Predictions of Stock Vaules using LinkedIn Data

Suren Rathnayake

29 July 2019

Data

The data contain daily records of number of employees of a number of companies (who uses LinkedIn) and the number of LinkedIn followers. The data were obtained from the website https://blog.thedataincubator.com/tag/data-sources/.

Here, I am investigating the predictability of the stock prices based on the number of employees and followers.

```r
library(data.table)
library(tidyverse)
library(tidyr)
library(quantmod)

dlink <- fread("temp_datalab_records_linkedin_company.csv")

summary(dlink)
```

```
##   dataset_id          as_of_date         company_name
##  Length:2426196     Length:2426196     Length:2426196
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##  followers_count    employees_on_platform     link
##  Length:2426196     Length:2426196        Length:2426196
##  Class :character   Class :character      Class :character
##  Mode  :character   Mode  :character      Mode  :character
##    industry          date_added         date_updated
##  Length:2426196     Length:2426196     Length:2426196
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##  description          website          entity_id       cusip
##  Length:2426196     Length:2426196     Mode:logical   Mode:logical
##  Class :character   Class :character   NA's:2426196   NA's:2426196
##  Mode  :character   Mode  :character
##    isin
##  Mode:logical
##  NA's:2426196
##
```

```r
head(dlink)
```

```
##    dataset_id as_of_date           company_name followers_count
## 1:      58329 2015-09-14          Goldman Sachs          552254
## 2:      58329 2015-09-15          Goldman Sachs          552862
## 3:      58363 2015-09-16    United Technologies           59157
## 4:      58366 2015-09-16           Novo Nordisk          336175
## 5:      58371 2015-09-16 Lowe's Companies, Inc.          134255
## 6:      58382 2015-09-16     UnitedHealth Group          221288
##    employees_on_platform                                 link
## 1:                 38124 https://www.linkedin.com/company/1382
```

```
## 2:                 38141 https://www.linkedin.com/company/1382
## 3:                 14982 https://www.linkedin.com/company/2426
## 4:                 26448 https://www.linkedin.com/company/2227
## 5:                 62574 https://www.linkedin.com/company/4128
## 6:                 77108 https://www.linkedin.com/company/1720
##                industry             date_added           date_updated
## 1:     Investment Banking 2015-09-14 00:00:00+00 2015-09-14 00:00:00+00
## 2:     Investment Banking 2015-09-15 00:00:00+00 2015-09-15 00:00:00+00
## 3:   Aviation & Aerospace 2015-09-16 00:00:00+00 2015-09-16 00:00:00+00
## 4:        Pharmaceuticals 2015-09-16 00:00:00+00 2015-09-16 00:00:00+00
## 5:                 Retail 2015-09-16 00:00:00+00 2015-09-16 00:00:00+00
## 6: Hospital & Health Care 2015-09-16 00:00:00+00 2015-09-16 00:00:00+00
##    description website entity_id cusip isin
## 1:                            NA    NA   NA
## 2:                            NA    NA   NA
## 3:                            NA    NA   NA
## 4:                            NA    NA   NA
## 5:                            NA    NA   NA
## 6:                            NA    NA   NA
```

Format the numeric and date columns.

Getting Stock Values Data

The `quantmod` provides a number functions to access and handle stock trading data. The following function a) downloads the data for a given company, b) aligns with the LinkedIn data based on the `added_date`, and c) extracts interesting information for the particular company, as well as those other who are in the same industry.

```
collate_data <- function(dlink, name, sym,
                     src = "yahoo") {

  linked_data <- dlink[company_name == name, .(date_added, company_name,
                                              followers_count, employees_on_platform)]
  # handle duplicates
  vars <- c("employees_on_platform", "followers_count")
  linked_data[, (vars) := lapply(.SD, function(y) as.vector(mean(y))),
                by = date_added, .SDcols = vars]
  linked_data <- linked_data[!duplicated(date_added)]

    date_range <- range(linked_data$date_added)
    getSymbols(sym, src = src, from = date_range[1], to = date_range[2])
  stock_data <- get(sym)

    comm_dates <- linked_data$date_added[linked_data$date_added %in% as.Date(time(stock_data))]

  linked_data <- linked_data[date_added %in% comm_dates]
  stock_data <- stock_data[as.Date(time(stock_data)) %in% comm_dates]

  # "Open"     "High"     "Low"      "Close"    "Volume" "Adjusted"
  dt <- data.table(employees_on_platform = linked_data$employees_on_platform,
                                      followers_count = linked_data$followers_count,
                                      stock_close = as.numeric(stock_data[, 4]),
                                      date_added = comm_dates)
```

```
  cindustry <- unique(dlink[company_name == name]$industry)
  cindustry <- cindustry[!cindustry %in% ""]
  ccompanies <- unique(dlink[industry %in% cindustry]$company_name)

  ind_data <- dlink[company_name %in% ccompanies, .(date_added, company_name,
                                                   followers_count, employees_on_platform)]

  # dplyer seems to be more conveniet here
  ind_data <- ind_data %>% group_by(date_added, company_name) %>%
         mutate(ind_followers_count = mean(followers_count),
                ind_employees_on_platform = mean(employees_on_platform)) %>%
         group_by(date_added) %>%
         summarise(ind_followers_count = sum(ind_followers_count),
                ind_employees_on_platform = sum(ind_employees_on_platform))
  setDT(ind_data)

  ind_data <- ind_data[!duplicated(date_added)]

  setkey(dt, date_added)
  setkey(ind_data, date_added)
  dt <- merge(dt, ind_data, by = "date_added")
  dt
}
```

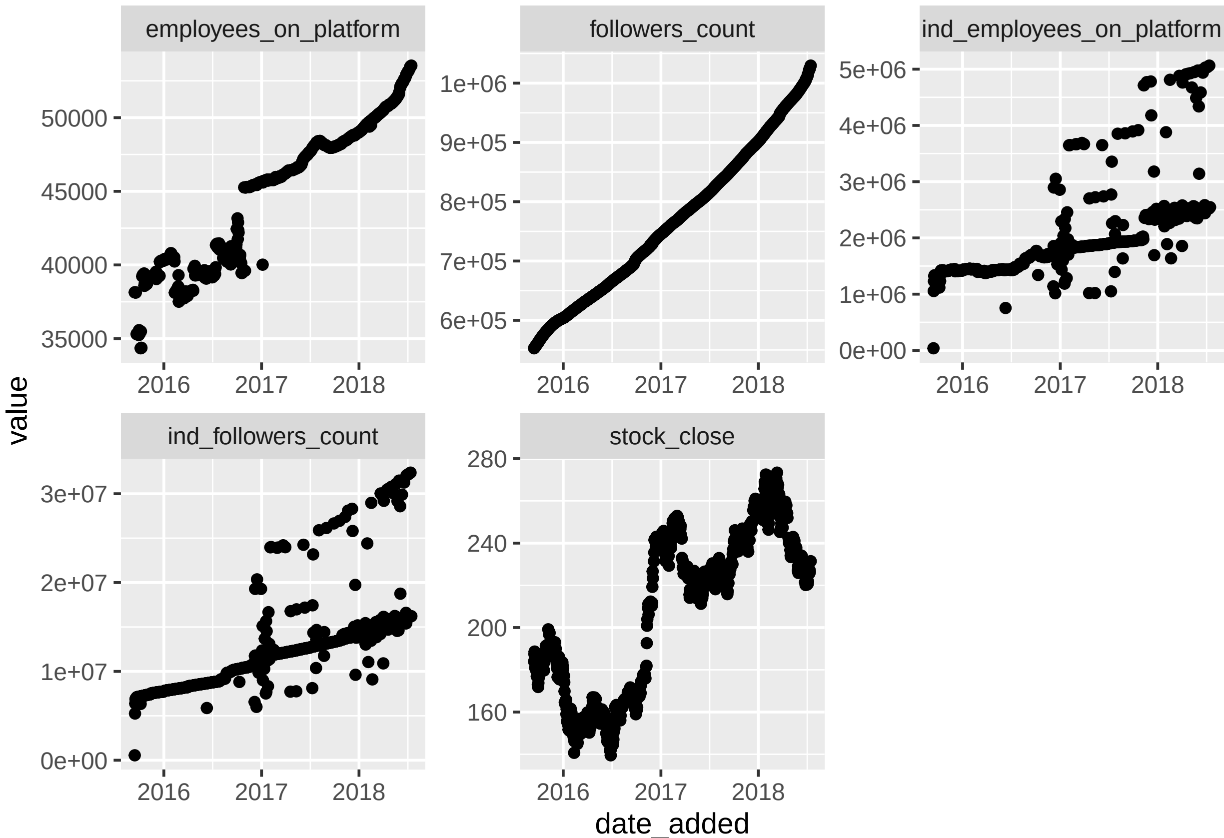
Goldman Sachs

Here we look at the Goldman Sachs company.

```
dt <- collate_data(dlink, name = "Goldman Sachs", sym = "GS")
```

```
## '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.
```

```
tall_df <- dt %>% gather(type, value, -date_added)
ggplot(tall_df, aes(x = date_added, y = value)) + geom_point() +
  facet_wrap(. ~ type, scales="free")
```

