974A3

Yaolun Yin

2022-10-30

Import library

```
shhh <- suppressPackageStartupMessages # It's a library, so shhh!
shhh(library(tseries))
shhh(library(PerformanceAnalytics))
shhh(library(rugarch))
shhh(library(FinTS))
shhh(library(fGarch))
```

Prepare data

```
# Use command from R3.pdf
msftPrices = get.hist.quote(instrument="msft", start="2000-01-03",
                            end="2015-10-16", quote="AdjClose",
                            provider="yahoo", origin="1970-01-01",
                            compression="d",retclass="zoo")
```

```
## time series ends   2015-10-15
```

```
# return = log(S_t/S_{t-1})
ret = diff(log(msftPrices), lag=1)
# square return
sqr_ret = ret^2
```

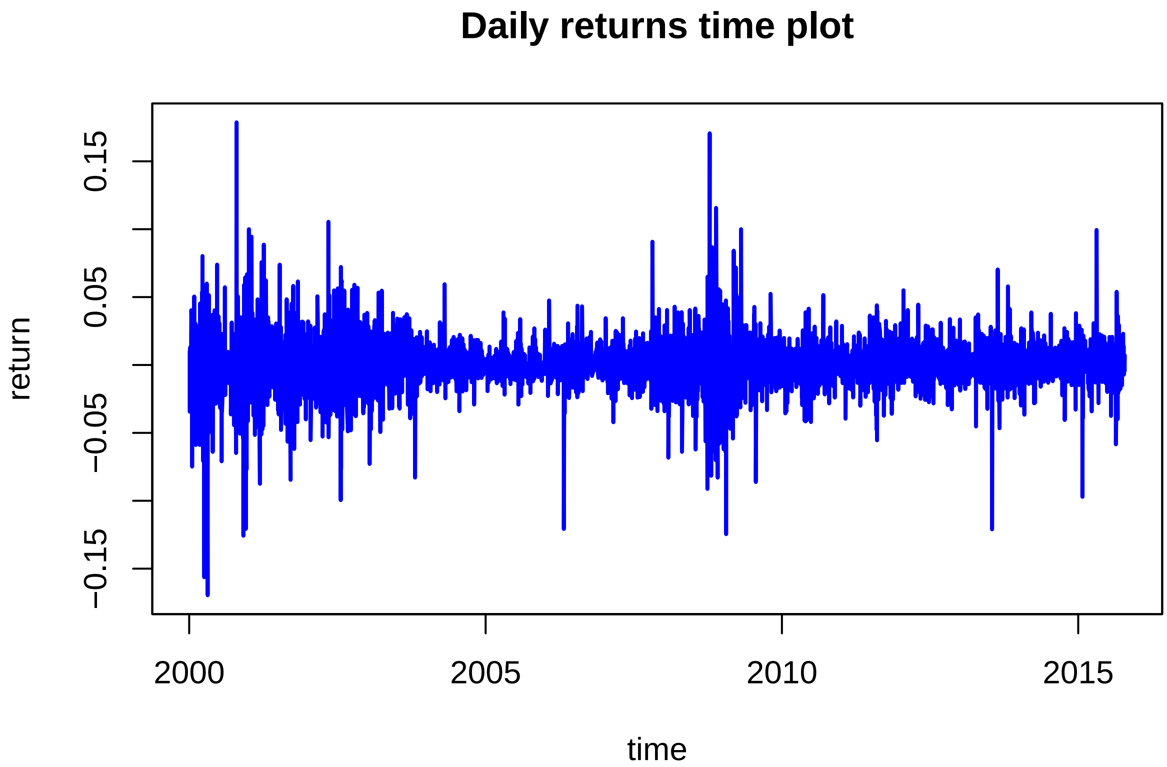
Q1

(a)

(i)

Daily returns plot

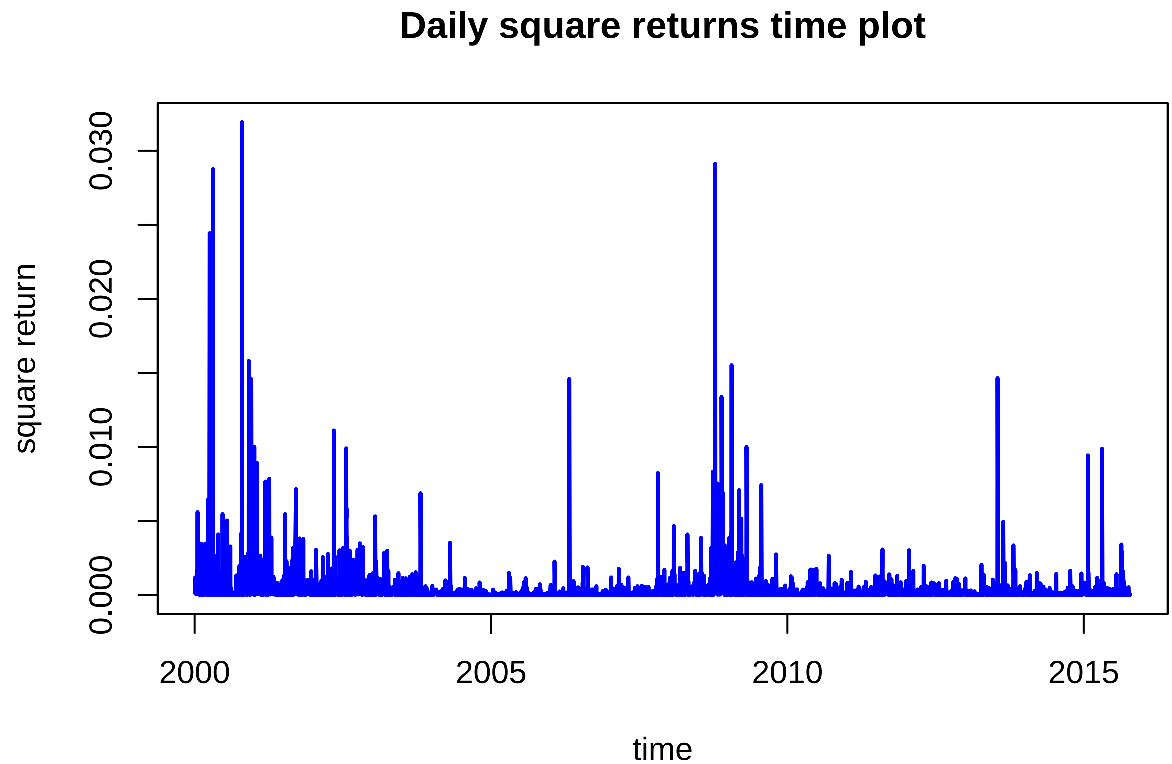
```
plot(ret, main="Daily returns time plot",
     lwd=2, col="blue", xlab="time", ylab="return")
```



Comment: All returns fall within the range [-0.2, 0.2]. Most of them are very small (i.e., the magnitude is less than 0.05). The significant returns happen in 2000, 2008, 2014, and etc.

Daily square returns plot

```
plot(sqr_ret, main="Daily square returns time plot",
     lwd=2, col="blue", xlab="time", ylab="square return")
```

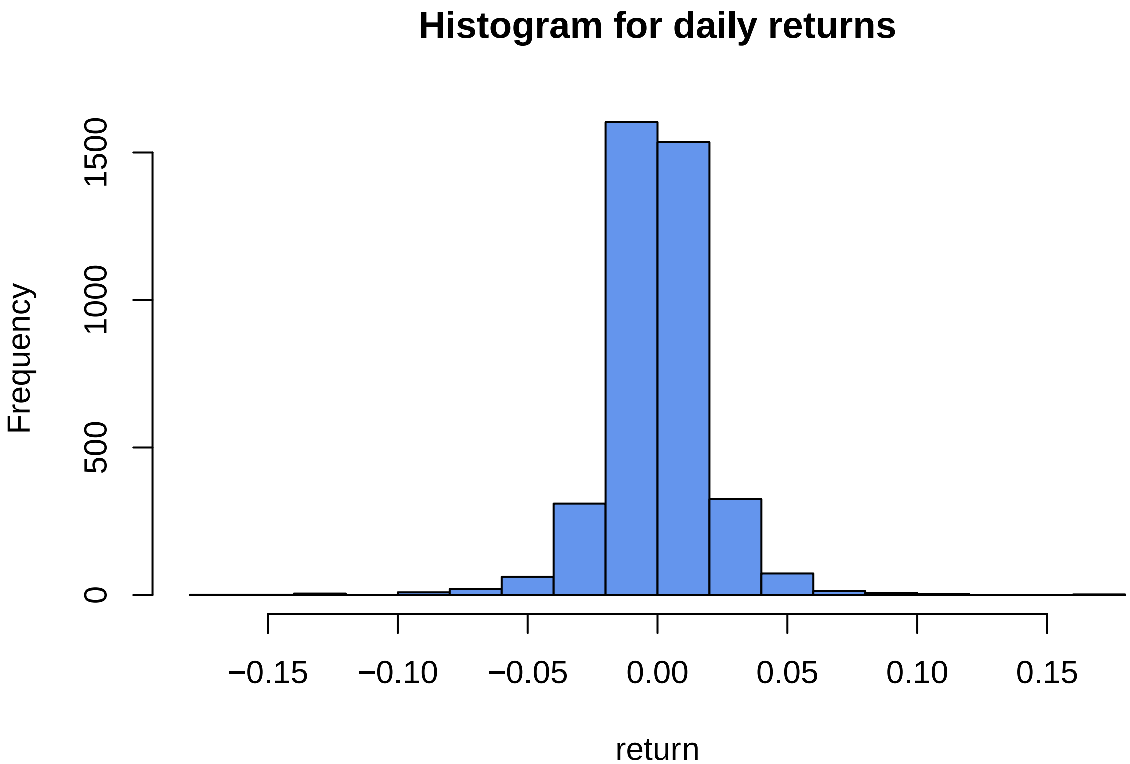


Comment: All square returns are less than 0.03. Most of them are very small (i.e., the magnitude is less than 0.005). The significant returns happen in 2000, 2008, 2014, and etc.

(ii)

Histogram

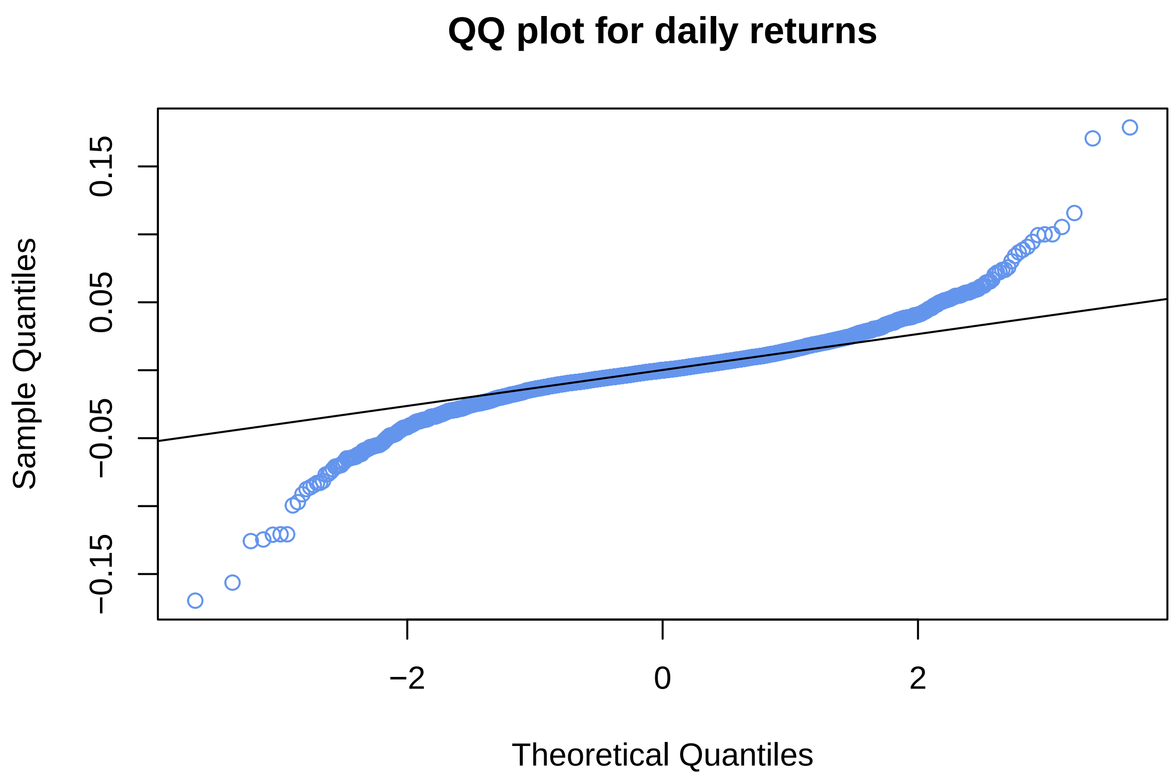
```
hist(ret, main="Histogram for daily returns", col="cornflowerblue", xlab="return")
```



Comment: The are many smaller values that are within [-0.025, 0.025]. A few large values exist but their magnitude do not exceed 0.15.

Normal QQ plot

```
qqnorm(ret, main="QQ plot for daily returns", col="cornflowerblue")
qqline(ret)
```



Comment: the QQ curve of daily returns does not align with the straight line, which means the distribution of returns is not normal.

(iii)

Get sample statistics

```
kurt <- function(ret) {
  kurtosis(ret, method=c("sample"))
}

statsRet = rbind(apply(ret, 2, mean),
                 apply(ret, 2, var),
                 apply(ret, 2, sd),
                 apply(ret, 2, skewness),
                 apply(ret, 2, kurt))
rownames(statsRet) = c("Mean", "Variance", "Std Dev", "Skewness", "Kurtosis")
round(statsRet, digits=4)
```

```
##          Adjusted
## Mean       0.0000
## Variance   0.0004
## Std Dev    0.0201
## Skewness  -0.1353
## Kurtosis  12.4078
```

JB test: the **null hypothesis** is that the daily returns follow the normal distribution. The **alternative hypothesis** is that the daily returns do not follow the normal distribution.

