430 Project2

Xinran Liu Shaoyang Sun Zefeng Lu

2019/12/5

I. Introduction

The Nasdaq Stock Market is an American stock exchange located at One Liberty Plaza in New York City. It is ranked second on the list of stock exchanges by market capitalization of shares traded, behind only the New York Stock Exchange.The exchange platform is owned by Nasdaq, Inc., which also owns the Nasdaq Nordic stock market network and several U.S. stock and options exchanges.

Amazon.com, Inc., is an American multinational technology company based in Seattle that focuses on e-commerce, cloud computing, digital streaming, and artificial intelligence. Amazon is known for well-established industries through technological innovation. It is the world's largest online marketplace, AI assistant provider, and cloud computing platform as measured by revenue and market capitalization.

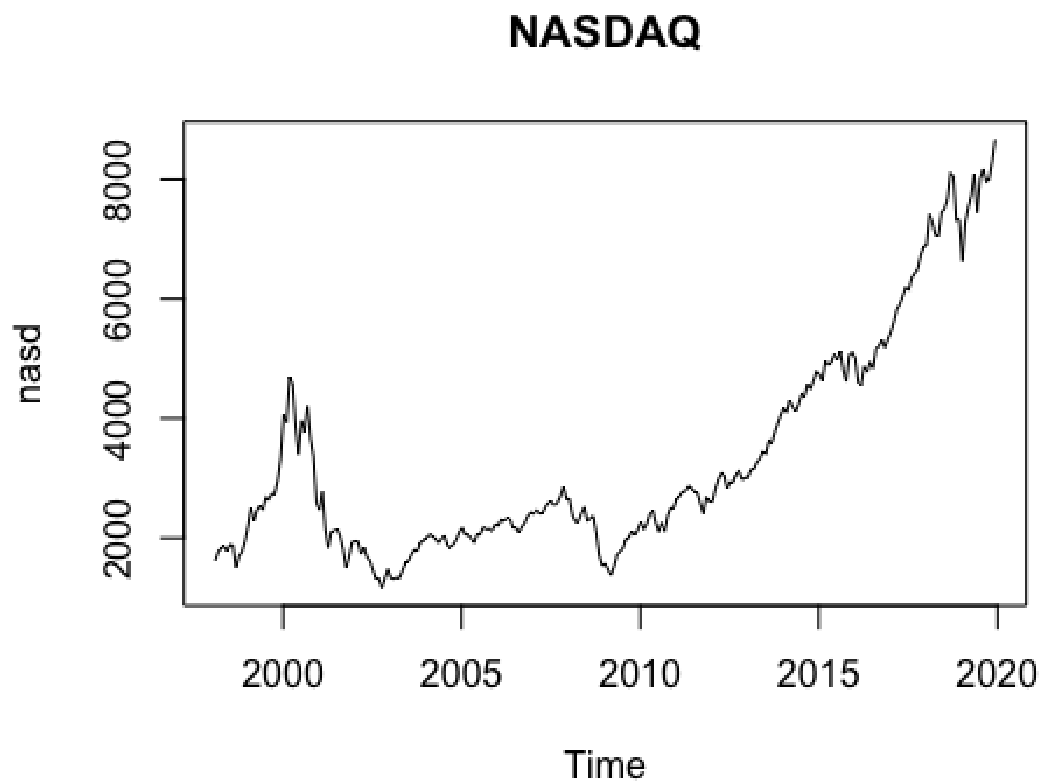
For this project, we collect data of Nasdaq Stock and Amazon stock price from January 1998 to November 2019 from Yahoo Finance. We pick adjusted close price as our dependent variable. By making time series models for both companies, we want to investigate whether there are trends, seasonalities, or cycles in the models. Also, we try to fit a VAR model by using both data of Nasdaq and Amazon stock, and see if Nasdaq value is a good predictor of Amazon stock price.

II. Results

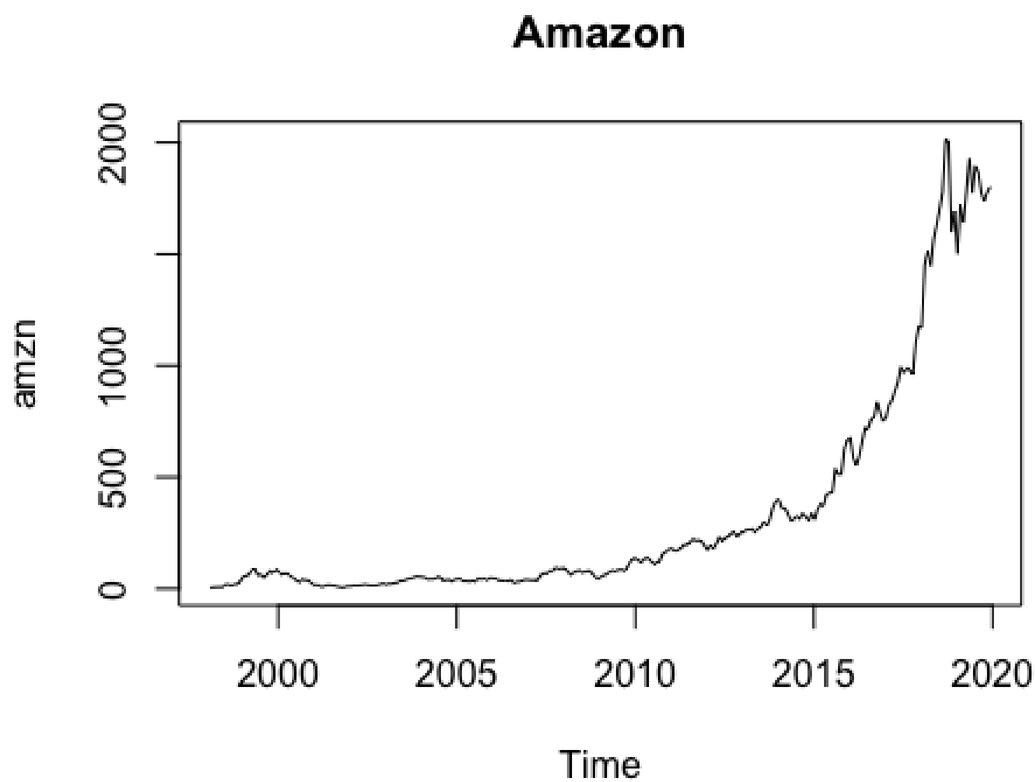
```
library(readxl)
stock <- read_excel("/Users/sunshaoyang/Desktop/NASDAQAMZN.xlsx")

library(psych)
describe(stock)

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; ret
urning
## Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; ret
urning
## -Inf

##        vars   n     mean       sd   median trimmed     mad     min    
max
## Date      1 263      NaN       NA       NA     NaN      NA     Inf    -
```

```
Inf
## NASDAQ    2 263 3348.56 1839.49 2596.36 3059.16 1096.83 1172.06 8665
.47
## AMZN      3 263  329.77  498.96   83.66  205.64  100.96    4.92 2012
.71
##            range skew kurtosis     se
## Date        -Inf   NA       NA     NA
## NASDAQ 7493.41 1.21     0.42 113.43
## AMZN   2007.79 2.00     2.94  30.77
```

```r
nasd=ts(stock$NASDAQ,start=1998.1,freq=12)
amzn=ts(stock$AMZN,start=1998.1,freq=12)
plot(nasd, main="NASDAQ")
```

