Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

Introduction

In 2015, Amazon celebrated its 20th anniversary - a company that has come a long way. Back in 2005, Amazon reported $6.9 billion in annual revenues. At the end of fiscal 2015, that figure was $107.9 billion. Over these ten years, Amazon has evolved from an online retailer into a jack-of-all-trades in the consumer electronics, digital media, and cloud computing services spaces. Shares of Amazon have soared 760% over the past decade. Will it surge higher or plummet over the next ten years?

The objective of the project is to apply advanced GARCH modeling to the analysis of heteroscedastic time series, such as Amazon's daily stock returns. Stock price forecasting is a popular and important topic in financial and academic studies. Time series analysis is the most common and fundamental method used to perform this task. Technical analysis utilizes the information captured by the stock price to interpret what the market is saying with the purpose of forming a view on the future. This report aims to combine the conventional time series analysis technique with information from the Yahoo finance website to analyze and predict the daily changes in stock price for Amazon.com, Inc.

Data source:
Dataset: Amazon.com, Inc. (AMZN) Historical Price data
Dataset source: http://finance.yahoo.com/quote/AMZN/history?p=AMZN
Period: 01/03/2005 – 11/16/2016 (Present)

Non-Technical Summary

Analysis of the high frequency data such as Amazon's historical price data focuses on the study of past market action to predict future price movement. One of the basic assumption underlying this analysis is the notion that price changes are not random, both short term and long term, can be identified, enabling market traders to profit from investing according to the existing trend.

In the recent years, Amazon has become a titan of retail that has branched out into many other areas. The objective of this analysis is to model the daily stock prices as a function of time. Prior to the beginning of the exploratory analysis phase, the dataset was examined and it was concluded that it did not require any preprocessing or data cleaning. Furthermore, prior to performing any analysis on the data, it was necessary to ensure that the dataset was adequate for the chosen analysis. Amazon's daily stock price data was perfectly suitable for applying a time-series modeling process.

Initial plots and analysis of the data revealed that Amazon's stock price data is highly varying and non-constant over time. The plot of the stock price data over time revealed periods of high variance in 2006 and in 2008-2009 during the economic crisis. There were also some positive spikes noticeable in the most recent years (2015-16). Observing these trends in the data allowed to identify the correct analysis to be performed on the data. Many different models that were suitable for the stock prices were applied on the dataset and were examined. The models were examined based on the various statistical parameters and graphs and the best model was determined using a model selection criteria.

Further to test the effectiveness of the model, two of the models which were considered most appropriate were tested and validated using the modeling procedures. This process highlighted the importance of cross validating and testing the model procedures. Since these models have a lot of variations, after a certain number of forecasts the model becomes useless for forecasting. Backtesting procedure was used recursively to compare the two models and choose best model that minimizes error - Backtesting is the process of testing a trading strategy on relevant historical data to ensure its viability before the trader risks any actual capital. A trader can simulate the trading of a strategy over an appropriate period and analyze the results for the levels of profitability and risk. The model chosen best captured the trends of the Amazon's daily stock prices data with minimized error for prediction accuracy.

Technical Summary

Exploratory Analysis of the data:

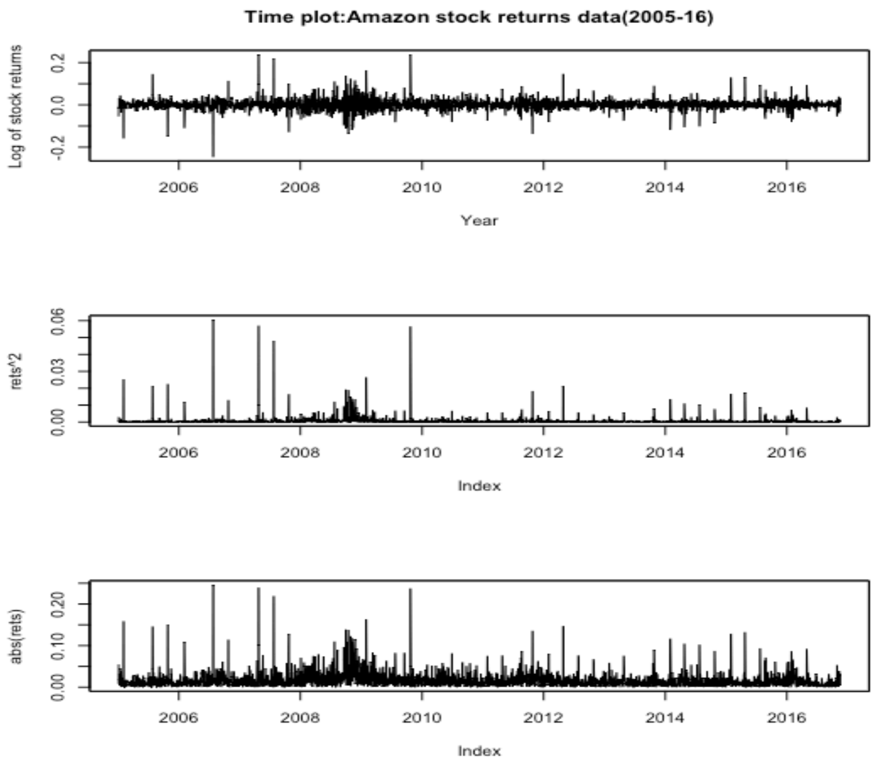
The data exploration began by analyzing the distribution and statistics for the Amazon stock returns data. Table-1a below shows the basic statistics for the data. The log returns data has a mean and median value of zero. The Kurtosis value of 14.468 represents excess kurtosis and means that the distribution has fatter tails and that there are lesser chances of extreme outcomes compared to a normal distribution. Furthermore, the skewness value is 0.48 indicates positively skewed distribution and would mean frequent small negative outcomes, and extremely bad scenarios are not as likely. The range of data varies from a minimum of -0.246182 to a maximum of 0.238621. The Histogram further confirms that the log returns data is not normally and not symmetrically distributed.

nobs	2990.000000
NAs	0.000000
Minimum	-0.246182
Maximum	0.238621
1. Quartile	-0.010703
3. Quartile	0.012830
Mean	0.000943
Median	0.000201
Sum	2.819444
SE Mean	0.000468
LCL Mean	0.000026
UCL Mean	0.001860
Variance	0.000654
Stdev	0.025566
Skewness	0.480386
Kurtosis	14.468528

Table 1a: Basic Statistics

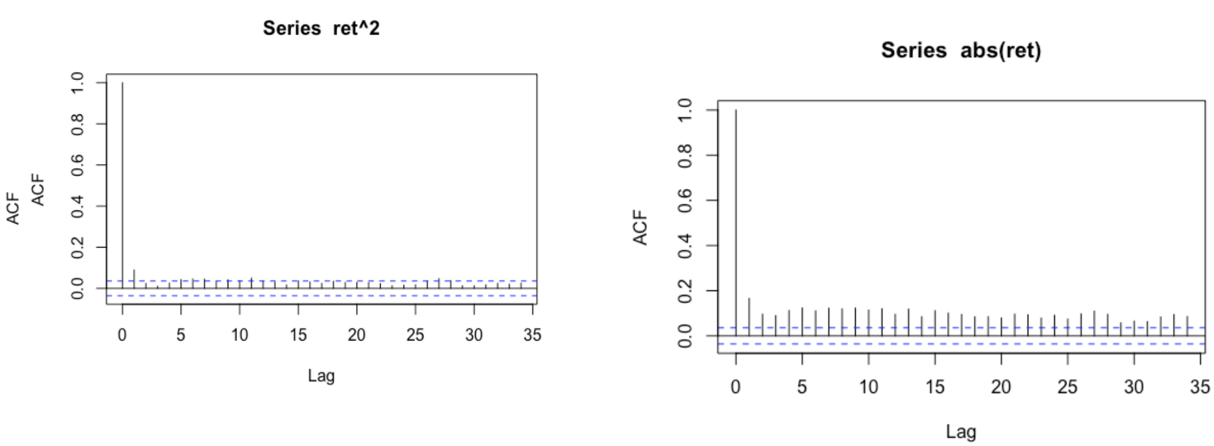
Time plots show returns varying around the zero line with an extreme negative return (-0.2) in 2006 followed by a high volatility period, after which there are no significant negative shocks. Conditional volatility is non-constant over time with periods of high volatility in 2006 and in 2008-2009 during the economic crisis. There are some positive spikes noticeable in the most recent years

(2015-16). Shocks influence the process volatility, where high volatility does not decrease quickly.



ACF plots in Figure 1 below depict that the Amazon stock log returns are not correlated – indicating a constant mean model for r_t. The square returns time series show autocorrelations for some of the lagged values, but the ARCH effect is evident in the absolute returns time series with large autocorrelations. Furthermore, the Lung-Box tests on squared and absolute returns confirm that the returns are autocorrelated (p-values < 0.01 -> we can reject H_0 of no autocorrelation). Hence, we can conclude that the log returns process has a strong non-linear dependence and there is a significant ARCH/GARCH effect for the log returns data.

Figure 1: ACF plots for log returns squared and absolute returns



Model Fitting:

Since the autocorrelation analysis showed that the returns are not correlated but have non -constant volatility, the process began by fitting an AR (0) - GARCH (1,1) model. The model was applied for different distributions such as normal, student's t and skewed t distribution (See Appendix for detailed results). Based on the tests for distribution goodness of fit, the most suitable for Amazon's daily stock returns data is the student's t distribution. Below are the results and fitted model for two of the best suitable models:

MODEL 1: AR (0) mean model with standard GARCH (1,1) model for variance and t- distribution:

Table 2: MODEL 1: AR (0) – GARCH (1,1) with t distributed errors

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000654    0.000323   2.0234 0.043034
omega   0.000005    0.000002   2.3719 0.017698
alpha1  0.023268    0.003292   7.0679 0.000000
beta1   0.967858    0.001744 554.9706 0.000000
shape   3.520292    0.219236  16.0571 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000654    0.000297   2.1983 0.027931
omega   0.000005    0.000006   0.8342 0.404166
alpha1  0.023268    0.006944   3.3508 0.000806
beta1   0.967858    0.001431 676.4558 0.000000
shape   3.520292    0.241549  14.5738 0.000000
```

Fitted Model:

r_t=0.000654+ a_t --> Mean model

$a_t = \sigma_t e_t$

σ_t^2=0.00 + 0.023268 a^2_{t-1} + 0.967858σ^2_{t-1} --> Variance model

with t distribution with 4 degrees of freedom (approximated to nearest integer).

- alpha1 represents volatility reaction to shocks
- beta1 represents persistence of shocks on volatility
- A large a_{t-1} or a large σ^2_{t-1} gives rise to a large σ_t^2
- Cannot capture asymmetric reaction of volatility to positive or negative innovations.

MODEL 2: AR (0) mean model with standard EGARCH (1,1) model for variance and t-distribution: -- Applying EGARCH model to fit possible leverage effect.

Table 3:MODEL 2: AR (0) – EGARCH (1,1) with t distributed errors

```
Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(1,1)
Mean Model      : ARFIMA(0,0,0)
Distribution    : std

Optimal Parameters
------------------------------------
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.000520    0.000314    1.6593 0.097054
omega  -0.077466    0.004346  -17.8250 0.000000
alpha1 -0.047202    0.010498   -4.4963 0.000007
beta1   0.989791    0.000563 1757.5086 0.000000
gamma1  0.099616    0.017738    5.6159 0.000000
shape   3.726901    0.254305   14.6552 0.000000

Robust Standard Errors:
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.000520    0.000288    1.8094 0.070395
omega  -0.077466    0.004531  -17.0953 0.000000
alpha1 -0.047202    0.013088   -3.6065 0.000310
beta1   0.989791    0.000619 1598.4903 0.000000
gamma1  0.099616    0.027427    3.6320 0.000281
shape   3.726901    0.320116   11.6423 0.000000
```

Fitted Model:

$r_t = 0.00052 + a_t$ --> Mean model

$a_t = \sigma_t e_t$

$\ln(\sigma_t^2) = -0.07746 + (-0.0472e_{t-1} + 0.0996(|e_{t-1}| - E(|e_{t-1}|)) + 0.9897 \ln(\sigma^2_{t-1})$ --> Variance model

with t distribution with 4 degrees of freedom (approximated to nearest integer).

- r_t = log return of an asset at time t
- Omega ω is constant term
- Leverage parameter is α_1 (multiple of θ in other expression)
- If $\alpha_1 < 0$, process reacts more strongly to negative shocks

The leverage theta parameter is negative and significantly different from zero, indicating that the volatility of Amazon stock returns has a significant asymmetric behavior and react more heavily to negative shocks. Shape parameter is significant as well, indicating that the t-distribution is a good choice.

Table 4 below shows the model selection criteria – The AIC and BIC values of all the different models were compared and the one with the lowest BIC value was considered the best model and most appropriate.

Table 4: Model selection criteria

	garch11	garch11.t	egarch11	gjrgarch11
Akaike	-4.573121	-4.900425	-4.926650	-4.907798
Bayes	-4.565090	-4.890387	-4.914604	-4.893744
Shibata	-4.573125	-4.900431	-4.926658	-4.907809
Hannan-Quinn	-4.570232	-4.896814	-4.922316	-4.902742

Best model according to the BIC model selection criteria is the **AR (0) – EGARCH (1,1)** model with **t-distribution**.

Residual Analysis and Model diagnostics:

The analysis and diagnostics of the residuals for the best model selected i.e. **AR (0) – EGARCH (1,1)** is presented below:

Table 5: Residual diagnostics

