

ECOM90024 - Assignment 3 - Question 3

Josh Copeland

2024-05-22

```
# Packages used for this assignment

# Library(tidyverse)
# Library(Lubridate)
# Library(forecast)
# Library(urca)
# Library(tseries)
# Library(rugarch)
# Library(janitor)
```

Question 3

The file “aapl.csv” contains observations of the daily closing price of Apple from 15/05/2015 to 15/05/2019. Using *R* you are required to do the following:

- Generate the daily returns of Apple’s stock price as the log difference of the daily closing price. Plot the data and provide a brief description of the observed time series.

Chart 1 shows a chart of Apple’s daily returns from 2015-05-15 to 2019-05-14. The mean of this series looks to be constant over time, but there is clear evidence of volatility clustering throughout this period. Notably, this occurs in late 2015, as well as early and late 2018 to call out the biggest examples (but there are other smaller episode throughout this history).

This volatility clustering suggests this time series has a conditional variance, where a (G)ARCH specification would likely aid in modelling these returns. However, more analysis is required to confirm this.

```
##### IMPORTING DATA #####
```

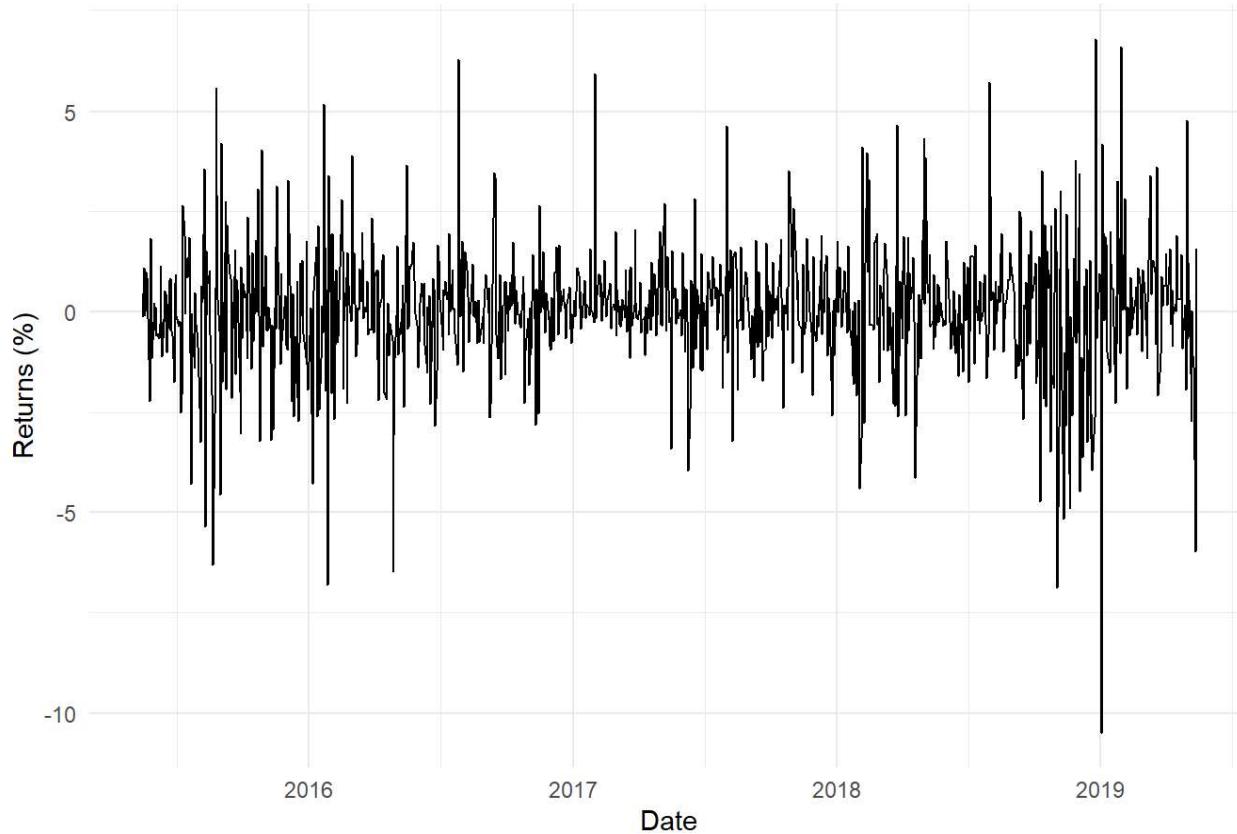
```
data <- read_csv("aapl.csv") %>%
  select(date, price = aapl) %>%
  mutate(
    date = dmy(date),
    returns = (log(price) - lag(log(price))) * 100
  ) %>%
  na.omit()
```

```
## Rows: 1007 Columns: 2
## └─ Column specification ──────────────────────────────────────────
##   ## Delimiter: ","
##   ## chr (1): date
##   ## dbl (1): aapl
## 
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
##### CHARTING DATA #####
```

```
ggplot(data, aes(date, returns)) +
  geom_line() +
  ggtitle("Chart 1: Daily Apple log returns from 2015-05-15 to 2019-05-14") +
  ylab("Returns (%)") +
  xlab("Date")
```

Chart 1: Daily Apple log returns from 2015-05-15 to 2019-05-14



b) Using the methodology outlined in previous lectures, specify and estimate an appropriate and parsimonious model for the conditional mean of the daily returns. Do not include any deterministic trend or seasonal dummies in your specification.

- As we have been instructed to not include any deterministic components to our model, we begin by testing for a unit root in this series.
 - We use the “none” option for our ADF test because there is no drift to this series: it centres at zero.
 - After determining the optimal number of lags for our regression (0), we complete the ADF test on the level. The test statistic is generated and is equal to -31.17, which is much greater than any of the listed critical values. Therefore, we reject the null hypothesis of this series having a unit root and conclude it is I(0).

```
##### SPECIFYING THE NUMBER OF MAX ADF LAGS #####
```

```
kmax <- ceiling(12*(length(data$price_level)/100)^(1/4))
```

```
## Warning: Unknown or uninitialized column: `price_level`.
```

```
##### ADF TEST ON LEVEL SERIES #####
```

```
for (i in 1:(kmax-1)){  
  
  k = kmax+1-i  
  
  df_data_level = ur.df(data$returns, type = "none", lags = k)  
  
  tstat = df_data_level@testreg$coefficients[k+1,3]  
  
  if(abs(tstat) >= 1.6){break}  
  
}
```

```
k  
  
## [1] 0
```

```
# SUMMARY OF ADF TEST
```

```
summary(df_data_level)
```

```

## #####
## # Augmented Dickey-Fuller Test Unit Root Test #
## #####
##
## Test regression none
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 - 1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.4942  -0.5877   0.0615   0.8331   6.8467
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## z.lag.1 -0.98417    0.03157  -31.17   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.582 on 1004 degrees of freedom
## Multiple R-squared:  0.4918, Adjusted R-squared:  0.4913
## F-statistic: 971.8 on 1 and 1004 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -31.1732
##
## Critical values for test statistics:
##      1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62

```

Conclusion: The test statistic is very Large (~-31). Therefore, we can reject the null hypothesis at any reasonable significance level and conclude this series is I(0).

- Next we need to confirm our series is not white noise before specifying any mean equation with an ARIMA model. If they are white noise, this series cannot be usefully modelled using such a process.
 - Using our portmanteau tests below, we conclude it is not suitable to use an ARIMA process to model the conditional mean of this series.
 - This is because the p-values from both our tests are both very large (much greater than 0.05), meaning we cannot reject the null hypothesis of there being no autocorrelation in the time series up to 32 lags (m) at any reasonable significance level. This means these series are effectively white noise, and cannot be usefully modelled as an ARIMA process..

```
m <- ceiling((sqrt(length(data$returns))))
```

```
m
```

```
## [1] 32
```

```
Box.test(data$returns, lag = m, type = "Box-Pierce")
```

```
##  
## Box-Pierce test  
##  
## data: data$returns  
## X-squared = 39.28, df = 32, p-value = 0.176
```

```
Box.test(data$returns, lag = m, type = "Ljung-Box")
```

```
##  
## Box-Ljung test  
##  
## data: data$returns  
## X-squared = 40.049, df = 32, p-value = 0.1552
```

- Because we've concluded these series are effectively white noise, we can skip generating the ACF and PACF to determine its dependence structure (there is none). From this we generate the conditional mean equation as an ARIMA(0,0,0)

```
mean_eqn <- Arima(data$returns,  
                    order = c(0,0,0),  
                    include.mean = FALSE,  
                    method = "CSS-ML")
```

c) Using your estimates from part b, compute and plot h-step ahead dpoint and 95% interval forecasts for the daily returns for $h = 1, 2, \dots, 10$.

