Time Series Analysis - Dow Jones Industrial

Siyuan Zheng, 1000726814, STA457H

Data Description

- The data is monthly excess return of Dow Jones Industrial Average from Jan 2005 to Feb 2017
- Where monthly excess return is defined as $\frac{Close-Open}{Open}$
- Historical data is downloaded from http://finance.yahoo.com

```
#The following commands are run in the console
#install.packages("dygraphs")
#install.packages("ltsa")
#install.packages("forecast")
#install.packages("tseries")
#install.packages("entropy")
#install.packages("arfima")
#install.packages("ggplot2")
#install.packages("ggfortify")
#install.packages("gamlss")

#import the libraries
library(dygraphs)
library(forecast)
library(ltsa)
library(tseries)
library(entropy)
library(arfima)
```

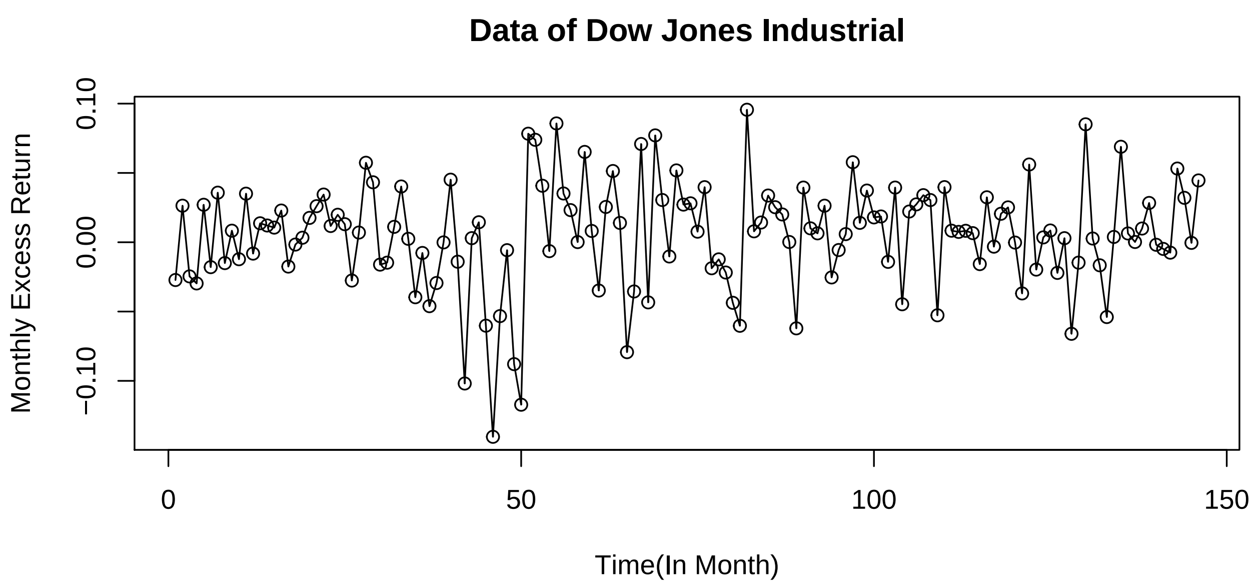
```
##
## Attaching package: 'arfima'

## The following object is masked from 'package:forecast':
##
##     arfima

## The following object is masked from 'package:stats':
##
##     BIC
```

```
library(ltsa)
# Data entry in the csv files have been adjusted to date-ascending order
data <-read.csv("table.csv", header=TRUE)
# What does the data look like ?
head(data)
```

```
##       Date     Open     High      Low    Close    Volume Adj.Close
## 1 05-01-03 10783.75 10867.39 10368.61 10489.94 286149500  10489.94
## 2 05-02-01 10489.72 10853.43 10489.64 10766.23 280355700  10766.23
```

```
## 3 05-03-01 10769.04 10984.46 10396.24 10503.76 276336800  10503.76
## 4 05-04-01 10504.57 10568.93 10000.46 10192.51 293144200  10192.51
## 5 05-05-02 10192.00 10560.81 10075.55 10467.48 242084700  10467.48
## 6 05-06-01 10462.86 10656.29 10253.49 10274.97 242206300  10274.97
```

```
# As required, defined the data of monthly excess return as (Close-Open)/Open
DJI = ts((data$Close-data$Open)/data$Open)
# The plot of the historical data
plot(DJI,main="Data of Dow Jones Industrial", xlab="Time(In Month)"
     ,ylab="Monthly Excess Return",type="o")
```




```
# making the plot interative(only available for html)
# dygraph(DJI)
```

