Quantitative Analysis of Finance I

ECON90033

WEEK 7

FORECASTING WITH GARCH MODELS

EXTENSIONS TO THE BASIC GARCH MODEL: IGARCH, EGARCH, TGARCH AND GARCH-M MODELS

Reference:

HMPY: § 13.3-13.6

Dr László Kónya
January 2023

FORECASTING WITH *GARCH* MODELS

- Once we managed to obtain a satisfactory (*G*)*ARCH* model, we can use it to forecast both the conditional mean and the conditional variance and to estimate forecast error variances.

  No matter which *GARCH* model (or heteroskedastic dynamic model, in general) is used, we can do so in three steps:

  1) Compute conditional mean and variance forecasts.
  2) Compute forecast errors and forecast error variances.
  3) Compute interval forecasts.

  For illustration, consider a stationary *AR*(1)-*GARCH*(1,1) model,

$$y_t = \varphi_0 + \varphi_1 y_{t-1} + \varepsilon_t \;\;,\;\; |\varphi_1| < 1 \;,\; \varepsilon_t = \nu_t \sqrt{h_t} \;\;,\;\; \nu_t : idN(0,1) \;\;,\;\; \rho_{h_t,\nu_t} = 0$$

$$h_t = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 h_{t-1} \quad , \quad \alpha_1 > 0 \,,\, \beta_1 > 0 \,,\, \alpha_1 + \beta_1 < 1$$

  and see these three steps in details.

Assume that the information set in time T, i.e., Ω_T, includes all ε_t and h_t for $t = 1, 2, \ldots, T$.

1) The conditional mean and variance can be forecast for $k = 1, 2, \ldots$ periods ahead the usual way by recursive substitution, i.e., by writing out the process for $T + k$ and replacing historical expectations by their realisations and all expectations of future innovations by zero.

recursive sub. to $E_T(y_{T+k})$

$$E_T(y_{T+1}) = \varphi_0 + \varphi_1 E_T(y_T) + E_T(\varepsilon_{T+1}) = \varphi_0 + \varphi_1 y_T$$

$$E_T(y_{T+2}) = \varphi_0 + \varphi_1 E_T(y_{T+1}) + E_T(\varepsilon_{T+2})$$
$$= \varphi_0 + \varphi_1(\varphi_0 + \varphi_1 y_T) = \varphi_0(1 + \varphi_1) + \varphi_1^2 y_T$$

$$E_T(y_{T+3}) = \varphi_0 + \varphi_1 E_T(y_{T+2}) + E_T(\varepsilon_{T+3})$$
$$= \varphi_0 + \varphi_1(\varphi_0(1 + \varphi_1) + \varphi_1^2 y_T) = \varphi_0(1 + \varphi_1 + \varphi_1^2) + \varphi_1^3 y_T$$

$$\longrightarrow \quad E_T(y_{T+k}) = \underbrace{\varphi_0 \sum_{i=0}^{k-1} \varphi_1^i}_{\text{sum b/c it is an AR process}} + \varphi_1^k y_T \quad , \quad k = 1, 2, \ldots$$

Likewise,

$$E_T(h_{T+1}) = E_T(\alpha_0 + \alpha_1\varepsilon_T^2 + \beta_1 h_T) = \alpha_0 + \alpha_1\varepsilon_T^2 + \beta_1 h_T$$

the same process for y_{t+k} applies to h_{t}

$$E_T(h_{T+k}) = E_T(\alpha_0 + \alpha_1\varepsilon_{T+k-1}^2 + \beta_1 h_{T+k-1})$$

$$= \alpha_0 + \alpha_1 \underbrace{E_T(\varepsilon_{T+k-1}^2)}_{E_T(\nu_{T+k-1}^2 h_{T+k-1}) = E_T(h_{T+k-1})} + \beta_1 E_T(h_{T+k-1})$$

because $\sigma_\nu^2 = 1$ and ν_{T+k-1} is independent of h_{T+k-1}.

$$E_T(h_{T+k}) = \alpha_0 + (\alpha_1 + \beta_1)E_T(h_{T+k-1})$$

$$= \dots = \underbrace{\alpha_0[1 + (\alpha_1+\beta_1) + (\alpha_1+\beta_1)^2 + \dots + (\alpha_1+\beta_1)^{k-1}]}_{\to \frac{\alpha_0}{1-\alpha_1-\beta_1}} + \underbrace{(\alpha_1+\beta_1)^k}_{\to 0} h_T$$

because $\alpha_1 > 0$, $\beta_1 > 0$, and $\alpha_1 + \beta_1 < 1$.

$$\longrightarrow \quad \lim_{k\to\infty} E_T(h_{T+k}) = \frac{\alpha_0}{1-\alpha_1-\beta_1}$$