Chart 2 below charts the point estimate of our 10-step ahead forecast. For ease of viewing, we have only included observations from the beginning of 2019 onwards. The point estimate forecast is zero for all forecast period, this is because the expected mean of a white noise series is unconditionally zero unless there is a mean to the series. Our mean is zero. The prediction intervals reflect the normal distribution of the white noise error term.

```
#####
##### FORECASTING VALUES #####
mean_fc <- forecast(mean_eqn, h = 10) %>%
  data.frame() %>%
  select(point_estimate = Point.Forecast, low_prediction_interval = Lo.95, high_prediction_in-
terval = Hi.95)

#####
##### CREATING DATAFRAME FOR FORECASTING #####
data_fc <- data.frame(date = seq(from = max(data$date) + days(1), by = "days", length.out = 1
0)) %>%
  cbind(mean_fc)

#####
##### COMBINING TOGETHER FOR CHART #####
data_final <- data %>%
  full_join(data_fc, by = "date") %>%
  filter(date >= "2019-01-01")

#####
##### CHART #####
mean_eqn_fc <- ggplot(data_final, aes(x = date)) +
  geom_line(aes(y = returns)) +
  geom_line(aes(y = point_estimate, colour = "Point estimate")) +
  geom_line(aes(y = high_prediction_interval, colour = "Prediction interval")) +
  geom_line(aes(y = low_prediction_interval, colour = "Prediction interval")) +
  scale_colour_manual(name = "Forecast", values = c("Point estimate" = "red", "Prediction i-
nterval" = "blue")) +
  ggtitle("Chart 2: Daily Apple log returns from 2019-01-01 with forecast")+
  ylab("Returns (%)") +
  xlab("Date") +
  theme(plot.title = element_text(size = 10),
        plot.caption = element_text(hjust = 0),
        legend.position = c(0.1, 0.4),
        legend.justification = c(0, 1))
```

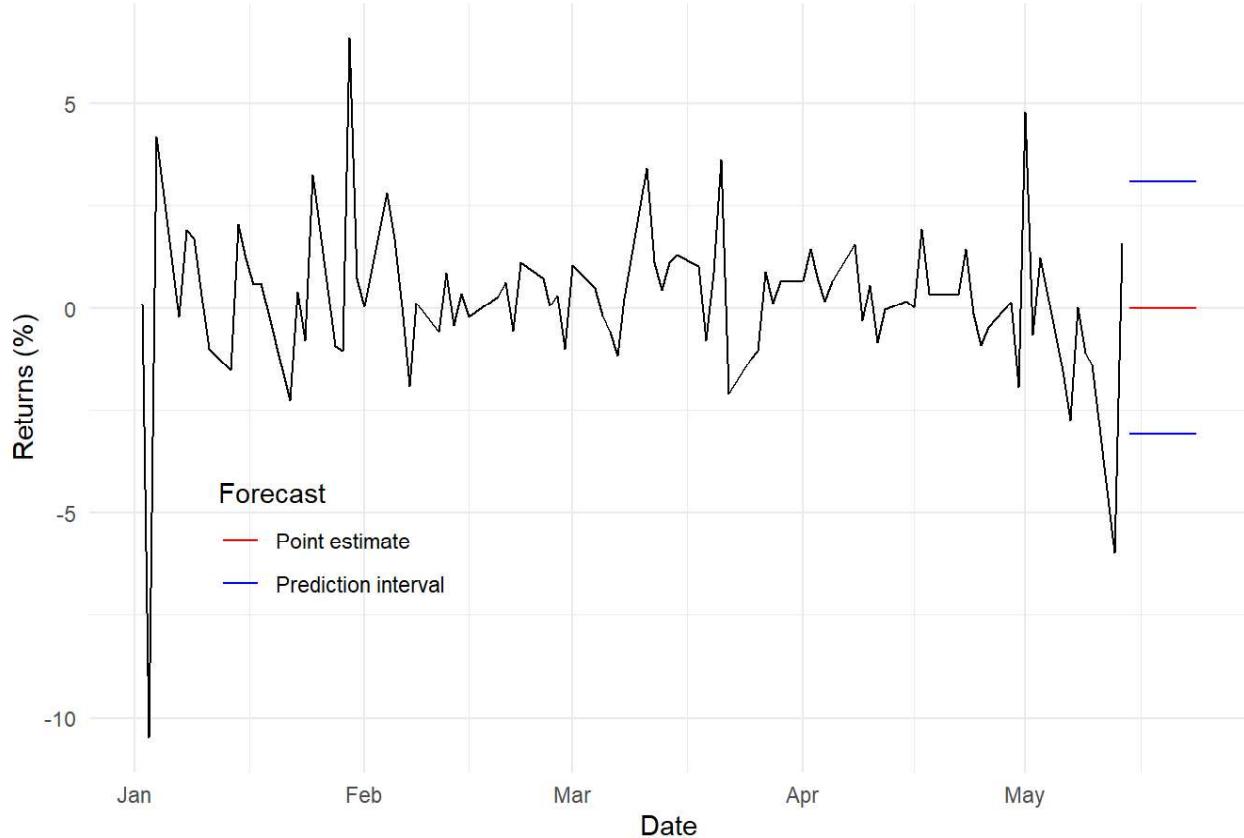
```
## Warning: A numeric `legend.position` argument in `theme()` was deprecated in ggplot2
## 3.5.0.
## i Please use the `legend.position.inside` argument of `theme()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
mean_eqn_fc
```

```
## Warning: Removed 10 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

```
## Warning: Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

Chart 2: Daily Apple log returns from 2019-01-01 with forecast

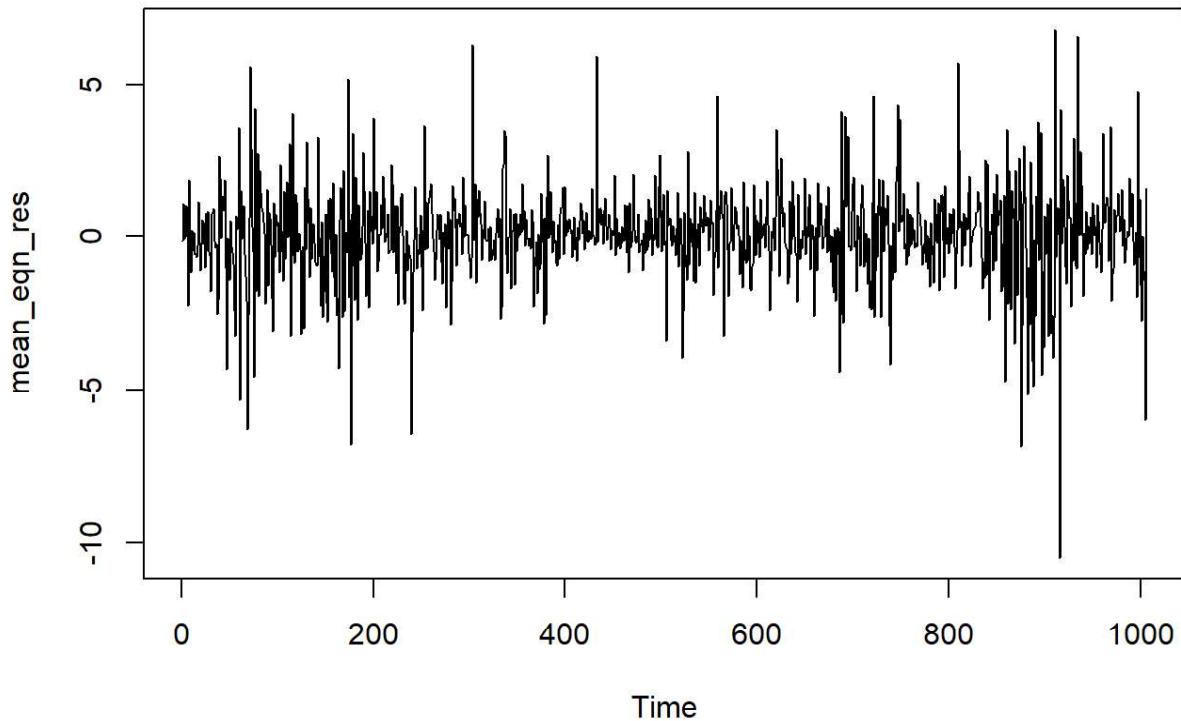


d) Using the method described in lecture 9, test for the presence of ARCH effects in the residuals obtained from the conditional mean estimated in part (b).

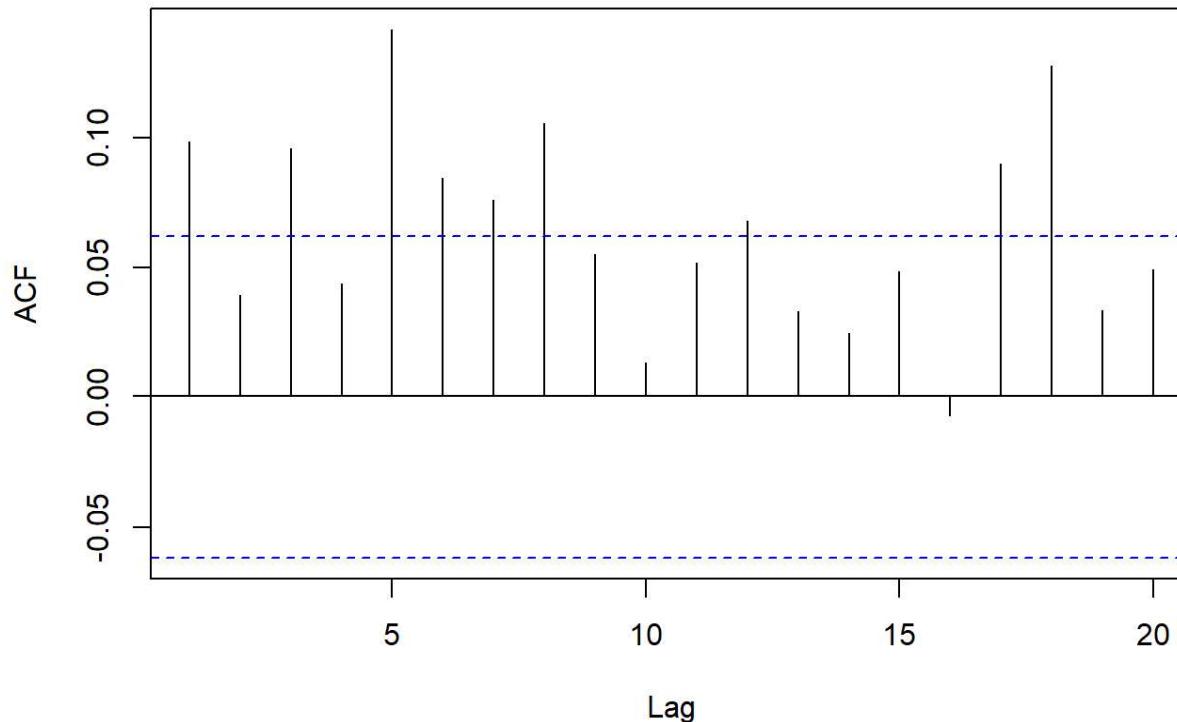
- Before estimating, it's worth noting that Chart 3 indicates residuals with high degrees of volatility clustering. This just reflects the returns series as we did not actually model the mean with an ARIMA(0,0,0).
- Chart 4 and Chart 5 show the ACF and PACF respectively of the squared residuals from our conditional mean equation, both of which show many significant lags, suggesting that ARCH effects are present.
- This is confirmed by the portmanteau tests below on the squared residuals from our conditional mean equation, which confirm these series are not white noise, and that there is indeed dependence in the squared residuals of our conditional mean equation. This justifies us pursuing the estimation of an ARCH/GARCH model.

```
mean_eqn_res <- mean_eqn$resid  
  
plot(mean_eqn_res, main = "Chart 3: Residuals from ARIMA(0,0,0) model")
```

