MATH 8090: Univariate Volatility Modeling

Whitney Huang, Clemson University

10/19/2021

Contents

Load the Apple stock data

```
library(quantmod)
getSymbols("AAPL", src = "yahoo")
```

```
## [1] "AAPL"
```

```
dim(AAPL)
```

```
## [1] 3726    6
```

```
head(AAPL)
```

```
##            AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
## 2007-01-03  3.081786  3.092143 2.925000   2.992857  1238319600      2.569716
## 2007-01-04  3.001786  3.069643 2.993571   3.059286   847260400      2.626753
## 2007-01-05  3.063214  3.078571 3.014286   3.037500   834741600      2.608048
## 2007-01-08  3.070000  3.090357 3.045714   3.052500   797106800      2.620926
## 2007-01-09  3.087500  3.320714 3.041071   3.306071  3349298400      2.838647
## 2007-01-10  3.383929  3.492857 3.337500   3.464286  2952880000      2.974493
```

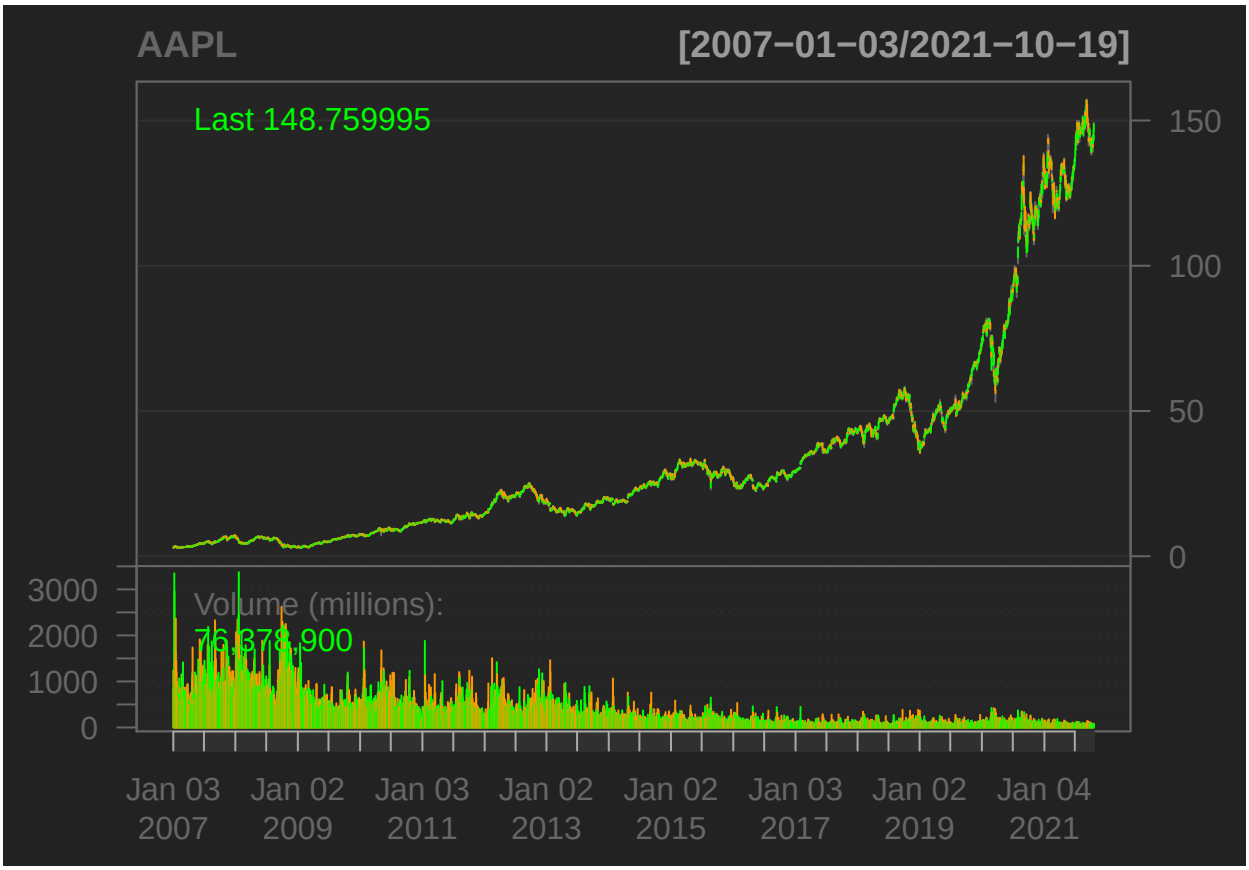
```
tail(AAPL)
```

```
##            AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
## 2021-10-12    143.23    143.25   141.04     141.51    73035900        141.51
## 2021-10-13    141.24    141.40   139.20     140.91    78762700        140.91
## 2021-10-14    142.11    143.88   141.51     143.76    69907100        143.76
## 2021-10-15    143.77    144.90   143.51     144.84    67885200        144.84
## 2021-10-18    143.45    146.84   143.16     146.55    85589200        146.55
## 2021-10-19    147.01    149.17   146.55     148.76    76378900        148.76
```

```
summary(AAPL)
```

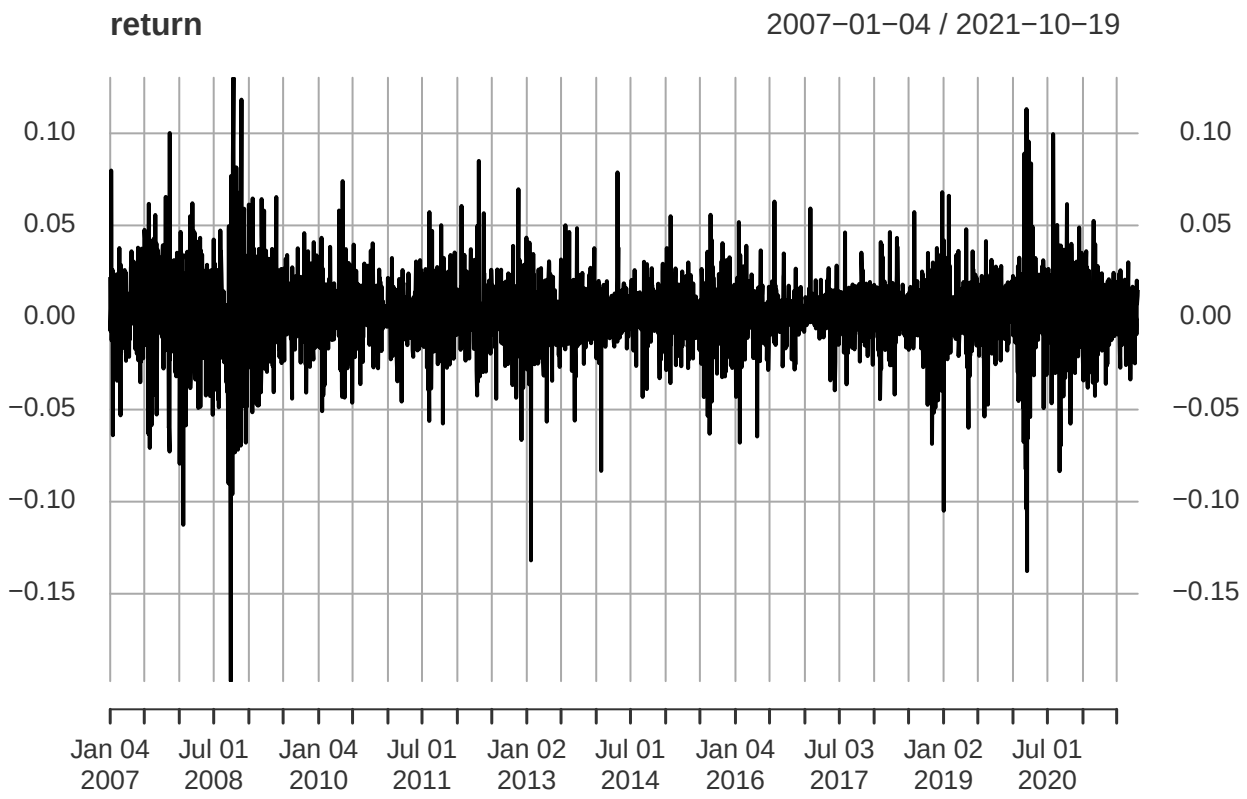
```
##      Index              AAPL.Open         AAPL.High         AAPL.Low      
##  Min.   :2007-01-03   Min.   :  2.835   Min.   :  2.929   Min.   :  2.793  
##  1st Qu.:2010-09-14   1st Qu.:  9.744   1st Qu.:  9.862   1st Qu.:  9.642  
##  Median :2014-05-28   Median : 23.392   Median : 23.564   Median : 23.145  
##  Mean   :2014-05-27   Mean   : 33.419   Mean   : 33.767   Mean   : 33.066  
##  3rd Qu.:2018-02-06   3rd Qu.: 42.291   3rd Qu.: 42.688   3rd Qu.: 41.763  
##  Max.   :2021-10-19   Max.   :156.980   Max.   :157.260   Max.   :154.390  
##    AAPL.Close       AAPL.Volume        AAPL.Adjusted    
##  Min.   :  2.793   Min.   :4.545e+07   Min.   :  2.398  
##  1st Qu.:  9.782   1st Qu.:1.268e+08   1st Qu.:  8.399  
##  Median : 23.376   Median :2.656e+08   Median : 20.709  
##  Mean   : 33.430   Mean   :3.993e+08   Mean   : 31.930  
##  3rd Qu.: 42.267   3rd Qu.:5.466e+08   3rd Qu.: 40.553  
##  Max.   :156.690   Max.   :3.373e+09   Max.   :156.690
```

```
chartSeries(AAPL)
```

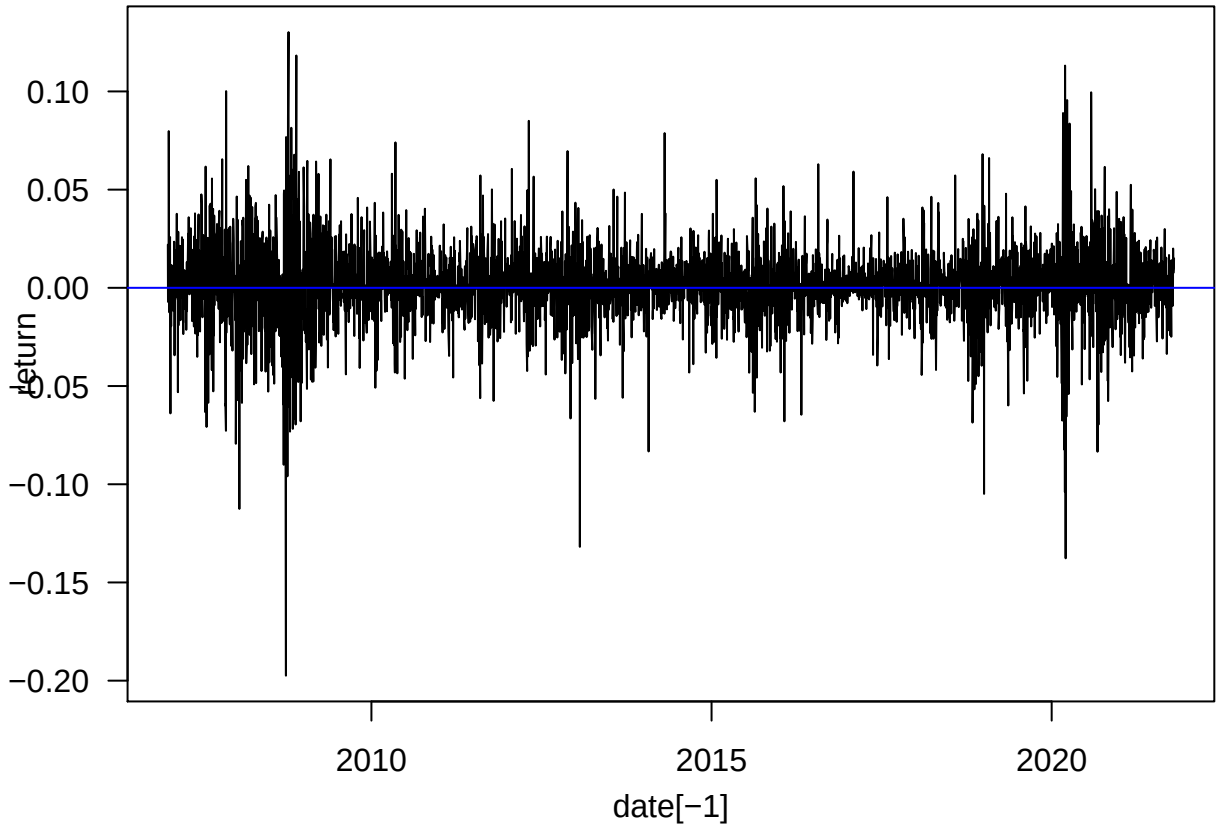


Plot the stock time series

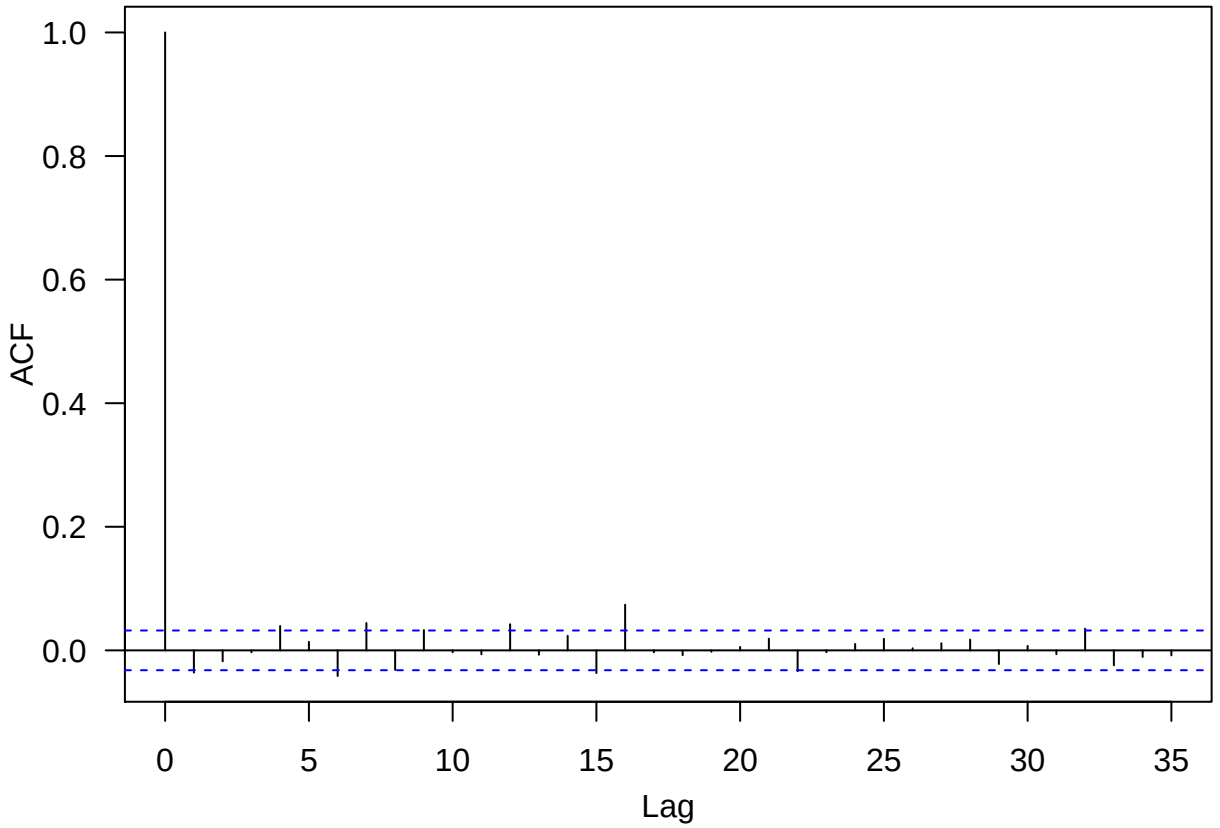
```
closing <- AAPL$AAPL.Close
return <- diff(log(closing))[!is.na(diff(log(closing))$AAPL.Close)]
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6))
plot(return)
```



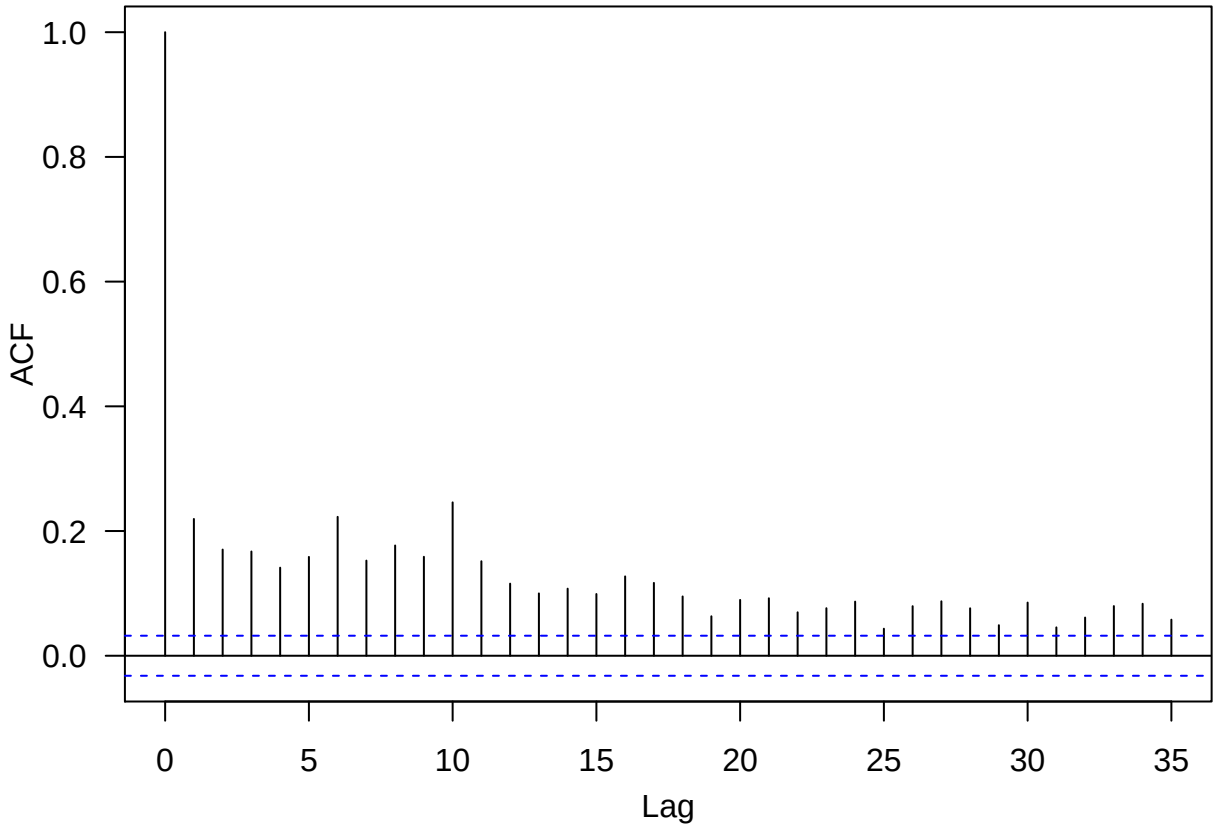

```
library(timetk)
date <- tk_index(AAPL)
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6))
plot(date[-1], return, type = "l")
abline(h = 0, col = "blue", lwd = 1)
```