Test statistic

$$\text{JB} = \frac{n}{6}\left(\frac{\hat{\mu}_3}{\hat{\mu}_2^{3/2}}\right)^2 + \frac{n}{24}\left(\frac{\hat{\mu}_4}{\hat{\mu}_2} - 3\right)^2$$

where $\hat{\mu}_2 = \frac{1}{n}\sum_{i=1}^{n}(X_t - \bar{X})^2$ is the sample variance. $\frac{\hat{\mu}_3}{\hat{\mu}_2^{3/2}} = \frac{1}{n}\sum_{i=1}^{n}\frac{(X_t-\bar{X})^3}{\hat{\mu}_2^{3/2}}$ is the sample (standardized) skewness. $\frac{\hat{\mu}_4}{\hat{\mu}_2} = \frac{1}{n}\sum_{i=1}^{n}\frac{(X_t-\bar{X})^4}{\hat{\mu}_2}$ is the sample (standardized) kurtosis.

Asymptotic Distribution

$$\text{JB} \sim \chi^2(2) \text{ under } H_0$$

```
jarque.bera.test(ret)
```

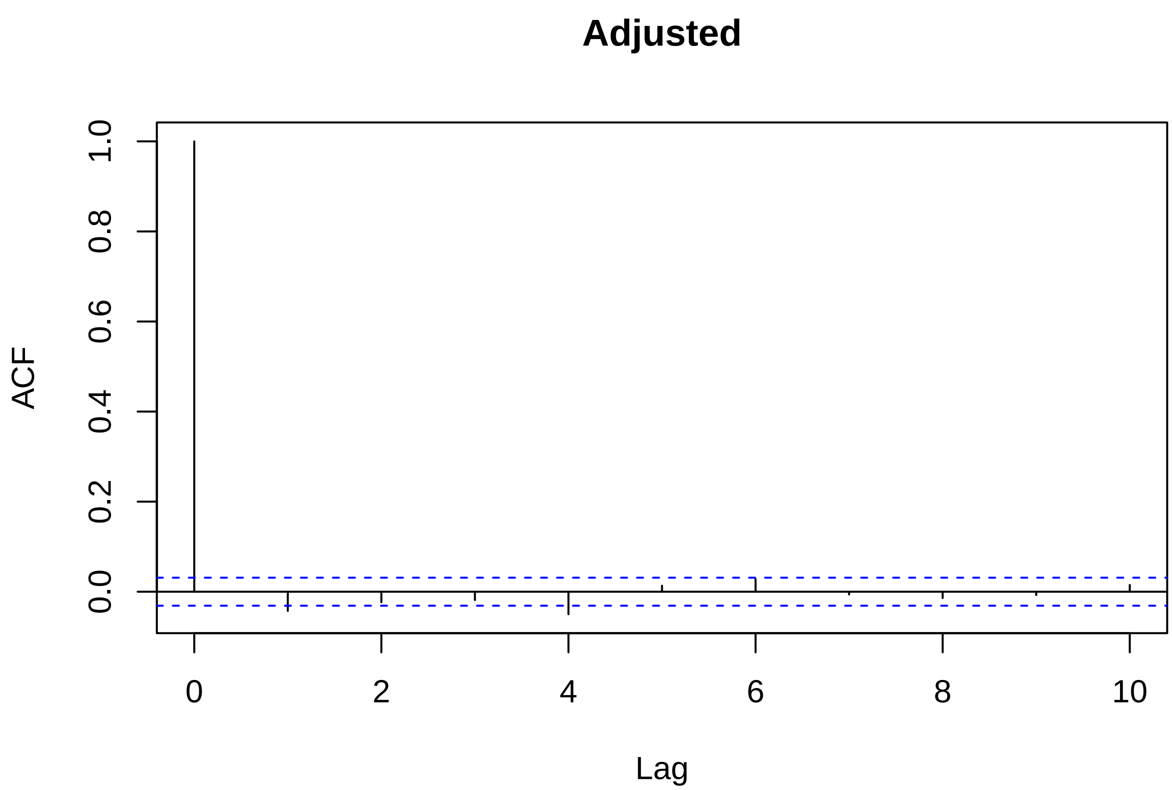
```
##
##  Jarque Bera Test
##
## data:  ret
## X-squared = 14608, df = 2, p-value < 2.2e-16
```

Since the p-value is very small (less than 0.01), we reject the null hypothesis on 99% confidence level (actually almost 100% level). So, the distribution of daily return is not normal.

(b)

(i)

```
acf(coredata(ret),lag.max = 10, plot=T)
```



The ACF very high for $h = 0$ but considerably low for $h > 1$, which means the daily returns are not correlated with past values.

(ii)

Portmanteau Test

The **null hypothesis** is $\rho(1) = ... = \rho(10) = 0$. The **alternative hypothesis** is at least one of $\rho(h) \neq 0$ for $h = 1, ..., 10$.

Test statistic and its **asymptotic distribution**

$$Q_{10}^{LB} := n(n+2)\sum_{i=1}^{10}\frac{\hat{\rho}(i)^2}{n-i} \xrightarrow{L} \chi^2(10) \text{ under } H_0$$

Implementation

```
lags = c(1:10)
test_stat = c(1:10)
pvalues = c(1:10)

for (i in 1:10) {
  BL = Box.test(ret, type='Ljung-Box', lag = i)
  test_stat[i] = BL$statistic
  pvalues[i] = BL$p.value
}

data.frame(lags, test_stat, pvalues)
```

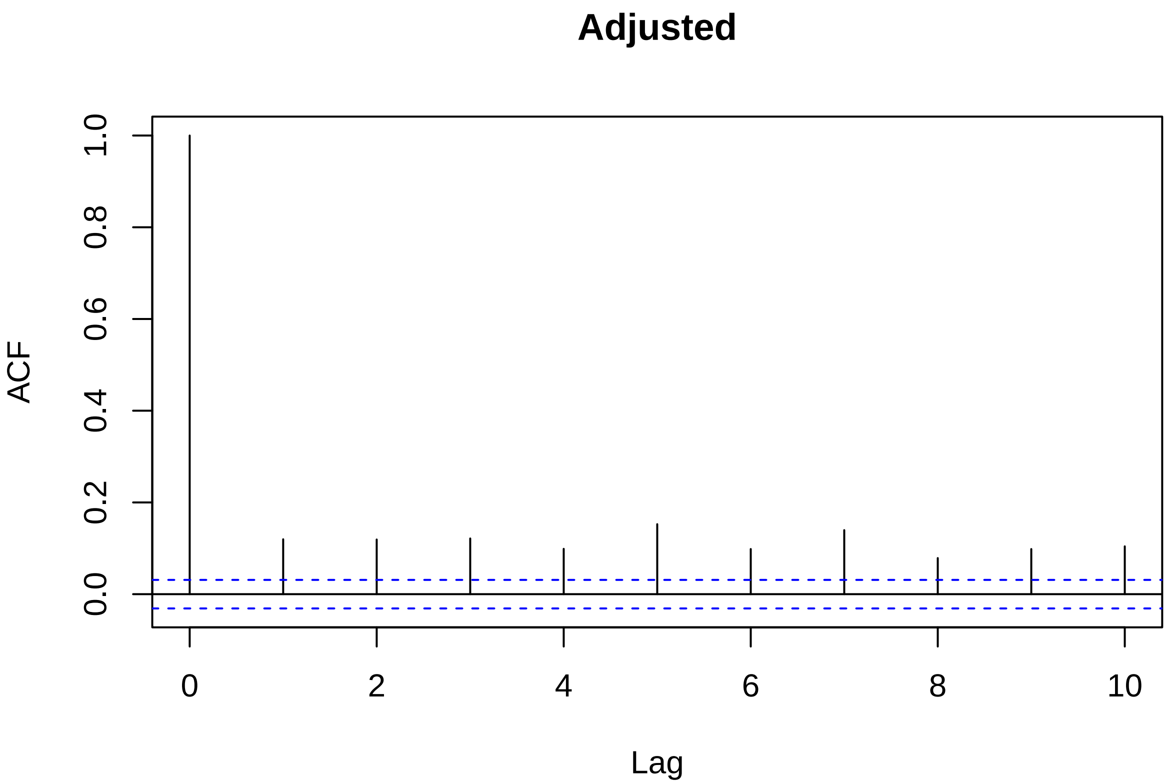
```
##    lags test_stat      pvalues
## 1     1  10.70879 1.066277e-03
## 2     2  10.71413 4.714718e-03
## 3     3  11.28274 1.029122e-02
## 4     4  28.86096 8.342605e-06
## 5     5  29.69109 1.696377e-05
## 6     6  35.82529 2.980747e-06
## 7     7  35.97343 7.333463e-06
## 8     8  42.62178 1.034868e-06
## 9     9  45.98783 6.053510e-07
## 10   10  46.23558 1.299943e-06
```

Since the p-values are very small (less than 0.01) for $h = 1, ..., 10$, we reject the null hypothesis on 99% level. At least one correlation is non-zero.

(iii)

Square return ACF

```r
acf(coredata(sqr_ret),lag.max = 10, plot=T)
```



The ACF is still high for $h = 0$. Notice that the ACF for $h > 1$ increased compared to that of daily returns.

(iv)

Portmanteau Test

The **null hypothesis** is $\rho(1) = ... = \rho(10) = 0$. The **alternative hypothesis** is at least one of $\rho(1), ..., \rho(10) \neq 0$.

Test statistic and its **asymptotic distribution**

$$Q_{10}^{LB} := n(n+2)\sum_{i=1}^{10}\frac{\hat{\rho}(i)^2}{n-i} \xrightarrow{L} \chi^2(10) \text{ under } H_0$$

Implementation

```r
lags = c(1:10)
test_stat = c(1:10)
pvalues = c(1:10)

for (i in 1:10) {
  BL = Box.test(sqr_ret, type='Ljung-Box', lag = i)
  test_stat[i] = BL$statistic
  pvalues[i] = BL$p.value
}

data.frame(lags, test_stat, pvalues)
```

```
##    lags test_stat pvalues
```

```
## 1      1  70.39576        0
## 2      2 130.86682        0
## 3      3 186.45326        0
## 4      4 229.73956        0
## 5      5 262.69421        0
## 6      6 320.10913        0
## 7      7 388.64583        0
## 8      8 446.32155        0
## 9      9 513.63786        0
## 10    10 555.66636        0
```

The p-values are all 0 due to computer precision. We reject the null hypothesis. At least one correlation is non-zero.

(v)

No. If the returns depart from the iid assumption, then the asymptotic distribution of the test statistic will no longer be $\chi^2(m)$ under H_0.

(c)

(i)

Lagrange Multiplier test

The **null hypothesis** is that no ARCH effect (i.e., $p = q = 0$). The **alternative hypothesis** is that returns have ARCH effect.

Test statistic and its **asymptotic distribution**

$$LM = nR^2 \overset{L}{\sim} \chi^2_m$$

where n is the total number of observations used to estimate the auxiliary linear regression below and R^2 R2 is an uncentered Coefficient of Determination (or Goodness-of-Fit measure) from estimating the following regression:

$$\hat{\epsilon}_t^2 = b_0 + \sum_{i=1}^{m} b_i \hat{\epsilon}_{t-i}^2 + u_t$$

where $u_t \sim iid(0, \sigma_u^2)$.

```
# lag = 5
ArchTest(ret, lags=5)
```

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  ret
## Chi-squared = 209.03, df = 5, p-value < 2.2e-16
```

```
# lag = 10
ArchTest(ret, lags=10)
```

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  ret
## Chi-squared = 265.98, df = 10, p-value < 2.2e-16
```

Since the p-values for lag = 5, 10 are less than 0.01, we reject the null hypothesis on 99% level. Hence, the returns have an ARCH effect.

(ii)

Lagrange Multiplier test

The **null hypothesis** is that no ARCH effect (i.e., $p = q = 0$). The **alternative hypothesis** is that returns have ARCH effect.

Test statistic and its **asymptotic distribution**

$$LM = nR^2 \stackrel{L}{\sim} \chi^2_m$$

where n is the total number of observations used to estimate the auxiliary linear regression below and R^2 R2 is an uncentered Coefficient of Determination (or Goodness-of-Fit measure) from estimating the following regression:

$$\hat{\epsilon}_t^2 = b_0 + \sum_{i=1}^{m} b_i \hat{\epsilon}_{t-i}^2 + u_t$$

where $u_t \sim iid(0, \sigma_u^2)$.

```
# lag = 5
ArchTest(sqr_ret, lags=5)
```

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqr_ret
## Chi-squared = 3.0706, df = 5, p-value = 0.6891
```

```
# lag = 10
ArchTest(sqr_ret, lags=10)
```

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqr_ret
## Chi-squared = 6.7407, df = 10, p-value = 0.7497
```

Since the p-values for lag = 5, 10 are large (greater than 0.5), we do not have enough evidence to reject the null hypothesis on 99% level. Hence, the squared returns may not have ARCH effect. However, b(iv) shows that the current value is correlated to past values. This may suggest that the squared returns only correlate to values with lag < 5.

(d)

(i)

A good candidate model for daily returns is ARCH (5). i.e.,

$$\epsilon_t = \sigma_t \eta_t$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{5} \alpha_i \epsilon_{t-i}^2$$

since we have shown that the daily returns do not follow normal distribution. Also, the current return is correlated to past values. LM test shows that the current return has ARCH effect with lag = 5 and 10. To simplify the modelling and promote computational efficiency, we can pick ARCH(5) over ARCH(10).

