ECON3350 - Applied Econometrics for Macroeconomics and Finance

Tutorial 5: Modelling Volatility - I

Tutor: Francisco Tavares Garcia

Report 1 is due next week!

Instructions

The project consists of three research questions. Please answer all questions as clearly, concisely and completely as possible. Each question is worth 50 marks, for a total of 150 marks. This report will constitute 20% of your overall grade in this course.

We suggest that you use R for all empirical work involved. However, you should be able to use another statistical software (e.g. Eviews, Stata, Python, etc.) without a problem. If you do choose to work with an alternative software, please note that support for software-specific issues from the course coordinator and tutors may be very limited.

Please upload your report via the "Turnitin" submission link (in the "Assessment / Research Report" folder). Please note that hard copies *will not* be accepted. At the moment, the due date is 1:00 PM on 28 March 2025, but please check BlackBoard regularly for announcements regarding any changes to this. Your report should be a write-up of your answers (in PDF format, single-spaced, and in 12 font size).[1]

You are allowed to work on this assignment with others, i.e., you can discuss how to answer the questions with your classmate(s) and even use AI as a research tool. However, this is not a group assignment, which means that the report must be written individually and by you: you must answer all the questions in your own words and submit your report separately. The marking system will check for similarities and AI content, and UQ's student integrity and misconduct policies on plagiarism *strictly apply*.

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

Report 1 – due 11 April - Question 1

Questions

The dataset for Questions 1 and 2 is contained in `report1.csv`. The variables are quarterly time-series of macroeconomic indicators in Australia for the period 1995Q1—2023Q4 (116 observations). In particular, the dataset contains the following variables:

1. Use the data provided to choose three (3) ARIMA(p, d, q) models for inflation, π_t. Use each of these three models to forecast π_t for 2023 and 2024 (two years or equivalently eight quarters past the end of the sample). In doing so, please consider how such forecasts may be useful for policy, and consequently, ensure your inference is aligned with the underlying motivation. Make sure to address all potential sources of uncertainty on a conceptual level, and to the extent possible, quantitatively.

Report 1 – due 11 April - Question 2

2. Use the data provided to obtain inference on the stability of the term structure of interest rates. In particular, investigate the following questions:

   (a) Is there evidence of nonstationarity in inflation, Δp_t, or in any of the following four interest rates $\{r_{M1,t}, r_{M3,t}, r_{Y2,t}, r_{Y3,t}\}$?

   (b) Are there any identifiable equilibrium relationships among the four interest rates?

   (c) Are each of the following spreads stationary?

   - $s_{t,m3-m1} = r_{M3,t} - r_{M1,t}$,
   - $s_{t,y2-3m} = r_{Y2,t} - r_{M3,t}$,
   - $s_{t,y3-y2} = r_{Y3,t} - r_{Y2,t}$, and
   - $s_{t,y5-r} = r_{Y5,t} - r_t$

   (d) Use a regression of Δp_t on $s_{t,y5-r}$ estimated by ols to investigate support for a relationship between these two.

In answering these questions, please consider how the answers may be useful for policy, and consequently, ensure your inference is aligned with the underlying motivation. Also, please discuss the key assumptions underlying any conclusions you obtain.

Report 1 – due 11 April - Question 3

3. The dataset for Question 3 is contained in `report2.csv`. The variables are daily time-series of two equity returns and a measure of market volatility for the period 29/06/2011—28/06/2021 (2541 observations, note the absence of weekends and holidays). The dataset contains the following variables:

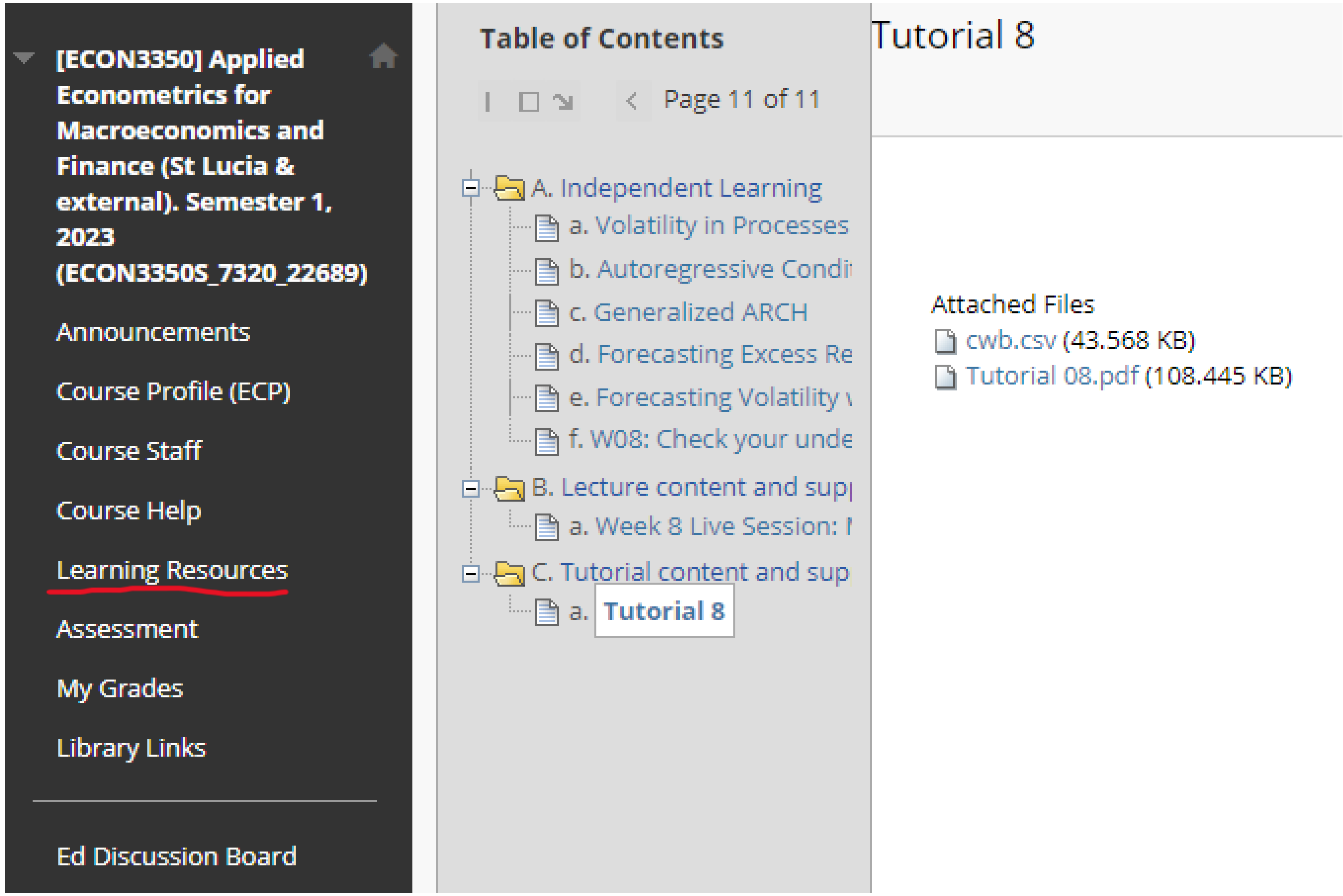
   (a) Use the data provided to obtain inference on the volatility of $r_{WES,t}$ and $r_{WPL,t}$. This should include discussion of any testing for the presence of volatility and model selection. Report only the important results that guide your conclusions, the estimated final model and estimated volatility for each process.

