Stock Trends Throughout The Years

Erik Mercado, emm4376

Introduction

The data sets I used in this report will be named *sp500* and *stock_data*. I chose these data sets because I was interested in how stock prices changed throughout the years across a wide range of categories. The data set *sp500* contains information about Fortune 500 companies such as their ticker, name of the company, industry & sub-industry of the company (Categorical), location of headquarters of the company (Categorical), and when the company was added to the Fortune 500. Each row in *sp500* represents one company. The data set *stock_data* has information on the stocks of Fortune 500 companies across a range of dates from 2013 to 2018. The information it includes is the date of the data, the highest price of the stock the day (Numeric), the lowest price of the stock that day (Numeric), the opening price of the stock on that day (Numeric), the closing price of the stock on that day (Numeric), the volume of the stocks traded that day (Numeric), and the ticker of the company. Each row represents on day of the specified stock. I got the data for *sp500* & *stock_data* from https://www.kaggle.com/datasets/alexanderxela/sp-500-companies & https://www.kaggle.com/datasets/camnugent/sandp500, respectively. The key that I will use to join the two data sets is the ticker of the stock, which are the columns "Ticker" in *sp500* and "Name" in *stock_data*. The research question I will be exploring is if there is any differences among industries in terms of how stock prices change.

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   1.0.1
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.5.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(dplyr)
library(ggplot2)

sp500 <- read.csv("/Users/erik/Downloads/sp500-constituents.csv")
stock_data <- read.csv("/Users/erik/Downloads/archive (3)/all_stocks_5yr.csv")
```

Joining/Merging

After loading up the data sets, I had to figure out how I was going to merge them. *sp500* had 503 observations and *stock_data* had 619,040 observations. I decided to go with a left join, but then I had to figure out which dataframe I was going to use as my main one. I ran both joins below to see which one would be better suited for what I needed.

```
merged_companies <- sp500 |>
  left_join(stock_data, by = c("Ticker" = "Name"))

merged_companies2 <- stock_data |>
  left_join(sp500, by = c("Name" = "Ticker"))
```

After creating both merged data sets, I checked them to see how they turned out and how many observations I could use from both.

```
merged_companies |>
  group_by(Ticker) |>
  summarize(n=n()) |>
  filter(n == 1)
```

```
## # A tibble: 106 x 2
##    Ticker     n
##    <chr>  <int>
##  1 ACGL       1
##  2 AMCR       1
##  3 ANET       1
##  4 ATO        1
##  5 BALL       1
##  6 BBWI       1
##  7 BIO        1
##  8 BKNG       1
##  9 BKR        1
## 10 BR         1
## # ... with 96 more rows
```

```
merged_companies2 |>
  mutate(Ticker = Name) |>
  select(-Name) |>
  mutate(Name = Name.y) |>
  select(-Name.y) |>
  group_by(Ticker) |>
  summarize(n=n()) |>
  filter(n == 1)
```

```
## # A tibble: 0 x 2
## # ... with 2 variables: Ticker <chr>, n <int>
```

```
common_ids <- merged_companies |>
  group_by(Ticker) |>
  summarize(n=n())
nrow(common_ids)
```

```
## [1] 503
```

After exploring the data after joining them, I decided that that *merged_companies2* would be the better data set for my analysis. I did this by checking how many companies would have the least amount of companies with just one row post merge, which would mean that there was no data on that company's stocks. The data set *stock_data* ended up having all the IDs that were present in *sp500*, but *sp500* seemed to have been missing 103 of the IDs that were present in *stock_data*. No observations were dropped with *merged_companies*, but this will change during the tidying stage because I would have to drop companies that don't have an Industry label.

Tidying

```
tidy_data <- merged_companies2 |>
  mutate(Ticker = Name) |>
  select(-Name) |>
  mutate(Name = Name.y) |>
  select(-Name.y) |>
  separate(date, into = c("Year", "Month", "Day"), sep = "-") |>
  select(Ticker, Name, Industry, Sub.Industry, open:close, Month, Day, Year) |>
  filter(!is.na(open)) |>
  filter(!is.na(close)) |>
  filter(!is.na(high)) |>
  filter(!is.na(low)) |>
  filter(!is.na(Industry))


common_ids2 <- tidy_data |>
  group_by(Ticker) |>
  summarize(n=n())
nrow(common_ids2)
```

```
## [1] 397
```

```
nrow(merged_companies2) - nrow(tidy_data)
```

```
## [1] 128695
```

After finalizing the joining process, I decided to divide up the dates because I wanted to focus on the yearly data. I also decided to rename some columns so that it would be easier for me to use. Then I decided to only keep observations that have an Industry label and have information for all 4 stock variables becasue these were the main variables that I was going to be exploring. After this, I ended up with only 397 companies left to use and 128695 observations were dropped.

Wrangling

After the tidying process, I did a bit of wrangling. I created two variables to represent the difference between the open and close price per observation, and then another on to represent the difference between the high and low price per observation.

