IST 687
APPLIED DATA SCIENCE

Professor Carlos Caicedo

School of Information Studies
SYRACUSE UNIVERSITY

\

PROJECT PROPOSAL
Predictive Analytics on Stock Market Data

By:
Rajith Jayadevan, Saurabh Jape

TABLE OF CONTENTS

INTRODUCTION

➢ AREA OF INTEREST

The world of stock market is constantly thriving under the process of modifications and alterations. Considering the fluctuations, it brings every day, making profit from it requires intensive planning. It is in the context of this fact that a step by step process is defined as the ideal approach. Stock market analysis refers to the entire procedure of monitoring and analyzing the stocks and thereby calculating the future trends. With the stock prices having the tendency to rise and fall, the whole scenario becomes volatile. However, since a defined pattern is followed by the stocks an insight can be procured subsequent to a thorough analysis. Stock market analysis is a process abided by most of the investors. Providing essential information to them, it is a proven way to extract the best out of the current status.

Having studied Enterprise Risk Management in our coursework at the iSchool and having had a background in the financial industry, we realized that analyzing stock data is an essential aspect of any financial organization. Financial giants such as Chase Manhattan corporation performs Market Risk management as part of its daily businesses. Through this project, we would be answering various questions related to stock related data. Since, data is being used by financial institutions for determining various hidden patterns and insights which helps them take better financial decisions.

➢ INTRODUCTION TO DATASET

We are going to analyze Financial stock market data and perform a Stock market analysis of different organizations. For our analysis we would be concentrating on 3 major revenue generating sectors- IT Services, Investment Banking and Internet Media. Within these sectors, we have chosen the top 3 performing companies over the past 1 year. The top 3 companies of each business have been chosen, according to the latest Bloomberg Industry Market Leaders Analysis published in February 2016.
URL: http://www.bloomberg.com/visual-data/industries/q/market-leaders

The organizations chosen are as follows:

a) IT Services:

1. IBM	2. HP	3. Accenture

b) Investment Banking:

1. JP Morgan	2. Goldman Sachs	3. Citigroup

c) Internet Media:

1. Google Inc.	2. Tencent Holdings Ltd.	3. Facebook

DATASET CHOSEN

Following are the datasets that we have chosen for this project.

Industry 1: IT Services

Dataset 1: IBM:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=IBM%2C+&ql=1

Dataset 2: Accenture:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=ACN%2C+&ql=1

Dataset 3: HP:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=HPE%2C+&ql=1

Industry 2: Investment Banking

Dataset 4: JP Morgan
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=JPM%2C+&ql=1

Dataset 5: Goldman Sachs:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=GS%2C+&ql=1

Dataset 6: Citigroup:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=C%2C+&ql=1

Industry 3: Internet Media

Dataset 7: Google:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=GOOGL%2C+&ql=1

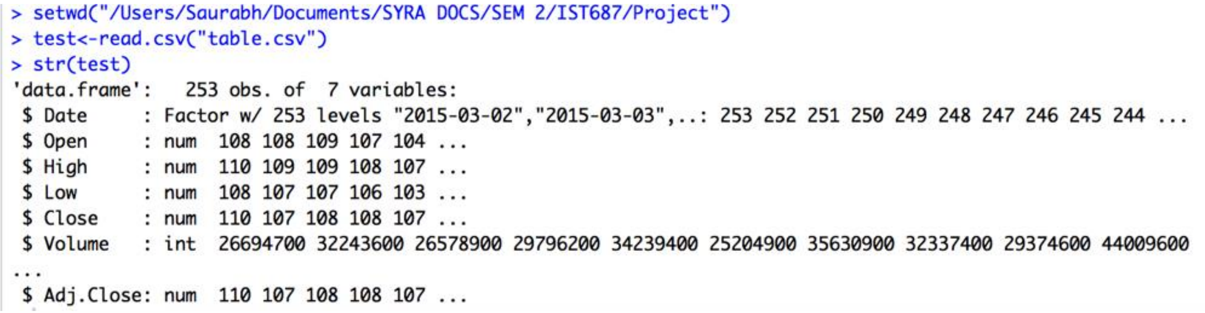
Dataset 8: Tencent:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=TCTZF%2C+&ql=1

Dataset 9: Facebook:
http://finance.yahoo.com/q/hp?a=02&b=1&c=2015&d=02&e=1&f=2016&g=d&s=FB%2C+&ql=1

Dataset 10:
http://data.okfn.org/data/core/s-and-p-500-companies#data
This data set contains other parameters such as Market/share, Price/Earnings etc. of various companies. We will be using this for further analysis.

DATASET FINDINGS

A. DATASETS (1 to 9)

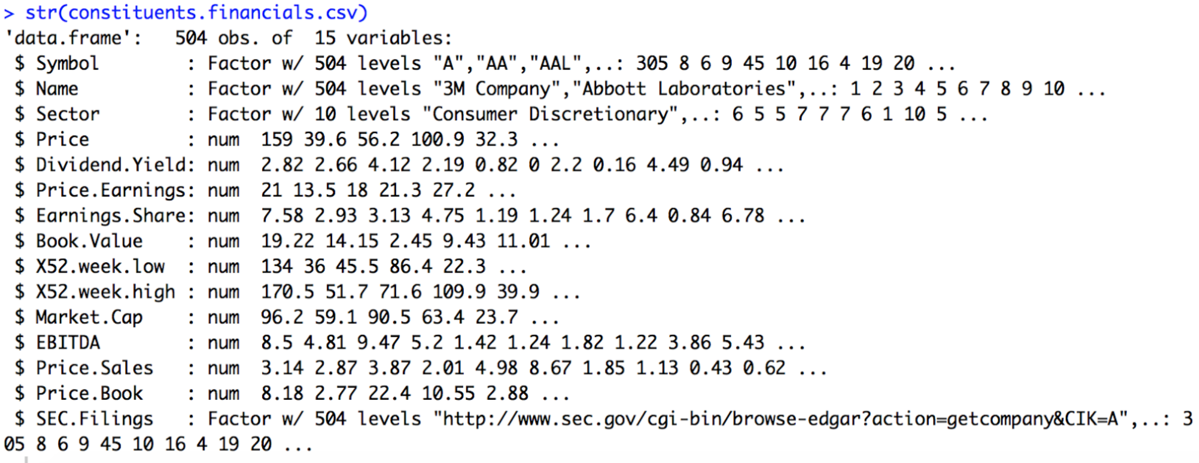


```
> setwd("/Users/Saurabh/Documents/SYRA DOCS/SEM 2/IST687/Project")
> test<-read.csv("table.csv")
> str(test)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  108 108 109 107 104 ...
 $ High     : num  110 109 109 108 107 ...
 $ Low      : num  108 107 107 106 103 ...
 $ Close    : num  110 107 108 108 107 ...
 $ Volume   : int  26694700 32243600 26578900 29796200 34239400 25204900 35630900 32337400 29374600 44009600
...
 $ Adj.Close: num  110 107 108 108 107 ...
```

Variable Name	Data Type	Description
Date	String	This denotes the date of the stock value
Open	Numeric	Opening stock value
High	Numeric	Highest stock value of the day
Low	Numeric	Lowest stock value of the day
Close	Numeric	Stock value at closing of the day
Volume	Integer	Stock Volume

As shown above, each of the 9 companies have 253 observations of 7 variables each. Thus, we have a total of: **15939** (9*253*7) data points

B. DATASET 10



```
> str(constituents.financials.csv)
'data.frame':	504 obs. of  15 variables:
 $ Symbol        : Factor w/ 504 levels "A","AA","AAL",..: 305 8 6 9 45 10 16 4 19 20 ...
 $ Name          : Factor w/ 504 levels "3M Company","Abbott Laboratories",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ Sector        : Factor w/ 10 levels "Consumer Discretionary",..: 6 5 5 7 7 7 6 1 10 5 ...
 $ Price         : num  159 39.6 56.2 100.9 32.3 ...
 $ Dividend.Yield: num  2.82 2.66 4.12 2.19 0.82 0 2.2 0.16 4.49 0.94 ...
 $ Price.Earnings: num  21 13.5 18 21.3 27.2 ...
 $ Earnings.Share: num  7.58 2.93 3.13 4.75 1.19 1.24 1.7 6.4 0.84 6.78 ...
 $ Book.Value    : num  19.22 14.15 2.45 9.43 11.01 ...
 $ X52.week.low  : num  134 36 45.5 86.4 22.3 ...
 $ X52.week.high : num  170.5 51.7 71.6 109.9 39.9 ...
 $ Market.Cap    : num  96.2 59.1 90.5 63.4 23.7 ...
 $ EBITDA        : num  8.5 4.81 9.47 5.2 1.42 1.24 1.82 1.22 3.86 5.43 ...
 $ Price.Sales   : num  3.14 2.87 3.87 2.01 4.98 8.67 1.85 1.13 0.43 0.62 ...
 $ Price.Book    : num  8.18 2.77 22.4 10.55 2.88 ...
 $ SEC.Filings   : Factor w/ 504 levels "http://www.sec.gov/cgi-bin/browse-edgar?action=getcompany&CIK=A",..: 3
05 8 6 9 45 10 16 4 19 20 ...
```

Variable Name	Data Type	Description
Name	string	Name of the company
Sector	string	Business sector the company is into
Price	string	Current Stock price
Dividend Yield	number	Dividend expressed as a percentage of a current share price
Price/Earnings	number	Current market price of a company share divided by the earnings per share of the company
Earnings/Share	number	Company's profit divided by the number of shares outstanding.
Book Value	number	The value of a security or asset as entered in a company's books
52 week low	number	Lowest price that a stock has traded at during the previous year.
52 week high	number	Highest price that a stock has traded at during the previous year.
Market Cap	number	Market capitalization is the total value of the issued shares of a publicly traded company
EBITDA	number	EBITDA margin is a measurement of a company's operating profitability
Price/Sales	number	It is the per-share stock price divided by the per-share revenue.
Price/Book	number	It is a financial ratio used to compare a company's current market price to its book value.
Symbol	string	Initials of the company
Sector	string	Industry sector that the company belongs to

As shown above, there are 504 observations of 15 variables.
Thus, we have a total of: **7560** (504*15) additional data points

MAJOR DATA QUESTIONS

With above collected data, we will be using various algorithms and performing different exploratory analysis to find answers to following questions about above mentioned data:

1. We will compare the results of each of these 2 ARIMA and Monte Carlo to come up with the best prediction model using training and test algorithm.

2. We will be predicting the stock price using cluster-then-predict to analyze rise or fall of stock prices using historical stock data.

3. We will be comparing the trends of the historical stock prices of the top 3 companies of each industry and predict which company is doing the best in the respective industry.

4. We will be comparing the trends of the historical stock prices of the best companies of each industry and predict which industry is doing the best and is safe to invest in.

5. We will be predicting the stock price of the next day, based on the past 1 year of historical data of the company using the best model for the Top 3 best performing stocks. (March 01, 2015 to March 01, 2016).

6. We shall identify if a relationship exists between the Market Capitalization of a company and the Dividend Yield that it provides to shareholders.

7. We shall analyze the company stock market value with factors such as Market Capitalization, Price/Earnings and identify if there is any correlation

8. We shall arrive at the best performing stock and conclude which company is good to invest in.

DATA PREPROCESSING, CLEANING, IMPUTATION AND TRANSFORMATION

We have included tasks of data cleaning, data mitigation, data analysis and data visualizations in this part.

A. Data Cleaning:

The financial stock market data used here is obtained from yahoo finance data set. Stock details for each company is taken from March 1st, 2015 to March 1st, 2016. Hence, the data set obtained is in a standard acceptable format. However, there are still some changes required:

1. *Data Format*: Dates, when the stock data was recorded, is in string format. In order to process the same this will have to be converted into numeric format.
2. *Missing Values*: Some rows do not have high, low, close, and volume values. Hence, these rows will have to be deleted as they don't play a major role in our data analysis. Data set - Constituents Financial - have some "NA" values in its few rows. Analysis cannot be performed on N/A values. Hence, they will have to be converted into special values or remove these rows while analyzing.

B. Data Mitigation:

Step 2 is Data Mitigation. Once the data is cleaned, the choice of data cleaning might impact accuracy of prediction models significantly.

During our data modelling and predictions, we will be using one way of data imputation:

1. *Code the missing values by a specific number*: As stated above, some statistical models do not take "NA" as an input value. In such cases, we will replace these values by some specific number (For example: 1)

As we move further in the project, we will be exploring more possible data imputation techniques.

C. Data Analysis:

For Data Analysis, we will be predicting the future stock value, using 3 modelling techniques: Monte Carlo model, ARIMA model and Cluster-then-predict to predict future stock prices. Also, for prediction, we will train and test data based on the above models.

We will try to answer all the business questions mentioned above with different approaches.

For the questions that focus on comparisons, we will perform exploratory analysis using correlations. With the help of these correlations, we will identify how the correlated variables relate with each other.
Depending upon the statistical model we will be choosing the data imputation technique. We will also try and use association rules wherever they could be applied. Based on which we can find different set of rules, confidence, support and lift of these rules. Making inferences and assumptions based on these rules.

As we move ahead with analysis, we will add few more analytical steps in this document which will be used for this case study.

D. Data Visualization:

In the Analysis part, we will be also visualizing our data-set by plotting the data using R.

