Lab3. ALTSALES

1. Load Libraries

```
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
```

2. Import Data

```
setwd("~/Desktop/CSC425/week3/")
myd = read.table("ALTSALES.csv", header = T, sep = ',')
head(myd)
```

```
##       date     rate
## 1 2/1/1976  4.25192
## 2 3/1/1976  0.31432
## 3 4/1/1976 -1.29920
## 4 5/1/1976 -2.09059
## 5 6/1/1976  0.64848
## 6 7/1/1976  3.10364
```

```
tail(myd)
```

```
##          date     rate
## 474  7/1/2015  3.04943
## 475  8/1/2015  1.47673
## 476  9/1/2015  1.91776
## 477 10/1/2015  0.30439
## 478 11/1/2015 -0.37519
## 479 12/1/2015 -4.64665
```

3. Create the tiems object ratets

```
# use function ts()
ratets = ts(myd[,2], start = c(1976,2), freq = 12)
ratets
```

```
##             Jan       Feb       Mar       Apr       May       Jun       Jul
## 1976            4.25192   0.31432  -1.29920  -2.09059   0.64848   3.10364
## 1977  -1.25379   2.25408   3.11824  -0.19618  -1.00319  -0.23280  -1.81868
## 1978  -9.83839   4.76226   5.33324   7.08849   0.46329   0.24005  -4.92816
## 1979  -1.43541   1.57254  -0.11443  -3.60536  -2.36298 -11.47787  10.91159
## 1980   6.94802  -8.13319  -9.02290 -13.55429  -8.70759   9.35553  11.78666
## 1981   2.96697   9.78615   0.56949 -16.19204  -0.74422   0.68074   0.00980
## 1982  13.85467   6.45161  -2.49883  -8.75968  14.24379 -13.57798   1.75159
## 1983  -3.47693  -2.31100   3.74485   6.23154   3.11115   7.75390  -0.40673
## 1984  -1.96106   0.94299  -1.15357   0.90929   3.37023  -1.20118   0.42379
## 1985   6.70348   1.09401  -2.48186   1.16287   0.30215  -3.07793   2.85135
## 1986   2.56060  -4.15056  -8.10525  12.07914   2.94627  -2.50031  -1.06158
## 1987 -31.64436  22.00740   1.26096   0.99887  -5.82844   7.40741   0.20207
## 1988   2.70939   2.54914  -2.00346  -4.08253   2.66430  -0.51903  -0.96618
## 1989  -6.71512  -2.32403  -2.08617   9.30233  -4.81590  -5.86936   2.30359
## 1990  20.62081 -12.14241   0.97398  -1.36591  -1.89878   0.90228  -0.56249
## 1991  -8.70729   4.99310   3.62218  -6.26982   3.76321   1.91524   1.89524
## 1992  -0.42673   2.67648  -0.83478  -2.23952   4.46791   3.55365  -5.38410
## 1993  -2.37531  -3.75199   2.70121   9.15574  -0.16157  -0.16887  -0.31717
## 1994   2.42523   1.05753  -1.44794   4.71484  -9.12628   3.60727  -1.63246
## 1995  -5.48171   0.65446   2.82216  -5.95358   3.13305   4.16840  -4.61416
## 1996  -8.90808   5.30140   3.04962  -3.61630   3.38804  -4.86431  -1.10334
## 1997   3.25274  -2.71328   3.53982  -4.85625   0.15653  -3.64884   7.15092
## 1998 -10.33066   3.04861   1.09172   3.46644   7.26757  -1.62773 -12.96251
## 1999  -5.01092   2.93277  -1.41253   0.55107   3.87894  -1.28964   1.97161
## 2000   1.80408   4.22325  -5.59881  -2.13220   0.14333  -2.22706  -1.32920
## 2001   8.96974   1.05501  -3.24098  -1.99787  -0.16938   3.64176  -5.73550
## 2002   0.45210   4.84587  -1.28190   3.30593  -8.64326   4.93562   7.23565
## 2003  -6.63900  -3.71994   2.27014   1.60762  -1.70997   3.30609   0.53937
## 2004  -4.04308   2.01779   1.22039  -2.04312   7.67598 -11.25063   7.03636
## 2005  -7.13111   0.19548   3.21302   2.03792  -2.00301   6.14367  14.68722
## 2006   