Time Series Modelling

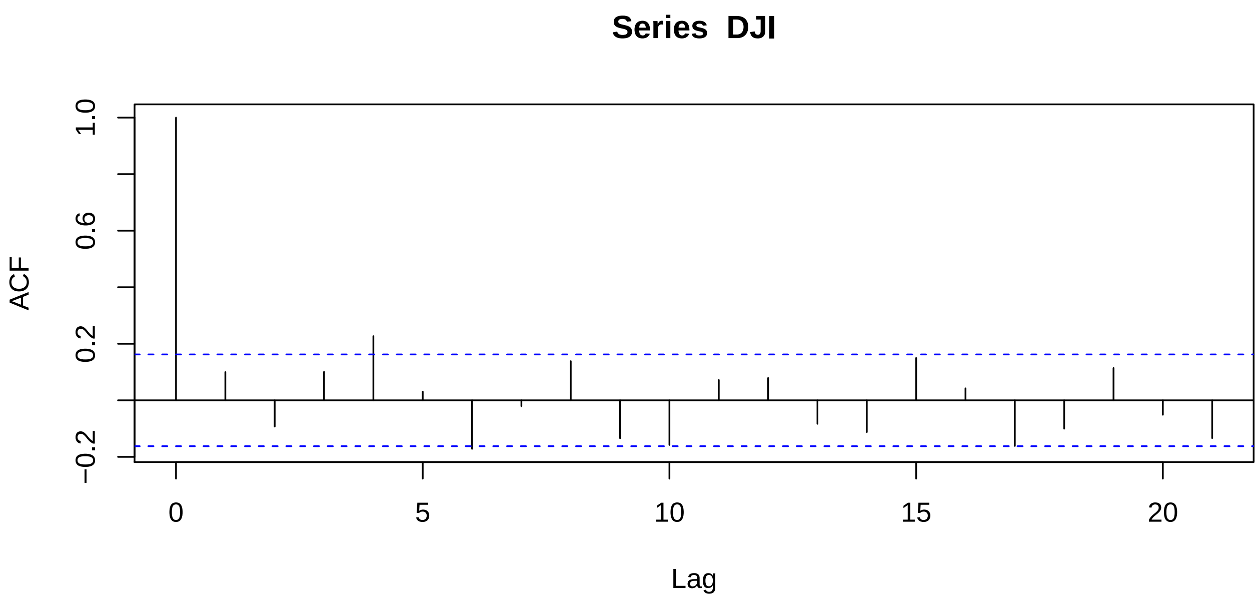
```
# To determine whether the data needs transformation
lambda <- BoxCox.lambda(DJI)
lambda
```

```
## [1] 0.894564
```

