A study of time series modeling of NFLX returns

Brennan Hall

University of California Santa Barbara

PSTAT 274

December 9, 2018

Abstract

This paper investigates various ways to model Netflix (NFLX) stock price using time series techniques. A model is fit to historical daily price data[1] from 2017-1-3 to 2018-10-28 and its predictive power is tested on the following month, 2018-10-29 to 2018-11-30. An initial modeling[2] attempt was to consider a basic Autoregressive Integrated Moving-Average (ARIMA) model but was quickly seen as inadequate due to the constant variance assumption in the ARIMA model so an ARMA+GARCH (Generalized Autoregressive Conditional Heteroscedastic) model is considered for the growth rate of the stock price to account for this feature, resulting in a more accurate model in terms of the two forecasts's error measurements. Lastly, a spectral analysis is conducted to consider whether seasonal or cyclic periods may exist in the time series. No seasonality is found in the price process; however, apparent peaks and troughs in the volatility process's spectral density indicates some periodicity.

[1]Data accessed via Yahoo! Finance API

[2]all code was written and implemented in the R programming language, see Appendix B for select code chunks.

1 Introduction

To invesatigate potential models for forecasting the NFLX stock price, we gather recent historical price data from 2017-01-03 to 2018-11-29. Only the daily closing price is chosen as the process used for modeling as a simplification to avoid intraday seasonality. Understanding that stock price data, particularly in the technology sector, is highly volatile and frequent to significant changes in short time frames, the goal of the model is restricted to forecasting next-month trend only. So the data is subset into a training set and test set - the test set including the last month's worth of price data (23 days of trading).

Figure 1: Daily closing price for NFLX

In its raw form, stock price data is typically not a stationary process and is difficult to fit a time series model to so transformations will be applied to the process in order to accomodate these data features. These transforms will be addressed and briefly summarized as they are introduced throughout the paper. ARIMA modeling techniques will be applied in the first portion of the paper (Sections 2-3), which will be improved upon by adding a GARCH component in subsequent sections (Sections 4-5).

The sections of the paper will be structured as follows: Section 2 will cover the process of model estimation using ARIMA techinques. In Section 3, forecasting and error analysis will be conducted using the ARIMA model. Sections 4 and 5 will be analogous to sections 2 and 3 but in relation to ARMA+GARCH modeling. Section 6 covers a short spectral analysis of the time series. Section 7 will conclude the paper with final thoughts and remarks on the models and modeling process.

2 ARIMA Model Estimation

2.1 Exploratory data analysis

Figure 1 displays the entire pricing process, call it S_t where t is the time index, that is to be analyzed. We note a general upward trend that then begins to decline by mid-2018. Additionally, there is evident volatility change throughout the process, particularly after January 2018 as the process has major upward and downward movements in relation to prices prior to 2018. This is an intial indication to implementing a GARCH model, but for the first portion of the paper, a Box-Cox transformation will be used as a variance stabilizing transform. Another feature of S_t is that it does not appear stationary due to its trend component so we will attempt to use a differencing operator to adjust this. Before moving to the actual model estimation, the next subsection reviews the ARIMA model and introduces some useful notation.

2.2 The ARIMA model

A time series, Y_t, is modeled by an ARIMA(p, d, q) model if it has the form described in the following:

Let Y_t be a non-stationary time series, and $X_t = (1 - B)^d Y_t$, where B is the backshift operator. Then X_t is stationary and of the form

$$X_t - \phi_1 X_{t-1} - \ldots - \phi_p X_{t-p} = a_t + \theta_1 a_{t-1} + \ldots + \theta_q a_{t-q}$$
$$\Phi_p(B) Y_t = \Theta_q(B) a_t$$

where $a_t \overset{\text{i.i.d}}{\sim} WN(0, \sigma^2)$ is a white noise process with constant variance, ie. X_t follows an ARMA(p, q) model.

For this study, Y_t will be a (soon-to-be-explained) transformation of S_t and $X_t = (1 - B)Y_t$ will be the lag-1 differenced process, and it is of interest to find optimal estimates of ϕ and θ under the stationary and invertibility constraints to provide meaningful and accurate forecasts. Towards the definition of stationarity and invertibility, these conditions are verified by the roots of $\Phi_p(B)Y_t = 0$ and $\Theta_q(B)a_t = 0$ being outside the unit circle.

2.3 Data preparation

Prior to fitting a model, certain transformations to the process S_t are necessary. A Box-Cox transformation is attempted here in an attempt to make the transformed data have constant variance. Figure 2 indicates an inverse transform will be best. Applying a lag-1 difference to this transformed process yields a mean zero process with stable (relative to the original process) variance.

