DATA 607: Final Project

Cindy Lin

2025-05-03

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

```
library(DBI)
library(RMySQL)
library(stringr)
library(tidyRSS)
library(purrr)
library(tidyquant)
```

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

```
## -- Attaching core tidyquant packages ----------------------- tidyquant 1.0.11 --
## v PerformanceAnalytics 2.0.8      v TTR                0.24.4
## v quantmod             0.4.26     v xts                0.14.1
## -- Conflicts ------------------------------------------ tidyquant_conflicts() --
## x zoo::as.Date()                 masks base::as.Date()
## x zoo::as.Date.numeric()         masks base::as.Date.numeric()
## x dplyr::filter()                masks stats::filter()
## x xts::first()                   masks dplyr::first()
## x dplyr::lag()                   masks stats::lag()
## x xts::last()                    masks dplyr::last()
## x PerformanceAnalytics::legend() masks graphics::legend()
## x quantmod::summary()            masks RMySQL::summary(), base::summary()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(syuzhet)
library(ggplot2)
```

INTRODUCTION

Stock price movement are driven by different forces. It can be fundamental, technical, or even just how the general public perceives the company. For the purpose of this project, we will look at the fundamental which is the company's performance and the public sentiment using the news.

```r
stocks <- c("UBER", "DASH", "Z", "DIS", "CVNA")

get_yahoo_news_filtered <- function(symbol) {
  rss_url <- paste0("https://feeds.finance.yahoo.com/rss/2.0/headline?s=", symbol, "&region=US&lang=en-
  feed <- tidyfeed(rss_url)

  feed <- feed %>%
    filter(

      str_detect(tolower(item_title), fixed(tolower(symbol))) |
      str_detect(tolower(item_description), fixed(tolower(symbol)))
    ) %>%
    mutate(symbol = symbol)
  return(feed)
}

all_news_filtered <- lapply(stocks, get_yahoo_news_filtered) %>%
  bind_rows()
```

```
## GET request successful. Parsing...
##
## GET request successful. Parsing...
##
## GET request successful. Parsing...
##
## GET request successful. Parsing...
##
## GET request successful. Parsing...
```

```r
#lapply runs the function in a vector
#bind_rows put the result into dataframe as one


head(all_news_filtered %>% select(symbol, item_title, symbol))
```

```
## # A tibble: 6 x 2
##   symbol item_title
##   <chr>  <chr>
## 1 UBER   Is Uber Technologies (UBER) the Unstoppable Growth Stock to Invest in ~
## 2 UBER   Checking In on the Latest From Disney and Uber
## 3 UBER   Volkswagen and Uber to test, deploy robotaxis
```

```
## 4 UBER   Uber VS. Lyft Earnings: ETFs in Focus
## 5 UBER   Uber price targets revamped as CEO unveils autonomous-vehicle expansio~
## 6 UBER   Uber is hedging its bets when it comes to robotaxis
```

Since I have five stocks, I need to create a function to find and filter through the news. The idea is that I would continuously pull the news each day prior to the earning report and on the day before earnings, I would generate sentiment on it.

```
news_data <- all_news_filtered %>%
  select(symbol, item_title, item_description, item_pub_date, item_link, item_description) %>%
  mutate(item_title = substr(item_title, 1, 255))



con <- dbConnect(
  RMySQL::MySQL(),
  user = "root",

  dbname = "mydatabase",
  host = "127.0.0.1"
)


  dbWriteTable(con, name = "stock_news", value = news_data, append = TRUE, row.names = FALSE)
```

```
## [1] TRUE
```

I saved the pulled news articles into MYSQL database and then in the news_data table.

```
dbListTables(con)
```

```
## [1] "mytable"    "stock_news"
```

```
news_df <- dbGetQuery(con, "SELECT * FROM stock_news")
```