```
tidy_data <- tidy_data |>
  group_by(Industry, Sub.Industry, Year) |>
  mutate(open_close_difference = open - close,
         high_low_difference = high - low)
```

```
companies_per_industry <- tidy_data |>
  group_by(Ticker, Industry) |>
  summarize(n=n())
```

```
## `summarise()` has grouped output by 'Ticker'. You can override using the
## `.groups` argument.
```

```
table(companies_per_industry$Industry)
```

```
##
## Communication Services Consumer Discretionary       Consumer Staples
##                     17                     47                     31
##                 Energy             Financials            Health Care
##                     19                     55                     50
##            Industrials Information Technology              Materials
##                     54                     52                     21
##            Real Estate              Utilities
##                     24                     27
```

```
summary(tidy_data$open_close_difference)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -81.3800  -0.4300  -0.0300  -0.0228   0.3700  75.8100
```

```
summary(tidy_data$high_low_difference)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -0.255   0.615   0.995   1.509   1.670 138.260
```

```
summary(tidy_data$open)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.62   42.44   64.36   84.27   96.38 1477.39
```

```
summary(tidy_data$close)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.62   42.45   64.39   84.29   96.44 1450.89
```

```
summary(tidy_data$high)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.69   42.83   64.93   85.01   97.19 1498.00
```

```
summary(tidy_data$low)
```

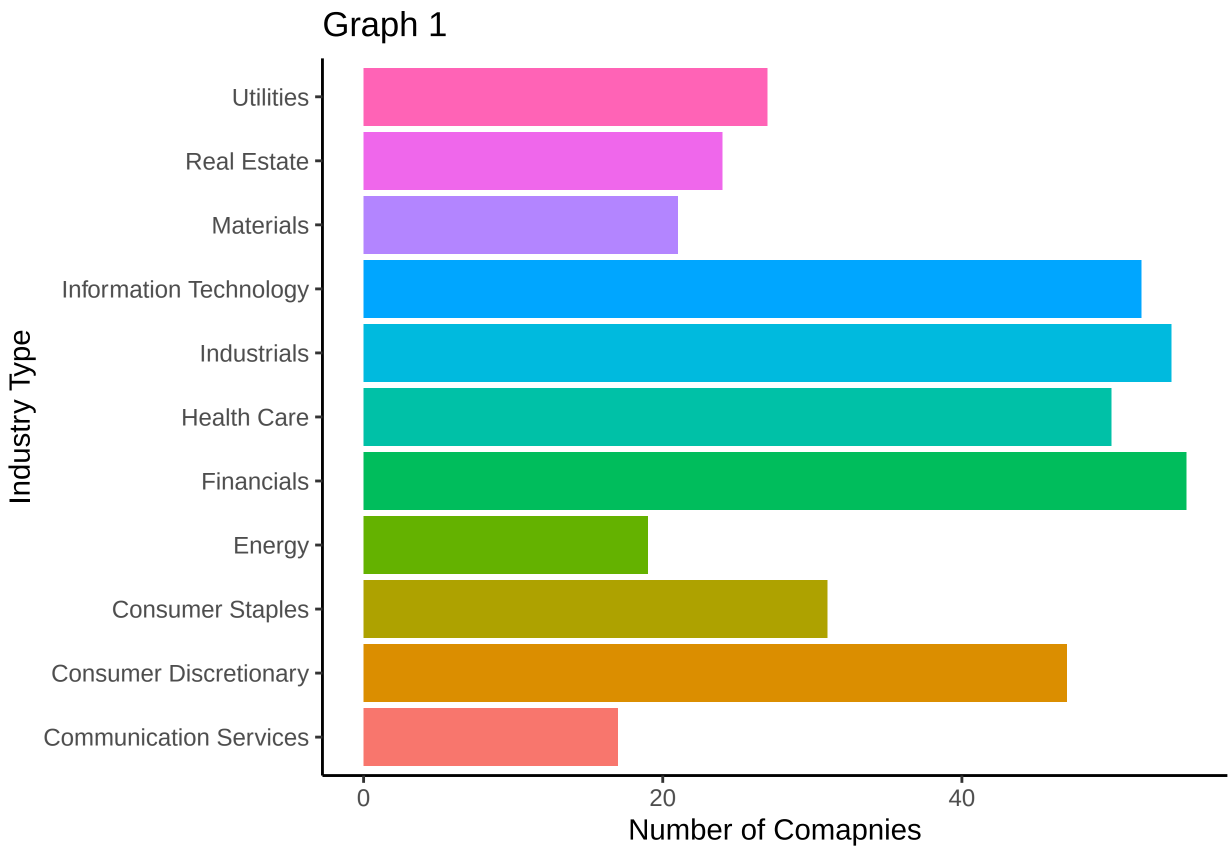
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.61   42.03   63.79   83.51   95.60 1450.04
```

After creating those two variables, I chose to summarize those two, as well as the number of companies per industry category. It seems that the industry category with the most companies is *Financials* and the category with the least amount of companies is *Communication Services*. For the variable *open_close_difference*, it ranged from -81.3800 dollars to 75.8100 dollars. For the variable *high_low_difference*, it ranged from -0.255 dollars to 138.260. For the variable *open*, it ranged from 1.62 dollars to 1477.39 dollars. For the variable *close*. it ranged from 1.62 dollars to 1450.89 dollars. For the variable *high*, it ranged from 1.69 dollars to 1498.00 dollars. For the variable *low*, it ranged from 1.61 dollars to 1450.04 dollars. It should be noted that these summary numbers are calculated across all calculations, there is no categorization by industry or year happening yet. That analysis will be done later on in the report.