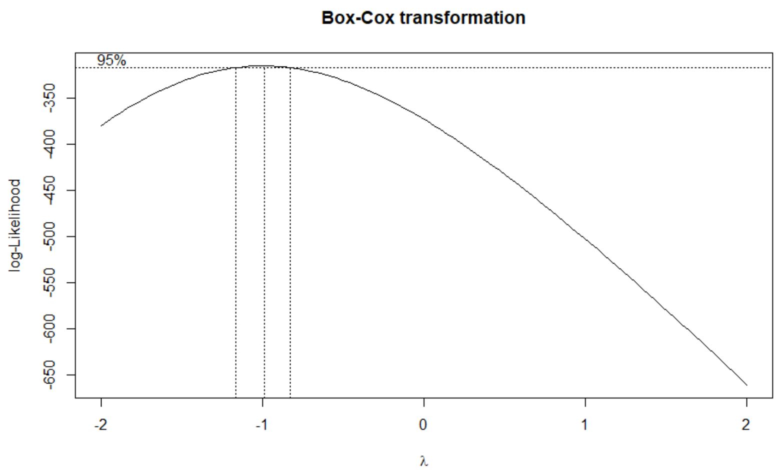

Figure 2: Log-likelihood of Box-Cox λ transformation

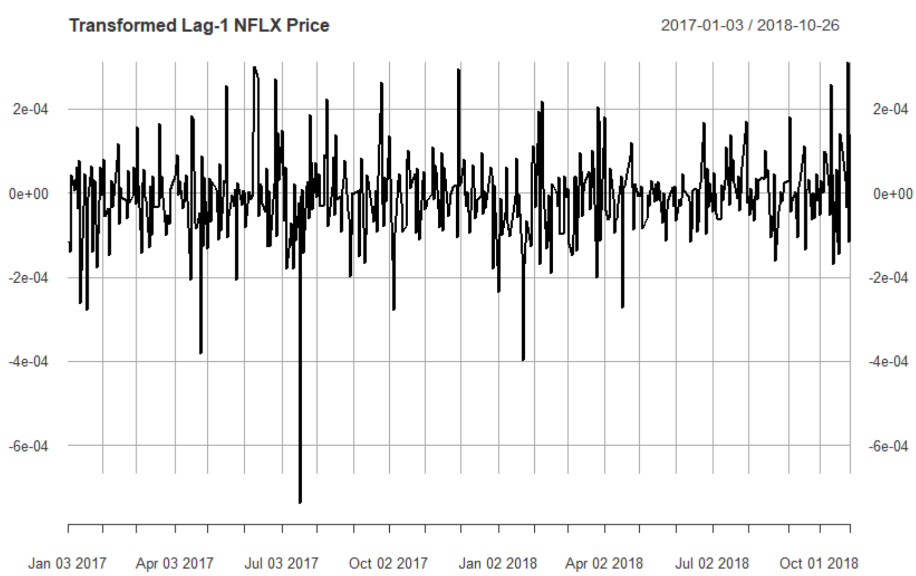

Figure 3: NFLX price after Box-Cox transformation and differencing

It is clear from Figure 3 that there is still some time dependence for the variance as there are periods of high and low volatility; however, this is as close to a stationary process that can be achieved under the ARIMA framework. Looking at the sample ACF and sample PACF plots of this transformed process (from now on referred to as Y_t) in Figure 4, we can verify Y_t as a suitable white noise process. Thus the process S_t can be fit to an ARIMA model.

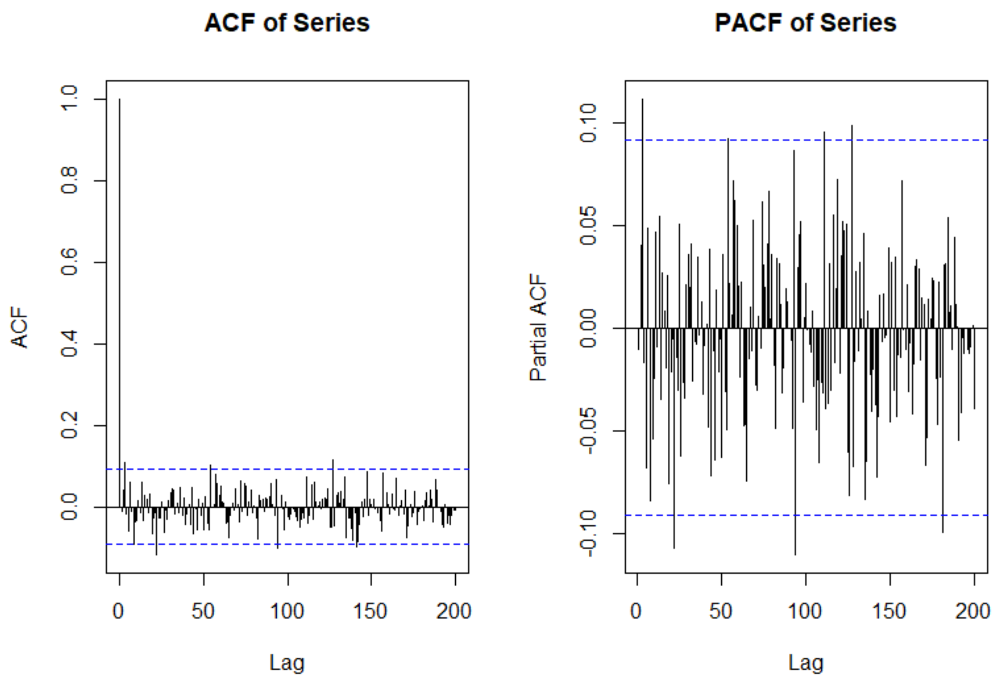

Figure 4: ACF and PACF plots of Y_t similar to a white noise process

2.4 Fitting the model

To fit the ARIMA model, a stepwise approach is used to identify the optimum orders p and q, assuming $d = 1$ as this has already been determined to provide a stationary process. The *auto.arima*() function from the *forecast* package is utilized in this context with the selection method chosen for the stepwise algorithm to be AIC and the MLE method for estimating the parameters once the orders are chosen. As seen in Appendix Table 1, the model selected is an ARIMA(3,1,0) model of the form

$$Y_t + -0.0056Y_{t-1} - 0.0494Y_{t-2} - 0.1234Y_{t-3} = a_t$$

Before proceeding to forecasting, diagnostics are performed on the model to determine whether the model assumptions are met, and, if not, where to expect shortcomings in our forecasts. To test for serial correlation and independence of the residuals, a Ljung-Box test is performed on the residuals of the ARIMA model. The null hypothesis of the Box-Ljung test is that the error terms of the model are uncorrelated and uses the χ^2 distributed test statistic,

$$Q = n(n+2)\sum_{k=1}^{K}(n-k)^{-1}\hat{\rho}_k^2$$

where K is the maximum lag length (3 in this case), n is the number of observations, and $\hat{\rho}_k$ is the sample ACF at lag k. The result of the Ljung-Box test yields a p-value of 0.7462, indicating no signifcant evidence to reject the null hypothesis. This is the only assumption that is met for this model, however.

