NASDAQ Composite Index

Final Project - Volatility Models - Financial Statistics

Alexander J Ohrt

06 april, 2022

1 Abstract

The NASDAQ Composite Index (IXIC) is studied using data from 01.01.2012 to 01.03.2022. The mean and variance of the series are modeled. The work shows that the volatility of the returns of the index, which is a stationary process, can be modeled by a ARMA(2,2)-EGARCH model. Moreover, it shows that the GARCH-based models are more reliable for volatility modeling than the use of historical volatility or EWMA, without the need of setting any hyperparameters. Lastly, it shows that a multivariate DCC GARCH model can be used to model the correlation between two stationary series, in this case the correlation between the returns of IXIC and the returns of the Stratus Properties Inc. stock, which is one of the components of the index.

2 Introduction

Describe

- Scenario and objective of the work. What will be analyzed.
- Precise description of variable (NASDAQ Composite) used in the analysis and description of where the data is gathered from (Yahoo Finance)
- Summary of structure of the work (description of what is done in each part)

The NASDAQ Composite Index (IXIC) is analyzed from 01.01.2012 to 01.03.2022 (2556 days of data). The data is downloaded from Yahoo Finance and can be found here. As NASDAQ explains in this article "The Nasdaq Composite Index, popularly referred to as 'The Nasdaq' by the media, covers more than 3,000 stocks, all of which are listed on the Nasdaq Stock Market". It is a market-cap weighted index, such that it represents the value of all its listed stocks. Moreover, technology dominates almost half of the composite weight.

As noted, the data is downloaded from Yahoo Finance. This is a free portal that aggregates financial information like market news, stock prices, personal finance information, portfolio management resources and much more.

The mean and conditional variance of the financial time series are modelized, in order to study its volatility. The volatility models can be used to learn several things about the index. First of all, they can be used to predict and interpret future volatility. Additionally, they can be used to interpret the impact of news on the index. Moreover, they can be used to calculate the Value at Risk (VaR). All of these applications are shown in this work. Finally, a multivariate analysis is done, explicitly including the stock of Stratus Properties Inc. (STRS), in order to study some multivariate properties between the two financial time series. Note that STRS is one of the top 30 components of IXIC. Stratus Properties Inc. is a publicly traded company, listed on the NASDAQ. It is a real estate company located in Austin, Texas. As they write on their website, they are "primarily engaged in the acquisition, entitlement, development, management, operation and sale of commercial, hotel, entertainment, and multi- and single-family residential real estate properties, primarily located in the Austin, Texas area, but including projects in certain other select markets in Texas".

The table of contents is shown below. The rest of the report is split into a univariate part and a multivariate part, where the univariate part is the largest and most detailed. Part 3.1 loads and describes the data in a concise fashion, detailing some events that may be related to the changes in the daily adjusted closing price. Section 3.2 analyzes the stationarity of the series, which leads to the conclusion that IXIC is in fact non-stationary. The returns of the series are stationary however, which means that they are used in the remainder of the work instead. Section 3.3 presents some basic statistical properties of the returns. Section 3.4 and 3.5 build models for the mean and variance, respectively, of the stationary series. Section 3.6 presents a grafic and interpretation of the volatility series estimated by the model found in previous sections, while section 3.7 shows the news impact curve of the modelized series. Part 3.8 does volatility predictions and interpretations, based on the models estimated previously. Section 3.9 calculates volatility with two other methods which have not been used earlier in the work; historical volatility and Exponentially Weighted Moving Average (EWMA). Finally in section 3, part 3.10 calculates and interprets the Value at Risk (VaR). The second main part of the report treats a multivariate analysis of IXIC, coupled with STRS. In section 4.1 a multivariate DCC GARCH model is fitted to the residuals of the time series. In this case, the identification, estimation and diagnostics of the models for the mean and variance of the returns of STRS is not shown. Section 4.2 estimates the correlation between the two financial series and shows the news impact surface, which is the bivariate equivalent to the news impact curve. Finally, a conclusion of the work is formulated.

Contents

3 Univariate Analysis

3.1 Description of Data

First, we load the NASDAQ Composite Index data from Yahoo Finance. Note that I downloaded the data in csv format instead of loading directly via the `quantmod getSymbols` API, in order to make sure that I always have access to the data.

```
#getSymbols("^IXIC",from="2012-01-01", to="2022-03-01", warnings = F)
Ixic <- read.csv("IXIC.csv")
dim(Ixic)
```

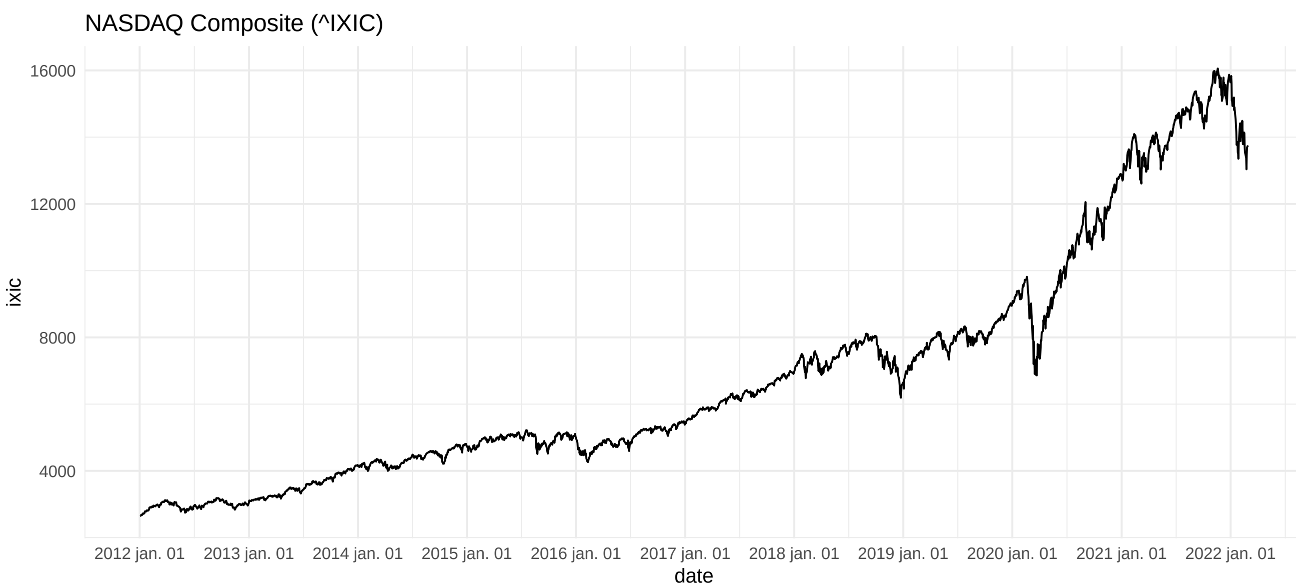
```
#> [1] 2556    7
```

```
any(is.na(Ixic))
```

```
#> [1] FALSE
```

```
#dim(IXIC)
# Want the adjusted closing price.
ixic <- Ixic[,6]
#IXIC <- IXIC[,6]
```

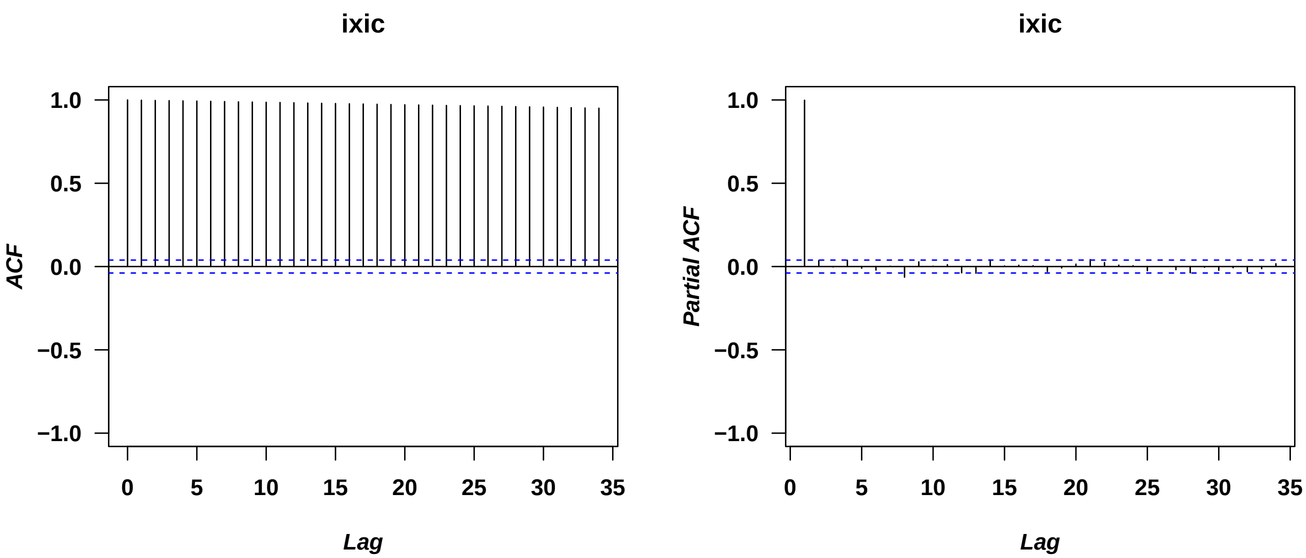
The data does not have any NA values (weekends and holidays have been removed already), such that we can start working with the data without the need to replace missing values. The series is plotted below.



We note some important moments during the time in question. The index fell in January 2016, perhaps in relation to the 2015-2016 stock market sellof or a slowdown in China and falling oil prices, as noted here. In January 2019 there was a stock market crash following an announcement from Apple's CEO Tim Cook. Moreover, the market slump was dependent on weak Chinese manufacturing data, as noted here. The relatively large fall in price in the beginning of 2020 was a result of the spread of COVID-19. This event leads up to the fall in the beginning of 2022, when Russia eventually launched an invasion on Ukraine on February 24.

3.2 Analysis of Stationarity

In order to see if the series is stationary, we will employ both informal and formal tests. Immediately, by looking at the plot of the series, it does not look stationary, since the mean of the process looks to change quite dramatically with time. Some more informal tests are done. The function of autocorrelation and partial autocorrelation (empirical) for the series are plotted below.



As is seen from the function of autocorrelation (ACF), the coefficients decrease slowly towards zero. This suggests that the time series is non-stationary, since a stationary series would show quickly decreasing autocorrelation coefficients.

Next, some Ljung-Box tests are done. Here we are testing the joint hypothesis that all m of the correlation coefficients are simultaneously equal to zero. Below we are testing for $m \in \{1, 5, 10, 15, 20\}$. Only the first output is shown, because all of them give very low p-values.

```
Box.test(ixic, lag = 1, type = c("Ljung-Box"))
```

```
#>
#>  Box-Ljung test
#>
#> data:  ixic
#> X-squared = 2551.4, df = 1, p-value < 2.2e-16
```

```
Box.test(ixic, lag = 5, type = c("Ljung-Box"))
Box.test(ixic, lag = 10, type = c("Ljung-Box"))
Box.test(ixic, lag = 15, type = c("Ljung-Box"))
Box.test(ixic, lag = 20, type = c("Ljung-Box"))
```

The low p-values mean that, to any reasonably prespecified significance level (often at 5%), the null hypothesis that all m correlation coefficients are simultaneously equal to zero is rejected. This further suggests that the series is non-stationary.

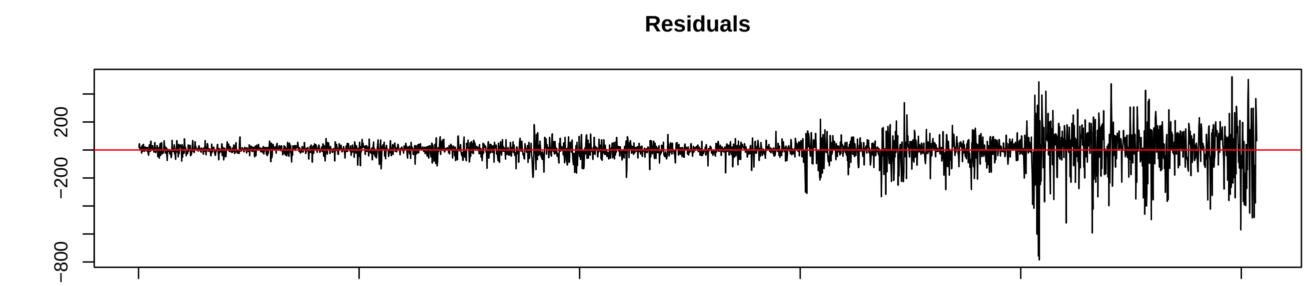
Next, some formal tests are done to check stationarity of the series. First, the Augmented-Dickey-Fuller (ADF) unit root test is done. The null hypothesis of this test states that the series is integrated of order 1, i.e. that it is non-stationary. Below, the ADF test is done assuming both a stochastic and deterministic trend in the data. The maximum number of lags considered are 20 and the number of lags used are chosen by BIC.

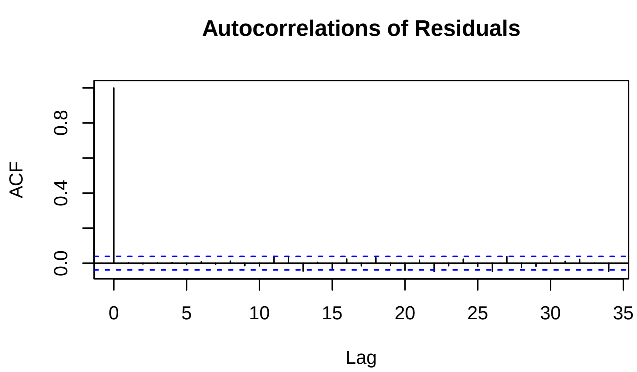
```
ixic.df<-ur.df(ixic, type = c("trend"), lags=20, selectlags = c("BIC"))
summary(ixic.df)
```

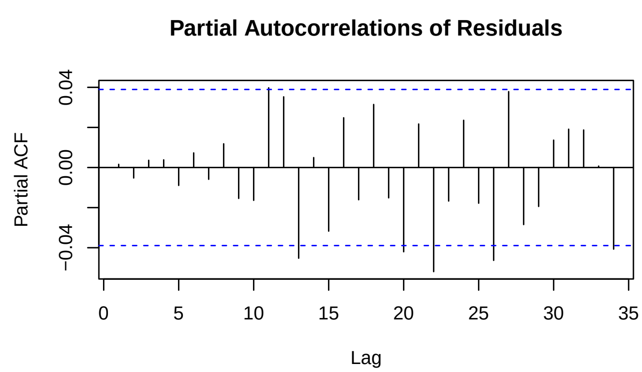
```
#>
#> ###############################################
#> # Augmented Dickey-Fuller Test Unit Root Test #
#> ###############################################
#>
#> Test regression trend
#>
#>
#> Call:
#> lm(formula = z.diff ~ z.lag.1 + 1 + tt + z.diff.lag)
#>
#> Residuals:
#>     Min      1Q  Median      3Q     Max
#> -785.57  -28.42    3.81   35.46  523.61
```

```
#>
#> Coefficients:
#>              Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  4.013754   4.319321   0.929  0.35285
#> z.lag.1     -0.002490   0.001457  -1.709  0.08751 .
#> tt           0.013739   0.006845   2.007  0.04482 *
#> z.diff.lag1 -0.090268   0.019879  -4.541 5.87e-06 ***
#> z.diff.lag2  0.051188   0.019870   2.576  0.01005 *
#> z.diff.lag3 -0.004620   0.019903  -0.232  0.81646
#> z.diff.lag4 -0.062694   0.019939  -3.144  0.00168 **
#> z.diff.lag5  0.007115   0.019994   0.356  0.72197
#> z.diff.lag6 -0.026554   0.019982  -1.329  0.18401
#> z.diff.lag7  0.084723   0.020069   4.222 2.51e-05 ***
#> z.diff.lag8 -0.104673   0.020111  -5.205 2.10e-07 ***
#> z.diff.lag9  0.062169   0.020173   3.082  0.00208 **
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 97.42 on 2523 degrees of freedom
#> Multiple R-squared:  0.05128,    Adjusted R-squared:  0.04715
#> F-statistic:  12.4 on 11 and 2523 DF,  p-value: < 2.2e-16
#>
#>
#> Value of test-statistic is: -1.7093 3.3031 2.0816
#>
#> Critical values for test statistics:
#>       1pct  5pct 10pct
#> tau3 -3.96 -3.41 -3.12
#> phi2  6.09  4.68  4.03
#> phi3  8.27  6.25  5.34
```

From the output it is apparent that BIC chooses 9 lags in the ADF test. Moreover, the value of the test-statistic clearly suggests that we cannot reject the null-hypothesis, since the value is much larger than the critical values for this left-sided test. Thus, we would conclude that the series is non-stationary. Note that the test leads to the same conclusion when assuming no trends and when assuming only a drift. Moreover, the same amount of lags were chosen automatically for all three variants. Below, the residuals and the autocorrelation functions of the residuals are plotted, in order to check if the number of lags chosen via BIC is satisfactory.





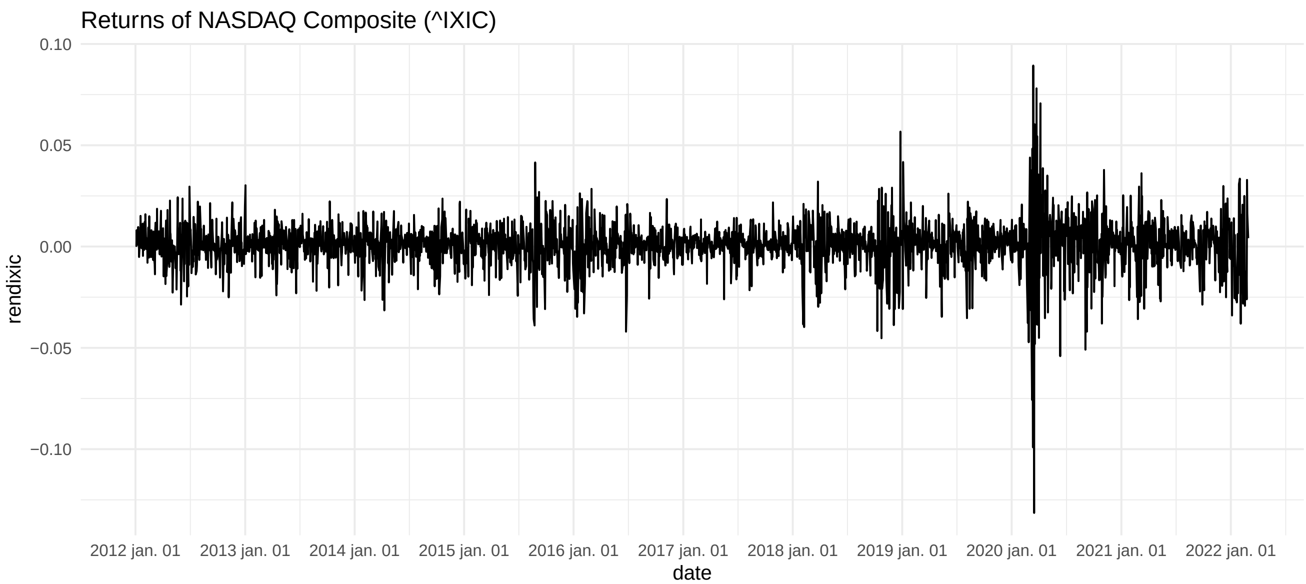


The autocorrelation function of the residuals has no significant coefficients, which leads us to conclude that the amount of lags chosen via BIC is satisfactory.

