Time Series Analysis of Apple Stock Prices Using Hidden Markov models

by

Remya Kannan

Data Description and Research Objective:

The motivation of this study is to analyze the trend in financial data. For this purpose, I collected the data from the daily historical Apple stock prices (open, high, low, close, and adjusted prices) from February 1, 2002 to January 31, 2017 extracted from the Yahoo Finance website.

The data has logged the prices of the Apple stock every day and comprises of the open, close, low, high and the adjusted close prices of the stock for the span of 15 years. The goal of the study is to discover an interesting trend in the apple stock prices over the past 15 years (3775 attributes) and to design and develop the best model for forecasting.

Non-Technical Summary:

The stock market is one of the most vital areas of a market economy. It provides companies with the access to capital by allowing investors to buy shares of ownership of the company, while allowing investors to profit from the company's future prosperity. Although there are millions gained by buying shares and then selling them for a profit, not all investors are successful in gaining a profitable return on their investments. This is because, the stock prices are continuously fluctuating at a given moment. A possible solution for the investors, is to sell their shares before the value diminishes than when it was purchased.

In this study, the technical data on the historical Apple stock prices are collected from the Yahoo Finance website. I chose this stock mainly because it is popular and there is a large amount of information that is relevant to the research and can help evaluate the analysis to make a better-informed decision.

After the initial data cleaning and preprocessing, we first explore the data to get a deeper understanding of the parameters that influence the modelling process. We then apply the Hidden Markov Model (HMM) analysis on the stock prices of aapl to obtain forecasting results. We chose HMM for analysis as it works well with volatility clustering, a phenomenon where there are periods of spikes and fall in volatility, typical of market data.

This study is structured as follows: We proceed with the technical aspects of the analysis by explaining the exploratory analysis performed on the data in section 1, section 2 discusses in detail the model fitting process, section 3 explains the residual analysis and model diagnostics towards model evaluation to choose the best model and section 4 gives an analysis of our results discussing the statistical significance of the results.

Technical Summary:

1. Exploratory Analysis:

Apple daily stock price

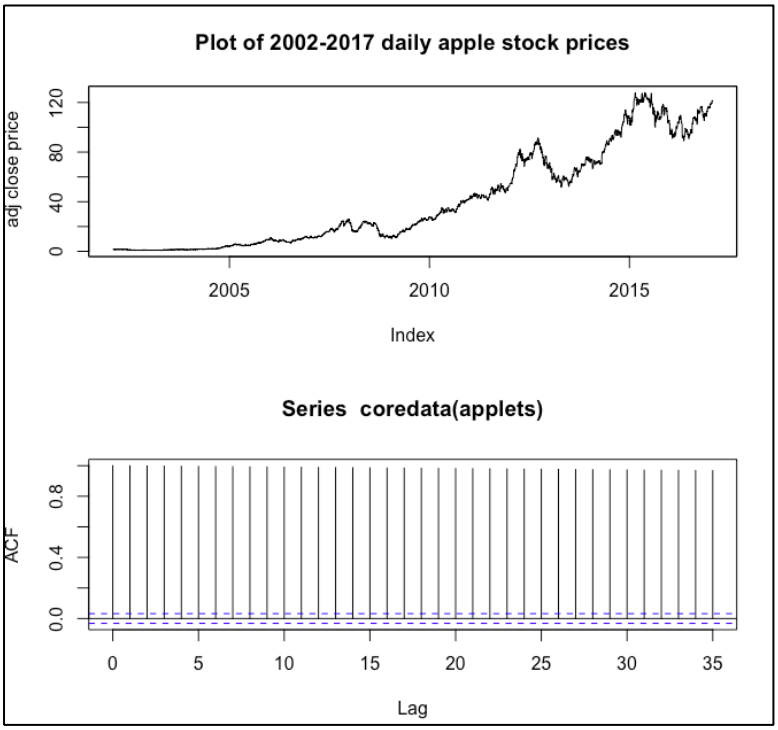


Fig.1. Time plot for the daily apple stock prices (2002-2017)

