Post02: Stock market data analysis using quantmod package

Sicheng Ouyang

2017/12/02

Introduction:

- In this post, I will illustrate how to use R to do stock market data analysis. First, I will show how to get the stock data by introducing an R package called “quantmod”. After that, I will do some calculation using the package “PerformanceAnalytics”. And then we’re going analyze the trend of the stock price by visualizing the stock data and interpreting the meaning of the graph that we draw using the functions in quantmod package. In order to analyze the stock price trend, I will also introduce a couple of technical indicators.

source:http://milenia-finance.com/stock-exchange-nature-operates/

1) First of all, we need to install and load the needed package

```
install.packages(c('quantmod','zoo','xts','TTR'))
```

```
library(quantmod)
```

```
## Warning: package 'quantmod' was built under R version 3.4.2
```

```
## 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
```

```
## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
library(zoo)
library(xts)
library(TTR)
```

2) Get the stock data.

In order to get the stock data, we need to use an R package called “quantmod”.

Here is a brief introduction of “quantmod”:

- The quantmod is an R package which can assist the quantitative trader in the development, testing, and deployment of statistically based trading models. With the comprehensive tools for data management and visualization in this package, we can quickly and cleanly explore and build trading models.
- For more information: https://cran.r-project.org/web/packages/quantmod/quantmod.pdf

Using the quantmod package, we can download the stock data that we need from different sources (ie:‘Yahoo’, ‘Google’). In this post, I want to use ‘Yahoo’ as our data source.

Let’s download Apple’s stock data by calling the function `getSymbols()`. We can get the stock data in a specific period by setting the parameters.

```
getSymbols("AAPL",from = "2017-06-01",to = "2017-12-01",src = "yahoo")
```

```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```
##
## WARNING: There have been significant changes to Yahoo Finance data.
## Please see the Warning section of '?getSymbols.yahoo' for details.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.yahoo.warning"=FALSE).
```

```
## [1] "AAPL"
```

```
head(AAPL,10)
```

```
##            AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume
## 2017-06-01    153.17    153.33   152.22     153.18    16404100
## 2017-06-02    153.58    155.45   152.89     155.45    27770700
## 2017-06-05    154.34    154.45   153.46     153.93    25331700
## 2017-06-06    153.90    155.81   153.78     154.45    26624900
## 2017-06-07    155.02    155.98   154.48     155.37    21069600
## 2017-06-08    155.25    155.54   154.40     154.99    21250800
## 2017-06-09    155.19    155.19   146.02     148.98    64882700
## 2017-06-12    145.74    146.09   142.51     145.42    72307300
## 2017-06-13    147.16    147.45   145.15     146.59    34165400
## 2017-06-14    147.50    147.50   143.84     145.16    31531200
##            AAPL.Adjusted
## 2017-06-01      152.0343
## 2017-06-02      154.2873
## 2017-06-05      152.7787
## 2017-06-06      153.2948
## 2017-06-07      154.2079
## 2017-06-08      153.8307
## 2017-06-09      147.8657
## 2017-06-12      144.3323
## 2017-06-13      145.4936
## 2017-06-14      144.0743
```

- From the data table, we can access the opening price, the lowest price, closing price, volume and adjusted closing price for Apple.

---

3) Calculate the return rate

After getting the data for Apple’s stock price, we can use the data to calculate two kinds of rate of return: simple rate of return and logarithmic rate of return.

3.1.1) Calculate simple rate of return

Here is the formula for the simple rate of return:

$$Rt = \frac{Close(t)-Close(t-1)}{Colse(t-1)}$$

We need to use the closing price of the current day minus the closing price a day before. And then divide the subtraction result by the closing price a day before to get the rate.

Get the closing price for Apple.

```
cPrice <- AAPL[,4]
```

We use `lag()` function to shift the time base back by 1 day by setting the parameter to 1.

- For more information about lag function: https://www.rdocumentation.org/packages/stats/versions/3.4.1/topics/lag

```
lagcPrice <- lag(cPrice,1)
head(lagcPrice)
```

```
##            AAPL.Close
## 2017-06-01         NA
## 2017-06-02     153.18
## 2017-06-05     155.45
## 2017-06-06     153.93
## 2017-06-07     154.45
## 2017-06-08     155.37
```

Calculate the simple rate of return:

```
calCprice <- merge(cPrice,lagcPrice)
simpleRate <- (cPrice-lagcPrice)/lagcPrice
names(simpleRate)="simplerate"
calrate=merge(calCprice,simpleRate)
head(calrate)
```

```
##            AAPL.Close AAPL.Close.1   simplerate
## 2017-06-01     153.18           NA           NA
## 2017-06-02     155.45       153.18  0.014819194
## 2017-06-05     153.93       155.45 -0.009778090
## 2017-06-06     154.45       153.93  0.003378185
## 2017-06-07     155.37       154.45  0.005956607
## 2017-06-08     154.99       155.37 -0.002445710
```

3.1.2) Calculate logarithmic rate of return.