```
Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                              statistic   p-value
Lag[1]                          1.152     0.2830
Lag[2*(p+q)+(p+q)-1][2]         3.367     0.1116
Lag[4*(p+q)+(p+q)-1][5]         5.278     0.1323
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                              statistic   p-value
Lag[1]                          6.173     0.01297
Lag[2*(p+q)+(p+q)-1][5]         6.702     0.06120
Lag[4*(p+q)+(p+q)-1][9]         7.474     0.16264
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
                Statistic Shape Scale P-Value
ARCH Lag[3]     0.3488 0.500 2.000  0.5548
ARCH Lag[5]     0.4816 1.440 1.667  0.8890
ARCH Lag[7]     1.2865 2.315 1.543  0.8630
```

Table 5 shows the results for Ljung-Box test for serial correlation is computed on residuals, and Ljung-Box test for ARCH/GARCH effect computed on squared residuals.

There is no evidence of autocorrelation in residuals. They behave as a white noise process. There is no evidence of serial correlation in squared residuals as well (considering 0.01 significance level). Weighted ARCH LM tests for the absolute value of returns further prove that residuals behave as white noise.

Table 6: Pearson goodness of fit test

Adjusted Pearson Goodness-of-Fit Test:

	group	statistic	p-value(g-1)
1	20	33.01	0.02398
2	30	49.73	0.00965
3	40	48.66	0.13808
4	50	60.30	0.12919

Table 6 shows the Goodness of fit test for distribution of the error term. The null hypothesis states that the distribution of error terms in the model is adequate. In this case, t-distribution cannot be rejected. Amazon's daily stock returns data supports the choice of t-distribution.

Figure 1a: QQ-Plot of Standardized Residuals

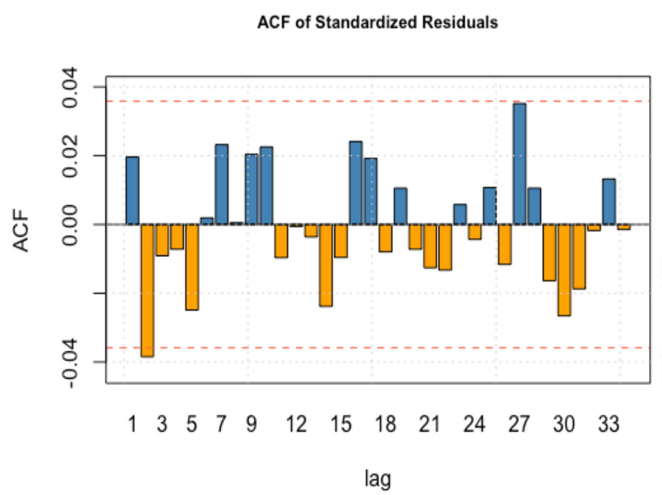


Figure 1b: ACF of Standardized Residuals

The t-distribution probability plot for residuals depicted in Figure 1a show that the t-distribution is appropriate. However, some departure is seen under the right tail for extreme residuals (also since the distribution is right skewed). ACF of Standardized residuals shown in Figure 1b further proves that there are no autocorrelation values (except for lag 2 which might be considered an outlier) and residuals are white noise.

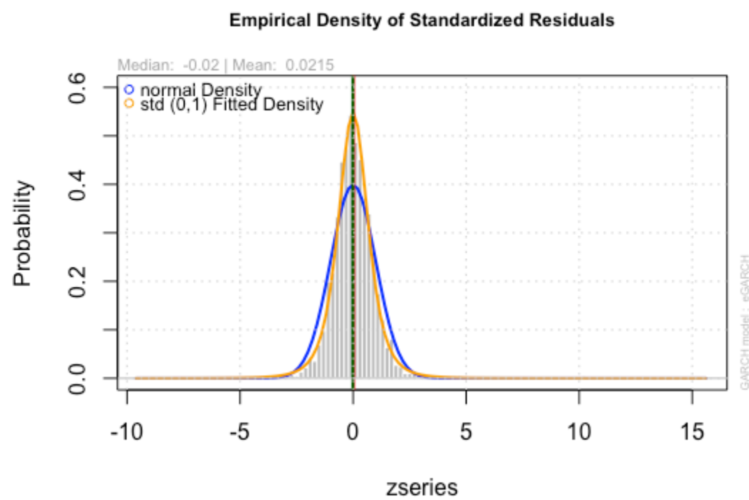


Figure 1c: Empirical density of Standardized Residuals

Empirical density of Standardized residuals is displayed in Figure 1c. The distribution appears to be symmetric and is consistent with a t-distribution.

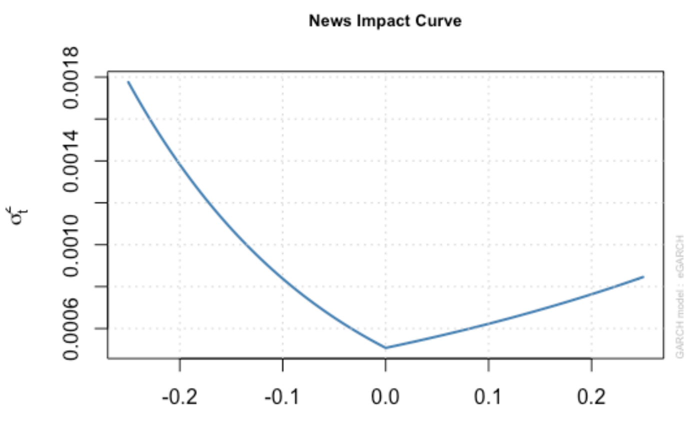


Figure 1d: Effect of volatility

Figure 1d displays the effect of the volatility on the shock. The y-axis represents volatility and x-axis is the shock value. The shock value is the t-distribution and majority of the values are varied within the range (-0.2 to 0.2). The impact curve captures the asymmetric behavior as the leverage parameter is significant.

Summary: Residual analysis of EGARCH model with t-distributed error terms shows that the model fits the data adequately --> residuals are white noise and show no ARCH effect. Ljung Box test (Q-statistic) on residuals is not significant, showing that hypothesis of no correlation for residuals cannot be rejected. Similarly, the Ljung Box test (Q-statistic) on the squared standardized residuals is not significant suggesting that residuals show no ARCH/GARCH effect. The Adjusted Pearson goodness of fit test is not significant indicating that the distribution of the error terms can be described by a t-distribution with 4 degrees of freedom, as also shown by the QQ plot of the residuals.

Forecast Analysis and Backtesting:

Table 8: One-step ahead forecasts for Amazon's stock returns

