S&P 500
DSC 324
Prof. Johnson
Group 4 - Erik Johnson, Alex Look, and Alex Schertler

Introduction

Our dataset contained the day to day records of every company within the current S&P 500 dating back to January 4th, 2010. The data was segmented into three distinct sets; one that consisted of the historical trading data, such as opening price, closing price, high and low price, and daily volume of the given stock on a day to day basis between January 4th 2010 and February of 2022; another set contained merely the date and the closing value of the S&P 500 on those dates within the same date range listed above; lastly, a detailed analysis of each company currently within the S&P 500 was included, which featured data such as the current price as of the last day of the data set above of each company, the current market cap, their EBITDA and revenue gain rates in addition to the amount of full-time employees, and some other character string data such as the city, state, country and sector/industry the company was within.

Our goal was to review the data holistically in addition to in a segmented fashion; meaning we wanted to find any patterns or correlations between the intertwinement of the three sets, but we also wanted to look at each set on their own in hopes of identifying unique analysis that could have been muddied based on the size of the combined data, and the fact that some stocks had not been on the S&P for the entirety of the 12 year sample and therefore defaulted to $0 values on a given date outside of their presence within the S&P. At first we performed a surface level look at the data and removed unnecessary data while cleaning up other values.

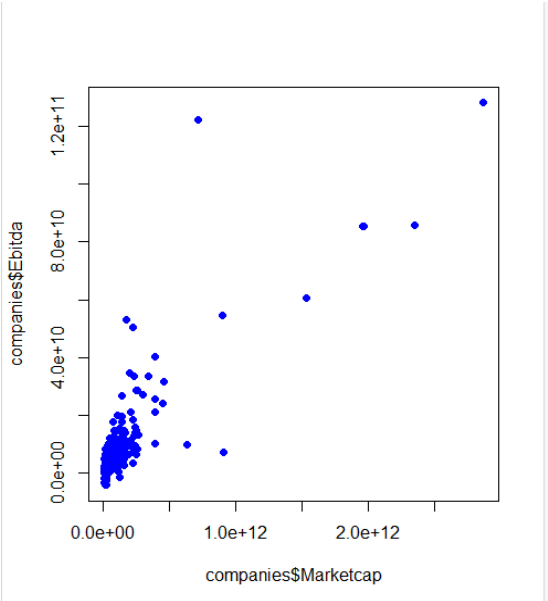
We deleted the multiple name listings (Shortname & Longname in sp500_companies) and kept only the ticker abbreviation (Symbol in sp500_companies) column. We chose to do that because the stock abbreviation was listed in the stocks and companies data set and could be considered a key. Additionally we removed the long business summary because it was a long text description.

We converted the market cap and EBITDA column to be a numerical value. Additionally we made two sets for analysis. We scaled down the EBITDA and Marketcap by 100,000,000 so that the units wouldn’t be substantially higher than the other numerical values for any PCA analysis. In the other set for analysis we did not scale it down and keyed those fields against each other. Additionally we reformatted the date field in the stocks data set to match the formatting of the date column in the index dataset.

During our cleanup we created columns that analyzed the mean trading price over the duration of the data, as well as the median trading price. The point of this was to identify and analyze whether these points took place 50% through the 10 year sample, or if they occurred sooner or later. This would help us to identify whether a stock grew steadily throughout the 10 years or saw a balloon hike in price; meaning the price remained lower for much longer and then saw a stark increase recently. It would also help us to see whether the stock tracked below the mean price longer than it did above it. We also did this same analysis on the median and mean of the overall S&P.

Exploratory Analysis of the Data

There were many areas to tackle our exploratory analysis from. One example is when we plotted EBITDA against the market cap, we saw that Tesla has a much higher marketcap than expected when comparing to the others. In general, there is linearity to the plot, as market cap scales pretty consistently with a company's EBITDA, but in the case of Tesla we saw a market cap that was roughly 125 times the EBITDA value, compared to other top top market cap stocks that came in around 20 times the value of the EBITDA.