   (b) Compare and contrast the estimates of volatility from your models in part (a) to the $p_{VIX,t}$.

   (c) Investigate the probability of a return less than 0.01% for $r_{WES,t}$ and $r_{WBC,t}$ on each of the days 29/06/2021, 30/06/2021 and 1/07/2021.

In answering these questions, please consider how the answers may be useful for risk management, and consequently, ensure your inference is aligned with the underlying motivation. Also, please discuss the key assumptions underlying any conclusions you obtain.

Let’s download the tutorial and the dataset.



Now, let's download the script for the tutorial.

- Copy the code from Github,
  - https://github.com/tavaresgarcia/teaching
- Save the scripts in the same folder as the data.

Tutorial 5: Modelling Volatility - I

At the end of this tutorial you should be able to:

- use R to infer the presence of heteroscedasticity using the Breusch-Pagan test;
- construct an adequate set of models with possible GARCH errors;
- use R to estimate volatilities based on models with GARCH errors.

Problems

Consider the daily share prices of Commonwealth Bank (CWB) for the period 5 September 1996—30 August 2006 ($T = 2605$) in the data file `cwb.csv`. Let $\{y_t\}$ denote the process of share prices.

	A	B
1	date	y
2	9/05/1996	10.74
3	9/06/1996	10.75
4	9/09/1996	10.83
5	9/10/1996	10.95
6	9/11/1996	11.1
7	9/12/1996	11.24

Solution For this tutorial, we load the following useful packages.

```
library(forecast)
library(dplyr)
library(rugarch)
```

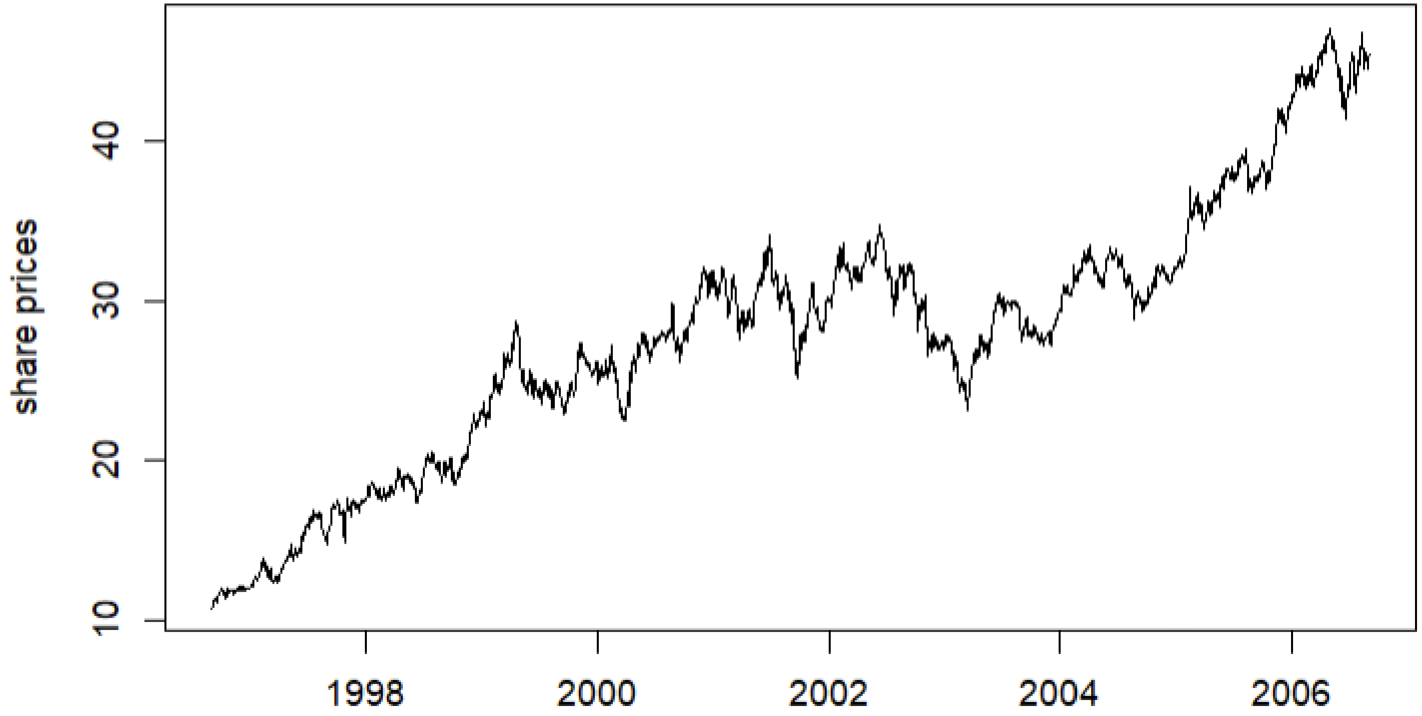
The package `rugarch` provides functions for working with conditional heteroscedasticity specifications. In this tutorial we will use the functions `ugarchspec` and `ugarchfit` to estimate volatilities based on standard GARCH models.