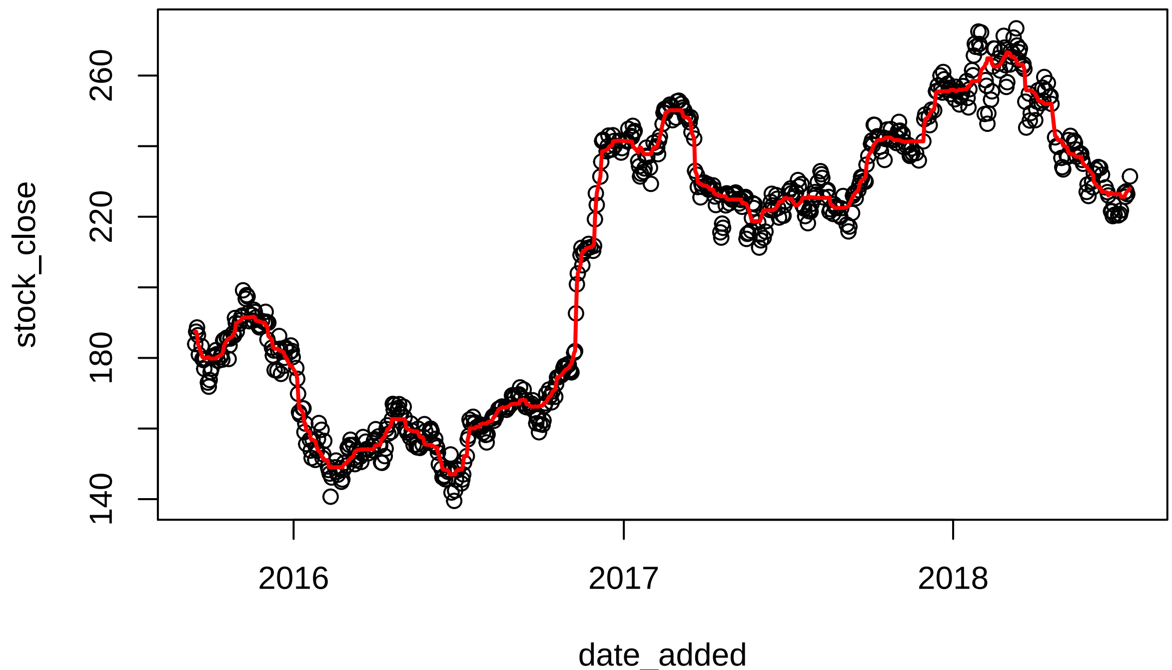


There are a large amount of explainable deviations in the data. For example, it is difficult to believe some of the changes in the number of employee in LinkedIn over a short period of time. Let's try a median filter.

```
median_filter <- function(x, n = 21){runmed(x, n)}
# filter on stock closing data
dt[, plot(stock_close ~ date_added)]
```

```
## NULL
```

```
dt[, points(median_filter(stock_close) ~ date_added, type = "l", lwd = 2, col = "red")]
```

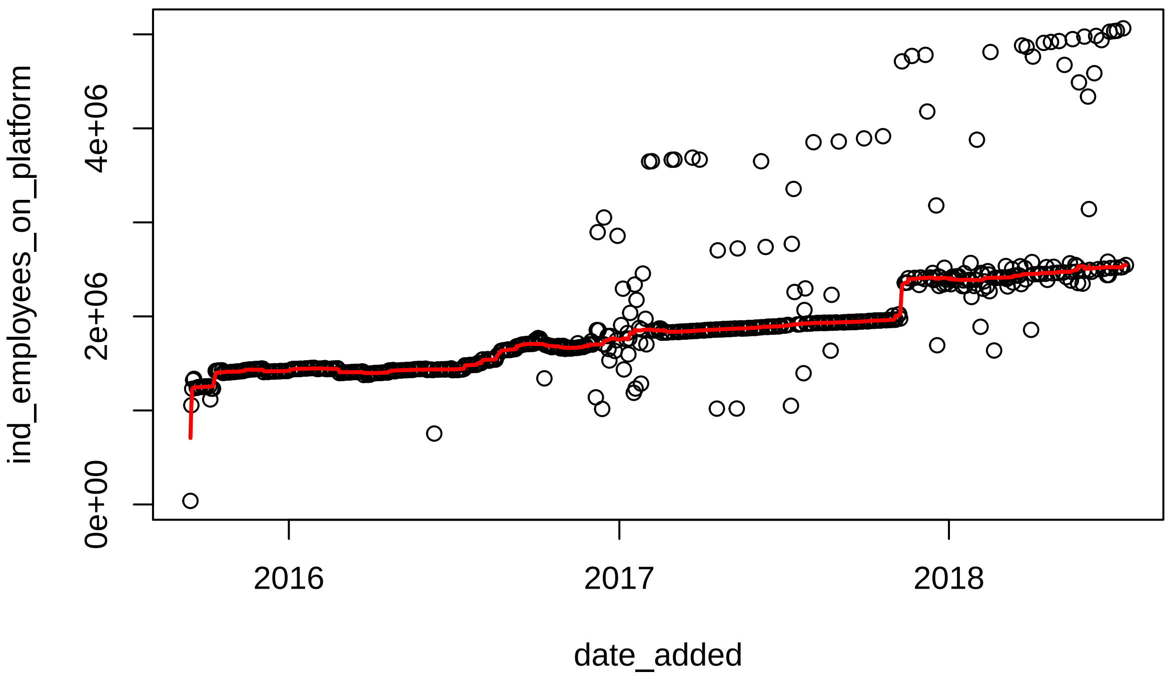



```
## NULL
```

```r
# filter on total number of employee on comanies in same industry as Golsman Sachs
dt[, plot(ind_employees_on_platform ~ date_added)]
```

```
## NULL
```

```r
dt[, points(median_filter(ind_employees_on_platform) ~ date_added, type = "l", lwd = 2, col = "red")]
```



