R Lab Project

Alessandro Valerani 758717

# Introduction

These stocks have been chosen for personal preferences. Some of the picks have been influenced by a very well received book: “The Four - The hidden DNA of Amazon, Apple, Facebook and Google” by Prof. Scott Galloway at NYU.

I picked 6 stocks as a way to compare the pairs against each other in their different market sectors.

[**Introduction**](#_hkwas8b6twn4) **1**

[**Data Summary (File F0)**](#_in22tesajyhi) **3**

[Amazon vs Walmart (Retail)](#_geaa5o4cv91v) 3

[Amazon](#_fhocncezwxae) 3

[Walmart](#_ttvskm5siup3) 4

[Amazon vs Walmart](#_vmwaoxx6q899) 6

[Facebook vs Google](#_yb7zjnu9qzqt) 7

[Comcast & Charter (Internet Service Providers)](#_ucova79ogkp5) 9

[Comcast](#_spfwbanu5wu8) 10

[Charter](#_qxwynyp9b7a) 11

[In R](#_sz09dxknml7m) 12

[**Descriptive analytics (Files: F1 - F3)**](#_qe4plcf0coz3) **14**

[Boxplots](#_aij2yyt54cm3) 17

[4Plots (File F2)](#_fq4nnmi4k1su) 21

[Univariate Analysis](#_tzcdc5ccg4op) 25

[Amazon](#_fuoo80l6o2jj) 25

[Walmart](#_4znr10pt5eyt) 26

[Facebook](#_1h50kziepyqq) 27

[Google](#_3wf1g7kttrj7) 28

[Comcast](#_tqfi3qy3krro) 29

[Charter](#_1wbet3b11fw) 30

[Sample Covariance Matrix (File F3)](#_agyfurrzkxsv) 31

[Sample Correlation Matrix](#_ub6o0l3zfjqj) 32

[AllPairs](#_1q9a9vmu1rdx) 33

[**Predictive Analysis (File F4)**](#_elltty495kay) **34**

[Beta Computation (File F5)](#_7rsilsb0hys9) 39

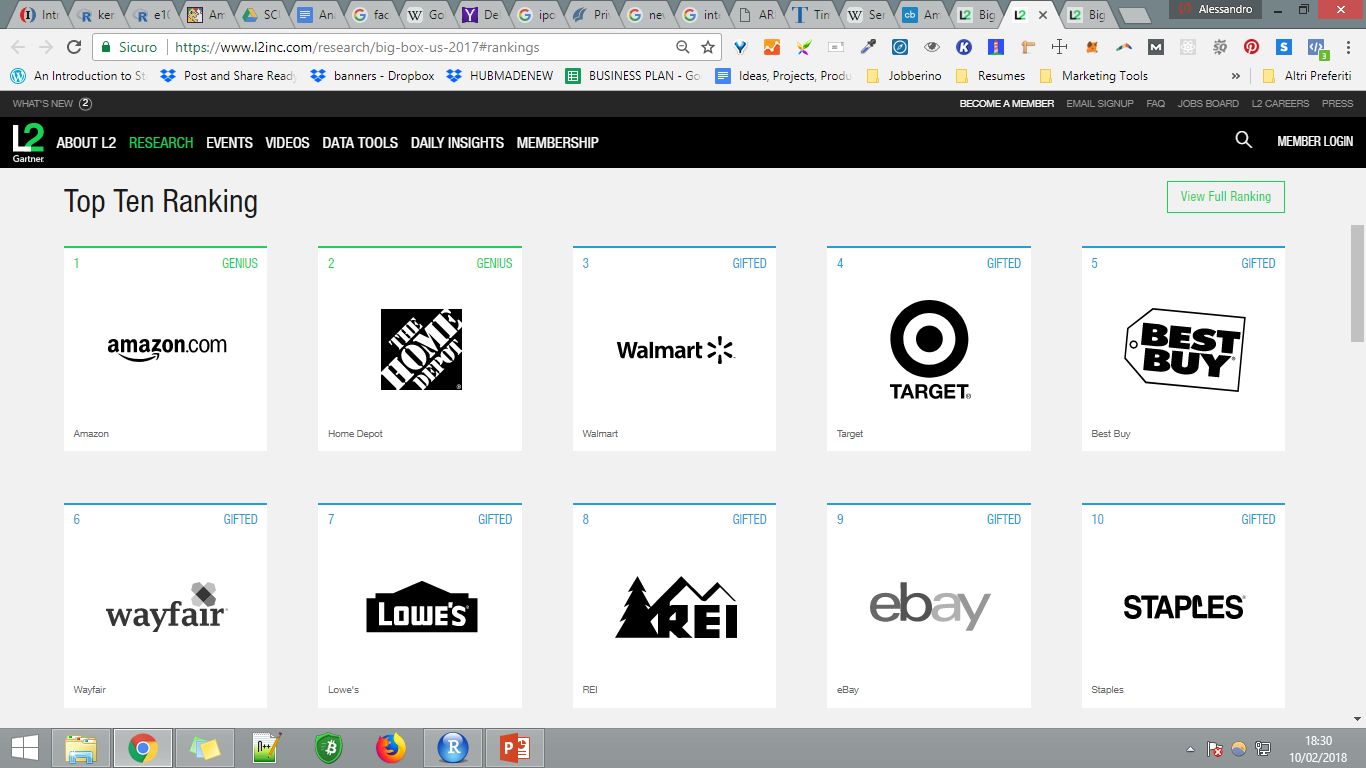
[**Portoflio Management via Mean-Variance (F6)**](#_y7o6eiz3q1) **40**