The formula for log logarithmic rate of return: \[Rt = \ln(Close(t)/Close(t-1))\]

Here's an introduction of the difference between simple and logarithmic rate of return: http://www.dcfnerds.com/94/arithmetic-vs-logarithmic-rates-of-return/

Calculate the logarithmic rate of return:

```
lograte <- log(cPrice/lagcPrice)
names(lograte)<-"lograte"
callog<-merge(calCprice,lograte)
head(callog)
```

```
##            AAPL.Close AAPL.Close.1      lograte
## 2017-06-01     153.18           NA           NA
## 2017-06-02     155.45       153.18  0.014710462
## 2017-06-05     153.93       155.45 -0.009826209
## 2017-06-06     154.45       153.93  0.003372492
## 2017-06-07     155.37       154.45  0.005938937
## 2017-06-08     154.99       155.37 -0.002448706
```

Compare the simple rate of return with the logarithmic one:

```
rrate<-merge(calrate,lograte)
head(rrate)
```

```
##            AAPL.Close AAPL.Close.1   simplerate      lograte
## 2017-06-01     153.18           NA           NA           NA
## 2017-06-02     155.45       153.18  0.014819194  0.014710462
## 2017-06-05     153.93       155.45 -0.009778090 -0.009826209
## 2017-06-06     154.45       153.93  0.003378185  0.003372492
## 2017-06-07     155.37       154.45  0.005956607  0.005938937
## 2017-06-08     154.99       155.37 -0.002445710 -0.002448706
```

3.2) Caculate simple return rate using PerformanceAnalytics package

- If we don't want to calculate the rates of return by hand, we can do the calculation using a package called 'PerformanceAnalytics'. It is an R package with a collection of econometric functions for performance and risk analysis.
- For more imformation about 'PerformanceAnalytics': https://cran.r-project.org/web/packages/PerformanceAnalytics/PerformanceAnalytics.pdf

We load the package first:

```
library(PerformanceAnalytics)
```

```
##
## Attaching package: 'PerformanceAnalytics'
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

Calculate the rates by calling `periodReturn()` and setting the parameters:

```
rate=periodReturn(cPrice,period="daily",type="arithmetic")
head(rate)
```

```
##            daily.returns
## 2017-06-01   0.000000000
## 2017-06-02   0.014819194
## 2017-06-05  -0.009778090
## 2017-06-06   0.003378185
## 2017-06-07   0.005956607
## 2017-06-08  -0.002445710
```

```
rate2=periodReturn(cPrice,period="daily",type="log")
head(rate2)
```

```
##            daily.returns
## 2017-06-01   0.000000000
## 2017-06-02   0.014710462
## 2017-06-05  -0.009826209
## 2017-06-06   0.003372492
## 2017-06-07   0.005938937
## 2017-06-08  -0.002448706
```

```
names(rate)<-"simplerate"
names(rate2)<-"lograte"

rrrate<-merge(rate,rate2)
head(rrrate)
```

```
##              simplerate      lograte
## 2017-06-01  0.000000000  0.000000000
## 2017-06-02  0.014819194  0.014710462
## 2017-06-05 -0.009778090 -0.009826209
## 2017-06-06  0.003378185  0.003372492
## 2017-06-07  0.005956607  0.005938937
## 2017-06-08 -0.002445710 -0.002448706
```

Compare the result we get with the calculation by hand in the last part. We found that they are exactly the same.

