Stock Market Prediction using Random Forest Algorithm

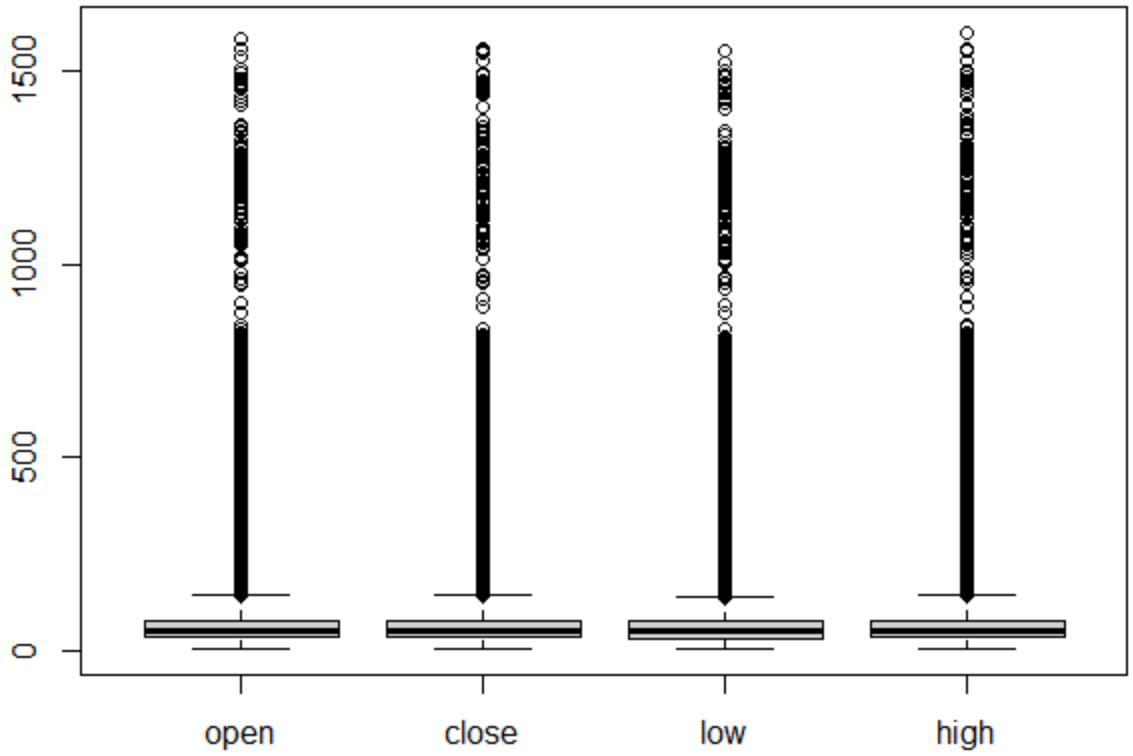
```
setwd("F:/DA Project")
data = read.csv ("Data.csv",nrows=100000)
data[,1] = as.Date(data[,1])
summary(data)
```

```
> setwd("F:/DA Project")
> data = read.csv ("Data.csv",nrows=100000)
> data[,1] = as.Date(data[,1])
> summary(data)
      date               symbol              open              close
 Min.   :0001-02-20   Length:100000      Min.   :   1.66   Min.   :   1.78
 1st Qu.:0003-12-20   Class :character   1st Qu.:  31.26   1st Qu.:  31.29
 Median :0006-12-20   Mode  :character   Median :  48.28   Median :  48.29
 Mean   :0007-01-19                      Mean   :  64.85   Mean   :  64.88
 3rd Qu.:0010-02-20                      3rd Qu.:  74.90   3rd Qu.:  74.90
 Max.   :0012-12-20                      Max.   :1584.44   Max.   :1557.98
      low                high              volume             class
 Min.   :   1.50   Min.   :   1.81   Min.   :        0   Length:100000
 1st Qu.:  30.94   1st Qu.:  31.63   1st Qu.:  1255800   Class :character
 Median :  47.76   Median :  48.78   Median :  2514750   Mode  :character
 Mean   :  64.18   Mean   :  65.51   Mean   :  5412135
 3rd Qu.:  74.12   3rd Qu.:  75.60   3rd Qu.:  5240150
 Max.   :1549.94   Max.   :1600.93   Max.   :553080300
```

```
dim(data)
head(data)
summary(is.na(data))
```

