Problem 2

a) Expain why the MA(q) models are regarded as finite memory models

A MA(q) process has finite memory since a shock only affects the process for periods of q ahead. Also, all autocorrelation functions for lags larger than q are equal to zero. Lastly, forecasts for MA(q) process converge to the mean after q steps ahead.

b) Discuss the ACF behavior of an ARMA(p,q) process

ACF of an Arma() decays geometically towards zero

Problem 3

```
library(tseries)
library(fBasics)
```

```
## Loading required package: timeDate
## Loading required package: timeSeries
##
## Rmetrics Package fBasics
## Analysing Markets and calculating Basic Statistics
## Copyright (C) 2005-2014 Rmetrics Association Zurich
## Educational Software for Financial Engineering and Computational Science
## Rmetrics is free software and comes with ABSOLUTELY NO WARRANTY.
## https://www.rmetrics.org --- Mail to: info@rmetrics.org
```

```
library(forecast)
library(lmtest)
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
## The following object is masked from 'package:timeSeries':
##
##     time<-
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

a) Import the data. In R you need to create a time object for index using the ts() function where the starting date is the first month of 1998, and frequencey is set equal to 12.

```r
setwd("~/Desktop/CSC425/hwork3/")
myd = read.table("INDPRO.csv", header = T, sep = ',')
head(myd)
```

```
##       date         rate
## 1 1/1/1998  0.005042245
## 2 2/1/1998  0.001123778
## 3 3/1/1998  0.000800165
## 4 4/1/1998  0.003449379
## 5 5/1/1998  0.006472101
## 6 6/1/1998 -0.006410126
```

```r
tail(myd)
```

```
##          date         rate
## 223  7/1/2016  0.002881344
## 224  8/1/2016 -0.000836184
## 225  9/1/2016 -0.002270310
## 226 10/1/2016  0.001949174
## 227 11/1/2016 -0.006613915
## 228 12/1/2016  0.008298091
```

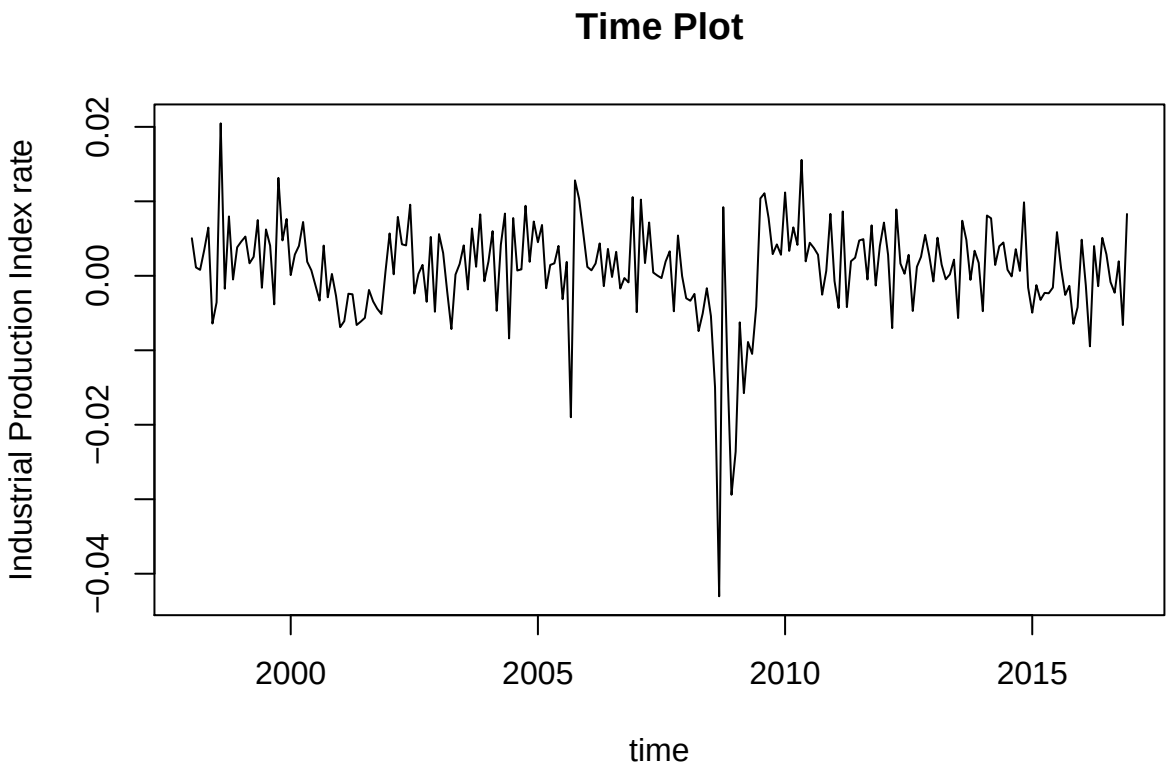
```r
date = myd[,1]
rate = myd[,2] # INDPRO index growth rate series
#create time series object
ratets = ts(rate, start = c(1998,1), freq = 12)
ratets
```

