Financial Time Series Analysis

Submitted to: Dr. Ginger Holt

Haritha Peyyeti, Huong Huynh, Vaibhav Kishore

5/15/15

Math 265

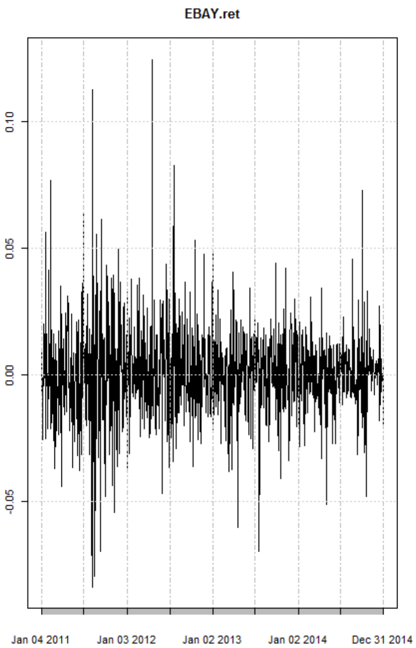
1. Introduction

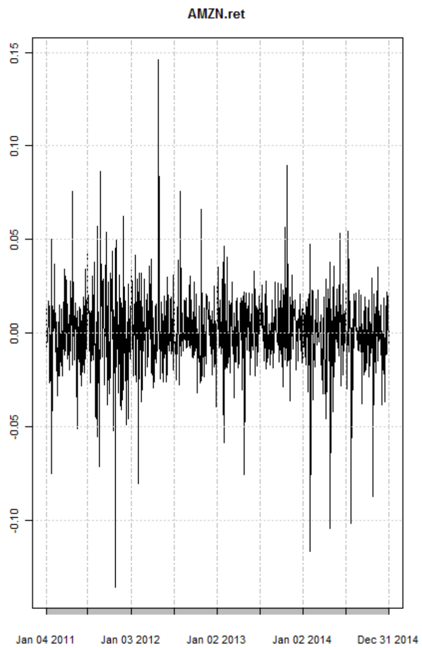
Stock price forecasting has always attracted interest because of the direct financial benefit. From our literature study, we were interested in examining the relationship between the stock prices of competitors in the e-commerce sector. In this project, we have chosen 3 companies – Ebay, Amazon and Walmart to examine if the return and volatility behavior of one competitor influence the others.

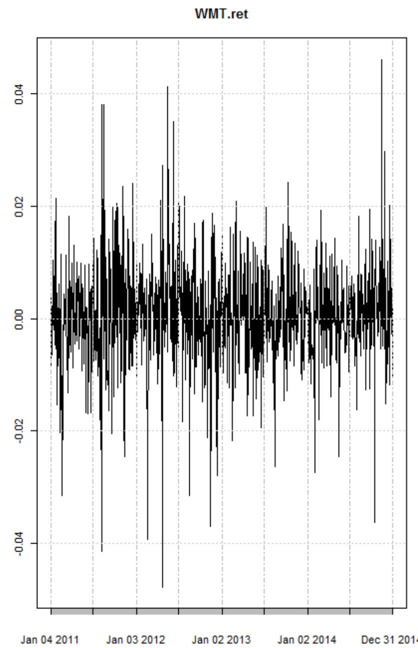
Stock Price of one company influences the price movements of its competitors and related companies in other sectors. Due to these inter-dependencies, meaningful insights can be drawn by analyzing them jointly to better understand the dynamic structure of the global finance. One company may lead the other company under certain time periods, yet the relationship may be reversed under other circumstances. Consequently, knowing how the markets are interrelated is of great importance in finance. A system that can identify which companies are doing well and which companies are not in the dynamic stock market will make it easy for investors or market or finance professionals to make decisions. Having an excellent knowledge about share price movement in the future helps the investors and finances personals significantly. Since, it is necessary to identify a model to analyze trends of stock prices with relevant information for decision making.

In the current project, we have considered the daily stock prices for the following companies: Ebay (EBAY), Amazon (AMZN), and Walmart (WMT) in the date range January 1, 2011 – December 31, 2014. In the first section, we examined the basic statistics of each of the return series. In the next section, we used the Box-Jenkins methodology to perform univariate time series model fitting to each of the series and forecast the daily returns for the first month in 2015. We then modeled the volatility of each return series by fitting an appropriate ARMA-GARCH model. Finally, we looked at the cross correlation between the three return series and compared the models.

Download daily stock prices for the following companies: Ebay (EBAY), Amazon (AMZN), and Walmart (WMT) and compute the log returns using the date range January 1, 2011 – December 31, 2014.







(a) Compute the sample mean, standard deviation, skewness, excess kurtosis, minimum, and maximum of the log returns for each series.

Basic Statistics	EBAY	AMZN	WMT
nobs	1005	1005	1005
NAs	0	0	0
Minimum	-0.0836	-0.13533	-0.04772
Maximum	0.124359	0.146225	0.046141
1. Quartile	-0.01008	-0.00945	-0.00461
3. Quartile	0.010563	0.012603	0.005649
Mean	0.000668	0.000519	0.000551
Median	0.000356	0.000574	0.000691
Sum	0.671292	0.52157	0.554037
SE Mean	0.000596	0.000649	0.000293
LCL Mean	-0.0005	-0.00076	-2.3E-05
UCL Mean	0.001838	0.001793	0.001125
Variance	0.000357	0.000423	0.000086
Stdev	0.018899	0.020578	0.009274
Skewness	0.444115	-0.32523	-0.22507
Kurtosis	4.577069	7.591271	3.598821

(b) Test the null hypothesis that the mean of each of the series log returns is zero. Also, construct a 95% confidence interval for the daily log returns of each stock.

Statistic	EBAY	AMZN	WMT
t-test	**1.120400**	**0.799500**	**1.884500**
P-value	**0.262800**	**0.424200**	**0.059780**
Lower confidence limits (95%)	**-0.000502**	**-0.000755**	**-0.000023**
Mean	**0.000668**	**0.000519**	**0.000551**
Upper confidence limits (95%)	**0.001838**	**0.001793**	**0.001125**

Here, the log returns of all the three stocks have a p-value of greater than 0.05. So, at 5% significance level we fail to reject the null hypothesis that the mean of each of the series log returns is zero.

(c) Test Ho: m3 = 0 vs. Ha: m3 != 0, where m3 denotes the skewness of the log return.

Skewness	EBAY	AMZN	WMT
Test Statistic	**5.756407**	**4.215475**	**2.917199**
P-value	**0.000000**	**0.000025**	**0.003532**

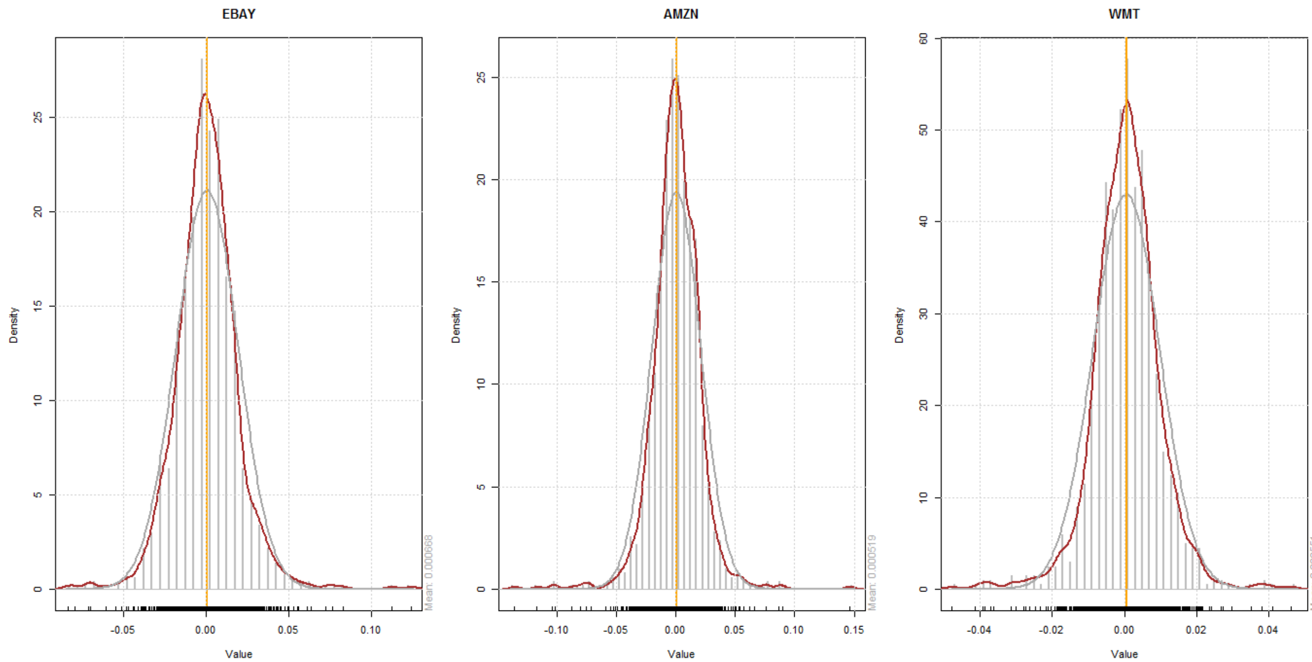
Here, the log returns of all the three stocks have a p-value of less than 0.05. So, at 5% significance level we reject the null hypothesis that the skewness of each of the series log returns is zero. Therefore, the distributions are not symmetric about the mean but skewed.

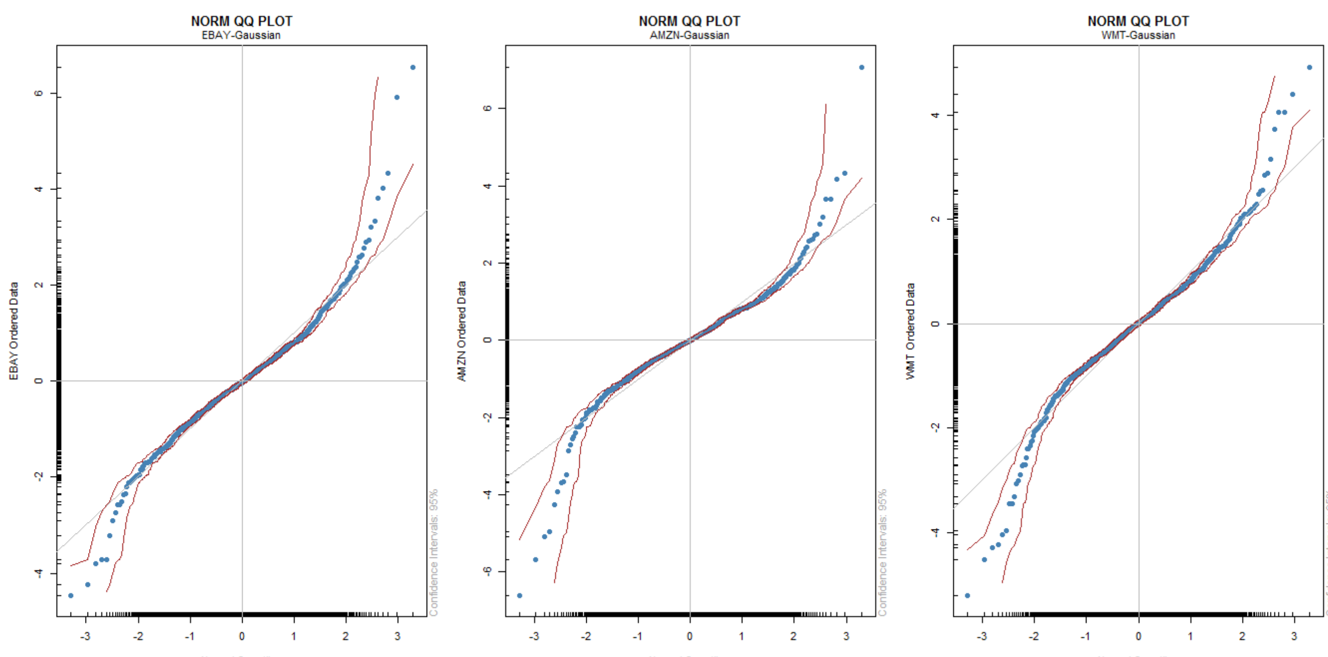
(d) Test Ho: K = 3 vs. Ha: K != 3, where K denotes the kurtosis.

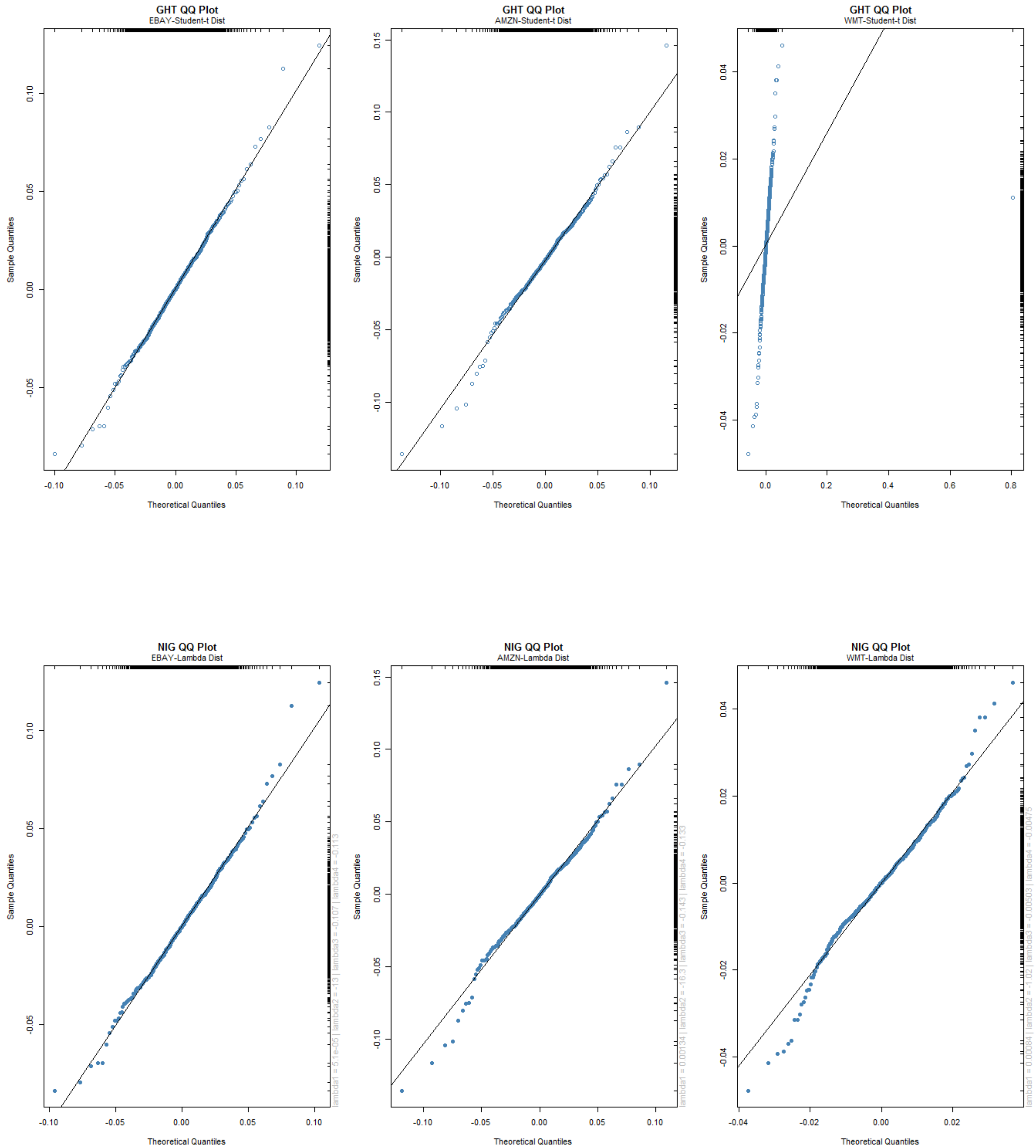
Excess Kurtosis	EBAY	AMZN	WMT
Test Statistic	**29.7163**	**49.2604**	**23.3734**
P-value	**0**	**0**	**0**

Here, the log returns of all the three stocks have large test statistic values and p-values of less than 0.05. So, at 5% significance level we reject the null hypothesis that the kurtosis of each of the series log returns is three. Thus, the distributions of these stocks have heavy tails.

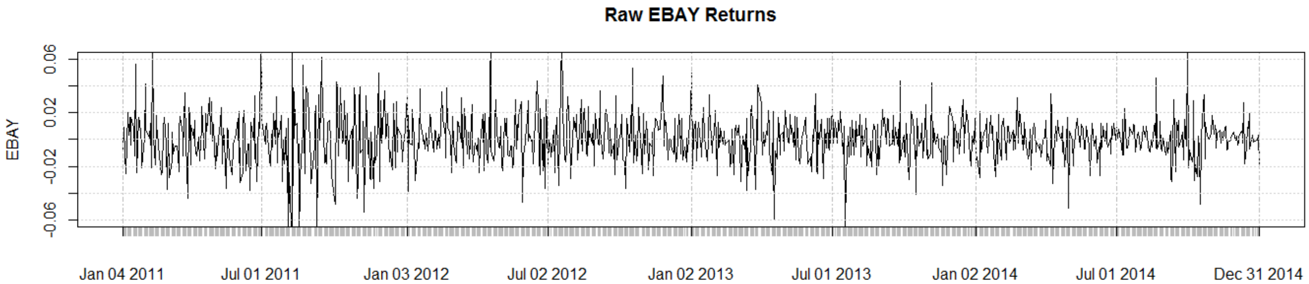
(e) Obtain the empirical density plot of the daily log returns of each series, and select an appropriate distribution (Gaussian, t, etc.).