Some of the visualizations would be:

1. Plotting prediction models for each stock market data companies
2. Plotting multiple prediction model for stock data of each company
3. Plot market capitalization, earnings per share for each company over different industries
4. Plotting comparison of different companies in same industry
5. Plotting comparison of top company in different industries

Note: More visualizations will be added as per data analysis performed

DATA ANALYSIS AND VISUALIZATION

STEP 1: READING THE DATASET

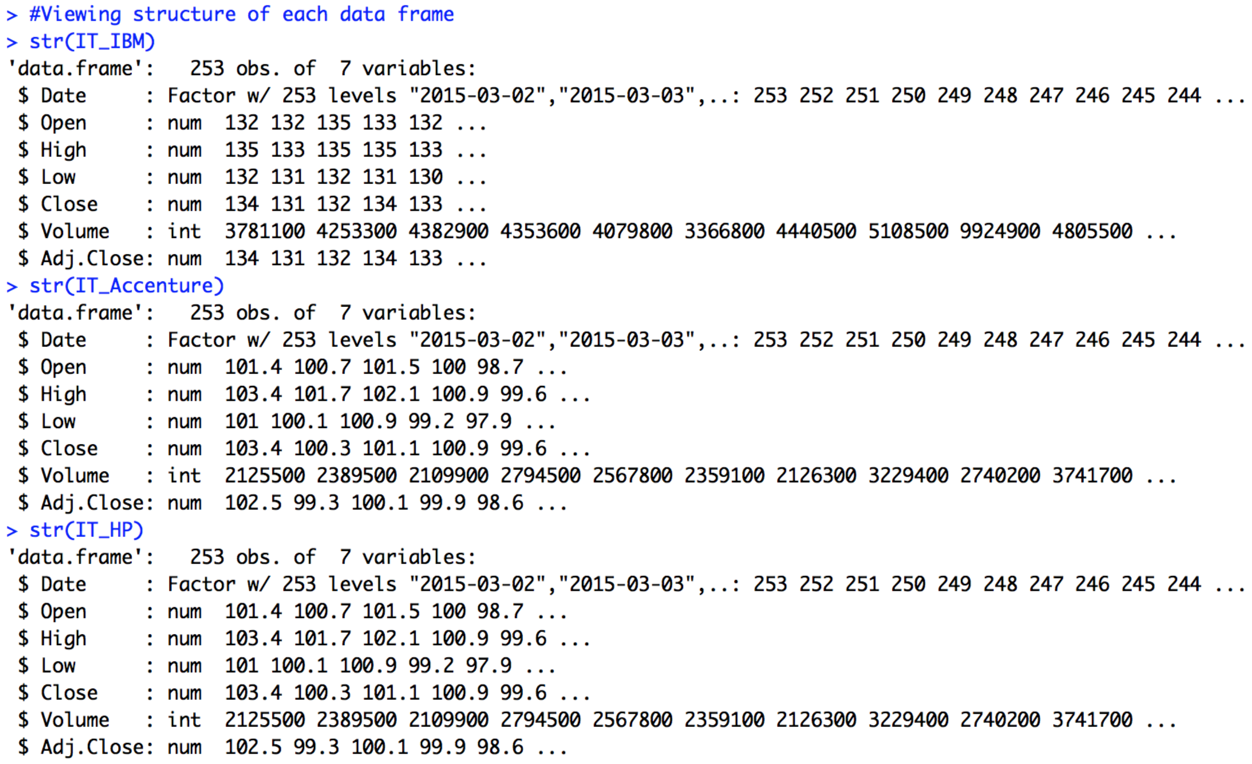
```r
#Stock Market data analysis
#Obtained stock market data from Yahoo stock market database. Daily Stock market data is taken from March 1, 2015 to March 1, 2016


#Reading each csv file

#IT Industry top 3 companies
IT_Accenture <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/accenture.csv", header=TRUE)
IT_HP <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/hp.csv", header=TRUE)
IT_IBM <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/ibm.csv", header=TRUE)



#Banking industry top 3 companies
Banking_JPMorgan <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/jpm.csv", header=TRUE)
Banking_GoldmanSachs <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/goldman.csv", header=TRUE)
Banking_Citigroup <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/citi.csv", header=TRUE)



#Internet Media industry top 3 companies
InternetMedia_Google <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/google.csv", header=TRUE)
InternetMedia_Tencent <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/tencent.csv", header=TRUE)
InternetMedia_Facebook <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/facebook.csv", header=TRUE)



#Reading Financial Constituent Dataset
financialConstituent <- read.csv("/Users/Saurabh/Documents/SYRA_DOCS/SEM2/IST687/Project/Datasets/constituents-financials.csv", header=TRUE)
```

STEP 2: VIEWING DATASET STRUCTURE

Viewing the Structure of each Dataset:

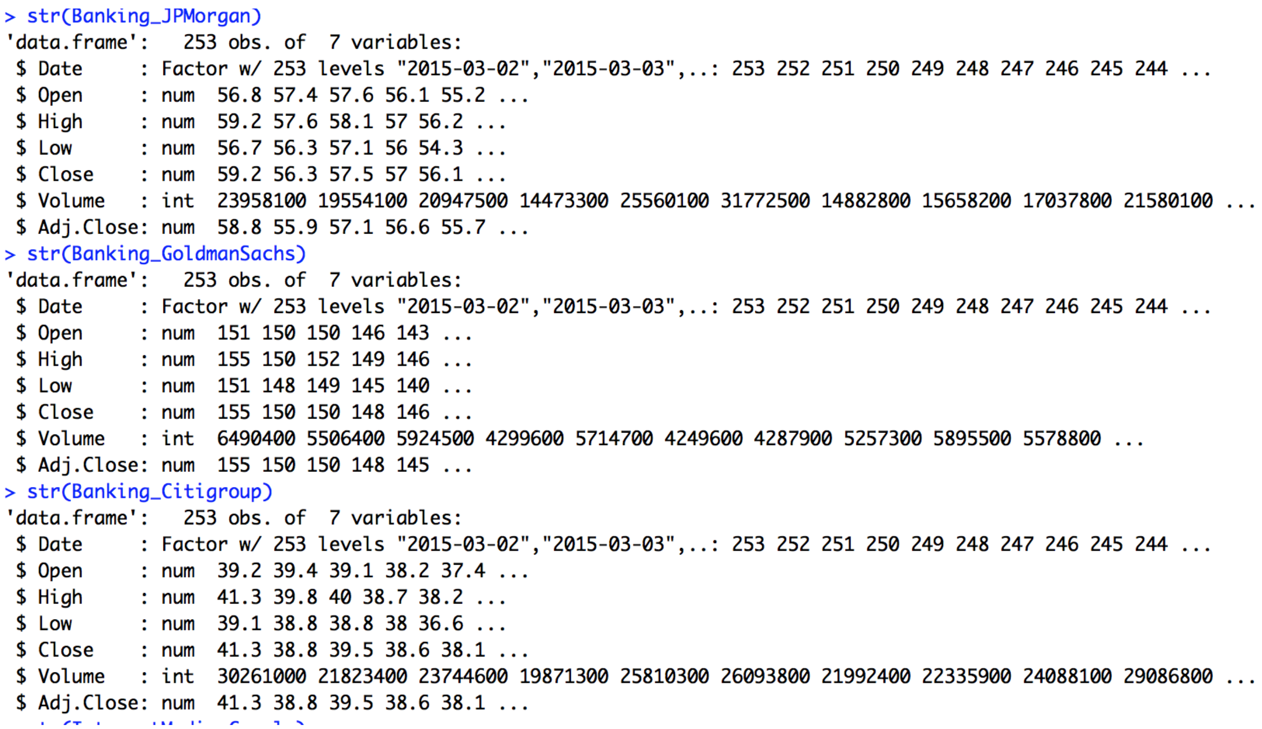
```
str(IT_IBM)
str(IT_Accenture)
str(IT_HP)

str(Banking_JPMorgan)
str(Banking_GoldmanSachs)
str(Banking_Citigroup)

str(InternetMedia_Google)
str(InternetMedia_Tencent)
str(InternetMedia_Facebook)

str(financialConstituent)
```

a) IT INDUSTRY:



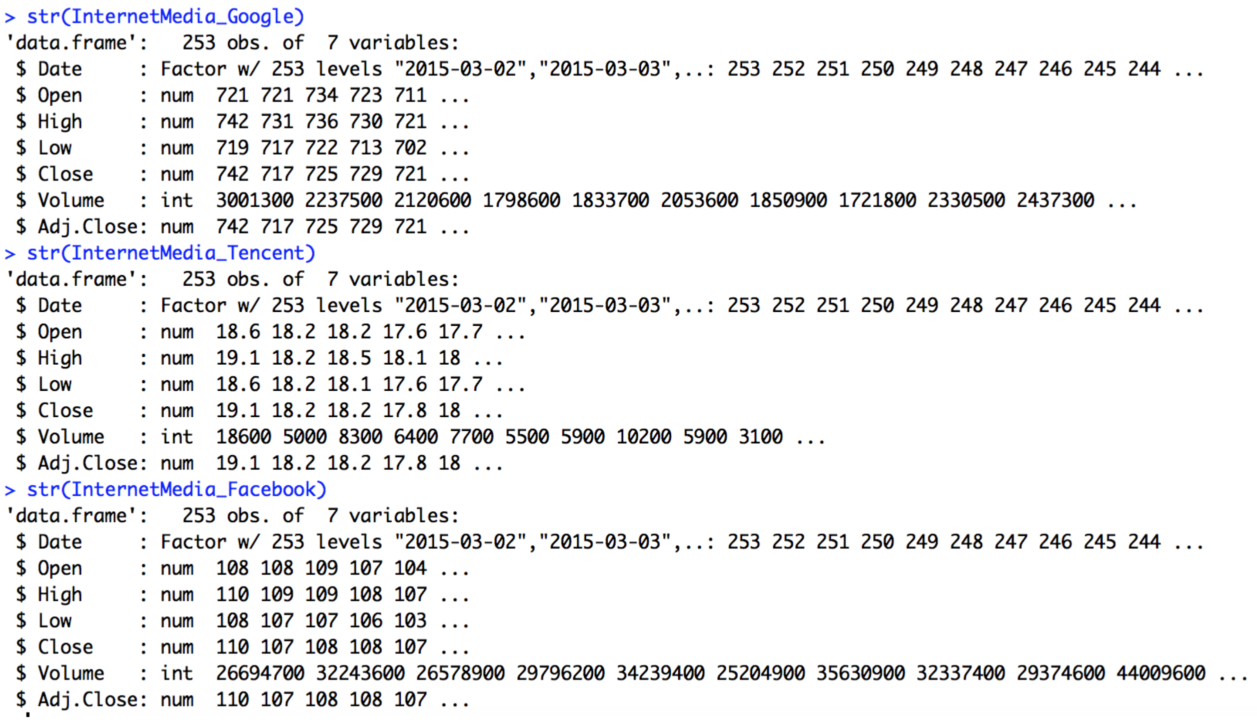
```
> #Viewing structure of each data frame
> str(IT_IBM)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  132 132 135 133 132 ...
 $ High     : num  135 133 135 135 133 ...
 $ Low      : num  132 131 132 131 130 ...
 $ Close    : num  134 131 132 134 133 ...
 $ Volume   : int  3781100 4253300 4382900 4353600 4079800 3366800 4440500 5108500 9924900 4805500 ...
 $ Adj.Close: num  134 131 132 134 133 ...
> str(IT_Accenture)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  101.4 100.7 101.5 100 98.7 ...
 $ High     : num  103.4 101.7 102.1 100.9 99.6 ...
 $ Low      : num  101 100.1 100.9 99.2 97.9 ...
 $ Close    : num  103.4 100.3 101.1 100.9 99.6 ...
 $ Volume   : int  2125500 2389500 2109900 2794500 2567800 2359100 2126300 3229400 2740200 3741700 ...
 $ Adj.Close: num  102.5 99.3 100.1 99.9 98.6 ...
> str(IT_HP)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  101.4 100.7 101.5 100 98.7 ...
 $ High     : num  103.4 101.7 102.1 100.9 99.6 ...
 $ Low      : num  101 100.1 100.9 99.2 97.9 ...
 $ Close    : num  103.4 100.3 101.1 100.9 99.6 ...
 $ Volume   : int  2125500 2389500 2109900 2794500 2567800 2359100 2126300 3229400 2740200 3741700 ...
 $ Adj.Close: num  102.5 99.3 100.1 99.9 98.6 ...
```

b) BANKING INDUSTRY



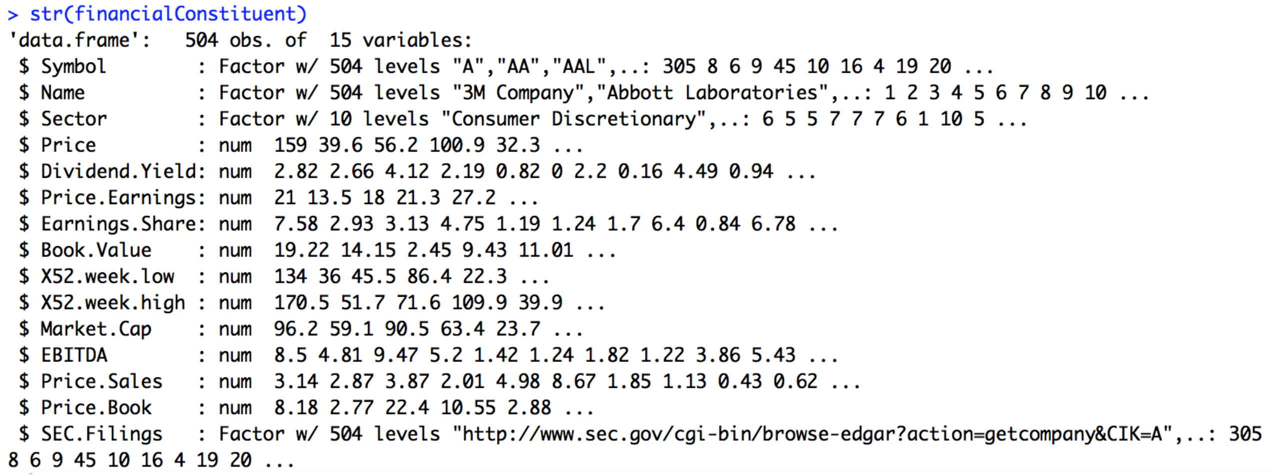
```
> str(Banking_JPMorgan)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  56.8 57.4 57.6 56.1 55.2 ...
 $ High     : num  59.2 57.6 58.1 57 56.2 ...
 $ Low      : num  56.7 56.3 57.1 56 54.3 ...
 $ Close    : num  59.2 56.3 57.5 57 56.1 ...
 $ Volume   : int  23958100 19554100 20947500 14473300 25560100 31772500 14882800 15658200 17037800 21580100 ...
 $ Adj.Close: num  58.8 55.9 57.1 56.6 55.7 ...
> str(Banking_GoldmanSachs)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  151 150 150 146 143 ...
 $ High     : num  155 150 152 149 146 ...
 $ Low      : num  151 148 149 145 140 ...
 $ Close    : num  155 150 150 148 146 ...
 $ Volume   : int  6490400 5506400 5924500 4299600 5714700 4249600 4287900 5257300 5895500 5578800 ...
 $ Adj.Close: num  155 150 150 148 145 ...
> str(Banking_Citigroup)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  39.2 39.4 39.1 38.2 37.4 ...
 $ High     : num  41.3 39.8 40 38.7 38.2 ...
 $ Low      : num  39.1 38.8 38.8 38 36.6 ...
 $ Close    : num  41.3 38.8 39.5 38.6 38.1 ...
 $ Volume   : int  30261000 21823400 23744600 19871300 25810300 26093800 21992400 22335900 24088100 29086800 ...
 $ Adj.Close: num  41.3 38.8 39.5 38.6 38.1 ...
```