Plotting the standardized residuals against the fitted values, as in Figure 5, indicates a pattern amongst residuals, violating the homoscedacticity assumption of the model. This is most likely a result of the time dependency of the variance noticed in the process S_t that was not completely remedied with the Box-Cox transformation. Lastly, conducting a Shapiro-Wilks test and plotting a Q-Q plot of the residuals[3], indicates the residuals do not follow a Gaussian white noise process. Since the parameter estimates were found via MLE, the estimates' error terms are likely larger than those reported. As a result, the forecasts are likely to be inaccurate and report smaller confidence intervals than appropriate.

3 ARIMA Forecasting and error analysis

Despite the likely inaccuracies, it is still reasonable to attempt to forecast the process since the process is seen to be invertible and stationary per the root requirements mentioned in Section 2.2. The AR roots are outside the unit circle (or inverse AR roots inside the unit circle as shown in Figure 7), and since there are no MA terms, the MA root equation is automatically satisfied. Therefore, the process is both stationary and invertible.

To conduct the forecast, the $forecast()$ function from the *forecast* package was used with a 23-step forecasting period (2018-10-26 to 2018-11-29). Figure 8 shows the plot of the original stock price with the forecasted prices along with 80- and 95% confidence intervals while Figure 9 shows the error between the point forecasts and actual prices from the test data set. Furthermore, Table 2 presents additional error

[3]See Appendix Figure 1

Figure 5: Standardized Residuals vs Fitted plot indicates heteroscedasticity amongst residuals

measurements based on the test data set. These measures will be used to quantify the improvements that including a GARCH model will provide. All together, it seems evident that the ARIMA model performs quite poorly at forecasting the NFLX stock price.

SSE	RMSE	MAE	MAPE
9455.761	20.28	17.12	0.06

Table 1: ARIMA(3,1,0) model forecast error measurments

4 ARMA+GARCH Model Estimation

Recognizing some of the shortcomings of using only an ARIMA model to fit the stock price, the more standard approach in financial literature is considered by modeling the logarithmic daily returns of the stock price. These returns are particularly useful as they are good approximations to the discrete returns of the stock (typically considered as the growth rate of the stock or the interest rate of return for investing in the stock). The logarithmic returns are defined for the process S_t as $Y_t = \log(\frac{S_t}{S_{t-1}})$. X_t is then