```
##                Jan           Feb           Mar           Apr           May
## 1998  0.0050422450  0.0011237780  0.0008001650  0.0034493790  0.0064721010
## 1999  0.0046014330  0.0052679620  0.0016842040  0.0025505560  0.0074867960
## 2000  0.0000916004  0.0028551460  0.0040217260  0.0072260030  0.0018903420
## 2001 -0.0068977730 -0.0060888770 -0.0024057330 -0.0024802850 -0.0066110990
## 2002  0.0056910940  0.0002026480  0.0079092930  0.0042245990  0.0040830650
## 2003  0.0055950230  0.0030564320 -0.0021913050 -0.0071516760  0.0001631760
## 2004  0.0019191370  0.0059970420 -0.0047030050  0.0041650850  0.0083579440
## 2005  0.0045134750  0.0068247890 -0.0016704050  0.0014600470  0.0016707600
## 2006  0.0011971940  0.0007391090  0.0016551290  0.0043494400 -0.0013891170
## 2007 -0.0048811600  0.0102301170  0.0017096720  0.0071624270  0.0004386400
## 2008 -0.0030144070 -0.0033479780 -0.0024301790 -0.0074237560 -0.0050624540
## 2009 -0.0236079830 -0.0062487840 -0.0157564830 -0.0088870320 -0.0104840950
## 2010  0.0111894110  0.0033403510  0.0065044710  0.0041491900  0.0155667910
## 2011 -0.0007005480 -0.0043265310  0.0086406680 -0.0042027780  0.0019598960
## 2012  0.0071427200  0.0028084490 -0.0070230330  0.0089098500  0.0016486440
## 2013 -0.0007601090  0.0051177970  0.0014988460 -0.0004693490  0.0001880250
## 2014 -0.0047517610  0.0080986040  0.0077227740  0.0014535550  0.0039638950
## 2015 -0.0049531440 -0.0012548280 -0.0032364230 -0.0022916170 -0.0023491260
## 2016  0.0048469320 -0.0012960370 -0.0094671210  0.0039932130 -0.0013944730
##                Jun           Jul           Aug           Sep           Oct
## 1998 -0.0064101260 -0.0035706240  0.0204701510 -0.0017317020  0.0079760570
## 1999 -0.0016092890  0.0062247160  0.0040485260 -0.0038331740  0.0131475210
## 2000  0.0007428980 -0.0012755520 -0.0033277210  0.0040606860 -0.0028860960
## 2001 -0.0061502270 -0.0056427030 -0.0018998550 -0.0034379360 -0.0044194740
## 2002  0.0095466440 -0.0023792210  0.0002479610  0.0014661180 -0.0034920720
## 2003  0.0015824390  0.0040850570 -0.0018385900  0.0063555630  0.0012149510
```