0.10
0.05
0.00
−0.05
−0.10
−0.15
−0.20
return
2010
2015
2020
date[−1]

```
acf(return)
```


1.0
0.8
0.6
0.4
0.2
0.0
ACF
0
5
10
15
20
25
30
35
Lag

```
acf(return^2)
```



ARCH Engle (1982)

An ARCH(m) model:

$$a_t = \sigma_t \epsilon_t, \quad \sigma_t^2 = \alpha_0 + \alpha_1 a_{t-1}^2 + \cdots + \alpha_m a_{t-m}^2,$$

where $\{\epsilon_t\}$ is a sequence of i.i.d. r.v. with

- $\mathbb{E}(\epsilon_t) = 0$
- $\text{Var}(\epsilon_t) = 1$
- $\alpha_i \geq 0$ for $1 \leq i \leq m$
- *Distribution*: standard normal, standardize Student-t, generalized error distribution, or their skewed counterparts

Simulation

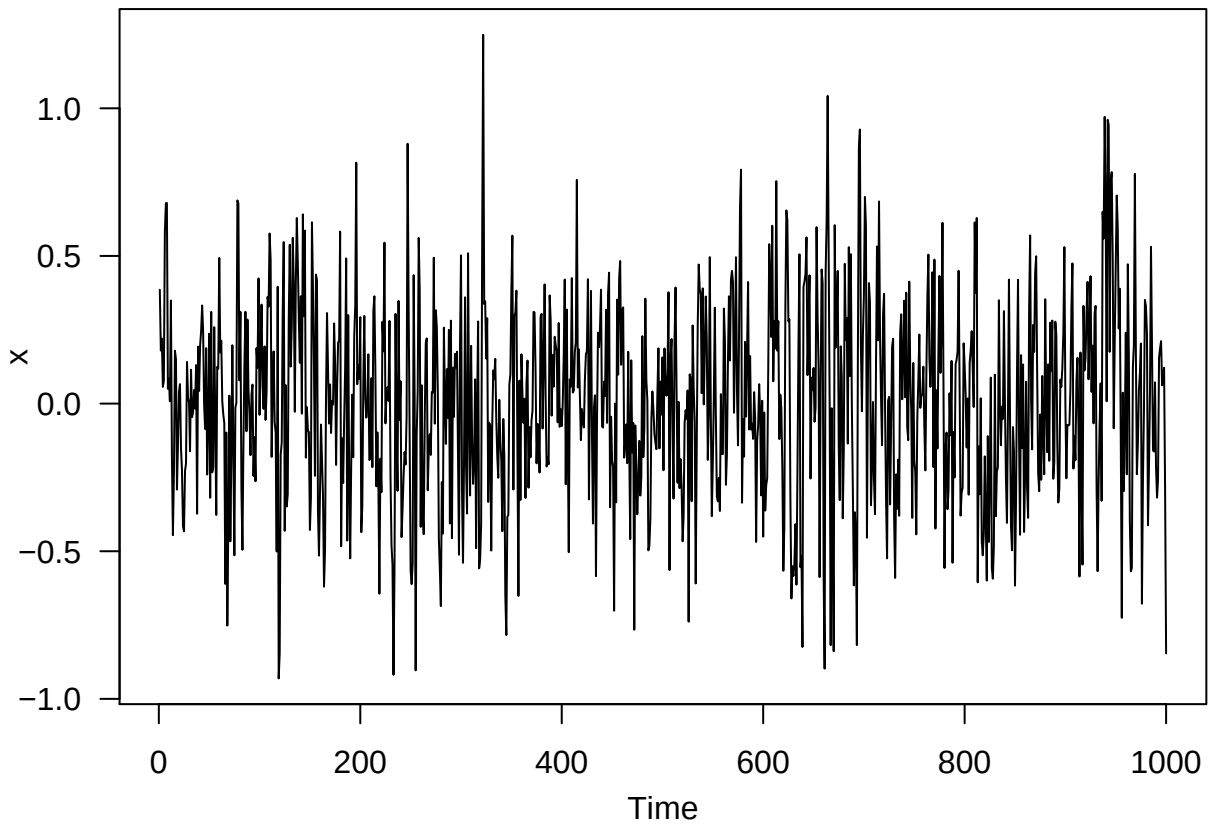
```
library(fGarch)
```

```
## Loading required package: timeDate
```

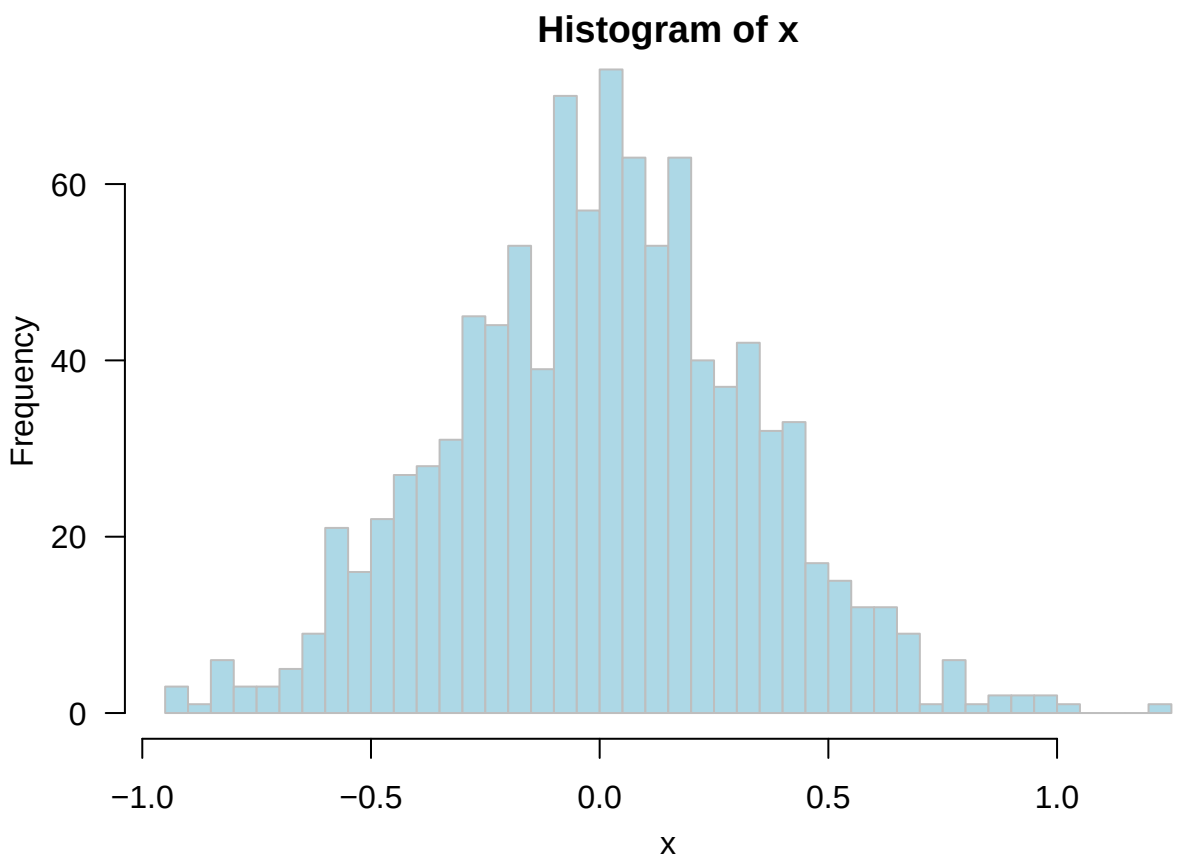
```
## Loading required package: timeSeries
```

```
##
## Attaching package: 'timeSeries'

## The following object is masked from 'package:zoo':
##
##     time<-

## Loading required package: fBasics

##
## Attaching package: 'fBasics'

## The following object is masked from 'package:TTR':
##
##     volatility
```

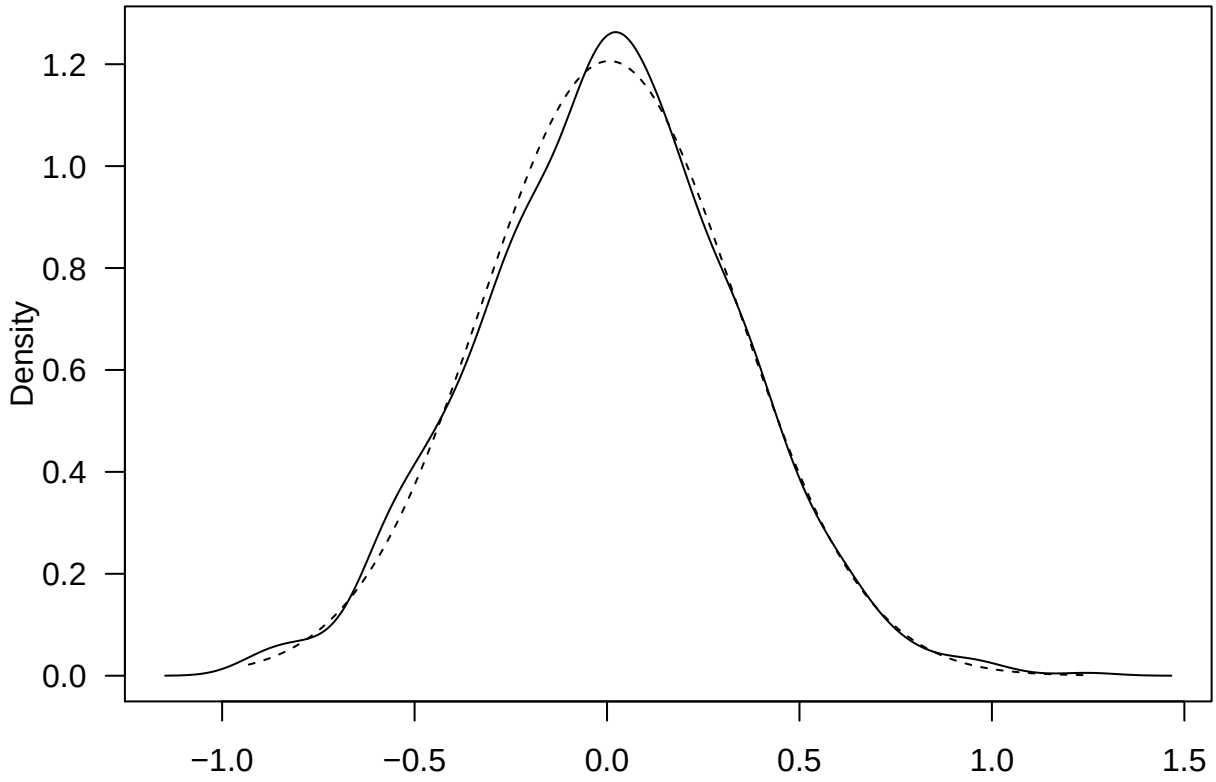
```
mod_spec <- garchSpec(model = list(ar = c(.35), omega = 0.01))
x <- garchSim(spec = mod_spec, n = 1000)
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6))
ts.plot(x)
```



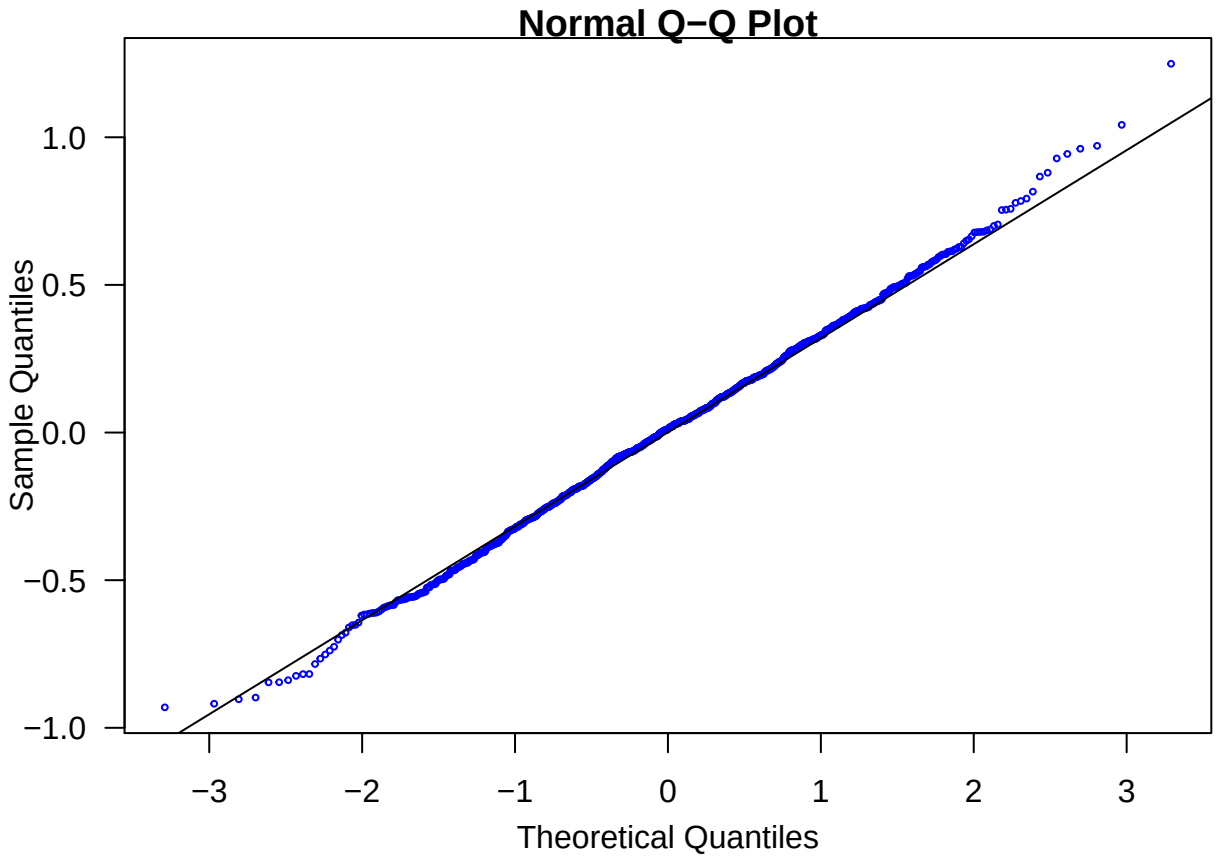

```
hist(x, nclass = 50, col = "lightblue", border = "gray")
```




```
den <- density(x)
rg <- range(x)
xg <- seq(rg[1], rg[2], .001)
yg <- dnorm(xg, mean(x), stdev(x))
plot(den$x, den$y, xlab = "", ylab = "Density", type = "l")
lines(xg, yg, lty = 2)
```




```
qqnorm(x, col = "blue", cex = 0.4); qqline(x)
```



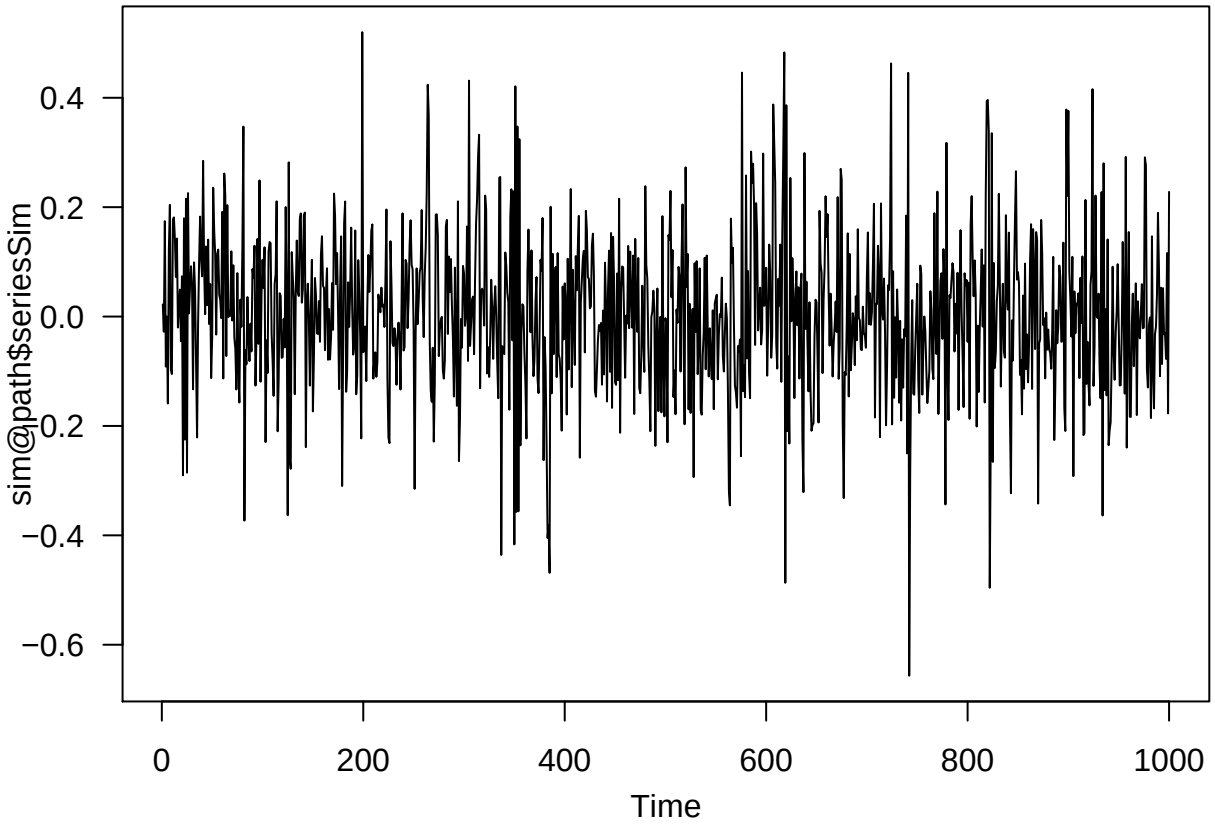
```
library(rugarch)
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'rugarch'
```