Next, we check if the returns are stationary. To be clear, the word "returns" is used to refer to the continuously compounded returns. Thus, whenever we talk about returns, we are referring to the continuously compounded returns of the series.

```
rendixic <- diff(log(ixic))
```

The returns are plotted below.



Then, the ADF test is calculated without trends, since there does not look to be any trends in the plot of the returns. Note that, as earlier, the conclusion of the test and the amount of lags that are chosen via BIC are the same when assuming a drift or both types of trends.

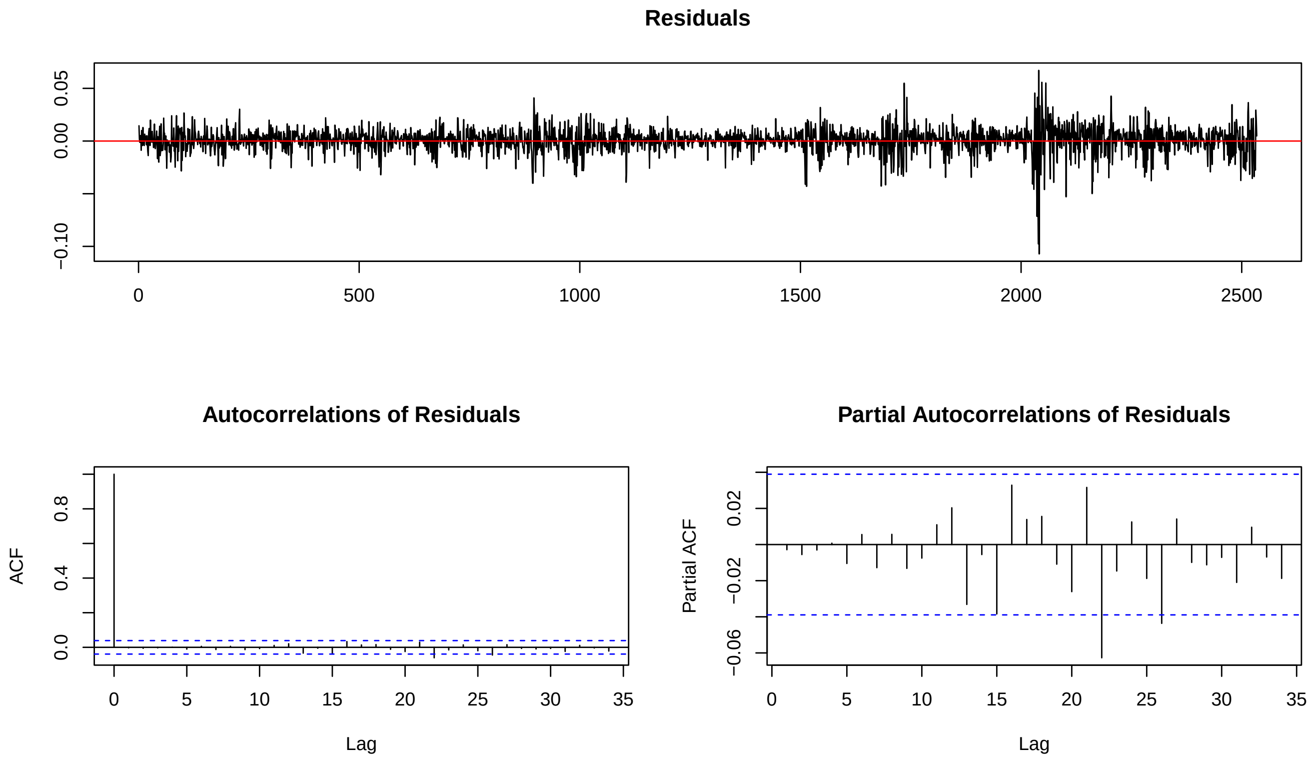
```
rendixic.df<-ur.df(rendixic, type = c("none"), lags=20, selectlags = c("BIC"))
summary(rendixic.df)
```

```
#>
#> ###############################################
#> # Augmented Dickey-Fuller Test Unit Root Test #
#> ###############################################
#>
#> Test regression none
#>
#>
#> Call:
#> lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)
#>
#> Residuals:
#>       Min        1Q    Median        3Q       Max
#> -0.107159 -0.004275  0.001424  0.006875  0.067084
#>
#> Coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> z.lag.1     -1.07192    0.06437 -16.651  < 2e-16 ***
#> z.diff.lag1 -0.02421    0.06026  -0.402 0.687881
#> z.diff.lag2  0.02301    0.05659   0.407 0.684281
#> z.diff.lag3  0.02650    0.05193   0.510 0.609830
#> z.diff.lag4 -0.02609    0.04723  -0.552 0.580728
#> z.diff.lag5 -0.01914    0.04188  -0.457 0.647707
#> z.diff.lag6 -0.06599    0.03630  -1.818 0.069208 .
#> z.diff.lag7  0.02827    0.02966   0.953 0.340668
#> z.diff.lag8 -0.06861    0.01996  -3.437 0.000597 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#>
#> Residual standard error: 0.01167 on 2525 degrees of freedom
#> Multiple R-squared:  0.5826, Adjusted R-squared:  0.5811
#> F-statistic: 391.6 on 9 and 2525 DF,  p-value: < 2.2e-16
#>
#>
#> Value of test-statistic is: -16.6512
#>
#> Critical values for test statistics:
#>       1pct  5pct 10pct
#> tau1 -2.58 -1.95 -1.62
```

It is apparent that 8 lags are chosen. Moreover, from the test-statistic above we would reject the null-hypothesis, which means that we have found evidence against the hypothesis that the returns are $I(1)$. Thus, we conclude that the returns are $I(0)$ or, equivalently, the original series is $I(1)$. This means that the original series is not stationary according to this test as well, but the returns are stationary and can be used in the analysis.

As earlier, the plot below shows that the amount of lags for the ADF test chosen via BIC is satisfactory.



For completeness, we also use the Philips-Perron (PP) test to check stationarity of the series. This test defines the same null-hypothesis as the ADF test, which means that this also is a left-tailed test. The output from the code blocks below are not printed, as they yield the same results as the ADF test above.

```
ixic.pp<-ur.pp(ixic, type = c("Z-tau"), model = c("trend"), lags = c("long"))
```

All combinations of trend assumptions and long or short lags yield the same conclusions; we have not found sufficient evidence to reject the null-hypothesis of non-stationarity of the series. Below the PP-test is done with the returns.

```
rendixic.pp<-ur.pp(rendixic, type = c("Z-tau"), model = c("constant"), lags = c("short"))
```

When referring to the returns, the conclusion is the same as for the ADF test; the returns are stationary while the original series is not.

Finally, we use the KPSS test to check stationarity of the series. The null hypothesis of this test states that the series is stationary. In the test below we have chosen to assume the deterministic component as a constant with a linear trend, and we have used short lags. Notice that the conclusion is the same with all different variations of assumptions for the test.

```r
ixic.kpss<-ur.kpss(ixic, type = c("tau"), lags = c("short"))
summary(ixic.kpss)
```

```
#>
#> #######################
#> # KPSS Unit Root Test #
#> #######################
#>
#> Test is of type: tau with 8 lags.
#>
#> Value of test-statistic is: 4.8701
#>
#> Critical value for a significance level of:
#>                 10pct  5pct 2.5pct  1pct
#> critical values 0.119 0.146  0.176 0.216
```

Since this is a right-tailed test, the test-statistic is clearly sufficiently large to reject the null-hypothesis to the lowest significance level shown (0.01). Thus, we conclude that the series is non-stationary, as expected. The test below shows that the returns are stationary, in line with what we have concluded earlier, since we cannot find strong evidence against the null-hypothesis.

```r
rendixic.kpss <- ur.kpss(rendixic, type = c("mu"), lags = c("short"))
summary(rendixic.kpss)
```

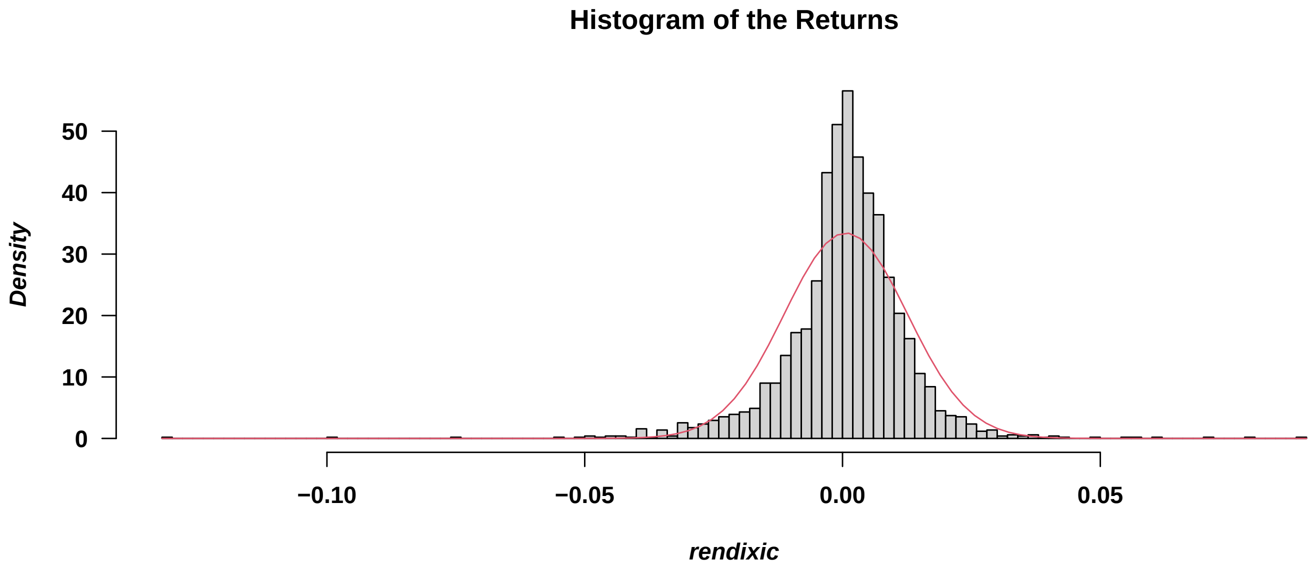
```
#>
#> #######################
#> # KPSS Unit Root Test #
#> #######################
#>
#> Test is of type: mu with 8 lags.
#>
#> Value of test-statistic is: 0.0313
#>
#> Critical value for a significance level of:
#>                 10pct  5pct 2.5pct  1pct
#> critical values 0.347 0.463  0.574 0.739
```

Conclusively, the original time series is not stationary, but the returns are stationary, which means that the returns will be used in the following analysis. We can be relatively certain that this is the case, since all three formal tests, as well as the informal tests, point to this conclusion.

3.3 Basic Statistical Properties of the Stationary Series

Some basic statistical properties of the stationary series, the returns, are shown below.

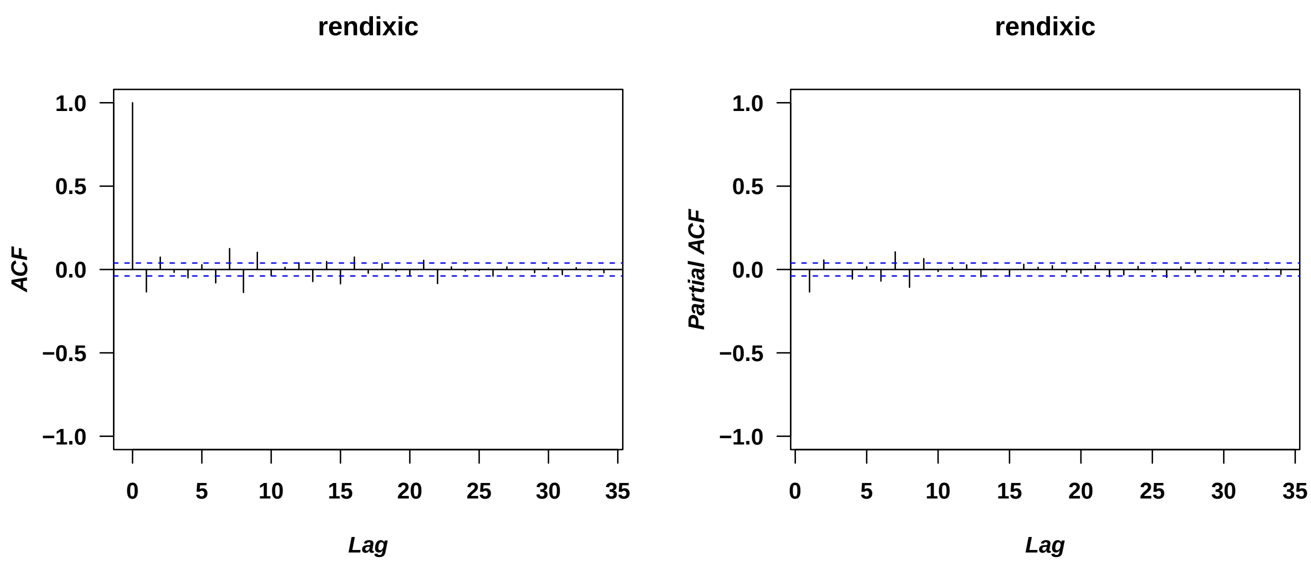
```
#>                rendixic
#> nobs        2555.000000
#> NAs            0.000000
#> Minimum       -0.131492
#> Maximum        0.089347
#> 1. Quartile   -0.003980
#> 3. Quartile    0.006759
#> Mean           0.000645
#> Median         0.001093
#> Sum            1.647064
#> SE Mean        0.000236
#> LCL Mean       0.000182
#> UCL Mean       0.001108
#> Variance       0.000142
#> Stdev          0.011933
#> Skewness      -0.841975
#> Kurtosis      12.309021
```



It becomes apparent that the series is leptokurtic, both from the kurtosis value and from the histogram. The superposed red curve is a Gaussian distribution with empirical mean and standard error according to the returns of IXIC. Moreover, the skewness is negative, which means that the distribution of the returns is heavy-tailed in the left tail. This is also apparent from the histogram above. Without any further comments, the rest of the statistical properties may be interesting to have in mind.

3.4 Identification, Estimation and Diagnostics of a Model for the Mean

The autocorrelation functions of the returns are plotted below.



Note that the third coefficient of both ACF and PACF seems to be non-significant, which might be a hint to what order of model would be fitting. Notice also that both the ACF and the PACF have significant coefficients after the third lag; 6, 7, 8 and 9 seem to be significant. An ARMA of order 6, 7, 8 or 9 seems like a too large order of model to estimate, so we will try with smaller models instead, noting that the third coefficient is non-significant. The table below shows the BIC and the AIC for different orders of ARMA-models. The largest model that is considered is ARMA(3,2) (or ARMA(2,3)), since an ARMA(3,3) yields NaNs in the estimates.

Note that all models we have estimated here have significant coefficient estimates to a predetermined significance level of $\alpha = 0.05$.

Table 1: AIC and BIC of different estimated models for the returns of IXIC

Model	BIC	AIC
AR(1)	-15402.7116191511	-15420.249041659
MA(1)	-15397.436369982	-15414.9737924899
ARMA(1,1)	-15400.9107843157	-15424.2940143262
AR(2)	-15403.1541772126	-15426.5374072231
MA(2)	-15402.9296919165	-15426.312921927
ARMA(2,2)	-15470.3358282124	-15505.4106732282
AR(3)	-15395.308473024	-15424.5375105372
MA(3)	-15397.4156685398	-15426.644706053
ARMA(3,2)	-15386.1619828535	-15427.082635372

The table above clearly shows that ARMA(2,2) yields the lowest AIC and BIC. The estimated model ARMA(2,2) is shown below

```
(mean.model <- arima(rendixic, order = c(2,0,2),include.mean = TRUE))
```

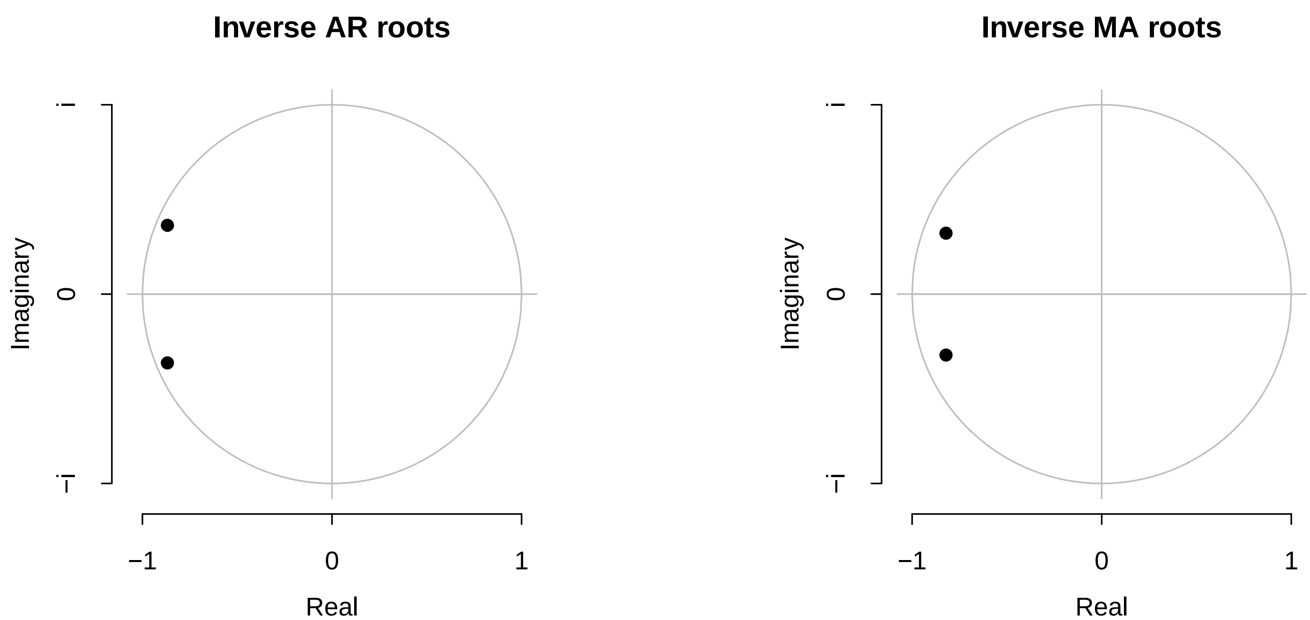
```
#>
#> Call:
#> arima(x = rendixic, order = c(2, 0, 2), include.mean = TRUE)
#>
#> Coefficients:
#>           ar1      ar2     ma1    ma2  intercept
#>       -1.7362  -0.8856  1.6425  0.778      6e-04
#> s.e.   0.0241   0.0226  0.0326  0.030      2e-04
#>
#> sigma^2 estimated as 0.0001349:  log likelihood = 7758.71,  aic = -15505.41
```

```
pnorm(c(abs(mean.model$coef)/sqrt(diag(mean.model$var.coef))), mean=0, sd=1, lower.tail=FALSE)
```

```
#>           ar1           ar2           ma1           ma2     intercept
#>  0.000000e+00  0.000000e+00  0.000000e+00 7.280516e-149  1.595550e-03
```