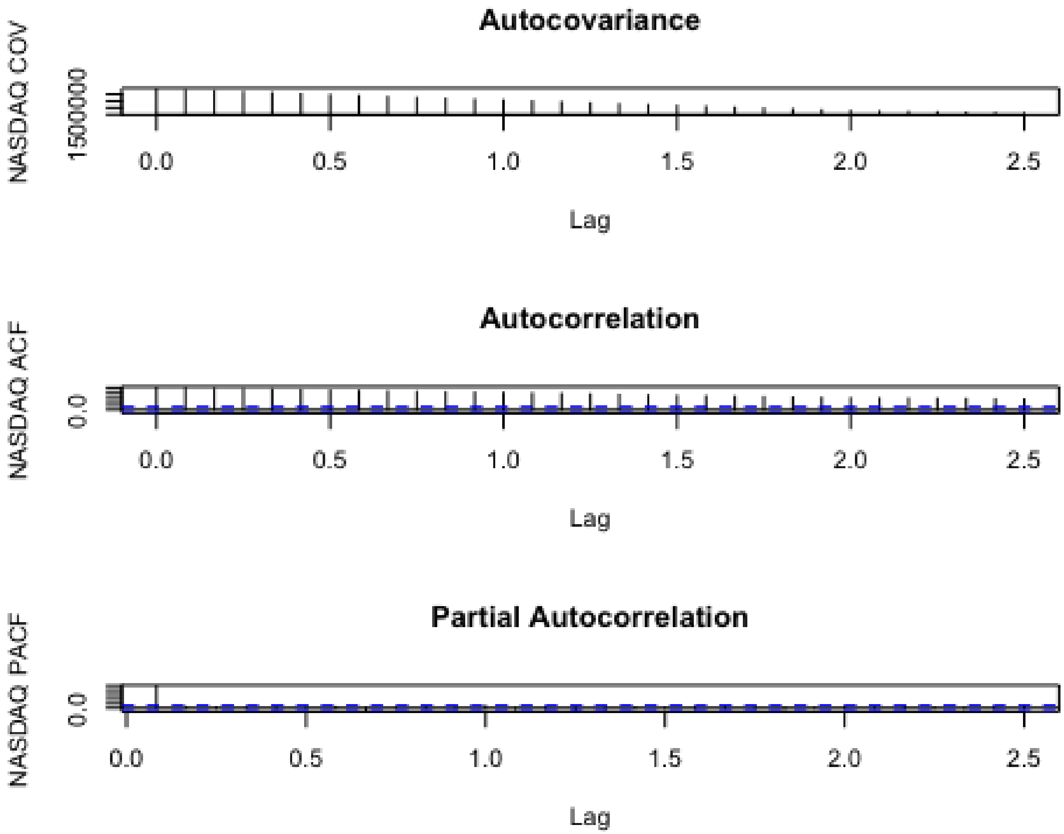



```r
plot(amzn, main="Amazon")
```

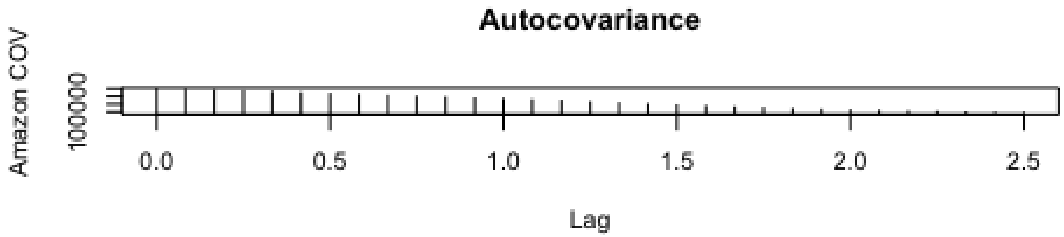


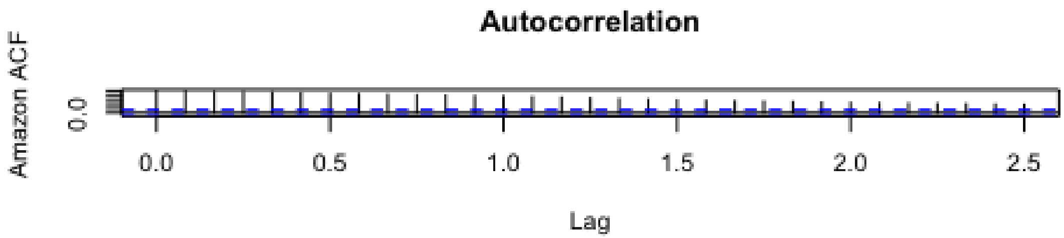
1\.
Produce a time-series plot of your data including the respective ACF and PACF plots.

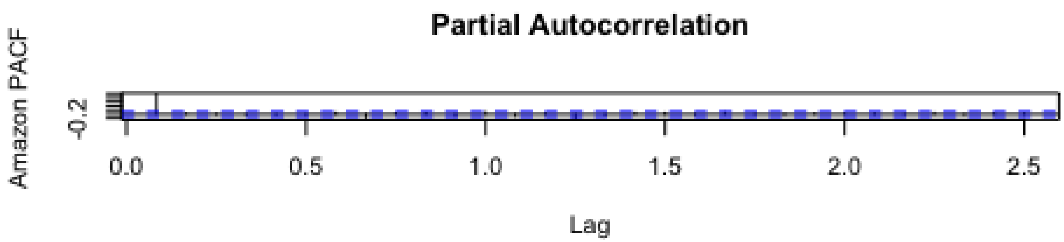
```
par(mfrow=c(3,1))
acf(nasd, type = "covariance", main="Autocovariance", lag.max=30, ylab=
"NASDAQ COV")
acf(nasd, type = "correlation", main="Autocorrelation", lag.max=30, yla
b= "NASDAQ ACF")
acf(nasd, type = "partial",main="Partial Autocorrelation",lag.max=30, y
lab="NASDAQ PACF")
```




```r
par(mfrow=c(3,1))
acf(amzn, type = "covariance", main="Autocovariance", lag.max=30, ylab=
"Amazon COV")
acf(amzn, type = "correlation", main="Autocorrelation", lag.max=30, yla
b= "Amazon ACF")
acf(amzn, type = "partial",main="Partial Autocorrelation",lag.max=30, y
lab="Amazon PACF")
```







2. As a baseline model, fit an ARIMA model to each series and comment on the fit. For the next questions, you will instead use the model estimated in (3) for their respective answers.

```
library(forecast)
library(tseries)
auto.arima(nasd)

## Series: nasd
## ARIMA(0,1,0)(0,0,1)[12] with drift
##
## Coefficients:
##          sma1    drift
##       -0.1275  26.0452
## s.e.   0.0694  11.3970
##
## sigma^2 estimated as 44353:  log likelihood=-1772.55
## AIC=3551.09   AICc=3551.19   BIC=3561.8
```

According to the result of auto.arima, we can conclude the model has one-period difference cycle, seasonal-MA(1) monthly seasonality and 26.0452 drift.

```
auto.arima(amzn)
```

```
## Series: amzn
## ARIMA(0,2,2)(0,0,1)[12]
##
## Coefficients:
##           ma1    ma2     sma1
##       -1.1492  0.178  -0.1330
## s.e.   0.0593  0.059   0.0723
##
## sigma^2 estimated as 2373:  log likelihood=-1384.84
## AIC=2777.68   AICc=2777.84   BIC=2791.94
```

According to the result of auto.arima, we choose the second-period difference, original MA(2) cycle, and seasonal MA(1) as our baseline model of AMZN data.

3. Fit a model that includes, trend, seasonality and cyclical components. Make sure to discuss your model in detail.

```
model_nasd=Arima(nasd, order=c(0,1,0), seasonal=list(order=c(0,0,1)),in
clude.drift = TRUE)
summary(model_nasd)

## Series: nasd
## ARIMA(0,1,0)(0,0,1)[12] with drift
##
## Coefficients:
##          sma1    drift
##       -0.1275  26.0452
## s.e.   0.0694  11.3970
##
## sigma^2 estimated as 44353:  log likelihood=-1772.55
## AIC=3551.09   AICc=3551.19   BIC=3561.8
##
## Training set error measures:
##                     ME     RMSE     MAE        MPE    MAPE      MASE
## Training set 0.3621672 209.3965 146.557 -0.6792378 4.95952 0.2343467
##                     ACF1
## Training set -0.03368376

library(lmtest)

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

coeftest(model_nasd)
```

```
##
## z test of coefficients:
##
##        Estimate Std. Error z value Pr(>|z|)
## sma1  -0.127462   0.069439 -1.8356  0.06642 .
## drift 26.045171  11.397002  2.2853  0.02230 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The model for NASDAQ has one-period difference as cycle, seasonaly MA(1) monthly as seasonaliy and 26.045171 as drift. According to the result of coefficient test, we can conclude that all of the coefficients are statistically significant at 5% confidence level.

```r
model_amzn=Arima(amzn, order=c(0,2,2), seasonal=list(order=c(0,0,1)))
summary(model_amzn)
```

```
## Series: amzn
## ARIMA(0,2,2)(0,0,1)[12]
##
## Coefficients:
##           ma1    ma2     sma1
##       -1.1492  0.178  -0.1330
## s.e.   0.0593  0.059   0.0723
##
## sigma^2 estimated as 2373:  log likelihood=-1384.84
## AIC=2777.68   AICc=2777.84   BIC=2791.94
##
## Training set error measures:
##                    ME     RMSE      MAE        MPE     MAPE      MAS
E
## Training set 2.796741 48.25302 21.38496 -0.6646715 10.60522 0.218391
4
##                     ACF1
## Training set -0.01632644
```

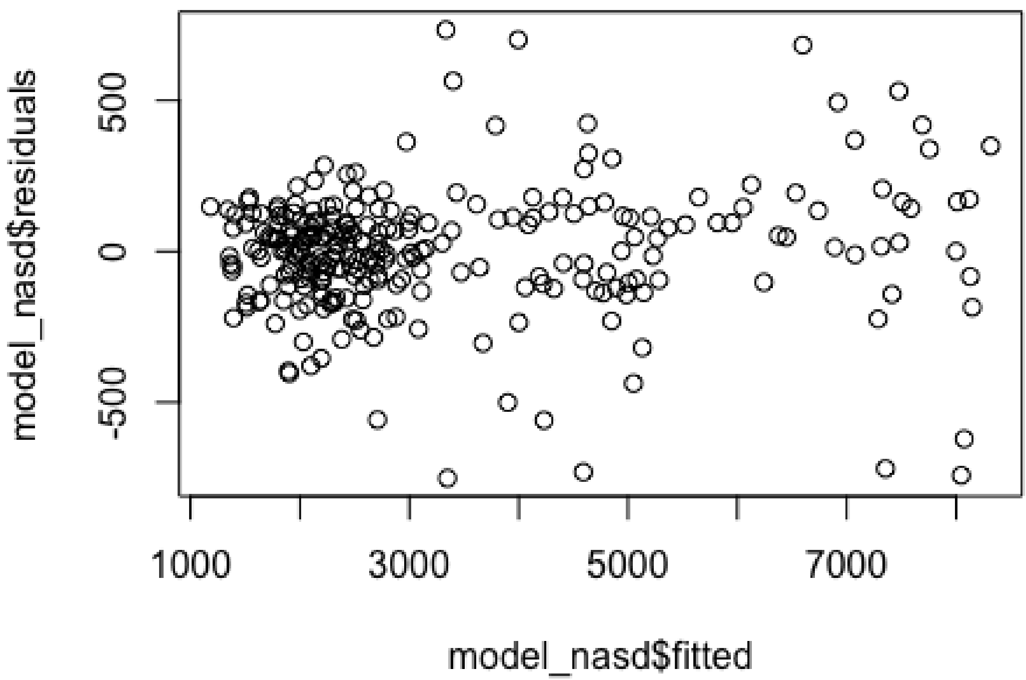
```r
coeftest(model_amzn)
```

```
##
## z test of coefficients:
##
##       Estimate Std. Error  z value  Pr(>|z|)
## ma1  -1.149229   0.059345 -19.3653 < 2.2e-16 ***
## ma2   0.177985   0.059013   3.0160  0.002561 **
## sma1 -0.133028   0.072341  -1.8389  0.065929 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The baseline model of amzn is 2 periods differenced, MA(2) cycle, seasonal MA(1). According to the coefficients test, the coefficients of cycle are significant under 5% significance level.