Next, load the data and and extract the variables.

```
mydata <- read.delim("cwb.csv", header = TRUE,  sep = ",")
```

1. Plot the *share prices* (y_t) and comment on the possible features of the DGP.

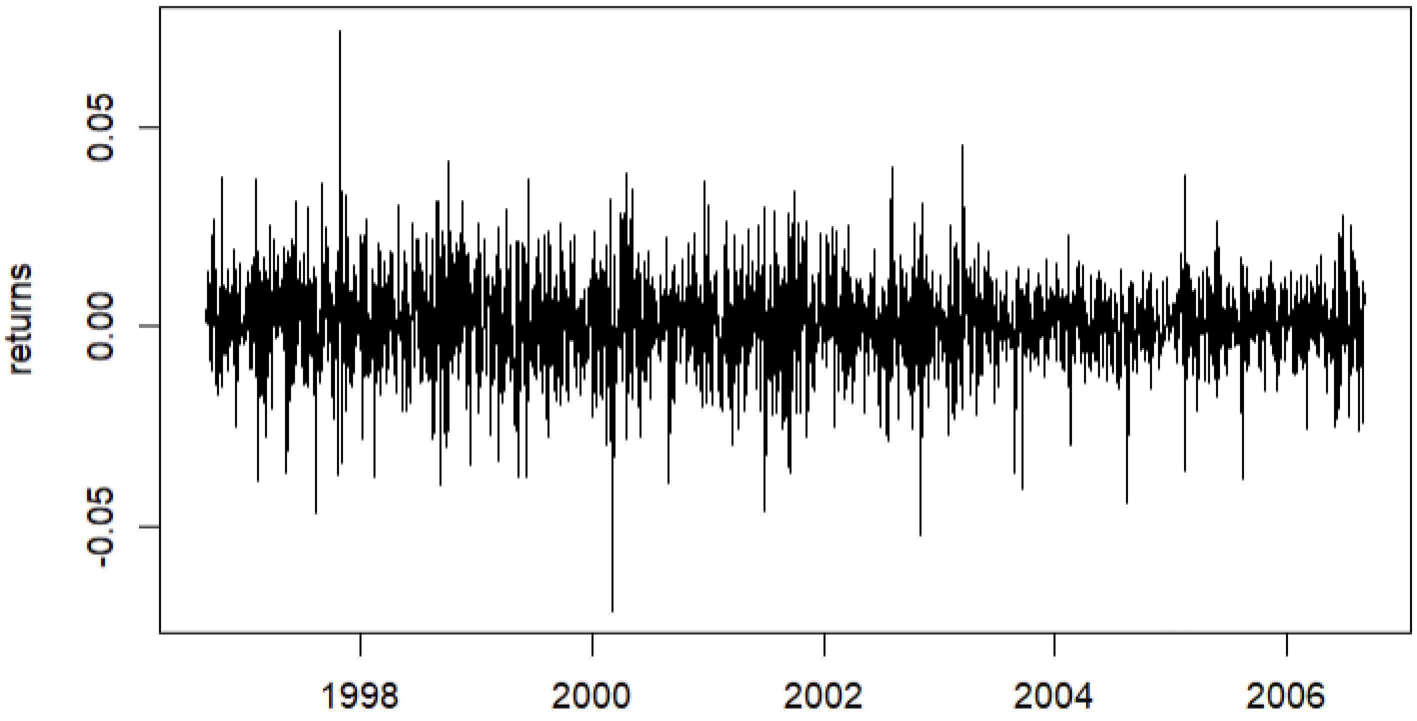
```
date <- as.Date(mydata$date, format = "%m/%d/%Y")
y <- mydata$y
r <- diff(log(y))
plot(date, y, type = "l", xlab = "", ylab = "share prices")
```



Share prices are clearly trending upwards, being around \$10 in the beginning of the sample and reaching about \$45 towards the end.