From the time plot shown in Fig.1, we observe that the daily apple stock prices in dollar from first day of February 2002 to the last day of January 2017 has a clear non-linear and upward trend. From 2002-2005, the prices show little growth and has a flatter trend. However, the stock prices start growing rapidly and the prices increase more dramatically from 2006 to 2013 followed by the rapid decrease on 2014. In the same year, the growth recovers and the upward trend of price continue till 2015 reaching the highest peak. We observe the apple stock prices fall in 2008, 2014 and 2016. There could many reasons why this happened. In the year 2008, Apple stocks had a market meltdown with an all-time low stock prices and was forever remembered as the day the market lost faith in Apple! There are theories on the effects of recession and the company's growth wasn't significant enough to contribute to the stock prices. Whereas, in the years 2014 and 2016, the largest market for the Apple users, China, had all Apple products go dark following the censorship order of the government.

Overall, the upward trend indicates that the mean changes over time, and the multiple jumps indicate that the variance is not constant. Both are signs of the process being non-stationary, moreover, another sign of the non-stationary process can be detected from the ACF plot: as we observe that the ACF value do not decay to zero meaning the shock affects the process permanently.

Therefore, based on the time plot and ACF plot of the data, we determine that analyzing the daily return (log return) of stock price is an appropriate approach for further analysis.

Apple daily return (log return) of stock price

- ***Normality Analysis***

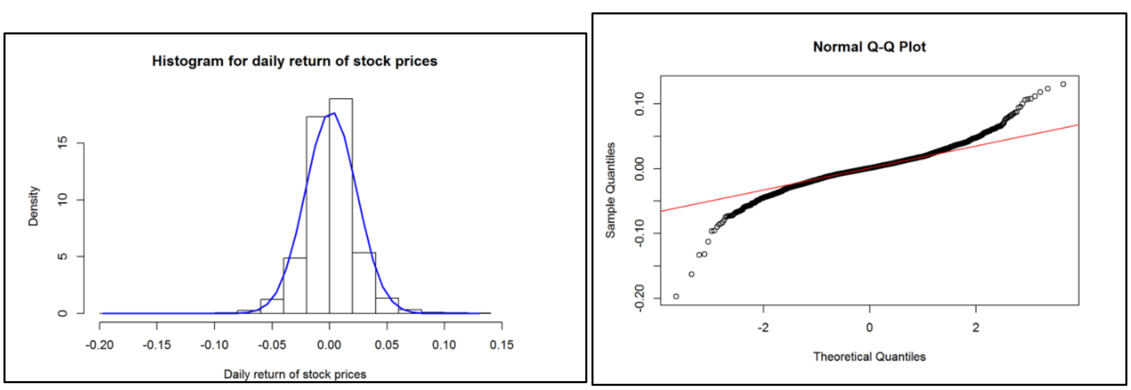


Fig.2. Histogram and QQ-plot for Normality test

The daily return of apple stock prices has zero mean and a large kurtosis value (5.411) indicating that the distribution has fat tails (**Appendix A1**) and hence is not normal. To confirm our observation, the normal probability plot, histogram, and the normality test (**Appendix A2**) is conducted. From the normal probability plot and histogram in Fig.2, we observe fat tails on both the ends. The normality test (Jarque-Bera test at 5% significance level) suggest that we reject the null hypothesis of the data being normally distributed and we may conclude that the distribution of apple daily stock return is not normal.