5.29722  -6.01525  -0.64184   0.96898  -2.40222   1.12554   4.72725
## 2007  -1.01394   1.84745  -4.12476   1.24883   0.49954  -2.90255  -2.05397
## 2008  -2.13131  -1.41065  -2.44626  -3.56201   0.67283  -2.08159  -9.59829
## 2009  -5.59116  -5.75517   5.86280  -3.71650   8.68762  -0.40016  14.20249
## 2010  -3.54430  -5.24934  14.29561  -2.63135   5.09379  -3.69650   2.89855
## 2011   1.16288   2.50659   0.39717   1.45051  -8.20399  -3.06513   6.10930
## 2012   3.70316   4.12798  -2.48971   1.81473  -2.91537   0.56927  -0.15566
## 2013   0.95920   1.05421  -1.10117   0.18883  -0.02600   1.83969   0.22980
```

```
## 2014  -0.86900   1.48502   6.10456  -1.52491   2.64668   0.58300  -1.70899
## 2015  -1.01172  -1.88180   4.55882  -2.14487   5.60546  -3.85619   3.04943
##            Aug       Sep       Oct       Nov       Dec
## 1976  -4.02378   4.88328  -5.08744   3.74093   9.15731
## 1977   1.27219  -1.47708   1.66036  -0.79250   2.72992
## 1978   3.91091  -9.44756   8.42550  -2.44299  -1.17216
## 1979   2.09306  -0.32859  -8.74364  -0.51834   4.19199
## 1980  -4.57781  -2.19153   4.93316  -1.18428  -2.68749
## 1981  22.05565 -12.79602 -15.22600   0.92301  -4.78804
## 1982   2.02400  12.05645  -8.91586  18.88588  -6.68296
## 1983  -5.37972   1.74303   6.89346  -1.55690  10.69075
## 1984  -3.10619  -2.45609   4.97731   0.71120  -0.95541
## 1985   9.26948  13.08243 -26.16832   3.21164   7.35366
## 1986   9.94764  24.75603 -31.59606   0.19978  22.25507
## 1987   9.19854  -5.75480 -11.66245   1.80322   7.92156
## 1988  -1.93171  -2.44064   0.71380   1.50523   7.04881
## 1989  12.82229  -5.37760 -13.61240  -1.52223   1.56872
## 1990  -1.50120   3.26167  -3.86453  -4.49455  -1.35901
## 1991  -3.06074   4.35557  -6.78821   2.90470   0.45293
## 1992   0.15873   3.91442   1.93686  -3.06703   4.29079
## 1993  -6.13024   3.01296   6.77830  -0.43827   0.96293
## 1994   3.21581  -0.52038   4.06411  -0.10956  -1.96129
## 1995   4.74661  -0.31321  -3.26894   3.74568   5.66214
## 1996   2.91156   0.37017  -1.39621   2.80524  -3.52781
## 1997   3.75806  -6.68570   2.20923   2.56052   4.13722
## 1998   0.72255  10.47500   5.15067  -6.28335   8.39358
## 1999  -0.29119  -0.02920   0.13438  -0.39092   4.22329
## 2000   1.41831   6.75834  -6.18252  -5.01256  -2.63239
## 2001  -0.66365   0.24975  35.20802 -18.35184  -8.90268
## 2002   1.54804  -9.82601  -2.46846   1.70822   8.63229
## 2003   6.89080  -5.50413  -4.73296   6.52915  -1.19207
## 2004  -0.90694   4.27708  -2.15695  -0.84428   4.22777
## 2005 -17.90654  -2.89649  -9.61831   7.91406   4.15678
## 2006  -7.14161   3.25745  -0.59074  -1.20076   2.74075
## 2007   3.45851   1.02906  -0.44447  -0.55187  -1.99526
## 2008   8.80063  -8.28394 -15.93632  -3.85337  -1.11165
## 2009  28.11785 -35.84129  10.95656   4.31051   2.24646
## 2010   0.74264  -0.83884   4.23823  -1.05746   2.59321
## 2011  -0.37250   6.03918   2.51418  -0.82997   1.80201
## 2012   0.27638   4.68551  -2.24803   4.13674   0.94171
## 2013  -0.07005  -1.88643  -0.17538   4.83472  -4.28900
## 2014   4.71153  -4.65629   0.24967   3.38942  -1.28077
## 2015   1.47673   1.91776   0.30439  -0.37519  -4.64665
```