c) INTERNET MEDIA INDUSTRY



```
> str(InternetMedia_Google)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  721 721 734 723 711 ...
 $ High     : num  742 731 736 730 721 ...
 $ Low      : num  719 717 722 713 702 ...
 $ Close    : num  742 717 725 729 721 ...
 $ Volume   : int  3001300 2237500 2120600 1798600 1833700 2053600 1850900 1721800 2330500 2437300 ...
 $ Adj.Close: num  742 717 725 729 721 ...
> str(InternetMedia_Tencent)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  18.6 18.2 18.2 17.6 17.7 ...
 $ High     : num  19.1 18.2 18.5 18.1 18 ...
 $ Low      : num  18.6 18.2 18.1 17.6 17.7 ...
 $ Close    : num  19.1 18.2 18.2 17.8 18 ...
 $ Volume   : int  18600 5000 8300 6400 7700 5500 5900 10200 5900 3100 ...
 $ Adj.Close: num  19.1 18.2 18.2 17.8 18 ...
> str(InternetMedia_Facebook)
'data.frame':	253 obs. of  7 variables:
 $ Date     : Factor w/ 253 levels "2015-03-02","2015-03-03",..: 253 252 251 250 249 248 247 246 245 244 ...
 $ Open     : num  108 108 109 107 104 ...
 $ High     : num  110 109 109 108 107 ...
 $ Low      : num  108 107 107 106 103 ...
 $ Close    : num  110 107 108 108 107 ...
 $ Volume   : int  26694700 32243600 26578900 29796200 34239400 25204900 35630900 32337400 29374600 44009600 ...
 $ Adj.Close: num  110 107 108 108 107 ...
```

d) FINANCIAL CONSTITUENT DATASET



```
> str(financialConstituent)
'data.frame':	504 obs. of  15 variables:
 $ Symbol        : Factor w/ 504 levels "A","AA","AAL",..: 305 8 6 9 45 10 16 4 19 20 ...
 $ Name          : Factor w/ 504 levels "3M Company","Abbott Laboratories",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ Sector        : Factor w/ 10 levels "Consumer Discretionary",..: 6 5 5 7 7 7 6 1 10 5 ...
 $ Price         : num  159 39.6 56.2 100.9 32.3 ...
 $ Dividend.Yield: num  2.82 2.66 4.12 2.19 0.82 0 2.2 0.16 4.49 0.94 ...
 $ Price.Earnings: num  21 13.5 18 21.3 27.2 ...
 $ Earnings.Share: num  7.58 2.93 3.13 4.75 1.19 1.24 1.7 6.4 0.84 6.78 ...
 $ Book.Value    : num  19.22 14.15 2.45 9.43 11.01 ...
 $ X52.week.low  : num  134 36 45.5 86.4 22.3 ...
 $ X52.week.high : num  170.5 51.7 71.6 109.9 39.9 ...
 $ Market.Cap    : num  96.2 59.1 90.5 63.4 23.7 ...
 $ EBITDA        : num  8.5 4.81 9.47 5.2 1.42 1.24 1.82 1.22 3.86 5.43 ...
 $ Price.Sales   : num  3.14 2.87 3.87 2.01 4.98 8.67 1.85 1.13 0.43 0.62 ...
 $ Price.Book    : num  8.18 2.77 22.4 10.55 2.88 ...
 $ SEC.Filings   : Factor w/ 504 levels "http://www.sec.gov/cgi-bin/browse-edgar?action=getcompany&CIK=A",..: 305
8 6 9 45 10 16 4 19 20 ...
```

STEP 3: INSTALLING PACKAGES

```
#STEP 3: INSTALLING PACKAGES REQUIRED FOR THE ENTIRE PROJECT

install.packages("dygraphs")
install.packages('data.table')
install.packages("forecast")
install.packages("xts")
install.packages("caret")
install.packages("flexclust")
install.packages("caTools")
```

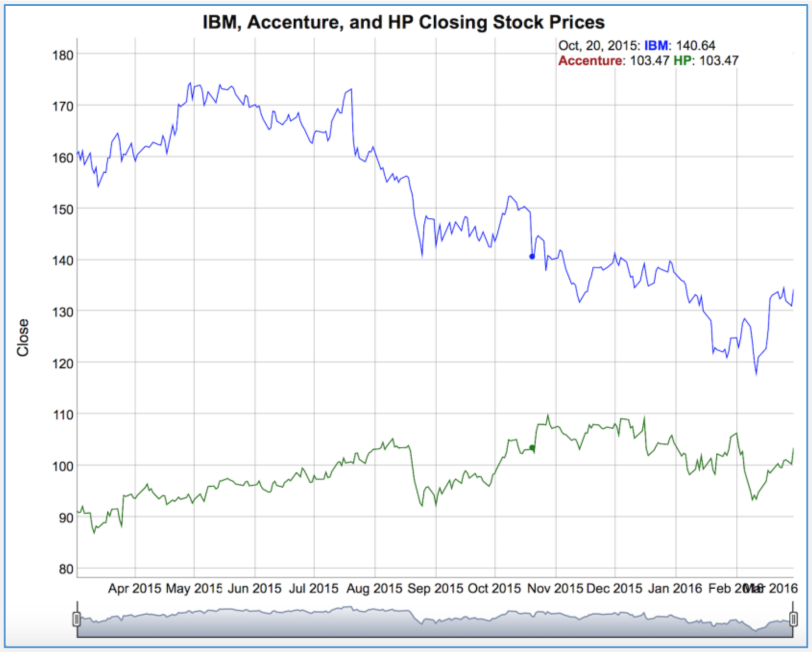
STEP 4: LOADING PACKAGES

```
#STEP 4: LOADING PACKAGES REQUIRED FOR THE ENTIRE PROJECT

library(data.table)
library(dygraphs)
library(xts)
library(forecast)
library(caret)
library(flexclust)
library(caTools)
```

STEP 5: PLOTTING TIME SERIES GRAPHS OF TOP 3 COMPANIES OF EACH INDUSTRY

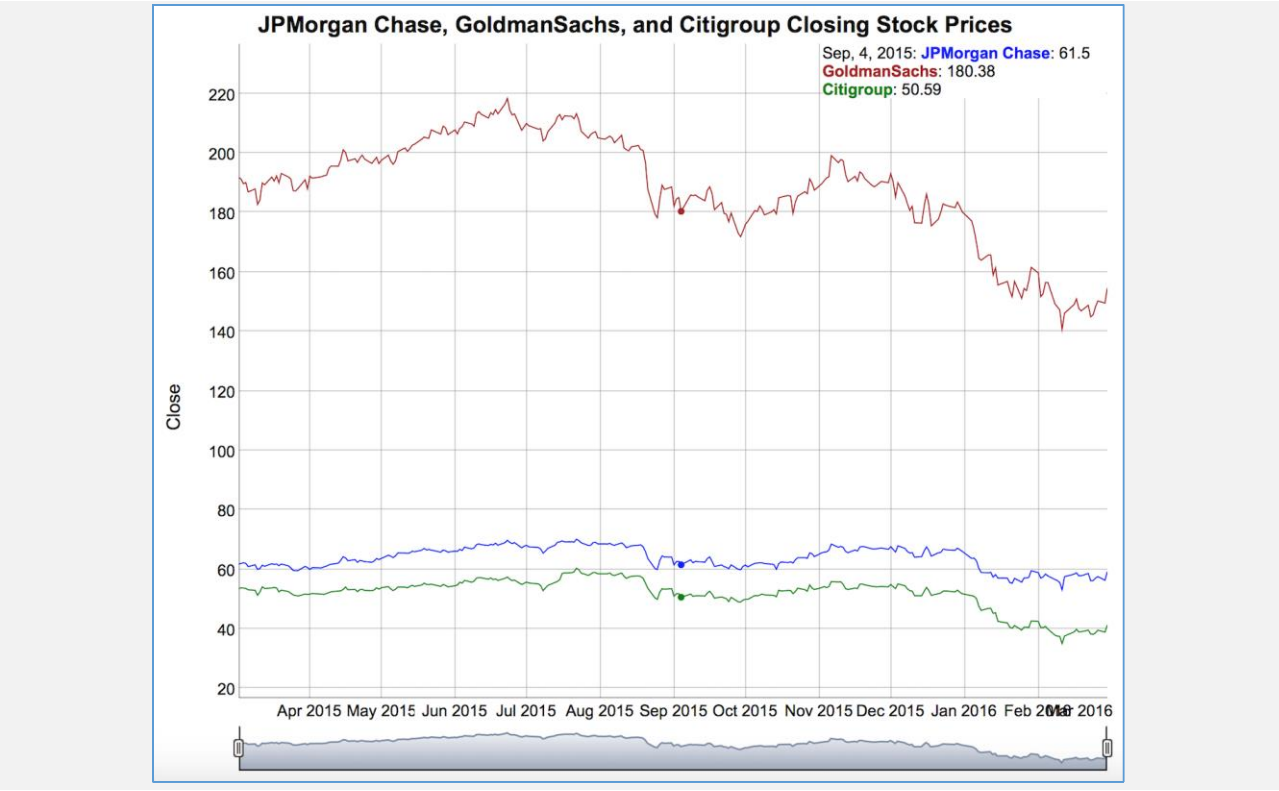
A) IT INDUSTRY

```
# Plotting Time Series of Top-3 companies from each industry
# dygraph() needs xts time series objects for IT industry
ibm_xts <- xts(IT_IBM$Close,order.by=as.POSIXct(IT_IBM$Date),frequency=365)
accen_xts <- xts(IT_Accenture$Close,order.by=as.POSIXct(IT_Accenture$Date),frequency=365)
hp_xts <- xts(IT_HP$Close,order.by=as.POSIXct(IT_HP$Date),frequency=365)
# Creating a new vector for IT industry
stocksITService <- cbind(ibm_xts,accen_xts,hp_xts)
#Creating IT Sector dygraph
dygraph(stocksITService,ylab="Close",
main="IBM, Accenture, and HP Closing Stock Prices") %>%
        dySeries("..1",label="IBM") %>%
        dySeries("..2",label="Accenture") %>%
        dySeries("..3",label="HP") %>%
        dyOptions(colors = c("blue","brown","green")) %>%
        dyRangeSelector()
```



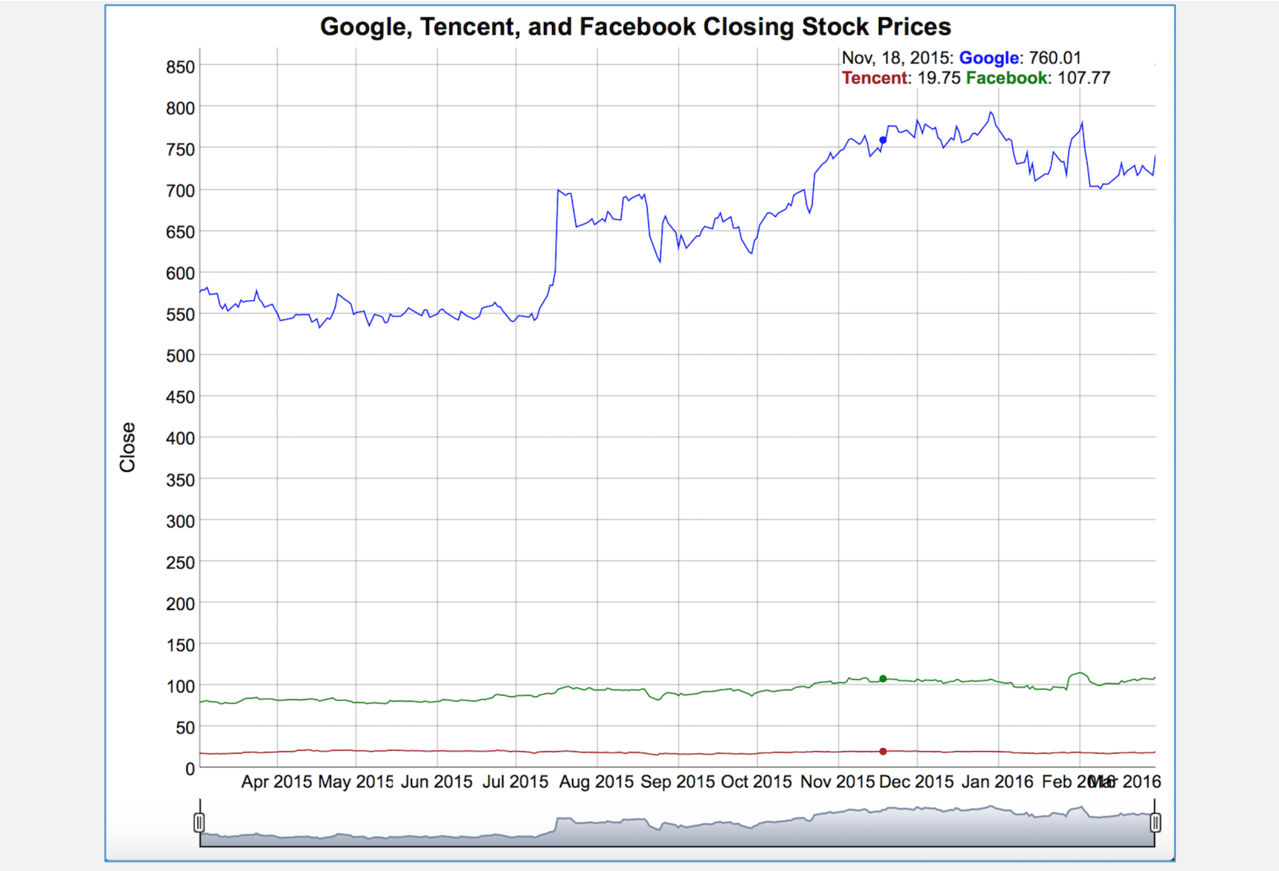
B) BANKING INDUSTRY

```
# dygraph() needs xts time series objects for Banking industry

JpMorgan_xts <- xts(Banking_JPMorgan$Close,order.by=as.POSIXct(Banking_JPMorgan$Date),frequency=365)

GoldmanSachs_xts <- xts(Banking_GoldmanSachs$Close,order.by=as.POSIXct(Banking_GoldmanSachs$Date),frequency=365)

Citigroup_xts <- xts(Banking_Citigroup$Close,order.by=as.POSIXct(Banking_Citigroup$Date),frequency=365)

# Creating a new vector for Banking industry

stocksBanking <- cbind(JpMorgan_xts,GoldmanSachs_xts,Citigroup_xts)

dygraph(stocksBanking,ylab="Close",

    main="JPMorgan Chase, GoldmanSachs, and Citigroup Closing Stock Prices") %>%

        dySeries("..1",label="JPMorgan Chase") %>%

         dySeries("..2",label="GoldmanSachs") %>%

         dySeries("..3",label="Citigroup") %>%

         dyOptions(colors = c("blue","brown","green")) %>%

         dyRangeSelector()
```



C) INTERNET MEDIA INDUSTRY

```
# dygraph() needs xts time series objects for Banking industry

Google_xts <- xts(InternetMedia_Google$Close,order.by=as.POSIXct(InternetMedia_Google$Date),frequency=365)

Tencent_xts <-xts(InternetMedia_Tencent$Close,order.by=as.POSIXct(InternetMedia_Tencent$Date),frequency=365)

Facebook_xts <-
xts(InternetMedia_Facebook$Close,order.by=as.POSIXct(InternetMedia_Facebook$Date),frequency=365)

stocksInternetMedia <- cbind(Google_xts,Tencent_xts,Facebook_xts)

dygraph(stocksInternetMedia,ylab="Close",

                main="Google, Tencent, and Facebook Closing Stock Prices") %>%

           dySeries("..1",label="Google") %>%

           dySeries("..2",label="Tencent") %>%

           dySeries("..3",label="Facebook") %>%

           dyOptions(colors = c("blue","brown","green")) %>%

           dyRangeSelector()
```