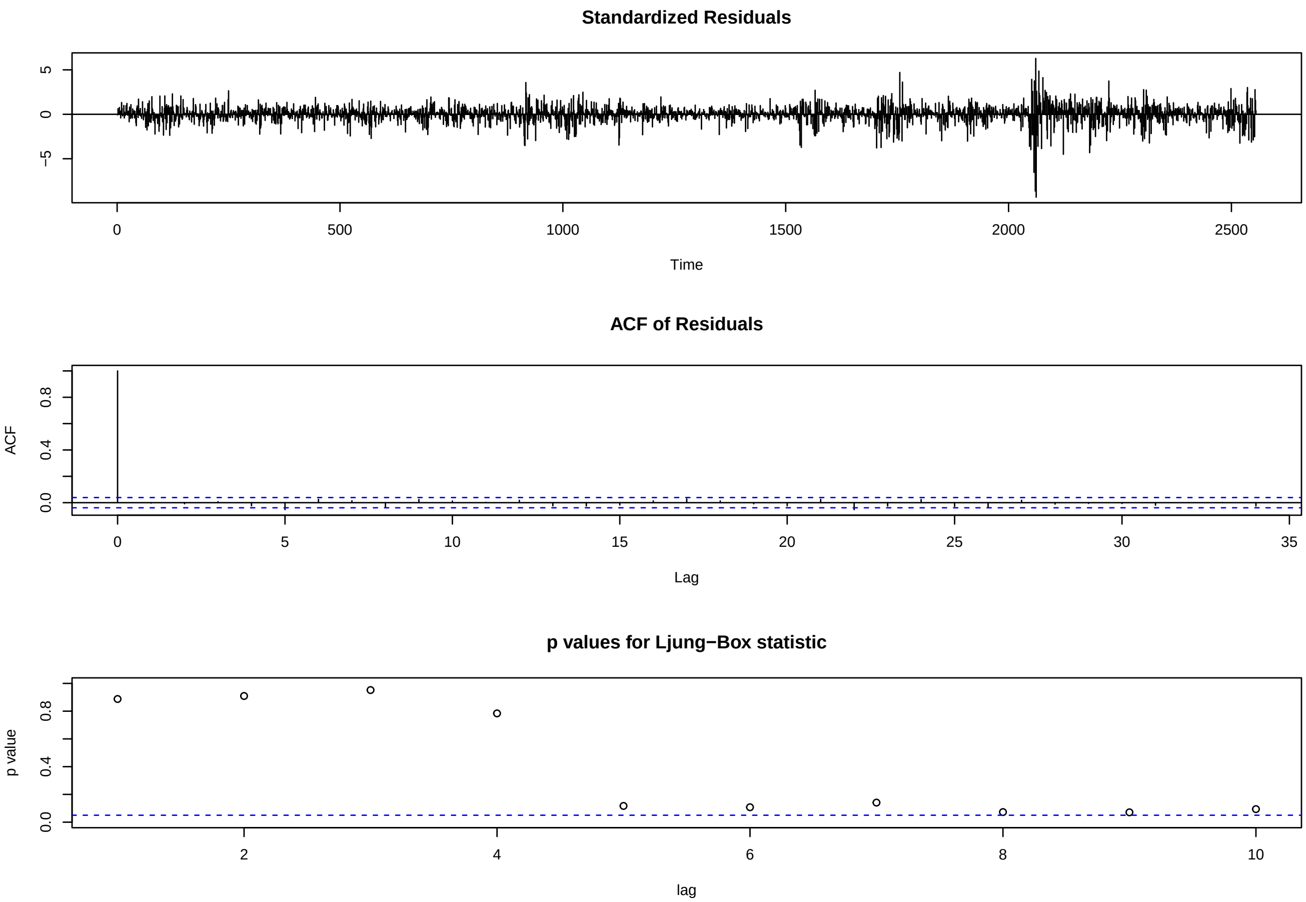
```
residuals <- c(mean.model$residuals)
```

Some model diagnostics have to be done to check if the model is adequate. We must check if the model is stationary. The inverse roots of the characteristic polynomial of AR and MA are plotted.



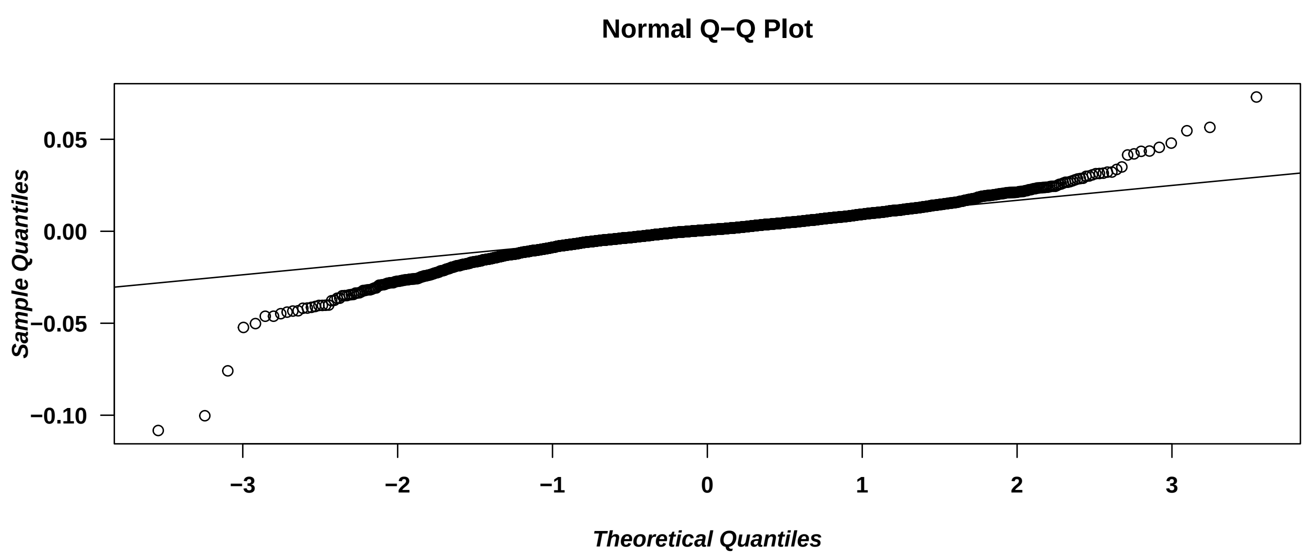
The stationarity condition for the AR-process is satisfied, since the roots have absolute values greater than one. Moreover, the invertibility condition holds for the MA process, since the roots of this process also have absolute values greater than one. Thus, the full model is stationary.

The residuals of the model are analyzed next.

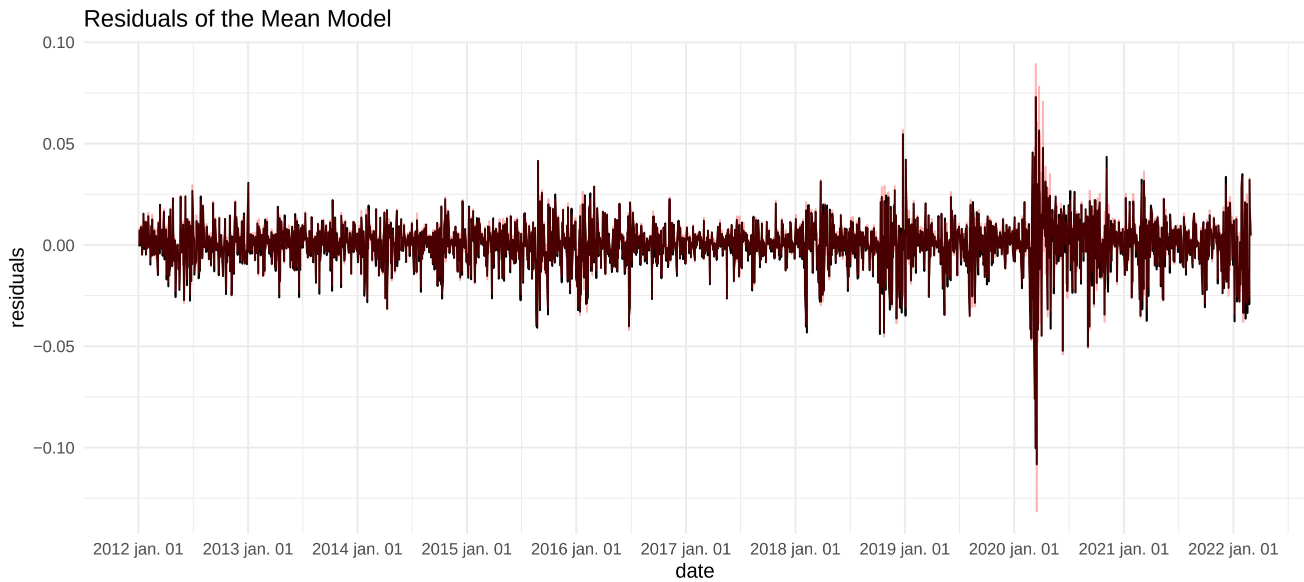


There are no significant coefficients in the autocorrelation function, which suggests that the model has adequately captured the information in the data. Moreover, the Ljung-Box statistic p-values are all relatively large, which means that we will not reject the Ljung-Box null hypothesis. This further suggests that the residuals are not correlated and we have found a model that seems reasonable in this regard.

Next, a QQ-plot of the theoretical normal quantiles, and the residuals themselves (not standardized), is plotted.



Also, the residuals of the estimated model are plotted, with the returns of IXIC superposed in light red. Notice that, qualitatively, they look similar, which indicates that the model for the mean does not have a whole lot of explicative power on the returns.



The Jarque-Bera Normality test is also applied.

```
normalTest(residuals,method="jb")
```

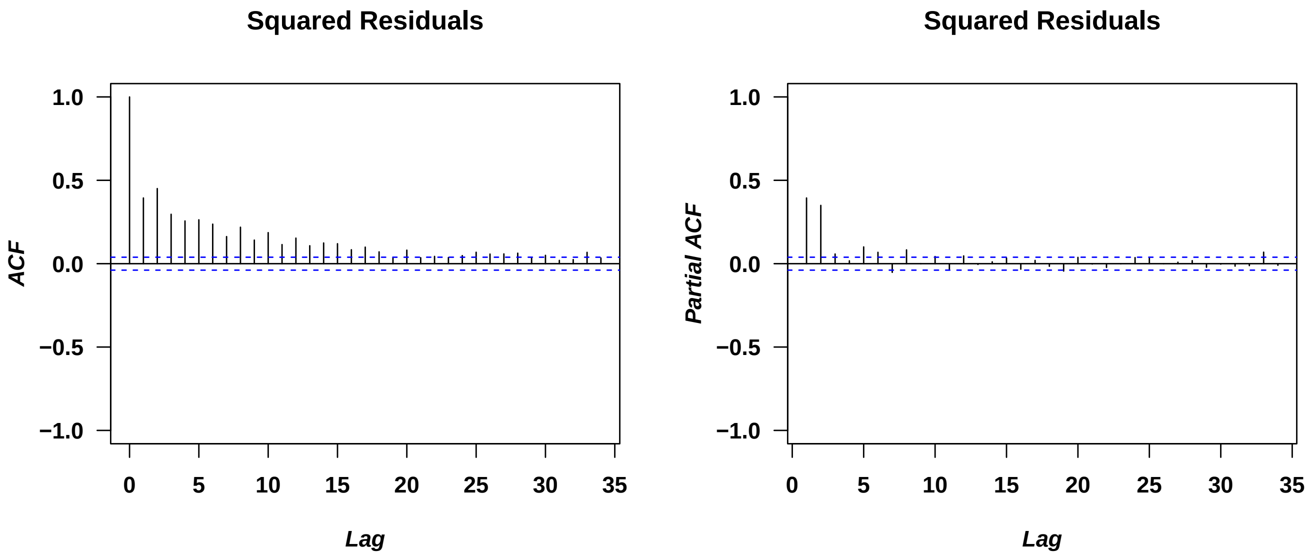
```
#>
#> Title:
#>  Jarque - Bera Normalality Test
#>
#> Test Results:
#>   STATISTIC:
#>     X-squared: 7472.6779
#>   P VALUE:
#>     Asymptotic p Value: < 2.2e-16
#>
#> Description:
#>  Wed Apr  6 12:42:16 2022 by user: ajo
```

It is apparent that the residuals have heavy tails. It is not reasonable to assume normality of the residuals, an argument that the Jarque-Bera Normality test further substantiates because its null hypothesis of normality is rejected following the very small p-value.

3.5 Identification, Estimation and Diagnostics of a Model for the Variance

First we test for ARCH effects using the residuals of the mean model.

```
residuals.squared <- residuals^2
```



As seen in the ACF and PACF of the squared residuals, they are clearly presenting autocorrelation, i.e. there are ARCH effects present.

```
Box.test(residuals.squared, lag=1,type='Ljung')
```

```
#>
#>  Box-Ljung test
#>
#> data:  residuals.squared
#> X-squared = 397.36, df = 1, p-value < 2.2e-16
```

```
Box.test(residuals.squared,lag=5,type='Ljung')
Box.test(residuals.squared,lag=15,type='Ljung')
```

The argument is further substantiated by the Ljung-Box tests, where only the first result is shown, since they all lead to the conclusion that the squared residuals are correlated, because the null hypothesis is rejected. Thus, it is relevant to identify and estimate a model for the volatility. Joint estimation of the mean and volatility equations, for different types of models, is done in the following.

Now over to estimation of GARCH models for the variance of the returns. First we estimate a ARMA(2,2)-GARCH(1,1), assuming the error term is conditionally t-student distributed. We will make this same assumption in all the following estimations.

```
spec1 <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1,1)),
          mean.model=list(armaOrder=c(2,2)), distribution.model = "std")
(m <- ugarchfit(spec = spec1, data = rendixic))
```

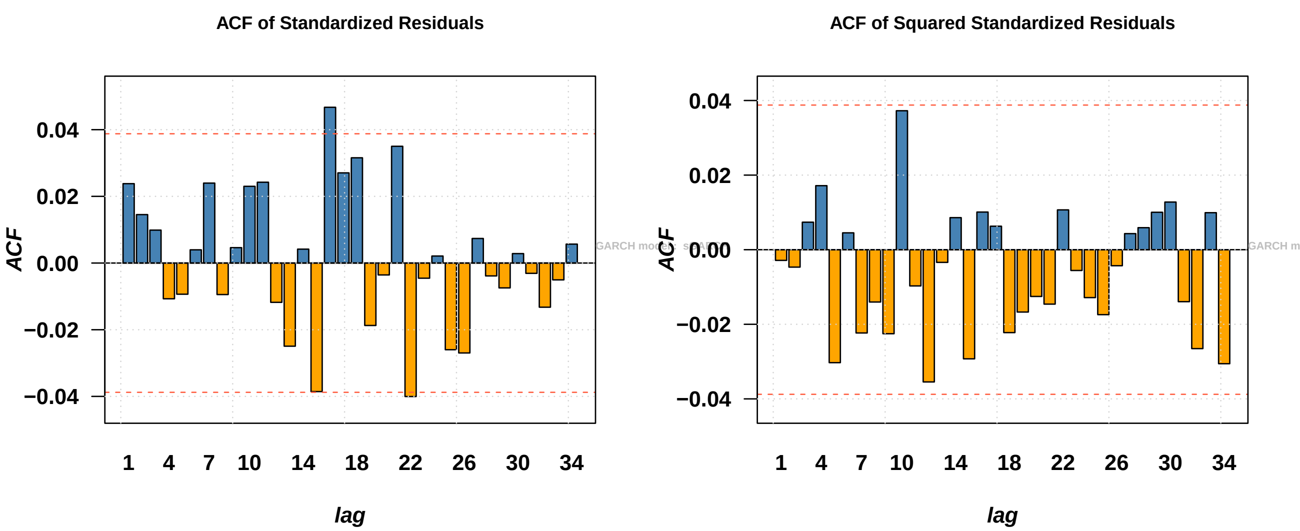
```
#>
#> *---------------------------------*
#> *          GARCH Model Fit        *
#> *---------------------------------*
#>
#> Conditional Variance Dynamics
#> -----------------------------------
#> GARCH Model  : sGARCH(1,1)
#> Mean Model   : ARFIMA(2,0,2)
#> Distribution : std
#>
#> Optimal Parameters
#> ------------------------------------
#>         Estimate  Std. Error  t value Pr(>|t|)
#> mu      0.001230    0.000085  14.4538 0.000000
#> ar1     0.411215    0.010896  37.7392 0.000000
#> ar2     0.468396    0.002407 194.5656 0.000000
#> ma1    -0.467394    0.010241 -45.6404 0.000000
#> ma2    -0.455194    0.009398 -48.4364 0.000000
#> omega   0.000005    0.000002   2.2329 0.025555
#> alpha1  0.163393    0.020408   8.0062 0.000000
```

```
#> beta1   0.813486    0.023304  34.9081 0.000000
#> shape   5.240364    0.590405   8.8759 0.000000
#>
#> Robust Standard Errors:
#>         Estimate  Std. Error   t value Pr(>|t|)
#> mu      0.001230    0.000101  12.15400  0.00000
#> ar1     0.411215    0.026259  15.65987  0.00000
#> ar2     0.468396    0.026987  17.35607  0.00000
#> ma1    -0.467394    0.016152 -28.93808  0.00000
#> ma2    -0.455194    0.015014 -30.31821  0.00000
#> omega   0.000005    0.000005   0.95667  0.33873
#> alpha1  0.163393    0.020752   7.87359  0.00000
#> beta1   0.813486    0.039750  20.46509  0.00000
#> shape   5.240364    0.726690   7.21127  0.00000
#>
#> LogLikelihood : 8262.277
#>
#> Information Criteria
#> ------------------------------------
#>
#> Akaike       -6.4605
#> Bayes        -6.4399
#> Shibata      -6.4605
#> Hannan-Quinn -6.4530
#>
#> Weighted Ljung-Box Test on Standardized Residuals
#> ------------------------------------
#>                          statistic p-value
#> Lag[1]                       1.450  0.2286
#> Lag[2*(p+q)+(p+q)-1][11]     3.676  1.0000
#> Lag[4*(p+q)+(p+q)-1][19]     8.810  0.6701
#> d.o.f=4
#> H0 : No serial correlation
#>
#> Weighted Ljung-Box Test on Standardized Squared Residuals
#> ------------------------------------
#>                         statistic p-value
#> Lag[1]                    0.02101  0.8847
#> Lag[2*(p+q)+(p+q)-1][5]   0.92104  0.8773
#> Lag[4*(p+q)+(p+q)-1][9]   2.69363  0.8083
#> d.o.f=2
#>
#> Weighted ARCH LM Tests
#> ------------------------------------
#>             Statistic Shape Scale P-Value
#> ARCH Lag[3]    0.1391 0.500 2.000  0.7092
#> ARCH Lag[5]    2.1473 1.440 1.667  0.4396
#> ARCH Lag[7]    3.0338 2.315 1.543  0.5072
#>
#> Nyblom stability test
#> ------------------------------------
#> Joint Statistic:  1.8626
#> Individual Statistics:
#> mu     0.37609
#> ar1    0.06996
#> ar2    0.12329
#> ma1    0.07412
#> ma2    0.11176
#> omega  0.24242
#> alpha1 0.64422
#> beta1  0.45667
#> shape  0.47015
#>
#> Asymptotic Critical Values (10% 5% 1%)
#> Joint Statistic:          2.1 2.32 2.82
#> Individual Statistic:     0.35 0.47 0.75
#>
#> Sign Bias Test
#> ------------------------------------
#>                    t-value     prob sig
```

```
#> Sign Bias           2.3681 0.017953  **
#> Negative Sign Bias  0.1143 0.909045
#> Positive Sign Bias  0.9489 0.342756
#> Joint Effect       15.5615 0.001395 ***
#>
#>
#> Adjusted Pearson Goodness-of-Fit Test:
#> ------------------------------------
#>   group statistic p-value(g-1)
#> 1    20     66.58    3.371e-07
#> 2    30     89.80    3.904e-08
#> 3    40     96.76    8.200e-07
#> 4    50    118.17    1.208e-07
#>
#>
#> Elapsed time : 0.8611135
```

```
m.AIC <- -6.4605
```