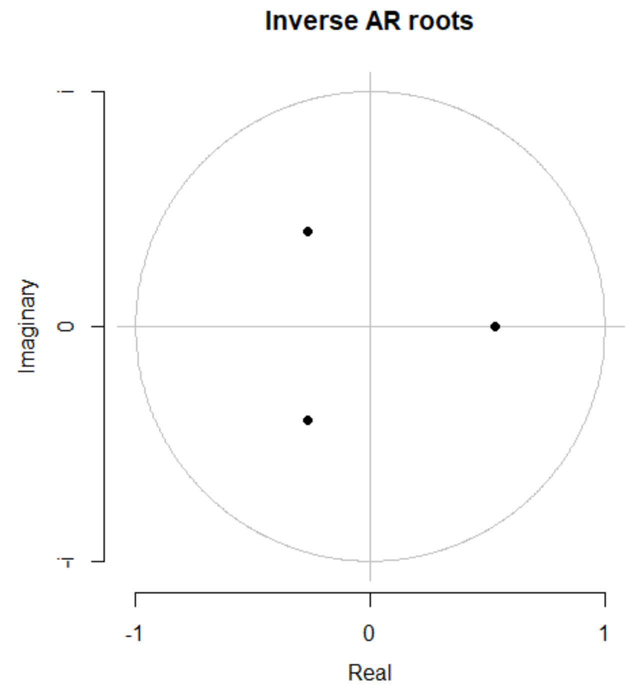

Figure 6: Inverse AR roots of ARIMA(3,1,0) model

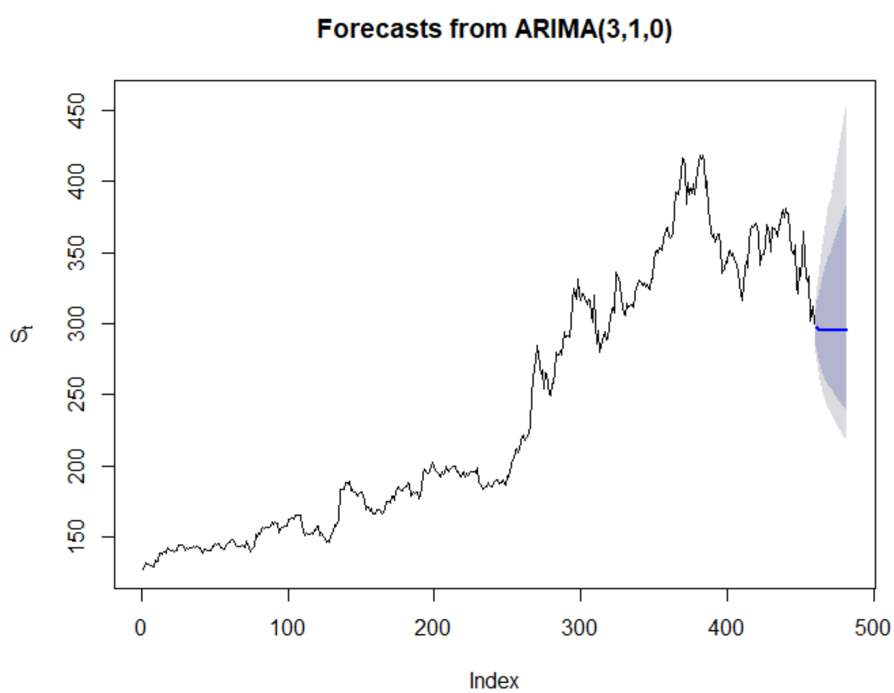

Figure 7: Forecast point estimates with 80- and 90% confidence intervals

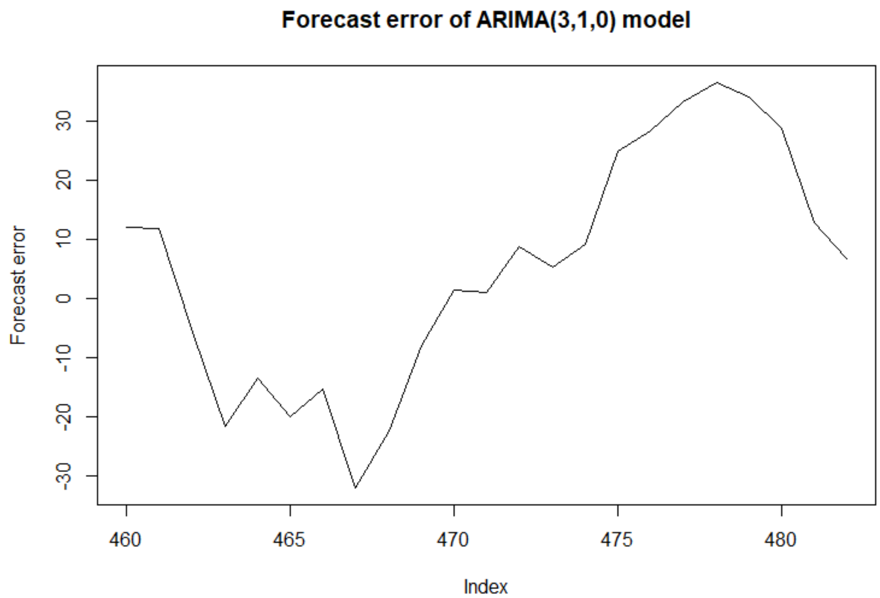

Figure 8: Forecast point estimate errors

modeled as a variance process (GARCH component) with a potential mean process (ARMA component). The NFLX data is presented in Y_t form in Figure 9.

4.1 The ARMA+GARCH model

To establish notation, a brief summary of the general form of the model follows. Let Y_t be a time series, then it has an ARMA$p, q)$ + GARCH(m, s) structure if it is of the form:

$$\begin{aligned}
\Phi(B)Y_t &= \delta + \Theta(B)a_t \\
a_t &= \sigma_t \epsilon_t \\
\sigma_t^2 &= \omega + \sum_{j=1}^{m} \alpha_j a_{t-j}^2 + \sum_{j=1}^{s} \beta_j \sigma_{t-j}^2
\end{aligned}$$

where ϵ_t is a white noise process with unit variance, and the parameters are subject to non-negativity constraints (since variance is non-negative) and stationarity constraints:

$$\omega > 0,\ \alpha_1, \ldots, \alpha_{m-1} \geq 0, \alpha_m > 0,\ \beta_1, \ldots, \beta_{s-1} \geq 0, \beta_s > 0$$

$$(\alpha_1 + \ldots + \alpha_m) + (\beta_1 + \ldots + \beta_s) < 1.$$

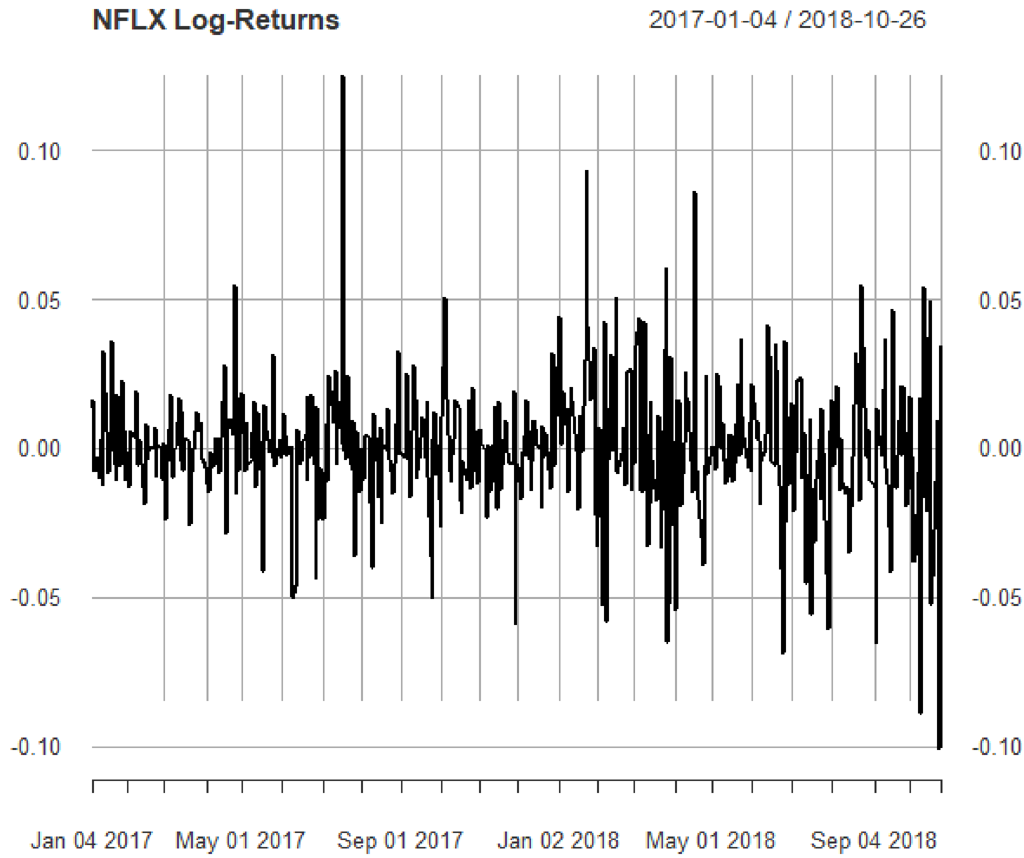

Figure 9: Logarithmic daily returns of NFLX stock price

The intent of the GARCH component is to model the variance of the process so that, conditionally, the variance is a time-dependent, non-constant process such that the unconditional process is stationary with constant variance.