Q2

(a)

(i)

```
spec = ugarchspec(variance.model = list(garchOrder = c(5, 0)),
                  mean.model = list(armaOrder = c(0,0)))
arch5 <- ugarchfit(spec=spec, data = ret$Adjusted)
# the coefficient
arch5_coef <- arch5@fit$coef
```

The sum of α_1 to α_5 is

```
sum(arch5_coef[c(3:length(arch5_coef))])
```

```
## [1] 0.7875188
```

Since $\alpha_1 + ... + \alpha_5 < 1$, we know that the second order stationary solution of this model exists. The solution is casual and unique.

(ii)

The coefficients are

```
arch5_coef[2:length(arch5_coef)]
```

```
##        omega       alpha1       alpha2       alpha3       alpha4       alpha5
## 0.0001187155 0.1525064620 0.1879124449 0.1414971449 0.1699574410 0.1356452608
```

ω and all of α_i are non-negative, which satisfy the assumption of ARCH(5) model. The α's magnitude are similar to the values in the ACF graph of squared returns. The ω is very small compared to α's.

(iii)

The CML standard error estimates are

```
cml_stderr = arch5@fit$se.coef[2:7]
names(cml_stderr) = c("mu", "alpha1", "alpha2", "alpha3", "alpha4", "alpha5")
cml_stderr
```

```
##           mu       alpha1       alpha2       alpha3       alpha4       alpha5
## 6.404450e-06 2.539957e-02 2.737063e-02 2.056058e-02 2.557046e-02 2.312678e-02
```

The standard errors are small, meaning the distribution of estimates closely surrounds the estimate.

The QML standard error estimates are

```
qml_stderr = arch5@fit$robust.se.coef[2:7]
names(qml_stderr) = c("mu", "alpha1", "alpha2", "alpha3", "alpha4", "alpha5")
qml_stderr
```

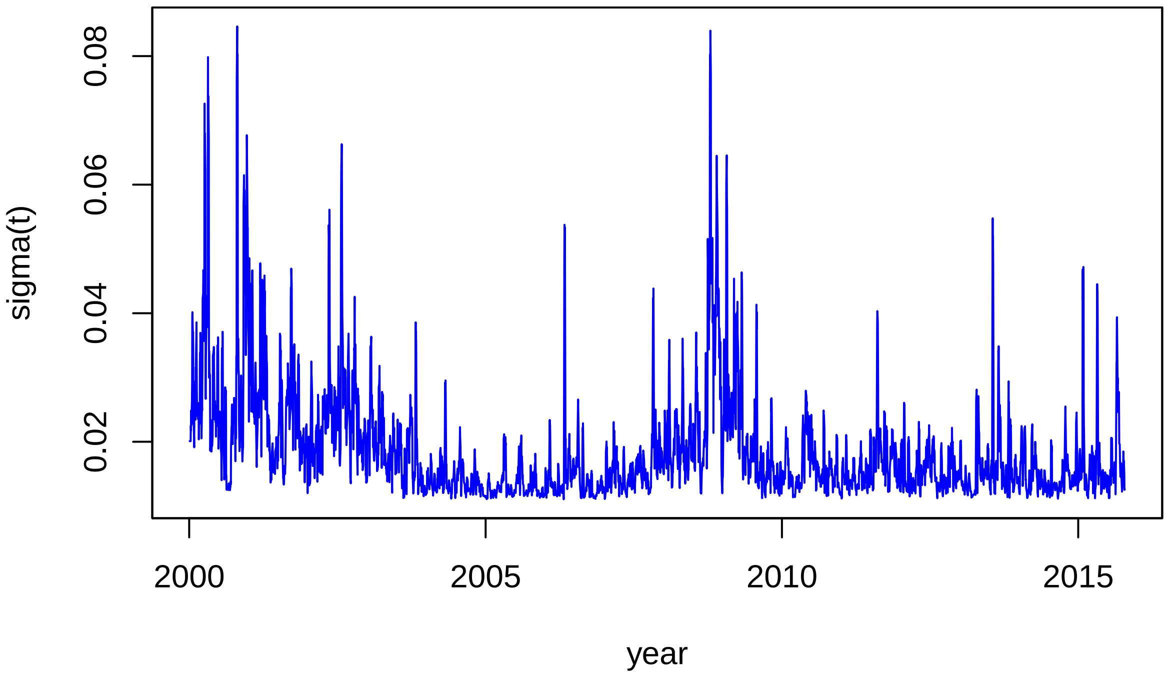
```
##           mu       alpha1       alpha2       alpha3       alpha4       alpha5
## 2.155098e-05 4.723155e-02 5.143172e-02 4.626768e-02 4.702430e-02 4.154647e-02
```

The QML standard errors are almost twice larger than CML errors.

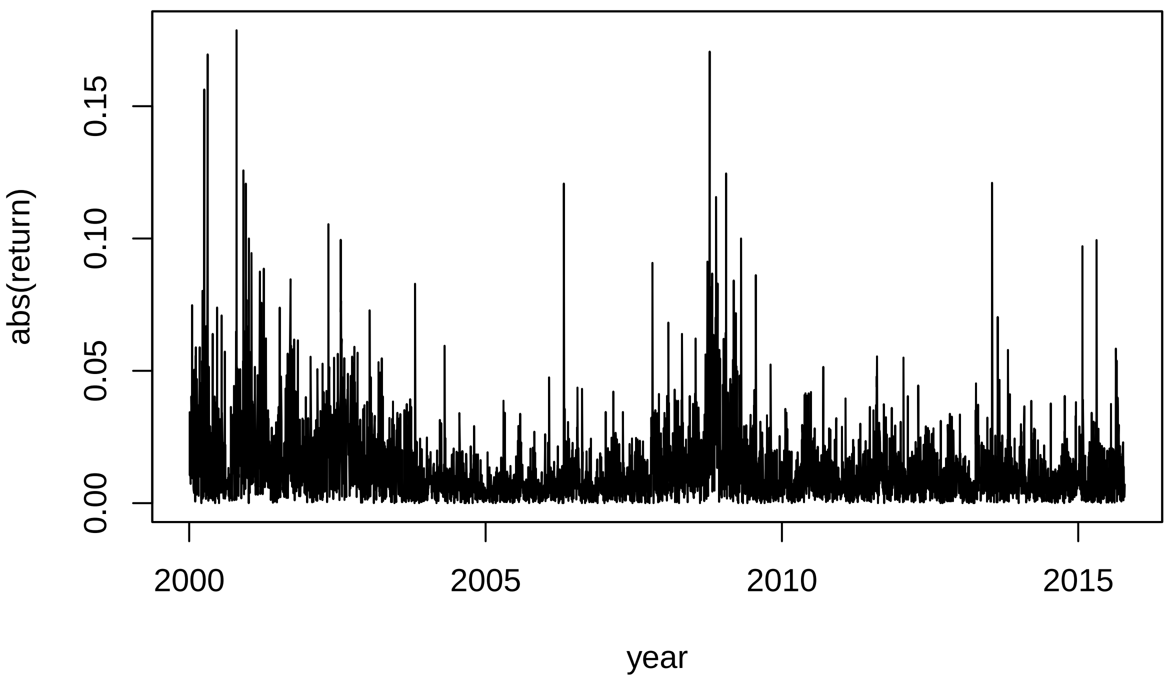
(iv)

Plot the conditional volatility for daily returns vs absoute return

```r
# conditional volatility
plot.zoo(sigma(arch5), xlab = "year", ylab="sigma(t)", col="blue")
```




```r
# absolute return
plot.zoo(abs(ret$Adjusted), xlab = "year", ylab = "abs(return)")
```



The two graphs are very similar. Whenever the first graph has a spike, the second graph will also have one. It makes sense because the log returns will be large if the volatility is high.

(v)

```
# unconditional volatility
res = c(sqrt(uncvariance(arch5)), sd(ret$Adjusted))
# sample standard deviation
names(res) = c("Unconditional volatility", "Sample standard deviation")
res
```

```
##  Unconditional volatility Sample standard deviation
##                0.02363706                0.02008916
```

The unconditional volatility is similar to the sample standard deviation. The former is slightly larger than the latter.

(b)

(i)

fit GARCH(1,1)

```
spec = ugarchspec(mean.model = list(armaOrder=c(0,0)))
garch11 <- ugarchfit(spec=spec, data = ret$Adjusted)
```

```
alpha = garch11@fit$coef['alpha1']
beta = garch11@fit$coef['beta1']
est = alpha + beta
names(est) = 'alpha + beta'
est
```

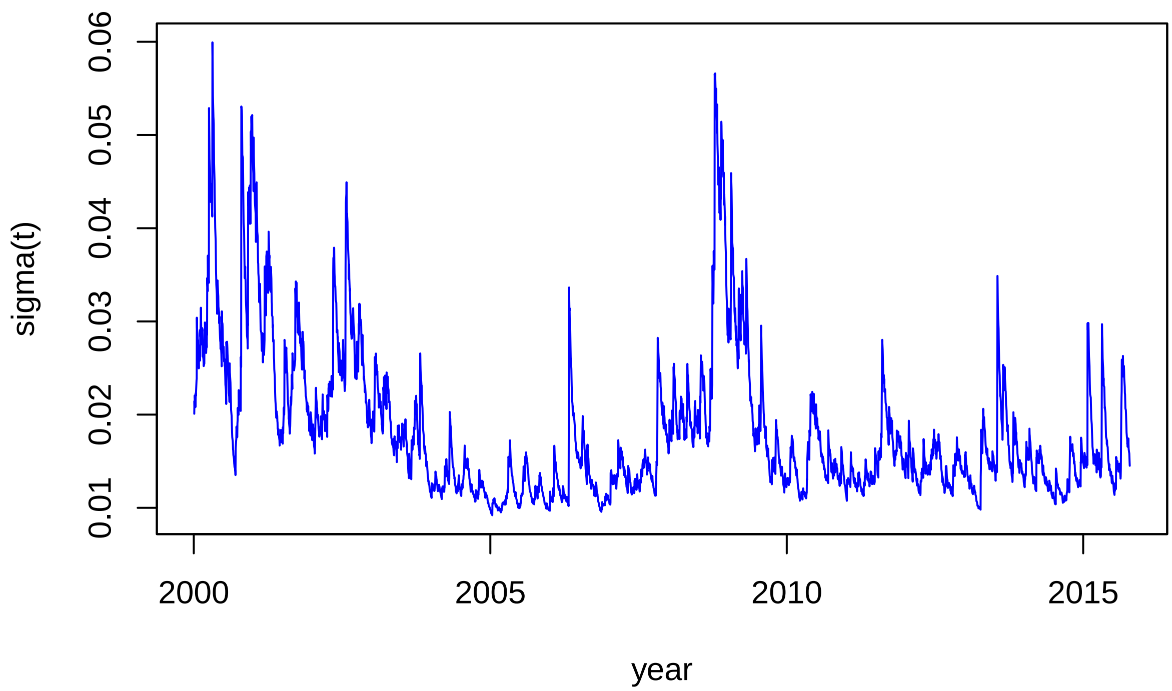
```
## alpha + beta
##    0.9858363
```

Since $\alpha + \beta < 1$, we know that the second order stationary solution of this model exists. The solution is casual and unique.

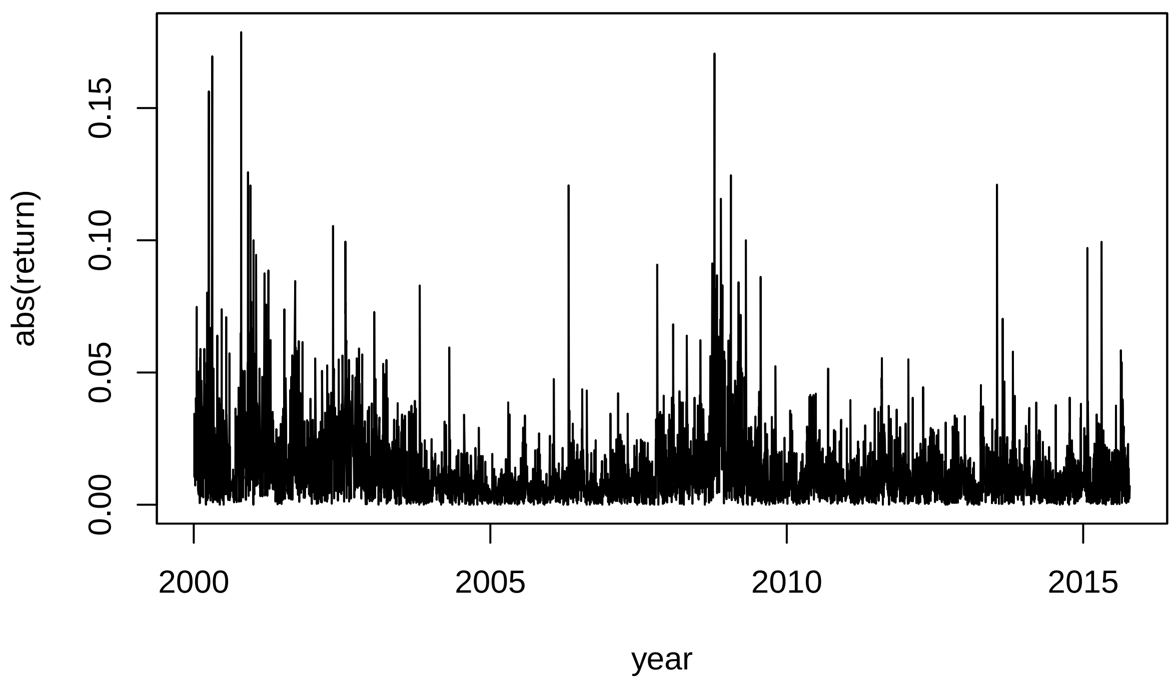
(ii)

Plot the conditional volatility for daily returns vs absoute return

```
# conditional volatility
plot.zoo(sigma(garch11), xlab = "year", ylab="sigma(t)", col="blue")
```




```r
# absolute return
plot.zoo(abs(ret$Adjusted), xlab = "year", ylab = "abs(return)")
```



The shape of conditional volatility of GARCH(1,1) looks similar to ARCH(5). The volatility of GARCH(1,1) tends to be higher than that of ARCH(5).