2. Plot the *returns* ($r_t = \ln y_t - \ln y_{t-1}$) and comment on the possible features of the DGP.

```
plot(date[-1], r, type = "l", xlab = "", ylab = "returns")
```



Returns look stable with possible heteroscedasticity. Observe that if returns are heteroscedastic then so must be the share prices. However, evidence of heteroscedasticity may be less obvious in plots of share prices than in plots of returns.

3. Assuming homoscedasticity, identify an adequate set of ARMA models for r_t.

Solution We use the familiar approach to building an adequate set of homoscedastic ARMA models.

```
ARMA_est <- list()
ic_arma <- matrix( nrow = 4 * 4, ncol = 4 )
colnames(ic_arma) <- c("p", "q", "aic", "bic")
for (p in 0:3)
{
  for (q in 0:3)
  {
    i <- p * 4 + q + 1
    ARMA_est[[i]] <- Arima(r, order = c(p, 0, q))
    ic_arma[i,] <- c(p, q, ARMA_est[[i]]$aic, ARMA_est[[i]]$bic)
  }
}
```

```
ic_aic_arma <- ic_arma[order(ic_arma[,3]),][1:10,]
ic_bic_arma <- ic_arma[order(ic_arma[,4]),][1:10,]

ic_int_arma <- intersect(as.data.frame(ic_aic_arma),
                         as.data.frame(ic_bic_arma))

adq_set_arma <- as.matrix(arrange(as.data.frame(
                                  rbind(ic_int_arma[c(1:3, 6),],
                                   ic_bic_arma[2,])), p, q))
adq_idx_arma <- match(data.frame(t(adq_set_arma[, 1:2])),
                      data.frame(t(ic_arma[, 1:2])))

nmods <- length(adq_idx_arma)
for (i in 1:nmods)
{
  checkresiduals(ARMA_est[[adq_idx_arma[i]]])
}
```

3. Assuming homoscedasticity, identify an adequate set of ARMA models for r_t.

```
        Ljung-Box test

data:  Residuals from ARIMA(0,0,1) with non-zero mean
Q* = 7.7683, df = 9, p-value = 0.5577

Model df: 1.   Total lags used: 10


        Ljung-Box test

data:  Residuals from ARIMA(0,0,2) with non-zero mean
Q* = 4.2044, df = 8, p-value = 0.8382

Model df: 2.   Total lags used: 10


        Ljung-Box test

data:  Residuals from ARIMA(1,0,0) with non-zero mean
Q* = 8.7674, df = 9, p-value = 0.459

Model df: 1.   Total lags used: 10


        Ljung-Box test

data:  Residuals from ARIMA(1,0,1) with non-zero mean
Q* = 3.4143, df = 8, p-value = 0.9057

Model df: 2.   Total lags used: 10


        Ljung-Box test

data:  Residuals from ARIMA(2,0,0) with non-zero mean
Q* = 4.632, df = 8, p-value = 0.7961

Model df: 2.   Total lags used: 10
```

	p	q	aic	bic
1	0	1	-15860.90	-15843.31
2	0	2	-15862.79	-15839.33
3	1	0	-15859.80	-15842.21
4	1	1	-15863.67	-15840.21
5	2	0	-15862.30	-15838.84

The adequate set consists of five models: MA(1), MA(2), AR(1), AR(2) and ARMA(1, 1).

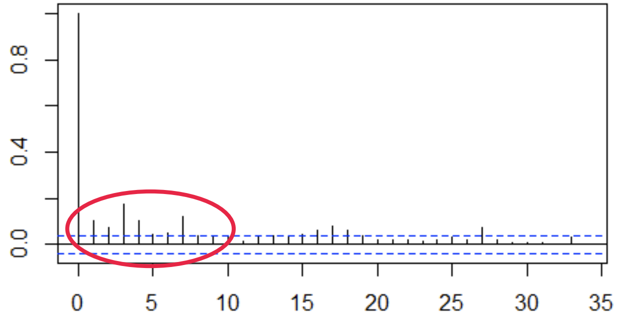
4. Generate estimated squared residuals for the set of models chosen in Question 3. Plot the squared residuals along with the sample ACFs. Interpret your findings.

Solution For each model in the adequate set, extract the estimated residuals using the `resid` function. Then, use `plot` and `acf` to obtain the desired graphs.

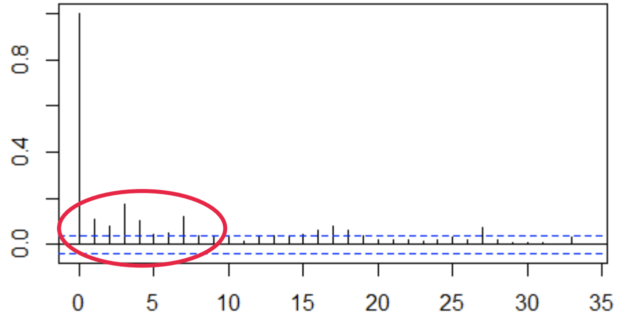
```
e2_arma <- list()
for (i in 1:nmods)
{
  e2_arma[[i]] <- resid(ARMA_est[[adq_idx_arma[i]]]) ^ 2
  title_p_q <- paste("ARMA(",
                     as.character(adq_set_arma[i, 1]), ", ",
                     as.character(adq_set_arma[i, 2]), ")",
                     sep = "")
  plot(date[-1], e2_arma[[i]], type = "l",
               xlab = "", ylab = "squared resid",
               main = paste("Plot: ", title_p_q))

  acf(e2_arma[[i]], xlab = "", ylab = "",
               main = paste("SACF: ", title_p_q))
}
```

Squared residuals appear to be strongly autocorrelated even though autocorrelation in the residual levels appears to be quite low. We interpret this as evidence of conditional heteroscedasticity possibly being present in the DGP.

