RV

Collaborative project

4/9/2021

```
library(tidyquant)
```

```
## Loading required package: lubridate

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

## Loading required package: PerformanceAnalytics

## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

##
## Attaching package: 'PerformanceAnalytics'

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

## Loading required package: quantmod

## Loading required package: TTR

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

## == Need to Learn tidyquant? ====================================================
## Business Science offers a 1-hour course - Learning Lab #9: Performance Analysis & Portfolio Optimiza
## </> Learn more at: https://university.business-science.io/p/learning-labs-pro </>
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.6
## v tidyr   1.1.3     v stringr 1.4.0
```

```
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ----------------------------------------- tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x dplyr::first()           masks xts::first()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x dplyr::last()            masks xts::last()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

```
head(FANG)
```

```
## # A tibble: 6 x 8
##   symbol date        open  high   low close    volume adjusted
##   <chr>  <date>     <dbl> <dbl> <dbl> <dbl>     <dbl>    <dbl>
## 1 FB     2013-01-02  27.4  28.2  27.4  28    69846400     28
## 2 FB     2013-01-03  27.9  28.5  27.6  27.8  63140600     27.8
## 3 FB     2013-01-04  28.0  28.9  27.8  28.8  72715400     28.8
## 4 FB     2013-01-07  28.7  29.8  28.6  29.4  83781800     29.4
## 5 FB     2013-01-08  29.5  29.6  28.9  29.1  45871300     29.1
## 6 FB     2013-01-09  29.7  30.6  29.5  30.6 104787700     30.6
```

Comments by Martín in italics.

Natalia.

1. *I recommend using the pre-loaded FANG database as it is (with the corresponding length). Here, Natalia downloaded the data and took the first months of the 2021 year. She includes two plots. Are both plots the same? I would recommend adding some comments to understand the plots. Also, remember the objective is to link the volume with the returns as explained in the instructions (readme repo). Martín.*
