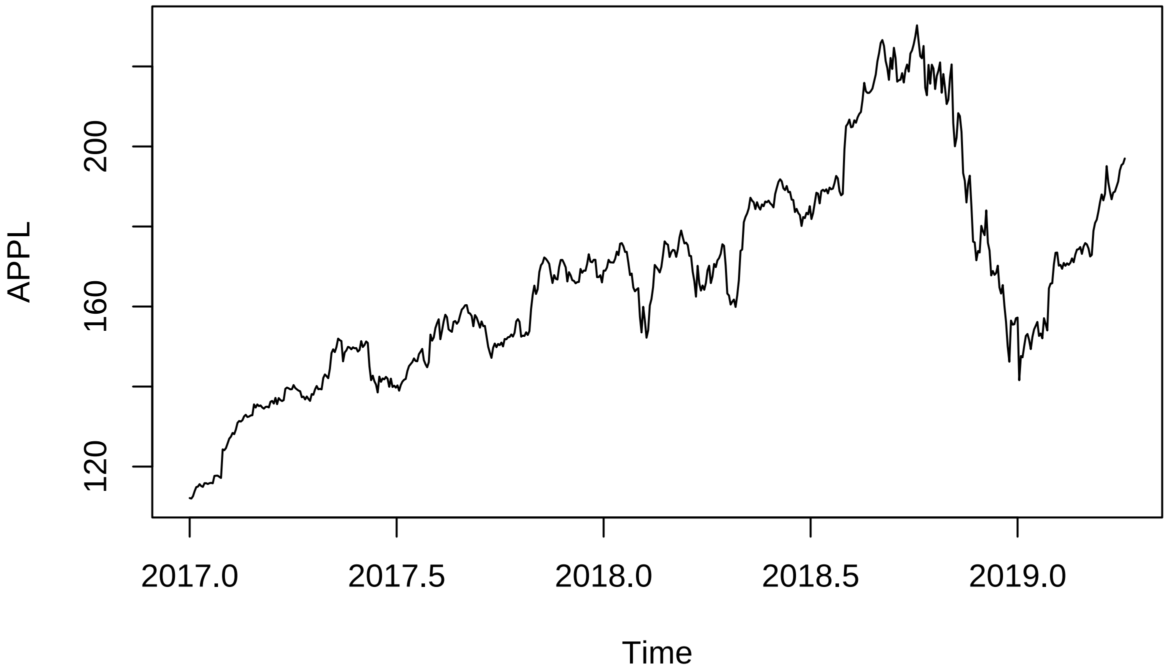
Modeling APPL Stock

```
load("~/Downloads/Stocks.RData")
dat <- Stocks
View(dat)
APPL <- dat[,1] #Extracts APPL as first time series
View(APPL)
```

The dataset chosen is the Stocks.RData file. Viewing the first time series, we have the daily adjusted closing prices from the beginning of 2017 until 2019.