- ***Analysis of Apple stock return***

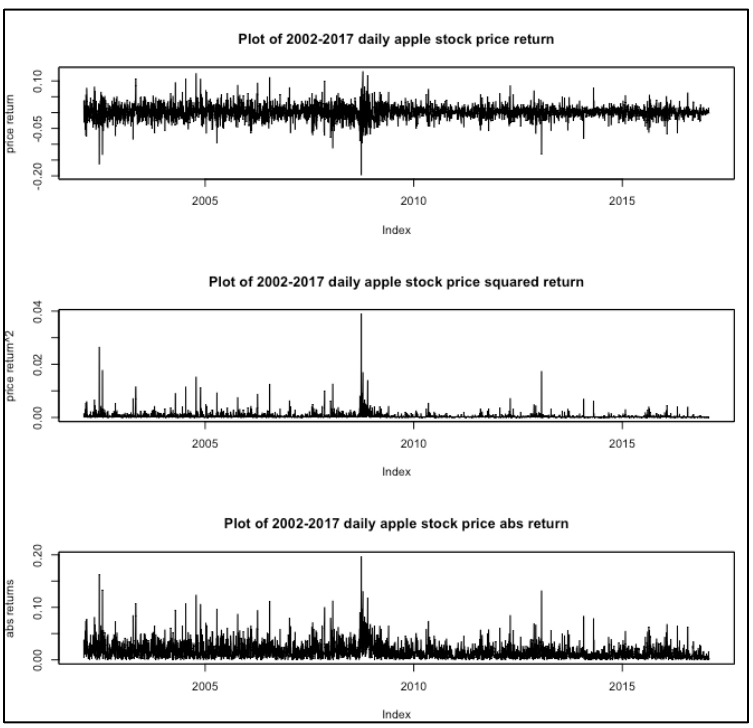


Fig.3. Time plot for log return of the Apple stock prices

The time plots in Fig.3, show returns varying around the zero line with an extremely large log return around 2002(-0.15), 2008(-0.20) and 2013(-0.15) followed by a high volatility period. Conditional volatility is non-constant over time with periods of high volatility in 2005 and in 2008-2009 during the economic crisis. Negative shocks influence the volatility of the process, where high volatility does not decrease quickly.

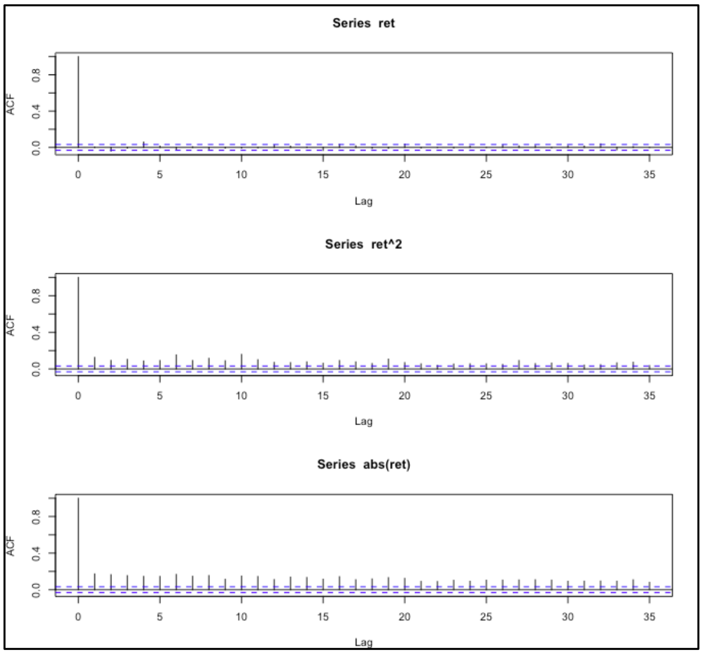


Fig.4. ACF plot for log returns of the Apple stock prices

The ACF plots in Fig.4 show that the log returns of the Apple stock prices are not correlated which suggest a constant mean model for r_t. Large autocorrelations on both the squared return and absolute return time series indicate the evidence of non-linear dependence on the log return process. Additionally, The LB tests at 5% significance level (**Appendix B**) confirm that there is no autocorrelation on stock return but there is autocorrelation among both squared returns and absolute return; therefore, we may conclude that there is evidence of non-linear dependency in the log return data.

2. Experimental Results:

In this section, since the return on the apple stock prices have a non-linear dependency, the HMM models are examined with different perspectives to capture non-constant volatility (**Note**: since the fat tail is detected in the previous section, and for financial data, the assumption of normal distribution of error term is usually rejected, we skip examining the model with the normal distribution assumption). Also, since there is no autocorrelation on the ACF plot of stock return, we can say that our mean model should be a constant mean model.