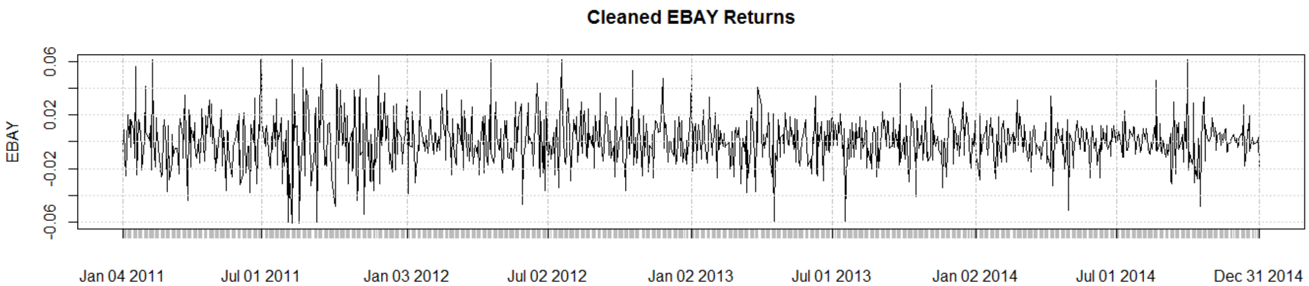


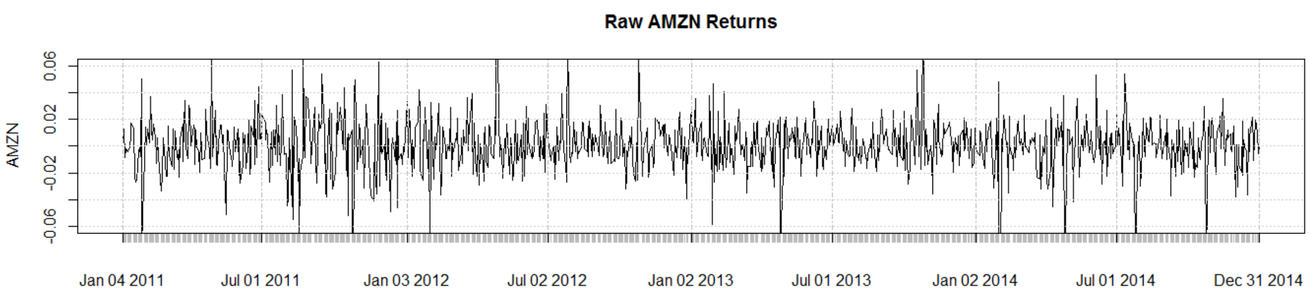


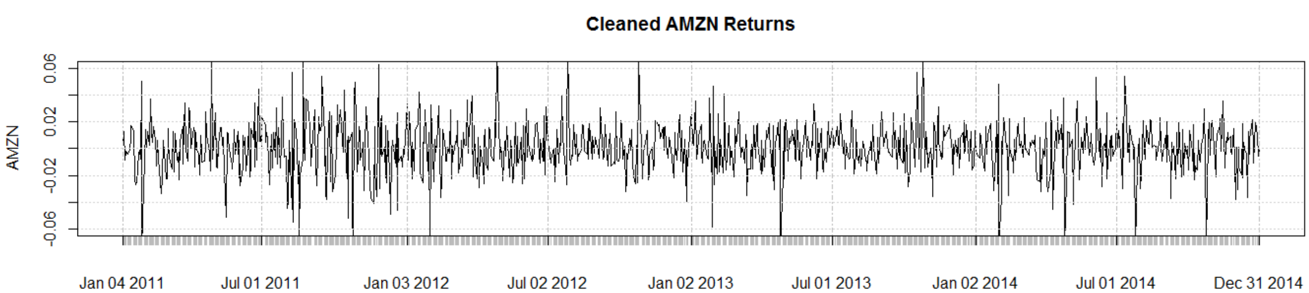


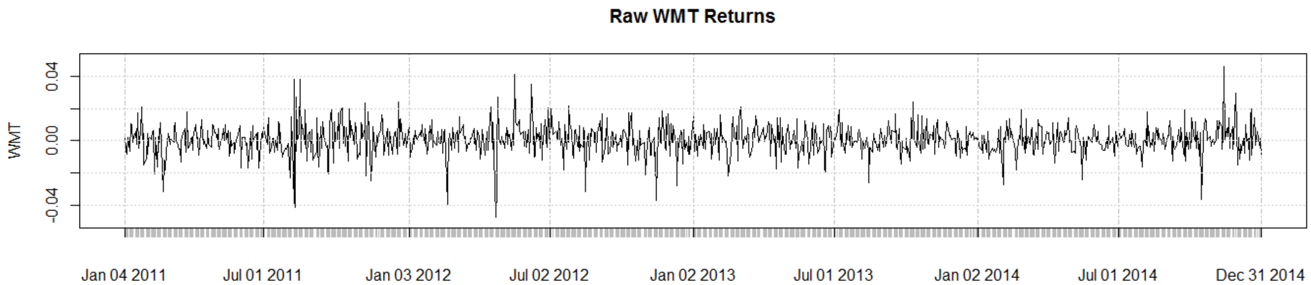
Looking at the qq-plots we can see that the Student-t distribution appears to provide the best distribution fit for Ebay and Amazon and the general lambda distribution appears to provide the best distribution fit for Walmart.

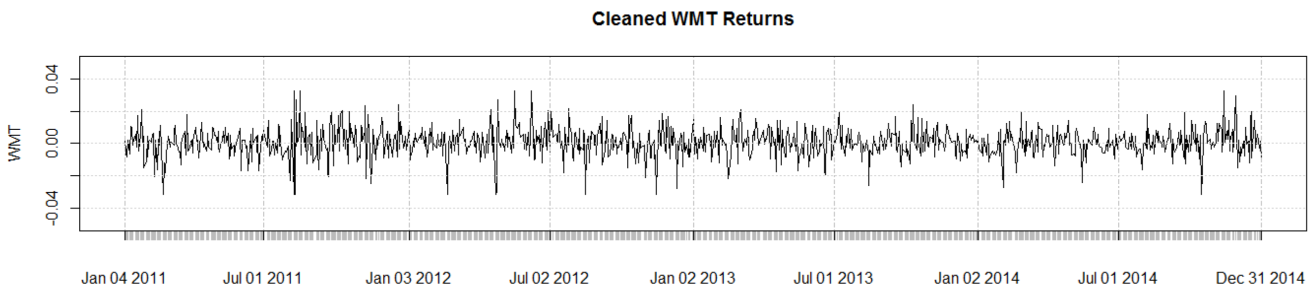

Raw EBAY Returns
EBAY
0.06
0.02
-0.02
-0.06
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014


Cleaned EBAY Returns
EBAY
0.06
0.02
-0.02
-0.06
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014


Raw AMZN Returns
AMZN
0.06
0.02
-0.02
-0.06
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014


Cleaned AMZN Returns
AMZN
0.06
0.02
-0.02
-0.06
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014

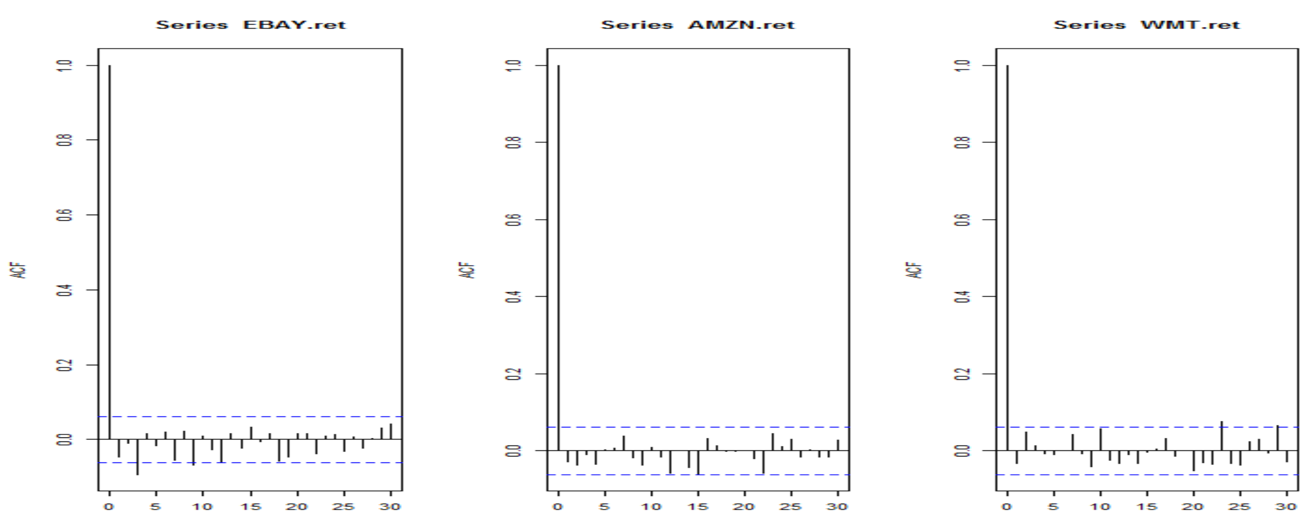


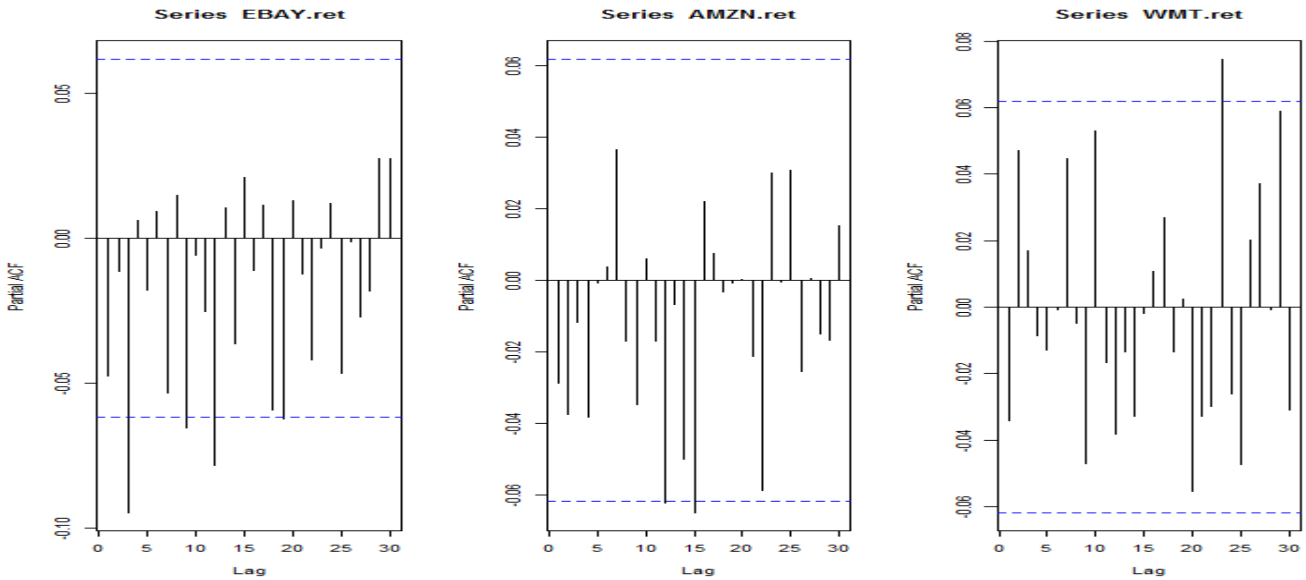


We used data cleaning method to get rid of any outliers. But from the above plots, the raw log returns and the cleaned log returns for all the three stocks look very similar. Furthermore, it does not have any significant impact on further analysis.

(f) Use the Box-Jenkins methodology to perform univariate time series model fitting to each of the series. Include details of each step of the process, and support your final model selection for each series.

Step 1: Model Identification





The above plot suggest ARMA(3,3) model for EBay. While for Amazon and Walmart, we observe that ACF and PACF are marginally significant only at higher lags.

Model Estimation:

EBAY: ARIMA(3,0,3) with non-zero mean

Coefficients:

	ar1	ar2	ar3	ma1	ma2	ma3	intercept
	0.019	-0.169	0.846	-0.062	0.141	-0.932	0.001
s.e.	0.048	0.038	0.039	0.034	0.026	0.026	0.000

AMZN: ARIMA(1,0,1) with non-zero mean

Coefficients:

	ar1	ma1	intercept
	0.929	-0.961	0.001
s.e.	0.069	0.056	0.000

WMT: ARIMA(0,0,0) with non-zero mean

Coefficients:

	intercept
	0.001
s.e.	0.000

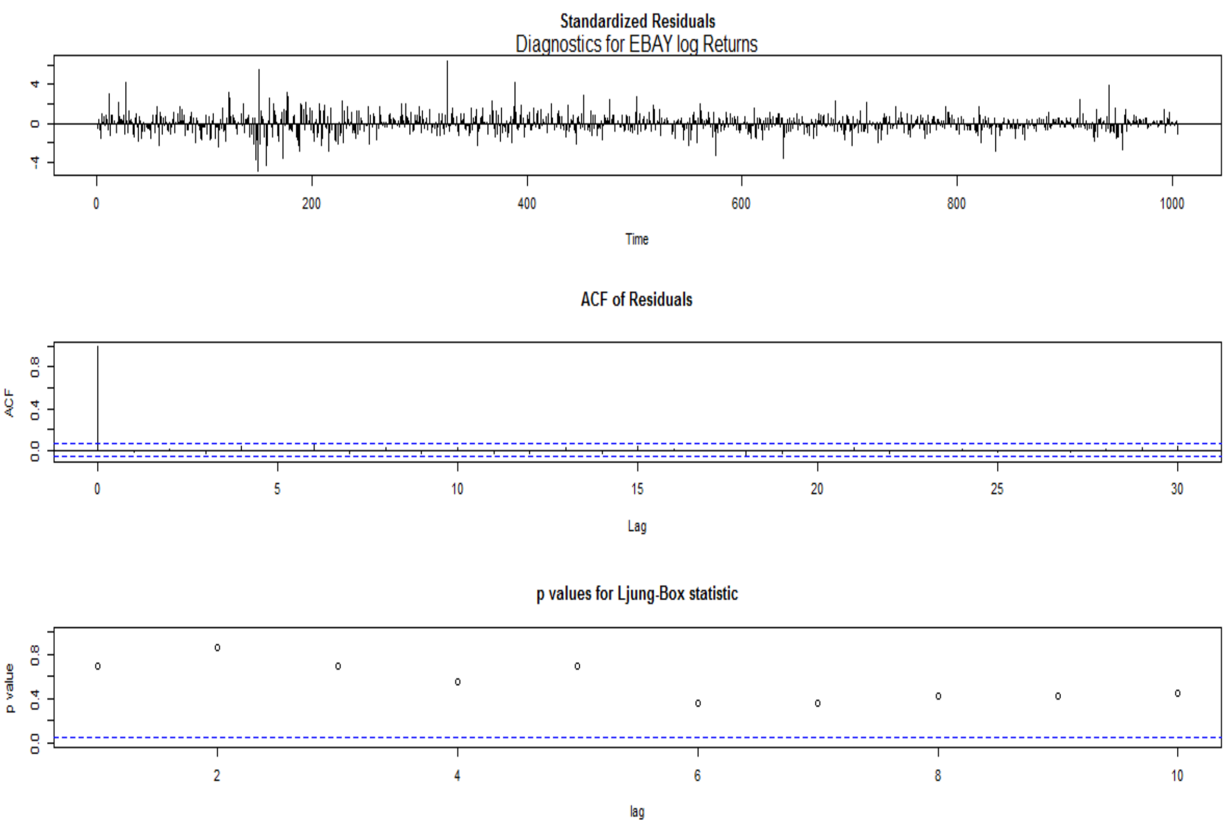
Using the AIC criteria to select the model, ARMA(3,3) best fits Ebay. While, AMZN has ARMA(1,1) as the suggested model and Walmart has the means model.

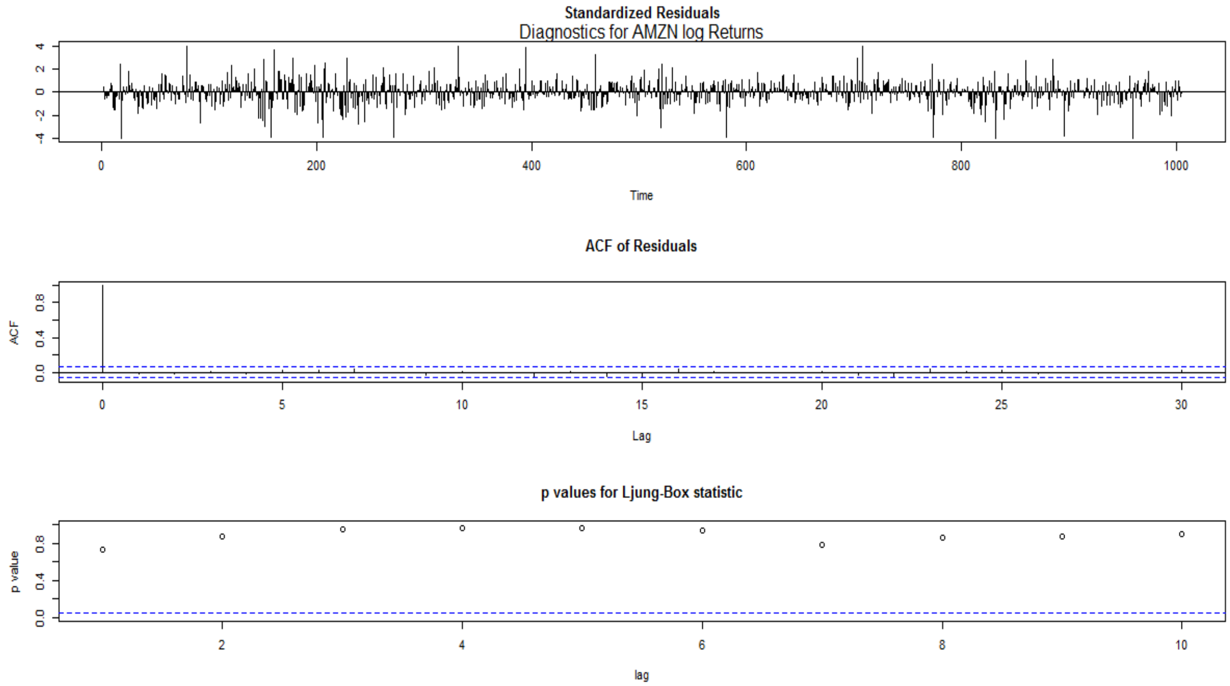
Ebay: $r_t = 0.001 + 0.019r_{t-1} - 0.169r_t\text{-}2 + 0.846r_{t-3} + a_t + 0.062a_{t-1} - 0.141a_{t-2} + 0.932a_{t-3}$

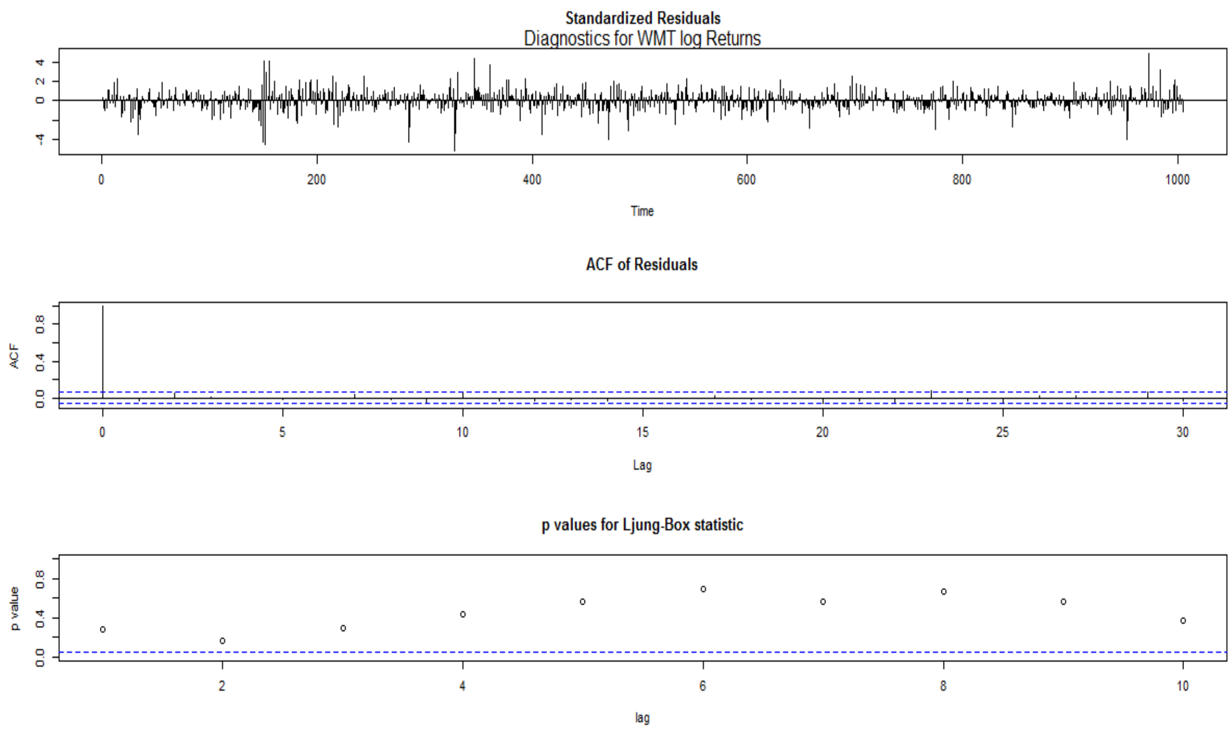
Amazon: $r_t = 0.001 + 0.929r_{t-1} + a_t + 0.961a_{t-1}$

Walmart: $r_t = 0.001$

Step 3: Model Verification:




Standardized Residuals
Diagnostics for AMZN log Returns
Time
ACF of Residuals
ACF
Lag
p values for Ljung-Box statistic
p value
lag


Standardized Residuals
Diagnostics for WMT log Returns
Time
ACF of Residuals
ACF
Lag
p values for Ljung-Box statistic
p value
lag

From the above diagnostic plots, we conclude that the model seems to be adequate from the above residuals plot (ACFs within the limit and Ljung box statistics with high p-val indicating that the residuals are uncorrelated).