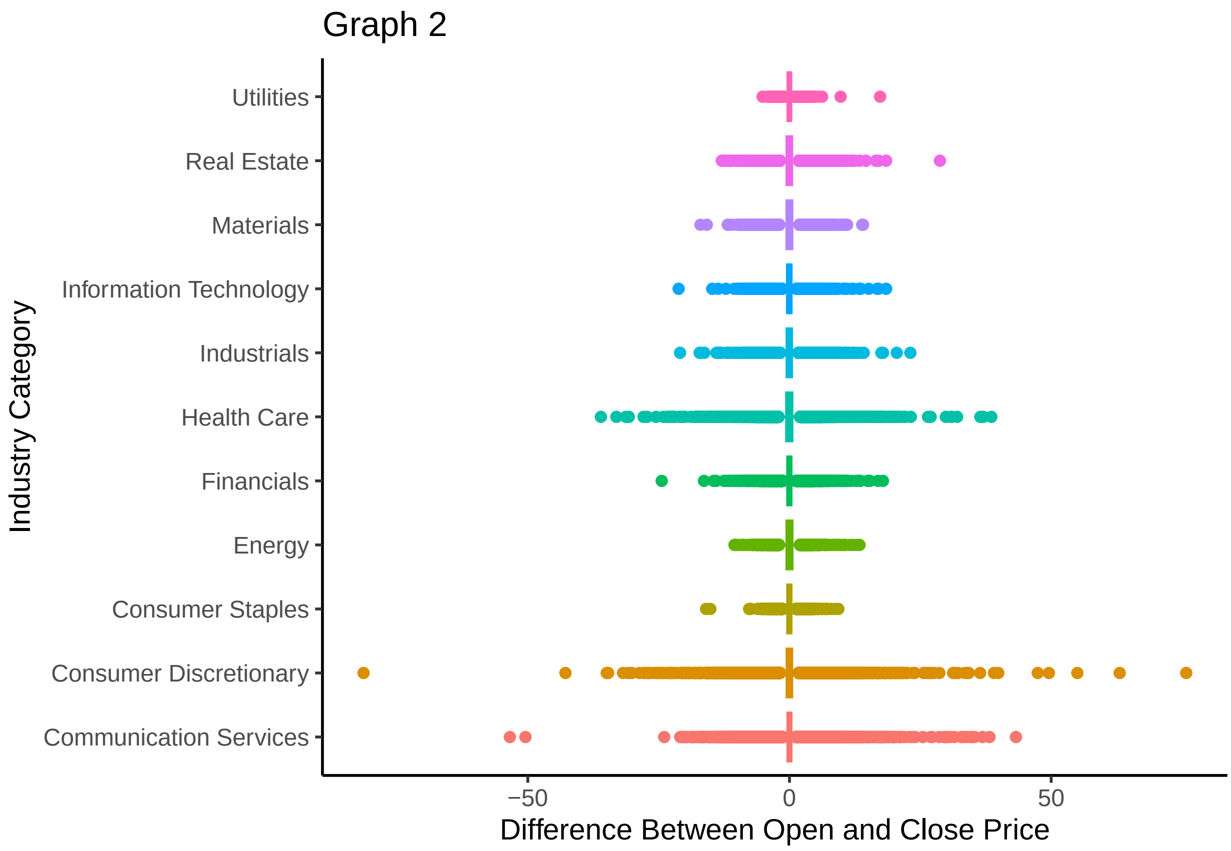
Vizualizing

I decided to go with more charts than were asked of us because there were several relationships that I wanted to explore.

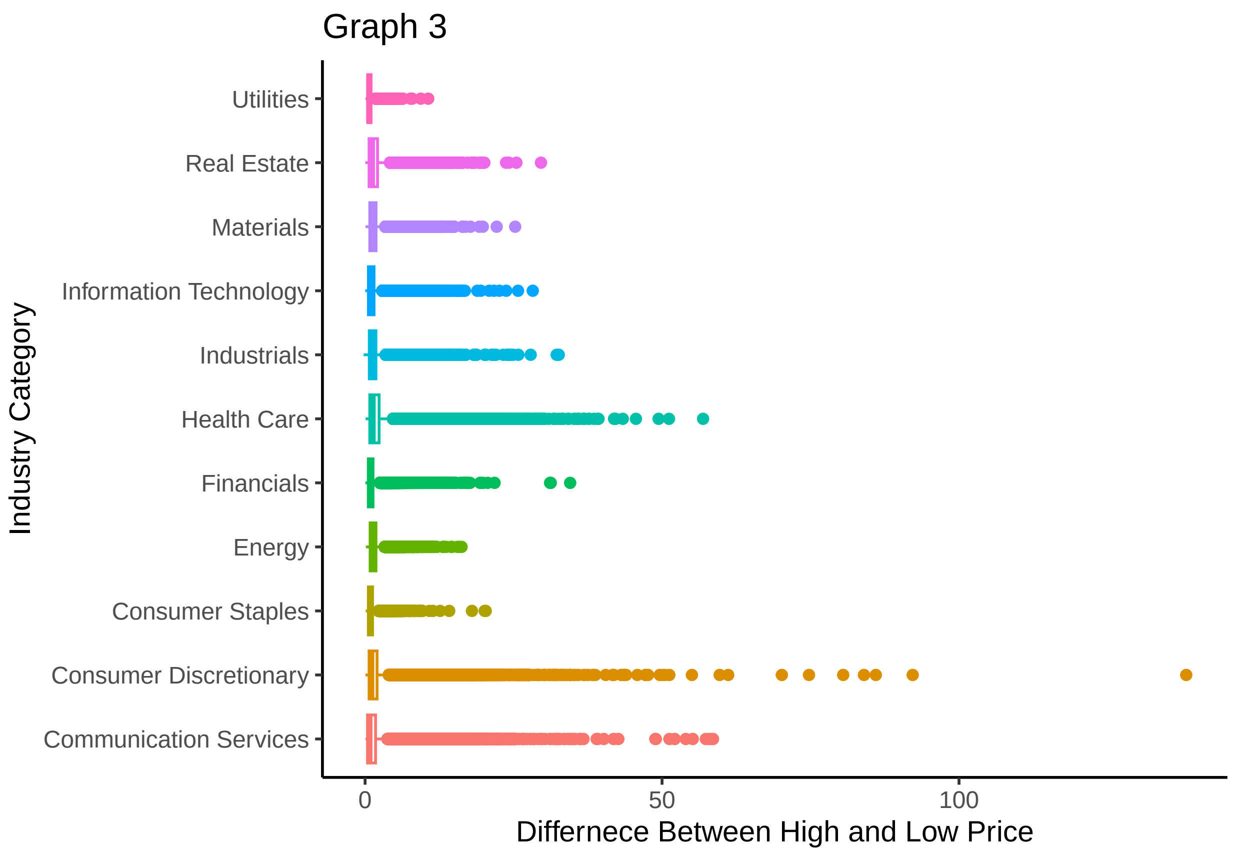
```
companies_per_industry |>
  ggplot(aes(y = Industry)) +
  geom_bar(aes(fill = Industry)) +
  labs(title = "Graph 1", x = "Number of Comapnies", y = "Industry Type") +
  theme_classic() + theme(legend.position = "none")
```



```
tidy_data |>
  group_by(Industry) |>
  ggplot(aes(y = Industry, x = open_close_difference )) +
  geom_boxplot(aes(color = Industry)) +
  labs(title = "Graph 2", x = "Difference Between Open and Close Price", y = "Industry Category") +
  theme_classic()+ theme(legend.position = "none")
```