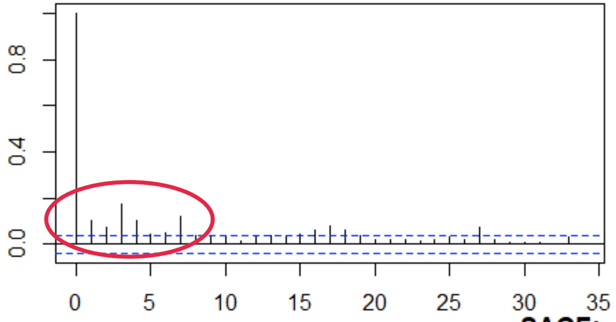
SACF: ARMA(0, 1)



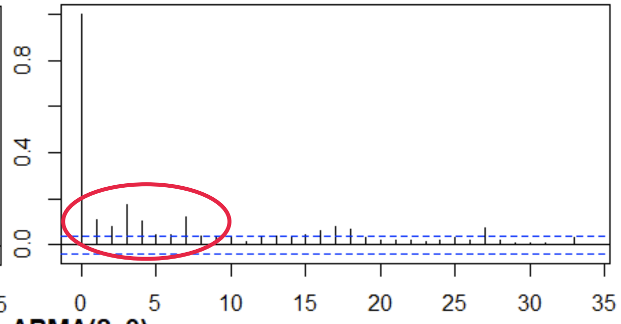
SACF: ARMA(0, 2)



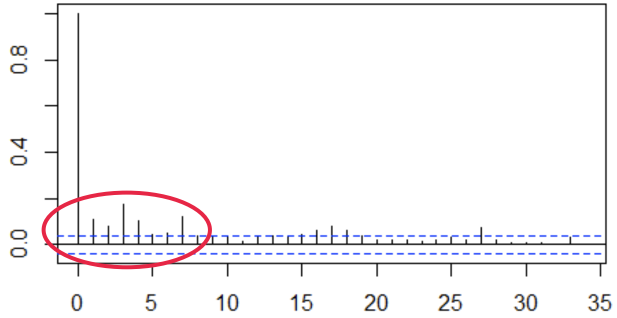
SACF: ARMA(1, 0)



SACF: ARMA(1, 1)



SACF: ARMA(2, 0)



5. Test if the errors in your set of models contain ARCH or GARCH effects.

Solution We use the Breush-Pagan test, which is a type of Lagrange-Multiplier (LM) test. The test is implemented for every model in the adequate set constructed in Question 3. It also requires a specification of j lags in the regression of squared residuals on lags. We will try lags ranging from 1 to 10. If there are discrepancies, we will need to delve deeper and construct an adequate set of B-P regressions for the test using the same approach we have learned based on information criteria and residuals analysis.

Note that the residual in the B-P regression involving squared residuals from the ARMA specification is assumed to be mean-independent. This is necessary to justify the χ^2 distribution of the LM test statistic. You should notice the connection to the corresponding role of mean-independence in the ADF test.

Also, the B-P regression is in fact an AR(j). However, because we need the R^2 from the B-P regression, it is easier to use the `lm` function in R rather than `Arima`.

```
bptest <- matrix(nrow = 10 * nmods, ncol = 5)
colnames(bptest) <- c("p", "q", "j", "LM-stat", "p-value")
for (i in 1:nmods)
{
  e2_i <- as.vector(e2_arma[[i]])
  f <- formula(e2_i ~ 1)
  for (j in 1:10)
  {
    # lag lengths in the auto-regression of squared residuals
    k <- 10 * (i - 1) + j
    f <- update.formula(f, paste("~ . + lag(e2_i, n = ", j, ")"));
    bp_reg_j <- lm(f)
    LM_j <- length(e2_i) * summary(bp_reg_j)$r.squared
    p_val_j <- 1 - pchisq(LM_j, df = j)
    bptest[k,] <- c(adq_set_arma[i, 1:2], j, LM_j, p_val_j)
  }
}
```

Breusch-Pagan test for heteroscedasticity: $H_0 : \alpha_1 = \cdots = \alpha_q = 0$, H_1 : ARCH(q).

1. **Estimate** the conditional mean using OLS and save residuals in $\{\widehat{\varepsilon}_t\}$.
2. **Regress** $\widehat{\varepsilon}_t^2 = \alpha_0 + \alpha_1 \widehat{\varepsilon}_{t-1}^2 + \cdots + \alpha_q \widehat{\varepsilon}_{t-q}^2$ and compute R_ε^2.
3. **Test** using $\text{LM}_{\text{ARCH}} \equiv T R_\varepsilon^2 \sim \chi_q^2$.