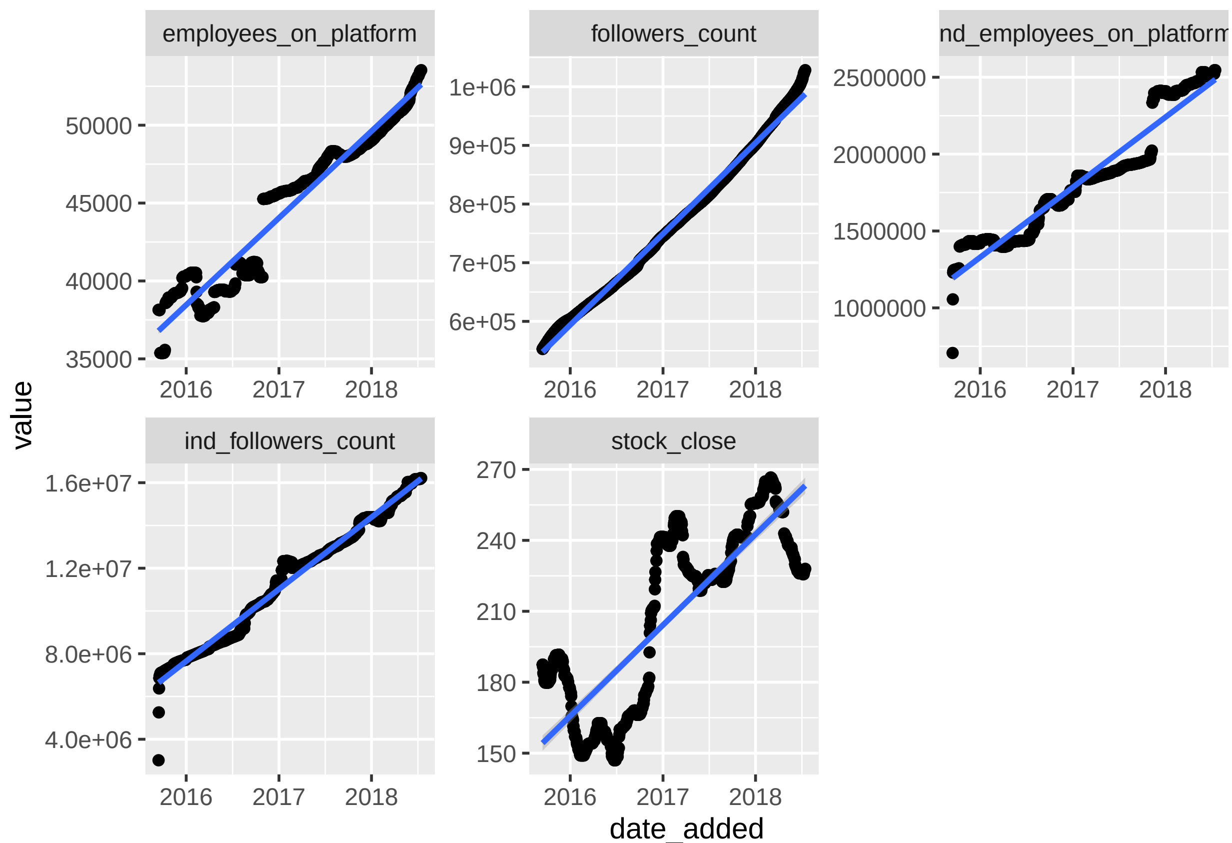

```
## NULL
```

Seems to work fine. Apply the median filter to the data.

```
vars <- colnames(dt)[-1]
dt[, (vars) := lapply(.SD, function (y) as.vector(median_filter(y))), .SDcols = vars]
```

Plot the extracted variables to visualize the data to see a relationship.

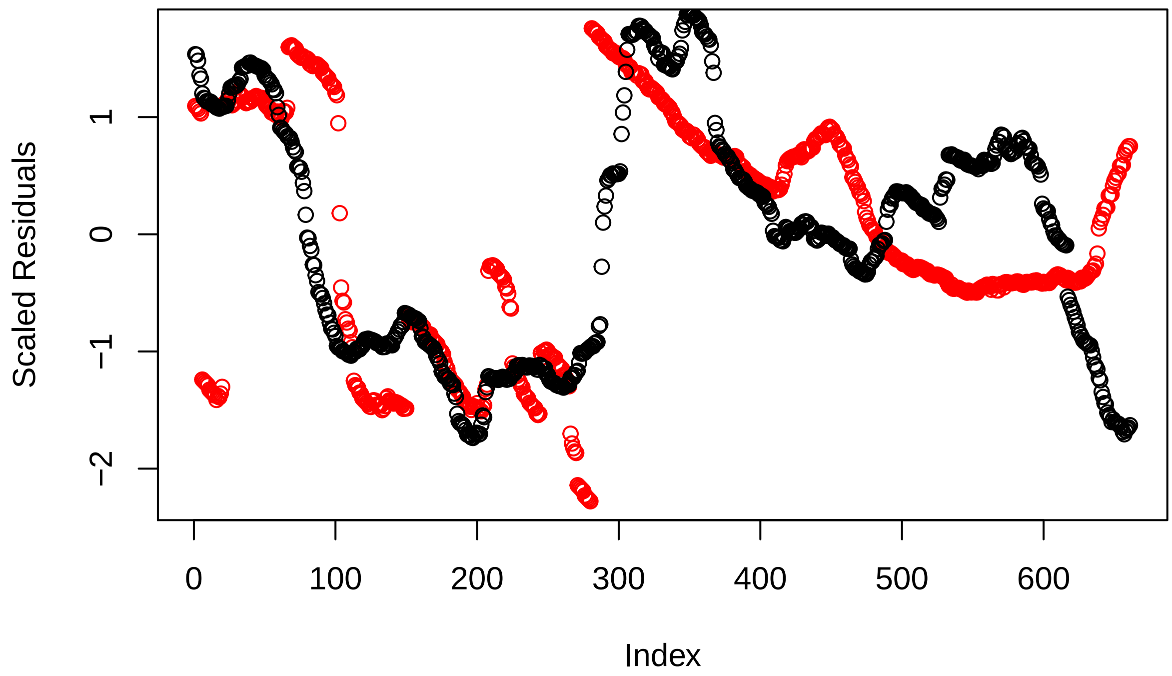
```
tall_df <- dt %>% gather(type, value, -date_added)
ggplot(tall_df, aes(x = date_added, y = value)) + geom_point() +
  geom_smooth(method = "lm") + facet_wrap(. ~ type, scales="free")
```



As everything here increases with time, the stock closing value would show a positive relationship with any of the other four - even if they are meaningless. Let's try to see if the changes in the increase in the number of employees with time can predict the changes in increase in the stock closing value with time.

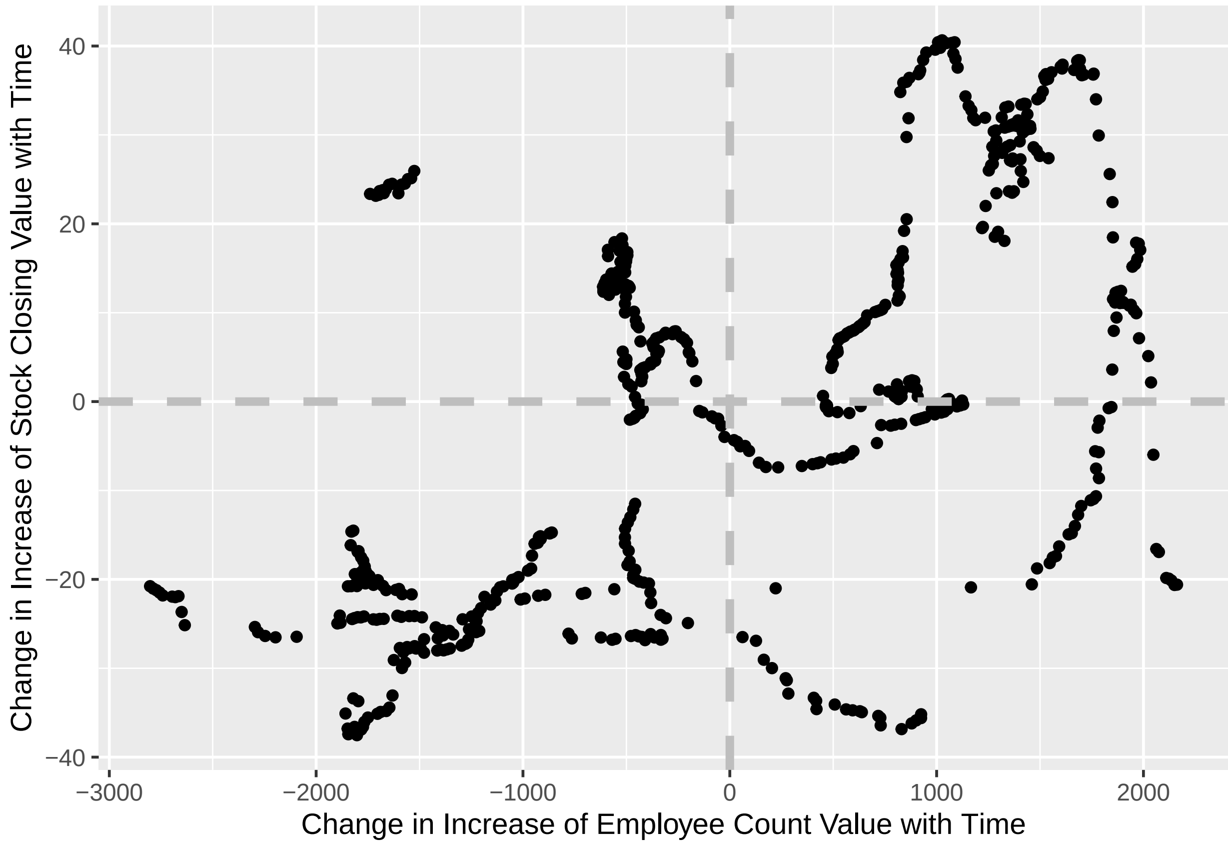
```
fitc <- dt[, lm(stock_close ~ date_added)]
fite <- dt[, lm(employees_on_platform ~ date_added)]

resid <- data.frame(close_resid = fitc$residuals,  emp_resid = fite$residuals)
plot(scale(resid$emp_resid), col="red", ylab = "Scaled Residuals")
points(scale(resid$close_resid))
```