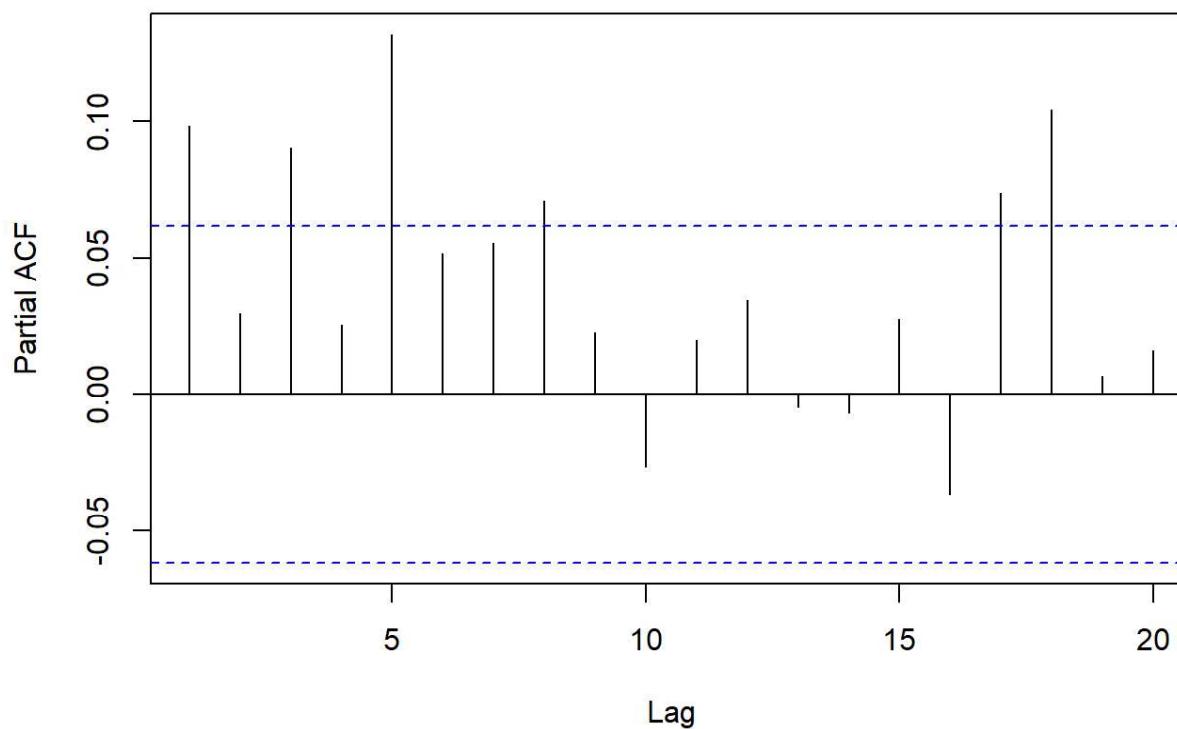
Chart 3: Residuals from ARIMA(0,0,0) model



```
mean_eqn_res_sq <- mean_eqn_res^2  
  
acf.res <- acf(mean_eqn_res_sq, plot = FALSE)  
pacf.res <- pacf(mean_eqn_res_sq, plot = FALSE)  
  
plot(acf.res[1:20], main = "Chart 4: ACF of squared residuals from ARIMA(0,0,0) model")
```

Chart 4: ACF of squared residuals from ARIMA(0,0,0) model

```
plot(pacf.res[1:20], main = "Chart 5: PACF of squared residuals from ARIMA(0,0,0) model")
```

Chart 5: PACF of squared residuals from ARIMA(0,0,0) model

```
Box.test(mean_eqn_res_sq, type = "Box-Pierce", l = m)
```

```
##  
## Box-Pierce test  
##  
## data: mean_eqn_res_sq  
## X-squared = 139.68, df = 32, p-value = 2.109e-15
```

```
Box.test(mean_eqn_res_sq, type = "Ljung-Box", l = m)
```

```
##  
## Box-Ljung test  
##  
## data: mean_eqn_res_sq  
## X-squared = 141.74, df = 32, p-value = 8.882e-16
```

e) Estimate an appropriate ARCH/GARCH model for the innovations. Verify that your specification is adequate.

- From Chart 4 above, it can be seen there are significant ACF lags up to lag 18. Therefore, we will test for model with ARCH terms up to order 18. Note, estimating these models produces many pages of output. I have reduced the amount of this output for the pdf for brevity.
- AIC and BIC disagree on which model to use (ARCH(12) and ARCH(8)) respectively. We will go with the more parsimonious ARCH(8) model.
- After estimating the ARCH(8) model we re-examine the residuals and note the residuals now look much more like white noise (Chart 6). This is confirmed by our portmanteau tests below in the squared standardised squared residuals. Given their large p-values (>0.05), we can confirm the residuals from this model are now white noise.
- Therefore, we conclude this ARCH(8) model successfully accounts for the ARCH effects in this return series.
- However, we also test to see if a more parsimonious GARCH(1,1) sufficiently arrives at the same solution and instead prefer this specification: the residuals also look very similar and like white noise (Chart 8) and the portmanteau tests also confirm this is the case for the same reason as above: large p-values which tell us these residuals are white noise.
- Because the GARCH(1,1) achieves the same outcome as the ARCH(8) more parsimoniously, we adopt this model instead as our preferred model.

```
##### ESTIMATING ARCH MODELS #####
z = 18

prefix.arch <- "arch"
suffix.arch <- seq(1,z)

arch.label <- paste(prefix.arch, suffix.arch, sep = " ")

arch.info <- data.frame(row.names = arch.label, matrix(NA,z,2))

colnames(arch.info) <- c("aic","bic")

T <- length(mean_eqn_res)

for(i in 1:z){
  archmod <- garch(mean_eqn_res, order = c(0,i))
  arch.info[i,1] <- AIC(archmod)
  arch.info[i,2] <- AIC(archmod, k = log(T-i-1))
}
```

```

##  

## ***** ESTIMATION WITH ANALYTICAL GRADIENT *****  

##  

##  

##      I      INITIAL X(I)          D(I)  

##  

##      1      2.373799e+00      1.000e+00  

##      2      5.000000e-02      1.000e+00  

##  

##      IT      NF      F          RELDF      PRELDF      RELDX      STPPAR      D*STEP      NPRELDF  

##      0      1      9.549e+02  

##      1      4      9.534e+02      1.59e-03      9.61e-03      2.4e-02      6.8e+02      1.2e-01      3.27e+00  

##      2      5      9.516e+02      1.90e-03      2.02e-03      2.5e-02      2.0e+00      1.2e-01      1.08e-01  

##      3      6      9.512e+02      4.17e-04      6.81e-04      2.6e-02      1.7e+00      1.2e-01      4.45e-03  

##      4      8      9.512e+02      2.64e-05      4.74e-05      2.1e-03      1.1e+01      9.1e-03      2.48e-04  

##      5      9      9.512e+02      2.53e-07      2.86e-07      6.3e-04      0.0e+00      2.9e-03      2.86e-07  

##      6      10     9.512e+02      3.55e-09      3.55e-09      6.9e-05      0.0e+00      3.1e-04      3.55e-09  

##      7      11     9.512e+02      9.92e-15      1.11e-14      9.6e-08      0.0e+00      4.1e-07      1.11e-14  

##  

## ***** RELATIVE FUNCTION CONVERGENCE *****  

##  

##  

##      FUNCTION      9.511546e+02      RELDX      9.585e-08  

##      FUNC. EVALS      11      GRAD. EVALS      8  

##      PRELDF      1.114e-14      NPRELDF      1.114e-14  

##  

##      I      FINAL X(I)          D(I)          G(I)  

##  

##      1      2.141265e+00      1.000e+00      1.497e-06  

##      2      1.580069e-01      1.000e+00      4.321e-06  

##  

##  

## ***** ESTIMATION WITH ANALYTICAL GRADIENT *****  

##  

##  

##      I      INITIAL X(I)          D(I)  

##  

##      1      2.248863e+00      1.000e+00  

##      2      5.000000e-02      1.000e+00  

##      3      5.000000e-02      1.000e+00  

##  

##      IT      NF      F          RELDF      PRELDF      RELDX      STPPAR      D*STEP      NPRELDF  

##      0      1      9.496e+02  

##      1      4      9.478e+02      1.81e-03      8.91e-03      1.8e-02      9.4e+02      9.5e-02      4.17e+00  

##      2      5      9.461e+02      1.89e-03      2.05e-03      2.1e-02      2.0e+00      9.5e-02      1.68e-01  

##      3      6      9.456e+02      5.07e-04      1.58e-03      1.6e-02      9.8e+00      9.5e-02      2.38e-01  

##      4      7      9.448e+02      8.72e-04      1.30e-03      1.7e-02      2.0e+00      9.5e-02      3.55e-02  

##      5      8      9.446e+02      2.03e-04      7.86e-04      2.1e-02      1.9e+00      9.5e-02      6.66e-03  

##      6      10     9.444e+02      1.73e-04      4.14e-04      8.8e-03      2.1e+00      3.5e-02      3.77e-03  

##      7      12     9.443e+02      8.06e-05      8.25e-05      2.2e-03      2.1e+00      1.3e-02      2.58e-04  

##      8      13     9.443e+02      8.47e-06      1.28e-05      3.2e-03      1.8e+00      1.3e-02      1.26e-04  

##      9      15     9.443e+02      7.17e-07      4.53e-06      1.2e-03      1.9e+00      4.8e-03      1.32e-05  

##      10     16     9.443e+02      7.21e-08      3.49e-06      9.7e-04      1.3e+00      4.8e-03      4.32e-06  

##      11     17     9.443e+02      2.26e-06      2.55e-06      6.1e-04      2.5e+00      2.4e-03      1.10e-05  

##      12     19     9.443e+02      2.89e-07      4.60e-07      2.4e-04      3.5e+00      1.0e-03      1.93e-06  

##      13     20     9.443e+02      9.62e-08      1.12e-07      2.5e-04      1.4e+00      1.0e-03      2.45e-07