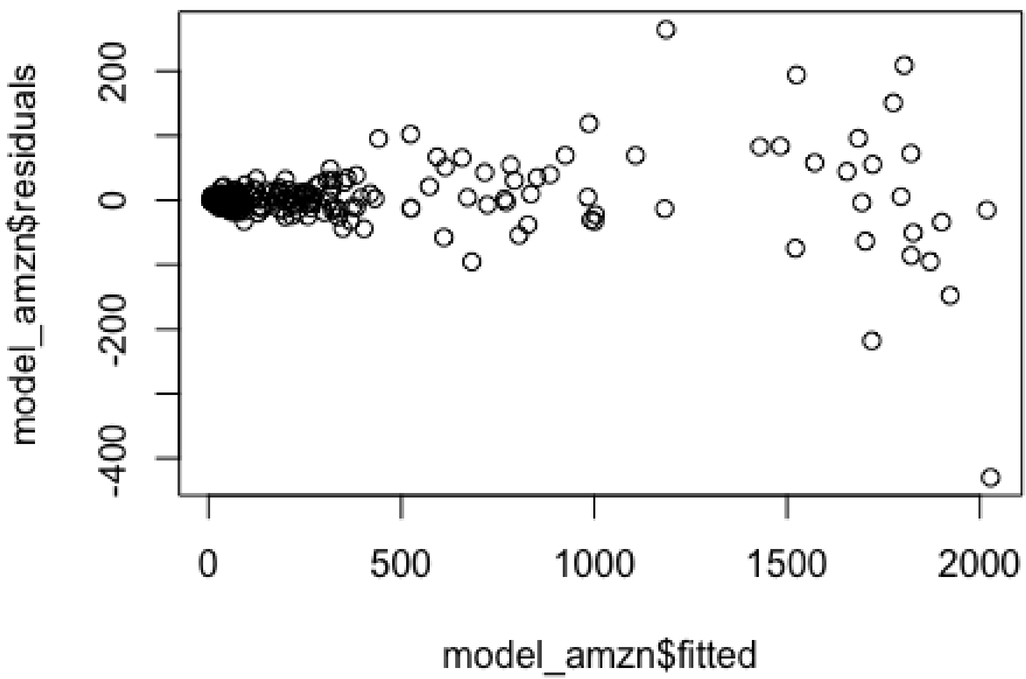
4. Plot the respective residuals vs. fitted values and discuss your observations.

```
plot(model_nasd$fitted, model_nasd$residuals)
```



According to the plot of residuals vs. fitted values of model NASDAQ, we could conclude that the residuals distribute around 0 randomly. And there is no heterskedasticity almostly.

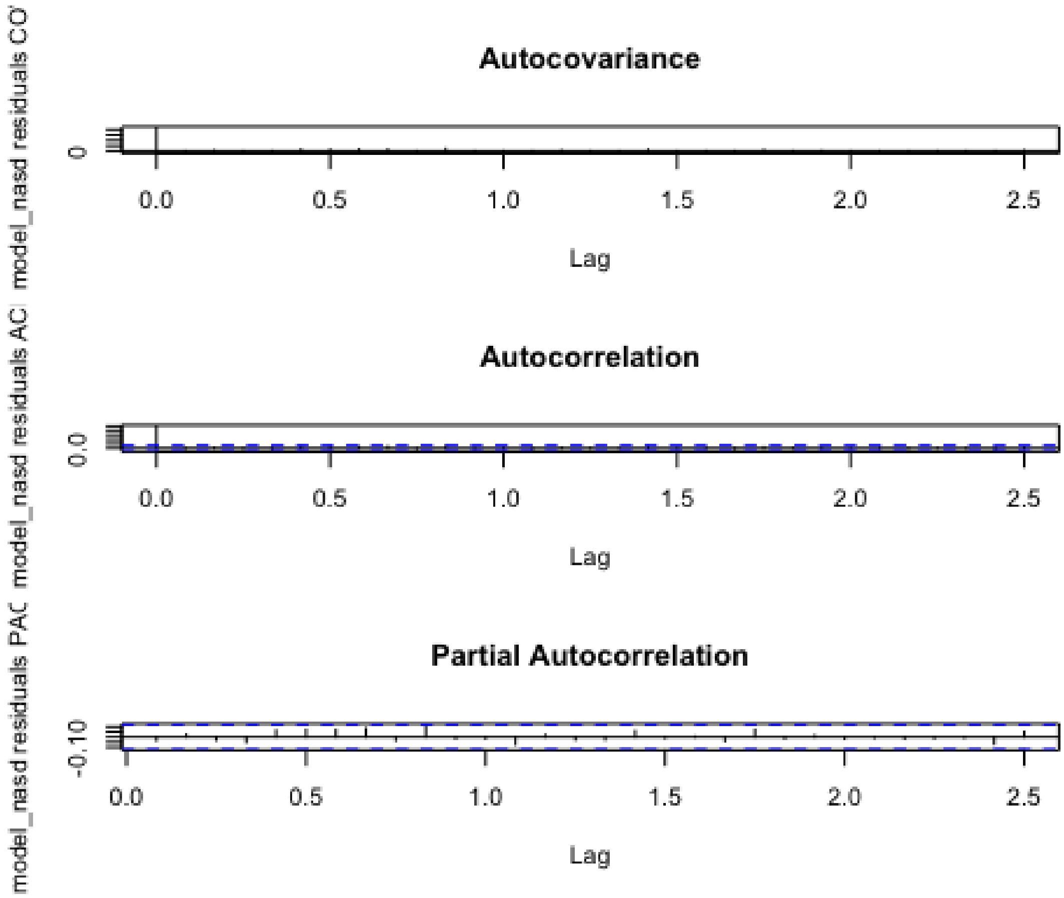
```
plot(model_amzn$fitted, model_amzn$residuals)
```



Most of the residuals of the amzn model is distributed around zero randomly with several extreme values.

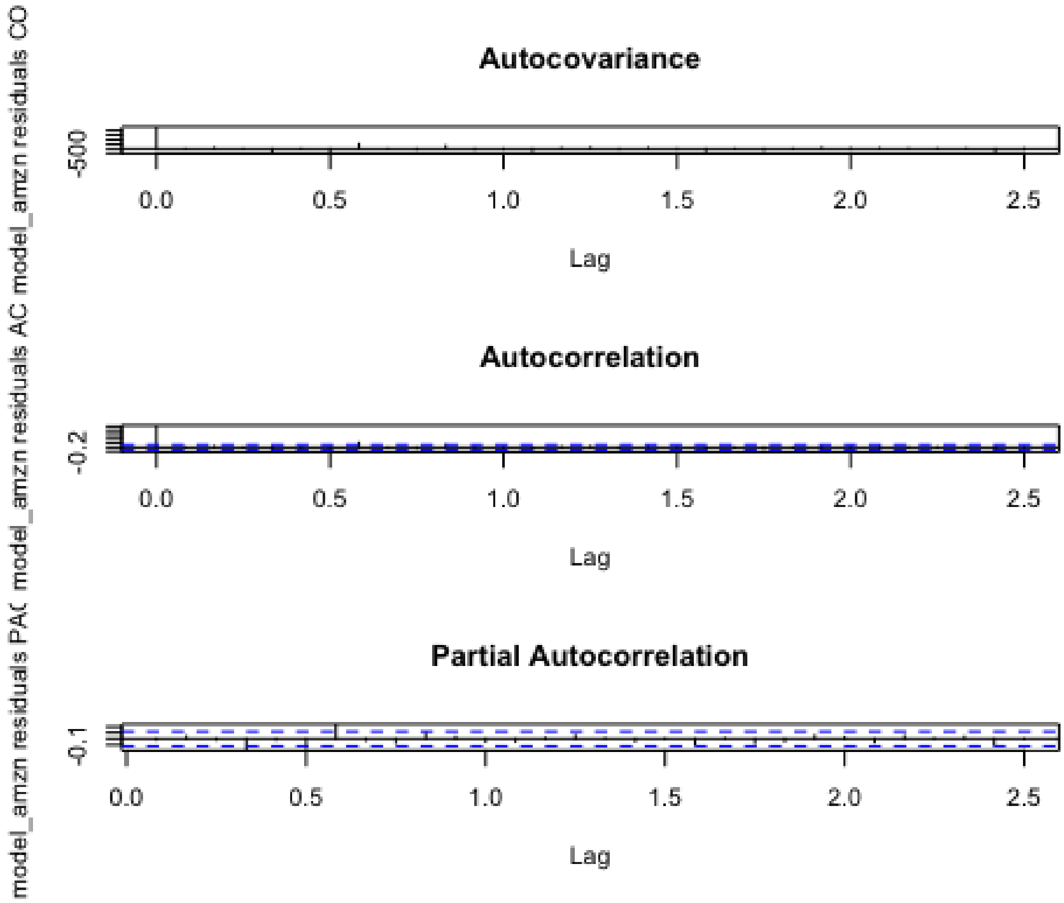
5. Plot the ACF and PACF of the respective residuals and interpret the plots.

```
par(mfrow=c(3,1))
acf(model_nasd$residuals, type = "covariance", main="Autocovariance", l
ag.max=30, ylab="model_nasd residuals COV")
acf(model_nasd$residuals, type = "correlation", main="Autocorrelation",
 lag.max=30, ylab= "model_nasd residuals ACF")
acf(model_nasd$residuals, type = "partial",main="Partial Autocorrelatio
n",lag.max=30, ylab="model_nasd residuals PACF")
```

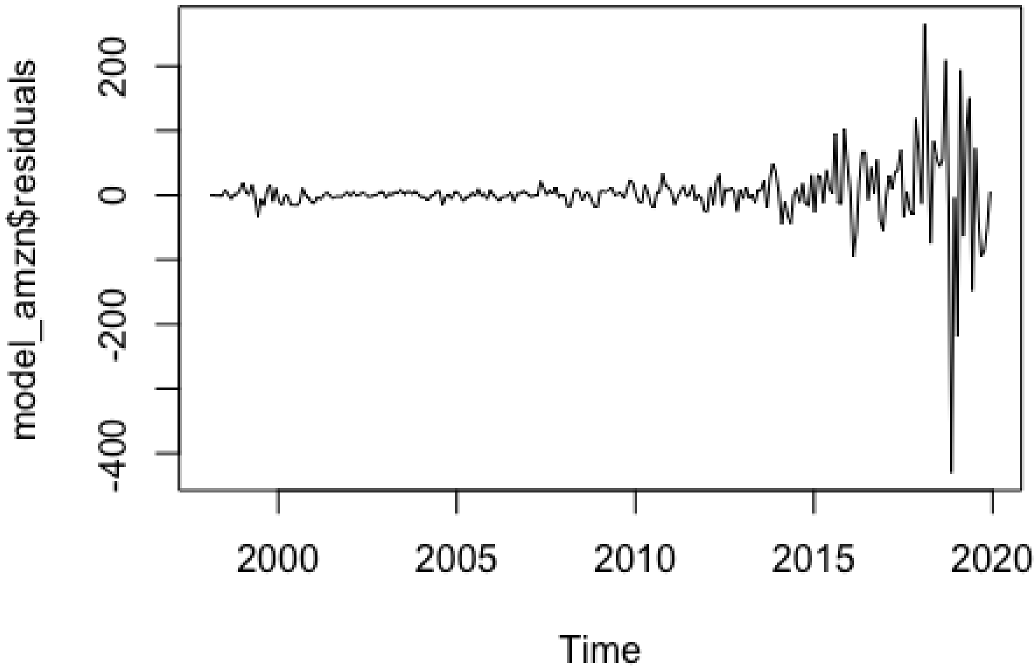


According to the ACF and PACF plots of residuals in model NASDAQ, there is no spike in these two plots which means the residuals are white noise and there is no pattern left in the estimated model.

```
par(mfrow=c(3,1))
acf(model_amzn$residuals, type = "covariance", main="Autocovariance", l
ag.max=30, ylab="model_amzn residuals COV")
acf(model_amzn$residuals, type = "correlation", main="Autocorrelation",
 lag.max=30, ylab= "model_amzn residuals ACF")
acf(model_amzn$residuals, type = "partial",main="Partial Autocorrelatio
n",lag.max=30, ylab="model_amzn residuals PACF")
```