```
## 2004 -0.0084071360  0.0077605390  0.0007123480  0.0008673950  0.0094033850
## 2005  0.0040133120 -0.0031387960  0.0018668800 -0.0189901820  0.0128105410
## 2006  0.0036112460 -0.0001589900  0.0032193330 -0.0017027160 -0.0003009920
## 2007  0.0000361410 -0.0003090900  0.0019188530  0.0032881810 -0.0047642080
## 2008 -0.0016667620 -0.0054112600 -0.0149665490 -0.0430291090  0.0092165800
## 2009 -0.0041389540  0.0103921060  0.0110811820  0.0077402360  0.0029102970
## 2010  0.0019397950  0.0044404470  0.0037300310  0.0028260880 -0.0025365200
## 2011  0.0024280010  0.0047451250  0.0049189810 -0.0004928650  0.0067746050
## 2012  0.0002759880  0.0028131050 -0.0047281910  0.0012119520  0.0025480220
## 2013  0.0021702410 -0.0056746030  0.0073930090  0.0048228890 -0.0005464230
## 2014  0.0044737860  0.0008085820 -0.0000798424  0.0035751320  0.0006545110
## 2015 -0.0015805630  0.0058706900  0.0009746340 -0.0025677630 -0.0013512850
## 2016  0.0050967420  0.0028813440 -0.0008361840 -0.0022703100  0.0019491740
##                Nov           Dec
## 1998 -0.0004994950  0.0038232170
## 1999  0.0047530830  0.0076279430
## 2000  0.0002462260 -0.0027046990
## 2001 -0.0051251340  0.0005470560
## 2002  0.0052207490 -0.0048266760
## 2003  0.0082476380 -0.0007309140
## 2004  0.0018762090  0.0072892630
## 2005  0.0103324840  0.0058066590
## 2006 -0.0008837600  0.0105706120
## 2007  0.0054155830 -0.0000321577
## 2008 -0.0123207760 -0.0294012250
## 2009  0.0042148320  0.0028289890
## 2010  0.0006692030  0.0083145070
## 2011 -0.0013067830  0.0039977210
## 2012  0.0054932120  0.0027102640
## 2013  0.0033633150  0.0017329420
## 2014  0.0098699310 -0.0015803270
## 2015 -0.0064451160 -0.0042292300
## 2016 -0.0066139150  0.0082980910
```

b) Create the time plot of the index growth rate Xt and analyze trends displayed by the plot

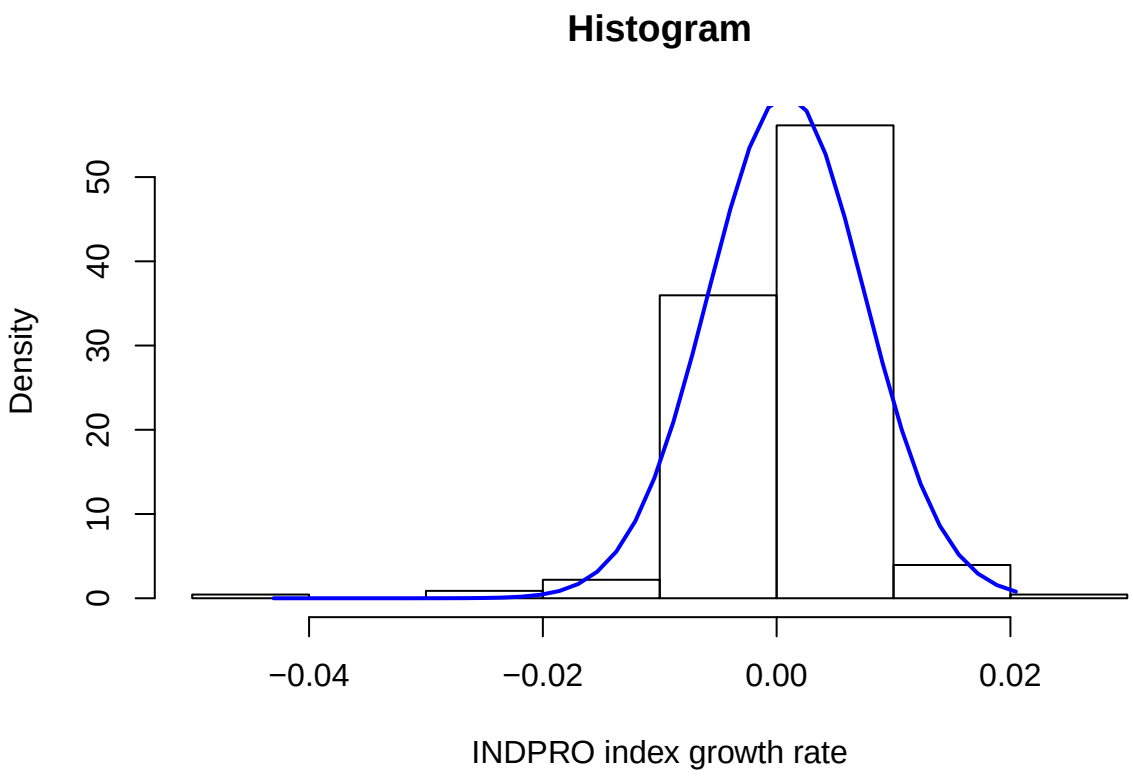
```
plot(ratets, type = 'l', xlab = 'time', ylab = 'Industrial Production Index rate', main = 'Time Plot')
```



By looking at the plot, it fluctuates over times and has no clear trends. It seems like its mean and variance over time are more likely constatnt rather than not constant. However, I see that there is a huge fluctuation between 2008 and 2009.

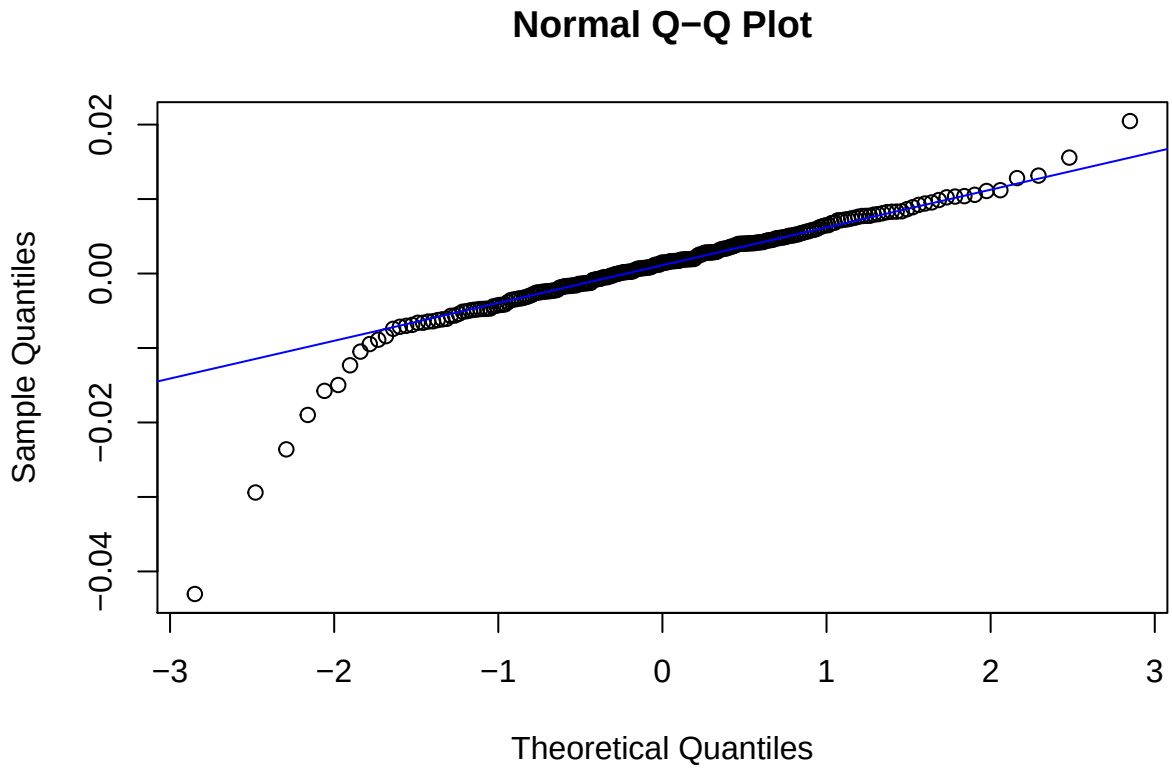
c) Analyze the distrigutrion of Xt. Can you assume that Xt is normally distributed?