```

```

##  

## ***** FALSE CONVERGENCE *****  

##  

## FUNCTION      9.089316e+02    RELDX      1.155e-14  

## FUNC. EVALS     87          GRAD. EVALS     39  

## PRELDF       3.363e-15    NPRELDF   -1.214e-01  

##  

##  

## I      FINAL X(I)        D(I)        G(I)  

##  

## 1      4.231184e-01    1.000e+00    -6.170e+01  

## 2      9.014490e-02    1.000e+00    -1.917e+01  

## 3      6.016252e-02    1.000e+00    -1.311e+00  

## 4      8.022037e-02    1.000e+00    -3.160e+01  

## 5      7.178485e-02    1.000e+00    -1.878e+01  

## 6      9.550914e-02    1.000e+00    -3.707e+01  

## 7      6.399412e-02    1.000e+00    -1.671e+00  

## 8      6.533758e-02    1.000e+00    -1.327e+01  

## 9      7.698626e-02    1.000e+00    -1.749e+01  

## 10     7.851005e-02    1.000e+00    -1.721e+01  

## 11     3.968291e-02    1.000e+00    2.891e+01  

## 12     3.383903e-02    1.000e+00    3.005e+01  

## 13     1.883743e-02    1.000e+00    4.963e+01  

## 14     1.632684e-02    1.000e+00    3.962e+01  

## 15     3.416687e-02    1.000e+00    3.784e+01  

## 16     2.510333e-15    1.000e+00    4.293e+01  

## 17     3.542141e-02    1.000e+00    4.222e+01  

## 18     8.736352e-02    1.000e+00    -2.845e+01  

##  

##  

## ***** ESTIMATION WITH ANALYTICAL GRADIENT *****  

##  

##  

## I      INITIAL X(I)        D(I)  

##  

## 1      2.498736e-01    1.000e+00  

## 2      5.000000e-02    1.000e+00  

## 3      5.000000e-02    1.000e+00  

## 4      5.000000e-02    1.000e+00  

## 5      5.000000e-02    1.000e+00  

## 6      5.000000e-02    1.000e+00  

## 7      5.000000e-02    1.000e+00  

## 8      5.000000e-02    1.000e+00  

## 9      5.000000e-02    1.000e+00  

## 10     5.000000e-02    1.000e+00  

## 11     5.000000e-02    1.000e+00  

## 12     5.000000e-02    1.000e+00  

## 13     5.000000e-02    1.000e+00  

## 14     5.000000e-02    1.000e+00  

## 15     5.000000e-02    1.000e+00  

## 16     5.000000e-02    1.000e+00  

## 17     5.000000e-02    1.000e+00  

## 18     5.000000e-02    1.000e+00  

## 19     5.000000e-02    1.000e+00  

##  

## IT      NF      F        RELDF      PRELDF      RELDX      STPPAR      D*STEP      NPRELDF
## 0       1      9.505e+02

```

```

##      1   3  9.292e+02  2.25e-02  5.83e-02  7.8e-02  5.5e+03  1.0e-01  1.62e+02
##      2   5  9.271e+02  2.28e-03  2.26e-03  9.3e-03  2.0e+01  1.0e-02  2.35e+01
##      3   7  9.231e+02  4.33e-03  4.36e-03  1.9e-02  5.2e+00  2.0e-02  2.05e+01
##      4   9  9.160e+02  7.64e-03  8.04e-03  3.7e-02  3.9e+00  4.0e-02  1.58e+01
##      5  12  9.159e+02  1.37e-04  1.38e-04  6.3e-04  7.7e+02  8.0e-04  7.44e+00
##      6  14  9.156e+02  2.72e-04  2.72e-04  1.3e-03  2.0e+01  1.6e-03  1.76e+01
##      7  16  9.151e+02  5.32e-04  5.33e-04  2.6e-03  1.0e+01  3.2e-03  1.82e+01
##      8  18  9.150e+02  1.05e-04  1.05e-04  5.3e-04  1.8e+02  6.4e-04  1.92e+01
##      9  20  9.150e+02  2.08e-05  2.08e-05  1.1e-04  8.5e+02  1.3e-04  1.94e+01
##     10  22  9.150e+02  4.16e-05  4.16e-05  2.1e-04  1.1e+02  2.6e-04  1.93e+01
##     11  24  9.150e+02  8.31e-06  8.31e-06  4.2e-05  2.1e+03  5.1e-05  1.94e+01
##     12  27  9.149e+02  6.63e-05  6.63e-05  3.4e-04  6.6e+01  4.1e-04  1.94e+01
##     13  30  9.149e+02  1.32e-06  1.32e-06  6.8e-06  1.3e+04  8.2e-06  1.95e+01
##     14  32  9.149e+02  2.65e-06  2.65e-06  1.4e-05  1.6e+03  1.6e-05  1.94e+01
##     15  34  9.149e+02  5.30e-07  5.30e-07  2.7e-06  3.3e+04  3.3e-06  1.94e+01
##     16  36  9.149e+02  1.06e-06  1.06e-06  5.5e-06  4.1e+03  6.6e-06  1.94e+01
##     17  38  9.149e+02  2.12e-07  2.12e-07  1.1e-06  8.1e+04  1.3e-06  1.94e+01
##     18  40  9.149e+02  4.24e-08  4.24e-08  2.2e-07  4.1e+05  2.6e-07  1.94e+01
##     19  43  9.149e+02  3.39e-07  3.39e-07  1.7e-06  1.3e+04  2.1e-06  1.94e+01
##     20  46  9.149e+02  6.78e-09  6.78e-09  3.5e-08  2.5e+06  4.2e-08  1.94e+01
##     21  48  9.149e+02  1.36e-09  1.36e-09  7.0e-09  1.3e+07  8.4e-09  1.94e+01
##     22  50  9.149e+02  2.71e-09  2.71e-09  1.4e-08  1.6e+06  1.7e-08  1.94e+01
##     23  52  9.149e+02  5.42e-09  5.42e-09  2.8e-08  7.9e+05  3.4e-08  1.94e+01
##     24  55  9.149e+02  1.08e-10  1.08e-10  5.6e-10  1.6e+08  6.7e-10  1.94e+01
##     25  57  9.149e+02  2.17e-10  2.17e-10  1.1e-09  2.0e+07  1.3e-09  1.94e+01
##     26  59  9.149e+02  4.34e-11  4.34e-11  2.2e-10  2.0e+00  2.7e-10  -1.31e-01
##     27  62  9.149e+02  3.47e-10  3.47e-10  1.8e-09  1.2e+07  2.1e-09  1.94e+01
##     28  65  9.149e+02  6.94e-12  6.94e-12  3.6e-11  2.0e+00  4.3e-11  -1.31e-01
##     29  67  9.149e+02  1.39e-12  1.39e-12  7.1e-12  2.0e+00  8.6e-12  -1.31e-01
##     30  69  9.149e+02  2.78e-12  2.78e-12  1.4e-11  2.0e+00  1.7e-11  -1.31e-01
##     31  71  9.149e+02  5.55e-13  5.55e-13  2.9e-12  2.0e+00  3.4e-12  -1.31e-01
##     32  73  9.149e+02  1.11e-12  1.11e-12  5.7e-12  2.0e+00  6.9e-12  -1.31e-01
##     33  75  9.149e+02  2.24e-13  2.22e-13  1.1e-12  2.0e+00  1.4e-12  -1.31e-01
##     34  77  9.149e+02  4.42e-13  4.44e-13  2.3e-12  2.0e+00  2.7e-12  -1.31e-01
##     35  79  9.149e+02  8.91e-13  8.88e-13  4.6e-12  2.0e+00  5.5e-12  -1.31e-01
##     36  81  9.149e+02  1.78e-13  1.78e-13  9.1e-13  2.0e+00  1.1e-12  -1.31e-01
##     37  83  9.149e+02  3.42e-14  3.55e-14  1.8e-13  2.0e+00  2.2e-13  -1.31e-01
##     38  85  9.149e+02  7.13e-14  7.11e-14  3.7e-13  2.0e+00  4.4e-13  -1.31e-01
##     39  87  9.149e+02  1.35e-14  1.42e-14  7.3e-14  2.0e+00  8.8e-14  -1.31e-01
##     40  89  9.149e+02  2.92e-14  2.84e-14  1.5e-13  2.0e+00  1.8e-13  -1.31e-01
##     41  92  9.149e+02  -1.09e+07  5.68e-16  3.0e-15  2.0e+00  3.5e-15  -1.31e-01
##
## ***** FALSE CONVERGENCE *****
##
##      FUNCTION    9.149129e+02    RELDX      2.955e-15
##      FUNC. EVALS    92        GRAD. EVALS    41
##      PRELDF      5.685e-16    NPRELDF   -1.313e-01
##
##      I      FINAL X(I)          D(I)          G(I)
##
##      1      3.099387e-01      1.000e+00     -7.625e+01
##      2      8.925303e-02      1.000e+00     -1.719e+01
##      3      6.212353e-02      1.000e+00     3.711e-01
##      4      7.397905e-02      1.000e+00     -2.512e+01
##      5      7.773623e-02      1.000e+00     -2.056e+01
##      6      1.061840e-01      1.000e+00     -4.482e+01