There appear to be relationship in this case - even though there is some time lag is present in it. Quantify the relationship.

```
ggplot(resid, aes(x = emp_resid, y = close_resid)) + geom_point() +
  #geom_smooth(method = "lm", se = FALSE) +
  geom_hline(yintercept=0, linetype="dashed", color = "grey", size = 1.5) +
  geom_vline(xintercept=0, linetype="dashed", color = "grey", size = 1.5)+
  ylab("Change in Increase of Stock Closing Value with Time") +
  xlab("Change in Increase of Employee Count Value with Time")
```



Proportion that would indicate change in the hiring increase would correctly predict the change in increase of the closing value of the stock prices.

```
# proporation of points in +,+ or -, - quadrents
resid %>%
  summarise(prop = sum((close_resid < 0 & emp_resid < 0) | (close_resid > 0 & emp_resid > 0))/ n())
```

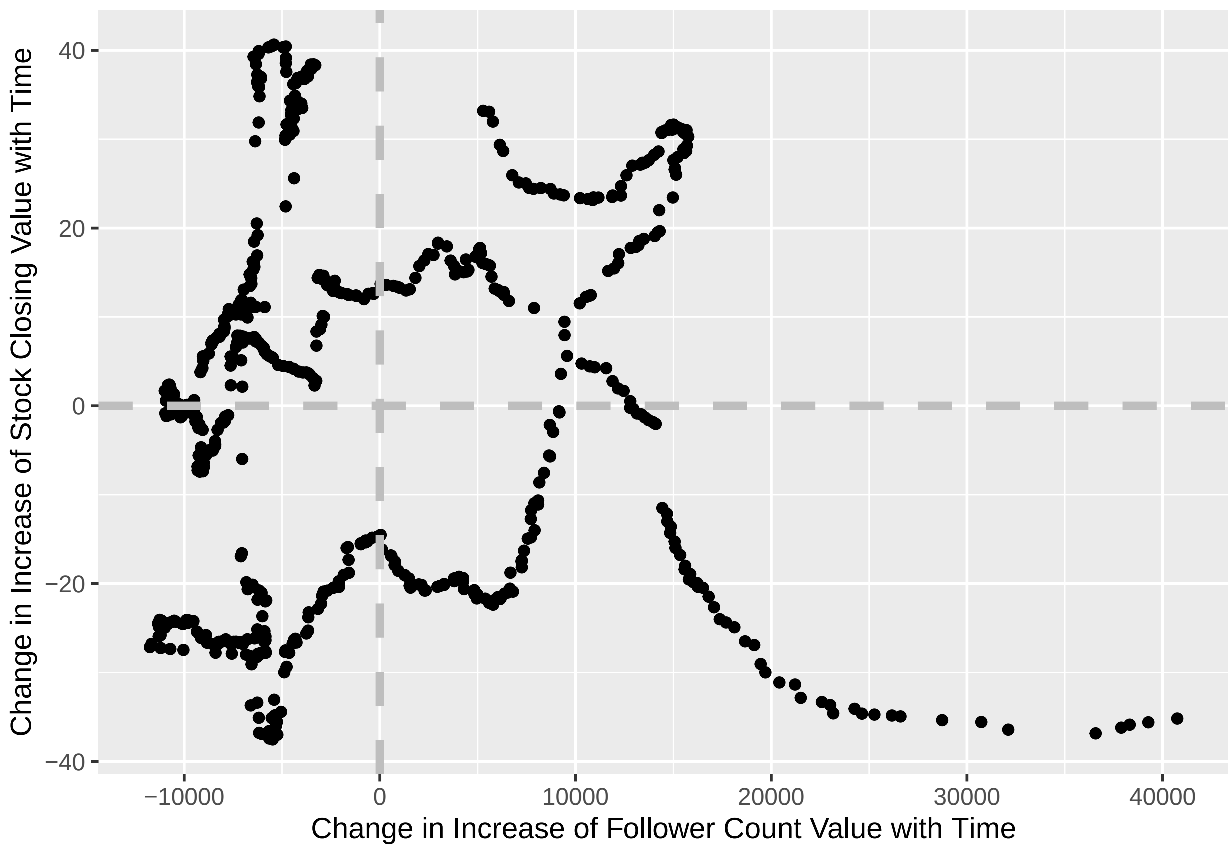
```
##        prop
## 1 0.6429652
```

In this case, around two thirds of the time, changes in employees numbers may indicate the change in stock prices.

Here I consider the change in number of followers values.

```
fite <- dt[, lm(followers_count ~ date_added)]

resid <- data.frame(close_resid = fitc$residuals,  foll_resid = fite$residuals)
ggplot(resid, aes(x = foll_resid, y = close_resid)) + geom_point() +
  #geom_smooth(method = "lm", se = FALSE) +
  geom_hline(yintercept=0, linetype="dashed", color = "grey", size = 1.5) +
  geom_vline(xintercept=0, linetype="dashed", color = "grey", size = 1.5) +
  ylab("Change in Increase of Stock Closing Value with Time") +
  xlab("Change in Increase of Follower Count Value with Time")
```



Proportion that would indicate change in the number of followers indicate the increase would correctly predict the change in increase of the closing value of the stock prices.

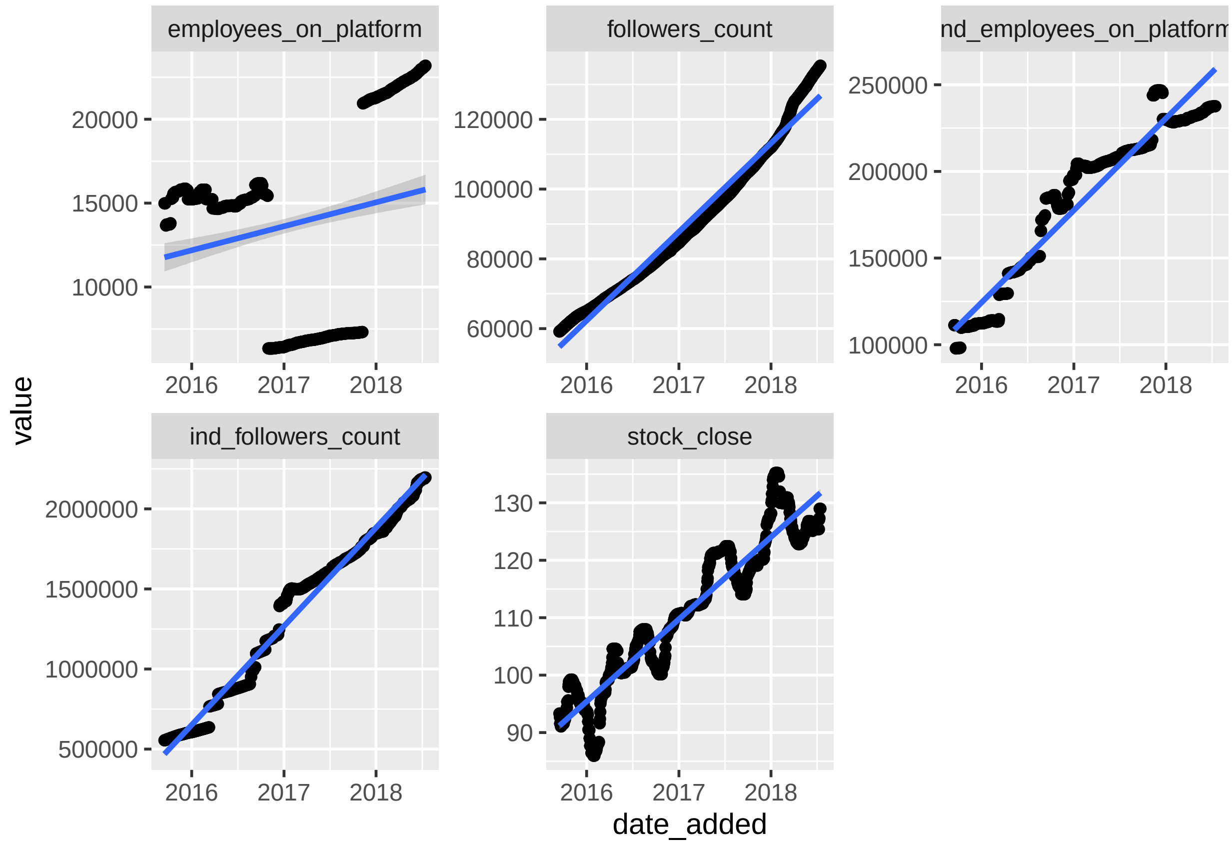
```
# proporation of points in +,+ or -, - quadrents
resid %>%
  summarise(prop = sum((close_resid < 0 & foll_resid < 0) | (close_resid > 0 & foll_resid > 0))/ n())
```

```
##        prop
## 1 0.4871407
```

I have considered various linear fits to the stock values. Although, the effect of linear fit appear to be statistically significant, there actually isn't meaningful relationship.