EBAY: sigma^2 estimated as 0.0003477762: log likelihood=2575.5

AIC=-5135   AICc=-5134.85   BIC=-5095.7

AMZN: sigma^2 estimated as 0.0003477762: log likelihood=2575.5

AIC=-5135   AICc=-5134.85   BIC=-5095.7

WMT: sigma^2 estimated as 8.591659e-05: log likelihood=3278.44

AIC=-6552.88   AICc=-6552.87   BIC=-6543.05

Training Set	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
EBAY	4.37063E-05	0.019	0.013457119	NaN	Inf	0.673796036	-0.012425584
AMZN	4.37063E-05	0.018648759	0.013457119	NaN	Inf	0.673796036	-0.012425584
WMT	-1.03886E-20	0.00926912	0.006693019	-Inf	Inf	0.682054649	-0.03430679

Box-Ljung tests

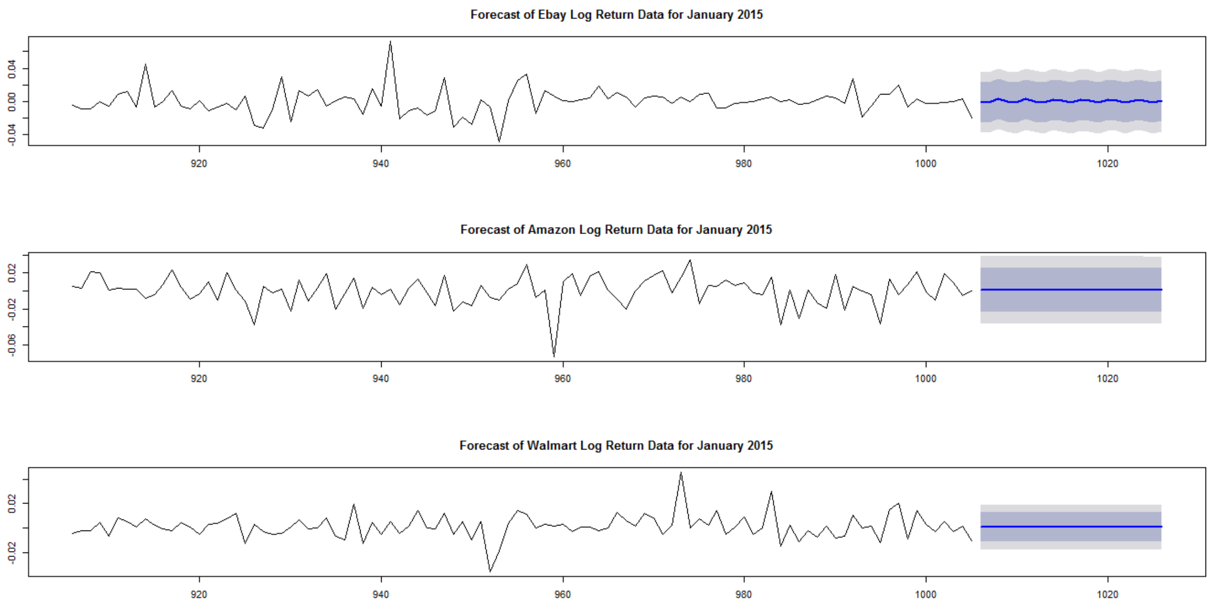
Ebay: X-squared = 20.1814, df = 24, p-value = 0.6864

Amazon:X-squared = 23.6909, df = 24, p-value = 0.4794

Amazon (Cleaned):X-squared = 16.9359, df = 24, p-value = 0.8514

Walmart: X-squared = 27.1645, df = 24, p-value = 0.2968

(g) Using the model you selected in part f), compute forecasts for the daily returns for the first month in 2015 as well as 95% confidence intervals for the forecast.

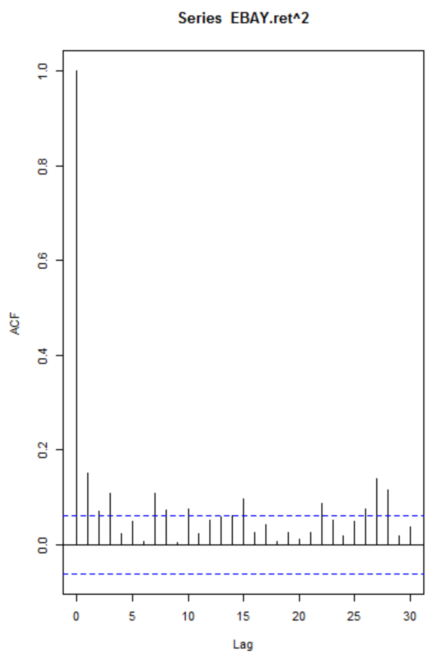


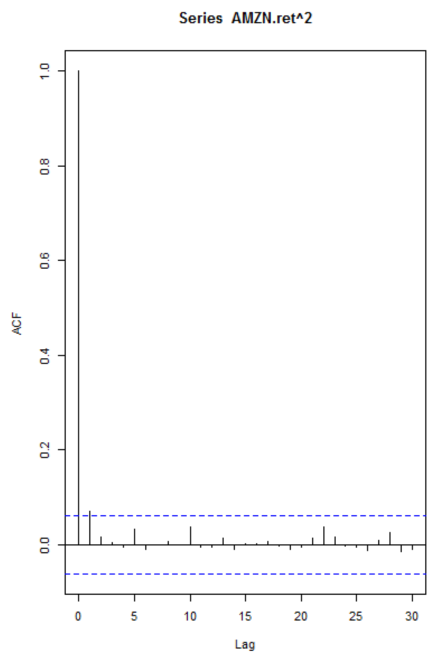
EBAY	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
1006	-0.0005	-0.0244	0.0234	-0.0371	0.0360
1007	-0.0010	-0.0249	0.0229	-0.0376	0.0356
1008	0.0032	-0.0207	0.0271	-0.0334	0.0398
1009	0.0000	-0.0240	0.0240	-0.0367	0.0367
1010	-0.0012	-0.0252	0.0228	-0.0379	0.0356
1011	0.0029	-0.0211	0.0269	-0.0339	0.0396
1012	0.0004	-0.0236	0.0245	-0.0364	0.0372
1013	-0.0013	-0.0254	0.0228	-0.0381	0.0355
1014	0.0025	-0.0215	0.0266	-0.0343	0.0394
1015	0.0008	-0.0233	0.0249	-0.0360	0.0377
1016	-0.0013	-0.0254	0.0228	-0.0382	0.0356
1017	0.0022	-0.0219	0.0263	-0.0347	0.0390
1018	0.0012	-0.0230	0.0253	-0.0357	0.0381
1019	-0.0012	-0.0254	0.0229	-0.0382	0.0357
1020	0.0018	-0.0223	0.0259	-0.0351	0.0387
1021	0.0014	-0.0227	0.0256	-0.0355	0.0383
1022	-0.0011	-0.0253	0.0230	-0.0381	0.0358
1023	0.0015	-0.0227	0.0256	-0.0355	0.0384
1024	0.0016	-0.0225	0.0258	-0.0353	0.0386
1025	-0.0010	-0.0251	0.0232	-0.0379	0.0360
1026	0.0011	-0.0230	0.0253	-0.0358	0.0381

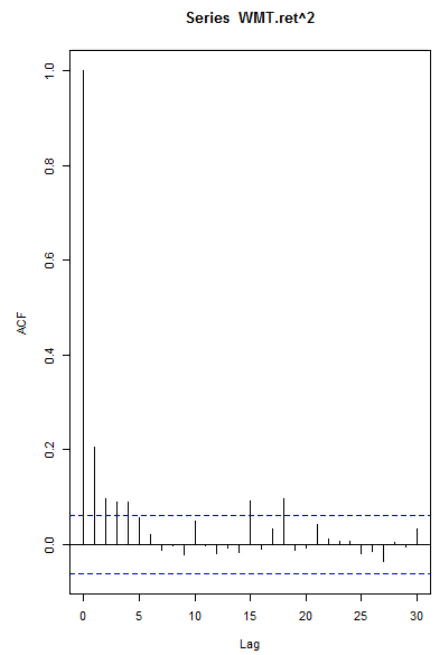
Amazon	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
1006	0.0012	-0.0233	0.0258	-0.0363	0.0388
1007	0.0012	-0.0234	0.0257	-0.0364	0.0387
1008	0.0011	-0.0234	0.0257	-0.0365	0.0387
1009	0.0011	-0.0235	0.0257	-0.0365	0.0387
1010	0.0011	-0.0235	0.0257	-0.0365	0.0387
1011	0.0010	-0.0236	0.0256	-0.0366	0.0387
1012	0.0010	-0.0236	0.0256	-0.0366	0.0386
1013	0.0010	-0.0236	0.0256	-0.0367	0.0386
1014	0.0009	-0.0237	0.0256	-0.0367	0.0386
1015	0.0009	-0.0237	0.0255	-0.0367	0.0386
1016	0.0009	-0.0237	0.0255	-0.0368	0.0386
1017	0.0009	-0.0237	0.0255	-0.0368	0.0385
1018	0.0009	-0.0238	0.0255	-0.0368	0.0385
1019	0.0008	-0.0238	0.0255	-0.0368	0.0385
1020	0.0008	-0.0238	0.0255	-0.0368	0.0385
1021	0.0008	-0.0238	0.0254	-0.0369	0.0385
1022	0.0008	-0.0238	0.0254	-0.0369	0.0385
1023	0.0008	-0.0238	0.0254	-0.0369	0.0385
1024	0.0008	-0.0239	0.0254	-0.0369	0.0385
1025	0.0008	-0.0239	0.0254	-0.0369	0.0384
1026	0.0008	-0.0239	0.0254	-0.0369	0.0384

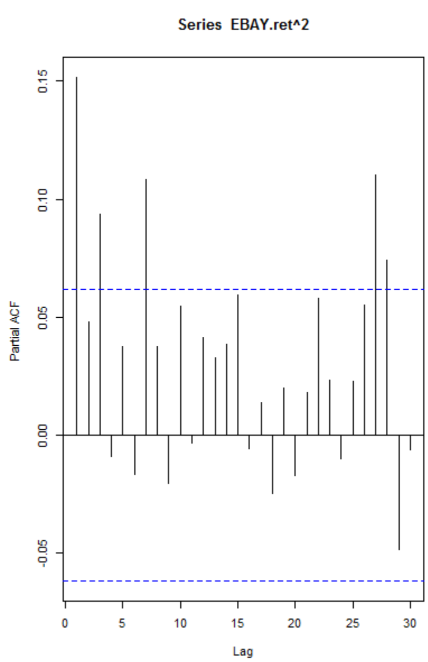
Walmart	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
1006	0.0006	-0.0113	0.0124	-0.0176	0.0187
1007	0.0006	-0.0113	0.0124	-0.0176	0.0187
1008	0.0006	-0.0113	0.0124	-0.0176	0.0187
1009	0.0006	-0.0113	0.0124	-0.0176	0.0187
1010	0.0006	-0.0113	0.0124	-0.0176	0.0187
1011	0.0006	-0.0113	0.0124	-0.0176	0.0187
1012	0.0006	-0.0113	0.0124	-0.0176	0.0187
1013	0.0006	-0.0113	0.0124	-0.0176	0.0187
1014	0.0006	-0.0113	0.0124	-0.0176	0.0187
1015	0.0006	-0.0113	0.0124	-0.0176	0.0187
1016	0.0006	-0.0113	0.0124	-0.0176	0.0187
1017	0.0006	-0.0113	0.0124	-0.0176	0.0187
1018	0.0006	-0.0113	0.0124	-0.0176	0.0187
1019	0.0006	-0.0113	0.0124	-0.0176	0.0187
1020	0.0006	-0.0113	0.0124	-0.0176	0.0187
1021	0.0006	-0.0113	0.0124	-0.0176	0.0187
1022	0.0006	-0.0113	0.0124	-0.0176	0.0187
1023	0.0006	-0.0113	0.0124	-0.0176	0.0187
1024	0.0006	-0.0113	0.0124	-0.0176	0.0187
1025	0.0006	-0.0113	0.0124	-0.0176	0.0187
1026	0.0006	-0.0113	0.0124	-0.0176	0.0187

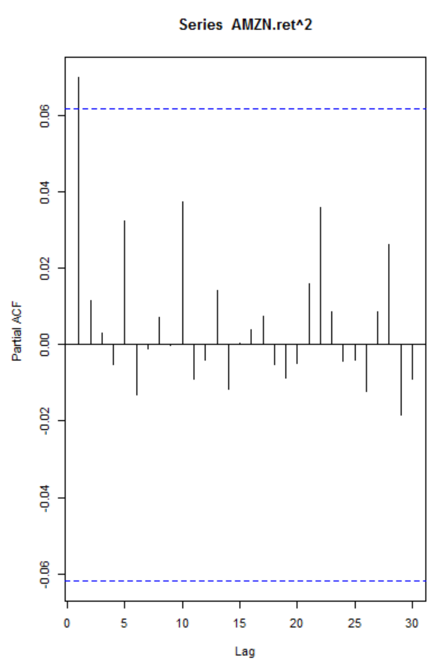
(h) Are there ARCH effect in the log return series? Why or why not?

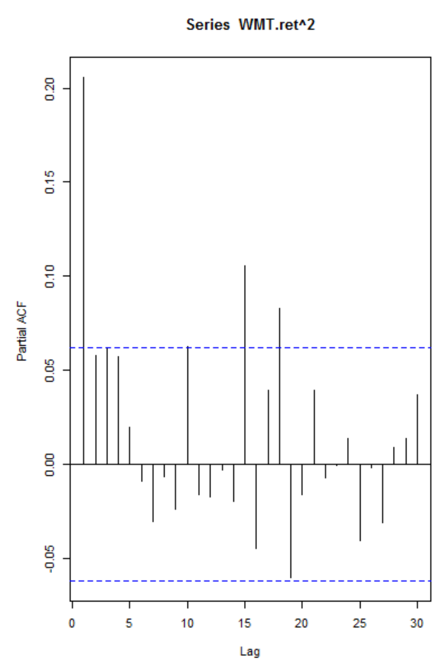


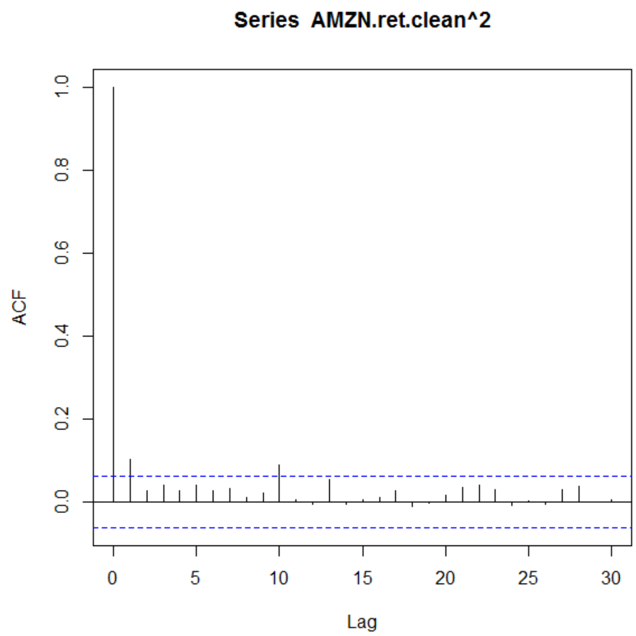


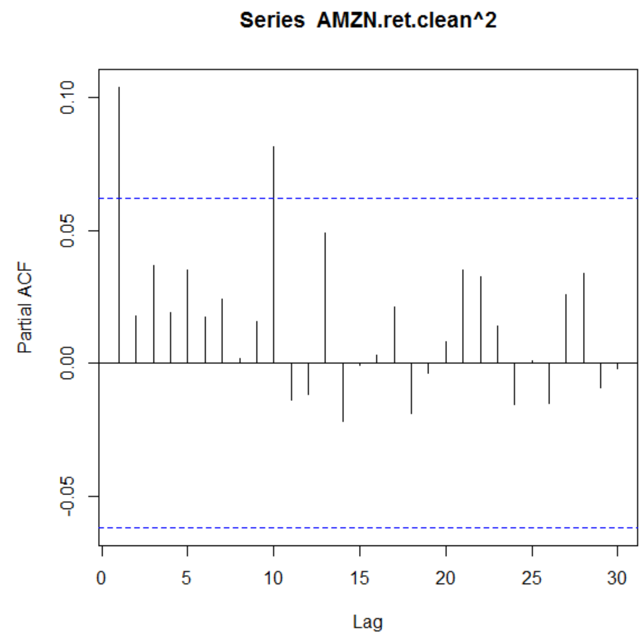










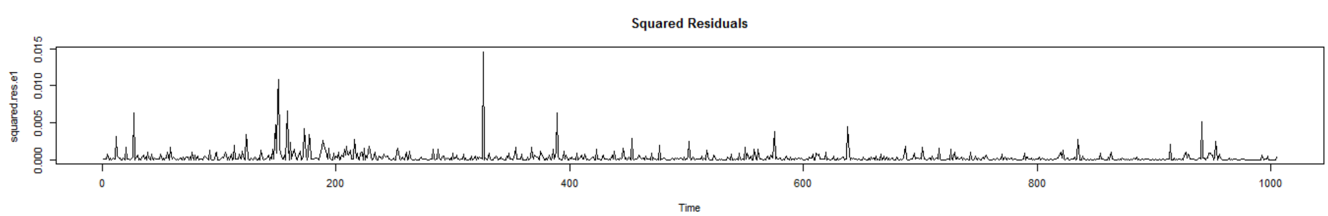


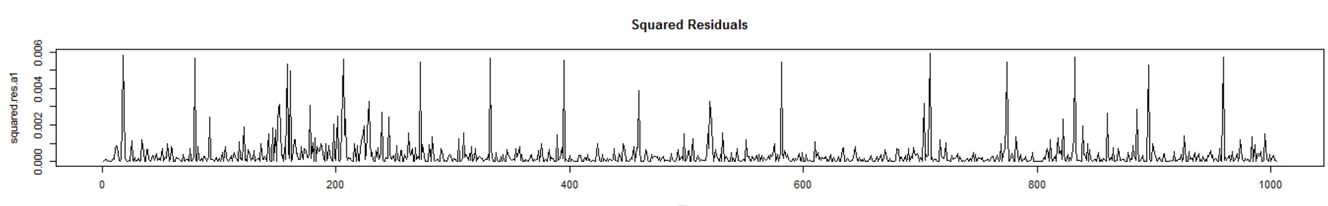
Squared Residuals:

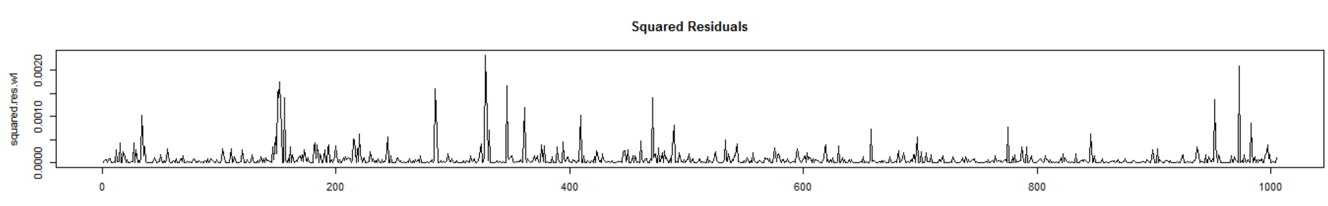
Ebay: X-squared = 97.9194, df = 24, p-value = 6.785e-11

Amazon: X-squared = 33.3083, df = 24, p-value = 0.09773

Walmart: X-squared = 97.1882, df = 24, p-value = 9.025e-11



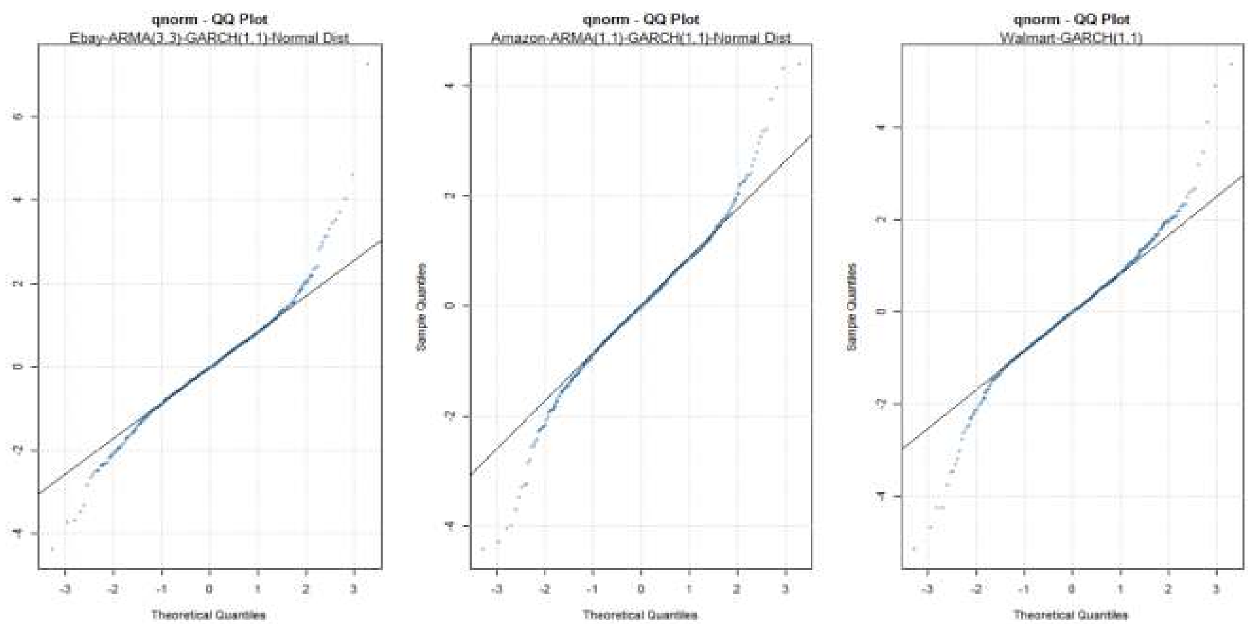




All the three stocks Ebay, Amazon and Walmart indicate the presence of ARCH effects based on the ACF plots, PACF plots. From the Box-Ljung test of autocorrelation in squared residuals, at alpha= 0.1 we see that p-values are significant, indicating the presence of ARCH effect. ARCHLM test was also performed for heteroskedasticity.