2) The forecast errors and forecast error variances also can be calculated as earlier for *ARMA* models.

$$e_{T+1} = y_{T+1} - E_T(y_{T+1}) = \varphi_0 + \varphi_1 y_T + \varepsilon_{T+1} - \varphi_0 - \varphi_1 y_T = \varepsilon_{T+1}$$

$$Var_T(e_{T+1}) = Var_T(\varepsilon_{T+1}) = h_{T+1}$$

theoretical ARMA f/c process

$$e_{T+2} = y_{T+2} - E_T(y_{T+2}) = \ldots = \varepsilon_{T+2} + \varphi_1 \varepsilon_{T+1}$$

$$Var_T(e_{T+2}) = Var_T(\varepsilon_{T+2}) + Var_T(\varphi_1 \varepsilon_{T+1}) = h_{T+2} + \varphi_1^2 h_{T+1}$$

$$\begin{aligned} e_{T+k} &= y_{T+k} - E_T(y_{T+k}) = \ldots \\ &= \varepsilon_{T+k} + \varphi_1 \varepsilon_{T+k-1} + \varphi_1^2 \varepsilon_{T+k-2} + \ldots + \varphi_1^{k-1} \varepsilon_{T+1} = \sum_{i=0}^{k-1} \varphi_1^i \varepsilon_{T+k-i} \end{aligned}$$

$$Var_T(e_{T+k}) = Var_T\left(\sum_{i=0}^{k-1} \varphi_1^i \varepsilon_{T+k-i}\right) = \sum_{i=0}^{k-1} \varphi_1^{2i} Var_T(\varepsilon_{T+k-i}) = \sum_{i=0}^{k-1} \varphi_1^{2i} h_{T+k-i}$$

To make this forecasting procedure operational in practice,

i. The unknown parameters are replaced by their estimates.
ii. The unknown ε_T^2 is replaced by e_T^2 (last squared residual).
iii. The unknown h_T is replaced by h_T-hat (last estimate of the conditional variance).

The *GARCH*(1,1) conditional mean and variance forecasts for k periods ahead, and the corresponding forecast error variances are computed as

$$\hat{y}_{T+k} = \hat{\varphi}_0 \sum_{i=0}^{k-1} \hat{\varphi}_1^i + \hat{\varphi}_1^k y_T \quad , \quad k = 1, 2, \ldots$$

$$\hat{h}_{T+1} = \hat{\alpha}_0 + \hat{\alpha}_1 e_T^2 + \hat{\beta}_1 \hat{h}_T$$ and $$\hat{h}_{T+k} = \hat{\alpha}_0 + (\hat{\alpha}_1 + \hat{\beta}_1) \hat{h}_{T+k-1} \quad , \quad k > 1$$

$$\widehat{Var_T}(e_{T+k}) = \sum_{i=0}^{k-1} \hat{\varphi}_1^{2i} \hat{h}_{T+k-i}$$

3) Interval forecasts can be obtained from the conditional mean forecasts and the estimates of the corresponding forecast error variances.

Since we assumed normality, the $(1-\alpha)\times 100\%$ prediction intervals are

$$\hat{y}_{T+k} \pm t_{\alpha/2,df} \sqrt{\widehat{Var}_T(e_{T+k})}$$

- Similar result can be obtained for finite-order *GARCH* models in general, granted that the *ARCH* lag polynomial is stable, i.e., its characteristic roots lie outside the unit circle.

<u>Ex 1</u>:

Last week in Ex 1 we estimated an *AR*(1)-*ARCH*(1) model for the approximate rate of change (i.e., the first difference of the logarithm) of the daily closing US dollar to Australian dollar exchange rate (*DLNEXR*) using data from 16 May 2006 to 2 June 2023.

We used the *ugarchspec()* and *ugarchfit()* functions of the *rugarch R* package to estimate the *AR*(1)-*ARCH*(1) model and named it *fit_v1* (see slide #24 of the week 6 lecture notes). Although it failed some diagnostics, let's use it to illustrate forecasting with *GARCH* models.

Using the fitted model, we can forecast both the conditional mean and the conditional variance with the *ugarchforecast()* function of the *rugarch* package.