```
#Histogram
par(mfcol = c(1,1))
hist(rate, xlab = 'INDPRO index growth rate', main = "Histogram", prob = T)
xfit<-seq(min(rate), max(rate), length = 40)
yfit<-dnorm(xfit, mean = mean(rate), sd = sd(rate))
lines(xfit, yfit, col = "blue", lwd = 2)
```



based on the histogram, it is skewed to the left, but I could say it is close to normal distribution

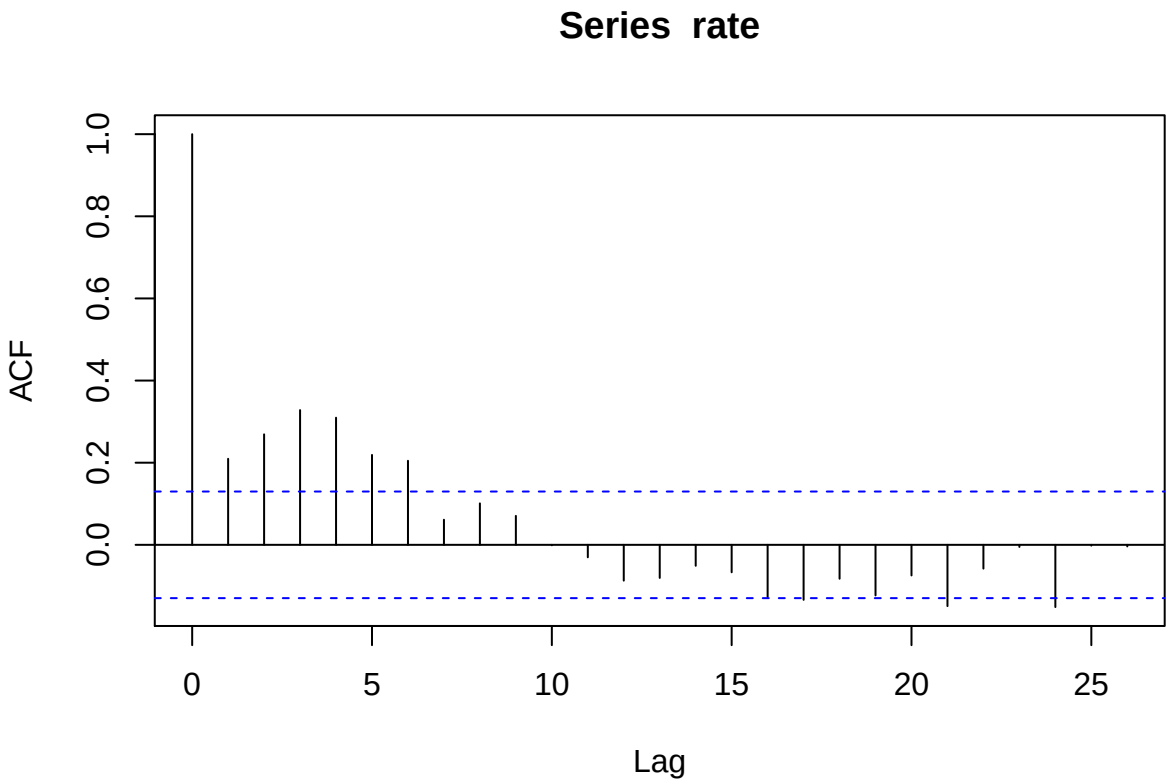
```
#qq-plot
qqnorm(rate)
qqline(rate, col = "blue")
```



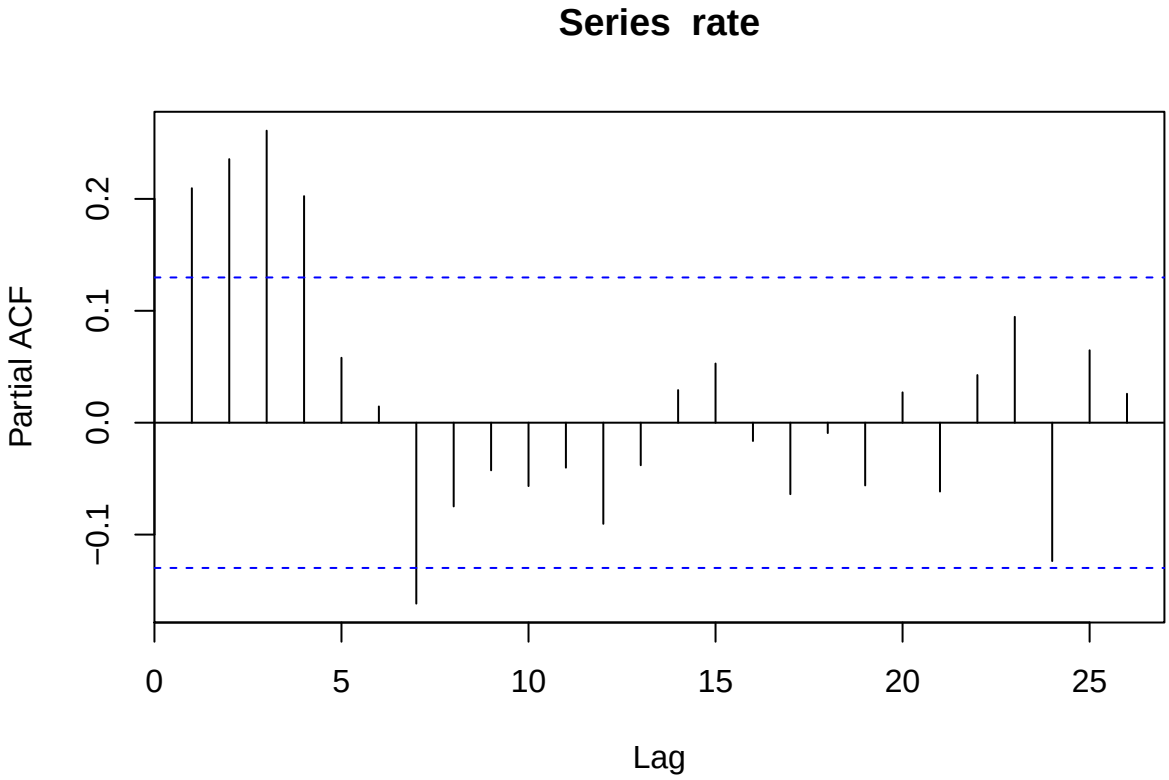
based on the qq-plot, it is very close to normal distribution.

d) Analyze the ACF and the PACF functions and discuss the following questions