5. Test if the errors in your set of models contain ARCH or GARCH effects.

```
> bptest
      p q  j    LM-stat      p-value
 [1,] 0 1  1   27.47889 1.588182e-07
 [2,] 0 1  2   37.76870 6.289701e-09
 [3,] 0 1  3  106.99361 0.000000e+00
 [4,] 0 1  4  118.68557 0.000000e+00
 [5,] 0 1  5  118.87024 0.000000e+00
 [6,] 0 1  6  118.90678 0.000000e+00
 [7,] 0 1  7  140.08030 0.000000e+00
 [8,] 0 1  8  140.04199 0.000000e+00
 [9,] 0 1  9  140.20233 0.000000e+00
[10,] 0 1 10  140.46339 0.000000e+00
[11,] 0 2  1   30.18079 3.935886e-08
[12,] 0 2  2   42.29405 6.545801e-10
[13,] 0 2  3  108.87246 0.000000e+00
[14,] 0 2  4  121.08711 0.000000e+00
[15,] 0 2  5  121.23972 0.000000e+00
[16,] 0 2  6  121.26869 0.000000e+00
[17,] 0 2  7  140.99921 0.000000e+00
[18,] 0 2  8  140.96602 0.000000e+00
[19,] 0 2  9  141.10184 0.000000e+00
[20,] 0 2 10  141.31670 0.000000e+00
[21,] 1 0  1   27.02864 2.004632e-07
[22,] 1 0  2   37.35731 7.726137e-09
[23,] 1 0  3  107.15954 0.000000e+00
[24,] 1 0  4  118.88632 0.000000e+00
[25,] 1 0  5  119.07991 0.000000e+00
[26,] 1 0  6  119.11105 0.000000e+00
[27,] 1 0  7  140.20514 0.000000e+00
[28,] 1 0  8  140.16945 0.000000e+00
[29,] 1 0  9  140.33090 0.000000e+00
[30,] 1 0 10  140.58678 0.000000e+00
[31,] 1 1  1   29.29636 6.211212e-08
[32,] 1 1  2   41.53401 9.572017e-10
[33,] 1 1  3  105.86215 0.000000e+00
[34,] 1 1  4  118.54213 0.000000e+00
[35,] 1 1  5  118.72142 0.000000e+00
[36,] 1 1  6  118.72429 0.000000e+00
[37,] 1 1  7  138.79392 0.000000e+00
[38,] 1 1  8  138.76541 0.000000e+00
[39,] 1 1  9  138.89403 0.000000e+00
[40,] 1 1 10  139.11567 0.000000e+00
[41,] 2 0  1   30.17108 3.955645e-08
[42,] 2 0  2   41.99959 7.584133e-10
[43,] 2 0  3  109.69155 0.000000e+00
[44,] 2 0  4  121.82022 0.000000e+00
[45,] 2 0  5  121.96896 0.000000e+00
[46,] 2 0  6  122.01129 0.000000e+00
[47,] 2 0  7  141.74894 0.000000e+00
[48,] 2 0  8  141.71517 0.000000e+00
[49,] 2 0  9  141.84991 0.000000e+00
[50,] 2 0 10  142.06939 0.000000e+00
```

Breusch-Pagan test for heteroscedasticity:

$H_0 : \alpha_1 = \cdots = \alpha_q = 0,\ H_1 : \text{ARCH}(q).$

In specifications, the null hypothesis of homoscedastic errors is rejected in favor of ARCH/GARCH errors at very low significance levels. This suggests we should consider expanding our adequate set of models to include conditional heteroscedasticity.

THE UNIVERSITY OF QUEENSLAND AUSTRALIA

6. Expand the adequate set of models to specifications with heteroscedasticity. To this end, only consider conditional variances modelled with ARCH/GARCH residuals. Hint: Use the `rugarch` package with `ugarchspec` and `ugarchfit` functions.

Solution Based on the adquate set constructed with homodcedastic ARMAs, we begin by considering ARMA variants with $p_m = 0, \cdots, 2$ and $q_m = 0, ..., 2$ in the conditional mean along with GARCH lags $p_h = 0, \cdots, 2$ and $q_h = 0, ..., 2$ in the conditional variance.

There is a total of $3^4 = 81$ models altogether—it will take a few minutes to estimate them all! Also, some specifications will be so bad that numerical optimisation will fail, so it is helpful to use the `try` function to avoid interruptions.

```
ARMA_GARCH_est <- list()
ic_arma_garch <- matrix( nrow = 3 ^ 4, ncol = 6 )
colnames(ic_arma_garch) <- c("pm", "qm", "ph", "qh", "aic", "bic")
i <- 0
for (pm in 0:2)
{
  for (qm in 0:2)
  {
    for (ph in 0:2)
    {
      for (qh in 0:2)
      {
        i <- i + 1
        ic_arma_garch[i, 1:4] <- c(pm, qm, ph, qh)
```

```
        if (ph == 0 && qh == 0)
        {
          # for models with constant variance, the ugarchspec and
          # ugarchfit functions do not work well; instead, the
          # documentation advises to use arfimaspec and arfimafit
          ARMA_GARCH_mod <- arfimaspec(
            mean.model = list(armaOrder = c(pm, qm)))

          ARMA_GARCH_est[[i]] <- arfimafit(ARMA_GARCH_mod, r)

          ic_arma_garch[i,5:6] <- infocriteria(
                                    ARMA_GARCH_est[[i]])[1:2]
        }
        else
        {
          try(silent = T, expr =
          {
            ARMA_GARCH_mod <- ugarchspec(
                  mean.model = list(armaOrder = c(pm, qm)),
                  variance.model = list(garchOrder = c(ph, qh)))

            ARMA_GARCH_est[[i]] <- ugarchfit(ARMA_GARCH_mod, r,
                                             solver = 'hybrid')
```

6. Expand the adequate set of models to specifications with heteroscedasticity. To this end, only consider conditional variances modelled with ARCH/GARCH residuals. Hint: Use the `rugarch` package with `ugarchspec` and `ugarchfit` functions.

```
            ic_arma_garch[i,5:6] <- infocriteria(
                                    ARMA_GARCH_est[[i]])[1:2]
          })
        }
      }
    }
  }
}

ic_aic_arma_garch <- ic_arma_garch[
                     order(ic_arma_garch[,5]),][1:40,]
ic_bic_arma_garch <- ic_arma_garch[
                     order(ic_arma_garch[,6]),][1:40,]

ic_int_arma_garch <- intersect(as.data.frame(ic_aic_arma_garch),
                               as.data.frame(ic_bic_arma_garch))
```

