## 5 2000-01-07     2.3867
## 6 2000-01-10     2.3835
```

Conversion of Gmt time to date format

```r
library(dplyr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
GBPCADGARCH$Date <- lubridate::ymd(GBPCADGARCH$Date)
head(GBPCADGARCH)
```

```
##         Date RateGBPCAD
## 1 2000-01-03     2.3675
## 2 2000-01-04     2.3778
## 3 2000-01-05     2.3822
## 4 2000-01-06     2.4037
## 5 2000-01-07     2.3867
## 6 2000-01-10     2.3835
```

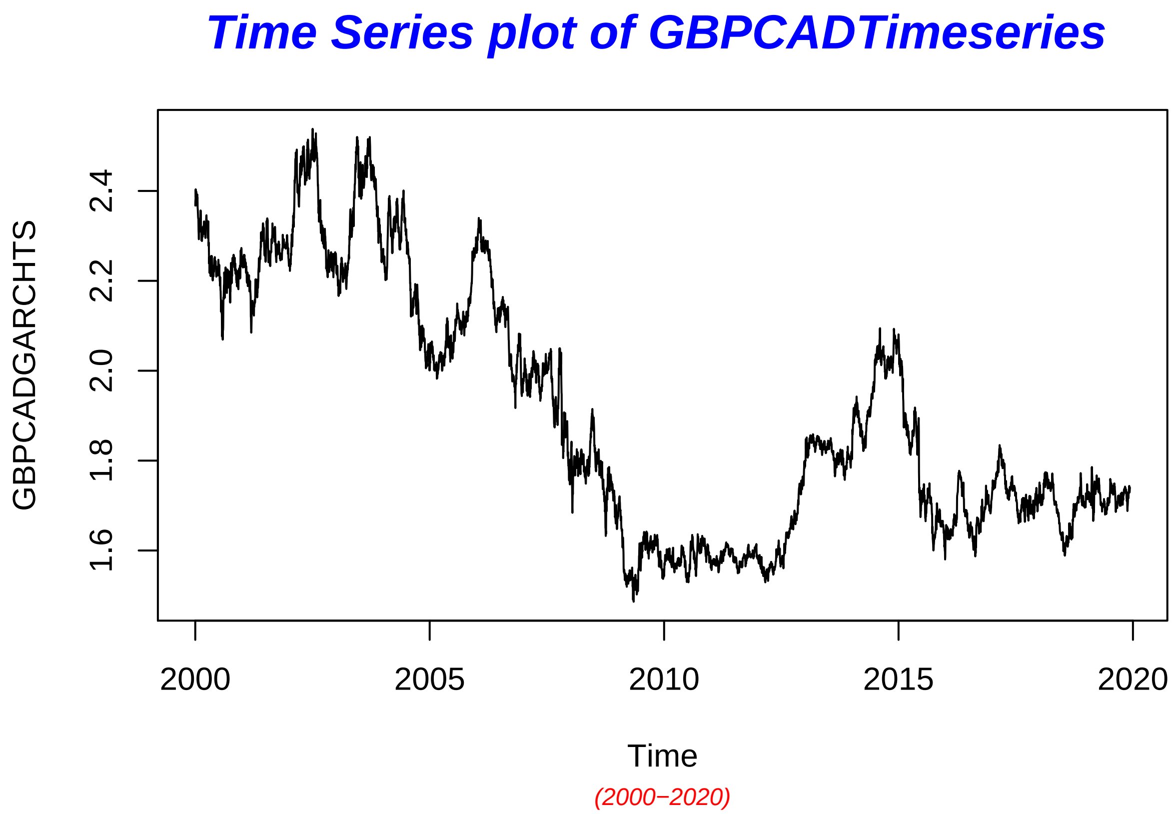
##Checking for obvious errors or missingg value

```r
#Checking for obvious errors
which(is.na(GBPCADGARCH))
```

```
## integer(0)
```

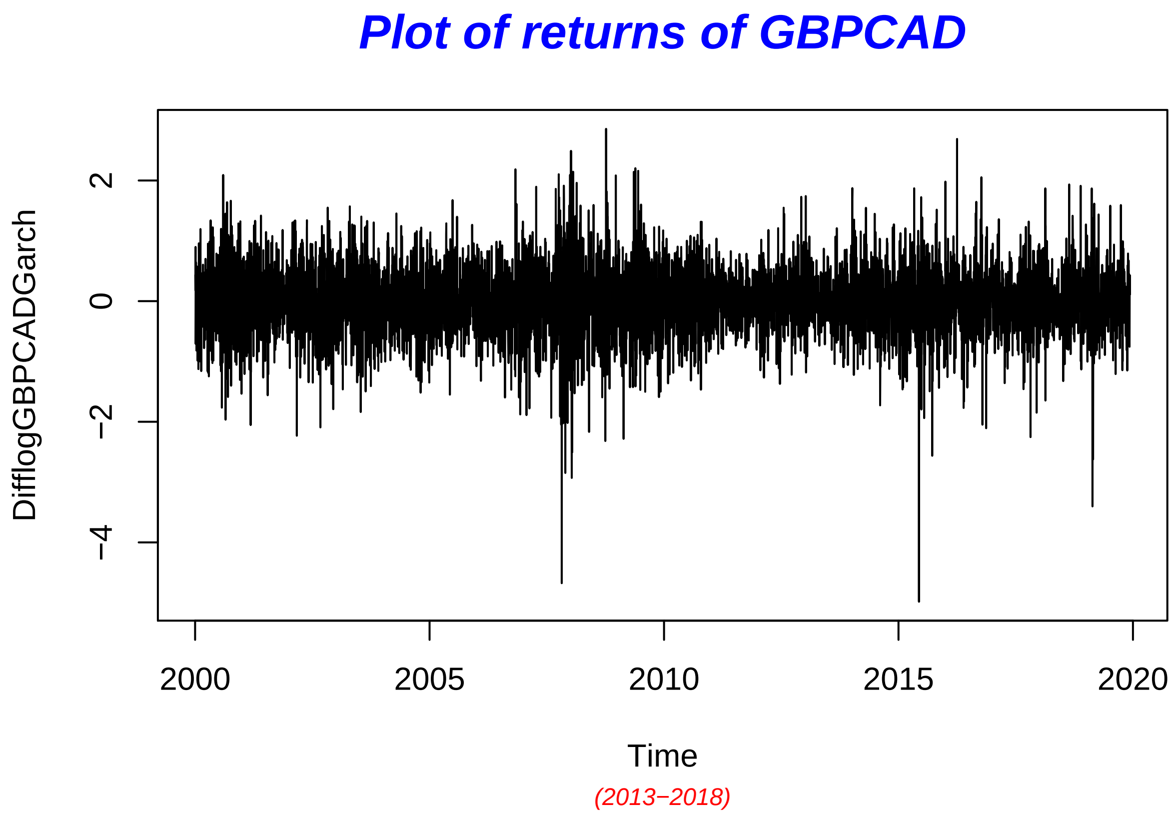
##Converting the data set into time series object