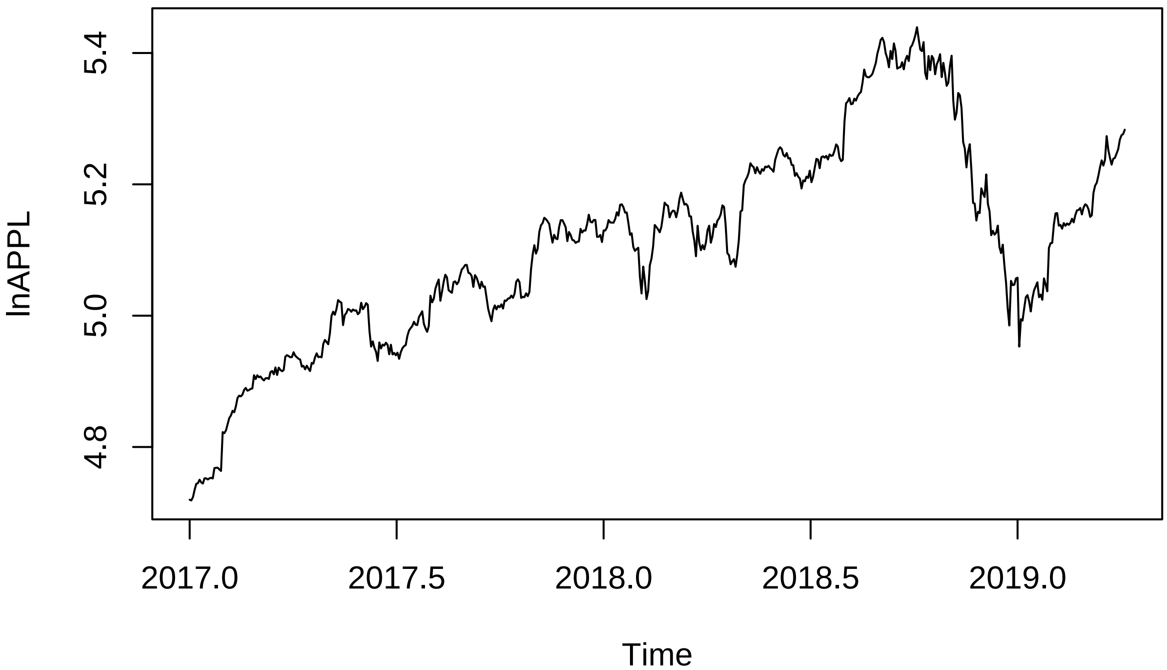
```
plot(APPL)
```



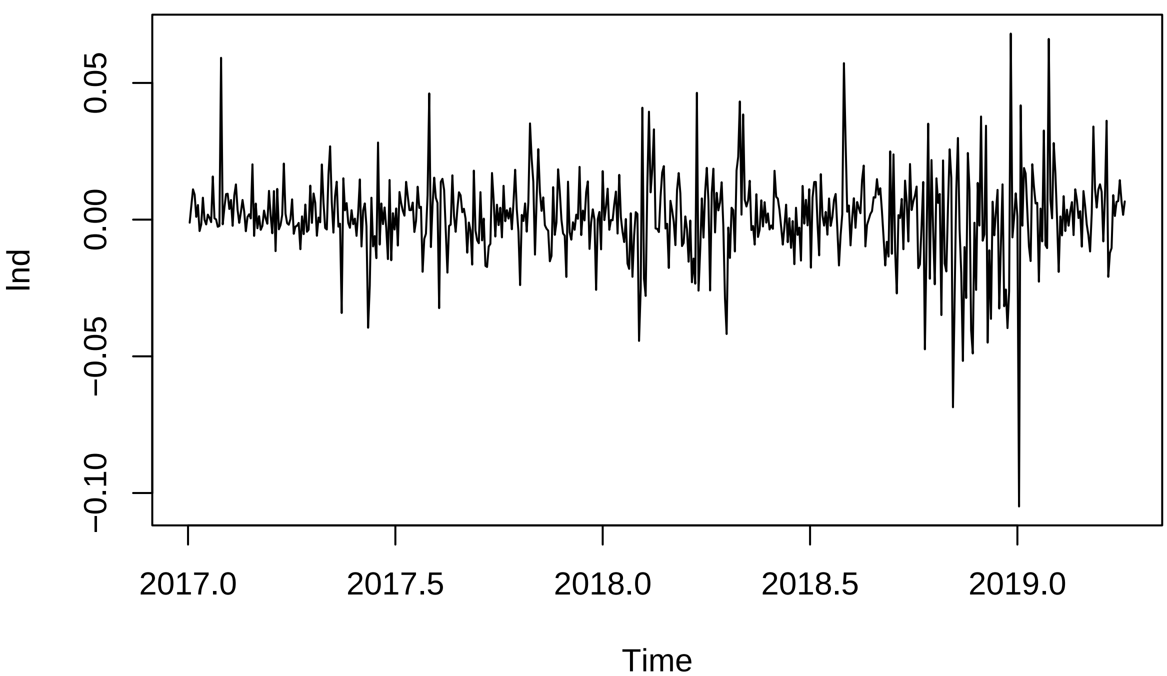
The time series appears to have a positve trend effect from the year 2017 until about 3/4 through 2018. At this point, the series' values declines abruptlu intil it starts to increase again at the beginning of 2019.

There also appears to be a quarterly effect where the closing prices at starting point and half point of the year are lower than the other points of the year. The variability before the time point 2018.5 appears to be relatively stable.

```
####Log transformation (for scaling)
lnAPPL <- log(APPL)
plot(lnAPPL)
```