(iii)

```r
# unconditional volatility
res = c(sqrt(uncvariance(garch11)), sd(ret$Adjusted))
```

```r
# sample standard deviation
names(res) = c("Unconditional volatility", "Sample standard deviation")
res
```

```
##  Unconditional volatility Sample standard deviation
##                0.02019474                0.02008916
```

The unconditional volatility and sample standard deviation is very similar (same for 4 decimal places).

Q3

(a)

```
show(arch5)
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(5,0)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000508    0.000242   2.0979 0.035913
## omega   0.000119    0.000006  18.5364 0.000000
## alpha1  0.152506    0.025400   6.0043 0.000000
## alpha2  0.187912    0.027371   6.8655 0.000000
## alpha3  0.141497    0.020561   6.8820 0.000000
## alpha4  0.169957    0.025570   6.6466 0.000000
## alpha5  0.135645    0.023127   5.8653 0.000000
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000508    0.000308   1.6483 0.099286
## omega   0.000119    0.000022   5.5086 0.000000
## alpha1  0.152506    0.047232   3.2289 0.001243
## alpha2  0.187912    0.051432   3.6536 0.000259
## alpha3  0.141497    0.046268   3.0582 0.002226
## alpha4  0.169957    0.047024   3.6142 0.000301
## alpha5  0.135645    0.041546   3.2649 0.001095
##
## LogLikelihood : 10422.3
##
## Information Criteria
## ------------------------------------
##
## Akaike       -5.2457
## Bayes        -5.2346
## Shibata      -5.2457
## Hannan-Quinn -5.2418
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                      1.318  0.2509
## Lag[2*(p+q)+(p+q)-1][2]     1.337  0.4008
```

```
## Lag[4*(p+q)+(p+q)-1][5]     2.641  0.4764
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.8649  0.3524
## Lag[2*(p+q)+(p+q)-1][14]    7.2998  0.4677
## Lag[4*(p+q)+(p+q)-1][24]   10.0932  0.6963
## d.o.f=5
##
## Weighted ARCH LM Tests
## ------------------------------------
##              Statistic Shape Scale P-Value
## ARCH Lag[6]      0.413 0.500 2.000  0.5205
## ARCH Lag[8]      1.088 1.480 1.774  0.7399
## ARCH Lag[10]     1.155 2.424 1.650  0.9152
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  3.5721
## Individual Statistics:
## mu     0.3158
## omega  0.1906
## alpha1 0.5531
## alpha2 1.0553
## alpha3 0.7026
## alpha4 0.3122
## alpha5 2.2228
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.69 1.9 2.35
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias           1.1949 0.2322
## Negative Sign Bias  0.1936 0.8465
## Positive Sign Bias  0.2776 0.7813
## Joint Effect        1.8108 0.6126
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     231.0    2.125e-38
## 2    30     233.7    7.140e-34
## 3    40     260.1    1.776e-34
## 4    50     278.1    9.044e-34
##
##
## Elapsed time : 0.367857
```

(i)

See the "Weighted Ljung-Box Test on Standardized Residuals" section of the above summary.

The null hypothesis is H_0: the standardized residuals are not serially correlated. The alternative hypothesis is H_A: at least two residuals are serially correlated. The p-values for three lags are all large (greater than 0.05). Thus, we do not have enough evidence to reject H_0 on 95% confidence level. Even if we choose a different confidence level, the p-values are still too large. To sum up, the standardized residuals may not have serial correlation. Using ARCH(5) is a valid choice.

The objective of this test is to verify that the fitted ARCH(5) model satisfies the assumption of iid standardized errors. If not, then we cannot use ARCH(5).

(ii)

See the summary in (i) for the Weighted Ljung-Box Test on Standardized Squared Residuals part. The p-values for lags are 0.3524, 0.4676, 0.6963.

The null hypothesis is H_0: the standardized squared residuals are not serially correlated. The alternative hypothesis is H_A: at least two squared residuals are serially correlated. The p-values for three lags are all large (greater than 0.05). Thus, we do not have enough evidence to reject H_0 on 95% confidence level. No serial correlations for squared residuals implies that ARCH(5) is a valid model.

The objective of this test is to verify that the fitted ARCH(5) model presents autogressive conditional heterskedasticity. If not, then we cannot use ARCH(5).

(iii)

The null hypothesis H_0 is no ARCH effect. The alternative hypothesis H_A is there is ARCH effect.

From the "Weighted ARCH LM Tests" section of the summary in (a), we observe that the p-values (0.5205, 0.7309, 0.9152) are very large. Thus, we do not have enough evidence to reject H_0 on 95% confidence level. As a result, there is no ARCH effect left in the residual after it has been fitted to a ARCH(5) model.

(iv)

Test in (ii) verifies that the squared residuals are not serial correlated, which are the foundation of a ARCH model. If the model passes test in (ii), then the purpose of test in (iii) is to explore if we have fully exploited the ARCH effect in the residuals. If not, then we can make our model better by fitting ARCH(m), where m is greater than 5.

(b)

```
show(garch11)
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,1)
```

```
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000402    0.000237   1.6982 0.089472
## omega   0.000006    0.000005   1.0552 0.291325
## alpha1  0.070214    0.005534  12.6870 0.000000
## beta1   0.915622    0.015348  59.6562 0.000000
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000402    0.000807 0.498001 0.618483
## omega   0.000006    0.000157 0.036749 0.970685
## alpha1  0.070214    0.100280 0.700185 0.483812
## beta1   0.915622    0.490417 1.867026 0.061898
##
## LogLikelihood : 10519.46
##
## Information Criteria
## ------------------------------------
##
## Akaike       -5.2961
## Bayes        -5.2898
## Shibata      -5.2961
## Hannan-Quinn -5.2939
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                      2.185  0.1394
## Lag[2*(p+q)+(p+q)-1][2]     2.230  0.2274
## Lag[4*(p+q)+(p+q)-1][5]     3.605  0.3076
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.1512  0.6974
## Lag[2*(p+q)+(p+q)-1][5]    0.2654  0.9871
## Lag[4*(p+q)+(p+q)-1][9]    0.4191  0.9991
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3] 0.0007849 0.500 2.000  0.9776
## ARCH Lag[5] 0.2036567 1.440 1.667  0.9646
## ARCH Lag[7] 0.2511761 2.315 1.543  0.9951
##
## Nyblom stability test
## ------------------------------------
```

```
## Joint Statistic:  3.4302
## Individual Statistics:
## mu     0.1349
## omega  0.1845
## alpha1 0.3297
## beta1  0.1164
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.07 1.24 1.6
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias          1.32068 0.1867
## Negative Sign Bias 0.66600 0.5055
## Positive Sign Bias 0.05818 0.9536
## Joint Effect       2.42656 0.4887
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     181.0    1.996e-28
## 2    30     186.7    5.847e-25
## 3    40     220.4    3.580e-27
## 4    50     218.2    3.264e-23
##
##
## Elapsed time : 0.4537859
```

(i)

See the "Weighted Ljung-Box Test on Standardized Residuals" section of the above summary.

The null hypothesis is H_0: the standardized residuals are not serially correlated. The alternative hypothesis is H_A: at least two residuals are serially correlated. Observe that the p-values are greater than 0.05. Thus, we do not have enough evidence to reject H_0 on 95% confidence level. No serial correlations for residuals implies that GARCH(1,1) is a valid model.

The objective of this test is to verify that the fitted GARCH(1,1) model satisfies the assumption of iid standardized errors. If not, then we cannot use GARCH(1,1).

(ii)

See the "Weighted Ljung-Box Test on Standardized Squared Residuals" section of the above summary.

The null hypothesis is H_0: the standardized squared residuals are not serially correlated. The alternative hypothesis is H_A: at least two residuals are serially correlated. Observe that the p-values (0.6974, 0.9871, 0.9991) are very large. Thus, we do not have enough evidence to reject H_0 on 95% confidence level. No serial squared correlations for residuals implies that GARCH(1,1) is a valid model.

The objective of this test is to verify that the fitted GARCH(1,1) model presents autogressive conditional heterskedasticity. If not, then we cannot use GARCH(1,1).

(iii)

The null hypothesis H_0 is no ARCH effect. The alternative hypothesis H_A is there is ARCH effect.

From the "Weighted ARCH LM Tests" section of the summary in (b), we observe that the p-values (0.9777, 0.9646, 0.9951) are very large. Thus, we do not have enough evidence to reject H_0 on 95% confidence level. As a result, there is no ARCH effect left in the residual after it has been fitted to a GARCH(1,1) model.

(c)

(i)

I will recommend GARCH(1,1) for the following reasons:

1. In Question 2, we have shown that the unconditional volatility of GARCH(1,1) is closer to the sample standard deviation than ARCH(5).
2. From the p-values of "Weighted ARCH LM Tests" in Question 3, ARCH(5) has (0.5205, 0.7399, 0.9152) and GARCH(1,1) has (0.9777, 0.9646, 0.9951). I believe that GARCH(1,1) have better exploited the remaining ARCH effect than ARCH(5).

Q4

(a)

(i)

Compute h-step ahead volitality forecasts

```
arch5_fcst = ugarchforecast(arch5 , n.ahead=100)
arch5_fcst
```

```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: sGARCH
## Horizon: 100
## Roll Steps: 0
## Out of Sample: 0
##
## 0-roll forecast [T0=2015-10-15]:
##           Series   Sigma
## T+1   0.0005081 0.01185
## T+2   0.0005081 0.01242
## T+3   0.0005081 0.01342
## T+4   0.0005081 0.01434
## T+5   0.0005081 0.01534
## T+6   0.0005081 0.01625
## T+7   0.0005081 0.01685
## T+8   0.0005081 0.01744
## T+9   0.0005081 0.01799
## T+10  0.0005081 0.01850
## T+11  0.0005081 0.01894
## T+12  0.0005081 0.01933
## T+13  0.0005081 0.01969
## T+14  0.0005081 0.02002
## T+15  0.0005081 0.02032
## T+16  0.0005081 0.02059
## T+17  0.0005081 0.02084
## T+18  0.0005081 0.02107
## T+19  0.0005081 0.02128
## T+20  0.0005081 0.02146
## T+21  0.0005081 0.02164
## T+22  0.0005081 0.02180
## T+23  0.0005081 0.02194
## T+24  0.0005081 0.02208
## T+25  0.0005081 0.02220
## T+26  0.0005081 0.02231
## T+27  0.0005081 0.02242
## T+28  0.0005081 0.02251
## T+29  0.0005081 0.02260
## T+30  0.0005081 0.02268
## T+31  0.0005081 0.02275
## T+32  0.0005081 0.02282
```

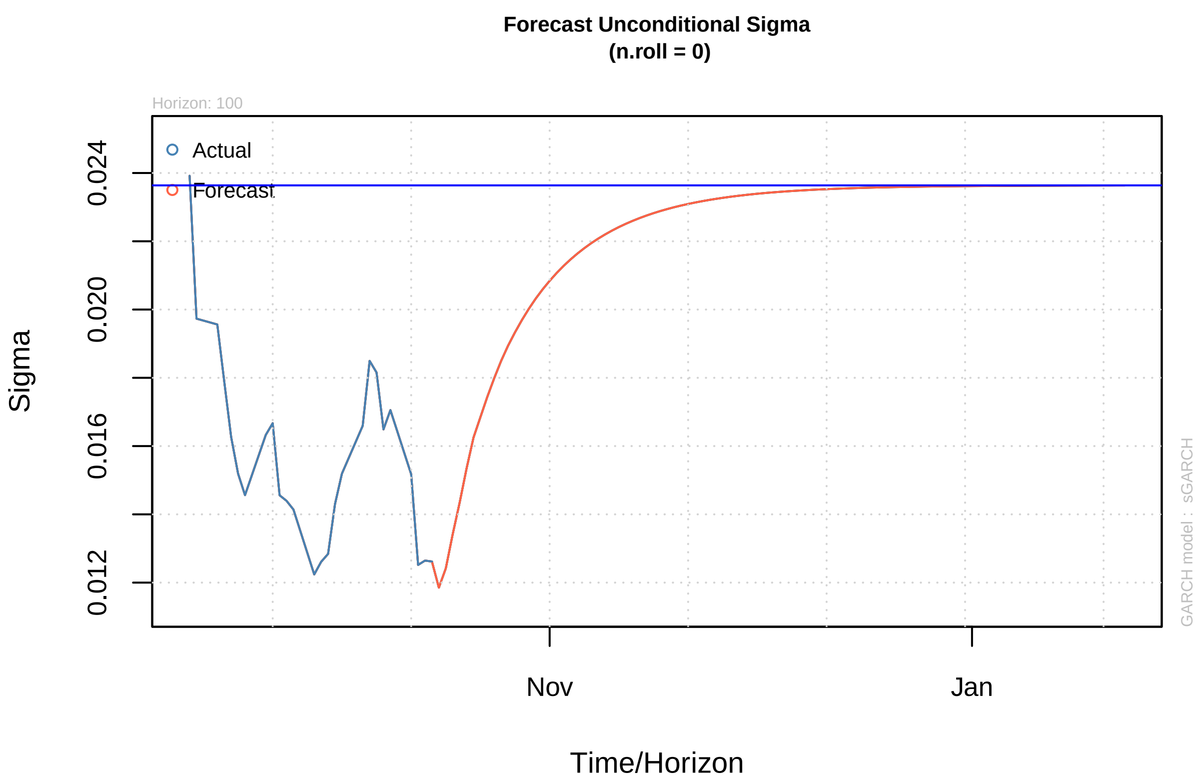
```
## T+33  0.0005081 0.02289
## T+34  0.0005081 0.02294
## T+35  0.0005081 0.02300
## T+36  0.0005081 0.02305
## T+37  0.0005081 0.02309
## T+38  0.0005081 0.02313
## T+39  0.0005081 0.02317
## T+40  0.0005081 0.02321
## T+41  0.0005081 0.02324
## T+42  0.0005081 0.02327
## T+43  0.0005081 0.02330
## T+44  0.0005081 0.02333
## T+45  0.0005081 0.02335
## T+46  0.0005081 0.02337
## T+47  0.0005081 0.02339
## T+48  0.0005081 0.02341
## T+49  0.0005081 0.02343
## T+50  0.0005081 0.02344
## T+51  0.0005081 0.02346
## T+52  0.0005081 0.02347
## T+53  0.0005081 0.02349
## T+54  0.0005081 0.02350
## T+55  0.0005081 0.02351
## T+56  0.0005081 0.02352
## T+57  0.0005081 0.02353
## T+58  0.0005081 0.02354
## T+59  0.0005081 0.02354
## T+60  0.0005081 0.02355
## T+61  0.0005081 0.02356
## T+62  0.0005081 0.02356
## T+63  0.0005081 0.02357
## T+64  0.0005081 0.02357
## T+65  0.0005081 0.02358
## T+66  0.0005081 0.02358
## T+67  0.0005081 0.02359
## T+68  0.0005081 0.02359
## T+69  0.0005081 0.02359
## T+70  0.0005081 0.02360
## T+71  0.0005081 0.02360
## T+72  0.0005081 0.02360
## T+73  0.0005081 0.02361
## T+74  0.0005081 0.02361
## T+75  0.0005081 0.02361
## T+76  0.0005081 0.02361
## T+77  0.0005081 0.02361
## T+78  0.0005081 0.02362
## T+79  0.0005081 0.02362
## T+80  0.0005081 0.02362
## T+81  0.0005081 0.02362
## T+82  0.0005081 0.02362
## T+83  0.0005081 0.02362
## T+84  0.0005081 0.02362
## T+85  0.0005081 0.02363
## T+86  0.0005081 0.02363
```

```
## T+87   0.0005081 0.02363
## T+88   0.0005081 0.02363
## T+89   0.0005081 0.02363
## T+90   0.0005081 0.02363
## T+91   0.0005081 0.02363
## T+92   0.0005081 0.02363
## T+93   0.0005081 0.02363
## T+94   0.0005081 0.02363
## T+95   0.0005081 0.02363
## T+96   0.0005081 0.02363
## T+97   0.0005081 0.02363
## T+98   0.0005081 0.02363
## T+99   0.0005081 0.02363
## T+100  0.0005081 0.02363
```

