H4 - Q1 - Model Estimation of (G)ARCH Models

2023-11-15

```
# Packages
set.seed(11)
library(PerformanceAnalytics)
```

```
## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

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

##
## Attaching package: 'PerformanceAnalytics'

## The following object is masked from 'package:graphics':
##
##     legend
```

```
library(quantmod)
```

```
## Loading required package: TTR

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

```
library(rugarch)
```

```
## Loading required package: parallel

##
## Attaching package: 'rugarch'

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

```
library(FinTS)

# download data
getSymbols("MSFT", from = "2000-01-03", to = "2015-10-16")
```

```
## [1] "MSFT"
```

```
# extract adjusted closing prices
MSFT = MSFT[, "MSFT.Adjusted", drop = F]

# calculate log-returns
```

```r
MSFT.ret = CalculateReturns(MSFT, method = "log")
colnames(MSFT.ret) = "MSFT"

# remove first NA observation
MSFT.ret = MSFT.ret[-1,]
```

Q1 : Estimating (G)ARCH models using the ruGARCH package

Estimating (G)ARCH models for daily returns on Microsoft, using ARCH(5) and GARCH(1,1) as examples (conditional maximum likelihood approach). An ARCH(5) model and GARCH(1,1) model are contrasted.

(a) Estimation of an ARCH(5) model.

```r
MSFT.arch5.spec = ugarchspec(variance.model = list(garchOrder=c(5,0)), mean.model = list(armaOrder = c(
MSFT.arch5.fit = ugarchfit(MSFT.arch5.spec, MSFT.ret, solver.control=list(trace = 0))

print(coef(MSFT.arch5.fit))
```

(i) Estimated values of $\alpha_1 + \cdots + \alpha_5$.

```
##        mu     omega    alpha1    alpha2    alpha3    alpha4    alpha5
## 0.0005081 0.0001187 0.1525188 0.1879168 0.1414962 0.1699558 0.1356410
```

```r
print(sum(coef(MSFT.arch5.fit)[-(1:2)]))
```

```
## [1] 0.7875
```

Looking at the parameters, we see the mean is effectively zero and that the ARCH(5) model does not find a significant shift when fitting the volatility. The dependence of volatility in this model on the previous five returns is approximately equal ($\alpha_i = 0.14 - 0.19$) and positive. In other words, the fit model is approximately just a moving average with a multiplicative coefficient.

Additionally, a measure of the persistence is the sum of the (α) coefficients, which is 0.7876. There is significant persistence (as we can see in the data after a shock), but since $\sum_i \alpha_i = 0.7876 < 1$ the process is mean reverting.

```r
show(MSFT.arch5.fit)
```

(ii) Conditional maximum likelihood (CML) vs quasi-maximum likelihood (QML) standard error estimates

```
##
## *---------------------------------*
## *          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   2.0980 0.035905
## omega   0.000119    0.000006  18.5362 0.000000
## alpha1  0.152519    0.025401   6.0044 0.000000
## alpha2  0.187917    0.027371   6.8655 0.000000
## alpha3  0.141496    0.020561   6.8819 0.000000
## alpha4  0.169956    0.025570   6.6466 0.000000
## alpha5  0.135641    0.023127   5.8652 0.000000
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000508    0.000308   1.6484 0.099271
## omega   0.000119    0.000022   5.5085 0.000000
## alpha1  0.152519    0.047234   3.2290 0.001242
## alpha2  0.187917    0.051434   3.6536 0.000259
## alpha3  0.141496    0.046268   3.0582 0.002227
## alpha4  0.169956    0.047024   3.6143 0.000301
## alpha5  0.135641    0.041546   3.2648 0.001095
##
## LogLikelihood : 10422
##
## 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  0.2510
## 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]                      0.8651  0.3523
## 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]     0.4129 0.500 2.000  0.5205
## ARCH Lag[8]     1.0882 1.480 1.774  0.7399
## ARCH Lag[10]    1.1547 2.424 1.650  0.9152
```

```
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  3.572
## Individual Statistics:
## mu     0.3158
## omega  0.1906
## alpha1 0.5529
## alpha2 1.0552
## 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  0.1935 0.8466
## Positive Sign Bias  0.2777 0.7813
## Joint Effect        1.8108 0.6126
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     230.9    2.247e-38
## 2    30     233.7    7.237e-34
## 3    40     259.9    1.970e-34
## 4    50     277.8    1.036e-33
##
##
## Elapsed time : 0.503
```

```r
show(MSFT.arch5.fit)
```

```
##
## *---------------------------------*
## *          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   2.0980 0.035905
## omega   0.000119    0.000006  18.5362 0.000000
```

```
## alpha1  0.152519    0.025401   6.0044 0.000000
## alpha2  0.187917    0.027371   6.8655 0.000000
## alpha3  0.141496    0.020561   6.8819 0.000000
## alpha4  0.169956    0.025570   6.6466 0.000000
## alpha5  0.135641    0.023127   5.8652 0.000000
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000508    0.000308   1.6484 0.099271
## omega   0.000119    0.000022   5.5085 0.000000
## alpha1  0.152519    0.047234   3.2290 0.001242
## alpha2  0.187917    0.051434   3.6536 0.000259
## alpha3  0.141496    0.046268   3.0582 0.002227
## alpha4  0.169956    0.047024   3.6143 0.000301
## alpha5  0.135641    0.041546   3.2648 0.001095
##
## LogLikelihood : 10422
##
## 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  0.2510
## 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]                      0.8651  0.3523
## 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]     0.4129 0.500 2.000  0.5205
## ARCH Lag[8]     1.0882 1.480 1.774  0.7399
## ARCH Lag[10]    1.1547 2.424 1.650  0.9152
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  3.572
## Individual Statistics:
```

```
## mu     0.3158
## omega  0.1906
## alpha1 0.5529
## alpha2 1.0552
## 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  0.1935 0.8466
## Positive Sign Bias  0.2777 0.7813
## Joint Effect        1.8108 0.6126
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     230.9    2.247e-38
## 2    30     233.7    7.237e-34
## 3    40     259.9    1.970e-34
## 4    50     277.8    1.036e-33
##
##
## Elapsed time : 0.503
```

MLE standard errors ("Standard error" in the output above) on α_i are $\sim 0.02 - -0.027$, whereas the QML errors are $\sim 0.046 - -0.051$ - a factor of ~ 2 times larger. Since QML's standard error is notably larger, this indicates that with the assumed model, an i.i.d Normal distribution for η_t in $\epsilon_t = \sigma_t \eta_t$ is not an ideal fit. In both cases it is statistically significant that the estimated parameters are larger than zero.

```r
sigma_hat = sigma(MSFT.arch5.fit)
p<-plot(abs(MSFT.ret), ylab='Conditional volatility of daily return', col='red', main='ARCH(5) vs ABS(r
lines(sigma_hat, col='blue')
```