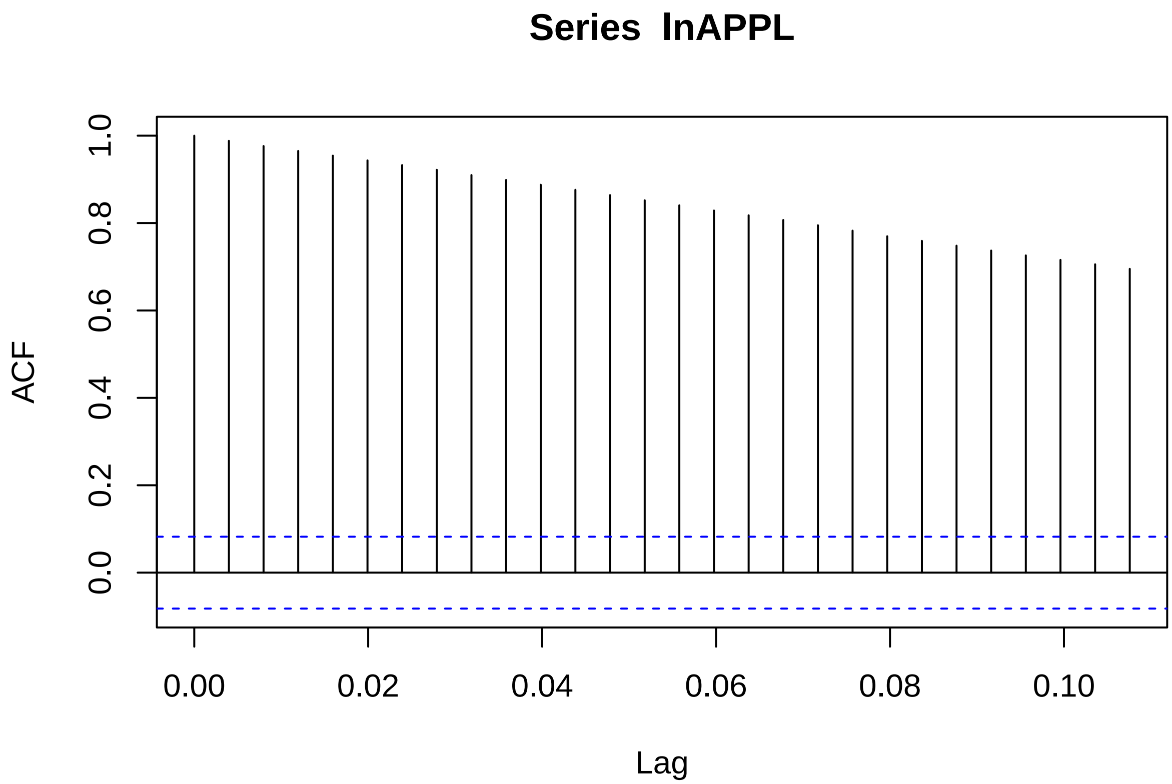



```
####Quarterly Growth rate of log
lnd <- diff(lnAPPL)
plot(lnd)
```

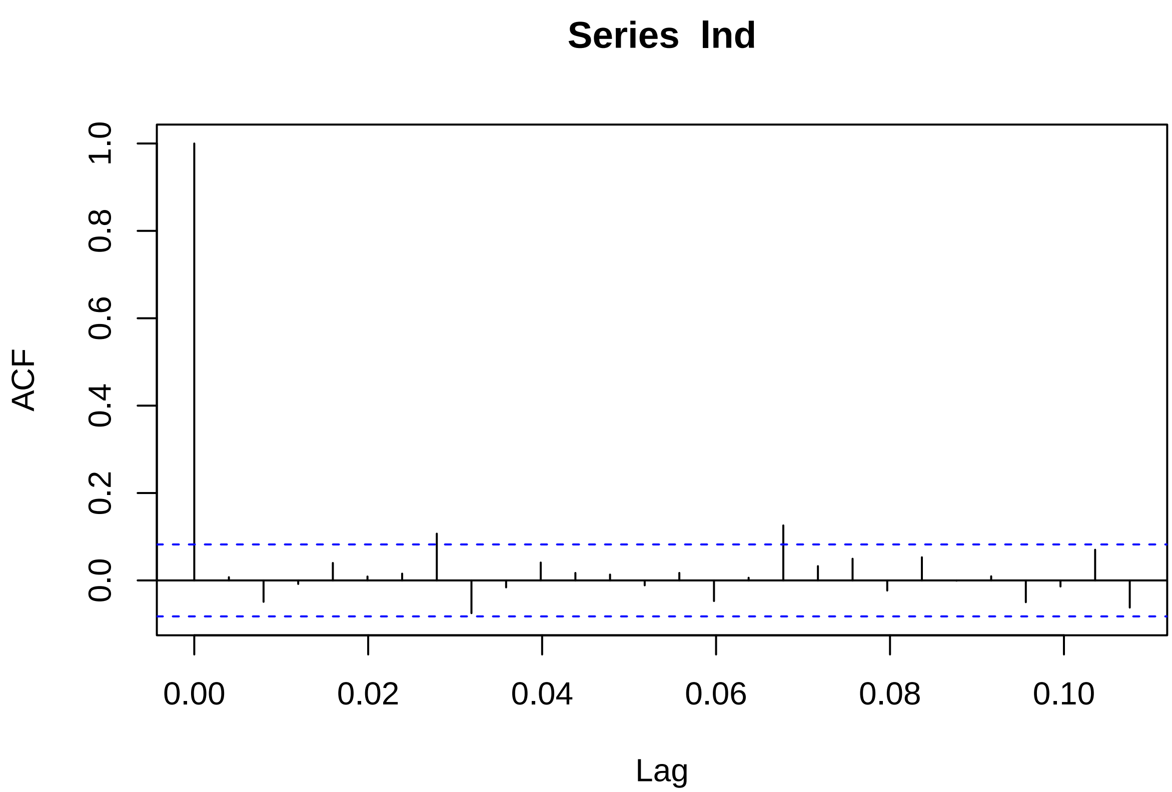


First, using log transformation, we are able to scale the data to a more manageable level. However, the overall behavior (trend features, varaibility, etc.) is preserved. Additionally, using the difference function, we can closely approximated the Quarterly Growth rate with first difference of the logged series.

```
acf(lnAPPL)
```