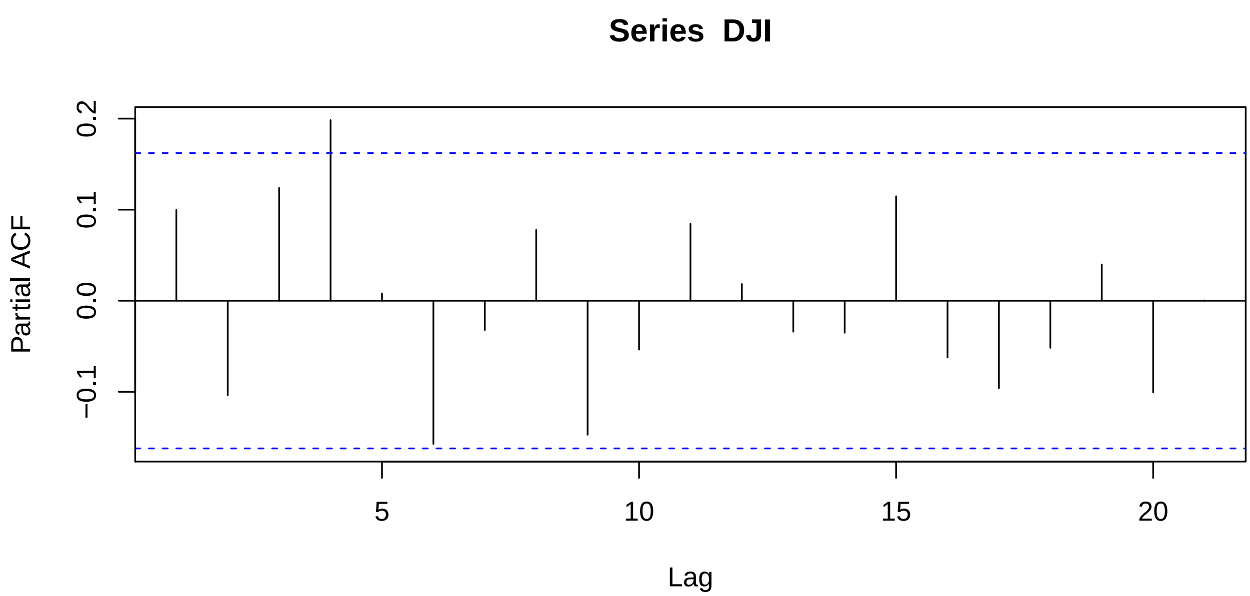
Since lambda is very cloes to 1, we do not need to make transformation to the data

```
# plot(stl(DJI)), this causes the halt of the program, which says there
# is no periodic component, thus ignoring the procedure of
# decomposition(removing trend and seasonal)
```

```r
# Using ACF to determine the p of MA model
acf(DJI)
```




```r
# Using PACF to determine the q of AR model
pacf(DJI)
```

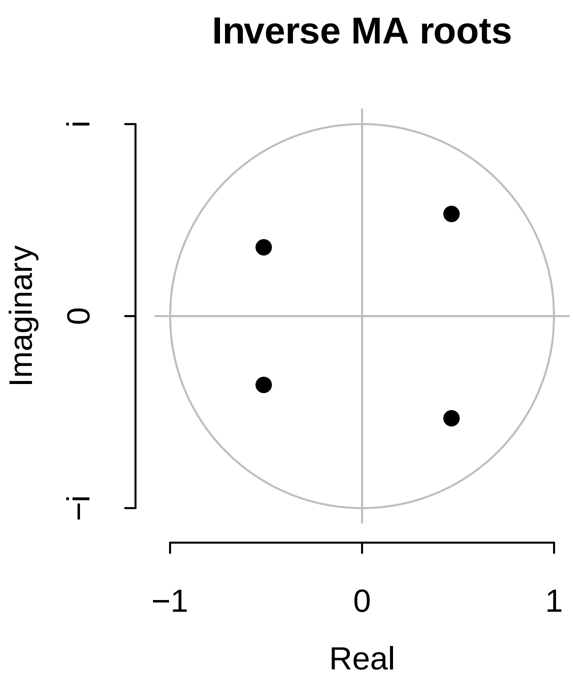


Fitting data using MA(4) model

```r
# based on the above diagrams, we should choose 4 as p of MA model
ma <- arima(DJI,c(0,0,4),include.mean=FALSE)
ma
```

```
##
## Call:
## arima(x = DJI, order = c(0, 0, 4), include.mean = FALSE)
##
## Coefficients:
##          ma1      ma2     ma3     ma4
##       0.0927  -0.0636  0.1482  0.1954
## s.e.  0.0812   0.0823  0.0829  0.0734
##
## sigma^2 estimated as 0.001393:  log likelihood = 272.8,  aic = -535.59
```

```
plot(ma)
```




```
## => There exists at least one root such that |B| <= 1
names(ma)
```

```
##  [1] "coef"      "sigma2"    "var.coef"  "mask"      "loglik"
##  [6] "aic"       "arma"      "residuals" "call"      "series"
## [11] "code"      "n.cond"    "nobs"      "model"
```

```
## "The AIC of MA(4) model is :"
ma$aic
```

```
## [1] -535.5925
```

```
## The p-value of the Ljung-Box test of MA model is:
Box.test(ma$resid,type=c("Ljung-Box"))$p.value
```