```r
#Converting the data set into time series object
GBPCADGARCHTS<- ts(as.vector(GBPCADGARCH$Rate),  frequency = 314, start= c(2000,01,03))
plot.ts(GBPCADGARCHTS)
title("Time Series plot of GBPCADTimeseries ", sub = "(2000-2020)",
      cex.main = 1.5,   font.main= 4, col.main= "blue",
      cex.sub = 0.75, font.sub = 3, col.sub = "red")
```



##Dealing with Conditional Heteroscedaticity:

```
DifflogGBPCADGarch= diff(log(GBPCADGARCHTS))*100
plot(DifflogGBPCADGarch)
title("Plot of returns of GBPCAD", sub = "(2013-2018)",
      cex.main = 1.5,   font.main= 4, col.main= "blue",
      cex.sub = 0.75, font.sub = 3, col.sub = "red")
```

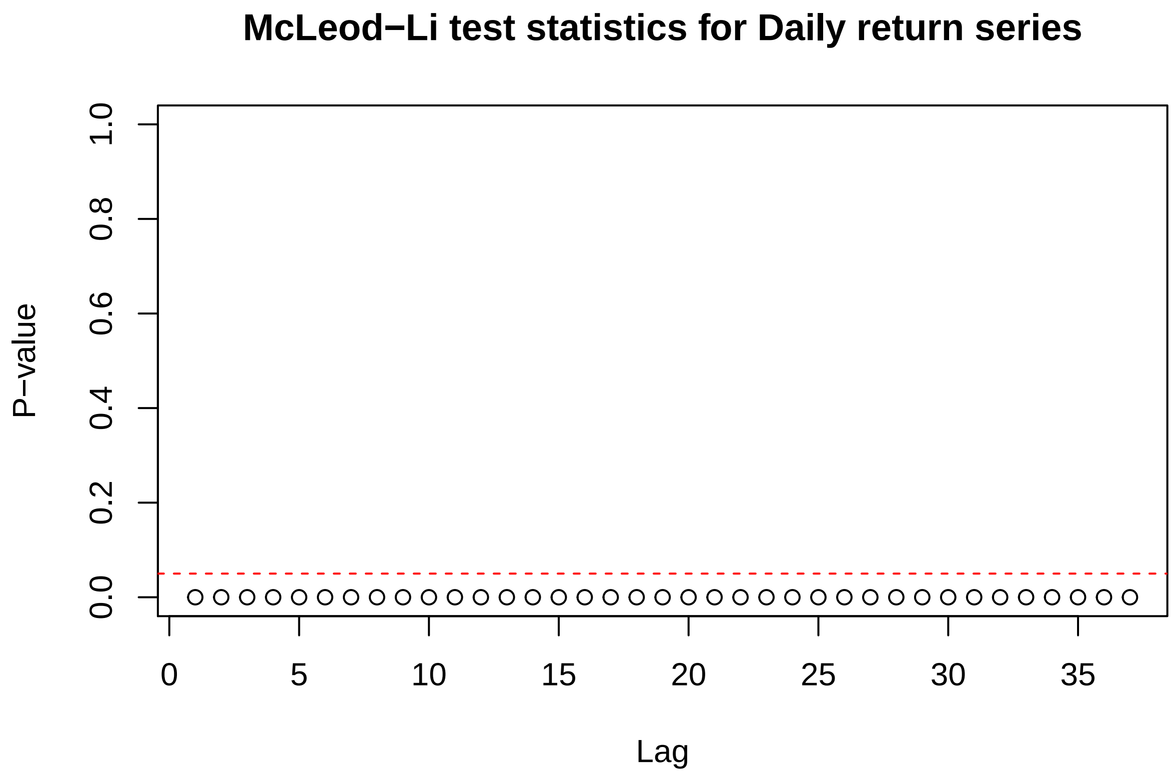


##nature as almost at all lags the p-values fall below the significance levels.