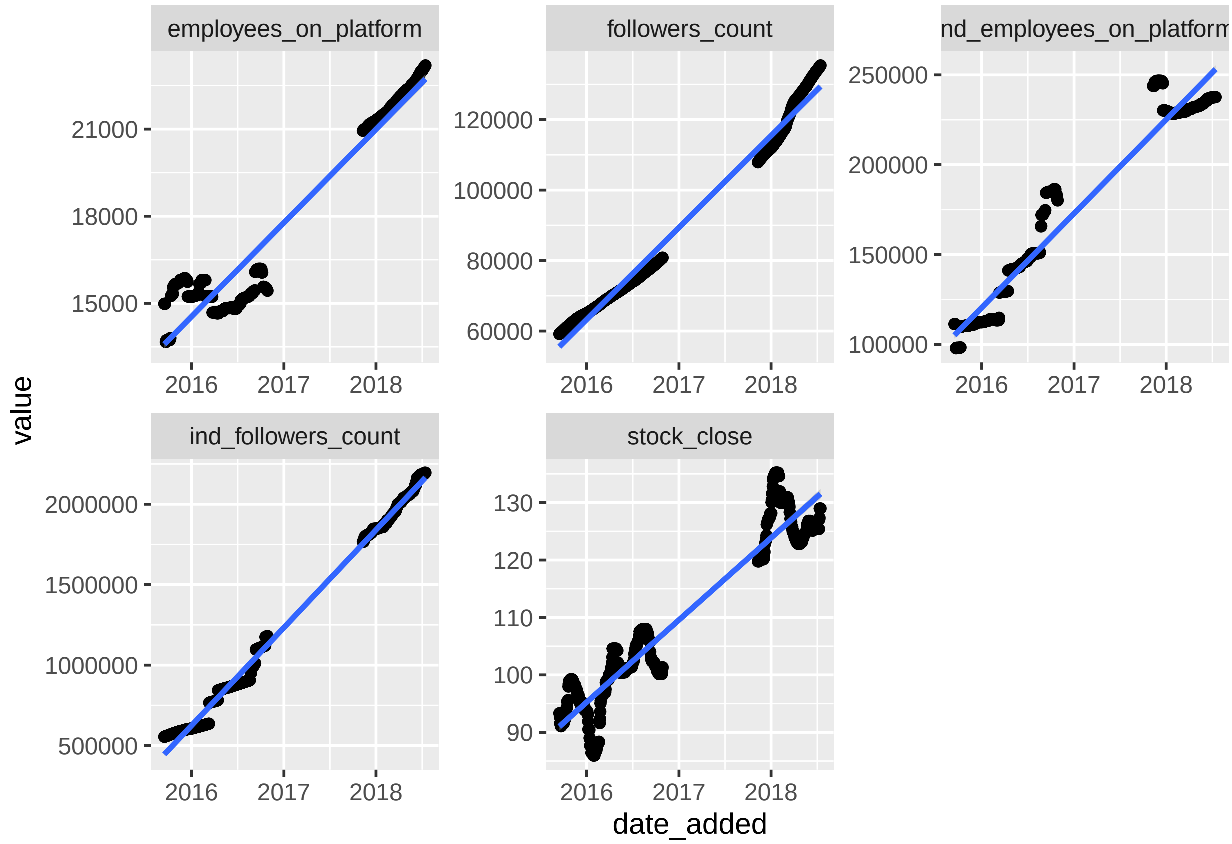
United Technologies Corporation (UTX)

```
# data
dt <- collate_data(dlink, name = "United Technologies", sym = "UTX")
# filter
dt[, (vars) := lapply(.SD, function (y) as.vector(median_filter(y))), .SDcols = vars]
# plot
tall_df <- dt %>% gather(type, value, -date_added)
ggplot(tall_df, aes(x = date_added, y = value)) + geom_point() +
  geom_smooth(method = "lm") + facet_wrap(. ~ type, scales="free")
```



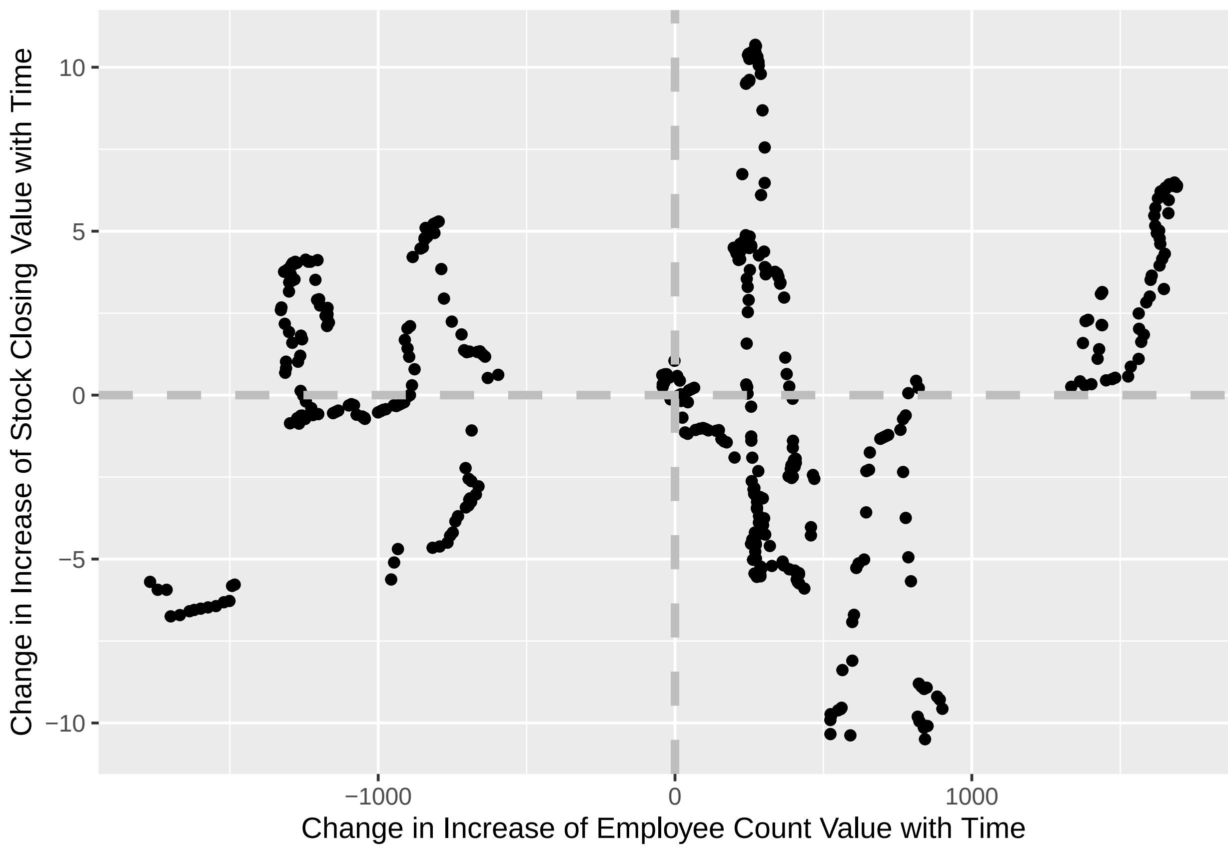
There appears to be some sort of a problem in the data where the total number of employees in LinkedIn has dropped significantly. We remove that part of the data from further analysis.

```
dt <- dt[employees_on_platform > 10000]
tall_df <- dt %>% gather(type, value, -date_added)
ggplot(tall_df, aes(x = date_added, y = value)) + geom_point() +
  geom_smooth(method = "lm") + facet_wrap(. ~ type, scales="free")
```