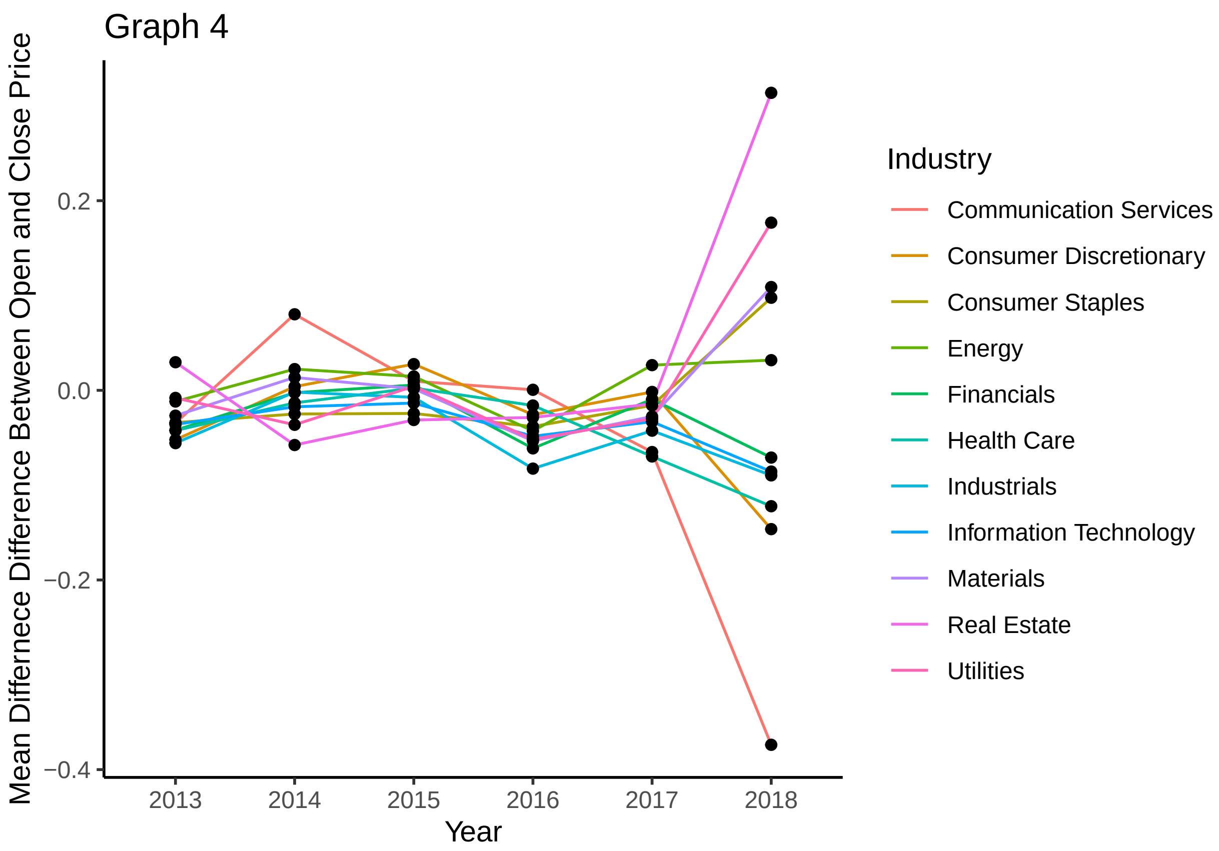


```
tidy_data |>
  group_by(Industry) |>
  ggplot(aes(y = Industry, x = high_low_difference )) +
  geom_boxplot(aes(color = Industry)) +
  labs(title = "Graph 3", x = "Differnece Between High and Low Price", y = "Industry Category") +
  theme_classic()+ theme(legend.position = "none")
```



