Ch 13: ARCH/GARCH in R

Overview

• Subject is a key part of research in finance and accounting

• There is a cottage industry on time series models in financial econometrics

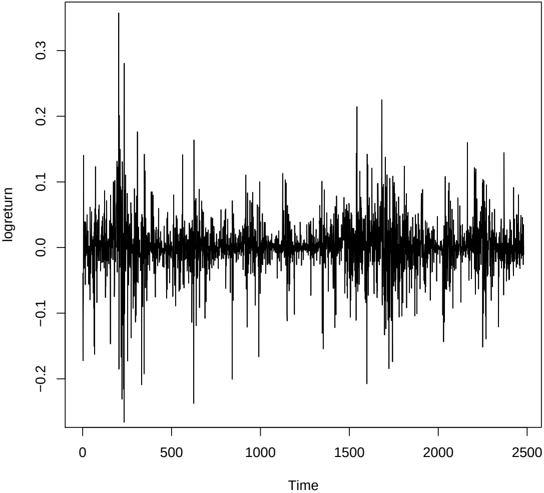
• **Deeper analyses are possible using key concepts from mathematical finance and from statistical physics**

- ARCH/GARCH models mark the starting point of the professional analysis of financial time series
- Lots of generalisations – another cottage industry – e.g. E-GARCH, M-GARCH, T-GARCH etc
- **Econometric analysis of cryptocurrency data a live topic of academic research – see e.g. Katsiampa (2017)**

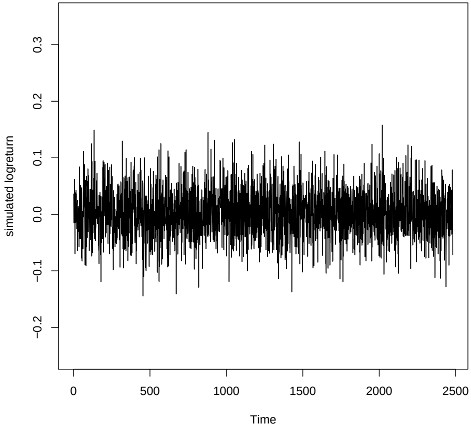
Motivation

- In the last lecture we saw that typically in financial time series the volatility is not constant
- Market returns tend to cluster into periods of extreme volatility and more tranquil periods
- Market returns also show marked differences from simulated returns from mathematical models
- **All markets, especially cryptocurrency markets, can show very volatile behaviour**
- **This volatility will ultimately be more extreme than any mathematical/statistical model used to describe it**

Example volatility: Time series plot of Bitcoin returns



Real data more volatile than simulated data: Time series plot of simulated Bitcoin returns based on normal random walk model



R-code for Figure 1

- **Same commands as before to read the data in**

```
>BitcoinData<-read.table("E:BitcoinData.txt")
> ncol(BitcoinData)
4
> price<-BitcoinData[,4]
> price<-rev(price)
> length(price)
2482
> logreturn<-log(price[-1])-log(price[-2482])
```

- **Plot the log-returns and choose a sensible *y*-axis scale. (Need to do the first plot in order to see what a sensible scale would be.)**

```
ts.plot(logreturn)
ts.plot(logreturn, ylim=c(-0.25, 0.35))
```

R code for Figure II

- **Simulate log-returns from the normal random walk model and use the same y-axis to show how different the two series look**

```
>mean(logreturn)
0.001716242
> sd(logreturn)
0.04244594
ts.plot(rnorm(2481, 0.001716242, 0.04244594),
ylim=c(-0.25, 0.35), ylab=‘‘simulated logreturn")
```

The purpose of ARCH/GARCH

- ARCH/GARCH solve the following problem
- Mathematical models such as Black-Scholes often assume that volatility is constant
- In reallity volatility is not constant and market returns can typically be clustered into periods of high-volatility and more tranquil low-volatility periods
- This basic problem has various different names
  - e.g. volatility clustering
  - e.g. long-range dependence in volatility

Outline of the lecture

1. Mathematical formulation of the model
   - What precisely do ARCH and GARCH do?
2. ARCH/GARCH in R
3. Example 1: Ethereum data
4. Example 2: Ripple data
5. Example 3: Bitcoin data
6. References

1.1 Basic Math: Comparing Black-Scholes to ARCH(1) (Gujarati and Porter, Ch. 22)

- Under the Black-Scholes model the first differences of the log-price ΔX_t satisfy

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \sigma^2 = \text{Constant} \end{aligned}$$

- **However this model is at odds with real stock market data**
- The ARCH(1) model is a partial fix to this problem