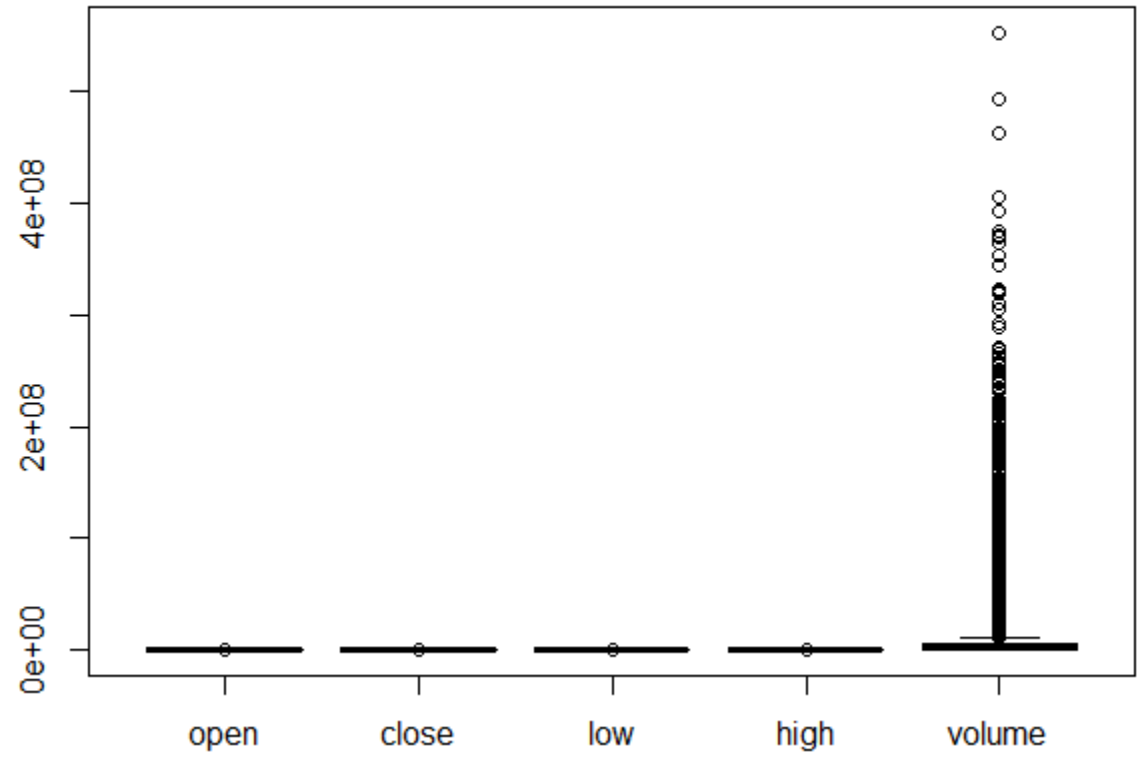
```
> dim(data)
[1] 100000      8
> head(data)
        date symbol    open  close    low   high  volume   class
1 0007-01-20   TSCO  59.375  58.19  57.90  59.39 1391400     low
2 0005-09-20   ADSK  39.950  39.75  39.56  40.22 2664400 Neutral
3 0004-01-20    PSA 111.560 111.00 110.61 111.67  700100     low
4 0008-01-20    MAA  67.600  66.25  65.77  67.97  600700     low
5 0005-05-20     WY  31.520  31.12  31.07  31.61 3103800     low
6 0003-02-20    ROP  83.100  84.71  83.02  84.98  586100    high
> summary(is.na(data))
    date            symbol           open            close
 Mode :logical   Mode :logical   Mode :logical   Mode :logical
 FALSE:100000    FALSE:100000    FALSE:100000    FALSE:100000
    low             high           volume          class
 Mode :logical   Mode :logical   Mode :logical   Mode :logical
 FALSE:100000    FALSE:100000    FALSE:100000    FALSE:100000
> boxplot(data[,c('open','close','low','high')])
```

```
boxplot(data[,c('open','close','low','high')])
```



boxplot(data[,c('open','close','low','high','volume')])

```
> boxplot(data[,c('open','close','low','high','volume')])
```