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

```
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6))
mod_spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 0)),mean.model = list(a
sim <- ugarchpath(mod_spec, n.sim = 1000)
ts.plot(sim@path$seriesSim)
```

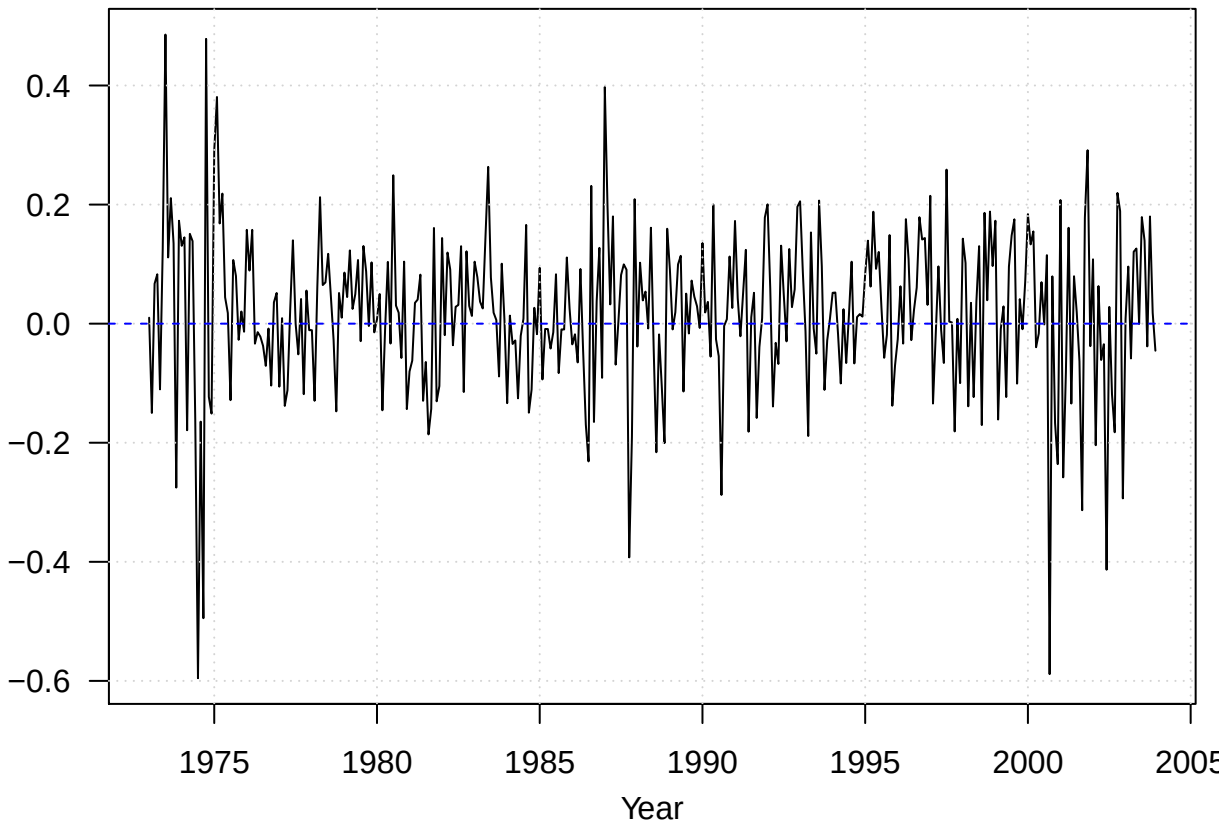


Load and plot the monthly log returns of Intel stock

```
url <- "https://www.chicagobooth.edu/-/media/faculty/ruey-s-tsay/teaching/fts2//m-intc7303.txt"
dat1 <- read.table(url)
names(dat1) <- c("Date", "Return"); head(dat1)
```

```
##       Date   Return
## 1 19730131  0.01005
## 2 19730228 -0.13930
## 3 19730330  0.06936
## 4 19730430  0.08649
## 5 19730531 -0.10448
## 6 19730629  0.13333
```

```
intc <- log(dat1$Return + 1)
return <- ts(intc, frequency = 12, start = c(1973, 1))
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6))
plot(return, type = "l", xlab = "Year", ylab = "")
grid()
abline(h = 0, lty = 2, col = "blue")
```



Testing ARCH effect

```
t.test(intc)
```

```
##
##  One Sample t-test
##
## data:  intc
## t = 2.5944, df = 371, p-value = 0.00985
## alternative hypothesis: true mean is not equal to 0
```

```
## 95 percent confidence interval:
##  0.004354971 0.031624693
## sample estimates:
##  mean of x
## 0.01798983
```

```
y <- intc - mean(intc)
Box.test(y^2, lag = 12, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  y^2
## X-squared = 68.67, df = 12, p-value = 5.676e-10
```

```
source("archTest.R")
archTest(y, 12)
```

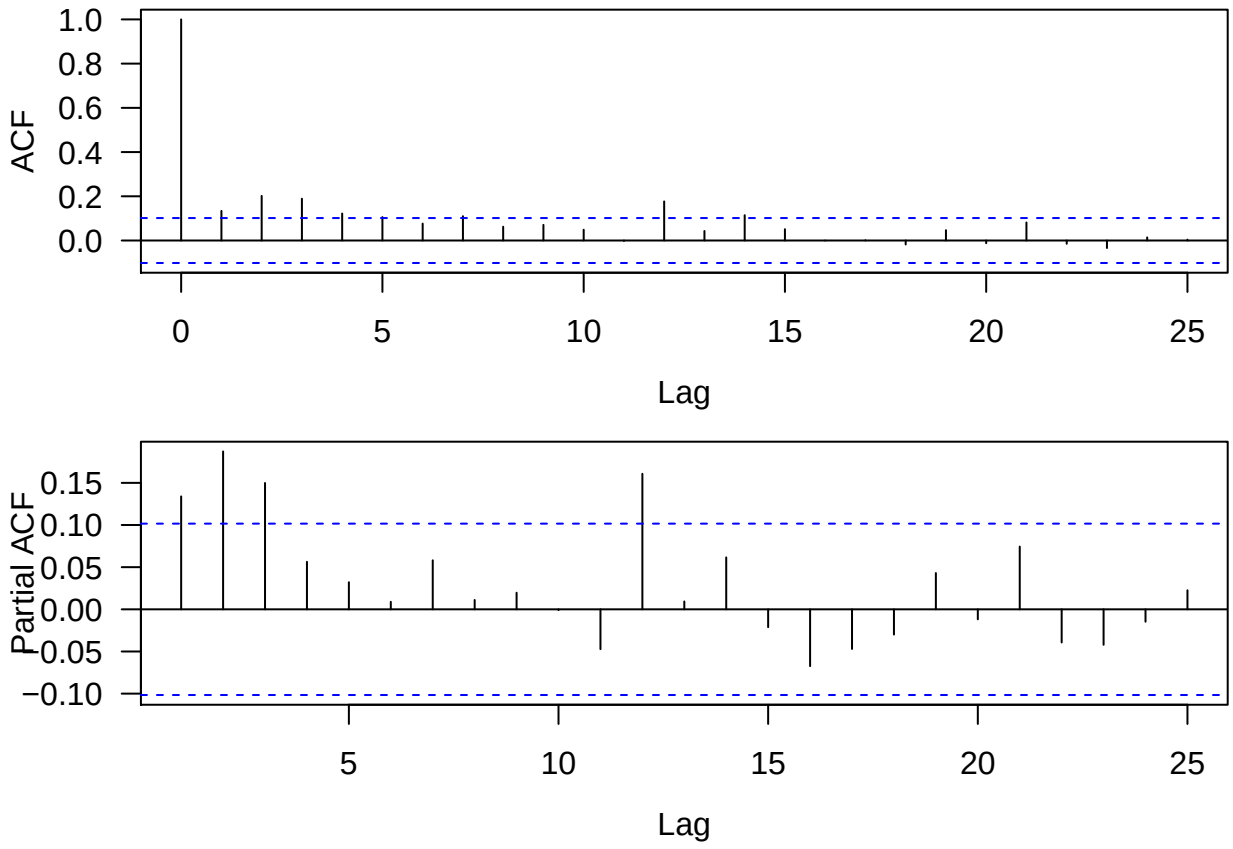
```
##
## Call:
## lm(formula = atsq ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.07368 -0.01295 -0.00729  0.00450  0.35621
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.007029   0.002752   2.554  0.01107 *
## x1           0.090001   0.052911   1.701  0.08984 .
## x2           0.155741   0.052830   2.948  0.00342 **
## x3           0.148341   0.053414   2.777  0.00578 **
## x4           0.020289   0.053994   0.376  0.70732
## x5           0.004670   0.053971   0.087  0.93110
## x6           0.007733   0.051753   0.149  0.88131
## x7           0.055361   0.051756   1.070  0.28552
## x8           0.009982   0.051805   0.193  0.84731
## x9           0.002042   0.051674   0.040  0.96850
## x10         -0.021888   0.051218  -0.427  0.66939
## x11         -0.057741   0.050622  -1.141  0.25481
## x12          0.162048   0.050563   3.205  0.00148 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03689 on 347 degrees of freedom
## Multiple R-squared:  0.1189, Adjusted R-squared:  0.0884
## F-statistic: 3.901 on 12 and 347 DF,  p-value: 1.236e-05
```

```
library(FinTS)
ArchTest(y)
```

```
##
```

```
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  y
## Chi-squared = 42.794, df = 12, p-value = 2.446e-05
```

```
par(las = 1, mgp = c(2.6, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6), mfrow = c(2, 1))
acf(y^2)
pacf(y^2)
```



Fitting ARCH

```
Intel_m1 <- garchFit(~ 1 + garch(3, 0), data = intc, trace = F)
```

```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
summary(Intel_m1)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
```

```
##  garchFit(formula = ~1 + garch(3, 0), data = intc, trace = F)
##
## Mean and Variance Equation:
##  data ~ 1 + garch(3, 0)
## <environment: 0x7fa4e3fea530>
##  [data = intc]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##       mu     omega    alpha1    alpha2    alpha3
## 0.016572  0.012043  0.208649  0.071837  0.049045
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.016572    0.006423    2.580  0.00988 **
## omega   0.012043    0.001579    7.627  2.4e-14 ***
## alpha1  0.208649    0.129177    1.615  0.10626
## alpha2  0.071837    0.048551    1.480  0.13897
## alpha3  0.049045    0.048847    1.004  0.31536
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  233.4286    normalized:  0.6274962
##
## Description:
##  Wed Oct 20 16:11:08 2021 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  169.773   0
##  Shapiro-Wilk Test  R    W      0.960696  1.970626e-08
##  Ljung-Box Test     R    Q(10)  10.97025  0.3598405
##  Ljung-Box Test     R    Q(15)  19.59024  0.1882211
##  Ljung-Box Test     R    Q(20)  20.82192  0.40768
##  Ljung-Box Test     R^2  Q(10)  5.376602  0.8646439
##  Ljung-Box Test     R^2  Q(15)  22.7346   0.08993974
##  Ljung-Box Test     R^2  Q(20)  23.70577  0.255481
##  LM Arch Test       R    TR^2   20.48506  0.05844884
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -1.228111 -1.175437 -1.228466 -1.207193
```

```
M1 = ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(3, 0)),
                      mean.model = list(armaOrder = c(0, 0), include.mean = T),
                      distribution.model = "norm", fixed.pars = list())
Intel_m1 <- ugarchfit(M1, data = intc)
```

```
Intel_m1
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(3,0)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.016560    0.006419  2.57981 0.009885
## omega   0.012050    0.001591  7.57490 0.000000
## alpha1  0.212953    0.131646  1.61763 0.105743
## alpha2  0.071933    0.048928  1.47016 0.141519
## alpha3  0.049129    0.049221  0.99813 0.318219
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.016560    0.006895   2.4016 0.016322
## omega   0.012050    0.002493   4.8345 0.000001
## alpha1  0.212953    0.193580   1.1001 0.271297
## alpha2  0.071933    0.036720   1.9590 0.050118
## alpha3  0.049129    0.031957   1.5373 0.124211
##
## LogLikelihood : 233.4331
##
## Information Criteria
## ------------------------------------
##
## Akaike       -1.2281
## Bayes        -1.1755
## Shibata      -1.2285
## Hannan-Quinn -1.2072
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.03327  0.8553
## Lag[2*(p+q)+(p+q)-1][2]   0.06686  0.9435
## Lag[4*(p+q)+(p+q)-1][5]   2.04550  0.6077
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                          statistic p-value
## Lag[1]                      0.5564  0.4557
```

```
## Lag[2*(p+q)+(p+q)-1][8]     1.4875  0.9287
## Lag[4*(p+q)+(p+q)-1][14]    6.7064  0.5442
## d.o.f=3
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[4]    0.5757 0.500 2.000  0.4480
## ARCH Lag[6]    0.8634 1.461 1.711  0.7873
## ARCH Lag[8]    1.7352 2.368 1.583  0.7945
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  1.8622
## Individual Statistics:
## mu     0.04824
## omega  0.31431
## alpha1 0.25826
## alpha2 0.57419
## alpha3 0.20981
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.28 1.47 1.88
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias          0.22799 0.8198
## Negative Sign Bias 0.07266 0.9421
## Positive Sign Bias 0.27306 0.7850
## Joint Effect       0.10621 0.9911
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     32.95      0.02439
## 2    30     44.29      0.03444
## 3    40     47.78      0.15799
## 4    50     64.29      0.07027
##
##
## Elapsed time : 0.112042
```

```
Intel_m2 <- garchFit(~ 1 + garch(1, 0), data = intc, trace = F)
```

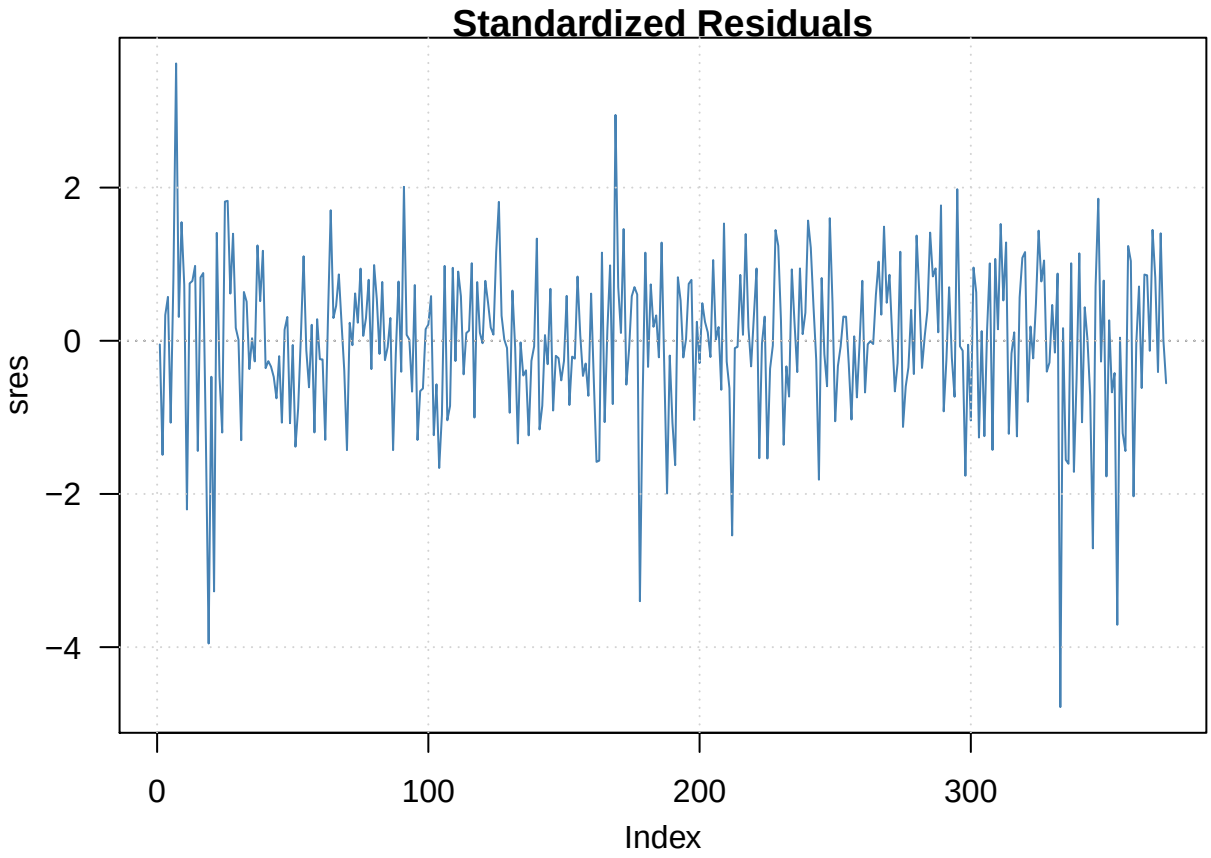
```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
summary(Intel_m2)
```