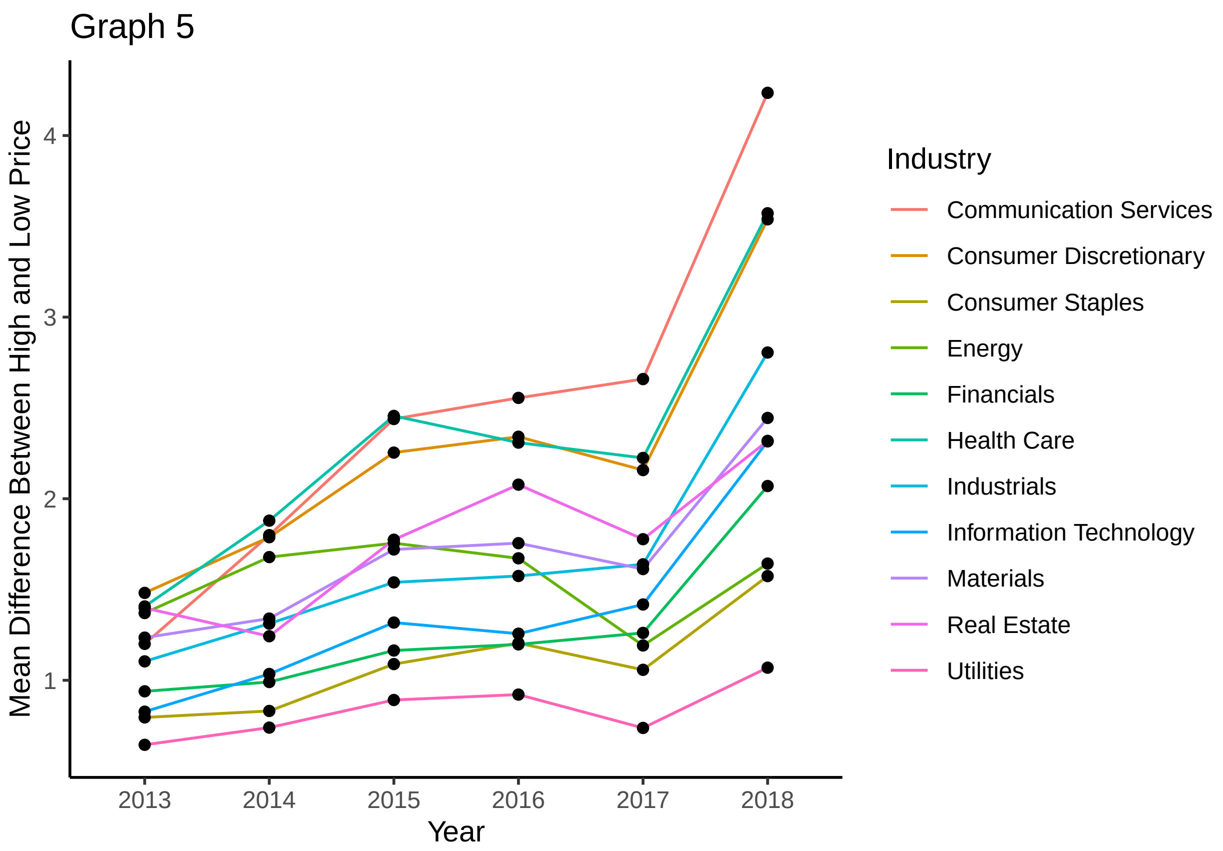
```
tidy_data |>
 group_by(Industry, Year) |>
 summarize(mean_open_close_difference = mean(open_close_difference)) |>
 ggplot(aes(x = Year, y = mean_open_close_difference)) +
 geom_line(aes(color = Industry, group = Industry)) +
 geom_point() +
 labs(title = "Graph 4", x ="Year", y = "Mean Differnece Difference Between Open and Close Price", y =
 theme_classic()
```

```
## `summarise()` has grouped output by 'Industry'. You can override using the
## `.groups` argument.
```