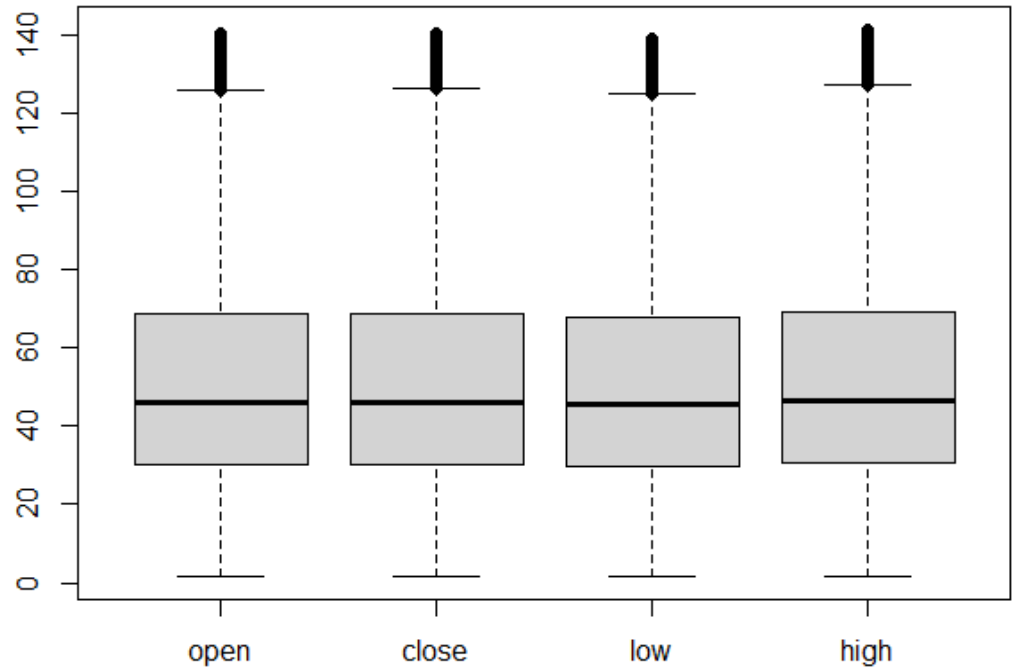


for(x in c('open','close','low','high','volume')){

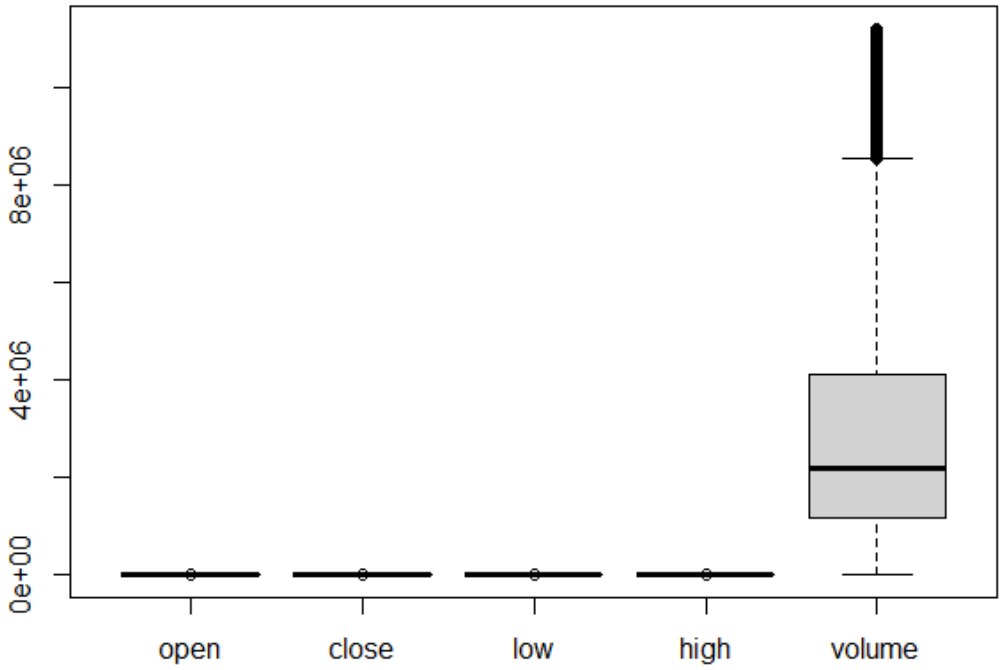
val = data[,x][data[,x] %in% boxplot.stats(data[,x])$out]
data[,x][data[,x] %in% val] = NA
}

```
> for(x in c('open','close','low','high','volume')){
+     val = data[,x][data[,x] %in% boxplot.stats(data[,x])$out]
+     data[,x][data[,x] %in% val] = NA
+ }
> boxplot(data[,c('open','close','low','high')])
> boxplot(data[,c('open','close','low','high','volume')])
```