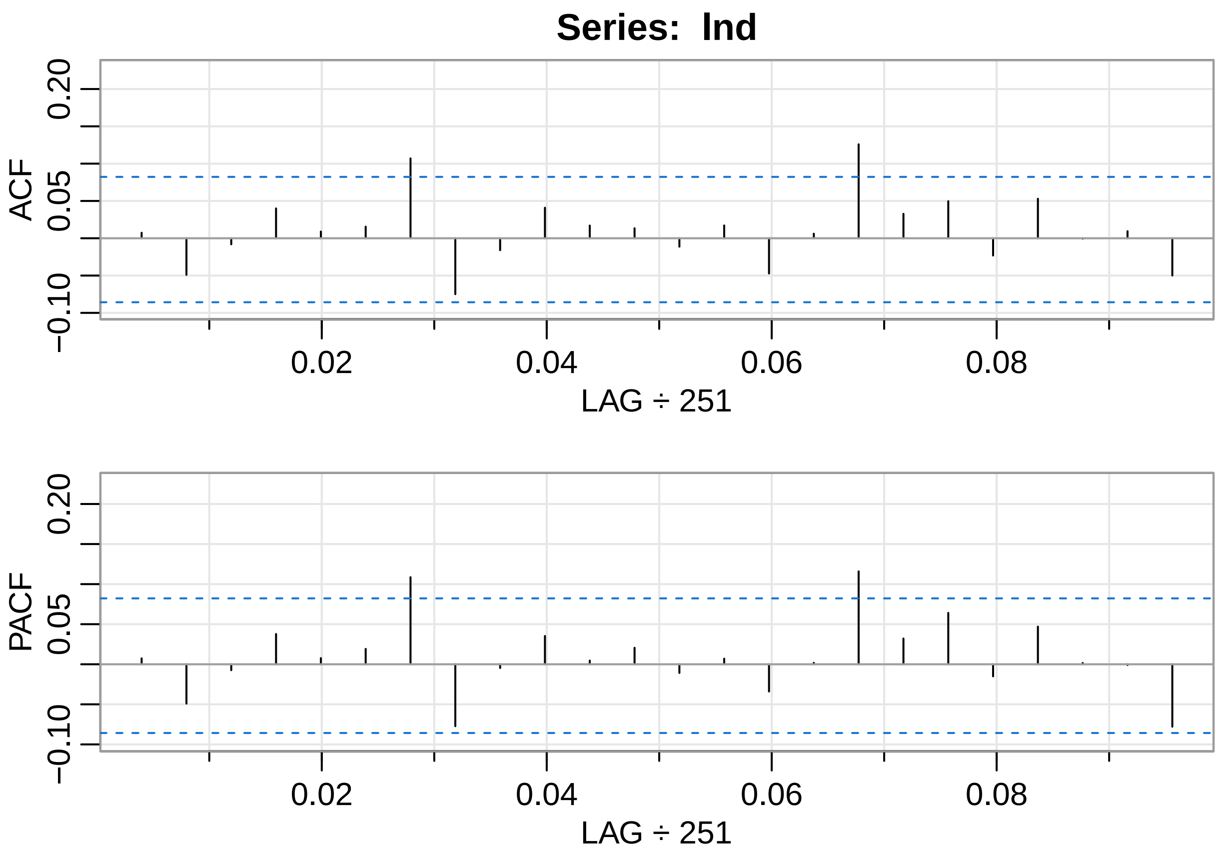



```
acf(lnd)
```



Differencing in the previous step seems to help create a series that looks stationary eliminating the obvious trend of the orignal data of Stocks. We can treat this as an ARIMA(p,1,q) model. There is no need to further difference because the ACF of the once differenced data drops to zero quickly suggesting it has the properties of a stationary time series.

```
library(astsa)
acf2(lnd,24)
```




```
##      [,1]  [,2]  [,3] [,4] [,5] [,6] [,7]  [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.01 -0.05 -0.01 0.04 0.01 0.02 0.11 -0.07 -0.02  0.04  0.02  0.01 -0.01
## PACF 0.01 -0.05 -0.01 0.04 0.01 0.02 0.11 -0.08  0.00  0.04  0.00  0.02 -0.01
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF   0.02 -0.05  0.01  0.13  0.03  0.05 -0.02  0.05     0  0.01 -0.05
## PACF  0.01 -0.03  0.00  0.12  0.03  0.06 -0.02  0.05     0  0.00 -0.08
```

The sample ACF and PACF are consistent with the data for APPL closing time stocks being either ARIMA(7,1,0) or ARIMA(0,1,7). I used 7 rather than 17 because this is quarterly data and too many lags are not reasonable.