(iii) In-sample estimates of the conditional volatility of the daily return, $\hat{\sigma}_t$, and comparision with its absolute return, $|X_t|$. The volatility spike locations are at the same locations as spikes in the absolute return, but the magnitude is clearly quite different (unsurprisingly, these are of course different quantities). The absolute value of the returns (red) also regularly drops to zero, whereas the conditional volatility (blue) never drops below ~ 0.15.

```r
# simple sample standard deviation
print(StdDev(MSFT.ret))
```

(iv) Estimate of the unconditional volatility (estimated standard deviation) of the return.

```
##              [,1]
## StdDev 0.02009
```

```
# unconditional variance: omega/(alpha1 + beta1)
print(paste("Unconditional volatility:", sqrt(uncvariance(MSFT.arch5.fit))))
```

```
## [1] "Unconditional volatility: 0.023637580538411"
```

The unconditional volatility, $\bar{\sigma} = \sqrt{\omega/(1-\sum_i \alpha_i)}$ is fairly close but somewhat larger than the sample standard deviation of the MSFT daily returns, indicating the ARCH(5) model is over predicting the volatility by only $\sim 20\%$ overall.

(b) Estimation of a GARCH(1,1) model.

```
MSFT.garch11.spec = ugarchspec(variance.model = list(garchOrder=c(1,1)), mean.model = list(armaOrder = c
MSFT.garch11.fit = ugarchfit(MSFT.garch11.spec, MSFT.ret)
print(coef(MSFT.garch11.fit))
```

(i) Estimated params from the fit

```
##         mu     omega    alpha1     beta1
## 4.019e-04 5.776e-06 7.022e-02 9.156e-01
```

```
# persistence: alpha1 + beta1
alpha_beta_sum <- coef(MSFT.garch11.fit)["alpha1"] + coef(MSFT.garch11.fit)["beta1"]
print(paste("Persistence: alpha + beta =", alpha_beta_sum))
```

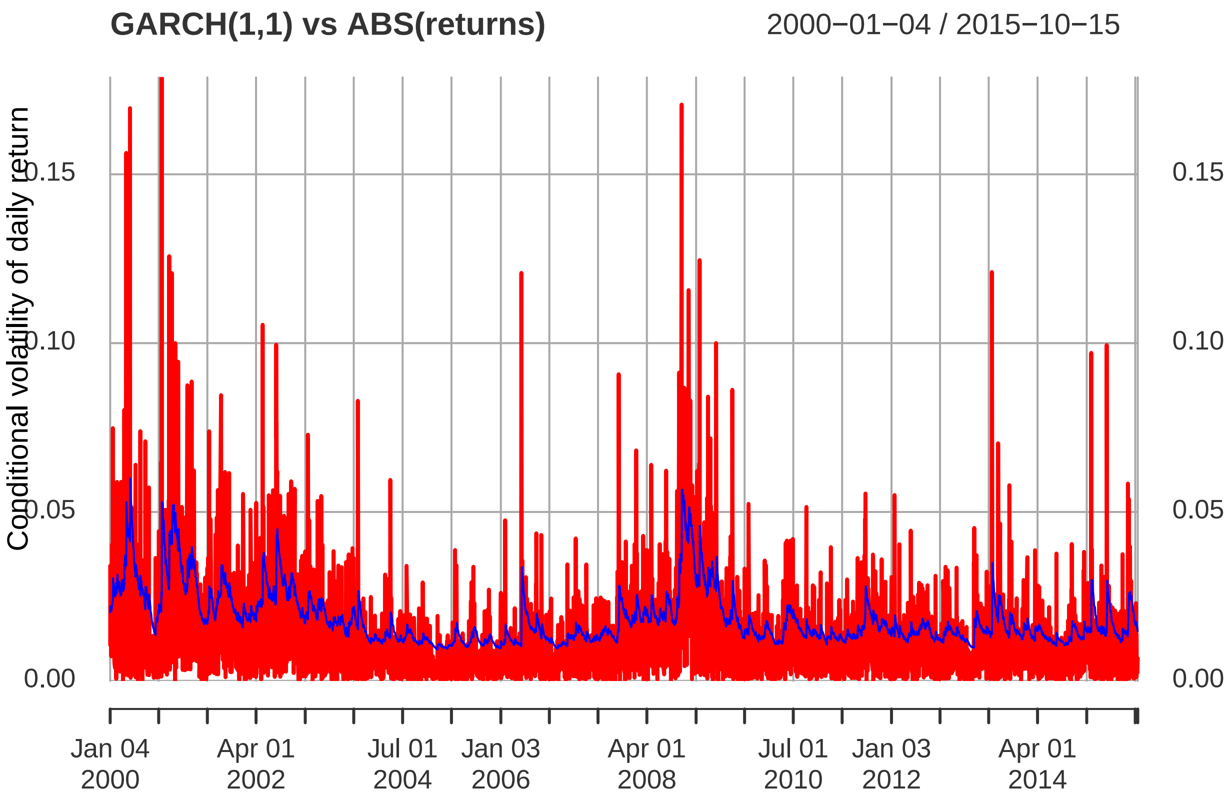
```
## [1] "Persistence: alpha + beta = 0.98583612881904"
```

```
# half-life:
print(paste("Half life", halflife(MSFT.garch11.fit), "days"))
```

```
## [1] "Half life 48.5902954975708 days"
```