boxplot(data[,c('open','close','low','high')])



boxplot(data[,c('open','close','low','high','volume')])



```
library(tidyr)
data=drop_na(data)
dim(data)
as.data.frame(colSums(is.na(data)))
```

```
> library(tidyr)
Warning message:
package 'tidyr' was built under R version 4.0.5
> data=drop_na(data)
> dim(data)
[1] 84036     8
> as.data.frame(colSums(is.na(data)))
       colSums(is.na(data))
date                      0
symbol                    0
open                      0
close                     0
low                       0
high                      0
volume                    0
class                     0
```

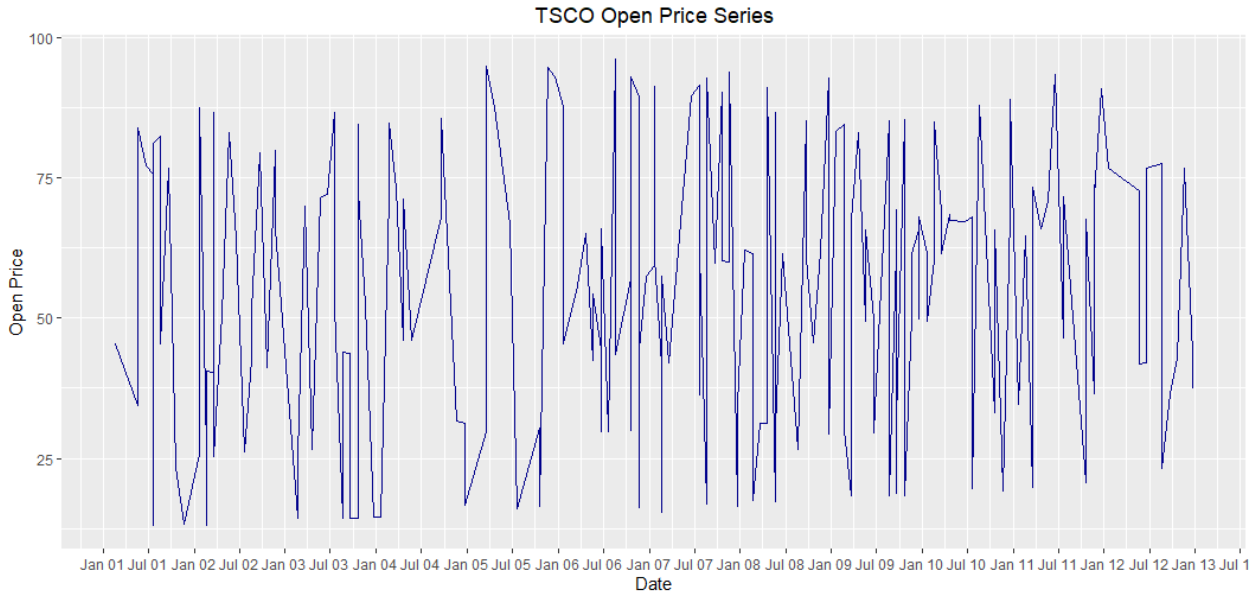
```
install.packages("quantmod")
install.packages("ggplot2")
library(quantmod)
library(ggplot2)
```

```
> library(quantmod)
Loading required package: xts
Loading required package: zoo

Attaching package: 'zoo'

The following objects are masked from 'package:base':

    as.Date, as.Date.numeric

Loading required package: TTR
Registered S3 method overwritten by 'quantmod':
  method            from
  as.zoo.data.frame zoo
Warning messages:
1: package 'quantmod' was built under R version 4.0.5
2: package 'xts' was built under R version 4.0.5
3: package 'zoo' was built under R version 4.0.5
4: package 'TTR' was built under R version 4.0.5
> library(ggplot2)
Warning message:
package 'ggplot2' was built under R version 4.0.5
```

```
data1 = subset(data, symbol == "TSCO")
```