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \alpha_0 + \alpha_1 u_{t-1}^2 \\ u_t &= (\Delta X_t - \mu) \text{ the residual} \end{aligned}$$

- **Allows for high values of volatility to follow each other**
- **Allows for low values of volatility to follow each other**

1.2 The full ARCH model

- Under the Black-Scholes model the first differences of the log-price ΔX_t satisfy

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \sigma^2 = \text{Constant} \end{aligned}$$

- **To address volatility clustering the ARCH(p) model is**

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \alpha_0 + \alpha_1 u_{t-1}^2 + \ldots + \alpha_p u_{t-p}^2 \\ u_t &= (\Delta X_t - \mu) \text{ the residual} \end{aligned}$$

- **Allows for high values of volatility to follow each other**
- **Allows for low values of volatility to follow each other**

1.3 ARCH and GARCH

• ARCH and GARCH models spawned a cottage industry in financial econometrics e.g. E-GARCH, M-GARCH, S-GARCH etc

• ARCH=Autoregressive Conditional Heteroskedasticity

• GARCH=Generalised Autoregressive Conditional Heteroskedasticity

• **As the name suggests GARCH is a generalisation of ARCH that allows for slightly richer behaviour in volatility**

1.4 Idiot's guide to time series

- There is a whole statistical subject of time series that is inherently specialist
- Financial time series is a little less specialist but is a subject in its own right and very different from the regular ARMA time series models commonly studied in mainstream statistics
- **We cannot hope to be able to do justice to either subject...**

$$\begin{aligned} \text{ARCH} &= \text{Autoregressive Conditional Heteroskedasticity} \\ \text{GARCH} &= \text{Generalised Autoregressive Conditional Heteroskedasticity} \\ \text{ARCH} &= \text{Autoregressive model for volatility} \\ \text{GARCH} &= \text{Autoregressive Moving Average model for volatility} \end{aligned}$$

1.5 The GARCH(1, 1) model

• Under the Black-Scholes model the first differences of the log-price ΔX_t satisfy

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \sigma^2 = \text{Constant} \end{aligned}$$

• **However this model is at odds with real stock market data**

• The GARCH(1, 1) model is a slightly more advanced partial fix to this problem

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \alpha_0 + \alpha_1 u_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \end{aligned}$$

• Model is more complicated as it involves an unobserved volatility component but the basic idea remains the same.

• **Allows for high values of volatility to follow each other**

• **Allows for low values of volatility to follow each other**

1.6 The GARCH(p, q) model

- Under the Black-Scholes model the first differences of the log-price ΔX_t satisfy

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \sigma^2 = \text{Constant} \end{aligned}$$

- **However this model is at odds with real stock market data**
- The ARCH(1) model is a partial fix to this problem

$$\begin{aligned} \Delta X_t &\sim N\left(\mu, \sigma_t^2\right) \\ \sigma_t^2 &= \alpha_0 + \alpha_1 u_{t-1}^2 + \ldots + \alpha_p u_{t-p}^2 + \beta_1 \sigma_{t-1}^2 + \ldots + \beta_q \sigma_{t-q}^2 \\ u_t &= (\Delta X_t - \mu) \text{ the residual} \\ \sigma_t^2 &= \text{unobserved volatility component} \end{aligned}$$

- **Allows for high values of volatility to follow each other**
- **Allows for low values of volatility to follow each other**

1.7 Choosing the order of an ARCH/GARCH model

- Choosing the order of an ARCH/GARCH model is difficult and there are several different approaches

1. ARCH(1)/GARCH$(1, 1)$
   - Often easiest to assume that both ARCH and GARCH effects are present and just use a first-order model
2. Choose the order to coincide with certain time periods
3. Use t-tests to determine the order
   - Similar to t-tests in regression
   - See the worked example
4. Choose the model with the lowest Schwarz criterion
   - Similar to advanced approaches in computer science/Bayesian statistics
   - Chooses the model with the highest probability of being correct

1.8 Choosing the order of the ARCH/GARCH model to coincide with certain time intervals – a non-exhaustive list

1. Daily data
   - Order 1: Volatility from yesterday affects today
   - Order 5: Volatility from last week affects today
   - Order 10: Volatility from last 2 weeks affects today
2. Weekly data
   - Order 1: Volatility from last week affects this week
   - Order 4: Volatility from last month affects this week
3. Monthly data
   - Order 3: Volatility from last quarter affects this month
   - Order 12: Volatility from last year affects this quarter
4. Quarterly data
   - Order 4: volatility from last year affects this quarter