```

```
##    7  6.662175e-02  1.000e+00  -1.875e+00
##    8  6.902189e-02  1.000e+00  -1.464e+01
##    9  7.647960e-02  1.000e+00  -1.581e+01
##   10  8.212329e-02  1.000e+00  -1.871e+01
##   11  3.732246e-02  1.000e+00  3.604e+01
##   12  3.756050e-02  1.000e+00  3.318e+01
##   13  2.523498e-02  1.000e+00  4.706e+01
##   14  2.176802e-02  1.000e+00  4.485e+01
##   15  3.742583e-02  1.000e+00  3.536e+01
##   16  9.123184e-16  1.000e+00  5.098e+01
##   17  4.618901e-02  1.000e+00  3.528e+01
##   18  7.590265e-02  1.000e+00  -1.127e+01
##   19  5.929747e-02  1.000e+00  3.953e+00
```

```
## Warning in sqrt(pred$e): NaNs produced
```

```
akaike.choice <- rownames(arch.info)[which.min(arch.info[,1])]
bayes.choice <- rownames(arch.info)[which.min(arch.info[,2])]
```

```
akaike.choice
```

```
## [1] "arch12"
```

```
bayes.choice
```

```
## [1] "arch8"
```

```
##### REXAMINING RESIDUALS #####
```

```
archmod <- garch(mean_eqn_res, order = c(0,8))
```

```

##  

## ***** ESTIMATION WITH ANALYTICAL GRADIENT *****  

##  

##  

##      I      INITIAL X(I)          D(I)  

##  

##      1      1.499242e+00      1.000e+00  

##      2      5.000000e-02      1.000e+00  

##      3      5.000000e-02      1.000e+00  

##      4      5.000000e-02      1.000e+00  

##      5      5.000000e-02      1.000e+00  

##      6      5.000000e-02      1.000e+00  

##      7      5.000000e-02      1.000e+00  

##      8      5.000000e-02      1.000e+00  

##      9      5.000000e-02      1.000e+00  

##  

##      IT      NF      F          RELDF      PRELDF      RELDX      STPPAR      D*STEP      NPRELDF  

##      0      1      9.010e+02  

##      1      3      8.994e+02      1.75e-03      1.18e-02      2.2e-02      1.1e+03      1.0e-01      6.24e+00  

##      2      5      8.983e+02      1.24e-03      1.25e-03      1.5e-03      8.1e+00      1.0e-02      3.46e+00  

##      3      7      8.963e+02      2.20e-03      2.89e-03      6.7e-03      2.2e+00      4.5e-02      4.74e-01  

##      4      8      8.954e+02      1.05e-03      1.38e-03      1.1e-02      2.0e+00      4.5e-02      4.37e-01  

##      5      9      8.939e+02      1.70e-03      2.30e-03      3.1e-02      2.0e+00      9.0e-02      2.72e-01  

##      6      10     8.927e+02      1.29e-03      2.11e-03      2.9e-02      2.0e+00      9.0e-02      3.34e-01  

##      7      12     8.923e+02      4.52e-04      7.48e-04      1.2e-02      2.0e+00      4.4e-02      1.04e-01  

##      8      14     8.921e+02      2.15e-04      5.80e-04      4.3e-03      2.2e+00      2.2e-02      6.76e-02  

##      9      15     8.919e+02      2.97e-04      4.19e-04      5.1e-03      2.0e+00      2.2e-02      5.91e-02  

##      10     16     8.917e+02      2.03e-04      2.48e-04      6.4e-03      2.0e+00      2.2e-02      3.35e-02  

##      11     17     8.915e+02      1.61e-04      4.80e-04      1.5e-02      2.0e+00      4.3e-02      2.33e-02  

##      12     18     8.912e+02      3.83e-04      6.67e-04      1.3e-02      2.0e+00      4.3e-02      3.40e-02  

##      13     20     8.911e+02      8.27e-05      1.79e-04      5.1e-03      2.0e+00      1.7e-02      4.13e-02  

##      14     21     8.910e+02      1.15e-04      2.49e-04      4.9e-03      2.0e+00      1.7e-02      2.12e-02  

##      15     22     8.909e+02      7.74e-05      1.32e-04      4.4e-03      2.0e+00      1.7e-02      4.66e-03  

##      16     23     8.909e+02      7.42e-05      1.06e-04      5.0e-03      2.0e+00      1.7e-02      9.17e-03  

##      17     24     8.909e+02      5.03e-06      1.00e-04      5.0e-03      2.0e+00      1.7e-02      8.46e-03  

##      18     25     8.908e+02      5.62e-05      9.17e-05      2.2e-03      2.0e+00      8.6e-03      5.22e-03  

##      19     26     8.908e+02      2.60e-05      4.90e-05      2.3e-03      2.0e+00      8.6e-03      5.03e-03  

##      20     27     8.908e+02      2.75e-05      3.57e-05      2.5e-03      2.0e+00      8.6e-03      1.58e-03  

##      21     28     8.908e+02      5.26e-06      4.82e-05      6.1e-03      2.0e+00      1.7e-02      1.50e-03  

##      22     30     8.908e+02      1.96e-05      4.99e-05      2.7e-03      2.0e+00      8.5e-03      1.50e-03  

##      23     31     8.907e+02      2.18e-05      5.69e-05      2.6e-03      2.0e+00      8.5e-03      3.30e-03  

##      24     32     8.907e+02      6.09e-06      2.84e-05      2.1e-03      2.0e+00      8.5e-03      1.17e-03  

##      25     33     8.907e+02      1.67e-05      2.55e-05      2.0e-03      2.0e+00      8.5e-03      1.17e-03  

##      26     34     8.907e+02      9.82e-07      1.55e-05      2.4e-03      2.0e+00      8.5e-03      6.77e-04  

##      27     35     8.907e+02      4.87e-06      1.75e-05      8.8e-04      2.1e+00      4.3e-03      4.54e-04  

##      28     36     8.907e+02      8.18e-06      1.37e-05      1.2e-03      2.0e+00      4.3e-03      4.49e-04  

##      29     37     8.907e+02      3.22e-06      5.35e-06      1.4e-03      1.9e+00      4.3e-03      1.18e-04  

##      30     38     8.907e+02      2.60e-06      5.01e-06      1.0e-03      2.2e+00      4.3e-03      1.03e-04  

##      31     40     8.907e+02      1.29e-06      3.32e-06      4.3e-04      2.8e+00      1.6e-03      1.47e-04  

##      32     41     8.907e+02      1.55e-06      2.04e-06      4.4e-04      2.0e+00      1.6e-03      8.17e-05  

##      33     42     8.907e+02      8.05e-07      1.16e-06      7.0e-04      1.9e+00      1.6e-03      2.66e-05  

##      34     43     8.907e+02      4.34e-07      2.23e-06      9.7e-04      1.6e+00      3.2e-03      9.66e-06  

##      35     44     8.907e+02      1.31e-06      2.47e-06      7.7e-04      2.0e+00      3.2e-03      3.60e-05  

##      36     46     8.907e+02      1.02e-07      1.30e-06      3.3e-04      2.3e+00      1.3e-03      3.75e-05  

##      37     47     8.907e+02      7.78e-07      8.30e-07      1.2e-04      2.2e+00      6.4e-04      1.47e-05

```

```

##   38   49  8.907e+02  4.29e-07  5.33e-07  7.6e-04  2.3e+00  1.8e-03  1.10e-05
##   39   51  8.907e+02  3.53e-07  6.26e-07  6.2e-04  1.9e+00  1.6e-03  8.89e-06
##   40   52  8.907e+02  1.56e-07  5.41e-07  4.3e-04  1.9e+00  1.6e-03  5.57e-06
##   41   55  8.907e+02  8.64e-08  1.38e-07  9.5e-05  3.4e+00  2.8e-04  1.83e-06
##   42   56  8.907e+02  4.29e-08  7.18e-08  6.0e-05  2.4e+00  2.8e-04  3.37e-06
##   43   57  8.907e+02  4.04e-08  6.64e-08  7.5e-05  2.0e+00  2.8e-04  1.45e-06
##   44   58  8.907e+02  3.03e-08  4.82e-08  8.1e-05  1.9e+00  2.8e-04  4.05e-07
##   45   59  8.907e+02  2.54e-08  3.30e-08  9.9e-05  1.8e+00  2.8e-04  3.14e-07
##   46   60  8.907e+02  1.59e-08  2.19e-08  1.2e-04  1.7e+00  2.8e-04  1.00e-07
##   47   61  8.907e+02  1.17e-08  2.25e-08  9.2e-05  1.9e+00  2.8e-04  2.76e-07
##   48   63  8.907e+02  3.14e-09  7.79e-09  4.2e-05  1.9e+00  1.2e-04  5.30e-08
##   49   64  8.907e+02  5.75e-09  1.04e-08  2.8e-05  1.9e+00  1.2e-04  6.59e-08
##   50   65  8.907e+02  2.84e-09  5.44e-09  3.5e-05  1.7e+00  1.2e-04  2.76e-08
##   51   66  8.907e+02  2.82e-09  3.52e-09  3.6e-05  2.0e+00  1.2e-04  6.93e-08
##   52   69  8.907e+02  1.13e-09  2.31e-09  9.4e-06  2.9e+00  3.9e-05  3.49e-08
##   53   70  8.907e+02  1.08e-09  1.22e-09  1.2e-05  1.9e+00  3.9e-05  1.49e-08
##   54   71  8.907e+02  1.52e-09  1.86e-09  3.4e-05  1.7e+00  7.7e-05  8.65e-09
##   55   72  8.907e+02  1.19e-09  1.73e-09  2.8e-05  1.8e+00  7.7e-05  1.23e-08
##   56   73  8.907e+02  6.93e-10  1.53e-09  2.1e-05  2.5e+00  7.7e-05  8.15e-09
##   57   74  8.907e+02  6.60e-10  1.30e-09  2.6e-05  1.5e+00  7.7e-05  2.48e-09
##   58   75  8.907e+02  4.62e-10  1.24e-09  2.2e-05  1.8e+00  7.7e-05  7.44e-09
##   59   77  8.907e+02  1.44e-10  4.62e-10  6.3e-06  2.2e+00  3.0e-05  1.31e-09
##   60   79  8.907e+02  2.08e-10  3.33e-10  2.9e-06  2.3e+00  1.4e-05  1.64e-09
##   61   80  8.907e+02  7.24e-11  8.25e-11  6.3e-06  1.5e+00  1.4e-05  2.68e-10
##   62   81  8.907e+02  3.99e-11  7.24e-11  5.1e-06  5.6e+00  1.4e-05  2.84e-10
##   63   83  8.907e+02  6.05e-11  9.16e-11  1.4e-05  8.1e-01  3.0e-05  1.38e-10
##   64   84  8.907e+02  2.26e-11  8.54e-11  1.3e-05  7.9e-01  3.0e-05  1.20e-10
##   65   85  8.907e+02  2.09e-12  1.54e-11  4.5e-06  0.0e+00  1.3e-05  1.54e-11
##
## ***** RELATIVE FUNCTION CONVERGENCE *****
##
##   FUNCTION      8.906906e+02    RELDX      4.453e-06
##   FUNC. EVALS      85          GRAD. EVALS      66
##   PRELDF      1.538e-11    NPRELDF      1.538e-11
##
##   I      FINAL X(I)          D(I)          G(I)
##
##   1      1.040612e+00      1.000e+00      2.249e-04
##   2      1.112691e-01      1.000e+00      9.552e-04
##   3      2.275198e-02      1.000e+00      2.865e-04
##   4      1.397528e-01      1.000e+00      2.828e-04
##   5      4.969623e-02      1.000e+00     -2.172e-04
##   6      1.128825e-01      1.000e+00      1.896e-03
##   7      4.307090e-02      1.000e+00      4.642e-04
##   8      4.750205e-02      1.000e+00      4.004e-04
##   9      9.495435e-02      1.000e+00     -2.883e-03

