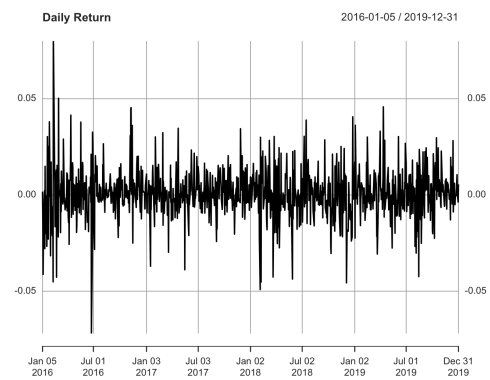
Problem Set 4: Computer Exercises

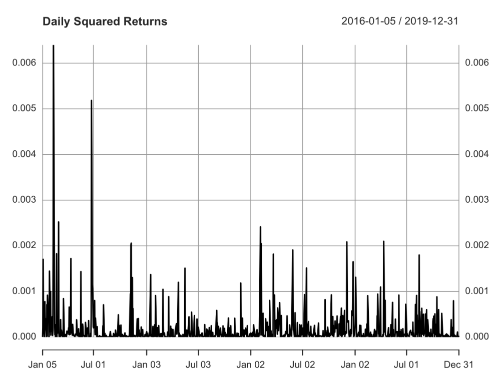
Jenny Petrova

March 27, 2025

5. The following exercises use the daily stock closing price data for JPMorgan Chase & Co. (JPM) from the period 2016-2019, obtained from Yahoo Finance. This data is accessed using the **quantmod** package in **R**. The outputs presented below are generated using the **R** programming language. The daily log returns from the closing prices are computed as differences of the log prices.

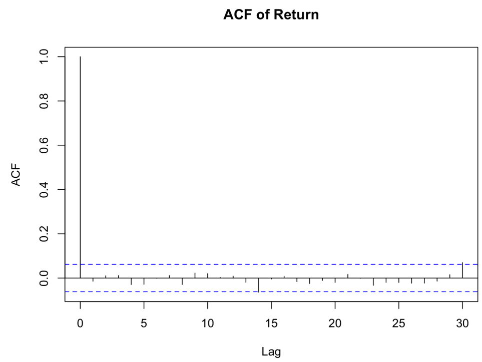
   (a) Examine the plots of the daily log returns and the squared daily stock returns, where the daily log returns from closing prices are computed as differences of the log prices:

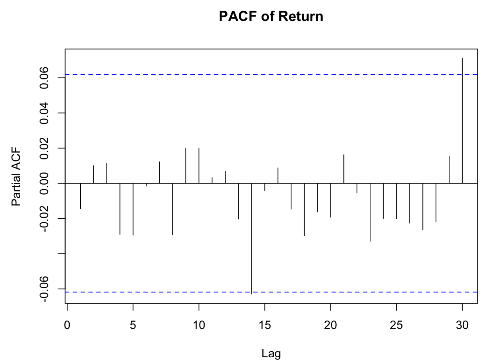




   In the plots above, there appears to be some volatility clustering in the return series. In the 'Daily Return' plot (top), we observe large positive and negative swings in the volatility in the first 6 months (from January up to July), followed by a period of relatively low swings in the volatility, from July up to ≈ October. This corresponds to the observations, and is better observed, in the plot of the 'Daily Squared Returns' (bottom). We see periods of high spikes in the volatility followed by periods of much lower volatility.