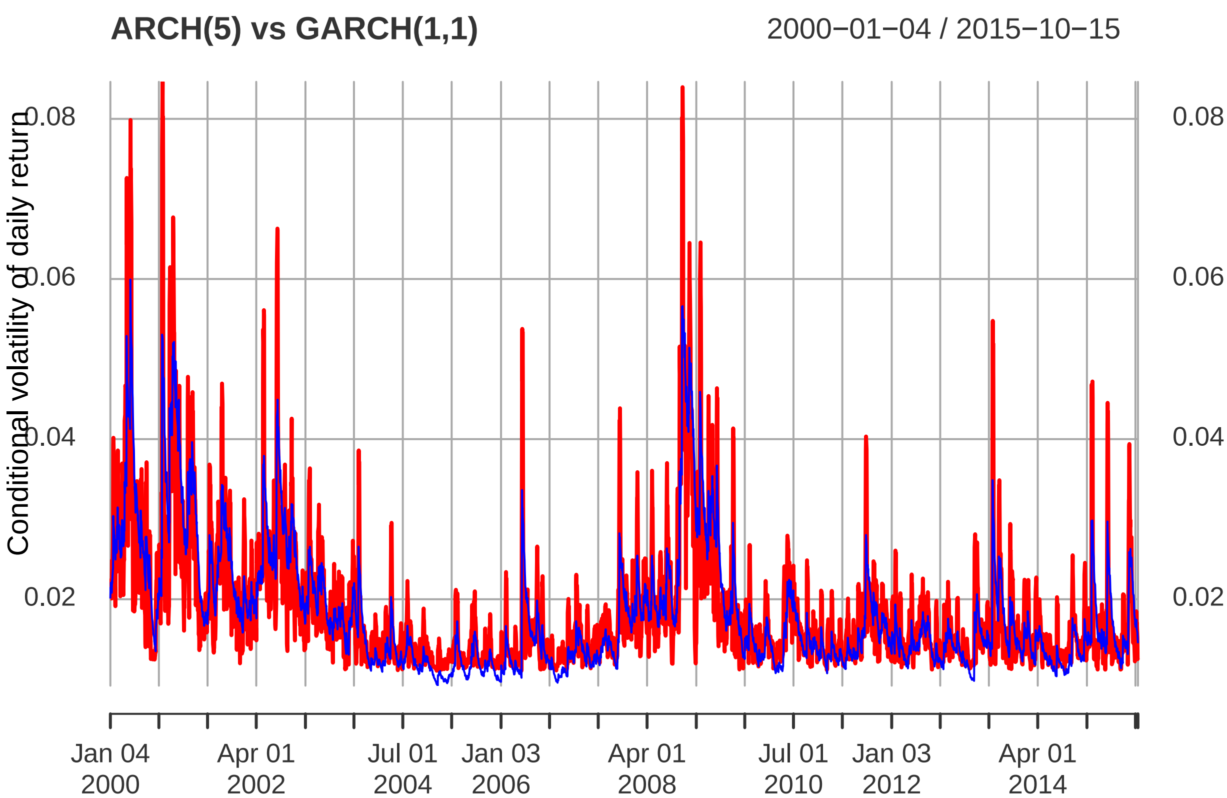
We see $\alpha + \beta = 0.9858 < 1$, which indicates that the fitted solution is second order stationary and therefore will also be both causal and unique. Such a solution does not explode to infinity, but given its closeness to one, there is high persistence in volatility. Such a model finds that volatility effects have a half-life of ~ 49 days. As expected for a stock market return, we see a small α and large β (close to 1).

```
sigma_hat_garch = sigma(MSFT.garch11.fit)
p<-plot(abs(MSFT.ret), ylab='Conditional volatility of daily return', col='red', main='GARCH(1,1) vs ABS
lines(sigma_hat_garch, col='blue')
```



(ii)

```
sigma_hat_garch = sigma(MSFT.garch11.fit)
p<-plot(sigma_hat, ylab='Conditional volatility of daily return', col='red', main='ARCH(5) vs GARCH(1,1
lines(sigma_hat_garch, col='blue')
```



Here we can see the differences between the GARCH(1,1) and ARCH(5) fitted models. Although the general larger scale trends in the conditional volatility follow a similar pattern, it's clear that the GARCH(1,1) model is much less erratic (less rapid variability in the fit volatility) on short timescales.

```r
# simple sample standard deviation
print(StdDev(MSFT.ret))
```

(iii) Estimate of the unconditional volatility (estimated standard deviation) of the return.

```
##            [,1]
## StdDev 0.02009
```

```r
# unconditional variance: omega/(alpha1 + beta1)
print(paste("Unconditional volatility:", sqrt(uncvariance(MSFT.garch11.fit))))
```

```
## [1] "Unconditional volatility: 0.0201947758894161"
```

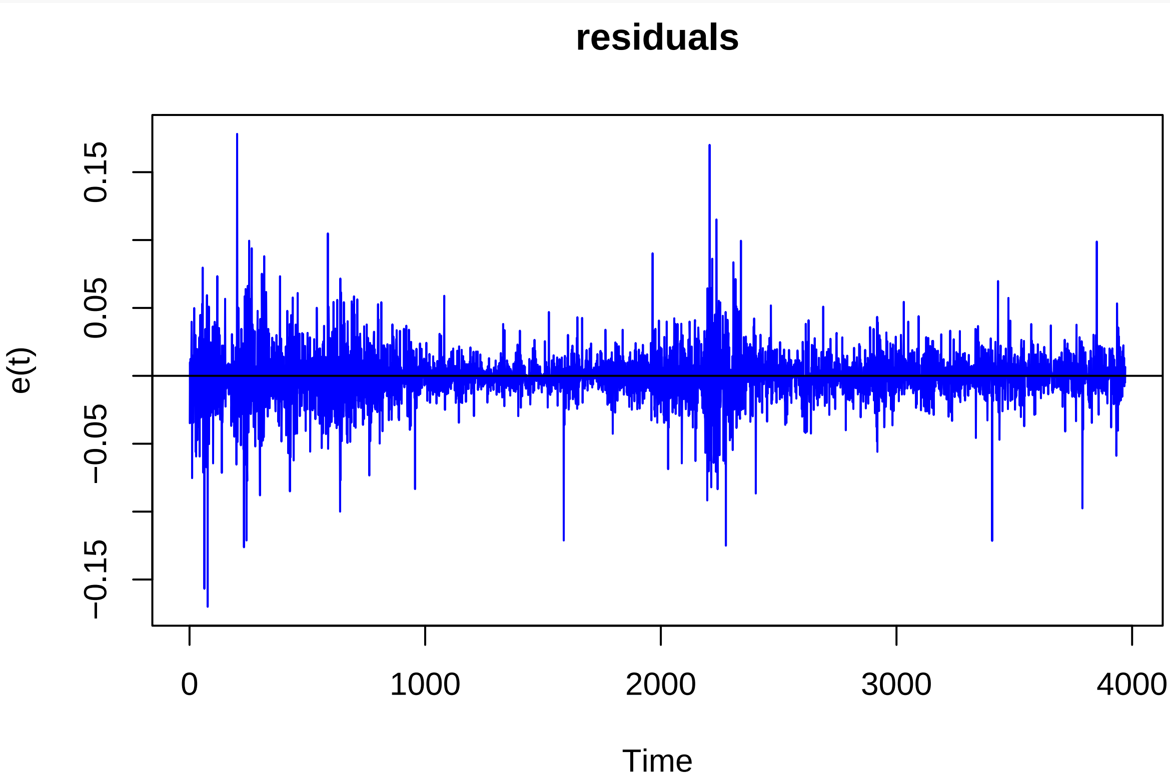
The unconditional volatility, $\bar{\sigma} = \sqrt{\omega/(1-\alpha-\beta)}$, is the same as for the sample standard deviation for the MSFT daily returns, indicating a good fit to the volatility seen in the data.