4. Compute summary statistics

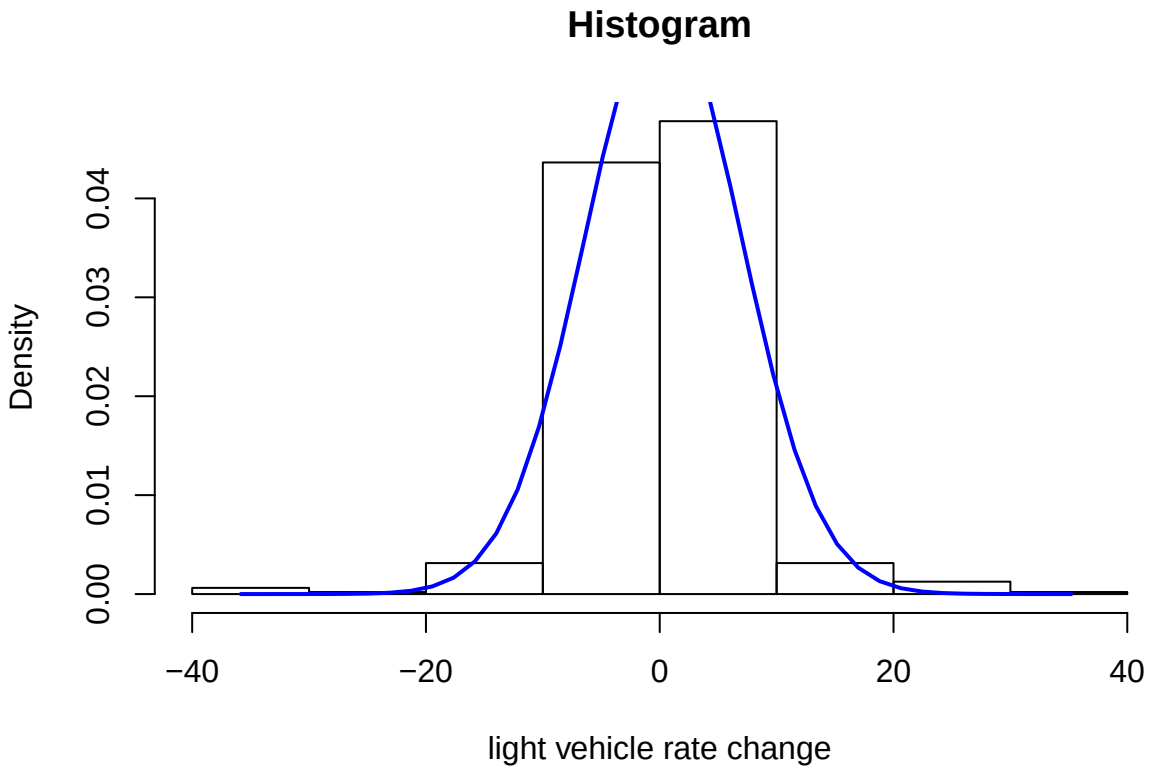
```
basicStats(myd$rate)
```

```
##             X..myd.rate
## nobs         479.000000
## NAs            0.000000
## Minimum      -35.841290
## Maximum       35.208020
```

```
## 1. Quartile   -2.441815
## 3. Quartile    3.306010
## Mean           0.296688
## Median         0.249670
## Sum          142.113560
## SE Mean        0.305955
## LCL Mean      -0.304494
## UCL Mean       0.897870
## Variance      44.838339
## Stdev          6.696144
## Skewness      -0.201682
## Kurtosis       6.432816
```