For example, let's generate ex ante forecasts for 20 business days following the end of the sample period.

```
DLNEXR_eaf = ugarchforecast(fit_v1,
                    data = DLNEXR, n.ahead = 20)
print(DLNEXR_eaf)
```

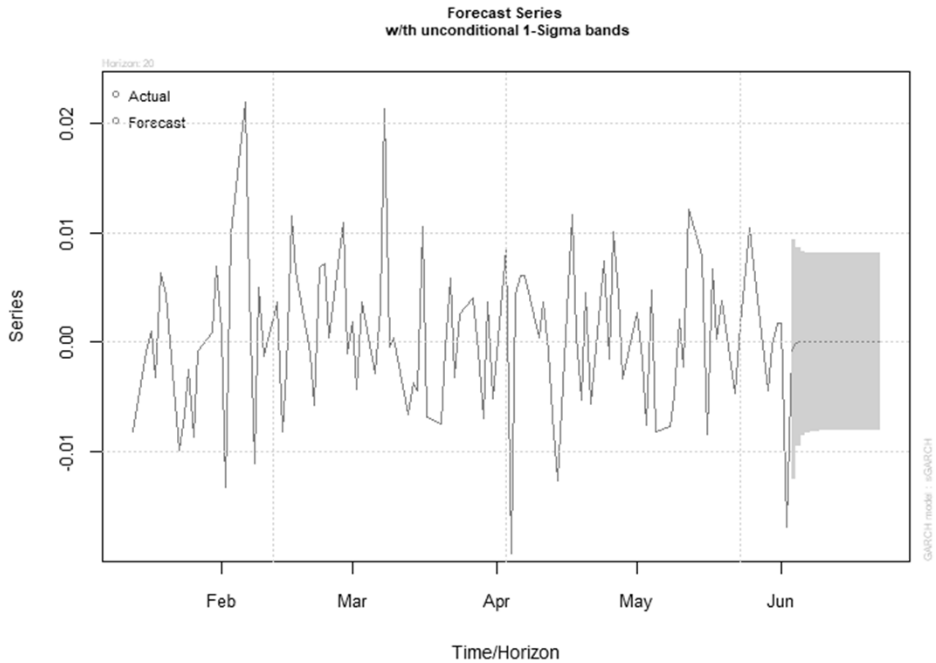
```
*------------------------------------*
*       GARCH Model Forecast         *
*------------------------------------*
Model: sGARCH
Horizon: 20
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=2023-06-02]:
         Series    Sigma
T+1  -7.578e-04 0.011734
T+2  -1.383e-05 0.009444
T+3   2.053e-05 0.008574
T+4   2.212e-05 0.008273
T+5   2.219e-05 0.008174
T+6   2.220e-05 0.008142
T+7   2.220e-05 0.008131
T+8   2.220e-05 0.008128
T+9   2.220e-05 0.008127
T+10  2.220e-05 0.008126
T+11  2.220e-05 0.008126
T+12  2.220e-05 0.008126
T+13  2.220e-05 0.008126
T+14  2.220e-05 0.008126
T+15  2.220e-05 0.008126
T+16  2.220e-05 0.008126
T+17  2.220e-05 0.008126
T+18  2.220e-05 0.008126
T+19  2.220e-05 0.008126
T+20  2.220e-05 0.008126
```

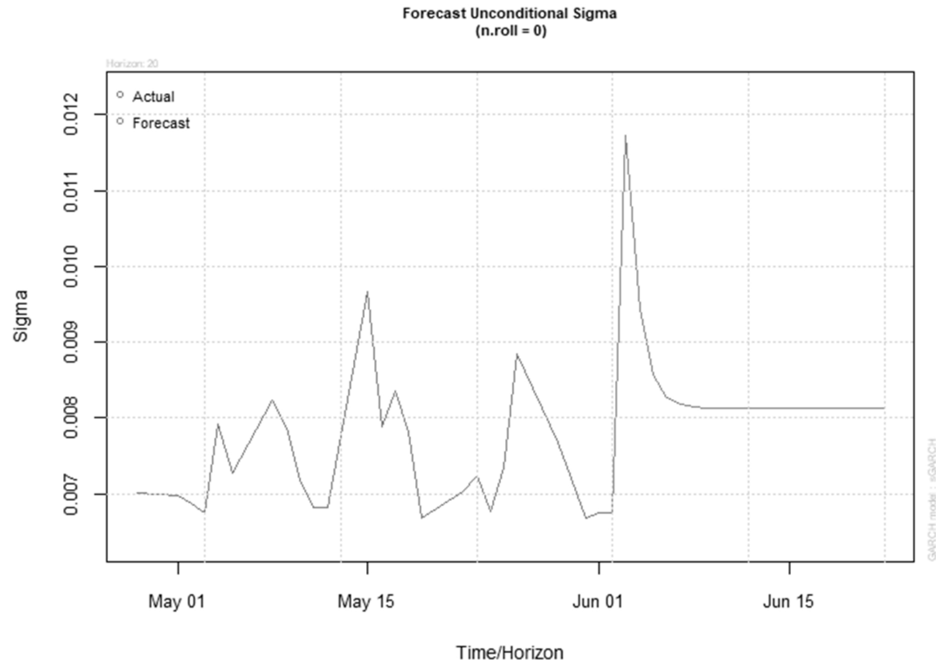
These are 1, 2, …, 20 days ahead forecasts following 2 June 2023, so they are not rolling forecasts.