2. *This is much better Natalia. I understand the value of the trend, it is useful, and it is a good idea to incorporate it. I agree with your contribution. However, the trend in loess reduces some detail which may be important later, when you link volume with stock returns. Ideally, this is where others start contributing to the project.*

Gonzalo.

1. *Interesting piece of evidence. Looks good and it was important to look at this at this moment. However, more questions arise. First, I recommend using ggplot to be consistent with the class and the previous analysis. The ggplot also will allow you to show differences by stocks. Gonzalo took all stocks together and there might be differences between them. There might also be differences by year. There is something else that could bring more contributions on board, this is to compare returns not only with respect to volume but changes in the volume (percentage changes). Just as you do a percentage change in price you can also calculate percentage change in volume. This will allow you to compare both variables under the same units.*

Diana.

1. *I have similar comments as in the case of Gonzalo. First, I recommend using ggplot to be consistent with the class and the previous analysis. The ggplot also will allow you to show differences by stocks. There is something else that could bring more contributions on board, this is to compare returns not only with respect to volume but changes in the volume (percentage changes). Just as you do a percentage*

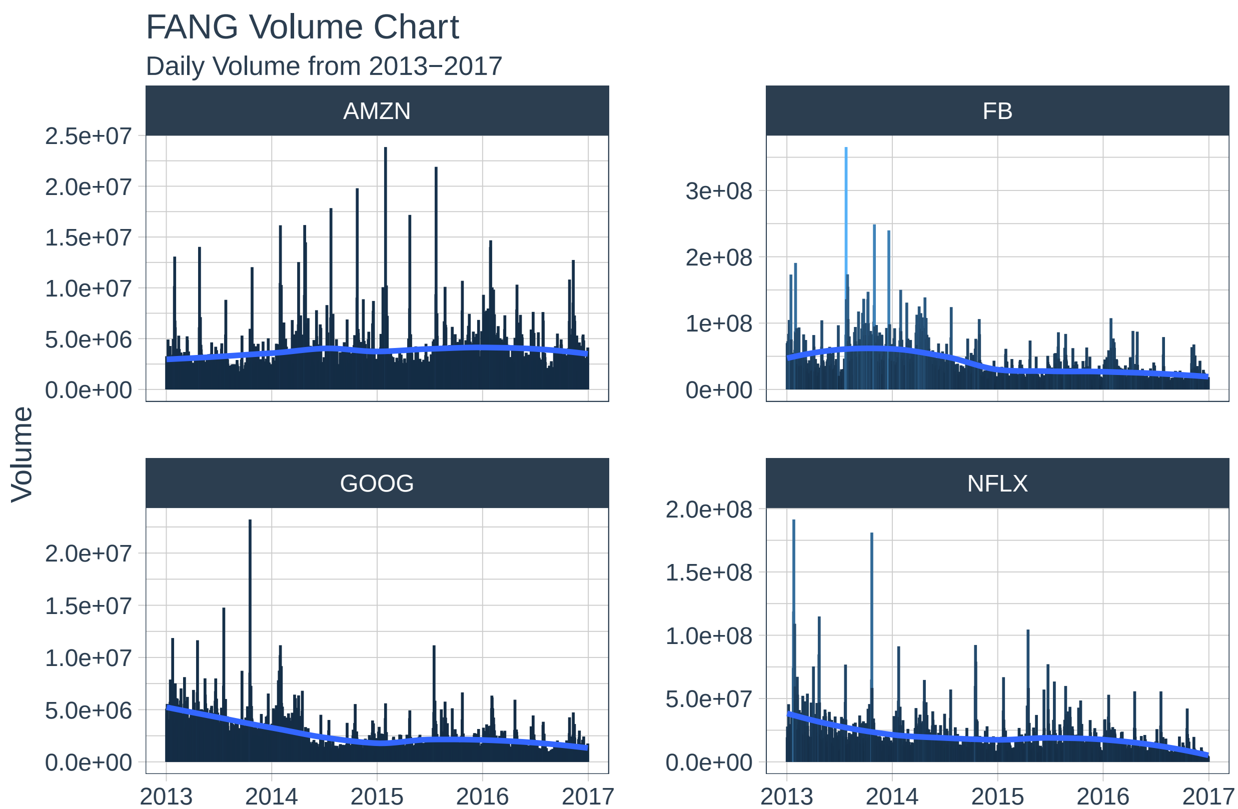
change in price you can also calculate percentage change in volume. This will allow you to compare both variables under the same units.

FANGs Volume in Recent Years.

Looking at volume patterns over time can help get a sense of the conviction behind rises and falls in specific stocks and entire markets, for that reason it is proposed to analyze the volume charts for each FANG stock, in order to understands its trend and then relate it with the stocks' returns.