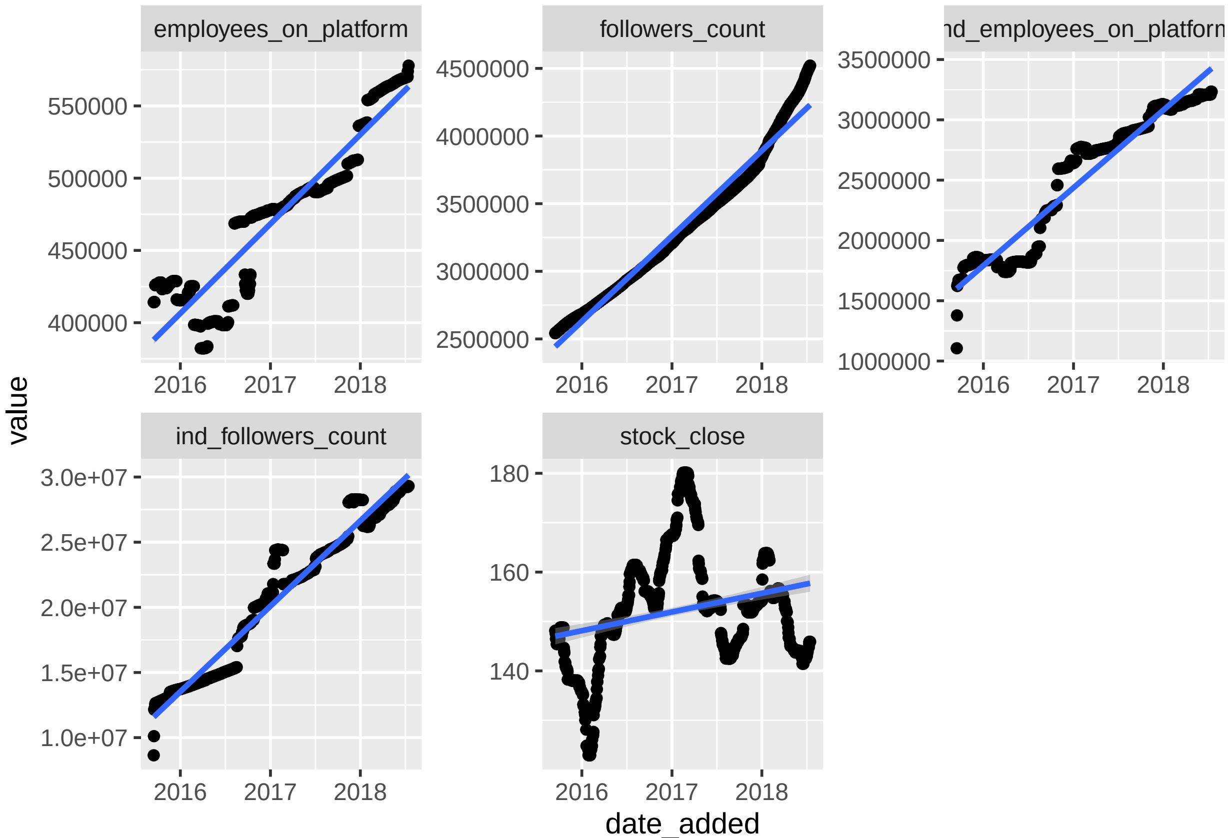

```
fitc <- dt[, lm(stock_close ~ date_added)]
fite <- dt[, lm(employees_on_platform ~ date_added)]

resid <- data.frame(close_resid = fitc$residuals,  emp_resid = fite$residuals)
ggplot(resid, aes(x = emp_resid, y = close_resid)) + geom_point() +
  #geom_smooth(method = "lm", se = FALSE) +
  geom_hline(yintercept=0, linetype="dashed", color = "grey", size = 1.5) +
  geom_vline(xintercept=0, linetype="dashed", color = "grey", size = 1.5)+
  ylab("Change in Increase of Stock Closing Value with Time") +
  xlab("Change in Increase of Employee Count Value with Time")
```



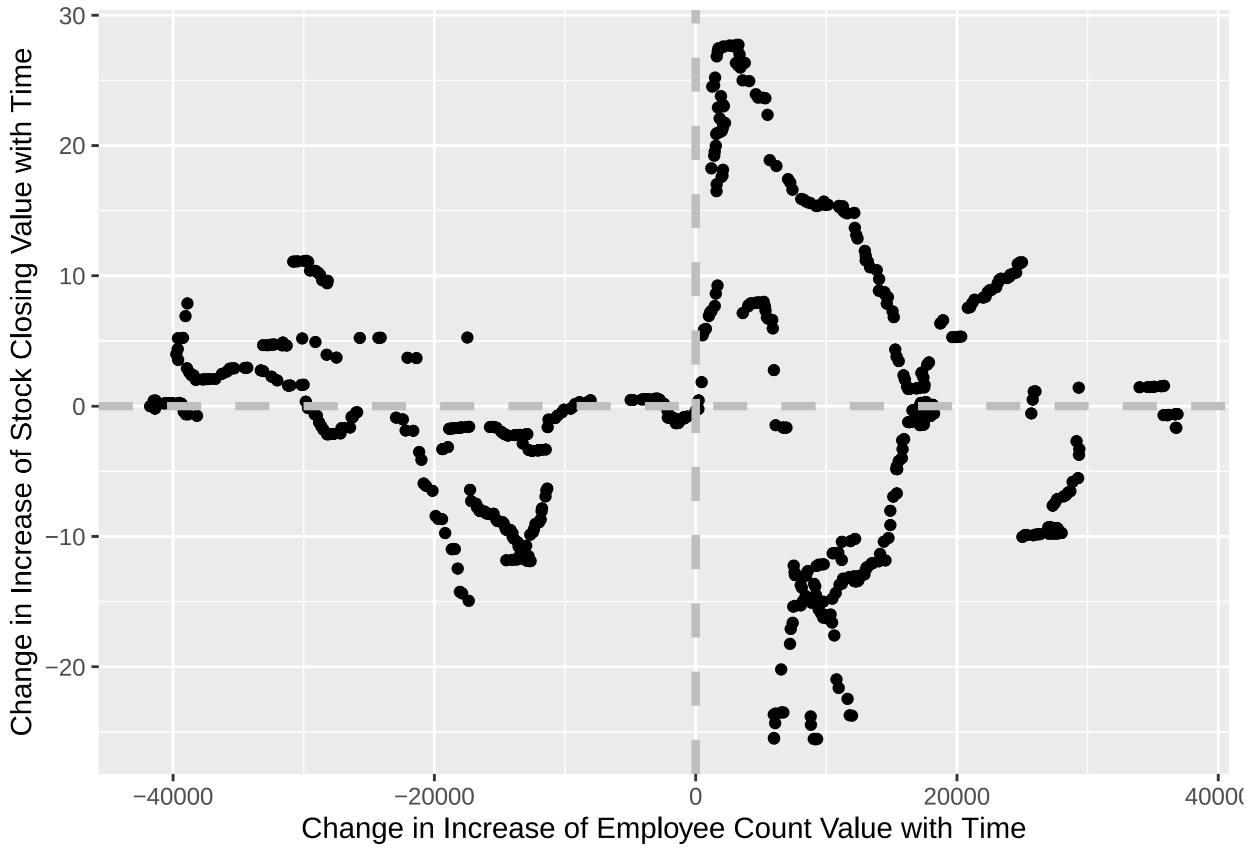
IBM

```
dt <- collate_data(dlink, name = "IBM", sym = "IBM")
# filter
dt[, (vars) := lapply(.SD, function (y) as.vector(median_filter(y))), .SDcols = vars]
# plot
tall_df <- dt %>% gather(type, value, -date_added)
ggplot(tall_df, aes(x = date_added, y = value)) + geom_point() +
  geom_smooth(method = "lm") + facet_wrap(. ~ type, scales="free")
```




```r
fitc <- dt[, lm(stock_close ~ date_added)]
fite <- dt[, lm(employees_on_platform ~ date_added)]

resid <- data.frame(close_resid = fitc$residuals,  emp_resid = fite$residuals)
ggplot(resid, aes(x = emp_resid, y = close_resid)) + geom_point() +
  #geom_smooth(method = "lm", se = FALSE) +
  geom_hline(yintercept=0, linetype="dashed", color = "grey", size = 1.5) +
  geom_vline(xintercept=0, linetype="dashed", color = "grey", size = 1.5)+
  ylab("Change in Increase of Stock Closing Value with Time") +
  xlab("Change in Increase of Employee Count Value with Time")
```