Additionally for our statistical analysis we wanted to centralize our focus around Goog vs Googl. As noted in milestone 2, these two ticker symbols represent shares in the same company, but the classification of the shares varies between Class C and Class A shares. The difference between Goog and Googl is that Goog does not provide shareholders a right to vote while Googl does. The expectation throughout our review was that Goog would always trade at a slightly discounted price because of that lack of votability. Using the entirety of the data's history, that conclusion did ring true:

```
> summary(googl)
     Date              Symbol            Adj.Close          Close             High              Low               Open             Volume         
 Length:5046        Length:5046        Min.   : 279.8   Min.   : 279.8   Min.   : 282.5   Min.   : 278.5   Min.   : 280.5   Min.   :  465600  
 Class :character   Class :character   1st Qu.: 551.2   1st Qu.: 551.2   1st Qu.: 556.7   1st Qu.: 546.6   1st Qu.: 551.7   1st Qu.: 1386600  
 Mode  :character   Mode  :character   Median : 835.1   Median : 835.1   Median : 839.0   Median : 829.0   Median : 833.0   Median : 1818700  
                                       Mean   :1015.1   Mean   :1015.1   Mean   :1024.2   Mean   :1005.2   Mean   :1014.9   Mean   : 2442886  
                                       3rd Qu.:1211.0   3rd Qu.:1211.0   3rd Qu.:1220.8   3rd Qu.:1203.2   3rd Qu.:1211.1   3rd Qu.: 3005600  
                                       Max.   :2996.8   Max.   :2996.8   Max.   :3030.9   Max.   :2978.0   Max.   :3025.0   Max.   :24859915  
                                       NA's   :2529     NA's   :2529     NA's   :2529     NA's   :2529     NA's   :2529     NA's   :2529      
  S.P.Close        
 Length:5046        
 Class :character   
 Mode  :character   


> summary(goog)
     Date              Symbol            Adj.Close          Close             High              Low               Open             Volume         
 Length:2523        Length:2523        Min.   : 278.5   Min.   : 278.5   Min.   : 281.2   Min.   : 277.2   Min.   : 279.1   Min.   :    7922  
 Class :character   Class :character   1st Qu.: 539.8   1st Qu.: 539.8   1st Qu.: 543.8   1st Qu.: 535.7   1st Qu.: 539.6   1st Qu.: 1271000  
 Mode  :character   Mode  :character   Median : 813.7   Median : 813.7   Median : 816.7   Median : 805.1   Median : 811.7   Median : 1689526  
                                       Mean   :1010.1   Mean   :1010.1   Mean   :1019.1   Mean   :1000.4   Mean   :1009.6   Mean   : 2310214  
                                       3rd Qu.:1206.0   3rd Qu.:1206.0   3rd Qu.:1216.2   3rd Qu.:1196.7   3rd Qu.:1205.0   3rd Qu.: 2805900  
                                       Max.   :3014.2   Max.   :3014.2   Max.   :3042.0   Max.   :2997.8   Max.   :3037.3   Max.   :24978074  
                                       NA's   :6        NA's   :6        NA's   :6        NA's   :6        NA's   :6        NA's   :6         
```

As we can see from the data above, the mean of the googl adjusted close over the roughly 10 year sample was roughly $5 or .047% higher than that of Goog. Meanwhile the median came in $21.40 higher for googl than it did for goog which was a variance of roughly 2.5% higher for googl than goog. Based on the data one could infer that the amount of volume, daily, correlated directly to the lower price and lower median as goog had a daily volume, on average, 132,672 lower than that of googl, which equates out to a volume-by-the-day being roughly 5.4% higher for googl than goog.

```
Call:
lm(formula = DoubleClose ~ ., data = googlSelect)

Residuals:
     Min       1Q   Median       3Q      Max 
-509.98  -97.70   -5.97  104.07  799.87 

Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.348e+03  8.570e+00 157.335  < 2e-16 ***
`Adj Close`  5.355e-01  4.085e-01   1.311  0.19001    
Close               NA         NA      NA       NA    
High         2.484e+00  4.366e-01   5.689 1.43e-08 ***
Low         -1.180e+00  4.330e-01  -2.725  0.00648 ** 
Open        -6.285e-01  4.048e-01  -1.553  0.12065    
Volume      -4.565e-05  1.864e-06 -24.483  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 143.1 on 2511 degrees of freedom
  (6 observations deleted due to missingness)
Multiple R-squared:  0.9711,    Adjusted R-squared:  0.9711 
F-statistic: 1.69e+04 on 5 and 2511 DF,  p-value: < 2.2e-16

Warning: NAs introduced by coercion

Call:
lm(formula = DoubleClose ~ ., data = googSelect)

Residuals:
    Min      1Q  Median      3Q     Max 
-522.74  -95.80   -5.64  102.34  865.52 

Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.383e+03  8.394e+00 164.724  < 2e-16 ***
`Adj Close`  8.098e-01  4.172e-01   1.941  0.05235 .  
Close               NA         NA      NA       NA    
High         2.130e+00  4.383e-01   4.859 1.25e-06 ***
Low         -1.368e+00  4.390e-01  -3.115  0.00186 ** 
Open        -3.852e-01  4.089e-01  -0.942  0.34620    
Volume      -4.901e-05  1.854e-06 -26.442  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 143.9 on 2511 degrees of freedom
  (6 observations deleted due to missingness)
Multiple R-squared:  0.9708,    Adjusted R-squared:  0.9707 
F-statistic: 1.669e+04 on 5 and 2511 DF,  p-value: < 2.2e-16
```