```
FANG %>%
ggplot(aes(x = date, y = volume, group = symbol)) +
    geom_segment(aes(xend = date, yend = 0, color = volume)) +
    geom_smooth(method = "loess", se = FALSE) +
    labs(title = "FANG Volume Chart",
         subtitle = "Daily Volume from 2013-2017",
         y = "Volume", x = "") +
  facet_wrap(~ symbol, ncol = 2, scale = "free_y") +
  theme_tq() +
    theme(legend.position = "none")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



The chart above shows the daily volumes and their trend line of FANGs stocks in the last few years. The bars represent the number of shares traded daily, and the line represents the trend that the volumes of each share have followed from 2013-2017.

It is clear that to analyze daily volumes, and see their highest and lowest points, it is necessary to have a graph focused on a shorter period of time, in order to obtain more clarity in the data and get more objective interpretations.

Still, trend lines are useful for analyzing data over long periods of time.In the graph, it is observed that FB, NFLX and GOOG stocks' volume has had a negative behavior, meaning that from 2013 to 2017 the average volume of shares decreased. In the case of FB, it is important to highlight that at the beginning it had a positive behavior, but at the end of 2014 it took a downward trend.In contrast, the trend line for AMZN looks stable throughout that period, so we infer that it has maintained a constant average in its volumes.

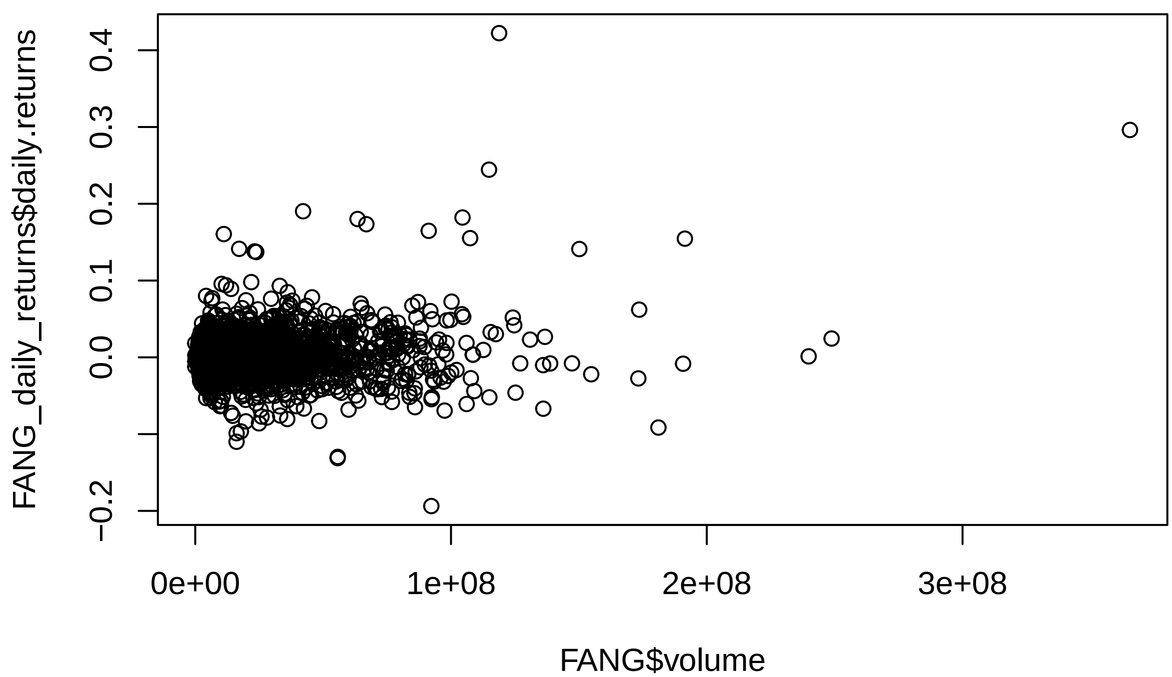
Now, it is proposed to analyze the a graph that shows the volumes by price