(i) Fit a Gaussian ARMA-GARCH model to each of the log return series. Obtain the normal QQ-plot of the standardized residuals, and write down the fitted model. Is the model adequate? Why or why not?

After trying many orders, we see that ARMA(3,3) and GARCH(1,1) is the best model with smallest AIC for log return of Ebay series. For Amazon, it appears that ARMA(1,1) and GARCH(1,1) is the best model with cleaned amazon data set. Since ARMA(1,1) is not significant at alpha = .05 for Walmart series, we refit with only pure GARCH(1,1) model. Below are the normal QQ-plot of the standardized residuals:



The plots show that the model violates the normality assumption since a lot of points at the beginning and at the end are not on the line for all three series. The R-output of the three models are shown below. All coefficients are significant at alpha = .05 except the mu parameter of Ebay and Amazon. All the models are adequate since the p-value in Ljung-Box test are all very large, this means that the residuals are not correlated.

Ebay:

```
Error Analysis:
           Estimate  Std. Error  t value Pr(>|t|)    
mu        2.719e-04   1.770e-04    1.536  0.12464    
ar1       7.739e-02   5.477e-02    1.413  0.15766    
ar2      -2.646e-01   4.579e-02   -5.778 7.58e-09 ***
ar3       8.384e-01   5.356e-02   15.654  < 2e-16 ***
ma1      -1.029e-01   4.625e-02   -2.225  0.02605 *  
ma2       2.204e-01   3.843e-02    5.736 9.69e-09 ***
ma3      -9.001e-01   4.401e-02  -20.450  < 2e-16 ***
omega     8.620e-06   4.442e-06    1.941  0.05230 .  
alpha1    4.231e-02   1.498e-02    2.825  0.00473 ** 
beta1     9.323e-01   2.483e-02   37.544  < 2e-16 ***
---

Standardised Residuals Tests:
                                Statistic p-Value     
 Jarque-Bera Test   R    Chi^2  809.2296  0           
 Shapiro-Wilk Test  R    W      0.9622079 1.831348e-15
 Ljung-Box Test     R    Q(10)  11.24479  0.3387672   
 Ljung-Box Test     R    Q(15)  15.04573  0.4481283   
 Ljung-Box Test     R    Q(20)  17.95988  0.5900512   
 Ljung-Box Test     R^2  Q(10)  4.561984  0.9184557   
 Ljung-Box Test     R^2  Q(15)  7.24801   0.9503986   
 Ljung-Box Test     R^2  Q(20)  10.6087   0.9557459   
 LM Arch Test       R    TR^2   5.015022  0.9574753   

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC 
-5.182282 -5.133399 -5.182478 -5.163708 
```

Amazon:

```
Error Analysis:
           Estimate  Std. Error  t value Pr(>|t|)    
mu        6.847e-05   5.178e-05    1.322  0.18608    
ar1       9.164e-01   4.926e-02   18.605  < 2e-16 ***
ma1      -9.470e-01   4.109e-02  -23.044  < 2e-16 ***
omega     4.694e-05   1.970e-05    2.383  0.01719 *  
alpha1    7.987e-02   2.561e-02    3.118  0.00182 ** 
beta1     7.942e-01   6.903e-02   11.506  < 2e-16 ***
---
```

```
Standardised Residuals Tests:
                                Statistic p-Value
 Jarque-Bera Test   R    Chi^2  263.2837  0
 Shapiro-Wilk Test  R    W      0.9714997 3.8325e-13
 Ljung-Box Test     R    Q(10)  3.952023  0.9494854
 Ljung-Box Test     R    Q(15)  10.74849  0.7702114
 Ljung-Box Test     R    Q(20)  12.79633  0.8859506
 Ljung-Box Test     R^2  Q(10)  5.083361  0.8855403
 Ljung-Box Test     R^2  Q(15)  9.394289  0.8560158
 Ljung-Box Test     R^2  Q(20)  11.58147  0.9297287
 LM Arch Test       R    TR^2   6.288555  0.900843
```

```
Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-5.091945 -5.062616 -5.092016 -5.080801
```

Walmart:

```
        Estimate  Std. Error  t value Pr(>|t|)
mu      6.269e-04   2.806e-04    2.234 0.025477 *
omega   3.042e-05   8.061e-06    3.774 0.000161 ***
alpha1  1.115e-01   3.267e-02    3.414 0.000641 ***
beta1   5.292e-01   1.101e-01    4.807 1.53e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Standardised Residuals Tests:
                                Statistic p-Value
 Jarque-Bera Test   R    Chi^2  510.3221  0
 Shapiro-Wilk Test  R    W      0.9629493 2.707259e-15
 Ljung-Box Test     R    Q(10)  10.21359  0.4219583
 Ljung-Box Test     R    Q(15)  13.27549  0.5810286
 Ljung-Box Test     R    Q(20)  17.08738  0.6472927
 Ljung-Box Test     R^2  Q(10)  6.373226  0.7829926
 Ljung-Box Test     R^2  Q(15)  21.02943  0.1358916
 Ljung-Box Test     R^2  Q(20)  34.86879  0.0208143
 LM Arch Test       R    TR^2   6.67328   0.8784281
```

```
Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-6.564658 -6.545105 -6.564689 -6.557228
```

The fitted models:

Ebay: $r_t = 2.719 \times 10^{-4} + a_t + 0.0774 r_{t-1} - 0.2646 r_{t-2} + 0.8383 r_{t-3} - 0.1029 a_{t-1} + 0.2204 a_{t-2}$

$- 0.9001 a_{t-3}$, $a_t = \sigma_t a_t$, $\varepsilon_t \sim N(0,1)$

$\sigma_t^2 = 8.62 \times 10^{-6} + 0.0423 a^2_{t-1} + 0.9323\, \sigma^2_{t-1}$

Amazon: $r_t = 6.847 \times 10^{-5} + 0.9164 r_{t-1} - 0.947 a_{t-1} + a_t$, $a_t = \sigma_t a_t$, $\varepsilon_t \sim N(0,1)$

$\sigma_t^2 = 4.694 \times 10^{-5} + 0.0799 a^2_{t-1} + 0.0794 \sigma^2_{t-1}$

Walmart: $r_t = 6.269 \times 10^{-4} + a_t$, $a_t = \sigma_t a_t$, $\varepsilon_t \sim N(0,1)$

$\sigma_t^2 = 3.042 \times 10^{-5} + 0.1115 a^2_{t-1} + 0.5292\, \sigma^2_{t-1}$

(j) Build an ARMA-GARCH model with Student-t innovations for the log return series. Perform model checking and write down the fitted model. Is this model better or worse than part (i)?

The QQ-norm plots show that the ARMA-GARCH(1,1) model with Student-t innovations is much better than part (i) models. Also, from the residual plots we can see that there are constant and random variances. The ACFs plots show that models are adequate. The AICs of these model are also better since they are smaller than AICs in part (i). All the coefficients are significant except mu and omega of Ebay and mu, omega and alpha1 for Amazon. All three models are adequate.

Ebay:

```
Error Analysis:
         Estimate  Std. Error  t value Pr(>|t|)    
mu      1.404e-04  8.156e-05    1.722 0.085158 .  
ar1     1.189e-01  3.332e-02    3.570 0.000357 ***
ar2    -2.484e-01  3.454e-02   -7.193 6.36e-13 ***
ar3     8.763e-01  3.611e-02   24.267  < 2e-16 ***
ma1    -1.507e-01  2.705e-02   -5.573 2.50e-08 ***
ma2     2.073e-01  3.024e-02    6.855 7.12e-12 ***
ma3    -9.283e-01  2.639e-02  -35.172  < 2e-16 ***
omega   1.225e-06  1.303e-06    0.940 0.347165    
alpha1  2.278e-02  7.777e-03    2.930 0.003393 ** 
beta1   9.732e-01  9.299e-03  104.652  < 2e-16 ***
shape   4.884e+00  6.940e-01    7.037 1.96e-12 ***
---

Standardised Residuals Tests:
                                Statistic    p-Value
 Jarque-Bera Test   R    Chi^2  959.43312    0
 Shapiro-Wilk Test  R    W      0.9564323397 0
 Ljung-Box Test     R    Q(10)  13.96774961  0.1744677376
 Ljung-Box Test     R    Q(15)  16.77013602  0.3327904041
 Ljung-Box Test     R    Q(20)  19.01516782  0.5208402926
 Ljung-Box Test     R^2  Q(10)  5.185209357  0.8784672636
 Ljung-Box Test     R^2  Q(15)  9.623327356  0.8427412151
 Ljung-Box Test     R^2  Q(20)  12.74377237  0.8881049363
 LM Arch Test       R    TR^2   6.138381248  0.9089461175

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC 
-5.289968 -5.236197 -5.290204 -5.269536 
```

Amazon:

```
Error Analysis:
         Estimate  Std. Error  t value Pr(>|t|)    
mu      5.536e-05   3.902e-05    1.419    0.156    
ar1     9.357e-01   3.468e-02   26.981  < 2e-16 ***
ma1    -9.627e-01   2.776e-02  -34.675  < 2e-16 ***
omega   2.921e-05   2.844e-05    1.027    0.304    
alpha1  5.232e-02   3.659e-02    1.430    0.153    
beta1   8.705e-01   1.073e-01    8.112 4.44e-16 ***
shape   4.894e+00   7.433e-01    6.585 4.56e-11 ***
---
```

```
Standardised Residuals Tests:
                                Statistic   p-Value
 Jarque-Bera Test   R    Chi^2  263.2837077 0
 Shapiro-Wilk Test  R    W      0.9714996718 3.832500108e-13
 Ljung-Box Test     R    Q(10)  3.952022996 0.9494853516
 Ljung-Box Test     R    Q(15)  10.74848743 0.7702113648
 Ljung-Box Test     R    Q(20)  12.79632962 0.8859505614
 Ljung-Box Test     R^2  Q(10)  5.083360625 0.8855402792
 Ljung-Box Test     R^2  Q(15)  9.394289173 0.8560158312
 Ljung-Box Test     R^2  Q(20)  11.58147296 0.9297286529
 LM Arch Test       R    TR^2   6.288554816 0.9008430358

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-5.172365 -5.138147 -5.172462 -5.159363
```

Walmart:

```
Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu     7.0505e-04  2.4744e-04   2.8493 0.004381 **
omega  3.4690e-05  1.1594e-05   2.9921 0.002770 **
alpha1 1.4493e-01  4.9004e-02   2.9574 0.003102 **
beta1  4.5263e-01  1.5101e-01   2.9975 0.002722 **
shape  4.7220e+00  6.7808e-01   6.9638 3.313e-12 ***
---

Standardised Residuals Tests:
                                Statistic   p-Value
 Jarque-Bera Test   R    Chi^2  530.8012978 0
 Shapiro-Wilk Test  R    W      0.9622315341 1.854121544e-15
 Ljung-Box Test     R    Q(10)  10.32157348 0.4127479364
 Ljung-Box Test     R    Q(15)  13.41802798 0.5700437133
 Ljung-Box Test     R    Q(20)  17.17206863 0.6417738948
 Ljung-Box Test     R^2  Q(10)  6.89303965  0.7355026175
 Ljung-Box Test     R^2  Q(15)  22.21415065 0.1023104129
 Ljung-Box Test     R^2  Q(20)  35.72762226 0.01655427229
 LM Arch Test       R    TR^2   7.044143036 0.8546826479

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-6.668692 -6.644251 -6.668742 -6.659405
```

The fitted models:

Ebay: $r_t = 1.404\text{x}10^{-4} + a_t + 0.1189r_{t-1} – 0.2484r_{t-2} + 0.8763r_{t-3} – 0.1507a_{t-1} + 0.2073a_{t-2}$

$– 0.9283a_{t-3},\ a_t = \sigma_t a_t,\quad \varepsilon_t \sim N(0,1)$

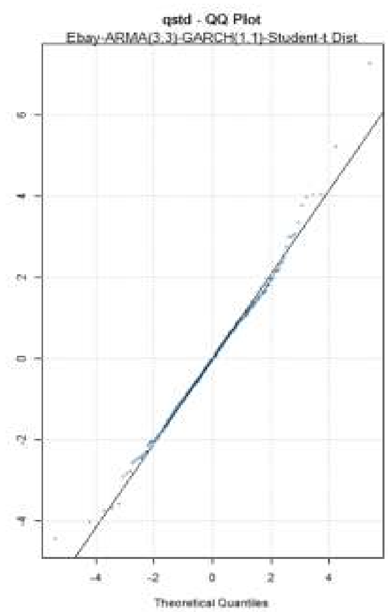
$\sigma_t^2 = 1.2249\text{x}10^{-6} + 0.0278a^2_{t-1} + 0.9732\ \sigma^2_{t-1}$

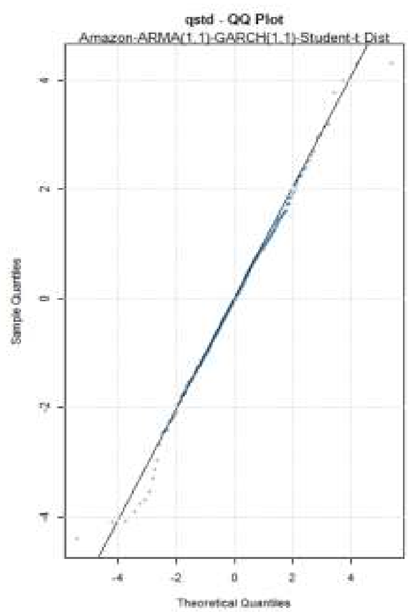
Amazon: $r_t = 5.5361\text{x}10^{-5} + 0.9356r_{t-1} – 0.9626a_{t-1} + a_t,\ a_t = \sigma_t a_t,\quad \varepsilon_t \sim N(0,1)$

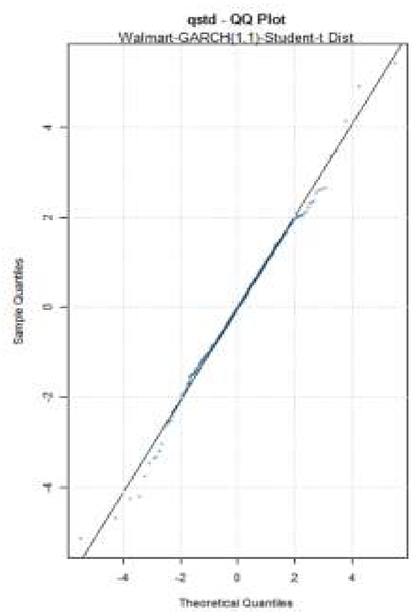
$\sigma_t^2 = 2.9208\text{x}10^{-5} + 0.0523a^2_{t-1} + 0.8705\sigma^2_{t-1}$

Walmart: $r_t = 7.051\text{x}10^{-4} + a_t,\ a_t = \sigma_t a_t,\quad \varepsilon_t \sim N(0,1)$

$\sigma_t^2 = 3.469\text{x}10^{-5} + 0.1449a^2_{t-1} + 0.4526\sigma^2_{t-1}$

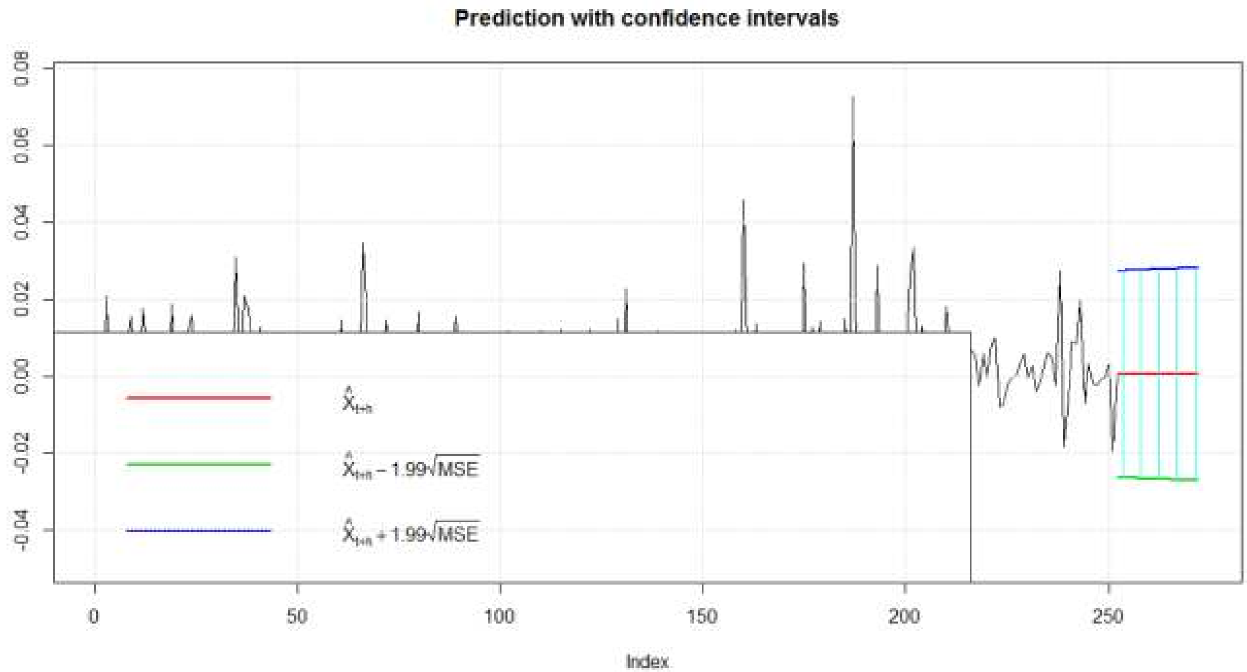




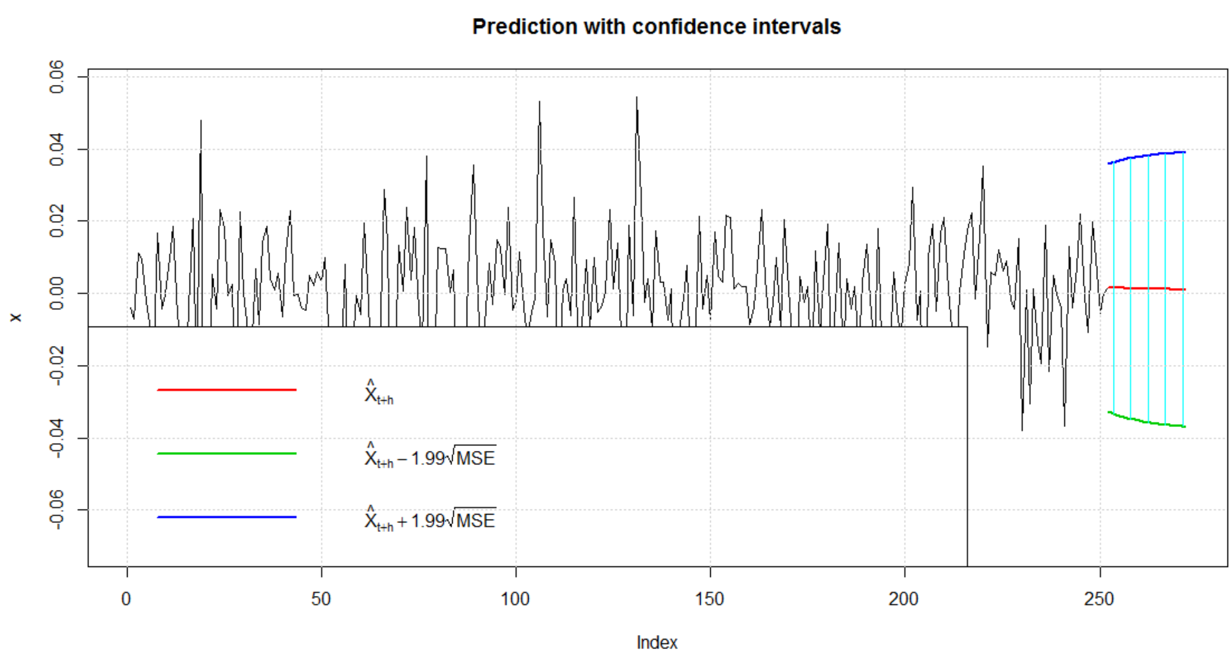


(k) Obtain 1-step ahead mean and volatility forecasts using the fitted ARMA-GARCH model with Student-t innovations with 95% confidence intervals for the first month in 2015.