The paraemter estimation is done via maximum-likelhood estimation, and so the conditional distribution of the variance process will need to be specified. Because GARCH models have a mixture distribution due to the time dependent variance, the marginal distribution will have a heavy-tailed distribution, making a Gaussian white noise inadepquate. For this reason, the Skewed General Error Distribution (sGED) is used.

4.2 Fitting the model

Similar to the method of fitting the ARIMA model, an exhaustive, stepwise algorithm approach is used with minimum AIC being used as the selection criteria. To implement this approach in R, the function *garchAuto()*[2] was used. This function exhaustively fits all combinations of a pre-specified set of model orders – in this case 0 through 5 for the ARMA orders and 0 through 2 for the GARCH orders – then selects the model with the smallest AIC value. The resulting optimum model is

$$Y_t - 0.818Y_{t-1} - 0.9411Y_{t-2} = a_t + 0.738a_{t-1} + 0.8807a_{t-2} - 0.06911a_{t-3} + 0.0036a_{t-4}$$
$$a_t = \sigma_t\epsilon_t$$
$$\sigma_t^2 = 8.872\text{e-}06 + 0.07605a_{t-1}^2 + 0.9143\sigma_{t-1}^2$$

where ϵ_t is a sGED white noise process with zero mean, unit variance, skew parameter $\xi = 1.061$, and shape parameter $\nu = 1$.

Some diagnostic tests that are performed on the residuals of the fitted model are the Engle's ARCH test and Ljung-Box test. All tests results have non-significant p-values so it can be concluded that the squared residual is a sGED white noise process and conditionally autocorrelated as desired for the model.

Test	Residual	Statistic	p-value
Ljung-Box Test	R	14.09163	0.1688531
Ljung-Box Test	R	19.79026	0.180127
Ljung-Box Test	R	23.08273	0.2847504
Ljung-Box Test	R^2	2.448689	0.9916025
Ljung-Box Test	R^2	6.101127	0.9779985
Ljung-Box Test	R^2	9.638658	0.9742609
LM Arch Test	R	3.425025	0.9917268

Table 2: Diagnostic test results

5 ARMA+GARCH Forecasting and error analysis

With the better fit by adding the GARCH component, it is expected to have much more accurate forecasts and smaller error measurements. First, it is verified that the fitted process is stationary and inveritble via the roots of the AR and MA equations being strictly outside the unit cirlce and the sum of the GARCH parameters being strictly less than one.

The 23-step forecast is shown in Figure 11. The plotted process is the logarithmic return process along with the 95% confidence intervals for the forecasted points. Considering the forecast errors plotted in Figure 12 and the error measurements

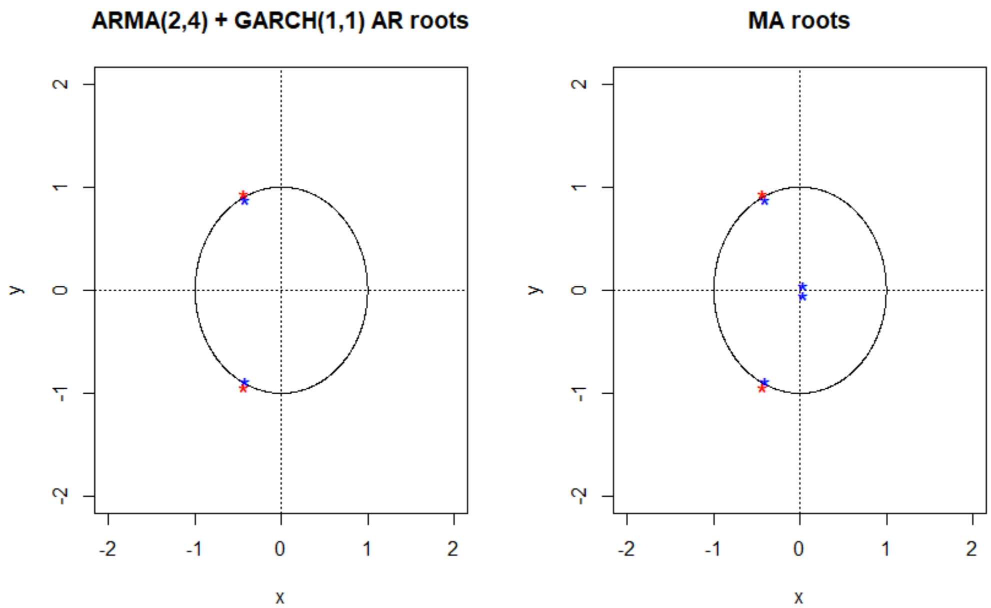

Figure 10: AR and MA equation roots indicating stationarity of ARMA+GARCH model

presented in Table 2, it is clear that the ARMA+GARCH model performs much better at modeling the

	SSE	RMSE	MAE	MAPE
ARMA+GARCH	0.0257	0.0334	0.0280	0.9943
ARIMA	9455.761	20.28	17.12	0.06

Table 3: ARMA(2,4)+GARCH(1,1) and ARIMA(3,1,0) forecast error measurements

6 Spectral analysis

An additional consideration to modeling a time series is to model in its frequency domain rather than its time domain. In this regard, the spectral density of the time series is found with the *spectrum*() function in R.