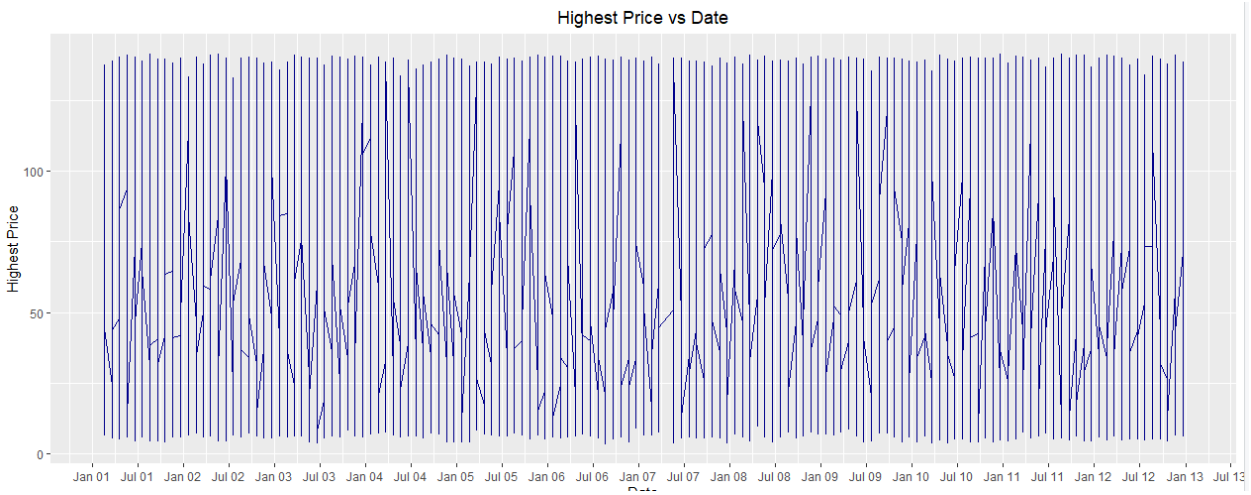
```
> data1 = subset(data, symbol == "TSCO")
> ggplot(data1, aes(x = data1[,1], y = data1[,3])) + geom_line(color = "darkblue") + ggtitle
("TSCO Open Price Series") + xlab("Date") + ylab("Open Price") + theme(plot.title = element_te
xt(hjust = 0.5)) + scale_x_date(date_labels = "%b %y", date_breaks = "6 months")
```

```
ggplot(data1, aes(x = data1[,1], y = data1[,3])) + geom_line(color = "darkblue") + ggtitle("TSCO Open Price Series") + xlab("Date") + ylab("Open Price") + theme(plot.title = element_text(hjust = 0.5)) + scale_x_date(date_labels = "%b %y", date_breaks = "6 months")
```