(b) Examine the ACF and PACF plots of the returns:





In the ACF and PACF plots, we examine some potential serial correlation at lags 14 and 30. This suggests past returns may have some influence on future returns. To confirm these observations, we perform the following portmanteau test:

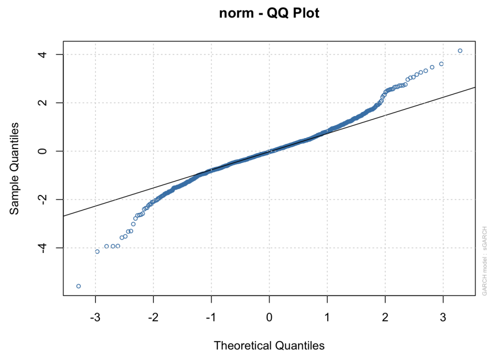
```
    Box-Ljung test
data:  r
X-squared = 4.0214, df = 10, p-value = 0.9464
```

Since our sample size is 1005 observations, we set the number of lags (df) to 10 (about 10% of the sample size). For the Box-Ljung test, the p-value is > 0.05, therefore we *fail* to reject the null hypothesis of the test and conclude that the residual returns are independently distributed (uncorrelated). To identify whether the variance of the returns is uncorrelated, we test whether the model has ARCH effects, by performing Engle's ARCH-LM test:

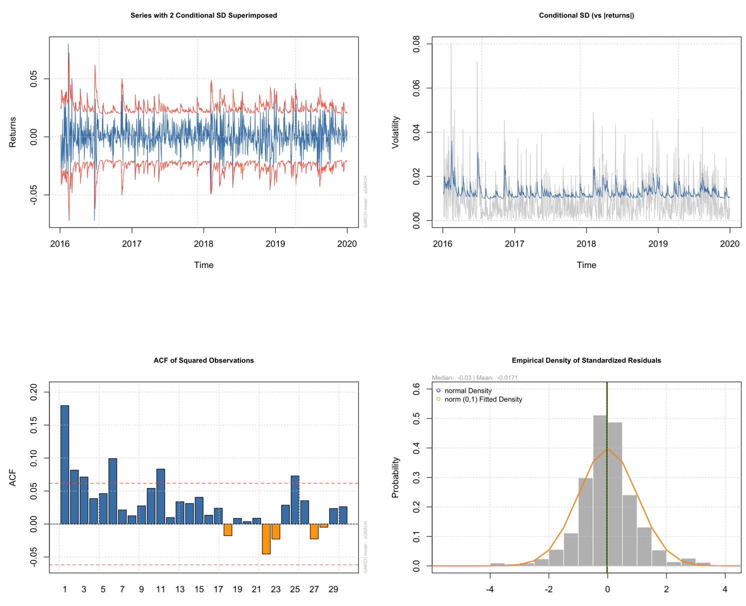
```
    ARCH LM-test; Null hypothesis: no ARCH effects
data:  r
Chi-squared = 47.967, df = 10, p-value = 6.295e-07
```

Since the p-value is < 0.05, we reject the null hypothesis of no ARCH effects. Therefore, there is evidence to suggest the variance of the returns is autocorrelated, which is an indication of volatility clustering. This supports our observations in part (a).

(c) We fit a Gaussian ARMA(0,0)-GARCH(1,1) model to the return series, and check the QQ-plot of the standardized residuals:

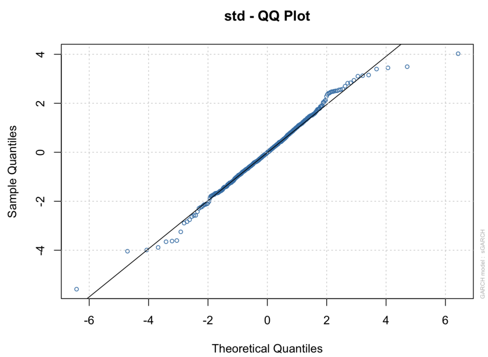


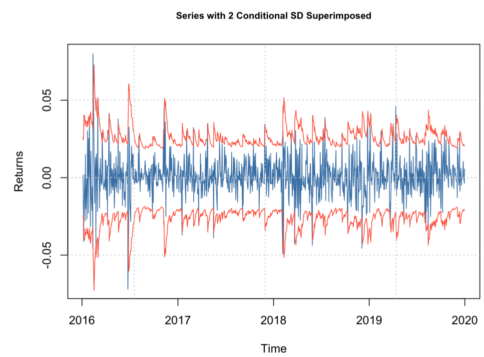
Observing the QQ-plot of the standardized residuals, we see that the residuals do not follow a straight line, but rather deviate at the beginning and end of the line. We also spot extreme outliers at both ends. This means the residuals have heavy tails in their distribution, and are therefore not normally distributed. Modeling the residuals with a Student-t distribution may be more useful for this data, since Student-t innovations are better at handling data that exhibits heavy tails or a non-Normal distribution. Our analysis is further supported by the following plots:



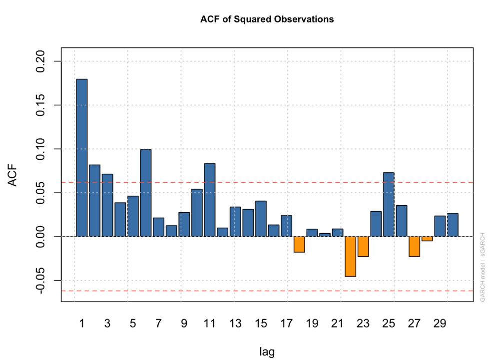
The plot of the series with 2 conditional SD superimposed (top left) shows that the pattern of the conditional standard deviation (volatility, depicted in red) follows the pattern of high and low periods of returns (depicted in blue). This suggests the model captures volatility clustering. We also see spikes in the returns, which exceed the volatility. These extreme events support our findings in the QQ-plot. Additionally, in the plot of the conditional SD vs absolute returns (top right) we see that the volatility (blue) tends to increase when the returns (grey) are higher, supporting the existence of volatility clustering and therefore time-varying volatility residuals (as seen in the non-constant residuals of the QQ-plot). In the ACF of squared observations (bottom left), there is significant serial correlation at lags 1, 2, 3, 6, 11, and 25, which is further evidence of volatility clustering. And in the plot of the empirical density of the standardized residuals (bottom right), the histogram shows both tails exhibit deviation from the Normal distribution curve (orange) - which supports our observations in the QQ-plot of the standardized residuals.

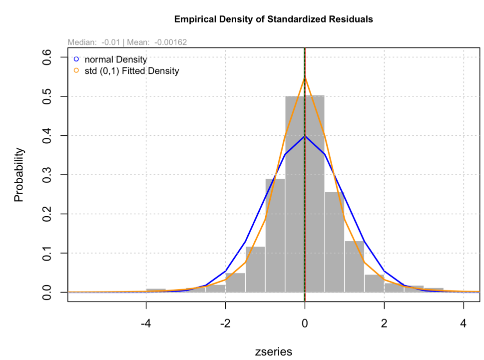
(d) Following from the observations in part (c), we fit a Gaussian ARMA(0,0)-GARCH(1,1) model with standardized Student-t innovations to the returns, and check the QQ-plot of the standardized residuals and the plot of the series with two conditional standard deviations superimposed:





The residuals in the QQ-plot (left) fit with Student-t innovations follow the line more closely, suggesting this distribution is better fit to model our data. There still exist some extreme outliers, on both ends of the line, but the overall fit has still improved. The plot of the series with 2 conditional SD superimposed (right) continues to show the pattern of the conditional standard deviation (volatility, red) follows the pattern of high and low periods of returns (blue). This model still captures the volatility clustering occurring in the data. We make further analyses from the following plots:





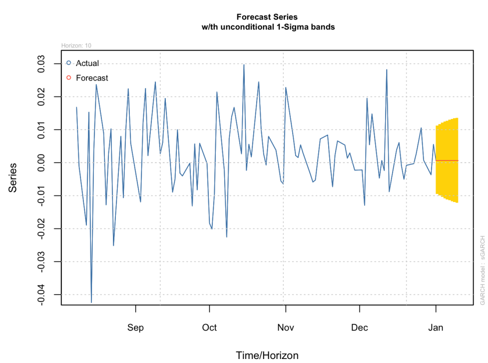
The ACF of squared observations (left) show that this model still captures the significant serial correlation at lags 1, 2, 3, 6, 11, and 25, supporting the evidence of volatility clustering. In the plot of the empirical density of the standardized residuals, we see that the Student-t distribution (orange) is a better fit for the data, as it is better aligned with the histogram, compared to the Normal distribution (blue).

