IranKhodro timeseries Analysis

Mehrab Atighi

12/20/2021

In The Name Of God

Contents

problem statement

We want to use the Timeseries Algorithms On the shares of Irankhodro Company in the Iranian stock market.
we should chek it with all of ARIMA state until the second time.

Dawnloading Data

So We went Tse website and dawnload needed Excel file.for this step we should go to the Site and after dawnloding Excel file, we should save that as CSV format.

Importing Data To R

Now we want to import the Dataset in to R. so we should use read.csv("...") function.

```
df = read.csv(file = "F:/lessons/time series/project/Iran.Khodro
head(df , 4)
```

```
##      X.TICKER. X.DTYYYYMMDD. X.FIRST. X.HIGH. X.LOW. X.CLOSE.
## 1 Iran.Khodro      20211219     1965    1970   1856      1876
## 2 Iran.Khodro      20211218     1900    1961   1890      1953
## 3 Iran.Khodro      20211215     1890    1950   1825      1868
## 4 Iran.Khodro      20211214     1922    1969   1917      1921
##       X.VOL. X.OPENINT. X.PER. X.OPEN. X.LAST.
## 1 438255155      11188      D    1953    1856
## 2 701269890      16307      D    1868    1961
## 3 733471286      17475      D    1921    1856
## 4 809024025      14825      D    2017    1917
```

PreProccesing of data

we should separate our data and we just need to Date and close value for each day. so we use separate function from tidyverse Package.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
df= separate(df,2,c("year","mouth","day"),
             sep =c(4,6),remove = TRUE)
df= df[,c(2,3,4,8)]
head(df , 4 )
```

```
##   year mouth day X.CLOSE.
## 1 2021    12  19     1876
## 2 2021    12  18     1953
## 3 2021    12  15     1868
## 4 2021    12  14     1921
```

Regression modeling for Data

Now we want to make a regression model for our data and calculating the coefficients estimate.

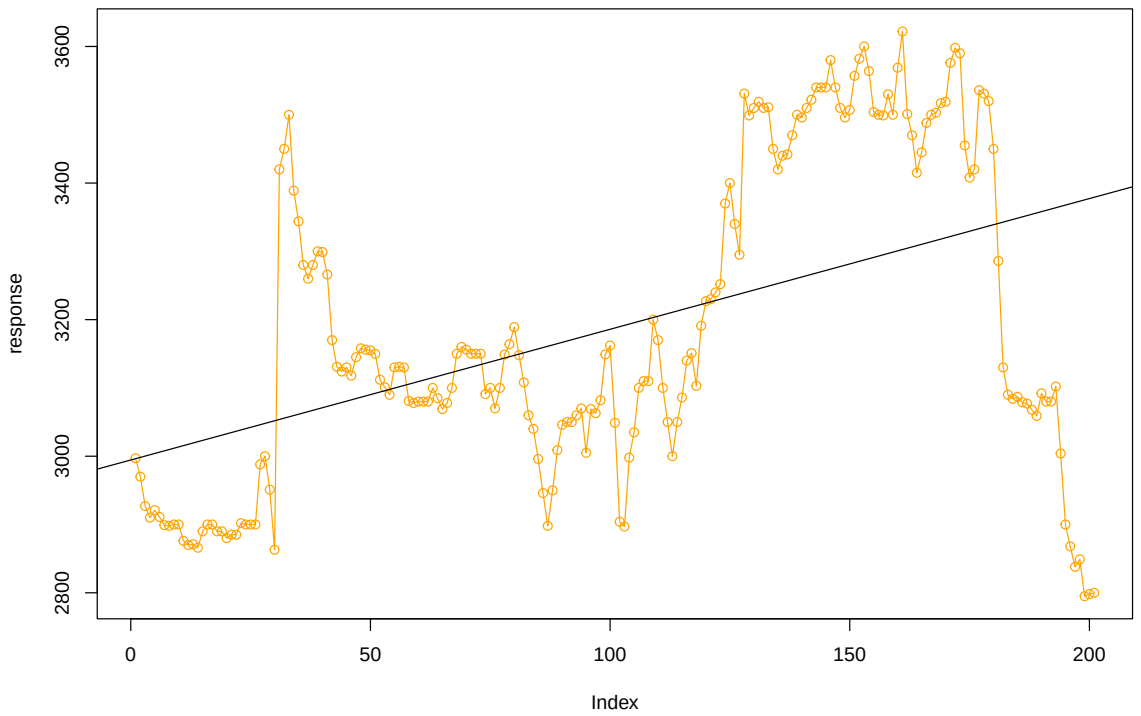
```
response = df$X.CLOSE.[(nrow(df)-200):nrow(df)]
model_reg = lm(response ~ (time(response)))
summary(model_reg)
```