[Conclusions](#_b9rztqn8oee4) 42

# Data Summary (File F0)

## Amazon vs Walmart (Retail)

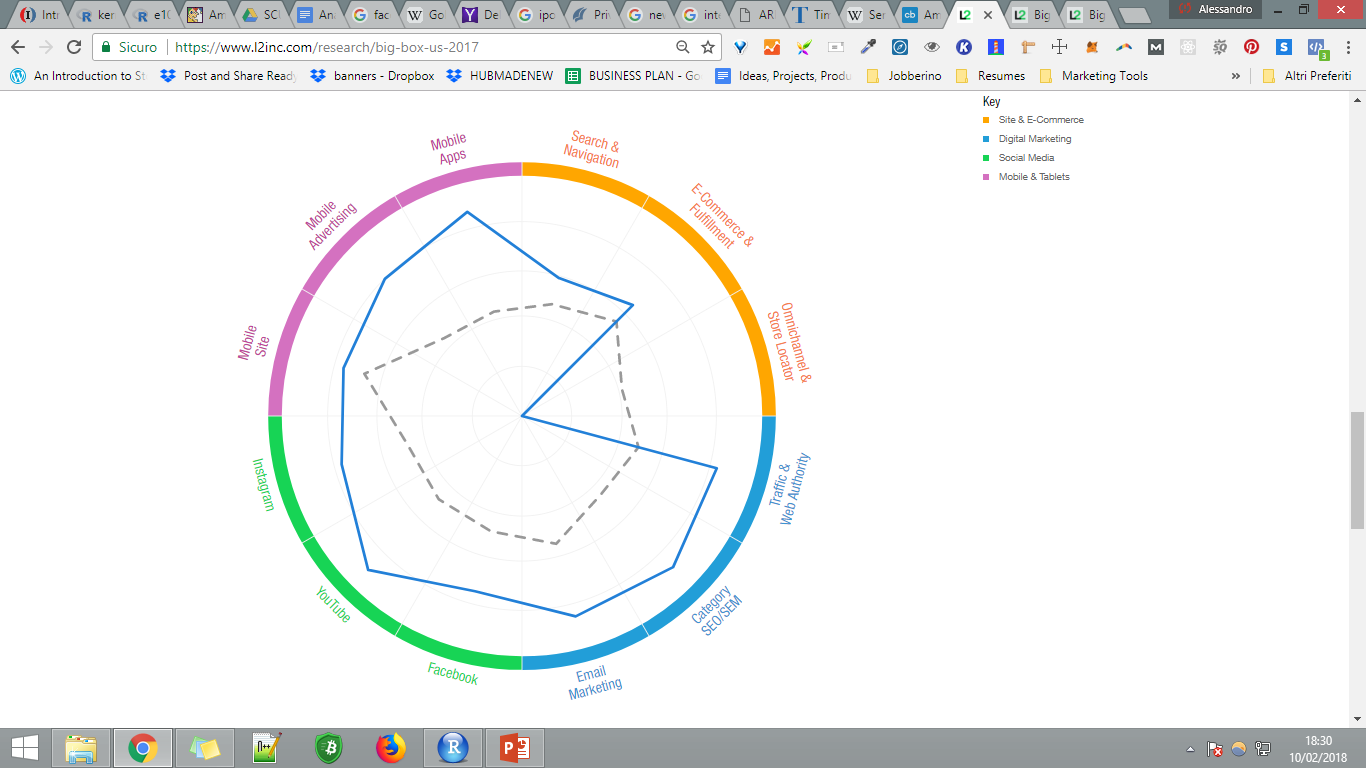
Category: Retail



Source: L2Inc

### Amazon

<https://www.crunchbase.com/organization/amazon>

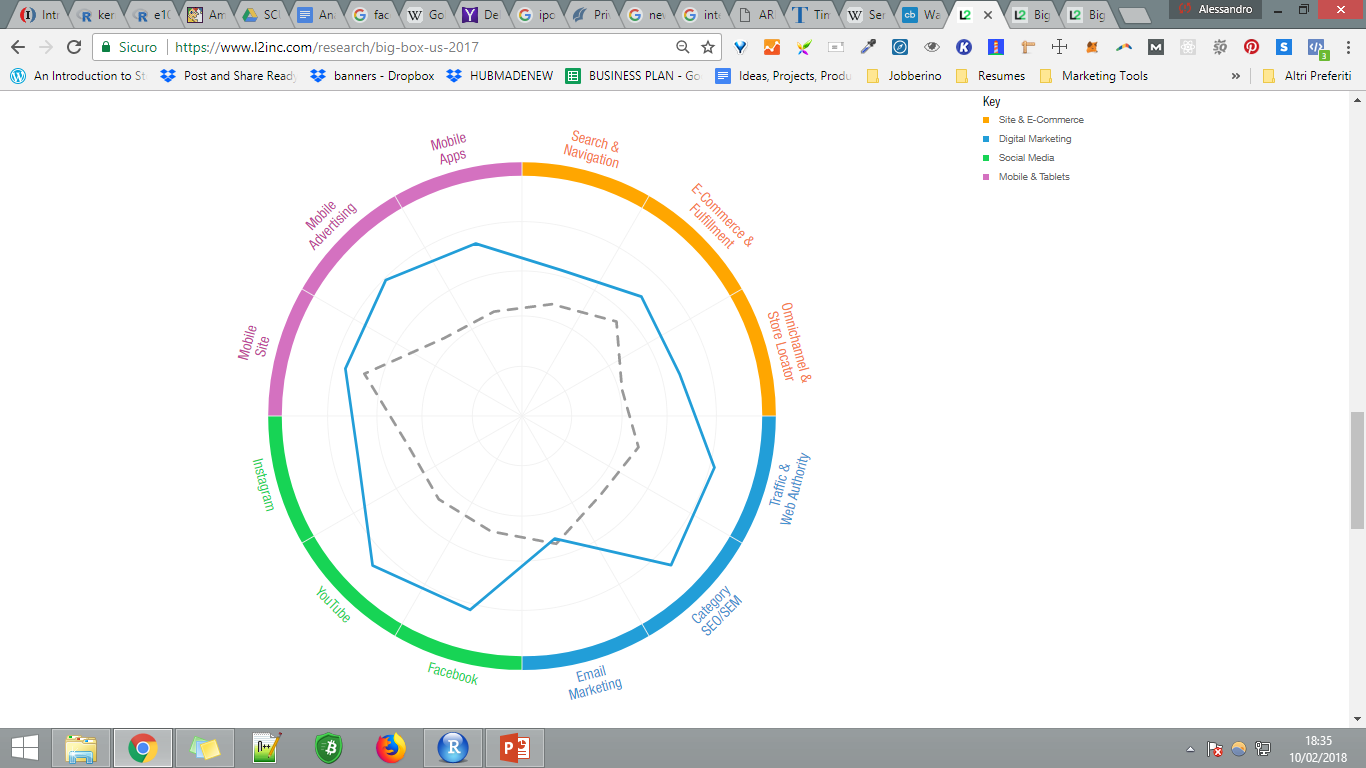


Source: L2Inc

## 

### Walmart

<https://www.crunchbase.com/organization/walmart>



Source: L2Inc

Walmart and Amazon are competing strongly in the search based retail space.

At the end of 2016, Walmart acquired Jet.com for over $3B as a way to bolster its online sales presence.

<https://www.youtube.com/watch?v=6rPEhFTFE9c>

At the time the acquisition was considered a short term fix to a long term proble, a band-aid over a gaping wound as Jet.com was spending over $25MLN a month on its advertising without turning any profit. In order to be profitable Jet.com would have needed to $20BLN in sales.

The acquisition was best considered as an acqui-hire which actually proved very successful as they shown YoY online sales growth of over 20%

<http://www.businessinsider.com/walmarts-online-sales-soar-2017-5?IR=T>

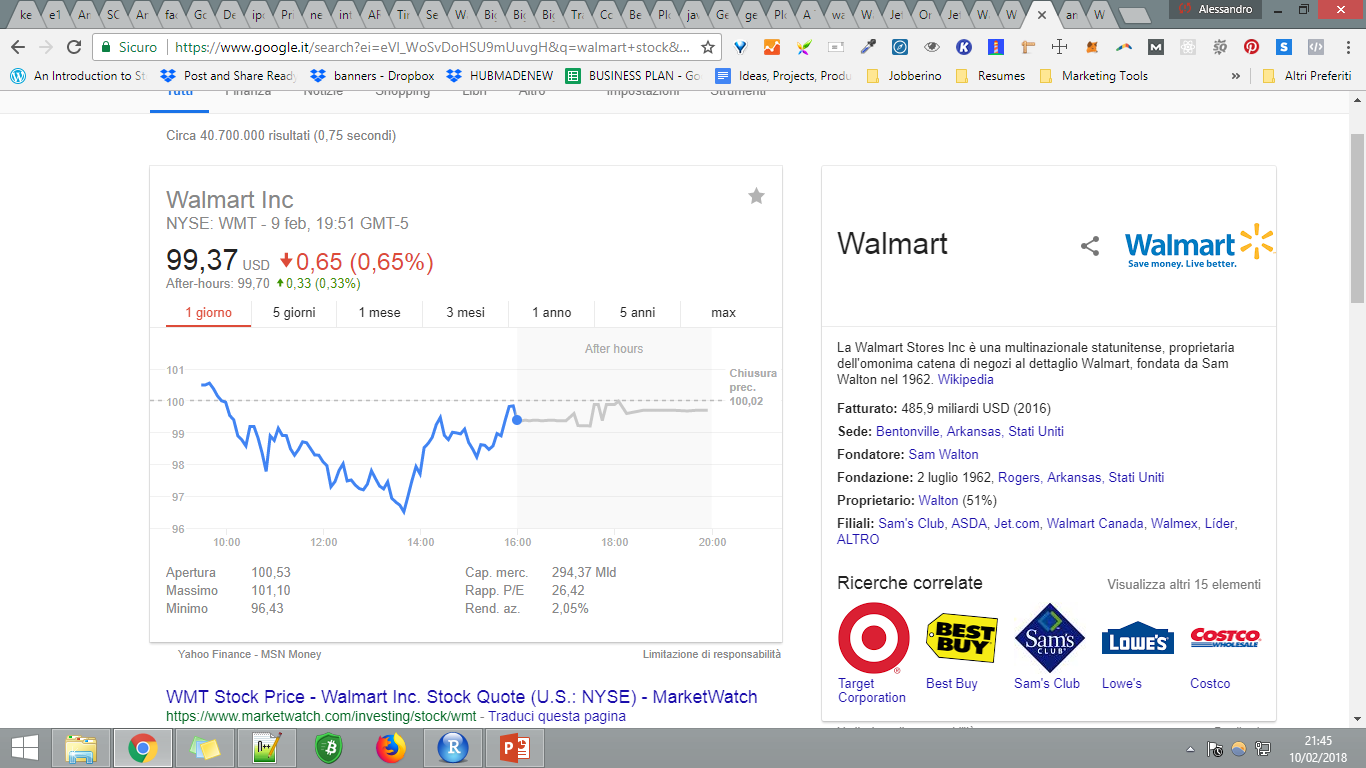
<https://www.digitalcommerce360.com/2017/02/21/wal-marts-online-sales-grow-29-q4/>

As Revenue went up, however, Walmart Profits shrinked by 7% as (I assume) Walmart re-invest it’s capital in growth rather than giving it back as dividends.