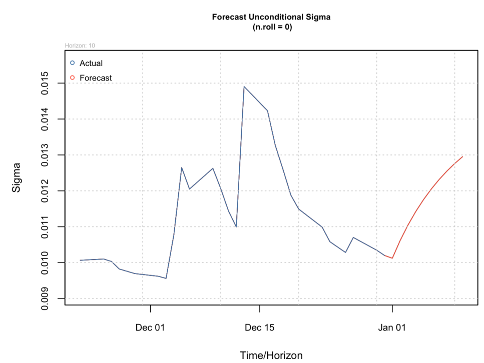
(e) Lastly, we compute the 1- to 10-step ahead forecasts of the returns and conditional standard deviation based on our model in part (d). The forecasts are

```
*------------------------------------*
*       GARCH Model Forecast         *
*------------------------------------*
Model: sGARCH
Horizon: 10
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=2019-12-31]:
        Series   Sigma
T+1  0.0006612 0.01012
T+2  0.0006612 0.01062
T+3  0.0006612 0.01105
T+4  0.0006612 0.01142
T+5  0.0006612 0.01176
T+6  0.0006612 0.01205
T+7  0.0006612 0.01232
T+8  0.0006612 0.01255
T+9  0.0006612 0.01276
T+10 0.0006612 0.01295
```

The output shows that, with each additional time-step ahead, the forecasted value of the returns remains the same, while the volatility (sigma) is expected to gradually increase over the next 1- to 10-step ahead forecasts. We examine the following plots for the forecasted volatility:





The plot of the forecasted series with the unconditional 1-sigma bands (left) shows the forecasted range of returns (highlighted in yellow). The range of returns gradually increases with each additional one-step-ahead forecast, corresponding to the forecasted, increasing volatility displayed in the Garch Model Forecast output above. The plot of the forecasted unconditional sigma (right) further visualizes this forecast, displaying the forecasted long-term volatility over the same time horizon. Therefore, over the next few time steps, we expect the volatility to sharply increase.

Corresponding Code

```
library(quantmod)
library(forecast)
library(rugarch)
library(FinTS)

jpm_full <- getSymbols("JPM", src="yahoo", auto.assign = F)
jpm <- jpm_full["2016/2019"]

# 5(a)
r <- diff(log(jpm$JPM.Close))[-1]
length(r)
plot(r, main = "Daily Return")
plot(r^2, main = "Daily Squared Return")

# 5(b)
(acf(r, main = "ACF of Return"))
(pacf(r, main = "PACF of Return"))
tsdisplay(r)
Box.test(r, lag=10, type = "Ljung-Box")
Box.test(r, lag=10, type = "Box-Pierce")
ArchTest(r, 10)

# 5(c)
# Fit a GARCH(1,1) model
spec_garch11 <- ugarchspec(variance.model = list(garchOrder=c(1,1)),
                           mean.model = list(armaOrder=c(0,0)))
fit_garch11 <- ugarchfit(spec=spec_garch11, data=r)
fit_garch11
plot(fit_garch11)

10# 5(d)
spec_garch11std <- ugarchspec(variance.model = list(garchOrder=c(1,1)),
                              mean.model = list(armaOrder=c(0,0)),
                              distribution.model = "std")
fit_garch11std <- ugarchfit(spec=spec_garch11std, data=r)
fit_garch11std
plot(fit_garch11std)

# 5(e)
(fcst_garch11std <- ugarchforecast(fit_garch11std, n.ahead=10))
plot(fcst_garch11std, which=1)
plot(fcst_garch11std, which=3)
```