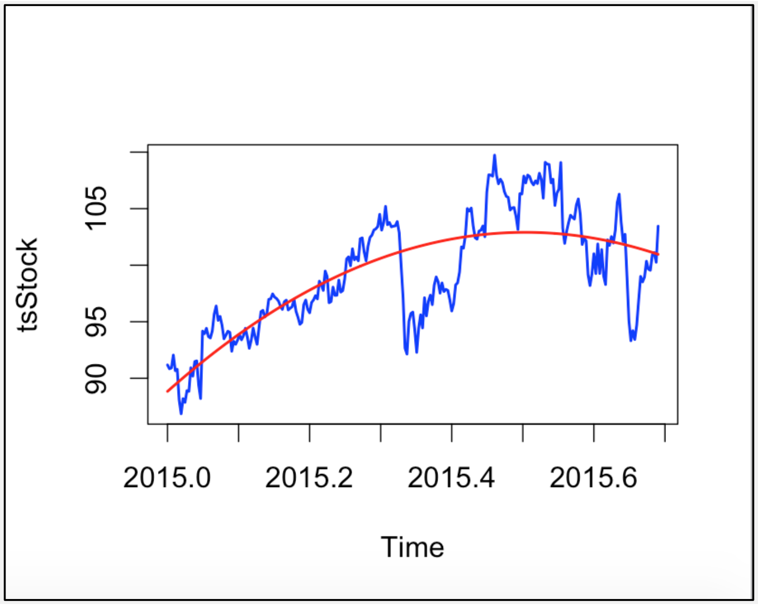


STEP 6: PREDICTING COMPANY STOCK PRICE USING DIFFERENT MODELS

A) POLYNOMIAL MODEL

Polynomial regression is used here to fit to stock market data, and model the non-linear relationship for predicting future values. The below function is used for polynomial regression.

```
#Function to calculate time series of polynomial trend

polynomial <- function(stockData) {

  train <- window(stockData, end=c(2015, 190))

  tl = seq(1,253,length=length(train))

  tl2=tl^7

  str(stockData)

  polyStock = lm(train ~tl+tl2)

  tsStocktrend1=ts(polyStock$fit,start=c(2015,1),frequency = 365)

  plot(tsStock,lw=2,col='blue')

  lines(tsStocktrend1,lw=2,col='red')

  return(tsStocktrend1)

}
```

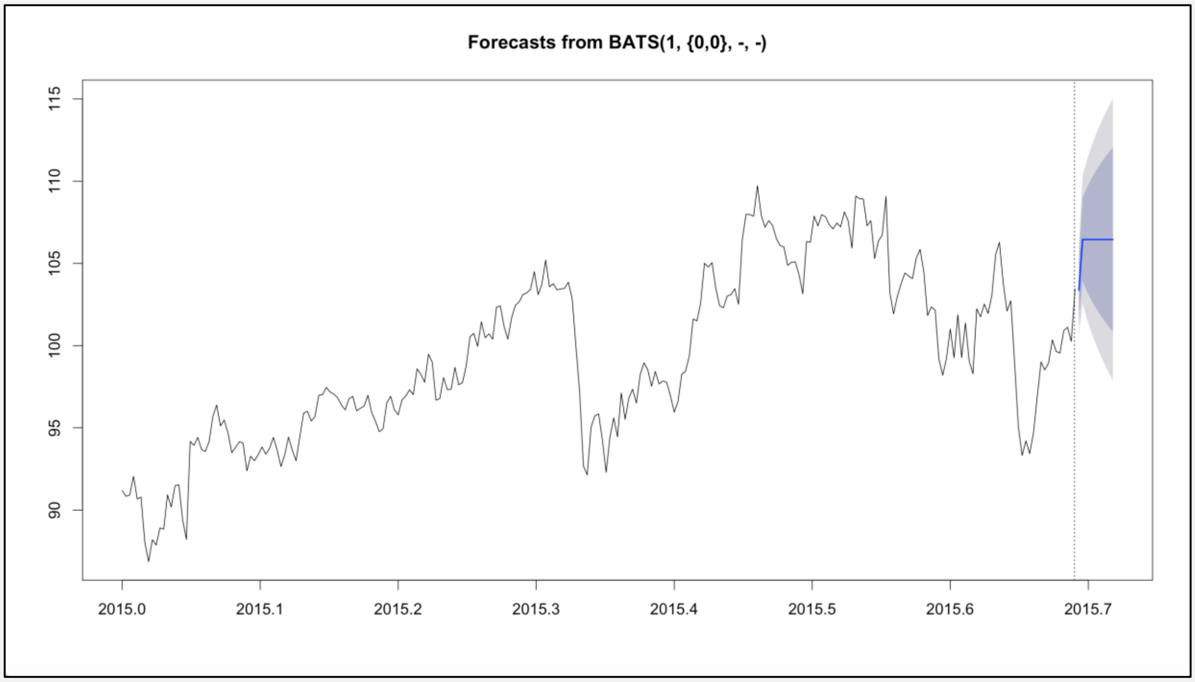


***Example**: The graph above shows the use of Polynomial model to calculate time series of polynomial trend*

B) TBATS MODEL

Since we have one year or more data for each stock on a daily basis we also went with weekly seasonality model calculation using multiple seasonal model i.e. TBATS. This model is designed for use when there are multiple cyclic patterns (e.g. daily, weekly and yearly patterns) in a single time series. It may detect complicated patterns in our time series.

```
#Forecasting using multiple seasonal model such as TBATS

stlStock <- msts(rev(accenture$Close), start=c(2015),seasonal.periods=c(7,365.25))

stocktrend2 <- tbats(stlStock)

plot(forecast(stocktrend2))

tsStocktrend2 = stlStock

abline(v=2015.69,lty=3)
```



***Example**: The graph above shows the use of TBATS model to forecast stock value*

C) ARIMA MODEL

ARIMA model combines unit root tests, minimization of AICc and MLE to obtain an ARIMA model. It processes enough elements for regression and averages, it fits an approximation to almost any time series.

```
#Function to calculate arima model and return total error
arimaModel <- function(polyTrend, stockData){
  fitArima<-auto.arima(polyTrend)
  plot(forecast(fitArima,h=50),xlim=c(2015,2016.2),ylim=c(90,120), lw=2, col="red", xlab="Time", ylab="Stock Price", main="Predictions of the Polynomial trend")
  test <- window(stockData, start=c(2015,191))
  predfitr =  window(forecast(fitArima,h=39)$mean, start=c(2015,191))
  mae = matrix(NA,25,length(test)+1)
  for(i in 1:length(test))
  {
    mae[1,i]<-abs(predfitr[i]-test[i])
  }
  mae[1,]
  return(sum(mae[1,], na.rm = TRUE))
}
```

D) MONTE CARLO MODEL

Monte Carlo method is significantly used for predicting future values as it considers a wide range of possibilities, and helps reduce uncertainty. Because of its advantages it is quite widely used across finance industries for portfolio management, asset management etc. Hence, we decided to use the same for predicting stock data.

```
################# MONTE CARLO PREDICTION ################

montCarlo<-function(stockData){

  #Dividing Dataset into Test and Train Data

  trainDataStockMC <-stockData[1:(floor(0.75*nrow(stockData))),]

  testDataStockMC <- stockData[(ceiling(0.75*nrow(stockData))):nrow(stockData),]


  #Finding Periodic Daily Return

setDT(trainDataStockMC)[,PeriodicDailyRetun:=log(trainDataStockMC$Adj.Close/shift(trainDataStockMC$Adj.Close,1,type="lead"))]

  trainDataStockMC$PeriodicDailyRetun[is.na(trainDataStockMC$PeriodicDailyRetun)] <- 0

  averageStock <- mean(trainDataStockMC$PeriodicDailyRetun)

  varianceStock<-var(trainDataStockMC$PeriodicDailyRetun)

  stdDeviationStock<- sd(trainDataStockMC$PeriodicDailyRetun)


  #Calculating Drift

  StockDrift<-averageStock-(varianceStock/2)

  #Setting Seed for Random Variable in Drift

  set.seed(123)


  previousDayPriceMC=trainDataStockMC$Adj.Close[nrow(trainDataStockMC)]

  predictedList=c()
```

```
 for(i in 1:nrow(testDataStockMC)){
  futurePriceMC=previousDayPriceMC*exp(StockDrift+stdDeviationStock*(qnorm(runif(1))))
  previousDayPriceMC= futurePriceMC
  predictedList[i]<-previousDayPriceMC
 }

 #For loop to calculate Total error
 errorSum<-0
 for(i in 1:nrow(testDataStockMC)){
  errorSum = errorSum + abs( predictedList[i]-testDataStockMC$Adj.Close[i])
 }
 return(errorSum)
}
```

E) CLUSTER THEN PREDICT MODEL

Cluster-then-predict model we first cluster observations and then build cluster-specific prediction models. We use this model to identify the trends of the stocks. The stocks which have more than 80% probability of having a future positive increase will be considered to be top 3 stocks. This function below calculates the sum of all future values with more than 80% of probability. The stocks that have highest sum have positive trends. Top 3 stocks will be selected for further analysis.

```
#Cluster-Then-Predict Function to identify trends in stocks


clusterPredict<-function(stockData){
  stockData<-dichotomousCalc(stockData)


  spl <- sample.split(stockData$PositiveChange, SplitRatio = 0.75)
  stocksTrain <- subset(stockData, spl == T)
  stocksTest <- subset(stockData, spl == F)


  limitedTrain <- stocksTrain
  limitedTrain$PositiveChange <- NULL
  limitedTest <- stocksTest
  limitedTest$PositiveChange <- NULL


  preproc <- preProcess(limitedTrain)
  str(preproc)
  normTrain <- predict(preproc, limitedTrain)
  normTest <- predict(preproc, limitedTest)


  summary(normTest)
```

```
set.seed(144)
km <- kmeans(normTrain[,-1], centers = 2)
str(km)

# CLUSTERING STOCKS
# test-set observations assigned to Cluster 2
#install.packages("flexclust")
library(flexclust)
km.kcca <- as.kcca(km, normTrain[,-1])
clusterTrain <- predict(km.kcca)
clusterTest <- predict(km.kcca, newdata = normTest[,-1])
table(clusterTest)
length(clusterTrain)
length(stocksTrain)
stockTrain1 <- subset(stocksTrain, clusterTrain == 1)
stockTrain2 <- subset(stocksTrain, clusterTrain == 2)
#stockTrain3 <- subset(stocksTrain, clusterTrain == 3)

stockTest1 <- subset(stocksTest, clusterTest == 1)
stockTest2 <- subset(stocksTest, clusterTest == 2)
#stockTest3 <- subset(stocksTest, clusterTest == 3)

#Training set data frame that has the highest average value of the dependent variable
tapply(stocksTrain$PositiveChange, clusterTrain, mean)
```

```
  #CLUSTER-SPECIFIC PREDICTIONS: Building logistic regression models
  stocksModel1 <- glm(PositiveChange ~ Open+High+Low+Close+Volume+Adj.Close, data = stockTrain1,
family = binomial)
  stocksModel2 <- glm(PositiveChange ~Open+High+Low+Close+Volume+Adj.Close, data = stockTrain2,
family = binomial)
  stocksModel3 <- glm(PositiveChange ~Open+High+Low+Close+Volume+Adj.Close, data = stockTrain3,
family = binomial)


  #Using StocksModel, make test-set predictions called PredictTest on the data frame stocksTest
  predictTest1 <- predict(stocksModel1, newdata = stockTest1, type = "response")
  predictTest2 <- predict(stocksModel2, newdata = stockTest2, type = "response")
  predictTest3 <- predict(stocksModel3, newdata = stockTest3, type = "response")


  allPredictions <- c(predictTest1, predictTest2)
  allOutcomes <- c(stockTest1$PositiveChange, stockTest2$PositiveChange)


  #Calculate number of predictions with more than 0.8 probability
  predictCount1 = sum(predictTest1>=0.8)
  predictCount2 = sum(predictTest2>=0.8)
  predictCount3 = sum(predictTest3>=0.8)


  return(predictCount1+predictCount2+predictCount3)
}


dichotomousCalc<-function(stockData){
  stockData$PositiveChange[1] <- T
  for (i in 2:nrow(stockData)) {
    if (stockData$Close[i] > stockData$Close[i-1] ) {
      stockData$PositiveChange[i] <- T
    } else {
      stockData$PositiveChange[i] <- F
    }
  }
  return(stockData)
}
```

F) ARIMA PREDICTION MODEL

ARIMA model is used to calculate/forecast future values for top 3 stocks. This function returns the future values for top 3 stocks.

```
########### ARIMA MODEL PREDICTION #################
arimaData<-function(stockData){


  #Converting date format
  rdate<-as.Date(stockData$Date, "%m%d%y")


  #Converting stock data into time series
  tsStock = ts(rev(stockData$Close),start=c(2015,1),frequency = 365)


  #polynomial trend generation
  tl = seq(1,253,length=length(tsStock))
  tl2=tl^7
  polyStock = lm(tsStock ~tl+tl2)


  tsStockPolyTrend=ts(polyStock$fit,start=c(2015,1),frequency = 365)


  #Auto arima function calculation and forecasting future values
  fitArima<-auto.arima(tsStockPolyTrend)
  return(forecast(fitArima, h=1)$mean)
}
```

STEP 7: ANALYZING WHICH MODEL IS BETTER?

Explanation

First we assign two variables with values of character type. "allStocks" contains names of all vectors which contains stock data. "nameOfTopStocks" contain names of stock data picked from each industry.