The most ideal model fitting technique is using the ARCH models. Thus, analysis is done to perform a comparative study of the effect of HMM against the ARCH models to see which model returns the best results. The significance of parameters is tested, and the BIC values of the models are compared to decide which model would be appropriate for this time series process.

1.1. Model 1: AR(0)-GARCH(1,1) with normally distributed errors

The first model to be fitted is the AR(0)-GARCH(1,1) model with normally distributed errors. From the R output, we determine the fitted model is represented as:

$$r_t = 0.0016 + a_t;\ a_t = \sigma_t * e_t$$

$$\sigma_t^2 = 0.00 + 0.055a_{t-1}^2 + 0.9355\sigma_{t-1}^2$$

The model has AIC value = -4.9238 and BIC value = -4.9172

1.2. Model 2: ARMA(0,0)-GARCH(1,1) model with t-distribution

The second model to be fitted is the ARMA(0,0)-GARCH(1,1) model with t-distribution of 5 degrees of freedom. From the R output, we determine the fitted model is represented as follows:

$$\boxed{\begin{gathered} r_t = 0.0014 + a_t;\ a_t = \sigma_t * e_t \\ \sigma_t^2 = 0.00 + 0.0477a_{t-1}^2 + 0.949\sigma_{t-1}^2 \end{gathered}}$$

Where the error term e_t has t-distribution with 5 degrees of freedom. The GARCH model satisfies the parameter constraint as 0.0477 + 0.949 < 1.

The model has AIC value = -5.0171 and BIC value = -5

1.3. Model 3: ARMA(0,0)-GARCH(1,1) model with skewed t-distribution

The third model to be fitted is the ARMA(0,0)-GARCH(1,1) model with skewed t-distribution. From the R output, we determine the fitted model is represented as follows:

$$\boxed{\begin{gathered} r_t = 0.0015 + a_t;\ a_t = \sigma_t * e_t \\ \sigma_t^2 = 0.000002 + 0.0484a_{t-1}^2 + 0.949\sigma_{t-1}^2 \end{gathered}}$$

Where the error term e_t has t-distribution with 5 degrees of freedom. The GARCH model satisfies the parameter constraint as 0.0484 + 0.949 < 1.

The model has AIC value = -5.0167 and BIC value = -5

1.4. Model 4: Fit ARMA(0,0)-eGARCH(1,1) model with t-distribution

The fourth model explored is the ARMA(0,0)-eGARCH(1,1) model). From the R output, we determine the fitted model is represented as follows:

$$r_t = 0.001257 + a_t;\ a_t = \sigma_t * e_t$$

$$ln\sigma_t^2 = -0.124376 + (-0.05131e_{t-1} + 0.155907(|e_{t-1}| - E(|e_{t-1}|)) + 0.984111\sigma_{t-1}^2$$

Where the error term e_t has t-distribution with 5 degrees of freedom. The "leverage" θ parameter, is negative and significantly different from zero, indicating that the volatility of the Apple stock returns has a significant asymmetric behavior.

The model has AIC value = -5.0291 and BIC value = -5.0192.

1.5. Model 5: Fit ARMA(0,0)-fGARCH(1,1) model with t-distribution

The fGARCH model is the next model to be fitted with t-distribution of 5 degrees of freedom. The model has AIC value = -5.0277 and has BIC value = -5.0162.

1.6. Model 6: IGARCH model

The IGARCH model is another technique applied to the data. The model has AIC value = -5.0174 and BIC value = -5.0108

1.7.Model 7: HMM model

The HMM model is the last technique applied to the data. The model has AIC value = -5.0299 and BIC value = -5.0293

From the above fitted model, we observe that the model most suited to analyze and forecast the Apple stock returns is the HMM model as it has the lowest BIC value of -5.0293, as seen in Fig.5. Additionally, we observe that the alpha value being less than 1, captures the leverage effect (negative and significantly different from zero) very well, indicating that the volatility of the Apple stock returns has a significant asymmetric behavior.