I pulled from saved articles in the database and pull into R

```
news_df <- news_data %>%
  mutate(full_text = paste(item_title, item_description, sep = ". "))


news_df <- news_df %>%
  rowwise() %>%
  mutate(sentiment = mean(syuzhet::get_sentiment(syuzhet::get_sentences(full_text)), na.rm = TRUE)) %>%
  ungroup()


sentiment_by_stock <- news_df %>%
  group_by(symbol) %>%
  summarise(avg_sentiment = mean(sentiment, na.rm = TRUE))
```

Using Syuzhet sentiment analysis, each stocks was calculated and scored on how positive or negative the news was. Here the most positive was UBER, and the least positive was DIS.

```
earnings <- read.csv("earnings.csv")

sentiment_by_stocks <- read.csv("news_data.csv")

print(earnings)
```

```
##   symbol  EPS EPS_Forecast Percent_Surprise
## 1   UBER 0.83         0.51            62.75
## 2   DASH 0.44         0.40            10.00
## 3      Z 0.03         0.02            50.00
## 4    DIS 1.45         1.18            22.88
## 5   CVNA 1.51         0.75           101.33
```

```
print(sentiment_by_stocks)
```

```
##   symbol avg_sentiment
## 1   CVNA     0.3708333
## 2   DASH     0.5660256
## 3    DIS     0.1455000
## 4   UBER     0.8603646
## 5      Z     0.2500000
```

I have the earning results in a CSV file so I am loading it here.

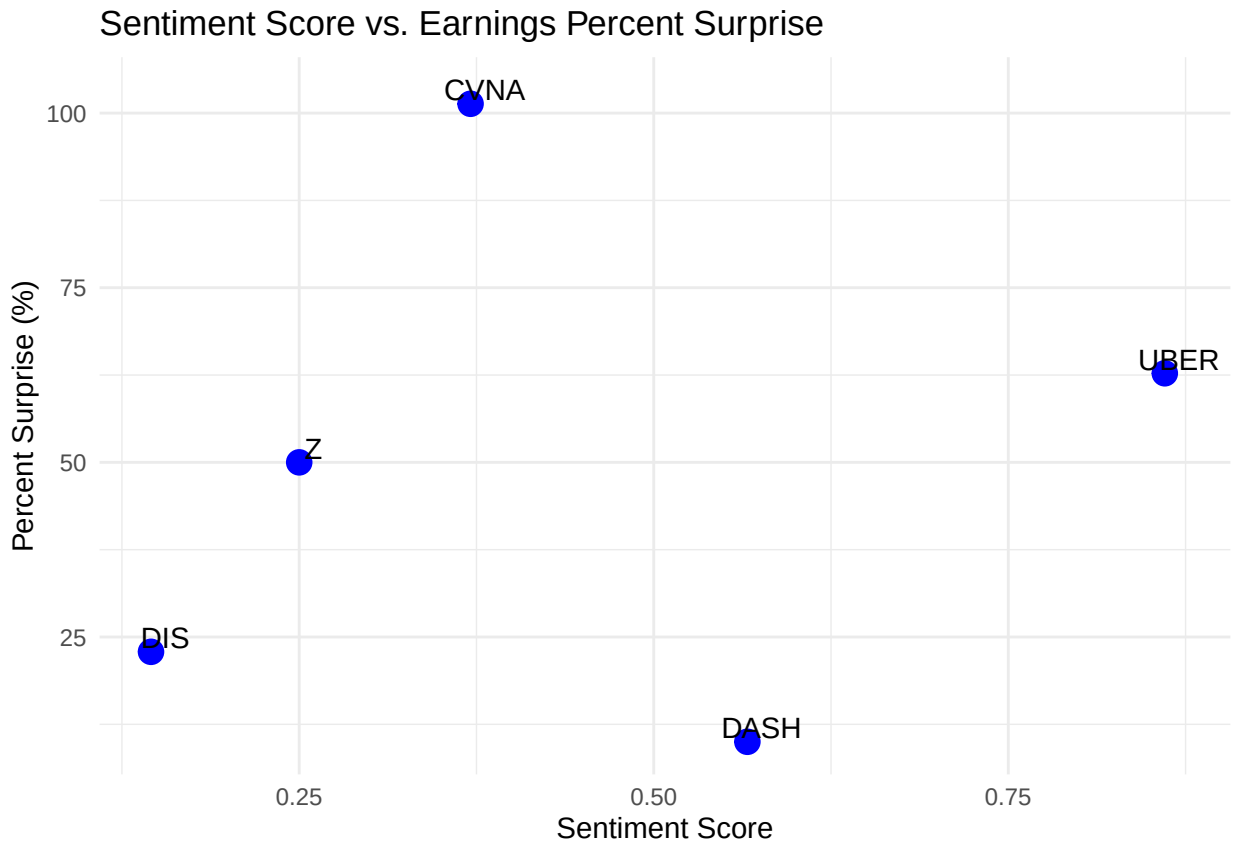
```
combined <- merge(earnings, sentiment_by_stocks, by = "symbol")

combined$earnings_result <- with(combined, ifelse(
  EPS > EPS_Forecast, "beat",
  ifelse(EPS < EPS_Forecast, "miss", "meet")
))

head(combined)
```

```
##   symbol  EPS EPS_Forecast Percent_Surprise avg_sentiment earnings_result
## 1   CVNA 1.51         0.75           101.33     0.3708333            beat
## 2   DASH 0.44         0.40            10.00     0.5660256            beat
## 3    DIS 1.45         1.18            22.88     0.1455000            beat
## 4   UBER 0.83         0.51            62.75     0.8603646            beat
## 5      Z 0.03         0.02            50.00     0.2500000            beat
```