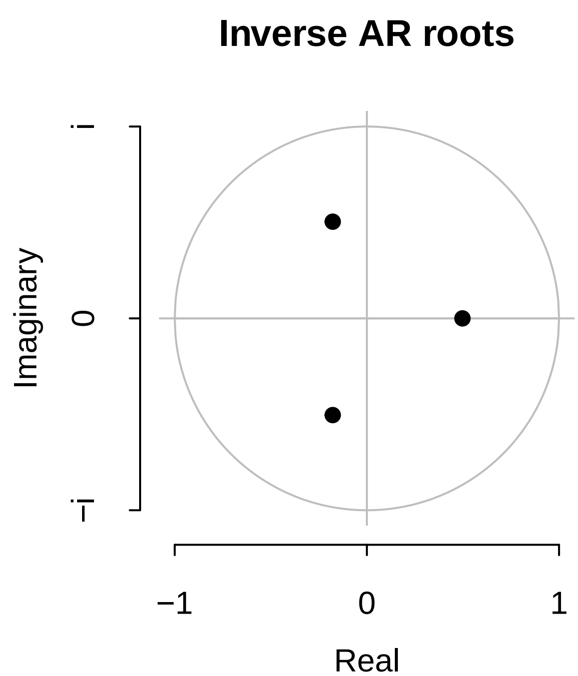
```
## [1] 0.991493
```

Fitting data using AR(3) model

```r
# based on the above diagrams, we should choose 3 as q of AR model
ar <- arima(DJI,c(3,0,0),include.mean=FALSE)
ar
```

```
##
## Call:
## arima(x = DJI, order = c(3, 0, 0), include.mean = FALSE)
##
## Coefficients:
##          ar1      ar2     ar3
##       0.1418  -0.1088  0.1420
## s.e.  0.0823   0.0825  0.0821
##
## sigma^2 estimated as 0.001444:  log likelihood = 270.24,  aic = -532.48
```

```r
plot(ar)
```




```r
## => There exists at least one root such that |B| <= 1
## The AIC of AR(3) model is :
ar$aic
```

```
## [1] -532.4813
```

```r
## "The p-value of the Ljung-Box test of AR(3) model is:
Box.test(ar$resid,type = c("Ljung-Box"))$p.value
```

