Quantitative Analysis of Finance I

ECON90033

WEEK 8

CAPM WITH GARCH

HIGH FREQUENCY DATA IN FINANCE

REALISED VARIANCE

MICROSTRUCTURE NOISE, BIPOWER VARIATION AND JUMPS

Reference:

HMPY: § 15.1-15.2, 15.4-15.5

Dr László Kónya
January 2023

CAPM WITH *GARCH*

- In week 2 you learnt about the capital asset pricing model (*CAPM*) and in weeks 6 and 7 about autoregressive conditional heteroskedasticity (*ARCH*).

  This week we combine the two, i.e., we model the excess return on asset *i* as a function of the excess return on the market (among others), assuming that the stochastic error is conditionally heteroskedastic.

The mean eqn can be any eqn, including CAPM!!

Ex 1:

In Ex 2 of week 2 we estimated the original one-factor *CAPM* and the Fama-French five-factor *CAPM* for the *Cnsrm* portfolio using monthly data from July 1963 to March 2023.

Although due to the *SMB* (Small Minus Big), *HML* (High Minus Low), *RMW* (Robust Minus Weak) and *CMA* (Conservative Minus Aggressive) extra regressors the five-factor model had a better fit to the data, both models failed some of the diagnostic tests. In particular, the *BG* and *LM* tests detected autocorrelation and *ARCH* errors.

a) Try to improve the specifications by allowing for *ARMA* errors.

One-factor *CAPM*:

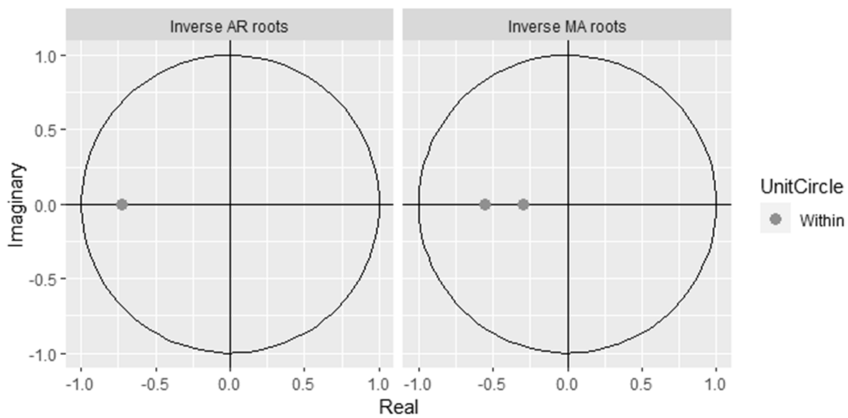
library(forecast)
best.capm1 = auto.arima(ER.Cnsmr, xreg = ER.Mkt, seasonal = FALSE, approximation = FALSE, stepwise= FALSE)
summary(best.capm1)

```
Series: ER.Cnsmr
Regression with ARIMA(1,0,2) errors

Coefficients:
          ar1     ma1     ma2  intercept    xreg
      -0.7240  0.8513  0.1644     0.1173  0.9283
s.e.   0.1807  0.1801  0.0368     0.0822  0.0154

sigma^2 = 3.535:  log likelihood = -1467.55
AIC=2947.1   AICc=2947.21   BIC=2974.55
```

autoplot(best.capm1)



library(lmtest)
coeftest(best.capm1)