```
Model: eGARCH
Horizon: 12
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=2016-11-16]:
       Series    Sigma
T+1  0.0005203 0.02237
T+2  0.0005203 0.02237
T+3  0.0005203 0.02237
T+4  0.0005203 0.02237
T+5  0.0005203 0.02237
T+6  0.0005203 0.02237
T+7  0.0005203 0.02237
T+8  0.0005203 0.02238
T+9  0.0005203 0.02238
T+10 0.0005203 0.02238
T+11 0.0005203 0.02238
T+12 0.0005203 0.02238
```

Based on the analysis above, the best model is the AR (0) – EGARCH (1,1) model with t-distributed error terms. This analysis supports the choice of the t-distribution and the model has also the smallest BIC value. One step ahead forecasts were computed using the ugarchforecast function and the results are displayed in Table 7.

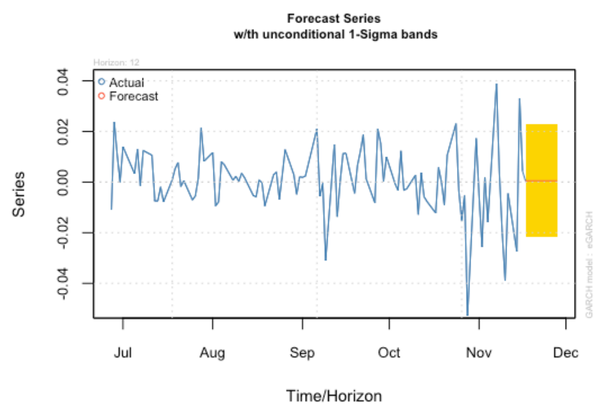


Figure 2a: Time series Prediction (unconditional)

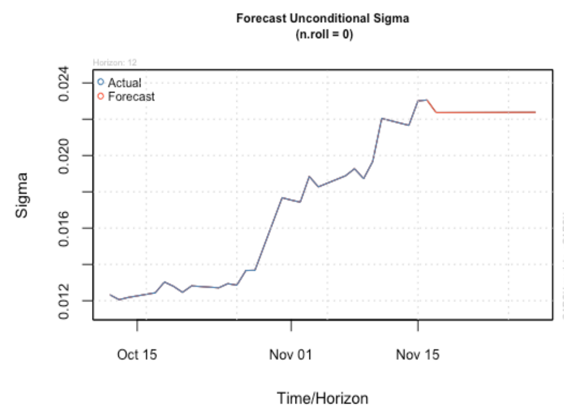


Figure 2b: Sigma prediction (unconditional)

Sigma is the predicted conditional volatility at time t+h and is displayed in Figure 2a. Predicted volatility converges to overall (unconditional) standard deviation of time series. Series is the predicted conditional mean at time t+h, displayed in Figure 2b. Predicted mean remains constant because the mean model on r_t is constant.

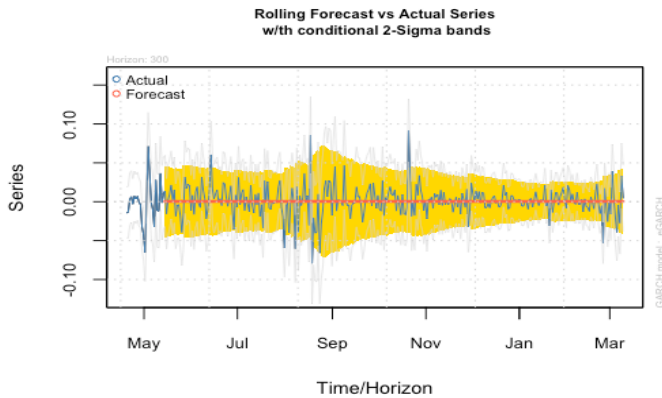


Figure 2c: Rolling Forecast Vs Actual Series

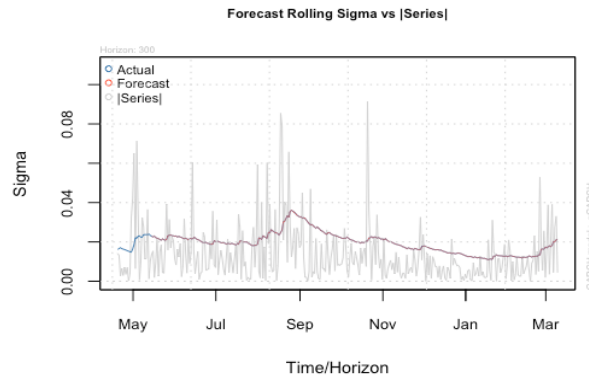


Figure 2d: Rolling Sigma vs |Series|

The graphs display the results of applying step ahead forecasts with a rolling window of 300. The rolling window step ahead forecast for the prediction of conditional mean is shown in Figure 2c. Prediction of stochastic volatility with a rolling window to move forward and plot the volatility based on the rolling forecast is depicted in Figure 2d.

GARCH Roll Mean Forecast Performance Measures --- Model : eGARCH No.Refits : 9 No.Forecasts: 1690 Stats MSE 0.0004192 MAE 0.0142200 DAC 0.5136000	GARCH Roll Mean Forecast Performance Measures --- Model : sGARCH No.Refits : 9 No.Forecasts: 1690 Stats MSE 0.0004192 MAE 0.0142200 DAC 0.5136000

Table 9: Forecast evaluation statistics using the "fpm" method

Since the model has a lot of variations, after a certain number of forecasts the model becomes useless for forecasting. Backtesting procedure was used recursively to compare the two models and was refit every 200 days. The forecast evaluation statistics were calculated using the fpm method. The GARCH and EGARCH models models are compared to choose best model that minimizes error (MSE - Mean squared error). The results of the procedure are displayed in Table 10 - Both models had similar values for MSE.

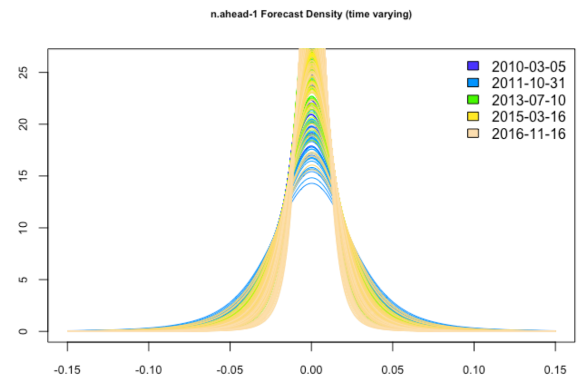


Figure 3: Density forecast plot

Figure 3 depicts the density forecast plot for the EGARCH model. The graph shows how the distribution changes depending on the time period used for the forecast. The forecasts in 2015 and 2016 for the Amazon's daily stock returns data have a tighter distribution and values have a smaller range and are more similar to each other. While 2010 and 2011 have a large fluctuation and the values are distributed.

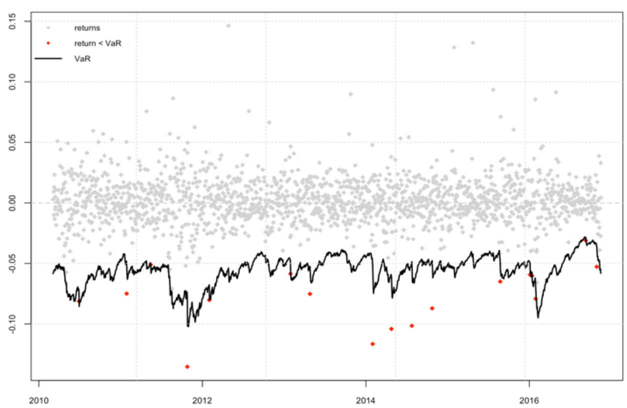


Figure 4: VaR Forecast

The value at Risk forecast for the Amazon's daily stock returns data after applying the backtesting procedure is displayed in Figure 4. The graph shows the potential expected loss that can be incurred by a trader or bank over a given period and for a given portfolio of assets. The red dots in the figure are the missed negative returns which are beyond the estimated Value at risk values.

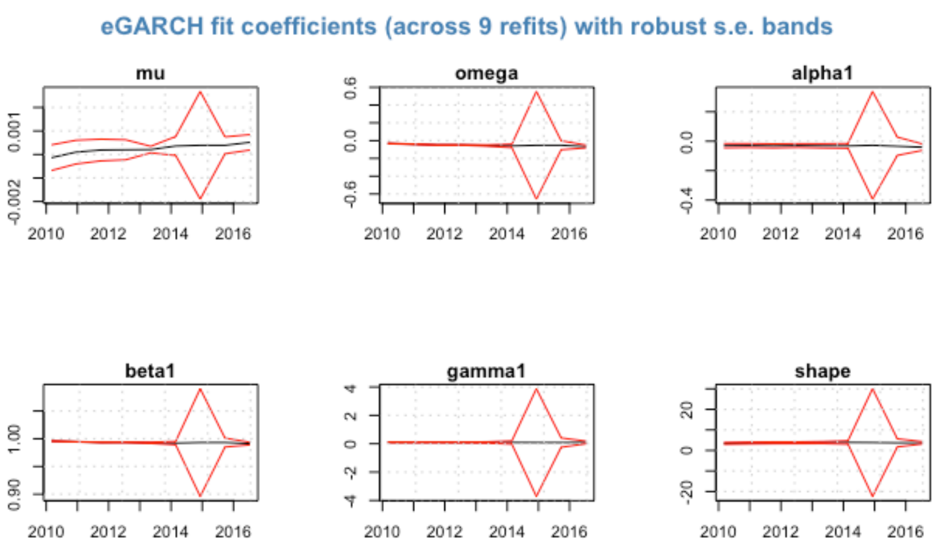


Figure 5: Fit Coefficients (with s.e. bands)

Figure 5 above displays the fir coefficients for the EGARCH model. The plot shows how the different coefficients of the model change based on the rolling window specified in the backtesting procedure. As we can see, there are lot of changes seen during the 2014-16 period. The leverage parameter alpha1 is significant during the 2015 period. Monitoring the coefficients of the model seems to be extremely important for time series analysis to maintain the model quality.

Conclusion

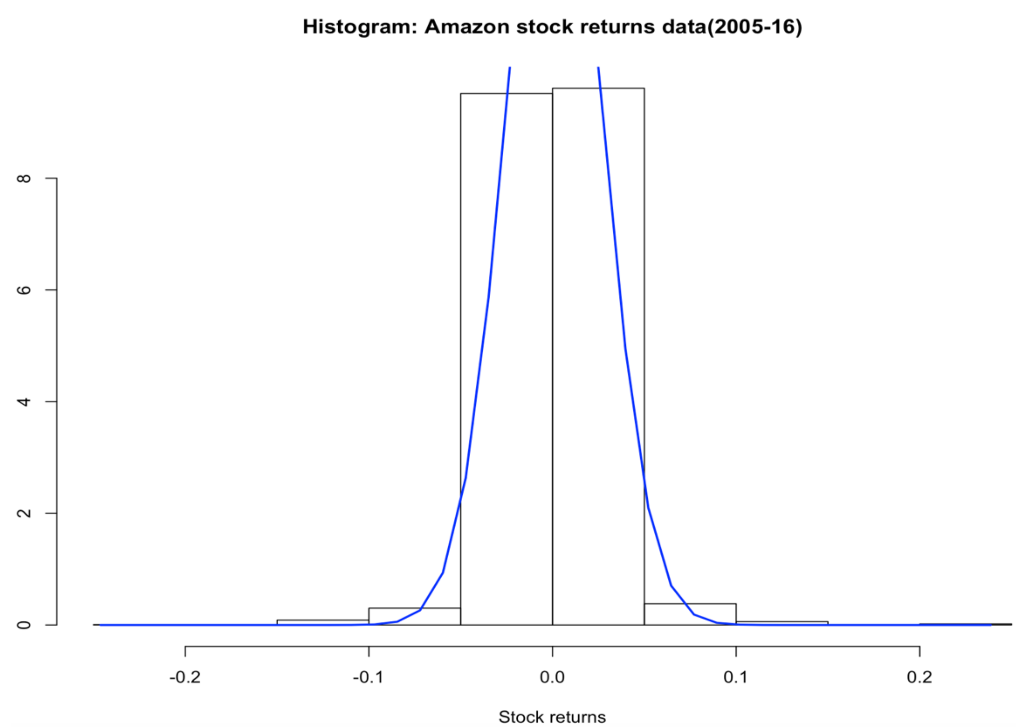
Amazon's daily stock returns data was perfectly suited for applying a time-series modeling process.

- Initial analysis of the data showed that the data was highly volatile – appeared in clusters: high in certain periods and low for the other, evolved over time in a continuous fashion.
- Time plots revealed that conditional volatility is non-constant over time with periods of high volatility and shocks influence Shocks influence the process volatility, where high volatility does not decrease quickly.
- ACF plots for the returns depicted that the Amazon stock log returns are not correlated. While, the square returns time series and series of absolute returns showed significant ACF values and proved that ARCH/GARCH effect is evident. The Ljung-Box test for squared and absolute returns further proved the presence of ARCH effect in the heteroscedastic data.
- Based on the tests for distribution goodness of fit, the most suitable for Amazon's daily stock returns data was the student's t distribution.
- Best model based on the BIC model selection criteria is the **AR (0) – EGARCH (1,1)** model with t-distribution.
- Residual analysis of EGARCH model with t-distributed error terms showed that the model fits the data adequately - residuals are white noise and show no ARCH effect. Ljung Box test (Q-statistic) further proved the same and showed no ARCH/GARCH effect.
- One step ahead forecasts applied on the EGARCH model showed that the predicted volatility converges to overall (unconditional) standard deviation of time series and predicted mean remains constant because the mean model on r_t is constant.
- Backtesting procedure was used recursively to compare the two models and was refit every 200 days. The GARCH and EGARCH models were compared to choose best model that minimizes MSE. Both the models had similar values for MSE.

Appendix

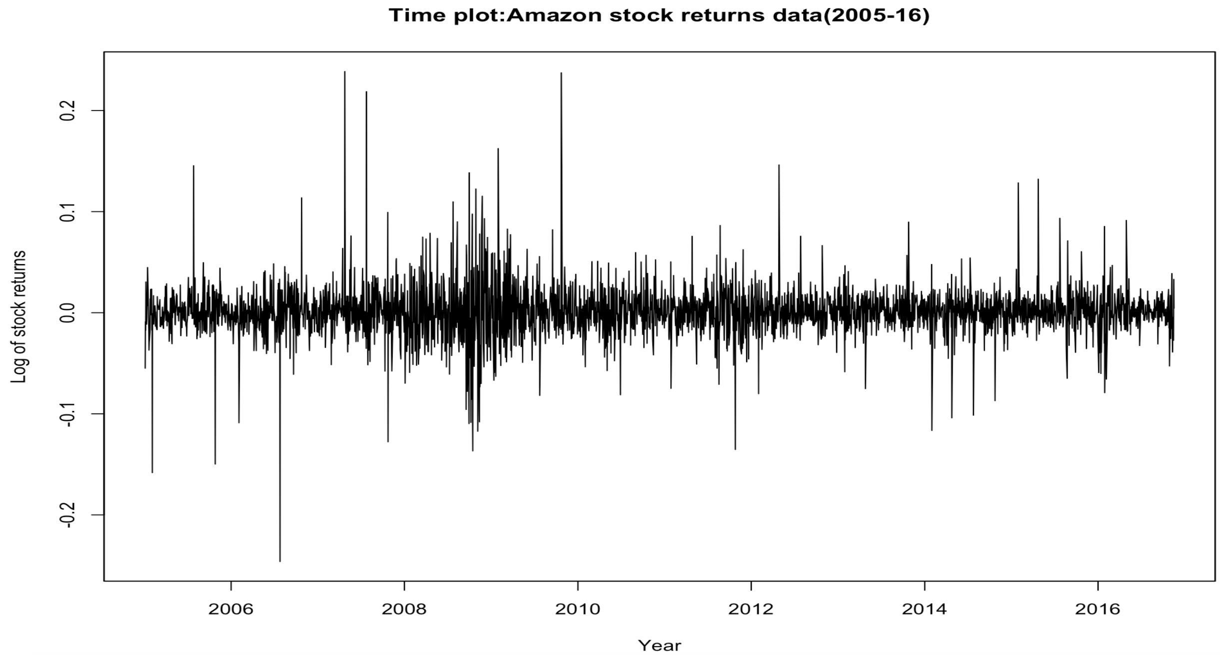
Code Output and graphs: Exploratory analysis

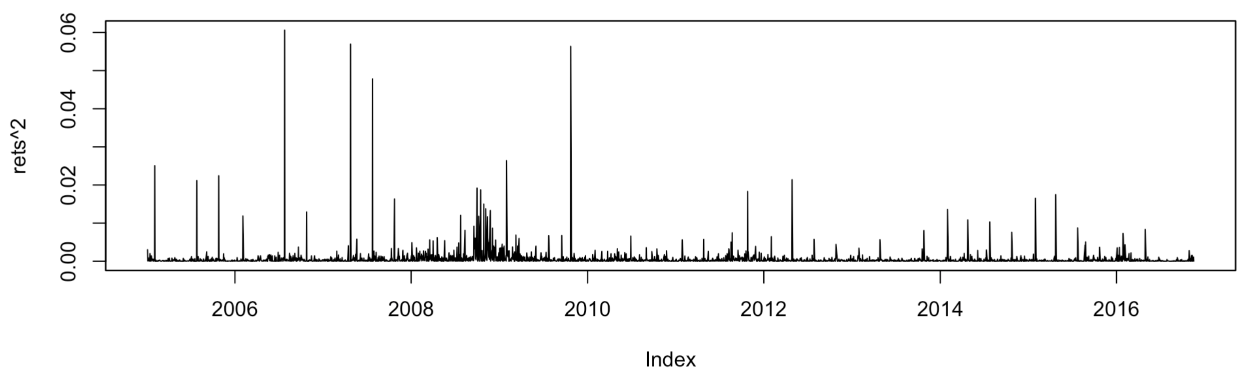
```
> ##################################################################
> ##EXPLORATORY ANALYSIS OF THE DATA
> ##################################################################
> #compute statistics
> basicStats(rets)
                       x
nobs         2990.000000
NAs             0.000000
Minimum        -0.246182
Maximum         0.238621
1. Quartile    -0.010703
3. Quartile     0.012830
Mean            0.000943
Median          0.000201
Sum             2.819444
SE Mean         0.000468
LCL Mean        0.000026
UCL Mean        0.001860
Variance        0.000654
Stdev           0.025566
Skewness        0.480386
Kurtosis       14.468528
```

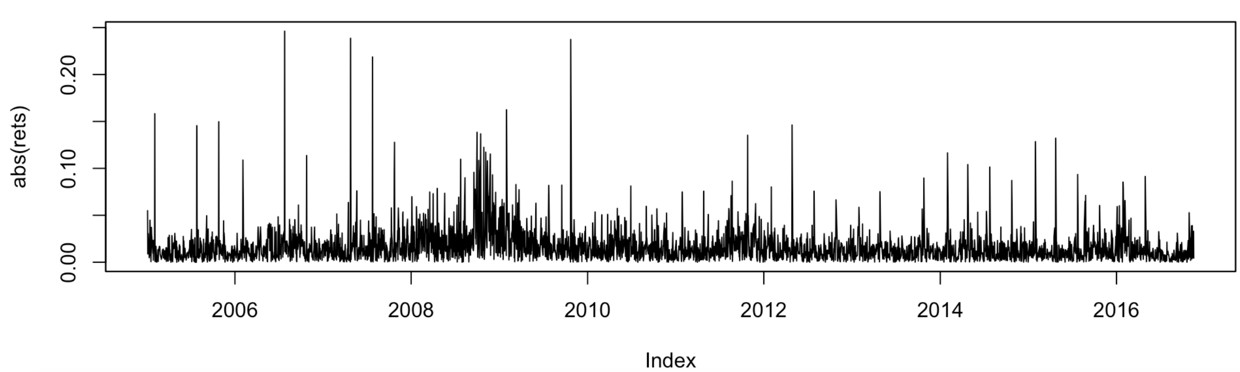


Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

TIME PLOTS:

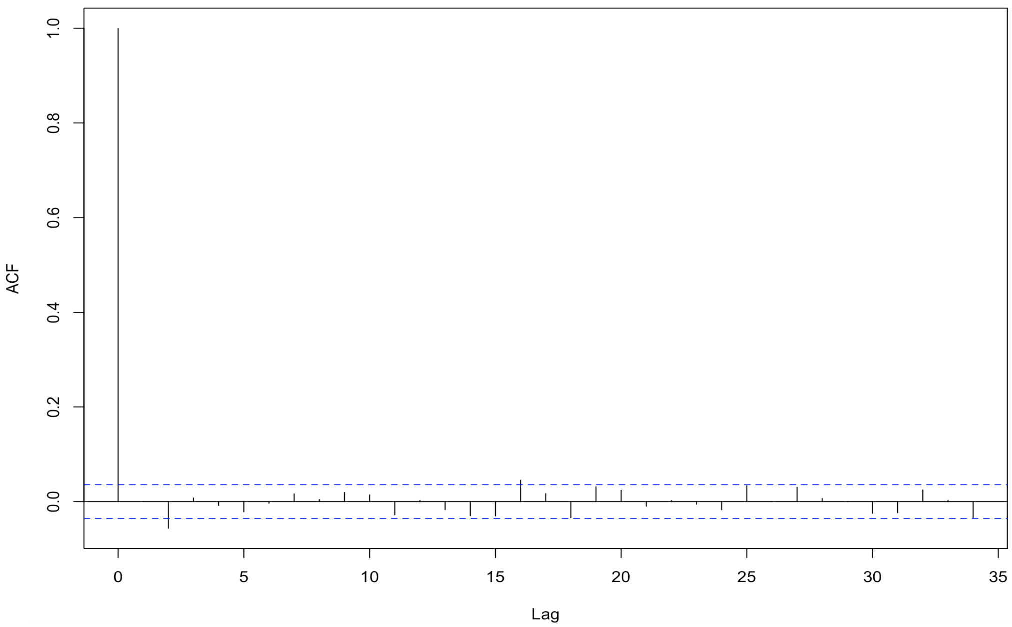


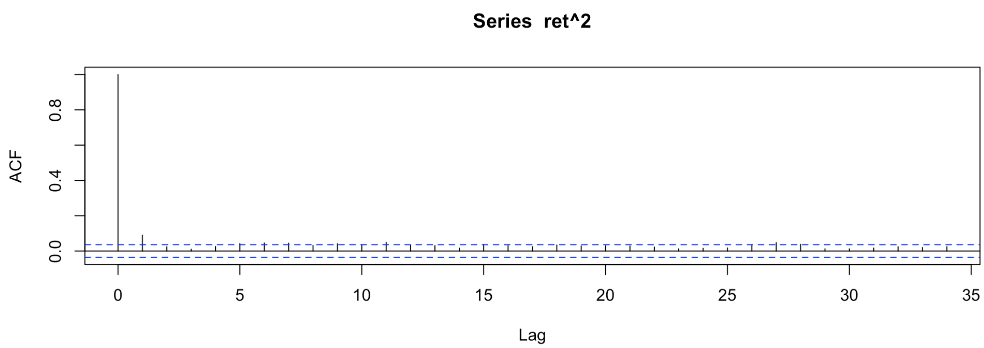


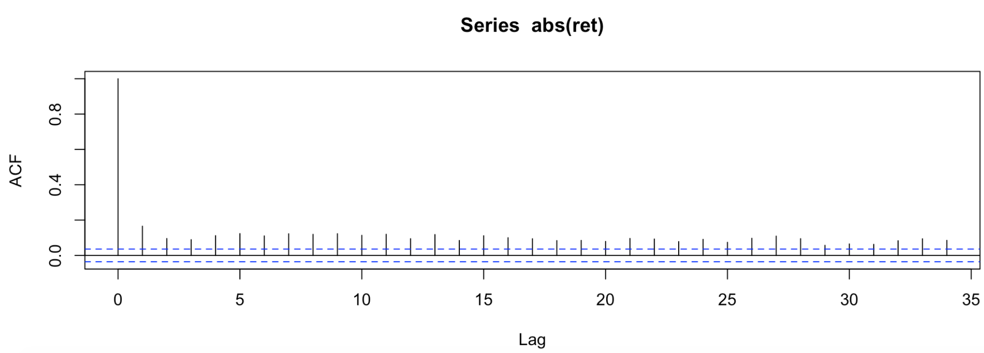


Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

ACF PLOTS:







Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

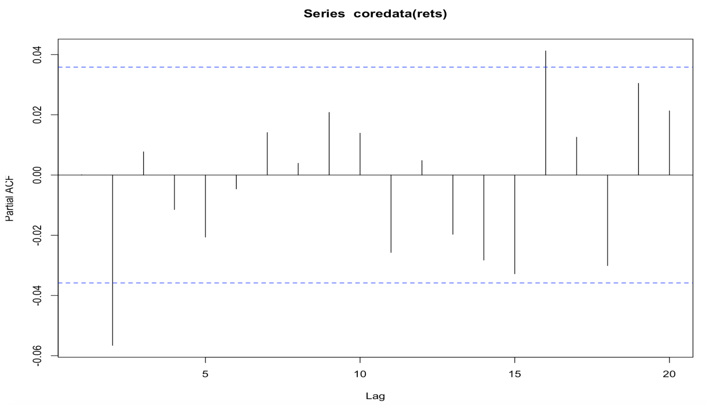
LJUNG BOX TEST RESULTS:

```
> # Computes Ljung-Box test on returns to test  independence
> Box.test(coredata(rets),lag=6,type='Ljung')

        Box-Ljung test

data:  coredata(rets)
X-squared = 11.387, df = 6, p-value = 0.07713

> Box.test(coredata(rets),lag=9,type='Ljung')

        Box-Ljung test

data:  coredata(rets)
X-squared = 13.363, df = 9, p-value = 0.1468

> Box.test(coredata(rets),lag=12,type='Ljung')

        Box-Ljung test

data:  coredata(rets)
X-squared = 16.338, df = 12, p-value = 0.1762
```

```
> # Computes Ljung-Box test on squared returns to test non-linear independence
> Box.test(coredata(rets^2),lag=6,type='Ljung')

        Box-Ljung test

data:  coredata(rets^2)
X-squared = 39.372, df = 6, p-value = 6.052e-07

> Box.test(coredata(rets^2),lag=12,type='Ljung')

        Box-Ljung test

data:  coredata(rets^2)
X-squared = 68.18, df = 12, p-value = 7.003e-10

> # Computes Ljung-Box test on absolute returns to test non-linear independence
> Box.test(abs(coredata(rets)),lag=6,type='Ljung')

        Box-Ljung test

data:  abs(coredata(rets))
X-squared = 253.05, df = 6, p-value < 2.2e-16

> Box.test(abs(coredata(rets)),lag=12,type='Ljung')

        Box-Ljung test

data:  abs(coredata(rets))
X-squared = 494.68, df = 12, p-value < 2.2e-16
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

PACF of returns:



<u>Code output and Graphs:</u> MODEL FITTING

AR (0) – GARCH (1,1) model with Normal distribution:

```
> ####################################################################
> ##MODEL FITTING, RESIDUAL ANALYSIS AND MODEL DIAGNOSTICS
> ####################################################################
> #specify model using functions in rugarch package
> #Fit ARMA(0,0)-GARCH(1,1) model with Normal distribution
> garch11.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)))
> #estimate model
> garch11.fit=ugarchfit(spec=garch11.spec, data=rets)
> garch11.fit

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.001014    0.000436    2.3275 0.019936
omega   0.000002    0.000001    3.4337 0.000595
alpha1  0.007019    0.000443   15.8600 0.000000
beta1   0.989724    0.000207 4786.8663 0.000000

Robust Standard Errors:
        Estimate  Std. Error    t value Pr(>|t|)
mu      0.001014    0.000408    2.48209 0.013061
omega   0.000002    0.000006    0.34473 0.730301
alpha1  0.007019    0.002177    3.22422 0.001263
beta1   0.989724    0.000158 6281.27717 0.000000

LogLikelihood : 6840.816
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

```
Information Criteria
------------------------------------

Akaike       -4.5731
Bayes        -4.5651
Shibata      -4.5731
Hannan-Quinn -4.5702

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.1504  0.6981
Lag[2*(p+q)+(p+q)-1][2]    3.3313  0.1142
Lag[4*(p+q)+(p+q)-1][5]    5.6069  0.1111
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      5.117  0.0237
Lag[2*(p+q)+(p+q)-1][5]     5.199  0.1379
Lag[4*(p+q)+(p+q)-1][9]     5.411  0.3717
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]   0.08359 0.500 2.000  0.7725
ARCH Lag[5]   0.15167 1.440 1.667  0.9764
ARCH Lag[7]   0.35551 2.315 1.543  0.9895

Nyblom stability test
------------------------------------
Joint Statistic:  35.275
Individual Statistics:
mu     0.07329
omega  1.08937
alpha1 0.39900
beta1  0.39348
```

```
Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.07 1.24 1.6
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
                   t-value    prob sig
Sign Bias            1.627 0.10378
Negative Sign Bias   1.744 0.08127   *
Positive Sign Bias   2.479 0.01323  **
Joint Effect        12.093 0.00707 ***


Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     329.5    1.764e-58
2    30     341.4    4.615e-55
3    40     359.2    2.072e-53
4    50     372.0    3.200e-51


Elapsed time : 0.1881399

>
```

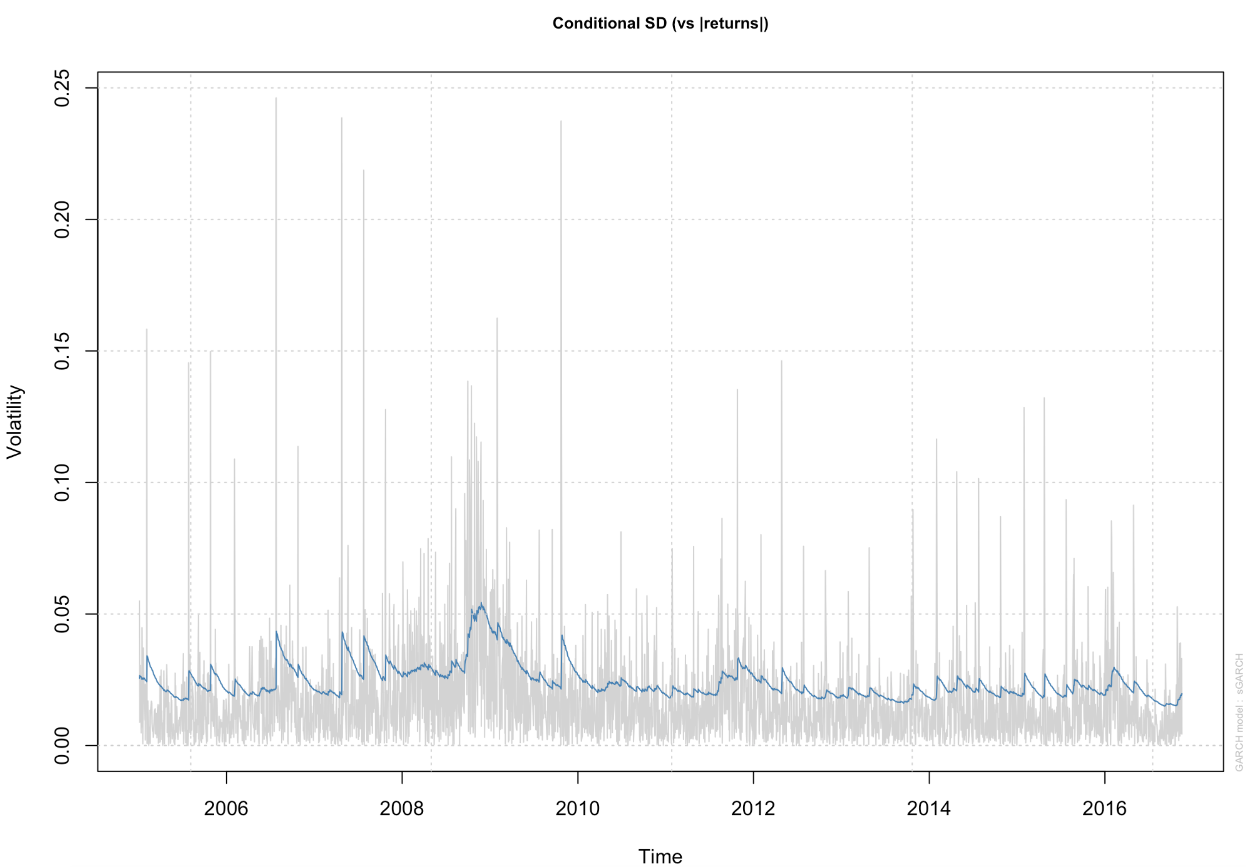
AR (0) – GARCH (1,1) model with t distribution:

```
> #Fit ARMA(0,0)-GARCH(1,1) model with t-distribution
> garch11.t.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)), distribution.model = "std")
> #estimate model
> garch11.t.fit=ugarchfit(spec=garch11.t.spec, data=rets)
> garch11.t.fit

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000654    0.000323   2.0234 0.043034
omega   0.000005    0.000002   2.3719 0.017698
alpha1  0.023268    0.003292   7.0679 0.000000
beta1   0.967858    0.001744 554.9706 0.000000
shape   3.520292    0.219236  16.0571 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000654    0.000297   2.1983 0.027931
omega   0.000005    0.000006   0.8342 0.404166
alpha1  0.023268    0.006944   3.3508 0.000806
beta1   0.967858    0.001431 676.4558 0.000000
shape   3.520292    0.241549  14.5738 0.000000

LogLikelihood : 7331.136
```

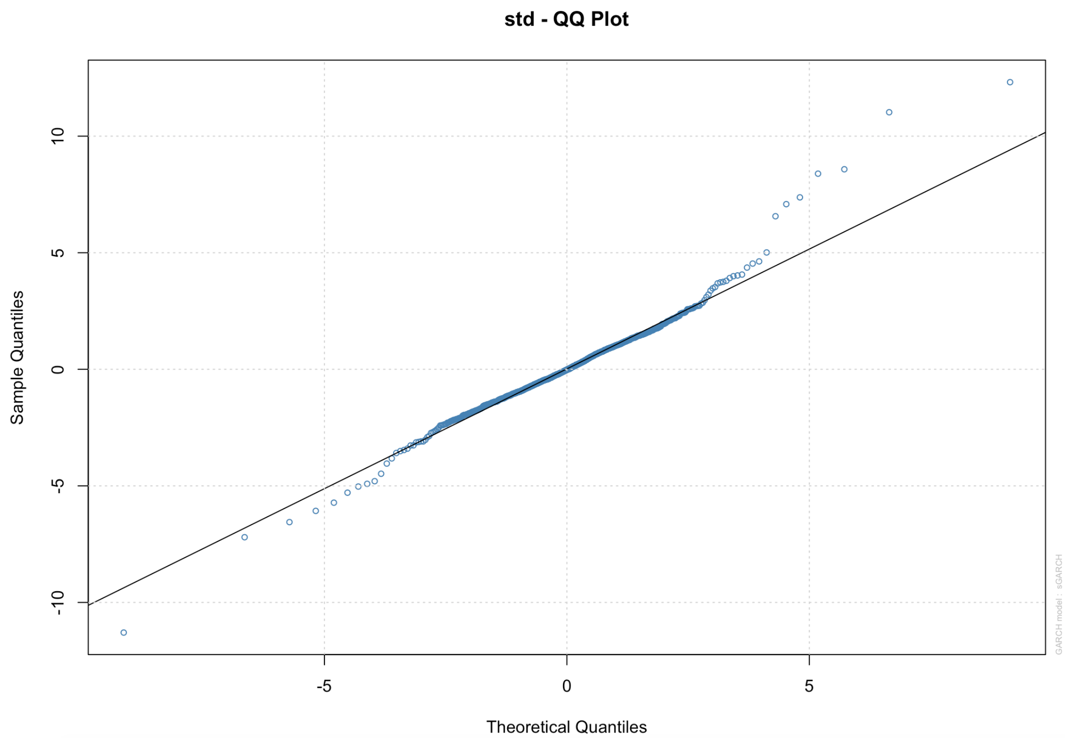
```
Information Criteria
------------------------------------

Akaike       -4.9004
Bayes        -4.8904
Shibata      -4.9004
Hannan-Quinn -4.8968

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                    0.07406  0.7855
Lag[2*(p+q)+(p+q)-1][2]   2.28295  0.2198
Lag[4*(p+q)+(p+q)-1][5]   4.00753  0.2531
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.8157  0.3664
Lag[2*(p+q)+(p+q)-1][5]    1.3013  0.7886
Lag[4*(p+q)+(p+q)-1][9]    1.6807  0.9395
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.3374 0.500 2.000  0.5613
ARCH Lag[5]    0.5563 1.440 1.667  0.8669
ARCH Lag[7]    0.8265 2.315 1.543  0.9401
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     40.74     0.002611
2    30     42.47     0.050940
3    40     54.23     0.053315
4    50     61.00     0.116678


Elapsed time : 0.30674

> #using extractors
> #estimated coefficients:
> coef(garch11.fit)
          mu        omega       alpha1        beta1
1.013923e-03 1.973723e-06 7.019118e-03 9.897242e-01
> #unconditional mean in mean equation
> uncmean(garch11.fit)
[1] 0.001013923
> #unconditional varaince: omega/(alpha1+beta1)
> uncvariance(garch11.fit)
[1] 0.0006060535
> #persistence = alpha1+beta1
> persistence(garch11.fit)
[1] 0.9967433
> #half-life: ln(0.5)/ln(alpha1+beta1)
> halflife(garch11.fit)
[1] 212.4917
>
```



Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy



AR (0) – GARCH (1,1) model with skewed t distribution:

```
> garch11.skt.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)), distribution.mode
l = "sstd")
> #estimate model
> garch11.skt.fit=ugarchfit(spec=garch11.skt.spec, data=rets)
> garch11.skt.fit

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001191    0.000382   3.1164 0.001831
omega   0.000005    0.000002   2.3438 0.019090
alpha1  0.023842    0.003453   6.9047 0.000000
beta1   0.967460    0.001749 553.2030 0.000000
skew    1.069798    0.027359  39.1024 0.000000
shape   3.495642    0.216538  16.1433 0.000000

Robust Standard Errors:
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.001191    0.000359   3.32003 0.000900
omega   0.000005    0.000006   0.80138 0.422913
alpha1  0.023842    0.007697   3.09735 0.001953
beta1   0.967460    0.001477 654.89059 0.000000
skew    1.069798    0.029016  36.86961 0.000000
shape   3.495642    0.238741  14.64198 0.000000

LogLikelihood : 7334.61

Information Criteria
------------------------------------

Akaike       -4.9021
Bayes        -4.8900
Shibata      -4.9021
Hannan-Quinn -4.8977

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                    0.07519  0.7839
Lag[2*(p+q)+(p+q)-1][2]   2.26088  0.2229
Lag[4*(p+q)+(p+q)-1][5]   3.97215  0.2576
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.8013  0.3707
Lag[2*(p+q)+(p+q)-1][5]    1.2771  0.7944
Lag[4*(p+q)+(p+q)-1][9]    1.6519  0.9422
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.3355 0.500 2.000  0.5624
ARCH Lag[5]    0.5537 1.440 1.667  0.8677
ARCH Lag[7]    0.8189 2.315 1.543  0.9411
```

```
Nyblom stability test
------------------------------------
Joint Statistic:  4.5436
Individual Statistics:
mu     0.2645
omega  0.2776
alpha1 1.0073
beta1  0.9291
skew   0.3181
shape  0.9467

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.49 1.68 2.12
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
                   t-value    prob sig
Sign Bias           1.7742 0.07613   *
Negative Sign Bias  0.3648 0.71528
Positive Sign Bias  1.2668 0.20531
Joint Effect        4.9444 0.17591


Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     40.37     0.002927
2    30     34.54     0.219980
3    40     57.22     0.029946
4    50     67.76     0.039080


Elapsed time : 0.57265

> persistence(garch11.skt.fit)
[1] 0.9913018
```

AR (0) – EGARCH (1,1) model with Gaussian Normal distribution:

```
> #Fit ARMA(0,0)-eGARCH(1,1) model with  Gaussian distribution
> egarch11.spec=ugarchspec(variance.model=list(model = "eGARCH", garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)))
> #estimate model
> egarch11.fit=ugarchfit(spec=egarch11.spec, data=ret)
> egarch11.fit

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001573    0.000389   4.0451 0.000052
omega  -0.332082    0.005676 -58.5019 0.000000
alpha1 -0.027958    0.010436  -2.6791 0.007382
beta1   0.953261    0.001080 882.2554 0.000000
gamma1  0.124391    0.007649  16.2625 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001573    0.000370   4.2496 0.000021
omega  -0.332082    0.023103 -14.3741 0.000000
alpha1 -0.027958    0.025400  -1.1007 0.271016
beta1   0.953261    0.003678 259.1852 0.000000
gamma1  0.124391    0.019168   6.4895 0.000000

LogLikelihood : 6876.352
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

```
Information Criteria
------------------------------------

Akaike       -4.5962
Bayes        -4.5862
Shibata      -4.5962
Hannan-Quinn -4.5926

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                      0.271  0.6026
Lag[2*(p+q)+(p+q)-1][2]     2.438  0.1994
Lag[4*(p+q)+(p+q)-1][5]     4.223  0.2275
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      1.813  0.1782
Lag[2*(p+q)+(p+q)-1][5]     2.611  0.4826
Lag[4*(p+q)+(p+q)-1][9]     3.280  0.7120
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.5478 0.500 2.000  0.4592
ARCH Lag[5]    0.8000 1.440 1.667  0.7931
ARCH Lag[7]    1.3678 2.315 1.543  0.8475

Nyblom stability test
------------------------------------
Joint Statistic:  1.1621
Individual Statistics:
mu     0.3130
omega  0.6619
alpha1 0.1809
beta1  0.6554
gamma1 0.4858

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.28 1.47 1.88
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
                   t-value    prob sig
Sign Bias          2.023669 0.04309  **
Negative Sign Bias 0.001678 0.99866
Positive Sign Bias 1.612243 0.10701
Joint Effect       5.417916 0.14363


Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     301.8    8.450e-53
2    30     314.7    1.013e-49
3    40     324.3    1.194e-46
4    50     342.8    1.069e-45
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

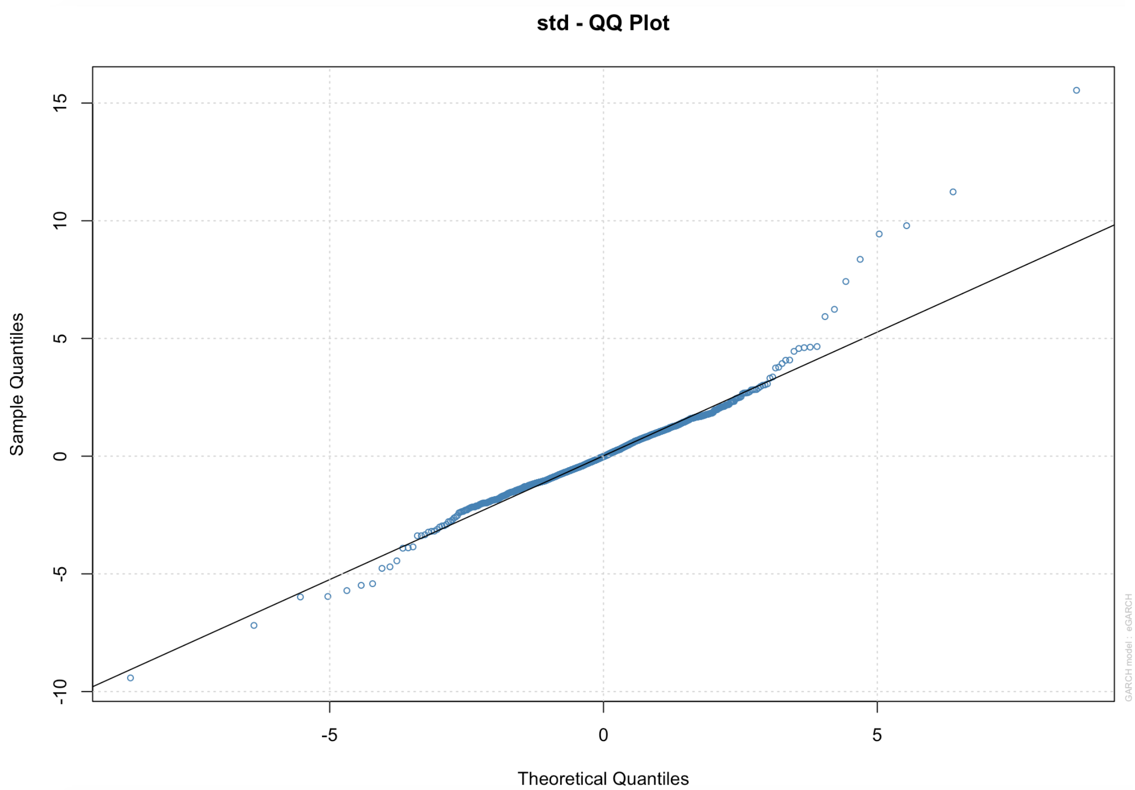
AR (0) – EGARCH (1,1) model with t distribution:

```
> #Fit ARMA(0,0)-eGARCH(1,1) model with t-distribution
> egarch11.t.spec=ugarchspec(variance.model=list(model = "eGARCH", garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)),
distribution.model = "std")
> #estimate model
> egarch11.t.fit=ugarchfit(spec=egarch11.t.spec, data=rets)
> egarch11.t.fit

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

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(1,1)
Mean Model      : ARFIMA(0,0,0)
Distribution    : std

Optimal Parameters
------------------------------------
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.000520    0.000314    1.6593 0.097054
omega  -0.077466    0.004346  -17.8250 0.000000
alpha1 -0.047202    0.010498   -4.4963 0.000007
beta1   0.989791    0.000563 1757.5086 0.000000
gamma1  0.099616    0.017738    5.6159 0.000000
shape   3.726901    0.254305   14.6552 0.000000

Robust Standard Errors:
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.000520    0.000288    1.8094 0.070395
omega  -0.077466    0.004531  -17.0953 0.000000
alpha1 -0.047202    0.013088   -3.6065 0.000310
beta1   0.989791    0.000619 1598.4903 0.000000
gamma1  0.099616    0.027427    3.6320 0.000281
shape   3.726901    0.320116   11.6423 0.000000

LogLikelihood : 7371.342

Information Criteria
------------------------------------

Akaike       -4.9267
Bayes        -4.9146
Shibata      -4.9267
Hannan-Quinn -4.9223

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                      1.152  0.2830
Lag[2*(p+q)+(p+q)-1][2]     3.367  0.1116
Lag[4*(p+q)+(p+q)-1][5]     5.278  0.1323
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      6.173 0.01297
Lag[2*(p+q)+(p+q)-1][5]     6.702 0.06120
Lag[4*(p+q)+(p+q)-1][9]     7.474 0.16264
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.3488 0.500 2.000  0.5548
ARCH Lag[5]    0.4816 1.440 1.667  0.8890
ARCH Lag[7]    1.2865 2.315 1.543  0.8630
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

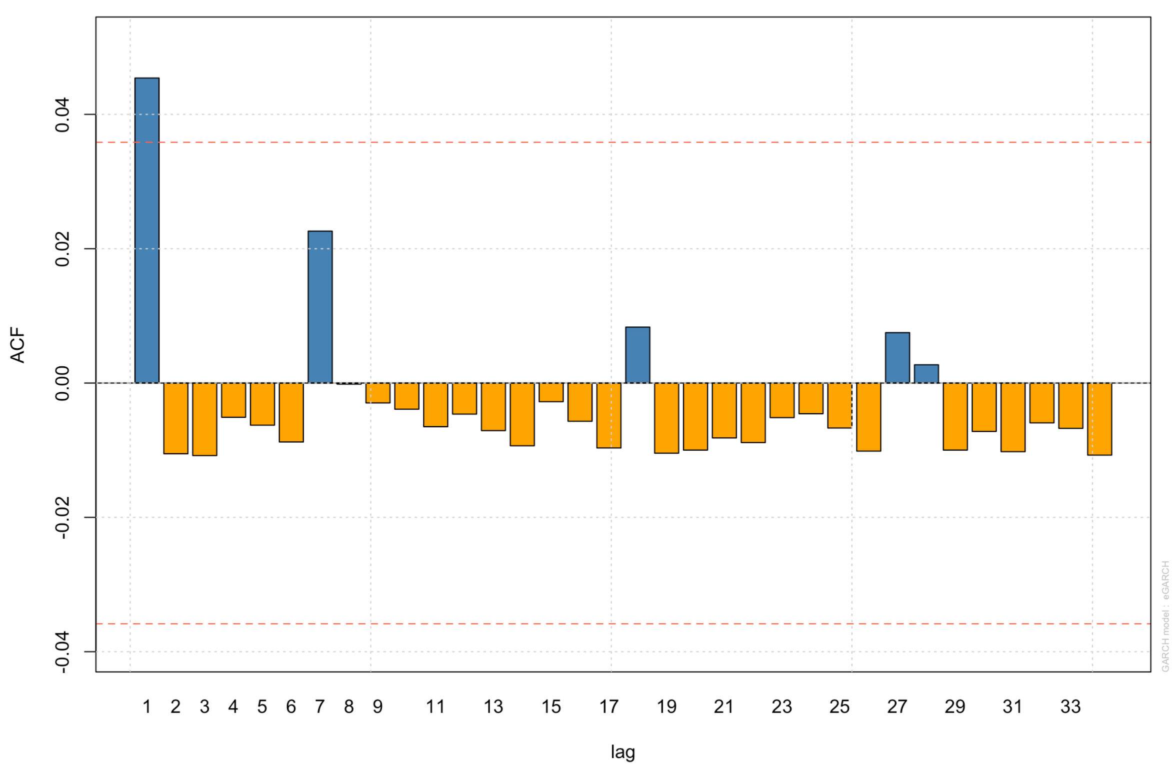
```
Nyblom stability test
------------------------------------
Joint Statistic:  2.6379
Individual Statistics:
mu     0.1543
omega  0.3893
alpha1 1.1301
beta1  0.3405
gamma1 0.1296
shape  0.3324

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.49 1.68 2.12
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
                   t-value    prob sig
Sign Bias           2.3400 0.01935  **
Negative Sign Bias  0.7564 0.44949
Positive Sign Bias  1.7484 0.08050   *
Joint Effect        6.0739 0.10807


Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     33.01      0.02398
2    30     49.73      0.00965
3    40     48.66      0.13808
4    50     60.30      0.12919


Elapsed time : 0.5454619

> persistence(egarch11.t.fit)
[1] 0.9897913
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

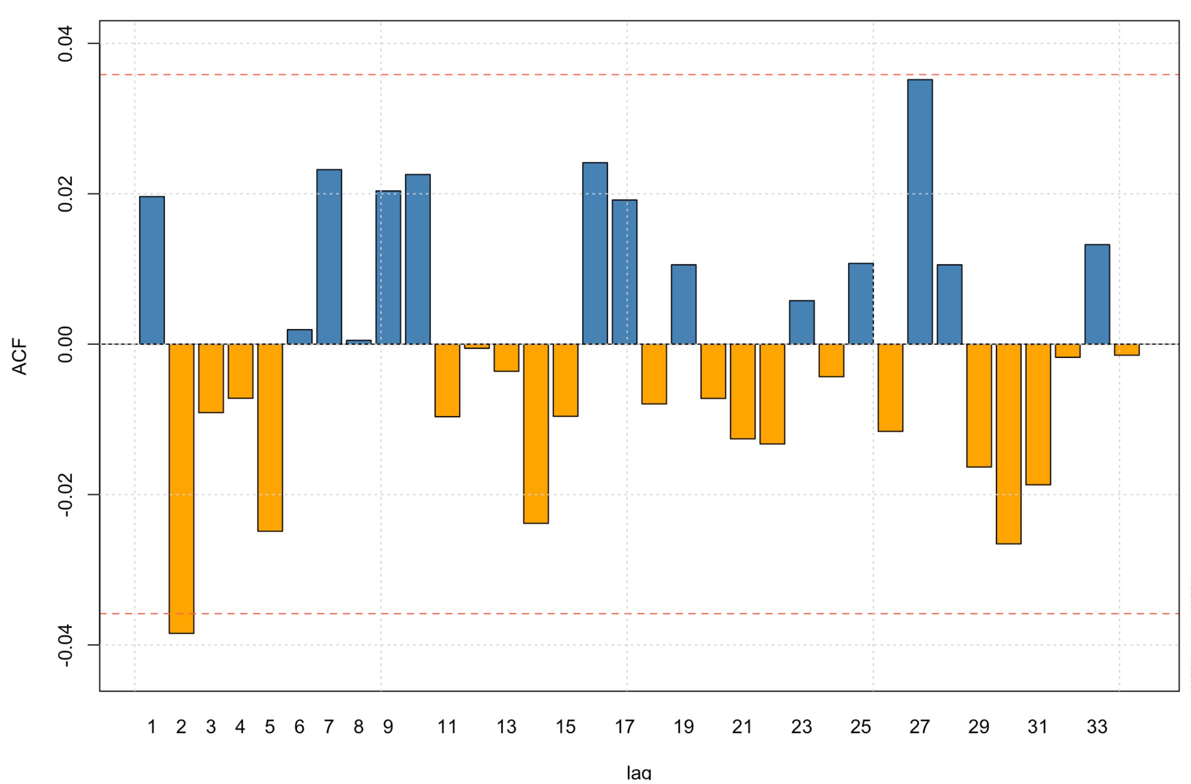


Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

ACF of Squared Standardized Residuals



ACF of Standardized Residuals



```
> egarch11.t.fcst=ugarchforecast(egarch11.t.fit, n.ahead=12)
> egarch11.t.fcst

*------------------------------------*
*       GARCH Model Forecast         *
*------------------------------------*
Model: eGARCH
Horizon: 12
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=2016-11-16]:
        Series   Sigma
T+1  0.0005203 0.02237
T+2  0.0005203 0.02237
T+3  0.0005203 0.02237
T+4  0.0005203 0.02237
T+5  0.0005203 0.02237
T+6  0.0005203 0.02237
T+7  0.0005203 0.02237
T+8  0.0005203 0.02238
T+9  0.0005203 0.02238
T+10 0.0005203 0.02238
T+11 0.0005203 0.02238
T+12 0.0005203 0.02238
```

AR (0) – TGARCH (1,1) model with t distribution:

```
> #Fit ARMA(0,0)-TGARCH(1,1) model with t-distribution
> gjrgarch11.t.spec=ugarchspec(variance.model=list(model = "gjrGARCH", garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0
)), distribution.model = "std")
> #estimate model
> gjrgarch11.t.fit=ugarchfit(spec=gjrgarch11.t.spec, data=ret)
> gjrgarch11.t.fit

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

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : gjrGARCH(1,1)
Mean Model      : ARFIMA(0,0,0)
Distribution    : std

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000586    0.000317   1.8485 0.064536
omega   0.000008    0.000002   5.1364 0.000000
alpha1  0.013866    0.002528   5.4845 0.000000
beta1   0.951581    0.004447 213.9717 0.000000
gamma1  0.045377    0.010258   4.4237 0.000010
shape   3.602001    0.201408  17.8841 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000586    0.000302   1.9427 0.052058
omega   0.000008    0.000003   2.6467 0.008128
alpha1  0.013866    0.006676   2.0771 0.037792
beta1   0.951581    0.006057 157.1009 0.000000
gamma1  0.045377    0.014884   3.0486 0.002299
shape   3.602001    0.313859  11.4765 0.000000

LogLikelihood : 7340.752
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

```
Information Criteria
------------------------------------

Akaike       -4.9062
Bayes        -4.8941
Shibata      -4.9062
Hannan-Quinn -4.9019

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.1777  0.6734
Lag[2*(p+q)+(p+q)-1][2]    2.3725  0.2078
Lag[4*(p+q)+(p+q)-1][5]    4.1891  0.2314
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      1.313  0.2518
Lag[2*(p+q)+(p+q)-1][5]     1.939  0.6329
Lag[4*(p+q)+(p+q)-1][9]     2.434  0.8474
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.4208 0.500 2.000  0.5165
ARCH Lag[5]    0.6309 1.440 1.667  0.8444
ARCH Lag[7]    1.0218 2.315 1.543  0.9100

Nyblom stability test
------------------------------------
Joint Statistic:  12.1167
Individual Statistics:
mu     0.1608
omega  3.4685
alpha1 1.2749
beta1  1.2152
gamma1 1.2857
shape  1.1194

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.49 1.68 2.12
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
                   t-value    prob sig
Sign Bias           2.0601 0.03947  **
Negative Sign Bias  0.3545 0.72296
Positive Sign Bias  1.3485 0.17761
Joint Effect        4.7807 0.18858


Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     27.06      0.10334
2    30     44.13      0.03564
3    40     55.81      0.03955
4    50     55.15      0.25341
```