```
acf(rate, plot = T, lag = 26)
```




```
pacf(rate, lag = 26)
```



d-1) Does the TS exhibits a stationary behavior?

It is not a perfect stationary, but close to stationary. There are just few significances to be observed over time. Other points are close to zeros.

d-2) Does the process show more of an AR behavior or MA behavior

PACF cuts off after the 4th lag, while the ACF tail off to zero. In this reason, I decide that a model of AR(4) could work for this analysis.

e) Apply the BIC criteria using the auto.arima(xvar, ic = c("bic"))function, and fit the model suggested by the BIC criterion. (M1)

```
m1 = auto.arima(ratets, max.P=8, max.Q=8, ic = ("bic"), trace = T, stationary = T)
```

```
##
##  Fitting models using approximations to speed things up...
##
##  ARIMA(2,0,2)(1,0,1)[12] with non-zero mean : -1657.383
##  ARIMA(0,0,0)            with non-zero mean : -1627.824
##  ARIMA(1,0,0)(1,0,0)[12] with non-zero mean : -1632.173
##  ARIMA(0,0,1)(0,0,1)[12] with non-zero mean : -1626.184
##  ARIMA(0,0,0)            with zero mean     : -1629.715
##  ARIMA(2,0,2)(0,0,1)[12] with non-zero mean : -1655.759
##  ARIMA(2,0,2)(2,0,1)[12] with non-zero mean : -1649.472
##  ARIMA(2,0,2)(1,0,0)[12] with non-zero mean : -1659.219
```

```
##  ARIMA(1,0,2)(1,0,0)[12] with non-zero mean : -1659.184
##  ARIMA(3,0,2)(1,0,0)[12] with non-zero mean : -1653.089
##  ARIMA(2,0,1)(1,0,0)[12] with non-zero mean : -1653.734
##  ARIMA(2,0,3)(1,0,0)[12] with non-zero mean : -1654.633
##  ARIMA(1,0,1)(1,0,0)[12] with non-zero mean : -1654.316
##  ARIMA(3,0,3)(1,0,0)[12] with non-zero mean : -1648.008
##  ARIMA(2,0,2)(1,0,0)[12] with zero mean     : -1664.188
##  ARIMA(2,0,2)            with zero mean     : -1663.718
##  ARIMA(2,0,2)(2,0,0)[12] with zero mean     : -1659.891
##  ARIMA(2,0,2)(1,0,1)[12] with zero mean     : -1662.4
##  ARIMA(2,0,2)(2,0,1)[12] with zero mean     : -1654.537
##  ARIMA(1,0,2)(1,0,0)[12] with zero mean     : -1664.324
##  ARIMA(1,0,1)(1,0,0)[12] with zero mean     : -1659.476
##  ARIMA(1,0,3)(1,0,0)[12] with zero mean     : -1665.736
##  ARIMA(0,0,2)(1,0,0)[12] with zero mean     : -1633.876
##  ARIMA(2,0,4)(1,0,0)[12] with zero mean     : -1656.216
##  ARIMA(1,0,3)(1,0,0)[12] with non-zero mean : -1660.681
##  ARIMA(1,0,3)            with zero mean     : -1663.921
##  ARIMA(1,0,3)(2,0,0)[12] with zero mean     : -1659.129
##  ARIMA(1,0,3)(1,0,1)[12] with zero mean     : -1664.133
##  ARIMA(1,0,3)(2,0,1)[12] with zero mean     : -1653.7
##  ARIMA(0,0,3)(1,0,0)[12] with zero mean     : -1640.814
##  ARIMA(2,0,3)(1,0,0)[12] with zero mean     : -1659.646
##  ARIMA(1,0,4)(1,0,0)[12] with zero mean     : -1660.895
##
##  Now re-fitting the best model(s) without approximations...
##
##  ARIMA(1,0,3)(1,0,0)[12] with zero mean     : -1661.046
##
##  Best model: ARIMA(1,0,3)(1,0,0)[12] with zero mean
```

e-1) Examine the significance of the model coefficients and analyze the residuals to check adequacy of the model

BIC() suggests that ARIMA(1,0,3)(1,0,0)[12] with zero mean is the best model.