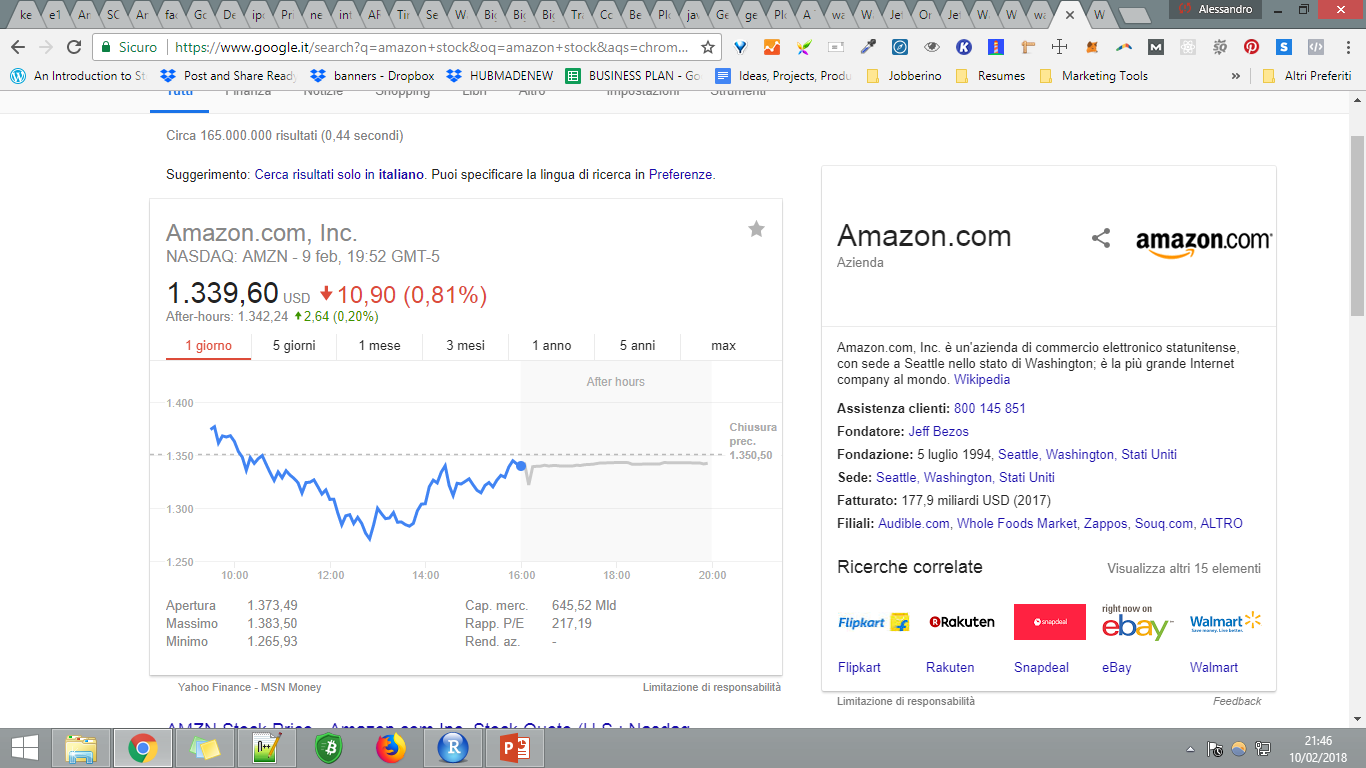
This might reflect poorly on Walmart’s stock as it’s shareholder might be used to receiving a fat dividend.

[While Amazon stock has yet to pay any dividend](https://www.fool.com/investing/2017/12/28/will-amazon-start-paying-a-dividend-in-2018.aspx), ever. As Prof.Scott Gallaways argues, Amazon sold it’s vision rather than it’s capital and as such it is considered different from other Big Box retials.