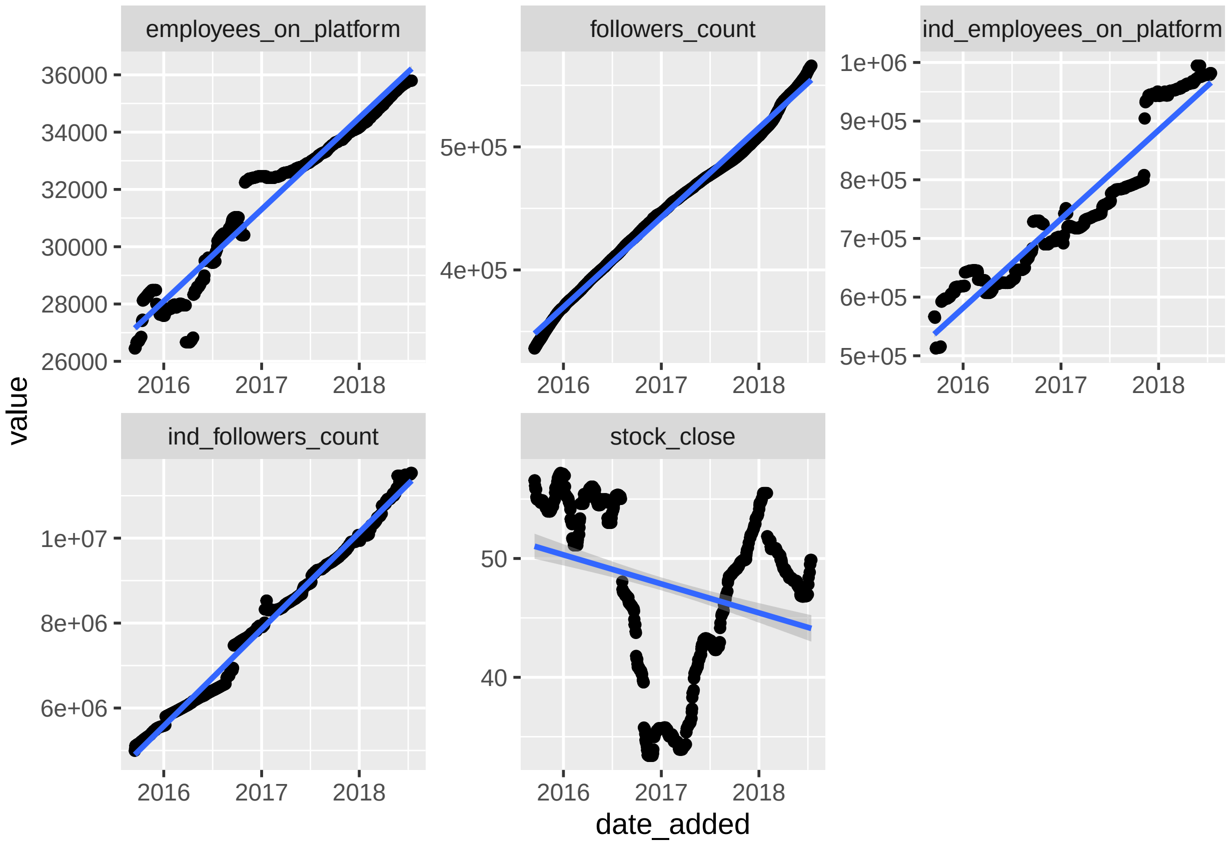
Proportion that would indicate change in the hiring increase would correctly predict the change in increase of the closing value of the stock prices.

```
# proporation of points in +,+ or -, - quadrents
resid %>%
  summarise(prop = sum((close_resid < 0 & emp_resid < 0) | (close_resid > 0 & emp_resid > 0))/ n())
```

```
##         prop
## 1 0.5675266
```

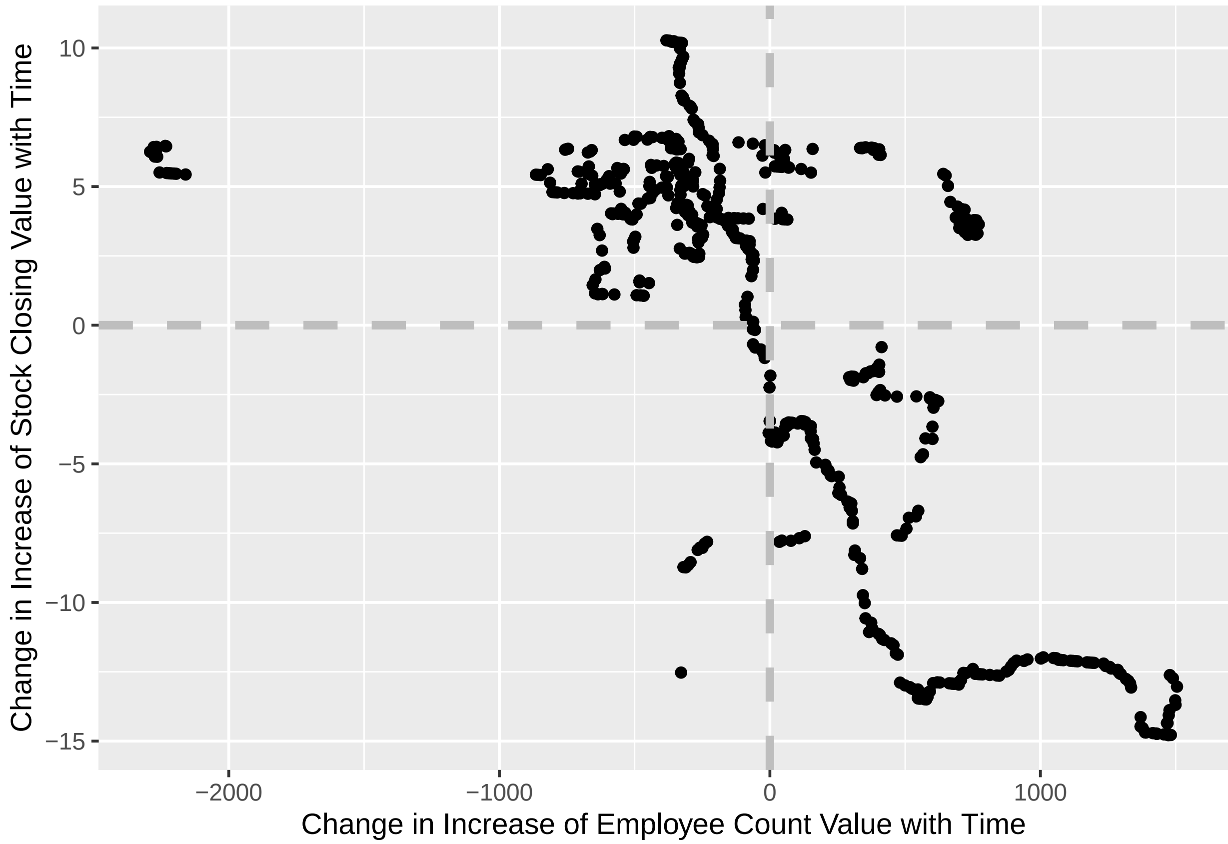
Novo Nordisk

```
dt <- collate_data(dlink, name = "Novo Nordisk", sym = "NVO")
# filter
dt[, (vars) := lapply(.SD, function (y) as.vector(median_filter(y))), .SDcols = vars]
# plot
tall_df <- dt %>% gather(type, value, -date_added)
ggplot(tall_df, aes(x = date_added, y = value)) + geom_point() +
  geom_smooth(method = "lm") + facet_wrap(. ~ type, scales="free")
```




```
fitc <- dt[, lm(stock_close ~ date_added)]
fite <- dt[, lm(employees_on_platform ~ date_added)]

resid <- data.frame(close_resid = fitc$residuals,  emp_resid = fite$residuals)
ggplot(resid, aes(x = emp_resid, y = close_resid)) + geom_point() +
  #geom_smooth(method = "lm", se = FALSE) +
  geom_hline(yintercept=0, linetype="dashed", color = "grey", size = 1.5) +
  geom_vline(xintercept=0, linetype="dashed", color = "grey", size = 1.5)+
  ylab("Change in Increase of Stock Closing Value with Time") +
  xlab("Change in Increase of Employee Count Value with Time")
```