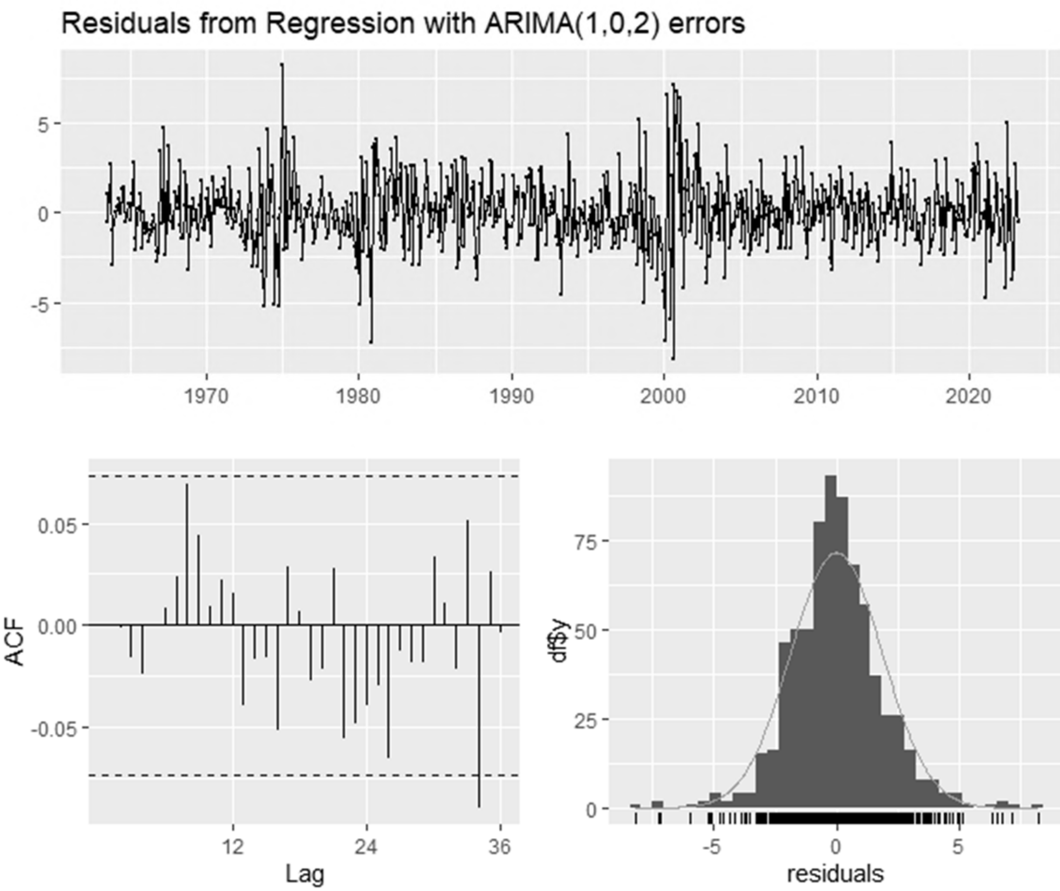
```
z test of coefficients:

           Estimate Std. Error z value  Pr(>|z|)    
ar1       -0.723990   0.180731 -4.0059 6.178e-05 ***
ma1        0.851283   0.180103  4.7267 2.283e-06 ***
ma2        0.164447   0.036786  4.4703 7.810e-06 ***
intercept  0.117310   0.082237  1.4265    0.1537    
xreg       0.928258   0.015418 60.2054 < 2.2e-16 ***
```

The *AR* and *MA* characteristic roots are well inside the unit circle,

and each term but the intercept is strongly significant.

checkresiduals(best.capm1)




```
        Ljung-Box test

data:  Residuals from Regression with ARIMA(1,0,2)
Q* = 17.339, df = 21, p-value = 0.6903

Model df: 3.   Total lags used: 24
```

According to the Bartlett and *LB* tests, the error terms are serially uncorrelated (do not worry about the single significant spike on the correlogram),
but the histogram and the *JB* test indicate that they are not normally distributed.

library(tseries)
jarque.bera.test(best.capm1$residuals)

```
        Jarque Bera Test

data:  best.capm1$residuals
X-squared = 129.26, df = 2, p-value < 2.2e-16
```

library(FinTS)
ArchTest(best.capm1$residuals, lags = 12)

```
        ARCH LM-test; Null hypothesis: no ARCH effects

data:  best.capm1$residuals
Chi-squared = 153.8, df = 12, p-value < 2.2e-16
```

In addition, there are *ARCH* effects.

Five-factor *CAPM*:

best.capm5 = auto.arima(ER.Cnsmr, xreg = cbind(ER.Mkt, SMB, HML, RMW, CMA),
seasonal = FALSE, approximation = FALSE,
stepwise= FALSE)

summary(best.capm5)

```
Series: ER.Cnsmr
Regression with ARIMA(1,0,0) errors

Coefficients:
         ar1  ER.Mkt     SMB      HML     RMW     CMA
      0.1008  0.9643  0.1143  -0.0132  0.4063  0.1422
s.e.  0.0375  0.0154  0.0229   0.0296  0.0310  0.0443

sigma^2 = 2.854:  log likelihood = -1390.38
AIC=2794.76   AICc=2794.92   BIC=2826.79
```

AIC, *AICc* and *BIC* all favour this five-factor *CAPM*.