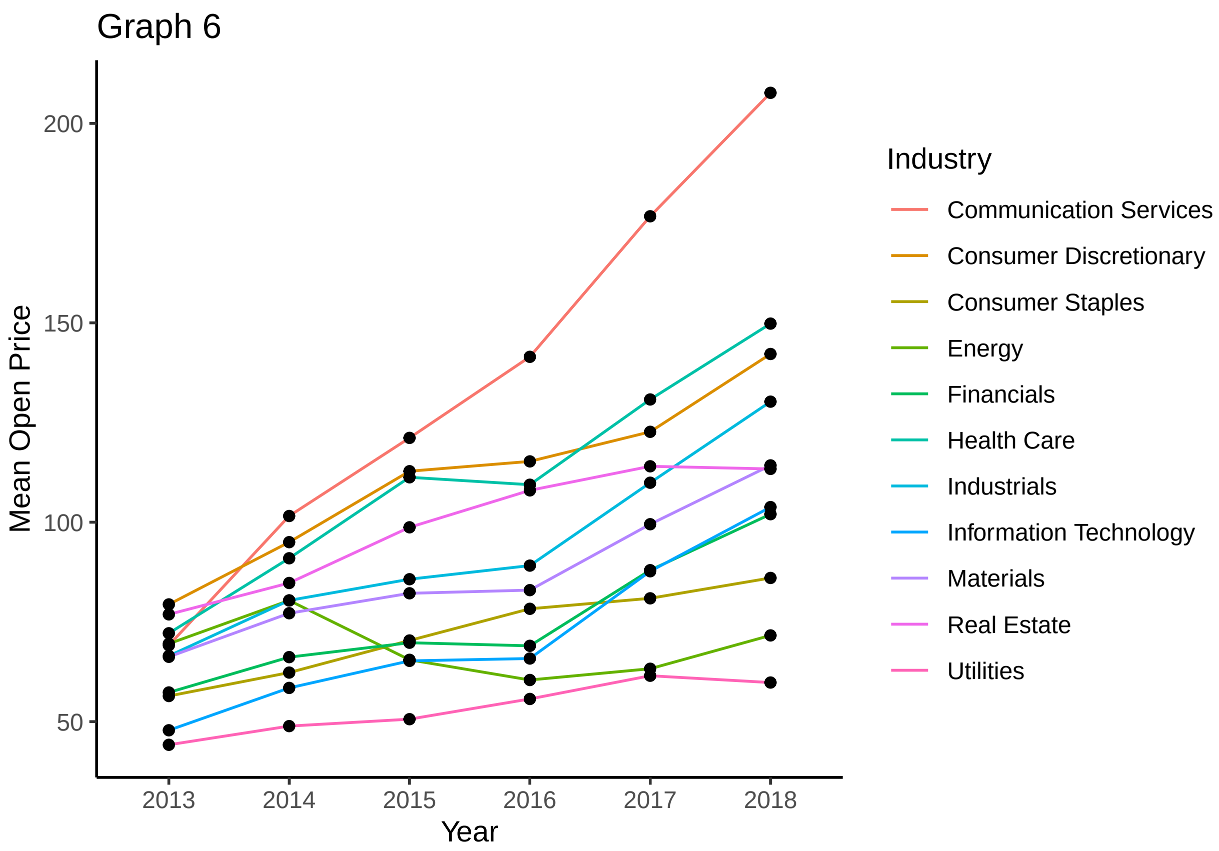
```
tidy_data |>
  group_by(Industry, Year) |>
  summarize(mean_high_low_difference = mean(high_low_difference)) |>
  ggplot(aes(x = Year, y = mean_high_low_difference)) +
  geom_line(aes(color = Industry, group = Industry)) +
  geom_point() +
  labs(title = "Graph 5", x = "Year", y = "Mean Difference Between High and Low Price", y = "Industry Ca
  theme_classic()
```

```
## `summarise()` has grouped output by 'Industry'. You can override using the
## `.groups` argument.
```



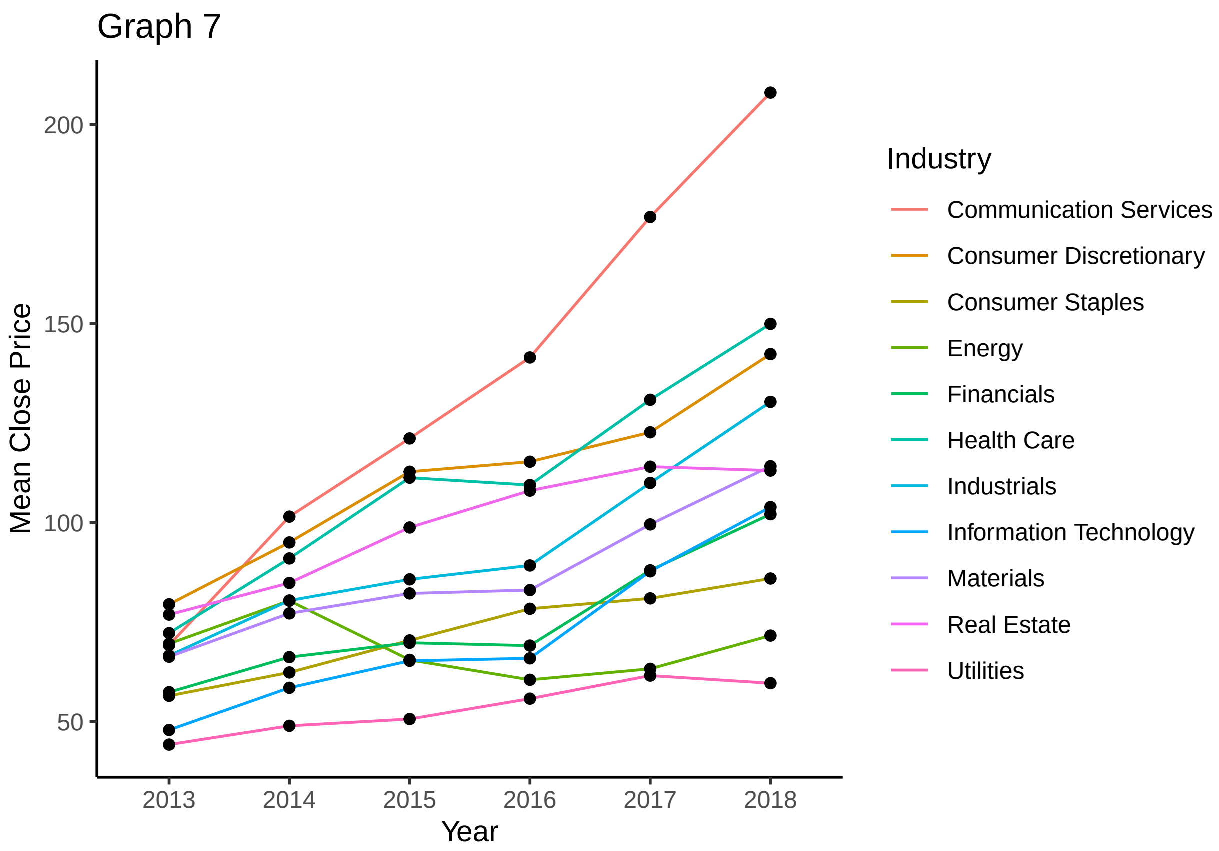
```
tidy_data |>
  group_by(Industry, Year) |>
  summarize(mean_open = mean(open)) |>
  ggplot(aes(x = Year, y = mean_open)) +
  geom_line(aes(color = Industry, group = Industry)) +
  geom_point() +
  labs (title = "Graph 6", x = "Year", y = "Mean Open Price")  +
  theme_classic()
```

```
## `summarise()` has grouped output by 'Industry'. You can override using the
## `.groups` argument.
```



