

Chapter6

Contents

Import data & Create xts	1
GARCH(1, 1)	1
Figure 6.1	1
Plot Figure 6.1 / Page 165	1
Column W	2
Figure 6.2	2
Plot Figure 6.2 / Page 166	2
Column X	3
Figure 6.3	3
Plot Figure 6.3 / Page 167	3
Figure 6.4	4
Create data for Figure 6.4	4
Column E	4
Column G	5
Column Y	5
Plot Figure 6.4	6
Implementing GARCH(1,1)	6
Results from xl	6
Estimated Parameters with tseries	6
Estimated Parameters with fGARCH	6
Estimated Parameters with rugarch	7

Import data & Create xts

```
rawdata = read.csv("S&P500.csv", sep = ";", dec = ".")

library(xts)
SP500 <- xts(rawdata[, -1], order.by = as.Date(rawdata[, 1], "%Y-%m-%d"))

SP500_logReturn <- diff(log(SP500))[-1]
```

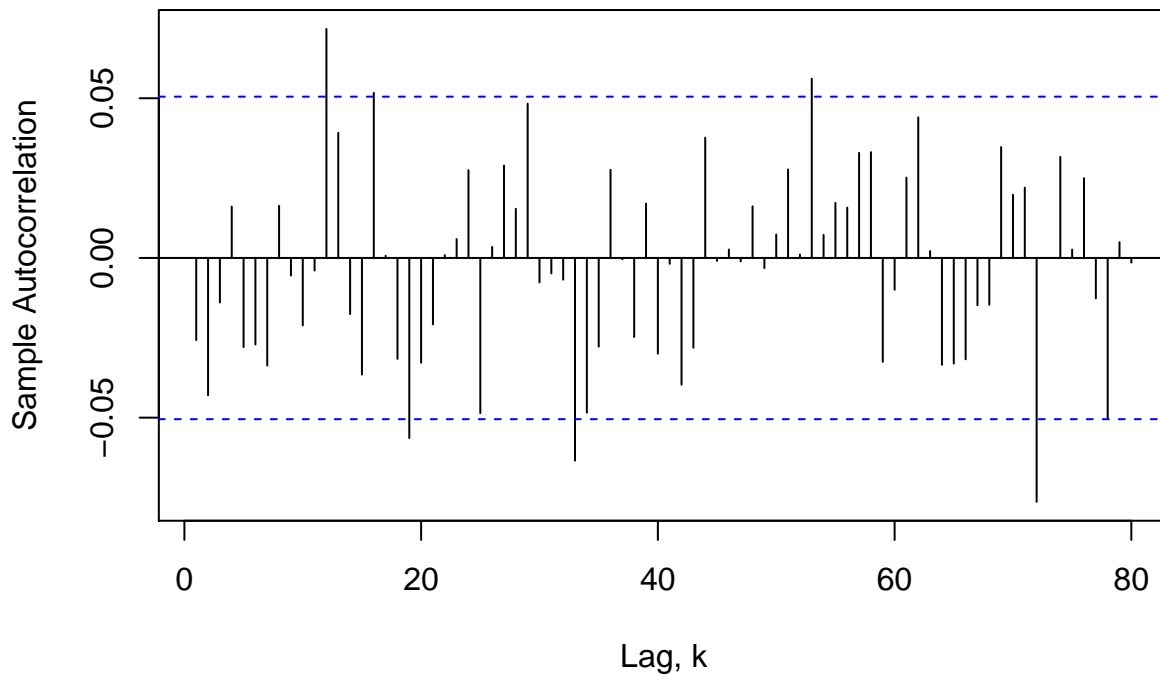
GARCH(1, 1)

Figure 6.1

Plot Figure 6.1 / Page 165

```
plot(acf(SP500_logReturn, lag.max = 80, plot = FALSE)[1:80], xlab = "Lag, k",
     ylab = "Sample Autocorrelation", main = "FIGURE 6.1")
```