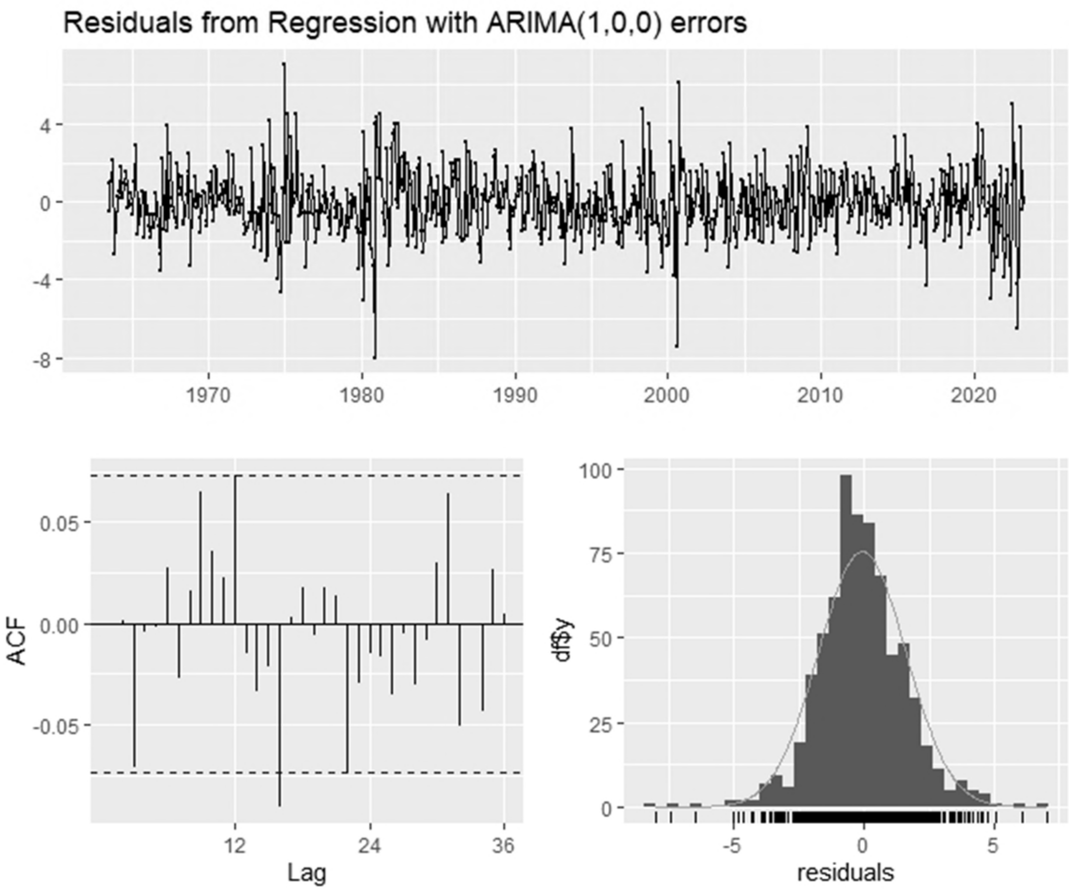
coeftest(best.capm5)

```
z test of coefficients:

        Estimate Std. Error z value  Pr(>|z|)
ar1     0.100826   0.037500  2.6887  0.007173 **
ER.Mkt  0.964347   0.015409 62.5842 < 2.2e-16 ***
SMB     0.114296   0.022877  4.9962 5.847e-07 ***
HML    -0.013237   0.029649 -0.4465  0.655271
RMW     0.406299   0.031011 13.1019 < 2.2e-16 ***
CMA     0.142228   0.044309  3.2100  0.001328 **
```

… and the single AR characteristic root is smaller than one in absolute value.

Each term but *HML* is significant at the 0.1% or stronger level, *checkresiduals(best.capm5)*




```
        Ljung-Box test

data:  Residuals from Regression with ARIMA(1,0,0)
Q* = 26.071, df = 23, p-value = 0.2975

Model df: 1.   Total lags used: 24
```

The error terms are serially uncorrelated, but they are not normally distributed,

library(tseries)
jarque.bera.test(best.capm5$residuals)

```
        Jarque Bera Test

data:  best.capm5$residuals
X-squared = 117.97, df = 2, p-value < 2.2e-16
```

ArchTest(best.capm5$residuals, lags = 12)

```
        ARCH LM-test; Null hypothesis: no ARCH effects

data:  best.capm5$residuals
Chi-squared = 64.272, df = 12, p-value = 3.716e-09
```

… and there are still *ARCH* effects.