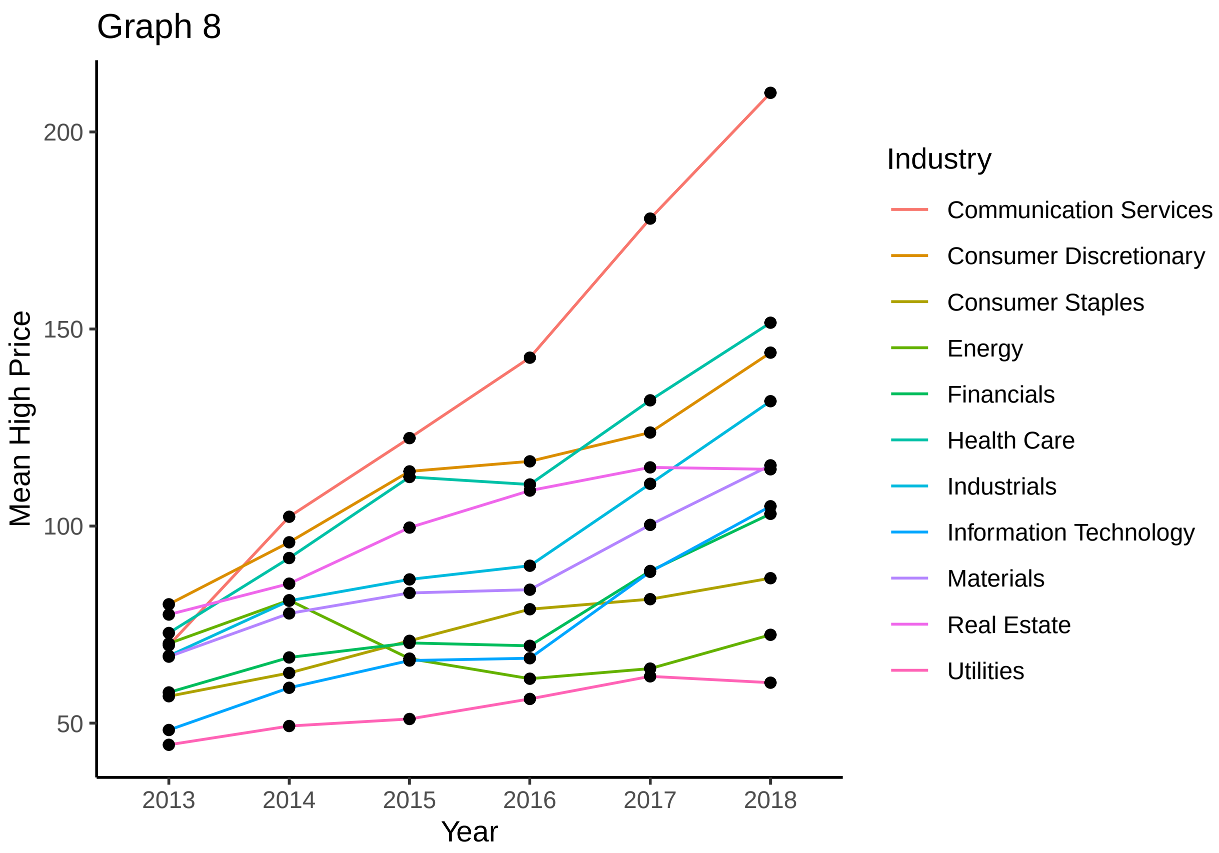
```
tidy_data |>
 group_by(Industry, Year) |>
 summarize(mean_close = mean(close)) |>
 ggplot(aes(x = Year, y = mean_close)) +
 geom_line(aes(color = Industry, group = Industry)) +
 geom_point() +
 labs (title = "Graph 7", x = "Year", y = "Mean Close Price") +
 theme_classic()
```

```
## `summarise()` has grouped output by 'Industry'. You can override using the
## `.groups` argument.
```