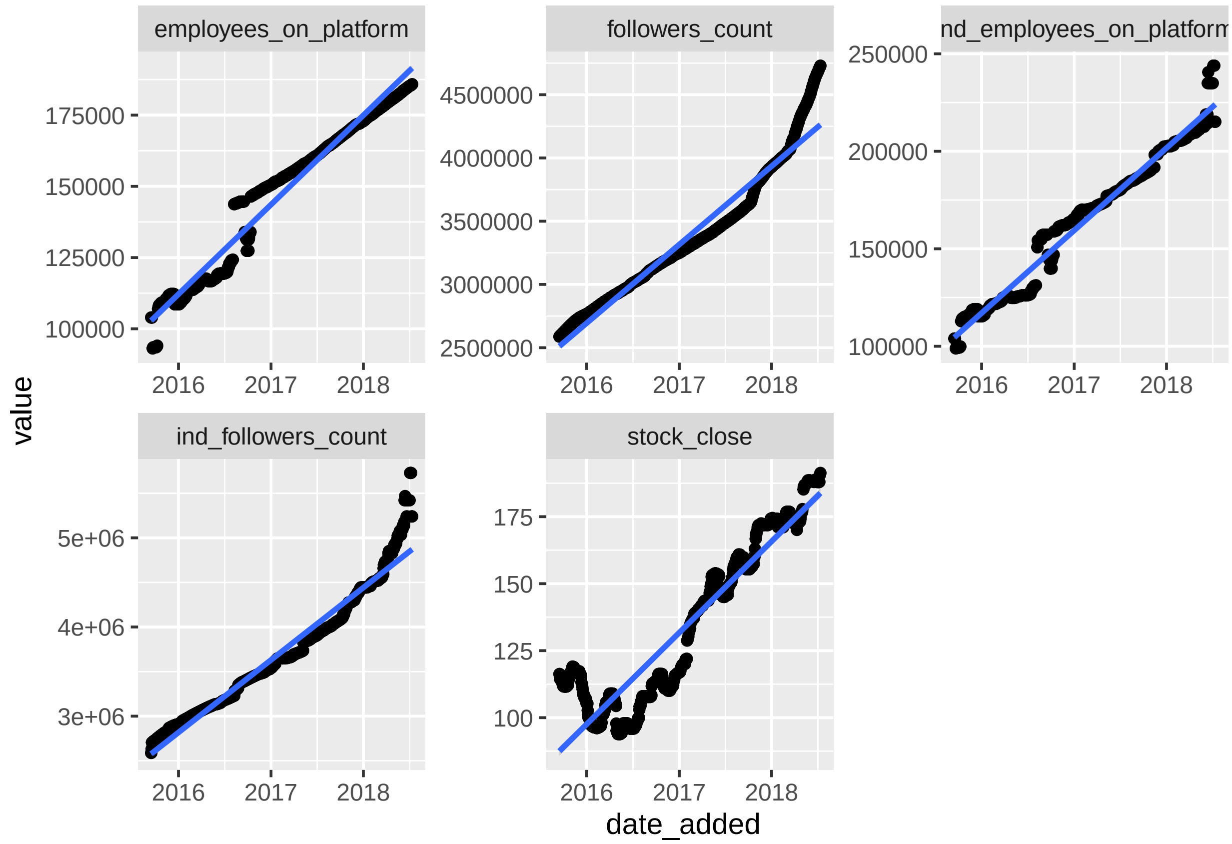
Proportion that would indicate change in the hiring increase would correctly predict the change in increase of the closing value of the stock prices.

```
# proporation of points in +,+ or -, - quadrents
resid %>%
  summarise(prop = sum((close_resid < 0 & emp_resid < 0) | (close_resid > 0 & emp_resid > 0))/ n())
```

```
##        prop
## 1 0.1365706
```

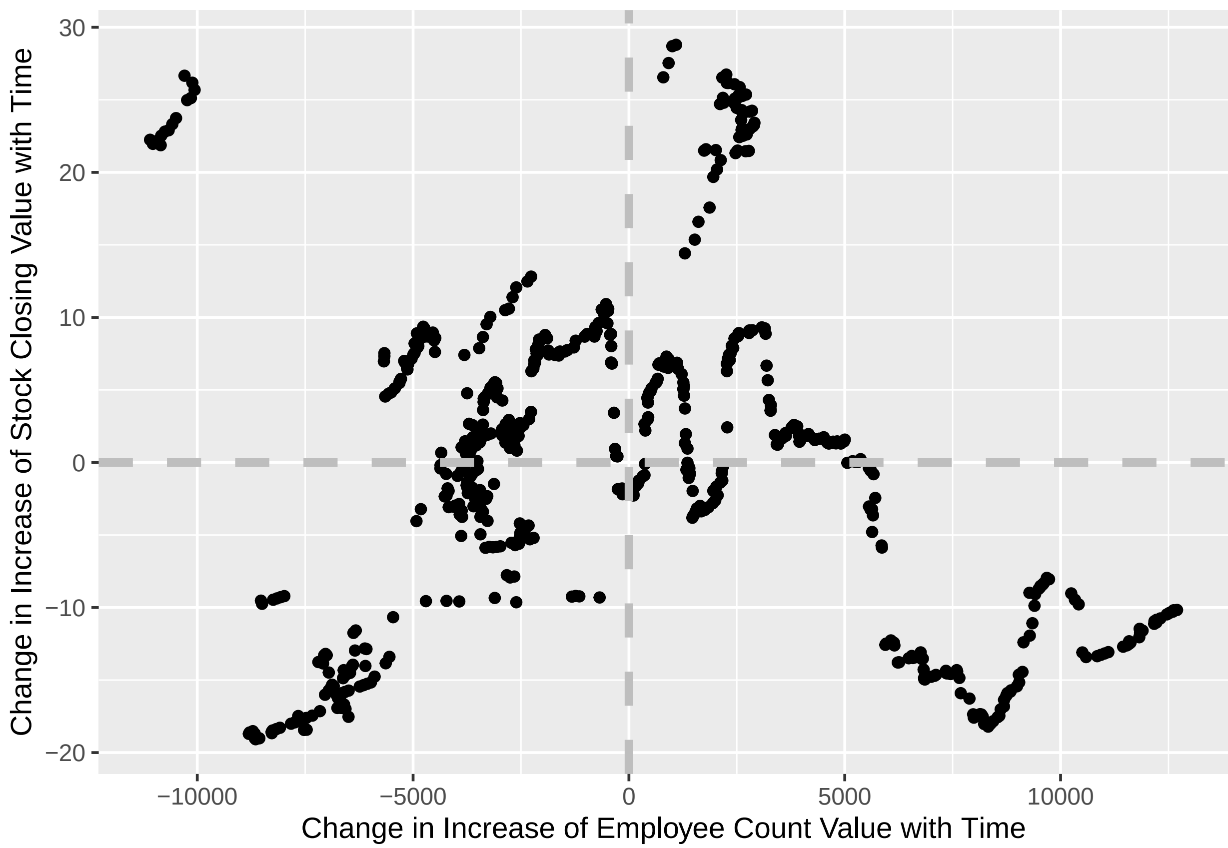
Apple (AAPL)

```
dt <- collate_data(dlink, name = "Apple", sym = "AAPL")
# filter
dt[, (vars) := lapply(.SD, function (y) as.vector(median_filter(y))), .SDcols = vars]
# plot
tall_df <- dt %>% gather(type, value, -date_added)
ggplot(tall_df, aes(x = date_added, y = value)) + geom_point() +
  geom_smooth(method = "lm") + facet_wrap(. ~ type, scales="free")
```




```
fitc <- dt[, lm(stock_close ~ date_added)]
fite <- dt[, lm(employees_on_platform ~ date_added)]

resid <- data.frame(close_resid = fitc$residuals,  emp_resid = fite$residuals)
ggplot(resid, aes(x = emp_resid, y = close_resid)) + geom_point() +
  #geom_smooth(method = "lm", se = FALSE) +
  geom_hline(yintercept=0, linetype="dashed", color = "grey", size = 1.5) +
  geom_vline(xintercept=0, linetype="dashed", color = "grey", size = 1.5)+
  ylab("Change in Increase of Stock Closing Value with Time") +
  xlab("Change in Increase of Employee Count Value with Time")
```



Proportion that would indicate change in the hiring increase would correctly predict the change in increase of the closing value of the stock prices.

```
# proporation of points in +,+ or -, - quadrents
resid %>%
  summarise(prop = sum((close_resid < 0 & emp_resid < 0) | (close_resid > 0 & emp_resid > 0))/ n())
```

```
##        prop
## 1 0.4818731
```

Conclustion

Present analysis was done as an explanatory analysis to investigate the use of LinkedIn data for the prediction of stock prices. From the companies consisted in this analysis, it has shown promising results from those involved in techno local industry. Further analysis, and possibly including other predictors, may reveal in what type companies such prediction can be maid.