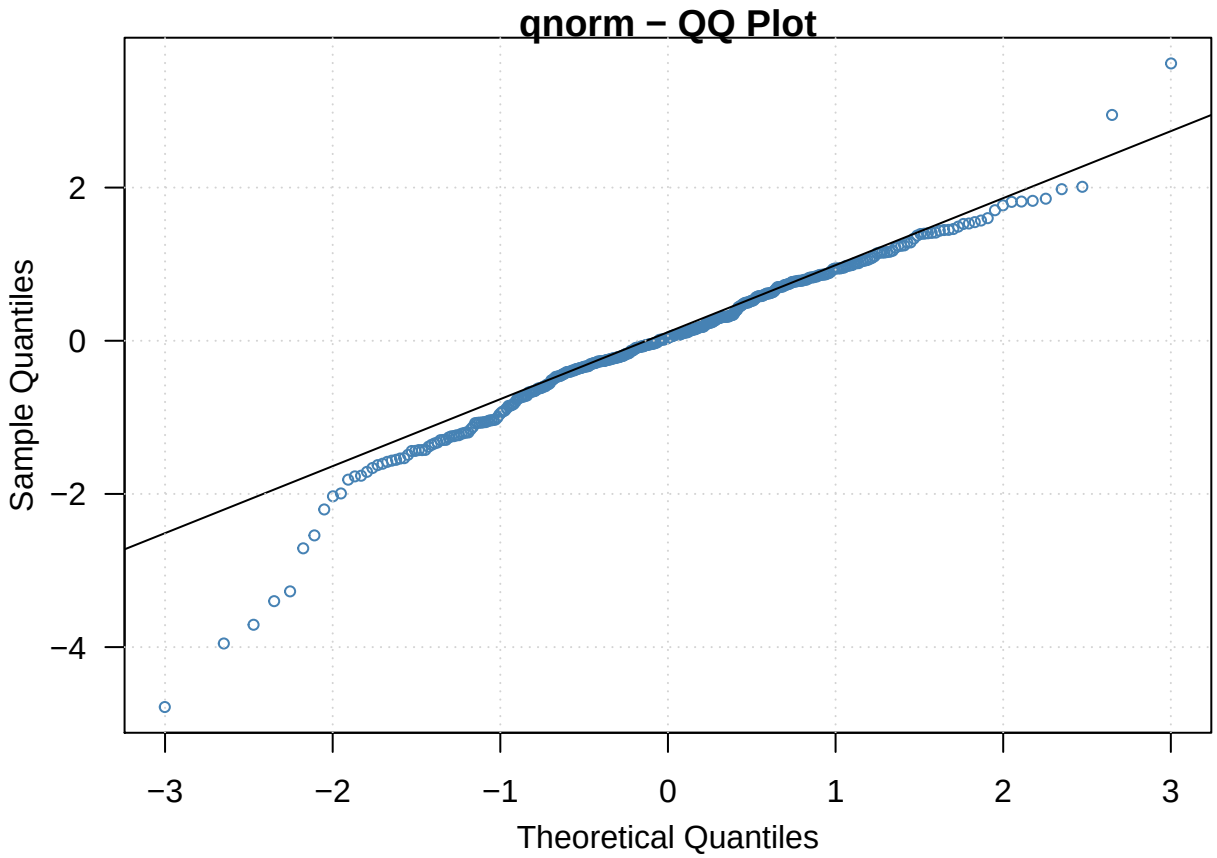
```
##
```

```
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~1 + garch(1, 0), data = intc, trace = F)
##
## Mean and Variance Equation:
##  data ~ 1 + garch(1, 0)
## <environment: 0x7fa4e5e21678>
##  [data = intc]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##      mu    omega   alpha1
## 0.01657  0.01249  0.36345
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.016570    0.006161    2.689  0.00716 **
## omega   0.012490    0.001549    8.061 6.66e-16 ***
## alpha1  0.363447    0.131598    2.762  0.00575 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  230.2423    normalized:  0.6189309
##
## Description:
##  Wed Oct 20 16:11:08 2021 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  122.404   0
##  Shapiro-Wilk Test  R    W      0.9647625 8.273101e-08
##  Ljung-Box Test     R    Q(10)  13.72604  0.1858587
##  Ljung-Box Test     R    Q(15)  22.31714  0.09975386
##  Ljung-Box Test     R    Q(20)  23.88257  0.2475594
##  Ljung-Box Test     R^2  Q(10)  12.50025  0.25297
##  Ljung-Box Test     R^2  Q(15)  30.11276  0.01152131
##  Ljung-Box Test     R^2  Q(20)  31.46404  0.04935483
##  LM Arch Test       R    TR^2   22.036    0.0371183
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -1.221733 -1.190129 -1.221861 -1.209182
```

```
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6))
plot(Intel_m2, which = 9)
```




```
plot(Intel_m2, which = 13)
```



ARCH with Student-t Innovations

```
Intel_m3 <- garchFit(~ 1 + garch(1, 0), data = intc, cond.dist = "std", trace = F)
```