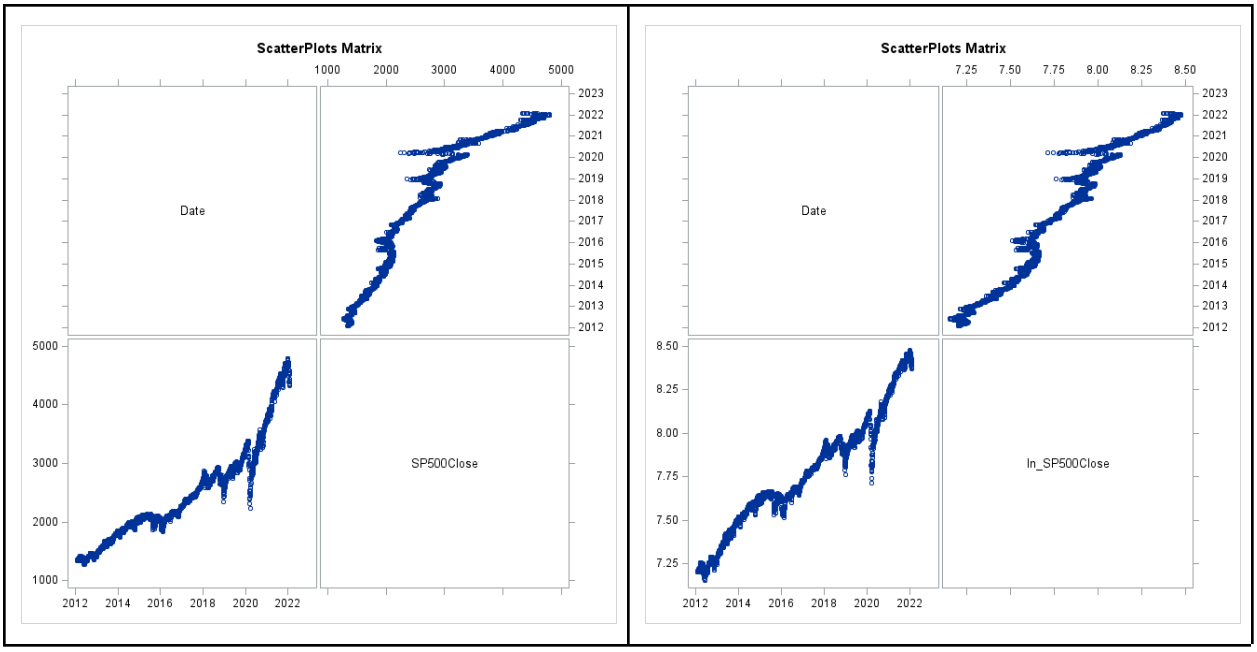
(Top is Googl and bottom is Goog)

What was interesting about our findings is that although googl on average traded higher, as we can see from all our categories across the board above including the high, low and open, it wasn't true that googl closed or traded higher every single day. In fact, of the 2523 days that we analyzed, the adjusted close for Goog was higher than googl in 514 of those days (20.3%) . Of those 514 days, 450 of them have come since 1/9/2020. That means of 756 days within our data set since 1/9/2020, Goog has had a higher adj close than Googl in 59.5% of them. While some can be explained by a higher

volume, as Goog has traded at a higher volume in 122 of those 450 days, the correlation isn't strong enough to conclude that. Another area of analysis is the impact full-time employees may have on the Goog stock. As the Class C shares (Goog) are the shares given as compensation to employees, it's possible that those shares are flipped with greater frequency than the Googl shares which may draw more long-term investors who desire voting rights and elongated company growth.