AR (0) – gjrGARCH (1,1) model with t distribution:

```
> #Fit ARMA(0,0)-gjrGARCH(1,1) model with skewed t-distribution
> gjrgarch11.t.spec=ugarchspec(variance.model=list(model = "gjrGARCH", garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0
)), distribution.model = "sstd")
> #estimate model
> gjrgarch11.t.fit=ugarchfit(spec=gjrgarch11.t.spec, data=ret)
> gjrgarch11.t.fit

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

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : gjrGARCH(1,1)
Mean Model      : ARFIMA(0,0,0)
Distribution    : sstd

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001087    0.000367   2.9635 0.003042
omega   0.000008    0.000002   4.2601 0.000020
alpha1  0.014076    0.002500   5.6304 0.000000
beta1   0.951934    0.004305 221.1311 0.000000
gamma1  0.044474    0.009797   4.5395 0.000006
skew    1.069057    0.027183  39.3278 0.000000
shape   3.586937    0.196370  18.2662 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001087    0.000364   2.9892 0.002797
omega   0.000008    0.000004   1.9471 0.051523
alpha1  0.014076    0.007565   1.8605 0.062816
beta1   0.951934    0.006068 156.8800 0.000000
gamma1  0.044474    0.014220   3.1276 0.001762
skew    1.069057    0.028297  37.7792 0.000000
shape   3.586937    0.330196  10.8630 0.000000

LogLikelihood : 7344.158


Information Criteria
------------------------------------

Akaike       -4.9078
Bayes        -4.8937
Shibata      -4.9078
Hannan-Quinn -4.9027

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                     0.1664  0.6833
Lag[2*(p+q)+(p+q)-1][2]    2.3909  0.2054
Lag[4*(p+q)+(p+q)-1][5]    4.2478  0.2247
d.o.f=0
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      1.309  0.2525
Lag[2*(p+q)+(p+q)-1][5]     1.917  0.6382
Lag[4*(p+q)+(p+q)-1][9]     2.406  0.8515
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.4163 0.500 2.000  0.5188
ARCH Lag[5]    0.6231 1.440 1.667  0.8468
ARCH Lag[7]    1.0097 2.315 1.543  0.9120
```

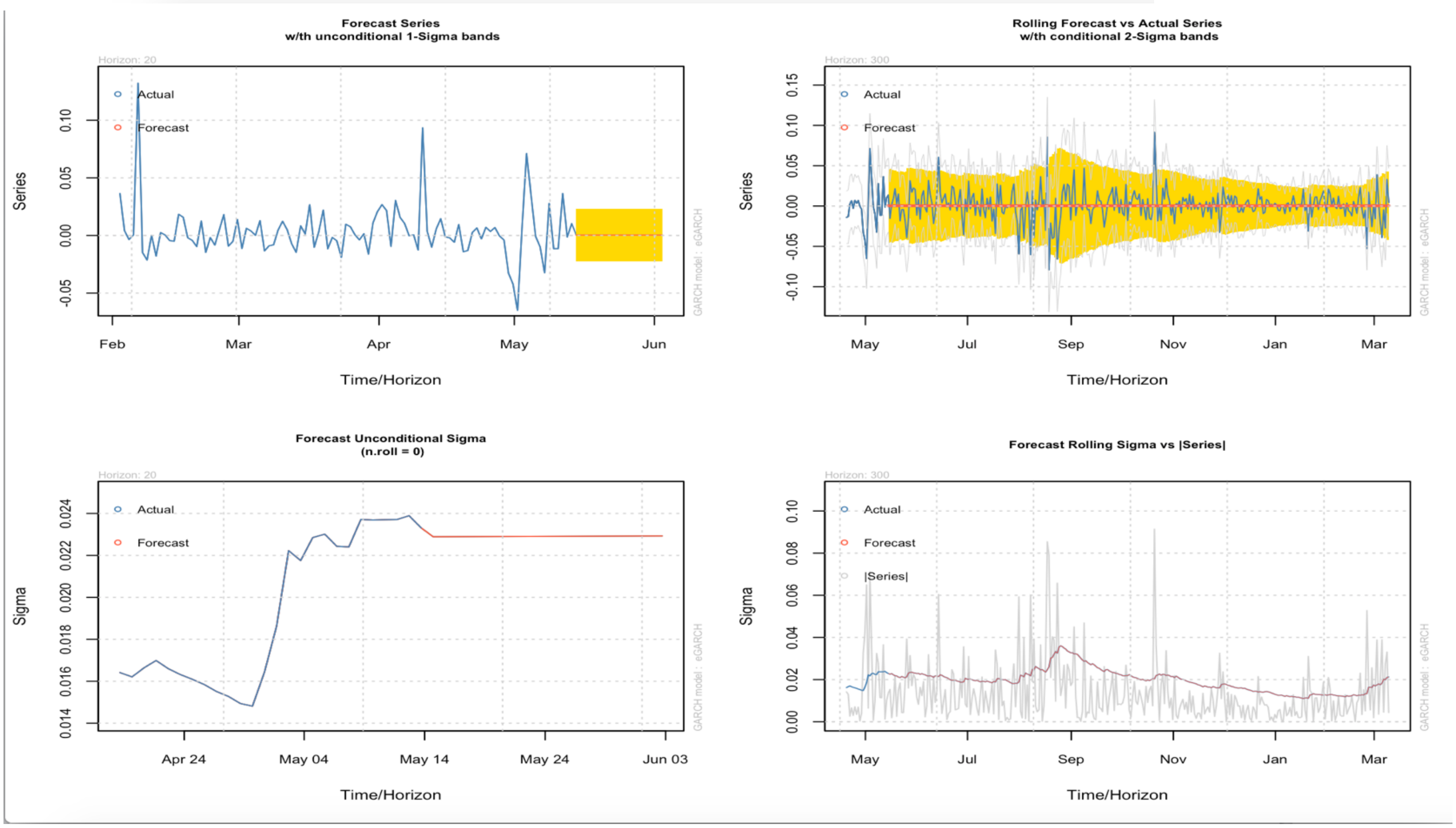
```
Nyblom stability test
------------------------------------
Joint Statistic:  10.152
Individual Statistics:
mu     0.1387
omega  3.0212
alpha1 1.2744
beta1  1.2134
gamma1 1.2619
skew   0.2607
shape  1.1213

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.69 1.9 2.35
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
                   t-value    prob sig
Sign Bias           2.0708 0.03846  **
Negative Sign Bias  0.3211 0.74812
Positive Sign Bias  1.3972 0.16246
Joint Effect        4.8889 0.18011


Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     26.39      0.11974
2    30     40.60      0.07456
3    40     51.55      0.08598
4    50     63.48      0.08001
```

MODEL COMPARISON:

```
> # MODEL COMPARISON
> # compare information criteria
> model.list = list(garch11 = garch11.fit, garch11.t = garch11.t.fit,
+                   egarch11 = egarch11.t.fit,
+                   gjrgarch11 = gjrgarch11.t.fit)
>
> info.mat = sapply(model.list, infocriteria)
> rownames(info.mat) = rownames(infocriteria(garch11.fit))
> info.mat
               garch11 garch11.t  egarch11 gjrgarch11
Akaike       -4.573121 -4.900425 -4.926650  -4.907798
Bayes        -4.565090 -4.890387 -4.914604  -4.893744
Shibata      -4.573125 -4.900431 -4.926658  -4.907809
Hannan-Quinn -4.570232 -4.896814 -4.922316  -4.902742
>
```

FORECAST ANALYSIS AND BACKTESTING:

```
> ####################################################################
> ##FORECAST ANALYSIS AND BACKTESTING
> ####################################################################
> # re-fit models leaving 300 out-of-sample observations for forecast
> # evaluation statistics
> egarch11.t.fit = ugarchfit(egarch11.t.spec, data=ret, out.sample=300)
>
> egarch11.t.fcst = ugarchforecast(egarch11.t.fit, n.roll=300, n.ahead=20)
> egarch11.t.fcst

*------------------------------------*
*       GARCH Model Forecast         *
*------------------------------------*
Model: eGARCH
Horizon: 20
Roll Steps: 300
Out of Sample: 20

0-roll forecast [T0=1977-05-13 17:00:00]:
        Series   Sigma
T+1  0.0003799 0.02289
T+2  0.0003799 0.02289
T+3  0.0003799 0.02289
T+4  0.0003799 0.02290
T+5  0.0003799 0.02290
T+6  0.0003799 0.02290
T+7  0.0003799 0.02290
T+8  0.0003799 0.02290
T+9  0.0003799 0.02291
T+10 0.0003799 0.02291
T+11 0.0003799 0.02291
T+12 0.0003799 0.02291
T+13 0.0003799 0.02291
T+14 0.0003799 0.02292
T+15 0.0003799 0.02292
T+16 0.0003799 0.02292
T+17 0.0003799 0.02292
T+18 0.0003799 0.02292
T+19 0.0003799 0.02293
T+20 0.0003799 0.02293
```



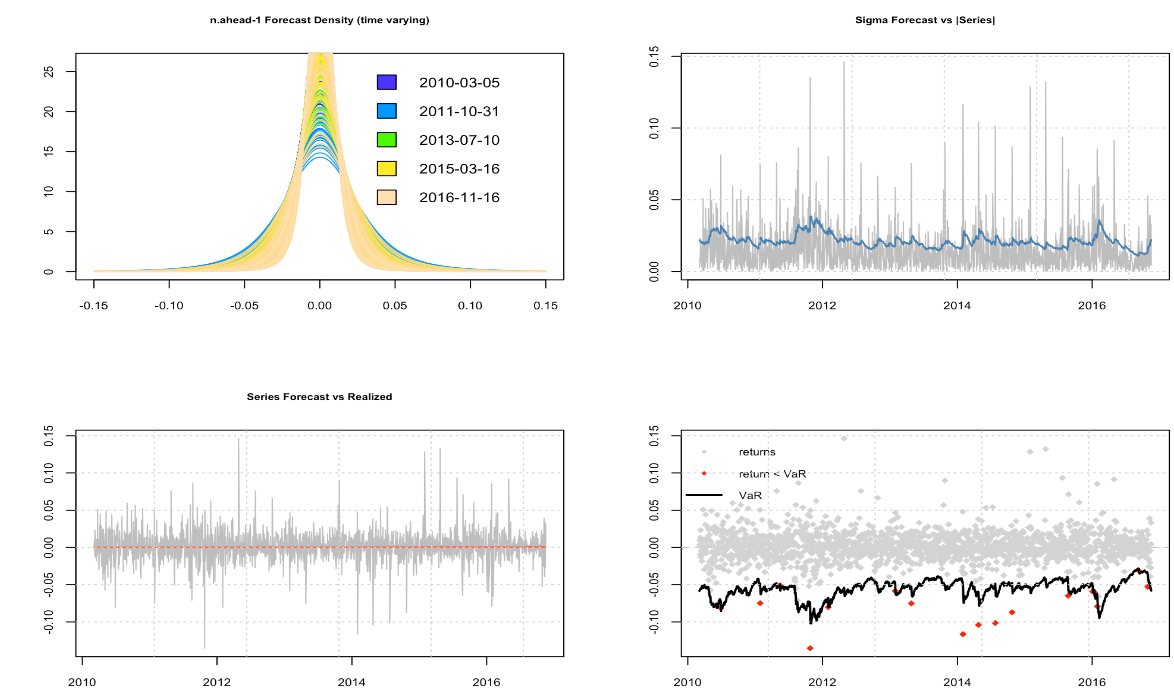
Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

```
> ## Backtesting method to compare EGARCH and GARCH models:
> mod_egarch = ugarchroll(egarch11.t.spec, data = rets, n.ahead = 1,
+                         n.start = 1300, refit.every = 200, refit.window = "recursive")
>
> mod_garch = ugarchroll(garch11.t.spec, data = rets, n.ahead = 1,
+                       n.start = 1300, refit.every = 200, refit.window = "recursive")
> report(mod_egarch, type="fpm")

GARCH Roll Mean Forecast Performance Measures
---------------------------------------------
Model         : eGARCH
No.Refits     : 9
No.Forecasts: 1690

        Stats
MSE 0.0004193
MAE 0.0142200
DAC 0.5136000


> report(mod_garch, type="fpm")

GARCH Roll Mean Forecast Performance Measures
---------------------------------------------
Model         : sGARCH
No.Refits     : 9
No.Forecasts: 1690

        Stats
MSE 0.0004192
MAE 0.0142200
DAC 0.5136000
```