FIGURE 6.1



Column W

```
acf(SP500_logReturn, lag.max = 80, plot = FALSE)[1:80]
```

```
##
## Autocorrelations of series 'SP500_logReturn', by lag
##
##      1      2      3      4      5      6      7      8      9     10
## -0.026 -0.043 -0.014  0.016 -0.028 -0.027 -0.034  0.016 -0.006 -0.021
##     11     12     13     14     15     16     17     18     19     20
## -0.004  0.072  0.039 -0.018 -0.037  0.052  0.001 -0.032 -0.056 -0.033
##     21     22     23     24     25     26     27     28     29     30
## -0.021  0.001  0.006  0.027 -0.049  0.003  0.029  0.015  0.048 -0.008
##     31     32     33     34     35     36     37     38     39     40
## -0.005 -0.007 -0.063 -0.048 -0.028  0.028  0.000 -0.025  0.017 -0.030
##     41     42     43     44     45     46     47     48     49     50
## -0.002 -0.040 -0.028  0.038 -0.001  0.003 -0.001  0.016 -0.003  0.007
##     51     52     53     54     55     56     57     58     59     60
##  0.028  0.001  0.056  0.007  0.017  0.016  0.033  0.033 -0.033 -0.010
##     61     62     63     64     65     66     67     68     69     70
##  0.025  0.044  0.002 -0.033 -0.033 -0.032 -0.015 -0.015  0.035  0.020
##     71     72     73     74     75     76     77     78     79     80
##  0.022 -0.076  0.000  0.032  0.003  0.025 -0.013 -0.050  0.005 -0.001
```

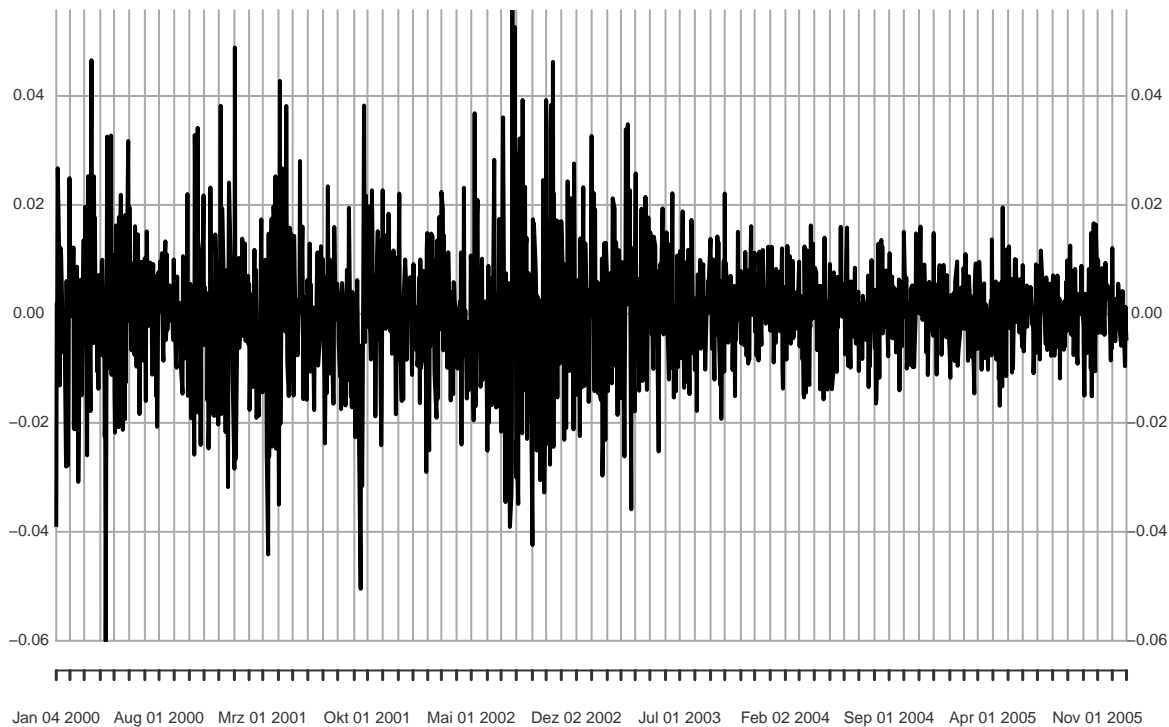
Figure 6.2

Plot Figure 6.2 / Page 166

```
plot(SP500_logReturn, xlab = "Date", ylab = "S&P500 Returns", type = "l", main = "FIGURE 6.2")
```