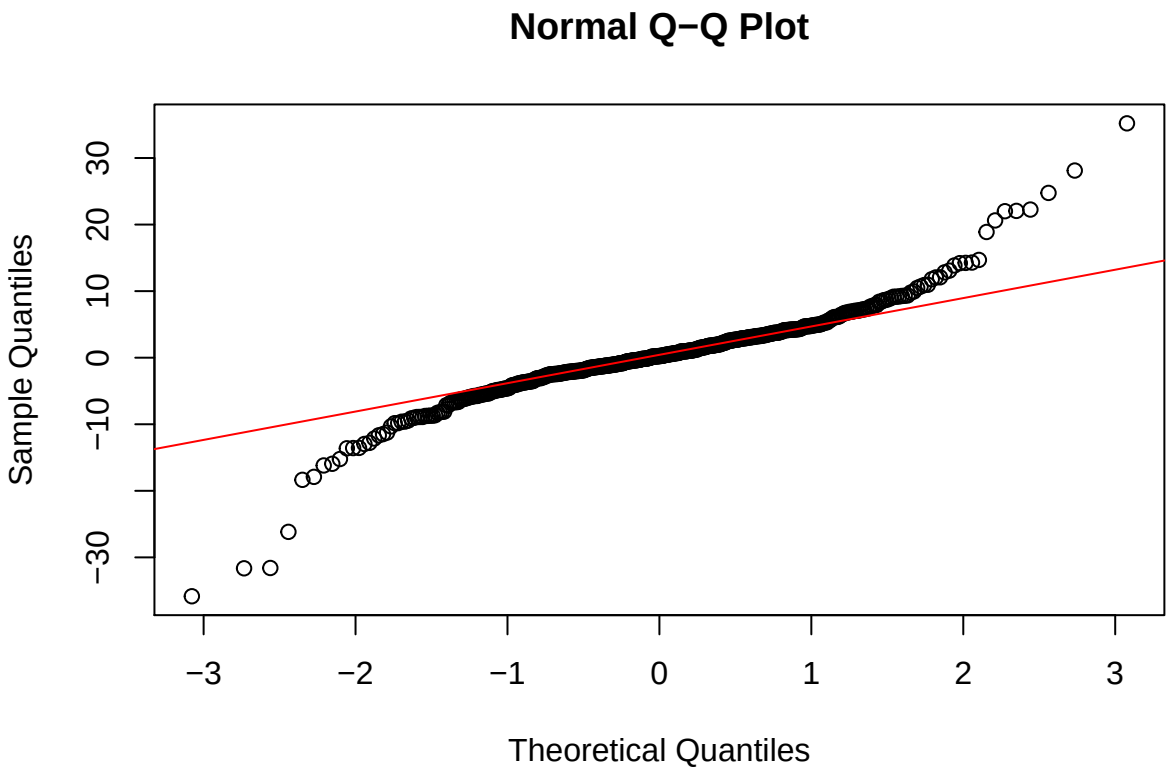
5. Creat Histogram

```
hist(myd$rate, xlab = "light vehicle rate change", prob = T, main = "Histogram")
xfit <- seq(min(myd$rate), max(myd$rate), length = 40)
yfit <- dnorm(xfit, mean = mean(myd$rate), sd = sd(myd$rate))
lines(xfit, yfit, col = "blue", lwd = 2)
```



6. Creat normal probability plot

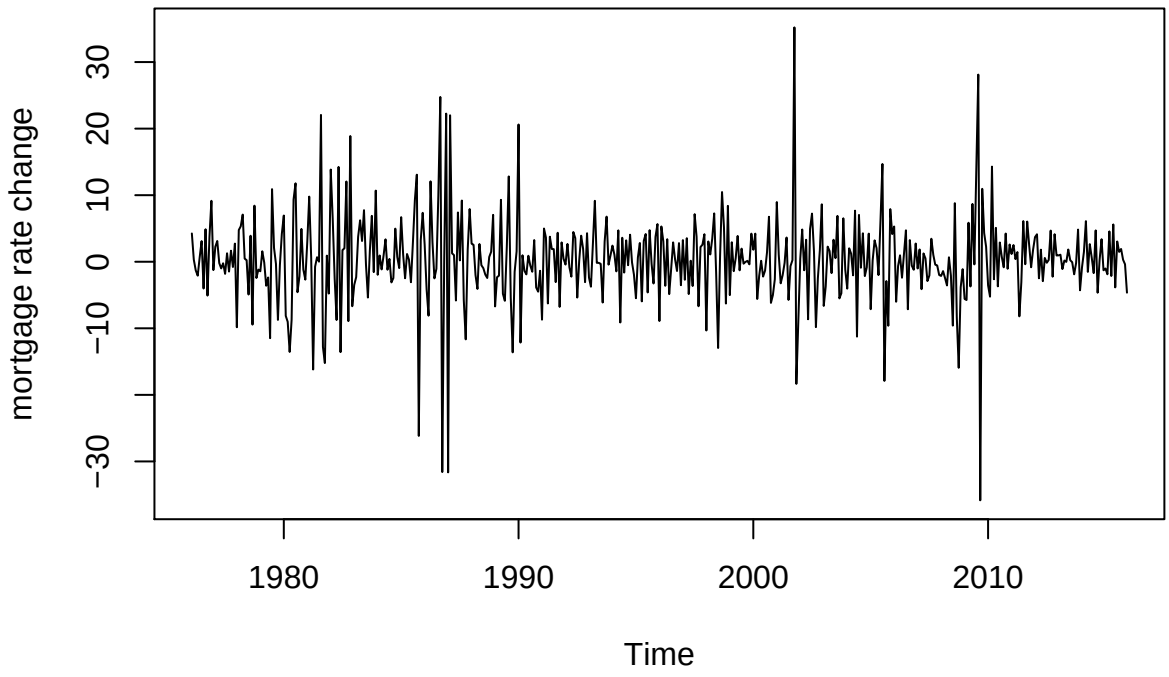
```
qqnorm(myd$rate)
qqline(myd$rate, col = 2)
```