A quick look at Walmart’s vs Amazon’s P/E Ratio can be a good indicator of this



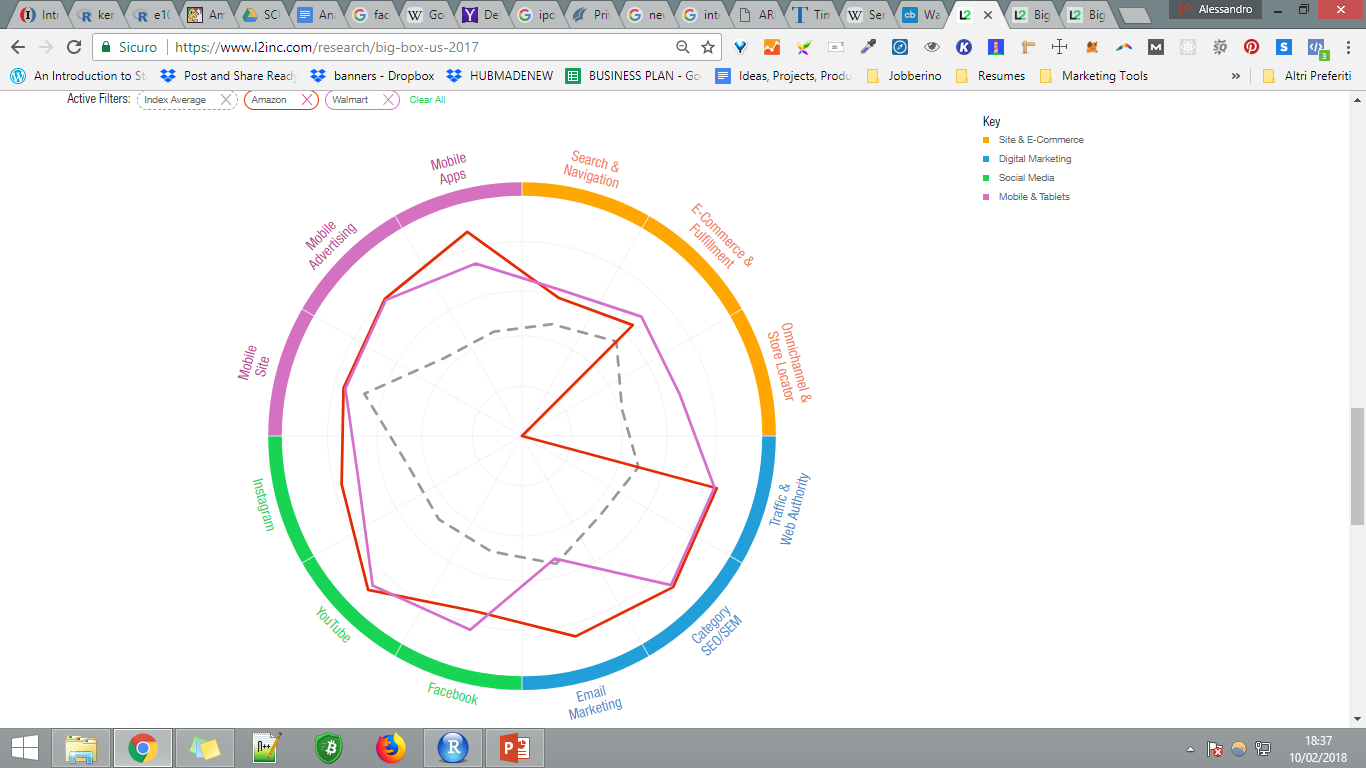
P/E Ratio: 26.42



P/E Ratio: 217.19

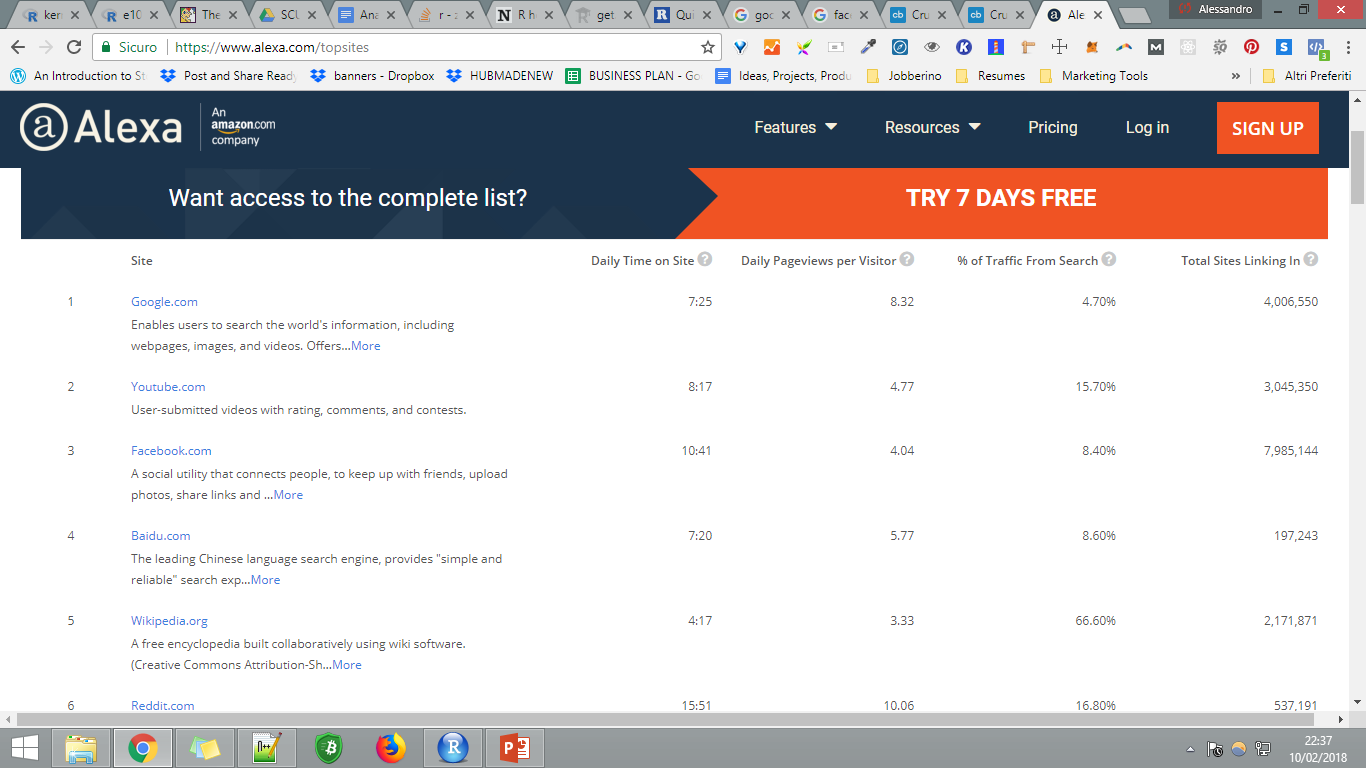
Amazon has approximately 3 times the market cap and 8 times the P/E Ratio of Walmart.

### Amazon vs Walmart



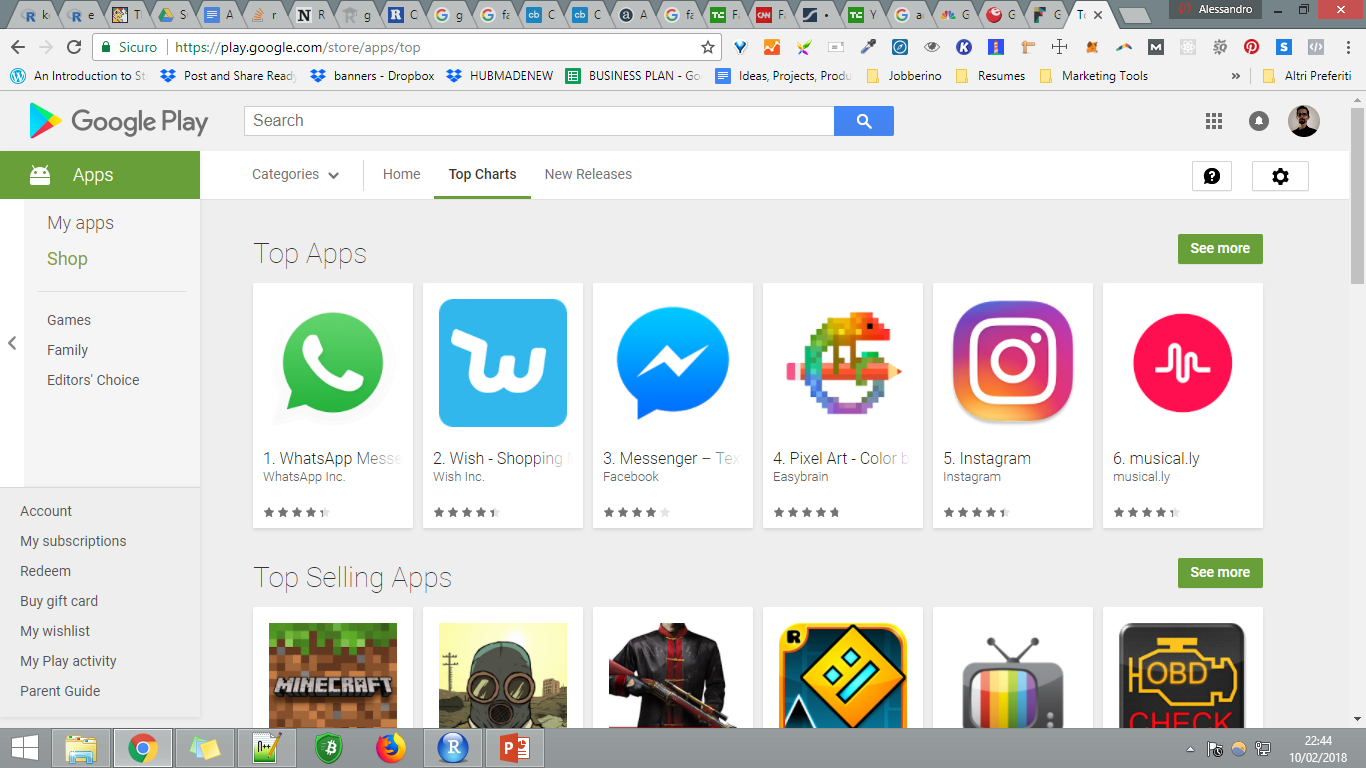
Amazon in Red, Walmart in Purple. Source: L2Inc

## Facebook vs Google



Source: Alexa.com

Alphabet owns the top 2 websites in the world with Facebook following in 3rd place. (Instagram.com being #16)



Source: play.google.com (Whatsapp, Messenger and Instagram are owned by Facebook)

Facebook reach 2BLN monthly active users in June 2017

<http://money.cnn.com/2017/06/27/technology/facebook-2-billion-users/index.html>

Google global revenue ($106BLN in 2017, $74.9BLN of which from advertising)