We observe that all the parameter estimates are significant to a 5% significance level. Moreover, we note that the condition of positivity holds, because $\hat{\alpha}_1 \geq 0$ and $\hat{\beta}_1 \geq 0$, where we follow the standard statistical notation of a hat indicating an estimate. Also, we note that the condition of stationarity holds, because $\hat{\alpha}_1 + \hat{\beta}_1 < 1$. We record the AIC of this first model in order to compare to other models later. The ACF of the residuals, plotted below, shows that the residuals do not present any autocorrelation (before moving to around 15 lags, which is a large number of lags), which indicates that this model has modeled the data in a sufficient or reasonable way. However, note that the p-values of the `Weighted Ljung-Box Test on Standardized Residuals` are quite large, which means we cannot reject the null hypothesis of no serial correlation for the different lags. As will become apparent in the following, this is actually the case for all the models we estimate, which means that it cannot be used to differentiate between them. However, it can be noted as a disadvantage of all these models in general.



Some quick interpretations of some of the parameters in the standard GARCH-model; as noted in the documentation of `rugarch`, α_1 is the ARCH term and β_1 is the GARCH term in the estimated model. This means that α_1 essentially measures the reaction of conditional volatility due to market shocks and that β_1 essentially measures the persistence of conditional volatility. When $\alpha_1 > 0.1$, which is the case here, it is regarded as large and we say that the volatility is very sensitive to market events. When $\beta_1 > 0.9$, we say that the volatility is very persistent, which is not the case here.

Next we will fit a ARMA(2,2)-GJR-GARCH(1,1) model, assuming a t-distribution.

```
spec.mgjr <- ugarchspec(variance.model=list(model="gjrGARCH", garchOrder = c(1,1)),
                    mean.model=list(armaOrder=c(2,2)), distribution.model = "std")
(mgjr <- ugarchfit(spec = spec.mgjr, data = rendixic))
```

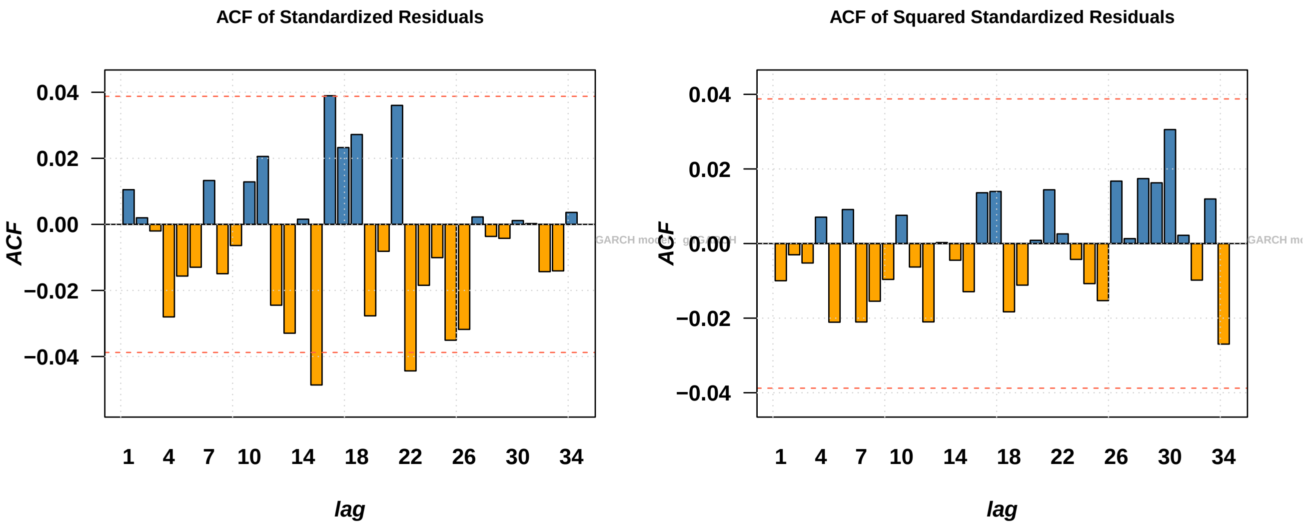
```
#>
#> *---------------------------------*
#> *          GARCH Model Fit        *
```

```
#> *---------------------------------*
#>
#> Conditional Variance Dynamics
#> -----------------------------------
#> GARCH Model  : gjrGARCH(1,1)
#> Mean Model   : ARFIMA(2,0,2)
#> Distribution : std
#>
#> Optimal Parameters
#> ------------------------------------
#>         Estimate  Std. Error   t value Pr(>|t|)
#> mu      0.001047    0.000138  7.606420 0.000000
#> ar1     0.275227    0.319606  0.861146 0.389157
#> ar2     0.470193    0.205718  2.285615 0.022277
#> ma1    -0.320370    0.321325 -0.997029 0.318750
#> ma2    -0.455726    0.214102 -2.128549 0.033292
#> omega   0.000005    0.000000 15.301373 0.000000
#> alpha1  0.000000    0.009195  0.000002 0.999999
#> beta1   0.823974    0.014252 57.815933 0.000000
#> gamma1  0.260477    0.031890  8.168122 0.000000
#> shape   5.594916    0.591457  9.459543 0.000000
#>
#> Robust Standard Errors:
#>         Estimate  Std. Error   t value Pr(>|t|)
#> mu      0.001047    0.000130  8.037487 0.000000
#> ar1     0.275227    0.196628  1.399736 0.161593
#> ar2     0.470193    0.119749  3.926475 0.000086
#> ma1    -0.320370    0.198050 -1.617626 0.105743
#> ma2    -0.455726    0.122730 -3.713240 0.000205
#> omega   0.000005    0.000001  9.596872 0.000000
#> alpha1  0.000000    0.012423  0.000001 0.999999
#> beta1   0.823974    0.013038 63.195840 0.000000
#> gamma1  0.260477    0.035268  7.385755 0.000000
#> shape   5.594916    0.550880 10.156318 0.000000
#>
#> LogLikelihood : 8303.976
#>
#> Information Criteria
#> ------------------------------------
#>
#> Akaike       -6.4923
#> Bayes        -6.4695
#> Shibata      -6.4924
#> Hannan-Quinn -6.4841
#>
#> Weighted Ljung-Box Test on Standardized Residuals
#> ------------------------------------
#>                          statistic p-value
#> Lag[1]                      0.2822  0.5953
#> Lag[2*(p+q)+(p+q)-1][11]    3.0139  1.0000
#> Lag[4*(p+q)+(p+q)-1][19]    8.8996  0.6552
#> d.o.f=4
#> H0 : No serial correlation
#>
#> Weighted Ljung-Box Test on Standardized Squared Residuals
#> ------------------------------------
#>                         statistic p-value
#> Lag[1]                     0.2543  0.6141
#> Lag[2*(p+q)+(p+q)-1][5]    0.5938  0.9423
#> Lag[4*(p+q)+(p+q)-1][9]    1.6828  0.9393
#> d.o.f=2
#>
#> Weighted ARCH LM Tests
#> ------------------------------------
#>             Statistic Shape Scale P-Value
#> ARCH Lag[3]   0.06952 0.500 2.000  0.7920
#> ARCH Lag[5]   0.85359 1.440 1.667  0.7768
#> ARCH Lag[7]   1.59629 2.315 1.543  0.8019
#>
#> Nyblom stability test
```

```
#> ------------------------------------
#> Joint Statistic:  17.2754
#> Individual Statistics:
#> mu     0.7414
#> ar1    0.1534
#> ar2    0.3645
#> ma1    0.1807
#> ma2    0.3876
#> omega  1.3411
#> alpha1 2.3979
#> beta1  0.9204
#> gamma1 0.8213
#> shape  0.3773
#>
#> Asymptotic Critical Values (10% 5% 1%)
#> Joint Statistic:          2.29 2.54 3.05
#> Individual Statistic:     0.35 0.47 0.75
#>
#> Sign Bias Test
#> ------------------------------------
#>                    t-value   prob sig
#> Sign Bias           1.2277 0.2197
#> Negative Sign Bias  1.2552 0.2095
#> Positive Sign Bias  0.2208 0.8253
#> Joint Effect        2.7117 0.4383
#>
#>
#> Adjusted Pearson Goodness-of-Fit Test:
#> ------------------------------------
#>   group statistic p-value(g-1)
#> 1    20     70.26    8.326e-08
#> 2    30     85.76    1.611e-07
#> 3    40    109.44    1.324e-08
#> 4    50    129.36    3.569e-09
#>
#>
#> Elapsed time : 1.421381
```

```
mgjr.AIC <- -6.4923
```

The stationarity conditions hold, since $\hat{\alpha}_1 + \hat{\beta}_1 + \frac{1}{2}\hat{\gamma} \approx 0.954 < 1$. Moreover, the positivity condition also holds, since $\hat{\alpha}_1 \geq 0$ and $\hat{\beta}_1 \geq 0$. Notice that the AIC is lower for this model compared to the standard model and the residuals do not present any autocorrelation according to the plots below. Notice that several of the ARMA-parameter coefficient estimates are not significant to a reasonable level, which is found to be the case no matter what `armaOrder` is used in the estimation. The parameters α_1, β_1 and γ_1 can be interpreted as; α_1 measures the reaction of conditional volatility due to market shocks. The estimate is equal to zero and is not significant in this case, which means that it will not be used to quantify this reaction. β_1 measures the persistence of conditional volatility, which has an estimate of $\hat{\beta}_1 \approx 0.82$ in this case, which is not regarded as large. γ_1 measures the extra reaction of the conditional volatility due to negative market shocks. In this case, $\hat{\gamma}_1$ is positive and significant, which means that the volatility increase is more pronounced following a negative return compared to a positive return of the same size. Couple this with the observation that $\hat{\alpha}_1 = 0$, we can interpret that the volatility reactions only take place for negative news, which will become apparent in the news impact curve later on.



Next, we will fit an ARMA(2,2)-EGARCH(1,1) model.

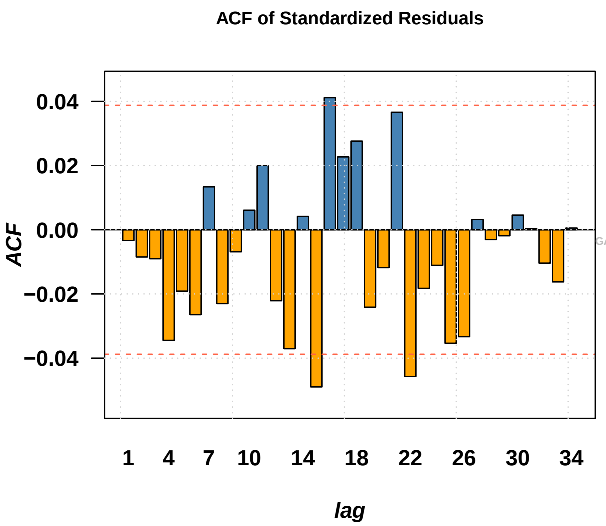
```
spec.egarch <- ugarchspec(variance.model = list(model="eGARCH", garchOrder = c(1,1)),
                    mean.model = list(armaOrder=c(2,2)), distribution.model = "std")
(m.egarch <- ugarchfit(spec = spec.egarch, data = rendixic))
```

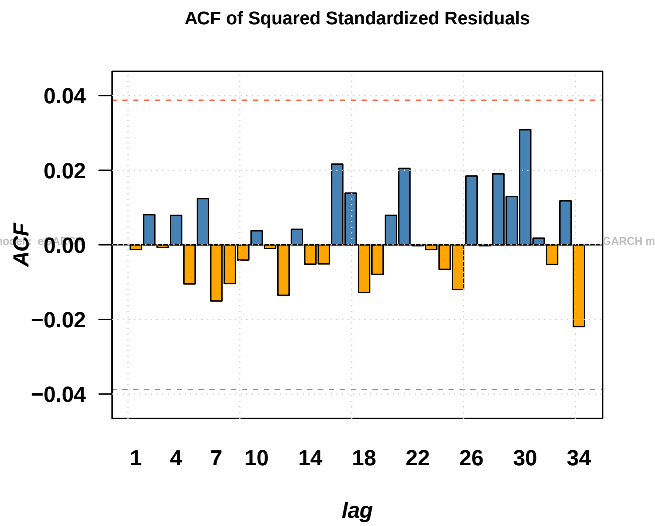
```
#>
#> *---------------------------------*
#> *          GARCH Model Fit        *
#> *---------------------------------*
#>
#> Conditional Variance Dynamics
#> -----------------------------------
#> GARCH Model  : eGARCH(1,1)
#> Mean Model   : ARFIMA(2,0,2)
#> Distribution : std
#>
#> Optimal Parameters
#> ------------------------------------
#>         Estimate  Std. Error  t value Pr(>|t|)
#> mu      0.000945    0.000155   6.1066 0.000000
#> ar1     0.137188    0.047974   2.8596 0.004241
#> ar2     0.310470    0.022509  13.7932 0.000000
#> ma1    -0.178777    0.045824  -3.9014 0.000096
#> ma2    -0.292323    0.022317 -13.0986 0.000000
#> omega  -0.389642    0.013992 -27.8483 0.000000
#> alpha1 -0.193182    0.019053 -10.1394 0.000000
#> beta1   0.958980    0.001550 618.8427 0.000000
#> gamma1  0.161978    0.021992   7.3654 0.000000
#> shape   5.708617    0.654727   8.7191 0.000000
#>
#> Robust Standard Errors:
#>         Estimate  Std. Error  t value Pr(>|t|)
#> mu      0.000945    0.000154   6.1427        0
#> ar1     0.137188    0.006823  20.1069        0
#> ar2     0.310470    0.010004  31.0345        0
#> ma1    -0.178777    0.016029 -11.1534        0
#> ma2    -0.292323    0.009662 -30.2545        0
#> omega  -0.389642    0.008250 -47.2320        0
#> alpha1 -0.193182    0.024672  -7.8300        0
#> beta1   0.958980    0.001166 822.5532        0
#> gamma1  0.161978    0.025847   6.2667        0
#> shape   5.708617    0.685084   8.3327        0
#>
#> LogLikelihood : 8308.361
#>
#> Information Criteria
#> ------------------------------------
#>
```

```
#> Akaike       -6.4958
#> Bayes        -6.4729
#> Shibata      -6.4958
#> Hannan-Quinn -6.4875
#>
#> Weighted Ljung-Box Test on Standardized Residuals
#> ------------------------------------
#>                          statistic p-value
#> Lag[1]                      0.0286  0.8657
#> Lag[2*(p+q)+(p+q)-1][11]    4.9973  0.9583
#> Lag[4*(p+q)+(p+q)-1][19]   11.6665  0.2290
#> d.o.f=4
#> H0 : No serial correlation
#>
#> Weighted Ljung-Box Test on Standardized Squared Residuals
#> ------------------------------------
#>                         statistic p-value
#> Lag[1]                   0.004198  0.9483
#> Lag[2*(p+q)+(p+q)-1][5]  0.258762  0.9877
#> Lag[4*(p+q)+(p+q)-1][9]  0.851583  0.9916
#> d.o.f=2
#>
#> Weighted ARCH LM Tests
#> ------------------------------------
#>             Statistic Shape Scale P-Value
#> ARCH Lag[3]  0.001218 0.500 2.000  0.9722
#> ARCH Lag[5]  0.297744 1.440 1.667  0.9409
#> ARCH Lag[7]  0.811170 2.315 1.543  0.9422
#>
#> Nyblom stability test
#> ------------------------------------
#> Joint Statistic:  4.2243
#> Individual Statistics:
#> mu     1.04997
#> ar1    0.07622
#> ar2    0.30588
#> ma1    0.08127
#> ma2    0.32174
#> omega  1.37628
#> alpha1 0.12155
#> beta1  1.23031
#> gamma1 0.94587
#> shape  0.16952
#>
#> Asymptotic Critical Values (10% 5% 1%)
#> Joint Statistic:         2.29 2.54 3.05
#> Individual Statistic:    0.35 0.47 0.75
#>
#> Sign Bias Test
#> ------------------------------------
#>                    t-value   prob sig
#> Sign Bias          0.32686 0.7438
#> Negative Sign Bias 0.48021 0.6311
#> Positive Sign Bias 0.06459 0.9485
#> Joint Effect       0.24566 0.9699
#>
#>
#> Adjusted Pearson Goodness-of-Fit Test:
#> ------------------------------------
#>   group statistic p-value(g-1)
#> 1    20     78.12    3.914e-09
#> 2    30     97.88    2.130e-09
#> 3    40    100.98    2.135e-07
#> 4    50    118.33    1.151e-07
#>
#>
#> Elapsed time : 0.905262
```

```r
m.egarch.AIC <- -6.4958
```

All estimated parameters are significant to a level of $\alpha = 0.05$. For this model, we do not require positivity of all the GARCH parameter estimates. WHAT ABOUT STATIONARITY, DO WE REQUIRE THIS? The residual plots below do not show any autocorrelation before moving to a large number of lags, which is a good sign that the model has modelized the data adequately. Moreover, the AIC for this model is the smallest value thus far. Notice that $\hat{\alpha}_1 < 0$, which indicates that the model estimates that negative news have a larger effect on the volatility compared to positive news. The reason behind this statement is that α_1 measures the reaction of conditional volatility due to market shocks, where the market shocks are quantified by the returns themselves (instead of some sort of transformation of returns, like squared returns or absolute returns). This implies that when the return is negative, the product of α_1 and the return will be positive, increasing the volatility, whereas when the return is negative the corresponding product will be negative, decreasing the volatility. The observation that negative news lead to higher volatility compared to positive news is coherent with the news impact curve produced by the model, which will be plotted later. As earlier, β_1 quantifies the persistence of conditional volatility. Notice that the parameter γ_1 has a different meaning in this model compared to in the GJR-GARCH; here it quantifies the reaction of conditional volatility due to the absolute value of the returns, i.e. this parameter does not discriminate between negative and positive news. This differentiation between the two types is done via α_1, referring to the previous discussion.





