CUHK RMSC4002 Tutorial 4

Benjamin Chan

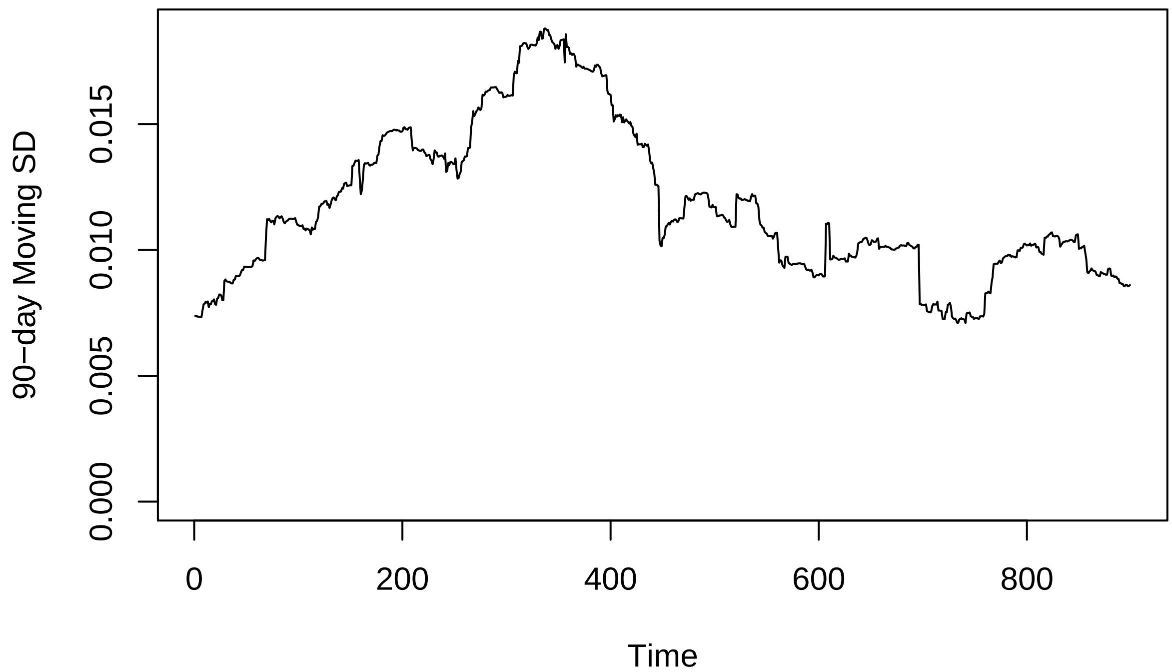
October 16, 2018

Read in and Manipulate Data

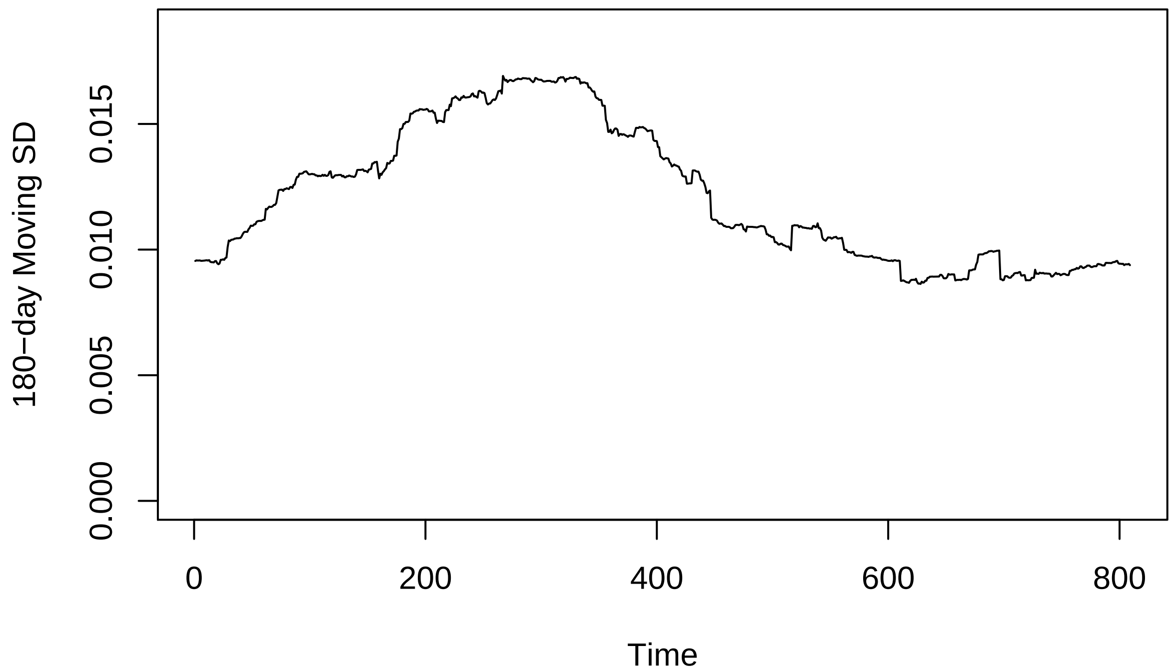
```
# Read in data (a CSV file) under Dataset
d <- read.csv("./../Dataset/stock_2018.csv")

# as.ts: coerce an object to a time-series
t1 <- as.ts(d$HSBC)                 # For stock HSBC (0005)
t2 <- as.ts(d$CLP)                  # For stock CLP (0002)
t3 <- as.ts(d$CK)                   # For stock Cheung Kong (0001)

# Compute daily percentage return
u1 <- (lag(t1)-t1)/t1               # lag: compute a lagged version of a time series
u2 <- (lag(t2)-t2)/t2
u3 <- (lag(t3)-t3)/t3
```