```
library(TSA)
```

```
## Warning: package 'TSA' was built under R version 4.0.5

##
## Attaching package: 'TSA'

## The following object is masked from 'package:readr':
##
##     spec

## The following objects are masked from 'package:stats':
##
##     acf, arima

## The following object is masked from 'package:utils':
##
##     tar
```

```
McLeod.Li.test(y= DifflogGBPCADGarch,main="McLeod-Li test statistics for Daily return series")
```

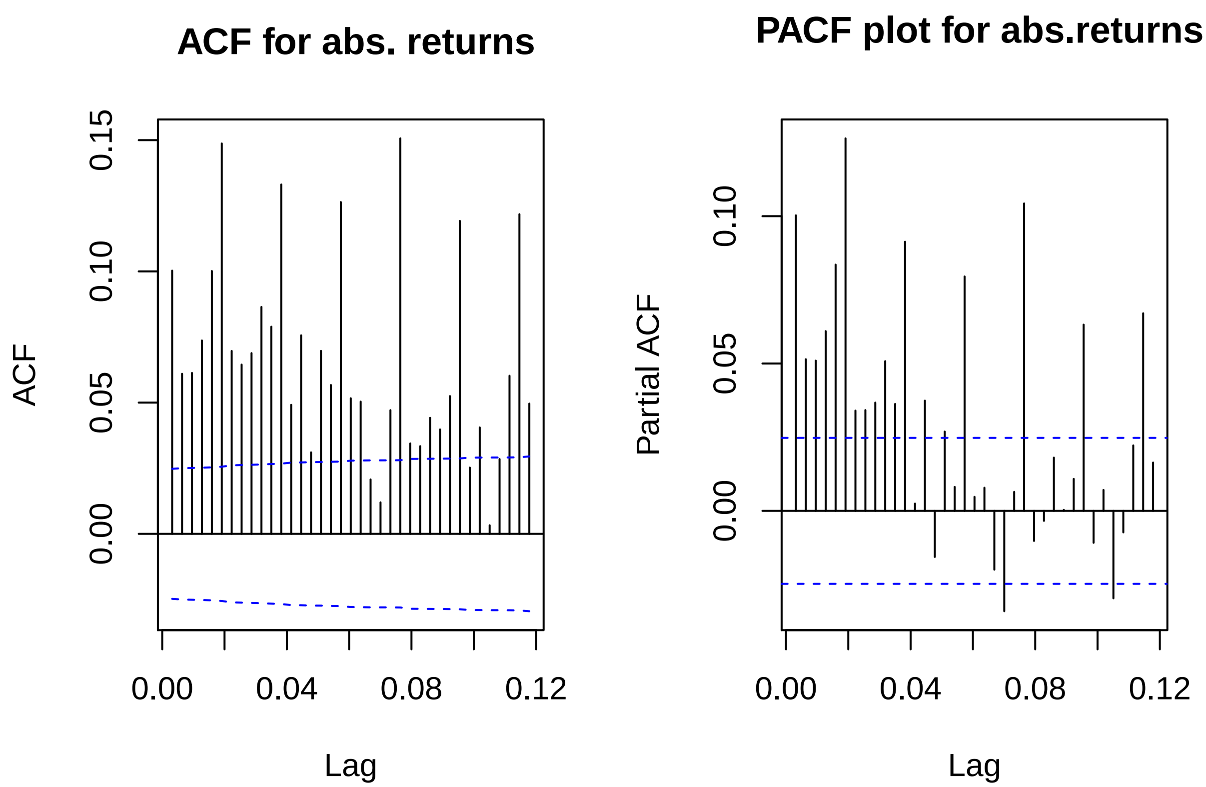


In order to get an order of GARCH , we further transform the return series into absolute values and squared return values.

```
abs = abs(DifflogGBPCADGarch)
sqr = DifflogGBPCADGarch^2
```

GARCH Model specification:

```
par(mfrow=c(1,2))
acf(abs, ci.type="ma",main=" ACF for abs. returns")
pacf(abs, main=" PACF plot for abs.returns")
```



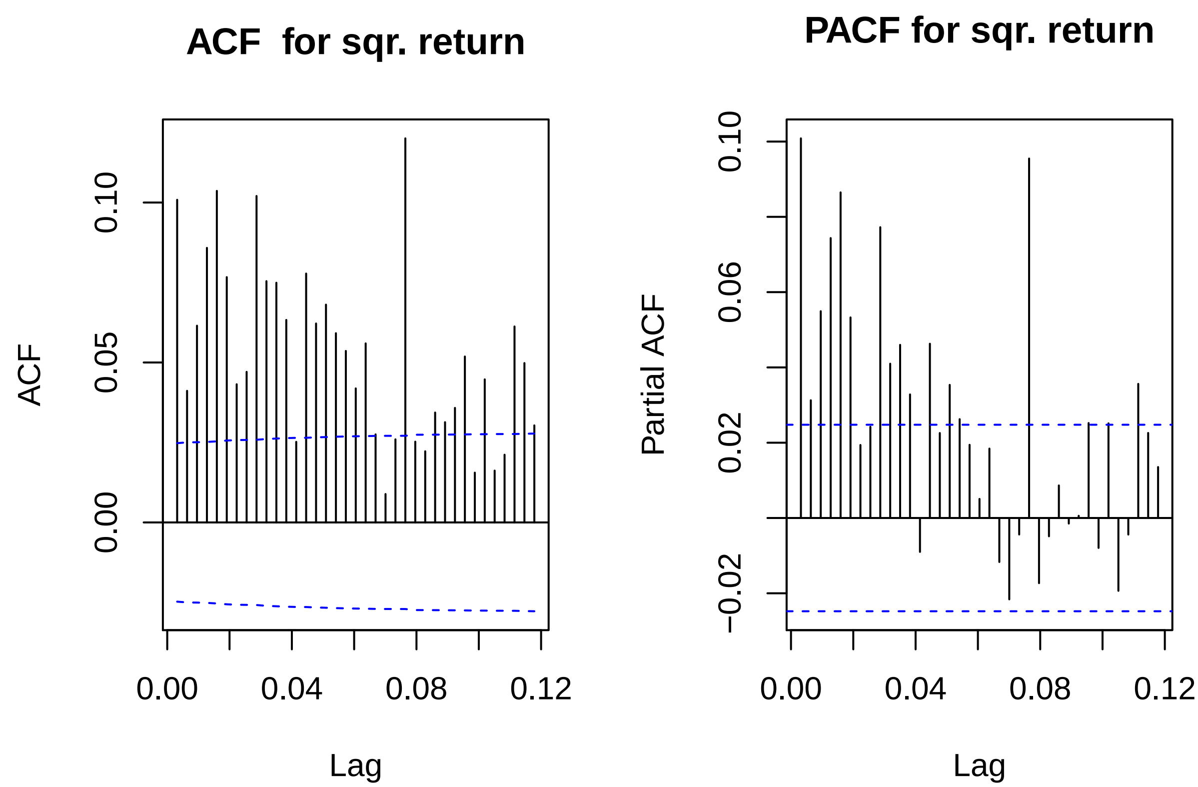
##From ACF and PACF we see many lags are significant. Hence, we plot EACF to get the candidate models