Next, we fit and ARMA(2,2)-IGARCH model.

```
spec.igarch <- ugarchspec(variance.model=list(model="iGARCH", garchOrder = c(1,1)),
                    mean.model=list(armaOrder=c(2,2)), distribution.model = "std")
(m.igarch <- ugarchfit(spec=spec.igarch,data=rendixic))
```

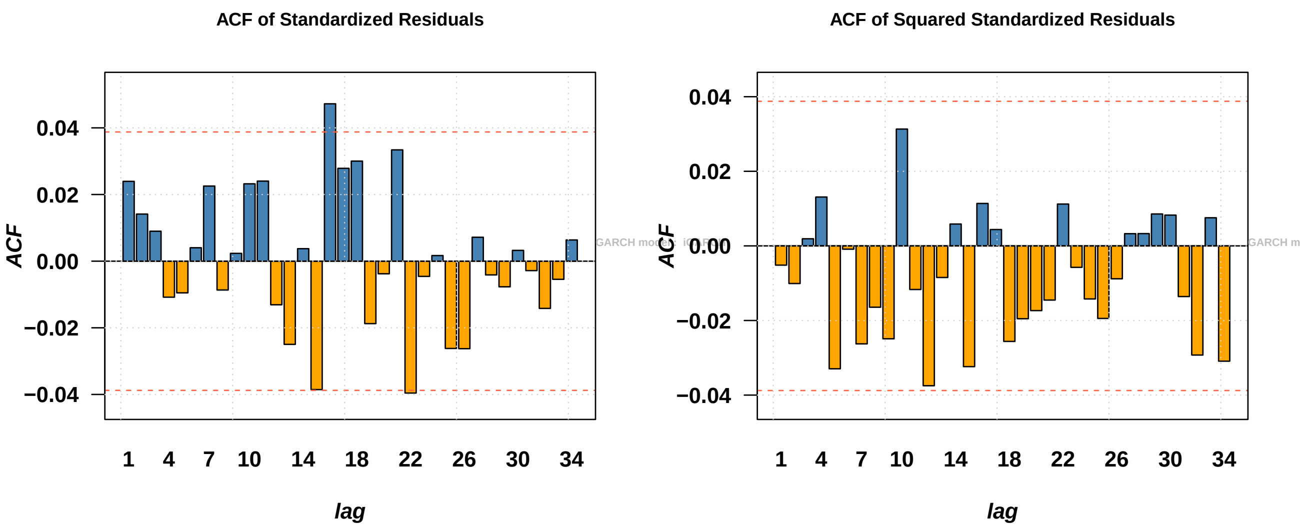
```
#>
#> *---------------------------------*
#> *          GARCH Model Fit        *
#> *---------------------------------*
#>
#> Conditional Variance Dynamics
#> -----------------------------------
#> GARCH Model  : iGARCH(1,1)
#> Mean Model   : ARFIMA(2,0,2)
#> Distribution : std
#>
#> Optimal Parameters
#> ------------------------------------
#>         Estimate  Std. Error  t value Pr(>|t|)
#> mu      0.001238    0.000084  14.7029 0.000000
#> ar1     0.411677    0.010557  38.9943 0.000000
#> ar2     0.467050    0.002286 204.2796 0.000000
#> ma1    -0.468876    0.010466 -44.8003 0.000000
#> ma2    -0.453161    0.009587 -47.2689 0.000000
#> omega   0.000004    0.000002   2.2337 0.025503
#> alpha1  0.181978    0.024164   7.5310 0.000000
#> beta1   0.818022          NA       NA       NA
```

```
#> shape   4.750542    0.428166  11.0951 0.000000
#>
#> Robust Standard Errors:
#>         Estimate  Std. Error   t value Pr(>|t|)
#> mu      0.001238    0.000107  11.53516 0.000000
#> ar1     0.411677    0.027459  14.99226 0.000000
#> ar2     0.467050    0.028844  16.19242 0.000000
#> ma1    -0.468876    0.017833 -26.29308 0.000000
#> ma2    -0.453161    0.016554 -27.37489 0.000000
#> omega   0.000004    0.000004   0.94899 0.342627
#> alpha1  0.181978    0.043129   4.21934 0.000025
#> beta1   0.818022          NA        NA       NA
#> shape   4.750542    0.511909   9.28005 0.000000
#>
#> LogLikelihood : 8260.896
#>
#> Information Criteria
#> ------------------------------------
#>
#> Akaike       -6.4602
#> Bayes        -6.4419
#> Shibata      -6.4602
#> Hannan-Quinn -6.4536
#>
#> Weighted Ljung-Box Test on Standardized Residuals
#> ------------------------------------
#>                          statistic p-value
#> Lag[1]                       1.467  0.2258
#> Lag[2*(p+q)+(p+q)-1][11]     3.540  1.0000
#> Lag[4*(p+q)+(p+q)-1][19]     8.661  0.6944
#> d.o.f=4
#> H0 : No serial correlation
#>
#> Weighted Ljung-Box Test on Standardized Squared Residuals
#> ------------------------------------
#>                         statistic p-value
#> Lag[1]                    0.06793  0.7944
#> Lag[2*(p+q)+(p+q)-1][5]   1.01394  0.8565
#> Lag[4*(p+q)+(p+q)-1][9]   3.06639  0.7480
#> d.o.f=2
#>
#> Weighted ARCH LM Tests
#> ------------------------------------
#>             Statistic Shape Scale P-Value
#> ARCH Lag[3]  0.009302 0.500 2.000  0.9232
#> ARCH Lag[5]  2.024476 1.440 1.667  0.4659
#> ARCH Lag[7]  3.123161 2.315 1.543  0.4906
#>
#> Nyblom stability test
#> ------------------------------------
#> Joint Statistic:  4.8042
#> Individual Statistics:
#> mu     0.36519
#> ar1    0.07278
#> ar2    0.12984
#> ma1    0.07754
#> ma2    0.11797
#> omega  1.56210
#> alpha1 0.13294
#> shape  0.42921
#>
#> Asymptotic Critical Values (10% 5% 1%)
#> Joint Statistic:          1.89 2.11 2.59
#> Individual Statistic:     0.35 0.47 0.75
#>
#> Sign Bias Test
#> ------------------------------------
#>                    t-value     prob sig
#> Sign Bias           2.3282 0.019982  **
#> Negative Sign Bias  0.4695 0.638737
```

```
#> Positive Sign Bias  1.2485 0.211975
#> Joint Effect       15.9491 0.001162 ***
#>
#>
#> Adjusted Pearson Goodness-of-Fit Test:
#> ------------------------------------
#>   group statistic p-value(g-1)
#> 1    20     64.44    7.537e-07
#> 2    30     85.41    1.820e-07
#> 3    40     98.48    4.757e-07
#> 4    50    121.14    4.823e-08
#>
#>
#> Elapsed time : 0.6907778
```

```r
m.igarch.AIC <- -6.4602
```

The residuals for the IGARCH look alright, but the AIC is the largest we have encountered thus far. Hence, I will not bother considering the other properties.



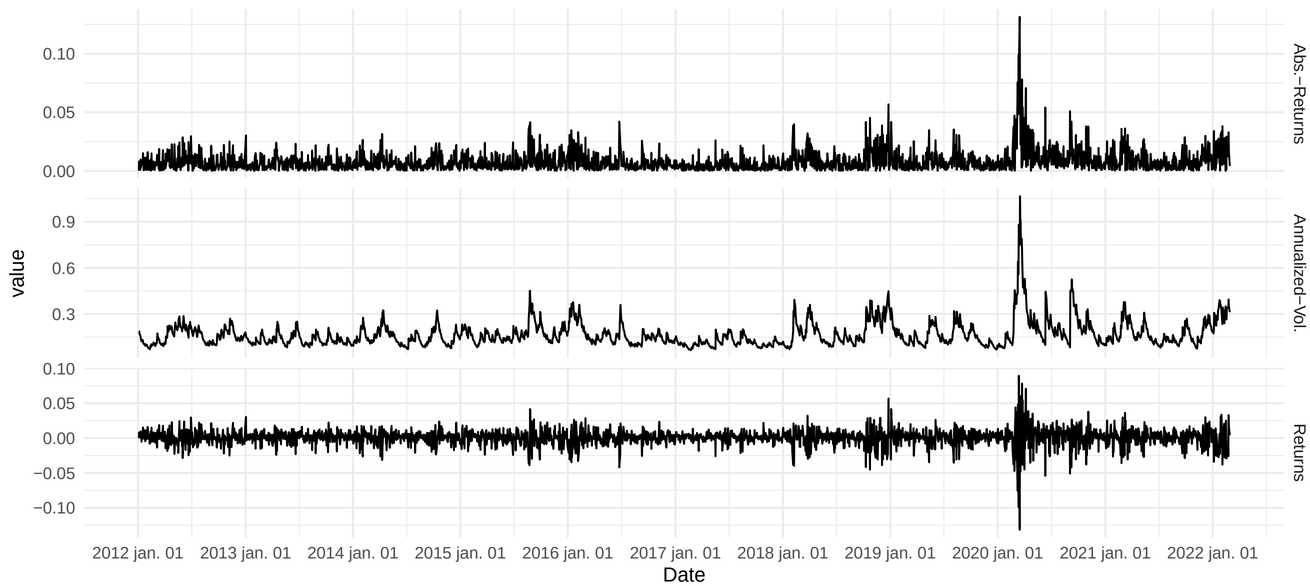
Thus, the conclusion is that the best model out of the four fitted is the EGARCH based model, since it satisfies the necessary conditions and has the lowest AIC, as seen in the table below.

Table 2: AIC for the Different Estimated GARCH Models¨

sGARCH	GJR-GARCH	EGARCH	IGARCH
-6.4605	-6.4923	-6.4958	-6.4602

3.6 Grafic and Interpretation of the Estimated Series of Volatility

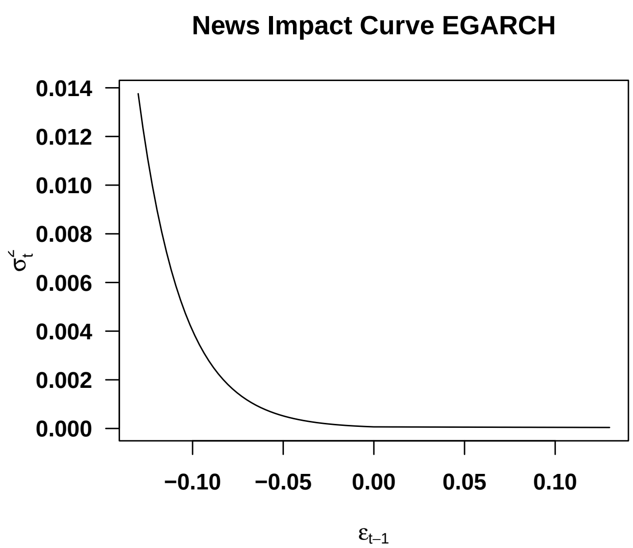
The estimated (annualized) series of volatility for the ARMA(2,2)-EGARCH model is plotted, alongside the returns and the absolute values of the returns.

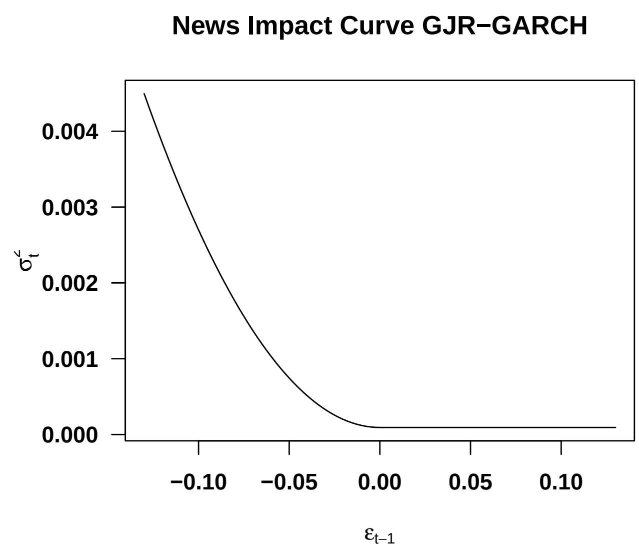


The general behaviour seems to match relatively well, i.e. the movements in the plots coincide relatively well. Days with larger (absolute) returns coincide with days with larger estimated volatilities. Note that we cannot compare the absolute values in plot however.

3.7 Grafic and Interpretation of the News Impact Curve

The news impact curve for our chosen model is shown below. Moreover, the news impact curve for the GJR-GARCH based model is also shown, since we will use this model later in the analysis again, for reasons that will become apparent.



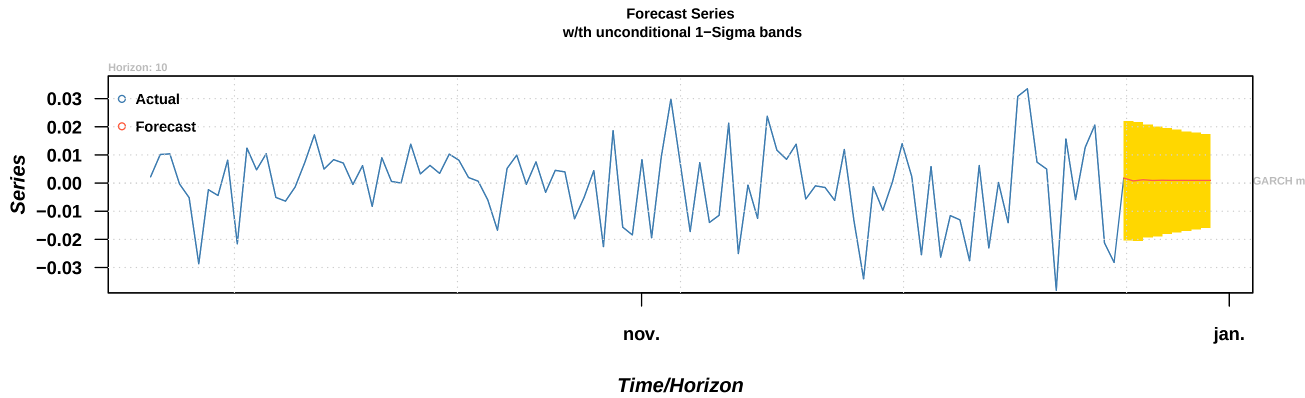


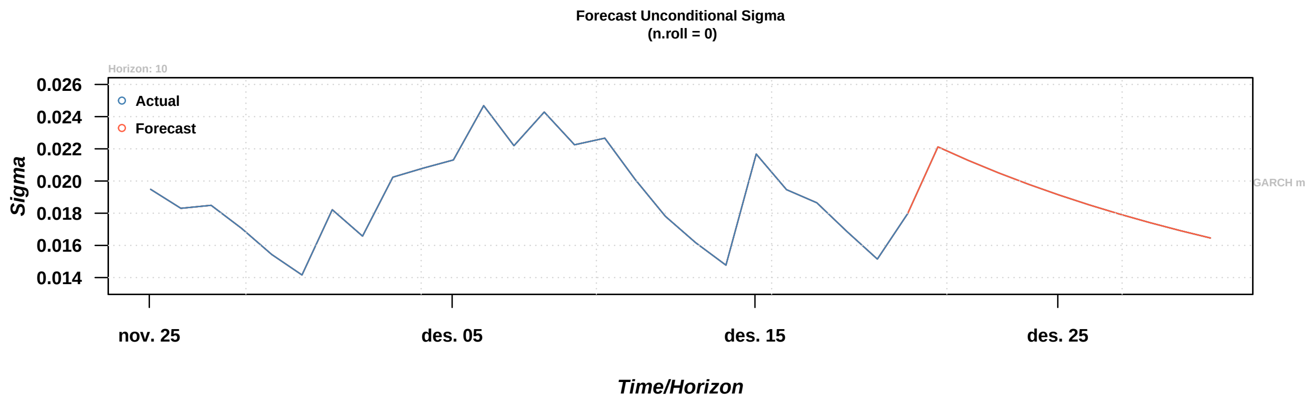
Both models consider the leverage effect, which is why the news impact curve is non-symmetric. The curves indicate that the volatility is impacted to a higher degree by negative news compared to the impact on the volatility following positive news, which decreases as the positivity of the news increases. The reader is referred to the discussion of the parameters in each of the two models for a more detailed discussion concerning this.