For financial data, it should be expected that no regular seasonal cycles could persist in the price process since the stock price may move up or down at any particular seasonal event; for example, quarterly reports, market announcements, or dividend distributions. However, there would be some intuitive reason for seasonal cycles around the same events as above for the volatility process since as an event such as quarterly reports approach, volatility typically increases then decreases once the

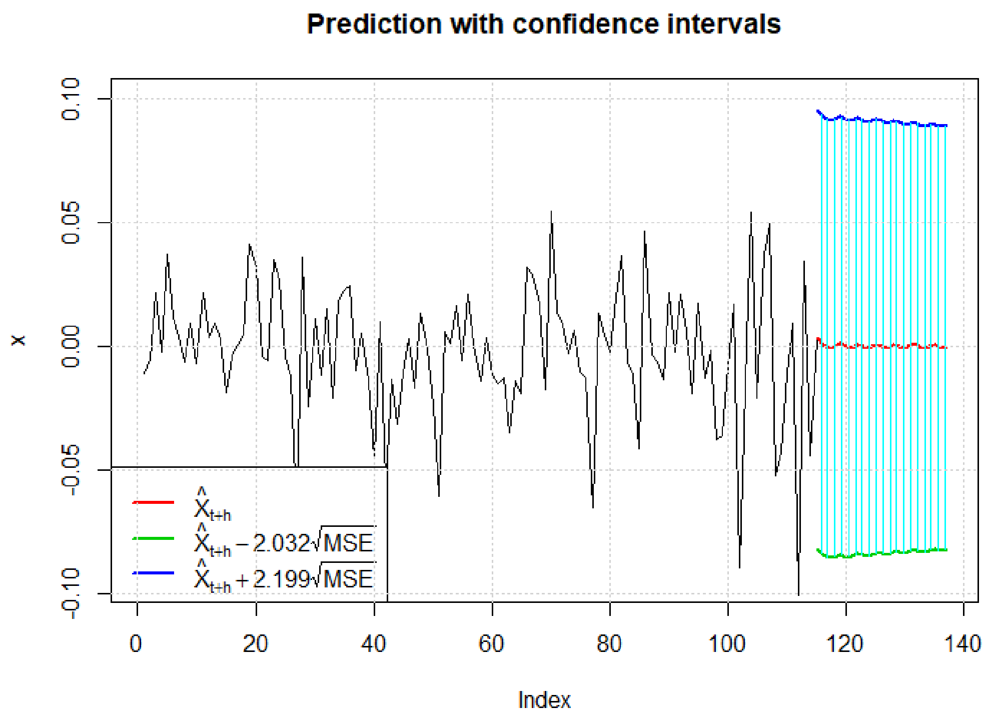

Figure 11: Forecast points for ARMA+GARCH with 95% confidence intervals

event has passed. With this in mind, the following plots in Figures 12 and 13 should confirm these intuitions. The spectral density of the pricing process indicates no cylic behavior while the spectral density of the logarithmic returns has at least three significant peaks.

The above analysis is performed using the *spec.mtm*() function in the *multitaper* package.

7 Concluding Thoughts

The ARIMA model is seen to be an inadequate model to fit and forecast stock price data due to its restrictive assumption on the variance of the error. It was also attempted to model the growth rate of NFLX with the ARIMA model, but the results of the optimum model performed even worse than the model presented in this paper so it was not compared to the optimum ARMA+GARCH model. Adding the GARCH component to model the volatility of the growth rate process provides a model fit with accurate and significant estimates along with small forecasting errors as seen by the error measurements conducted on the test data set.

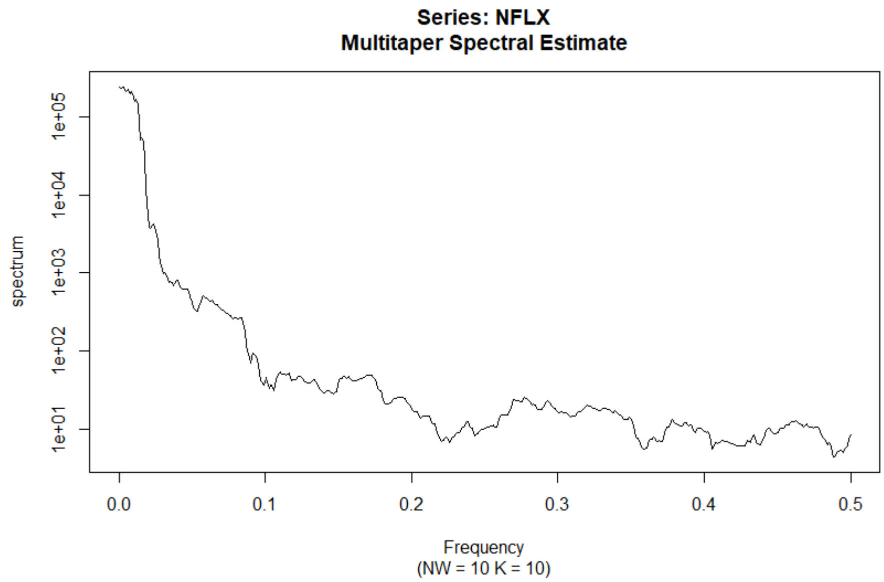

Figure 12: Spectral density of NFLX stock price

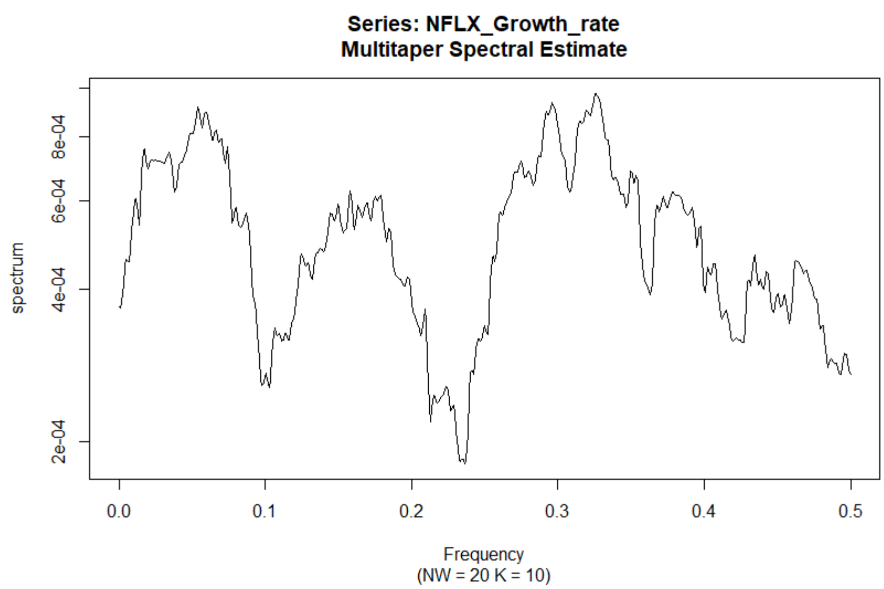

Figure 13: Spectral density of NFLX growth rate

References

[1] Peter Brockwell, Richard Davis, *Introduction to Time Series and Forecasting*, Third Edition, Springer 2016.