After loading the CSV file with the earning information, I want to know how the stocks performed, if they miss, beat, or meet the expectation of the forecast.

```
library(ggplot2)

ggplot(combined, aes(x = avg_sentiment, y = Percent_Surprise, label = symbol)) +
  geom_point(size = 4, color = "blue") +
  geom_text(nudge_x = 0.01, nudge_y = 2) +
```

```
  labs(
    title = "Sentiment Score vs. Earnings Percent Surprise",
    x = "Sentiment Score",
    y = "Percent Surprise (%)"
  ) +
  theme_minimal()
```



Here I am plotting the sentiment score with the earning results. The higher the score, the more positive it is. Here we see that while UBER had the most positive sentiment, it had a significant positive performance. While DASH had the second most positively scored sentiment, it had the lowest performance expectation.

```
earnings_dates <- data.frame(
  symbol = c("UBER", "DASH", "Z", "DIS", "CVNA"),
  earnings_date = as.Date(c("2025-05-07", "2025-05-06", "2025-05-07", "2025-05-07", "2025-05-07"))
)

combined <- left_join(combined, earnings_dates, by = "symbol")




get_price_window <- function(symbol, date) {
  tq_get(symbol, from = date - 2, to = date + 2) %>%
    mutate(symbol = symbol, earnings_date = date)
}
```