⟶ The *CAPM* models with *ARMA* errors are free of serial correlation, but there is still conditional heteroskedasticity and non-normality.

b) Re-estimate the five-factor *CAPM* using an *ARMA*(1,0) - *GARCH*(1,1) specification this time.

```
library(rugarch)
capm5.garch.v2 = ugarchspec(mean.model = list(armaOrder = c(1,0),
                            include.mean = FALSE,
                            external.regressors = cbind(ER.Mkt, SMB, HML, RMW, CMA)),
                            variance.model = list(model = "sGARCH", garchOrder = c(1,1)),
                            distribution.model = "norm")
estimate_capm5.garch.v2 = ugarchfit(spec = capm5.garch.v2, data = ER.Cnsmr)
print(estimate_capm5.garch.v2)
```

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model      : sGARCH(1,1)
Mean Model       : ARFIMA(1,0,0)
Distribution     : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
ar1     0.090540    0.041245   2.1952 0.028149
mxreg1  0.980433    0.014538  67.4387 0.000000
mxreg2  0.087605    0.021893   4.0016 0.000063
mxreg3 -0.131525    0.031899  -4.1231 0.000037
mxreg4  0.377102    0.034906  10.8033 0.000000
mxreg5  0.260235    0.044501   5.8479 0.000000
omega   0.205830    0.075701   2.7190 0.006548
alpha1  0.159482    0.037590   4.2427 0.000022
beta1   0.773146    0.051522  15.0062 0.000000
```

In the mean equation *mxreg1*, *mxreg2*, …, *mxreg5* are the external regressors, and every regressor is significant at least at the 2.9% level.

In the variance equation the intercept (*omega*), the *AR*(1) term (*alpha1*) and the *MA*(1) term (*beta1*) are all significant.

```
Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                    0.05679  0.8116
Lag[2*(p+q)+(p+q)-1][2]   0.05811  1.0000
Lag[4*(p+q)+(p+q)-1][5]   0.29851  0.9989
d.o.f=1
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.8709  0.3507
Lag[2*(p+q)+(p+q)-1][5]    4.2385  0.2258
Lag[4*(p+q)+(p+q)-1][9]    5.4102  0.3717
d.o.f=2
```

The standardized residuals and squared residuals are both serially uncorrelated.

```
Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]     5.075 0.500 2.000 0.02427
ARCH Lag[5]     5.580 1.440 1.667 0.07536
ARCH Lag[7]     5.671 2.315 1.543 0.16492
```

There are still some *ARCH* effects, but far less significant as before.

```
Nyblom stability test
------------------------------------
Joint Statistic:  5.7072
Individual Statistics:
ar1     0.31826
mxreg1  1.57698
mxreg2  1.33558
mxreg3  1.64039
mxreg4  0.74310
mxreg5  0.41450
omega   0.05336
alpha1  0.05490
beta1   0.07031

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         2.1 2.32 2.82
Individual Statistic:    0.35 0.47 0.75
```

The joint test statistic is significant, so the joint null hypothesis that each parameter is constant is rejected.

In the mean equation four individual test statistics are significant at the 5% or lower level, while in the variance equation every test statistic is insignificant.

```
Sign Bias Test
------------------------------------
                   t-value      prob sig
Sign Bias            1.659 0.097548   *
Negative Sign Bias   1.735 0.083107   *
Positive Sign Bias   2.740 0.006300 ***
Joint Effect        11.557 0.009065 ***
```

At the 10% level each test rejects the null hypothesis of no leverage effect.

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     15.13       0.7141
2    30     23.79       0.7390
3    40     33.74       0.7083
4    50     50.99       0.3952
```

No matter how many groups the observations are classified in, there is no evidence against the null hypothesis that the errors are normally distributed.

c) Try to take care of the leverage effect by re-estimating the five-factor *CAPM* using an *ARMA*(1,0) - *TGARCH*(1,1) specification this time.

```
library(rugarch)
capm5.garch.v3 = ugarchspec(mean.model = list(armaOrder = c(1,0),
                            include.mean = FALSE,
                            external.regressors = cbind(ER.Mkt, SMB, HML, RMW, CMA)),
                            variance.model = list(model="fGARCH", submodel="TGARCH",
                            garchOrder = c(1,1)),
                            distribution.model = "norm")
estimate_capm5.garch.v3 = ugarchfit(spec = capm5.garch.v3, data = ER.Cnsmr)
print(estimate_capm5.garch.v3)
```