correlation between daily volume and daily returns

```
FANG_daily_returns <- FANG %>%
  group_by(symbol) %>%
  tq_transmute(select = adjusted,
               mutate_fun = periodReturn,
               period = "daily",
               type = "arithmetic")
```

```
## Registered S3 method overwritten by 'tune':
##   method                   from
##   required_pkgs.model_spec parsnip
```

```
plot(FANG_daily_returns$daily.returns~FANG$volume)
```



Thanks to the scatter diagram, we can see that the relationship between daily returns and the volume of FANG have an independent and non-linear relationship. We can interpret it this way because their correlation coefficient is very close to zero (0.1356115) and its diagram does not indicate any curves. This coefficient can be seen below:

```
cor.test(FANG_daily_returns$daily.returns, FANG$volume)
```

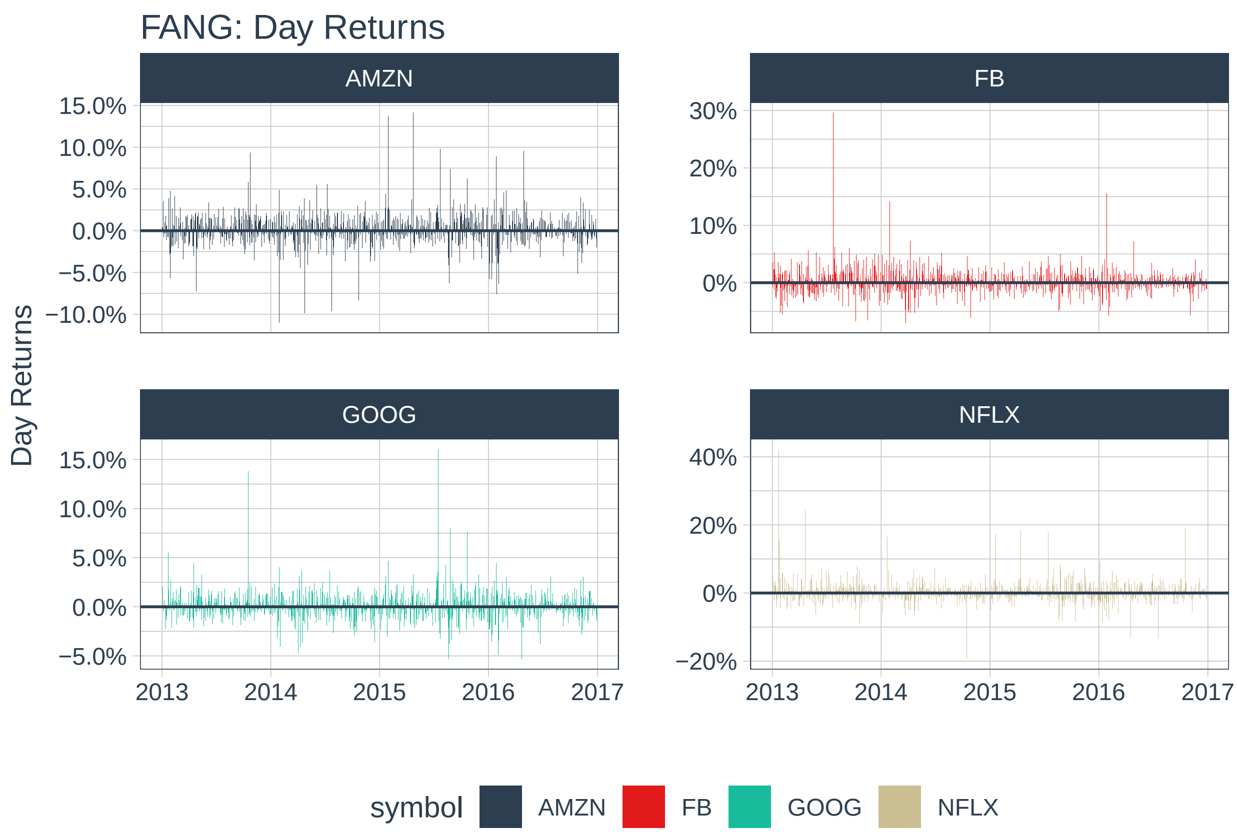
```
##
##  Pearson's product-moment correlation
##
## data:  FANG_daily_returns$daily.returns and FANG$volume
## t = 8.6892, df = 4030, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1051836 0.1657857
## sample estimates:
##       cor
## 0.1356115
```