<https://www.statista.com/statistics/266206/googles-annual-global-revenue/>

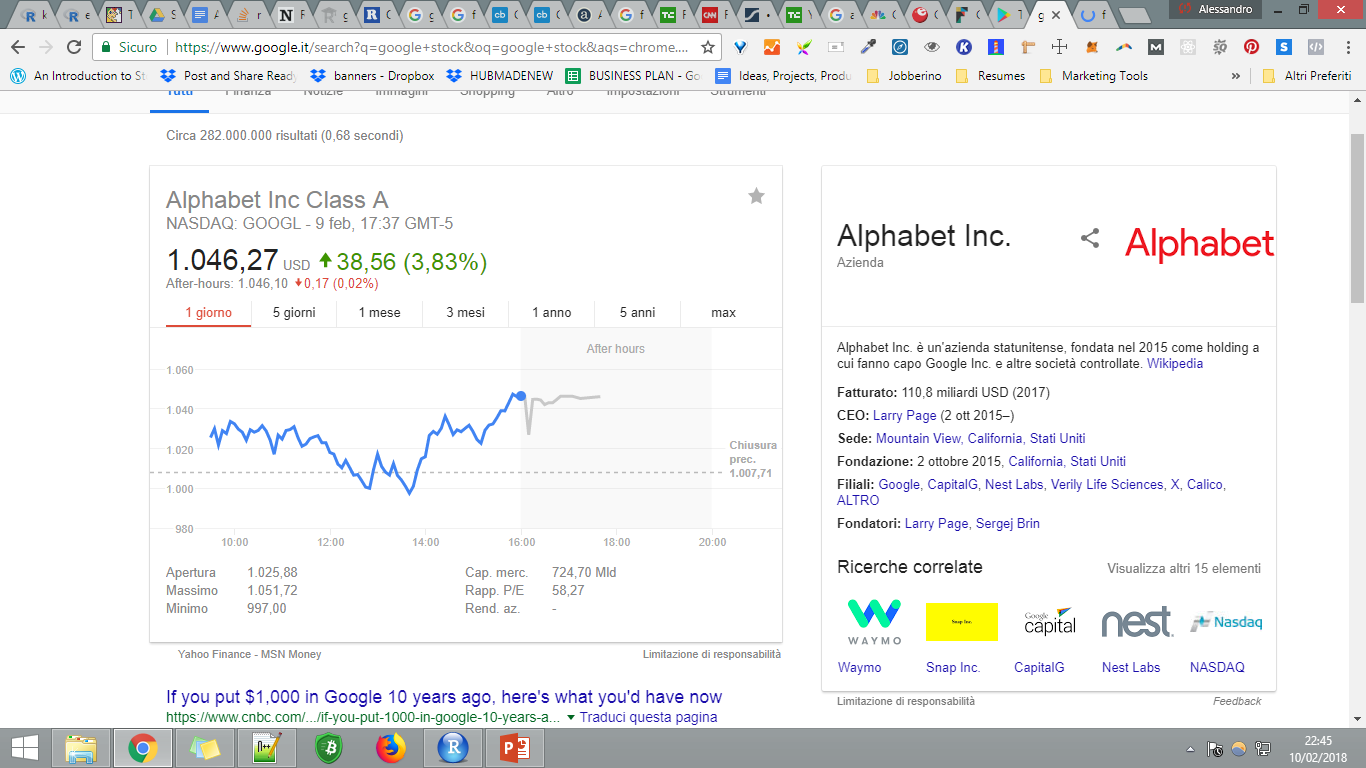
Youtube 1.5 Monthly logged in users:

<https://techcrunch.com/2017/06/22/youtube-has-1-5-billion-logged-in-monthly-users-watching-a-ton-of-mobile-video/>

60% of advertising revenue in US is made by Google and Facebook

<http://fortune.com/2017/07/28/google-facebook-digital-advertising/>

The digital advertising sector in 2016 saw Google and Facebook gaining a 103% growth which means that beside them the sector is actually declining.





## Comcast & Charter (Internet Service Providers)

<https://arstechnica.com/information-technology/2016/01/comcast-and-charter-may-soon-control-70-of-25mbps-internet-subscriptions/>

>70% combined market share

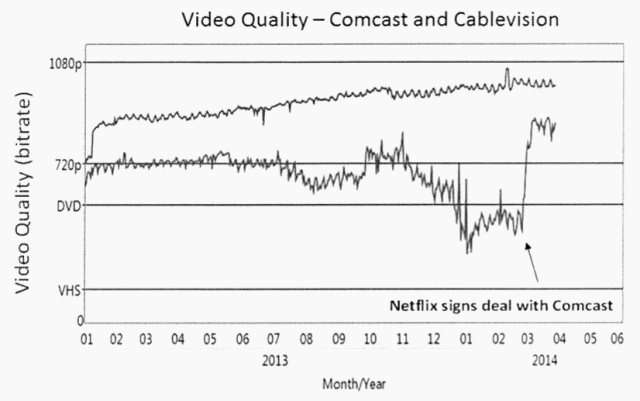
Net neutrality

<http://www.corriere.it/tecnologia/18_gennaio_17/pioggia-ricorsi-contro-abrogazione-net-neutrality-ebec458a-fb72-11e7-8302-0b7dac2e21fa.shtml>

<https://www.corrierecomunicazioni.it/telco/net-neutrality-appello-att-al-congresso-usa-serve-un-internet-bill-of-rights/>

Netflix pays Comcast not to be throttled down

<https://qz.com/256586/the-inside-story-of-how-netflix-came-to-pay-comcast-for-internet-traffic/>



Source: QZ.com

Net Neutraility has had a long and complicated history in the US.

With the recent deregulation, users expect ISPs to start charging content providers arbitrary prices and receive a downgraded service.

The stock market however should have a “more optimistic” perspective on the subject

<https://en.wikipedia.org/wiki/Net_neutrality_in_the_United_States#FCC_Open_Internet_Order_(2010)>

### Comcast



### Charter

## 

## 

## In R

File: F0 - GetData.r

##DATA SUMMARY

#Si connette a yahoo, e restituisce il valore della stock (passato tramite name)

library(tseries)

get <- function(name){

x <- get.hist.quote(instrument = name, start = "2008-01-01", end="2017-10-01",

quote = c("Adjusted"), compression="m")

}

##c("Open", "High", "Low", "Close", "Adjusted", "Volume")

#get.his.quote returns a zoo

Amazon.adj.z <- get("AMZN") #amazon

Walmart.adj.z <- get("WMT") #walmart

Facebook.adj.z <- get("FB") #facebook

Google.adj.z <- get("GOOGL") #Alphabet inc, owner of Google

Comcast.adj.z <- get("CMCSA") #Comcast

Charter.adj.z <- get("CHTR") #Charter

tutti <- cbind(Amazon.adj.z,

Walmart.adj.z,

Facebook.adj.z,

Google.adj.z,

Comcast.adj.z,

Charter.adj.z)

##Visualizza i prezzi adjusted

library(dygraphs)

##Mostra con dygraph, cambia data, width 100% per lables, selettore in basso

dygraph(tutti, main="Adjusted Monthly Prices of chosen Stock") %>%

dySeries(name="Adjusted.Amazon.adj.z", label = "\nAmazon Adj") %>%

dySeries(name="Adjusted.Walmart.adj.z", label = "Walmart Adj") %>%

dySeries(name="Adjusted.Facebook.adj.z", label = "Facebook Adj") %>%

dySeries(name="Adjusted.Google.adj.z", label = "Google Adj") %>%

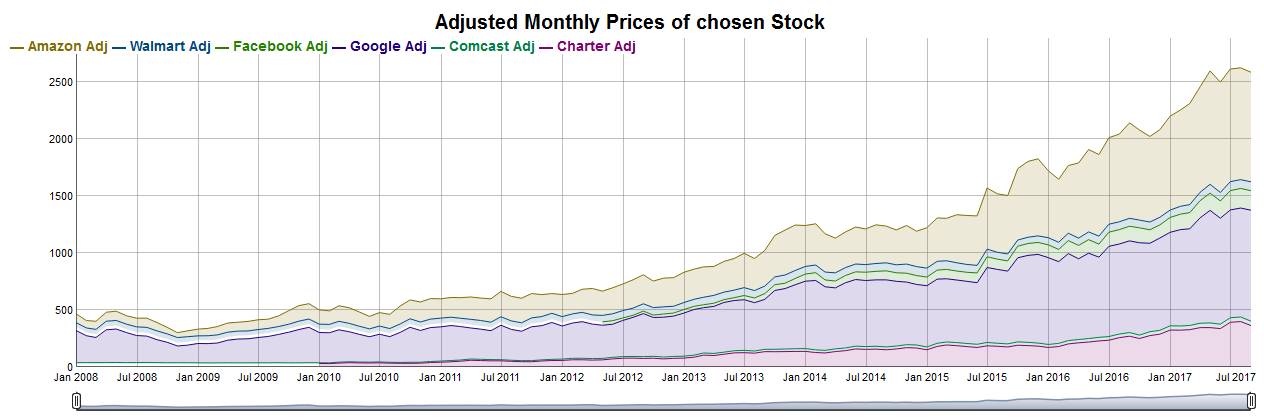
dySeries(name="Adjusted.Comcast.adj.z", label = "Comcast Adj") %>%

dySeries(name="Adjusted.Charter.adj.z", label = "Charter Adj") %>%

dyRangeSelector(height = 20) %>%

dyOptions(stackedGraph = TRUE) %>%

dyLegend(width = "100%")



Il grafo di output è interattivo grazie a Dygraphs.