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : fGARCH(1,1)
fGARCH Sub-Model      : TGARCH
Mean Model      : ARFIMA(1,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
ar1     0.086652    0.038393   2.2570 0.024010
mxreg1  0.976728    0.014484  67.4347 0.000000
mxreg2  0.080457    0.021316   3.7744 0.000160
mxreg3 -0.130946    0.032680  -4.0069 0.000062
mxreg4  0.366651    0.033333  10.9997 0.000000
mxreg5  0.244055    0.046072   5.2973 0.000000
omega   0.064815    0.024644   2.6300 0.008538
alpha1  0.091818    0.025744   3.5666 0.000362
beta1   0.887161    0.030229  29.3476 0.000000
eta11   0.573251    0.204237   2.8068 0.005004
```

In the mean equation every regressor is significant at least at the 2.5% level.

In the variance equation also, every term is strongly significant. Note that η_1-hat is significantly positive, so negative shocks have greater effect on expected volatility than positive shocks.

```
Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                    0.02591  0.8721
Lag[2*(p+q)+(p+q)-1][2]   0.04340  1.0000
Lag[4*(p+q)+(p+q)-1][5]   0.64465  0.9857
d.o.f=1
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      0.166  0.6837
Lag[2*(p+q)+(p+q)-1][5]     3.832  0.2758
Lag[4*(p+q)+(p+q)-1][9]     5.292  0.3879
d.o.f=2
```

The standardized residuals and squared residuals are both serially uncorrelated.

```
Weighted ARCH LM Tests
------------------------------------
             Statistic Shape Scale P-Value
ARCH Lag[3]      5.287 0.500 2.000 0.02149
ARCH Lag[5]      5.963 1.440 1.667 0.06127
ARCH Lag[7]      6.329 2.315 1.543 0.12061
```

There are still some uncontrolled *ARCH* effects,

the stability tests challenge the specification of the mean equation,

```
Nyblom stability test
------------------------------------
Joint Statistic:  6.12
Individual Statistics:
ar1     0.18056
mxreg1  2.00274
mxreg2  1.37359
mxreg3  1.71121
mxreg4  0.76638
mxreg5  0.42179
omega   0.07581
alpha1  0.08136
beta1   0.08246
eta11   0.24505

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         2.29 2.54 3.05
Individual Statistic:    0.35 0.47 0.75
```

```
Sign Bias Test
------------------------------------
                   t-value    prob sig
Sign Bias            2.007 0.04510  **
Negative Sign Bias   1.428 0.15378
Positive Sign Bias   1.761 0.07861   *
Joint Effect         5.390 0.14540
```

… and the leverage effects have not been fully accounted for either.

The normality assumption remains unchallenged.

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     18.54       0.4869
2    30     27.90       0.5235
3    40     39.54       0.4457
4    50     47.50       0.5339
```

How do the *ARMA*(1,0) - *GARCH*(1,1) and *ARMA*(1,0) - *TGARCH*(1,1) models compare to each other?

To answer this question, we need to compare the model specification values.

ARMA(1,0) - *GARCH*(1,1)

```
Information Criteria
---------------------

Akaike       3.7748
Bayes        3.8322
Shibata      3.7745
Hannan-Quinn 3.7969
```

ARMA(1,0) - *TGARCH*(1,1)

```
Information Criteria
--------------------

Akaike       3.7625
Bayes        3.8263
Shibata      3.7621
Hannan-Quinn 3.7871
```

Although none of these models is ‘perfect’, all four model specification criteria support the second model.

HIGH FREQUENCY DATA IN FINANCE

- High-frequency data are observations taken at fine time intervals, say daily, hourly or at an even finer time scale.

Ex 2:

The file *w8e2.xlsx* contains the intra-day log-prices of IBM shares on 3 January 1996 between 9:30 and 16:00 (taken from the TAQ database). A snapshot of these trading data is given below.

	A	B	C	D	E	F	G	H
1	DATEVEC	TS	TIME	TIMESTAMP	LOGPRICE	TRADE	DURATION	PRICE
26	03-01-96 9:30			24				
27	03-01-96 9:30			25				
28	03-01-96 9:30			26				
29	03-01-96 9:30	9:30:27	93027	27	4.413404	3	9	82.55
30	03-01-96 9:30			28				
31	03-01-96 9:30			29				
32	03-01-96 9:30			30				
33	03-01-96 9:30	9:30:31	93031	31	4.412435	4	4	82.47
34	03-01-96 9:30			32				
35	03-01-96 9:30	9:30:33	93033	33	4.412677	5	2	82.49
36	03-01-96 9:30			34				
37	03-01-96 9:30	9:30:35	93035	35	4.412556	6	2	82.48
38	03-01-96 9:30	9:30:36	93036	36	4.412313	7	1	82.46
39	03-01-96 9:30			37				
40	03-01-96 9:30			38				
41	03-01-96 9:30			39				

The structure of this data set is like that of in Exercise 3 of Tutorial 2.