Q2 : Validating (G)ARCH models

(a) Fitted ARCH(5) Model

(i) Weighted Ljung-Box Test Is the standardized residuals of the fitted model $\hat{\eta}_t \equiv \hat{\epsilon}_t/\hat{\sigma}_t$, serially correlated?

```r
# residuals: e(t)
plot.ts(residuals(MSFT.arch5.fit), ylab="e(t)", col="blue", main="residuals")
abline(h=0)
```



```r
print("Test for no serial correlation in standardized residuals")
```

```
## [1] "Test for no serial correlation in standardized residuals"
```

```r
# standardized residuals
epsilon_t = residuals(MSFT.arch5.fit) # residuals: epsilon(t)
sigma_t = sigma(MSFT.arch5.fit) # sigma(t) = conditional volatility
eta_t = epsilon_t/sigma_t
Box.test(eta_t, type="Ljung-Box", lag = 1)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 1.3, df = 1, p-value = 0.3
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.02166  1.00011
```

```r
Box.test(eta_t, type="Ljung-Box", lag = 6)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 5, df = 6, p-value = 0.5
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.02166  1.00011
```

```r
Box.test(eta_t, type="Ljung-Box", lag = 8)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 7.2, df = 8, p-value = 0.5
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.02166  1.00011
```

```r
Box.test(eta_t, type="Ljung-Box", lag = 10)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 11, df = 10, p-value = 0.4
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.02166  1.00011
```

Given the large p-values, the standardized residuals are consistent with the null (no serial correlation) left in the residuals of the fitted model.

```r
print("Test for no serial correlation in squared standardized residuals")
```

(ii) Are the squared standardized residuals serially correlated?

```
## [1] "Test for no serial correlation in squared standardized residuals"
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 1)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 0.87, df = 1, p-value = 0.4
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 6)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 7.3, df = 6, p-value = 0.3
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 8)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 8.1, df = 8, p-value = 0.4
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 10)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 8.1, df = 10, p-value = 0.6
```

The large p-values suggest no evidence of a serial correlation in the squared standardized residuals.

```r
ArchTest(eta_t)
```

(iii) Is there any ARCH effect left in the residuals after it has been fitted to an ARCH(5) model?

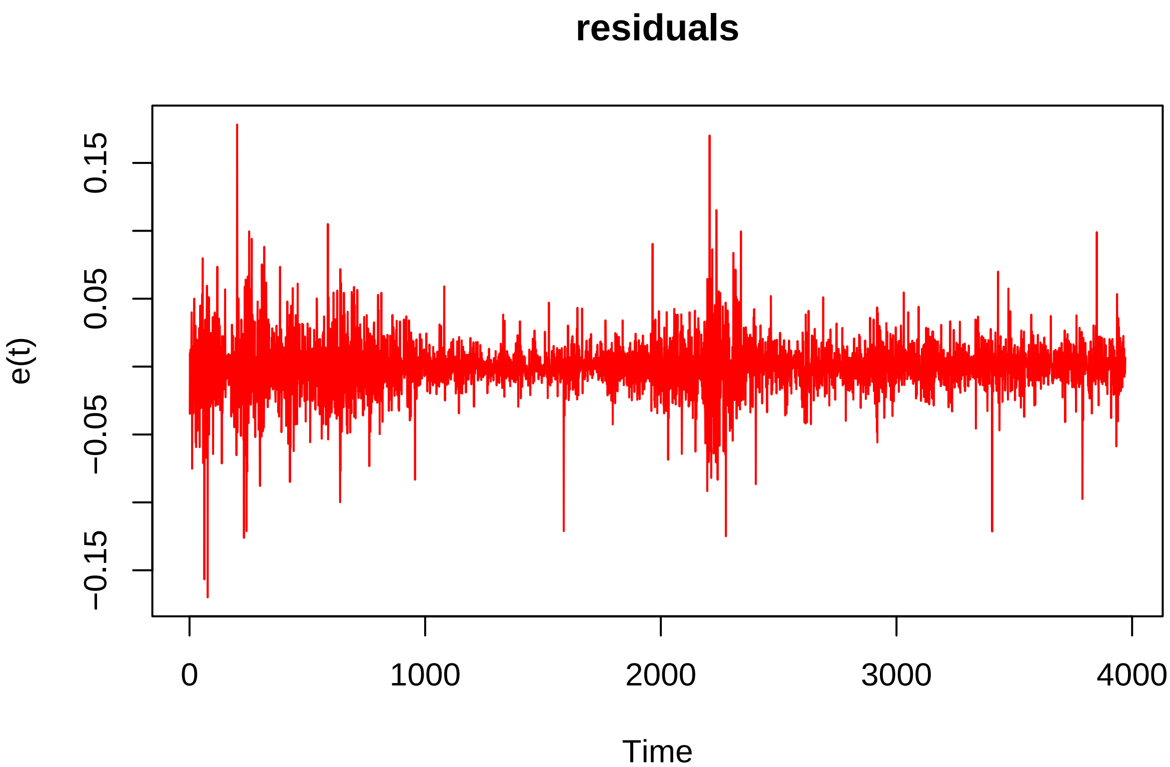
```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  eta_t
## Chi-squared = 12, df = 12, p-value = 0.5
```

Based on the ARCH LM test there is no evidence of an ARCH effect remaining in the model residuals.