The first column shows the 20 business days after 2 June 2023, and the estimated mean and volatility of *DLNEXR* are in the second and third columns.

After a few days, the mean forecast converges to 2.220e-05 = 0.0000222 and the volatility forecast to 0.008126.

plot(DLNEXR_eaf)





EXTENSIONS TO THE BASIC *GARCH* MODEL

- An important restriction of the basic *GARCH* model is that it tacitly assumes that the impact of a shock on future volatility depends only on the magnitude of the shock, but not on its sign.

  ← The conditional variance, h_t, is supposed to depend on ε_{t-i}^2 but not on ε_{t-i}.

  Moreover, in order to ensure positive conditional variances, all α_i, β_j coefficients are restricted to be non-negative and their sum ($i = 1, \ldots, q$, $j = 1, \ldots, p$) must be less than one.

Other conditional volatility models, based on alternative specifications of conditional heteroskedasticity, either relax these restrictions or extend the basic *GARCH* specification by relating the level and the volatility of some variable to each other.

In integrated-*GARCH* models, for example, the α_i, β_j coefficients add up to one. In exponential-*GARCH* and threshold-*GARCH* models h_t might react differently to negative and to positive ε_{t-i}. *GARCH-in-mean* models allow the mean of $\{y_t\}$ to depend on its own conditional variance.

- In a *GARCH* model the conditional variance is positive and finite and volatility is stationary if

$$\sum_{i=1}^{q} \alpha_i + \sum_{j=1}^{p} \beta_j < 1$$

Occasionally, however, it is realistic to assume that

$$\sum_{i=1}^{q} \alpha_i + \sum_{j=1}^{p} \beta_j = 1$$

implying a unit root in the conditional variance.

This model is called an integrated-*GARCH*(p,q) model and it is denoted as *IGARCH*(p,q).

In this case any shock to the conditional variance has a persistent effect, and in this sense *IGARCH* processes are like *ARIMA* processes ($d \geq 1$).

Yet, this analogy is misleading because while *ARIMA* processes are non-stationary, *IGARCH* processes are stationary and any apparent persistence of the shocks is likely due to some thick-tailed distribution.

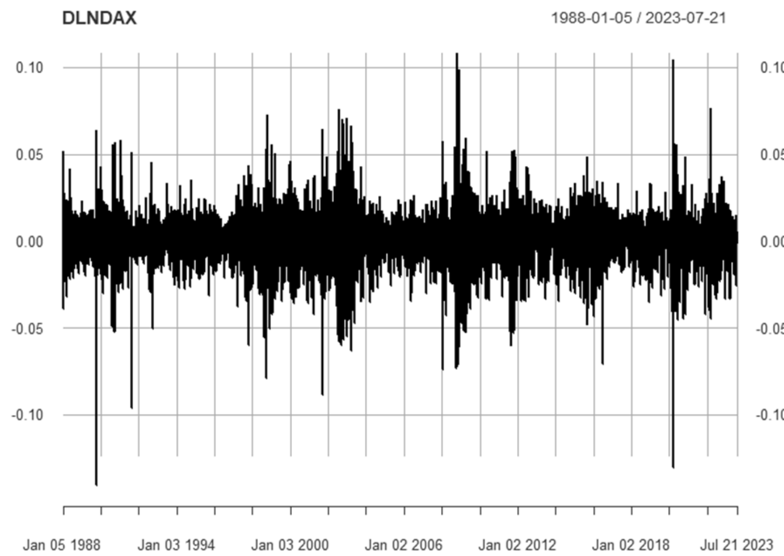
$\longrightarrow$ After having imposed the parameter restriction, *IGARCH* models can be estimated like standard *GARCH* models.

Ex 2:

The DAX (Deutscher Aktienindex) is a stock market index consisting of the 40 major German blue-chip companies trading on the Frankfurt Stock Exchange. Its daily closing values (*DAX*) from 4 January 1988 to 21 July 2023 are saved in *t7e2.xlsx* (downloaded from https://finance.yahoo.com).