For example, the 4th trade of the day occurred at 9:30:31 at the price \$82.47, and the next one 2 seconds later at 9:30:33 at the price \$82.49.

The duration time (column G) between these two trades is 2 seconds.

a) Plot the log prices (*LP*) of IBM shares.

The time unit is one second and there were 6326 trades in 23400 seconds that day. Consider only the seconds with trade.

```
attach(w8e1)
data = na.omit(w8e1)
detach(w8e1)
attach(data)

library(xts)
LP = xts(LOGPRICE,
        order.by = as.Date(DATEVEC))
plot.xts(LP, xlab = "Date", ylab = "LP",
       main = "Log price of IBM stocks",
       col = "darkgreen")
```

This plot of intra-day data provides information on volatility of log-prices for 3 January 1996. Namely, it shows that

(i) price fell in the first hour,

(ii) remained relatively flat for an extended period, and

(iii) eventually rose and reached a closing price below the opening price.

- How to measure daily volatility of high-frequency data?

  The simplest estimator is the range volatility, the difference between the largest and smallest log-prices of the day.

(Ex 2, cont.)

b) What is the range volatility of the intra-day log-prices of IBM shares on 3 January 1996?

```
Range = max(LP) - min(LP)
print(Range)
```

```
0.021303
```

→ The difference between the largest and smallest IBM share log-prices on 3 January 1996 was 0.021303, implying that the difference between the largest and smallest IBM share prices was $e^{0.021303}$ = \$1.0215.

As usual, the range is a very simple and convenient measure of dispersion, but since it does not use all available information of intra-day transactions prices, it is an inefficient estimator of volatility.

A far more comprehensive measure is the realised variance or volatility.

REALISED VARIANCE / VOLATILITY (*RV*)

- An alternative class of volatility models to *GARCH* models, known as Realised Variance or Realised Volatility (*RV)* has become popular recently.

  Unlike the range volatility estimator, the realized variance estimator uses all available information on intra-day prices.

  ⟵ Suppose that the price is recorded *M* times on a given trading day and that the opening and closing prices are P_0 and P_1,

$$P_0 \qquad P_{1/M}, P_{2/M}, \ldots, P_{(M-1)/M}, \qquad P_1$$

Intra-day prices

i.e. the number of possible intervals

The sample size for the day is *M* (the opening price is taken from the previous day), and it is

$6.5 \times 60 \times 60 = 23400$ for one-second frequency,
$23400 / 5 = 4680$ for five-second frequency,
$23400 / 60 = 390$ for one-minute frequency,
$23400 / (60 \times 5) = 78$ for five-minute frequency, etc.

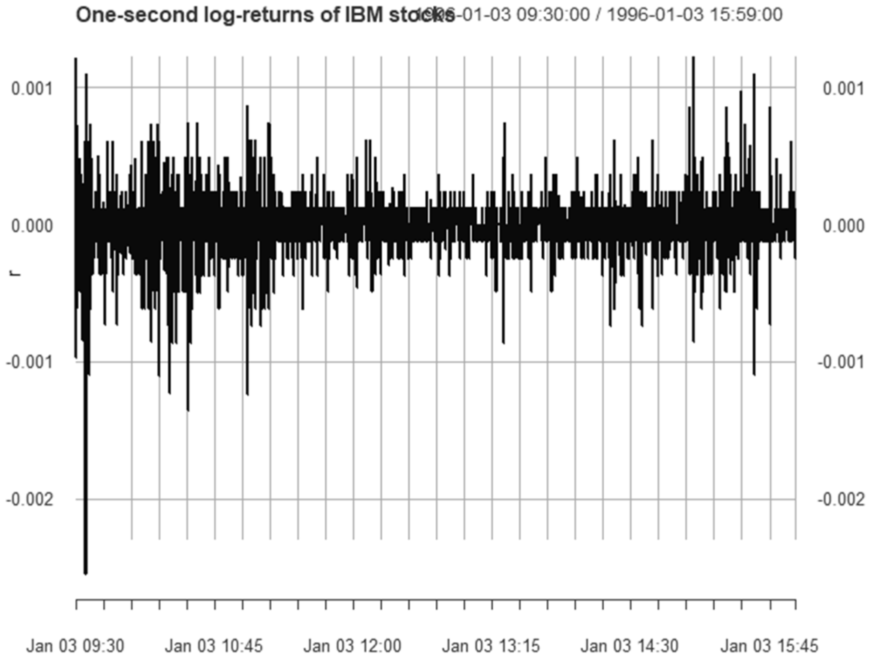
Given this setup, the log returns are

$$r_i = \ln P_{i/M} - \ln P_{(i-1)/M} \quad , \quad i = 1, 2, \ldots, M$$

(Ex 2)