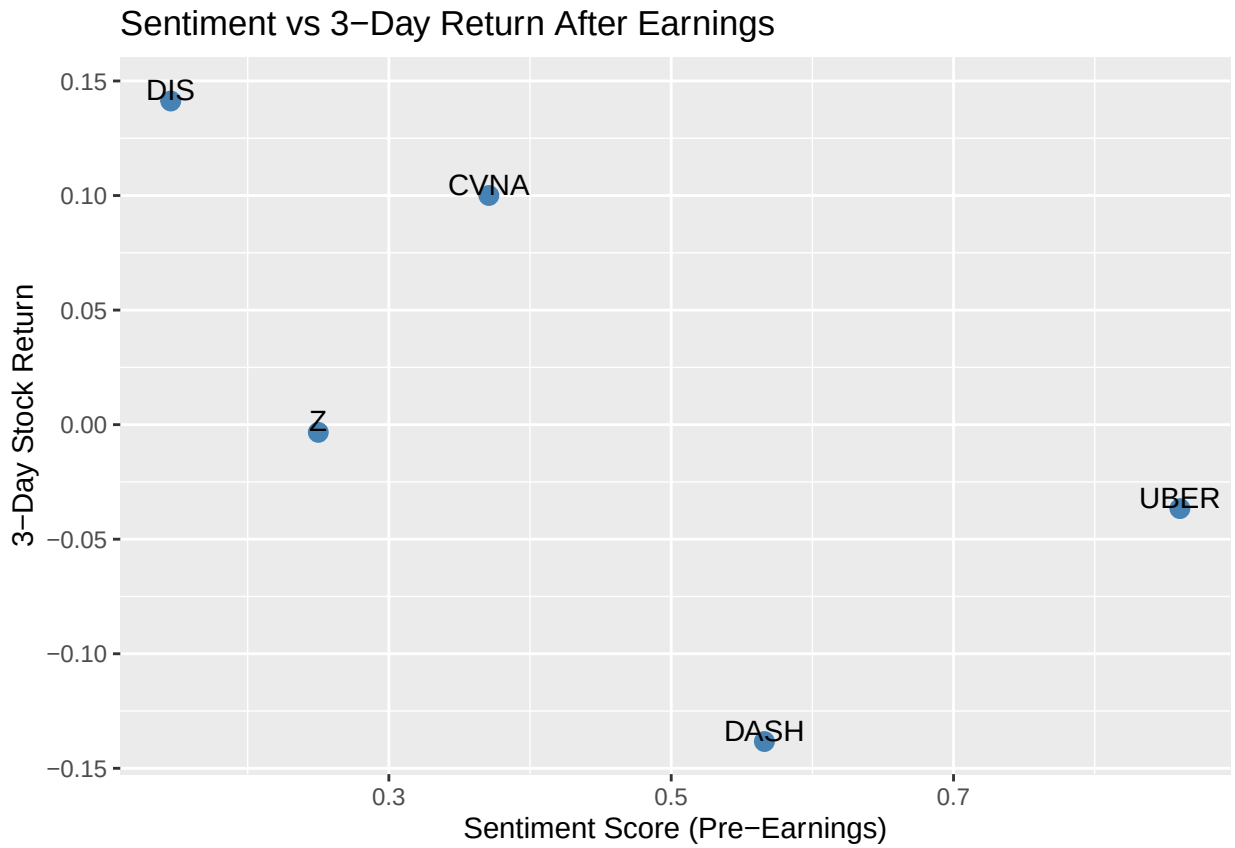
```r
price_data <- map2_df(combined$symbol, combined$earnings_date, get_price_window)



cumulative_returns <- price_data %>%
  filter(date >= earnings_date - 1 & date <= earnings_date + 1) %>%
  group_by(symbol) %>%
  arrange(date)


final_df <- cumulative_returns %>%
  left_join(combined, by = "symbol")

print(final_df)
```

```
## # A tibble: 19 x 15
## # Groups:   symbol [5]
##    symbol date        open  high   low close   volume adjusted earnings_date.x
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>    <dbl>    <dbl> <date>
##  1 CVNA   2025-05-05 254.  263.  254.  260.   2131700    260.  2025-05-05
##  2 DASH   2025-05-05 203.  207.  201.  205.   5617500    205.  2025-05-05
##  3 DIS    2025-05-05  89.7  93.1  89.6  92.1 10231000     92.1 2025-05-05
##  4 UBER   2025-05-05  83.1  86.6  83.0  85.4 25339000     85.4 2025-05-05
##  5 Z      2025-05-05  68.2  69    67.8  68.1  1987100     68.1 2025-05-05
##  6 CVNA   2025-05-06 255   261.  253.  259.   3192800    259.  2025-05-06
##  7 DASH   2025-05-06 194.  195.  185.  190.   9580800    190.  2025-05-06
##  8 DIS    2025-05-06  91.2  92.7  91    92.2 11839400     92.2 2025-05-06
##  9 UBER   2025-05-06  83.5  86.5  83.1  85.8 30378900     85.8 2025-05-06
## 10 Z      2025-05-06  67.0  68.3  66.8  67.2  2201100     67.2 2025-05-06
## 11 CVNA   2025-05-07 257.  263   256.  259.   5617100    259.  2025-05-07
## 12 DASH   2025-05-07 190.  190.  176.  177.   8621700    177.  2025-05-07
## 13 DIS    2025-05-07 102.  103.  100.  102.  36155300    102.  2025-05-07
## 14 UBER   2025-05-07  83.1  85.2  80.1  83.7 49238800     83.7 2025-05-07
## 15 Z      2025-05-07  67.4  68.6  67.0  67.9  3660100     67.9 2025-05-07
## 16 CVNA   2025-05-08 276.  294.  270   286.   9215800    286.  2025-05-08
## 17 DIS    2025-05-08 104.  106.  104.  105.  19265400    105.  2025-05-08
## 18 UBER   2025-05-08  83.9  84    82    82.3 24293100     82.3 2025-05-08
## 19 Z      2025-05-08  66.1  69.5  64.8  67.9  4527800     67.9 2025-05-08
## # i 6 more variables: EPS <dbl>, EPS_Forecast <dbl>, Percent_Surprise <dbl>,
## #   avg_sentiment <dbl>, earnings_result <chr>, earnings_date.y <date>
```