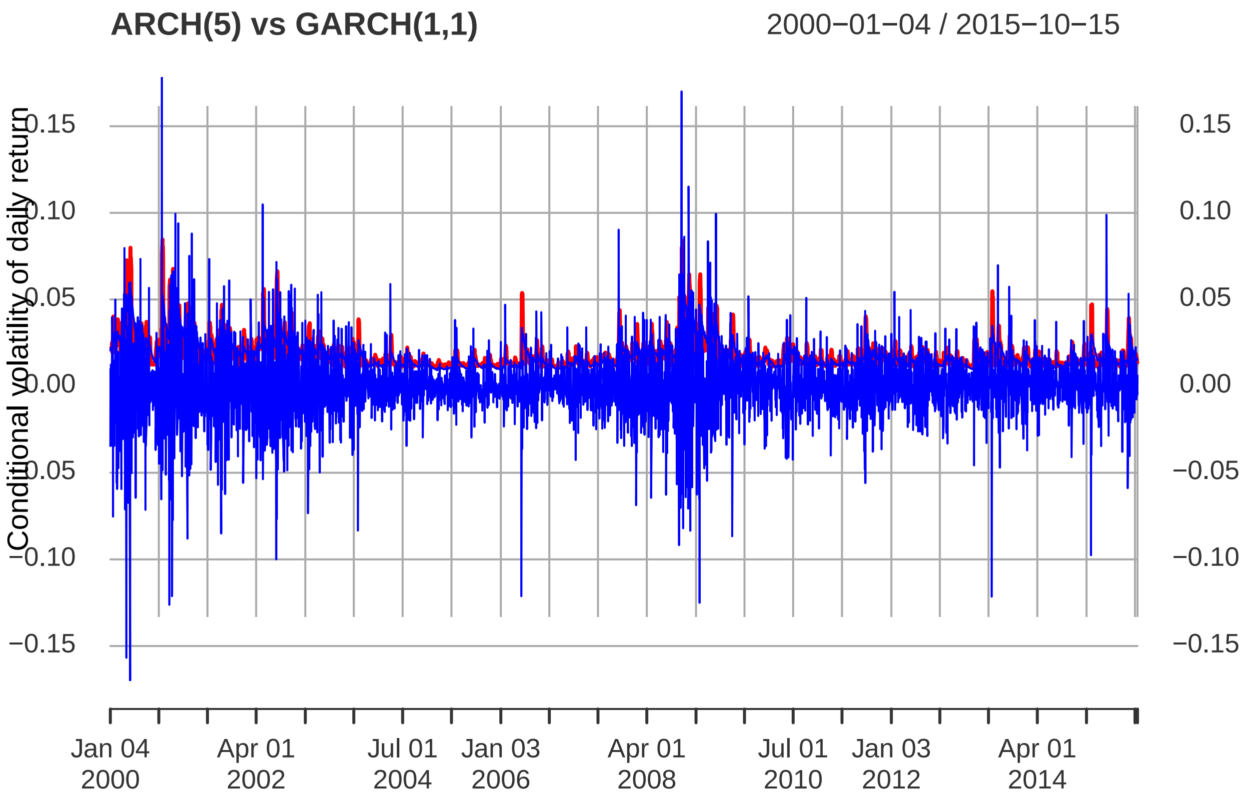
(b) Fitted GARCH(1,1) Model

(i) Weighted Ljung-Box Test Is the standardized residuals of the fitted model $\hat{\eta}_t \equiv \hat{\epsilon}_t/\hat{\sigma}_t$, serially correlated?

```
# residuals: e(t)
p<-plot.ts(residuals(MSFT.garch11.fit), ylab="e(t)", col="red", main="residuals")
```




```
lines(residuals(MSFT.arch5.fit), col='blue')
abline(h=0)
```




```r
print("Test for no serial correlation in standardized residuals")
```

```
## [1] "Test for no serial correlation in standardized residuals"
```

```r
# standardized residuals
epsilon_t = residuals(MSFT.garch11.fit) # residuals: epsilon(t)
sigma_t = sigma(MSFT.garch11.fit) # sigma(t) = conditional volatility
eta_t = epsilon_t/sigma_t
Box.test(eta_t, type="Ljung-Box", lag = 1)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 2.2, df = 1, p-value = 0.1
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.01551  1.00086
```

```r
Box.test(eta_t, type="Ljung-Box", lag = 6)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 6.6, df = 6, p-value = 0.4
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.01551  1.00086
```

```r
Box.test(eta_t, type="Ljung-Box", lag = 8)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 9.5, df = 8, p-value = 0.3
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.01551  1.00086
```

```r
Box.test(eta_t, type="Ljung-Box", lag = 10)
```

```
##
##  Box-Ljung test
##
## data:  eta_t
## X-squared = 14, df = 10, p-value = 0.2
```

```r
print(c(mean(eta_t),StdDev(eta_t)))
```

```
## [1] -0.01551  1.00086
```

The above p-values for various lags suggest no evidence for serial correlation in the standardized residuals.

```r
print("Test for no serial correlation in squared standardized residuals")
```

(ii) Are the squared standardized residuals serially correlated?

```
## [1] "Test for no serial correlation in squared standardized residuals"
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 1)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 0.15, df = 1, p-value = 0.7
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 6)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 0.51, df = 6, p-value = 1
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 8)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 0.59, df = 8, p-value = 1
```

```r
Box.test(eta_t^2, type="Ljung-Box", lag = 10)
```

```
##
##  Box-Ljung test
##
## data:  eta_t^2
## X-squared = 1, df = 10, p-value = 1
```

The above p-values for various lags suggest no evidence for serial correlation in the squared standardized residuals.

```r
ArchTest(eta_t)
```

(iii) Is there any ARCH effect left in the residuals after it has been fitted to an GARCH(1,1) model?

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  eta_t
## Chi-squared = 1.2, df = 12, p-value = 1
```

Based on the ARCH LM test, there is no ARCH effect in the standardized residuals from the GARCH(1,1) model.

(c) Summary remarks

```r
# extract information criteria for all models
print("ARCH(5) model:")
```

```
## [1] "ARCH(5) model:"
```

```r
infocriteria(MSFT.arch5.fit)
```

```
##
## Akaike       -5.246
## Bayes        -5.235
## Shibata      -5.246
## Hannan-Quinn -5.242
```