FIGURE 6.2

2000-01-04 / 2005-12-30



Column X

```
acf(SP500_logReturn^2, lag.max = 80, plot = FALSE)[1:80]
```

```
##
## Autocorrelations of series 'SP500_logReturn^2', by lag
##
##      1      2      3      4      5      6      7      8      9     10     11     12
## 0.163 0.225 0.266 0.148 0.187 0.174 0.165 0.181 0.111 0.178 0.152 0.157
##      13     14     15     16     17     18     19     20     21     22     23     24
## 0.146 0.107 0.099 0.135 0.087 0.172 0.136 0.119 0.134 0.096 0.070 0.073
##      25     26     27     28     29     30     31     32     33     34     35     36
## 0.116 0.115 0.088 0.135 0.081 0.123 0.092 0.103 0.117 0.042 0.076 0.054
##      37     38     39     40     41     42     43     44     45     46     47     48
## 0.081 0.063 0.084 0.110 0.080 0.046 0.102 0.079 0.113 0.144 0.120 0.121
##      49     50     51     52     53     54     55     56     57     58     59     60
## 0.082 0.124 0.137 0.100 0.128 0.109 0.140 0.149 0.060 0.149 0.095 0.063
##      61     62     63     64     65     66     67     68     69     70     71     72
## 0.073 0.068 0.016 0.091 0.072 0.078 0.022 0.096 0.072 0.040 0.083 0.108
##      73     74     75     76     77     78     79     80
## 0.054 0.047 0.038 0.018 0.049 0.038 0.021 0.027
```

Figure 6.3

Plot Figure 6.3 / Page 167

```
# acf(SP500_logReturn^2, lag.max=80, xlab='Lag, k', ylab='Autocorrelation',
# main='FIGURE 6.3')
plot(acf(SP500_logReturn^2, lag.max = 80, plot = FALSE)[1:80], xlab = "Lag, k",
     ylab = "Autocorrelation", main = "FIGURE 6.3")
```