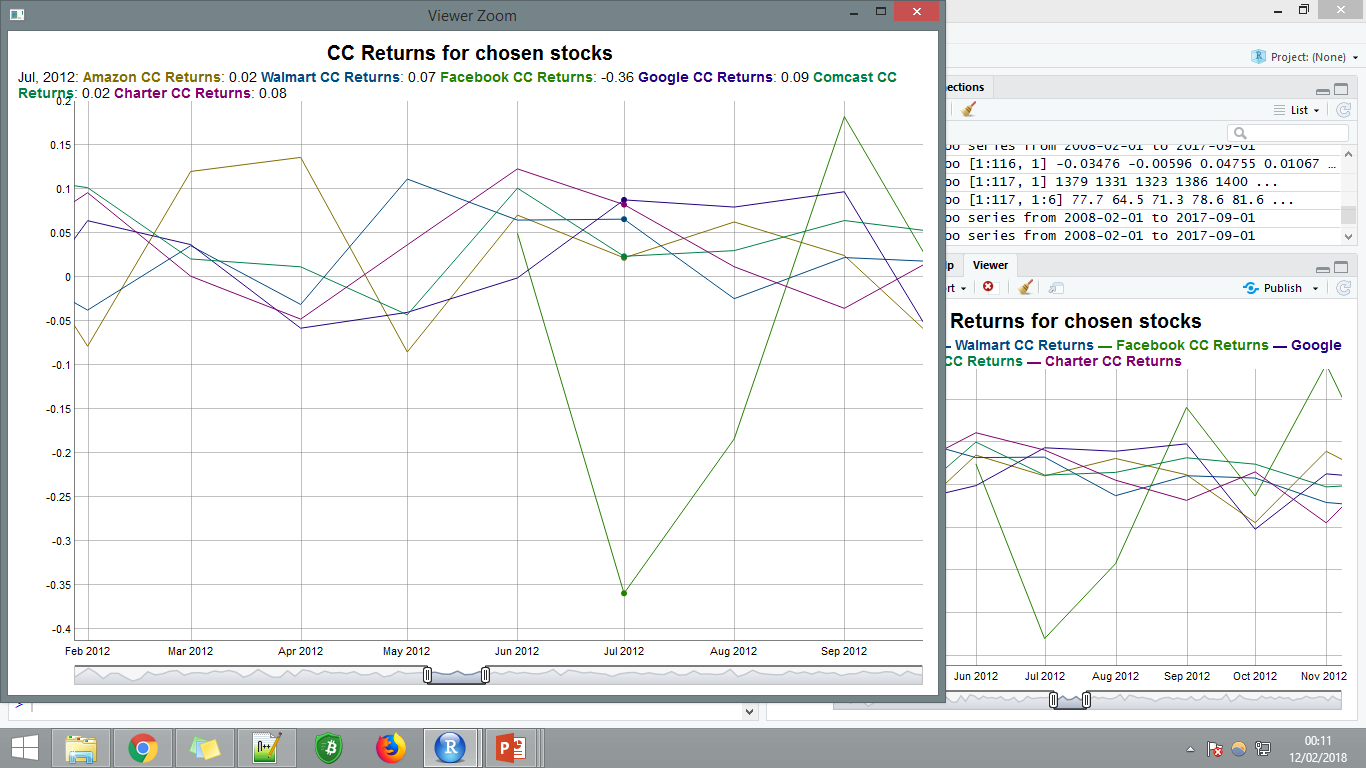
# Descriptive analytics (Files: F1 - F3)

# 

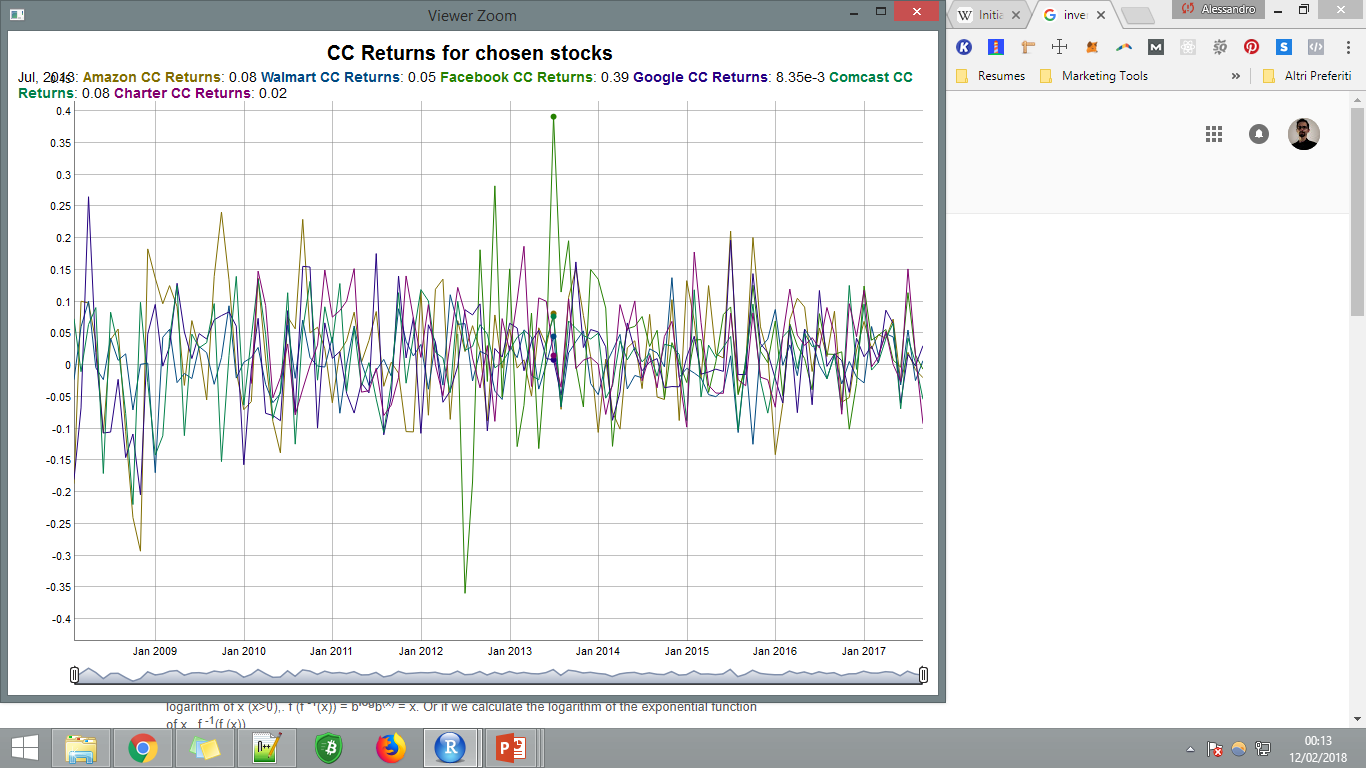
# 

Alcuni ritorni molto significativi sono il “dip” post IPO di Facebook e dei forti spike di Amazon