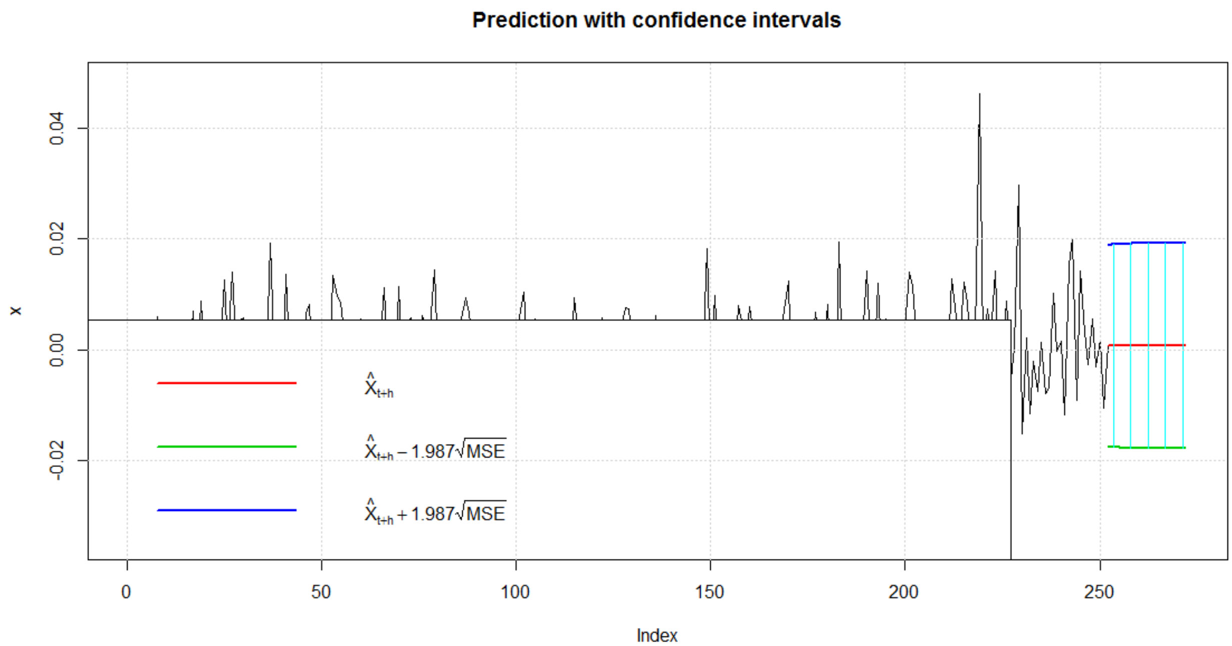
Plot of prediction with confidence intervals for first month in 2015 for Ebay:



Plot of prediction with confidence intervals for first month in 2015 for Amazon:



Plot of prediction with confidence intervals for first month in 2015 for Walmart:



Assuming 21 trading days in the month of January, the forecasted values, standard deviation and 95% lower interval, upper interval for Ebay, Amazon and Walmart are shown below:

Ebay:

	meanForecast	meanError	standardDeviation	lowerInterval	upperInterval
1	0.0005808137	0.01342058	0.01342058	-0.02612689	0.02728852
2	0.0005803505	0.01345531	0.01343990	-0.02619646	0.02735717
3	0.0005799368	0.01348680	0.01345910	-0.02625955	0.02741942
4	0.0005795673	0.01351570	0.01347820	-0.02631742	0.02747655
5	0.0005792373	0.01354251	0.01349718	-0.02637110	0.02752958
6	0.0005789427	0.01356764	0.01351607	-0.02642141	0.02757929
7	0.0005786795	0.01359141	0.01353484	-0.02646898	0.02762634
8	0.0005784445	0.01361408	0.01355352	-0.02651433	0.02767122
9	0.0005782347	0.01363586	0.01357208	-0.02655788	0.02771435
10	0.0005780472	0.01365690	0.01359055	-0.02659994	0.02775603
11	0.0005778799	0.01367733	0.01360891	-0.02664077	0.02779653
12	0.0005777304	0.01369726	0.01362717	-0.02668059	0.02783605
13	0.0005775969	0.01371678	0.01364533	-0.02671955	0.02787475
14	0.0005774777	0.01373594	0.01366339	-0.02675780	0.02791275
15	0.0005773712	0.01375479	0.01368135	-0.02679543	0.02795017
16	0.0005772762	0.01377339	0.01369921	-0.02683253	0.02798708
17	0.0005771912	0.01379176	0.01371697	-0.02686917	0.02802356
18	0.0005771154	0.01380993	0.01373464	-0.02690541	0.02805964
19	0.0005770477	0.01382792	0.01375220	-0.02694128	0.02809537
20	0.0005769872	0.01384575	0.01376968	-0.02697682	0.02813079
21	0.0005769332	0.01386343	0.01378705	-0.02701206	0.02816593

Amazon:

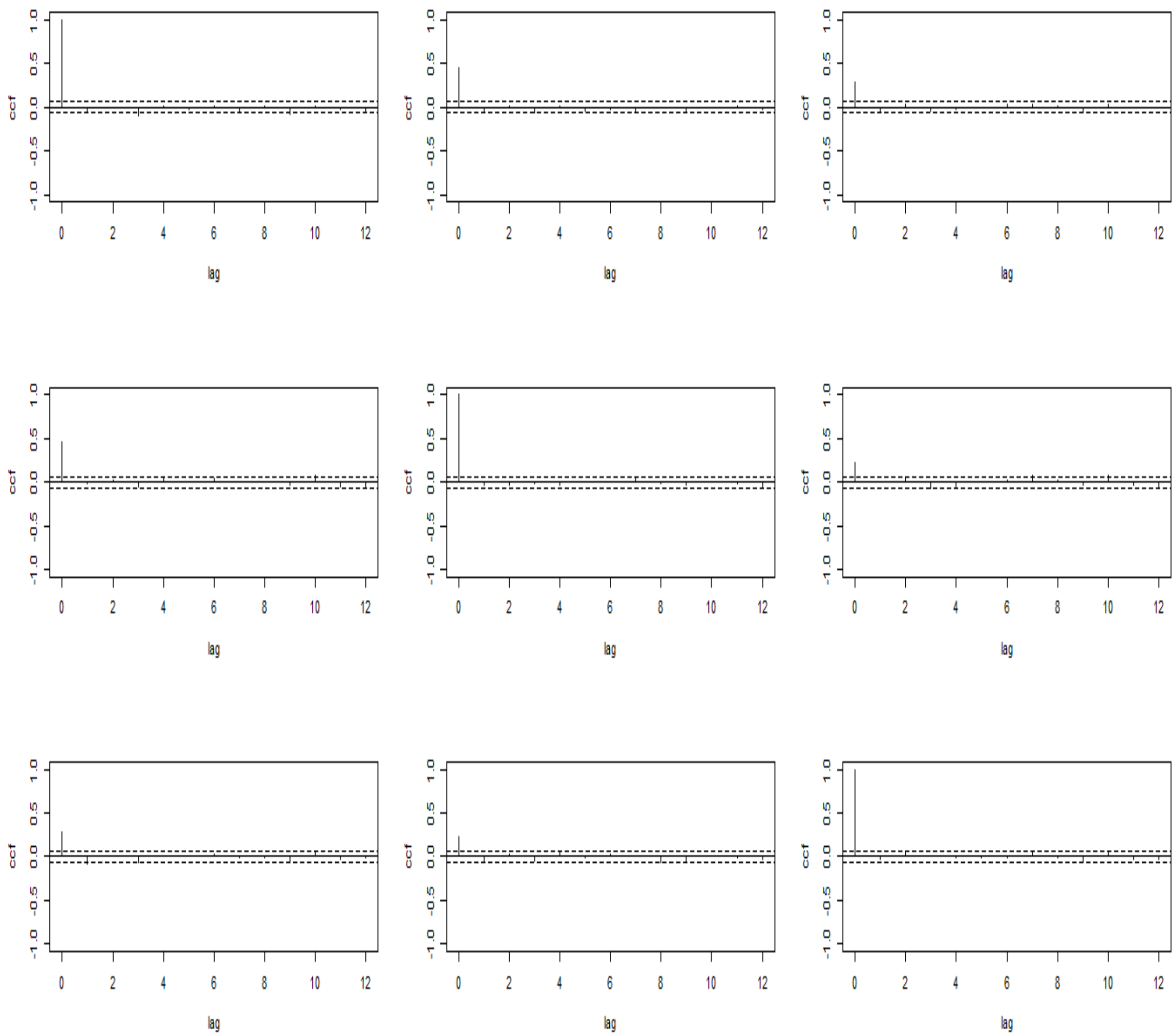
	meanForecast	meanError	standardDeviation	lowerInterval	upperInterval
1	0.001597749	0.01726478	0.01726478	-0.03275383	0.03594933
2	0.001550329	0.01744991	0.01744368	-0.03316960	0.03627026
3	0.001505959	0.01761887	0.01760717	-0.03355014	0.03656206
4	0.001464443	0.01777322	0.01775671	-0.03389877	0.03682765
5	0.001425598	0.01791436	0.01789360	-0.03421844	0.03706963
6	0.001389252	0.01804352	0.01801900	-0.03451178	0.03729028
7	0.001355244	0.01816181	0.01813396	-0.03478114	0.03749163
8	0.001323424	0.01827021	0.01823941	-0.03502865	0.03767550
9	0.001293651	0.01836961	0.01833618	-0.03525620	0.03784350
10	0.001265793	0.01846080	0.01842504	-0.03546550	0.03799709
11	0.001239727	0.01854451	0.01850666	-0.03565812	0.03813757
12	0.001215337	0.01862138	0.01858167	-0.03583545	0.03826612
13	0.001192517	0.01869199	0.01865062	-0.03599877	0.03838380
14	0.001171165	0.01875689	0.01871402	-0.03614924	0.03849157
15	0.001151186	0.01881654	0.01877234	-0.03628792	0.03859029
16	0.001132493	0.01887140	0.01882601	-0.03641577	0.03868076
17	0.001115002	0.01892187	0.01887539	-0.03653367	0.03876367
18	0.001098636	0.01896830	0.01892086	-0.03664241	0.03883968
19	0.001083323	0.01901103	0.01896271	-0.03674274	0.03890939
20	0.001068995	0.01905036	0.01900126	-0.03683533	0.03897332
21	0.001055589	0.01908657	0.01903676	-0.03692078	0.03903196

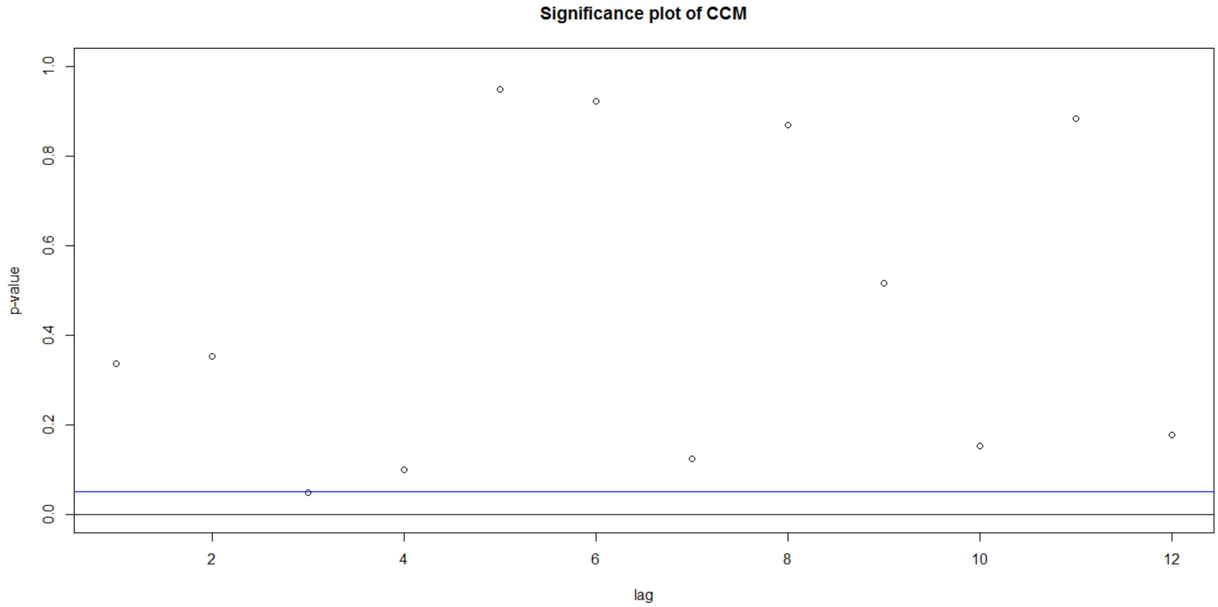
Walmart:

	meanForecast	meanError	standardDeviation	lowerInterval	upperInterval
1	0.0007050475	0.009111362	0.009111362	-0.01739713	0.01880722
2	0.0007050475	0.009181366	0.009181366	-0.01753621	0.01894630
3	0.0007050475	0.009222945	0.009222945	-0.01761882	0.01902891
4	0.0007050475	0.009247701	0.009247701	-0.01766800	0.01907810
5	0.0007050475	0.009262463	0.009262463	-0.01769733	0.01910742
6	0.0007050475	0.009271273	0.009271273	-0.01771483	0.01912493
7	0.0007050475	0.009276533	0.009276533	-0.01772528	0.01913538
8	0.0007050475	0.009279675	0.009279675	-0.01773153	0.01914162
9	0.0007050475	0.009281552	0.009281552	-0.01773525	0.01914535
10	0.0007050475	0.009282673	0.009282673	-0.01773748	0.01914758
11	0.0007050475	0.009283343	0.009283343	-0.01773881	0.01914891
12	0.0007050475	0.009283744	0.009283744	-0.01773961	0.01914970
13	0.0007050475	0.009283983	0.009283983	-0.01774008	0.01915018
14	0.0007050475	0.009284126	0.009284126	-0.01774037	0.01915046
15	0.0007050475	0.009284211	0.009284211	-0.01774054	0.01915063
16	0.0007050475	0.009284262	0.009284262	-0.01774064	0.01915073
17	0.0007050475	0.009284293	0.009284293	-0.01774070	0.01915080
18	0.0007050475	0.009284311	0.009284311	-0.01774074	0.01915083
19	0.0007050475	0.009284322	0.009284322	-0.01774076	0.01915085
20	0.0007050475	0.009284329	0.009284329	-0.01774077	0.01915087
21	0.0007050475	0.009284333	0.009284333	-0.01774078	0.01915087

(l) Is there significant cross-correlation in the log returns for the 3 companies?

Yes, there is significant cross-correlation at lag 4 in the log returns for the 3 companies. The Ljung-Box statistics show that p-value = 0.04 at lag 4.

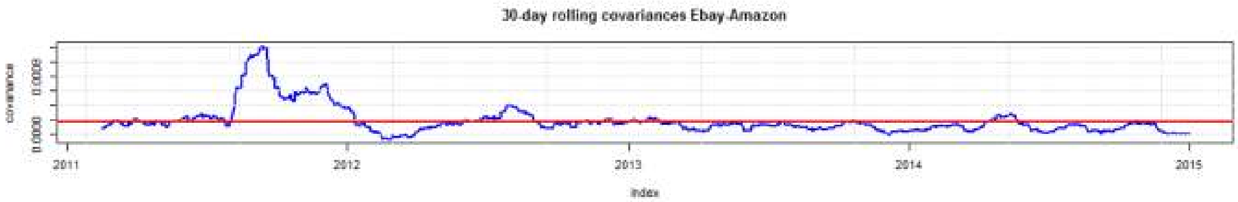


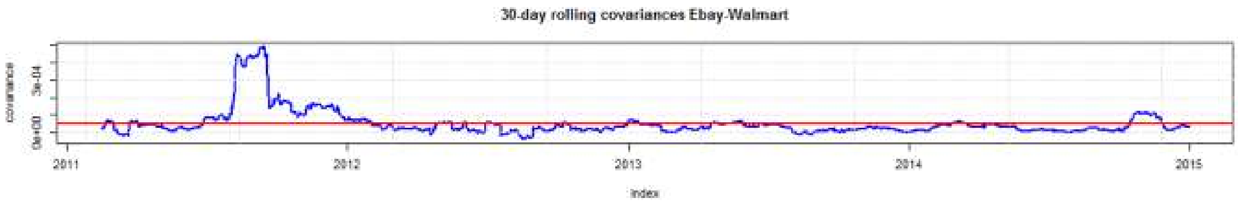


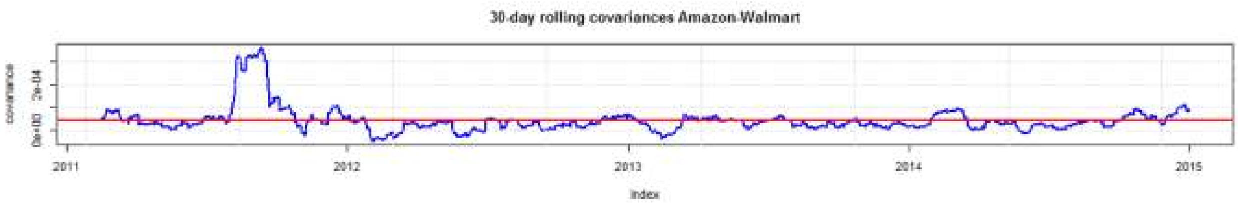

```
Ljung-Box Statistics:
         m       Q(m)      df    p-value
 [1,]   1.0      10.2     9.0      0.34
 [2,]   2.0      20.2    18.0      0.32
 [3,]   3.0      37.2    27.0      0.09
 [4,]   4.0      51.9    36.0      0.04
 [5,]   5.0      55.3    45.0      0.14
 [6,]   6.0      59.1    54.0      0.29
 [7,]   7.0      73.1    63.0      0.18
 [8,]   8.0      77.7    72.0      0.30
 [9,]   9.0      85.9    81.0      0.33
[10,]  10.0      99.1    90.0      0.24
[11,]  11.0     103.5    99.0      0.36
[12,]  12.0     116.2   108.0      0.28
```

(m) Using a 30-day moving window, compute and plot rolling covariances and correlations. Briefly comment on what you see.