```r
print("GARCH(1,1) model:")
```

```
## [1] "GARCH(1,1) model:"
```

```r
infocriteria(MSFT.garch11.fit)
```

```
##
## Akaike       -5.296
## Bayes        -5.290
## Shibata      -5.296
## Hannan-Quinn -5.294
```

The analysis in the previous parts indicates a GARCH(1,1) model should be given preference. As well, if we consider the information criterion given above, GARCH(1,1) is clearly lower-valued and is therefore a better fit to the data. Part of this improvement in the information criteria, lending additional preference to GARCH(1,1) is that it has only four fittable parameters, instead of 7.

Q3 : Model prediction of (G)ARCH Models

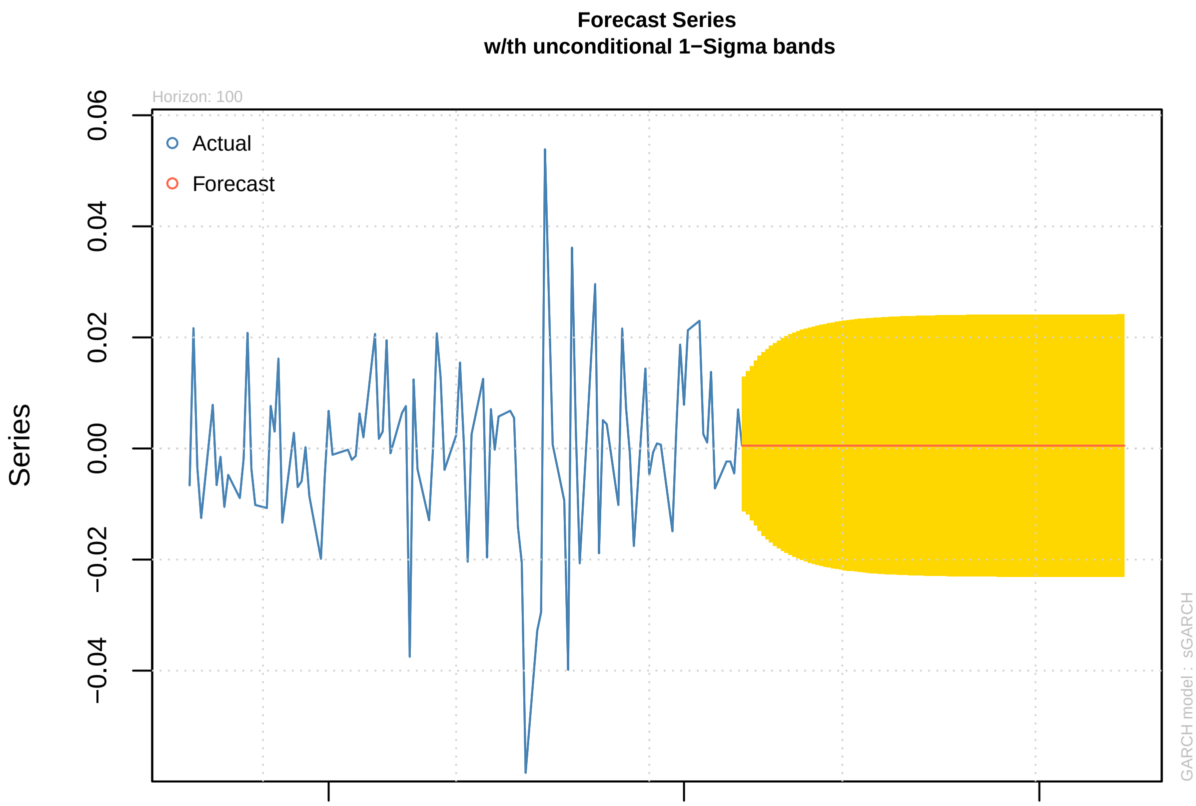
(a) Fitted ARCH(5) Model

Compute h-step ahead volatility forecasts of the daily returns on Microsoft for h=1,. . . ,100 days.

Plot these along with the estimate of unconditional volatility of the daily return on Microsoft. From the plot, do the forecasts appear to converge fast to the unconditional volatility of the returns? Explain.

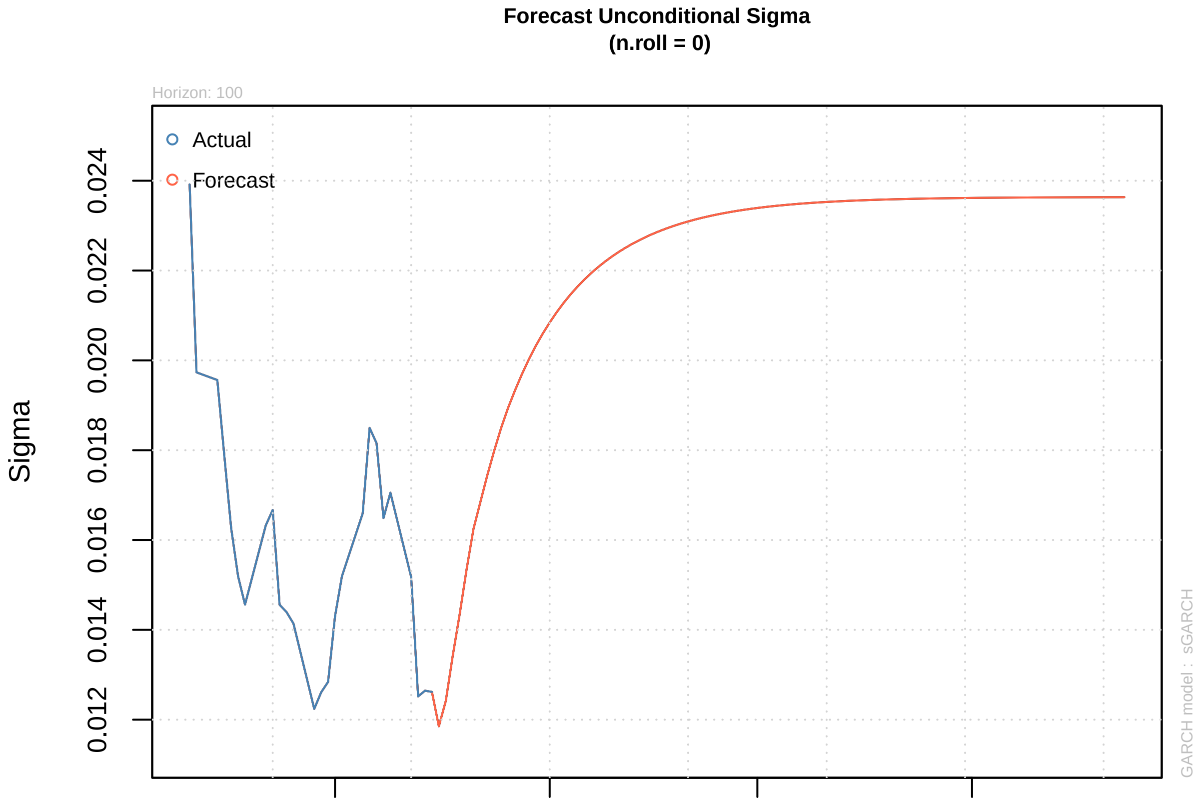
```
# Set larger bottom and top margins (the numbers are lines of text)
par(mfrow=c(1,1), mar=c(1, 4.1, 4, 2.1))

# Your plotting code
MSFT.arch5.fcst = ugarchforecast(MSFT.arch5.fit, n.ahead=100)
par(mfrow=c(1,1))
plot(MSFT.arch5.fcst, which=1)
```




```
# Set larger bottom and top margins (the numbers are lines of text)
par(mfrow=c(1,1), mar=c(1, 4.1, 4, 2.1))