Model name	BIC value
Model 1	-4.9172
Model 2	-5
Model 3	-5
Model 4	-5.0192
Model 5	-5.0162
Model 6	-5.0108
Model 7	-5.0293

Fig.5. Model comparison

In the next section, we will look at the residual analysis and perform the model diagnostics on the chosen HMM model to determine if this model is adequate for analysis.

3. Residual Analysis and Model diagnostics:

To check if the selected HMM model is adequate and is a good fit, we need to make sure that the parameters of the model are significant, the residuals are white noise, the square or absolute value of residuals are white noise, and that the distribution is appropriate for the data.

All the parameters in the model are significant as seen in the output, (**Appendix C1**) which tells us that the model contributes information and is adequate. Also, since the LB test (**Appendix C2**) on standardized residuals and standardized squared residuals are both not significant, we can conclude that both are white noise. As for the goodness of fit, the coefficient of the parameter is significant which indicates that the model is a good choice and the adjusted goodness of fit test being not significant indicates that the assumption of model cannot be rejected (**Appendix C1**),

Therefore, we may conclude that HMM model is adequate and can be used for forecasting. Furthermore, backtesting evaluation technique was performed, which helped us to determine the prediction power of our model. In this case, we use VaR (value of risk) as reference (**Appendix C3**). Since the alpha (1%) and the actual %(1.1%) are very close, we conclude that HMM can make a good predictive model for VaR.

5. Experimental Analysis and discussion:

The goal of this study is to analyze the historical Apple daily stock price and to build an effective and efficient model for forecasting. From the time series analysis performed in this study, the HMM model is found to be adequate for the times series process, that can best help us predict the mean and volatility of the stock prices. We know that the model captures the asymmetric volatility behavior of Apple stock return and that the negative shock will have larger impact on the Apple stock return volatility than positive shock. However, the applications of the model is a significant takeaway. For example, in finance, we can use the suggested model to calculate VaR (Value at Risk) and this helps the banker and investor to manage the risk. The illustration is provided below:

```
> #risk
> p01=qt(0.01, 5)
> p01
[1] -3.36493
> r01=100000*(exp(0.001257+p01*0.007904)-1)
> r01
[1] -2502.106
```

Assume that the asset is $100000, we calculate the 1% VaR to be $ -2502.106. This means there is a 1% chance that the potential loss of holding 100000 on the next day is $2502.106 or more and this provides very useful information for banker and investor.

However, practically, the model may not do well on predicting extremely low and high value since our analysis showed that the model cannot capture the extreme values. Therefore, we should be careful on the usage of this model for prediction and potentially search for other techniques that may help improve the model further.

Appendix

Appendix A: Normality Test

Appendix A1: Basic Statistic of log return of apple stock price

```
> #log retun of apple stock ts
> rets=log(applets/lag(applets, -1))
> basicStats(rets)
                       x
nobs         3775.000000
NAs             0.000000
Minimum        -0.197470
Maximum         0.130194
1. Quartile    -0.010006
3. Quartile     0.012837
Mean            0.001149
Median          0.000947
Sum             4.336122
SE Mean         0.000365
LCL Mean        0.000434
UCL Mean        0.001864
Variance        0.000502
Stdev           0.022410
Skewness       -0.192333
Kurtosis        5.441411
```

Appendix A2: Jarque-Bera Normalatity Test

```
> #normality test
> normalTest(rets,method=c("jb"))

Title:
 Jarque - Bera Normalality Test

Test Results:
  STATISTIC:
    X-squared: 3104.0816
  P VALUE:
    Asymptotic p Value: < 2.2e-16

Description:
 Tue Feb 28 16:28:12 2017 by user:
```