```

```

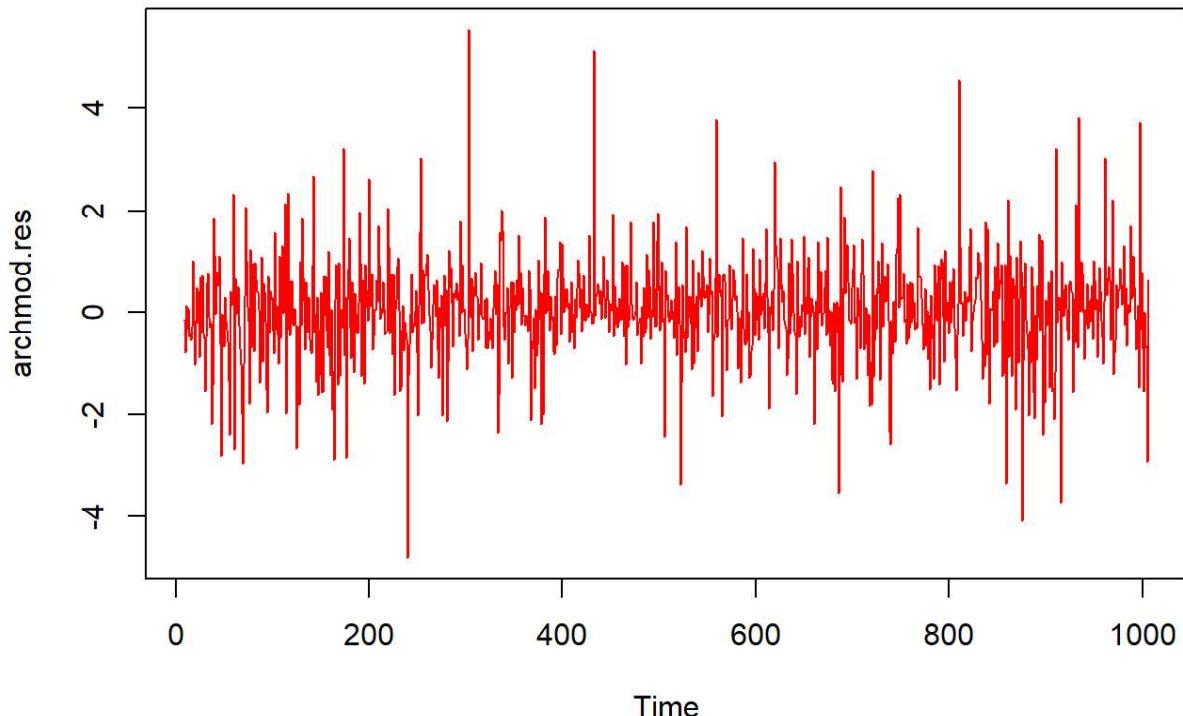
archmod.res <- archmod$residuals

archmod.res <- na.omit(archmod.res)

plot(archmod.res, col = "red", main = "Chart 6: Standardised Residuals from ARCH(8) Equation", type = "l")

```

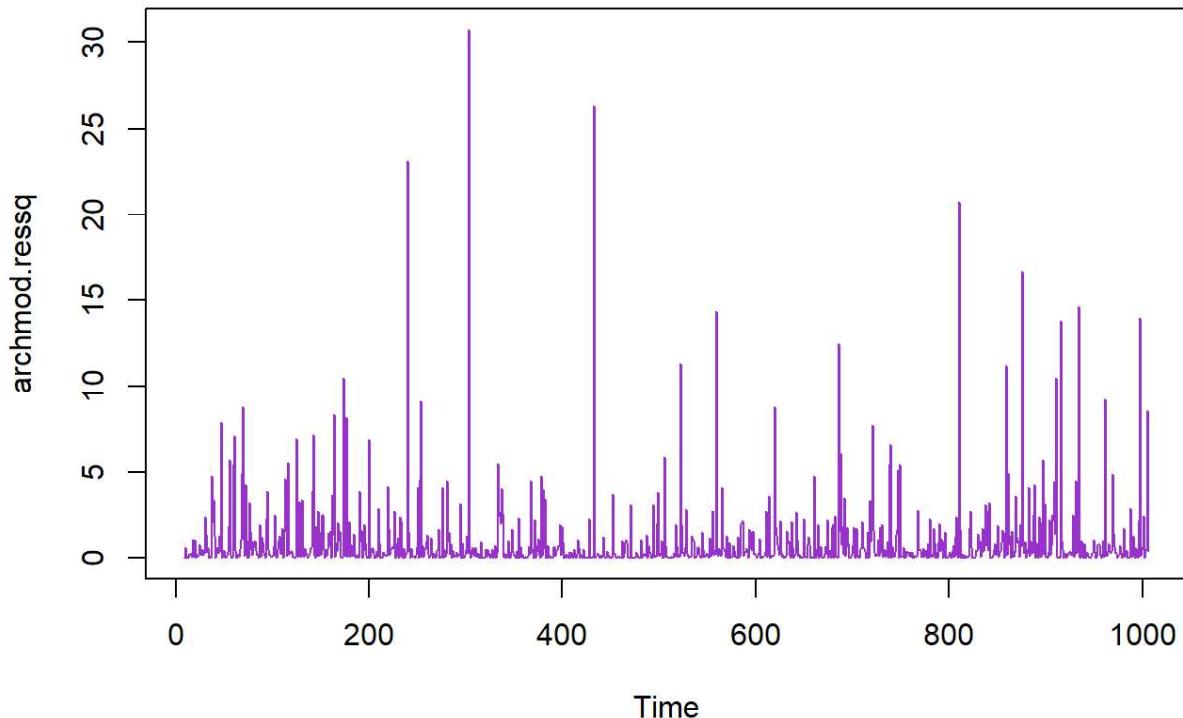
Chart 6: Standardised Residuals from ARCH(8) Equation



```
archmod.ressq <- archmod.res^2

plot(archmod.ressq, col = "darkorchid", main = "Chart 7: Squared Standardised Residuals from
ARCH(8) Equation", type = "l")
```

Chart 7: Squared Standardised Residuals from ARCH(8) Equation



```
##### VERIFYING IF ARCH EFFECTS ARE GONE #####

```

```
Box.test(archmod.ressq, lag = m, type = "Box-Pierce")
```

```
##
## Box-Pierce test
##
## data: archmod.ressq
## X-squared = 11.612, df = 32, p-value = 0.9996
```

```
Box.test(archmod.ressq, lag = m, type = "Ljung-Box")
```

```
##
## Box-Ljung test
##
## data: archmod.ressq
## X-squared = 11.879, df = 32, p-value = 0.9995
```

```
##### TESTING IF A GARCH(1,1) MODEL GIVES SIMILAR RESULTS #####