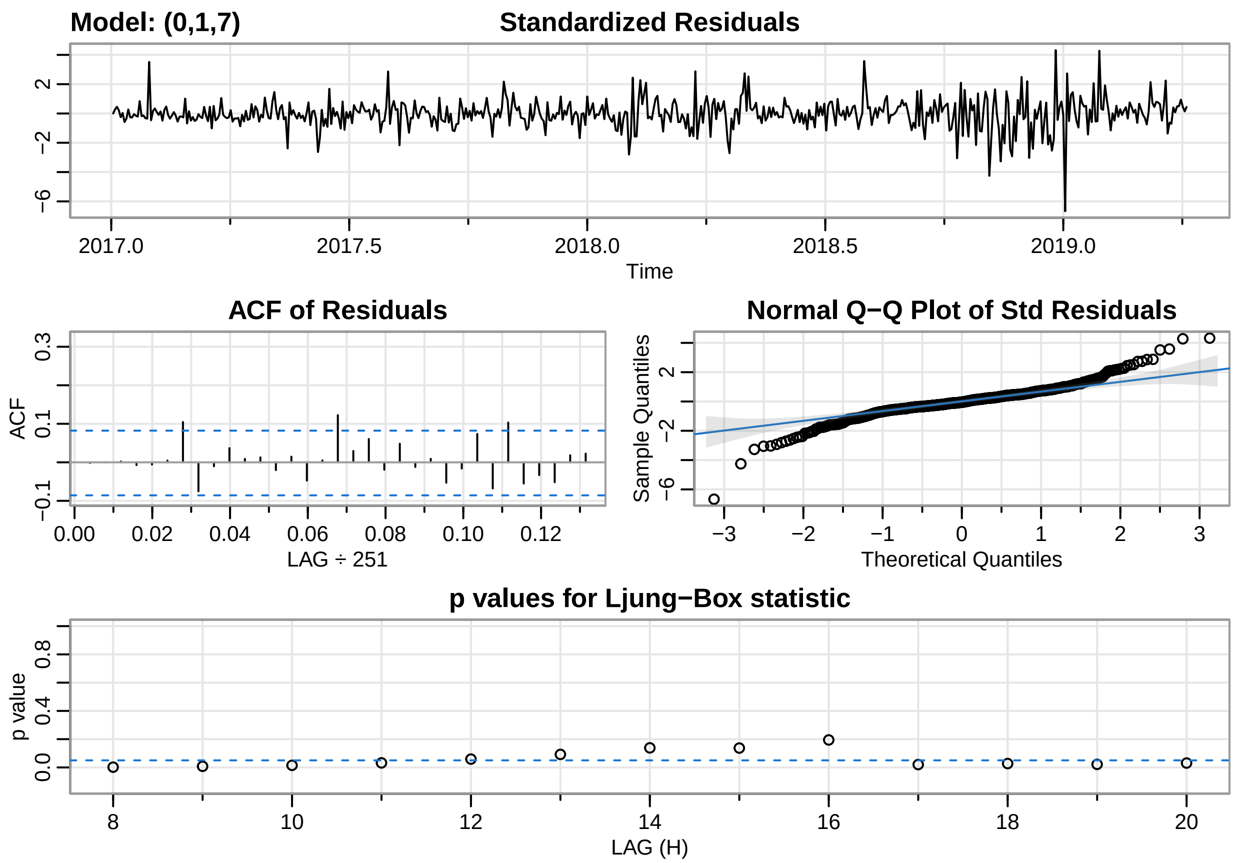
```
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

##
## Attaching package: 'forecast'

## The following object is masked from 'package:astsa':
##
##     gas
```

```r
fitMA7 <- Arima(lnd, order=c(0,1,7), method='ML', include.constant = TRUE)
fitAR7 <- Arima(lnd, order=c(7,1,0), method='ML', include.constant = TRUE)
```

```r
#MA model

sfitMA7 <- sarima(lnd, p=0, d=1, q=7)
```