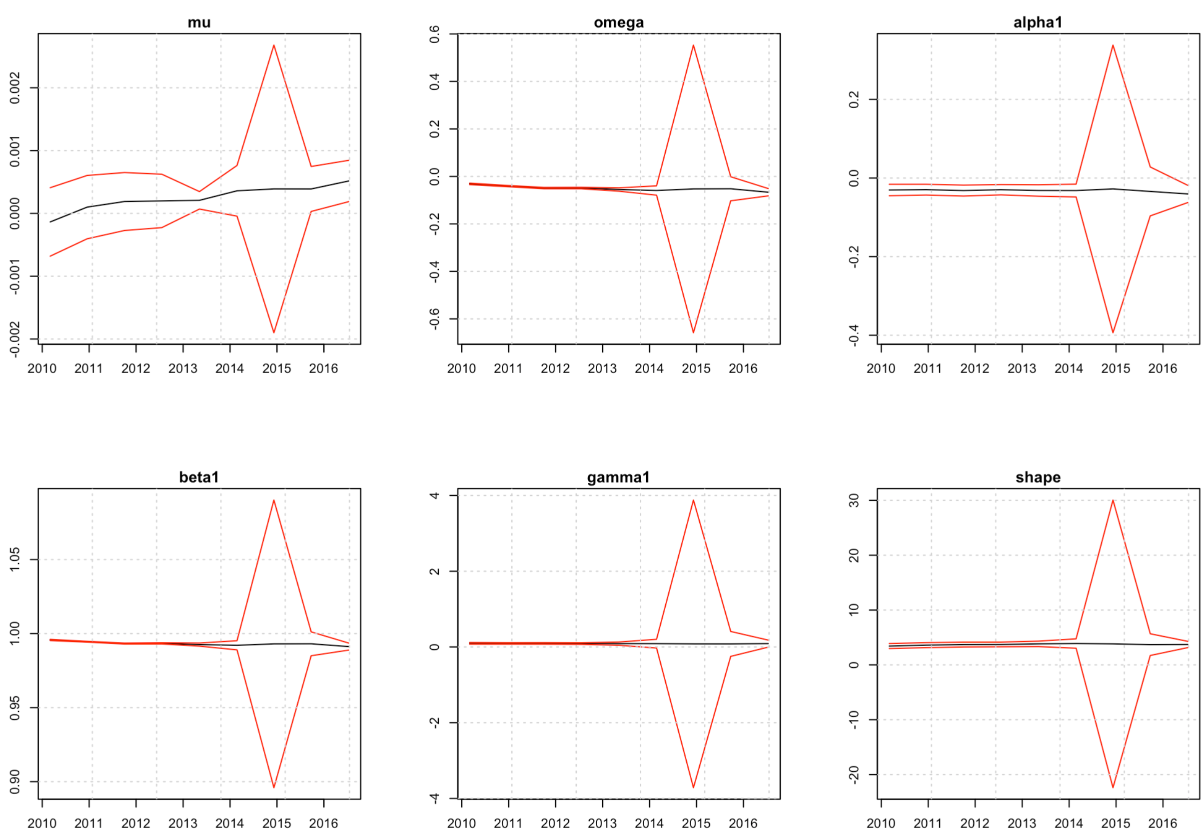
```
> #type=VaR shows VaR at 1% level: this is the tail probablity.
> report(mod_egarch, type = "VaR", VaR.alpha =0.01, conf.level = 0.95)
VaR Backtest Report
===========================================
Model:                          eGARCH-std
Backtest Length:        1690
Data:

==========================================
alpha:                          1%
Expected Exceed:        16.9
Actual VaR Exceed:      17
Actual %:                       1%

Unconditional Coverage (Kupiec)
Null-Hypothesis:        Correct Exceedances
LR.uc Statistic:        0.001
LR.uc Critical:         3.841
LR.uc p-value:          0.981
Reject Null:            NO

Conditional Coverage (Christoffersen)
Null-Hypothesis:        Correct Exceedances and
                                        Independence of Failures
LR.cc Statistic:        NaN
LR.cc Critical:         5.991
LR.cc p-value:          NaN
Reject Null:            NA
>
```

Analysis of Amazon's daily stock returns (2005-2016)

By: Brunda Chouthoy



eGARCH fit coefficients (across 9 refits) with robust s.e. bands



Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

R CODE:

```
###################################################################
#Name: Brunda Chouthoy
#CSC 425 Time Series Analysis and forecasting
#Amazon.com, Inc. (AMZN) Historical Price data
#Analysis of Amazon daily stock returns
#Time Period: 2005-2016(Present)
#Depaul ID: 1804455
###################################################################

# Load the libraries
library(rgl)
library(rugarch)
library(tseries)
library(fBasics)
library(zoo)
library(forecast)
library(fBasics)

# import data in R and compute log returns
# import libraries for TS analysis
myd= read.table('amazon_2005_16.csv', header=T, sep=',')
head(myd)
#Create a timeseries object with the zoo function
amazon.ts = zoo(myd$Close, as.Date(as.character(myd$Date),
format=c("%m/%d/%y")))
#log return time series
rets = log(amazon.ts/lag(amazon.ts, -1))
# strip off the dates and just create a simple numeric object
ret = coredata(rets);

###################################################################
##EXPLORATORY ANALYSIS OF THE DATA
###################################################################
#compute statistics
basicStats(rets)
hist(rets, xlab="Stock returns", prob=TRUE, main="Histogram: Amazon stock
returns data(2005-16)")
##add approximating normal density curve
xfit<-seq(min(rets),max(rets),length=40)
yfit<-dnorm(xfit,mean=mean(rets),sd=sd(rets))
lines(xfit, yfit, col="blue", lwd=2)

##and a normal quantile plot.
qqnorm(rets)
qqline(rets, col = 2)

# creates time plot of log returns
plot(rets, ylab="Log of stock returns", xlab='Year', main="Time plot:Amazon
stock returns data(2005-16)")
#plot returns, square returns and abs(returns)
par(mfrow=c(1,1))
plot(rets, ylab="Log of stock returns", xlab='Year', main="Time plot:Amazon
stock returns data(2005-16)")
plot(rets^2,type='l')
plot(abs(rets),type='l')
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

```
par(mfrow=c(1,1))
# Plots ACF function of vector data
acf(ret)
# Plot ACF of squared returns to check for ARCH effect
acf(ret^2)
# Plot ACF of absolute returns to check for ARCH effect
acf(abs(ret))

par(mfrow=c(1,1))
# Computes Ljung-Box test on returns to test  independence
Box.test(coredata(rets),lag=6,type='Ljung')
Box.test(coredata(rets),lag=9,type='Ljung')
Box.test(coredata(rets),lag=12,type='Ljung')
# Computes Ljung-Box test on squared returns to test non-linear independence
Box.test(coredata(rets^2),lag=6,type='Ljung')
Box.test(coredata(rets^2),lag=12,type='Ljung')
# Computes Ljung-Box test on absolute returns to test non-linear independence
Box.test(abs(coredata(rets)),lag=6,type='Ljung')
Box.test(abs(coredata(rets)),lag=12,type='Ljung')

# plots PACF of squared returns to identify order of AR model
pacf(coredata(rets),lag=20)

##################################################################
##MODEL FITTING, RESIDUAL ANALYSIS AND MODEL DIAGNOSTICS
##################################################################
#specify model using functions in rugarch package
#Fit ARMA(0,0)-GARCH(1,1) model with Normal distribution
garch11.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)),
mean.model=list(armaOrder=c(0,0)))
#estimate model
garch11.fit=ugarchfit(spec=garch11.spec, data=rets)
garch11.fit
#plot of residuals
plot(garch11.fit)

#Fit ARMA(0,0)-GARCH(1,1) model with t-distribution
garch11.t.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)),
mean.model=list(armaOrder=c(0,0)), distribution.model = "std")
#estimate model
garch11.t.fit=ugarchfit(spec=garch11.t.spec, data=rets)
garch11.t.fit
#plot of residuals
plot(garch11.t.fit)

#using extractors
#estimated coefficients:
coef(garch11.fit)
#unconditional mean in mean equation
uncmean(garch11.fit)
#unconditional varaince: omega/(alpha1+beta1)
uncvariance(garch11.fit)
#persistence = alpha1+beta1
persistence(garch11.fit)
#half-life: ln(0.5)/ln(alpha1+beta1)
halflife(garch11.fit)
```

Analysis of Amazon's daily stock returns (2005-2016)
By: Brunda Chouthoy

```
#Fit ARMA(0,0)-GARCH(1,1) model with skewed t-distribution
garch11.skt.spec=ugarchspec(variance.model=list(garchOrder=c(1,1)),
mean.model=list(armaOrder=c(0,0)), distribution.model = "sstd")
#estimate model
garch11.skt.fit=ugarchfit(spec=garch11.skt.spec, data=rets)
garch11.skt.fit
persistence(garch11.skt.fit)

#Fit ARMA(0,0)-eGARCH(1,1) model with  Gaussian distribution
egarch11.spec=ugarchspec(variance.model=list(model = "eGARCH",
garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)))
#estimate model
egarch11.fit=ugarchfit(spec=egarch11.spec, data=ret)
egarch11.fit

#Fit ARMA(0,0)-eGARCH(1,1) model with t-distribution
egarch11.t.spec=ugarchspec(variance.model=list(model = "eGARCH",
garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)), distribution.model =
"std")
#estimate model
egarch11.t.fit=ugarchfit(spec=egarch11.t.spec, data=rets)
egarch11.t.fit
persistence(egarch11.t.fit)
#plot of residuals
plot(egarch11.t.fit)

#FORECASTS for egarch.t.fit
#compute h-step ahead forecasts for h=1,2,...,10
egarch11.t.fcst=ugarchforecast(egarch11.t.fit, n.ahead=12)
egarch11.t.fcst
plot(egarch11.t.fcst)

#Fit ARMA(0,0)-TGARCH(1,1) model with t-distribution
gjrgarch11.t.spec=ugarchspec(variance.model=list(model = "gjrGARCH",
garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)), distribution.model =
"std")
#estimate model
gjrgarch11.t.fit=ugarchfit(spec=gjrgarch11.t.spec, data=ret)
gjrgarch11.t.fit

#Fit ARMA(0,0)-gjrGARCH(1,1) model with skewed t-distribution
gjrgarch11.t.spec=ugarchspec(variance.model=list(model = "gjrGARCH",
garchOrder=c(1,1)), mean.model=list(armaOrder=c(0,0)), distribution.model =
"sstd")
#estimate model
gjrgarch11.t.fit=ugarchfit(spec=gjrgarch11.t.spec, data=ret)
gjrgarch11.t.fit

# MODEL COMPARISON
# compare information criteria
model.list = list(garch11 = garch11.fit, garch11.t = garch11.t.fit,
                  egarch11 = egarch11.t.fit,
                  gjrgarch11 = gjrgarch11.t.fit)

info.mat = sapply(model.list, infocriteria)
rownames(info.mat) = rownames(infocriteria(garch11.fit))
```

```
info.mat

####################################################################
##FORECAST ANALYSIS AND BACKTESTING
####################################################################
# re-fit models leaving 300 out-of-sample observations for forecast
# evaluation statistics
egarch11.t.fit = ugarchfit(egarch11.t.spec, data=ret, out.sample=300)

egarch11.t.fcst = ugarchforecast(egarch11.t.fit, n.roll=300, n.ahead=20)
egarch11.t.fcst
plot(egarch11.t.fcst)

# compute forecast evaluation statistics using fpm method
# type="fpm" shows forecast performance measures
# (Mean Squared Error (MSE), mean absolute error(MAE) and directional
accuracy
# of the forecasts vs realized returns(DAC)).
fcst.list = list(garch11.t=garch11.t.fcst,
                 egarch11.t=egarch11.t.fcst)
fpm.mat = sapply(fcst.list, fpm)
fpm.mat

#to visualize results use plot
#plot(garch11.t.fcst)
#plot(egarch11.t.fcst)

## Backtesting method to compare EGARCH and GARCH models:
mod_egarch = ugarchroll(egarch11.t.spec, data = rets, n.ahead = 1,
                        n.start = 1300, refit.every = 200, refit.window =
"recursive")

mod_garch = ugarchroll(garch11.t.spec, data = rets, n.ahead = 1,
                      n.start = 1300, refit.every = 200, refit.window =
"recursive")
report(mod_egarch, type="fpm")
report(mod_garch, type="fpm")

#type=VaR shows VaR at 1% level: this is the tail probablity.
report(mod_egarch, type = "VaR", VaR.alpha =0.01, conf.level = 0.95)

#to visualize results use plot
plot(mod_egarch)
plot(mod_garch)
####################################################################
##R CODE ENDS HERE
####################################################################
```