###

7. Creat Time plot

```
plot(ratets, ylab = 'mortgage rate change')
```



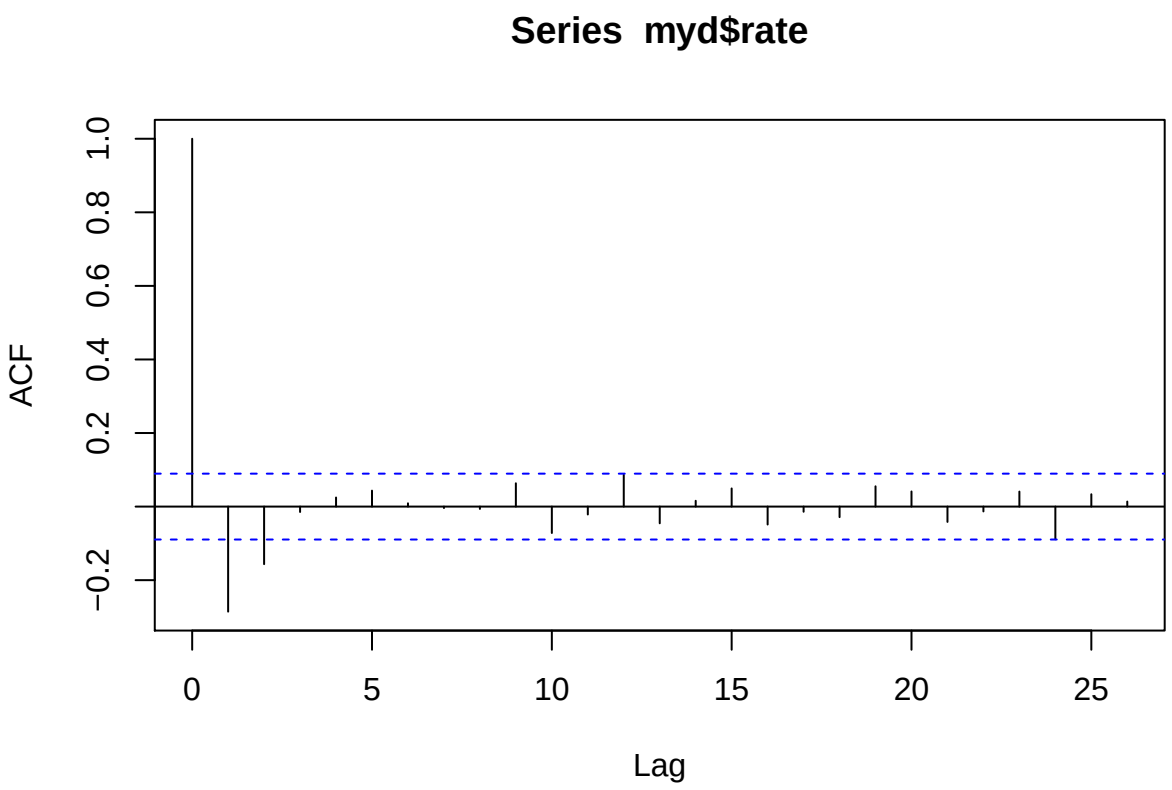
8. Normality tests

```
# Perform JB normality test
normalTest(myd$rate, method = c("jb"))
```