3.8 Volatility Predictions and Interpretations

Volatility is predicted while leaving out the last 10 observations when estimating the ARMA(2,2)-EGARCH model. The prediction is done 10 steps ahead into the future, first statically.

```r
m.egarch.pred <- ugarchfit(spec = spec.egarch, data = rendixic, out.sample = 10)
forc <- ugarchforecast(m.egarch.pred, n.ahead=10, n.roll= 0)
```



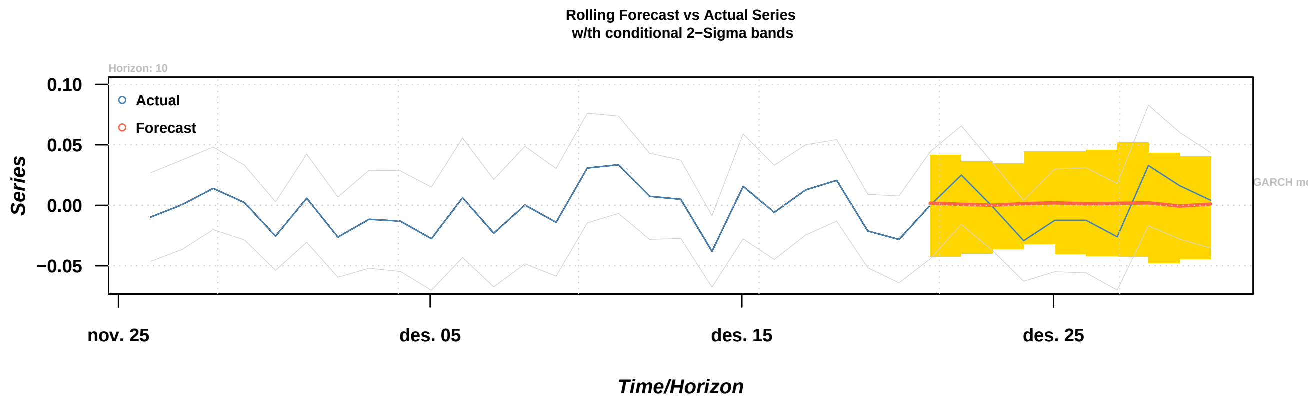


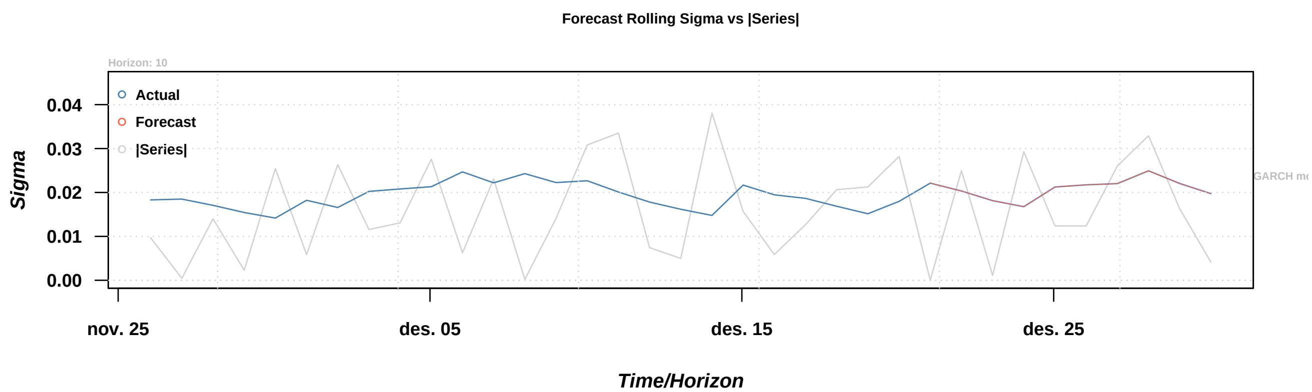

```
#> Long Run Unconditional Variance:  0.008652676
```

As we can see from the uppermost plot, these static predictions (for the mean) 10 steps ahead are relatively useless.

Next, let us predict 10 steps into the future with a rolling window. We reestimate the model at each time step and estimate one step into the future after each reestimation. After doing this 10 times, we have effectively predicted 10 days into the future.

```r
forc2 <- ugarchforecast(m.egarch.pred, n.ahead=1, n.roll= 10)
```



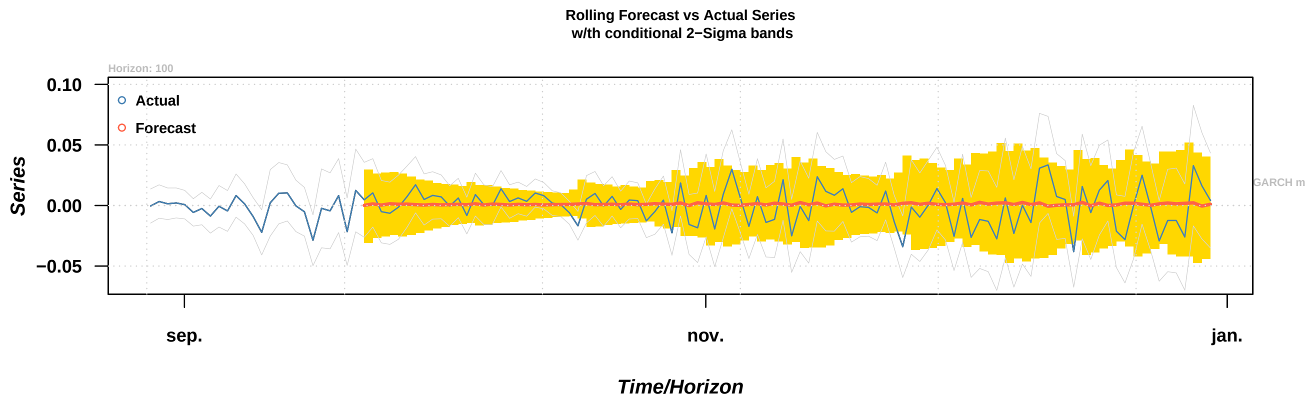


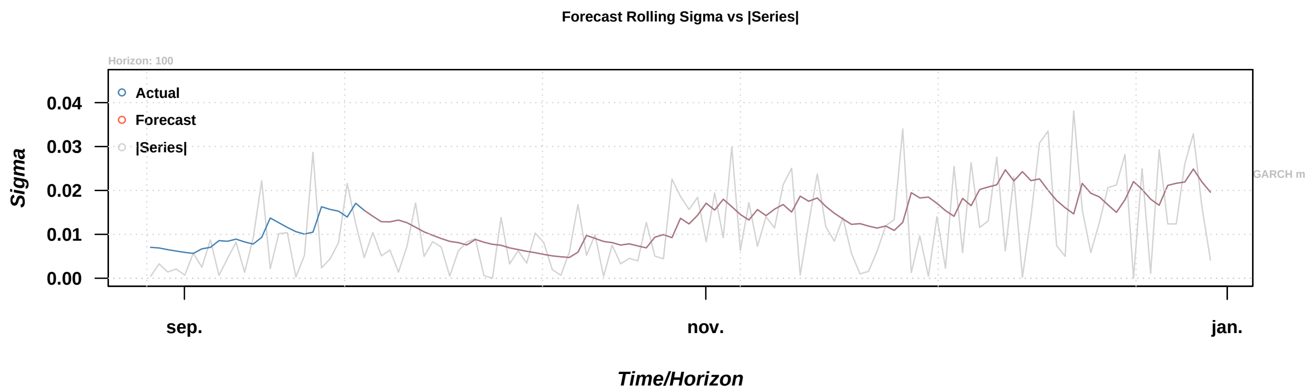

```
#> Long Run Unconditional Variance:  0.008652676
```

The predictions are still lousy, as can be seen from the predictions of the mean in the uppermost plot. However, from the second plot, it looks like the predictions of the variance are somewhat following similar movements as the absolute value of the series; when the absolute value of the series hits a spike, the predictions of the volatility increase as well.

The same type of movement can be seen when predicting with a rolling window 100 steps into the future, as done next.

```
m.egarch.pred2 <- ugarchfit(spec = spec.egarch, data = rendixic, out.sample = 100)
forc3 <- ugarchforecast(m.egarch.pred2, n.ahead=1, n.roll= 100)
```





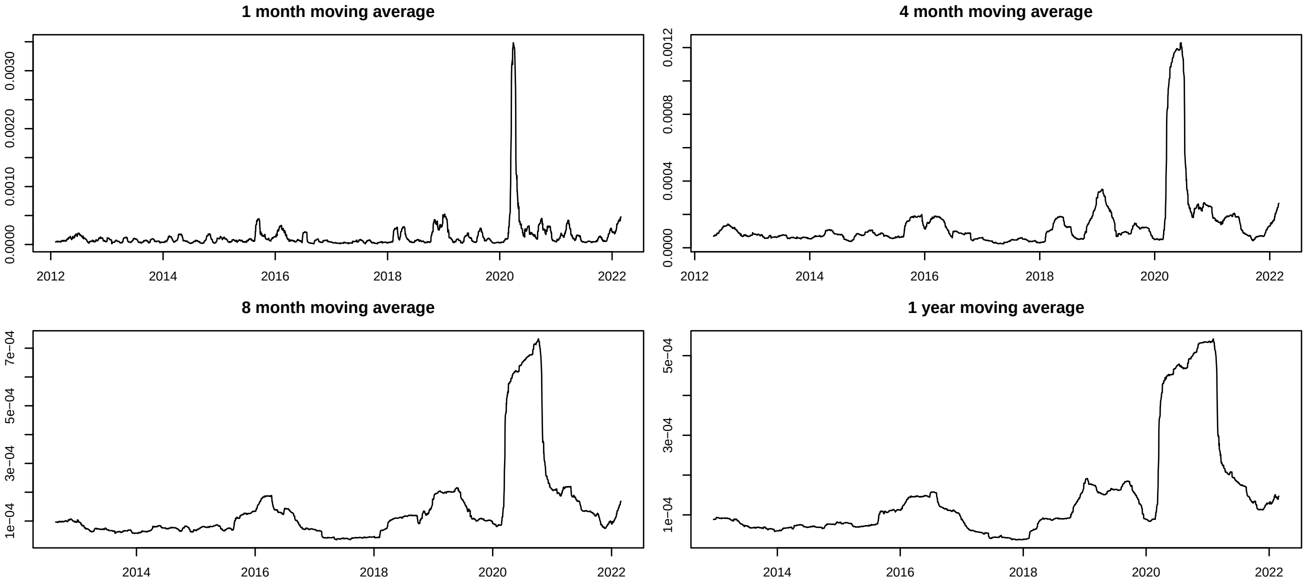

```
#> Long Run Unconditional Variance:  0.008571196
```

3.9 Calculations via Historical Volatility and EWMA

Historical volatility is calculated below. The historical volatility has been calculated using Simple Moving Average (SMA) over different time periods

$$\sigma_t^2 = \frac{1}{k}\sum_{i=1}^{k} r_{t-1}^2.$$

The results for different time periods k are shown below.



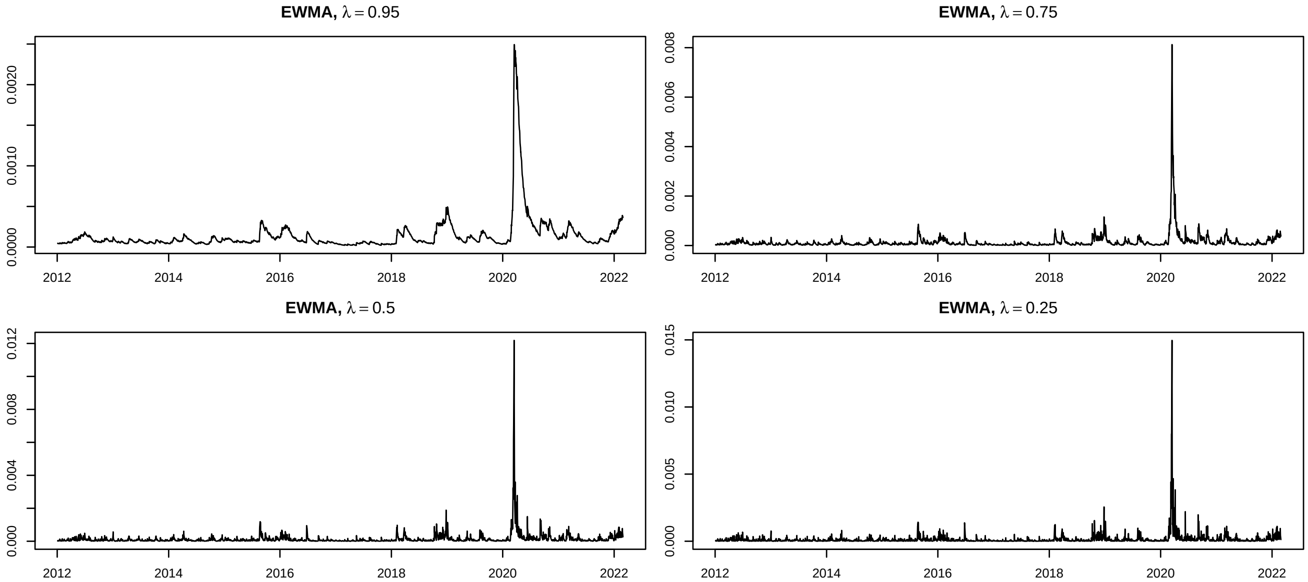
As is apparent from the plots, the volatility pattern is highly dependent on k, i.e. the number of observations used to calculate the moving average. Moreover, we can see that the results are greatly affected by extreme values, especially when k is small, which is clearly seen in the results for the 1 month moving average. The volatility pattern is smoother when k is larger. Which of these values for k gives the "best" results? This is difficult to answer.

Following the calculations from historical volatility, the Exponentially Weighted Moving Average (EWMA) model is used to calculate volatility

$$\sigma_t^2 = (1-\lambda)r_{t-1}^2 + \lambda\sigma_{t-1}^2 = (1-\lambda)\sum_{i=1}^{\infty}\lambda^{i-1}r_{t-i}^2, \;\; 0 < \lambda < 1.$$

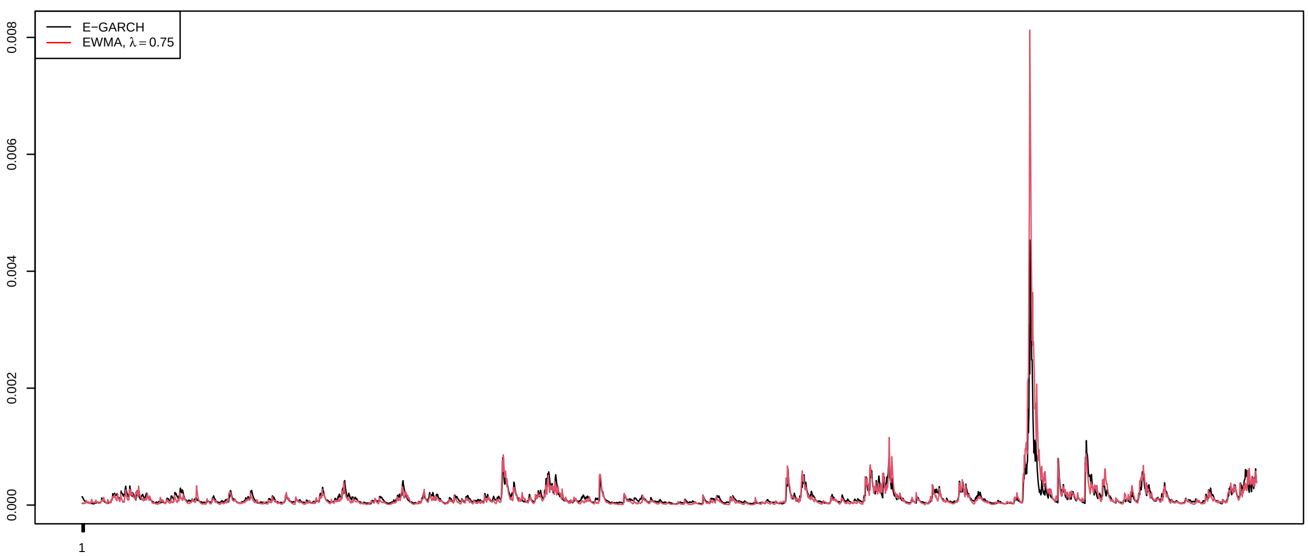
Different values of the parameter λ are used in order to see how the results depend on it. From the theoretical point of view, we know that the term $(1-\lambda)r_{t-1}^2$ determines the reaction of volatility to market events, i.e. the larger the term $(1-\lambda)$ the larger the reaction in the volatility stemming from yesterday's return. Moreover, the term $\lambda\sigma_{t-1}^2$ determines the persistence in volatility. In other terms, it decides how much of yesterday's volatility is allowed to persist to today's volatility: A larger value of λ gives larger persistence. Thus, the EWMA model gives a trade-off between persistence and reaction in the volatility, depending on the value of λ.

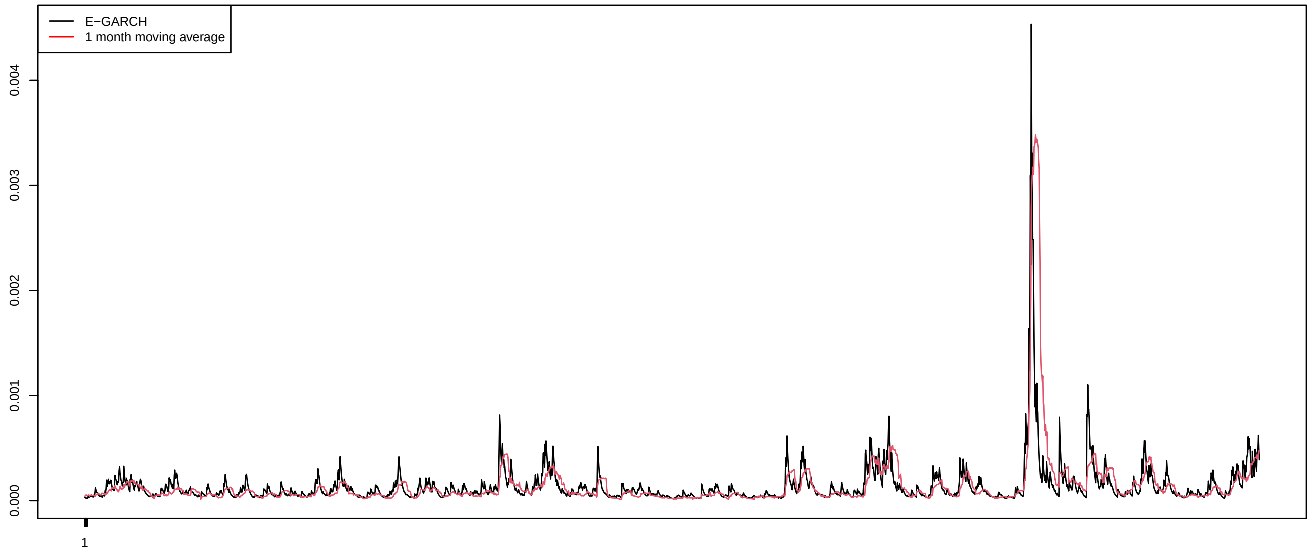
Some results from calculating the volatility using EWMA with different values of λ are plotted.