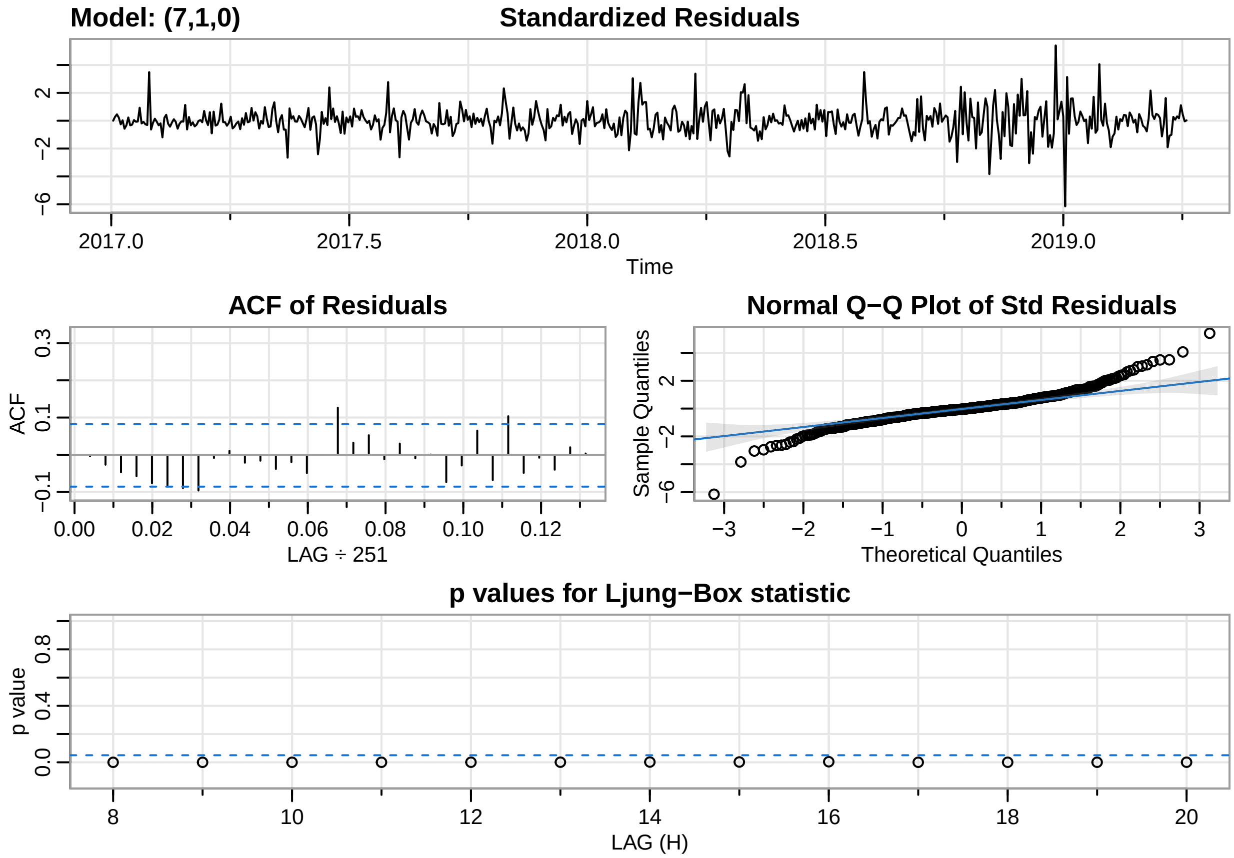
```
## initial  value -3.809896
## iter   2 value -4.005637
## iter   3 value -4.110756
## iter   4 value -4.124883
## iter   5 value -4.126329
## iter   6 value -4.138462
## iter   7 value -4.146561
## iter   8 value -4.148098
## iter   9 value -4.148460
## iter  10 value -4.149172
## iter  11 value -4.150176
## iter  12 value -4.150834
## iter  13 value -4.150858
## iter  14 value -4.150978
## iter  15 value -4.151031
## iter  16 value -4.151073
## iter  17 value -4.151094
## iter  18 value -4.151105
## iter  19 value -4.151109
## iter  20 value -4.151111
## iter  21 value -4.151111
## iter  22 value -4.151111
## iter  22 value -4.151111
## iter  22 value -4.151111
## final  value -4.151111
## converged
## initial  value -4.148594
## iter   2 value -4.149629
## iter   3 value -4.150124
## iter   4 value -4.150321
## iter   5 value -4.150353
## iter   6 value -4.150376
## iter   7 value -4.150377
## iter   7 value -4.150378
## iter   7 value -4.150378
## final  value -4.150378
## converged
```




```
# Now the AR model:

sfitAR7 <- sarima(lnd, p=7, d=1, q=0)
```

```
## initial  value -3.804142
## iter   2 value -3.953175
## iter   3 value -4.014171
## iter   4 value -4.050291
## iter   5 value -4.067679
## iter   6 value -4.081800
## iter   7 value -4.097848
## iter   8 value -4.103538
## iter   9 value -4.104341
## iter  10 value -4.104536
## iter  11 value -4.104677
## iter  12 value -4.104743
## iter  13 value -4.104746
## iter  13 value -4.104746
## iter  13 value -4.104746
## final  value -4.104746
## converged
## initial  value -4.108949
## iter   2 value -4.108952
## iter   3 value -4.108953
## iter   4 value -4.108953
## iter   5 value -4.108953
## iter   5 value -4.108953
```

```
## iter   5 value -4.108953
## final  value -4.108953
## converged
```