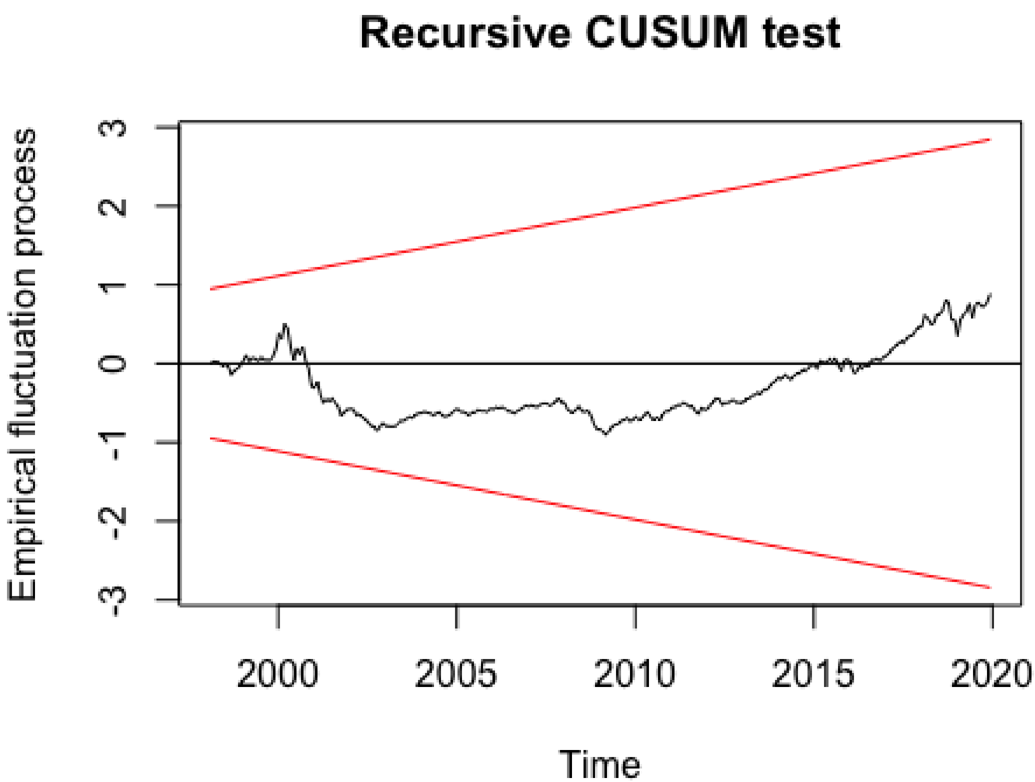

```
plot(model_amzn$residuals)
```



According to the acf and pacf plots of residuals, we found some spikes, or significant correlations in it and there may be missing factors in the arima model.

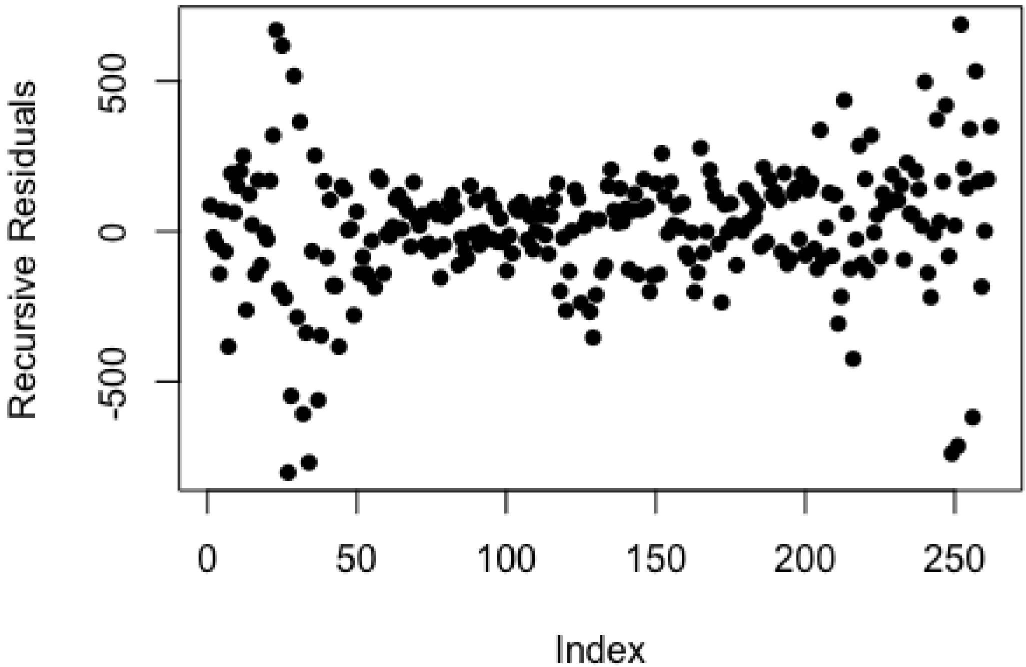
6. Plot the respective CUSUM and interpret the plot.

```
library(strucchange)

## Loading required package: sandwich

y=recresid(model_nasd$residuals~1)
plot(efp(model_nasd$residuals~1, type = "Rec-CUSUM"))
```



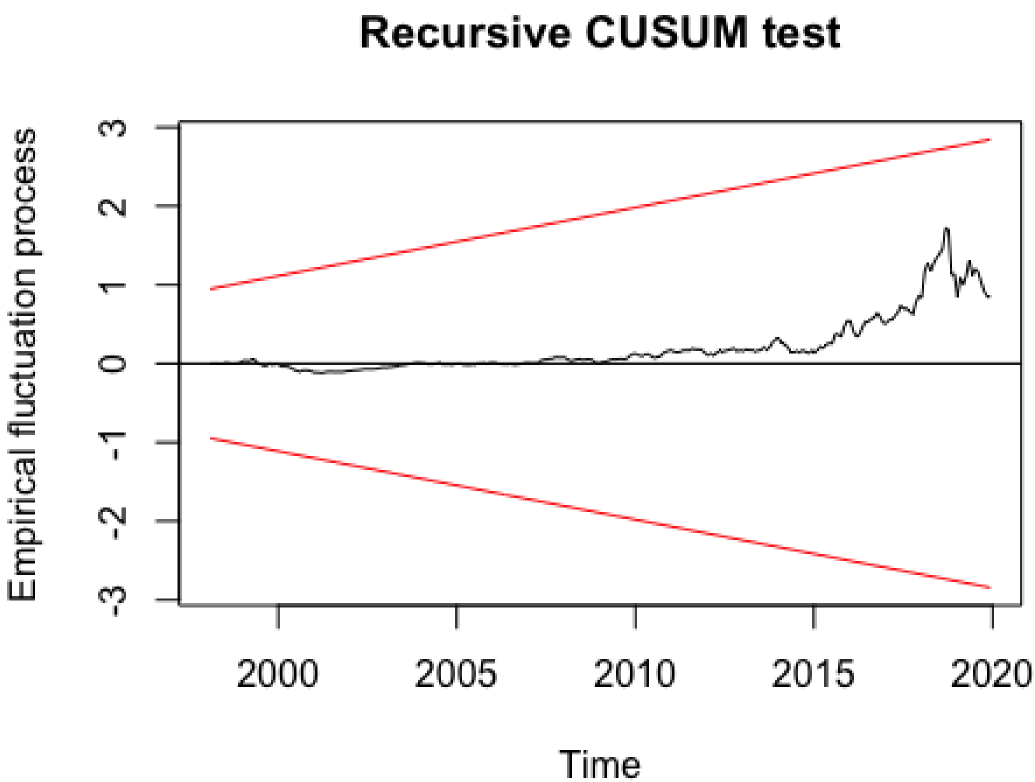
According to the Recursive CUSUM plot, we could conclude that even though the residual does not equal to 0 exactly, they are between the bond.

```
plot(y, pch=16,ylab="Recursive Residuals")
```



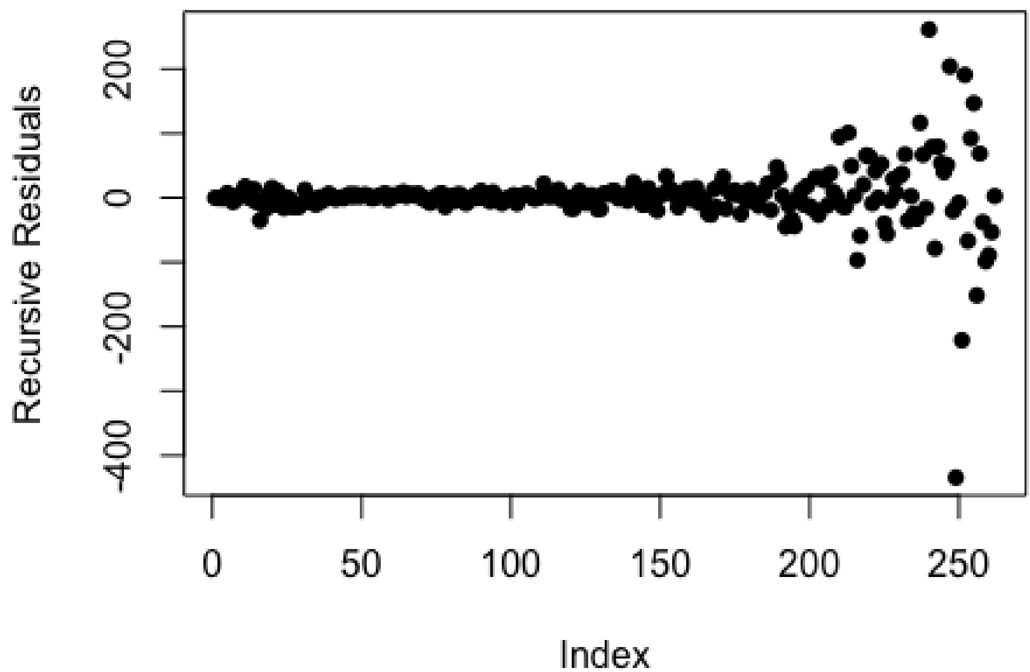
Recursive estimates provide information about parameter stability. From the Recursive Residuals plot, the residuals are stable in the middle periods, and the residuals are more scattered in the first and last several periods.

```
y=recresid(model_amzn$res~1)
plot(efp(model_amzn$res~1, type = "Rec-CUSUM"))
```



The residuals vary around zero before 2015, and the variance rises when it's around 2018 but it still in the red lines range.

```
plot(y, pch=16,ylab="Recursive Residuals")
```



The recursive estimates provide infomation about the stability of the residuals. From the Recursive Residuals plot, the residuals are stable from the beginning, and there are some extreme values in the recent two years.