(ii)

The plot for volatility forecasts

```
plot(arch5_fcst, which = 3)
arch5_uncon_vol = sqrt(uncvariance(arch5))
abline(h = arch5_uncon_vol, col = c('blue'))
```



The horizontal blue line is the unconditional volatility of ARCH(5). As shown in the graph, the forecasts quickly converges to the unconditional volatility.

According to slides, an important property of ARCH is that the forecasts of volatility will converge to its unconditional value as the prediction horizon increases. A second order stationary ARCH is mean reverting.

(b)

(i)

Compute h-step ahead volitality forecasts

```
garch11_fcst = ugarchforecast(garch11 , n.ahead=100)
garch11_fcst
```

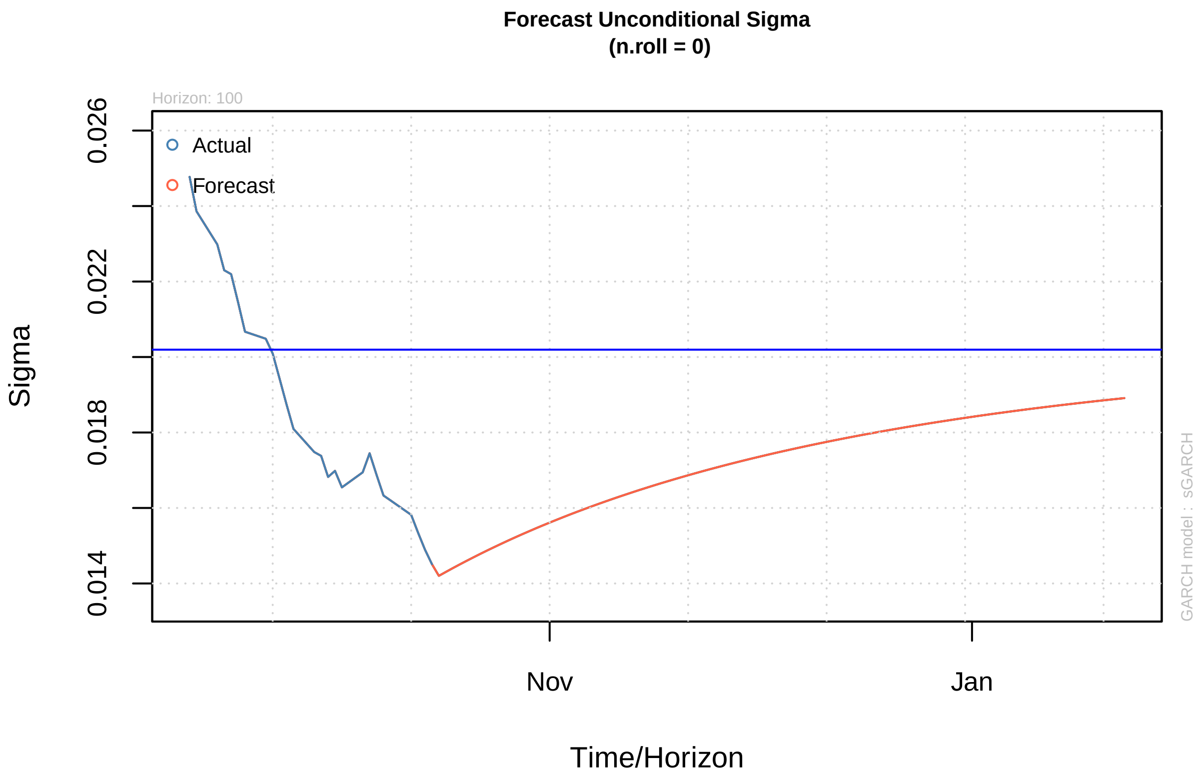
```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: sGARCH
## Horizon: 100
## Roll Steps: 0
## Out of Sample: 0
##
## 0-roll forecast [T0=2015-10-15]:
##           Series   Sigma
## T+1   0.0004019 0.01420
## T+2   0.0004019 0.01430
## T+3   0.0004019 0.01440
## T+4   0.0004019 0.01450
## T+5   0.0004019 0.01460
## T+6   0.0004019 0.01469
## T+7   0.0004019 0.01478
## T+8   0.0004019 0.01488
## T+9   0.0004019 0.01496
## T+10  0.0004019 0.01505
## T+11  0.0004019 0.01514
## T+12  0.0004019 0.01522
## T+13  0.0004019 0.01530
## T+14  0.0004019 0.01538
## T+15  0.0004019 0.01546
## T+16  0.0004019 0.01554
## T+17  0.0004019 0.01561
## T+18  0.0004019 0.01569
## T+19  0.0004019 0.01576
## T+20  0.0004019 0.01583
## T+21  0.0004019 0.01590
## T+22  0.0004019 0.01597
## T+23  0.0004019 0.01604
## T+24  0.0004019 0.01610
## T+25  0.0004019 0.01617
## T+26  0.0004019 0.01623
## T+27  0.0004019 0.01630
## T+28  0.0004019 0.01636
## T+29  0.0004019 0.01642
## T+30  0.0004019 0.01648
## T+31  0.0004019 0.01654
## T+32  0.0004019 0.01659
## T+33  0.0004019 0.01665
## T+34  0.0004019 0.01671
## T+35  0.0004019 0.01676
```

```
## T+36  0.0004019 0.01681
## T+37  0.0004019 0.01687
## T+38  0.0004019 0.01692
## T+39  0.0004019 0.01697
## T+40  0.0004019 0.01702
## T+41  0.0004019 0.01707
## T+42  0.0004019 0.01712
## T+43  0.0004019 0.01716
## T+44  0.0004019 0.01721
## T+45  0.0004019 0.01726
## T+46  0.0004019 0.01730
## T+47  0.0004019 0.01735
## T+48  0.0004019 0.01739
## T+49  0.0004019 0.01743
## T+50  0.0004019 0.01747
## T+51  0.0004019 0.01752
## T+52  0.0004019 0.01756
## T+53  0.0004019 0.01760
## T+54  0.0004019 0.01764
## T+55  0.0004019 0.01768
## T+56  0.0004019 0.01771
## T+57  0.0004019 0.01775
## T+58  0.0004019 0.01779
## T+59  0.0004019 0.01782
## T+60  0.0004019 0.01786
## T+61  0.0004019 0.01790
## T+62  0.0004019 0.01793
## T+63  0.0004019 0.01796
## T+64  0.0004019 0.01800
## T+65  0.0004019 0.01803
## T+66  0.0004019 0.01806
## T+67  0.0004019 0.01809
## T+68  0.0004019 0.01813
## T+69  0.0004019 0.01816
## T+70  0.0004019 0.01819
## T+71  0.0004019 0.01822
## T+72  0.0004019 0.01825
## T+73  0.0004019 0.01828
## T+74  0.0004019 0.01830
## T+75  0.0004019 0.01833
## T+76  0.0004019 0.01836
## T+77  0.0004019 0.01839
## T+78  0.0004019 0.01841
## T+79  0.0004019 0.01844
## T+80  0.0004019 0.01847
## T+81  0.0004019 0.01849
## T+82  0.0004019 0.01852
## T+83  0.0004019 0.01854
## T+84  0.0004019 0.01857
## T+85  0.0004019 0.01859
## T+86  0.0004019 0.01861
## T+87  0.0004019 0.01864
## T+88  0.0004019 0.01866
## T+89  0.0004019 0.01868
```

```
## T+90   0.0004019 0.01871
## T+91   0.0004019 0.01873
## T+92   0.0004019 0.01875
## T+93   0.0004019 0.01877
## T+94   0.0004019 0.01879
## T+95   0.0004019 0.01881
## T+96   0.0004019 0.01883
## T+97   0.0004019 0.01885
## T+98   0.0004019 0.01887
## T+99   0.0004019 0.01889
## T+100 0.0004019 0.01891
```

The plot for volatility forecasts

```
plot(garch11_fcst, which = 3)
garch11_uncon_vol = sqrt(uncvariance(garch11))
abline(h = garch11_uncon_vol, col = c('blue'))
```



The volatility of GARCH(1,1) does not converge to the unconditional volatility so fast because the fitted GARCH(1,1) may not be second order stationary. See the summary of GARCH(1,1) in Question 3b. In "Robust Standard Errors" section, the p-value of α estimate is 0.19, which is not a statistically significant value if our confidence level is 95%. It means that α may be greater than 0.07, leading to $\alpha + \beta > 1$, which violates the second-order stationary condition.

Q5

The 1-day conditional VaR is

$$VaR_{\alpha,t} = -(\hat{\mu}_t + \hat{\sigma}_{t+1} \times q^z_{1-\alpha})$$

where

- $\hat{\mu}_t$: the sample mean of MSFT returns based on t
- $\hat{\sigma}_{t+1}$: the sample standard deviation of MSFT returns based on t. We also have $\mathbb{E}(\sigma^2_{t+1}|\mathcal{F}_t) = \omega + \alpha\hat{\epsilon}^2_t + \beta\hat{\sigma}^2_t$.
- $q^z_{1-\alpha}$: the $(1-\alpha)$ quantile of the standard normal variable

The h-day conditional VaR is

$$VaR_{\alpha,n} = -(h\hat{\mu}_n + \sqrt{h}\hat{\sigma}_n \times q^z_{1-\alpha})$$

where

- n is the last return observation from estimation window, and $\sigma^2_n(h) = \mathbb{E}(\sigma^2_{n+1}|\mathcal{F}_n + ... + \mathbb{E}(\sigma^2_{n+h}|\mathcal{F}_n$ where $\mathbb{E}(\sigma^2_{n+1}|\mathcal{F}_n, ..., \mathbb{E}(\sigma^2_{n+h}|\mathcal{F}_n = \omega\sum_{i=0}^{h-1}(\alpha+\beta)^i + (\alpha+\beta)^{h-1}(\alpha\hat{\epsilon}^2_t + \beta\hat{\sigma}^2_t)$ are h-step ahead forecasts of conditional variance of returns from GARCH(1,1)
- $\hat{\mu}_n$ is the sample mean of MSFT returns based on t
- $\hat{\sigma}_t$ is the sample standard deviation of MSFT returns based on t
- $q^z_{1-\alpha}$ is the $(1-\alpha)$ quantile of the standard normal variable

(b)

ugarchroll setup

```
spec = ugarchspec(mean.model = list(armaOrder=c(0,0)))
garch11_roll = ugarchroll(spec = spec,
                          data = ret,
                          n.ahead = 1,
                          forecast.length = 1000,
                          refit.every = 20,
                          refit.window = 'moving')
```