The model seems to fit reasonably well. This is based on the fact that the standardized residuals of both models appear to resemble white noise, the ACF of residuals are within the blue dashed line ounds indicated on the graph and the QQ plot suggests the residuals have a distribution close to the theoretical normal distribution.

```
#####Information Criteria for Arima fit
fitMA7$aic
```

```
## [1] -3073.986
```

```
fitAR7$aic
```

```
## [1] -3027.097
```

```
fitMA7$aicc
```

```
## [1] -3073.662
```

```
fitAR7$aicc
```

```
## [1] -3026.773
```

```
fitMA7$bic
```

```
## [1] -3034.938
```

```
fitAR7$bic
```

```
## [1] -2988.049
```

```r
#####Information Criteria for sarima fit
sfitMA7$AIC
```

```
## [1] -5.431076
```

```r
sfitAR7$AIC
```

```
## [1] -5.348227
```

```r
sfitMA7$AICc
```

```
## [1] -5.430619
```

```r
sfitAR7$AICc
```

```
## [1] -5.34777
```

```r
sfitMA7$BIC
```

```
## [1] -5.362088
```

```r
sfitAR7$BIC
```

```
## [1] -5.279239
```

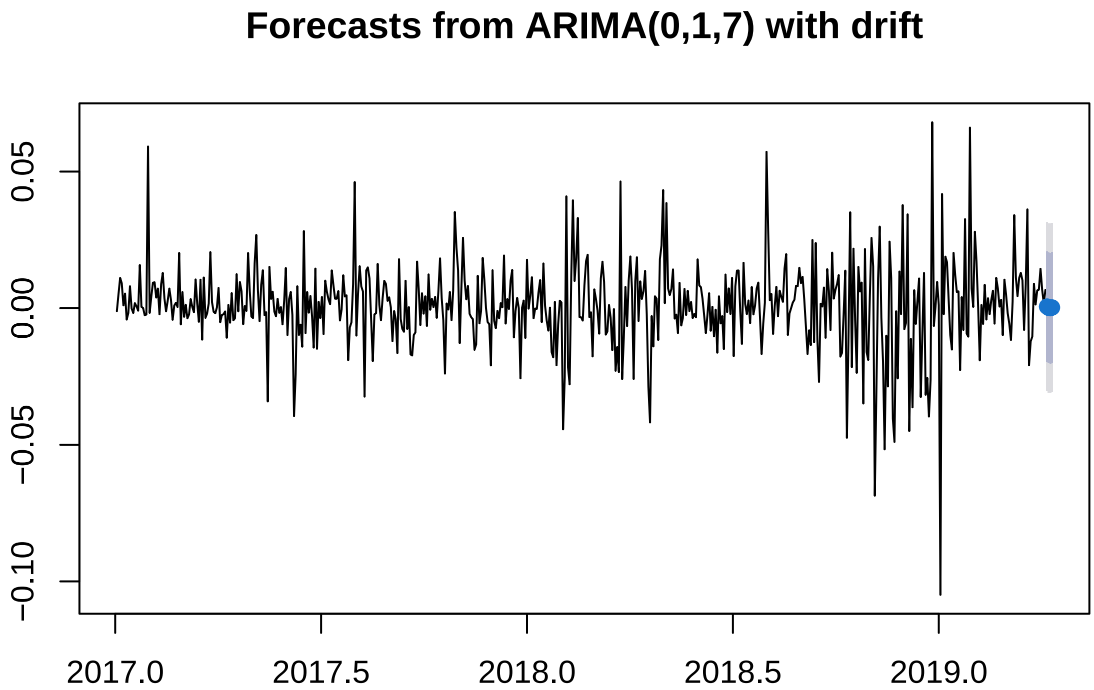
The AIC model seems to prefer the MA(7) model out of the Arima fit models and out of the sarima fit models. The AICc model has the same preference and prefers the MA(7) model out of the Arima fit models and out of the sarima fit models. The BIC model has the same preference.

I will choose the MA(7) model because it was chosen by all of the infomation criteria.