1.9 ARCH/GARCH model selection for daily cryptocurrency data

- Two main possibilities
  1. GARCH(1, 1)
     - Assumption is that a low order model (and indeed the lowest possible order model) may be sufficient to give a good (albeit imperfect) description of volatility correlations on real markets.
  2. GARCH(7, 7)
     - Since currency and cryptocurrency markets trade 7 days a week the interpretation of this model would be that volatility from the previous week feeds forward and affects today

2.1 ARCH/GARCH modelling in R

- Easiest way to fit ARCH/GARCH models in R is using the `tseries` package
- This fits a conditionally normally distributed model to the mean-corrected log-returns series
- More advanced ARCH/GARCH models may be possible using other packages e.g. `fGARCH`, `RUgarch`
- Fancier ARCH/GARCH models may be required for cryptocurrency data on occasion to account for
  - autocorrelation in the returns series (see e.g. Katsiampa, 2017)
  - heavier tails than the conditional normal distributions in the GARCH model (e.g. Student t-distribution)

2.2 ARCH/GARCH modelling using tseries in R

- The basic command is `garch` even if you fit a purely ARCH model without any GARCH terms
- The basic command is

`garch(x, order=(`q`, `p`)`

- There are two quirky features of this

1. x is a demeaned log-returns series
2. Mathematically tend to write GARCH models as GARCH(p, q) where p is the ARCH bit and q is the GARCH bit. In R in `tseries` this ordering is revered.

2.3 Using `tseries` to fit ARCH and GARCH models in R

- Basic way of working shows some similarities with how R works in relation to regression and extended regression models
  1. Fit the model using the command `garch` and save this as something
  2. Then use the command `summary` to display the required results e.g. t-statistics
- Can formally test for the presence of an ARCH and GARCH effect
  1. If any of a_1, a_2, ..., a_p are significant then we have evidence of the ARCH effect
  2. If any of b_1, b_2, ..., b_q are significant then we have evidence of the GARCH effect

2.4 Summary

• t-statistics for the ARCH and GARCH effect follow on from other topics discussed in the module
• However, there is a notable difference in that given the ubiquity of ARCH/GARCH effects in financial time series these ARCH and GARCH terms are more likely than not to show significant effects provided that the order of the model chosen is appropriate
• Some example questions can be found in the mock exam on the Canvas module page
• Some additional worked-examples can be found below in the remainder of the lecture

3.1 Ethereum example

- Price data on Ethereum from August 7th 2015-Feb 13th 2020
- Want to do the following tasks in R
    1. Read in the data
    2. Load the package `tseries` if you haven't done this already
    3. Fit low order ARCH(1) and GARCH(1, 1) models
    4. Experiment with the fitting of a higher order GARCH(7, 7) model that would have the interpretation of volatility from the previous week feeds forward and affects today

3.2 Loading in the Ethereum data

- Following previous examples

```
> EthereumData<-read.table(``E:EthereumData.txt")
> ncol(EthereumData)
4
> price<-EthereumData[,4]
> price<-rev(price)
> length(price)
1652
> logreturn<-log(price[-1])-log(price[-1652])
```

3.3 Model fitting in tseries in R

- Need to apply models to the mean-corrected log-returns

```
resid<-logreturn-mean(logreturn)
```

- Fit ARCH(1) and GARCH(1, 1) models to this data remembering that the R syntax works the other way round to standard econometric notation

```
arch1<-garch(resid, order=c(0, 1))
garch11<-garch(resid, order=c(1, 1))
summary(arch1)
summary(garch11)
```

3.4 Testing for the ARCH effect

- **Using the command** `summary` **in R gives conclusive evidence of the ARCH effect**

```
 Estimate Std.  Error t value Pr(>|t|)
a0 2.759e-03 5.856e-05 47.11 <2e-16 ***
a1 3.487e-01 2.960e-02 11.78 <2e-16 ***
```

- **Hand calculation for an exam-type question**

$$t = |\text{estimate}|/\text{e.s.e} = 3.487 \times 10^{-1} / 2.960 \times 10^{-2}$$

$$t = 11.78041 > 2, \ \ p < 0.05.$$

3.5 Testing for the GARCH effect

- **Using the command** `summary` **in R gives conclusive evidence of the both the ARCH and the GARCH effect**

```
Estimate Std.  Error t value Pr(>|t|)
a0 0.0003819 0.0000383 9.971 <2e-16 ***
a1 0.1949723 0.0181241 10.758 <2e-16 ***
b1 0.7070688 0.0216811 32.612 <2e-16 ***
```