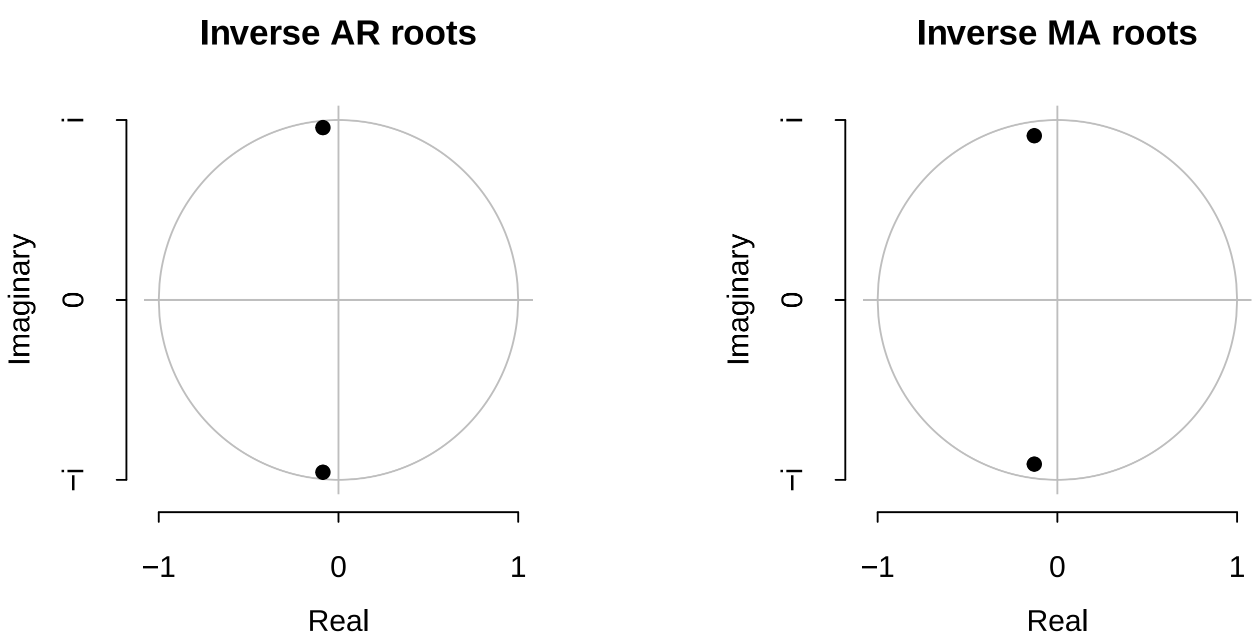
```
## [1] 0.5919148
```

Fitting data using ARMA model(auto)

```
auto <- auto.arima(DJI,seasonal=FALSE,stationary=TRUE,allowmean=FALSE)
auto
```

```
## Series: DJI
## ARIMA(2,0,2) with zero mean
##
## Coefficients:
##           ar1      ar2     ma1     ma2
##       -0.1731  -0.9251  0.2567  0.8499
## s.e.   0.0640   0.0535  0.0983  0.0739
##
## sigma^2 estimated as 0.001394:  log likelihood=274.59
## AIC=-539.18   AICc=-538.75   BIC=-524.27
```

```
plot(auto)
```




```
## => There exists at least one root such that |B| <= 1
## The AIC of ARMA(auto) model is :
auto$aic
```

```
## [1] -539.1836
```

```
## "The p-value of the Ljung-Box test of ARMA(auto) model is:
Box.test(auto$resid,type = c("Ljung-Box"))$p.value
```

```
## [1] 0.5705332
```

Comments on three models

AR Model

$$X_t = 0.1418X_{t-1} - 0.1088X_{t-2} + 0.1420X_{t-3} + Z_t$$

-p-value of ljung-box test is greater than 0.05 => fail to reject the null hypothesis that Z_t is a white noise.

-not causal(stationary)

MA Model

$$X_t = Z_t + 0.0927Z_{t-1} - 0.0636Z_{t-2} + 0.1482Z_{t-3} + 0.1954Z_{t-4}$$

-p-value of ljung-box test is greater than 0.05 => fail to reject the null hypothesis that Z_t is a white noise.

-not invertible

ARMA(2,2) (auto generated) Model

$$X_t - 0.1731X_{t-1} - 0.9251X_{t-2} = Z_t + 0.2567Z_{t-1} + 0.8499Z_{t-2}$$

-p-value of ljung-box test is greater than 0.05 => fail to reject the null hypothesis that Z_t is a white noise.

-neither invertible nor causal(stationary)

Conclusion

There is no transformation to the monthly excess return data, since lambda value given by boxcox is very close to 1.There is no seasonal or trend component for the time series. After using MA(4), AR(3), ARMA(2,2) models to fit data repectively,ARMA(2,2) model has the lowest AIC value, thus to this extent ARMA(2,2) is used to conduct the forecasting. However, ARCH model is not being tried for fitting the data, maybe there are some other better models as well.