```
coeftest(m1)
```

```
##
## z test of coefficients:
##
##       Estimate Std. Error z value  Pr(>|z|)
## ar1   0.821877   0.061570 13.3486 < 2.2e-16 ***
## ma1  -0.822426   0.083459 -9.8542 < 2.2e-16 ***
## ma2   0.138905   0.081621  1.7018  0.088787 .
## ma3   0.182975   0.068459  2.6728  0.007523 **
## sar1 -0.106368   0.068149 -1.5608  0.118566
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

By coefficient test, ma2 and sar1 have pretty high p-values for t-test and we could drop those two values. However, I am not able to see Intercept by using ARIMA(1,0,3)(1,0,0)

Thus, I will check arima(1,0,3) instead if this function shows the Intercept.

```
m11 = Arima(ratets, c(1,0,3), method = 'ML')
coeftest(m11)
```

```
##
## z test of coefficients:
##
##              Estimate  Std. Error z value  Pr(>|z|)
## ar1        0.81457954  0.06140278 13.2662 < 2.2e-16 ***
## ma1       -0.81045702  0.08385289 -9.6652 < 2.2e-16 ***
## ma2        0.12436064  0.07992196  1.5560  0.119702
## ma3        0.18883096  0.06627749  2.8491  0.004384 **
## intercept  0.00086275  0.00104399  0.8264  0.408581
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

I am now able to check intercept, but its p-value is so high as well as the p-value for MA2. Hence, I could drop these two, but I will put 0 by using fixed() in Arima.

Instead of drop them, I put 0 by using fixed option.

```
m111 = Arima(ratets, c(1,0,3), fixed = c(NA,NA,0,NA,0), method = 'ML')
coeftest(m111)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error  z value  Pr(>|z|)
## ar1  0.819617   0.059892  13.6849 < 2.2e-16 ***
## ma1 -0.753919   0.067651 -11.1442 < 2.2e-16 ***
## ma3  0.251288   0.055065   4.5634 5.032e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Finally, the t-tests are good.

Now, I am going to check adequacy of the models.

```
# Box test for m1$residuals
Box.test(m1$residuals, lag = 4, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  m1$residuals
## X-squared = 0.48873, df = 4, p-value = 0.9746
```

```
Box.test(m1$residuals, lag = 6, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  m1$residuals
## X-squared = 1.095, df = 6, p-value = 0.9818
```

Ljung-Box test on m1$residuals cannot reject hypothesis of white noise time series

```
# Box test for m11$residuals
Box.test(m11$residuals, lag = 4, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  m11$residuals
## X-squared = 0.7197, df = 4, p-value = 0.9489
```

```
Box.test(m11$residuals, lag = 6, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  m11$residuals
## X-squared = 1.2721, df = 6, p-value = 0.9732
```

The p-values for Ljung-Box test on m11$residuals are still high, so cannot reject hypothesis of white noise time series

```
# Box test for m111$residuals
Box.test(m111$residuals, lag = 4, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  m111$residuals
## X-squared = 4.1008, df = 4, p-value = 0.3925
```

```
Box.test(m111$residuals, lag = 6, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  m111$residuals
## X-squared = 4.8653, df = 6, p-value = 0.5612
```

Ljung-Box test on m1_1$residuals get imporved but still cannot reject hypothesis of white noise time series. Therefore,

e-2) Write down the model expression and discuss if this is a good model for the data

For m1 model, I have not leared expressing the models with (sar1), also this model does not have an intercept. so I conclude with this model cannot be expressed based on lecture we have covered.

For m11 model, the model expression could be X_t - 0.00086275 = 0.81457954(X_(t-1) - 0.00086275) + a_t - 0.81045702(a_(t-1)) + 0.12436064(a_(t-2)) + 0.18883096(a_(t-3))

This model is not a good model for this analysis. Since it does not have a white noise, and I had some bad coefficients results.