```
head(rrate[,c(-1,-2)])
```

```
##              simplerate      lograte
## 2017-06-01           NA           NA
## 2017-06-02  0.014819194  0.014710462
## 2017-06-05 -0.009778090 -0.009826209
## 2017-06-06  0.003378185  0.003372492
## 2017-06-07  0.005956607  0.005938937
## 2017-06-08 -0.002445710 -0.002448706
```

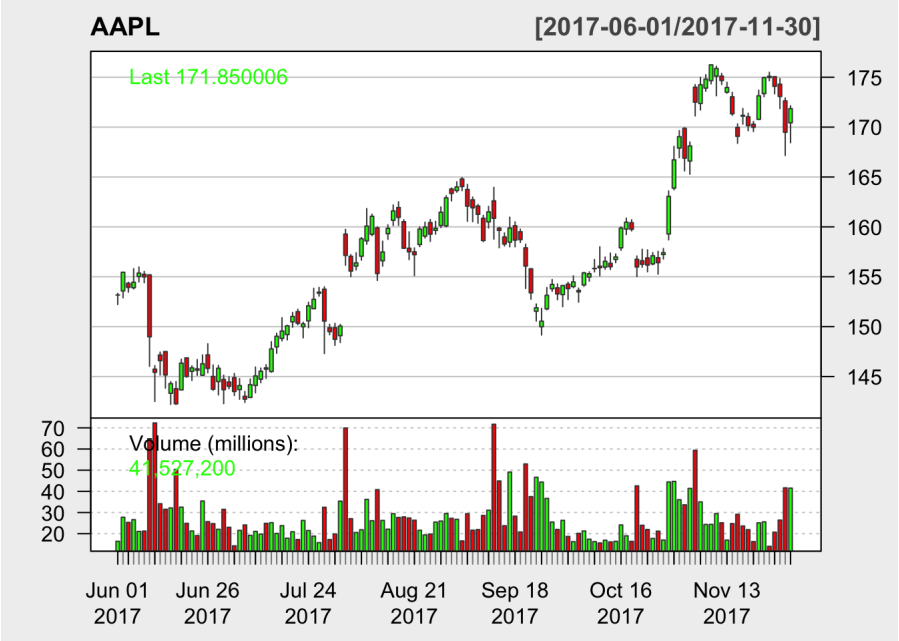
```
head(rrrate)
```

```
##              simplerate      lograte
## 2017-06-01  0.000000000  0.000000000
## 2017-06-02  0.014819194  0.014710462
## 2017-06-05 -0.009778090 -0.009826209
## 2017-06-06  0.003378185  0.003372492
## 2017-06-07  0.005956607  0.005938937
## 2017-06-08 -0.002445710 -0.002448706
```

4) Visualize the data

After getting the data and calculating the rate of return, we can analyze the trend of the stock price by visualizing it.

```
chartSeries(AAPL,up.col='green', dn.col='red',theme="white")
```

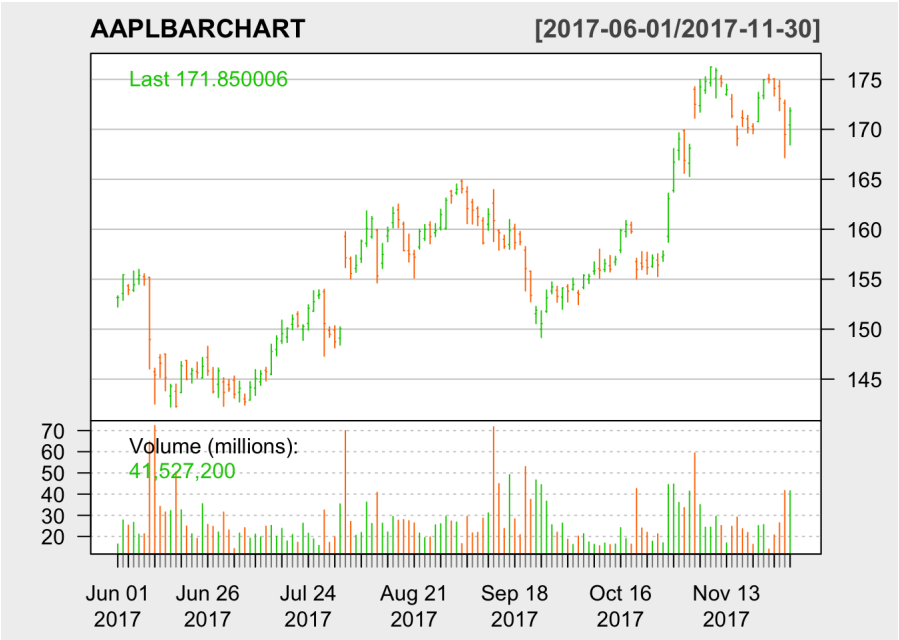


- For the upper part of the chart, we have many green and red candlestick shaded boxes. If the box is green, that means the stock price has increased during that day. The lower bound of the box shows the opening price of the stock, and the upper bound shows the closing price. If the box is red, that means the stock price on the corresponding day has decreased. The upper bound of the box shows the opening price and the lower bound shows the closing price. The top of the thin line which is placed above the box shows the highest point the price has reached, and the bottom of the thin line which is placed below the box shows the lowest point the price has reached.
- The bottom part of the chart shows the volume for each day. And the volume is defined as the number of shares or contracts traded in one day.

- We can draw other types of charts by putting different parameters.

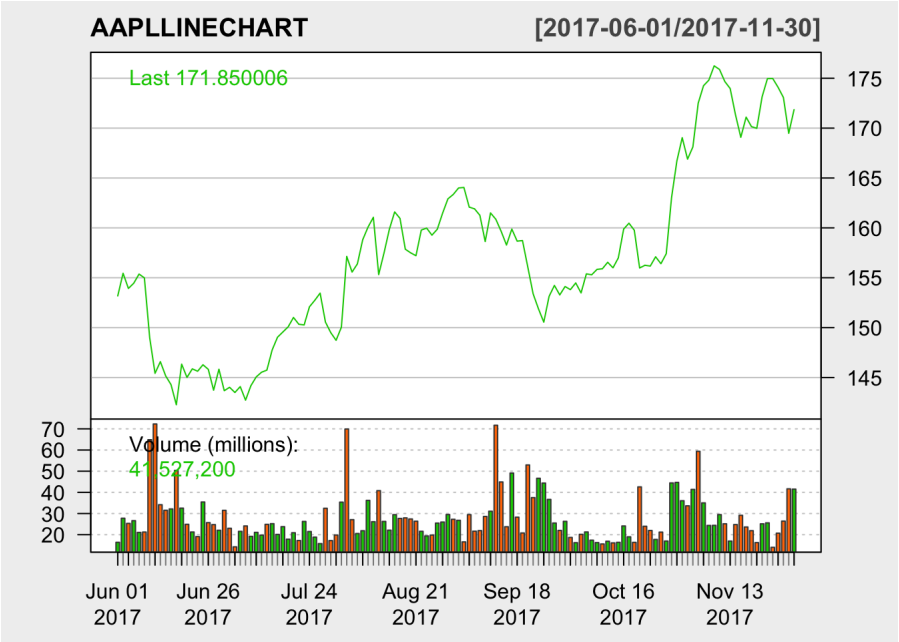
Barchart:

```
chartSeries(AAPL,name = "AAPLBARCHART", type="bars",theme="white")
```



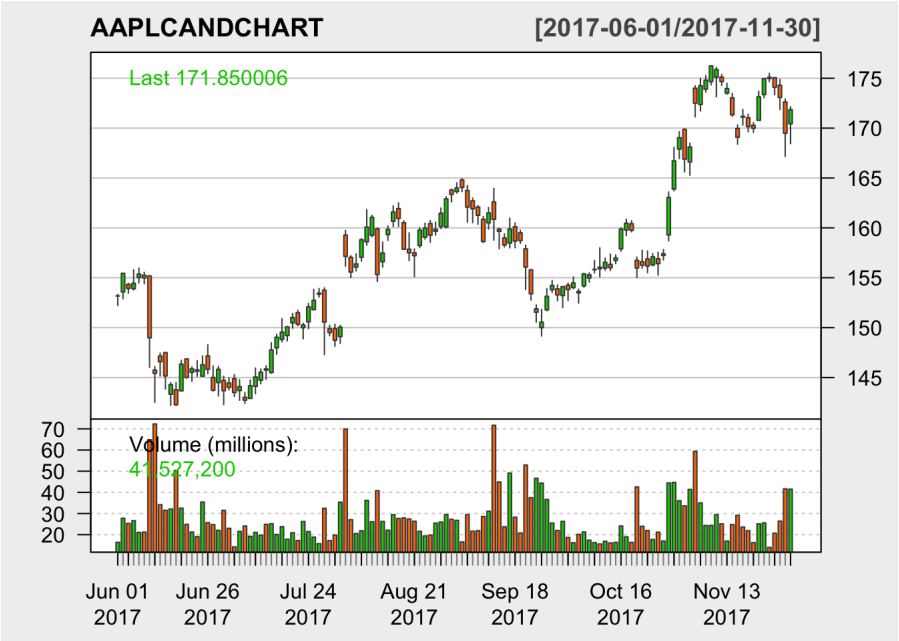
Linechart:

```
chartSeries(AAPL,name = "AAPLLINECHART", type="line",theme="white")
```



Candlestickschart:

```
chartSeries(AAPL,name = "AAPLCANDCHART", type="candlesticks",theme="white")
```



- From the charts, we can see that the overall trend of Apple's stock price is going up from Jun 01 to Nov 30 in 2017. In particular, the price kept going up in the period between the late June and the late July, and the price increased at a high rate from the late Oct to the early Nov. However, there is a big price drop in the early Jun. We will try to find out the reasons for those price changes later in this post.

---

5) Analyze the trend of stock price by adding different indicators and the Bollinger Bands on the chart

Base on the chart of the stock price, we can analyze the trend of the price by adding different technical indicators.

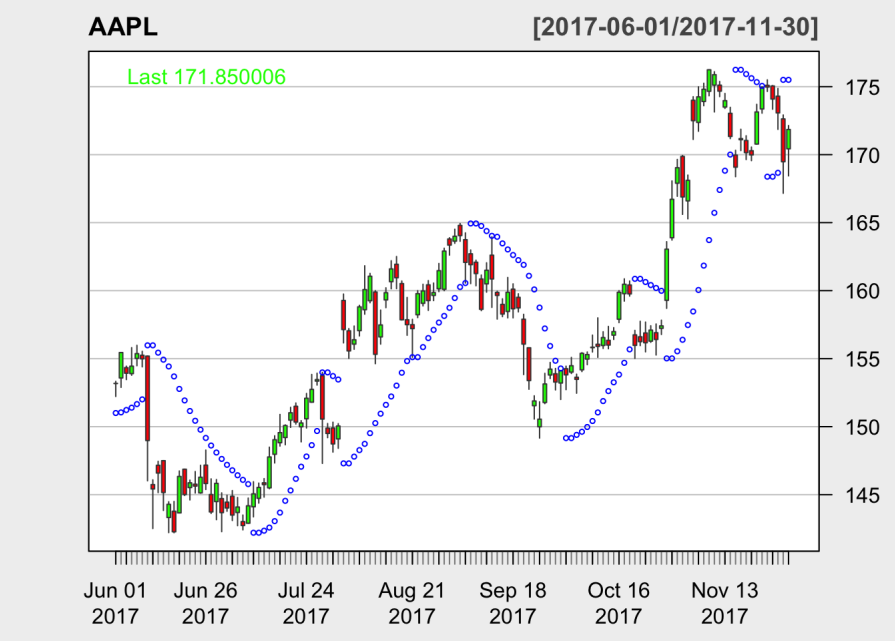
5.1) Add the parabolic SAR indicator on the chart

Here's a brief introduction of SAR indicator:

- The SAR indicator is developed by a technician called Welles Wilder to determine the direction of an asset's momentum and the point in time when this momentum has a higher-than-normal probability of switching directions.
- For more information: https://www.investopedia.com/articles/technical/02/042202.asp

We can use `addSAR()` function to plot the SAR indicator on the chart to see the trend of the stock price. In order to combine two graphs together, we can set the parameter 'TA' = '`addSAR()`' for the function chartSeries():

```
chartSeries(AAPL,up.col='green', dn.col='red',theme="white", TA='addSAR()')
```



- From this chart with SAR indicator, we can see the trend of Apple's stock price clearly. From Jun 01 to the beginning of Jul, the price kept going down. And the price was going up in general from Jul to the end of Aug. Starting from Oct 27, the stock has increased dramatically for over two weeks. If the position of the dot is below the price, that means the up-going momentum is expected to remain. So the trader may keep the stock for a little longer because the price of the stock is expected to increase. On the other hand, if the the dot is above the price, then the momentum has a great chance to remain downward. For example, on Aug 7, the position of the dot is way lower than the current price. That means we can keep the stock because it has a high probability to keep increasing. Conversely, on Jun 08, the dot is above the price. So we should consider selling the stock because the price is about to keep falling.

5.2) Add the parabolic ADX indicator on the chart

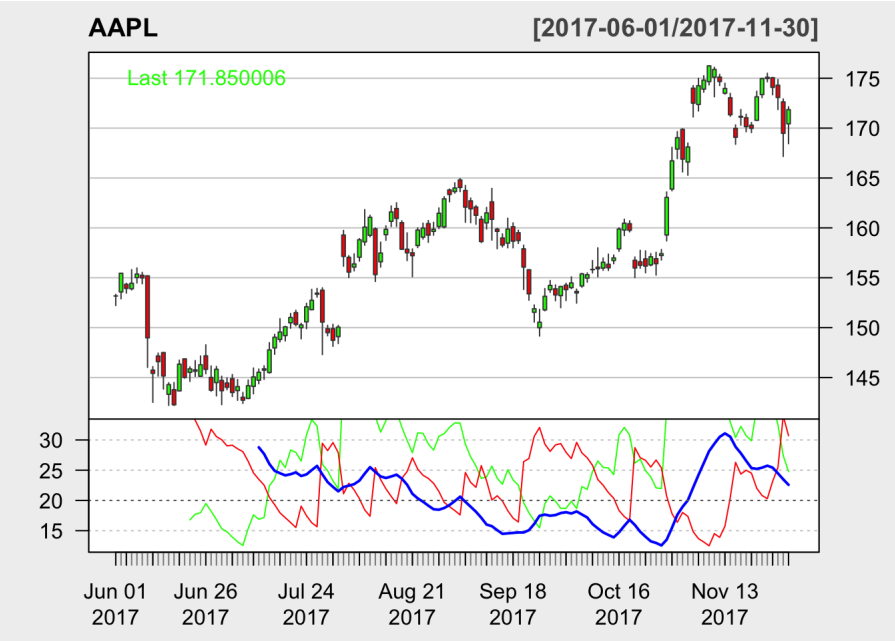
Here's a brief introduction of ADX indicator:

The average directional movement index (ADX) was developed in 1978 by J. Welles Wilder as an indicator of trend strength in a series of prices of a financial instrument. It is based on moving average of price range expansion over a given period of time.

- For more information: https://en.wikipedia.org/wiki/Average_directional_movement_index

We can set TA = '`addADX()`' to plot the SAR indicator on the chart to see the trend of the stock price.