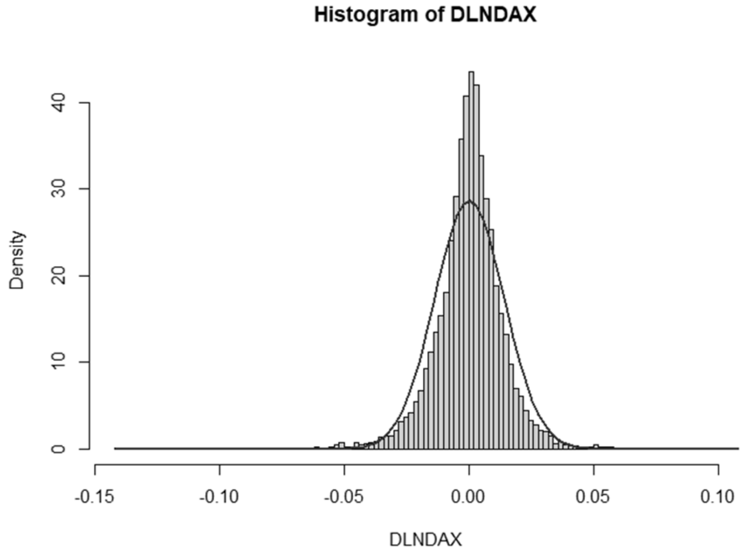
a) Calculate the daily log returns of *DAX* and illustrate it with a time series plot and a histogram.

```
library(xts)
DAX = xts(Close, order.by = as.Date(Date))
DLNDAX = na.omit(diff(log(DAX), 1))

plot.xts(DLNDAX)
```



The daily log returns of *DAX* fluctuate around some constant with seemingly changing volatility.

```
hist(DLNDAX, breaks = 100, freq = FALSE,
     col = "lightblue")

x = DLNDAX
curve(dnorm(x, mean = mean(DLNDAX),
      sd = sd(DLNDAX)), add = TRUE,
      col = "red“, lwd = 2)
```



The distribution of *DLNDAX* has a sharper peak and slightly fatter tails than a normal distribution with the same expected value and standard deviation, i.e., it is leptokurtic (Kurtosis = 9.69551).

b) According to the *ADF* tests on the level and on the first difference of the logarithm of *DAX* (not shown here), *DLNDAX* is stationary.

Estimate a *GARCH*(1,1) model with a constant mean equation for *DLNDAX*.

```
library(rugarch)
spec_v1 = ugarchspec(mean.model = list(armaOrder = c(0,0), include.mean = TRUE),
                     variance.model = list(model = "sGARCH", garchOrder = c(1,1)),
                     distribution.model = "norm")
fit_v1 = ugarchfit(spec = spec_v1, data = DLNDAX)
print(fit_v1)
```

If $\alpha_1 + \beta_1 \approx 1$ Use IGARCH

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000718    0.000111    6.4463  0e+00
omega   0.000004    0.000001    4.4122  1e-05
alpha1  0.099112    0.006259   15.8342  0e+00
beta1   0.880301    0.007110  123.8103  0e+00
```

$$\widehat{DLNDAX}_t = 0.000718 + e_t \quad , \quad e_t \sim N(0, \hat{h}_t)$$

$$\hat{h}_t = 0.000004 + 0.099112 e_{t-1}^2 + 0.880301 \hat{h}_{t-1}$$

The point estimates of α_1 and β_1 add up to 0.979413, very close to 1.

Without presenting the details, the rest of the printout suggests that

i. The robust standard errors make the *t*-ratios of the slopes much smaller, but only the point estimate of *omega* (intercept) becomes insignificant.
ii. The weighted *LB* tests do not detect autocorrelation, neither in the standardized residuals nor in the standardized squared residuals.
iii. According to the weighted *ARCH LM* tests, there are no *ARCH* effects left in the residuals.
iv. The joint Nyblom stability test indicates some parameter instability, but none of the individual tests.
v. The sign bias tests detect some leverage effect.
vi. The adjusted Pearson tests reject normality.

As regards normality, we stick to it this time because the results based on the *t* distribution are worse. We are going to focus instead on the possibility of a unit root in the conditional variance and on the leverage effect.

c) Estimate an *IGARCH*(1,1) model with a constant mean equation for *DLNDAX*.

```
spec_v2 = ugarchspec(mean.model = list(armaOrder = c(0,0), include.mean = TRUE),
                     variance.model = list(model = "iGARCH", garchOrder = c(1,1)),
                     distribution.model = "norm")
fit_v2 = ugarchfit(spec = spec_v2, data = DLNDAX)
print(fit_v2)
```