```
eacf(abs)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x x x x x x x x  x  x  x
## 1 x o o o o x x o o o o  x  x  x
## 2 x x o o o x x o o o o  x  o  o
## 3 x x o o o x x o o o o  x  o  o
## 4 x x o o o x x x o o o  x  x  o
## 5 x x x x x x o x o o o  x  x  o
## 6 x x x x x x o o o o o  o  o  o
## 7 x o x x x x x o o x o  o  o  o
```

##From the squared returns ACF and PACF plot, it is not that clear to derive the order of p and q. Hence, I approach EACF and the order of ARMA are ARMA (2,3), ARMA (3,3), ARMA (2,4). Thus, GARCH candidate models would be GARCH (3,2) GARCH (3,3) GARCH (4,2)

```
par(mfrow=c(1,2))
acf(sqr, ci.type="ma",main="ACF  for sqr. return")
pacf(sqr, main="PACF for sqr. return")
```




```r
eacf(sqr)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x x x x x x x x  x  o  x
## 1 x x o o x o o o x o o  x  o  x
## 2 x x o o o o o o x o o  o  x  x
## 3 x x o o o o o x x o o  o  x  x
## 4 x x x x o o o x x o o  o  x  x
## 5 x x x x x o o o x o o  o  x  x
## 6 x x x x o x o o x o o  o  o  o
## 7 x x x x o x o o o o o  o  o  o
```

With reference to the Dickey-Fuller Test, p-value is less than the 0.02 and we can reject the null hypothesis stating the non-stationarity. Hence , we can proceed further for model selection .

#MODEL ESTIMATION: ##GARCH (2,1): for GBP and CAD Curruency Pair

```r
# GARCH(2,1)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
GBPCADGARCHFit.21 = garch(DifflogGBPCADGarch,order=c(2,1),trace =FALSE)
summary(GBPCADGARCHFit.21)
```

```
##
## Call:
## garch(x = DifflogGBPCADGarch, order = c(2, 1), trace = FALSE)
##
## Model:
## GARCH(2,1)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -7.184615 -0.568618  0.009225  0.565076  5.723104
##
## Coefficient(s):
##     Estimate  Std. Error  t value Pr(>|t|)
## a0  0.006448    0.001154    5.587 2.31e-08 ***
## a1  0.063151    0.005357   11.787  < 2e-16 ***
## b1  0.446483    0.135483    3.295 0.000983 ***
## b2  0.470271    0.130204    3.612 0.000304 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Diagnostic Tests:
##  Jarque Bera Test
##
## data:  Residuals
## X-squared = 831.4, df = 2, p-value < 2.2e-16
##
##
##  Box-Ljung test
##
## data:  Squared.Residuals
## X-squared = 0.4531, df = 1, p-value = 0.5009
```

GARCH (2,2):

##This model can be interpreted as an overfit model of GARCH(2,1) and p values from residual tests confirms that residuals are highly correlated. Thus this model is not consider to be a good fit.

```r
GBPCADGARCHFit.22 = garch(DifflogGBPCADGarch, order =c(2,2),trace =FALSE)
summary(GBPCADGARCHFit.22)
```

```
##
## Call:
## garch(x = DifflogGBPCADGarch, order = c(2, 2), trace = FALSE)
##
## Model:
```

```
## GARCH(2,2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -7.132436 -0.568409  0.009232  0.565129  5.720988
##
## Coefficient(s):
##     Estimate  Std. Error  t value Pr(>|t|)
## a0 6.758e-03   1.464e-03    4.615 3.93e-06 ***
## a1 6.714e-02   7.451e-03    9.011  < 2e-16 ***
## a2 8.539e-15   1.410e-02    0.000 1.000000
## b1 3.359e-01   1.738e-01    1.933 0.053286 .
## b2 5.761e-01   1.630e-01    3.535 0.000408 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Diagnostic Tests:
##  Jarque Bera Test
##
## data:  Residuals
## X-squared = 818.96, df = 2, p-value < 2.2e-16
##
##
##  Box-Ljung test
##
## data:  Squared.Residuals
## X-squared = 0.21917, df = 1, p-value = 0.6397
```

##GARCH (3,1): ##This model can be interpreted as an overfit model of GARCH(2,1) and GARCH (2,2). This model may not be consider to be a good fit.