```
chartSeries(AAPL,up.col='green', dn.col='red',theme="white", TA= 'addADX()')
```



- The blue line in the graph above shows the ADX value. The range of ADX's value is from 0-100. However, the ADX value is not directional, it only shows the strength of trend (the trend of the price may be going upward or downward). Here is a ADX value table to help us to determine whether a trend is strong or not:

ADX Value	Trend Strength
0-25	Absent or Weak Trend
25-50	Strong Trend
50-75	Very Strong Trend
75-100	Extremely Strong Trend

- Base on the table, there're only two periods of time when the strength of the trend is strong enough for us to decide whether to buy/sell the stock or not. From Jul 10 to Jul 14, the momentum of the stock price was in the upward direction, and the ADX value is near or above 25. So that we can consider buying Apple's stock because we can predict that the stock price might keep going up. However, from Sep 11 to Oct 30, the ADX value is below 25. We should consider not to trade (buy or sell) the stock because the trend of the price is weak. We can't predict the price trend clearly.

5.3) Add the parabolic CCI indicator on the chart

Here's a brief introduction of CCI indicator: * The Commodity Channel Index(CCI), was developed by Donald Lambert in 1980. It compares the current price to an average price over a period of time. The indicator fluctuates above or below zero, moving into positive or negative territory. While most values, approximately 75%, will fall between -100 and +100, about 25% of the values will fall outside this range, indicating a lot of weakness or strength in the price movement.

- For more information: https://www.investopedia.com/articles/active-trading/031914/how-traders-can-utilize-cci-commodity-channel-index-trade-stock-trends.asp

Let's set TA = '`addCCI()`' to plot the CCI indicator on the chart to see the trend of the stock price.

```
chartSeries(AAPL,up.col='green', dn.col='red',theme="white", TA = 'addCCI()')
```

- When the CCI is above +100, we can consider buying the stock. On the other hand, we might want to sell it went CCI is below -100. From the graph, we can clearly see that there're 5 periods times which are good for us to buy the stock. And it is wise for us to sell the stock from Sep 20 to Sep 30 because the CII is below -100.

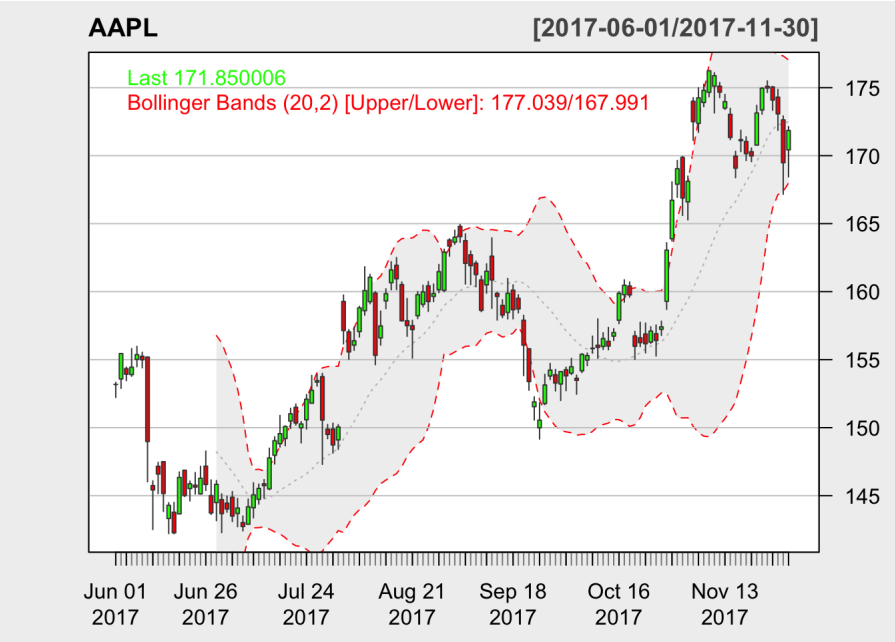
5.4) Add the Bollinger Bands on the chart

The last method I want to introduce to predict the price trend of a stock is called Bollinger Bands.
Here's a brief introduction of Bollinger Bands: Bollinger Bands are a technical indicator developed by John Bollinger. The indicator forms a channel around the price movements of an asset. The price of the stock is bounded by the upper and lower bound. The distance between the upper and lower band is determined by standard deviations.

- For more information: https://www.thebalance.com/day-trading-with-bollinger-bands-1031210

Let's set TA = '`addBBands()`'