**Facebook stock dipped just a week after it’s IPO**



With bearish sentiment remaining during july 2012



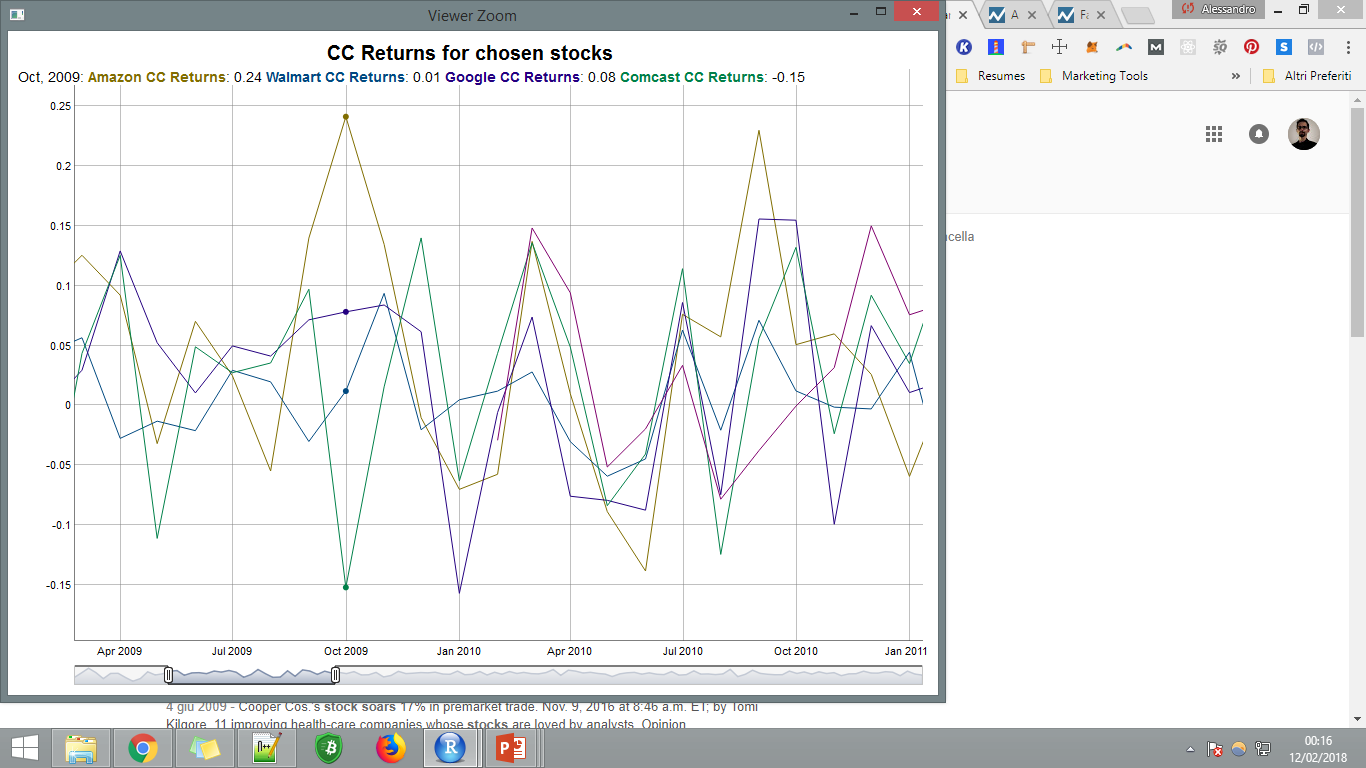
July 2013 very bullish after a strong earning statement

<http://buzz.money.cnn.com/2013/07/25/facebook-earnings-stock/>

"One year into Facebook's mobile advertising efforts, mobile has increased from zero to 41% of total ad revenue," Anmuth highlighted in a note to clients.

Facebook stock rose by 30% in one day.

**Amazon**



Oct 2009

<http://money.cnn.com/2009/10/22/technology/Amazon_earnings/?postversion=2009102306>

69% profit surge in Q3 2009

Sept 2010

<http://money.cnn.com/2010/10/21/technology/amazon_earnings/index.htm>

Amazon sales rose by 40% YoY in Q3 2010

## Boxplots

## 

## 4Plots (File F2)

## Univariate Analysis

### Amazon

#Amazon

> mean(Amazon.ret.mat)

[1] 0.0216852

> var(Amazon.ret.mat)

Adjusted

Adjusted 0.008253085

> sd(Amazon.ret.mat)

[1] 0.09084649

> skewness(Amazon.ret.mat)

[1] -0.4007064

> kurtosis(Amazon.ret.mat)

[1] 0.8515085

> summary(Amazon.ret.mat)

Adjusted

Min. :-0.29305

1st Qu.:-0.03882

Median : 0.02532

Mean : 0.02169

3rd Qu.: 0.07928

Max. : 0.24106

## 

### Walmart

#Walmart

> mean(Walmart.ret.mat)

[1] 0.005776232

> var(Walmart.ret.mat)

Adjusted

Adjusted 0.002160168

> sd(Walmart.ret.mat)

[1] 0.0464776

> skewness(Walmart.ret.mat)

[1] -0.3191834

> kurtosis(Walmart.ret.mat)

[1] 1.502392

> summary(Walmart.ret.mat)

Adjusted

Min. :-0.169408

1st Qu.:-0.022849

Median : 0.005247

Mean : 0.005776

3rd Qu.: 0.037200

Max. : 0.137816

### Facebook

#Facebook

> mean(Facebook.ret.mat)

[1] 0.02739264

> var(Facebook.ret.mat)

Adjusted

Adjusted 0.01118341

> sd(Facebook.ret.mat)

[1] 0.1057517

> skewness(Facebook.ret.mat)

[1] -0.03738282

> kurtosis(Facebook.ret.mat)

[1] 3.572187

> summary(Facebook.ret.mat)

Adjusted

Min. :-0.35943

1st Qu.:-0.02614

Median : 0.02177

Mean : 0.02739

3rd Qu.: 0.07744

Max. : 0.39143

### Google

#Google

> mean(Google.ret.mat)

[1] 0.01066969

> var(Google.ret.mat)

Adjusted

Adjusted 0.006100101

> sd(Google.ret.mat)

[1] 0.07810314

> skewness(Google.ret.mat)

[1] 0.1004066

> kurtosis(Google.ret.mat)

[1] 0.707263

> summary(Google.ret.mat)

Adjusted

Min. :-0.20429

1st Qu.:-0.04085

Median : 0.01207

Mean : 0.01067

3rd Qu.: 0.05682

Max. : 0.26529

### Comcast

#Comcast

> mean(Comcast.ret.mat)

[1] 0.01419726

> var(Comcast.ret.mat)

Adjusted

Adjusted 0.004866069

> sd(Comcast.ret.mat)

[1] 0.06975722

> skewness(Comcast.ret.mat)

[1] -0.6359832

> kurtosis(Comcast.ret.mat)

[1] 0.5068699

> summary(Comcast.ret.mat)

Adjusted

Min. :-0.21958

1st Qu.:-0.02490

Median : 0.02762

Mean : 0.01420

3rd Qu.: 0.05411

Max. : 0.13982

### Charter

#Charter

> mean(Charter.ret.mat)

[1] 0.02688673

> var(Charter.ret.mat)

Adjusted

Adjusted 0.004685212

> sd(Charter.ret.mat)

[1] 0.06844861

> skewness(Charter.ret.mat)

[1] 0.2676872

> kurtosis(Charter.ret.mat)

[1] -0.7446324

> summary(Charter.ret.mat)

Adjusted

Min. :-0.09764

1st Qu.:-0.02871

Median : 0.01588

Mean : 0.02689

3rd Qu.: 0.08202

Max. : 0.18725

## Sample Covariance Matrix (File F3)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | AmazonRetCC | WalmartRetCC | FacebookRetCC | GoogleRetCC | ComcastRetCC | CharterRetCC |
| AmazonRetCC | 5.13E-03 | 7.72E-05 | 0.0013655532 | 2.39E-03 | 0.0010604552 | 0.0010581641 |
| WalmartRetCC | 7.72E-05 | 1.99E-03 | -0.0003069339 | -4.86E-05 | 0.0004695905 | 0.0004546492 |
| FacebookRetCC | 1.37E-03 | -3.07E-04 | 0.0111834134 | 8.39E-04 | 0.0007733719 | -0.000989344 |
| GoogleRetCC | 2.39E-03 | -4.86E-05 | 0.0008387805 | 3.17E-03 | 0.0007224683 | 0.0004477737 |
| ComcastRetCC | 1.06E-03 | 4.70E-04 | 0.0007733719 | 7.22E-04 | 0.0023924539 | 0.0019691165 |
| CharterRetCC | 1.06E-03 | 4.55E-04 | -0.000989344 | 4.48E-04 | 0.0019691165 | 0.0044323028 |