There are *a lot* of models that have comparable AIC and BIC measures! We select the first 36 in the intersecting set.

```
adq_set_arma_garch <- as.matrix(arrange(as.data.frame(
                      ic_int_arma_garch[1:36,]), pm, qm, ph, qh))
adq_idx_arma_garch <- match(
                      data.frame(t(adq_set_arma_garch[, 1:4])),
                      data.frame(t(ic_arma_garch[, 1:4])))
```

Next, we check the residuals for specifications in the adequate set. Note that the standard Ljung-Box test is not well-defined for heteroscedastic models because the derivation of the sampling distribution assumes homoscedasticity. We will therefore avoid relying on the hypothesis test, but still examine the SACF for obvious for evidence of substantial autocorrelation.

```
nmods <- length(adq_idx_arma_garch)
sacf_garch <- matrix(nrow = nmods, ncol = 14)
colnames(sacf_garch) <- c("pm", "qm", "ph", "qh", 1:10)
for (i in 1:nmods)
{
  sacf_garch[i,1:4] <- adq_set_arma_garch[i,1:4]
  sacf_garch[i,5:14] <-
                acf(ARMA_GARCH_est[[adq_idx_arma_garch[i]]]@fit$z,
                lag = 10, plot = F)$acf[2:11]
}
```

Residual autocorrelations appear to be relatively small for all models in the adquate set.

6. Expand the adequate set of models to specifications with heteroscedasticity. To this end, only consider conditional variances modelled with ARCH/GARCH residuals. Hint: Use the `rugarch` package with `ugarchspec` and `ugarchfit` functions.