b) Calculate and plot the one-second log returns of IBM shares.

```
r = LP - lag(LP)
plot.xts(r, xlab = "Date", ylab = "r",
         main = "One-second log-returns of IBM
                 stocks",
         col = "darkblue")
```



The variance of the *M* log returns is

$$Var(r) = \frac{1}{M}\sum_{i=1}^{M}(r_i - \bar{r})^2 \approx \frac{1}{M}\sum_{i=1}^{M} r_i^2$$

It is reasonable to assume for high frequency data that the mean of *r* is zero.

Realised volatility: the estimate of the daily variability of a financial asset.

From the variance of the *M* intra-day log returns,

$$RV(M) = M \times Var(r) \approx \sum_{i=1}^{M} r_i^2$$

i.e., the realised volatility on a trading day is the sum of the squared log returns during that day.

(Ex 1)

c) What was the realised volatility of the IBM shares on 3 January 1996?

```
rvec = data.frame(drop(coredata(r)))[-1,]
RV = sum(rvec^2)
print(RV)
0.0001690085
```

$\longrightarrow$ The estimate of the realised volatility of IBM shares on 3 January 1996 is about 0.000169.

This estimate of *RV* was calculated from all 6326 log-prices of the day.

It is also possible, however, to compute *RV* for different frequencies, like one-minute, five-minute and ten-minute frequencies.

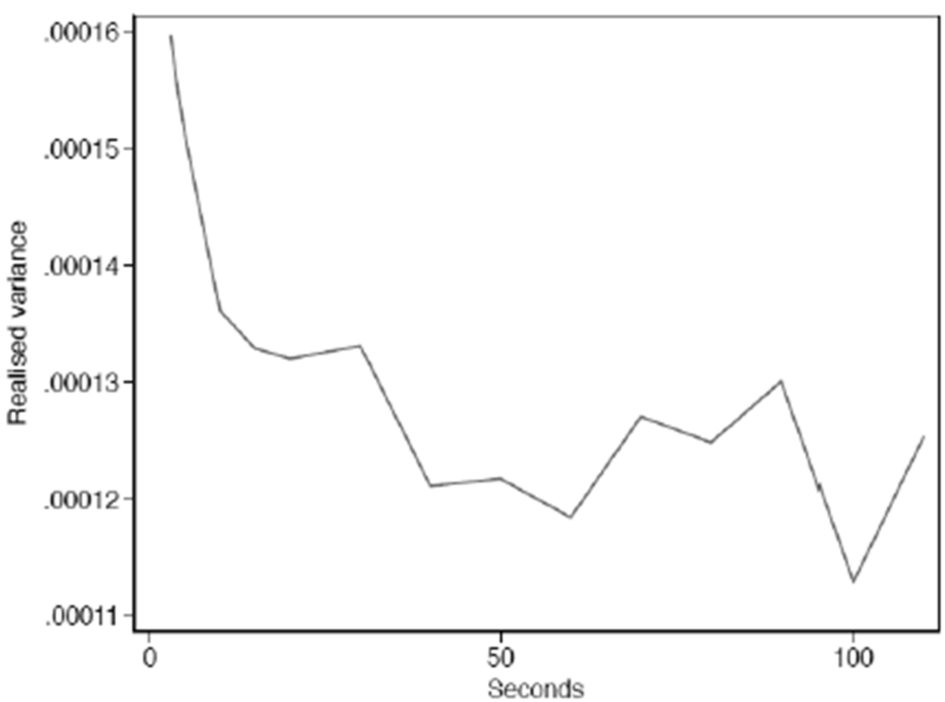
Without showing the details, these *RV* estimates are as follows.

One-minute frequency: $RV(M = 390) = 0.000118$

Five-minute frequency: $RV(M = 78) = 0.000163$

Ten-minute frequency: $RV(M = 39) = 0.000155$

Repeating the calculations for other possible frequencies and plotting the *RV* estimates against the corresponding frequencies, we get the so-called signature plot:



The sampling frequency ranges from 1 second to 110 seconds.