```
*---------------------------------*
*          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      0.000722    0.000111   6.4894 0.000000
omega   0.000003    0.000001   3.0288 0.002455
alpha1  0.117372    0.011137  10.5393 0.000000
beta1   0.882628          NA       NA       NA
```

$$\widehat{DLNDAX}_t = 0.000722 + e_t \quad , \quad e_t \sim N(0, \hat{h}_t)$$

$$\hat{h}_t = 0.000003 + 0.117372 e_{t-1}^2 + 0.882628 \hat{h}_{t-1}$$

$\beta_1 = 1 - \alpha_1$, so it is not estimated.

Without presenting the details, the rest of the printout suggests that

i. The robust standard errors make *omega* and *alpha1* insignificant.
ii. As one should expect, all four model specification criteria favour the *GARCH*(1,1) model over this restricted model.
iii. The weighted *LB* tests do not detect autocorrelation.
iv. The weighted *ARCH LM* tests do not detect any remaining *ARCH* effect.
v. The Nyblom stability tests reject stability for *omega*.
vi. The sign bias tests detect some leverage effect.
vii. The adjusted Pearson tests reject normality.

As an additional check, it is useful to test H_0: $\alpha_1 + \beta_1 = 1$ on the *GARCH* model.

This linear restriction can be tested with the likelihood ratio (*LR*), Wald or Lagrange multiplier (*LM*) tests based on the unrestricted *GARCH* model and the restricted *IGARCH* model.

In the *LR* test, for example, the test statistic is

$$\lambda = 2(\ln L_{ur} - \ln L_r)$$

where L_{ur} and L_r are the likelihood values of the unrestricted and restricted models, respectively,

Under H_0 it follows a chi-square distribution with degrees of freedom equal to the number of restrictions.

There is not a specific *R* function to perform the *LR* test on a *GARCH* model, but we can do it step-by-step.

Likelihood values:

url = likelihood(fit_v1)
print(round(url,2)) 27009.85

rl = likelihood(fit_v2)
print(round(rl,2)) 26988.86

Test statistic:

lambda = 2(log(url) - log(rl))*
print(round(lambda,5)) 0.00155

p-value:

pvalue = 1 - pchisq(q = lambda, df = 1)
print(round(pvalue,4)) 0.9685

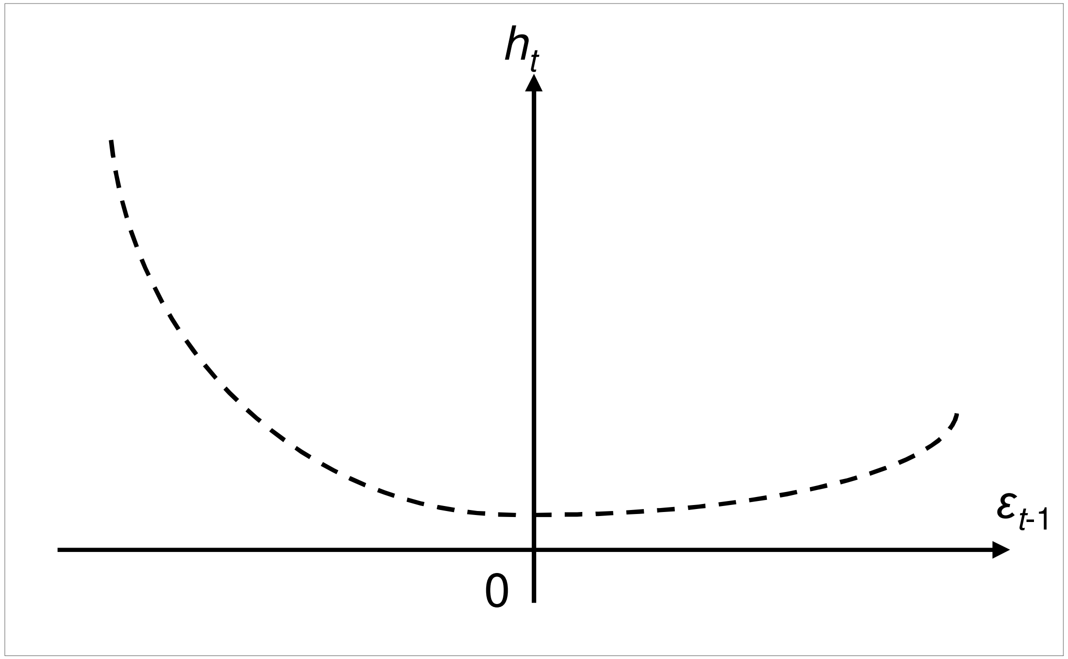
↓

H_0: $\alpha_1 + \beta_1 = 1$ is maintained, supporting the *IGARCH* model.

- Threshold-*GARCH* (*TGARCH*) and Exponential-*GARCH* (*EGARCH*) models allow for asymmetry.

    ⟵ An unexpected ‘bad news’ ($\varepsilon_t < 0$) is likely to have larger impact on future volatility than an unexpected ‘good news’ ($\varepsilon_t > 0$) of the same magnitude. This phenomenon is often referred to as leverage effect.

It can be illustrated as follows:



The curve on the left side of the origin is steeper than the curve on the right side of the origin.

Hence, if ‘new information’ is measured by ε_{t-1}, a negative shock has a bigger effect on volatility than a positive shock of the same magnitude.

This can be modelled in the following ways.

Threshold-*GARCH* (*TGARCH*) model:

d = dummy

$d_{t-i} = 1$ if $\varepsilon_{t-i} < 0$

$$y_t = \mu_t + \varepsilon_t \qquad \varepsilon_t : idN(0, h_t)$$

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^{q} \eta_i d_{t-i} \varepsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j h_{t-j}$$

where $d_{t-i} = 1$ if $\varepsilon_{t-i} < 0$ and zero otherwise, the α_i and β_i coefficients must satisfy the same requirements as in *GARCH* models (see week 6, slide #13), and η_i is expected to be positive.

Given that $q = p = 1$,

$$E_{t-1}\left(h_t \mid \varepsilon_{t-1} \geq 0\right) = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 h_{t-1}$$

$$E_{t-1}\left(h_t \mid \varepsilon_{t-1} < 0\right) = \alpha_0 + (\alpha_1 + \eta_1) \varepsilon_{t-1}^2 + \beta_1 h_{t-1}$$

→ A significantly positive η_1-hat implies that negative shocks have greater effect on expected volatility than positive shocks.

d) Estimate a *TGARCH*(1,1) model with a constant mean equation for *DLNDAX*.