Moving Standard Deviation

```
msd <- function(t, w) {             # Function to compute moving s.d.
   n <- length(t)-w+1
   out <- c()                       # Initialize an output vector

   for (i in 1:n) {
      j <- i+w-1
      s <- sd (window(t, i, j))     # Compute the sd of t(i) to t(j)
      out <- append(out, s)         # Append to out
   }

   out <- as.ts(out)                # Coerce to a time-series object
}

s1_90 <- msd(u1, 90)                # Compute 90-day moving sd of u1
s1_180 <- msd(u1, 180)              # Compute 180-day moving sd of u1

par(mfrow = c(1, 1))
plot(s1_90, ylim = c(0, max(s1_90, s1_180)), ylab = "90-day Moving SD")
```




```r
plot(s1_180, ylim = c(0, max(s1_90, s1_180)), ylab = "180-day Moving SD")
```



Note that the volatility varies with time.

The minimum and maximum of `s1_90` is 0.0071 and 0.0188. Hence the minimum and maximum annual volatility is 11.27% and 29.85%. Similarly, the minimum and maximum of `s1_180` is 0.0086 and 0.0169. Hence the minimum and maximum annual volatility is 13.7% and 26.85%.

Generalized Autoregressive Conditional Heteroskedasticity (GARCH) Model

Maximum Likelihood Estimation

```
library(tseries)
res <- garch(u1, order = c(1, 1))      # Fit GARCH(1,1) model
```

```
names(res)                             # names: get or set the names of an object
```

```
 [1] "order"          "coef"           "n.likeli"       "n.used"
 [5] "residuals"      "fitted.values"  "series"         "frequency"
 [9] "call"           "vcov"
```

```
round(res$coef, 6)                     # Display the coefficient using 6 digits
```

```
      a0       a1       b1
0.000011 0.082403 0.837725
```

```
# n.likeli: the negative log-likelihood function evaluated
# at the coefficient estimates (apart from some constant)
-2*res$n.likeli
```

```
[1] 7853.381
```