Code:

```
###########################################################################
### STEP 7: ANALYZING WHICH MODEL IS BETTER MONTE CARLO OR ARIMA ####
###########################################################################
#allStocks vector contains vector names of all stocks
allStocks<-
c("IT_Accenture","IT_IBM","IT_HP","Banking_Citigroup","Banking_GoldmanSachs","Banking_JPMorga
n","InternetMedia_Facebook","InternetMedia_Tencent","InternetMedia_Google")

#namesOfTopStocks contains vector names of all top stocks from top 3 industries
namesOfTopStocks<-c("IT_Accenture","Banking_JPMorgan","InternetMedia_Facebook")
```

Evaluating accuracy of ARIMA, MonteCarlo, and Cluster-Then-Predict on top company of each industry

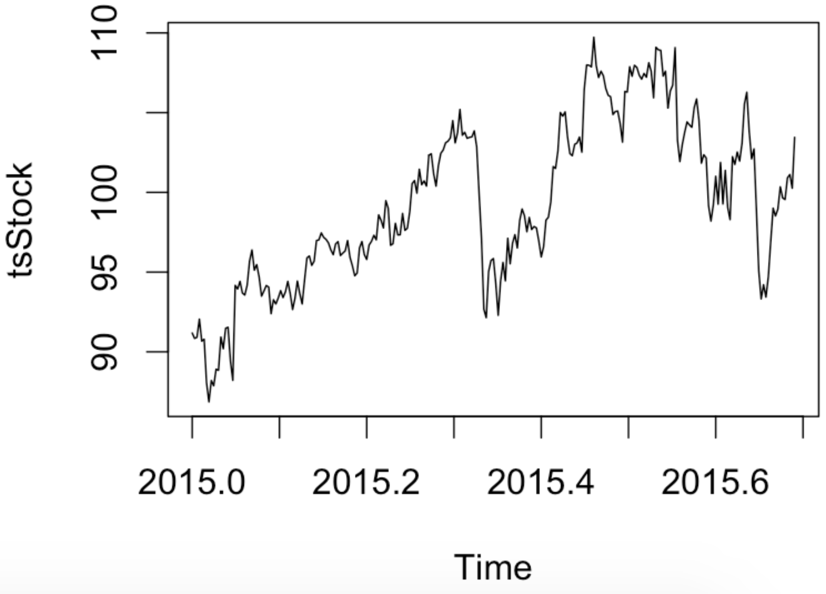
In the below for loop, we calculate sum of errors of 3 randomly picked stocks, one from each industry, using ARIMA and Monte Carlo Analysis. The model that gives least error will be used for further processing.

```
#Decalring an Array errorTable for having data in them
errorTable = array(NA,dim=c(3,2))
j = 1
```

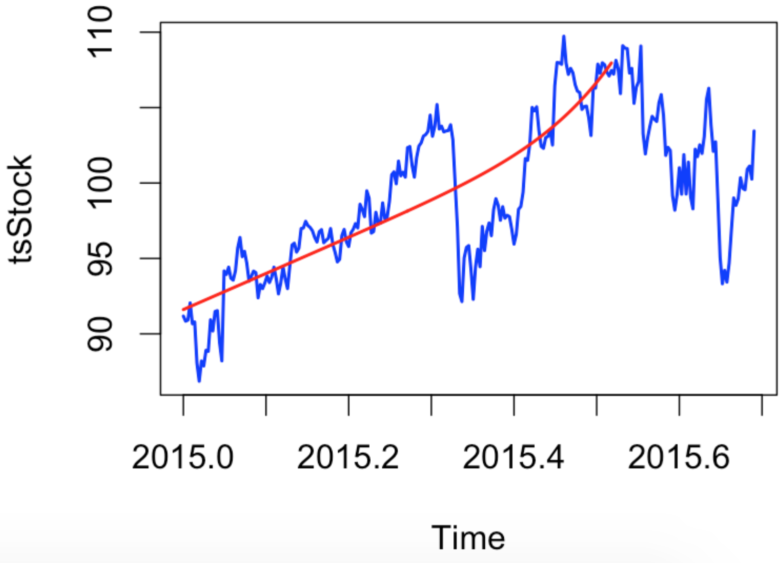
```
#For loop for iterating over top stock of each industry
for (i in namesOfTopStocks){
  #Converting date format
  rdate<-as.Date(get(i)$Date, "%m%d%y")

  #Converting stock data into time series
  tsStock = ts(rev(get(i)$Close),start=c(2015,1),frequency = 365)
  #General Plot of each stock
  plot(tsStock)

  #Generating polynomial trend of each stock
  polytrend<-polynomial(tsStock)

#Appending ARIMA error values of each stock one-by-one
  errorTable[j,1] =  arimaModel(polytrend, tsStock)
  #append(arimaError, arimaModel(polytrend, tsStock), after = length(arimaError))


  #Appending MonteCarlo error values of each stock one-by-one
  errorTable[j,2] = montCarlo(get(i))
  j = j+1
}
```

STOCK 1: IT_Accenture

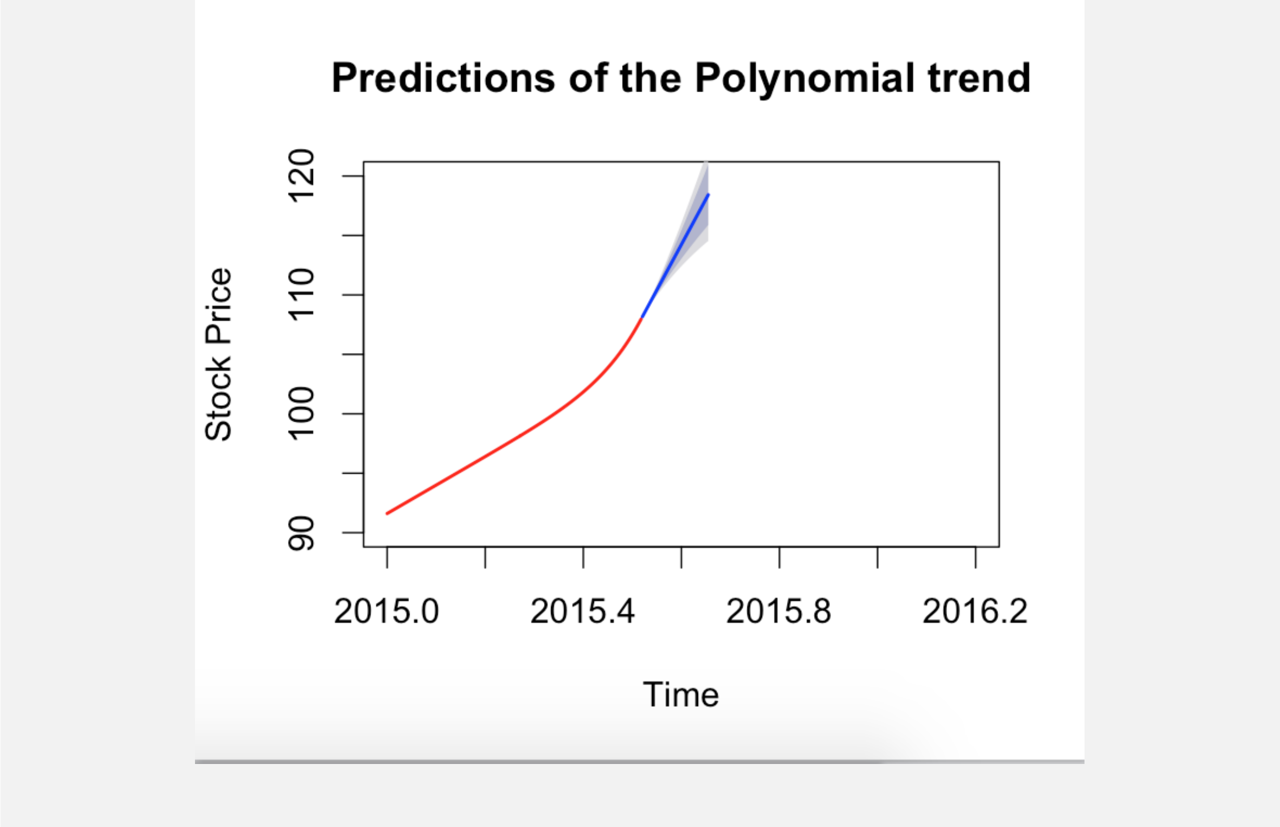
Time Series Model



Polynomial Model:

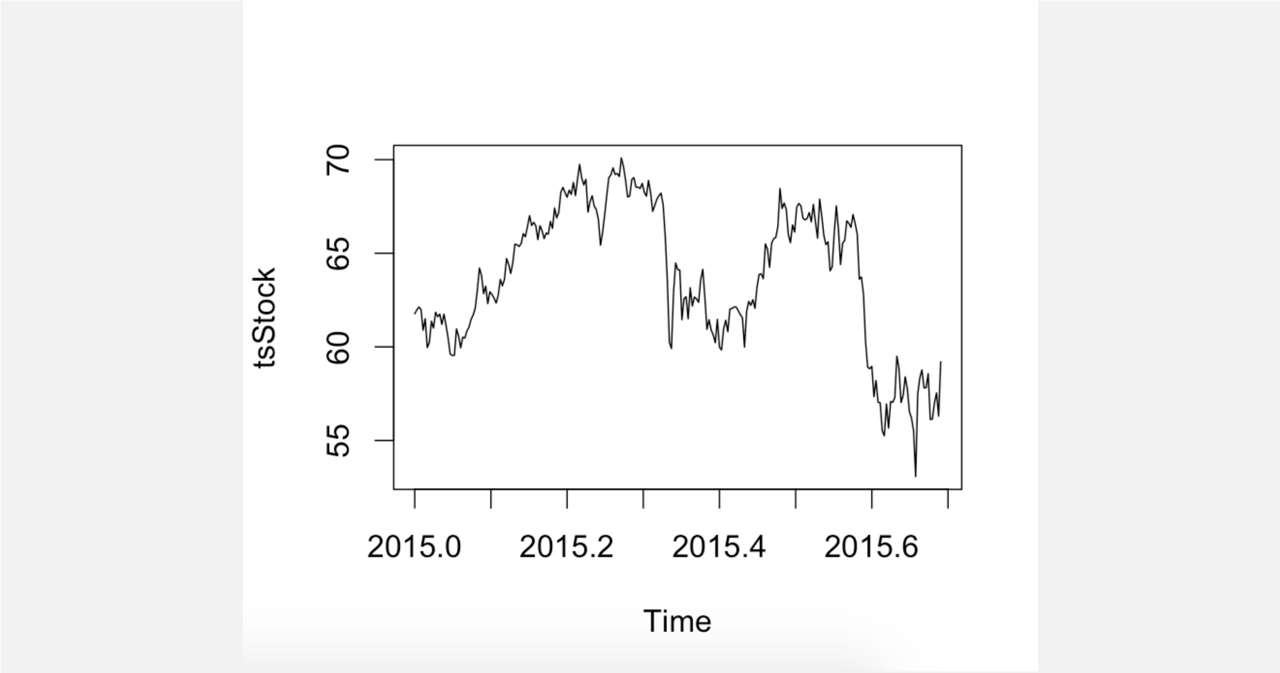


Prediction using Polynomial Model:

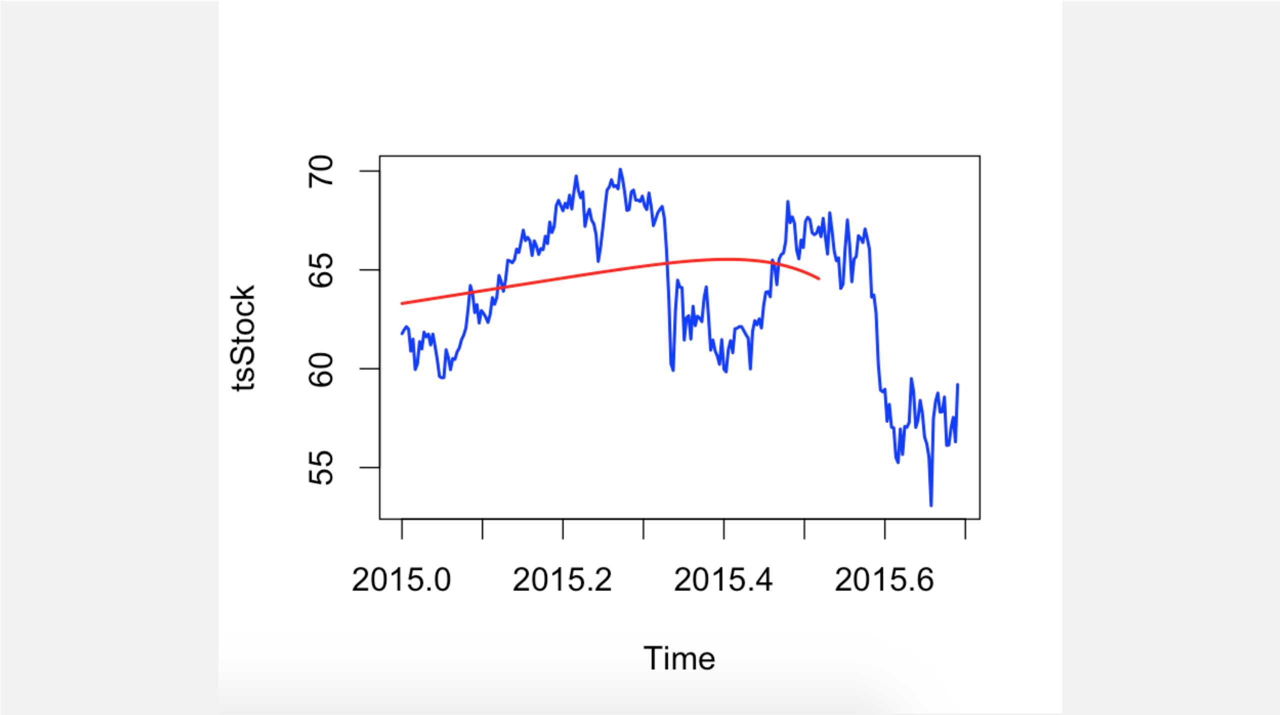


STOCK 2: Banking_JPMorgan

Time Series Model

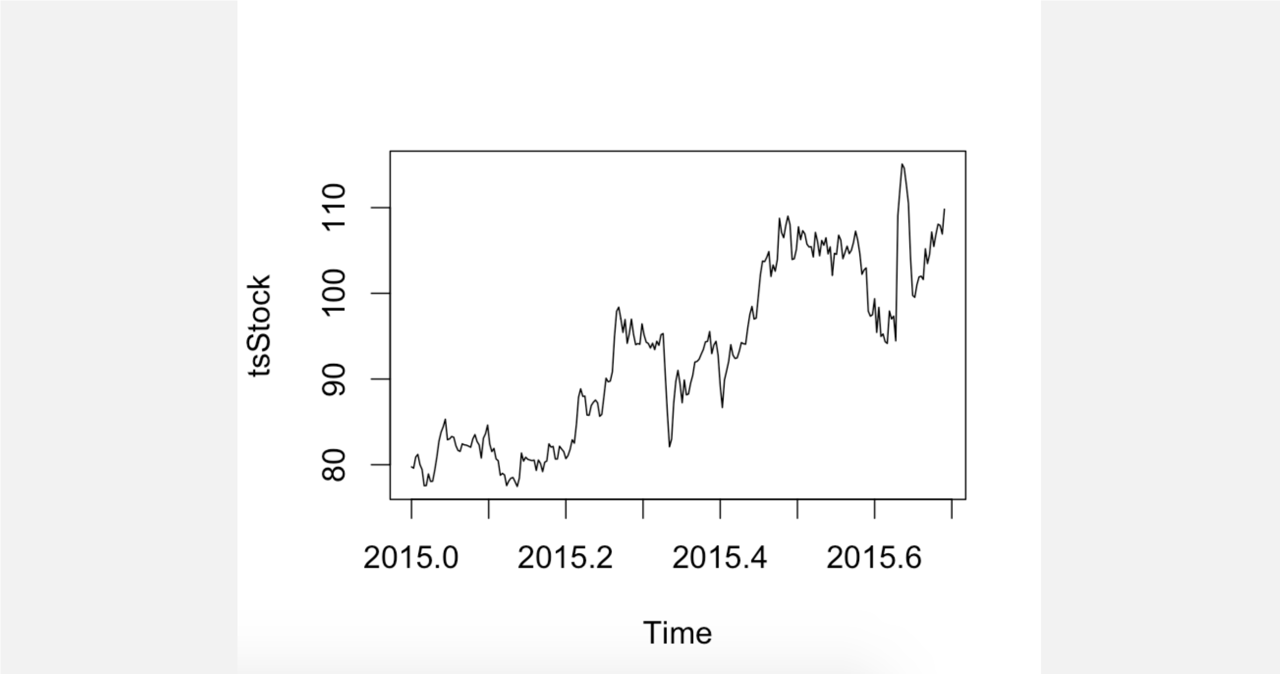


Polynomial Model:

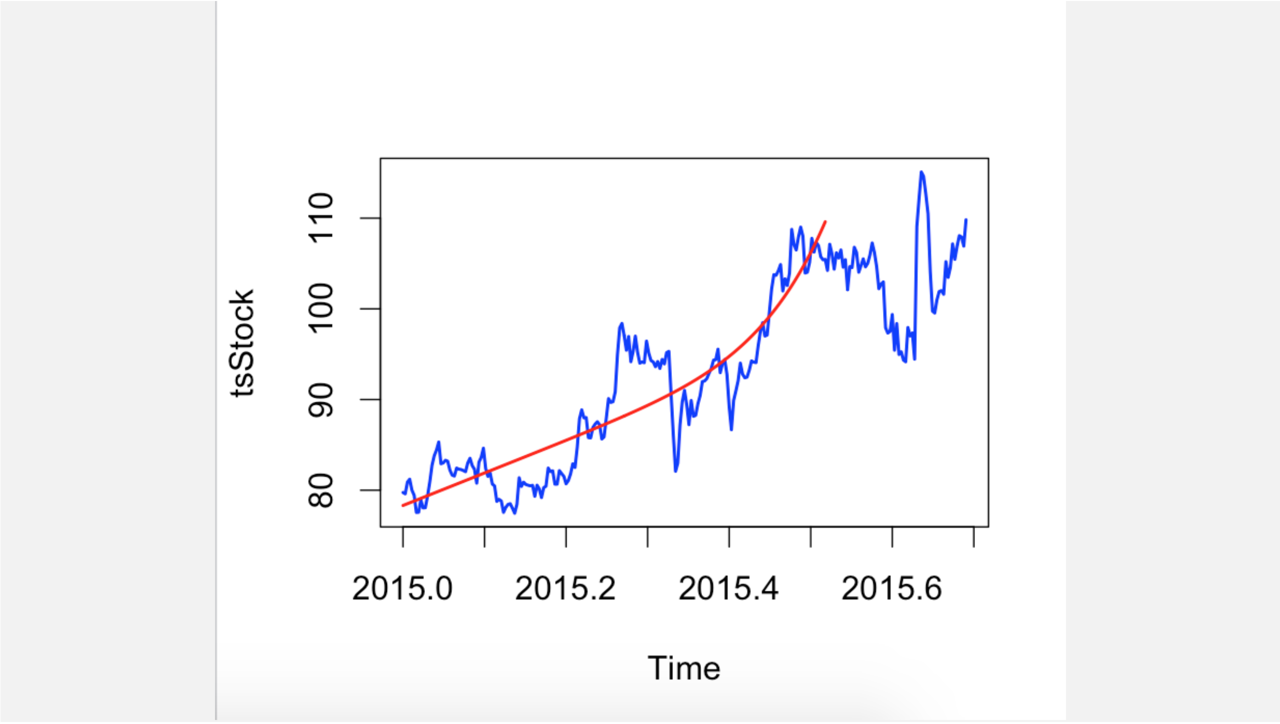


STOCK 3: InternetMedia_Facebook

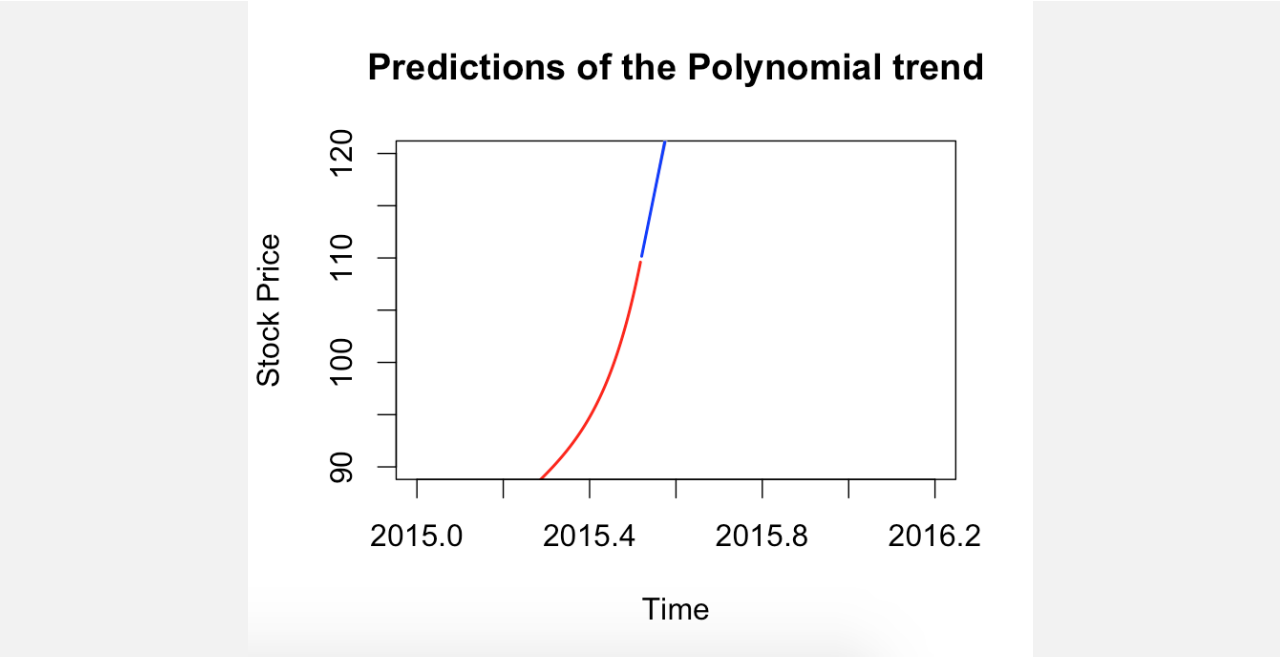
Time Series Model



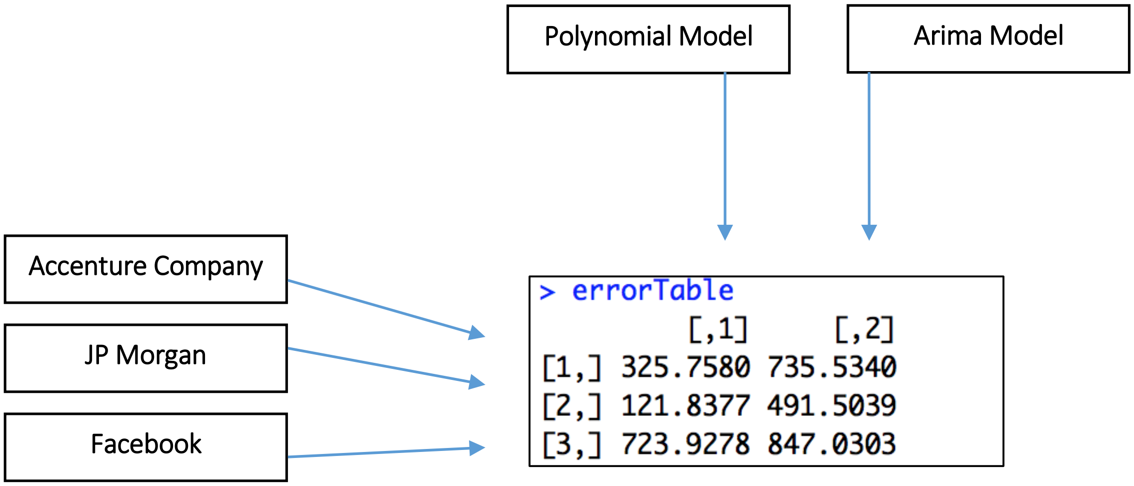
Polynomial Model:



Prediction using Polynomial Model:



Here, the Rows represent the 3 companies while the Columns represent error values using Polynomial Model and Arima Model



```
> errorTable
         [,1]     [,2]
[1,] 325.7580 735.5340
[2,] 121.8377 491.5039
[3,] 723.9278 847.0303
```

Thus we can conclude that ARIMA is better than Monte-Carlo model for prediction

MAJOR DATA QUESTION 1 ANSWERED

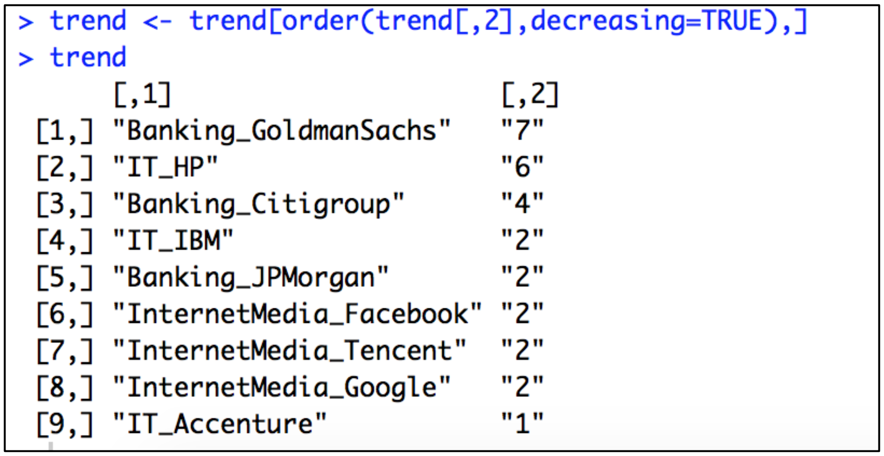
Thus, we have concluded that ARIMA model is the Best Model used for Stock Price Prediction from ARIMA model and Monte Carlo model

STEP 8: Evaluating Stock using Cluster-Predict Method

```
##########################################################################
############ STEP 8: Evaluating Stock using Cluster-Predict Method ############
##########################################################################
```

In the below for loop, we calculate trends of all stocks using Cluster-then-Predict, and sort the trends and corresponding stock names.

```
#Calculating trends of all stocks
trend = matrix(NA,nrow = 9, ncol = 2)
k = 1
#For loop for iterating over each stock
for (i in allStocks){
 #Name of the stock stored in matrix
 trend[k,1] = allStocks[k]
 #Trend calculation, one which has maximum number of increases or positivity in their trends
 trend[k,2] = clusterPredict(get(i))
 k = k+1
}
trend <- trend[order(trend[,2],decreasing=TRUE),]
```



```
> trend <- trend[order(trend[,2],decreasing=TRUE),]
> trend
      [,1]                       [,2]
 [1,] "Banking_GoldmanSachs"    "7"
 [2,] "IT_HP"                   "6"
 [3,] "Banking_Citigroup"       "4"
 [4,] "IT_IBM"                  "2"
 [5,] "Banking_JPMorgan"        "2"
 [6,] "InternetMedia_Facebook"  "2"
 [7,] "InternetMedia_Tencent"   "2"
 [8,] "InternetMedia_Google"    "2"
 [9,] "IT_Accenture"            "1"
```

*Thus, as shown from the above table***, Goldman Sachs, HP and Citigroup** *have maximum number of increases in their stock prices using cluster prediction value accuracy to be greater than 0.8.*

MAJOR DATA QUESTION 2 ANSWERED

- ***Thus, we have predicted the stock price using Cluster-Then-Predict to predict future stock RISE or FALL using historical stock Data***
- ***Goldman Sachs, HP and Citigroup are the 3 top performing stocks.***

MAJOR DATA QUESTION 3 ANSWERED

- ***Goldman Sachs is performing best in Banking Industry***
- ***HP is performing best in IT industry***
- ***Trends of Facebook, Tencent, and Google remain similar in Internet Media industry***

MAJOR DATA QUESTION 4 ANSWERED

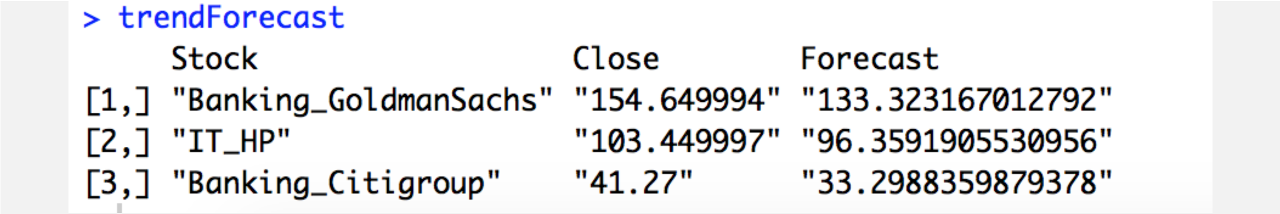
- ***We have obtained Goldman Sachs, and Citigroup among the top 3 best performing stocks. Hence, we can conclude that Banking Industry in general is showing a positive trend in its stocks.***

STEP 9: Forecasting THE TOP 3 STOCKS Prices Using ARIMA Model

In the below for loop, we calculate next day's future value for top 3 stocks using ARIMA model which was obtained as the better model among ARIMA and Monte Carlo Analysis.

```
## Calculating Forecast values for top 3 best trend stocks using ARIMA model
trendForecast = matrix(NA,nrow = 3, ncol = 3)
colnames(trendForecast) <- c("Stock","Close","Forecast")
k = 1
#For loop for iterating over each stock
for (i in trend){
  #Name of the stock stored in matrix
  trendForecast[k,1] = trend[k]

  #Last days closing Stock value
  trendForecast[k,2] = get(i)$Close[1]

  #Trend calculation, one which has maximum number of increases or positivity in their trends
  trendForecast[k,3] = arimaData(get(i))
  k = k+1
  if(k == 4){
    break
  }
}
```

OUTPUT:



```
> trendForecast
     Stock                    Close       Forecast
[1,] "Banking_GoldmanSachs" "154.649994" "133.323167012792"
[2,] "IT_HP"                "103.449997" "96.3591905530956"
[3,] "Banking_Citigroup"    "41.27"      "33.2988359879378"
```