- **Hand calculation for ARCH effect**

$$t = |\text{estimate}|/\text{e.s.e} = 0.1949723/0.0181241$$

$$t = 10.75763 > 2, \;\; p < 0.05.$$

- **Hand calculation for GARCH effect**

$$t = |\text{estimate}|/\text{e.s.e} = 0.7070688/0.0216811$$

$$t = 32.61222 > 2, \;\; p < 0.05.$$

3.6 Experimenting with higher order models

- Fit a GARCH(7, 7) model

```
garch77<-garch(resid, order=c(7, 7))
summary(garch(77)
```

- Some evidence of an ARCH effect. Though non-significant terms suggest that there is some redundancy in this model and really that a lower-order model is appropriate

3.7 Results for the GARCH(7, 7) model

```
Estimate Std.  Error t value Pr(>|t|)
a0 9.852e-04 9.396e-04 1.049 0.294
a1 1.324e-01 1.928e-02 6.867 6.57e-12 ***
a2 8.610e-02 1.489e-01 0.578 0.563
a3 5.899e-02 1.133e-01 0.521 0.602
a4 3.406e-02 7.287e-02 0.467 0.640
a5 4.179e-02 6.539e-02 0.639 0.523
a6 4.043e-02 5.282e-02 0.765 0.444
a7 1.596e-15 3.082e-02 0.000 1.000
b1 4.686e-02 1.125e+00 0.042 0.967
b2 4.258e-02 6.122e-01 0.070 0.945
b3 3.804e-02 4.072e-01 0.093 0.926
b4 3.961e-02 4.325e-01 0.092 0.927
b5 4.095e-02 4.196e-01 0.098 0.922
b6 4.109e-02 4.377e-01 0.094 0.925
b7 4.560e-02 2.418e-01 0.189 0.850
```

4.1 Ripple example

- Price data on Ripple from August 4th 2013-Feb 13th 2020
- Want to do the following tasks in R

1. Read in the data
2. Load the package `tseries` if you haven't done this already
3. Fit low order ARCH(1) and GARCH(1, 1) models
4. Experiment with the fitting of a higher order GARCH(7, 7) model that would have the interpretation of volatility from the previous week feeds forward and affects today

4.2 Loading in the Ripple data

- Following previous examples

```
>RippleData<-read.table(``E:RippleData.txt")
> ncol(RippleData)
4
> price<-RippleData[,4]
> price<-rev(price)
> length(price)
2385
> logreturn<-log(price[-1])-log(price[-2385])
```

4.3 Model fitting in `tseries` in R

- Need to apply models to the mean-corrected log-returns

```
resid<-logreturn-mean(logreturn)
```

- Fit ARCH(1) and GARCH(1, 1) models to this data remembering that the R syntax works the other way round to standard econometric notation

```
arch1<-garch(resid, order=c(0, 1))
garch11<-garch(resid, order=c(1, 1))
summary(arch1)
summary(garch11)
```

4.4 Testing for the ARCH effect

- **Using the command `summary` in R gives conclusive evidence of the ARCH effect**

```
 Estimate Std.  Error t value Pr(>|t|)
a0 2.486e-03 2.726e-05 91.17 <2e-16 ***
a1 9.300e-01 2.188e-02 42.51 <2e-16 ***
```

- **Hand calculation for an exam-type question**

$$t = |\text{estimate}|/\text{e.s.e} = 9.300\times 10^{-1}/2.188\times 10^{-2}$$

$$t = 42.50457 > 2, \;\; p < 0.05.$$

4.5 Testing for the GARCH effect

- **Using the command `summary` in R gives conclusive evidence of the both the ARCH and the GARCH effect**

```
Estimate Std.  Error t value Pr(>|t|)
a0 3.634e-04 1.453e-05 25.01 <2e-16 ***
a1 3.363e-01 1.652e-02 20.36 <2e-16 ***
b1 6.413e-01 1.261e-02 50.84 <2e-16 ***
```

- **Hand calculation for ARCH effect**

$$t = |\text{estimate}|/\text{e.s.e} = 3.363{\times}10^{-1}/1.652{\times}10^{-2}$$

$$t = 20.35714 > 2, \;\; p < 0.05.$$

- **Hand calculation for GARCH effect**

$$t = |\text{estimate}|/\text{e.s.e} = 6.413{\times}10^{-1}/1.261{\times}10^{-2}$$

$$t = 50.85646 > 2, \;\; p < 0.05.$$

4.6 Experimenting with higher order models

- Fit a GARCH(7, 7) model

```
garch77<-garch(resid, order=c(7, 7))
summary(garch(77)
```