f) Find and alternative model(M2) for the index growth rate time series, using either an MA(q) or an AR(p) model depending on your analysis in d). Make sure the model coefficients are significant and the residuals are white noise. Write down the model expression.

```
# alternative model AR(4) since the PACF cuts off after lag 4
m2<-Arima(ratets, c(4,0,0), method = 'ML')
coeftest(m2)
```

```
##
## z test of coefficients:
##
##             Estimate Std. Error z value  Pr(>|z|)
## ar1       0.04422845 0.06497972  0.6807 0.4960930
## ar2       0.15370358 0.06297554  2.4407 0.0146594 *
## ar3       0.23860823 0.06299084  3.7880 0.0001519 ***
## ar4       0.20093968 0.06480786  3.1005 0.0019316 **
## intercept 0.00087412 0.00106987  0.8170 0.4139121
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

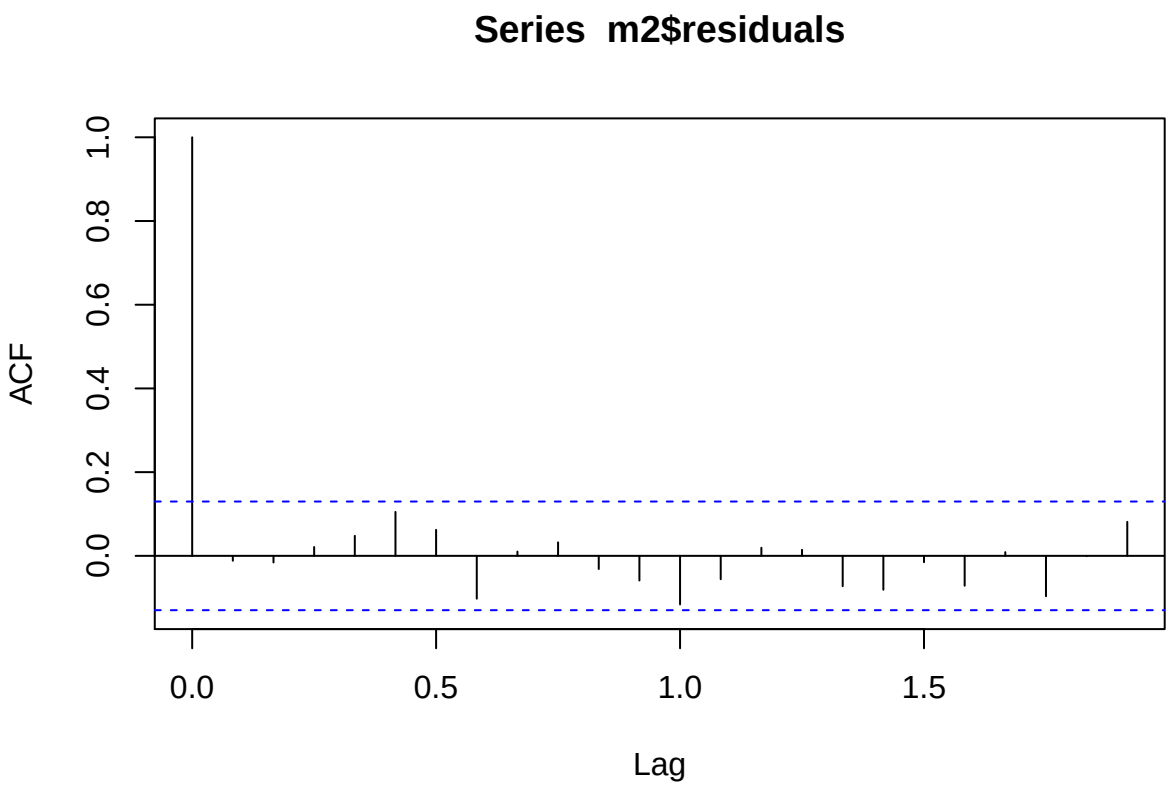
coeftest() shows that t-test for ar2,ar3,ar4 are very good, but failed for ar1 and intercept. so we could drop them

```
Box.test(m2$residuals, lag = 4, fitdf = 4, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  m2$residuals
## X-squared = 0.73156, df = 0, p-value < 2.2e-16
```

I got 2.2e-16 for p-value which is good.