Thus, we could conclude that the return of a stock has not relationship with its volume in stock market; these variables are independent.

Does anyone disagree? or someone to improve my conclusion

```
FANG_daily_returns <- FANG %>%
  group_by(symbol) %>%
  tq_transmute(select = adjusted,
               mutate_fun = periodReturn,
               period = "daily",
               type = "arithmetic",
               date = "2015-12-30" )
```

```
FANG_daily_returns %>%
ggplot(aes(x = date-1, y = daily.returns, fill = symbol)) + geom_col() +
geom_hline(yintercept = 0, color = palette_light()[[1]]) + scale_y_continuous(labels = scales::percent)
labs(title = "FANG: Day Returns",
         y = "Day Returns", x = "") +
    facet_wrap(~ symbol, ncol = 2, scales = "free_y") +
    theme_tq() + scale_fill_tq()
```




```
Netflix_daily_returns  <- FANG_daily_returns %>%
  mutate(day = as.factor(day(date))) %>%
  filter(symbol == "NFLX")

Netflix_volume <- FANG %>%
  filter(symbol == "NFLX")
Netflix_daily_returns
```

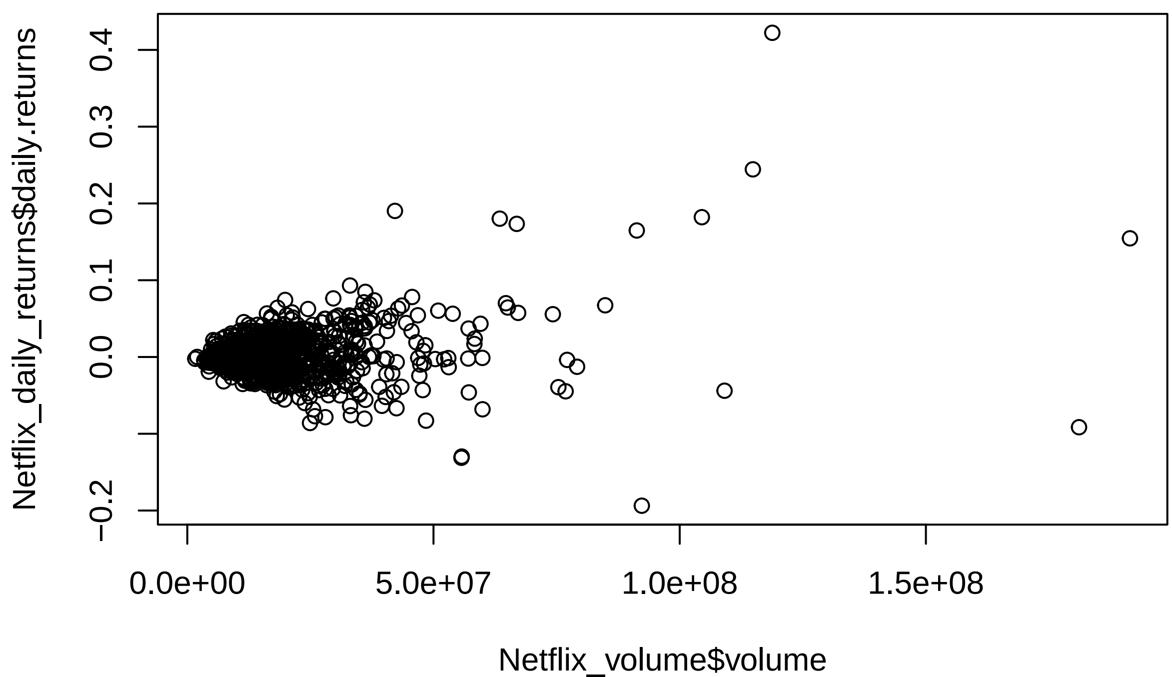
```
## # A tibble: 1,008 x 4
## # Groups:   symbol [1]
##    symbol date       daily.returns day
##    <chr>  <date>             <dbl> <fct>
##  1 NFLX   2013-01-02       0       2
##  2 NFLX   2013-01-03       0.0498  3
##  3 NFLX   2013-01-04      -0.00632 4
##  4 NFLX   2013-01-07       0.0335  7
##  5 NFLX   2013-01-08      -0.0206  8
##  6 NFLX   2013-01-09      -0.0129  9
##  7 NFLX   2013-01-10       0.0218  10
##  8 NFLX   2013-01-11       0.0336  11
##  9 NFLX   2013-01-14       0.0213  14
## 10 NFLX   2013-01-15      -0.0170  15
## # ... with 998 more rows
```

```
Netflix_volume
```

```
## # A tibble: 1,008 x 8
```

```
##    symbol date        open  high   low close   volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>    <dbl>    <dbl>
##  1 NFLX   2013-01-02  95.2  95.8  90.7  92.0 19431300     13.1
##  2 NFLX   2013-01-03  92.0  97.9  91.5  96.6 27912500     13.8
##  3 NFLX   2013-01-04  96.5  97.7  95.5  96.0 17761100     13.7
##  4 NFLX   2013-01-07  96.4 102.   96.1  99.2 45550400     14.2
##  5 NFLX   2013-01-08 100.  101.   96.8  97.2 24714900     13.9
##  6 NFLX   2013-01-09  97.1  97.9  94.6  95.9 20223000     13.7
##  7 NFLX   2013-01-10  96.6  99.9  95.7  98   26117700     14
##  8 NFLX   2013-01-11  98.2 102.   98   101.  29851500     14.5
##  9 NFLX   2013-01-14 101.  105.  101.  103.  23473100     14.8
## 10 NFLX   2013-01-15 103.  104.  101.  102.  17068100     14.5
## # ... with 998 more rows
```

```
plot(Netflix_daily_returns$daily.returns~Netflix_volume$volume)
```




```
cor.test(Netflix_daily_returns$daily.returns,Netflix_volume$volume)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Netflix_daily_returns$daily.returns and Netflix_volume$volume
## t = 8.0481, df = 1006, p-value = 2.358e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1870426 0.3030922
## sample estimates:
```

```
##       cor
## 0.2459486
```