Appendix B: The LB tests

```
#Test of independence
#Compute the Ljung's Box Test on stock price returns
#Ljung Box test on ret
#Box.test(apple_ret_num, lag=2, type="Ljung")
#Box.test(apple_ret_num, lag=4, type="Ljung"
#Box.test(apple_ret_num, lag=6, type="Ljung")

#Ljung Box test on squared values of the stock price returns
Box.test(apple_ret_num^2, lag=2, type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  apple_ret_num^2
## X-squared = 92.17, df = 2, p-value < 2.2e-16
```

```
Box.test(apple_ret_num^2, lag=4, type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  apple_ret_num^2
## X-squared = 161.84, df = 4, p-value < 2.2e-16
```

```
Box.test(apple_ret_num^2, lag=6, type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  apple_ret_num^2
## X-squared = 282.53, df = 6, p-value < 2.2e-16
```

```
#Ljung Box test on absolute values of the stock price returns
Box.test(abs(apple_ret_num), lag=2, type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  abs(apple_ret_num)
## X-squared = 214.65, df = 2, p-value < 2.2e-16
```

```
Box.test(abs(apple_ret_num), lag=4, type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  abs(apple_ret_num)
## X-squared = 384.2, df = 4, p-value < 2.2e-16
```

```
Box.test(abs(apple_ret_num), lag=6, type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  abs(apple_ret_num)
## X-squared = 569.79, df = 6, p-value < 2.2e-16
```

Appendix C: Residual and model diagnostic

Appendix C1

```
Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001257    0.000263   4.7776    2e-06
omega  -0.124376    0.009689 -12.8374    0e+00
alpha1 -0.051314    0.010297  -4.9832    1e-06
beta1   0.984111    0.001234 797.5046    0e+00
gamma1  0.155907    0.018030   8.6469    0e+00
shape   5.173706    0.431036  12.0030    0e+00

Robust Standard Errors:
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.001257    0.000265    4.7519    2e-06
omega  -0.124376    0.004142  -30.0262    0e+00
alpha1 -0.051314    0.010561   -4.8586    1e-06
beta1   0.984111    0.000560 1756.8258    0e+00
gamma1  0.155907    0.021157    7.3690    0e+00
shape   5.173706    0.454961   11.3718    0e+00
```

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     18.56       0.4852
2    30     25.00       0.6781
3    40     35.02       0.6518
4    50     44.35       0.6618
```

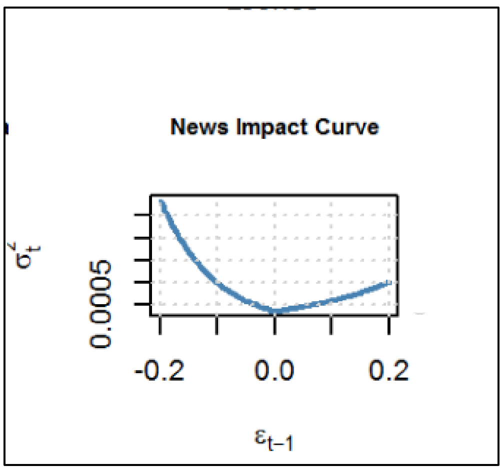
Appendix C2: LB tests of residuals

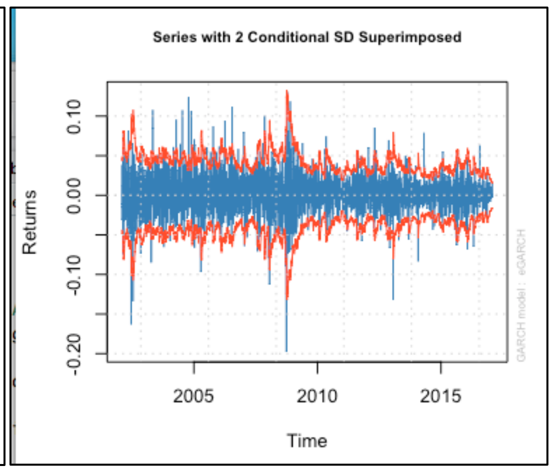
```
Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                      3.490 0.06173
Lag[2*(p+q)+(p+q)-1][2]     3.703 0.09062
Lag[4*(p+q)+(p+q)-1][5]     6.191 0.08107
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      0.369  0.5436
Lag[2*(p+q)+(p+q)-1][5]     2.180  0.5765
Lag[4*(p+q)+(p+q)-1][9]     2.825  0.7875
d.o.f=2
```

Appendix C3: backtesting (VaR)