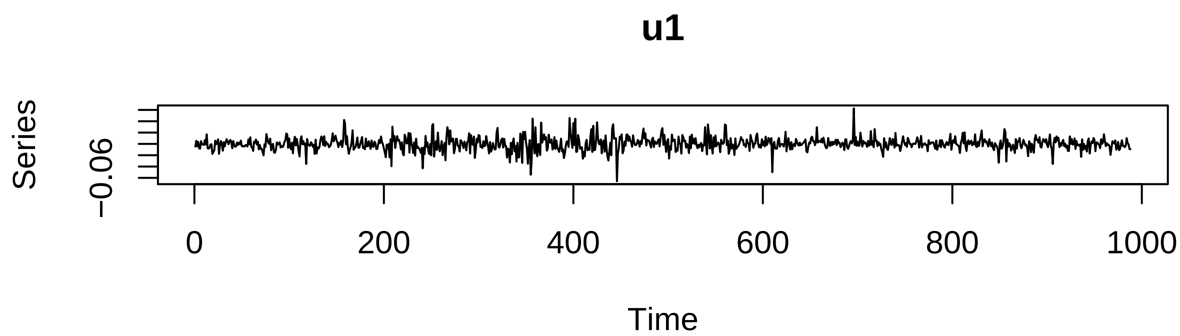
```r
summary(res)
```

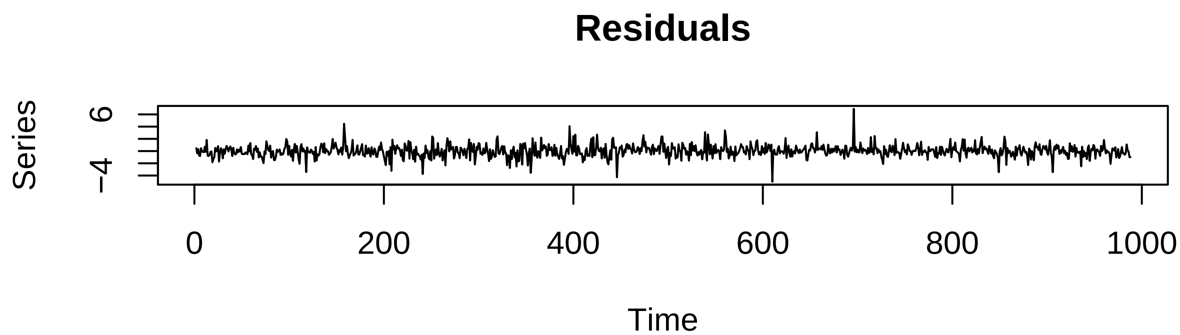
```
Call:
garch(x = u1, order = c(1, 1))

Model:
GARCH(1,1)

Residuals:
    Min      1Q  Median      3Q     Max
-5.0160 -0.5337  0.0000  0.5162  6.9064

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 1.118e-05   2.780e-06    4.020 5.81e-05 ***
a1 8.240e-02   1.853e-02    4.446 8.73e-06 ***
b1 8.377e-01   3.571e-02   23.459  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Diagnostic Tests:
    Jarque Bera Test

data:  Residuals
X-squared = 788.5, df = 2, p-value < 2.2e-16


    Box-Ljung test

data:  Squared.Residuals
X-squared = 0.12839, df = 1, p-value = 0.7201
```

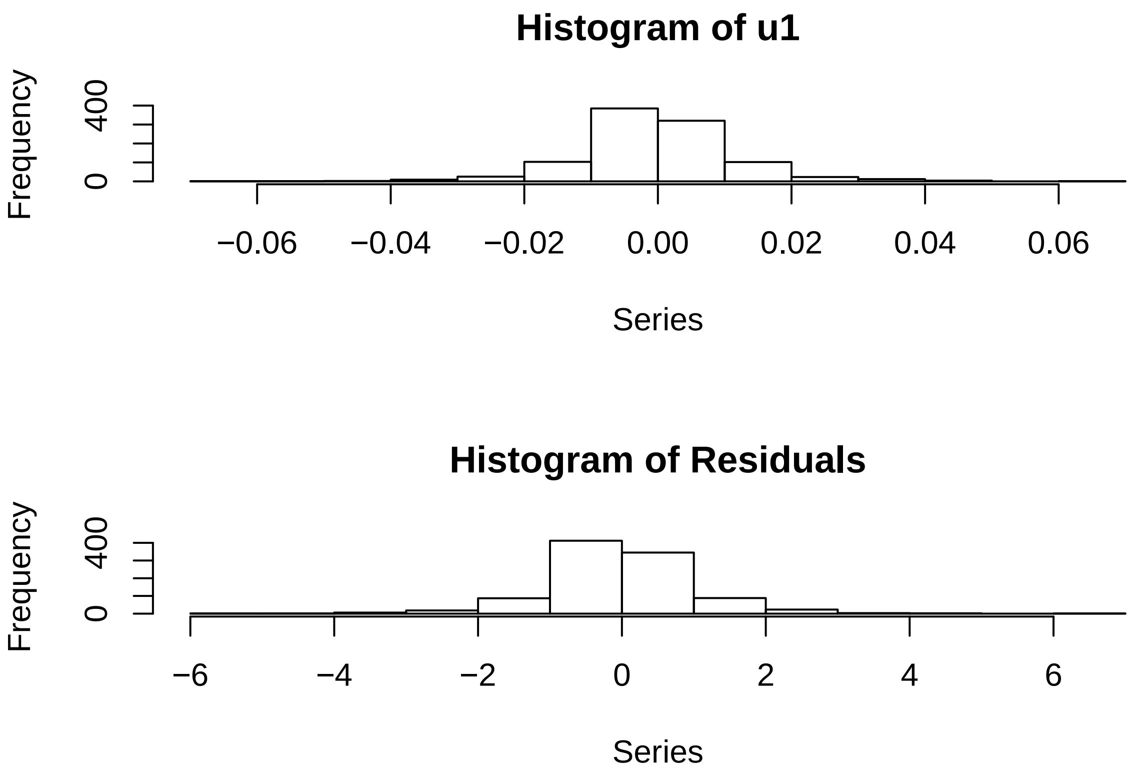
All the p-values are small and hence the coefficients are significantly different from zero.