```
##
## Call:
## lm(formula = response ~ (time(response)))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -580.36 -127.01  -15.06  186.97  442.36
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2994.4752    28.6114 104.660  < 2e-16 ***
## time(response)    1.9140     0.2456   7.792 3.58e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 202.1 on 199 degrees of freedom
## Multiple R-squared:  0.2338, Adjusted R-squared:  0.2299
## F-statistic: 60.72 on 1 and 199 DF,  p-value: 3.578e-13
```

Selecting and Ploting Data

Now we want to select the Two hundred days ago data to timeseries alg and ploting them fron dataset.

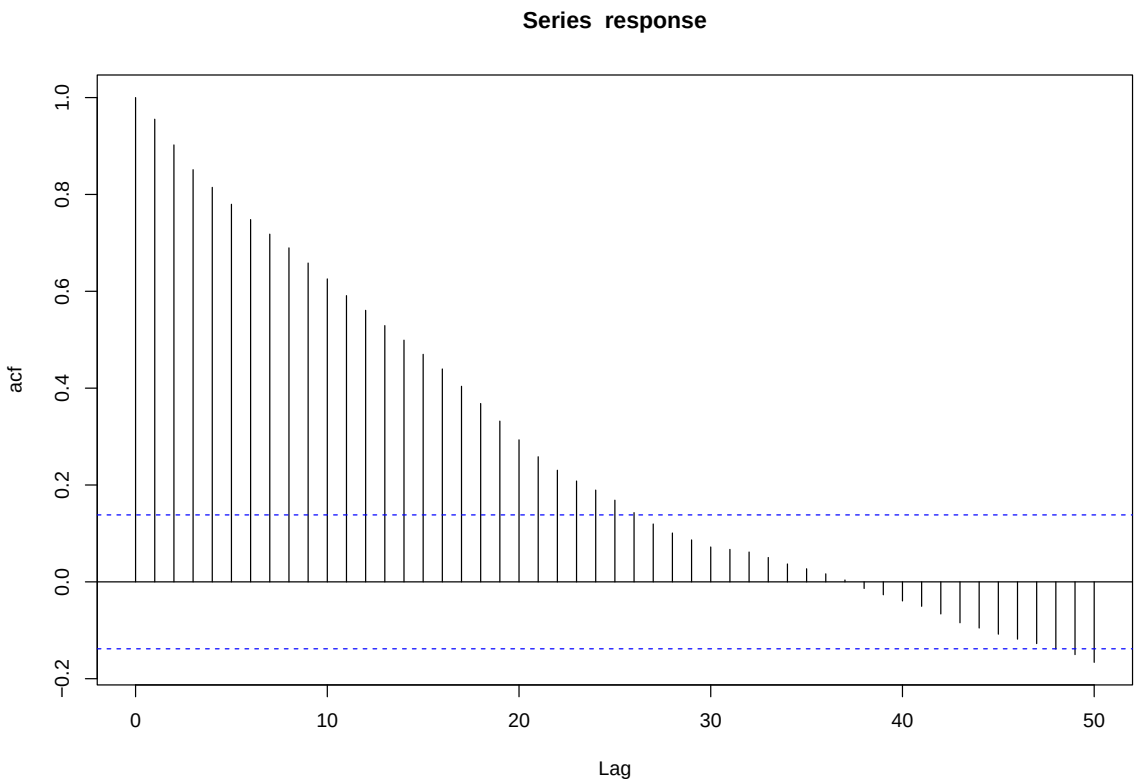
```
plot(response,type="o" , col = "orange")
abline(model_reg)
```



plot of auto correlation function

Now we want to ploting the auto correlation function values of dataset.

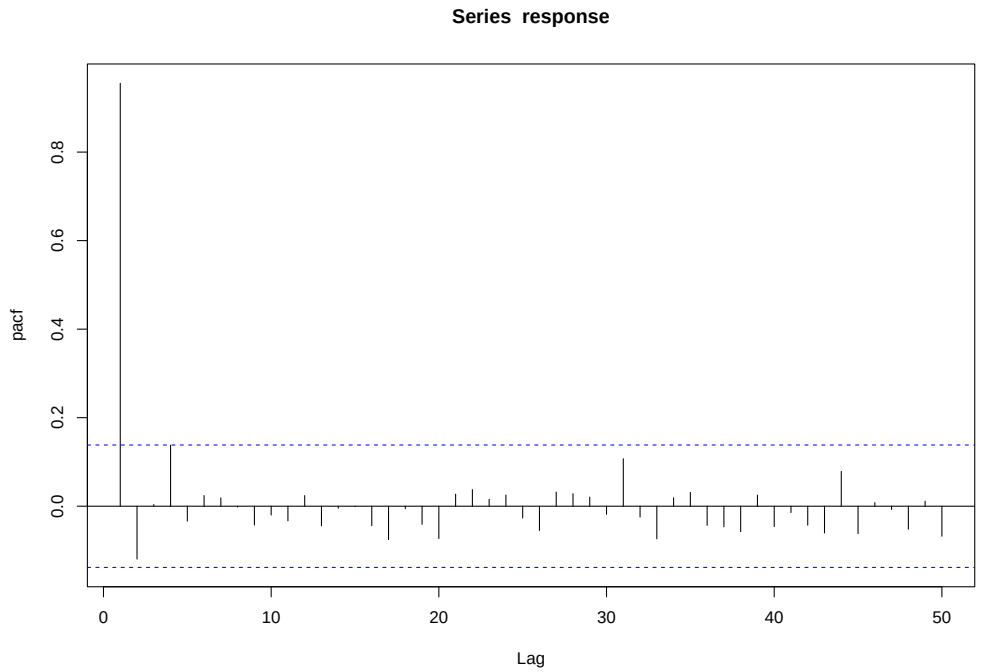
```
acf(response,lag=50,ylab="acf")
```



plot of partial auto correlation function

We want to ploting the partial auto correlation function values of dataset.