```

```
garchmod <- garch(mean_eqn_res, order = c(1,1))
```

```

##  

## ***** ESTIMATION WITH ANALYTICAL GRADIENT *****  

##  

##  

##      I      INITIAL X(I)          D(I)  

##  

##      1      2.248863e+00      1.000e+00  

##      2      5.000000e-02      1.000e+00  

##      3      5.000000e-02      1.000e+00  

##  

##      IT      NF      F      RELDF      PRELDF      RELDX      STPPAR      D*STEP      NPRELDF  

##      0      1  9.545e+02  

##      1      4  9.527e+02  1.92e-03  1.12e-02  2.8e-02  6.9e+02  1.3e-01  3.85e+00  

##      2      6  9.522e+02  5.52e-04  5.67e-04  2.0e-03  7.1e+00  1.3e-02  5.03e-01  

##      3      7  9.515e+02  7.18e-04  7.22e-04  3.7e-03  2.9e+00  2.5e-02  4.33e-02  

##      4      8  9.510e+02  4.56e-04  5.15e-04  1.0e-02  2.0e+00  5.0e-02  3.20e-02  

##      5      9  9.507e+02  3.45e-04  3.54e-04  9.5e-03  2.0e+00  5.0e-02  3.58e-02  

##      6     12  9.473e+02  3.61e-03  3.13e-03  9.3e-02  1.2e+00  4.0e-01  3.98e-02  

##      7     14  9.333e+02  1.48e-02  8.68e-03  2.7e-01  2.0e+00  8.0e-01  1.90e+02  

##      8     16  9.281e+02  5.57e-03  4.77e-03  8.1e-02  2.0e+00  1.6e-01  9.13e+04  

##      9     18  9.165e+02  1.26e-02  1.18e-02  2.1e-01  2.0e+00  3.2e-01  6.01e+06  

##     10     20  9.142e+02  2.44e-03  2.51e-03  4.9e-02  2.0e+00  6.4e-02  7.04e+08  

##     11     21  9.120e+02  2.45e-03  6.18e-03  8.1e-02  2.0e+00  1.3e-01  1.40e+07  

##     12     22  9.089e+02  3.36e-03  7.70e-03  8.9e-02  2.0e+00  1.3e-01  4.04e+06  

##     13     24  9.079e+02  1.13e-03  2.63e-03  2.2e-02  2.0e+00  4.0e-02  7.63e+04  

##     14     25  9.071e+02  8.99e-04  1.17e-03  2.3e-02  2.0e+00  4.0e-02  6.74e+04  

##     15     26  9.062e+02  9.05e-04  1.19e-03  4.6e-02  2.0e+00  8.0e-02  3.03e+04  

##     16     29  9.059e+02  4.02e-04  6.95e-04  4.5e-03  3.6e+00  8.1e-03  2.82e+05  

##     17     30  9.056e+02  2.95e-04  3.17e-04  5.1e-03  2.0e+00  8.1e-03  1.10e+05  

##     18     31  9.051e+02  5.35e-04  5.98e-04  8.7e-03  2.0e+00  1.6e-02  9.36e+04  

##     19     32  9.047e+02  4.32e-04  8.50e-04  1.3e-02  2.0e+00  3.2e-02  5.21e+04  

##     20     33  9.047e+02  8.62e-05  7.78e-04  1.6e-02  2.0e+00  3.2e-02  6.14e+03  

##     21     35  9.045e+02  1.53e-04  3.80e-04  5.6e-03  2.0e+00  1.3e-02  8.73e+01  

##     22     36  9.044e+02  1.35e-04  1.61e-04  5.9e-03  2.0e+00  1.3e-02  4.51e+02  

##     23     38  9.044e+02  1.95e-05  5.49e-05  3.5e-03  2.0e+00  7.0e-03  3.20e+01  

##     24     40  9.044e+02  3.41e-05  5.62e-05  3.4e-03  2.0e+00  6.6e-03  3.12e+00  

##     25     41  9.043e+02  1.35e-05  1.92e-05  2.6e-03  2.0e+00  6.6e-03  2.31e+01  

##     26     43  9.043e+02  7.24e-07  5.20e-06  3.4e-04  2.7e+01  6.6e-04  5.07e-01  

##     27     44  9.043e+02  3.68e-06  3.61e-06  3.8e-04  2.0e+00  6.6e-04  1.09e-01  

##     28     45  9.043e+02  2.77e-06  3.01e-06  6.8e-04  2.0e+00  1.3e-03  6.45e-02  

##     29     46  9.043e+02  2.46e-06  4.75e-06  1.4e-03  2.0e+00  2.6e-03  1.66e-02  

##     30     47  9.043e+02  2.99e-07  2.92e-06  1.2e-03  2.0e+00  2.6e-03  6.40e-04  

##     31     48  9.043e+02  7.29e-07  2.24e-06  1.2e-03  2.0e+00  2.6e-03  9.76e-05  

##     32     51  9.043e+02  1.37e-07  2.20e-07  4.9e-05  4.1e+00  9.9e-05  3.22e-05  

##     33     54  9.043e+02  4.94e-07  1.29e-07  5.5e-04  0.0e+00  1.0e-03  1.29e-07  

##     34     76  9.043e+02  2.51e-16  5.23e-18  7.8e-15  3.0e+10  1.6e-14  3.51e-06  

##     35     77  9.043e+02 -8.80e-16  1.05e-17  1.6e-14  1.5e+10  3.3e-14  2.49e-06  

##  

##      ***** FALSE CONVERGENCE *****  

##  

##      FUNCTION      9.043302e+02      RELDX      1.566e-14  

##      FUNC. EVALS      77      GRAD. EVALS      35  

##      PRELDF      1.045e-17      NPRELDF      2.490e-06  

##  

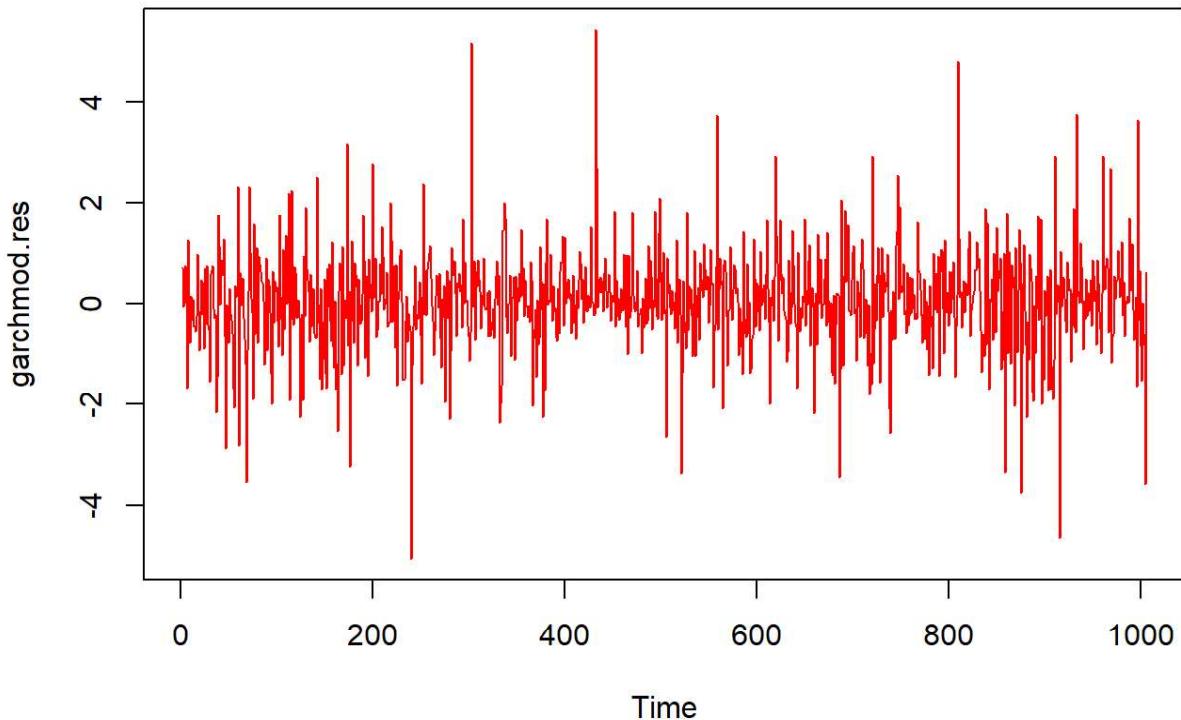
##      I      FINAL X(I)          D(I)          G(I)

```

```
##  
##      1  1.857971e-01  1.000e+00  2.270e-01  
##      2  1.122845e-01  1.000e+00  4.714e-02  
##      3  8.181193e-01  1.000e+00 -1.733e-01
```

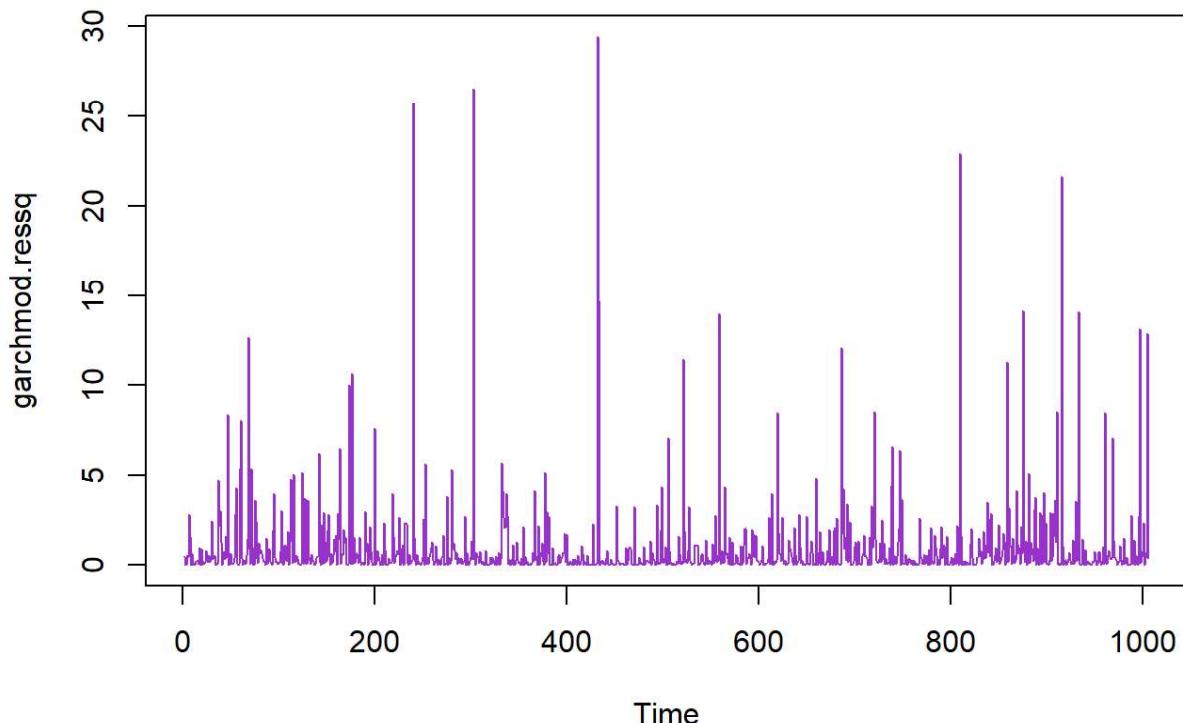
```
garchmod.res <- garchmod$residuals  
  
garchmod.res <- na.omit(garchmod.res)  
  
garchmod.ressq <- garchmod.res^2  
  
plot(garchmod.res, col = "red", main = "Chart 8: Standardised Residuals from GARCH(1,1) Equation", type = "l")
```

Chart 8: Standardised Residuals from GARCH(1,1) Equation



```
plot(garchmod.ressq, col = "darkorchid", main = "Chart 9: Squared Standardised Residuals from GARCH(1,1) Equation", type = "l")
```