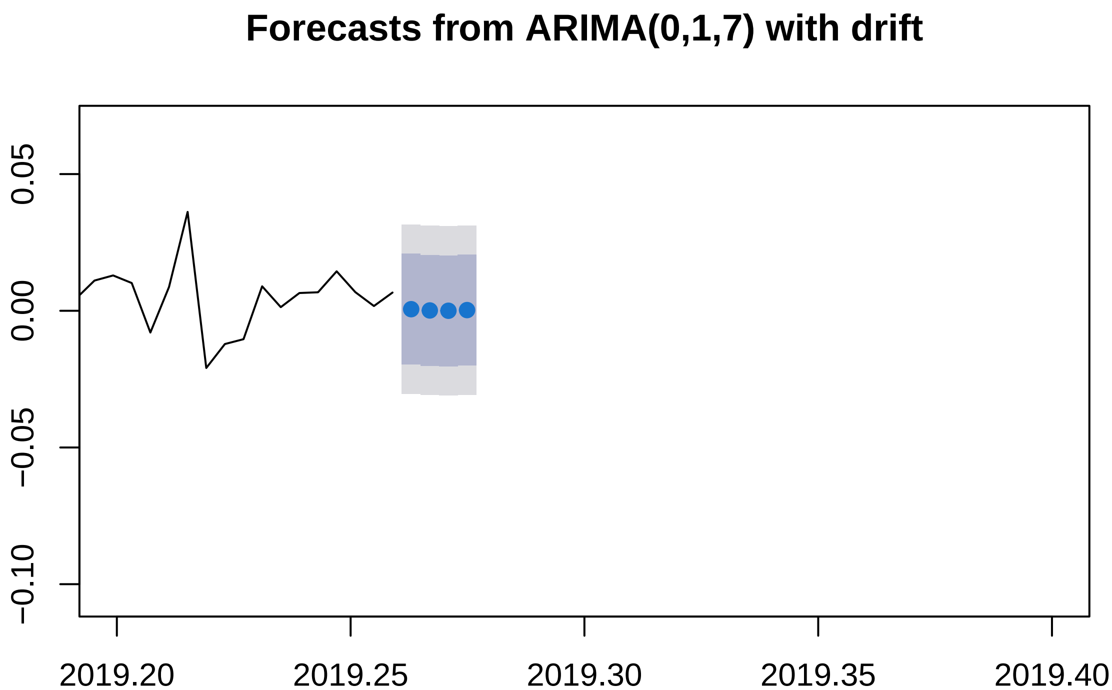
```r
fcMA7 <- forecast(fitMA7, h=4) ###Predicts next 5 values
fcMA7
```

```
##          Point Forecast       Lo 80      Hi 80       Lo 95      Hi 95
## 2019.263   5.844247e-04 -0.01965938 0.02082823 -0.03037581 0.03154466
## 2019.267   1.206997e-04 -0.02012359 0.02036499 -0.03084027 0.03108167
## 2019.271   1.185171e-05 -0.02025743 0.02028114 -0.03098735 0.03101105
## 2019.275   2.523155e-04 -0.02001873 0.02052336 -0.03074957 0.03125420
```

```r
plot(fcMA7)
```

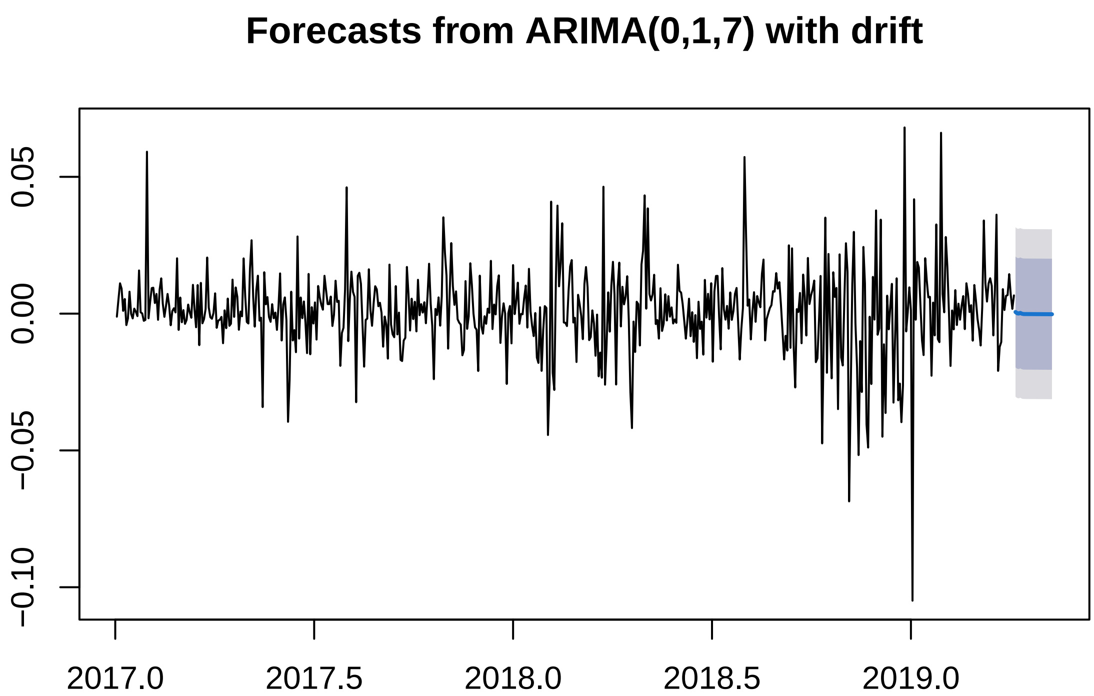



```
###Limits x-axis of plot to years 2019 to 2020
plot(fcMA7, xlim=c(2019.2, 2019.4))
```



Used the Arima fit because this was preferred in class.

```
fitARIMA017 <- Arima(lnd, order=c(0,1,7), include.drift=TRUE)
fcARIMA017 <- forecast(fitARIMA017, h=24)
plot(fcARIMA017)
```




```r
plot(fcARIMA017, xlim=c(2019.2, 2019.4))
```