```
pacf(response,lag=50,ylab="pacf")
```



We can see that the pacf values from the first times is lower than a blue bound and we can say its approximatly zero.

Cheking The ARIMA Models with Data

Now we want to chek all of ARIMA state until two times.
- ARIMA(0,0,1)13 - ARIMA(0,0,2)14 - ARIMA(1,0,0)15 - ARIMA(2,0,0)16 - ARIMA(1,0,1)17 - ARIMA(2,0,2)18 - ARIMA(0,1,1)19 - ARIMA(0,1,2)20 - ARIMA(1,1,0)21 - ARIMA(2,1,0)22 - ARIMA(1,1,1)23 - ARIMA(2,1,2)24 - ARIMA(0,2,1)25 - ARIMA(0,2,2)26 - ARIMA(1,2,0)27 - ARIMA(2,2,0)28 - ARIMA(1,2,1)29 - ARIMA(2,2,2)30

Modeling for all states

We Want to Chek the response with ARIMA(0,0,1): attention that we need to install and library the TSA Package.

```
library(TSA)
```

```
##
## Attaching package: 'TSA'

## The following object is masked from 'package:readr':
##
##     spec

## The following objects are masked from 'package:stats':
##
##     acf, arima

## The following object is masked from 'package:utils':
##
##     tar
```

```
(M1=arima(response,order=c(0,0,1)))
```

```
##
## Call:
## arima(x = response, order = c(0, 0, 1))
##
## Coefficients:
##          ma1  intercept
##       0.8176  3186.4328
## s.e.  0.0279    17.6125
##
## sigma^2 estimated as 18957:  log likelihood = -1275.68,  aic = 2555.36
```

Modeling for all states

We Want to Chek the response with ARIMA(0,0,1):