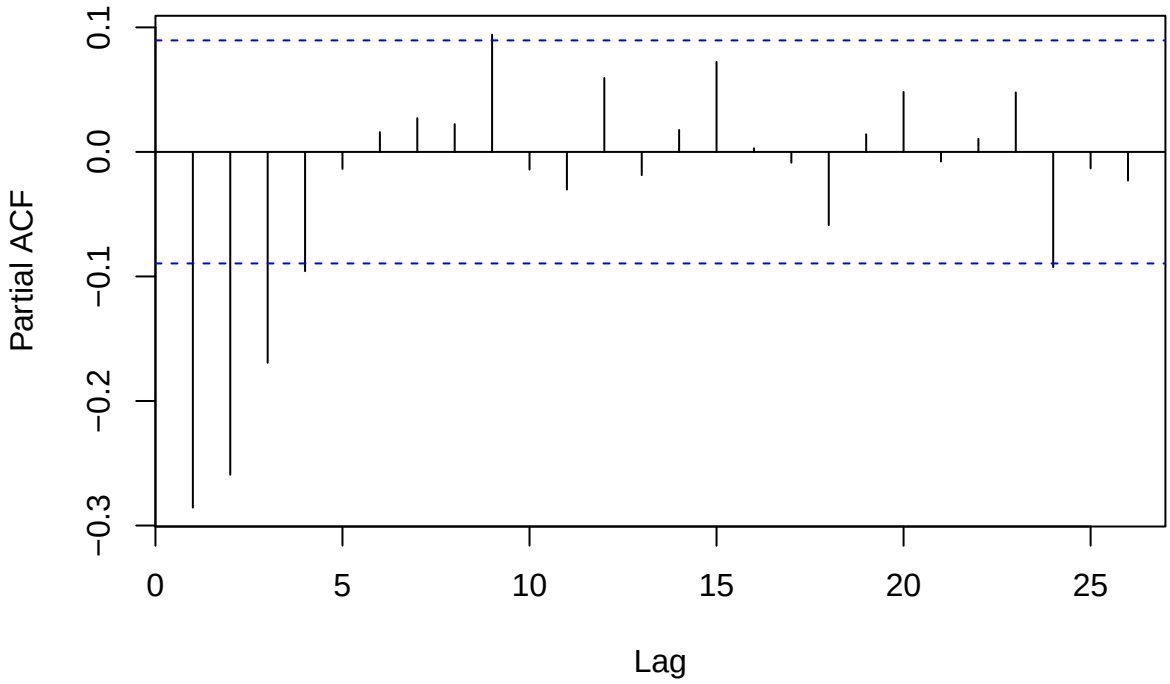
```
##
## Title:
##  Jarque - Bera Normalality Test
##
## Test Results:
##   STATISTIC:
##     X-squared: 839.3422
##   P VALUE:
##     Asymptotic p Value: < 2.2e-16
##
## Description:
##  Fri Sep 29 15:15:27 2017 by user:
```

since p value is less than 0.05, our objervation is passed by the test

```
#ACF autocorrelation plot
acf(myd$rate)
```




```
#PACF plot
pacf(myd$rate)
```




```
#

Box.test(myd$rate, lag = 2, type = 'Ljung')
```

```
##
##  Box-Ljung test
##
## data:  myd$rate
## X-squared = 51.155, df = 2, p-value = 7.795e-12
```

```
library(forecast)
#fit MA(2) model
m1 = Arima(ratets, order = c(0,0,2))
m1
```

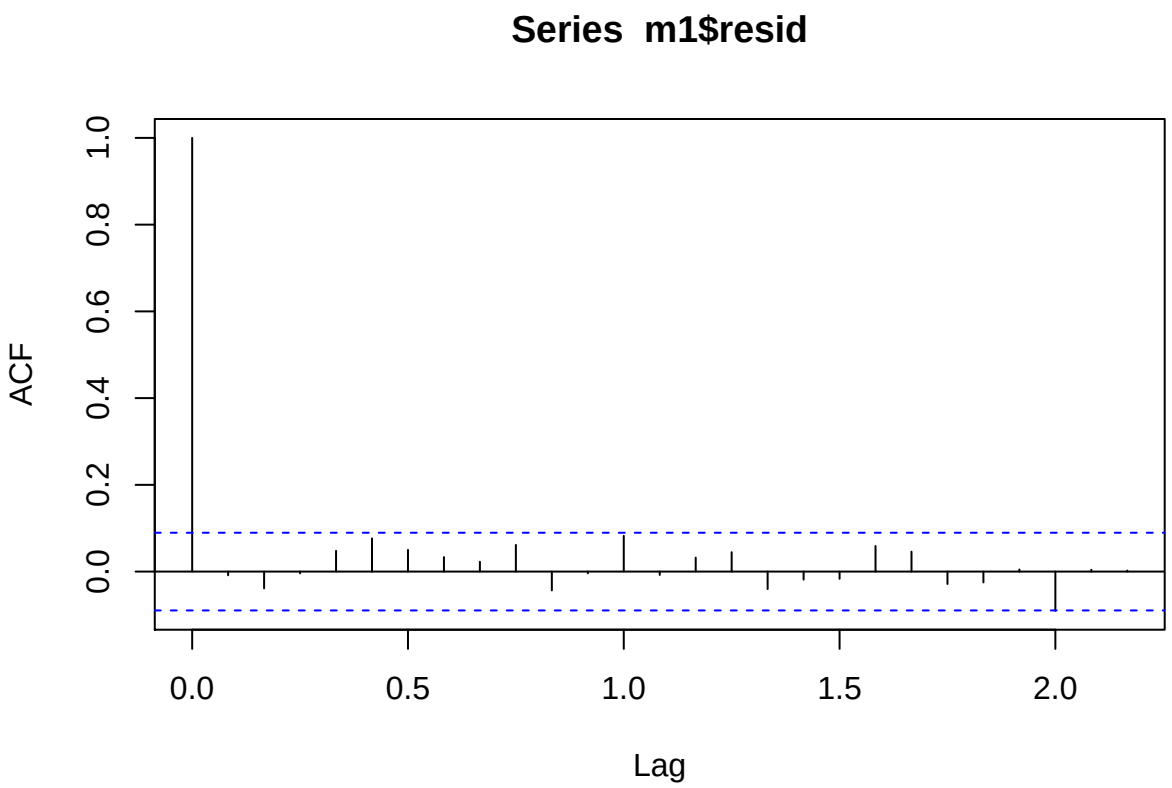
```
## Series: ratets
## ARIMA(0,0,2) with non-zero mean
##
## Coefficients:
##           ma1      ma2    mean
##       -0.4053  -0.1400  0.2986
## s.e.   0.0462   0.0432  0.1272
##
## sigma^2 estimated as 37.5:  log likelihood=-1546.37
## AIC=3100.74   AICc=3100.82   BIC=3117.43
```

```
#to view t-tests
coeftest(m1)
```

```
##
## z test of coefficients:
```

```
##
##             Estimate Std. Error z value Pr(>|z|)
## ma1        -0.405309   0.046202 -8.7726  < 2e-16 ***
## ma2        -0.140039   0.043173 -3.2436  0.00118 **
## intercept  0.298559   0.127221  2.3468  0.01894 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#analysis of residuals
#ACF of residuals
acf(m1$resid)
```




```
#LB test on residuals
Box.test(m1$resid, lag = 3, type = 'Ljung', fitdf = 2)
```

```
##
##  Box-Ljung test
##
## data:  m1$resid
## X-squared = 0.76713, df = 1, p-value = 0.3811
```