The ahead rolling estimates

```
garch11_roll@forecast$density$Mu
```

```
##    [1] 0.0002516148 0.0002516148 0.0002516148 0.0002516148 0.0002516148
##    [6] 0.0002516148 0.0002516148 0.0002516148 0.0002516148 0.0002516148
##   [11] 0.0002516148 0.0002516148 0.0002516148 0.0002516148 0.0002516148
##   [16] 0.0002516148 0.0002516148 0.0002516148 0.0002516148 0.0002516148
##   [21] 0.0002442970 0.0002442970 0.0002442970 0.0002442970 0.0002442970
##   [26] 0.0002442970 0.0002442970 0.0002442970 0.0002442970 0.0002442970
##   [31] 0.0002442970 0.0002442970 0.0002442970 0.0002442970 0.0002442970
##   [36] 0.0002442970 0.0002442970 0.0002442970 0.0002442970 0.0002442970
##   [41] 0.0002707215 0.0002707215 0.0002707215 0.0002707215 0.0002707215
##   [46] 0.0002707215 0.0002707215 0.0002707215 0.0002707215 0.0002707215
##   [51] 0.0002707215 0.0002707215 0.0002707215 0.0002707215 0.0002707215
##   [56] 0.0002707215 0.0002707215 0.0002707215 0.0002707215 0.0002707215
##   [61] 0.0002996353 0.0002996353 0.0002996353 0.0002996353 0.0002996353
##   [66] 0.0002996353 0.0002996353 0.0002996353 0.0002996353 0.0002996353
```

```
##   [71] 0.0002996353 0.0002996353 0.0002996353 0.0002996353 0.0002996353
##   [76] 0.0002996353 0.0002996353 0.0002996353 0.0002996353 0.0002996353
##   [81] 0.0003636781 0.0003636781 0.0003636781 0.0003636781 0.0003636781
##   [86] 0.0003636781 0.0003636781 0.0003636781 0.0003636781 0.0003636781
##   [91] 0.0003636781 0.0003636781 0.0003636781 0.0003636781 0.0003636781
##   [96] 0.0003636781 0.0003636781 0.0003636781 0.0003636781 0.0003636781
##  [101] 0.0004026738 0.0004026738 0.0004026738 0.0004026738 0.0004026738
##  [106] 0.0004026738 0.0004026738 0.0004026738 0.0004026738 0.0004026738
##  [111] 0.0004026738 0.0004026738 0.0004026738 0.0004026738 0.0004026738
##  [116] 0.0004026738 0.0004026738 0.0004026738 0.0004026738 0.0004026738
##  [121] 0.0003566800 0.0003566800 0.0003566800 0.0003566800 0.0003566800
##  [126] 0.0003566800 0.0003566800 0.0003566800 0.0003566800 0.0003566800
##  [131] 0.0003566800 0.0003566800 0.0003566800 0.0003566800 0.0003566800
##  [136] 0.0003566800 0.0003566800 0.0003566800 0.0003566800 0.0003566800
##  [141] 0.0003704028 0.0003704028 0.0003704028 0.0003704028 0.0003704028
##  [146] 0.0003704028 0.0003704028 0.0003704028 0.0003704028 0.0003704028
##  [151] 0.0003704028 0.0003704028 0.0003704028 0.0003704028 0.0003704028
##  [156] 0.0003704028 0.0003704028 0.0003704028 0.0003704028 0.0003704028
##  [161] 0.0003521799 0.0003521799 0.0003521799 0.0003521799 0.0003521799
##  [166] 0.0003521799 0.0003521799 0.0003521799 0.0003521799 0.0003521799
##  [171] 0.0003521799 0.0003521799 0.0003521799 0.0003521799 0.0003521799
##  [176] 0.0003521799 0.0003521799 0.0003521799 0.0003521799 0.0003521799
##  [181] 0.0003590397 0.0003590397 0.0003590397 0.0003590397 0.0003590397
##  [186] 0.0003590397 0.0003590397 0.0003590397 0.0003590397 0.0003590397
##  [191] 0.0003590397 0.0003590397 0.0003590397 0.0003590397 0.0003590397
##  [196] 0.0003590397 0.0003590397 0.0003590397 0.0003590397 0.0003590397
##  [201] 0.0003612664 0.0003612664 0.0003612664 0.0003612664 0.0003612664
##  [206] 0.0003612664 0.0003612664 0.0003612664 0.0003612664 0.0003612664
##  [211] 0.0003612664 0.0003612664 0.0003612664 0.0003612664 0.0003612664
##  [216] 0.0003612664 0.0003612664 0.0003612664 0.0003612664 0.0003612664
##  [221] 0.0003589680 0.0003589680 0.0003589680 0.0003589680 0.0003589680
##  [226] 0.0003589680 0.0003589680 0.0003589680 0.0003589680 0.0003589680
##  [231] 0.0003589680 0.0003589680 0.0003589680 0.0003589680 0.0003589680
##  [236] 0.0003589680 0.0003589680 0.0003589680 0.0003589680 0.0003589680
##  [241] 0.0003403907 0.0003403907 0.0003403907 0.0003403907 0.0003403907
##  [246] 0.0003403907 0.0003403907 0.0003403907 0.0003403907 0.0003403907
##  [251] 0.0003403907 0.0003403907 0.0003403907 0.0003403907 0.0003403907
##  [256] 0.0003403907 0.0003403907 0.0003403907 0.0003403907 0.0003403907
##  [261] 0.0003448883 0.0003448883 0.0003448883 0.0003448883 0.0003448883
##  [266] 0.0003448883 0.0003448883 0.0003448883 0.0003448883 0.0003448883
##  [271] 0.0003448883 0.0003448883 0.0003448883 0.0003448883 0.0003448883
##  [276] 0.0003448883 0.0003448883 0.0003448883 0.0003448883 0.0003448883
##  [281] 0.0002934802 0.0002934802 0.0002934802 0.0002934802 0.0002934802
##  [286] 0.0002934802 0.0002934802 0.0002934802 0.0002934802 0.0002934802
##  [291] 0.0002934802 0.0002934802 0.0002934802 0.0002934802 0.0002934802
##  [296] 0.0002934802 0.0002934802 0.0002934802 0.0002934802 0.0002934802
##  [301] 0.0003133008 0.0003133008 0.0003133008 0.0003133008 0.0003133008
##  [306] 0.0003133008 0.0003133008 0.0003133008 0.0003133008 0.0003133008
##  [311] 0.0003133008 0.0003133008 0.0003133008 0.0003133008 0.0003133008
##  [316] 0.0003133008 0.0003133008 0.0003133008 0.0003133008 0.0003133008
##  [321] 0.0003063021 0.0003063021 0.0003063021 0.0003063021 0.0003063021
##  [326] 0.0003063021 0.0003063021 0.0003063021 0.0003063021 0.0003063021
##  [331] 0.0003063021 0.0003063021 0.0003063021 0.0003063021 0.0003063021
##  [336] 0.0003063021 0.0003063021 0.0003063021 0.0003063021 0.0003063021
```

```
##  [341] 0.0002974739 0.0002974739 0.0002974739 0.0002974739 0.0002974739
##  [346] 0.0002974739 0.0002974739 0.0002974739 0.0002974739 0.0002974739
##  [351] 0.0002974739 0.0002974739 0.0002974739 0.0002974739 0.0002974739
##  [356] 0.0002974739 0.0002974739 0.0002974739 0.0002974739 0.0002974739
##  [361] 0.0002942707 0.0002942707 0.0002942707 0.0002942707 0.0002942707
##  [366] 0.0002942707 0.0002942707 0.0002942707 0.0002942707 0.0002942707
##  [371] 0.0002942707 0.0002942707 0.0002942707 0.0002942707 0.0002942707
##  [376] 0.0002942707 0.0002942707 0.0002942707 0.0002942707 0.0002942707
##  [381] 0.0003455463 0.0003455463 0.0003455463 0.0003455463 0.0003455463
##  [386] 0.0003455463 0.0003455463 0.0003455463 0.0003455463 0.0003455463
##  [391] 0.0003455463 0.0003455463 0.0003455463 0.0003455463 0.0003455463
##  [396] 0.0003455463 0.0003455463 0.0003455463 0.0003455463 0.0003455463
##  [401] 0.0003670072 0.0003670072 0.0003670072 0.0003670072 0.0003670072
##  [406] 0.0003670072 0.0003670072 0.0003670072 0.0003670072 0.0003670072
##  [411] 0.0003670072 0.0003670072 0.0003670072 0.0003670072 0.0003670072
##  [416] 0.0003670072 0.0003670072 0.0003670072 0.0003670072 0.0003670072
##  [421] 0.0003761813 0.0003761813 0.0003761813 0.0003761813 0.0003761813
##  [426] 0.0003761813 0.0003761813 0.0003761813 0.0003761813 0.0003761813
##  [431] 0.0003761813 0.0003761813 0.0003761813 0.0003761813 0.0003761813
##  [436] 0.0003761813 0.0003761813 0.0003761813 0.0003761813 0.0003761813
##  [441] 0.0003056000 0.0003056000 0.0003056000 0.0003056000 0.0003056000
##  [446] 0.0003056000 0.0003056000 0.0003056000 0.0003056000 0.0003056000
##  [451] 0.0003056000 0.0003056000 0.0003056000 0.0003056000 0.0003056000
##  [456] 0.0003056000 0.0003056000 0.0003056000 0.0003056000 0.0003056000
##  [461] 0.0003162425 0.0003162425 0.0003162425 0.0003162425 0.0003162425
##  [466] 0.0003162425 0.0003162425 0.0003162425 0.0003162425 0.0003162425
##  [471] 0.0003162425 0.0003162425 0.0003162425 0.0003162425 0.0003162425
##  [476] 0.0003162425 0.0003162425 0.0003162425 0.0003162425 0.0003162425
##  [481] 0.0002967143 0.0002967143 0.0002967143 0.0002967143 0.0002967143
##  [486] 0.0002967143 0.0002967143 0.0002967143 0.0002967143 0.0002967143
##  [491] 0.0002967143 0.0002967143 0.0002967143 0.0002967143 0.0002967143
##  [496] 0.0002967143 0.0002967143 0.0002967143 0.0002967143 0.0002967143
##  [501] 0.0003211049 0.0003211049 0.0003211049 0.0003211049 0.0003211049
##  [506] 0.0003211049 0.0003211049 0.0003211049 0.0003211049 0.0003211049
##  [511] 0.0003211049 0.0003211049 0.0003211049 0.0003211049 0.0003211049
##  [516] 0.0003211049 0.0003211049 0.0003211049 0.0003211049 0.0003211049
##  [521] 0.0003423612 0.0003423612 0.0003423612 0.0003423612 0.0003423612
##  [526] 0.0003423612 0.0003423612 0.0003423612 0.0003423612 0.0003423612
##  [531] 0.0003423612 0.0003423612 0.0003423612 0.0003423612 0.0003423612
##  [536] 0.0003423612 0.0003423612 0.0003423612 0.0003423612 0.0003423612
##  [541] 0.0003471599 0.0003471599 0.0003471599 0.0003471599 0.0003471599
##  [546] 0.0003471599 0.0003471599 0.0003471599 0.0003471599 0.0003471599
##  [551] 0.0003471599 0.0003471599 0.0003471599 0.0003471599 0.0003471599
##  [556] 0.0003471599 0.0003471599 0.0003471599 0.0003471599 0.0003471599
##  [561] 0.0003525277 0.0003525277 0.0003525277 0.0003525277 0.0003525277
##  [566] 0.0003525277 0.0003525277 0.0003525277 0.0003525277 0.0003525277
##  [571] 0.0003525277 0.0003525277 0.0003525277 0.0003525277 0.0003525277
##  [576] 0.0003525277 0.0003525277 0.0003525277 0.0003525277 0.0003525277
##  [581] 0.0003784092 0.0003784092 0.0003784092 0.0003784092 0.0003784092
##  [586] 0.0003784092 0.0003784092 0.0003784092 0.0003784092 0.0003784092
##  [591] 0.0003784092 0.0003784092 0.0003784092 0.0003784092 0.0003784092
##  [596] 0.0003784092 0.0003784092 0.0003784092 0.0003784092 0.0003784092
##  [601] 0.0003773196 0.0003773196 0.0003773196 0.0003773196 0.0003773196
##  [606] 0.0003773196 0.0003773196 0.0003773196 0.0003773196 0.0003773196
```

```
##  [611] 0.0003773196 0.0003773196 0.0003773196 0.0003773196 0.0003773196
##  [616] 0.0003773196 0.0003773196 0.0003773196 0.0003773196 0.0003773196
##  [621] 0.0003874375 0.0003874375 0.0003874375 0.0003874375 0.0003874375
##  [626] 0.0003874375 0.0003874375 0.0003874375 0.0003874375 0.0003874375
##  [631] 0.0003874375 0.0003874375 0.0003874375 0.0003874375 0.0003874375
##  [636] 0.0003874375 0.0003874375 0.0003874375 0.0003874375 0.0003874375
##  [641] 0.0003956570 0.0003956570 0.0003956570 0.0003956570 0.0003956570
##  [646] 0.0003956570 0.0003956570 0.0003956570 0.0003956570 0.0003956570
##  [651] 0.0003956570 0.0003956570 0.0003956570 0.0003956570 0.0003956570
##  [656] 0.0003956570 0.0003956570 0.0003956570 0.0003956570 0.0003956570
##  [661] 0.0003915292 0.0003915292 0.0003915292 0.0003915292 0.0003915292
##  [666] 0.0003915292 0.0003915292 0.0003915292 0.0003915292 0.0003915292
##  [671] 0.0003915292 0.0003915292 0.0003915292 0.0003915292 0.0003915292
##  [676] 0.0003915292 0.0003915292 0.0003915292 0.0003915292 0.0003915292
##  [681] 0.0004156257 0.0004156257 0.0004156257 0.0004156257 0.0004156257
##  [686] 0.0004156257 0.0004156257 0.0004156257 0.0004156257 0.0004156257
##  [691] 0.0004156257 0.0004156257 0.0004156257 0.0004156257 0.0004156257
##  [696] 0.0004156257 0.0004156257 0.0004156257 0.0004156257 0.0004156257
##  [701] 0.0004241334 0.0004241334 0.0004241334 0.0004241334 0.0004241334
##  [706] 0.0004241334 0.0004241334 0.0004241334 0.0004241334 0.0004241334
##  [711] 0.0004241334 0.0004241334 0.0004241334 0.0004241334 0.0004241334
##  [716] 0.0004241334 0.0004241334 0.0004241334 0.0004241334 0.0004241334
##  [721] 0.0004415483 0.0004415483 0.0004415483 0.0004415483 0.0004415483
##  [726] 0.0004415483 0.0004415483 0.0004415483 0.0004415483 0.0004415483
##  [731] 0.0004415483 0.0004415483 0.0004415483 0.0004415483 0.0004415483
##  [736] 0.0004415483 0.0004415483 0.0004415483 0.0004415483 0.0004415483
##  [741] 0.0004576880 0.0004576880 0.0004576880 0.0004576880 0.0004576880
##  [746] 0.0004576880 0.0004576880 0.0004576880 0.0004576880 0.0004576880
##  [751] 0.0004576880 0.0004576880 0.0004576880 0.0004576880 0.0004576880
##  [756] 0.0004576880 0.0004576880 0.0004576880 0.0004576880 0.0004576880
##  [761] 0.0004528165 0.0004528165 0.0004528165 0.0004528165 0.0004528165
##  [766] 0.0004528165 0.0004528165 0.0004528165 0.0004528165 0.0004528165
##  [771] 0.0004528165 0.0004528165 0.0004528165 0.0004528165 0.0004528165
##  [776] 0.0004528165 0.0004528165 0.0004528165 0.0004528165 0.0004528165
##  [781] 0.0005044616 0.0005044616 0.0005044616 0.0005044616 0.0005044616
##  [786] 0.0005044616 0.0005044616 0.0005044616 0.0005044616 0.0005044616
##  [791] 0.0005044616 0.0005044616 0.0005044616 0.0005044616 0.0005044616
##  [796] 0.0005044616 0.0005044616 0.0005044616 0.0005044616 0.0005044616
##  [801] 0.0004770593 0.0004770593 0.0004770593 0.0004770593 0.0004770593
##  [806] 0.0004770593 0.0004770593 0.0004770593 0.0004770593 0.0004770593
##  [811] 0.0004770593 0.0004770593 0.0004770593 0.0004770593 0.0004770593
##  [816] 0.0004770593 0.0004770593 0.0004770593 0.0004770593 0.0004770593
##  [821] 0.0004554233 0.0004554233 0.0004554233 0.0004554233 0.0004554233
##  [826] 0.0004554233 0.0004554233 0.0004554233 0.0004554233 0.0004554233
##  [831] 0.0004554233 0.0004554233 0.0004554233 0.0004554233 0.0004554233
##  [836] 0.0004554233 0.0004554233 0.0004554233 0.0004554233 0.0004554233
##  [841] 0.0004413853 0.0004413853 0.0004413853 0.0004413853 0.0004413853
##  [846] 0.0004413853 0.0004413853 0.0004413853 0.0004413853 0.0004413853
##  [851] 0.0004413853 0.0004413853 0.0004413853 0.0004413853 0.0004413853
##  [856] 0.0004413853 0.0004413853 0.0004413853 0.0004413853 0.0004413853
##  [861] 0.0004462245 0.0004462245 0.0004462245 0.0004462245 0.0004462245
##  [866] 0.0004462245 0.0004462245 0.0004462245 0.0004462245 0.0004462245
##  [871] 0.0004462245 0.0004462245 0.0004462245 0.0004462245 0.0004462245
##  [876] 0.0004462245 0.0004462245 0.0004462245 0.0004462245 0.0004462245
```

```
## [881] 0.0004399593 0.0004399593 0.0004399593 0.0004399593 0.0004399593
## [886] 0.0004399593 0.0004399593 0.0004399593 0.0004399593 0.0004399593
## [891] 0.0004399593 0.0004399593 0.0004399593 0.0004399593 0.0004399593
## [896] 0.0004399593 0.0004399593 0.0004399593 0.0004399593 0.0004399593
## [901] 0.0004604424 0.0004604424 0.0004604424 0.0004604424 0.0004604424
## [906] 0.0004604424 0.0004604424 0.0004604424 0.0004604424 0.0004604424
## [911] 0.0004604424 0.0004604424 0.0004604424 0.0004604424 0.0004604424
## [916] 0.0004604424 0.0004604424 0.0004604424 0.0004604424 0.0004604424
## [921] 0.0004356870 0.0004356870 0.0004356870 0.0004356870 0.0004356870
## [926] 0.0004356870 0.0004356870 0.0004356870 0.0004356870 0.0004356870
## [931] 0.0004356870 0.0004356870 0.0004356870 0.0004356870 0.0004356870
## [936] 0.0004356870 0.0004356870 0.0004356870 0.0004356870 0.0004356870
## [941] 0.0004159973 0.0004159973 0.0004159973 0.0004159973 0.0004159973
## [946] 0.0004159973 0.0004159973 0.0004159973 0.0004159973 0.0004159973
## [951] 0.0004159973 0.0004159973 0.0004159973 0.0004159973 0.0004159973
## [956] 0.0004159973 0.0004159973 0.0004159973 0.0004159973 0.0004159973
## [961] 0.0004584042 0.0004584042 0.0004584042 0.0004584042 0.0004584042
## [966] 0.0004584042 0.0004584042 0.0004584042 0.0004584042 0.0004584042
## [971] 0.0004584042 0.0004584042 0.0004584042 0.0004584042 0.0004584042
## [976] 0.0004584042 0.0004584042 0.0004584042 0.0004584042 0.0004584042
## [981] 0.0004704365 0.0004704365 0.0004704365 0.0004704365 0.0004704365
## [986] 0.0004704365 0.0004704365 0.0004704365 0.0004704365 0.0004704365
## [991] 0.0004704365 0.0004704365 0.0004704365 0.0004704365 0.0004704365
## [996] 0.0004704365 0.0004704365 0.0004704365 0.0004704365 0.0004704365
```