```
(M2=arima(response,order=c(0,0,2)))
```

```
##
## Call:
## arima(x = response, order = c(0, 0, 2))
##
## Coefficients:
##          ma1     ma2  intercept
##       1.1233  0.7259  3185.4451
## s.e.  0.0705  0.0561    19.5437
##
## sigma^2 estimated as 9543:  log likelihood = -1207.16,  aic = 2420.33
```

Modeling for all states

We Want to Chek the response with ARIMA(1,0,0):

```
(M3=arima(response,order=c(1,0,0)))
```

```
##
## Call:
## arima(x = response, order = c(1, 0, 0))
##
## Coefficients:
##          ar1  intercept
##       0.9668  3122.7292
## s.e.  0.0174   117.7902
##
## sigma^2 estimated as 3740:  log likelihood = -1113.38,  aic = 2230.76
```

Modeling for all states

We Want to Chek the response with ARIMA(2,0,0):

```
(M4=arima(response,order=c(2,0,0)))
```

```
##
## Call:
## arima(x = response, order = c(2, 0, 0))
##
## Coefficients:
##          ar1      ar2  intercept
##       1.0912  -0.1293  3138.0426
## s.e.  0.0697   0.0702   103.6656
##
## sigma^2 estimated as 3678:  log likelihood = -1111.7,  aic = 2229.4
```

Modeling for all states

We Want to Chek the response with ARIMA(1,0,1):

```
(M5=arima(response,order=c(1,0,1)))
```

```
##
## Call:
## arima(x = response, order = c(1, 0, 1))
##
## Coefficients:
##          ar1     ma1  intercept
##       0.9579  0.1309  3135.8185
## s.e.  0.0205  0.0690   105.7904
##
## sigma^2 estimated as 3678:  log likelihood = -1111.7,  aic = 2229.4
```

Modeling for all states

We Want to Chek the response with ARIMA(2,0,2):

```
(M6=arima(response,order=c(2,0,2)))
```

```
##
## Call:
## arima(x = response, order = c(2, 0, 2))
##
## Coefficients:
##          ar1     ar2     ma1     ma2  intercept
##       0.1107  0.8053  0.9984  0.1828  3139.8876
## s.e.  0.3945  0.3855  0.3855  0.0750   102.1744
##
## sigma^2 estimated as 3633:  log likelihood = -1110.48,  aic = 2230.96
```

Modeling for all states

We Want to Chek the response with ARIMA(0,1,1):

```
(M7=arima(response,order=c(0,1,1)))
```

```
##
## Call:
## arima(x = response, order = c(0, 1, 1))
##
## Coefficients:
##          ma1
##       0.1130
## s.e.  0.0687
##
## sigma^2 estimated as 3755:  log likelihood = -1106.87,  aic = 2215.73
```

Modeling for all states

We Want to Chek the response with ARIMA(0,1,2):

```
(M8=arima(response,order=c(0,1,2)))
```

```
##
## Call:
## arima(x = response, order = c(0, 1, 2))
##
## Coefficients:
##          ma1     ma2
##       0.1134  0.0008
## s.e.  0.0772  0.0781
##
## sigma^2 estimated as 3755:  log likelihood = -1106.87,  aic = 2217.73
```

Modeling for all states

We Want to Chek the response with ARIMA(1,1,0):

```
(M9=arima(response,order=c(1,1,0)))
```

```
##
## Call:
## arima(x = response, order = c(1, 1, 0))
##
## Coefficients:
##          ar1
##       0.1095
## s.e.  0.0701
##
## sigma^2 estimated as 3757:  log likelihood = -1106.93,  aic = 2215.86
```

Modeling for all states

We Want to Chek the response with ARIMA(2,1,0):

```
(M10=arima(response,order=c(2,1,0)))
```

```
##
## Call:
## arima(x = response, order = c(2, 1, 0))
##
## Coefficients:
##          ar1      ar2
##       0.1151  -0.0520
## s.e.  0.0705   0.0704
##
## sigma^2 estimated as 3747:  log likelihood = -1106.66,  aic = 2217.32
```

Modeling for all states

We Want to Chek the response with ARIMA(1,1,1):

```
(M11=arima(response,order=c(1,1,1)))
```

```
##
## Call:
## arima(x = response, order = c(1, 1, 1))
##
## Coefficients:
##          ar1     ma1
##       0.0017  0.1115
## s.e.  0.3128  0.3048
##
## sigma^2 estimated as 3755:  log likelihood = -1106.87,  aic = 2217.73
```

Modeling for all states

We Want to Chek the response with ARIMA(2,1,2):