FIGURE 6.3

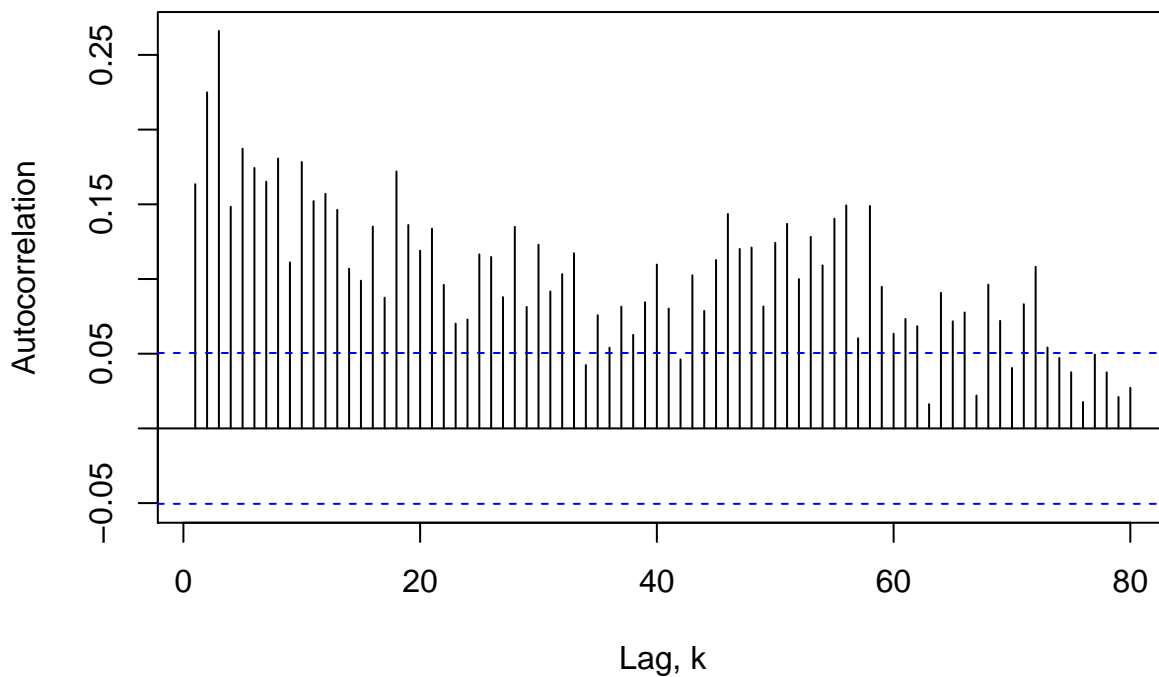


Figure 6.4

Create data for Figure 6.4

```
# Arbitrary Parameters
omega <- 2e-06
alpha <- 0.1
beta <- 0.85

SP500_logReturn_squared <- SP500_logReturn^2

GARCH_sigma_t_squared <- rep(-1, nrow(SP500_logReturn))
GARCH_sigma_t_squared[1] <- var(SP500_logReturn)

for (i in 2:length(GARCH_sigma_t_squared)) {
  GARCH_sigma_t_squared[i] <- omega + alpha * SP500_logReturn_squared[i -
    1, 1] + beta * GARCH_sigma_t_squared[i - 1]
}

# (Log Return)^2 / GARCH for sigma_t^2
StdReturn_squared <- SP500_logReturn_squared/GARCH_sigma_t_squared
```

Column E

Rows 3-9

```
rev(tail(GARCH_sigma_t_squared, n = 9))
```

```
## [1] 2.629451e-05 2.753354e-05 2.984174e-05 2.191419e-05 2.340713e-05
## [6] 2.309776e-05 2.407766e-05 2.596704e-05 2.416179e-05
```

Rows 1501-1509

```
GARCH_sigma_t_squared[c(9:1)]
```

```
## [1] 0.0001713072 0.0001819711 0.0002094572 0.0002237277 0.0002462880  
## [6] 0.0002033401 0.0002367634 0.0002757583 0.0001422157
```

Column G

Rows 3-9

```
rev(tail(StdReturn_squared, n = 9))
```

```
##  
## 2005-12-19 0.912633708  
## 2005-12-20 0.323606991  
## 2005-12-21 0.056316460  
## 2005-12-22 4.204893155  
## 2005-12-23 0.007743436  
## 2005-12-27 0.768055154  
## 2005-12-28 0.262381621  
## 2005-12-29 0.002183924  
## 2005-12-30 1.419396654
```

Rows 1501-1509

```
rev(StdReturn_squared[c(1:9)])
```

```
##  
## 2000-01-04 0.657728439  
## 2000-01-05 0.804074269  
## 2000-01-06 0.092260533  
## 2000-01-07 0.772754585  
## 2000-01-10 0.502778817  
## 2000-01-11 3.513765429  
## 2000-01-12 0.003854038  
## 2000-01-13 0.013373517  
## 2000-01-14 10.749510900
```