There are some fluctuations, but overall, the *RV* estimates are relatively high for shorter intervals and tend to decay as the interval increases.

(This plot is from HMPY, p. 441. It is not reproducible because the raw data are not available.)

MICROSTRUCTURE NOISE, BIPOWER VARIATION AND JUMPS

- In our example the *RV* estimates rapidly decrease from 0.000169 to 0.000132 as the sampling frequency decreases from 1 second to 10 seconds and then seem to converge at sampling frequencies around one minute.

  In general, we expect larger sample data to produce more accurate results. So, why are the *RV* estimates computed at higher frequencies, i.e., from more information (larger *M*), more volatile, less precise?

  Without discussing the details, it can be shown that as the sampling frequency of intra-daily log-returns is kept increasing (i.e., $M \to \infty$), the estimates of daily volatility, $RV(M)$, are expected to converge to the population parameter, called Integrated Volatility.

  Although this expectation is confirmed theoretically under certain conditions (in particular, considering continuous trading, i.e., shorter and shorter time intervals), working with real data extraneous factors might invalidate it.

⟵ Additional movements in prices at higher frequencies that are not the result of market movements but are the result of the trading system are called microstructure noise.

It might be caused by the discrete nature of real price changes, jumps in the prices, infrequent and irregular trading, asymmetric information between buyers and sellers, etc., and it makes high frequency estimates of some parameters (like *RV*) very unstable.

It can be represented by a random disturbance term, ε_t, in the following simple model of asset prices,

$$\boxed{P_t = F_t + \varepsilon_t}$$

where P_t is the actual price and F_t is the market fundamental price.

A possible solution is to estimate *RV* from lower frequency data that are not contaminated by microstructure noise.

⟶ A standard practice is to report *RV* computed using five- or ten-minute log-returns as they are considered not to be affected by the microstructure noise.

- Microstructure noise represents a possible mechanism for biasing volatility estimates based on market movements in price.

  Another mechanism that can bias volatility estimates are jumps, defined as one-off movements in the price caused by an external shock such as a policy announcement, a new CEO, release of some financial data, etc.

  If there are jumps, the *RV* estimate of volatility represents the sum of two terms due to the market and to the jump, respectively.

  → A jump-robust estimator of realised volatility is required to detect the persistent dynamics of daily volatility of asset returns.

  One such estimator is the realised bipower volatility estimator based on the autocorrelation of the absolute log-returns,

$$BV(M) = \frac{\pi}{2} \sum_{i=2}^{M} |r_i| \times |r_{i-1}|$$

  The rationale behind this estimator is that jumps tend to be infrequent, so if r_i is large (in absolute value), it is likely offset by r_{i-1} which is expected to be relatively small (in absolute value).

(Ex 2)

d) What is the realised bipower volatility estimate of IBM shares on 3 January 1996?

Using the original one-second frequency data,

```
lrvec = c(0, rvec[1:length(rvec) - 1])
BV = (pi/2)*crossprod(abs(rvec), abs(lrvec))
print(BV)
```

```
0.000111028
```

→ The estimate of the realised bipower volatility of IBM shares on 3 January 1996 is about 0.000111.

It is smaller than the realised volatility estimate in part (c) (*RV* = 0.000169, see slide 7), providing evidence of jumps that bias the *RV* estimate upwards.

Jump volatility, the contribution of jumps to daily volatility, can be measured by the difference between the two estimates, or as a percentage,

$$\boxed{RV - BV = 0.000169 - 0.000111 = 0.000058}$$

$$\boxed{\frac{RV - BV}{RV} = \frac{0.000169 - 0.000111}{0.000169} = 0.343 \quad \rightarrow \quad 34.3\%}$$

From one-minute frequency data,

$RV(M = 390) = 0.000118$ and $BV(M = 390) = 0.000103$, and

$$\boxed{RV - BV = 0.000118 - 0.000103 = 0.000015}$$

$$\boxed{\frac{RV - BV}{RV} = \frac{0.000118 - 0.000103}{0.000118} = 0.127 \quad \rightarrow \quad 12.7\%}$$

The jump volatility estimates computed from one-second frequency data is larger than the ones computed from one-minute frequency data.

WHAT SHOULD YOU KNOW?

- High frequency data
- Realized variance / volatility
- Microstructure noise
- Bipower variation and jumps