```
GBPCADGARCHFit.31 = garch(DifflogGBPCADGarch,order=c(3,1),trace =FALSE)
summary(GBPCADGARCHFit.31)
```

```
##
## Call:
## garch(x = DifflogGBPCADGarch, order = c(3, 1), trace = FALSE)
##
## Model:
## GARCH(3,1)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -6.869099 -0.569442  0.008929  0.564771  5.752239
##
## Coefficient(s):
##     Estimate  Std. Error  t value Pr(>|t|)
## a0 7.629e-03   1.366e-03    5.587 2.31e-08 ***
## a1 7.966e-02   6.026e-03   13.218  < 2e-16 ***
## b1 4.271e-01   9.301e-02    4.593 4.38e-06 ***
## b2 2.583e-14   1.103e-01    0.000        1
## b3 4.696e-01   8.748e-02    5.368 7.94e-08 ***
## ---
```

```
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Diagnostic Tests:
##  Jarque Bera Test
##
## data:  Residuals
## X-squared = 766.05, df = 2, p-value < 2.2e-16
##
##
##  Box-Ljung test
##
## data:  Squared.Residuals
## X-squared = 0.028104, df = 1, p-value = 0.8669
```

##GARCH (3,2): ##This model can be interpreted as an overfitting model and p values from residual tests confirms that residuals are highly correlated. Thus this model is not consider to be a good fit.

GARCH(3,2)

```
GBPCADGARCHFit.32 = garch(DifflogGBPCADGarch,order=c(3,2),trace =FALSE)
summary(GBPCADGARCHFit.32)
```

```
##
## Call:
## garch(x = DifflogGBPCADGarch, order = c(3, 2), trace = FALSE)
##
## Model:
## GARCH(3,2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.86127 -0.56894  0.00887  0.56388  5.74983
##
## Coefficient(s):
##     Estimate  Std. Error  t value Pr(>|t|)
## a0 8.232e-03   2.136e-03    3.854 0.000116 ***
## a1 8.342e-02   8.347e-03    9.994  < 2e-16 ***
## a2 3.167e-15   1.998e-02    0.000 1.000000
## b1 3.802e-01   2.056e-01    1.849 0.064497 .
## b2 1.088e-01   1.357e-01    0.802 0.422571
## b3 4.023e-01   1.412e-01    2.850 0.004378 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Diagnostic Tests:
##  Jarque Bera Test
##
## data:  Residuals
## X-squared = 769.95, df = 2, p-value < 2.2e-16
##
##
```

```
##  Box-Ljung test
##
## data:  Squared.Residuals
## X-squared = 0.11306, df = 1, p-value = 0.7367
```

GARCH (3,3):

This model can be interpreted as an overfitting model and p values from residual tests confirms that residuals are highly correlated. Thus, this model is not consider to be a good fit.

GARCH(3,3)

```r
GBPCADGARCHFit.33 = garch(DifflogGBPCADGarch,order=c(3,3),trace =FALSE)
summary(GBPCADGARCHFit.33)
```

```
##
## Call:
## garch(x = DifflogGBPCADGarch, order = c(3, 3), trace = FALSE)
##
## Model:
## GARCH(3,3)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -6.719741 -0.568636  0.008977  0.560144  5.751115
##
## Coefficient(s):
##     Estimate  Std. Error  t value Pr(>|t|)
## a0 1.018e-02   3.421e-03    2.975  0.00293 **
## a1 9.714e-02   9.201e-03   10.557  < 2e-16 ***
## a2 1.030e-03   5.361e-02    0.019  0.98467
## a3 7.921e-16   3.100e-02    0.000  1.00000
## b1 2.623e-01   4.995e-01    0.525  0.59941
## b2 2.894e-01   4.555e-01    0.635  0.52522
## b3 3.198e-01   1.413e-01    2.263  0.02365 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Diagnostic Tests:
##  Jarque Bera Test
##
## data:  Residuals
## X-squared = 756.89, df = 2, p-value < 2.2e-16
##
##
##  Box-Ljung test
##
## data:  Squared.Residuals
## X-squared = 0.84823, df = 1, p-value = 0.3571
```

##GARCH (4,2): ##This model can be interpreted as an overfitting model and p values from residual tests confirms that residuals are highly correlated. Thus, this model is not considered to be a good fit.