```
spec_v3 = ugarchspec(mean.model = list(armaOrder = c(0,0), include.mean = TRUE),
                     variance.model = list(model="fGARCH", submodel="TGARCH",
                                           garchOrder = c(1,1)),
                     distribution.model = "norm")
fit_v3 = ugarchfit(spec = spec_v3, data = DLNDAX)print(fit_v3)
```

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000307    0.000113   2.7264 0.006402
omega   0.000344    0.000044   7.8660 0.000000
alpha1  0.074508    0.007240  10.2905 0.000000
beta1   0.915468    0.008538 107.2171 0.000000
eta11   0.734277    0.057214  12.8339 0.000000
```

$$\widehat{DLNDAX}_t = 0.000307 + e_t \quad , \quad e_t \sim N(0, \hat{h}_t)$$

$$\hat{h}_t = 0.000344 + 0.074508 e_{t-1}^2 + 0.734277 d_{t-1} e_{t-1}^2 + 0.915468 \hat{h}_{t-1}$$

The estimate of the conditional variance for $e_{t-i} < 0$, $d = 1$ is

$$\hat{h}_t = 0.000344 + (0.074508 + 0.734277) e_{t-1}^2 + 0.915468 \hat{h}_{t-1}$$
$$= 0.000344 + 0.808785 e_{t-1}^2 + 0.915468 \hat{h}_{t-1}$$

while for $e_{t-i} > 0$, $d = 0$ it is

$$\hat{h}_t = 0.000344 + 0.074508 e_{t-1}^2 + 0.915468 \hat{h}_{t-1}$$

Without presenting the details, the rest of the printout suggests that

i. The robust standard errors are larger, but they do not make any coefficient insignificant.
ii. According to all four model specification criteria, the *TGARCH* model performs worse than the *GARCH* model, but it is better than the *IGARCH* model.
iii. The weighted *LB* tests do not detect autocorrelation.
iv. The weighted *ARCH LM* tests do not detect any remaining *ARCH* effect.
v. The Nyblom stability tests reject stability for *eta1*.
vi. The sign bias tests still detect some leverage effect.
vii. The adjusted Pearson tests reject normality.

Exponential-*GARCH* (*EGARCH*) model:

$$y_t = \mu_t + \varepsilon_t \qquad \varepsilon_t : idN(0, h_t)$$

$$\ln h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \frac{\varepsilon_{t-i}}{\sqrt{h_{t-i}}} + \sum_{i=1}^{q} \gamma_i \frac{|\varepsilon_{t-i}|}{\sqrt{h_{t-i}}} + \sum_{j=1}^{p} \beta_j \ln h_{t-j}$$

and it is logically expected that $\alpha_i + \eta_i > 0$ and $\alpha_i < 0$, so $\eta_i > 0$.

← Given that $q = p = 1$,

$$E_{t-1}(\ln h_t) = \alpha_0 + \alpha_1 \frac{\varepsilon_{t-1}}{\sqrt{h_{t-1}}} + \gamma_1 \frac{|\varepsilon_{t-1}|}{\sqrt{h_{t-1}}} + \beta_1 \ln h_{t-1}$$

→ The effect of a $\varepsilon_{t-1} > 0$ shock on the expected log volatility is $\alpha_1 + \gamma_1$ while that of $\varepsilon_{t-1} < 0$ is $-\alpha_1 + \gamma_1$, and the former is smaller than the latter if $\alpha_1 < 0$.