# Your plotting code
MSFT.arch5.fcst = ugarchforecast(MSFT.arch5.fit, n.ahead=100)
par(mfrow=c(1,1))
plot(MSFT.arch5.fcst, which=3)
```



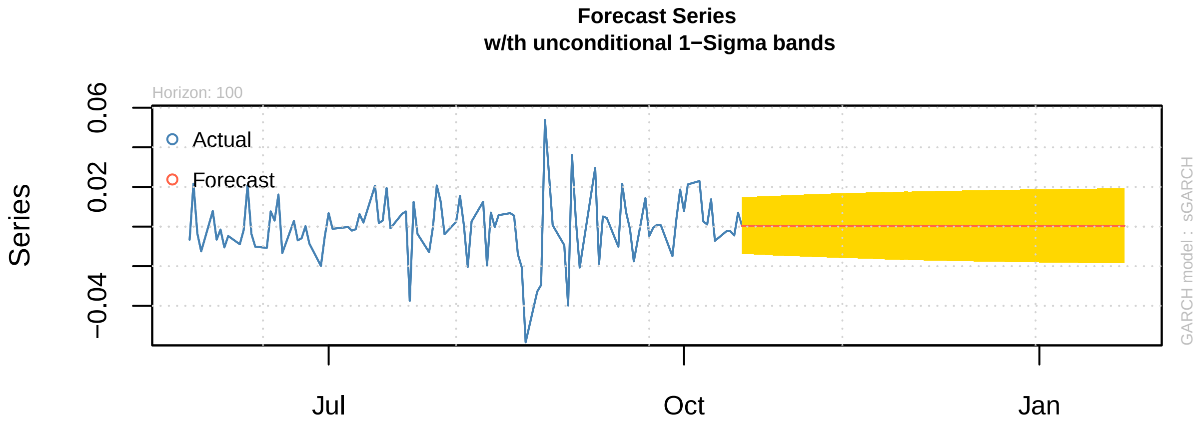
The forecast converges within 2-3 months to the conditional volatility of the returns for the ARCH(5) model (0.0236).

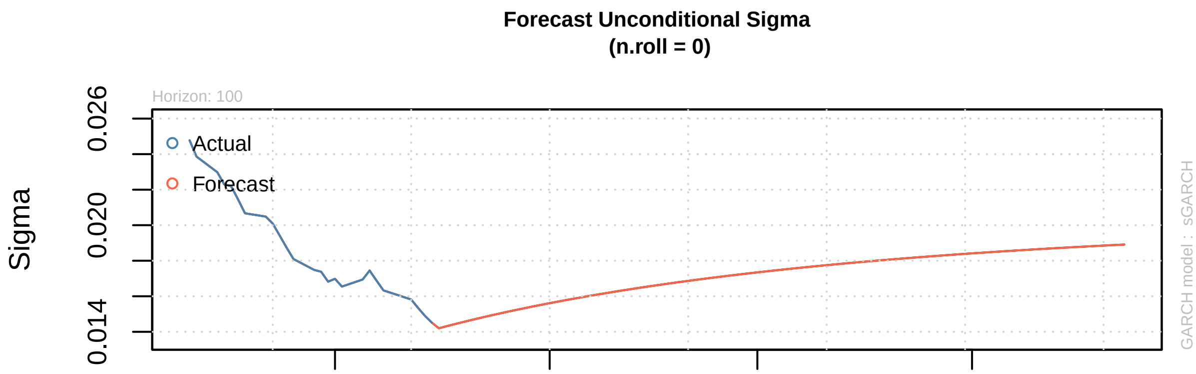
(b) Predictions of a fitted GARCH(1,1) Model

```
# Set larger bottom and top margins (the numbers are lines of text)
par(mfrow=c(1,1), mar=c(1, 4.1, 4, 2.1))

# Your plotting code
MSFT.garch11.fcst = ugarchforecast(MSFT.garch11.fit, n.ahead=100)

par(mfrow=c(2,1))
plot(MSFT.garch11.fcst, which=1)
plot(MSFT.garch11.fcst, which=3)
```






```
# plot.window(xlim=c(as.Date("2014-01-01"), as.Date("2016-03-28")))
```

The forecast in this case only slowly converges (over many months) to the conditional volatility of the returns, which in this case is equal to the unconditional volatility of the returns (0.020). $\alpha + \beta$ being close to one explains why the volatility in the GARCH(1,1) model takes so long to converge (high persistence).

Question 4. Rolling Estimation, Rolling Forecasts, and VaR Measures from Fitted GARCH(1,1) Model

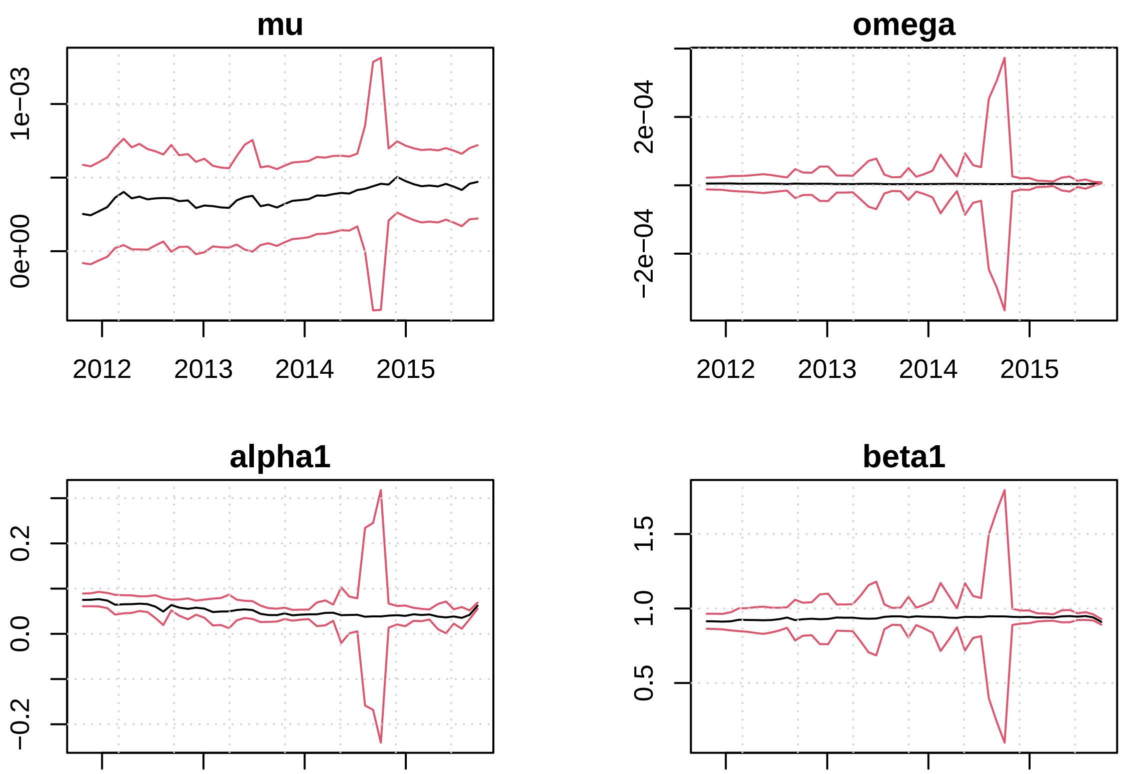
(a) Write formulas for 1-day conditional Value at Risk (VaR) and h-day conditional VaR for the daily return on MSFT.

(b) Compute a thousand 1-period ahead rolling forecasts of the normal GARCH(1,1) model for Microsoft, refitting the GARCH(1,1) model every 20 observations (this is a monthly rolling window).

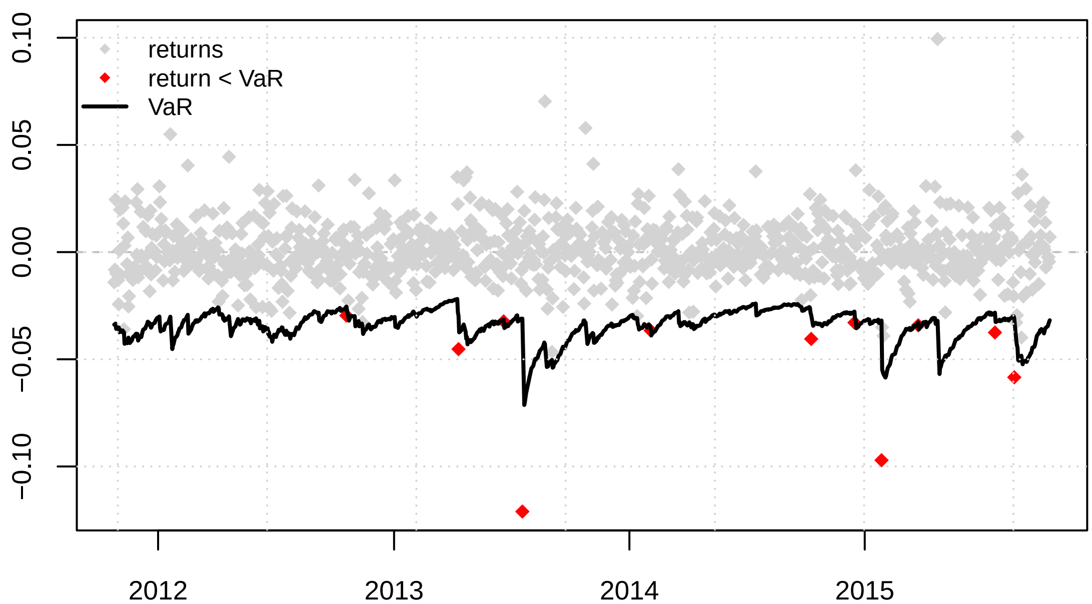
In ugarchroll(), n:ahead=1 means compute 1-step ahead fore- casts, forecast:length=1000 means set aside 1000 observations for the forecast, refit:every = 20 means you will refit the model every 20 days in the interest of the calculation speed, and refit:window = moving is the rolling forecast. Make sure that you calculate 1% VaR when you perform the rolling estimation.