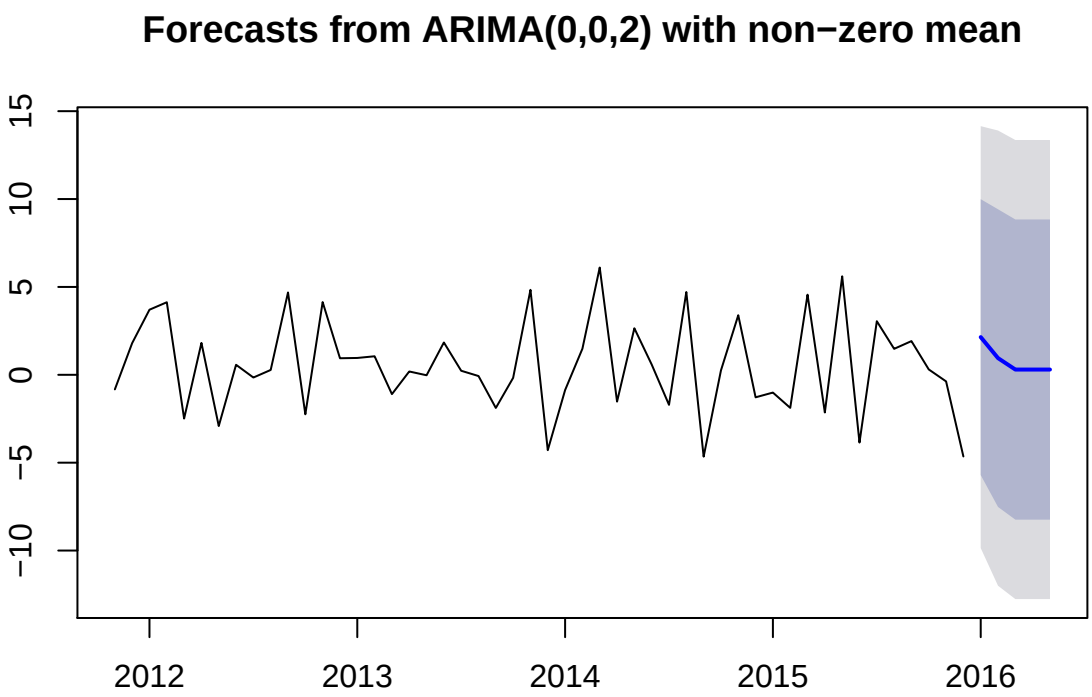
```
Box.test(m1$resid, lag = 6, type = 'Ljung', fitdf = 2)
```

```
##
##  Box-Ljung test
##
## data:  m1$resid
## X-squared = 5.9465, df = 4, p-value = 0.2032
```