As we can see from the plots, the larger values of λ give smoother plots, since the persistence is larger, while the smaller values of λ give a more reactive or non-smooth volatility pattern, since the persistence of the volatility is much lower in these cases. Comparing to the results obtained when using the historical volatility, all the volatility patterns obtained with EWMA are more non-smooth than the former, being most similar to the 1 month moving average. Note also that the choice of λ seems somewhat arbitrary in this case (similar to the choice of k for historical volatility), as it is difficult to be certain about the best choice of the parameter.

Doing a quick comparison between these two models and the results from the EGARCH model, it looks like the EWMA model with $\lambda = 0.75$ gives a relatively similar volatility pattern, whereas the 1 month moving average (which is the one among the four models that is most similar to the results from EGARCH) is lacking in comparison.





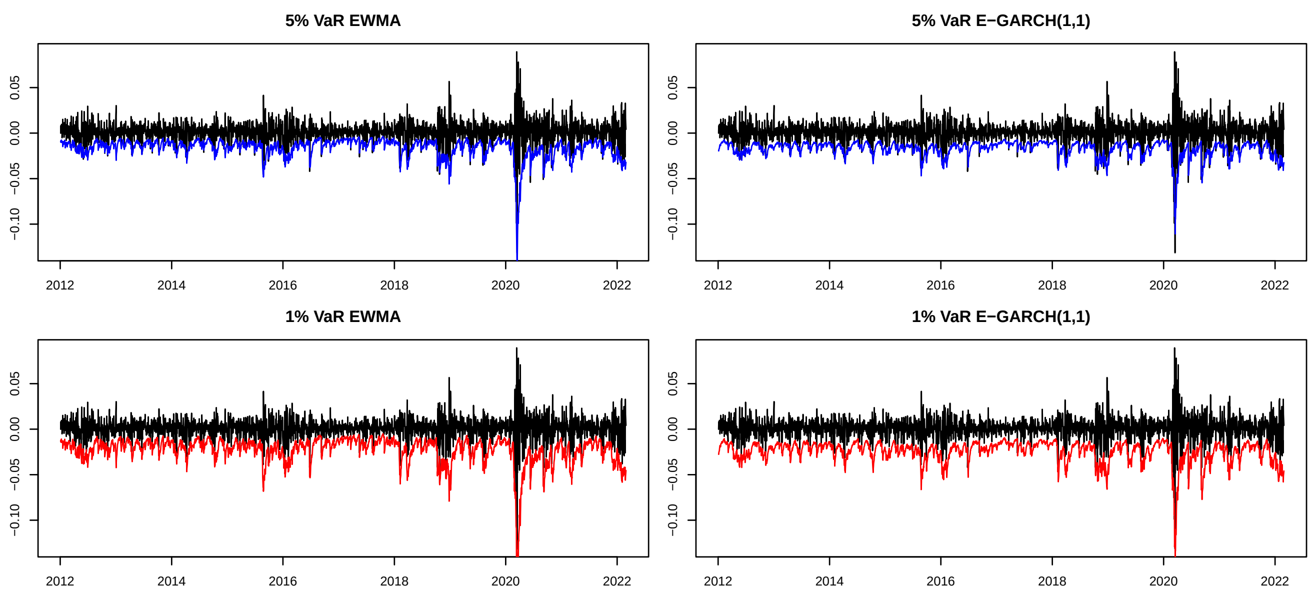
NOTE THAT THE TIMES ON THE X-AXIS ARE FUCKED UP. TRY TO FIX THIS LATER! THIS IS THE CASE SEVERAL PLACES!

3.10 Calculation and Interpretation of VaR

Here we will calculate and interpret the Value at Risk (VaR) using estimated volatilities from several different models.

First we calculate the VaR using estimates of volatility from the EWMA model with $\lambda = 0.75$ and from the ARMA(2,2)-EGARCH model. Thus, this is an in-sample comparison of the two models when calculating VaR using volatility estimations.

```
var5.ewma   <-  - qnorm(0.95) * sqrt(vol.ewma0.75)
var5.egarch <- - qnorm(0.95) * v
var1.ewma   <-  - qnorm(0.99) * sqrt(vol.ewma0.75)
var1.egarch <-  - qnorm(0.99) * v
```



The plots above show the estimated VaRs with the two different models, at two different significance levels (5% shown in blue and 1% shown in red), plotted together with the returns. To the naked eye it looks like the returns don't sink below the 1% VaR very often, while they sink below the 5% VaR somewhat more often, but still rarely. To quantify this, we calculate the fraction of the sample where the loss in returns exceeds each of the significance levels for the two models.

```
#> Fraction of sample where loss exceeds 5% VaR for EWMA:  0.03209393
#> Fraction of sample where loss exceeds 5% VaR for EGARCH:  0.05401174
#> Fraction of sample where loss exceeds 1% VaR for EWMA:  0
#> Fraction of sample where loss exceeds 1% VaR for EGARCH:  0.02191781
```

For both significance levels, we can see that the EWMA model overestimates the risk, since the fraction is much lower than necessary to meet the required significance level; in a case where we choose a significance level of $\alpha = 0.01$ we can see that the EWMA model with $\lambda = 0.75$ overestimates the risk, since the fraction of the sample where the loss exceeds the 1% VaR is 0. It also overestimates the risk when choosing a significance level of $\alpha = 0.05$, since the fraction of the sample where the loss exceeds the 5% VaR is around 3%. In practice this means that, based on volatility calculations using this model, the company is dedicating too much resources to the regulatory capital (minimum capital requirement).

On the other hand, the EGARCH-based model underestimates the risk, because the predefined significance level is violated in both cases. Since the EGARCH-based model does not give satisfactory in-sample predictions of VaR, we will in the following redo the calculations with the GJR-GARCH-based model, in order to check if that model performs any better. However, before redoing the calculations with ARMA(2,2)-GJR-GARCH, we will redo the EWMA-based calculations with a different value of λ and do out-of-sample predictions of VaR, in order to further quantify the performance of the models.

Redoing the calculations with the EWMA model with $\lambda = 0.95$ instead gives the fractions

```
#> Fraction of sample where loss exceeds 5% VaR for EWMA:  0.05401174
#> Fraction of sample where loss exceeds 1% VaR for EWMA:  0.02035225
```

These results are very similar to the results obtained when using the EGARCH, which now instead underestimate the risk. Thus, one of the main downfalls of the EWMA model again comes to show; the result is highly dependent on the value of the hyperparameter λ, which is not a trivial choice.

Next we use the variance-covariance method, calculating the VaR with a static forecast one time ahead, using the EGARCH-based model. Thus, this is an out-of-sample estimation of the VaR.

```
forc <- ugarchforecast(m.egarch, n.ahead=1, n.roll= 0)
show(forc)
```

```
#>
#> *------------------------------------*
#> *       GARCH Model Forecast         *
#> *------------------------------------*
#> Model: eGARCH
#> Horizon: 1
#> Roll Steps: 0
#> Out of Sample: 0
#>
#> 0-roll forecast [T0=1976-12-30 01:00:00]:
#>        Series   Sigma
#> T+1 0.0006739 0.01791
```

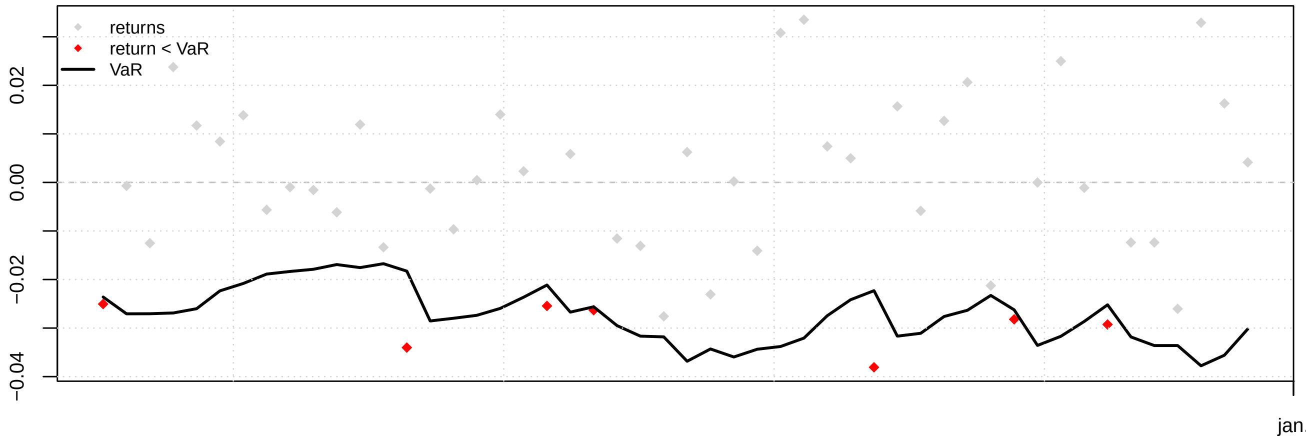
```
var5.garch <- - qnorm(0.95) * 0.01791
cat("VaR: ", var5.garch)
```

```
#> VaR:  -0.02945933
```

This value means that, with a confidence level of 95%, the largest expected loss for tomorrow in our index is $\approx 2.95\%$. In other terms, the probability of the return tomorrow being lower than -2.95% is 5%.

Next we calculate the VaR with a rolling window dynamic forecast (still out-of-sample), using the EGARCH-based model, with a significance level of 5%.

```
var.t <-  ugarchroll(spec.egarch, data = rendixic, n.ahead = 1, forecast.length = 50,
             refit.every = 10,  refit.window = "rolling",
             calculate.VaR = TRUE, VaR.alpha = 0.05)
```




```
#> VaR Backtest Report
#> ===========================================
#> Model:              eGARCH-std
#> Backtest Length: 50
#> Data:
#>
#> ==========================================
#> alpha:              5%
#> Expected Exceed: 2.5
#> Actual VaR Exceed:  7
#> Actual %:           14%
#>
#> Unconditional Coverage (Kupiec)
#> Null-Hypothesis: Correct Exceedances
#> LR.uc Statistic: 5.855
#> LR.uc Critical:     3.841
#> LR.uc p-value:      0.016
#> Reject Null:    YES
#>
#> Conditional Coverage (Christoffersen)
#> Null-Hypothesis: Correct Exceedances and
#>                 Independence of Failures
#> LR.cc Statistic: 7.839
#> LR.cc Critical:     5.991
#> LR.cc p-value:      0.02
#> Reject Null:    YES
```

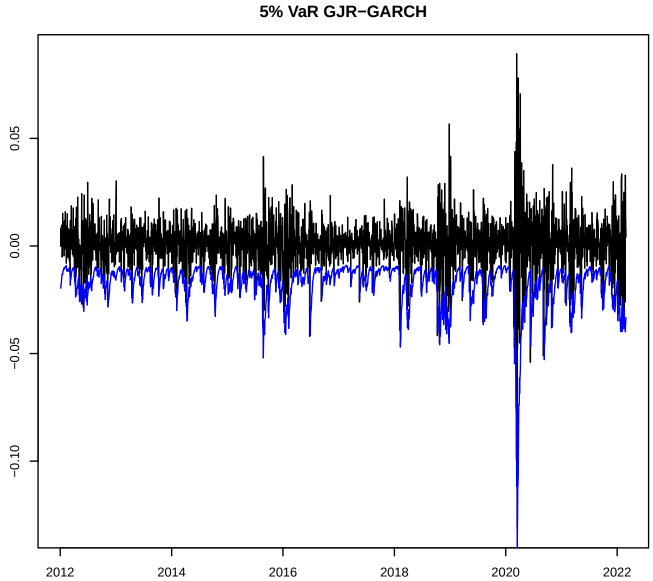
The report above shows that our predefined level of 5% significance is violated, i.e. that the largest expected loss cannot be quantified at the 5% significance level. Instead, the VaR is estimated to be 14%, which means that the probability of the return the next day being lower than the VaR is ≈ 14% instead of 5%. In practice, this means that the company should set aside more funds than expected, in order to cover the predefined significance level of 5%. As already noted during the in-sample estimations, it is apparent that the EGARCH-based model severely underestimates the risk, which is an indication that the model is not very well suited for use in practice, at least not for calculating VaRs.

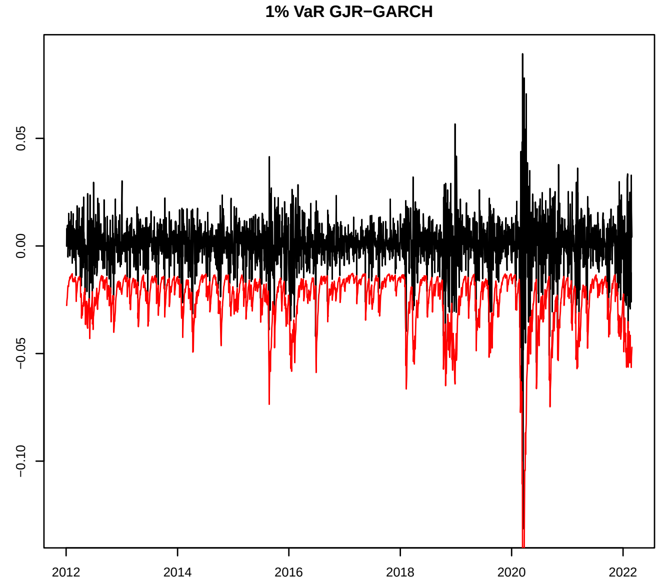
The in-sample and out-of-sample estimations of VaR are redone using an ARMA(2,2)-GJR-GARCH.

```
v.gjr <- sigma(mgjr)
var5.gjr.garch <- - qnorm(0.95) * v.gjr
var1.gjr.garch <-  - qnorm(0.99) * v.gjr
```

Without printing any values to the reader, notice that the conditional sigma values of the EGARCH-based model (`v`) and the GJR-GARCH-based model (`v.gjr`) look very similar. This already gives an indication of the fact that the in-sample estimations of VaR most likely will be similar when using the two models.

The plots for each of the significance levels are given, with the fractions of exceedances below.






```
#> Fraction of sample where loss exceeds 5% VaR for GJR-GARCH:  0.05088063
```

```
#> Fraction of sample where loss exceeds 1% VaR for GJR-GARCH:  0.02113503
```

The fractions are very similar to the fractions obtained with the EGARCH-based model, which indicates that the in-sample performance of the two models are very similar. Both models underestimate the risk.

Next, we redo the out-of-sample estimations.

The variance-covariance method is used, calculating the VaR with a static forecast one time ahead, this time using the GJR-GARCH-based model.

```
forc <- ugarchforecast(mgjr, n.ahead=1, n.roll= 0)
show(forc)
```

```
#>
#> *------------------------------------*
#> *       GARCH Model Forecast         *
#> *------------------------------------*
#> Model: gjrGARCH
#> Horizon: 1
#> Roll Steps: 0
#> Out of Sample: 0
#>
#> 0-roll forecast [T0=1976-12-30 01:00:00]:
#>        Series   Sigma
#> T+1 0.0009176 0.01849
```

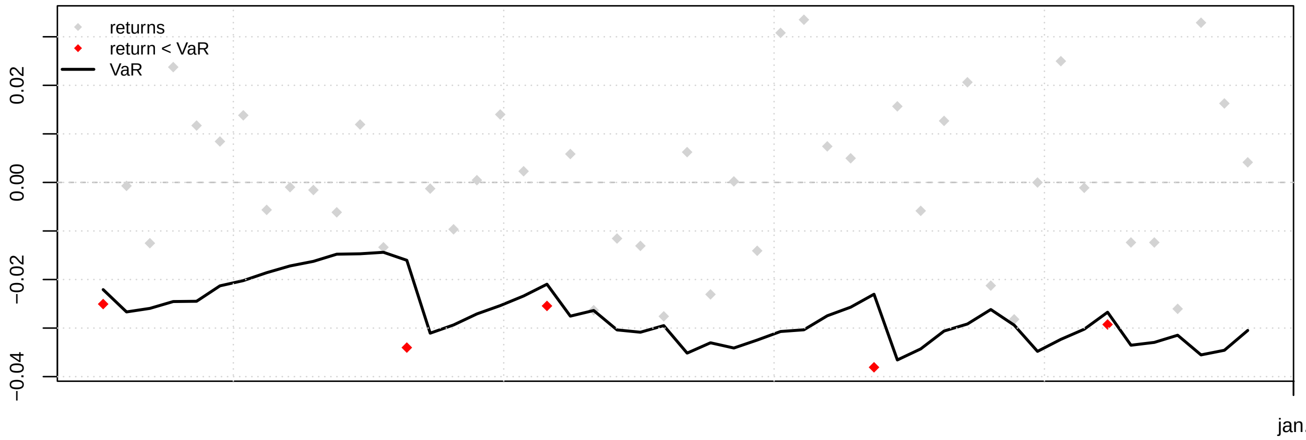
```
var5.gjr.garch <- - qnorm(0.95) * 0.01849
cat("VaR: ", var5.gjr.garch)
```

```
#> VaR:  -0.03041334
```

Compared to the value of 2.95% obtained with the EGARCH-based model, these results are very similar and the conclusions are the same.

Next we calculate the VaR with a rolling window dynamic forecast (still out-of-sample), using the GJR-GARCH-based model, with a significance level of 5%.

```
var.t.gjr <-  ugarchroll(spec.mgjr, data = rendixic, n.ahead = 1, forecast.length = 50,
              refit.every = 10,  refit.window = "rolling",
              calculate.VaR = TRUE, VaR.alpha = 0.05)
```




```
#> VaR Backtest Report
#> ===========================================
#> Model:              gjrGARCH-std
#> Backtest Length: 50
#> Data:
#>
#> ==========================================
#> alpha:              5%
#> Expected Exceed: 2.5
#> Actual VaR Exceed:  5
#> Actual %:           10%
#>
#> Unconditional Coverage (Kupiec)
#> Null-Hypothesis: Correct Exceedances
#> LR.uc Statistic: 2.065
#> LR.uc Critical:     3.841
#> LR.uc p-value:      0.151
#> Reject Null:     NO
#>
#> Conditional Coverage (Christoffersen)
#> Null-Hypothesis: Correct Exceedances and
#>                  Independence of Failures
#> LR.cc Statistic: 2.966
#> LR.cc Critical:     5.991
#> LR.cc p-value:      0.227
#> Reject Null:     NO
```

The plot and the report is still relatively similar to the ones obtained when using the EGARCH-based model. However, we can see that the amount of exceedances is reduced to 5 instead of 7. This can be viewed as an improvement over the performance of the EGARCH-based model, even though the predefined significance level of 5% is still violated. The conclusion is still the same as earlier: the company should set aside more funds than expected, in order to cover the predefined significance level.