In the three pair-wise 30-day rolling covariance charts, we see small positive covariance fluctuations throughout the series with a dramatic spike in the beginning of second half of 2011. The July-August spike is caused by the dramatic incidences of 2011: Greek default, S&P downgrades in the United States and Gold Hike from $1440 to $1840 happening at the same time frame followed by a dramatic drop a little more than a month later.

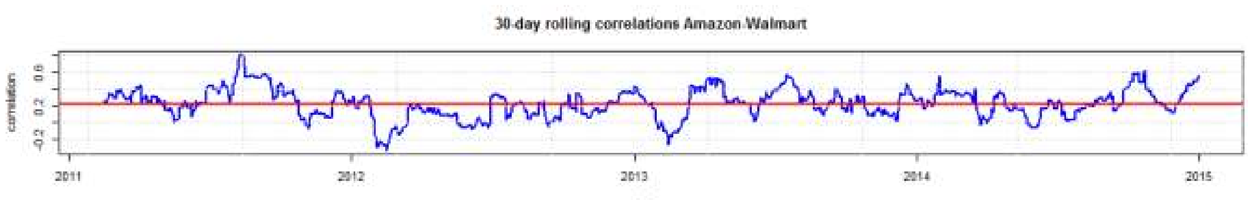

30-day rolling covariances Ebay-Amazon
covariance
0.0000
0.0008
2011
2012
2013
2014
2015
Index


30-day rolling covariances Ebay-Walmart
covariance
0e+00
3e-04
2011
2012
2013
2014
2015
Index


30-day rolling covariances Amazon-Walmart
covariance
0e+00
2e-04
2011
2012
2013
2014
2015
Index

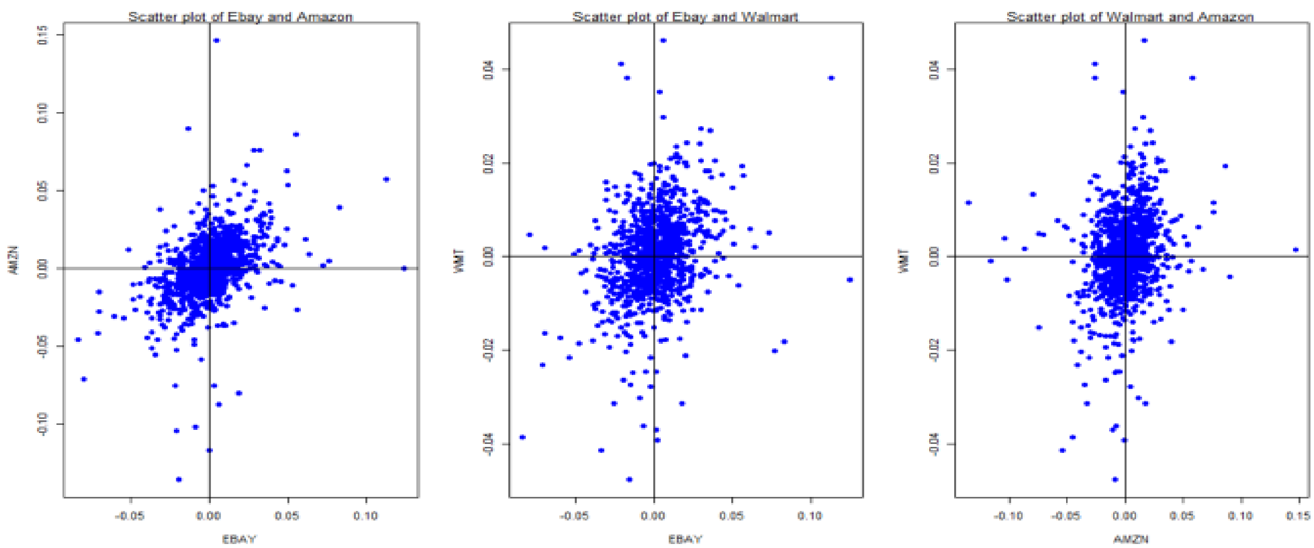

correlation
-0.2
0.2
0.6
2011
2012
2013
2014
2015
Index


30-day rolling correlations Ebay-Walmart
correlation
-0.2
0.2
0.6
2011
2012
2013
2014
2015
Index


30-day rolling correlations Amazon-Walmart
correlation
-0.2
0.2
0.6
2011
2012
2013
2014
2015
Index

(n) Let r_t = (r_EBAY,t, r_AMZN,t, r_WMT,t)^T. Using the dccfit() function from the rmgarch package, estimate the normal-DCC(1,1) model. Briefly comment on the estimated coefficients and the fit of the model.

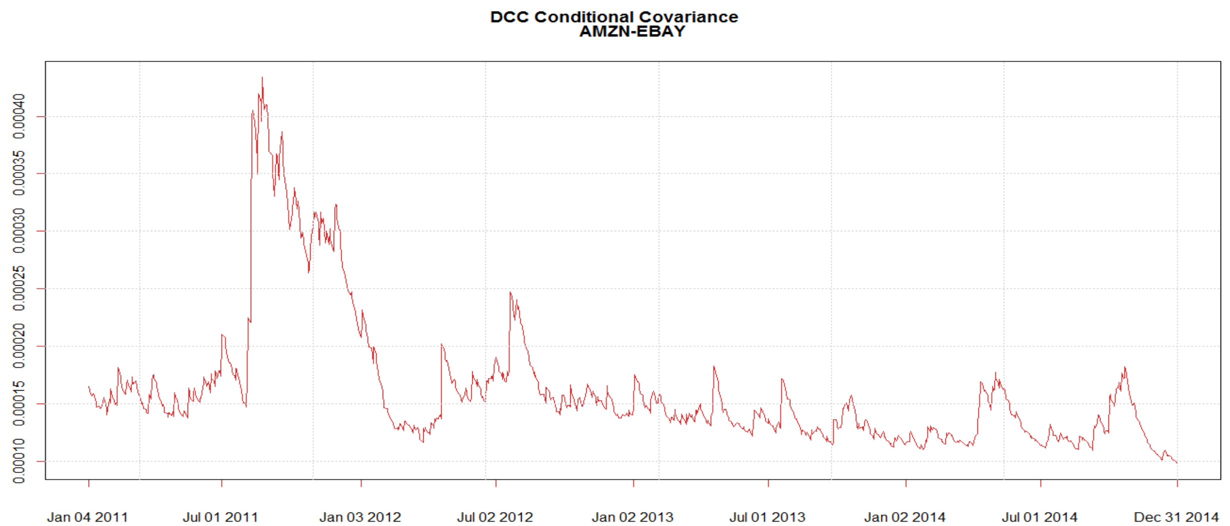
From the R-output, we can see that all of the estimated parameters were significant except the mu parameters for all three series, and alpha1 for Amazon. The parameter alpha1 and including the jointly estimated alpha were moderately significant except alpha1 parameter of Amazon series. The decay term (beta1) was the greatest significant coefficients with p-values equal 0.0. This is what we expected since in an analysis of the correlated return series in the scatterplots, we expected to see significant estimates in the joint estimated coefficients for alpha and beta coefficients.

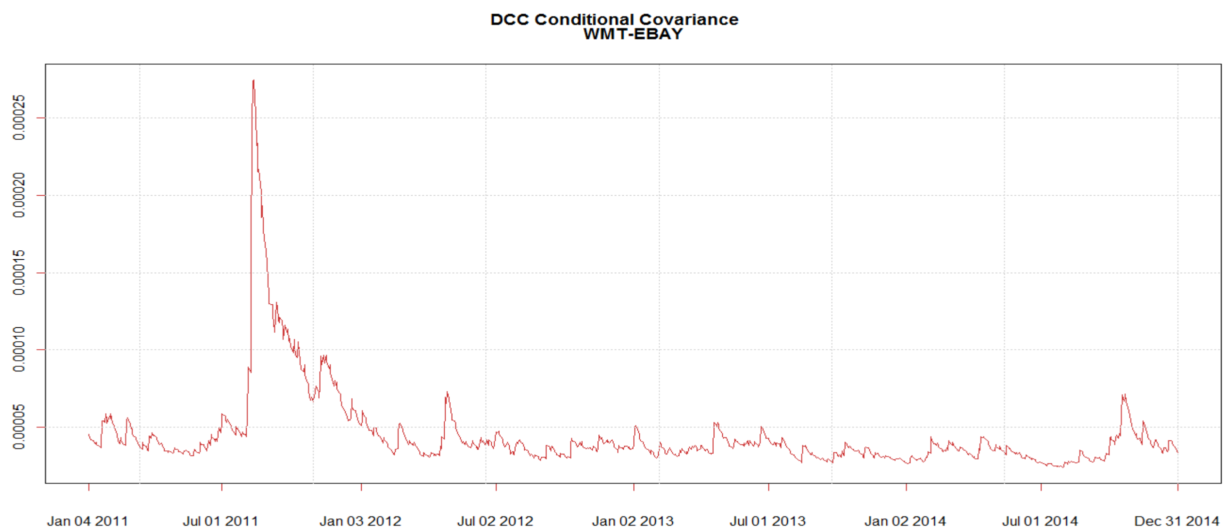


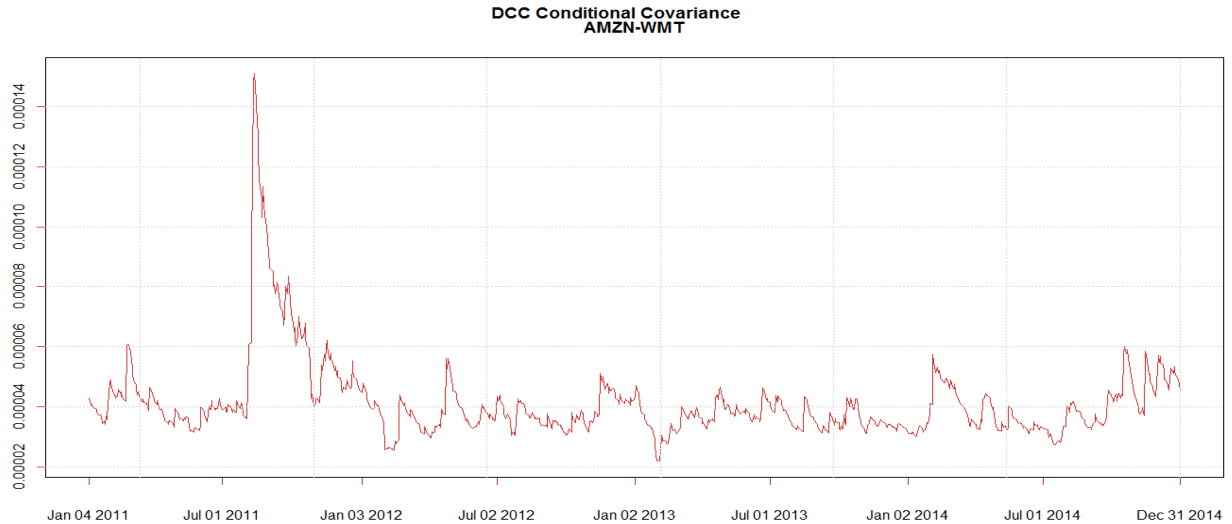

```
Optimal Parameters
------------------------------------
                  Estimate  Std. Error   t value Pr(>|t|)
[EBAY].mu         0.000890    0.000562   1.58293 0.113438
[EBAY].omega      0.000009    0.000001   7.41377 0.000000
[EBAY].alpha1     0.042214    0.004468   9.44711 0.000000
[EBAY].beta1      0.932536    0.007872 118.46076 0.000000
[AMZN].mu         0.000585    0.000644   0.90798 0.363889
[AMZN].omega      0.000005    0.000002   3.01767 0.002547
[AMZN].alpha1     0.007897    0.005539   1.42565 0.153970
[AMZN].beta1      0.980758    0.009459 103.68530 0.000000
[WMT].mu          0.000469    0.000281   1.66763 0.095389
[WMT].omega       0.000005    0.000000  19.81895 0.000000
[WMT].alpha1      0.047343    0.003248  14.57825 0.000000
[WMT].beta1       0.892312    0.010058  88.71278 0.000000
[Joint]dcca1      0.008884    0.004838   1.83612 0.066340
[Joint]dccb1      0.963136    0.015350  62.74630 0.000000
```

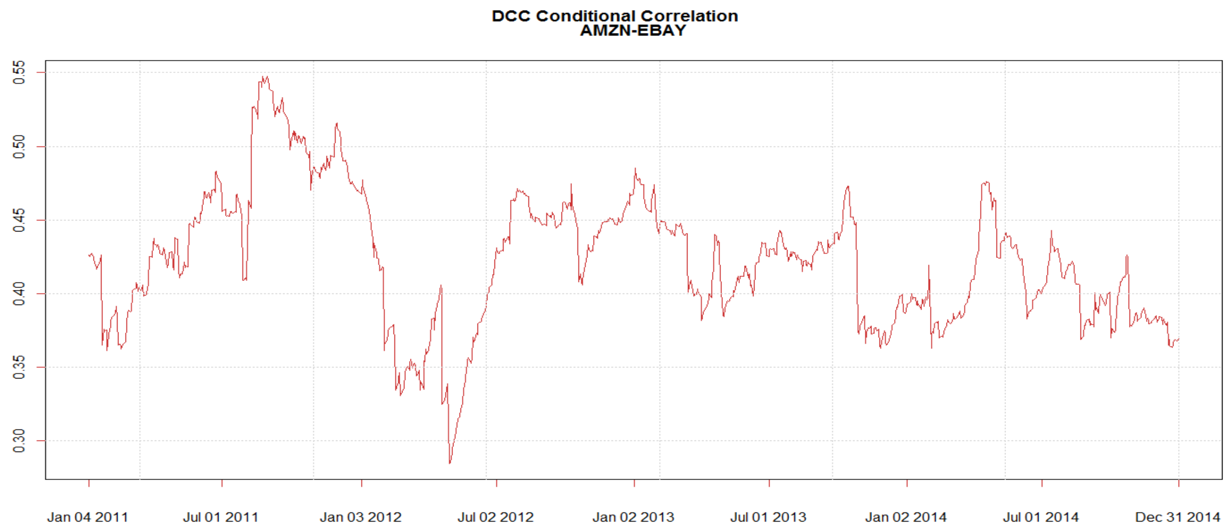
```
Information Criteria
---------------------

Akaike       -16.945
Bayes        -16.862
Shibata      -16.946
Hannan-Quinn -16.914


Elapsed time : 1.058794
```

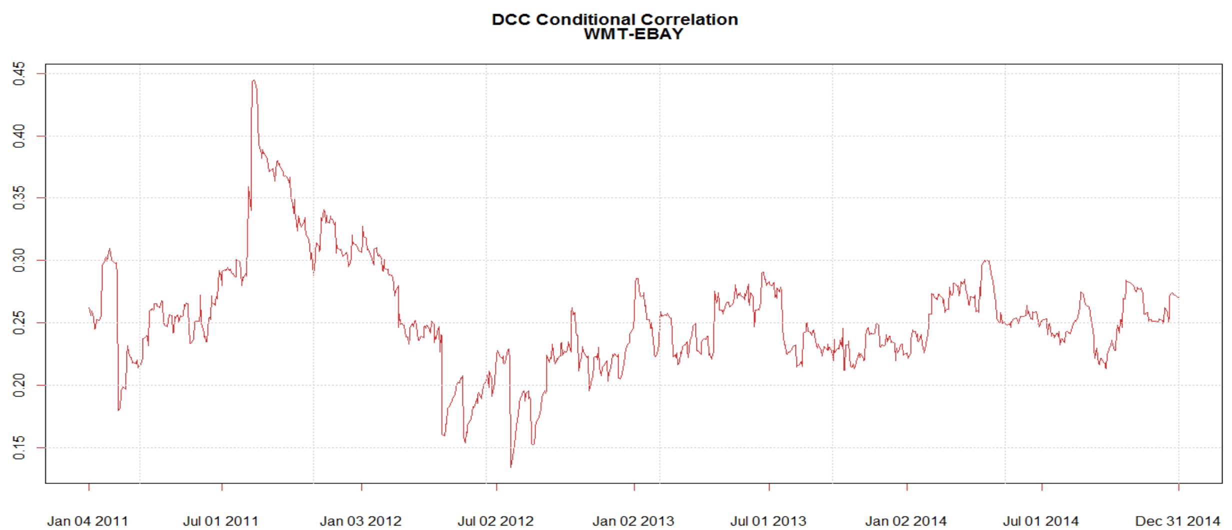
(o) Plot the estimated in-sample conditional covariances and correlations. Compare the EWMA and rolling estimates.