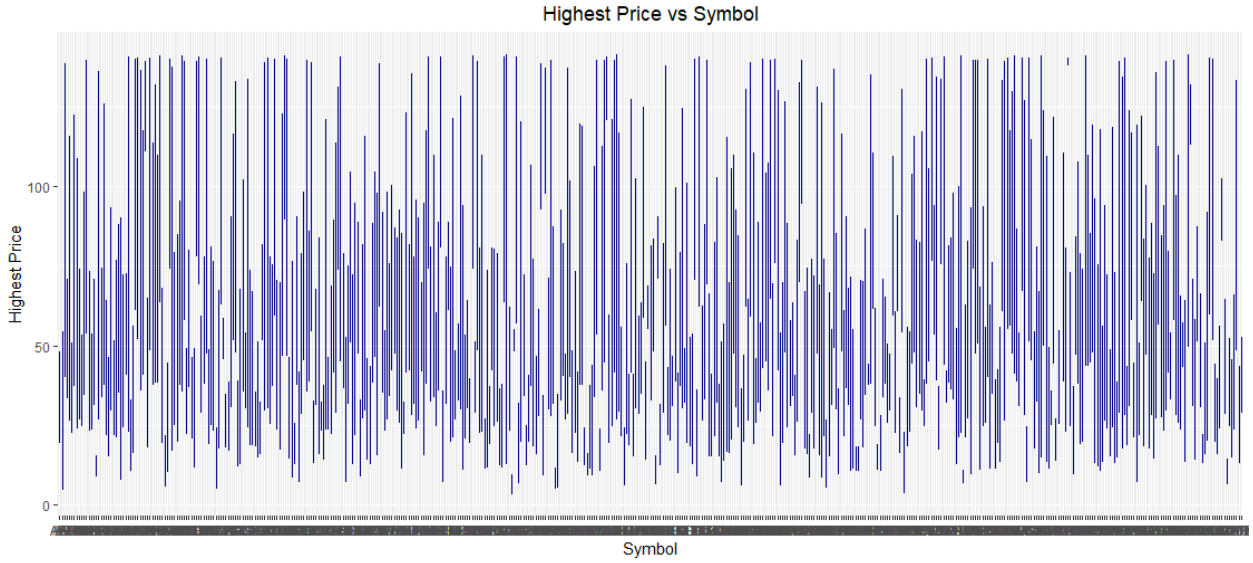
ggplot(data, aes(x =data[,1], y = data[,6])) + geom_line(color = "darkblue") + ggtitle("Highest Price vs Date") + xlab("Date") + ylab("Highest Price") + theme(plot.title = element_text(hjust = 0.5)) + scale_x_date(date_labels = "%b %y", date_breaks = "6 months")

```
> ggplot(data, aes(x =data[,1], y = data[,6])) + geom_line(color = "darkblue") + ggtitle("High
est Price vs Date") + xlab("Date") + ylab("Highest Price") + theme(plot.title = element_text(h
just = 0.5)) + scale_x_date(date_labels = "%b %y", date_breaks = "6 months")
```



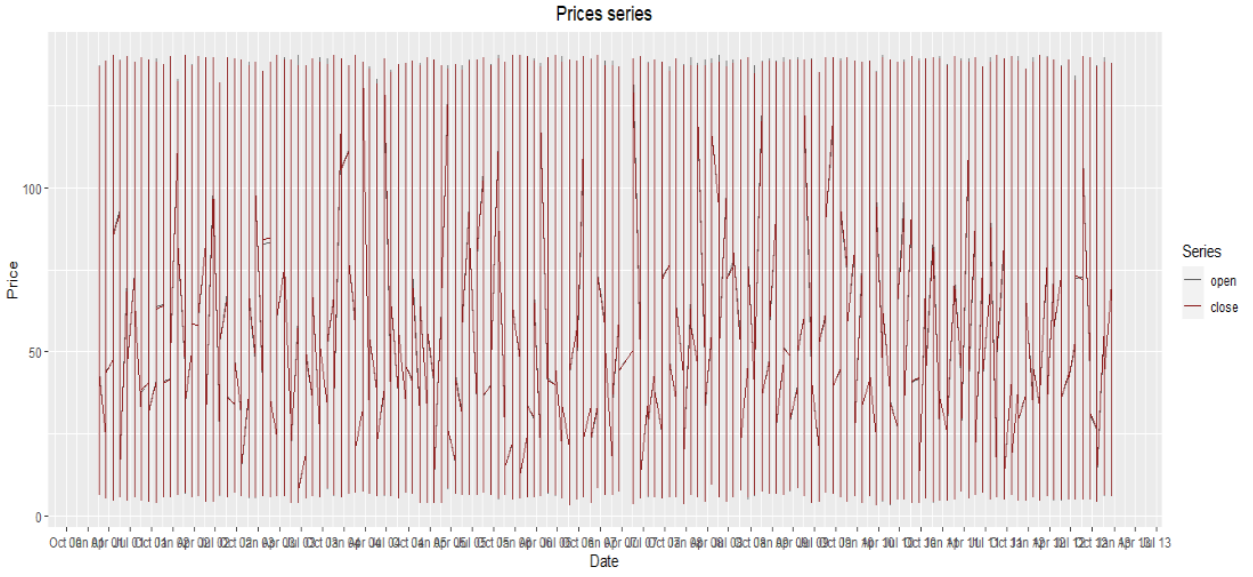
ggplot(data, aes(x = data[,2], y = data[,6])) + geom_line(color = "darkblue") + ggtitle("Highest Price vs Symbol") + xlab("Symbol") + ylab("Highest Price") + theme(plot.title = element_text(hjust = 0.5))

```
> ggplot(data, aes(x = data[,2], y = data[,6])) + geom_line(color = "darkblue") + ggtitle("Highest P
rice vs Symbol") + xlab("Symbol") + ylab("Highest Price") + theme(plot.title = element_text(hjust =
 0.5))
```