Forecasting

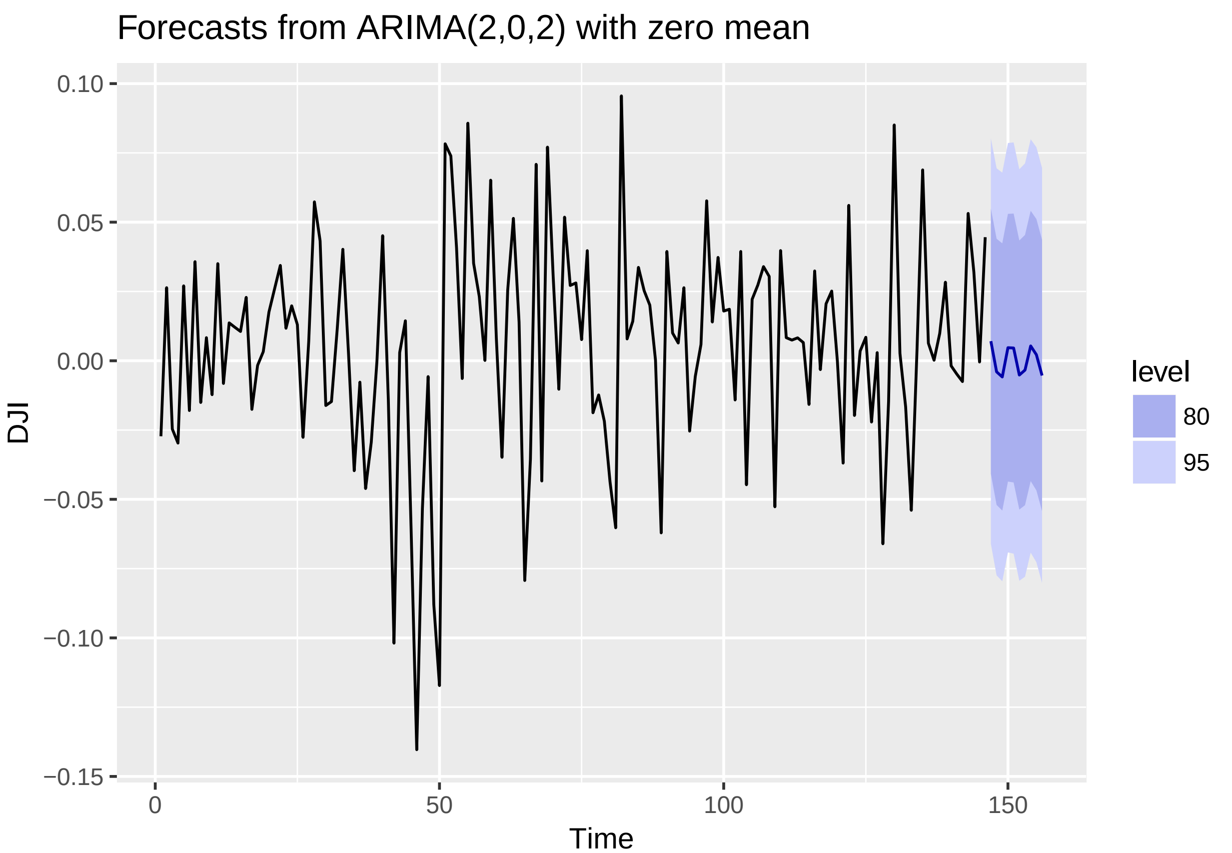
```r
library(ggplot2)
library(ggfortify)
```

```
##
## Attaching package: 'ggfortify'

## The following object is masked from 'package:forecast':
##
##     gglagplot
```

```r
forecast <- forecast(auto)
forecast
```

```
##     Point Forecast       Lo 80      Hi 80       Lo 95      Hi 95
## 147    0.007067221 -0.04078076 0.05491521 -0.06610996 0.08024440
## 148   -0.003995046 -0.05200999 0.04401990 -0.07742756 0.06943747
## 149   -0.005846250 -0.05405263 0.04236013 -0.07957155 0.06787905
## 150    0.004708027 -0.04358902 0.05300507 -0.06915593 0.07857199
## 151    0.004593280 -0.04391134 0.05309790 -0.06958814 0.07877470
## 152   -0.005150679 -0.05369490 0.04339354 -0.07939266 0.06909130
## 153   -0.003357512 -0.05210859 0.04539357 -0.07791586 0.07120083
## 154    0.005346225 -0.04341588 0.05410833 -0.06922898 0.07992143
## 155    0.002180440 -0.04677352 0.05113440 -0.07268818 0.07704906
## 156   -0.005323335 -0.05427774 0.04363107 -0.08019264 0.06954597
```

```r
autoplot(forecast)
```



The predicted monthly excess return of April 2017 is : -0.004