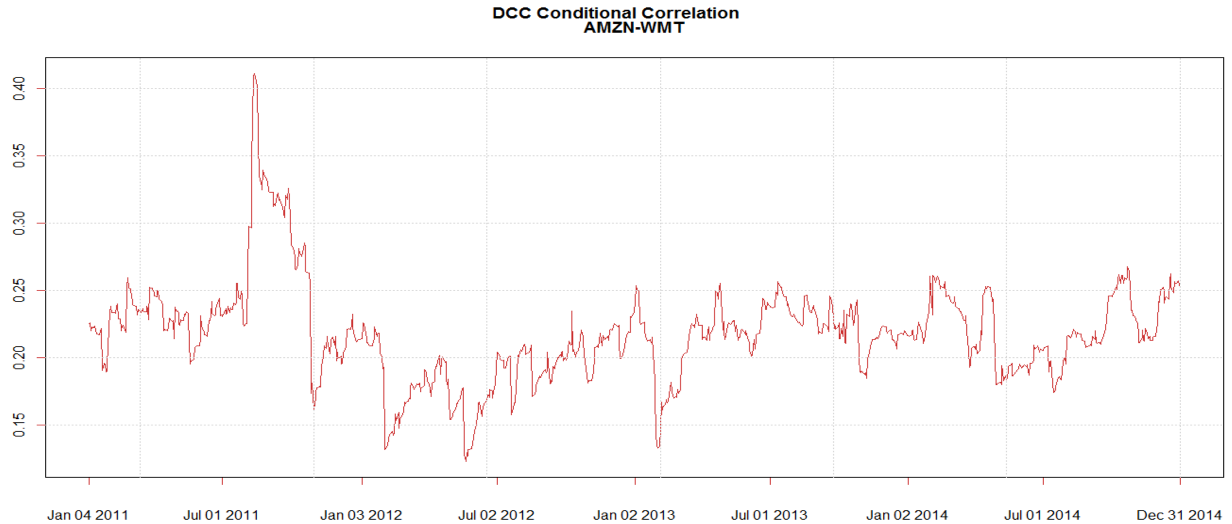


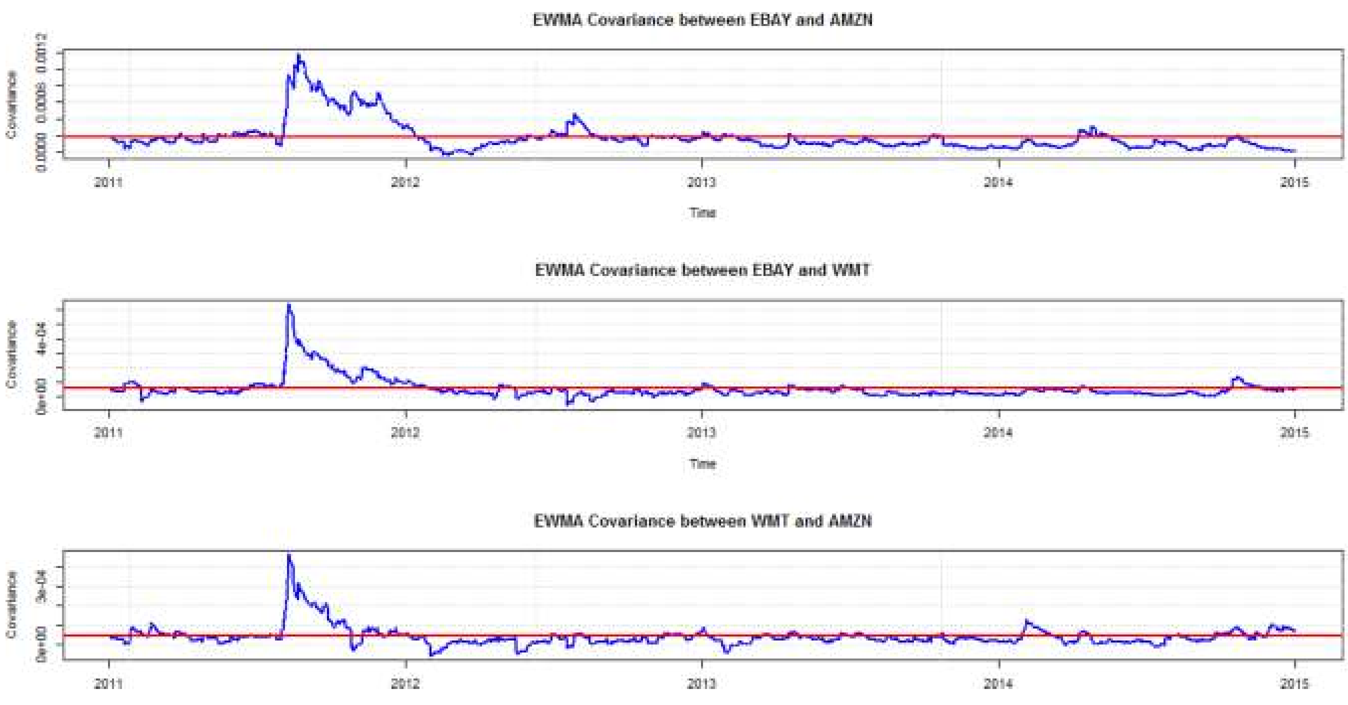


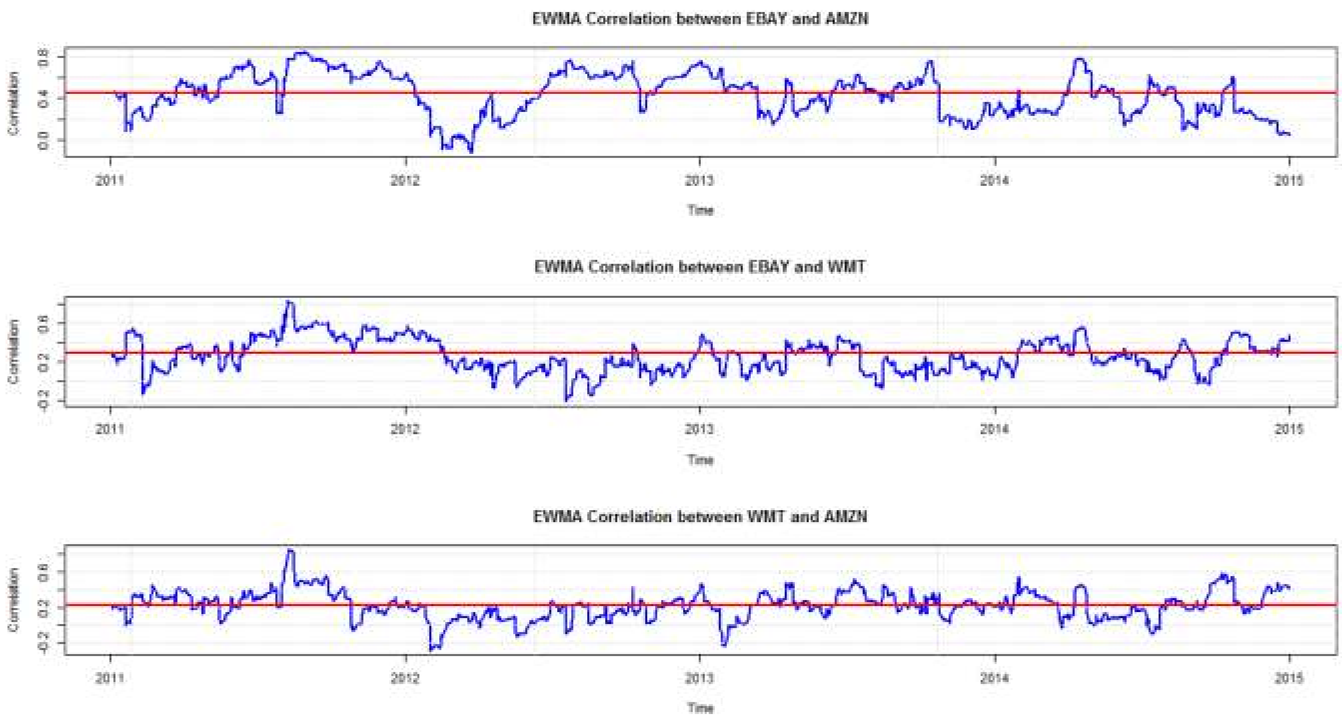

DCC Conditional Covariance
AMZN-WMT
0.00002
0.00004
0.00006
0.00008
0.00010
0.00012
0.00014
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014


DCC Conditional Correlation
AMZN-EBAY
0.30
0.35
0.40
0.45
0.50
0.55
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014

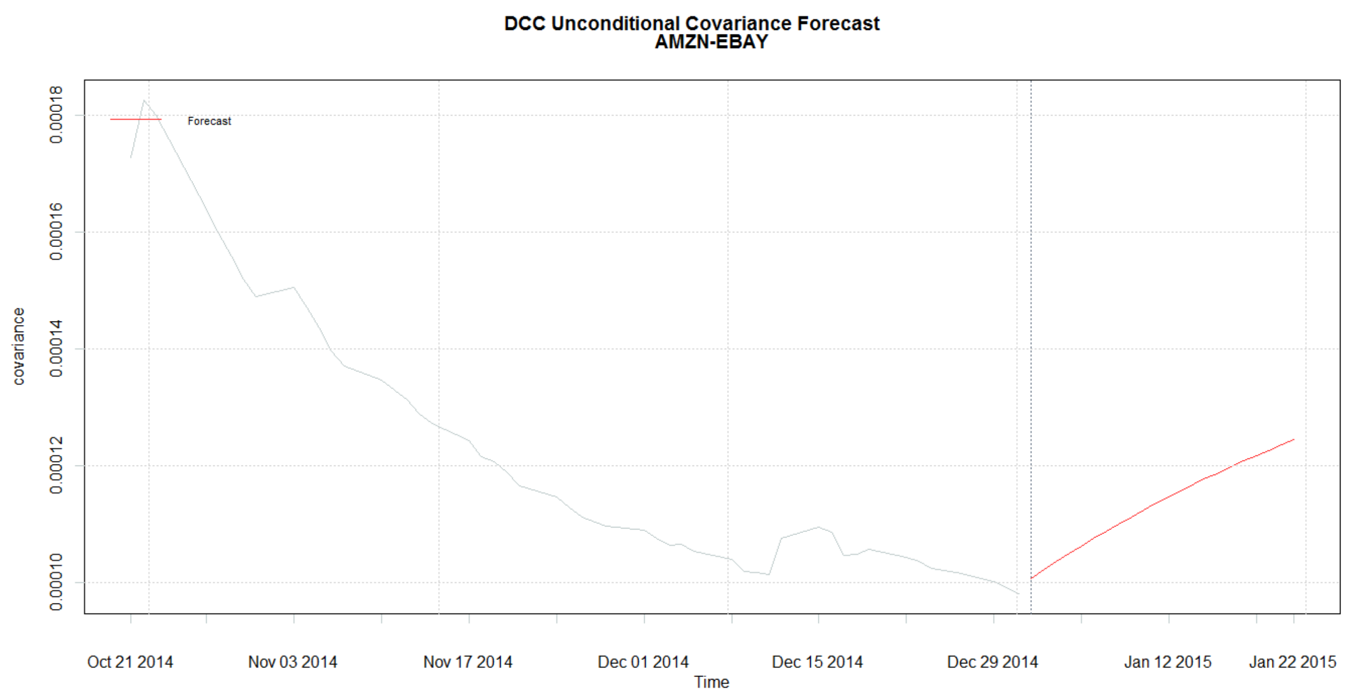

DCC Conditional Correlation
WMT-EBAY
0.15
0.20
0.25
0.30
0.35
0.40
0.45
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014

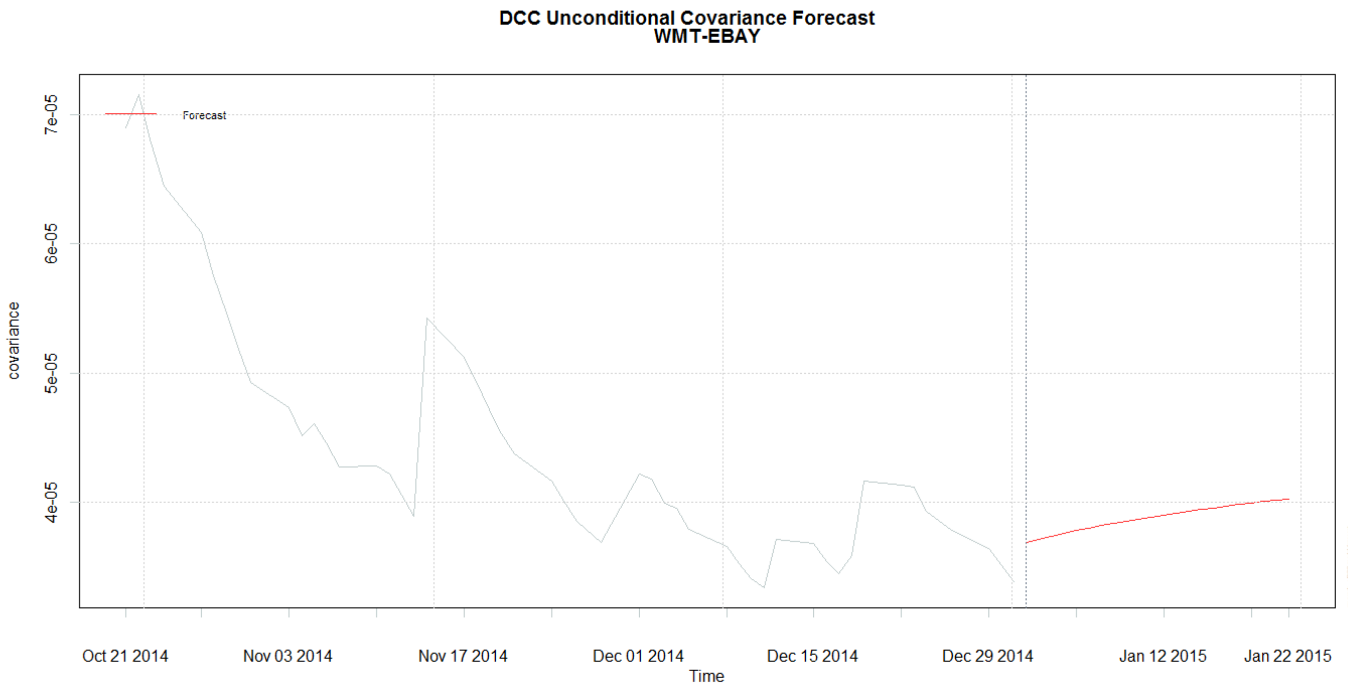

DCC Conditional Correlation
AMZN-WMT
0.40
0.35
0.30
0.25
0.20
0.15
Jan 04 2011
Jul 01 2011
Jan 03 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 31 2014

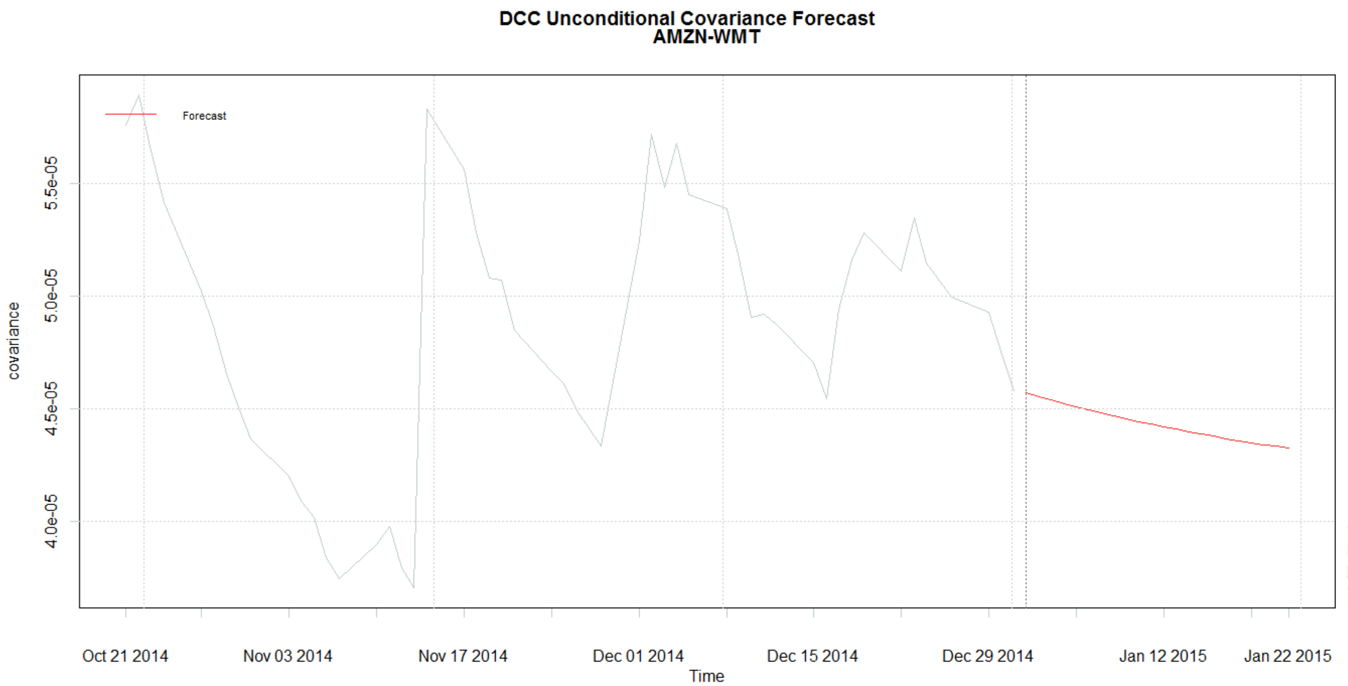

EWMA Covariance between EBAY and AMZN
Covariance
Time
2011
2012
2013
2014
2015
EWMA Covariance between EBAY and WMT
Covariance
4e-04
0e+00
Time
EWMA Covariance between WMT and AMZN
Covariance
3e-04
0e+00
Time



(p) Using the Estimated DCC(1,1) model, compute (using dccforecast() function) and plot the first month in 2015's 1-step ahead forecasts of conditional covariance and correlation.




DCC Unconditional Covariance Forecast
WMT-EBAY
Forecast
covariance
7e-05
6e-05
5e-05
4e-05
Oct 21 2014
Nov 03 2014
Nov 17 2014
Dec 01 2014
Dec 15 2014
Dec 29 2014
Jan 12 2015
Jan 22 2015
Time


DCC Unconditional Covariance Forecast
AMZN-WMT
Forecast
covariance
5.5e-05
5.0e-05
4.5e-05
4.0e-05
Oct 21 2014
Nov 03 2014
Nov 17 2014
Dec 01 2014
Dec 15 2014
Dec 29 2014
Jan 12 2015
Jan 22 2015
Time

Mean Forecasts:

	EBAY	AMZN	WMT
T+1	0.0008902927673	0.000584685502	0.0004692418079
T+2	0.0008902927673	0.000584685502	0.0004692418079
T+3	0.0008902927673	0.000584685502	0.0004692418079
T+4	0.0008902927673	0.000584685502	0.0004692418079
T+5	0.0008902927673	0.000584685502	0.0004692418079
T+6	0.0008902927673	0.000584685502	0.0004692418079
T+7	0.0008902927673	0.000584685502	0.0004692418079
T+8	0.0008902927673	0.000584685502	0.0004692418079
T+9	0.0008902927673	0.000584685502	0.0004692418079
T+10	0.0008902927673	0.000584685502	0.0004692418079
T+11	0.0008902927673	0.000584685502	0.0004692418079
T+12	0.0008902927673	0.000584685502	0.0004692418079
T+13	0.0008902927673	0.000584685502	0.0004692418079
T+14	0.0008902927673	0.000584685502	0.0004692418079
T+15	0.0008902927673	0.000584685502	0.0004692418079
T+16	0.0008902927673	0.000584685502	0.0004692418079
T+17	0.0008902927673	0.000584685502	0.0004692418079
T+18	0.0008902927673	0.000584685502	0.0004692418079
T+19	0.0008902927673	0.000584685502	0.0004692418079
T+20	0.0008902927673	0.000584685502	0.0004692418079
T+21	0.0008902927673	0.000584685502	0.0004692418079
T+22	0.0008902927673	0.000584685502	0.0004692418079

Sigma Forecasts:

	EBAY	AMZN	WMT
T+1	0.01404177393	0.01955252369	0.009339478063
T+2	0.01417472907	0.01956381818	0.009334440479
T+3	0.01430313775	0.01957497813	0.009329704408
T+4	0.01442720425	0.01958600521	0.009325251940
T+5	0.01454711982	0.01959690109	0.009321066217
T+6	0.01466306377	0.01960766739	0.009317131366
T+7	0.01477520457	0.01961830575	0.009313432447
T+8	0.01488370072	0.01962881774	0.009309955398
T+9	0.01498870156	0.01963920494	0.009306686986
T+10	0.01509034804	0.01964946889	0.009303614758
T+11	0.01518877334	0.01965961113	0.009300726999
T+12	0.01528410344	0.01966963316	0.009298012682
T+13	0.01537645769	0.01967953647	0.009295461437
T+14	0.01546594929	0.01968932253	0.009293063508
T+15	0.01555268569	0.01969899279	0.009290809717
T+16	0.01563676900	0.01970854866	0.009288691432
T+17	0.01571829636	0.01971799158	0.009286700533
T+18	0.01579736026	0.01972732291	0.009284829385
T+19	0.01587404885	0.01973654404	0.009283070807
T+20	0.01594844619	0.01974565633	0.009281418046
T+21	0.01602063254	0.01975466110	0.009279864752
T+22	0.01609068454	0.01976355967	0.009278404953

(q) Compare your mean and volatility forecast from part (p) with parts (g) and (k). Which is the best model?

The ARMA model in g had the lowest degree of confidence primarily because there was no model for Walmart. Also, it failed to take into account any market fluctuations or shocks and simply produces forecast based on the mean. Since the mean coefficient was very small, this may not be a dramatic deviation, though it leaves a great deal of information out. ARIMA model focuses on analyzing time series linearly and it does not reflect the new information is available. Therefore, in order to update the model, we need to incorporate new data and estimate parameters again. The variance in ARIMA model is unconditional variance and remains constant. ARCH effects is a method to measure volatility of the series, or more specifically, to model the noise term of ARIMA model. Volatility modeling in part k) incorporates new information and analyzes the series based on conditional variances where we can forecast future values with up-to-date information. The forecast interval for the ARIMA_GARCH model from part l) is closer than that of ARIMA-only model from part g)

Unlike the ARMA model, the volatility forecast shows trends based upon past shocks with the upward trend for Amazon showing closer intervals compared to the DCC volatility plot. The DCC mean forecast showed trends in the future forecast which were flat in the previous two models reflecting a stronger weight in more recent market activity. Overall the best model is the DCC model given the high degree of fluctuation and strong shocks being captured.

(r) Use your model to develop a trading strategy involving these stocks. Use out-of-sample data to test your trading strategy. How did it perform? Make sure you take into account transaction costs.