```
GBPCADGARCHFit.42 = garch(DifflogGBPCADGarch,order=c(4,2),trace =FALSE)
summary(GBPCADGARCHFit.42)
```

```
##
## Call:
## garch(x = DifflogGBPCADGarch, order = c(4, 2), trace = FALSE)
##
## Model:
## GARCH(4,2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -6.745367 -0.567743  0.009288  0.563699  5.709719
##
## Coefficient(s):
##     Estimate  Std. Error  t value Pr(>|t|)
## a0  0.009170    0.004487    2.044  0.04097 *
## a1  0.089631    0.008846   10.133  < 2e-16 ***
## a2  0.005132    0.046206    0.111  0.91156
## b1  0.235647    0.540111    0.436  0.66262
## b2  0.097458    0.214671    0.454  0.64984
## b3  0.327272    0.113209    2.891  0.00384 **
## b4  0.216316    0.241077    0.897  0.36956
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Diagnostic Tests:
##  Jarque Bera Test
##
## data:  Residuals
## X-squared = 745.41, df = 2, p-value < 2.2e-16
##
##
##  Box-Ljung test
##
## data:  Squared.Residuals
## X-squared = 0.33378, df = 1, p-value = 0.5634
```

Model Selection:

##Best possible model is selected by AIC scores of the models. From the below sort function, GARCH(3,1) would be the best model for the return series. From the p-value, 3.1 also has the lowest correlation

```
library(dLagM)
```

```
## Warning: package 'dLagM' was built under R version 4.0.5
```

```
## Loading required package: nardl
```

```
## Warning: package 'nardl' was built under R version 4.0.5

## Loading required package: dynlm

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
GARCHModelSelectionGBPCAD = AIC(GBPCADGARCHFit.21,GBPCADGARCHFit.22 ,GBPCADGARCHFit.31,GBPCADGARCHFit.32
sortScore(GARCHModelSelectionGBPCAD, score ="aic")
```

```
##                    df      AIC
## GBPCADGARCHFit.31  5 9979.524
## GBPCADGARCHFit.42  7 9979.598
## GBPCADGARCHFit.32  6 9982.103
## GBPCADGARCHFit.21  4 9987.087
## GBPCADGARCHFit.33  7 9988.882
## GBPCADGARCHFit.22  5 9989.913
```

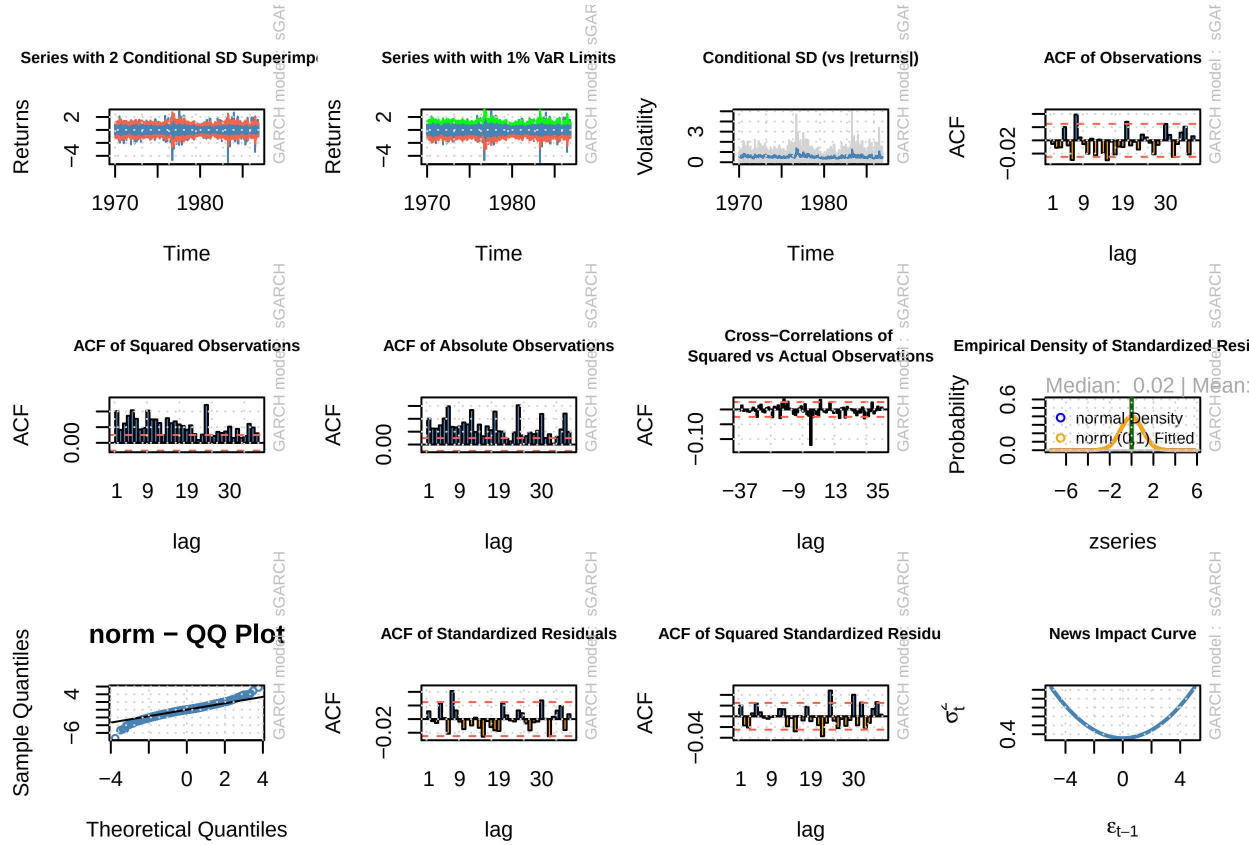
Model Fitting:

```
library(rugarch)
```

```
## Warning: package 'rugarch' was built under R version 4.0.5

## Loading required package: parallel

##
## Attaching package: 'rugarch'

## The following object is masked from 'package:stats':
##
##     sigma
```

```
GBPCADmodel3.1<-ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(3,1)),
                  mean.model = list(armaOrder = c(1, 1), include.mean = TRUE),
                  distribution.model = "norm")