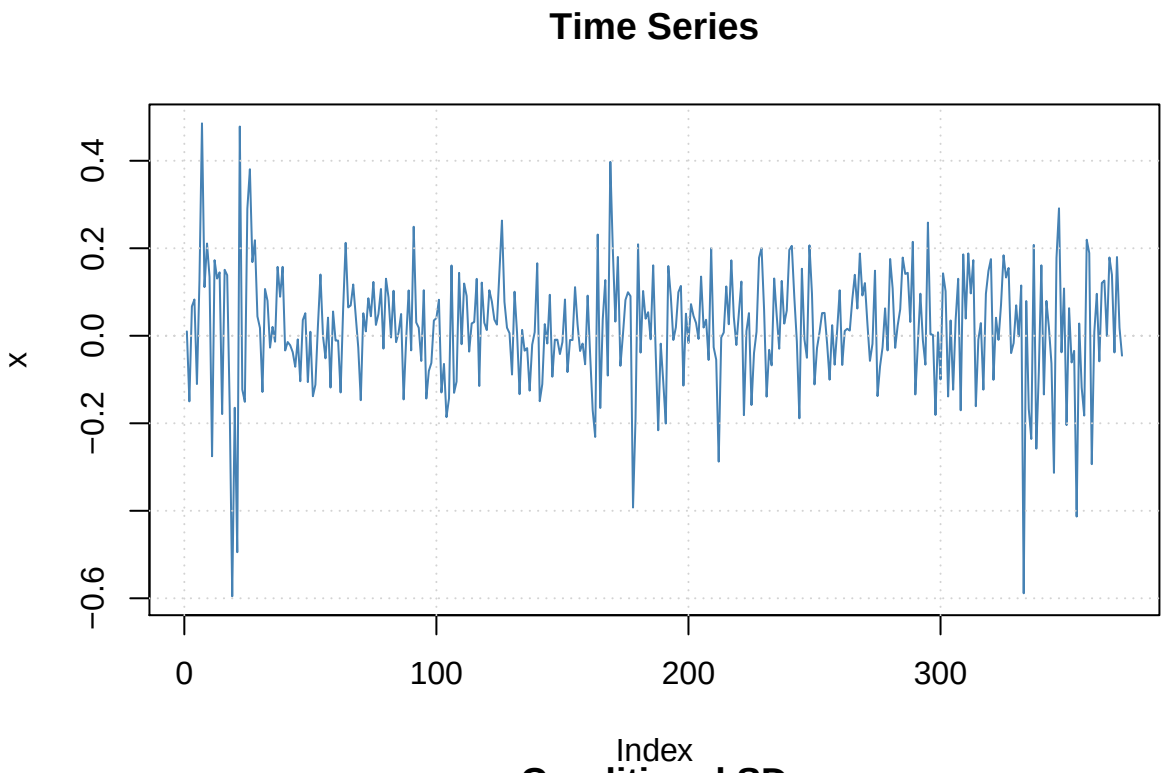
```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

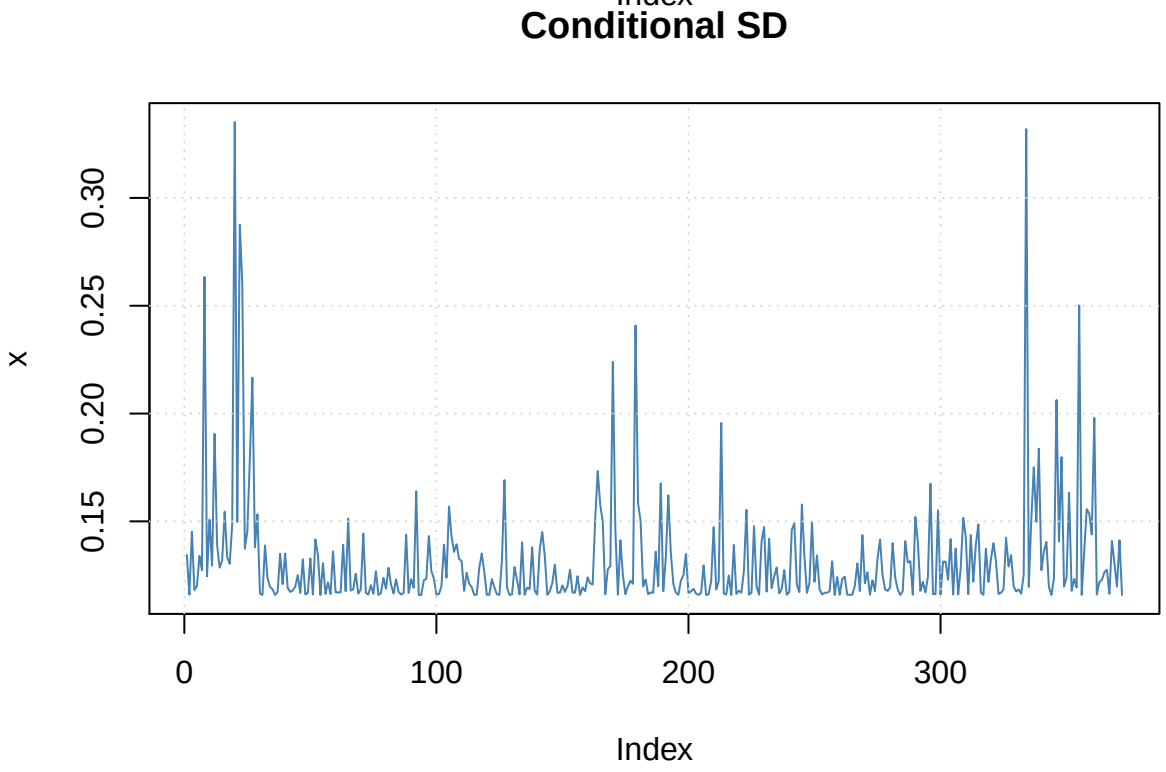
```
summary(Intel_m3)
```

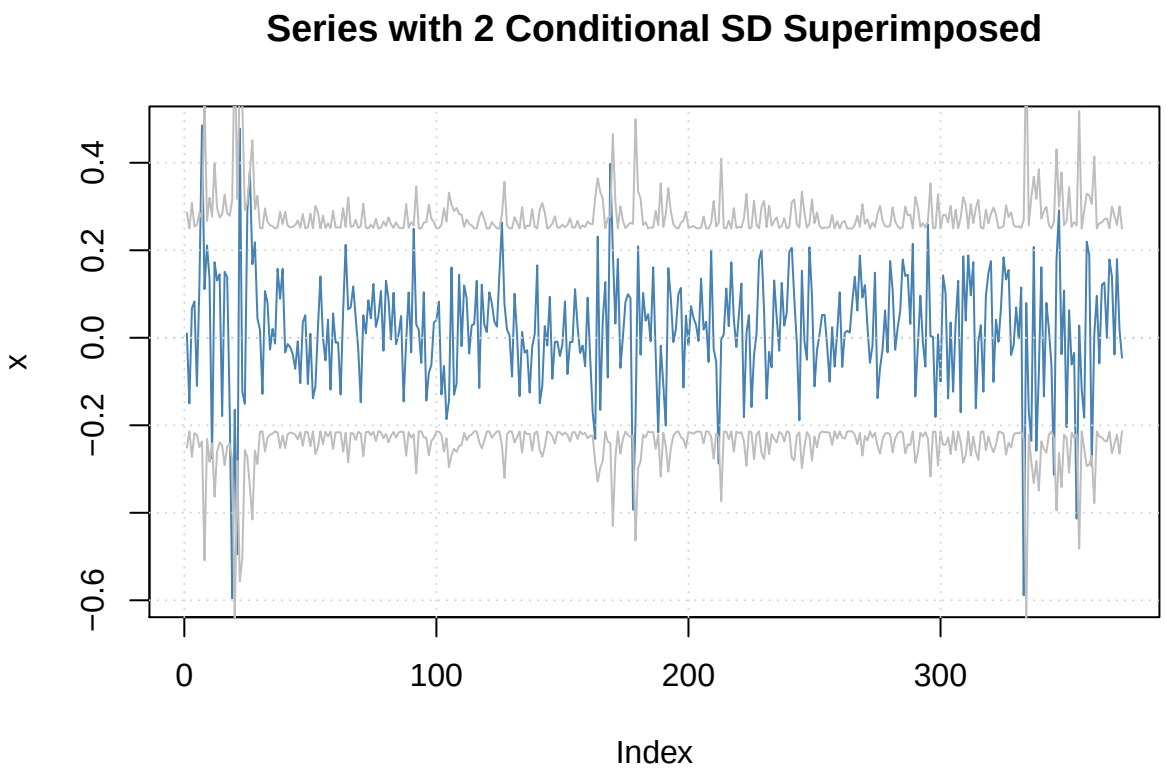
```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~1 + garch(1, 0), data = intc, cond.dist = "std",
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ 1 + garch(1, 0)
## <environment: 0x7fa4d5e97070>
##  [data = intc]
##
## Conditional Distribution:
```

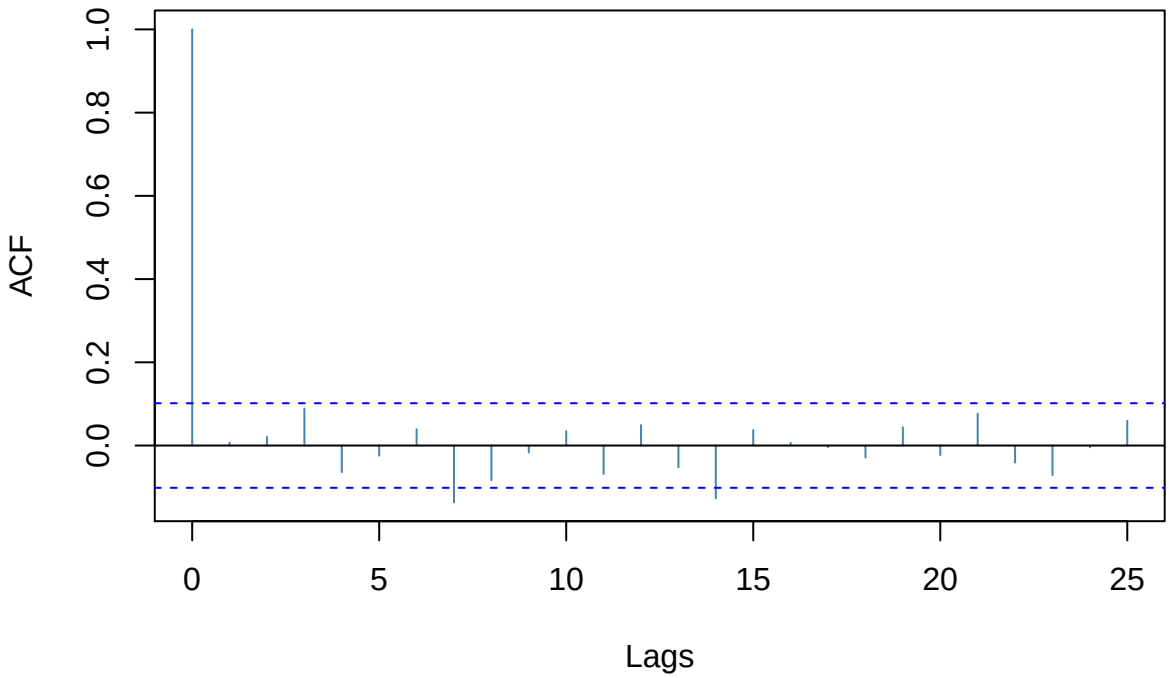
```
##  std
##
## Coefficient(s):
##       mu     omega    alpha1     shape
## 0.021571  0.013424  0.259867  5.985979
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.021571    0.006054    3.563 0.000366 ***
## omega   0.013424    0.001968    6.820 9.09e-12 ***
## alpha1  0.259867    0.119901    2.167 0.030209 *
## shape   5.985979    1.660030    3.606 0.000311 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  242.9678    normalized:  0.6531391
##
## Description:
##  Wed Oct 20 16:11:09 2021 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  130.8931  0
##  Shapiro-Wilk Test  R    W      0.9637533 5.744995e-08
##  Ljung-Box Test     R    Q(10)  14.31288  0.1591926
##  Ljung-Box Test     R    Q(15)  23.34043  0.07717449
##  Ljung-Box Test     R    Q(20)  24.87286  0.2063387
##  Ljung-Box Test     R^2  Q(10)  15.35917  0.1195054
##  Ljung-Box Test     R^2  Q(15)  33.96318  0.003446127
##  Ljung-Box Test     R^2  Q(20)  35.46828  0.01774746
##  LM Arch Test       R    TR^2   24.11517  0.01961957
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -1.284773 -1.242634 -1.285001 -1.268039
```

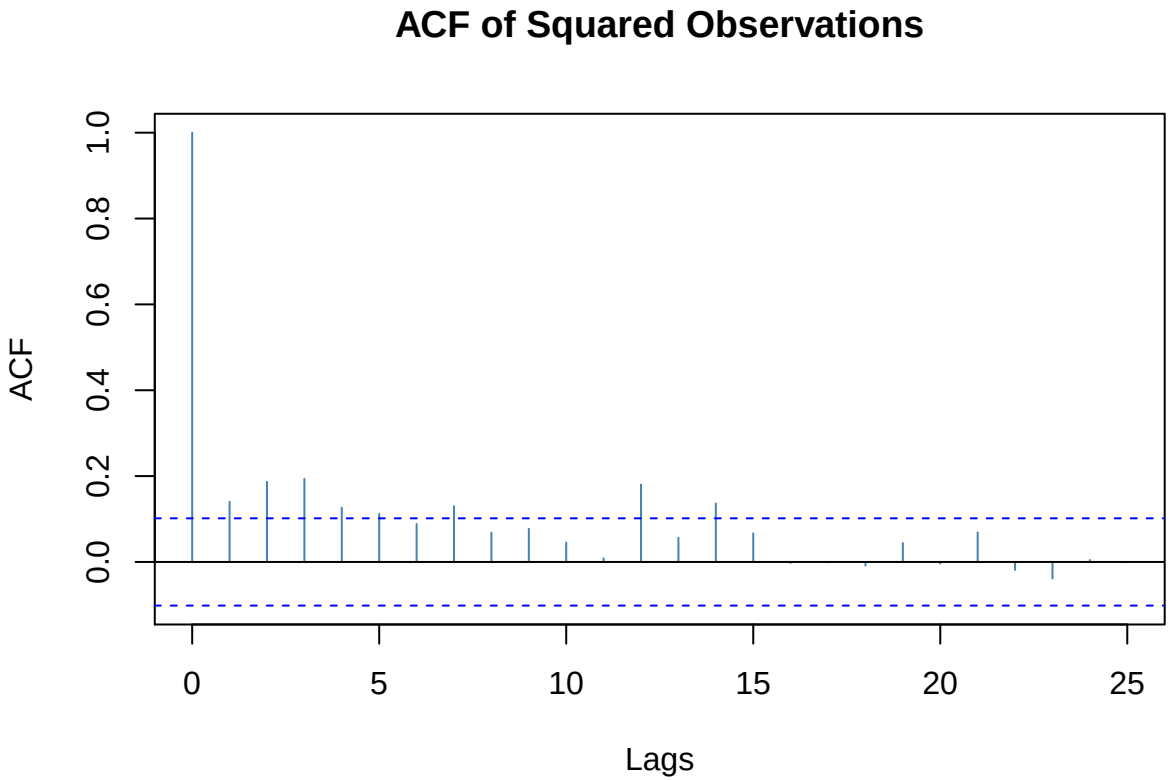
```
plot(Intel_m3, which = 1:13)
```

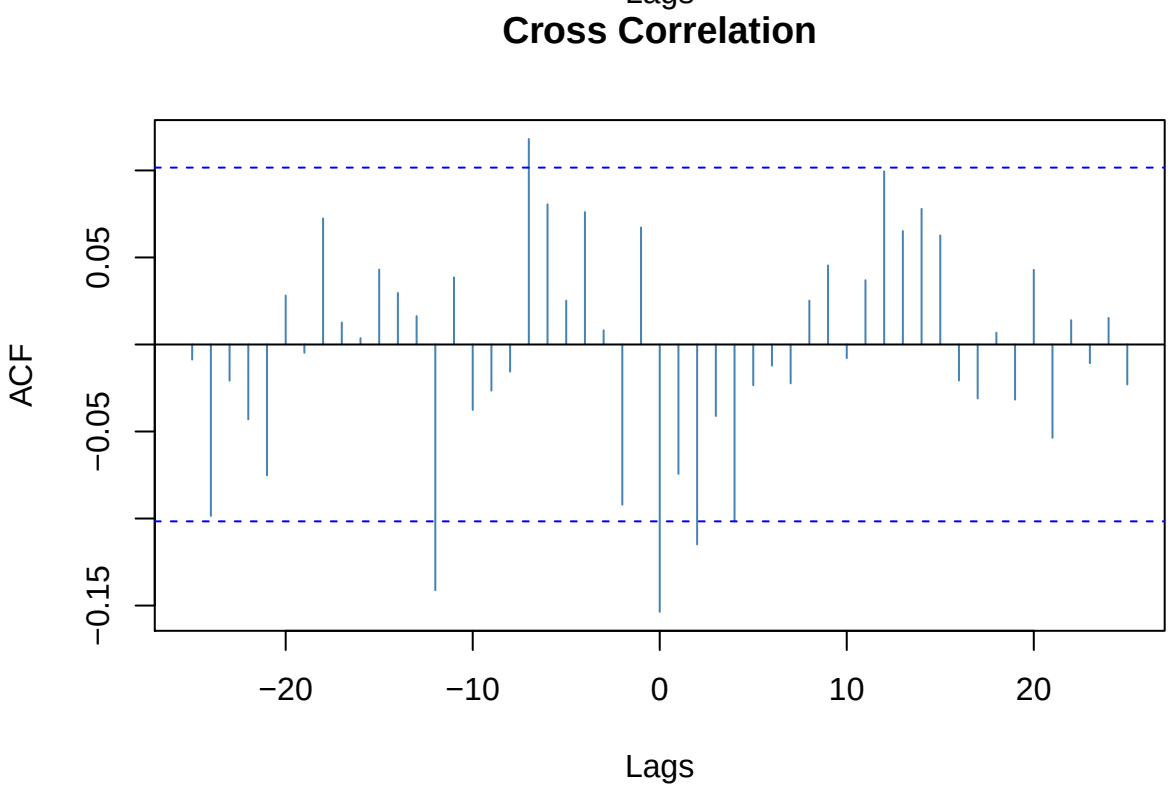

Time Series
x
0.4
0.2
0.0
−0.2
−0.6
0
100
200
300
Index

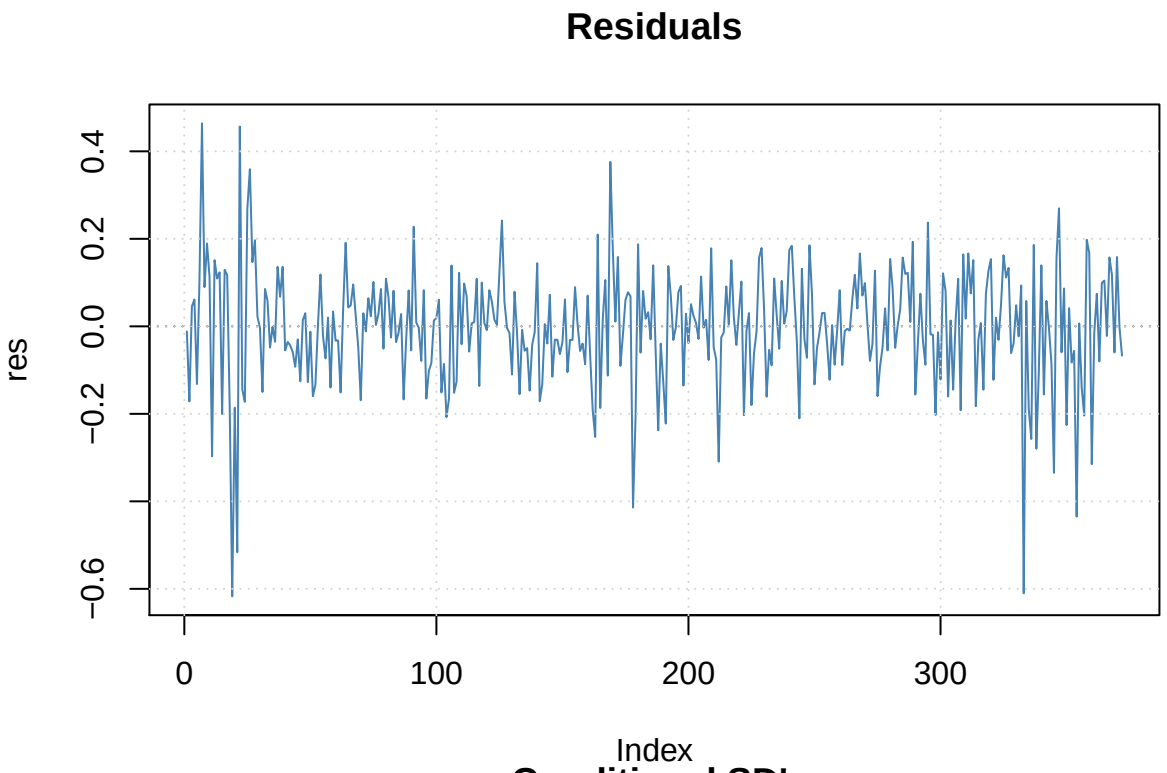

Conditional SD
x
0.30
0.25
0.20
0.15
0
100
200
300
Index

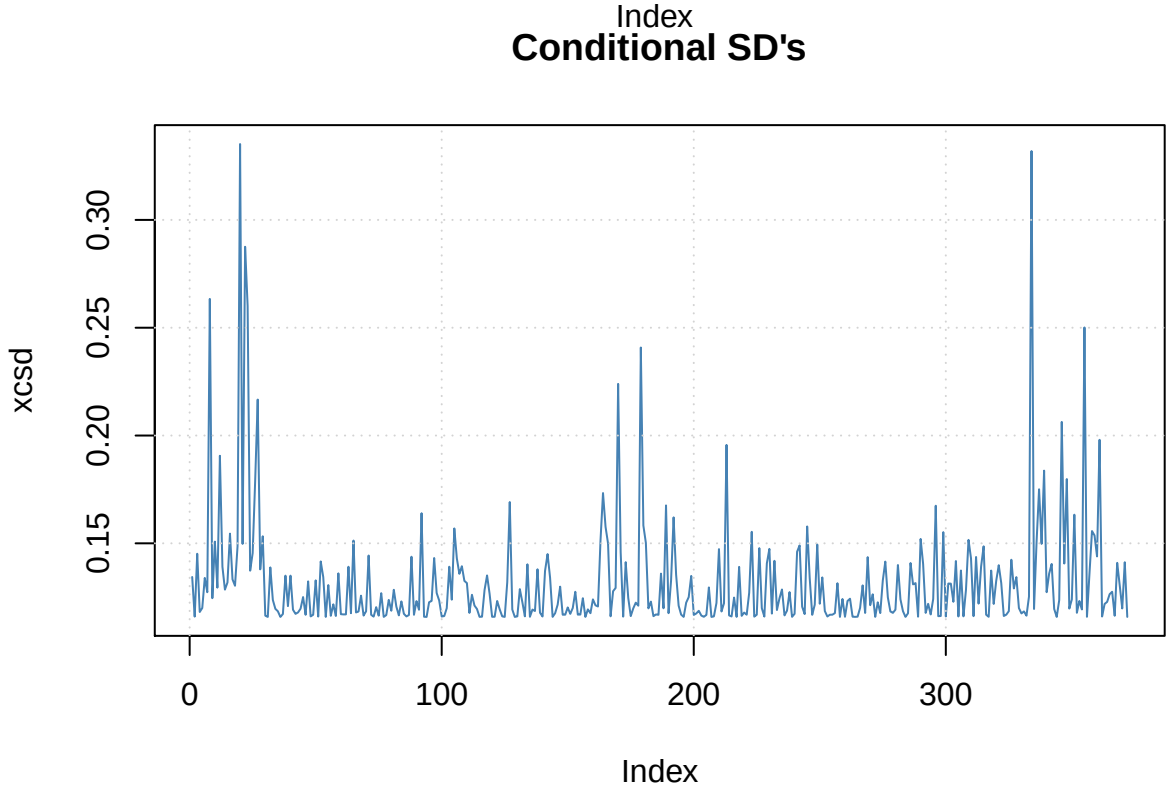

Series with 2 Conditional SD Superimposed
x
0.4
0.2
0.0
−0.2
−0.6
0
100
200
300
Index

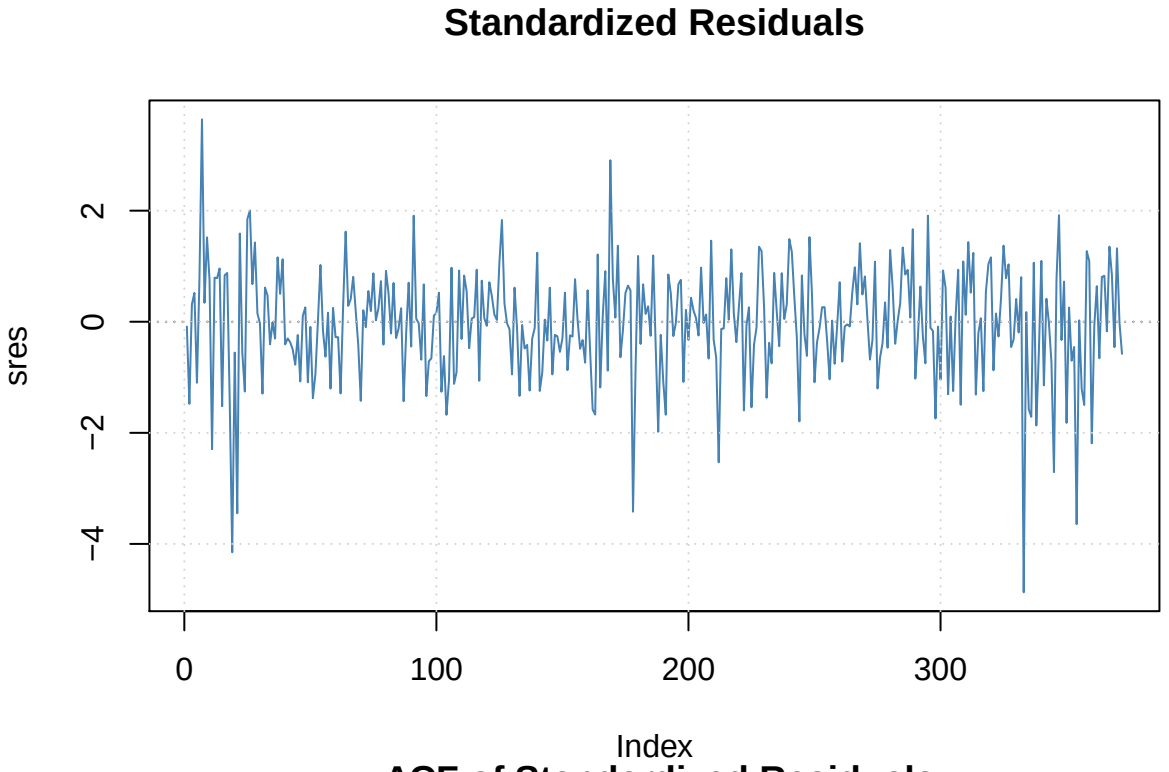

ACF of Observations
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Lags

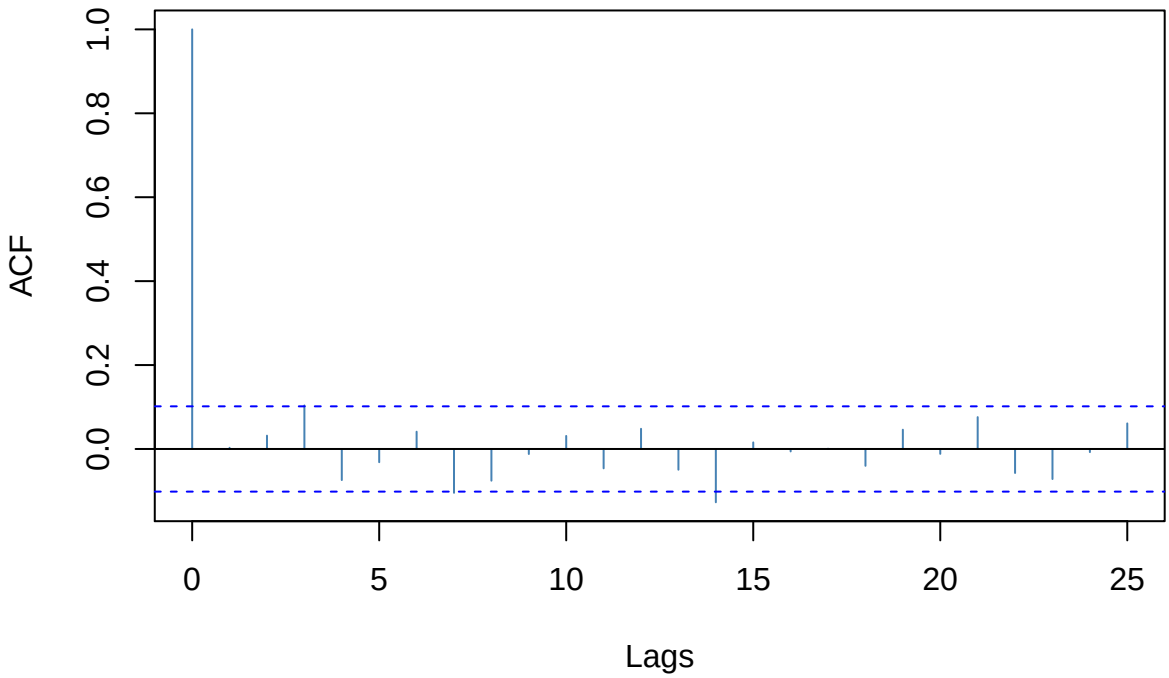

ACF of Squared Observations
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
Lags

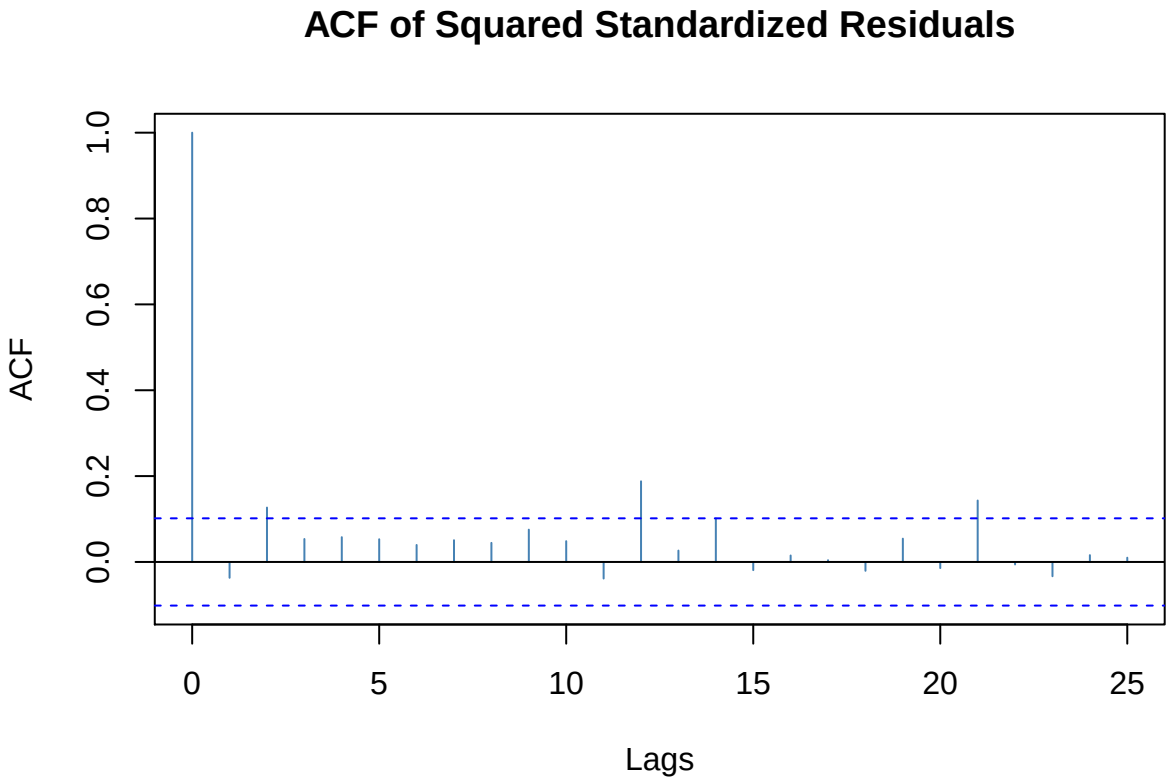

Cross Correlation
ACF
−0.15
−0.05
0.05
−20
−10
0
10
20
Lags

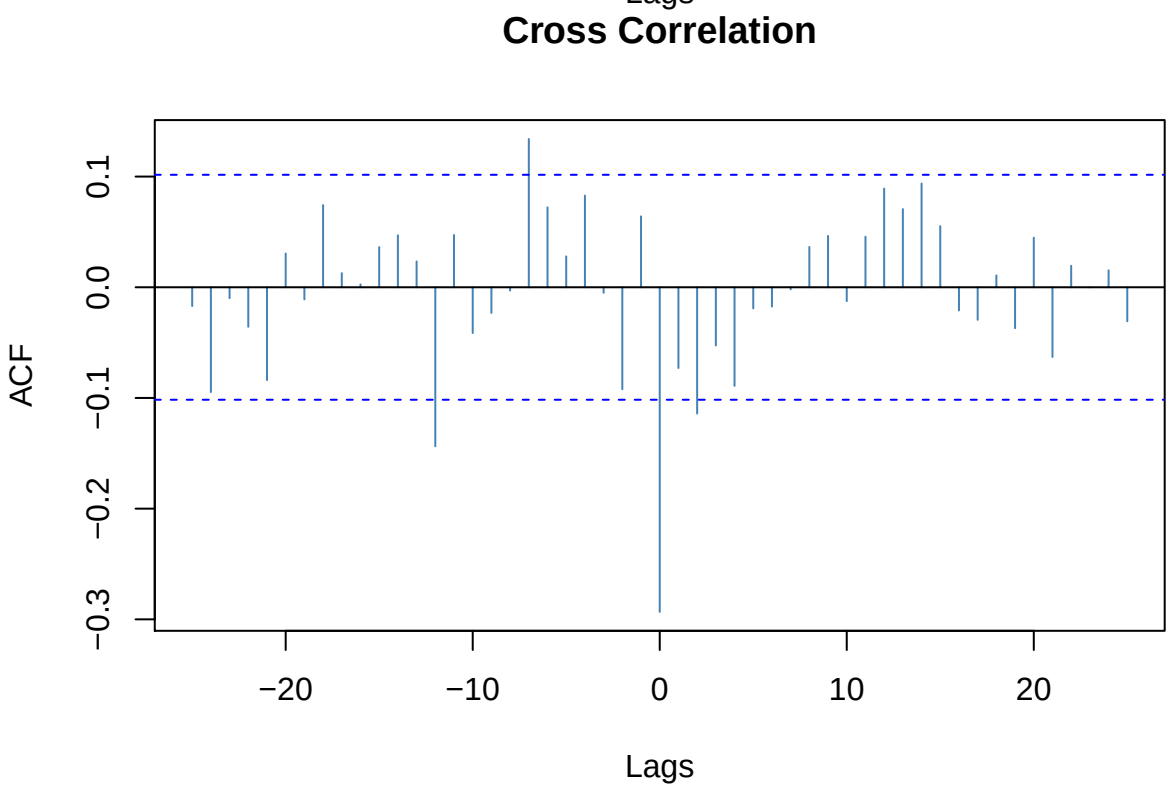

Residuals
res
0.4
0.2
0.0
−0.2
−0.6
0
100
200
300
Index

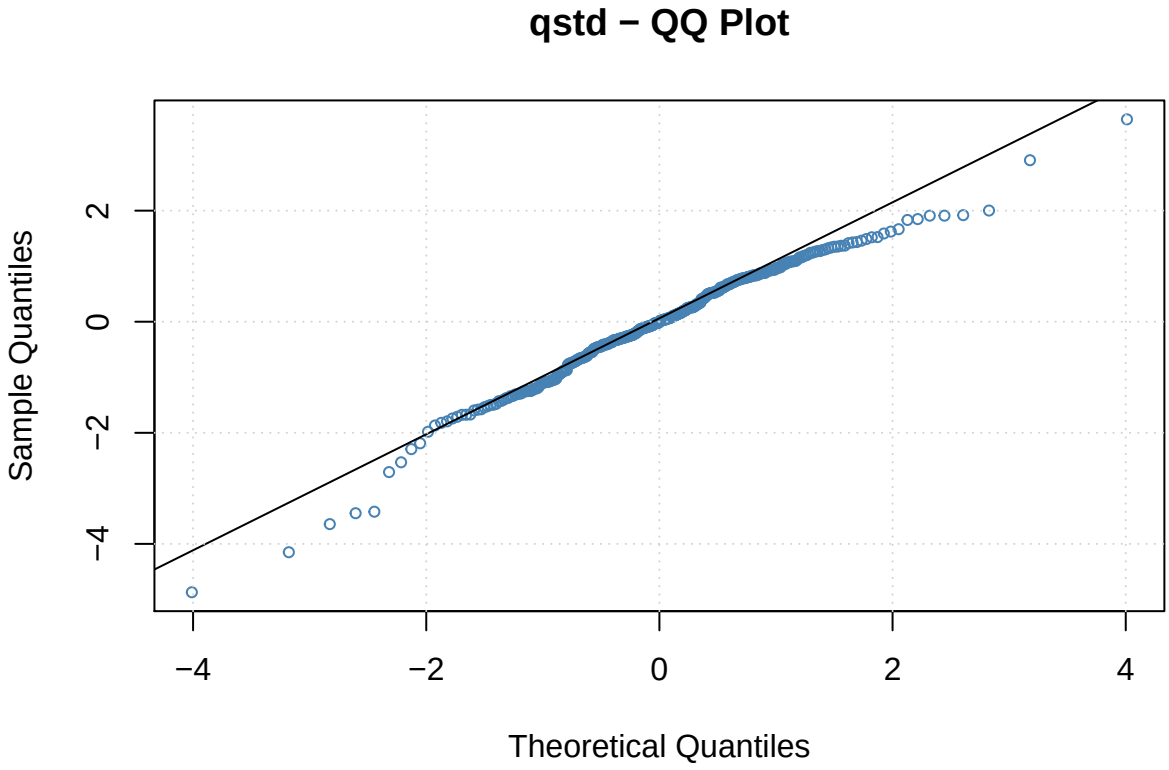

Conditional SD's
xcsd
0.30
0.25
0.20
0.15
0
100
200
300
Index


Standardized Residuals
sres
2
0
−2
−4
0
100
200
300
Index


ACF of Standardized Residuals
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
Lags


ACF of Squared Standardized Residuals
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
Lags


Cross Correlation
ACF
−0.3
−0.2
−0.1
0.0
0.1
−20
−10
0
10
20
Lags