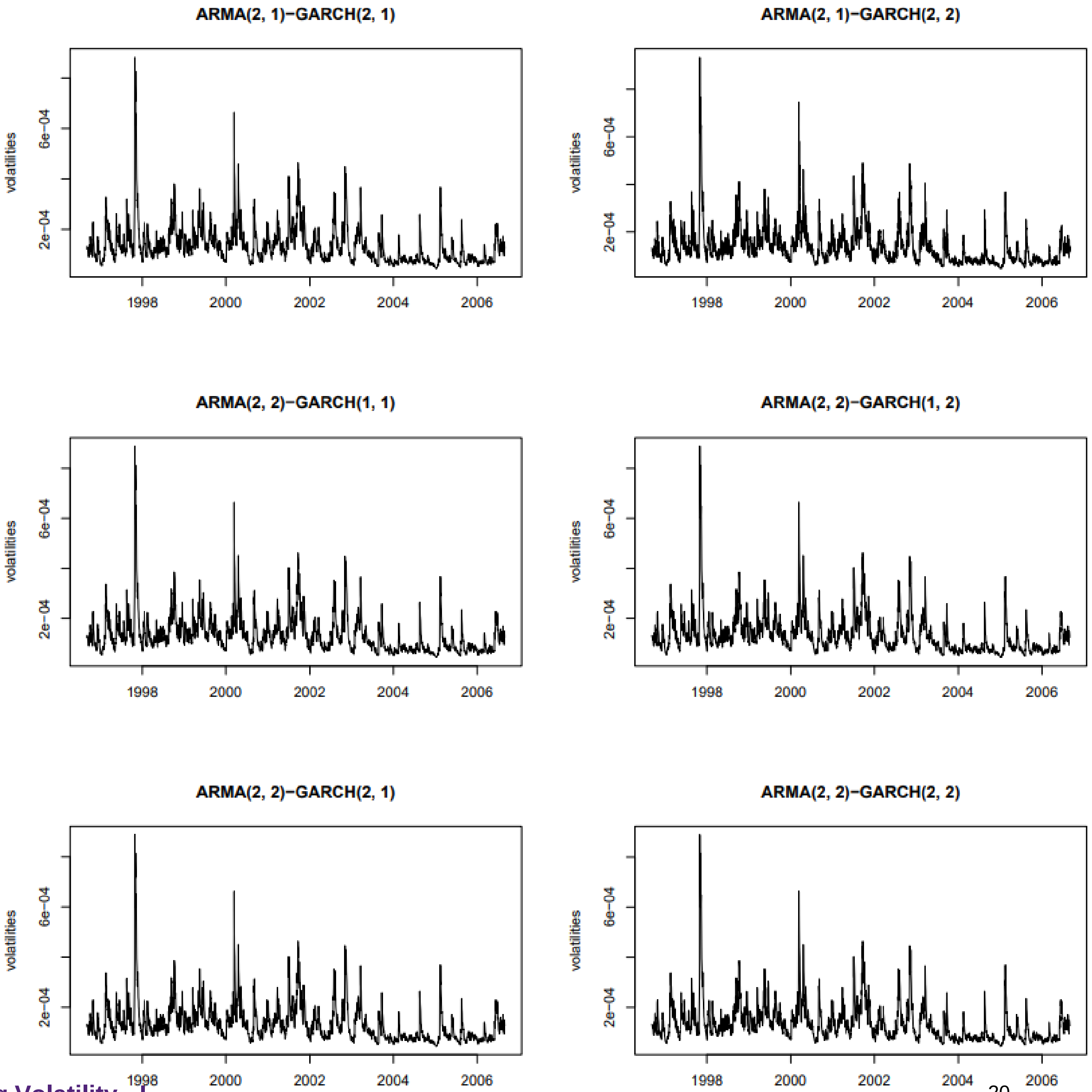
	pm	qm	ph	qh	1	2	3	4	5	6	7	8	9	10
1	0	0	1	1	0.080615454	-0.0248654847	0.004857558	-0.016930061	-0.006749342	-0.004187928	0.02715488	0.0006352833	0.01998621	0.01658524
2	0	0	1	2	0.080474149	-0.0249294169	0.004826002	-0.016922911	-0.006689254	-0.004156120	0.02724194	0.0006359106	0.01994307	0.01655781
3	0	0	2	1	0.080523790	-0.0249246782	0.004830379	-0.016913886	-0.006720514	-0.004162020	0.02722005	0.0006631367	0.01997585	0.01656224
4	0	0	2	2	0.080482182	-0.0249189508	0.004841645	-0.016907114	-0.006707058	-0.004158543	0.02722731	0.0006406836	0.01995563	0.01655541
5	0	1	1	1	0.009449754	-0.0268323402	0.008076573	-0.018222257	-0.004903785	-0.005393951	0.02706520	-0.0030019729	0.01934716	0.01647291
6	0	1	1	2	0.003473560	-0.0237859433	0.008629754	-0.017605930	-0.003091956	-0.004864554	0.02620156	-0.0061520688	0.01816856	0.01563880
7	0	1	2	1	0.009425841	-0.0268862753	0.008054388	-0.018198231	-0.004879847	-0.005375295	0.02712748	-0.0029775588	0.01933526	0.01645148
8	0	1	2	2	0.005634878	-0.0250179591	0.008099166	-0.017565618	-0.003746803	-0.004656915	0.02612422	-0.0056539717	0.01871211	0.01690101
9	0	2	1	1	0.013477139	0.0103881441	0.005662666	-0.017921984	-0.003148064	-0.006093518	0.02763679	-0.0024260814	0.02020789	0.01540749
10	0	2	1	2	0.012983659	0.0105276155	0.005765886	-0.017848316	-0.003044394	-0.006102749	0.02760864	-0.0026523587	0.02007500	0.01531045
11	0	2	2	1	0.013483138	0.0103851839	0.005656458	-0.017929474	-0.003140383	-0.006092689	0.02764221	-0.0024259979	0.02020499	0.01540907
12	0	2	2	2	0.010195787	0.0095593492	0.005633165	-0.017334245	-0.002332554	-0.005427437	0.02668520	-0.0049785680	0.01950636	0.01585315
13	1	0	1	1	0.015702342	-0.0319889935	0.007709074	-0.017896033	-0.005350210	-0.005182616	0.02699921	-0.0027400378	0.01929317	0.01663290
14	1	0	1	2	0.009915171	-0.0301429282	0.008267232	-0.017237613	-0.003579699	-0.004584744	0.02609792	-0.0059032538	0.01810394	0.01585452
15	1	0	2	1	0.015722209	-0.0320342193	0.007672808	-0.017881475	-0.005324133	-0.005163712	0.02706394	-0.0027048929	0.01928737	0.01661502
16	1	0	2	2	0.012102390	-0.0309688442	0.007722937	-0.017217820	-0.004229222	-0.004406866	0.02602236	-0.0053890824	0.01867928	0.01708916
17	1	1	1	1	0.011835953	0.0010122317	-0.005122116	-0.013232333	-0.006081715	-0.004904648	0.02675510	-0.0023229624	0.01988978	0.01568714
18	1	1	1	2	0.007136584	0.0028999391	-0.002949068	-0.013754060	-0.004041436	-0.004908959	0.02654393	-0.0049689260	0.01878145	0.01479802

7. Estimate each model in the set identified in Question 6 and plot the estimated volatilities ($\hat{h}_t$). Interpret the results.

Solution We have already estimated all the models in the adequate set, so the only thing left to do is plot the estimated volatilities. Unfortunately, the `rugarch` package *does not* provide standard errors or confidence intervals for estimated volatilities. We would need to write a customised function to compute these (left as practice... or for an advanced econometrics course)!

```
for (i in 1:nmods)
{
  title_p_q <- paste("ARMA(",
                     as.character(adq_set_arma_garch[i, 1]), ", ",
                     as.character(adq_set_arma_garch[i, 2]),
                     ")-GARCH(",
                     as.character(adq_set_arma_garch[i, 3]), ", ",
                     as.character(adq_set_arma_garch[i, 4]), ")",
                     sep = "")
  plot(date[-1], ARMA_GARCH_est[[adq_idx_arma_garch[i]]]@fit$var,
                     type = "l", xlab = "", ylab = "volatilities",
                     main = title_p_q)
}
```

We notice some subtle differences across these specifications, but overall the estimated volatilities over the sample period follow a similar pattern. Hence, even though our adequate set contains many models, the implied high degree of model specification uncertainty is of little practical consequence for inference on volatilities. Put in a positive light: the inference we obtain regarding volatilities is robust to model specification uncertainty.



Tutorial 5: Modelling Volatility - I

At the end of this tutorial you should be able to:

- use R to infer the presence of heteroscedasticity using the Breusch-Pagan test;
- construct an adequate set of models with possible GARCH errors;
- use R to estimate volatilities based on models with GARCH errors.

Thank you

Francisco Tavares Garcia
Academic Tutor | School of Economics
tavaresgarcia.github.io

Reference

Tsay, R. (2010). Analysis of Financial Time Series, 3rd Edition, John Wiley & Sons.