The calculated 1% VaR list

```
garch11_roll@forecast$VaR$`alpha(1%)`
```

```
##   [1] -0.03381362 -0.03399355 -0.03336217 -0.03581285 -0.03524338 -0.03513557
##   [7] -0.03741406 -0.03615728 -0.03710535 -0.03652132 -0.03763231 -0.03730104
##  [13] -0.04277363 -0.04127072 -0.04250243 -0.04114305 -0.03990112 -0.04181092
##  [19] -0.04242892 -0.04136666 -0.04065089 -0.03960298 -0.03914089 -0.03806408
##  [25] -0.03956023 -0.03819416 -0.04128203 -0.04054473 -0.03915076 -0.03960779
##  [31] -0.03824521 -0.03697511 -0.03610596 -0.03567405 -0.03485074 -0.03426845
##  [37] -0.03346696 -0.03243719 -0.03322434 -0.03427640 -0.03526083 -0.03477628
##  [43] -0.03365171 -0.03299926 -0.03196092 -0.03148916 -0.03091179 -0.03004347
##  [49] -0.03512321 -0.03703072 -0.03632141 -0.03641410 -0.03623455 -0.03507502
##  [55] -0.03404406 -0.03353737 -0.03293526 -0.03190064 -0.03093906 -0.03014274
##  [61] -0.04524825 -0.04359108 -0.04286887 -0.04157897 -0.04013758 -0.03920285
##  [67] -0.03868302 -0.03741605 -0.03692533 -0.03572182 -0.03506329 -0.03396176
##  [73] -0.03303656 -0.03262844 -0.03171659 -0.03131850 -0.03045056 -0.02973905
##  [79] -0.02925616 -0.03804544 -0.03622678 -0.03533498 -0.03449524 -0.03356993
##  [85] -0.03270110 -0.03193336 -0.03249737 -0.03173807 -0.03244768 -0.03184096
##  [91] -0.03146698 -0.03100450 -0.03061973 -0.02999039 -0.02925797 -0.02856711
##  [97] -0.03007260 -0.02937264 -0.02869287 -0.02841411 -0.02860566 -0.02822563
## [103] -0.02762030 -0.02703276 -0.02644069 -0.02787941 -0.02728154 -0.02739970
## [109] -0.02683016 -0.02635264 -0.02580113 -0.02614430 -0.02912656 -0.02896492
## [115] -0.02942002 -0.03123192 -0.03052024 -0.03203063 -0.03136382 -0.03093029
## [121] -0.03086748 -0.03063558 -0.02996441 -0.03921674 -0.03842160 -0.03744082
## [127] -0.03664093 -0.03559077 -0.03464248 -0.03364680 -0.03269988 -0.03206438
## [133] -0.03121141 -0.03387102 -0.03355853 -0.03276275 -0.03222177 -0.03135159
## [139] -0.03151100 -0.03208555 -0.03170282 -0.03149614 -0.03088505 -0.03148463
## [145] -0.03207418 -0.03119400 -0.03319311 -0.03227073 -0.03138301 -0.03212629
## [151] -0.03159259 -0.03091299 -0.03388160 -0.03295987 -0.03205038 -0.03556971
```

```
## [157] -0.03461105 -0.03460975 -0.03701631 -0.03671812 -0.03574035 -0.03492307
## [163] -0.03645737 -0.03556969 -0.03833183 -0.03739033 -0.03945666 -0.03969688
## [169] -0.04183472 -0.04058132 -0.03938899 -0.03853030 -0.03958349 -0.03835080
## [175] -0.03734936 -0.03624246 -0.03659996 -0.03572371 -0.03508523 -0.03524715
## [181] -0.03672399 -0.03859601 -0.03750092 -0.03666776 -0.03856509 -0.03766402
## [187] -0.03807003 -0.04040257 -0.03931519 -0.03872427 -0.03813188 -0.03874916
## [193] -0.03772223 -0.03682174 -0.03582393 -0.03512989 -0.03578058 -0.03498934
## [199] -0.03451217 -0.03361207 -0.03301108 -0.03232622 -0.03163992 -0.03099734
## [205] -0.03037417 -0.03126496 -0.03066776 -0.03017649 -0.02958183 -0.02935362
## [211] -0.02919863 -0.02904644 -0.02855770 -0.02804540 -0.02753033 -0.02763993
## [217] -0.02835623 -0.02879367 -0.02824665 -0.03190025 -0.03260039 -0.03204085
## [223] -0.03120386 -0.03039393 -0.02975870 -0.02942891 -0.02871361 -0.02804789
## [229] -0.02753663 -0.02788835 -0.02772306 -0.02823930 -0.02863096 -0.02831114
## [235] -0.02766038 -0.02820213 -0.02810365 -0.02764115 -0.02727920 -0.02686410
## [241] -0.02652572 -0.02605176 -0.02731107 -0.02740084 -0.02684814 -0.02671288
## [247] -0.02679740 -0.02627194 -0.02582436 -0.02541547 -0.03001555 -0.03198545
## [253] -0.03120047 -0.03060903 -0.02988711 -0.02988204 -0.02986116 -0.03461539
## [259] -0.03369928 -0.03289950 -0.03217890 -0.03463165 -0.03416148 -0.03330540
## [265] -0.03457003 -0.03818622 -0.03751985 -0.03671780 -0.03587832 -0.03518554
## [271] -0.03428845 -0.03375180 -0.03606426 -0.03568552 -0.03534169 -0.03485459
## [277] -0.03499022 -0.03478557 -0.03414610 -0.03332059 -0.03309981 -0.03227723
## [283] -0.03200932 -0.03273834 -0.03281627 -0.03203941 -0.03137748 -0.03127401
## [289] -0.03107340 -0.03169900 -0.03136661 -0.03147574 -0.03108818 -0.03141625
## [295] -0.03095655 -0.03028024 -0.03082980 -0.03026751 -0.03487572 -0.03480792
## [301] -0.03537087 -0.03446817 -0.03372444 -0.03300550 -0.03259553 -0.03265836
## [307] -0.03187759 -0.03174962 -0.03120346 -0.03074459 -0.03003833 -0.02944720
## [313] -0.03018618 -0.02950749 -0.02925144 -0.02862107 -0.02807613 -0.02770008
## [319] -0.02832549 -0.02926261 -0.03011799 -0.02953657 -0.02913187 -0.02860484
## [325] -0.02848784 -0.02850184 -0.02797422 -0.02758867 -0.02709989 -0.02664180
## [331] -0.02662355 -0.02640069 -0.02692588 -0.02689990 -0.02744253 -0.02696060
## [337] -0.02766562 -0.02717557 -0.02682915 -0.02660076 -0.02630040 -0.02631387
## [343] -0.02587986 -0.02559908 -0.02531499 -0.02492284 -0.02454234 -0.02448913
## [349] -0.02421048 -0.02388269 -0.02358834 -0.02339480 -0.02342790 -0.02324369
## [355] -0.02301430 -0.02272702 -0.02275038 -0.02285998 -0.02258043 -0.02254932
## [361] -0.02248813 -0.02221806 -0.02201577 -0.02186593 -0.02806436 -0.02978562
## [367] -0.03745383 -0.03664636 -0.03581327 -0.03529521 -0.03456338 -0.03376975
## [373] -0.03716789 -0.04043719 -0.03961148 -0.04305414 -0.04201881 -0.04101176
## [379] -0.04201840 -0.04161937 -0.04143502 -0.04088963 -0.04009977 -0.03921602
## [385] -0.03883734 -0.03818018 -0.03758717 -0.03661126 -0.03606918 -0.03699489
## [391] -0.03636959 -0.03560794 -0.03672962 -0.03591470 -0.03520106 -0.03454101
## [397] -0.03447409 -0.03367139 -0.03475742 -0.03397931 -0.03309154 -0.03235985
## [403] -0.03323395 -0.03374047 -0.03308355 -0.03238041 -0.03334187 -0.03268327
## [409] -0.03337793 -0.03263739 -0.03217051 -0.03183013 -0.03238696 -0.03161687
## [415] -0.03149983 -0.03543592 -0.03472678 -0.03456925 -0.03371097 -0.03453060
## [421] -0.03380124 -0.03300981 -0.03235948 -0.03230914 -0.03156625 -0.03098148
## [427] -0.03033192 -0.02966736 -0.02949150 -0.03240283 -0.03164697 -0.03174520
## [433] -0.03104105 -0.03138203 -0.03102231 -0.07131680 -0.06975552 -0.06760912
## [439] -0.06549241 -0.06416782 -0.05993461 -0.05840822 -0.05710037 -0.05564698
## [445] -0.05430870 -0.05304085 -0.05180569 -0.05055888 -0.04983635 -0.05014846
## [451] -0.04900575 -0.04786890 -0.04713437 -0.04602948 -0.04575795 -0.04466836
## [457] -0.04409270 -0.04319320 -0.04219021 -0.04285639 -0.05360000 -0.05300513
## [463] -0.05328192 -0.05215087 -0.05146344 -0.05031530 -0.05392676 -0.05366178
## [469] -0.05239752 -0.05118830 -0.05056168 -0.05052160 -0.04959679 -0.04846165
## [475] -0.04759032 -0.04663664 -0.04561661 -0.04492359 -0.04414101 -0.04487121
```

```
## [481] -0.04378782 -0.04304291 -0.04210874 -0.04135640 -0.04107102 -0.04019334
## [487] -0.03955626 -0.03900112 -0.03820296 -0.03741882 -0.03758432 -0.03706648
## [493] -0.03632681 -0.03687548 -0.03647935 -0.03600516 -0.03529635 -0.03466165
## [499] -0.03419208 -0.03354512 -0.03191838 -0.03185778 -0.03345489 -0.03277914
## [505] -0.04286377 -0.04189972 -0.04091155 -0.03994998 -0.03907230 -0.03820687
## [511] -0.03774578 -0.03806538 -0.04230864 -0.04227079 -0.04140961 -0.04051643
## [517] -0.03969680 -0.04012020 -0.03923643 -0.03841649 -0.03860465 -0.03788831
## [523] -0.03734854 -0.03679824 -0.03611585 -0.03540995 -0.03492676 -0.03438239
## [529] -0.03433071 -0.03388749 -0.03330315 -0.03353296 -0.03489897 -0.03449143
## [535] -0.03408093 -0.03426479 -0.03420988 -0.03393928 -0.03400493 -0.03344932
## [541] -0.03301189 -0.03238967 -0.03209868 -0.03227357 -0.03177116 -0.03171966
## [547] -0.03145353 -0.03095292 -0.03040186 -0.02990243 -0.02958205 -0.02927847
## [553] -0.03058863 -0.03025515 -0.03098818 -0.03060700 -0.03078853 -0.03348762
## [559] -0.03451178 -0.03613701 -0.03542710 -0.03536815 -0.03476670 -0.03422775
## [565] -0.03357937 -0.03432037 -0.03521743 -0.03462798 -0.03428954 -0.03369102
## [571] -0.03528517 -0.03884855 -0.03804438 -0.03791571 -0.03738571 -0.03690613
## [577] -0.03624606 -0.03578875 -0.03524047 -0.03455654 -0.03375319 -0.03308926
## [583] -0.03245440 -0.03195980 -0.03147331 -0.03112753 -0.03062639 -0.03009253
## [589] -0.02994684 -0.02992972 -0.03019761 -0.03066108 -0.03036445 -0.02982344
## [595] -0.02949225 -0.02900943 -0.02861843 -0.02830262 -0.02828495 -0.02794800
## [601] -0.02755789 -0.03313728 -0.03265034 -0.03454437 -0.03386370 -0.03337536
## [607] -0.03273638 -0.03281410 -0.03261188 -0.03396580 -0.03424535 -0.03387216
## [613] -0.03315731 -0.03273988 -0.03507576 -0.03431224 -0.03356336 -0.03377746
## [619] -0.03591300 -0.03516149 -0.03440665 -0.03437528 -0.03453850 -0.03415009
## [625] -0.03342056 -0.03270752 -0.03226554 -0.03165863 -0.03102139 -0.03257671
## [631] -0.03222898 -0.03160216 -0.03139231 -0.03103318 -0.03061840 -0.03042168
## [637] -0.03015704 -0.02969586 -0.02917882 -0.02911858 -0.02993177 -0.02950983
## [643] -0.03002392 -0.02961854 -0.02914274 -0.02868216 -0.02924705 -0.02893123
## [649] -0.02846138 -0.02801785 -0.02768055 -0.02750797 -0.02792308 -0.02756080
## [655] -0.02779685 -0.02737419 -0.02878403 -0.02846615 -0.02813272 -0.02776835
## [661] -0.02747188 -0.02727283 -0.02784376 -0.02756498 -0.02720893 -0.02680967
## [667] -0.02648186 -0.02616693 -0.02601917 -0.02582840 -0.02565626 -0.02558500
## [673] -0.02590227 -0.02633572 -0.02602690 -0.02567418 -0.02537052 -0.02512259
## [679] -0.02494027 -0.02467690 -0.02422352 -0.02432937 -0.02404567 -0.02399672
## [685] -0.02957550 -0.02942599 -0.02895688 -0.02850206 -0.02803383 -0.02758459
## [691] -0.02764464 -0.02722488 -0.02744545 -0.02704110 -0.02686968 -0.02689855
## [697] -0.02672741 -0.02688288 -0.02669633 -0.02659867 -0.02666917 -0.02631798
## [703] -0.02597660 -0.02583765 -0.02611158 -0.02583644 -0.02600922 -0.02585483
## [709] -0.02597675 -0.02595385 -0.02574785 -0.02544405 -0.02513712 -0.02490666
## [715] -0.02467295 -0.02439789 -0.02470420 -0.02468604 -0.02445546 -0.02435014
## [721] -0.02474801 -0.02502535 -0.02486247 -0.02457457 -0.02432929 -0.02425644
## [727] -0.02444122 -0.02465021 -0.02449583 -0.02425457 -0.02524905 -0.02536126
## [733] -0.02554897 -0.02568556 -0.02738236 -0.02720323 -0.02681392 -0.02645811
## [739] -0.02652245 -0.02620806 -0.02597350 -0.02563171 -0.02594650 -0.02830404
## [745] -0.02938320 -0.03436372 -0.03396217 -0.03332702 -0.03317256 -0.03298381
## [751] -0.03364354 -0.03331205 -0.03364737 -0.03343870 -0.03341325 -0.03453836
## [757] -0.03396063 -0.03376883 -0.03315155 -0.03305001 -0.03374777 -0.03340658
## [763] -0.03277671 -0.03225900 -0.03261513 -0.03200059 -0.03145771 -0.03088768
## [769] -0.03035590 -0.03078084 -0.03023858 -0.02974354 -0.02952289 -0.02948456
## [775] -0.02931832 -0.02969335 -0.02946151 -0.02900039 -0.02863559 -0.02817756
## [781] -0.02862147 -0.02820324 -0.02802199 -0.02847317 -0.02832988 -0.02879481
## [787] -0.02834379 -0.02876386 -0.02838948 -0.02802982 -0.02774684 -0.03146797
## [793] -0.03140680 -0.03551720 -0.03478944 -0.03419355 -0.03377607 -0.03325424
## [799] -0.03271168 -0.03238142 -0.03203287 -0.03197778 -0.03151660 -0.03126609
```

```
## [805] -0.03149717 -0.03142737 -0.03353507 -0.03314192 -0.03305948 -0.03253264
## [811] -0.03219817 -0.03200151 -0.03227297 -0.03169761 -0.03150312 -0.03308932
## [817] -0.03245868 -0.03190628 -0.05482707 -0.05591921 -0.05686629 -0.05851944
## [823] -0.05783185 -0.05637679 -0.05492150 -0.05388095 -0.05245578 -0.05108132
## [829] -0.04981112 -0.04859805 -0.04798617 -0.04751241 -0.04630442 -0.04514591
## [835] -0.04402570 -0.04310546 -0.04215898 -0.04115301 -0.04019624 -0.03925746
## [841] -0.03838501 -0.03753220 -0.03731893 -0.03659941 -0.03581534 -0.03607318
## [847] -0.03569128 -0.03616380 -0.03540312 -0.03640502 -0.03584128 -0.03513280
## [853] -0.03443107 -0.03484749 -0.03422401 -0.03412205 -0.03343491 -0.03277298
## [859] -0.03604184 -0.03541012 -0.03463833 -0.03390737 -0.03341333 -0.03273456
## [865] -0.03251339 -0.03501790 -0.03427285 -0.03358786 -0.03290170 -0.03234073
## [871] -0.03170384 -0.03112778 -0.03127547 -0.03071221 -0.03081028 -0.03348273
## [877] -0.03295624 -0.03250346 -0.03206875 -0.05688642 -0.05373506 -0.05348683
## [883] -0.05225316 -0.05120825 -0.05003176 -0.04906446 -0.04835773 -0.04900125
## [889] -0.04804987 -0.04799301 -0.04707002 -0.04602287 -0.04507439 -0.04521434
## [895] -0.04441457 -0.04354742 -0.04263085 -0.04172319 -0.04088061 -0.04036297
## [901] -0.03960584 -0.03993737 -0.03918046 -0.03883677 -0.03821900 -0.03760999
## [907] -0.03690187 -0.03653008 -0.03592141 -0.03550188 -0.03486715 -0.03540016
## [913] -0.03480264 -0.03449881 -0.03424183 -0.03378619 -0.03321566 -0.03337496
## [919] -0.03336159 -0.03280248 -0.03199294 -0.03156895 -0.03103177 -0.03078420
## [925] -0.03162827 -0.03118402 -0.03079400 -0.03029442 -0.02980744 -0.02935897
## [931] -0.02891838 -0.02860660 -0.02819373 -0.02923958 -0.02879859 -0.02839369
## [937] -0.02926530 -0.02882304 -0.02851990 -0.02828833 -0.03263159 -0.03251475
## [943] -0.03202991 -0.03204021 -0.03152342 -0.03225351 -0.03217068 -0.03170218
## [949] -0.03120740 -0.03140662 -0.03091151 -0.03174449 -0.03125088 -0.03120801
## [955] -0.03192815 -0.03154723 -0.03104780 -0.03065285 -0.03031042 -0.02993591
## [961] -0.02992591 -0.03104373 -0.04130824 -0.04330722 -0.04457630 -0.05032919
## [967] -0.05067766 -0.04936350 -0.04832134 -0.05082807 -0.05233018 -0.05097529
## [973] -0.05065382 -0.05124296 -0.05074460 -0.04947697 -0.04824070 -0.04728204
## [979] -0.04716753 -0.04608993 -0.04461231 -0.04423337 -0.04338277 -0.04193707
## [985] -0.04046947 -0.03908182 -0.03777471 -0.03760888 -0.03643000 -0.03681933
## [991] -0.03590282 -0.03680652 -0.03793648 -0.03671712 -0.03555270 -0.03530620
## [997] -0.03451108 -0.03351978 -0.03259206 -0.03181206
```