```
MSFT.garch11.roll = ugarchroll(MSFT.garch11.spec, MSFT.ret, n.ahead=1, forecast.length = 1000,
                                refit.every=20, refit.window="moving")
# Coef plot`
par(mfrow=c(1,1), mar=c(1, 4.1, 4, 2.1))
plot(MSFT.garch11.roll, which=5)
```

sGARCH fit coefficients (across 50 refits) with robust s.e. bands




```r
# VaR plot
plot(MSFT.garch11.roll, which=4)
```




```r
report(MSFT.garch11.roll, type="VaR")
```

```
## VaR Backtest Report
## ===========================================
## Model:                 sGARCH-norm
```

```
## Backtest Length: 1000
## Data:
##
## ==========================================
## alpha:              1%
## Expected Exceed: 10
## Actual VaR Exceed:   11
## Actual %:           1.1%
##
## Unconditional Coverage (Kupiec)
## Null-Hypothesis: Correct Exceedances
## LR.uc Statistic: 0.098
## LR.uc Critical:      3.841
## LR.uc p-value:       0.754
## Reject Null:     NO
##
## Conditional Coverage (Christoffersen)
## Null-Hypothesis: Correct Exceedances and
##                  Independence of Failures
## LR.cc Statistic: 0.343
## LR.cc Critical:      5.991
## LR.cc p-value:       0.842
## Reject Null:     NO
```

The above plot and results suggest that the risk captured in the rolling 1% estimates are underpredicted by our fit GARCH(1,1) model.

(c) What is the Mean Squared Error and Mean Absolute Errors of the forecast?

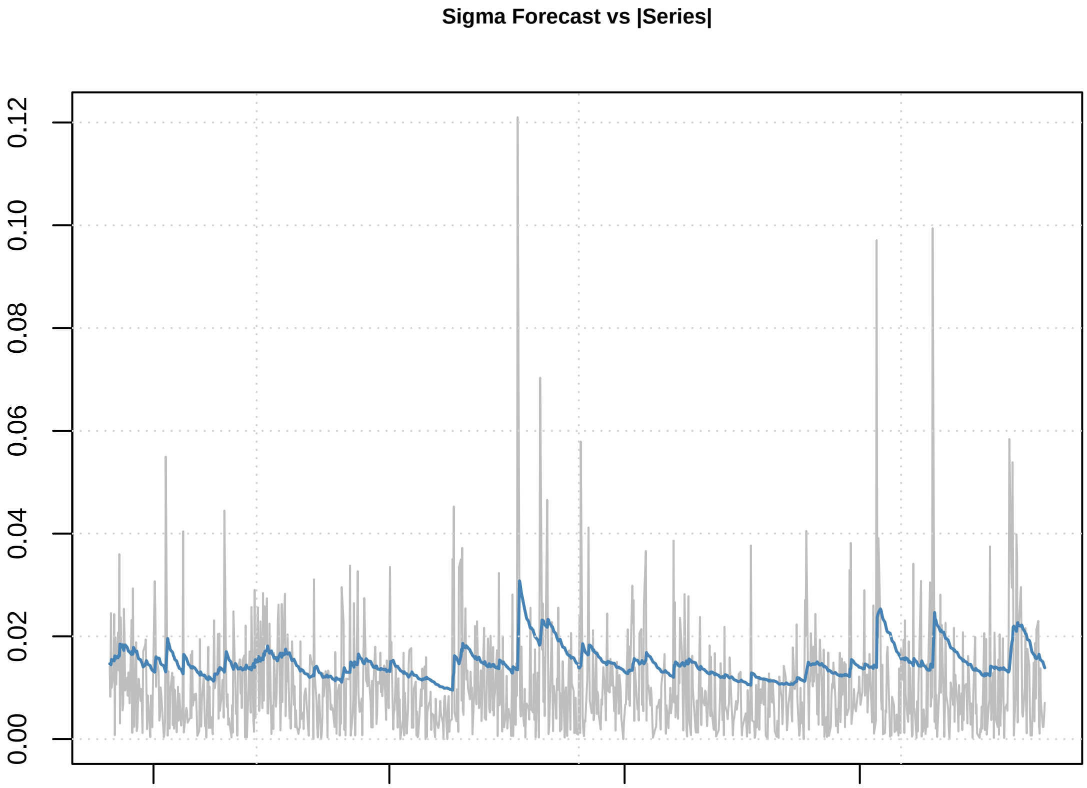
```r
report(MSFT.garch11.roll, type="fpm")
```

```
##
## GARCH Roll Mean Forecast Performance Measures
## ---------------------------------------------
## Model        : sGARCH
## No.Refits    : 50
## No.Forecasts: 1000
##
##         Stats
## MSE 0.0002128
## MAE 0.0101900
## DAC 0.4920000
```

The values are given in the above table as MSE and MAE, respectively. Since they are very small it indicates a good fit.

(d) Plot the forecast of conditional volatility and discuss the result.

```r
par(mfrow=c(1,1), mar=c(1, 4.1, 4, 2.1))
plot(MSFT.garch11.roll, which=2)
```



Sigma Forecast vs |Series|
0.00
0.02
0.04
0.06
0.08
0.10
0.12