```
VaR Backtest Report
===========================================
Model:
Backtest Length:        1275
Data:

==========================================
alpha:                          1%
Expected Exceed:        12.8
Actual VaR Exceed:      14
Actual %:                       1.1%

Unconditional Coverage (Kupiec)
Null-Hypothesis:        Correct Exceedances
LR.uc Statistic:        0.12
LR.uc Critical:         3.841
LR.uc p-value:          0.729
Reject Null:            NO

Conditional Coverage (Christoffersen)
Null-Hypothesis:        Correct Exceedances and
                                Independence of Failures
LR.cc Statistic:        NaN
LR.cc Critical:         5.991
LR.cc p-value:          NaN
Reject Null:            NA
```

Appendix C4:





Appendix D: Code

```
####################Analysis of the apple stock data########################

####################Load libraries and data##################################


library(rugarch)

library(tseries)

library(fBasics)

library(zoo)

library(lmtest)

library(depmixS4)

library(quantmod)

library(gridExtra)


setwd(".../Data")

apple <- read.table("day.csv",header=T, sep=',')

head(apple)


applets <- zoo(apple$Adj.Close, as.Date(as.character(apple$Date), format =
c("%Y-%m-%d")))

#log return time series

apple_rets <- log(applets/lag(applets,-1))

#strip off the dates and create numeric object

apple_ret_num <- coredata(apple_rets)


#Time series plot

plot(applets, type='l', ylab = " adj close price", main="Plot of 2002-2017
daily apple stock prices", col = 'red')


#ACF and PACF plot

acf(coredata(applets), main="ACF plot of the 2002-2017 daily apple stock
prices")

pacf(coredata(applets), main="PACF plot of the 2002-2017 daily apple stock
prices")
```

```
#Compute statistics

basicStats(apple_rets)


#Histogram

hist(apple_rets, xlab="Daily return of stock prices", prob=TRUE,
main="Histogram for daily return of stock prices")

xfit<-seq(min(apple_rets),max(apple_rets),length=40)

yfit<-dnorm(xfit,mean=mean(apple_rets),sd=sd(apple_rets))

lines(xfit, yfit, col="blue", lwd=2)


#QQ-plot

qqnorm(apple_rets)

qqline(apple_rets, col = 2)


#Time plot of log return of prices

plot(apple_rets, type='l', ylab = "stock price return", main="Plot of 2002-
2017 daily apple stock price return")


#Time plot of square of log return of prices

plot(apple_rets^2,type='l', ylab = "square of stock price return", main="Plot
of 2002-2017 daily apple stock price squared return")


#Time plot of absolute value of log return of prices

plot(abs(apple_rets),type='l', ylab = "abs value of stock price return",
main="Plot of 2002-2017 daily apple stock price abs return")


#ACF plot of log return of prices

par(mfrow=c(2,1))

acf(apple_ret_num)


#ACF plot of square of log return of prices

acf(apple_ret_num^2)


#ACF plot of absolute value of log return of prices
```

```
acf(abs(apple_ret_num))


#Test of independence

#Ljung Box test on squared values of the stock price returns

Box.test(apple_ret_num^2, lag=2, type="Ljung")

Box.test(apple_ret_num^2, lag=4, type="Ljung")

Box.test(apple_ret_num^2, lag=6, type="Ljung")


#Ljung Box test on absolute values of the stock price returns

Box.test(abs(apple_ret_num), lag=2, type="Ljung")

Box.test(abs(apple_ret_num), lag=4, type="Ljung")

Box.test(abs(apple_ret_num), lag=6, type="Ljung")


#Determine the order of the model

#PACF plot on the log return of the stock prices

pacf(apple_ret_num, lag=10, main="PACF plot of the log return of the stock
prices")