<u>Note</u>: The variance equation is in log-linear form. Consequently, no matter whether $\ln h_t$ is positive, negative or zero, h_t is always positive, and there is no need to impose any sign restriction on β_j.

e) Estimate an *EGARCH*(1,1) model with a constant mean equation for *DLNDAX*.

```
spec_v4 = ugarchspec(mean.model = list(armaOrder = c(0,0), include.mean = TRUE),
                     variance.model = list(model="eGARCH", garchOrder = c(1,1)),
                     distribution.model = "norm")
fit_v4 = ugarchfit(spec = spec_v4, data = DLNDAX)
print(fit_v4)
```

```
*---------------------------------*
*          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      0.000329    0.000093    3.5438 0.000394
omega  -0.203867    0.000674 -302.2640 0.000000
alpha1 -0.091653    0.004462  -20.5415 0.000000
beta1   0.976486    0.000155 6280.4241 0.000000
gamma1  0.121289    0.001190  101.9641 0.000000
```

$$\widehat{DLNDAX}_t = 0.000329 + e_t \quad , \quad e_t \sim N(0, \hat{h}_t)$$

$$\hat{h}_t = -0.203867 - 0.091653\frac{e_{t-1}}{\sqrt{\hat{h}_{t-1}}} + 0.121289\frac{|e_{t-1}|}{\sqrt{\hat{h}_{t-1}}} + 0.976486\ln\hat{h}_{t-1}$$

The point estimates are all significant and they satisfy $\alpha_1 + \gamma_1 = 0.02963 > 0$, $\alpha_1 = -0.091653 < 0$, and $\gamma_1 = 0.121289 > 0$.

Without presenting the details, the rest of the printout suggests that

i. The robust standard errors are smaller, so they do not make any qualitative difference.
ii. According to all four model specification criteria, the *EGARCH* model performs slightly worse than the *TGARCH* model.
iii. The weighted *LB* tests do not detect autocorrelation.
iv. The weighted *ARCH LM* tests do not detect any remaining *ARCH* effect.
v. The Nyblom stability tests reject stability for *alpha1*.
vi. The sign bias tests still detect some leverage effect.
vii. The adjusted Pearson tests reject normality.

- The *GARCH-in-mean* (*GARCH-M*) model allows the mean of $\{y_t\}$ to depend on its own conditional variance (or standard deviation).

  This approach is especially useful for modelling asset markets, where risk-averse agents are supposed to require risk premium, i.e., higher average returns, as compensation for holding a risky asset.

  → If y_t is the excess return from holding a risky asset,

$$y_t = \underbrace{\beta + \delta h_t}_{} + \varepsilon_t \quad , \quad \delta > 0$$

The expected risk premium is constant if the conditional variance is constant; otherwise, $E(y_t)$ is an increasing function of h_t.

… and h_t is a standard *GARCH*(q,p) process, i.e.,

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j h_{t-j}$$

y_t depends on conditional variance, $\delta > 0$.

f) Estimate a *GARCH-M*(1,1) model with a constant mean equation for *DLNDAX*.

```
spec_v5 = ugarchspec(mean.model = list(armaOrder = c(0,0),
                                       include.mean = TRUE, archm = TRUE, archpow = 2),
                     variance.model = list(model="sGARCH", garchOrder = c(1,1)),
                     distribution.model = "norm")
fit_v5 = ugarchfit(spec = spec_v5,  data = DLNDAX)
print(fit_v5)
```

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.000375    0.000177    2.1136 0.034554
archm   2.903712    1.158441    2.5066 0.012191
omega   0.000004    0.000001    4.4390 0.000009
alpha1  0.099579    0.006274   15.8707 0.000000
beta1   0.879628    0.007120  123.5366 0.000000
```

$$\widehat{DLNDAX}_t = 0.000375 + 2.903712\hat{h}_t + e_t$$

$$e_t \sim N(0, \hat{h}_t)$$

$$\hat{h}_t = 0.000004 + 0.099579 e_t^2 + 0.879628 \hat{h}_{t-1}$$

The point estimates are all significant at the 1.3% level.

Without presenting the details, the rest of the printout suggests that

i. The robust standard errors make *omega* insignificant.

ii. According to all four model specification criteria, this *GARCH-M* model outperforms only the *IGARCH* model.

iii. The weighted *LB* tests do not detect autocorrelation.

iv. The weighted *ARCH LM* tests do not detect any remaining *ARCH* effect.

v. The Nyblom stability tests do not detect instability.

vi. The sign bias tests detect some leverage effect.

vii. The adjusted Pearson tests reject normality.

All things considered, in this illustrative example none of the estimated models appears to perform well.

WHAT SHOULD YOU KNOW?

- Forecasting with *GARCH* models
- Extensions to the basic *GARCH* model: *IGARCH*, *TGARCH*, *EGARCH*, *GARCH-M*