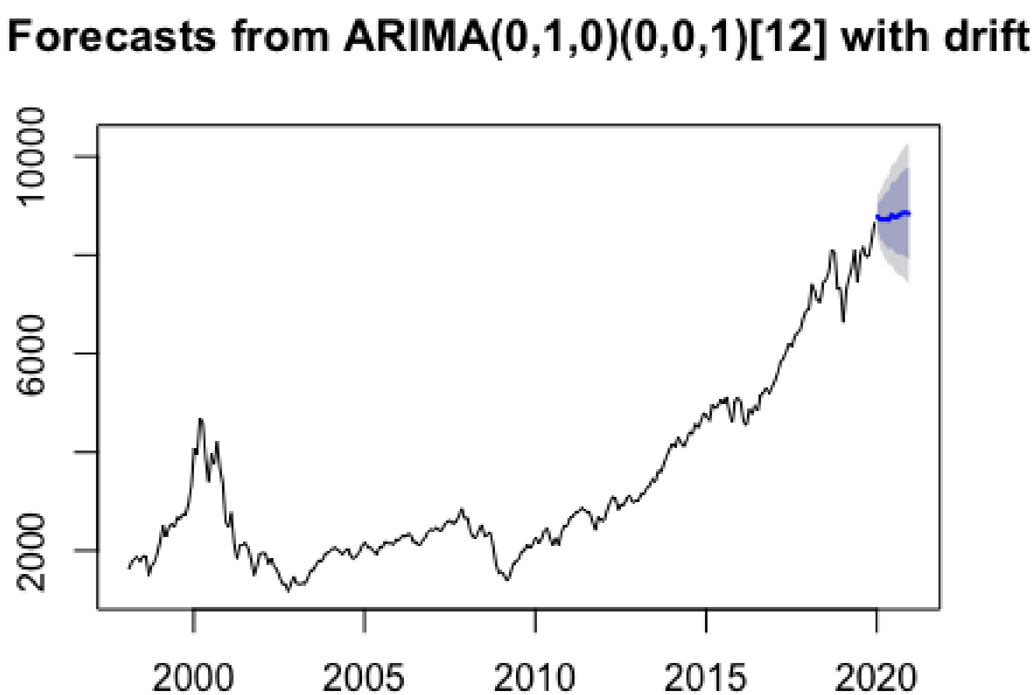
8. For your model, discuss the associated diagnostic statistics.

```
accuracy(model_nasd)

##                     ME     RMSE     MAE        MPE    MAPE      MASE
## Training set 0.3621672 209.3965 146.557 -0.6792378 4.95952 0.2343467
##                     ACF1
## Training set -0.03368376

accuracy(model_amzn)

##                    ME     RMSE      MAE        MPE     MAPE      MAS
E
## Training set 2.796741 48.25302 21.38496 -0.6646715 10.60522 0.218391
4
##                     ACF1
## Training set -0.01632644
```

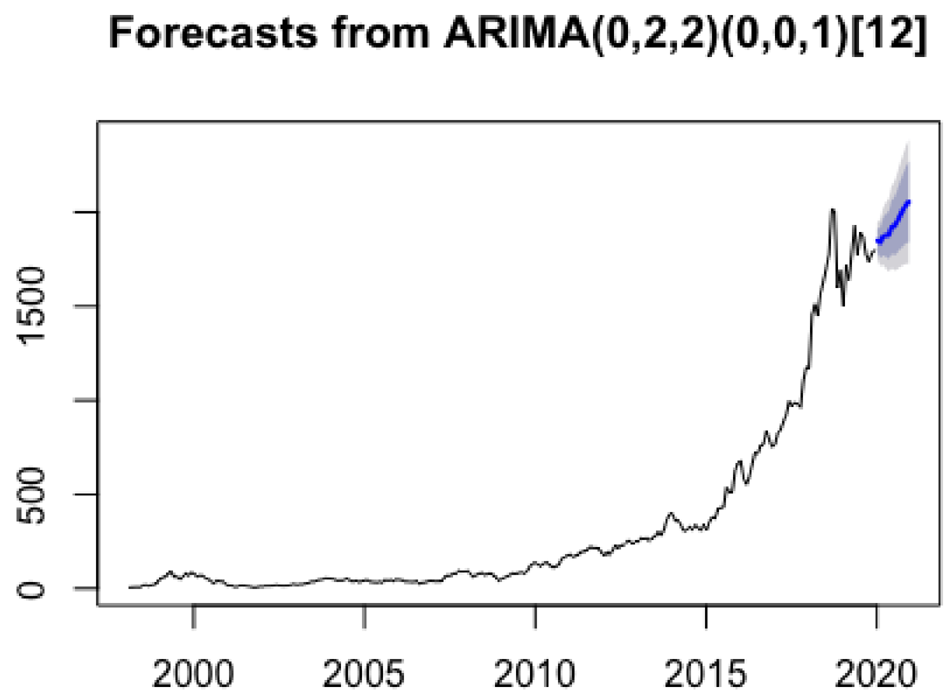
According to the error metrics, the root mean of standard error is about 48, which is much smaller than the RMSE of the whole market. In other words, the deviation between the observed values and the truth values are smaller in amzn model.
According to the error metrics, we can get the associated diagnostic statistics to show goodness of fit. Based on these indicators (ME, MPE, MAPE, MASE, etc.), we conclude that the two estimated models are good to fit the true value and have good forecasting performance.

9. Use your model to forecast 12-steps ahead. Your forecast should include the respective error bands.

```
plot(forecast(model_nasd,h=12))
```




```
plot(forecast(model_amzn,h=12))
```



10. Fit an appropriate VAR model using your two variables. Make sure to show the relevant plots and discuss your results from the fit.

```
library(vars)

## Loading required package: MASS

## Loading required package: urca

y=cbind(nasd,amzn)
y_tot=data.frame(y)
VARselect(y_tot, 10)

## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      8      1      1      8
##
## $criteria
##                    1            2            3            4
  5
## AIC(n) 1.820272e+01 1.819821e+01 1.822035e+01 1.824877e+01 1.821955e
+01
## HQ(n)  1.823643e+01 1.825440e+01 1.829902e+01 1.834992e+01 1.834316e
+01
## SC(n)  1.828651e+01 1.833787e+01 1.841587e+01 1.850016e+01 1.852680e
```

```
+01
## FPE(n) 8.041550e+07 8.005454e+07 8.184832e+07 8.421086e+07 8.178918e
+07
##                    6            7            8            9
 10
## AIC(n) 1.824572e+01 1.821536e+01 1.816899e+01 1.818548e+01 1.821019e
+01
## HQ(n)  1.839182e+01 1.838392e+01 1.836004e+01 1.839900e+01 1.844618e
+01
## SC(n)  1.860884e+01 1.863433e+01 1.864384e+01 1.871619e+01 1.879676e
+01
## FPE(n) 8.396455e+07 8.146092e+07 7.778025e+07 7.908615e+07 8.108033e
+07
```