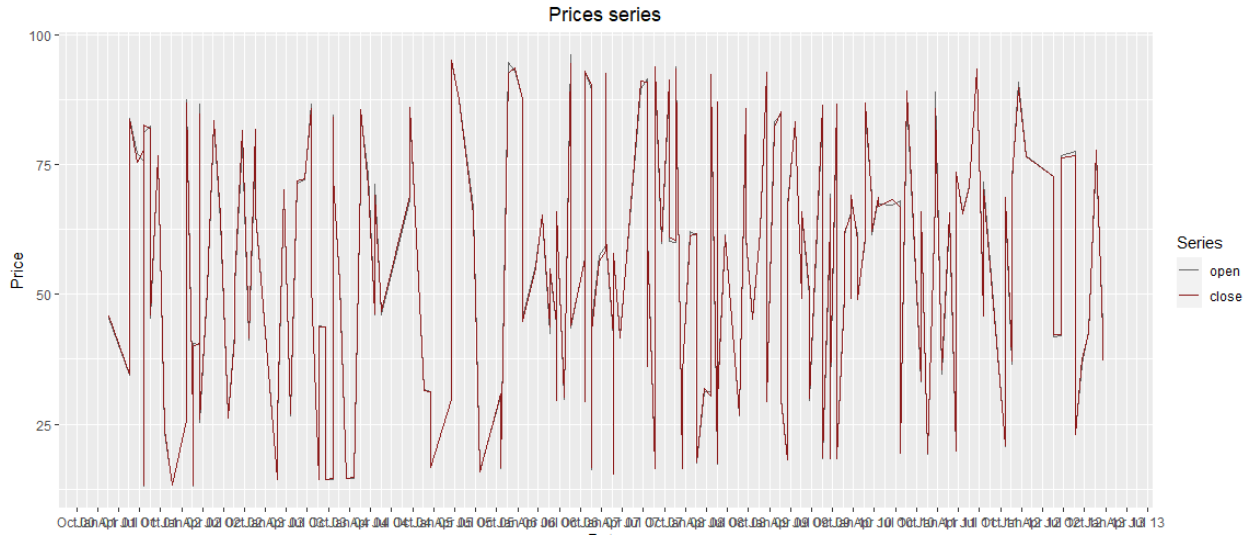

```
ggplot(data, aes(x = data[,1])) +
  geom_line(aes(y = open, color = "open")) + ggtitle("Prices series") +
  geom_line(aes(y = close, color = "close")) + xlab("Date") + ylab("Price") +
  theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank()) +
  scale_x_date(date_labels = "%b %y", date_breaks = "3 months") +
  scale_colour_manual("Series", values=c("open"="gray40", "close"="firebrick4"))
```

```
> ggplot(data, aes(x = data[,1])) +
+     geom_line(aes(y = open, color = "open")) + ggtitle("Prices series") +
+     geom_line(aes(y = close, color = "close")) + xlab("Date") + ylab("Price") +
+     theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank()) +
+     scale_x_date(date_labels = "%b %y", date_breaks = "3 months") +
+     scale_colour_manual("Series", values=c("open"="gray40", "close"="firebrick4"))
```



```
ggplot(data1, aes(x = data1[,1])) +
  geom_line(aes(y = open, color = "open")) + ggtitle("Prices series") +
  geom_line(aes(y = close, color = "close")) + xlab("Date") + ylab("Price") +
  theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank()) +
  scale_x_date(date_labels = "%b %y", date_breaks = "3 months") +
  scale_colour_manual("Series", values=c("open"="gray40", "close"="firebrick4"))
```

```
> ggplot(data1, aes(x = data1[,1])) +
+     geom_line(aes(y = open, color = "open")) + ggtitle("Prices series") +
+     geom_line(aes(y = close, color = "close")) + xlab("Date") + ylab("Price") +
+     theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank()) +
+     scale_x_date(date_labels = "%b %y", date_breaks = "3 months") +
+     scale_colour_manual("Series", values=c("open"="gray40", "close"="firebrick4"))
```




```
plot(data1[,4], main = "Closing Prices")
```

```
> plot(data1[,4], main = "Closing Prices")
```

```
index <- sample(1:nrow(data), size=0.2*nrow(data))
   test <- data[index, ]
   train <- data[-index, ]
install.packages('randomForest')
library(randomForest)
```

```
> index <- sample(1:nrow(data), size=0.2*nrow(data))
> test <- data[index, ]
> train <- data[-index, ]
> library(randomForest)
randomForest 4.6-14
Type rfNews() to see new features/changes/bug fixes.

Attaching package: 'randomForest'

The following object is masked from 'package:ggplot2':

    margin

Warning message:
package 'randomForest' was built under R version 4.0.5
```

```