```
predict(Intel_m3, 5)
```

```
##   meanForecast meanError standardDeviation
## 1     0.021571 0.1207911         0.1207911
## 2     0.021571 0.1312069         0.1312069
## 3     0.021571 0.1337810         0.1337810
## 4     0.021571 0.1344418         0.1344418
## 5     0.021571 0.1346130         0.1346130
```

```
M3 = ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 0)),
                      mean.model = list(armaOrder = c(0, 0), include.mean = T),
                      distribution.model = "std", fixed.pars = list())
Intel_m3 <- ugarchfit(M3, data = intc)
Intel_m3
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,0)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : std
##
## Optimal Parameters
## ------------------------------------
```

```
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.021565    0.006053   3.5627 0.000367
## omega   0.013476    0.001996   6.7508 0.000000
## alpha1  0.263910    0.121752   2.1676 0.030189
## shape   5.937781    1.655116   3.5875 0.000334
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.021565    0.006365   3.3880 0.000704
## omega   0.013476    0.002132   6.3203 0.000000
## alpha1  0.263910    0.153254   1.7220 0.085061
## shape   5.937781    1.475958   4.0230 0.000057
##
## LogLikelihood : 242.9753
##
## Information Criteria
## ------------------------------------
##
## Akaike       -1.2848
## Bayes        -1.2427
## Shibata      -1.2850
## Hannan-Quinn -1.2681
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                   0.003118  0.9555
## Lag[2*(p+q)+(p+q)-1][2]  0.193986  0.8579
## Lag[4*(p+q)+(p+q)-1][5]  3.649143  0.3012
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.5263 0.46815
## Lag[2*(p+q)+(p+q)-1][2]    3.5205 0.10150
## Lag[4*(p+q)+(p+q)-1][5]    6.6637 0.06253
## d.o.f=1
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[2]     5.924 0.500 2.000 0.01493
## ARCH Lag[4]     7.325 1.397 1.611 0.02349
## ARCH Lag[6]     8.323 2.222 1.500 0.03480
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  1.2181
## Individual Statistics:
## mu     0.05211
## omega  0.48258
## alpha1 0.37199
```

```
## shape  0.13634
##
## 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           0.18119 0.8563
## Negative Sign Bias 0.39211 0.6952
## Positive Sign Bias 0.06475 0.9484
## Joint Effect       0.16040 0.9837
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     32.95      0.02439
## 2    30     38.48      0.11195
## 3    40     42.62      0.31802
## 4    50     55.42      0.24547
##
##
## Elapsed time : 0.09191203
```

GARCH Bollerslev (1986)

$$a_t = \sigma_t \epsilon_t, \quad \sigma_t^2 = \alpha_0 + \sum_{i=1}^{m} \alpha_i a_{t-i}^2 + \sum_{j=1}^{s} \beta_j \sigma_{t-j}^2.$$

```
Intel_m4 <- garchFit(~ 1 + garch(1, 1), data = intc, trace = F)
```

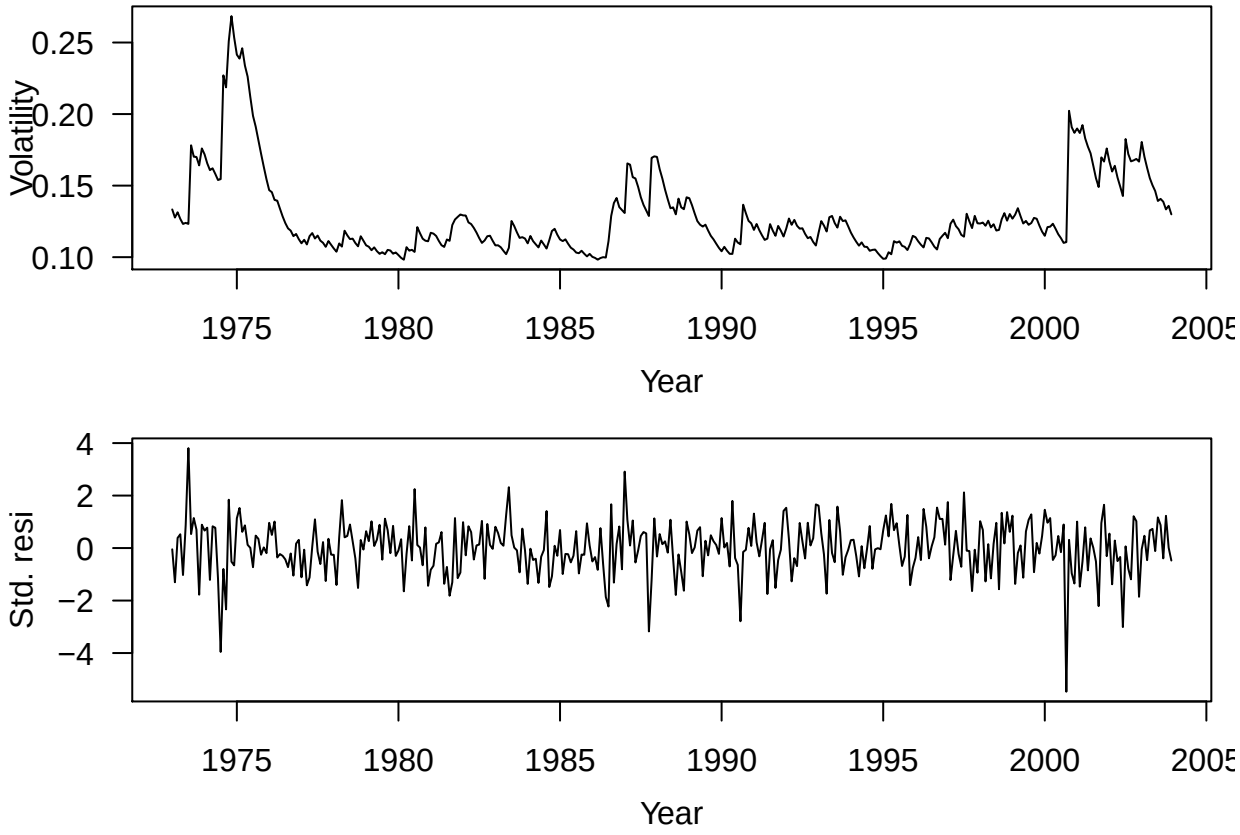
```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
summary(Intel_m4)
```

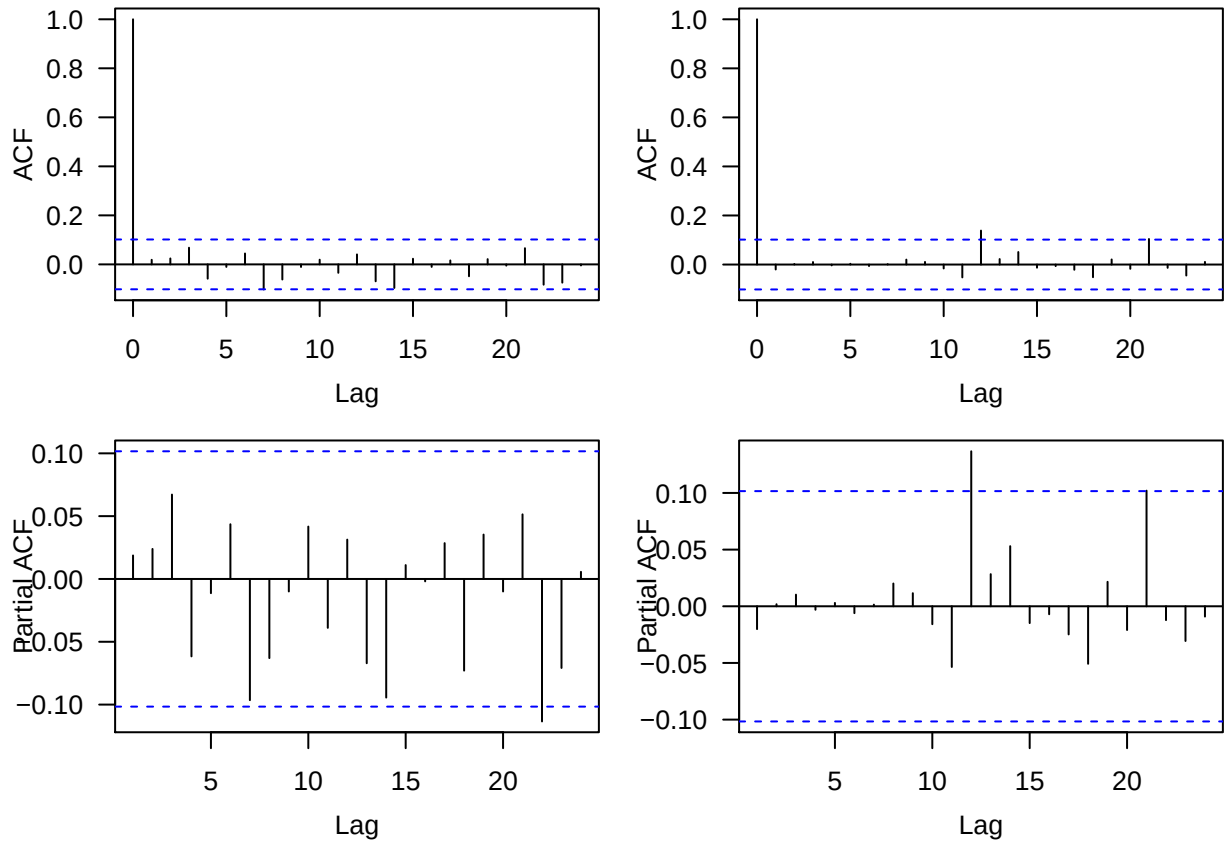
```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~1 + garch(1, 1), data = intc, trace = F)
##
## Mean and Variance Equation:
##  data ~ 1 + garch(1, 1)
## <environment: 0x7fa500156eb8>
##  [data = intc]
##
## Conditional Distribution:
```