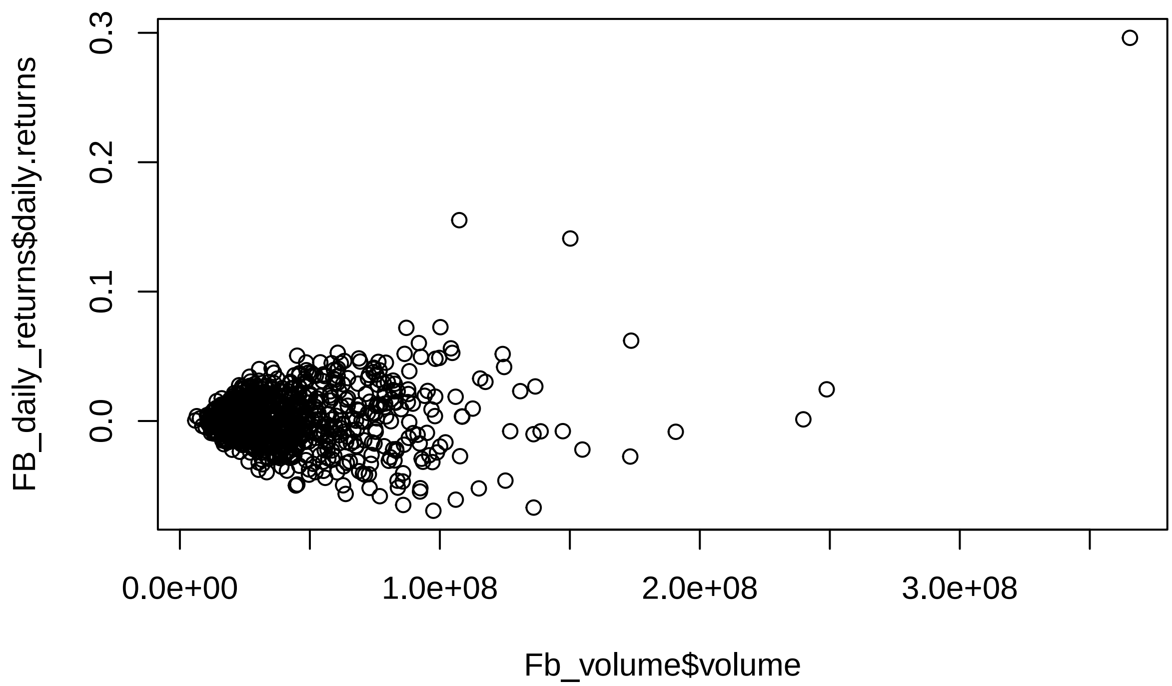
```
FB_daily_returns  <- FANG_daily_returns %>%
  mutate(day = as.factor(day(date))) %>%
  filter(symbol == "FB")

Fb_volume <- FANG %>%
  filter(symbol == "FB")

FB_daily_returns
```

```
## # A tibble: 1,008 x 4
## # Groups:   symbol [1]
##    symbol date       daily.returns day
##    <chr>  <date>             <dbl> <fct>
##  1 FB     2013-01-02       0       2
##  2 FB     2013-01-03      -0.00821 3
##  3 FB     2013-01-04       0.0356  4
##  4 FB     2013-01-07       0.0229  7
##  5 FB     2013-01-08      -0.0122  8
##  6 FB     2013-01-09       0.0526  9
##  7 FB     2013-01-10       0.0232  10
##  8 FB     2013-01-11       0.0134  11
##  9 FB     2013-01-14      -0.0243  14
## 10 FB     2013-01-15      -0.0275  15
## # ... with 998 more rows
```

```
Fb_volume
```

```
## # A tibble: 1,008 x 8
##    symbol date        open  high   low close    volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>     <dbl>    <dbl>
##  1 FB     2013-01-02  27.4  28.2  27.4  28    69846400     28
##  2 FB     2013-01-03  27.9  28.5  27.6  27.8  63140600     27.8
##  3 FB     2013-01-04  28.0  28.9  27.8  28.8  72715400     28.8
##  4 FB     2013-01-07  28.7  29.8  28.6  29.4  83781800     29.4
##  5 FB     2013-01-08  29.5  29.6  28.9  29.1  45871300     29.1
##  6 FB     2013-01-09  29.7  30.6  29.5  30.6 104787700     30.6
##  7 FB     2013-01-10  30.6  31.5  30.3  31.3  95316400     31.3
##  8 FB     2013-01-11  31.3  32.0  31.1  31.7  89598000     31.7
##  9 FB     2013-01-14  32.1  32.2  30.6  31.0  98892800     31.0
## 10 FB     2013-01-15  30.6  31.7  29.9  30.1 173242600     30.1
## # ... with 998 more rows
```

```
plot(FB_daily_returns$daily.returns~Fb_volume$volume)
```




```r
cor.test(FB_daily_returns$daily.returns,Fb_volume$volume)
```

```
##
##  Pearson's product-moment correlation
##
## data:  FB_daily_returns$daily.returns and Fb_volume$volume
## t = 6.8229, df = 1006, p-value = 1.539e-11
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1505105 0.2685616
## sample estimates:
##       cor
## 0.2103025
```

```r
AMZN_daily_returns  <- FANG_daily_returns %>%
  mutate(day = as.factor(day(date))) %>%
  filter(symbol == "AMZN")

AMZN_volume <- FANG %>%
  filter(symbol == "AMZN")

AMZN_daily_returns
```