Chart 9: Squared Standardised Residuals from GARCH(1,1) Equation



```
Box.test(garchmod.ressq, lag = m, type = "Box-Pierce")
```

```
##  
## Box-Pierce test  
##  
## data: garchmod.ressq  
## X-squared = 11.666, df = 32, p-value = 0.9996
```

```
Box.test(garchmod.ressq, lag = m, type = "Ljung-Box")
```

```
##  
## Box-Ljung test  
##  
## data: garchmod.ressq  
## X-squared = 11.896, df = 32, p-value = 0.9995
```

```
# The GARCH(1,1) model fails to remove ARCH effects from the squared residuals!
```

```
##### CREATING FINAL MODEL #####
```

```
final_model <- ugarchspec(  
  variance.model = list(model = "sGARCH", garchOrder = c(1,1)),  
  mean.model = list(armaOrder = c(0,0), include.mean = TRUE),  
  distribution.model = "norm"  
)
```

f) Using conditional varince forecasts from your ARCH/GARCH model, compute and plot h-step ahead point and 95% interval forecasts for the daily

returns for h = 1, 2, ..., 10.

```
#####
##### FORECASTING VALUES #####
#####

data_fc <- data_fc %>%
  select(date)

final_model_forecast_fit <- ugarchfit(spec = final_model, data = data$returns)

final_model_forecast <- ugarchforecast(final_model_forecast_fit, n.ahead = 10)

#####

#####
##### EXTRACTING MODEL OUTPUTS #####
#####

cond_mean_forecast <- data.frame(final_model_forecast@forecast$seriesFor) %>% clean_names() %>%
  select(cond_mean = x1972_10_03)

cond_var_forecast <- data.frame(final_model_forecast@forecast$sigmaFor) %>% clean_names() %>%
  select(cond_var = x1972_10_03) %>%
  mutate(upper_interval = cond_var * 1.96) %>%
  mutate(lower_interval = cond_var * -1.96) %>%
  select(upper_interval, lower_interval)

garch_forecast <- cond_mean_forecast %>%
  cbind(cond_var_forecast)

#####

#####
##### COMBINING INTO FORECAST FOR PLOT #####
#####

data_fc <- data_fc %>%
  cbind(garch_forecast)

data_final_garch <- data %>%
  full_join(data_fc, by = "date") %>%
  filter(date >= "2019-01-01")

#####

#####
##### CHARTING #####
#####

garch_fc <- ggplot(data_final_garch, aes(x = date)) +
  geom_line(aes(y = returns)) +
  geom_line(aes(y = cond_mean, colour = "Point estimate")) +
  geom_line(aes(y = upper_interval, colour = "Prediction interval")) +
  geom_line(aes(y = lower_interval, colour = "Prediction interval")) +
  scale_colour_manual(name = "Forecast", values = c("Point estimate" = "red", "Prediction interval" = "blue")) +
```

```

ggttitle("Chart 9: Daily Apple log returns from 2019-01-01 with GARCH forecast")+
  ylab("Returns (%)") +
  xlab("Date") +
  theme(plot.title = element_text(size = 10),
        plot.caption = element_text(hjust = 0),
        legend.position = c(0.1, 0.4),
        legend.justification = c(0, 1))

garch_fc

```

```

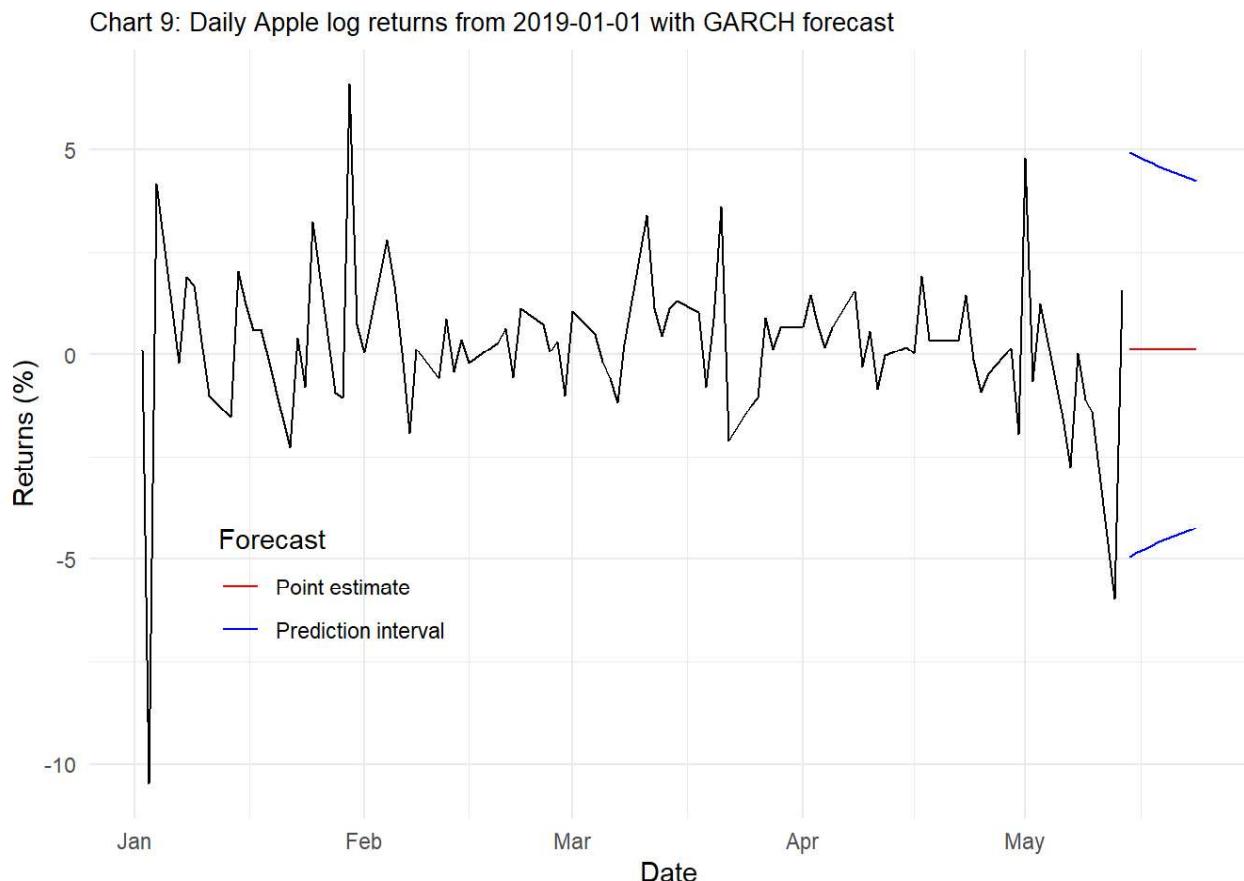
## Warning: Removed 10 rows containing missing values or values outside the scale range
## (`geom_line()`).

```

```

## Warning: Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).

```



g) Compare the forecast intervals from part (c) with those from part (f). Do you observe any difference?

- Chart 2 and Chart 9 have been replicated below for ease of reference.

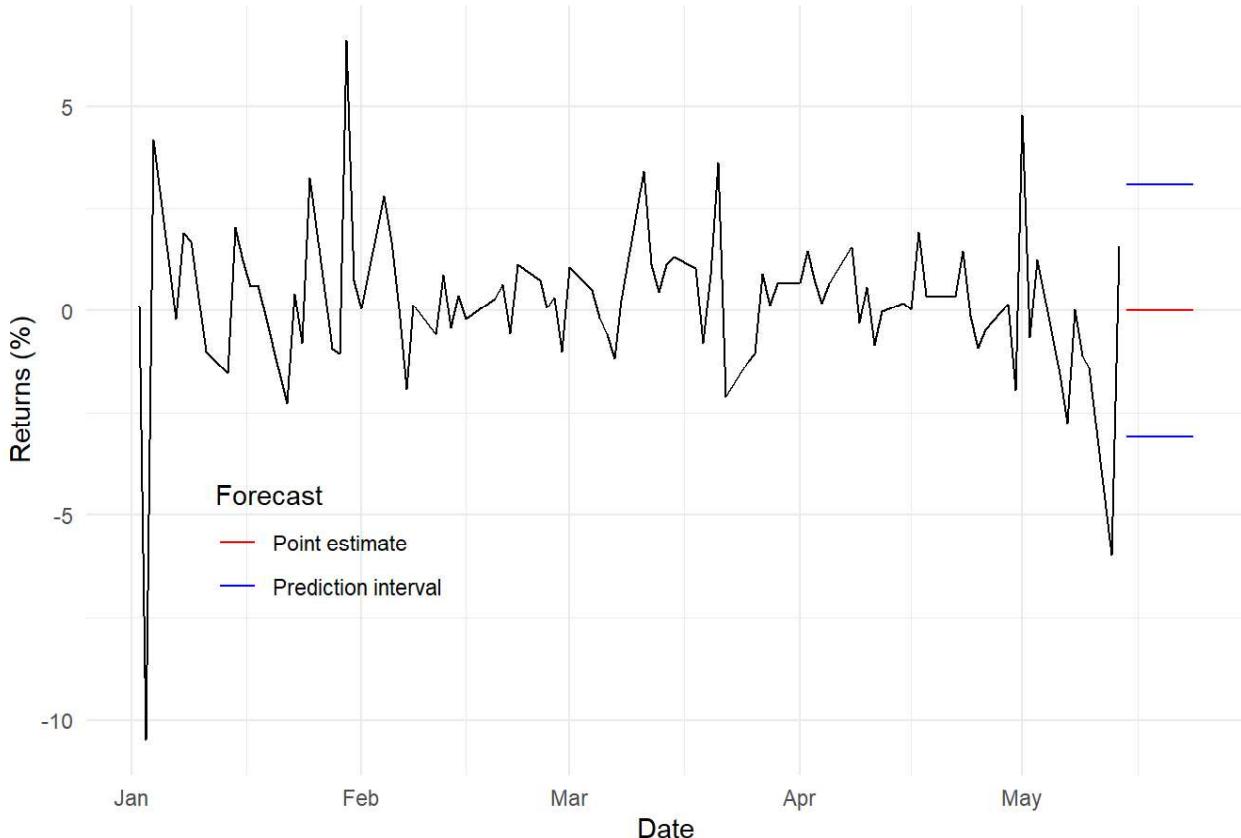
- The key difference between the forecast intervals between them is that in the GARCH model forecast, the intervals are sloping downwards whereas they are constant for the mean equation forecast.
- The reason mean equation forecast has constant variance is because that is a requirement for effectively modelling using an ARIMA model: variance must be constant and unconditional. Therefore, by definition, these models are unable to show any time variance.
- In comparison, the reason the variance is not constant is because the objective of GARCH models is to model conditional (i.e. time varying) variance. Specifically, it slopes downwards due to the nature of how volatility is modelled. That is, GARCH models incorporate mean reversion of volatility, which means that while volatility can spike in the short term due to shocks, it is always expected to revert to a long-run average over time. Furthermore, given our time series is stationary, therefore this conditional variance will always revert back to its unconditional variance following the unwinding of shocks.

```
mean_eqn_fc
```

```
## Warning: Removed 10 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

```
## Warning: Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

Chart 2: Daily Apple log returns from 2019-01-01 with forecast



garch_fc

```
## Warning: Removed 10 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 92 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

Chart 9: Daily Apple log returns from 2019-01-01 with GARCH forecast