```
acf(m2$residuals)
```



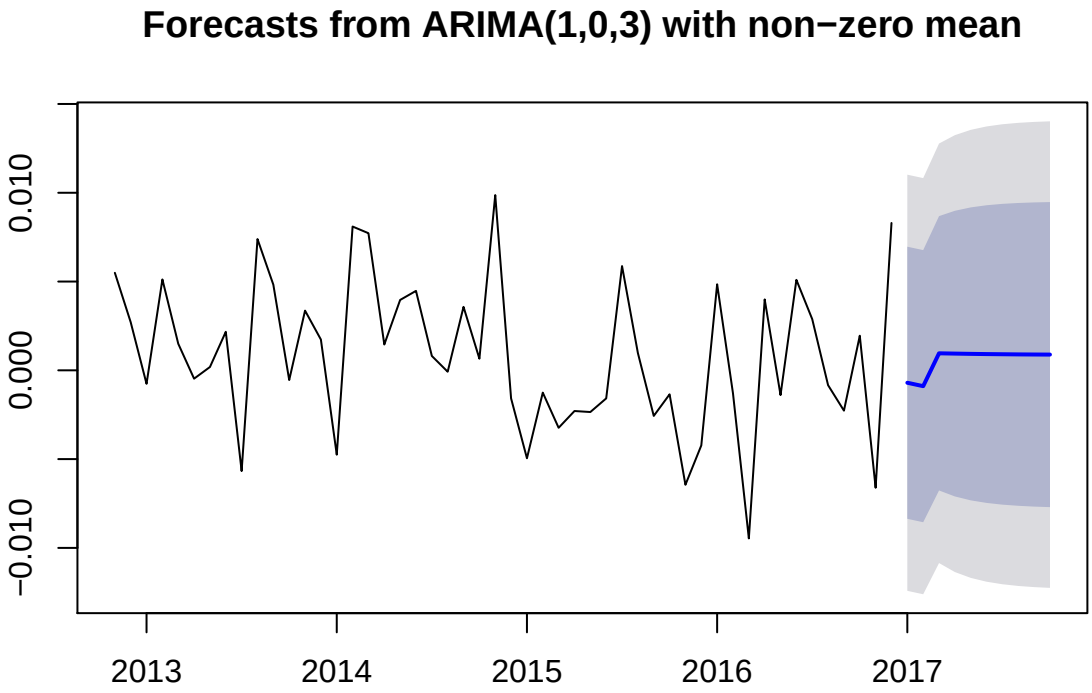
acf the residuals looks pretty good, thus white noise.

the model expression :

x_t - 0.00087412 = 0.04422845(x_(t-1) - 0.00087412) + 0.15370358(x_(t-2) - 0.00087412) + 0.23860823(x_(t-3) -0.00087412) + 0.20093968(x_(t-4) -0.00087412) + a_t

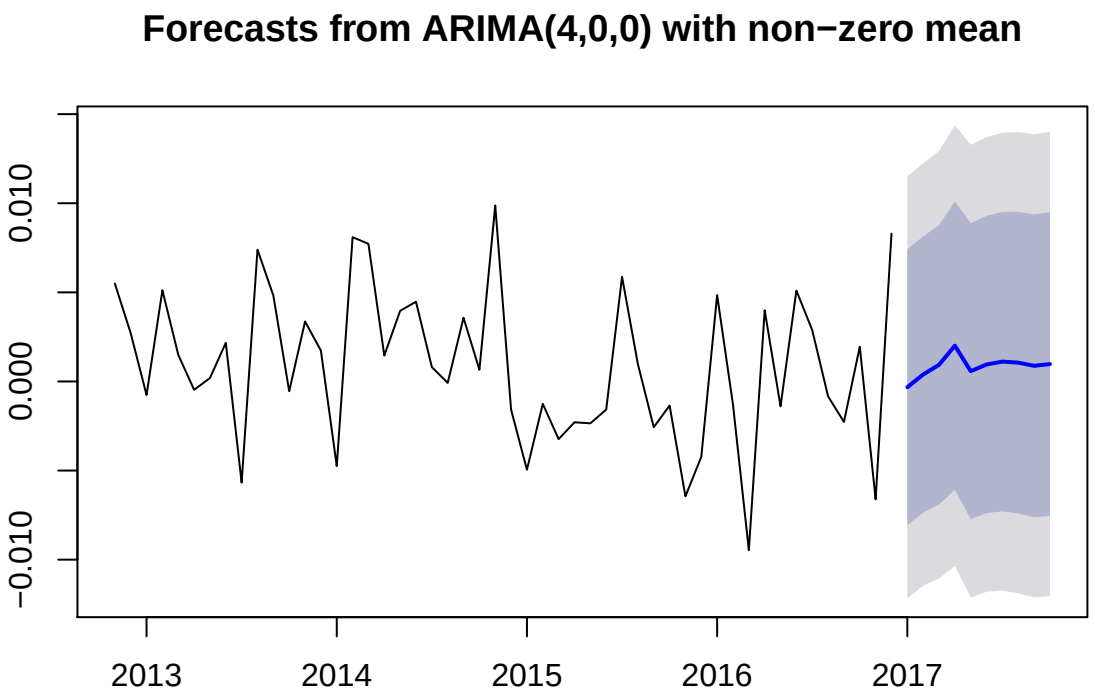
g) Plot the forecasts for the model M1 and M2 and compare their behavior. Are teh forecasts similar in trends? Do you notice any major difference?

```
pred1 = forecast(m1)
plot(forecast(m11, h = 10), include = 50)
```



This trend show there would be a flat trend after 2017 and this forcasting answers me with no interesting patterns.

```
#alternative model
plot(forecast(m2, h = 10), include = 50)
```



Based on the alternative model forecasting, there will be some gentle flcutuations after 2017 with a soft-postive trend. Also the changes around values of 0.000.

h) Apply the backtesting procdure using 85% (=194 values) of the data for training and 15% for testing to evaluate the forecasting power fo both model.

```
source('backtest.R')
pm1 = backtest(m11, ratets, 194, 1)
```

```
## [1] "RMSE of out-of-sample forecasts"
## [1] 0.004749266
## [1] "Mean absolute error of out-of-sample forecasts"
## [1] 0.003861951
## [1] "Mean Absolute Percentage error"
## [1] 2.573491
## [1] "Symmetric Mean Absolute Percentage error"
## [1] 1.478262
```

```
pm2 = backtest(m2, ratets, 194, 1)
```

```
## [1] "RMSE of out-of-sample forecasts"
## [1] 0.004635424
## [1] "Mean absolute error of out-of-sample forecasts"
## [1] 0.003664983
## [1] "Mean Absolute Percentage error"
## [1] 2.00443
## [1] "Symmetric Mean Absolute Percentage error"
## [1] 1.411719
```

Based on the results, both models are pretty good since all RMSE, MAE, MAPE, SMAPE are very low.

RMSE for m1 is 0.004749266 and m2 is 0.004635424. Thus m2 has more accuracy MAE for m1 is 0.003861951 and m2 is 0.003664983. Thus m2 has more accuracy MAPE for m1 is 2.573491 and m2 is 2.00443. Thus m2

has more accuracy SMAPE for m1 is 1.478262 and m2 is 1.411719 Thus m2 has more accuracy

i) At the end of this analysis, which model would you recommend for this analysis? Discusss the trend captured by it.

I think model 2 is more adequate for this analysis since it is more accurate based on the result from backtest. Also, it has a white noise, and shows a good result from coeftest(). Lastly, based on forecast(), the trend for model 2 shows some positive trend. In these reasons, model 2 is a better model in this analysis.