Time series analysis is a useful way to predict the stock prices. There are some points in forecasting based on ARIMA-ARCH/GARCH model that is to be taken into account. Firstly, ARIMA model focuses on analyzing time series linearly and it does not reflect the new information is available. Therefore, in order to update the model, we need to incorporate new data and estimate parameters again. The variance in ARIMA model is unconditional variance and remains constant. ARCH effects is a method to measure volatility of the series, or more specifically, to model the noise term of ARIMA model. Volatility modeling in part i) incorporates new information and analyzes the series based on conditional variances where we can forecast future values with up-to-date information. The forecast interval for the ARIMA_GARCH model from part l) is closer than that of ARIMA-only model from part g)

We computed forecast accuracy measures for in-sample and out-of-sample (Jan1 – Jan 31 2015) and we observed that DCC model works best for out of sample prediction.

R Code

```
library(fBasics)
library(fGarch)
library(psych)
library(forecast)
library(PerformanceAnalytics)
library(quantmod)
library(robustbase)
library(rugarch)
library(car)
library(FinTS)
library(rmgarch)
library(MTS)
options(digits=10)
symbol.vec = c("EBAY", "AMZN", "WMT")
getSymbols(symbol.vec, from ="2011-01-01", to = "2014-12-31")
colnames(EBAY)
start(EBAY)
end(EBAY)
colnames(AMZN)
start(AMZN)
end(AMZN)
colnames(WMT)
start(WMT)
end(WMT)
save(EBAY,file="EBAY.Rdata")
save(AMZN,file="AMZN.Rdata")
save(WMT,file="WMT.Rdata")
# Extract adjusted closing prices
EBAY = EBAY[, "EBAY.Adjusted", drop=F]
```

```
AMZN = AMZN[, "AMZN.Adjusted", drop=F]
WMT = WMT[, "WMT.Adjusted", drop=F]
# Calculate log-returns for GARCH analysis
EBAY.ret = CalculateReturns(EBAY, method="log")
AMZN.ret = CalculateReturns(AMZN, method="log")
WMT.ret = CalculateReturns(WMT, method="log")
save(EBAY.ret,file="EBAYreturn.Rdata")
save(AMZN.ret,file="AMZNreturn.Rdata")
save(WMT.ret,file="WMTreturn.Rdata")
# Remove first NA observation
EBAY.ret = EBAY.ret[-1,]
AMZN.ret = AMZN.ret[-1,]
WMT.ret = WMT.ret[-1,]
colnames(EBAY.ret) ="EBAY"
colnames(AMZN.ret) = "AMZN"
colnames(WMT.ret) = "WMT"
# Plot the log returns in a single 1x3 plot
par(mfcol=c(1,3))
# Plot log returns
plot(EBAY.ret)
plot(AMZN.ret)
plot(WMT.ret)
stock.ret = merge(EBAY.ret, AMZN.ret, WMT.ret)
#a)
basicStats(stock.ret)
#b)
t.test(stock.ret$EBAY)
t.test(stock.ret$AMZN)
t.test(stock.ret$WMT)
#c)
```

```
# Testing for skewness
skew = abs(skewness(stock.ret)/sqrt(6/nrow(stock.ret)))
skew
pvalue = 2*(1-(sapply(abs(skew),pnorm)))
pvalue
#d)
# Testing for Kurtosis
kurt = kurtosis(stock.ret)/sqrt(24/nrow(stock.ret))
kurt
pval = 2*(1-(sapply(abs(kurt),pnorm)))
pval
#e)
# Plot density plots
par(mfcol=c(1,3))
densityPlot(as.timeSeries(EBAY.ret))
densityPlot(as.timeSeries(AMZN.ret))
densityPlot(as.timeSeries(WMT.ret))
# Print QQ-norm plots to assess normal distribution
qqnormPlot(as.timeSeries(EBAY.ret))
mtext("EBAY-Gaussian",side=3,cex=0.7,padj=-0.5)
qqnormPlot(as.timeSeries(AMZN.ret))
mtext("AMZN-Gaussian",side=3,cex=0.7,padj=-0.5)
qqnormPlot(as.timeSeries(WMT.ret))
mtext("WMT-Gaussian",side=3,cex=0.7,padj=-0.5)
# Print QQ-GHT plots to assess Generalized Hyperbolic Student-t distribution
qqghtPlot(as.timeSeries(EBAY.ret))
mtext("EBAY-Student-t Dist",side=3,cex=0.7,padj=-0.5)
qqghtPlot(as.timeSeries(AMZN.ret))
mtext("AMZN-Student-t Dist",side=3,cex=0.7,padj=-0.5)
qqghtPlot(as.timeSeries(WMT.ret))
```

```
mtext("WMT-Student-t Dist",side=3,cex=0.7,padj=-0.5)
# Print QQ-GLD plots to assess generalized lambda distribution
qqgldPlot(as.timeSeries(EBAY.ret))
mtext("EBAY-Lambda Dist",side=3,cex=0.7,padj=-0.5)
qqgldPlot(as.timeSeries(AMZN.ret))
mtext("AMZN-Lambda Dist",side=3,cex=0.7,padj=-0.5)
qqgldPlot(as.timeSeries(WMT.ret))
mtext("WMT-Lambda Dist",side=3,cex=0.7,padj=-0.5)
# Use a robust method for cleaning Outliers from the dataset
EBAY.ret.clean = Return.clean(EBAY.ret, method="boudt")
par(mfrow=c(2,1))
plot(EBAY.ret, main="Raw EBAY Returns", ylab="EBAY", ylim= c(-0.06,0.06))
plot(EBAY.ret.clean, main="Cleaned EBAY Returns", ylab="EBAY", ylim= c(-0.06,0.06))
AMZN.ret.clean = Return.clean(AMZN.ret, method="boudt")
par(mfrow=c(2,1))
plot(AMZN.ret, main="Raw AMZN Returns", ylab="AMZN", ylim= c(-0.06,0.06))
plot(AMZN.ret.clean, main="Cleaned AMZN Returns", ylab="AMZN", ylim= c(-0.06,0.06))
WMT.ret.clean = Return.clean(WMT.ret, method="boudt")
par(mfrow=c(2,1))
plot(WMT.ret, main="Raw WMT Returns", ylab="WMT", ylim= c(-0.05,0.05))
plot(WMT.ret.clean, main="Cleaned WMT Returns", ylab="WMT", ylim= c(-0.05,0.05))
#f)
# Model identification
par(mfcol=c(1,3))
acf(EBAY.ret)
acf(AMZN.ret)
acf(WMT.ret)
pacf(EBAY.ret)
pacf(AMZN.ret)
pacf(WMT.ret)
```

```
e1auto=ar(as.ts(EBAY.ret),method="mle")
e1auto$order
e1auto
a1auto=ar(as.ts(AMZN.ret),method="mle")
a1auto$order
a1auto
w1auto=ar(as.ts(WMT.ret),method="mle")
w1auto$order
w1auto
e1auto=auto.arima(EBAY.ret)
summary(e1auto)
par(mfcol=c(1,1))
tsdiag(e1auto)
mtext("Diagnostics for EBAY log Returns",side=3,cex=1.0,padj=-2)
sqrt(e1auto$sigma2) # Calculate the residual standard error
Box.test(e1auto$resid,lag=24,type='Ljung')
a2=arima(AMZN.ret.clean, order=c(0,0,1),include.mean=F)
summary(a2)
a1=arima(AMZN.ret.clean, order=c(3,0,3),include.mean=F)
summary(a1)
a1auto=auto.arima(AMZN.ret.clean)
summary(a1auto)
coef(a1auto)
tsdiag(a1auto)
mtext("Diagnostics for AMZN log Returns",side=3,cex=1.0,padj=-2)
sqrt(a1auto$sigma2)
Box.test(a1auto$resid,lag=24,type='Ljung')
w1auto=auto.arima(WMT.ret)
summary(w1auto)
coef(w1auto)
```

```
tsdiag(w1auto)
mtext("Diagnostics for WMT log Returns",side=3,cex=1.0,padj=-2)
sqrt(w1auto$sigma2)
Box.test(w1auto$resid,lag=24,type='Ljung')
#g)
e1.forecast=forecast(e1auto, h=21,)
a1.forecast=forecast(a1auto, h=21,)
w1.forecast=forecast(w1auto, h=21,)
e1.forecast
a1.forecast
w1.forecast
par(mfcol=c(3,1))
plot(e1.forecast, include=100,main="Forecast of Ebay Log Return Data for January 2015")
plot(a1.forecast, include=100, main="Forecast of Amazon Log Return Data for January 2015")
plot(w1.forecast,include=100, main="Forecast of Walmart Log Return Data for January 2015")
#h)
Box.test(e1auto$resid^2,lag=24,type='Ljung')
Box.test(a1auto$resid^2,lag=24,type='Ljung')
Box.test(w1auto$resid^2,lag=24,type='Ljung')
Box.test(EBAY.ret^2,lag=12,type='Ljung')
Box.test(AMZN.ret.clean^2,lag=24,type='Ljung')
Box.test(WMT.ret^2,lag=24,type='Ljung')
ArchTest(EBAY.ret^2, lags=24)
ArchTest(AMZN.ret.clean^2, lags=24)
ArchTest(WMT.ret^2, lags=24)
par(mfcol=c(1,3))
acf(EBAY.ret^2)
acf(AMZN.ret^2)
acf(WMT.ret^2)
pacf(EBAY.ret^2)
```

```
pacf(AMZN.ret^2)
pacf(WMT.ret^2)
par(mfcol=c(1,2))
acf(AMZN.ret.clean^2)
pacf(AMZN.ret.clean^2)
squared.res.e1=e1auto$resid^2
squared.res.a1=a1auto$resid^2
squared.res.w1=w1auto$resid^2
par(mfcol=c(3,1))
plot(squared.res.e1,main='Squared Residuals')
plot(squared.res.a1,main='Squared Residuals')
plot(squared.res.w1,main='Squared Residuals')
#i)
ebay1=garchFit(~arma(3,3)+garch(1,1),data=EBAY.ret,trace=F)
summary(ebay1)
coef(ebay1)
plot(ebay1)
mtext("Ebay-ARMA(3,3)-GARCH(1,1)-Normal Dist",side=3,cex=0.8)
amzn1 = garchFit(~arma(1,1)+garch(1,1),data=AMZN.ret.clean,trace=F)
summary(amzn1)
coef(amzn1)
plot(amzn1)
mtext("Amazon-ARMA(1,1)-GARCH(1,1)-Normal Dist",side=3,cex=0.8)
wmt1=garchFit(~arma(1,1)+garch(1,1),data=WMT.ret,trace=F)
summary(wmt1)
wmt2=garchFit(~garch(1,1),data=WMT.ret,trace=F)
summary(wmt2)
coef(wmt2)
plot(wmt2)
mtext("Walmart-GARCH(1,1)-Normal Dist",side=3,cex=0.8)
```

```
#j)
ebay2=garchFit(~arma(3,3)+garch(1,1),data=EBAY.ret,trace=F,cond.dist="std")
summary(ebay2)
coef(ebay2)
plot(ebay2)
mtext("Ebay-ARMA(3,3)-GARCH(1,1)-Student-t Dist",side=3,cex=0.8)
amzn2 = garchFit(~arma(1,1)+garch(1,1),data=AMZN.ret.clean,trace=F,cond.dist="std")
summary(amzn2)
coef(amzn2)
plot(amzn2)
mtext("Amazon-ARMA(1,1)-GARCH(1,1)-Student-t Dist",side=3,cex=0.8)
wmt3=garchFit(~garch(1,1),data=WMT.ret,trace=F,cond.dist="std")
summary(wmt3)
coef(wmt3)
plot(wmt3)
mtext("Walmart-GARCH(1,1)-Student-t Dist",side=3,cex=0.8)
#k)
# Because there is bug in forcasting higher order of arma(1,1)-garch(1,1)
# we fixed model arma(1,1)-garch(1,1) to get predictions.
ebay3=garchFit(~arma(1,1)+garch(1,1),data=EBAY.ret,trace=F,cond.dist="std")
summary(ebay3)
pebay3=predict(ebay3,n.ahead=21,plot=TRUE)
pebay3
pamzn2=predict(amzn2,n.ahead=21,plot=TRUE)
pamzn2
pwmt3=predict(wmt3,n.ahead=21,plot=TRUE)
pwmt3
#l)
logreturn=data.frame(EBAY.ret,AMZN.ret,WMT.ret)
head(logreturn)
```

```
ccm(logreturn, level=TRUE)
mq(logreturn, lag=12)
#m)
ebay_amzn = chart.RollingCorrelation(EBAY.ret, AMZN.ret, width=30)
ebay_wmt = chart.RollingCorrelation(EBAY.ret, WMT.ret, width=30)
amzn_wmt = chart.RollingCorrelation(WMT.ret, AMZN.ret, width=30)
#####

cor.fun = function(x){
  cor(x)[1,2]
}

cov.fun = function(x){
  cov(x)[1,2]
}

#ebay-amzn
roll.cov1 = rollapply(as.zoo(cbind(EBAY.ret,AMZN.ret)), FUN=cov.fun, width=30,
                      by.column=FALSE, align="right")
roll.cor1 = rollapply(as.zoo(cbind(EBAY.ret,AMZN.ret)), FUN=cor.fun, width=30,
                      by.column=FALSE, align="right")
summary(roll.cov1)
par(mfrow=c(2,1))
plot(roll.cov1, main="30-day rolling covariances Ebay-Amazon",
     ylab="covariance", lwd=2, col="blue")
grid()
abline(h=cov(logreturn)[1,2], lwd=2, col="red")

plot(roll.cor1, main="30-day rolling correlations Ebay-Amazon",
     ylab="correlation", lwd=2, col="blue")
```

```
grid()
abline(h=cor(logreturn)[1,2], lwd=2, col="red")
summary(roll.cov1)
summary(roll.cor1)

###ebay-walmart

roll.cov2 = rollapply(as.zoo(cbind(EBAY.ret,WMT.ret)), FUN=cov.fun, width=30,
                  by.column=FALSE, align="right")
roll.cor2 = rollapply(as.zoo(cbind(EBAY.ret,WMT.ret)), FUN=cor.fun, width=30,
                  by.column=FALSE, align="right")
par(mfrow=c(2,1))
plot(roll.cov2, main="30-day rolling covariances Ebay-Walmart",
    ylab="covariance", lwd=2, col="blue")
grid()
abline(h=cov(logreturn)[1,3], lwd=2, col="red")
plot(roll.cor2, main="30-day rolling correlations Ebay-Walmart",
    ylab="correlation", lwd=2, col="blue")
grid()
abline(h=cor(logreturn)[1,3], lwd=2, col="red")

#amzn-walmart:
roll.cov3 = rollapply(as.zoo(cbind(AMZN.ret,WMT.ret)), FUN=cov.fun, width=30,
                  by.column=FALSE, align="right")
roll.cor3 = rollapply(as.zoo(cbind(AMZN.ret,WMT.ret)), FUN=cor.fun, width=30,
                  by.column=FALSE, align="right")
par(mfrow=c(2,1))
plot(roll.cov3, main="30-day rolling covariances Amazon-Walmart",
    ylab="covariance", lwd=2, col="blue")
```

```
grid()
abline(h=cov(logreturn)[2,3], lwd=2, col="red")
plot(roll.cor3, main="30-day rolling correlations Amazon-Walmart",
     ylab="correlation", lwd=2, col="blue")
grid()
abline(h=cor(logreturn)[2,3], lwd=2, col="red")
#n)
# DCC estimation
# univariate normal GARCH(1,1) for each series
garch11.spec = ugarchspec(mean.model = list(armaOrder = c(0,0)),
                          variance.model = list(garchOrder = c(1,1),
                                                model = "sGARCH"),
                          distribution.model = "norm")

# dcc specification - GARCH(1,1) for conditional correlations
dcc.garch11.spec = dccspec(uspec = multispec( replicate(3, garch11.spec) ),
                           dccOrder = c(1,1),
                           distribution = "mvnorm")
dcc.garch11.spec

dcc.fit = dccfit(dcc.garch11.spec, data = logreturn)
coef(dcc.fit)
names(dcc.fit)
class(dcc.fit)
slotNames(dcc.fit)
names(dcc.fit@mfit)
names(dcc.fit@model)
likelihood(dcc.fit)
# Scatter plots:
par(mfcol=c(1,3))
```

```
plot( coredata(EBAY.ret), coredata(AMZN.ret.clean), xlab="EBAY", ylab="AMZN",
    type="p", pch=16, lwd=2, col="blue")
abline(h=0,v=0)
mtext("Scatter plot of Ebay and Amazon",side=3,cex=0.8)
plot( coredata(EBAY.ret), coredata(WMT.ret), xlab="EBAY", ylab="WMT",
    type="p", pch=16, lwd=2, col="blue")
abline(h=0,v=0)
mtext("Scatter plot of Ebay and Walmart",side=3,cex=0.8)
plot( coredata(AMZN.ret), coredata(WMT.ret), xlab="AMZN", ylab="WMT",
    type="p", pch=16, lwd=2, col="blue")
abline(h=0,v=0)
mtext("Scatter plot of Walmart and Amazon",side=3,cex=0.8)
ts.plot
#o)
par(mfcol=c(1,3))
plot(dcc.fit, which=3, series=c(1,2))
plot(dcc.fit, which=3, series=c(1,3))
plot(dcc.fit, which=3, series=c(3,2))
plot(dcc.fit, which=4, series=c(1,2))
plot(dcc.fit, which=4, series=c(1,3))
plot(dcc.fit, which=4, series=c(3,2))
lambda = 0.94
cov.ewma = covEWMA(as.data.frame(logreturn), lambda=lambda)
cor.ewma = covEWMA(as.data.frame(logreturn), lambda=lambda,return.cor=TRUE)
head(cov.ewma)
dim(cov.ewma)
# Extract conditional variance and correlation
# conditional variance
EBAY.AMZN.cond.cov = cov.ewma[,1,2];
# Plots
```

```
par(mfrow=c(2,1))
plot(x=time(as.zoo(logreturn)), y=cov.ewma[,1,2],
    type="l", xlab="Time", ylab="Covariance", lwd=2, col="blue",
    main="EWMA Covariance between EBAY and AMZN");
grid()
abline(h=cov(logreturn)[1,2], lwd=2, col="red")

plot(x=time(as.zoo(logreturn)), y=cov.ewma[,1,3],
    type="l", xlab="Time", ylab="Covariance", lwd=2, col="blue",
    main="EWMA Covariance between EBAY and WMT");
grid()
abline(h=cov(logreturn)[1,3], lwd=2, col="red")

plot(x=time(as.zoo(logreturn)), y=cov.ewma[,2,3],
    type="l", xlab="Time", ylab="Covariance", lwd=2, col="blue",
    main="EWMA Covariance between WMT and AMZN");
grid()
abline(h=cov(logreturn)[2,3], lwd=2, col="red")
# PLOT EWMA CORRELATION
plot(x=time(as.zoo(logreturn)), y=cor.ewma[,1,2],
    type="l", xlab="Time", ylab="Correlation", lwd=2, col="blue",
    main="EWMA Correlation between EBAY and AMZN");
grid()
abline(h=cor(logreturn)[1,2], lwd=2, col="red")

plot(x=time(as.zoo(logreturn)), y=cor.ewma[,1,3],
    type="l", xlab="Time", ylab="Correlation", lwd=2, col="blue",
    main="EWMA Correlation between EBAY and WMT");
grid()
abline(h=cor(logreturn)[1,3], lwd=2, col="red")
```

```
plot(x=time(as.zoo(logreturn)), y=cor.ewma[,2,3],
    type="l", xlab="Time", ylab="Correlation", lwd=2, col="blue",
    main="EWMA Correlation between WMT and AMZN");
grid()
abline(h=cor(logreturn)[2,3], lwd=2, col="red")
#p)
dcc.fcst = dccforecast(dcc.fit, n.ahead=22)
plot(dcc.fcst)
fitted(dcc.fcst)
sigma(dcc.fcst)
plot(dcc.fcst, which=3, series=c(1,2))
plot(dcc.fcst, which=3, series=c(1,3))
plot(dcc.fcst, which=3, series=c(3,2))
plot(dcc.fcst, which=4, series=c(1,2))
plot(dcc.fcst, which=4, series=c(1,3))
plot(dcc.fcst, which=4, series=c(3,2))
```