```
##  norm
##
## Coefficient(s):
##        mu      omega     alpha1      beta1
## 0.0163276  0.0010918  0.0802716  0.8553014
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     0.0163276   0.0062624    2.607  0.00913 **
## omega  0.0010918   0.0005291    2.063  0.03907 *
## alpha1 0.0802716   0.0281162    2.855  0.00430 **
## beta1  0.8553014   0.0461374   18.538  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  239.5189    normalized:  0.6438681
##
## Description:
##  Wed Oct 20 16:11:14 2021 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  156.5138  0
##  Shapiro-Wilk Test  R    W      0.9676933 2.471139e-07
##  Ljung-Box Test     R    Q(10)  9.805485  0.4577215
##  Ljung-Box Test     R    Q(15)  16.54435  0.346824
##  Ljung-Box Test     R    Q(20)  17.8005   0.6005484
##  Ljung-Box Test     R^2  Q(10)  0.5130171 0.9999925
##  Ljung-Box Test     R^2  Q(15)  10.24557  0.8040151
##  Ljung-Box Test     R^2  Q(20)  11.77988  0.9234441
##  LM Arch Test       R    TR^2   9.334459  0.6741288
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -1.266231 -1.224092 -1.266459 -1.249496
```

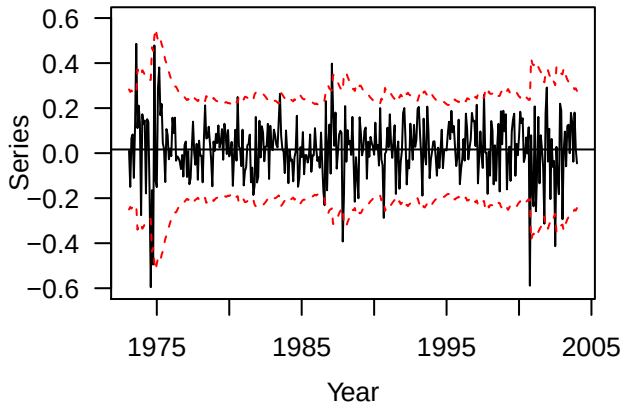
```r
mu <- Intel_m4@fit$par[1]

v1 <- volatility(Intel_m4)
resi <- residuals(Intel_m4, standardize = T)
vol <- ts(v1, frequency = 12, start = c(1973, 1))
res <- ts(resi, frequency = 12, start = c(1973, 1))

par(las = 1, mgp = c(2.4, 1, 0), mar = c(3.6, 3.8, 0.8, 0.6), mfcol = c(2, 1))
plot(vol, xlab = 'Year', ylab = 'Volatility',type = 'l')
plot(res, xlab = 'Year', ylab = 'Std. resi', type = 'l')
```




```
par(las = 1, mgp = c(2.4, 1, 0), mar = c(3.6, 3.8, 0.8, 0.6), mfcol = c(2, 2))
acf(resi, lag = 24)
pacf(resi, lag = 24)
acf(resi^2, lag = 24)
pacf(resi^2, lag = 24)
```




```
par(las = 1, mgp = c(2.4, 1, 0), mar = c(3.6, 3.8, 0.8, 0.6))
upp = mu + 2 * v1; low = mu - 2 * v1
tdx <- (1:length(intc)) / 12 + 1973
plot(tdx, intc, xlab = 'Year', ylab = 'Series', type = 'l', ylim = c(-0.6, 0.6))
lines(tdx, upp, lty = 2, col = 'red'); lines(tdx, low, lty = 2, col = 'red')
abline(h = mu)
```



IGARCH

```
source("Igarch.R")
IGARCH_fit <- Igarch(intc)
```

```
## Estimates:  0.9248279
## Maximized log-likehood:  -228.7737
##
## Coefficient(s):
##       Estimate  Std. Error  t value   Pr(>|t|)
## beta 0.9248279   0.0186781  49.5139 < 2.22e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
names(IGARCH_fit)
```

```
## [1] "par"        "volatility"
```

```
Intel_m5 = ugarchspec(variance.model = list(model = "iGARCH", garchOrder = c(1, 1)), mean.model = list(a
                      fixed.pars = list())

Intel_m5 <- ugarchfit(Intel_m5, data = intc)
Intel_m5
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : iGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.015365    0.006210   2.4741 0.013357
## omega   0.000336    0.000206   1.6318 0.102719
## alpha1  0.113395    0.035860   3.1621 0.001566
## beta1   0.886605          NA       NA       NA
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.015365    0.007326   2.0973 0.035968
## omega   0.000336    0.000213   1.5765 0.114903
## alpha1  0.113395    0.037279   3.0418 0.002352
## beta1   0.886605          NA       NA       NA
##
## LogLikelihood : 236.014
##
## Information Criteria
## ------------------------------------
##
## Akaike       -1.2528
## Bayes        -1.2212
## Shibata      -1.2529
```

```
## Hannan-Quinn -1.2402
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.1910  0.6621
## Lag[2*(p+q)+(p+q)-1][2]    0.3424  0.7726
## Lag[4*(p+q)+(p+q)-1][5]    1.7669  0.6744
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.1677  0.6822
## Lag[2*(p+q)+(p+q)-1][5]    0.3334  0.9801
## Lag[4*(p+q)+(p+q)-1][9]    0.4189  0.9991
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]   0.01600 0.500 2.000  0.8993
## ARCH Lag[5]   0.07147 1.440 1.667  0.9918
## ARCH Lag[7]   0.12253 2.315 1.543  0.9990
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  1.5876
## Individual Statistics:
## mu     0.05671
## omega  0.13078
## alpha1 0.49311
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          0.846 1.01 1.35
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias          0.00311 0.9975
## Negative Sign Bias 0.03265 0.9740
## Positive Sign Bias 0.02437 0.9806
## Joint Effect       0.00404 0.9999
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     29.08      0.06481
## 2    30     35.74      0.18120
## 3    40     46.28      0.19700
## 4    50     55.42      0.24547
```

```
##
##
## Elapsed time : 0.03114009
```

EGARCH Nelson (1991)

```
source("Egarch.R")
IBM <- read.table("m-ibmsp6709.txt", header = T)
ibm <- log(IBM$ibm + 1)
Box.test(ibm, lag = 12, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  ibm
## X-squared = 7.4042, df = 12, p-value = 0.8298
```

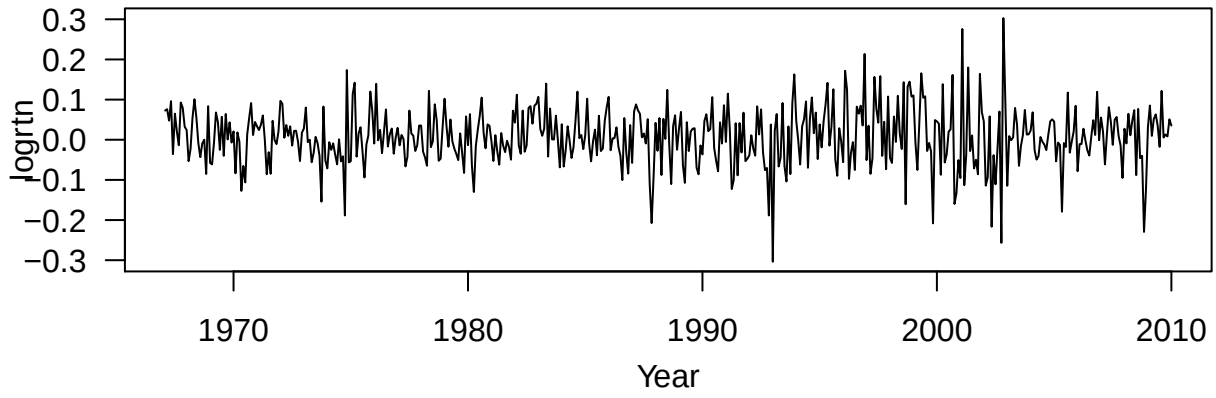
```
EGARCH_fit <- Egarch(ibm)
```

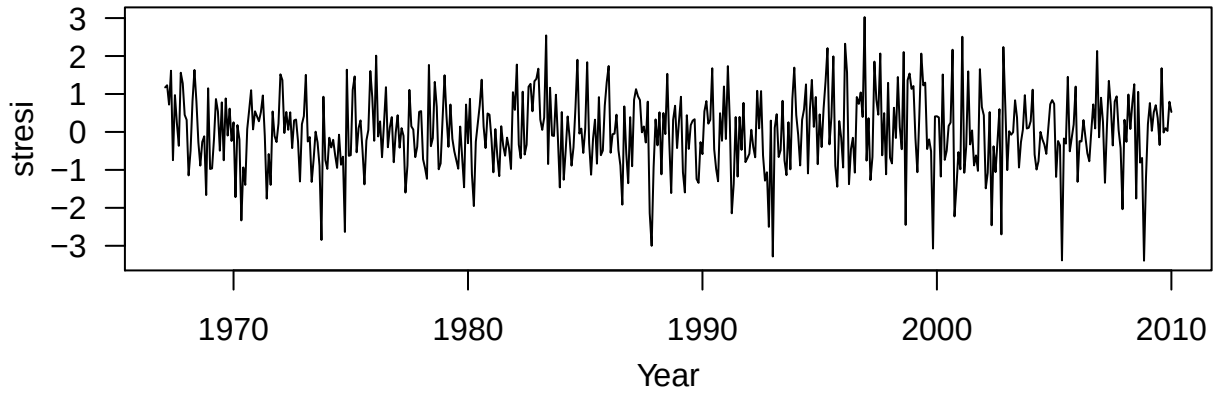
```
##
## Estimation results of EGARCH(1,1) model:
## estimates:  0.006732389 -0.5983263 0.217603 -0.4243245 0.92015
## std.errors:  0.002877666 0.2349172 0.05916528 0.1683064 0.0388656
## t-ratio:  2.339531 -2.546967 3.677882 -2.521144 23.67518
```

```
names(EGARCH_fit)
```

```
## [1] "residuals"  "volatility"
```

```
stresi <- EGARCH_fit$residuals / EGARCH_fit$volatility
tdx = (1:length(ibm)) / 12 + 1967
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6), mfcol = c(2, 1))
plot(tdx, ibm, xlab = 'Year', ylab = 'logrtn', type = 'l')
plot(tdx,stresi, xlab = 'Year', ylab = 'stresi',type = 'l')
```






```
Box.test(stresi, lag = 10, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  stresi
## X-squared = 5.2866, df = 10, p-value = 0.8712
```

```
Box.test(stresi, lag = 20, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  stresi
## X-squared = 20.983, df = 20, p-value = 0.3981
```

```
Box.test(stresi^2, lag = 10, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  stresi^2
## X-squared = 5.0469, df = 10, p-value = 0.888
```

```r
Box.test(stresi^2, lag = 20, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  stresi^2
## X-squared = 14.261, df = 20, p-value = 0.817
```

```r
IBM_egarch <- ugarchspec(variance.model = list(model = "eGARCH", garchOrder = c(1, 1)), mean.model = lis
                     fixed.pars = list())

(IBM_egarch <- ugarchfit(IBM_egarch, data = ibm))
```

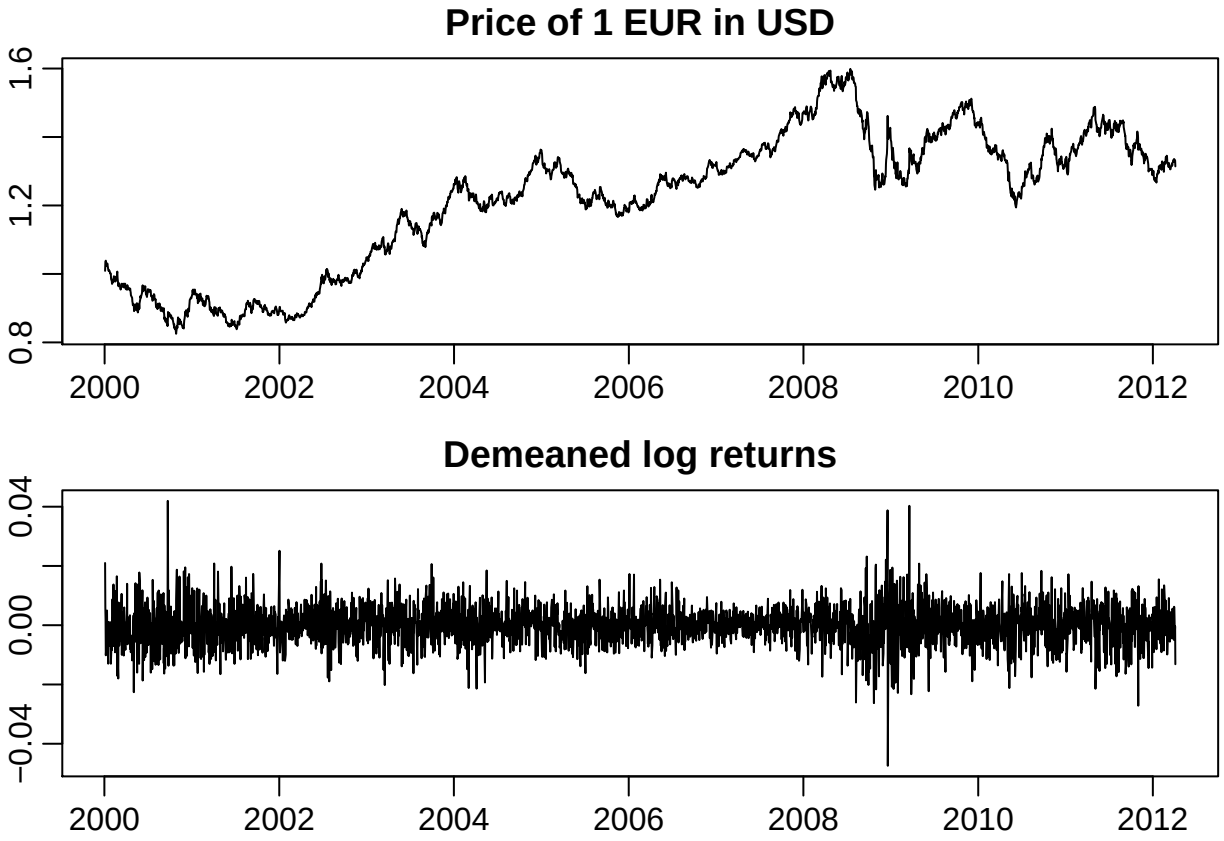
```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : eGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.006649    0.002963   2.2442 0.024820
## omega  -0.423208    0.223673  -1.8921 0.058480
## alpha1 -0.094813    0.039373  -2.4081 0.016037
## beta1   0.920485    0.041729  22.0586 0.000000
## gamma1  0.218711    0.060802   3.5971 0.000322
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.006649    0.003073   2.1635 0.030502
## omega  -0.423208    0.308230  -1.3730 0.169743
## alpha1 -0.094813    0.051835  -1.8291 0.067382
## beta1   0.920485    0.057270  16.0728 0.000000
## gamma1  0.218711    0.061770   3.5407 0.000399
##
## LogLikelihood : 651.634
##
## Information Criteria
## ------------------------------------
##
## Akaike       -2.5063
## Bayes        -2.4652
## Shibata      -2.5065
## Hannan-Quinn -2.4902
##
## Weighted Ljung-Box Test on Standardized Residuals
```

```
## -------------------------------------
##                           statistic p-value
## Lag[1]                        1.237  0.2661
## Lag[2*(p+q)+(p+q)-1][2]       1.344  0.3989
## Lag[4*(p+q)+(p+q)-1][5]       1.867  0.6501
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## -------------------------------------
##                           statistic p-value
## Lag[1]                     0.009633  0.9218
## Lag[2*(p+q)+(p+q)-1][5]    1.087446  0.8396
## Lag[4*(p+q)+(p+q)-1][9]    2.511467  0.8360
## d.o.f=2
##
## Weighted ARCH LM Tests
## -------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]   0.09128 0.500 2.000  0.7626
## ARCH Lag[5]   1.10479 1.440 1.667  0.7021
## ARCH Lag[7]   2.20197 2.315 1.543  0.6746
##
## Nyblom stability test
## -------------------------------------
## Joint Statistic:  1.1719
## Individual Statistics:
## mu     0.21948
## omega  0.61756
## alpha1 0.15868
## beta1  0.61824
## gamma1 0.06386
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.28 1.47 1.88
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## -------------------------------------
##                    t-value   prob sig
## Sign Bias           0.1014 0.9192
## Negative Sign Bias  0.2560 0.7980
## Positive Sign Bias  0.1888 0.8503
## Joint Effect        0.3726 0.9458
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## -------------------------------------
##   group statistic p-value(g-1)
## 1    20     13.07       0.8350
## 2    30     22.26       0.8094
## 3    40     28.03       0.9040
## 4    50     42.53       0.7314
##
```

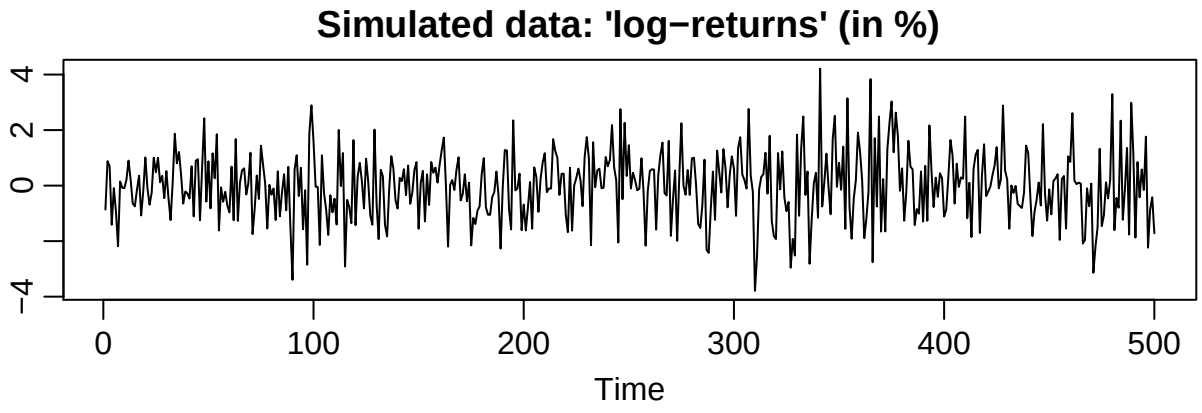
```
##
## Elapsed time : 0.06183004
```

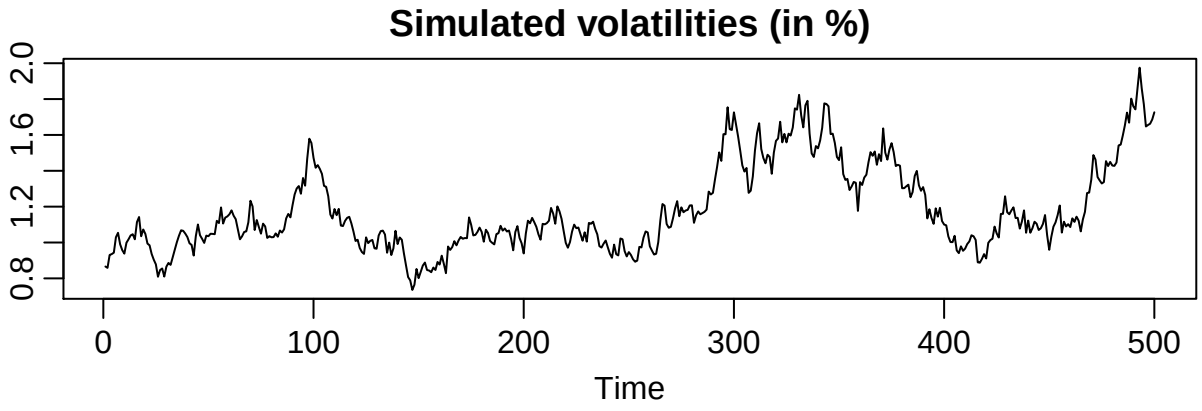
Stochastic Volatility (SV) Model Melino and Turnbull (1990); Harvey, Ruiz, and Shephard (1994); Jacquier, Polson, and Rossi (2002)