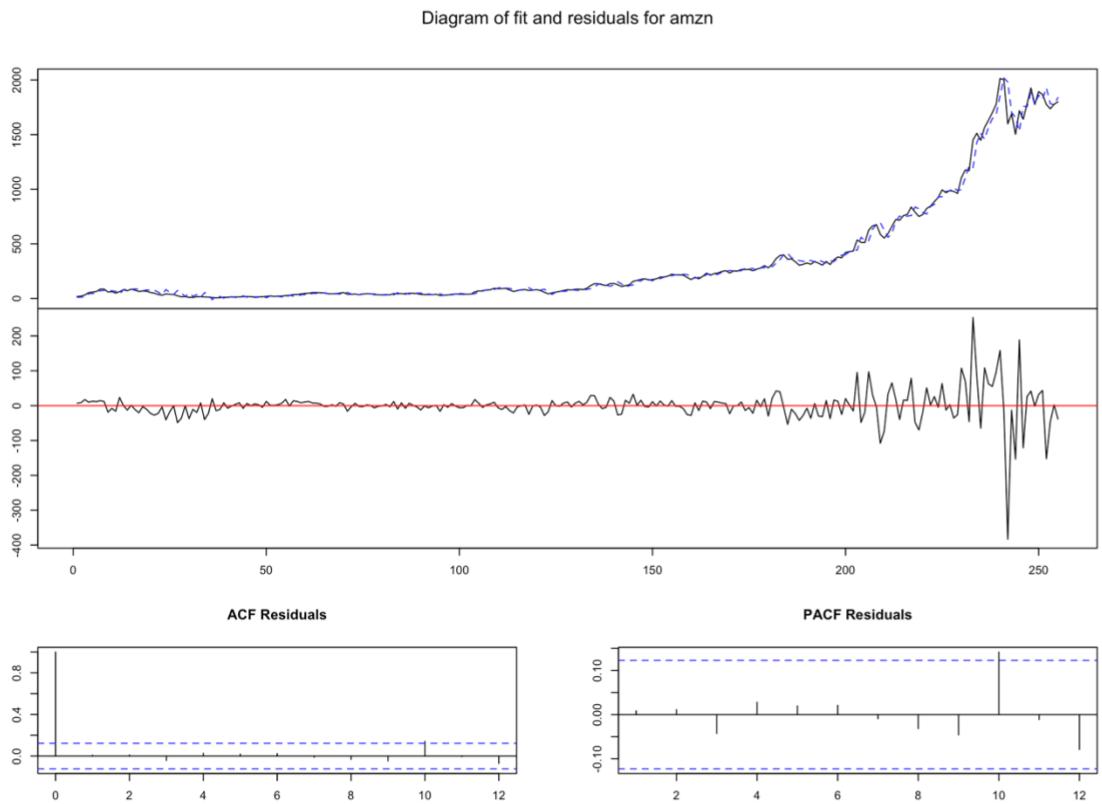
```
var_model=VAR(y_tot,p=8)
summary(var_model)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: nasd, amzn
## Deterministic variables: const
## Sample size: 255
## Log Likelihood: -3004.418
## Roots of the characteristic polynomial:
## 1.014 0.9138 0.9138 0.8739 0.8739 0.8426 0.8426 0.8341 0.8341 0.7559
 0.7559 0.6782 0.6782 0.6579 0.6246 0.4268
## Call:
## VAR(y = y_tot, p = 8)
##
##
## Estimation results for equation nasd:
## =====================================
## nasd = nasd.l1 + amzn.l1 + nasd.l2 + amzn.l2 + nasd.l3 + amzn.l3 + n
asd.l4 + amzn.l4 + nasd.l5 + amzn.l5 + nasd.l6 + amzn.l6 + nasd.l7 + am
zn.l7 + nasd.l8 + amzn.l8 + const
##
##          Estimate Std. Error t value Pr(>|t|)
## nasd.l1  1.02732    0.07516  13.668  < 2e-16 ***
## amzn.l1 -0.40910    0.32410  -1.262 0.208092
## nasd.l2 -0.04616    0.10692  -0.432 0.666348
## amzn.l2  0.27907    0.42845   0.651 0.515443
## nasd.l3 -0.03061    0.10663  -0.287 0.774318
## amzn.l3  0.36080    0.43159   0.836 0.404001
## nasd.l4  0.06953    0.10583   0.657 0.511832
## amzn.l4 -1.05324    0.43066  -2.446 0.015184 *
## nasd.l5 -0.02514    0.10552  -0.238 0.811917
## amzn.l5  0.82312    0.43448   1.894 0.059372 .
## nasd.l6  0.15850    0.10562   1.501 0.134766
## amzn.l6 -0.60832    0.43893  -1.386 0.167065
```

```
## nasd.l7 -0.20702    0.10740  -1.928 0.055099 .
## amzn.l7  1.50106    0.44181   3.398 0.000797 ***
## nasd.l8  0.02315    0.07638   0.303 0.762080
## amzn.l8 -0.71853    0.34486  -2.084 0.038271 *
## const   80.06550   51.13728   1.566 0.118748
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 205.5 on 238 degrees of freedom
## Multiple R-Squared: 0.9884,  Adjusted R-squared: 0.9876
## F-statistic:  1266 on 16 and 238 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation amzn:
## =====================================
## amzn = nasd.l1 + amzn.l1 + nasd.l2 + amzn.l2 + nasd.l3 + amzn.l3 + n
asd.l4 + amzn.l4 + nasd.l5 + amzn.l5 + nasd.l6 + amzn.l6 + nasd.l7 + am
zn.l7 + nasd.l8 + amzn.l8 + const
##
##            Estimate Std. Error t value Pr(>|t|)
## nasd.l1   0.005542   0.017109   0.324   0.7463
## amzn.l1   0.856349   0.073775  11.608  < 2e-16 ***
## nasd.l2  -0.002527   0.024338  -0.104   0.9174
## amzn.l2   0.144734   0.097526   1.484   0.1391
## nasd.l3   0.006218   0.024271   0.256   0.7980
## amzn.l3   0.018467   0.098241   0.188   0.8511
## nasd.l4  -0.027711   0.024089  -1.150   0.2512
## amzn.l4  -0.133449   0.098029  -1.361   0.1747
## nasd.l5   0.024262   0.024018   1.010   0.3135
## amzn.l5   0.029010   0.098900   0.293   0.7695
## nasd.l6   0.022124   0.024042   0.920   0.3584
## amzn.l6  -0.121310   0.099912  -1.214   0.2259
## nasd.l7  -0.048709   0.024447  -1.992   0.0475 *
## amzn.l7   0.524037   0.100568   5.211 4.08e-07 ***
## nasd.l8   0.031387   0.017387   1.805   0.0723 .
## amzn.l8  -0.338463   0.078500  -4.312 2.37e-05 ***
## const   -19.905477  11.640219  -1.710   0.0886 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 46.78 on 238 degrees of freedom
## Multiple R-Squared: 0.9919,  Adjusted R-squared: 0.9914
## F-statistic:  1824 on 16 and 238 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##       nasd amzn
```

```
## nasd 42232 5000
## amzn  5000 2188
##
## Correlation matrix of residuals:
##        nasd   amzn
## nasd 1.0000 0.5201
## amzn 0.5201 1.0000
```

```r
plot(var_model)
```



According to the VAR plot, when taking nasd as the dependent variable and the nasd lags and amzn lags as independent variables, the var model fit the true values well and the residuals are almost randomly vary arround 0 even though the residuals in the first and last seveal periods are more scattered.

```r
irf(var_model)
```

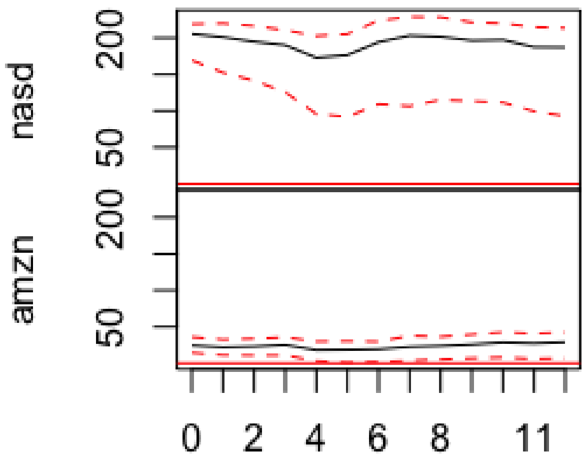
```
##
## Impulse response coefficients
## $nasd
##            nasd     amzn
##  [1,] 205.5031 24.33101
##  [2,] 201.1641 21.97467
##  [3,] 194.9748 22.93506
##  [4,] 190.2547 25.12036
```

```
##  [5,] 173.0094 18.10810
##  [6,] 176.5674 18.44151
##  [7,] 194.3574 19.25106
##  [8,] 203.2694 22.49272
##  [9,] 201.6901 23.71924
## [10,] 196.6026 25.66471
## [11,] 197.4178 28.41919
##
## $amzn
##              nasd     amzn
##  [1,]   0.0000000 39.95230
##  [2,] -16.3445790 34.21311
##  [3,] -19.6382313 34.99024
##  [4,]  -9.7721789 35.58593
##  [5,] -43.1612008 30.73232
##  [6,] -37.5908173 28.82389
##  [7,] -60.6525189 21.25262
##  [8,] -21.1131670 34.91444
##  [9,] -19.8901633 31.72337
## [10,] -11.3299108 31.88428
## [11,]   0.7078479 33.14725
##
##
## Lower Band, CI= 0.95
## $nasd
##           nasd      amzn
##  [1,] 171.3068 13.056497
##  [2,] 155.0689 11.225272
##  [3,] 145.8963 11.210433
##  [4,] 132.2856 11.923121
##  [5,] 115.1678  5.831653
##  [6,] 118.5917  6.518417
##  [7,] 124.6677  6.546421
##  [8,] 130.8690  8.621261
##  [9,] 125.7846  8.831510
## [10,] 118.6559 10.336976
## [11,] 113.1278 12.537239
##
## $amzn
##             nasd      amzn
##  [1,]    0.00000 27.244238
##  [2,]  -39.05658 20.896324
##  [3,]  -48.55046 20.952045
##  [4,]  -45.18355 21.402231
##  [5,]  -87.61598 17.419330
##  [6,]  -94.96302 14.580178
##  [7,] -118.86362  8.099597
##  [8,]  -85.95934 19.827346
##  [9,]  -81.46370 16.918294
## [10,]  -66.33931 14.999591
```

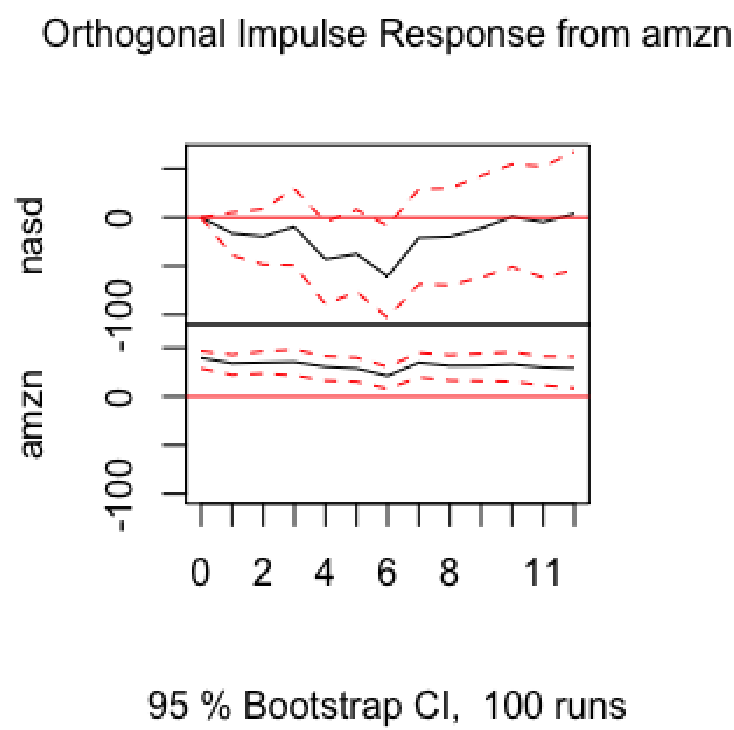
```
## [11,]  -56.10241 15.255597
##
##
## Upper Band, CI= 0.95
## $nasd
##             nasd     amzn
##  [1,] 223.1158 35.24403
##  [2,] 221.9505 30.01405
##  [3,] 221.5606 35.86138
##  [4,] 227.1121 37.50955
##  [5,] 207.5254 29.59820
##  [6,] 208.5506 32.59693
##  [7,] 233.9503 30.72972
##  [8,] 238.6157 32.63218
##  [9,] 239.9450 35.50894
## [10,] 228.7395 36.41610
## [11,] 223.5520 39.66017
##
## $amzn
##              nasd     amzn
##  [1,]  0.000000 47.40247
##  [2,] 12.316019 42.45884
##  [3,] 18.287973 45.72891
##  [4,] 36.497218 47.64971
##  [5,] 16.464157 43.01747
##  [6,] 21.113238 40.55682
##  [7,] -1.556363 30.97317
##  [8,] 40.529495 45.25019
##  [9,] 40.186166 42.00756
## [10,] 53.392926 43.37726
## [11,] 72.103535 43.54383
```

```
plot(irf(var_model, n.ahead=12))
```

Orthogonal Impulse Response from nasd



95 % Bootstrap CI, 100 runs



According to the implulse response plots, we could conclude that after 1 unit shock of NASDAQ, NASDAQ has initially a large movement in starts at all times, and Amazon has initially a little movement in starts at all times. After 1 unit shock of Amazon, NASDAQ initially produces no movement then builds up, reaching the lowest point around 6 months and then the effect decays slowly.