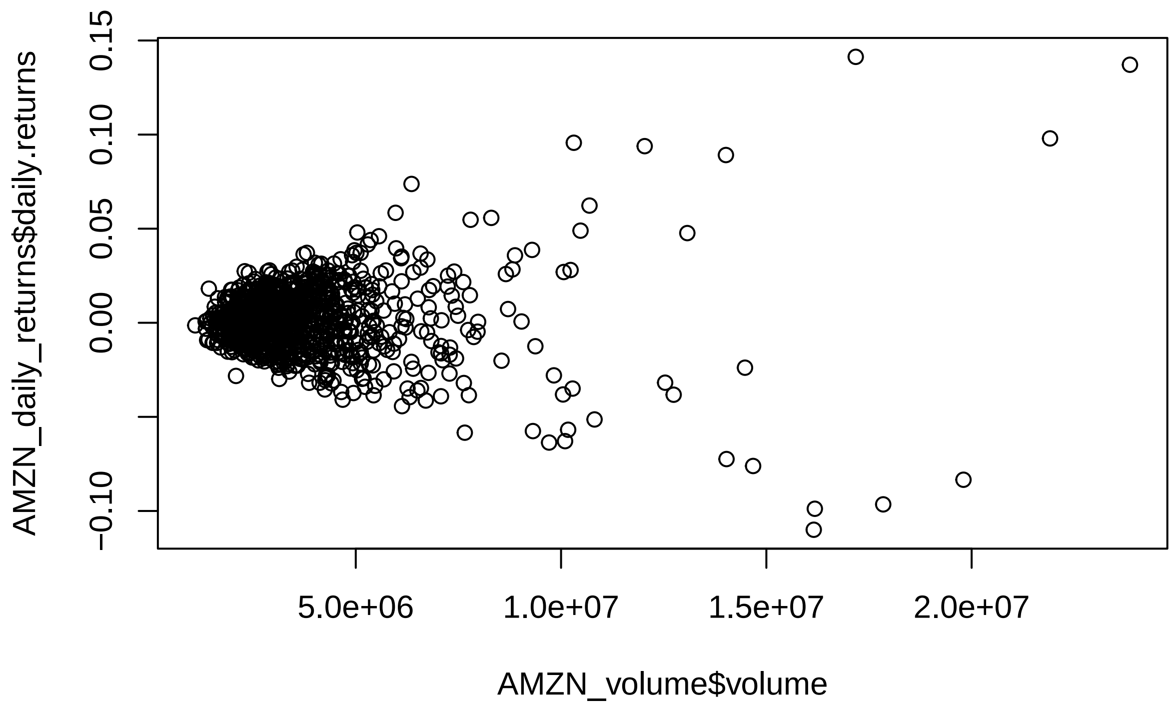
```
## # A tibble: 1,008 x 4
## # Groups:   symbol [1]
##    symbol date       daily.returns day
##    <chr>  <date>             <dbl> <fct>
```

```
##  1 AMZN   2013-01-02      0        2
##  2 AMZN   2013-01-03      0.00455  3
##  3 AMZN   2013-01-04      0.00259  4
##  4 AMZN   2013-01-07      0.0359   7
##  5 AMZN   2013-01-08     -0.00775  8
##  6 AMZN   2013-01-09     -0.000113 9
##  7 AMZN   2013-01-10     -0.00379  10
##  8 AMZN   2013-01-11      0.00980  11
##  9 AMZN   2013-01-14      0.0179   14
## 10 AMZN   2013-01-15     -0.00304  15
## # ... with 998 more rows
```

```
AMZN_volume
```

```
## # A tibble: 1,008 x 8
##    symbol date        open  high   low close  volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>   <dbl>    <dbl>
##  1 AMZN   2013-01-02  256.  258.  253.  257. 3271000     257.
##  2 AMZN   2013-01-03  257.  261.  256.  258. 2750900     258.
##  3 AMZN   2013-01-04  258.  260.  257.  259. 1874200     259.
##  4 AMZN   2013-01-07  263.  270.  263.  268. 4910000     268.
##  5 AMZN   2013-01-08  267.  269.  264.  266. 3010700     266.
##  6 AMZN   2013-01-09  268.  270.  265.  266. 2265600     266.
##  7 AMZN   2013-01-10  269.  269.  262.  265. 2863400     265.
##  8 AMZN   2013-01-11  265.  268.  264.  268. 2413300     268.
##  9 AMZN   2013-01-14  268   274.  268.  273. 4275000     273.
## 10 AMZN   2013-01-15  271.  273.  269.  272. 2326900     272.
## # ... with 998 more rows
```

```
plot(AMZN_daily_returns$daily.returns~AMZN_volume$volume)
```




```
cor.test(AMZN_daily_returns$daily.returns,AMZN_volume$volume)
```

```
##
##  Pearson's product-moment correlation
##
## data:  AMZN_daily_returns$daily.returns and AMZN_volume$volume
## t = 0.49499, df = 1006, p-value = 0.6207
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.04618667  0.07727634
## sample estimates:
##        cor
## 0.01560432
```

```
GOOG_daily_returns  <- FANG_daily_returns %>%
  mutate(day = as.factor(day(date))) %>%
  filter(symbol == "GOOG")

GOOG_volume <- FANG %>%
  filter(symbol == "GOOG")

GOOG_daily_returns
```