[2] Ivan Popivanov, garchAuto.R, 2012, https://gist.github.com/ivannp/5198580.

[3] Panayiotis Theodossiou, *Skewed Generalized Error Distribution of Financial Assets and Option Pricing*, Multinational Finance Journal, 2015.

[4] Ruey Tsay, *Analysis of Financial Time Series*, Wiley, 2002.

A Appendices

A.1 Plots and Tables

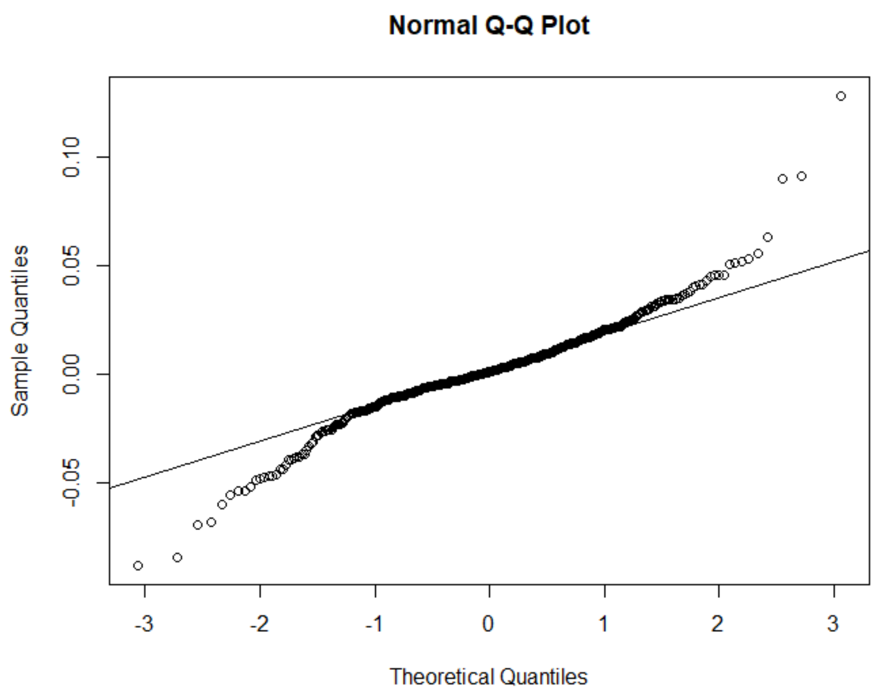

Figure 1: Normal Q-Q plot of residuals for ARIMA(3,1,0) model

	ϕ_1	-0.0056		
		(0.0465)		
	ϕ_2	0.0494		
		(0.0466)		
	ϕ_3	0.1234		
		(0.0471)		
	AIC	-7147.18		
	Log Likelihood	3577.59		
	RMSE	MAE	MAPE	MASE
Training set	6.4382	4.1291	1.5923	1.0111

Table 1: ARIMA(3,1,0) parameter estimates (with standard errors)

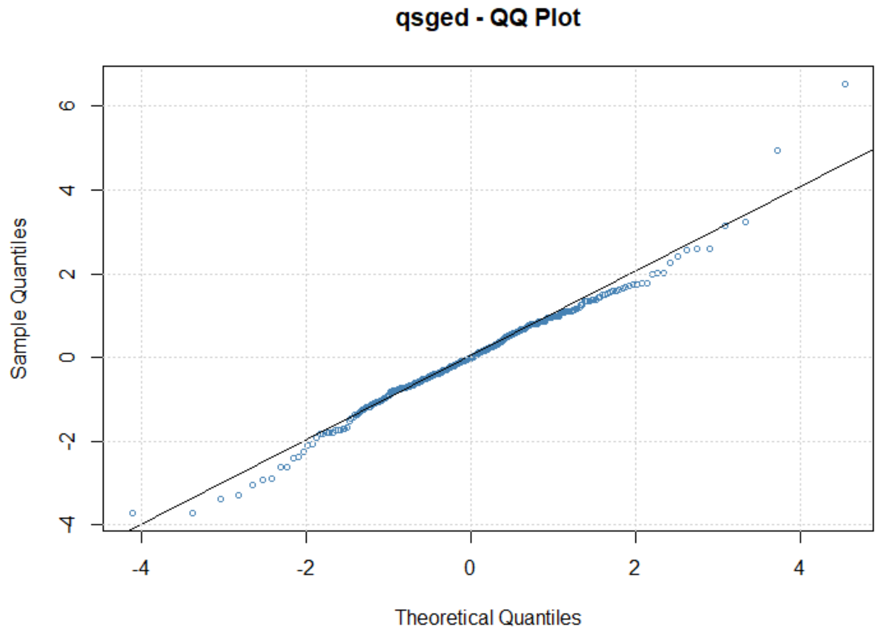

Figure 2: Skewed Generalized Error Distribution Q-Q plot of residuals for ARMA(2,4)+GARCH(1,1) model