12. Perform a Granger-Causality test on your variables and discuss your results from the test.

```
grangertest(nasd ~ amzn, order = 8)

## Granger causality test
##
## Model 1: nasd ~ Lags(nasd, 1:8) + Lags(amzn, 1:8)
## Model 2: nasd ~ Lags(nasd, 1:8)
##   Res.Df Df      F    Pr(>F)
## 1    238
## 2    246 -8 3.5631 0.0006381 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

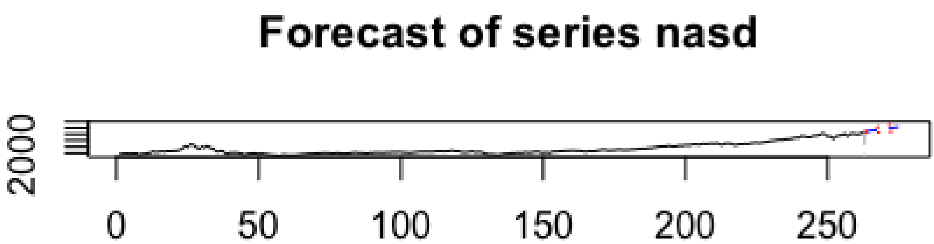
According to the results of Granger-Causality test, we could conclude that Amazon is the Granger-Cause of NASDAQ because the test is statistically significant.

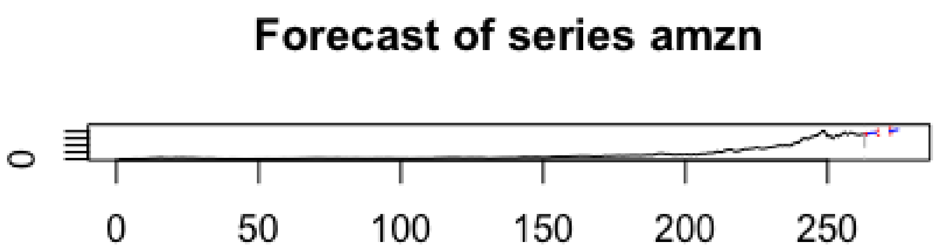
```
grangertest(amzn ~ nasd, order = 8)

## Granger causality test
##
## Model 1: amzn ~ Lags(amzn, 1:8) + Lags(nasd, 1:8)
## Model 2: amzn ~ Lags(amzn, 1:8)
##   Res.Df Df     F Pr(>F)
## 1    238
## 2    246 -8 1.251 0.2703
```

According to the results of Granger-Causality test, we could conclude that NASDAQ is not the Granger-Cause of Amazon because the test is not statistically significant.

13. Use your VAR model to forecast 12-steps ahead. Your forecast should include the respective error bands. Comment on the differences between the two forecasts (VAR vs. ARIMA).
    The difference between the VAR and ARIMA models is that VAR model includes more infomation to forecast. Both of the amzn lags and nasd lags are included in the VAR model and the ARIMA models include amzn only or nasd only for forecast.

```
var.predict = predict(object=var_model, n.ahead=12)
plot(var.predict)
```





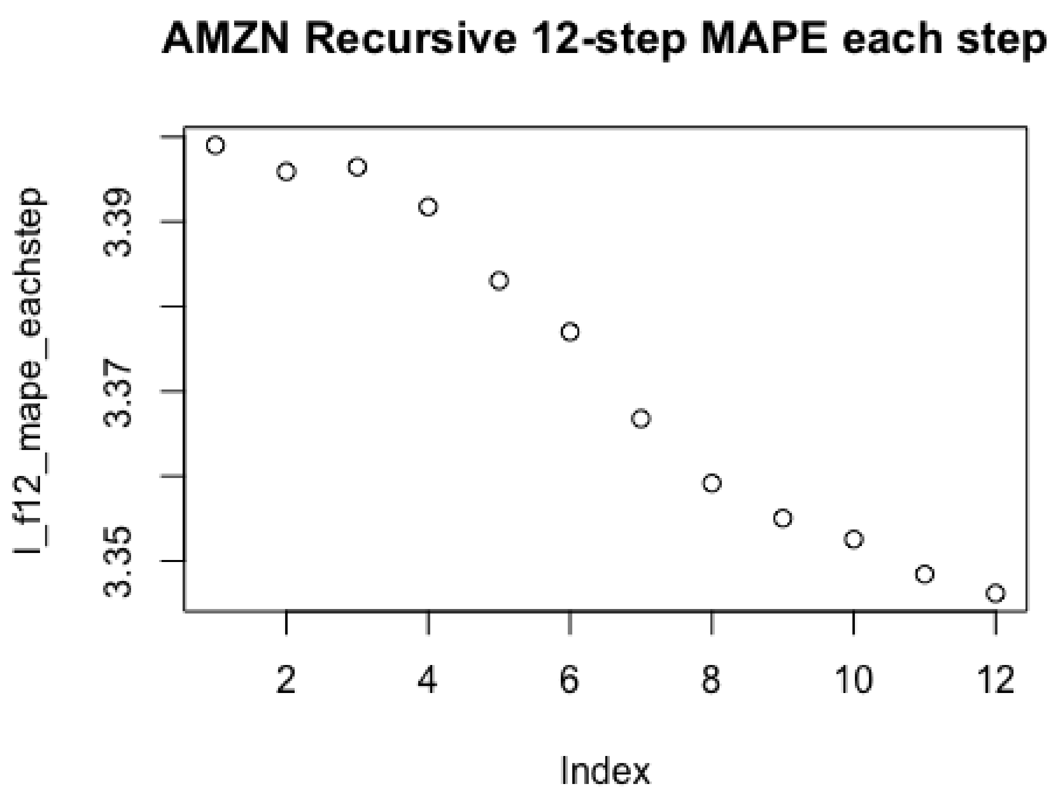

```
AIC(var_model,model_nasd,model_amzn)

## Warning in AIC.default(var_model, model_nasd, model_amzn): models ar
e not
## all fitted to the same number of observations

##            df      AIC
## var_model  34 6076.836
## model_nasd  3 3551.095
## model_amzn  4 2777.681

BIC(var_model,model_nasd,model_amzn)

## Warning in BIC.default(var_model, model_nasd, model_amzn): models ar
e not
## all fitted to the same number of observations

##            df      BIC
## var_model  34 6197.239
## model_nasd  3 3561.800
## model_amzn  4 2791.939
```

According to the results of AIC and BIC, the ARIMA models (model_nasd and model_amzn) have smaller AIC and BIC, which means the estimations are more close to true values and the models are better. As a result, the ARIMA models have better prediction.

14. Backtest your ARIMA model. Begin by partitioning your data set into an estimation set and a prediction set.

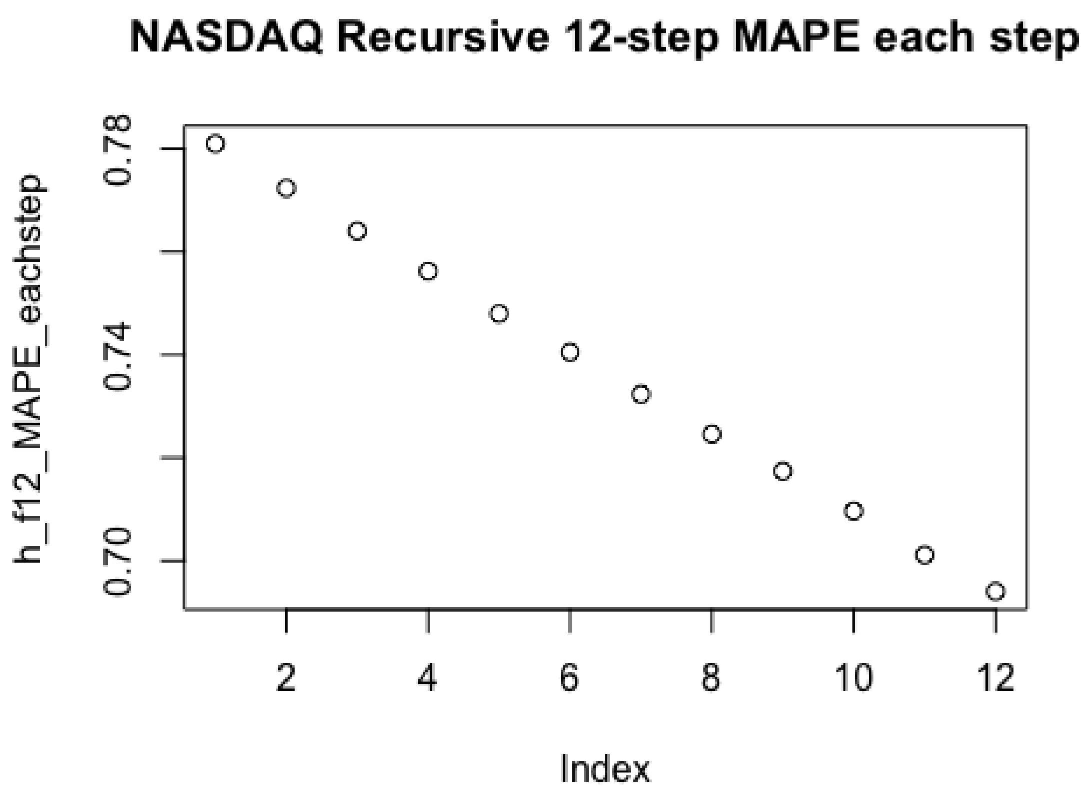
(a) Use a recursive backtesting scheme, and forecast 12-steps ahead at each iteration. Compute the mean absolute percentage error at each step. Provide a plot showing the MAPE over each iteration.