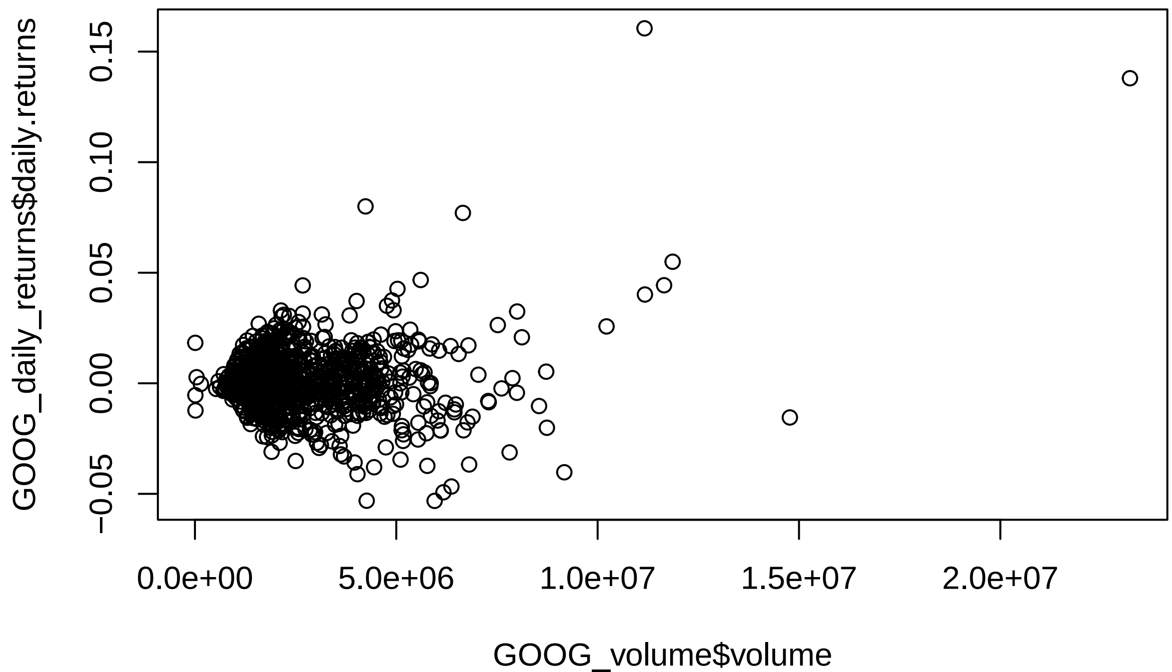
```
## # A tibble: 1,008 x 4
## # Groups:   symbol [1]
##    symbol date       daily.returns day
##    <chr>  <date>             <dbl> <fct>
```

```
##  1 GOOG   2013-01-02      0        2
##  2 GOOG   2013-01-03      0.000581 3
##  3 GOOG   2013-01-04      0.0198   4
##  4 GOOG   2013-01-07     -0.00436  7
##  5 GOOG   2013-01-08     -0.00197  8
##  6 GOOG   2013-01-09      0.00657  9
##  7 GOOG   2013-01-10      0.00455  10
##  8 GOOG   2013-01-11     -0.00201  11
##  9 GOOG   2013-01-14     -0.0226   14
## 10 GOOG   2013-01-15      0.00232  15
## # ... with 998 more rows
```

```
GOOG_volume
```

```
## # A tibble: 1,008 x 8
##    symbol date        open  high   low close  volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>   <dbl>    <dbl>
##  1 GOOG   2013-01-02  719.  727.  717.  723. 5101500     361.
##  2 GOOG   2013-01-03  725.  732.  721.  724. 4653700     361.
##  3 GOOG   2013-01-04  729.  741.  728.  738. 5547600     369.
##  4 GOOG   2013-01-07  735.  739.  731.  735. 3323800     367.
##  5 GOOG   2013-01-08  736.  736.  724.  733. 3364700     366.
##  6 GOOG   2013-01-09  732.  738.  729.  738. 4064500     369.
##  7 GOOG   2013-01-10  743.  745.  734.  741. 3685000     370.
##  8 GOOG   2013-01-11  742.  742.  736.  740. 2579900     370.
##  9 GOOG   2013-01-14  737.  742.  722.  723. 5749200     361.
## 10 GOOG   2013-01-15  719.  735.  712.  725. 7884700     362.
## # ... with 998 more rows
```

```
plot(GOOG_daily_returns$daily.returns~GOOG_volume$volume)
```




```
cor.test(GOOG_daily_returns$daily.returns,GOOG_volume$volume)
```

```
##
##  Pearson's product-moment correlation
##
## data:  GOOG_daily_returns$daily.returns and GOOG_volume$volume
## t = 5.3868, df = 1006, p-value = 8.931e-08
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1067969 0.2268405
## sample estimates:
##       cor
## 0.1674393
```

As we can see, the correlation between the daily return and the daily volume is not significant, because these are very small, and they are not representative. We can see that the smallest correlation is with Amazon, which is 0.0156, we can conclude that the daily return of Amazon is not related to the daily volume. In contrast, Netflix's correlation is the biggest, which is 0.2459. The correlation between the daily return and daily volume of Netflix is more than Amazon's correlation.