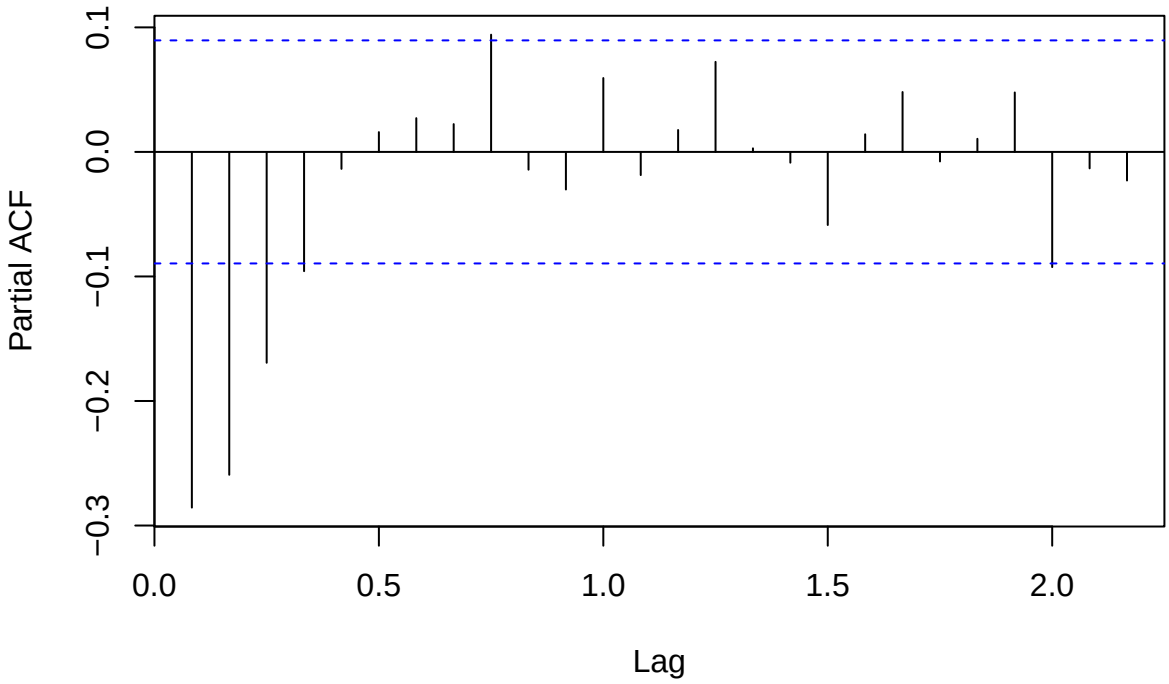
```
#compute forecasts up to 5 steps ahead
f = forecast(m1, h=5)
f
```

```
##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Jan 2016      2.1443574 -5.703974 9.992689  -9.85863 14.14734
## Feb 2016      0.9491186 -7.519355 9.417592 -12.00229 13.90053
## Mar 2016      0.2985590 -8.240938 8.838056 -12.76147 13.35859
## Apr 2016      0.2985590 -8.240938 8.838056 -12.76147 13.35859
## May 2016      0.2985590 -8.240938 8.838056 -12.76147 13.35859
```

```r
#plot forecasts
plot(f, include = 50)
lines(c(f$fitted, f$mean), col = "blue")
```




```r
#AR model
pacf(ratets)
```




```
#fit an AR(4) model
m2 = Arima(ratets, order = c(4,0,0))
coeftest(m2)
```

```
##
## z test of coefficients:
##
##             Estimate Std. Error z value  Pr(>|z|)
## ar1       -0.420201   0.045511 -9.2329 < 2.2e-16 ***
## ar2       -0.350852   0.048428 -7.2448 4.330e-13 ***
## ar3       -0.207789   0.048364 -4.2963 1.737e-05 ***
## ar4       -0.095387   0.045352 -2.1033   0.03544 *
## intercept  0.298758   0.134033  2.2290   0.02582 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
f2 = forecast(m2, h = 5)
f2
```

```
##          Point Forecast     Lo 80     Hi 80      Lo 95    Hi 95
## Jan 2016     2.45767771 -5.364359 10.279714  -9.505095 14.42045
## Feb 2016     1.26618768 -7.218357  9.750732 -11.709803 14.24218
## Mar 2016     0.22666921 -8.366697  8.820036 -12.915751 13.36909
## Apr 2016     0.01275135 -8.581205  8.606708 -13.130571 13.15607
## May 2016     0.03727684 -8.564763  8.639317 -13.118408 13.19296
```