model <- randomForest(as.factor(class)~ +low+high+open+close+volume, train,
importance=TRUE, ntree=100, mtry=4)
print(model)
```

```
> model <- randomForest(as.factor(class)~ +low+high+open+close+volume, trai
n, importance=TRUE, ntree=100, mtry=4)
> print(model)

Call:
 randomForest(formula = as.factor(class) ~ +low + high + open +      close +
 volume, data = train, importance = TRUE, ntree = 100,      mtry = 4)
               Type of random forest: classification
                     Number of trees: 100
No. of variables tried at each split: 4

        OOB estimate of  error rate: 5.44%
Confusion matrix:
         high   low Neutral class.error
high    18034     0    1226  0.06365524
low         0 16069    1170  0.06786937
Neutral   669   589   29472  0.04093719
```

```
attributes(model)
```

```
> attributes(model)
$names
 [1] "call"            "type"            "predicted"
 [4] "err.rate"        "confusion"       "votes"
 [7] "oob.times"       "classes"         "importance"
[10] "importanceSD"    "localImportance" "proximity"
[13] "ntree"           "mtry"            "forest"
[16] "y"               "test"            "inbag"
[19] "terms"

$class
[1] "randomForest.formula" "randomForest"
```

library(caret)
p2 = predict(model,test)

```
> library(caret)
Loading required package: lattice
Warning message:
package 'caret' was built under R version 4.0.5
> p2 = predict(model,test)
```

confusionMatrix(p2,as.factor(test$class))

```
> confusionMatrix(p2,as.factor(test$class))
Confusion Matrix and Statistics

          Reference
Prediction high  low Neutral
   high    4501    0     159
   low        0 3957     132
   Neutral  343  286    7429

Overall Statistics

               Accuracy : 0.9453
                 95% CI : (0.9417, 0.9487)
    No Information Rate : 0.4593
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9143

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: high Class: low Class: Neutral
Sensitivity               0.9292     0.9326         0.9623
Specificity               0.9867     0.9895         0.9308
Pos Pred Value            0.9659     0.9677         0.9219
Neg Pred Value            0.9718     0.9775         0.9667
Prevalence                0.2882     0.2525         0.4593
Detection Rate            0.2678     0.2354         0.4420
Detection Prevalence      0.2773     0.2433         0.4794
Balanced Accuracy         0.9579     0.9610         0.9465
```