GBPCADgarchMODEL3.1<-ugarchfit(spec=GBPCADmodel3.1,data=DifflogGBPCADGarch, out.sample = 100)
plot(GBPCADgarchMODEL3.1,which="all")
```

```
##
## please wait...calculating quantiles...
```



##Model Diagnostics

```
GBPCADgarchMODEL3.1
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model   : sGARCH(3,1)
## Mean Model    : ARFIMA(1,0,1)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error   t value Pr(>|t|)
## mu     -0.004324    0.006220 -0.695132 0.486973
## ar1     0.569569    0.234535  2.428502 0.015161
## ma1    -0.591873    0.229894 -2.574550 0.010037
## omega   0.004688    0.002616  1.792137 0.073111
## alpha1  0.045755    0.011669  3.920977 0.000088
## alpha2  0.000000    0.019909  0.000003 0.999998
## alpha3  0.000000    0.019200  0.000009 0.999993
## beta1   0.939803    0.021172 44.388379 0.000000
```

```
##
## Robust Standard Errors:
##         Estimate  Std. Error   t value Pr(>|t|)
## mu     -0.004324    0.007014 -0.616465 0.537588
## ar1     0.569569    0.172933  3.293573 0.000989
## ma1    -0.591873    0.170122 -3.479105 0.000503
## omega   0.004688    0.010215  0.458949 0.646271
## alpha1  0.045755    0.020791  2.200753 0.027753
## alpha2  0.000000    0.028230  0.000002 0.999998
## alpha3  0.000000    0.049337  0.000003 0.999997
## beta1   0.939803    0.085668 10.970286 0.000000
##
## LogLikelihood : -4924.072
##
## Information Criteria
## ------------------------------------
##
## Akaike       1.6013
## Bayes        1.6101
## Shibata      1.6013
## Hannan-Quinn 1.6044
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.8079 0.36876
## Lag[2*(p+q)+(p+q)-1][5]    2.3103 0.86773
## Lag[4*(p+q)+(p+q)-1][9]    8.2415 0.04695
## d.o.f=2
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                          statistic p-value
## Lag[1]                       2.468  0.1162
## Lag[2*(p+q)+(p+q)-1][11]     8.996  0.1453
## Lag[4*(p+q)+(p+q)-1][19]    13.211  0.1814
## d.o.f=4
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[5]     3.476 0.500 2.000 0.06227
## ARCH Lag[7]     3.803 1.473 1.746 0.21866
## ARCH Lag[9]     3.911 2.402 1.619 0.41073
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  3.9942
## Individual Statistics:
## mu     0.07659
## ar1    0.07397
## ma1    0.07733
## omega  0.23237
```

```
## alpha1 0.12325
## alpha2 0.12961
## alpha3 0.17861
## beta1  0.20643
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.89 2.11 2.59
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias           0.1932 0.8468
## Negative Sign Bias  1.6180 0.1057
## Positive Sign Bias  0.4814 0.6303
## Joint Effect        3.1783 0.3649
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     207.4    1.145e-33
## 2    30     226.4    1.837e-32
## 3    40     241.0    6.335e-31
## 4    50     260.7    1.179e-30
##
##
## Elapsed time : 0.545969
```

Forecasting

```
forcgarchGBPCAD= ugarchforecast(GBPCADgarchMODEL3.1, data = DiffGBPCADLogTran, n.ahead = 100, n.roll =1
print(forcgarchGBPCAD)
```

```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: sGARCH
## Horizon: 100
## Roll Steps: 10
## Out of Sample: 100
##
## 0-roll forecast [T0=1986-11-13 02:00:00]:
##          Series  Sigma
## T+1    0.005083 0.4693
## T+2    0.001034 0.4709
## T+3   -0.001272 0.4725
## T+4   -0.002586 0.4741
## T+5   -0.003334 0.4756
## T+6   -0.003760 0.4771
## T+7   -0.004003 0.4785
```

```
## T+8   -0.004141 0.4800
## T+9   -0.004220 0.4814
## T+10  -0.004264 0.4828
## T+11  -0.004290 0.4842
## T+12  -0.004305 0.4855
## T+13  -0.004313 0.4868
## T+14  -0.004318 0.4881
## T+15  -0.004320 0.4894
## T+16  -0.004322 0.4906
## T+17  -0.004323 0.4919
## T+18  -0.004323 0.4931
## T+19  -0.004323 0.4943
## T+20  -0.004324 0.4955
## T+21  -0.004324 0.4966
## T+22  -0.004324 0.4977
## T+23  -0.004324 0.4989
## T+24  -0.004324 0.5000
## T+25  -0.004324 0.5010
## T+26  -0.004324 0.5021
## T+27  -0.004324 0.5031
## T+28  -0.004324 0.5042
## T+29  -0.004324 0.5052
## T+30  -0.004324 0.5062
## T+31  -0.004324 0.5071
## T+32  -0.004324 0.5081
## T+33  -0.004324 0.5090
## T+34  -0.004324 0.5100
## T+35  -0.004324 0.5109
## T+36  -0.004324 0.5118
## T+37  -0.004324 0.5127
## T+38  -0.004324 0.5135
## T+39  -0.004324 0.5144
## T+40  -0.004324 0.5152
## T+41  -0.004324 0.5161
## T+42  -0.004324 0.5169
## T+43  -0.004324 0.5177
## T+44  -0.004324 0.5185
## T+45  -0.004324 0.5192
## T+46  -0.004324 0.5200
## T+47  -0.004324 0.5208
## T+48  -0.004324 0.5215
## T+49  -0.004324 0.5222
## T+50  -0.004324 0.5229
## T+51  -0.004324 0.5237
## T+52  -0.004324 0.5243
## T+53  -0.004324 0.5250
## T+54  -0.004324 0.5257
## T+55  -0.004324 0.5264
## T+56  -0.004324 0.5270
## T+57  -0.004324 0.5277
## T+58  -0.004324 0.5283
## T+59  -0.004324 0.5289
## T+60  -0.004324 0.5295
## T+61  -0.004324 0.5301
```