MAJOR DATA QUESTION 5 ANSWERED

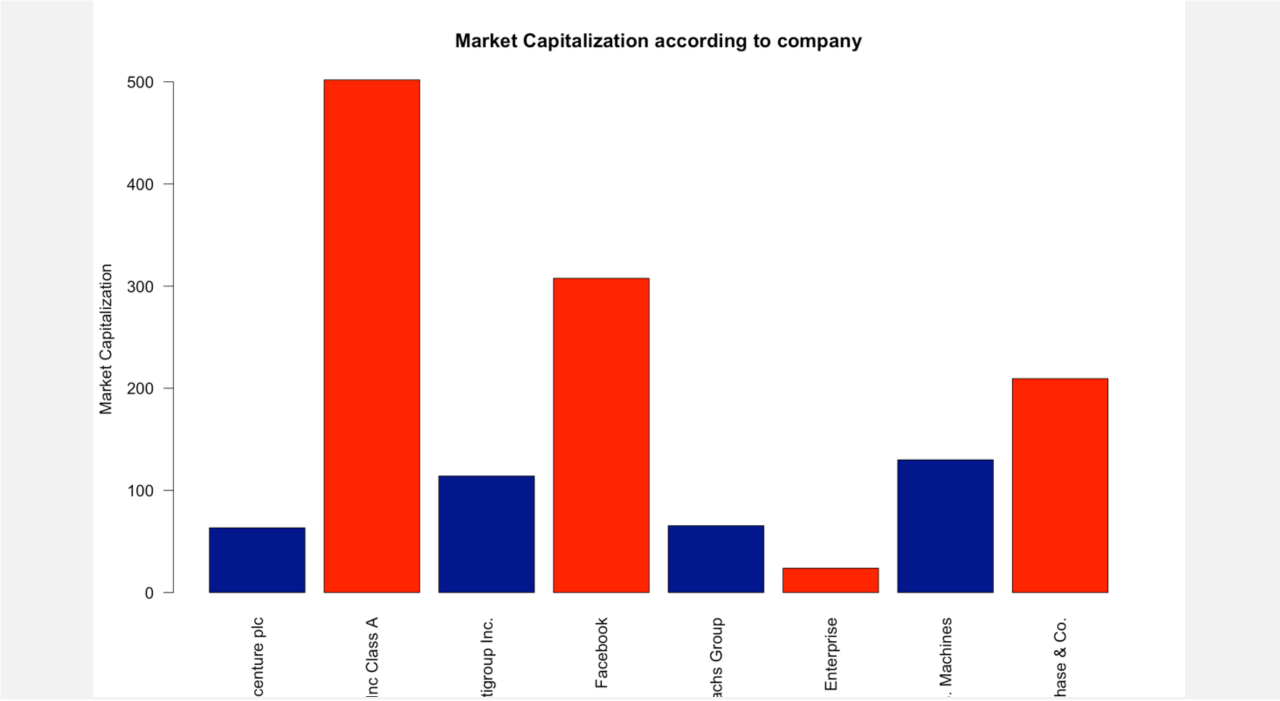
Thus, we have predicted the stock prices of the TOP 3 companies using ARIMA method

STEP 10: Analyzing Companies using Financial Constituents Dataset

We will now look at some other Financial numbers of the companies to understand which of the companies are performing well and in which, an Investor can invest in.

Analyzing Financial Constituents Data:

```
#Selecting the companies whose stock values we predicted

financialConstituentSelected<- financialConstituent[financialConstituent$Symbol %in%
c('ACN','GOOGL','C','FB','GS','HPE','IBM','JPM'),]

#Displaying the Market Capitalization of each company

barplot(financialConstituentSelected$Market.Cap, names.arg = financialConstituentSelected$Name, las =
2, ylab = "Market Capitalization", main = "Market Capitalization according to company",
col=c("darkblue","red"))
```

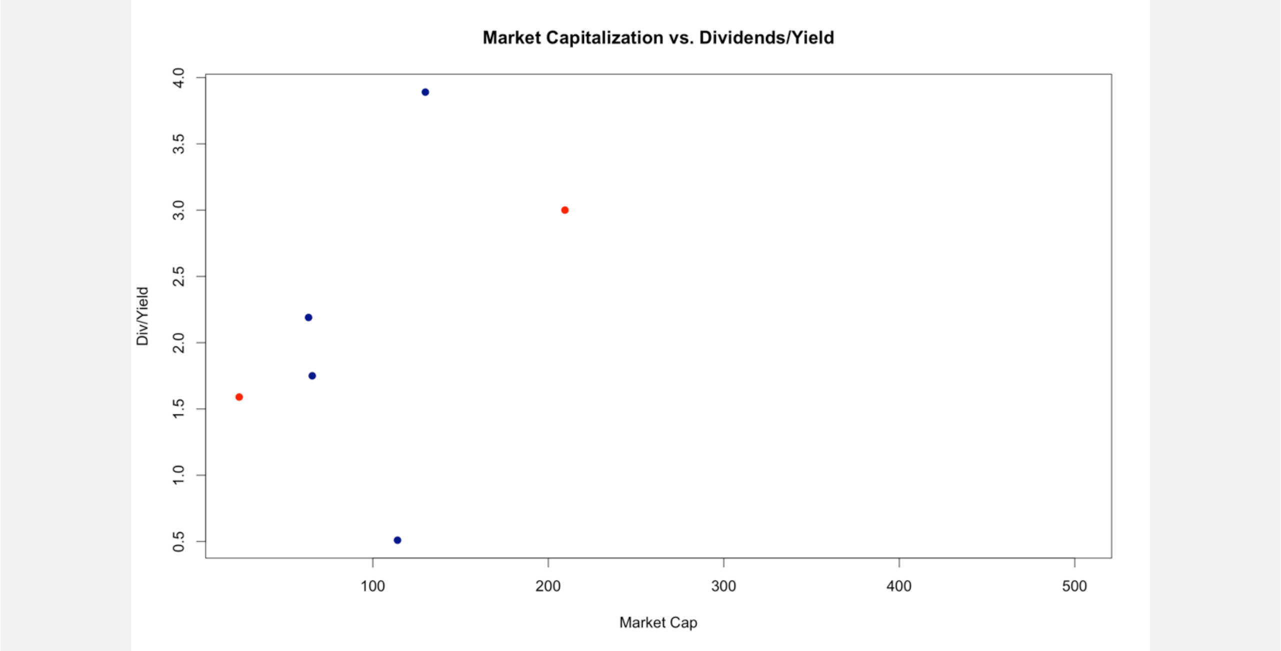


OBSERVATION

Thus, as we can see from the above graph Albhabet Class A (Google) has the highest Market Capitalization followed by Facebook and JP Morgan Chase

#Looking at the table, we might theorize that companies with higher Market capitalization could have higher dividends/Yield

plot(financialConstituentSelected$Market.Cap, financialConstituentSelected$Dividend.Yield, main = "Market Capitalization vs. Dividends/Yield", xlab="Market Cap", ylab = "Div/Yield", pch=19, col=c("darkblue","red"))



OBSERVATION

Thus, as we can see from the above graph companies with higher Market Capitalization tend to have higher Dividend/Yield.

MAJOR DATA QUESTION 6 ANSWERED

Thus, there is a relation between the Market Capitalization and the Dividend Yield

According to investors, the Features of a Company performing well are:

1. The company should have a high Price to Earnings ratio.

2. The dividend yield should be high.

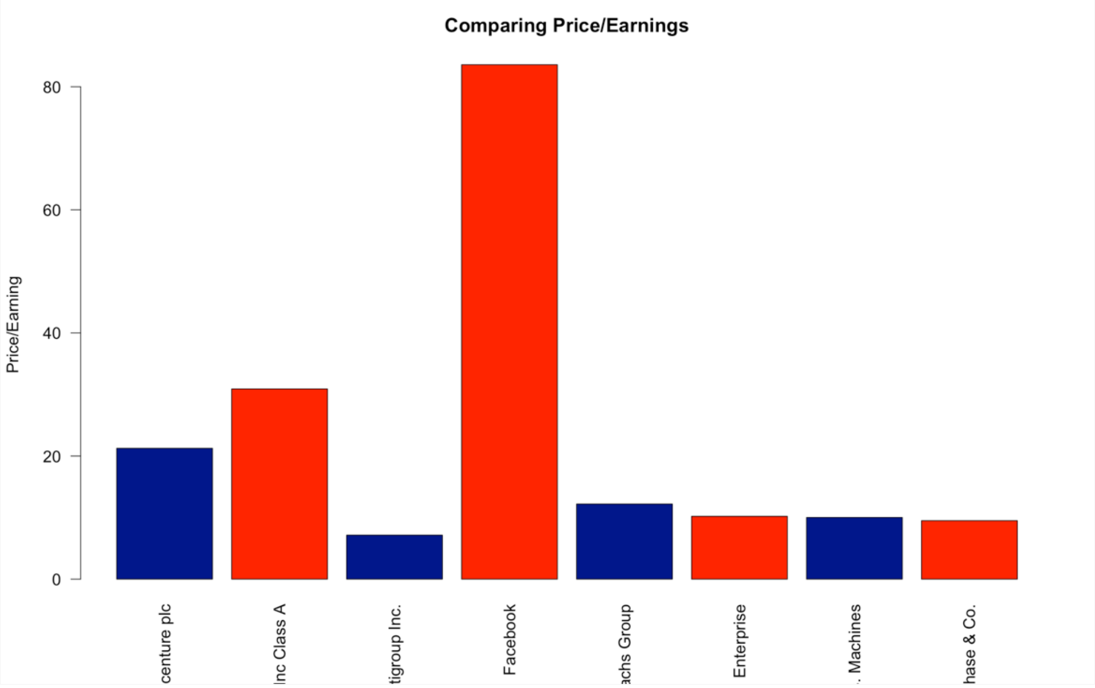
3. The Price to Book Ratio should be high.

4. The company should have a low price-to-sales ratio.

5. The predicted stock value should be near the 52-week high value.

Factor 1: Compare price/earnings ratios to operating performance and financial condition

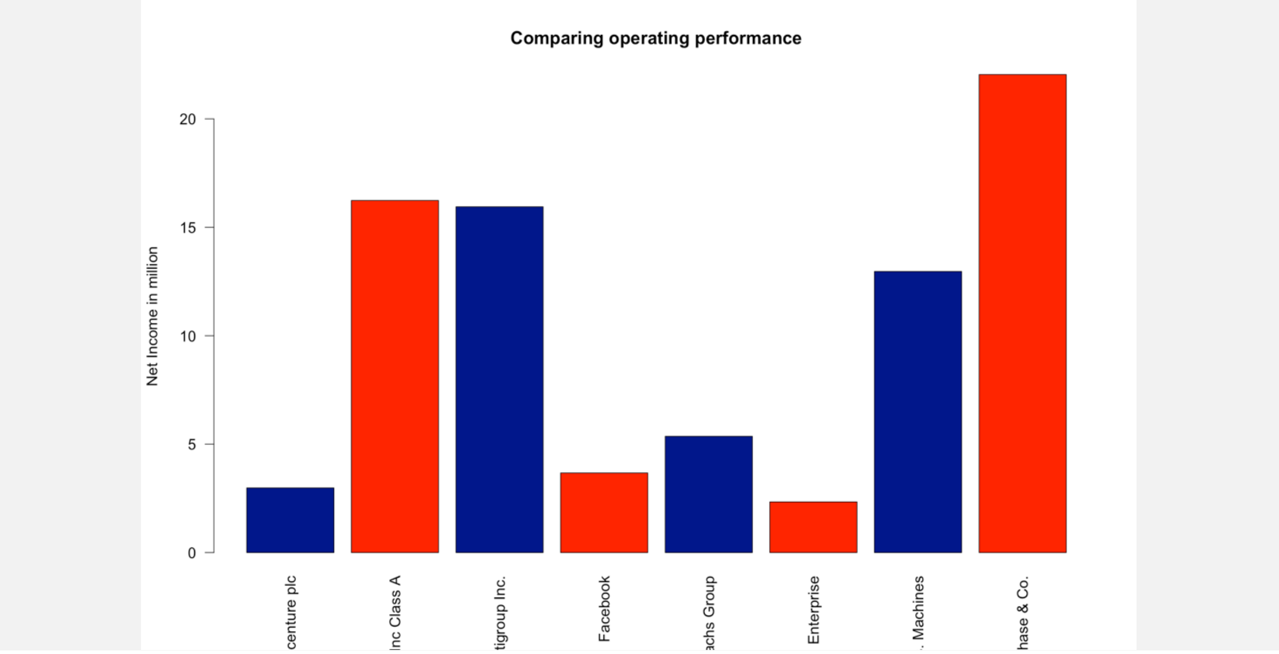
```
barplot(financialConstituentSelected$Price.Earnings, names.arg = financialConstituentSelected$Name, las
= 2, ylab = "Price/Earning", main = "Comparing Price/Earnings", col=c("darkblue","red"))
```



Thus, as we can see Facebook has the highest P/E ratio followed by Google and Accenture. Other companies have a similar Price/Earnings ratio.

Operating Performance:

```
#Calculate the number of Outstanding Shares

outstandingShares<-financialConstituentSelected$Market.Cap/financialConstituentSelected$Price*1000000

#Calculate the Net Income

netIncome<-outstandingShares*financialConstituentSelected$Earnings.Share

barplot(netIncome/1000000, names.arg = financialConstituentSelected$Name, las = 2, ylab = "Net Income in million", main = "Comparing operating performance", col=c("darkblue","red"))
```



P/E ratio is calculated by dividing the company's share price by its earnings per share. If a company were currently trading at a multiple (P/E) of 20, the interpretation is that an investor is willing to pay $20 for $1 of current earnings. In general, a high P/E suggests that investors are expecting higher earnings growth in the future compared to companies with a lower P/E. Also, a higher the Net Income of a company also signifies that the company is doing well with incoming funds.

Thus, a company with higher Price to Earnings ratio and higher Operating Performance denotes that the company is healthy and good to invest in.