```
chartSeries(AAPL,up.col='green', dn.col='red',theme="white", TA = 'addBBands()' )
```



- When the price is moving along the upper bound, that means the trend is strong and in an upward direction. If the price is touching the lower bound, that means the stock price is in a strong downward trend. Form this graph, we can see that the price was touching the upper bound from the beginning of Aug to Aug 14 and from Oct 27 to Nov 6. Those are the good time for us to buy the stock. However, the price was moving along the lower bound from Sep 18 to Sep 25. So it is probably not the good time for you to buy the stock.

6) Find out the reason for the huge price change

In the previous parts, we found that a few huge price changes occurred from Jun to Nov in 2017, and we want to figure out what causes the huge price change.

We want to list all the date when the change of price is over 1 percent:

```
head(AAPL)
```

```
##            AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume
## 2017-06-01    153.17    153.33   152.22     153.18    16404100
## 2017-06-02    153.58    155.45   152.89     155.45    27770700
## 2017-06-05    154.34    154.45   153.46     153.93    25331700
## 2017-06-06    153.90    155.81   153.78     154.45    26624900
## 2017-06-07    155.02    155.98   154.48     155.37    21069600
## 2017-06-08    155.25    155.54   154.40     154.99    21250800
##            AAPL.Adjusted
## 2017-06-01      152.0343
## 2017-06-02      154.2873
## 2017-06-05      152.7787
## 2017-06-06      153.2948
## 2017-06-07      154.2079
## 2017-06-08      153.8307
```

Use `Delt()` function to calculate the change percentage of the closing price. We can get the closing price by calling `CL(AAPL)`