I want to find the price before and after the earning so since the earning was released 5/7/2025, I looked at the movement on 5/6 and 5/8 as well.

```r
library(ggplot2)

price_movement_3day <- final_df%>%
  group_by(symbol) %>%
  summarize(
    start_price = first(adjusted),
    end_price = last(adjusted),
```

```
    return_3day = (end_price - start_price) / start_price
  )


sentiment_analysis <- price_movement_3day %>%
  left_join(combined, by = "symbol")

library(ggplot2)

ggplot(sentiment_analysis, aes(x = avg_sentiment, y = return_3day, label = symbol)) +
  geom_point(size = 3, color = "steelblue") +
  geom_text(nudge_y = 0.005) +
  labs(title = "Sentiment vs 3-Day Return After Earnings",
       x = "Sentiment Score (Pre-Earnings)",
       y = "3-Day Stock Return")
```



Here we have the 3 day price movement return and comparing it with the sentiment of each stocks. The highest return is DIS which has the lowest positive sentiment score. The two highest sentiment stocks had negative returns from the pre-earning price to the post earning price.

```
cor_test <- cor.test(sentiment_analysis$avg_sentiment, sentiment_analysis$return_3day)
cor_test
```

```
##
##  Pearson's product-moment correlation
```

```
##
## data:  sentiment_analysis$avg_sentiment and sentiment_analysis$return_3day
## t = -1.4403, df = 3, p-value = 0.2454
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.9728560  0.5572029
## sample estimates:
##        cor
## -0.6393872
```

The correlation coefficient is -0.639 which means that it has a negative relationship between sentiment and the 3-day return. It appears that stocks with higher pre-earnings sentiment perform worse. However, we do have a wide range for the confidence interval which makes sense since we have a low sample size.

CONCLUSION

The hypothesis of this project is simple: positive news about a company should be followed by positive performance, reflected in a higher stock price. While this idea appears intuitive, that's not what was found here.

In this analysis, even when both sentiment and earnings were positive, price movement in the days following earnings did not always align. The assumption that one factor directly and proportionally impacts the other does not consistently hold. In fact, we observed cases where stocks with strong sentiment and solid earnings still experienced a decline in price post-earnings. However, one theory could be that the sentiment was so positively strong, it drove the stock price higher than its worth. When the earnings report is release, it corrected the price which resulted what looks like a "decline".

In short, while sentiment and earnings may shape investor outlook, they do not guarantee a specific price response. However, this is also a very small sample of stocks and the news that drove the sentiment was also very limited. Even with that said, it is interesting to see the relationship between market sentiment and technical factors, and how that affects the short term return of a stock.