```
## T+62  -0.004324 0.5307
## T+63  -0.004324 0.5313
## T+64  -0.004324 0.5319
## T+65  -0.004324 0.5325
## T+66  -0.004324 0.5330
## T+67  -0.004324 0.5336
## T+68  -0.004324 0.5341
## T+69  -0.004324 0.5346
## T+70  -0.004324 0.5352
## T+71  -0.004324 0.5357
## T+72  -0.004324 0.5362
## T+73  -0.004324 0.5367
## T+74  -0.004324 0.5372
## T+75  -0.004324 0.5377
## T+76  -0.004324 0.5381
## T+77  -0.004324 0.5386
## T+78  -0.004324 0.5391
## T+79  -0.004324 0.5395
## T+80  -0.004324 0.5400
## T+81  -0.004324 0.5404
## T+82  -0.004324 0.5408
## T+83  -0.004324 0.5413
## T+84  -0.004324 0.5417
## T+85  -0.004324 0.5421
## T+86  -0.004324 0.5425
## T+87  -0.004324 0.5429
## T+88  -0.004324 0.5433
## T+89  -0.004324 0.5437
## T+90  -0.004324 0.5441
## T+91  -0.004324 0.5445
## T+92  -0.004324 0.5449
## T+93  -0.004324 0.5452
## T+94  -0.004324 0.5456
## T+95  -0.004324 0.5459
## T+96  -0.004324 0.5463
## T+97  -0.004324 0.5466
## T+98  -0.004324 0.5470
## T+99  -0.004324 0.5473
## T+100 -0.004324 0.5476
```

Forecasting the rate

```
p.t_1 = 1.73964
  R_t <- c( 0.005083, 0.001034, -0.001272, -0.002586, -0.003334, -0.003760, -0.004003, -0.004141, -0.004
)
  p_t= 0
  for (i in 1:100){
    p_t =  p.t_1  *((2.71828)^(R_t[i]/100))
   print(p_t)
    p.t_1=p_t
  }
```

```
## [1] 1.739728
## [1] 1.739746
## [1] 1.739724
## [1] 1.739679
## [1] 1.739621
## [1] 1.739556
## [1] 1.739486
## [1] 1.739414
## [1] 1.739341
## [1] 1.739267
## [1] 1.739192
## [1] 1.739117
## [1] 1.739042
## [1] 1.738967
## [1] 1.738892
## [1] 1.738817
## [1] 1.738742
## [1] 1.738666
## [1] 1.738591
## [1] 1.738516
## [1] 1.738441
## [1] 1.738366
## [1] 1.738291
## [1] 1.738215
## [1] 1.73814
## [1] 1.738065
## [1] 1.73799
## [1] 1.737915
## [1] 1.73784
## [1] 1.737765
## [1] 1.737689
## [1] 1.737614
## [1] 1.737539
## [1] 1.737464
## [1] 1.737389
## [1] 1.737314
## [1] 1.737239
## [1] 1.737164
## [1] 1.737088
## [1] 1.737013
## [1] 1.736938
## [1] 1.736863
## [1] 1.736788
## [1] 1.736713
## [1] 1.736638
## [1] 1.736563
## [1] 1.736488
## [1] 1.736413
## [1] 1.736337
## [1] 1.736262
## [1] 1.736187
## [1] 1.736112
## [1] 1.736037
## [1] 1.735962
```

```
## [1] 1.735887
## [1] 1.735812
## [1] 1.735737
## [1] 1.735662
## [1] 1.735587
## [1] 1.735512
## [1] 1.735437
## [1] 1.735362
## [1] 1.735287
## [1] 1.735212
## [1] 1.735137
## [1] 1.735062
## [1] 1.734987
## [1] 1.734912
## [1] 1.734837
## [1] 1.734762
## [1] 1.734687
## [1] 1.734612
## [1] 1.734537
## [1] 1.734462
## [1] 1.734387
## [1] 1.734312
## [1] 1.734237
## [1] 1.734162
## [1] 1.734087
## [1] 1.734012
## [1] 1.733937
## [1] 1.733862
## [1] 1.733787
## [1] 1.733712
## [1] 1.733637
## [1] 1.733562
## [1] 1.733487
## [1] 1.733412
## [1] 1.733337
## [1] 1.733262
## [1] 1.733187
## [1] 1.733112
## [1] 1.733037
## [1] 1.732962
## [1] 1.732887
## [1] 1.732812
## [1] 1.732737
## [1] 1.732662
## [1] 1.732588
## [1] 1.732513
```

```
RateGBPJPYGarch = 141.168
  RGBPJPYGARCH <-c(0.007108, 0.008990, 0.009627, 0.009843, 0.009916, 0.009940, 0.009949, 0.009951, 0.009

)
  GBPJPYgarch= 0
  for (i in 1:100){
    USCanadagarch =  RateGBPJPYGarch *((2.71828)^(RGBPJPYGARCH[i]/100))
```

```r
    print(GBPJPYgarch)
    RateUSGBPJPY=GBPJPYgarch
  }
```

```
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
```

```
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
```