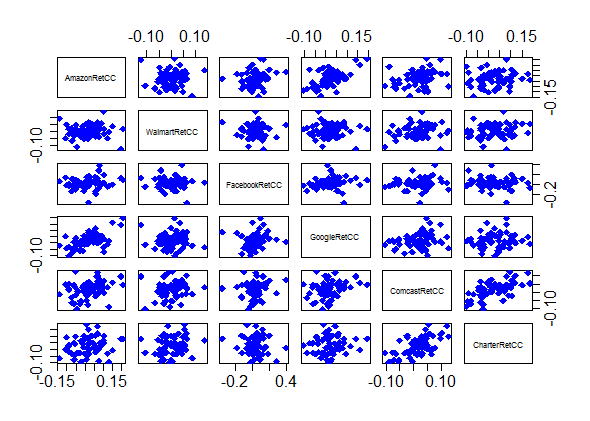
## Sample Correlation Matrix

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | AmazonRetCC | WalmartRetCC | FacebookRetCC | GoogleRetCC | ComcastRetCC | CharterRetCC |
| AmazonRetCC | 1 | 0.02412033 | 0.18022008 | 0.59385585 | 0.3025884 | 0.2218299 |
| WalmartRetCC | 0.02412033 | 1 | -0.06500293 | -0.01935963 | 0.215017 | 0.1529456 |
| FacebookRetCC | 0.18022008 | -0.06500293 | 1 | 0.14096049 | 0.1495132 | -0.1405224 |
| GoogleRetCC | 0.59385585 | -0.01935963 | 0.14096049 | 1 | 0.2625024 | 0.1195309 |
| ComcastRetCC | 0.30258842 | 0.21501704 | 0.14951316 | 0.26250239 | 1 | 0.6046929 |
| CharterRetCC | 0.2218299 | 0.15294561 | -0.14052236 | 0.11953088 | 0.6046929 | 1 |

Comcast and Charter share a high 0.6, suggesting that these stocks are being treated similarly on the market

Amazon is sharing a moderate correlation with all but Walmart which is its most direct competitor

## AllPairs



# Predictive Analysis (File F4)

I’ve used the following code to produce a ARIMA based predictive model

library( forecast )

returnsTrain <- returns[1:(n \* length(returns))] # Train dataset

returnsTest <- returns[(n \* length(returns) + 1):(n \* length(returns)+ 1 + m \* length(returns))] # Test dataset

fit <- arima(returnsTrain, order = c(2, 0, 2))

arma.preds <- predict(fit, n.ahead = (length(returns) - (n \* length(returns))))$pred

arma.forecast <- forecast(fit, h = 35)

plot(arma.forecast, main = "ARMA forecasts for Amazon returns")

accuracy(arma.preds, returnsTest)[2] # RMSE values

lines( returnsTest )

## 

Amazon Accuracy

[1] 0.08411751

Walmart Accuracy

0.04998011

Facebook Accuracy

0.05643034

Google Accuracy

0.05670273

Comcast Accuracy

0.05922246

Charter Accuracy

0.057656

**I didn’t validate the model by running the portfolio simulation.**

## Beta Computation (File F5)

All stocks belong to the S&P500 which I will reference

##Fatto con performance analytics

> CAPM.beta(Amazon.adj.ret, SP500.adj.ret, Rf = 0)

[1] 0.984947

> CAPM.beta(Walmart.adj.ret, SP500.adj.ret, Rf = 0)

[1] 0.3322981

> CAPM.beta(Facebook.adj.ret, SP500.adj.ret, Rf = 0)

[1] 0.6923669

> CAPM.beta(Google.adj.ret, SP500.adj.ret, Rf = 0)

[1] 1.01724

> CAPM.beta(Comcast.adj.ret, SP500.adj.ret, Rf = 0)

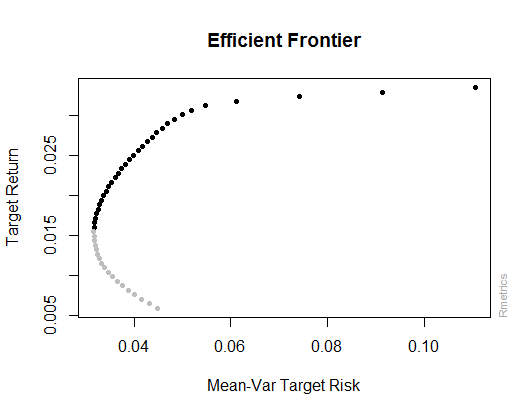
[1] 1.07621

> CAPM.beta(Charter.adj.ret, SP500.adj.ret, Rf = 0)

[1] 0.8444471

**I didn’t calculate the expected return of the stocks based on the beta value**

# Portoflio Management via Mean-Variance (F6)

I’ve used: pf <- portfolioFrontier(Data) to calculate the efficient frontier, minimizing risk and using an alpha of 0.05

**Portfolio Weights:**

Amazon.Adj Walmart.Adj Facebook.Adj Google.Adj Comcast.Adj Charter.Adj

1 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000

13 0.0000 0.5444 0.0346 0.2185 0.2024 0.0000

25 0.0216 0.3316 0.1184 0.2423 0.0617 0.2245

**37 0.0700 0.0890 0.1933 0.2320 0.0000 0.4156**  Chosen Weights

50 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000

**Covariance Risk Budgets:**

Amazon.Adj Walmart.Adj Facebook.Adj Google.Adj Comcast.Adj Charter.Adj

1 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000

13 0.0000 0.6059 0.0226 0.1892 0.1823 0.0000

25 0.0250 0.2288 0.1567 0.2494 0.0605 0.2796

**37 0.0708 0.0166 0.2501 0.1810 0.0000 0.4816**

50 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000

Target Returns and Risks:

mean Cov CVaR VaR

1 0.0059 0.0449 0.0913 0.0526

13 0.0127 0.0324 0.0565 0.0273

25 0.0194 0.0333 0.0413 0.0308

**37 0.0261 0.0417 0.0410 0.0317**

50 0.0335 0.1106 0.1929 0.1204

**I then used the weights to calculate the net returns**

budget <- 1000

## suppose to have a budget of $V at the begin of the l months (with the same value of l previously defined in the forecasting) and decide how to invest your budget to create your portoflio.

peso\_amazon <- 0.0216

peso\_walmart <- 0.3316

peso\_facebook <- 0.1184

peso\_google <- 0.2423

peso\_comcast <- 0.0617

peso\_charter <- 0.2245

transaction\_cost <- 15 #1.5% transaction cost

transazioni <- 6 #Compro le stock una volta

## compute your return (you must consider transaction costs) at the end of the period according to your investment decisions

ritorni\_netti <- sum((sum(tail(Amazon.adj.ret, minimo \* l)) \* peso\_amazon \* budget),

(sum(tail(Walmart.adj.ret, minimo \* l)) \* peso\_walmart \* budget),

(sum(tail(Facebook.adj.ret, minimo \* l)) \* peso\_facebook \* budget),

(sum(tail(Google.adj.ret, minimo \* l)) \* peso\_google \* budget),

(sum(tail(Comcast.adj.ret, minimo \* l)) \* peso\_comcast \* budget),

(sum(tail(Charter.adj.ret, minimo \* l)) \* peso\_charter \* budget),

-transaction\_cost \* transazioni)

##> ritorni\_netti

##[1] 35.27504 = 3.527504%

Which netted a 3.527504% return.

## Conclusions

NOTE:

Predictive Analysis: I didn’t validate the model by running the portfolio simulation.

Beta Computation: I didn’t calculate the expected return of the stocks based on the beta value

Facebook, Google and Amazon are very interesting stocks, with huge growth and interesting swings, as now leading the growth of the S&P500.

Walmart is the least volatile stock, might be useful to reduce risk.

Comcast and Charter seem to be moving at unison