```
library(greybox)

## Package "greybox", v0.5.4 loaded.

ourCall="predict(arima(amzn,order=c(0,2,2)),n.ahead=h)"
ourValue="pred"
# separation : training 78 test 185
returnedValues1=ro(amzn,h=12,origins = 78,call=ourCall,value = ourValue
,ci=F,co=T)
l_f12_mape_eachstep=apply(abs(returnedValues1$holdout-returnedValues1$p
red),1,mean,na.rm=TRUE)/ mean(returnedValues1$actuals)
plot(l_f12_mape_eachstep,main="AMZN Recursive 12-step MAPE each step")
```



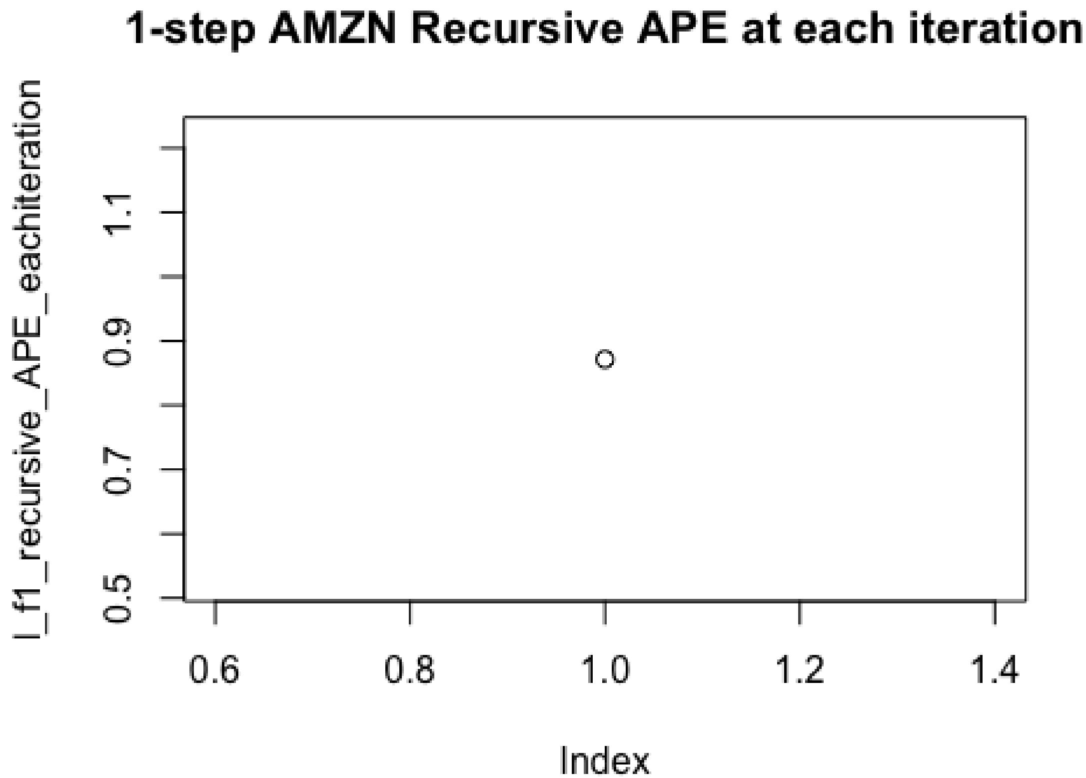

```
ourCall_N <- "predict(arima(nasd,order=c(0,1,0)),n.ahead=h)"
ourValue <- "pred"
```

```
N_returnedValues1 <- ro(nasd, h=12,origins = 68, call=ourCall_N, value=
ourValue)
N_returnedValues2 = ro(nasd,h=1,origins = 79,call=ourCall_N, value=ourV
alue)
h_f12_MAPE_eachstep=apply(abs(N_returnedValues1$holdout - N_returnedVal
ues1$pred),1,mean,na.rm=TRUE) / mean(N_returnedValues1$actuals)
plot(h_f12_MAPE_eachstep, main='NASDAQ Recursive 12-step MAPE each step
')
```

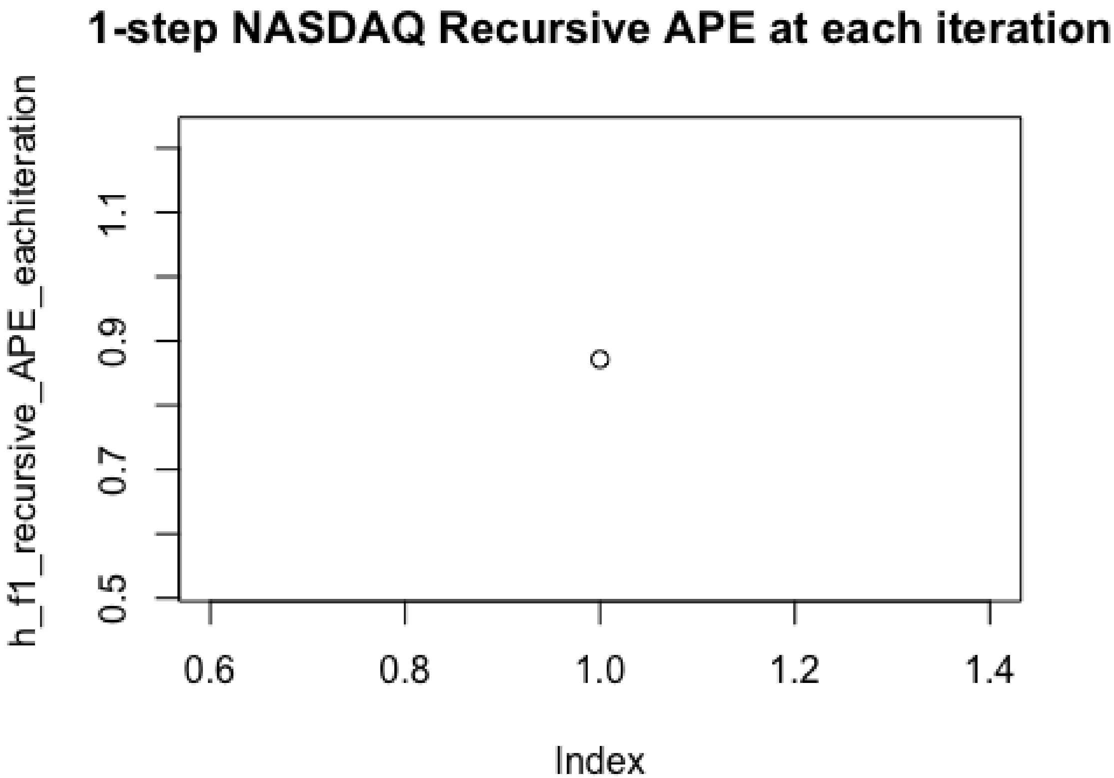


(b) Shorten your forecast horizon to only 1-step ahead. Compute the absolute percentage error at each iteration, and plot.

```
N_returnedValues2 = ro(nasd,h=1,origins = 79,call=ourCall_N, value=ourV
alue)
h_f12_MAPE_eachstep=apply(abs(N_returnedValues1$holdout - N_returnedVal
ues1$pred),1,mean,na.rm=TRUE) / mean(N_returnedValues1$actuals)
l_f1_recursive_APE_eachiteration=apply(abs(N_returnedValues2$holdout -
N_returnedValues2$pred),1,mean,na.rm=TRUE) / mean(N_returnedValues2$act
uals)
plot(l_f1_recursive_APE_eachiteration, main='1-step AMZN Recursive APE
at each iteration')
```




```
h_f1_recursive_APE_eachiteration=apply(abs(N_returnedValues2$holdout -
N_returnedValues2$pred),1,mean,na.rm=TRUE) / mean(N_returnedValues2$act
uals)
plot(h_f1_recursive_APE_eachiteration, main='1-step NASDAQ Recursive AP
E at each iteration')
```



(d) Now test your model using a moving window backtesting scheme. Forecast out 12-steps ahead at each iteration, and plot the forecast errors observed at each iteration. Repeat for a 1-step ahead forecast horizon. Provide plots of both.

```
library(greybox)
ourCall <- "predict(arima(stock$AMZN,order=c(0,2,2)),n.ahead=h)"
ourValue <- "pred"
nasdrol1<- ro(stock$AMZN, h=12,origins = 130, call=ourCall, value=ourVa
lue,TRUE,TRUE)
nasdrol12= ro(stock$AMZN,h=1,origins = 79,call=ourCall, value=ourValue,
TRUE,TRUE)
h_f1_rolling_FE_eachiteration = nasdrol1$mean-nasdrol1$holdout

plot(l_f1_rolling_FE_eachiteration)



library(greybox)
ourCall <- "predict(arima(stock$NASDAQ,order=c(0,1,0)),n.ahead=h)"
ourValue <- "pred"
nasdrol1<- ro(stock$NASDAQ, h=12,origins = 130, call=ourCall, value=our
Value,TRUE,TRUE)
nasdrol12= ro(stock$NASDAQ,h=1,origins = 79,call=ourCall, value=ourValu
```

```
e,TRUE,TRUE)
h_f1_rolling_FE_eachiteration = nasdrol1$mean-nasdrol1$holdout

plot(l_f1_rolling_FE_eachiteration)
```

III. Conclusion

Based on our result in Granger-Causality test, there is no casual effect between Nasdqe and Amazon stock price. Both Nasdqe and Amazon stock price show a positive trend based on our forecast. However, the stock market is highly volatile and unpredictable. In the future, there may be government policies or global issue that affect the trend of Nasdqe index and Amazon stock prices.

IV. References

Amazon.com, Inc. (AMZN) Stock Price, Quote, History & News. (2019, December 12). Retrieved from https://finance.yahoo.com/quote/AMZN?p=AMZN.

NASDAQ Composite (^IXIC) Charts, Data & News. (2019, December 13). Retrieved from https://finance.yahoo.com/quote/^IXIC?p=^IXIC.

Amazon (company). (2019, December 9). Retrieved from https://en.wikipedia.org/wiki/Amazon_(company). Nasdaq. (2019, November 30). Retrieved from https://en.wikipedia.org/wiki/Nasdaq.