- R reports problems with numerical instabilities here suggesting that the higher order model does not work properly and contains some redundancies

5.1 Bitcoin example

- Price data on Bitcoin from April 29th 2013-Feb 13th 2020
- Want to do the following tasks in R

1. Read in the data
2. Load the package `tseries` if you haven't done this already
3. Fit low order ARCH(1) and GARCH(1, 1) models
4. Experiment with the fitting of a higher order GARCH(7, 7) model that would have the interpretation of volatility from the previous week feeds forward and affects today

5.2 Loading in the Bitcoin data

- As previously,

```
>BitcoinData<-read.table("E:BitcoinData.txt")
> ncol(BitcoinData)
4
> price<-BitcoinData[,4]
> price<-rev(price)
> length(price)
2482
> logreturn<-log(price[-1])-log(price[-2482])
```

- **Now need to check that the package tseries is loaded**

5.3 Model fitting in `tseries` in R

- Need to apply models to the mean-corrected log-returns

```
resid<-logreturn-mean(logreturn)
```

- Fit ARCH(1) and GARCH(1, 1) models to this data remembering that the R syntax works the other way round to standard econometric notation

```
arch1<-garch(resid, order=c(0, 1))
garch11<-garch(resid, order=c(1, 1))
summary(arch1)
summary(garch11)
```

5.4 Testing for the ARCH effect

- **Using the command `summary` in R gives conclusive evidence of the ARCH effect**

```
Estimate Std.  Error t value Pr(>|t|)
a0 1.286e-03 2.132e-05 60.34 <2e-16 ***
a1 2.893e-01 2.023e-02 14.29 <2e-16 ***
```

- **Hand calculation for an exam-type question**

$$t = |\text{estimate}|/\text{e.s.e} = 2.893\times 10^{-1}/2.023\times 10^{-2}$$

$$t = 14.30054 > 2, \;\; p < 0.05.$$

5.5 Testing for the GARCH effect

- **Using the command `summary` in R gives conclusive evidence of the both the ARCH and the GARCH effect**

```
Estimate Std.  Error t value Pr(>|t|)
a0 7.265e-05 5.197e-06 13.98 <2e-16 ***
a1 1.411e-01 9.449e-03 14.93 <2e-16 ***
b1 8.256e-01 9.227e-03 89.48 <2e-16 ***
```

- **Hand calculation for ARCH effect**

$$t = |\text{estimate}|/\text{e.s.e} = 1.411\times10^{-1}/9.449\times10^{-3}$$

$$t = 14.9328 > 2, \; p < 0.05.$$

- **Hand calculation for GARCH effect**

$$t = |\text{estimate}|/\text{e.s.e} = 8.256\times10^{-1}/9.229\times10^{-3}$$

$$t = 89.47654 > 2, \; p < 0.05.$$

5.6 Experimenting with higher order models

- Fit a GARCH(7, 7) model

```
garch77<-garch(resid, order=c(7, 7))
summary(garch(77)
```

- Some evidence of an ARCH effect. Though non-significant terms suggest that there is some redundancy in this model and really that a lower-order model is appropriate

5.7 Results for GARCH(7, 7) model

```
 Estimate Std.  Error t value Pr(>|t|)
a0 3.138e-04 3.925e-04 0.799 0.4240
a1 1.494e-01 1.556e-02 9.604 <2e-16 ***
a2 7.778e-02 2.610e-01 0.298 0.7657
a3 4.519e-02 1.309e-01 0.345 0.7299
a4 2.248e-05 4.625e-02 0.000 0.9996
a5 8.639e-02 4.722e-02 1.829 0.0673 .
a6 2.898e-02 1.517e-01 0.191 0.8485
a7 1.267e-02 6.592e-02 0.192 0.8475
b1 5.031e-02 1.744e+00 0.029 0.9770
b2 4.727e-02 1.009e+00 0.047 0.9627
b3 5.221e-02 3.924e-01 0.133 0.8942
b4 5.608e-02 2.004e-01 0.280 0.7796
b5 5.591e-02 1.291e-01 0.433 0.6650
b6 5.588e-02 1.370e-01 0.408 0.6834
b7 6.721e-02 1.397e-01 0.481 0.6305
```

6. References

Gujarati, D. N. and Porter, D. C. (2009) *Basic econometrics*, 5th edition. McGraw Hill, New York.

Katsiampa, P. (2017) Volatility estimation for Bitcoin: A comparison of GARCH models. *Economics Letters* **158** 3-6.