	ϕ_1	$-8.180e-01$		
		$(1.487e-02)$		
	ϕ_2	$-9.411e-01$		
		$(1.496e-02)$		
	θ_1	$7.380e-01$		
		$(1.603e-02)$		
	θ_2	$8.807e-01$		
		$(1.690e-02)$		
	θ_3	$-6.911e-02$		
		$(1.214e-02)$		
	θ_4	$3.600e-03$		
		$(1.467e-02)$		
	ω	$8.872e-06$		
		$(6.977e-06)$		
	α_1	$7.605e-02$		
		$(2.448e-02)$		
	β_1	$9.143e-01$		
		$(2.784e-02)$		
	AIC	-4.9786		
	Log Likelihood	1151.099		
	RMSE	MAE	MAPE	MASE
Training set	0.0226	0.0154	1.5923	1.0111

Table 2: ARMA(2,4)+GARCH(1,1) parameter estimates (with standard errors)

A.2 Selected R code

```
##### Import data
getSymbols("NFLX", from = "2017-01-03", to = "2018-10-26")
NFLX_past <- NFLX$NFLX.Close
logrets_past <- logret(NFLX_past, demean = TRUE)

###################
# ARIMA Modeling
###################
## get Box-Cox transformation
boxcox(NFLX_past ~ as.numeric(1:n_past))
nflx_past_inv1 <- diff(NFLX_past^(-1))

acf(nflx_past_sqr1[-1], lag.max = 200, main="ACF of Series")
```

```
pacf(nflx_past_sqr1[-1], lag.max = 200, main="PACF of Series")
### ACF/PPACF may indicate ARIMA(3,1,0) for inverse transform
## but nothing conclusive -> use stepwise selection


##### Fit model
arma_fit_inv <- auto.arima(NFLX_past, d=1, max.order = 7, ic = "aicc", trace = T,
lambda=-1, stepwise = F, num.cores = NULL, allowdrift=F, allowmean=F)

## plot AR/MA roots
plot(arma_fit_inv)


##### Diagnostics tests
Box.test(residuals(arma_fit_sqrt_past), type = "Ljung") ## passes independence
plot(sqrt(abs(residuals(arma_fit_inv))) ~ fitted(arma_fit_inv)) # indicates NcV
shapiro.test(residuals(arma_fit_inv)) ## fails normality
qqnorm(arma_fit_inv$residuals)
qqline(arma_fit_inv$residuals)


##### Forecasting
forecast_arma_inv <- forecast(arma_fit_inv, h = 23)
er <- forecast_arma_inv$mean - as.numeric(NFLX_pres)
plot(forecast_arma_inv,xlab = "Index", ylab = expression(S[t]))
plot(er, ylab="Forecast error", xlab="Index",
                main="Forecast error of ARIMA(3,1,0) model")
## Error measurements
rmse_arma_sqr <- rmse(predicted = as.numeric(forecast_arma_sqrt_past$mean),
                actual = NFLX_pres)
mae_arma_sqr <- mae(predicted = as.numeric(forecast_arma_sqrt_past$mean),
                actual = NFLX_pres)
mape_arma_sqr <- mape(predicted = as.numeric(forecast_arma_sqrt_past$mean),
                actual = NFLX_pres)
sse_arma_sqr <- sse(predicted = as.numeric(forecast_arma_sqrt_past$mean),
                actual = NFLX_pres)
```

```
###################
# ARMA+GARCH Modeling
###################

##### Fit Model
best_fit_AIC <- garchAuto(data, cond.dists = "sged", trace = T, ic = "AIC")
plot(best_fit_AIC)
summary(best_fit_AIC)
predict(best_fit_AIC, n.ahead = 23, plot = T)

rmse(actual = logrets_past, predicted = best_fit_AIC@fitted)
mae(actual = logrets_past, predicted = best_fit_AIC@fitted)
mpe(actual = logrets_past, predicted = best_fit_AIC@fitted)
mape(actual = logrets_past, predicted = best_fit_AIC@fitted)
mase(actual = logrets_past, predicted = best_fit_AIC@fitted)

##### Diagnostics tests
res <- best_fit_AIC@residuals
plot(res^2, type='l', col='red')
par(mfrow=c(1,2))
Acf(res^2)
Pacf(res^2)
Box.test(res^2,lag=1, type="Ljung-Box", fitdf=0)

## Check stationarity/invertibility
coef(best_fit_AIC)

ar_coef_AIC <- coef(best_fit_AIC)[c(1:2)]
ma_coef_AIC <- coef(best_fit_AIC)[c(3:6)]
g_coef_AIC <- coef(best_fit_AIC)[c(8:9)]

par(mfrow=c(1,2))
## Stationarity of ARMA (passes)
ARroots_AIC <- polyroot(c(1,-ar_coef_AIC))
```

```
plot.roots(ma.roots = ARroots_AIC, main = "ARMA(2,4) + GARCH(1,1) AR roots")

## Invertibility of ARMA (passes)
MAroots_AIC <- polyroot(c(1,ma_coef_AIC))
plot.roots(ma.roots = MAroots_AIC, main = "MA roots")

## Stationarity of GARCH (passes)
sum(g_coef_AIC) < 1

##### Forecasting
forecast_garchA1_past <- predict(best_fit_AIC, n.ahead = 23, plot=T)
## Error measurements
test_res_gA1 <- as.numeric(forecast_garchA1_past$meanForecast) - logrets_pres[1:23]
plot(test_res_gA1)
rmse_gA1 <- rmse(predicted = as.numeric(forecast_garchA1_past$meanForecast),
                actual = logrets_pres)
mae_gA1 <- mae(predicted = as.numeric(forecast_garchA1_past$meanForecast),
                actual = logrets_pres)
mape_gA1 <- mape(predicted = as.numeric(forecast_garchA1_past$meanForecast),
                actual = logrets_pres)
sse_gA1 <- sse(predicted = as.numeric(forecast_garchA1_past$meanForecast),
                actual = logrets_pres)

##################
# Spectral analysis
##################
## Price process
spec.mtm(NFLX, k=10, nw=10, nFFT = "default", Ftest = TRUE, log="yes")
## Growth rate process
spec.mtm(NFLX_Growth_rate, k=10, nw=20, nFFT = "default",
        Ftest = TRUE, log="yes")
```