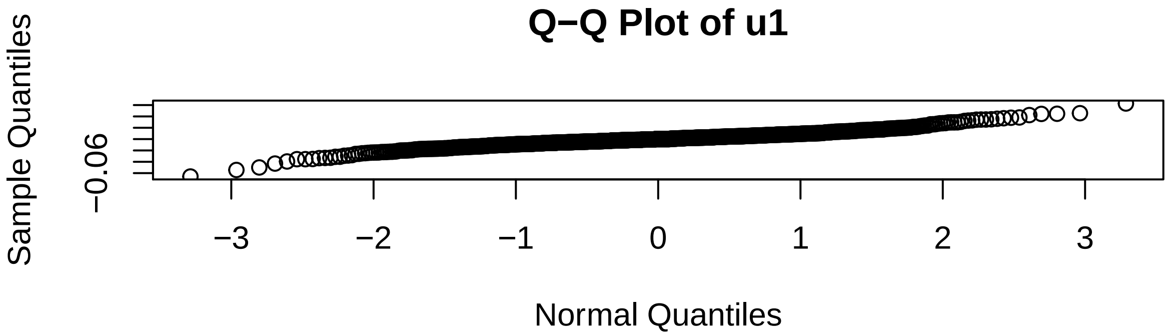
Model Diagnostic

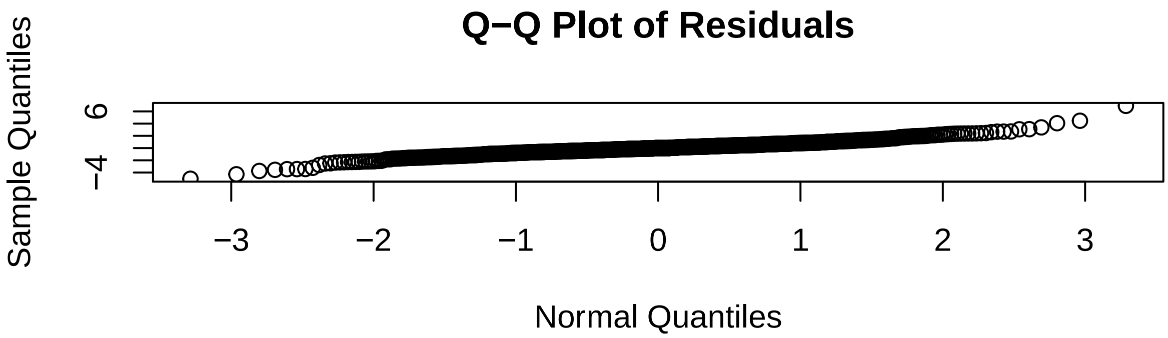
```r
plot(res)
```

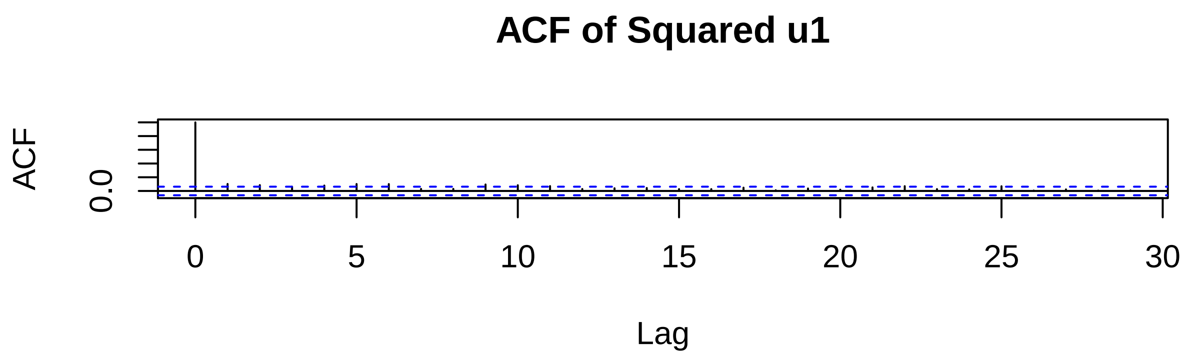

u1
Series
−0.06
0
200
400
600
800
1000
Time

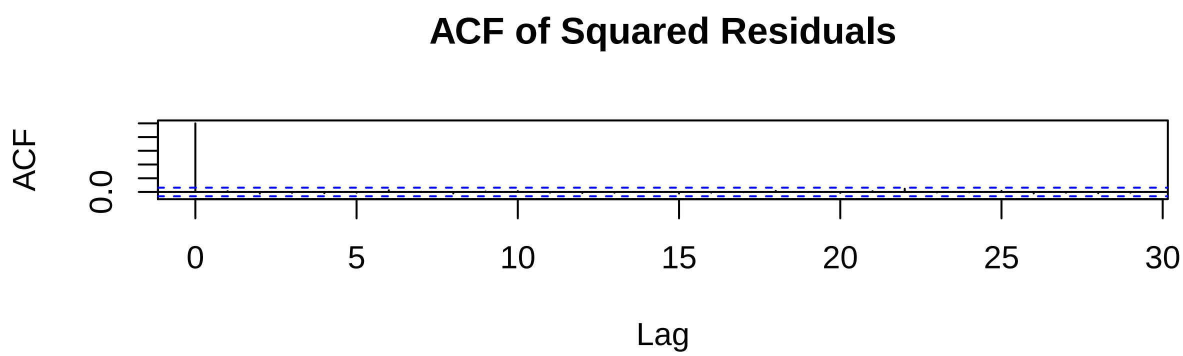

Residuals
Series
6
−4
0
200
400
600
800
1000
Time


Histogram of u1
Frequency
0
400
−0.06
−0.04
−0.02
0.00
0.02
0.04
0.06
Series
Histogram of Residuals
Frequency
0
400
−6
−4
−2
0
2
4
6
Series


Q−Q Plot of u1
Sample Quantiles
−0.06
−3
−2
−1
0
1
2
3
Normal Quantiles


Q−Q Plot of Residuals
Sample Quantiles
−4
6
−3
−2
−1
0
1
2
3
Normal Quantiles





From the plots, `u1` and the residuals are approximately normally distributed

```
# Box.test: compute the Box-Pierce or Ljung-Box test statistic
# for examining null hypothesis of independence in a time series
Box.test(u1^2, lag = 15, type = "Ljung")
```

```
    Box-Ljung test

data:  u1^2
X-squared = 76.023, df = 15, p-value = 3.698e-10
```

```
Box.test(res$resid^2, lag = 15, type = "Ljung")
```

```
    Box-Ljung test

data:  res$resid^2
X-squared = 3.4925, df = 15, p-value = 0.999
```

The p-value of Box-Ljung test statistic for u_i^2 is small, meaning that there exists autocorrelation in u_i^2. On the other hand, the p-value for u_i^2/σ_i^2 is large, meaning that there is no autocorrelation in u_i^2/σ_i^2. It indicates that the autocorrelation of u_i^2 is removed by GARCH(1,1) model. Hence σ_i^2 is a good estimate of the variance rate.

Plot the Fitted Values

```r
library(plotly)                                    # Create Interactive Web Graphics via 'plotly.js'
library(tidyr)                                     # Easily Tidy Data with 'spread()' and 'gather()' Functi
library(dplyr)                                     # A Grammar of Data Manipulation

t90 <- as.ts(c(rep(NA, 45), s1_90))                # Add 45 NA's in front of s1_90
t180 <- as.ts(c(rep(NA, 90), s1_180))              # Add 90 NA's in front of s1_180
s <- cbind(res$fitted.values[, 1], t90, t180)

# To be appeared in plot
colnames(s) <- c("GARCH", "90-day Moving SD", "180-day Moving SD")

# %>%: pipe operator
newseries <- as.data.frame(s) %>%
    gather(type, value) %>%
    mutate(time = rep(time(s), 3))

# plot_ly(x = newseries$time, y = newseries$value, color = newseries$type, mode = 'lines')
# plot_ly is not supported by LaTeX. Please refer to HTML version for the plot.
```

There are some clustered and spiky patterns in the estimated volatilities from GARCH(1,1) while the plot of the estimated volatilities using the moving s.d. tends to smooth out these spiky patterns.