Column Y

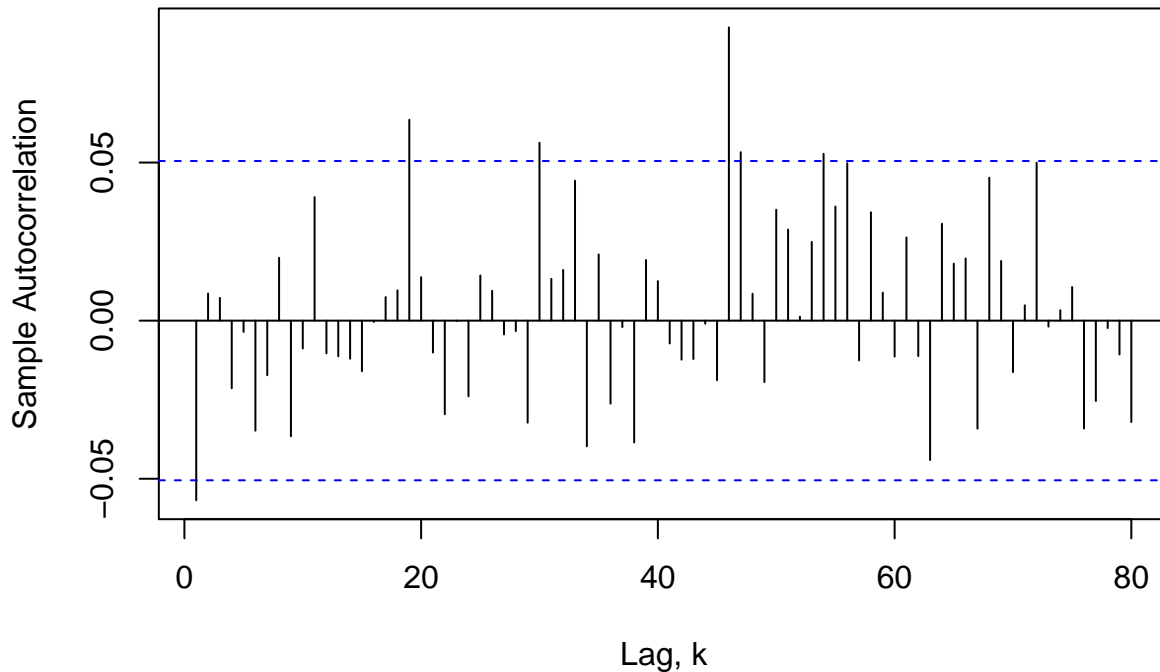
```
acf(StdReturn_squared, type = "partial", lag.max = 80, plot = FALSE)[1:80]
```

```
##  
## Partial autocorrelations of series 'StdReturn_squared', by lag  
##  
##      1      2      3      4      5      6      7      8      9     10  
## -0.057 0.009 0.007 -0.021 -0.004 -0.035 -0.017 0.020 -0.037 -0.009  
##     11     12     13     14     15     16     17     18     19     20  
## 0.039 -0.010 -0.011 -0.012 -0.016 0.000 0.007 0.010 0.064 0.014  
##     21     22     23     24     25     26     27     28     29     30  
## -0.010 -0.030 0.000 -0.024 0.014 0.009 -0.004 -0.003 -0.032 0.056  
##     31     32     33     34     35     36     37     38     39     40  
## 0.013 0.016 0.044 -0.040 0.021 -0.026 -0.002 -0.039 0.019 0.013  
##     41     42     43     44     45     46     47     48     49     50  
## -0.007 -0.012 -0.012 -0.001 -0.019 0.093 0.053 0.009 -0.019 0.035  
##     51     52     53     54     55     56     57     58     59     60  
## 0.029 0.001 0.025 0.053 0.036 0.050 -0.013 0.034 0.009 -0.011  
##     61     62     63     64     65     66     67     68     69     70  
## 0.026 -0.011 -0.044 0.031 0.018 0.020 -0.034 0.045 0.019 -0.016  
##     71     72     73     74     75     76     77     78     79     80  
## 0.005 0.050 -0.002 0.003 0.011 -0.034 -0.025 -0.002 -0.011 -0.032
```

Plot Figure 6.4

```
plot(acf(StdReturn_squared, type = "partial", lag.max = 80, plot = FALSE)[1:80],
     main = "FIGURE 6.4", xlab = "Lag, k", ylab = "Sample Autocorrelation")
```

FIGURE 6.4



Implementing GARCH(1,1)

Results from x1

Parameter	Value
ω	0.000000720549336714
α	0.074563989113391000
β	0.921009670203477000

Estimated Parameters with tseries

```
library(tseries)
out <- garch(SP500_logReturn, order = c(1, 1), trace = FALSE)
options(scipen = 999)
format(out$coef, digits = 9, nsmall = 1)
```

```
##              a0              a1              b1
## "0.000000720454738" "0.074563780160594" "0.921011037334454"
```

Estimated Parameters with fGARCH

```
library(fGarch)

fGarchmodel <- garchFit(formula = ~garch(1, 1), data = SP500_logReturn, trace = FALSE)
# print(fGarchmodel)
```

```
print(coef(fGarchmodel))
```

```
##              mu              omega              alpha1              beta1
## 0.0002138699512 0.0000007207968 0.0747184031615 0.9208875485281
```

Estimated Parameters with rugarch

```
library(rugarch)
# Specify a model using 'rugarch'
spec = ugarchspec(variance.model = list(model = "fGARCH", garchOrder = c(1,
  1), submodel = "GARCH", external.regressors = NULL, variance.targeting = FALSE),
  mean.model = list(armaOrder = c(0, 0), external.regressors = NULL), distribution.model = "norm",
  start.pars = list(), fixed.pars = list())

rugarchmodel = ugarchfit(spec = spec, data = SP500_logReturn, solver = "hybrid")

# Compare model coefficients
print(coef(rugarchmodel))
```

```
##              mu              omega              alpha1              beta1
## 0.0002179530269 0.0000007144172 0.0744863779527 0.9210657397310
```