(c)

The Mean Squared Error

```r
# Mean Squared Error
pred_ret = garch11_roll@forecast$density$Mu # predicted return
act_ret = garch11_roll@forecast$density$Realized # actual return
MSE = mean((pred_ret - act_ret)^2)
MSE
```

```
## [1] 0.00021283
```

The Mean Absolute Error

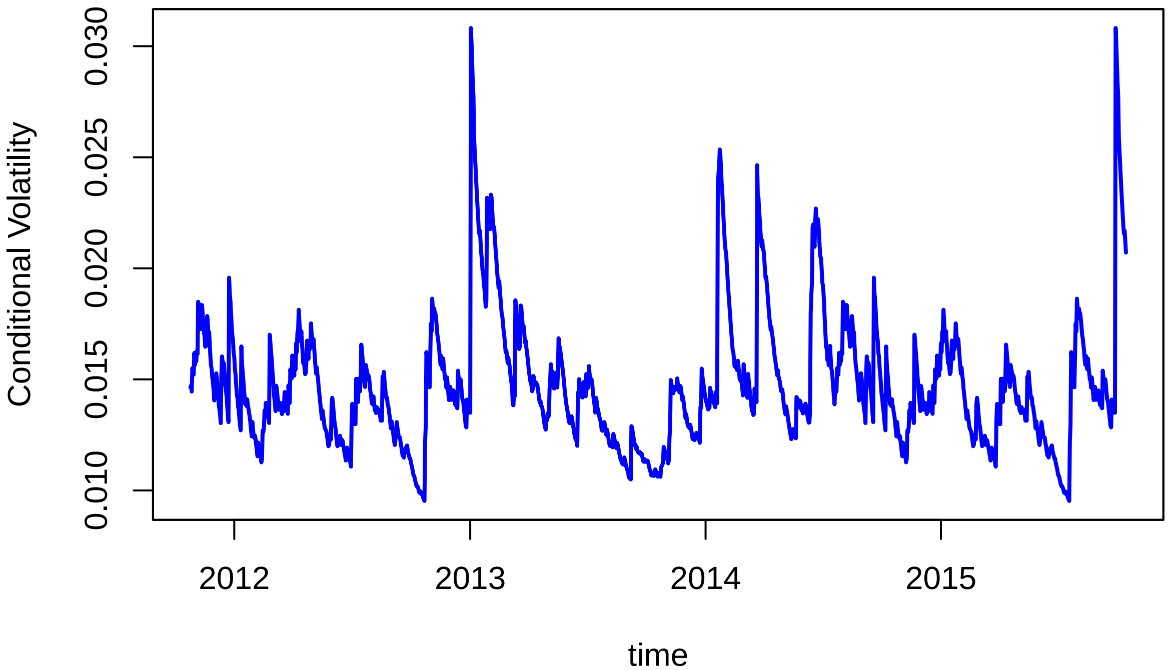
```r
# Mean Squared Error
pred_ret = garch11_roll@forecast$density$Mu # predicted return
act_ret = garch11_roll@forecast$density$Realized # actual return
MAE = mean(abs(pred_ret - act_ret))
MAE
```

```
## [1] 0.01018637
```

(d)

Plot the forecast of conditional volatility

```r
# dates
days <- seq(as.Date("2011-10-25"), as.Date("2015-10-15"), by = "days")
# create a time series
con_vol = zoo(garch11_roll@forecast$density$Sigma, days)
plot.zoo(con_vol, lwd=2, col="blue", xlab="time", ylab="Conditional Volatility")
```



All conditional volatility are within [0.01, 0.03], which are not very large. Most stay below 0.02. However, there are a few large spikes in 2013, 2014, and the end of 2015.