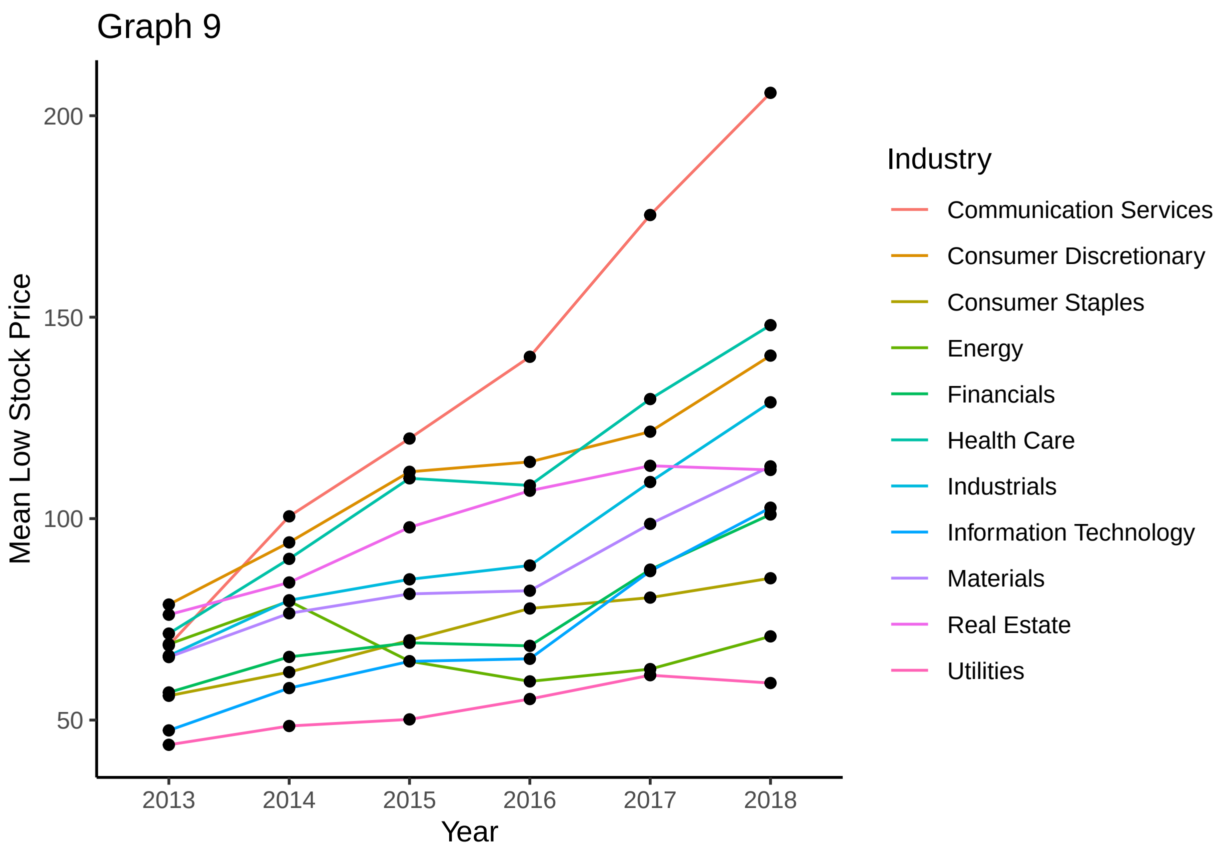
```
tidy_data |>
 group_by(Industry, Year) |>
 summarize(mean_high = mean(high)) |>
 ggplot(aes(x = Year, y = mean_high)) +
 geom_line(aes(color = Industry, group = Industry)) +
 geom_point() +
 labs (title = "Graph 8", x = "Year", y = "Mean High Price") +
 theme_classic()
```

```
## ‘summarise()‘ has grouped output by ’Industry’. You can override using the
## ‘.groups‘ argument.
```



```
tidy_data |>
 group_by(Industry, Year) |>
 summarize(mean_low = mean(low)) |>
 ggplot(aes(x = Year, y = mean_low)) +
 geom_line(aes(color = Industry, group = Industry)) +
 geom_point() +
 labs (title = "Graph 9", x = "Year", y = "Mean Low Stock Price") +
 theme_classic()
```

```
## ‘summarise()‘ has grouped output by ’Industry’. You can override using the
## ‘.groups‘ argument.
```



Discussion

After all of this exploration, I made a few observations. The first is that in Graph 4, you can see that from the years 2013. to 2017, stocks' opening and closing prices across most industries stay pretty stable within a certain range of change, but then in 2018, there is a big jump in the change. The second is that in Graph 5, it can be seen that companies in the *Communication Services* industry suffered in the highest volatility in stock prices and companies in the *Utilities* industry suffered the lowest volatility in stock prices. The third is that in Graphs 6 through 9, the order in which the different industries rank within the 4 stock variables (open, close, high, and low) stays the same. Based on all these observations, it seems that there is a difference in how stock prices change across different industries, not so much in the direction (positive or negative), but more in the amount it changes by.

The process of this analysis was very enlightening. I figured out how to do new things in R that I had not done before and it was a great way to test my skills. I also enjoyed the process of exploring the data and seeing what insights I could draw from it. If I were to do something different, I would have found a way to have all industries have the same amount of companies because there may have been some different if all the industry sizes were set to the same size. The challenging thing was trying to organize certain aspects of the data in a certain way so that it would be usable in the way that I wanted.