#PACF plot on the squared return of the stock prices

pacf(apple_ret_num^2, lag=10, main="PACF plot of the squared log return of
the stock prices")


#PACF plot on the absolute value of the return on the stock prices

pacf(abs(apple_ret_num), lag=10, main="PACF plot of the absolute value of the
log return of the stock prices")


#Model 1

garch11.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)),
mean.model=list(armaOrder=c(0,0)))

#estimate model

garch11.fit=ugarchfit(spec=garch11.spec, data=apple_rets)

garch11.fit


#using extractors
```

```
#estimated coefficients:

coef(garch11.fit)

#unconditional mean in mean equation

uncmean(garch11.fit)

#unconditional varaince: omega/(alpha1+beta1)

uncvariance(garch11.fit)


#persistence = alpha1+beta1

persistence(garch11.fit)


#Constraints on parameters < 1


#half-life: ln(0.5)/ln(alpha1+beta1)

halflife(garch11.fit)


#create selection list of plots for garch(1,1) fit

plot(garch11.fit, which = "all")


#conditional volatility plot

plot.ts(sigma(garch11.fit), ylab="sigma(t)", col="blue")



#Compute information criteria using infocriteria() function for model
selecton

infocriteria(garch11.fit)


#Model 2

garch11.t.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)),
mean.model=list(armaOrder=c(0,0)), distribution.model = "std")


#estimate model

garch11.t.fit=ugarchfit(spec=garch11.t.spec, data=apple_rets)

garch11.t.fit
```

```
#plot of residuals
plot(garch11.t.fit, which = "all")
persistence(garch11.t.fit)

#FORECASTS
compute h-step ahead forecasts for h=1,2,...,10
garch11.fcst=ugarchforecast(garch11.t.fit, n.ahead=12)
garch11.fcst
plot(garch11.fcst)

#rolling forecasts
garch11.t.fit=ugarchfit(spec=garch11.t.spec, data=apple_rets, out.sample=500)
garch11.fcst=ugarchforecast(garch11.t.fit, n.ahead=12, n.roll=450)
plot(garch11.fcst)

#Model 3
garch11.skt.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)),
mean.model=list(armaOrder=c(0,0)), distribution.model = "sstd")
#estimate model
garch11.skt.fit=ugarchfit(spec=garch11.skt.spec, data=apple_rets)
garch11.skt.fit

plot(garch11.skt.fit, which = "all")

#Model 4
egarch11.t.spec=ugarchspec(variance.model=list(model = "eGARCH",
garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)), distribution.model =
"std")
#estimate model
egarch11.t.fit=ugarchfit(spec=egarch11.t.spec, data=apple_rets)
egarch11.t.fit
```

```
plot(egarch11.t.fit, which = "all")


#Model 5

fgarch11.t.spec=ugarchspec(variance.model=list(model = "fGARCH",
garchOrder=c(1,1), submodel = "APARCH"), mean.model=list(armaOrder=c(0,0)),
distribution.model = "std")

#estimate model

fgarch11.t.fit=ugarchfit(spec=fgarch11.t.spec, data=apple_rets)

fgarch11.t.fit


plot(fgarch11.t.fit, which = "all")


#Model 6

igarch11.t.spec=ugarchspec(variance.model=list(model = "iGARCH",
garchOrder=c(1,1)), mean.model=list(armaOrder=c(0 , 0 )), distribution.model
= "std")


igarch11.t.fit=ugarchfit(spec=igarch11.t.spec, data=apple_rets)

igarch11.t.fit


plot(igarch11.t.fit, which = "all")


#Model 7

#Build Hidden Markov Model

set.seed(1)

hmm_apl<-
depmix(list(apple_rets~1,apple_ret_num~1),data=applets,nstates=3,family=list(
gaussian(),gaussian()))

#fit our model to the data set

hmm_apl_fit<-fit(hmm_apl, verbose = FALSE)

print(hmm_apl_fit)

summary(hmm_apl_fit)

#find the posterior odds for each state over our data set

hmm_post<-posterior(hmm_apl_fit)
```

```
head(hmm_apl_fit)
```