```
aaa<-Delt(Cl(AAPL))
aaaover<-aaa[which(aaa > 0.01), ]
length(aaaover)
```

```
## [1] 27
```

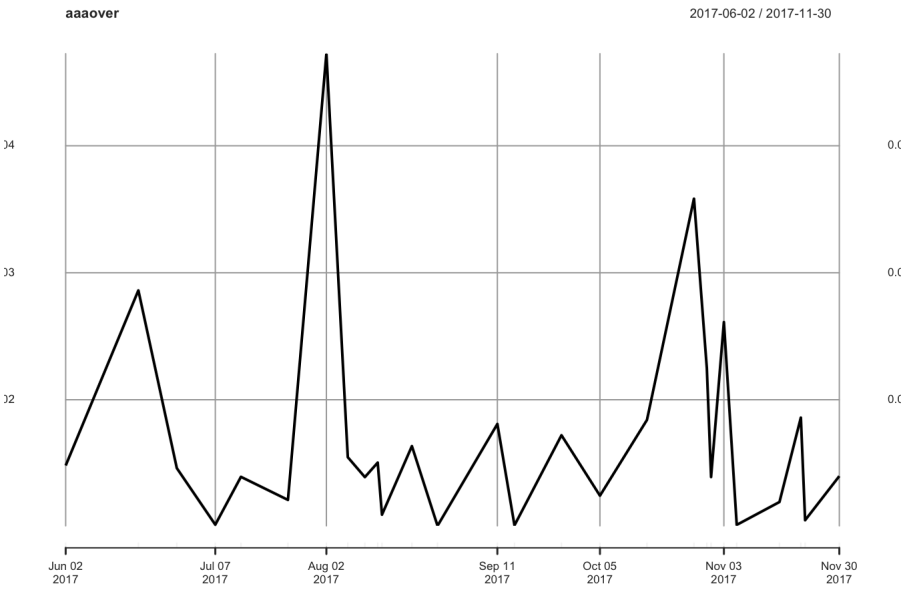
```
aaaover
```

```
##            Delt.1.arithmetic
## 2017-06-02        0.01481919
## 2017-06-19        0.02860752
## 2017-06-28        0.01461077
## 2017-07-07        0.01015902
## 2017-07-13        0.01392891
## 2017-07-24        0.01211148
## 2017-08-02        0.04725089
## 2017-08-07        0.01547413
## 2017-08-11        0.01390670
## 2017-08-14        0.01504959
## 2017-08-15        0.01094776
## 2017-08-22        0.01634751
## 2017-08-28        0.01007131
## 2017-09-11        0.01809238
## 2017-09-15        0.01010871
## 2017-09-26        0.01720356
## 2017-10-05        0.01244464
## 2017-10-16        0.01840882
## 2017-10-27        0.03582999
## 2017-10-30        0.02250842
## 2017-10-31        0.01391550
## 2017-11-03        0.02611385
## 2017-11-06        0.01014493
## 2017-11-16        0.01194703
## 2017-11-21        0.01859044
## 2017-11-22        0.01051177
## 2017-11-30        0.01398401
```

- We found that there're 27 days whoses price change percentage is above 1%.

We can also draw a line graph to visualize it:

```
plot(aaaover)
```



- From the line graph, we can see that there're two huge price changes on Aug 02 and Oct 27. On Aug 02, the price change percentage was over 4.5%, and the price change percentage on Oct 27 was over 3.5%. We can figure out what caused those price changes by searching what happens to Apple on those particular days.

After searching on the Internet, we found that:

- Apple's stock price increased by over 4.5% because Apple is likely to launch its newest iPhone model–iPhone 8. And the Apple gave the investors confidence that the next iPhone launch is on track. Therefore, Apple's stock price increased dramatically on people's optimism for iPhone 8 sales.
- Source:http://money.cnn.com/2017/08/01/technology/business/apple-earnings/index.html
- On Oct 27, the price has increased by over 3.5% because the preorder for iphone X began on Oct. So Apple's stock price increased over 4.5% because of the huge demands for iPhone X.
- Source: https://www.appleworld.today/blog/2017/10/27/awt-news-update-october-27-2017

Here's a quick summary of what we have done in this post (Take-

Here's a quick summary of what we have done in this post (Take home message):

1. I showed how to access and download the stock data from different sources by introducing an R package called 'quantmod'.
2. I showed how to calculate the simple rate of return and logarithmic rate of return by hand or using a package called 'PerformanceAnalytics'.
3. I illustrated how to visualize the stock using the `chartSeries()` function.
4. After that, I introduce a few indicators and bollinger band to help us to analyze the price trend and then interpreted the meaning the charts.
5. At last, I listed the date when the Apple's stock price change was over 1% and figured out what's the reason behind those changes.

I hope this post will be helpful to the readers who are interested in stock data analysis.

Thanks for reading!

References:

- Image: http://milenia-finance.com/wp-content/uploads/635963392980931659203580 9433_stock-market.jpg
- Introduction of 'quantmod' package: https://cran.r-project.org/web/packages/quantmod/quantmod.pdf
- Introduction of `lag()` function: https://www.rdocumentation.org/packages/stats/versions/3.4.1/topics/lag
- Difference between simple and logarithmic rate of change: http://www.dcfnerds.com/94/arithmetic-vs-logarithmic-rates-of-return/
- Information about 'PerformanceAnalytics': https://cran.r-project.org/web/packages/PerformanceAnalytics/PerformanceAnalytics.pdf
- SAR indicator: https://www.investopedia.com/articles/technical/02/042202.asp
- ADX indicator: https://en.wikipedia.org/wiki/Average_directional_movement_index
- CCI indicator: https://www.investopedia.com/articles/active-trading/031914/how-traders-can-utilize-cci-commodity-channel-index-trade-stock-trends.asp
- Bollinger Bands: https://www.thebalance.com/day-trading-with-bollinger-bands-1031210
- Apple news: http://money.cnn.com/2017/08/01/technology/business/apple-earnings/index.html
- Apple news: https://www.appleworld.today/blog/2017/10/27/awt-news-update-october-27-2017