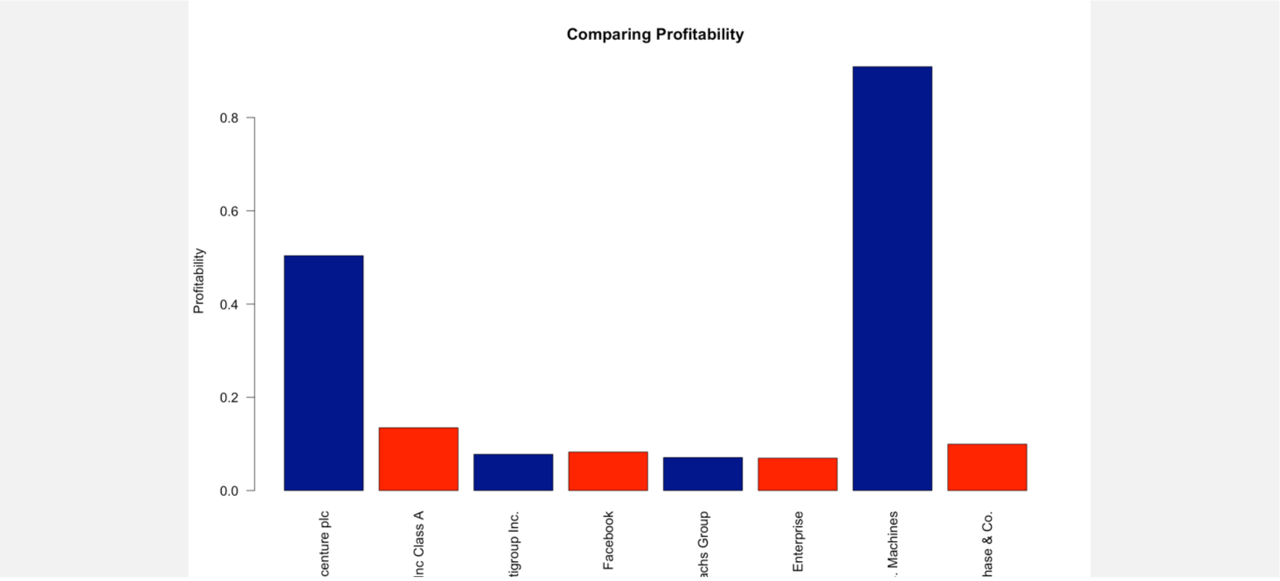
OBSERVATION

Thus, as we can see from the above two graphs JP Morgan Chase, Citigroup and Google have a higher Operating Performance followed by IBM.

Thus from the above 2 graphs, using the 1st Factor, doesn't provide a clear picture of which company to invest in from amongst Google, Facebook, JP Morgan, Citi, IBM and Accenture.

Factor 2: Comparing Profitability

```
barplot((financialConstituentSelected$Earnings.Share/financialConstituentSelected$Book.Value), names.arg = financialConstituentSelected$Name, las = 2, ylab = "Profitability",

	main = "Comparing Profitability", col=c("darkblue","red"))
```



A higher profitability means that the company has been doing well over the last few years. Thus, a company with a High Profitability would be wise to invest in.

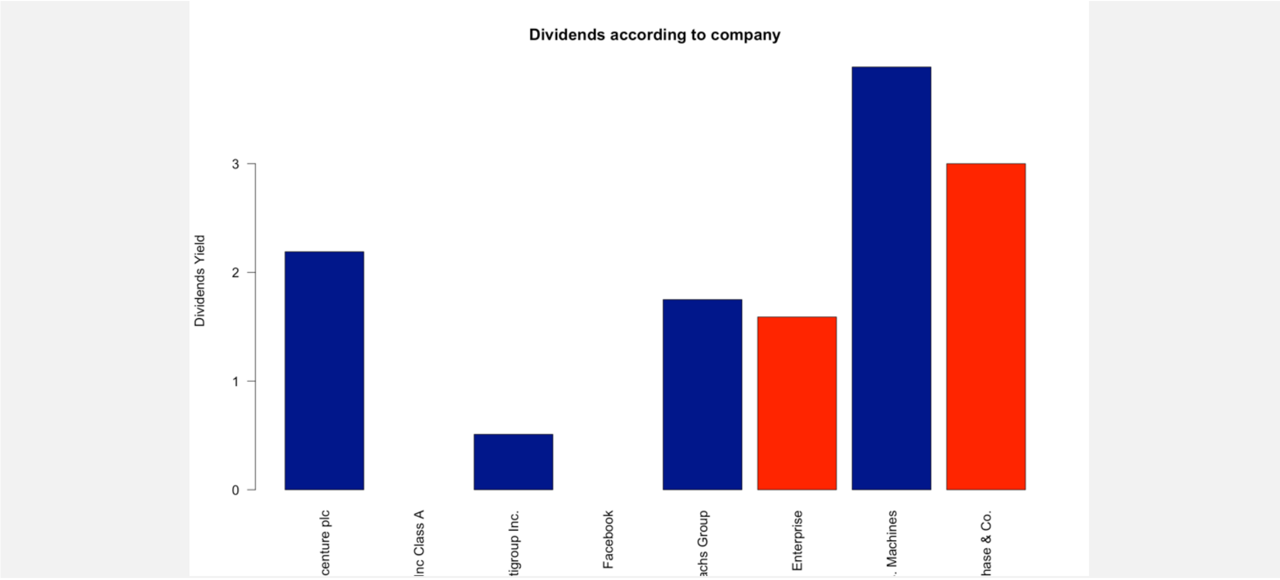
OBSERVATION

Thus, as we can see from the above graph IBM has the highest Profitability followed by Accenture.

Thus using the 2nd Factor, IBM and Accenture would be the companies to invest in.

Factor 3: Comparing Dividend Yield vs Price Earnings

```
barplot(financialConstituentSelected$Dividend.Yield, names.arg = financialConstituentSelected$Name, las
= 2, ylab = "Dividends Yield", main = "Dividends according to company", col=c("darkblue","red"))
```



In addition to evaluating the consistency of their dividend payments, investors would be wise to consider the dividend yields offered by each company. The “dividend yield” refers to the percentage of the price of the stock that is paid to shareholders in the form of a dividend. Because the dividend yield is calculated as a percentage of the price, it follows that a higher price corresponds to a lower dividend yield and vice-versa. Thus a company with a higher dividend yield tends to provide higher returns.

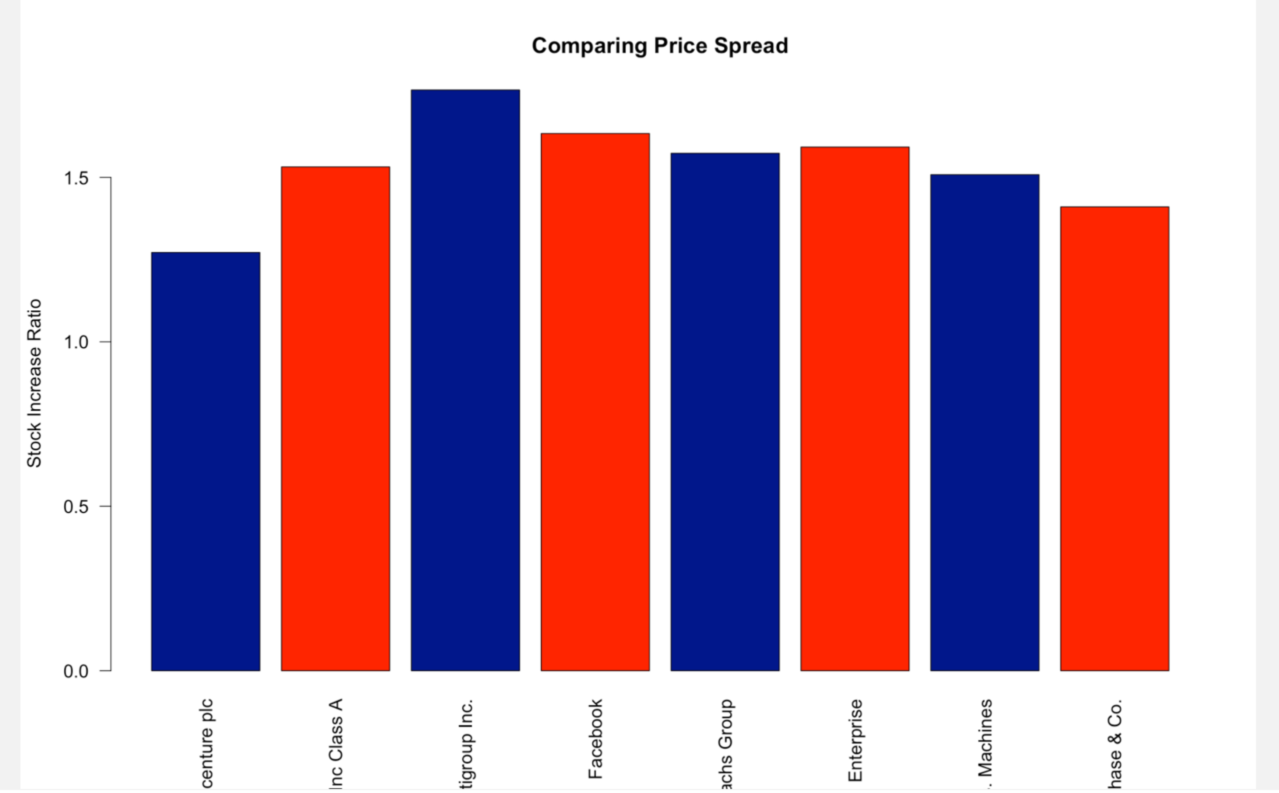
OBSERVATION

Thus, as we can see from the above graph IBM has the highest dividend yield, followed by the rest of the companies

Thus, using the 2nd Factor, IBM would be the companies to invest in, but at the same time, there is not much of a difference in the dividend yield factor, so this factor would have a lower analysis.

#Factor 4: Comparing Price Spread using Stock history

```
barplot(financialConstituentSelected$X52.week.high/financialConstituentSelected$X52.week.low,
names.arg = financialConstituentSelected$Name, las = 2, ylab = "Stock Increase Ratio",

    main = "Comparing Price Spread", col=c("darkblue","red"))
```



Comparing the price history of his companies by comparing their lowest to highest prices, is one of the most important factors to predict the stock to invest in. A higher ratio denotes that the company stock increases at a higher rate.

OBSERVATION

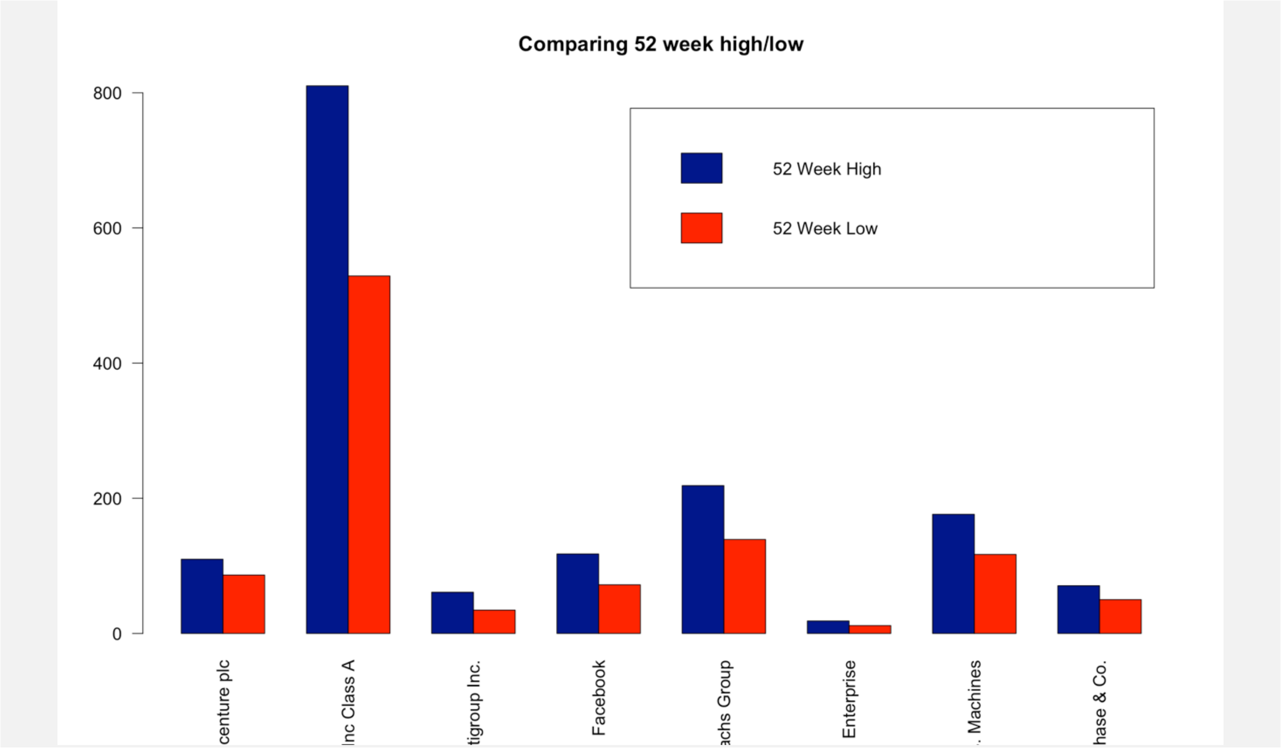
Thus, as we can see from the above graph Citigroup has the highest Stock Increase Ratio spread. Thus this signifies that the rise in stock prices over the past year is very high.

Thus, using the 4th Factor, Citigroup would be the company to invest in, but at the same time, there is not much of a difference in the dividend yield factor, so this factor would have a lower analysis.

#Factor 5: Is predicted Price near 52 Week High?

```
mydf <- data.frame( X1=financialConstituentSelected$X52.week.high,
X2=financialConstituentSelected$X52.week.low)
```

```
barplot(t(as.matrix(mydf)), names.arg = financialConstituentSelected$Name, las=2, beside=TRUE,
col=c("darkblue","red"),main = "Comparing 52 week high/low",legend = c("52 Week High","52 Week
Low"))
```



OBSERVATION

The predicted price of Citi is near its 52-week high value. Thus, using this Factor, Citi would be the company to invest in. The predicted value of the other stocks does not lie close to its highest value.

Thus, using the 5th Factor, Citigroup would be the company to invest in.

MAJOR DATA QUESTION 7 ANSWERED

On the basis of the above 5 factors, we can conclude that Citigroup and IBM are the companies that an investor can invest in.

MAJOR DATA QUESTION 8 ANSWERED

On the basis of the Major Data Question 8 that we answered, we arrived at Citigroup and IBM as the companies that an investor should invest in.

On the basis of the Major Data Question 2 that we answered, we arrived at Goldman Sachs, HP and Citigroup as the 3 Top Most Performing Stocks.

CONCLUSION

Thus, on the basis of Questions 2 and 8:

We can conclude that Citigroup is the most valuable stock from amongst all the companies.

REFERENCES:

1. http://www.dataapple.net/?p=59
2. https://www.youtube.com/watch?v=3gcLRU24-w0
3. https://www.youtube.com/watch?v=LMhlolM_hIU
4. https://www.youtube.com/watch?v=iMET2gbsbHY
5. https://www.econ.berkeley.edu/sites/default/files/Selene%20Yue%20Xu.pdf
6. http://data.okfn.org/data/core/s-and-p-500-companies#data
7. http://robjhyndman.com/hyndsight/dailydata/
8. http://www.r-bloggers.com/plotting-time-series-in-r-using-yahoo-finance-data/
9. https://vancouvervalueinvesting.net/2014/03/03/tii-chapter-13/
10. http://www.money-zine.com/investing/investing/market-ratios/
11. http://beginnersinvest.about.com/cs/newinvestors/a/040901a.htm
12. https://www.aaii.com/journal/article/quantitative-strategies-for-selecting-stocks.mobile
13. https://www.linkedin.com/pulse/forecasting-time-series-r-eric-kramer
14. http://robjhyndman.com/hyndsight/dailydata/
15. https://rpubs.com/jimu_xw/predicting_stock_returns