```
set.seed(123)
library("stochvol")
data("exrates")
ret <- logret(exrates$USD, demean = TRUE)
par(mfrow = c(2, 1), mar = c(1.9, 1.9, 1.9, 0.5), mgp = c(2, 0.6, 0))
plot(exrates$date, exrates$USD, type = "l", main = "Price of 1 EUR in USD")
plot(exrates$date[-1], ret, type = "l", main = "Demeaned log returns")
```




```
sim <- svsim(500, mu = -9, phi = 0.99, sigma = 0.1)
par(mfrow = c(2, 1))
plot(sim)
```






```
res <- svsample(ret, priormu = c(-10, 1), priorphi = c(20, 1.1), priorsigma = 0.1)
```

```
## Done!
```

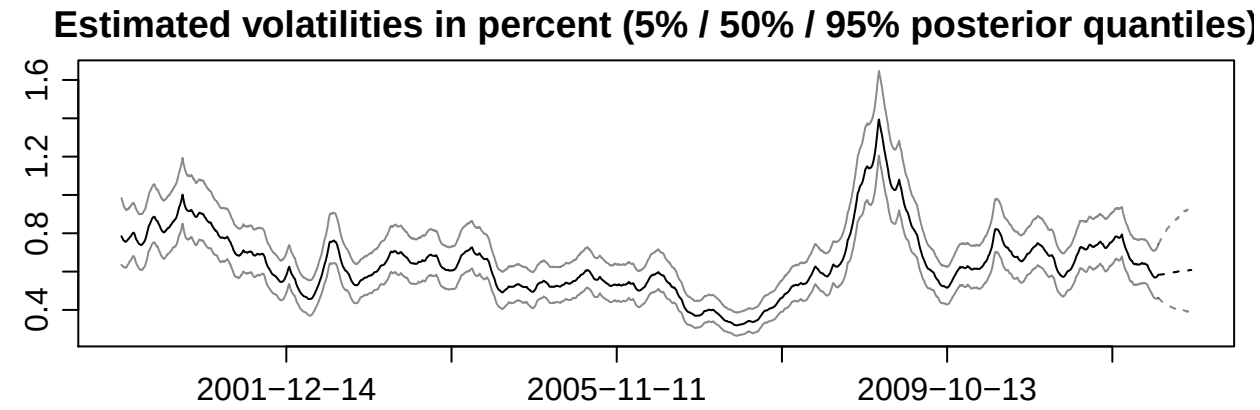
```
## Summarizing posterior draws...
```

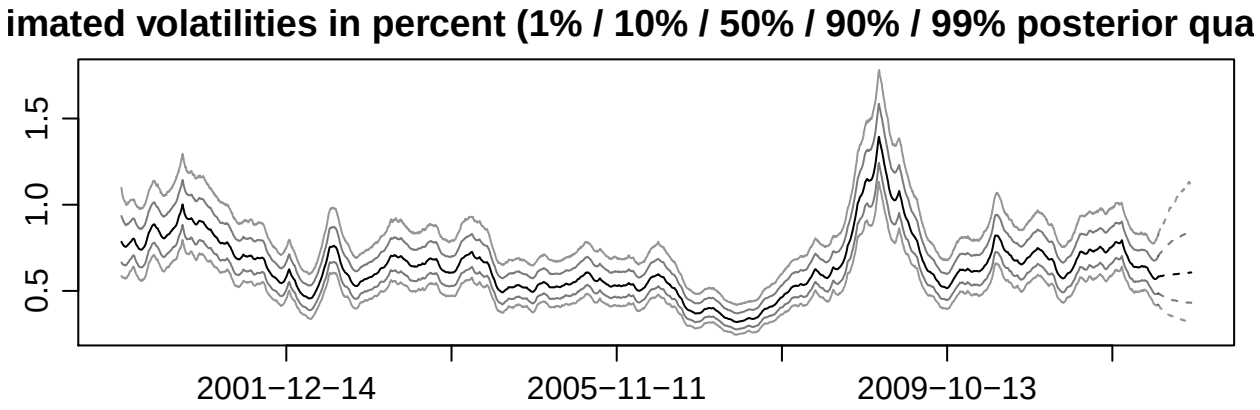
```
summary(res, showlatent = FALSE)
```

```
##
## Summary of 'svdraws' object
##
## Prior distributions:
## mu        ~ Normal(mean = -10, sd = 1)
## (phi+1)/2 ~ Beta(a = 20, b = 1.1)
## sigma^2   ~ Gamma(shape = 0.5, rate = 5)
## nu        ~ Infinity
## rho       ~ Constant(value = 0)
##
## Stored 10000 MCMC draws after a burn-in of 1000.
## No thinning.
##
## Posterior draws of SV parameters (thinning = 1):
##               mean      sd       5%      50%     95%  ESS
## mu        -10.1366 0.22711 -10.4749 -10.1399 -9.7933 4552
## phi         0.9935 0.00282   0.9886   0.9938  0.9977  397
```

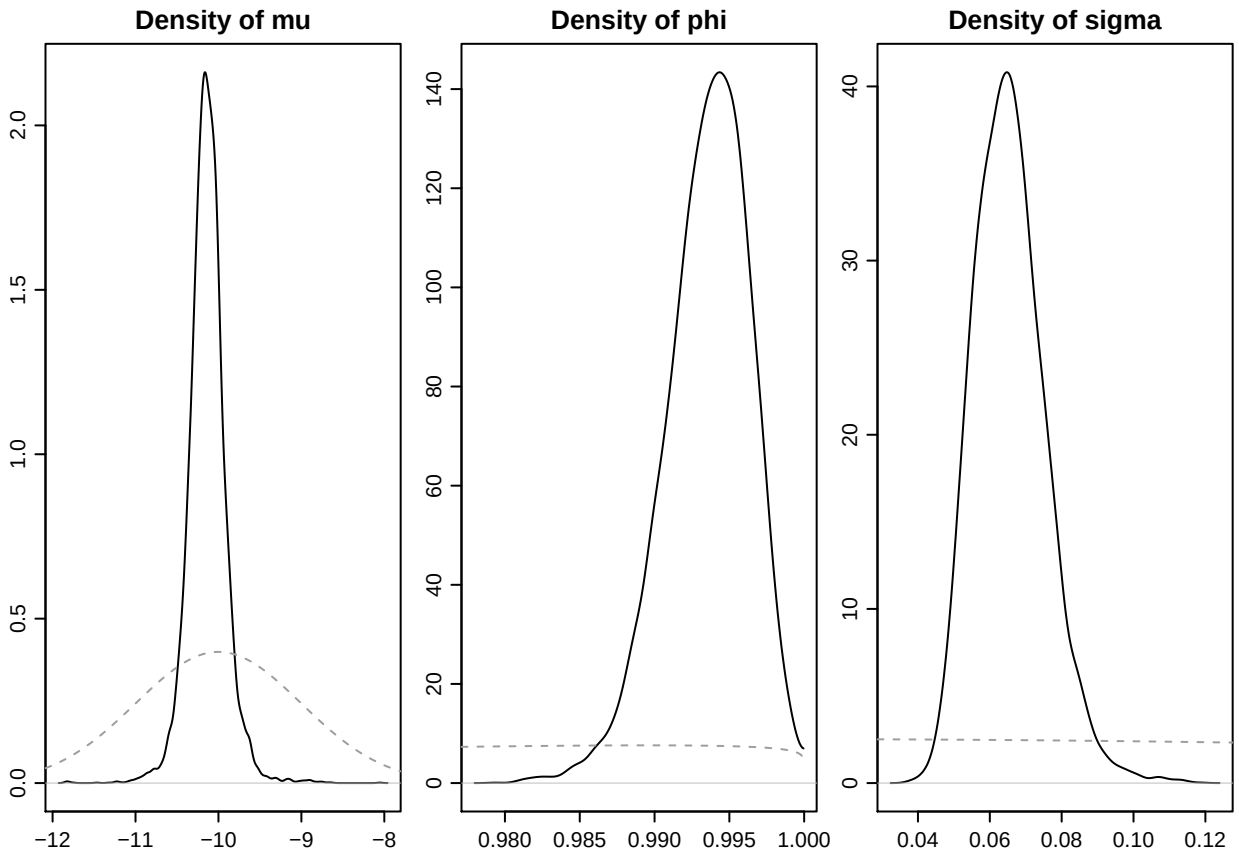
```
## sigma       0.0656 0.01001   0.0509   0.0649  0.0830  143
## exp(mu/2)   0.0063 0.00075   0.0053   0.0063  0.0075 4552
## sigma^2     0.0044 0.00139   0.0026   0.0042  0.0069  143
```

```
volplot(res, forecast = 100, dates = exrates$date[-1])
res <- updatesummary(res, quantiles = c(0.01, 0.1, 0.5, 0.9, 0.99))
volplot(res, forecast = 100, dates = exrates$date[-1])
```

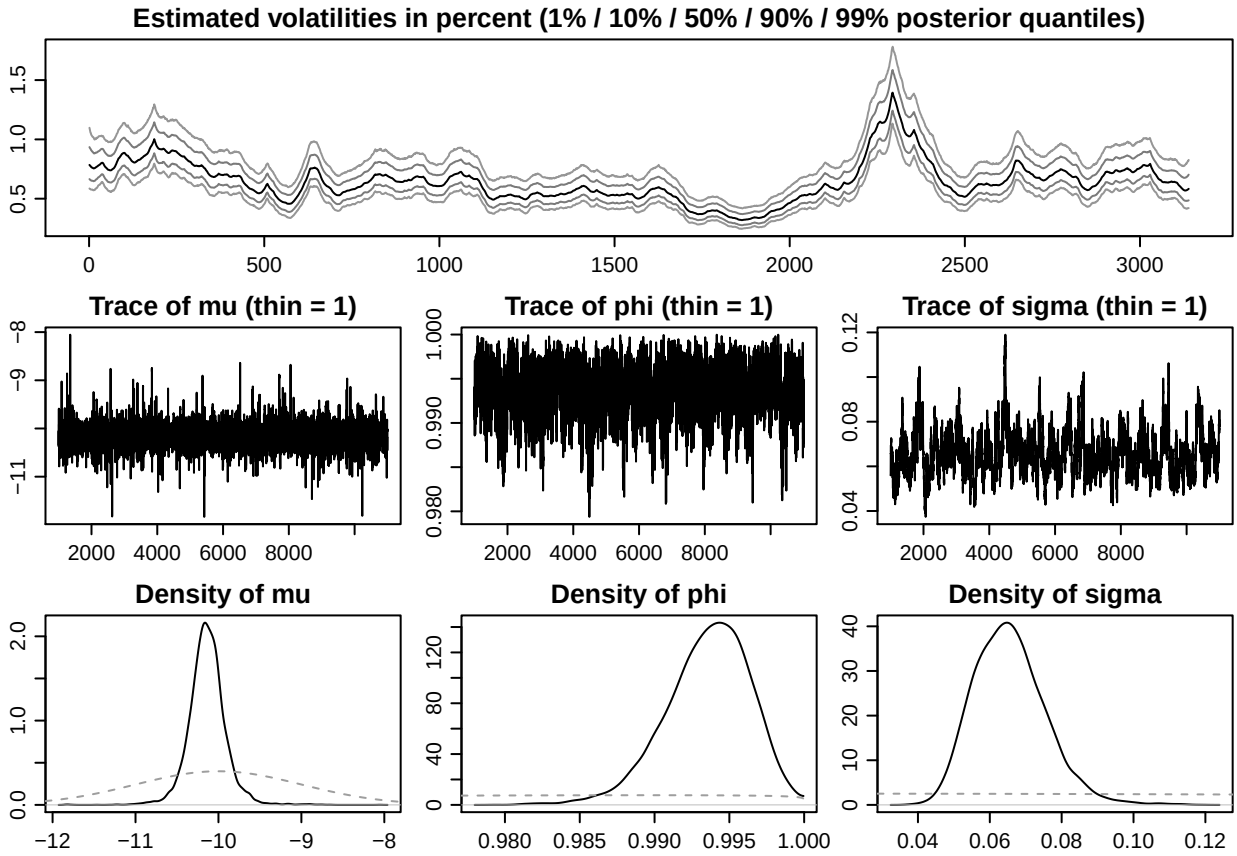




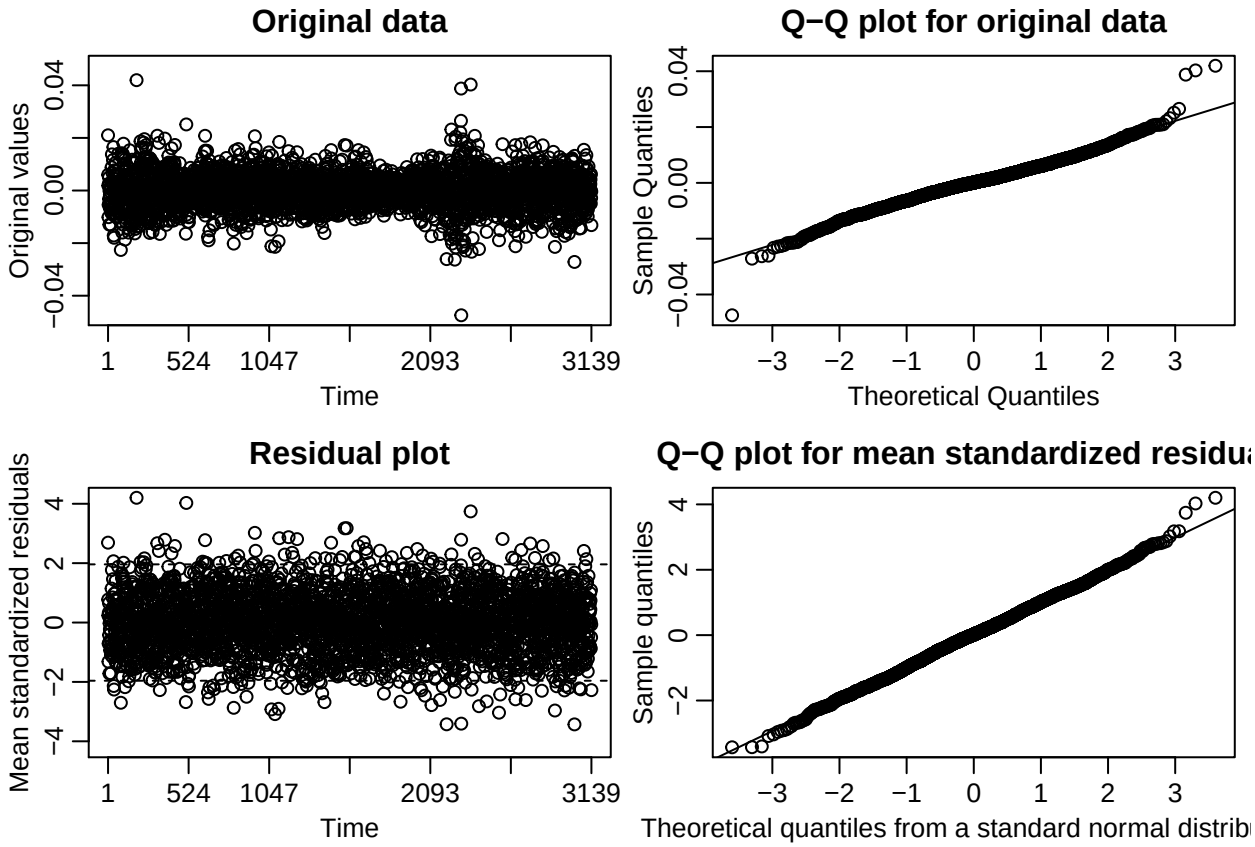

```
par(mfrow = c(1, 3))
paradensplot(res, showobs = FALSE)
```




```
plot(res, showobs = FALSE)
```




```
myresid <- resid(res)
plot(myresid, ret)
```



References

Bollerslev, Tim. 1986. "Generalized Autoregressive Conditional Heteroskedasticity." *Journal of Econometrics* 31 (3): 307–27.

Engle, Robert F. 1982. "Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation." *Econometrica: Journal of the Econometric Society*, 987–1007.

Harvey, Andrew, Esther Ruiz, and Neil Shephard. 1994. "Multivariate Stochastic Variance Models." *The Review of Economic Studies* 61 (2): 247–64.

Jacquier, Eric, Nicholas G Polson, and Peter E Rossi. 2002. "Bayesian Analysis of Stochastic Volatility Models." *Journal of Business & Economic Statistics* 20 (1): 69–87.

Melino, Angelo, and Stuart M Turnbull. 1990. "Pricing Foreign Currency Options with Stochastic Volatility." *Journal of Econometrics* 45 (1-2): 239–65.

Nelson, Daniel B. 1991. "Conditional Heteroskedasticity in Asset Returns: A New Approach." *Econometrica: Journal of the Econometric Society*, 347–70.