Before continuing, a quick note on the advantages of GARCH-based models over EWMA models is given. This section gives an explicit example of how a GARCH model is advantageous compared to an EWMA model, since we use the data to estimate the model ("the data talks") optimally with maximum likelihood and we need not to set a hyperparameter like λ which the results depend largely on. Another advantage to note concerning the use of GARCH models over EWMA is that the reaction (typically α_i) and persistence (typically β_i) coefficients are estimated separately, which means that there is no defined trade-off between the two, as is the case in the EWMA.

4 Multivariate Analysis

4.1 Multivariate DCC GARCH

In order to solve this problem I have chosen the stock of Stratus Properties Inc. (STRS), which is one of the top 30 components of the NASDAQ Composite Index. Similarly to the IXIC-data, this data has been downloaded in a csv-file directly from the Yahoo Finance website. The adjusted close of the time series is plotted below.

```
STRS <- read.csv("STRS.csv")
dim(STRS)
```

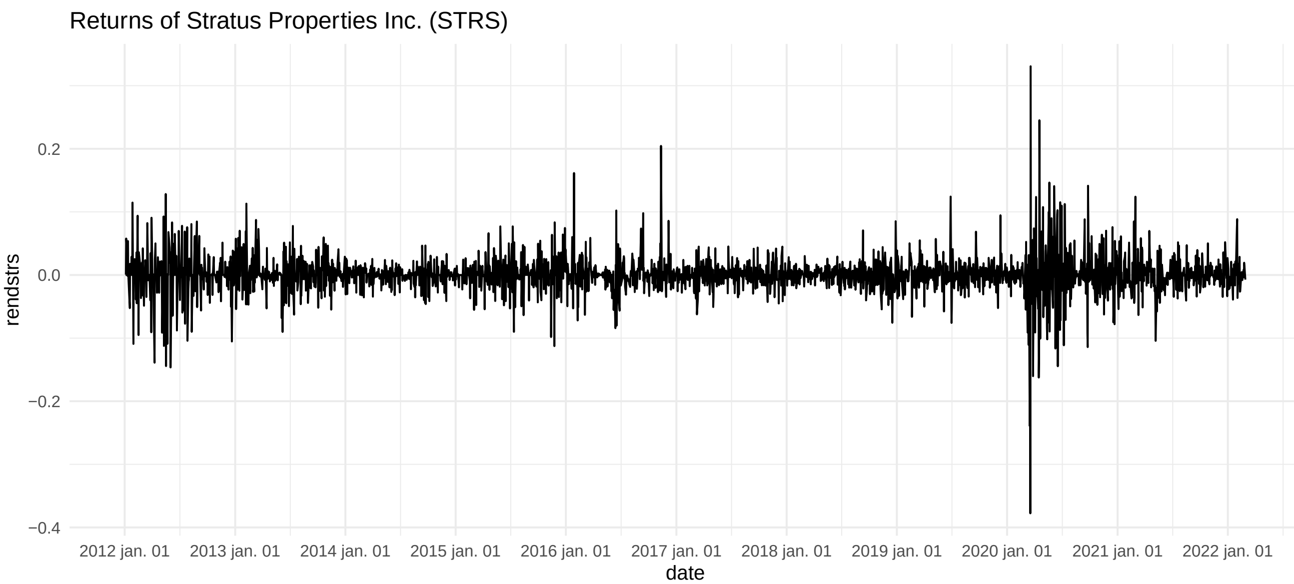
```
#> [1] 2556    7
```

```
any(is.na(STRS))
```

```
#> [1] FALSE
```

```
strs <- STRS[,6]
```

The returns of STRS are calculated and plotted below.



Analysis of stationarity shows that this series is integrated of order 1 as well, similarly to IXIC. Thus, we work with the returns of the series instead of the series itself.

REDO THIS TO CHECK IF GJR-GARCH IS BETTER IN THIS CASE AS WELL!

After doing a similar analysis of this series, the conclusion is that a MA(1)-EGARCH is the best model to estimate its volatility. This model is used, together with the ARMA(2,2)-EGARCH from IXIC, to estimate the multivariate DCC GARCH model for these two series (both of which are estimated on the logarithmic returns, not on the series themselves).

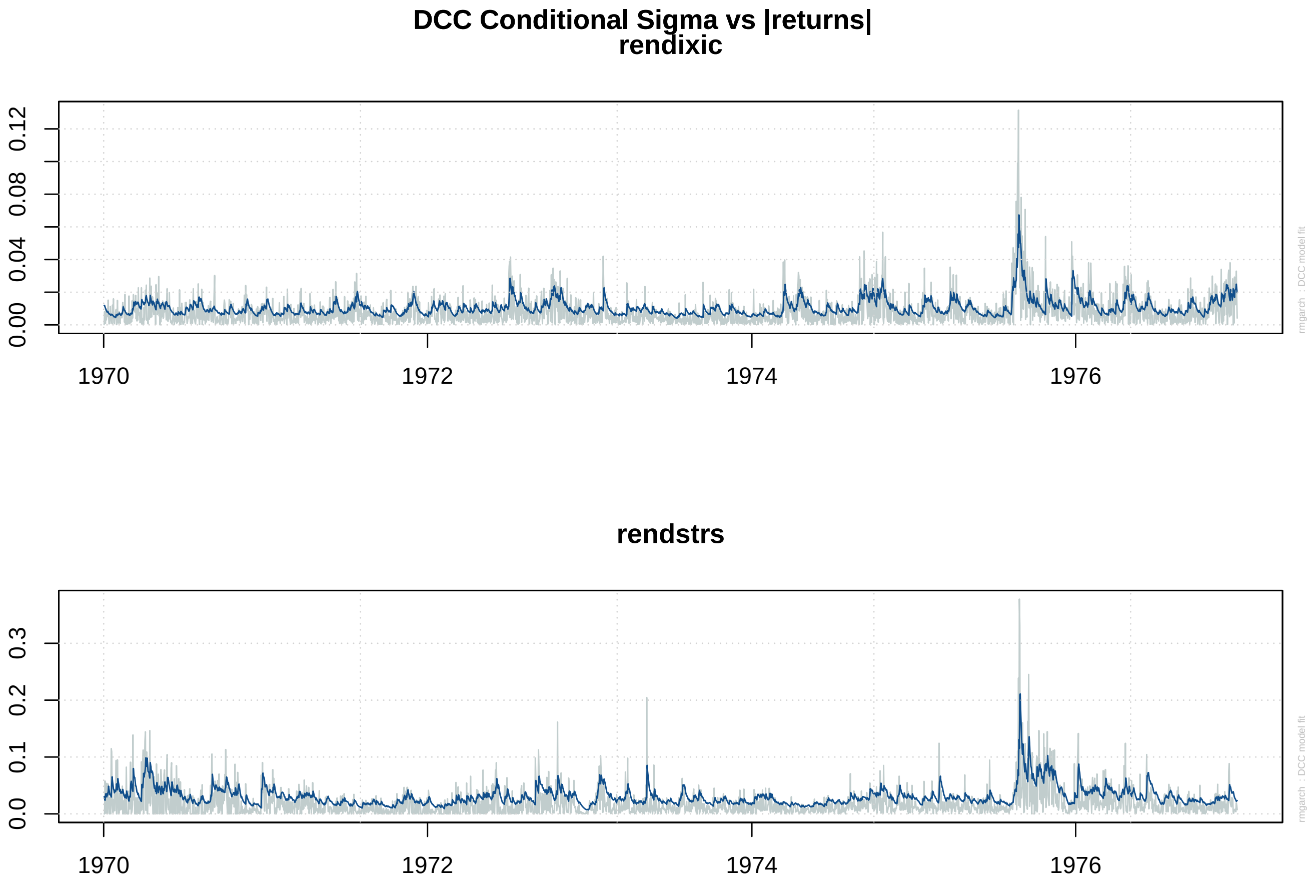
```r
returns <- cbind(rendixic,rendstrs)
spec1 <- ugarchspec(mean.model = list(armaOrder = c(2,2)), variance.model = list(garchOrder = c(1,1),
                                              model = "eGARCH"), distribution.model = "std")
spec2 <- ugarchspec(mean.model = list(armaOrder = c(0,1)), variance.model = list(garchOrder = c(1,1),
                                              model = "eGARCH"), distribution.model = "std")
dcc.garch11.spec <- dccspec(uspec = multispec(c(spec1, spec2)), dccOrder = c(1,1), distribution = "mvnorm")
(dcc.fit <- dccfit(dcc.garch11.spec, data = returns))
```

```
#>
#> *---------------------------------*
#> *          DCC GARCH Fit          *
#> *---------------------------------*
#>
#> Distribution         :  mvnorm
#> Model                :  DCC(1,1)
#> No. Parameters       :  20
#> [VAR GARCH DCC UncQ] : [0+17+2+1]
#> No. Series           :  2
#> No. Obs.             :  2555
#> Log-Likelihood       :  14075.25
#> Av.Log-Likelihood    :  5.51
#>
#> Optimal Parameters
#> -----------------------------------
#>                     Estimate  Std. Error   t value Pr(>|t|)
#> [rendixic].mu       0.000945    0.000168   5.63015 0.000000
#> [rendixic].ar1      0.137188    0.008107  16.92259 0.000000
#> [rendixic].ar2      0.310470    0.011440  27.13847 0.000000
#> [rendixic].ma1     -0.178777    0.017057 -10.48116 0.000000
#> [rendixic].ma2     -0.292323    0.011170 -26.17109 0.000000
#> [rendixic].omega   -0.389642    0.007845 -49.66907 0.000000
#> [rendixic].alpha1 -0.193182    0.023139  -8.34890 0.000000
#> [rendixic].beta1    0.958980    0.001084 884.69593 0.000000
#> [rendixic].gamma1  0.161978    0.022819   7.09833 0.000000
#> [rendixic].shape    5.708617    0.736620   7.74974 0.000000
#> [rendstrs].mu       0.000104    0.000411   0.25408 0.799435
#> [rendstrs].ma1     -0.189850    0.017865 -10.62700 0.000000
#> [rendstrs].omega   -0.271402    0.199469  -1.36062 0.173634
#> [rendstrs].alpha1 -0.064225    0.035459  -1.81126 0.070101
#> [rendstrs].beta1    0.961041    0.028377  33.86663 0.000000
#> [rendstrs].gamma1  0.398593    0.134602   2.96126 0.003064
#> [rendstrs].shape    2.846712    0.207328  13.73046 0.000000
#> [Joint]dcca1        0.016504    0.008918   1.85073 0.064208
#> [Joint]dccb1        0.924780    0.035361  26.15250 0.000000
#>
#> Information Criteria
#> ---------------------
#>
#> Akaike       -11.002
#> Bayes        -10.956
#> Shibata      -11.002
#> Hannan-Quinn -10.986
#>
#>
#> Elapsed time : 6.224618
```

The positivity and stationarity conditions of the DCC hold, because `dcca1` and `dccb1` are both positive and their sum is less than one. The first parameter can be used to interpret the response in the correlation to the news, which has a relatively low estimation in this case. The second parameter can be used to interpret the persistence of the correlation. This value is relatively close to one, which means that the persistence of the correlation is relatively large.

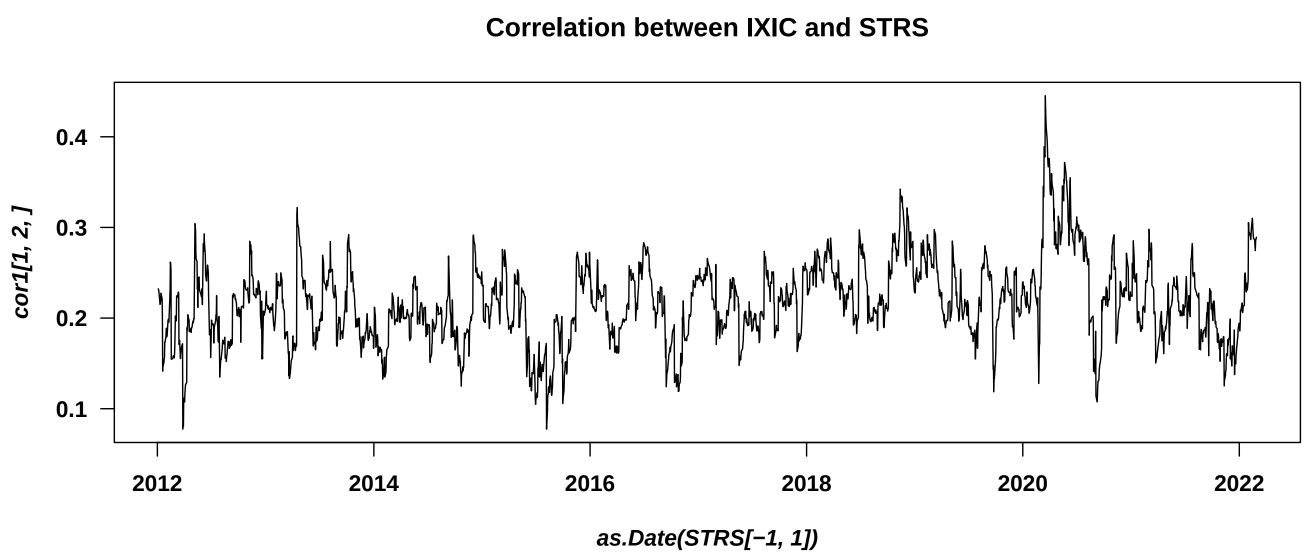
4.2 Estimated Correlation and News Impact Surface

The plot below shows the conditional standard error estimated from the model (in blue) and the realized absolute returns (in grey). Disregard the years shown on the x-axis. The time period of the data is the same as earlier.

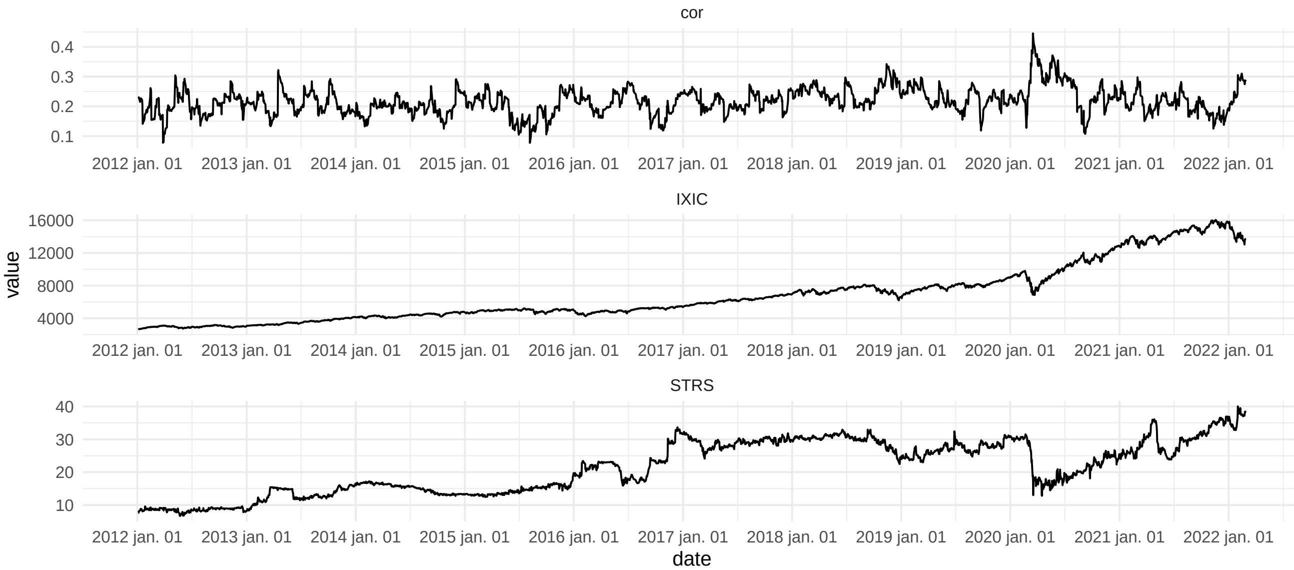


From this plot we can gather that the model has made good estimations of the standard error, because the behaviour of the graphs are similar. Note that we cannot conclude anything based on the absolute values of the quantities; we are only interested in the shape or the behaviour of the quantities.

The plot below shows the correlation between the two series estimated from the model.



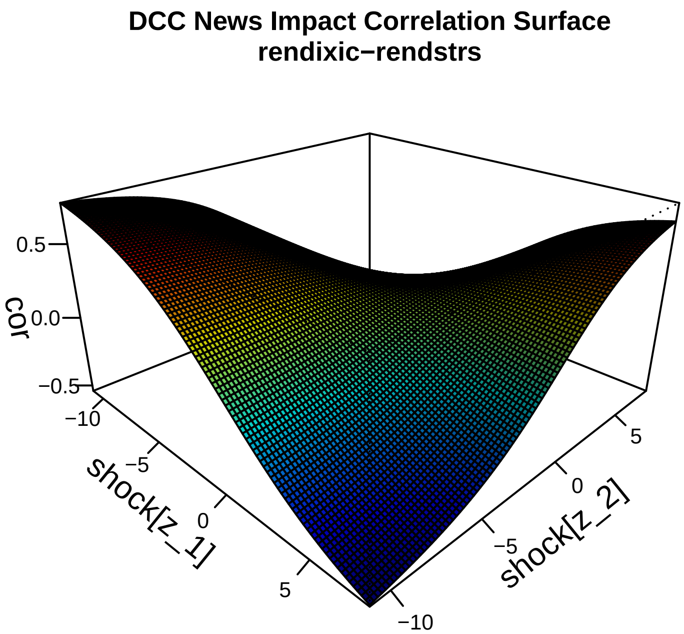
The correlation is plotted alongside the two original time series.



Just by looking at the two time series side by side, they look like they move in a similar fashion. However, STRS exhibits more volatile behaviour, i.e. it does not increase as stably as IXIC, likely because we are comparing one individual stock to an index. The peak in the correlation between the two series, which corresponds to a correlation of about 0.45, took place in the beginning of 2020, when both prices fell, most likely because of the outburst of COVID-19. One of the stylized facts about financial time series (daily returns) becomes apparent here; the correlation between assets increases during highly-volatile periods, particularly during crashes.

The news impact surface is plotted below.

```
nisurface(dcc.fit, type="cor")
```



From this we can learn that

- Simultaneous negative news in both series lead to the largest increase in correlation
- Simultaneous positive news in both series leads to an increase in correlation as well, but not to the same extent as the negative shocks. This shows that the leverage effect has been taken into account in the model.
- Increasingly negative news in one of the series and positive news in the other leads to an increasingly negative correlation, which (again) shows that the leverage effect is taken into account, since the negative news clearly are weighted more than the positive news.

5 Conclusions