```
(M12=arima(response,order=c(2,1,2)))
```

```
##
## Call:
## arima(x = response, order = c(2, 1, 2))
##
## Coefficients:
##           ar1     ar2     ma1     ma2
##       -0.2487  -0.580  0.3940  0.6228
## s.e.   0.5984   0.367  0.6184  0.2602
##
## sigma^2 estimated as 3682:  log likelihood = -1104.94,  aic = 2217.88
```

Modeling for all states

We Want to Chek the response with ARIMA(0,2,1):

```
(M13=arima(response,order=c(0,2,1)))
```

```
##
## Call:
## arima(x = response, order = c(0, 2, 1))
##
## Coefficients:
##           ma1
##       -0.9892
## s.e.   0.0271
##
## sigma^2 estimated as 3848:  log likelihood = -1105.69,  aic = 2213.39
```

Modeling for all states

We Want to Chek the response with ARIMA(0,2,2):

```
(M14=arima(response,order=c(0,2,2)))
```

```
##
## Call:
## arima(x = response, order = c(0, 2, 2))
##
## Coefficients:
##           ma1     ma2
##       -0.8829  -0.117
## s.e.   0.0984   0.069
##
## sigma^2 estimated as 3773:  log likelihood = -1104.35,  aic = 2212.7
```

Modeling for all states

We Want to Chek the response with ARIMA(1,2,0):

```
(M15=arima(response,order=c(1,2,0)))
```

```
##
## Call:
## arima(x = response, order = c(1, 2, 0))
##
## Coefficients:
##           ar1
##       -0.4145
## s.e.   0.0643
##
## sigma^2 estimated as 5621:  log likelihood = -1141.57,  aic = 2285.14
```

Modeling for all states

We Want to Chek the response with ARIMA(2,2,0):

```
(M16=arima(response,order=c(2,2,0)))
```

```
##
## Call:
## arima(x = response, order = c(2, 2, 0))
##
## Coefficients:
##           ar1      ar2
##       -0.5036  -0.2131
## s.e.   0.0691   0.0690
##
## sigma^2 estimated as 5361:  log likelihood = -1136.91,  aic = 2277.82
```

Modeling for all states

We Want to Chek the response with ARIMA(1,2,1):

```
(M17=arima(response,order=c(1,2,1)))
```

```
##
## Call:
## arima(x = response, order = c(1, 2, 1))
##
## Coefficients:
##          ar1      ma1
##       0.1148  -0.9999
## s.e.  0.0705   0.0526
##
## sigma^2 estimated as 3775:  log likelihood = -1104.41,  aic = 2212.81
```

Modeling for all states

We Want to Chek the response with ARIMA(2,2,2):

```
(M18=arima(response,order=c(2,2,2)))
```

```
##
## Call:
## arima(x = response, order = c(2, 2, 2))
##
## Coefficients:
##          ar1      ar2      ma1     ma2
##       0.8334  -0.1872  -1.7110  0.7228
## s.e.  0.2024   0.0727   0.1964  0.1941
##
## sigma^2 estimated as 3740:  log likelihood = -1102.49,  aic = 2212.98
```

Final AIC Comparing

Now we want to compare all models AIC .

```
(AIC=c(M1$aic,M2$aic,M3$aic,M4$aic,M5$aic,M6$aic,
  M7$aic,M8$aic,M9$aic,M10$aic,M11$aic,M12$aic,
  M13$aic,M14$aic,M15$aic,M16$aic,M17$aic,M18$aic))
```

```
##  [1] 2555.358 2420.329 2230.761 2229.399 2229.397 2230.9
##  [9] 2215.865 2217.320 2217.734 2217.876 2213.387 2212.7
## [17] 2212.813 2212.978
```

```
min(AIC)
```

```
## [1] 2212.702
```

```
which.min(AIC)
```

```
## [1] 14
```

So we can say that the M14 or ARIMA(0,2,2) is the best fit for our dataset.

Final predict Visual

```
plot(response,type="o" , col = "orange")
with(df , lines(x = time(response) , y = fitted(M14)))
```