Additionally, while this is speculatory based on the fact that employees receive Class C shares, it's possible that Google itself is buying back the Class C shares and not the Class A shares in order to escalate the Goog symbol price as an employee benefit. This is potentially supported by the data as we see our largest spikes in volume on Goog, and largest variance in volume between Goog and Googl, during the final weeks/first weeks of a new quarter. This could be anything from hiring season to buying back stock around earnings to limit the shares in circulation and inflate the share price (which may be why we see Goog higher than googl).

Lastly, from a surface level look, we wanted to look at the closing price over the duration of the data and identify any points that stood out. To do that we transformed the data and created some scatter plots:



After seeing the transformation it is felt that both plots are increasing and positively increasingly linear scatter plots meaning that with the further date the higher the S&P500 was at closing time; or that the market has seen a steady escalation in total close. We were able to identify one stark decline in the market in 2020, which we identified as being COVID's initial impact on the S&P 500. While there were other temporary flattenings or dips, the general plot was linear in a positive direction.We also notice that the recovery of the S&P took place more dramatically than the downturn. We also noticed a steady growth rate in 2012 to 2016, which we

identified as potentially being the company coming out of the 2008 recession with a significant recovery rate until 2016 where it began to level off.

While these three analyses were less regressional in nature, and centralized their focus more around a surface level statistical analysis, we did pivot from these analyses into specified techniques that would give us a better look at the totality of the information within our data set.

Application of Analysis and Techniques

Principal Component Analysis:

The first technique we wanted to utilize on our dataset was a principal component analysis. We wanted to utilize the following variables: current price, marketcap, EBITDA, revenuegain and full-time employees. For the PCA, our independent variable was current price. When we initially ran our PCA, we noticed that we had some scaling issues, as can be seen below with the high proportionate variance accounted for in PC1:

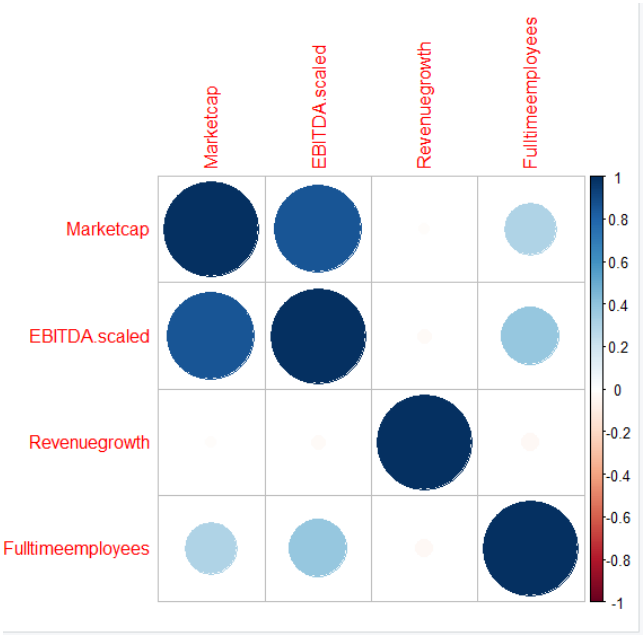
```
Standard deviations (1, .., p=6):
[1] 2.326420e+11 6.536499e+09 1.275279e+05 3.536727e+02
[5] 2.434643e+00 7.125505e-06

Rotation (n x k) = (6 x 6):
                              PC1           PC2           PC3
Currentprice       6.071158e-10 -5.458383e-09  2.356367e-04
Marketcap          9.990311e-01 -4.400922e-02 -3.908048e-09
Ebitda             4.400922e-02  9.990311e-01 -4.179373e-06
Revenuegrowth     -1.957633e-13 -1.683860e-11 -4.488302e-07
Fulltimeemployees  1.878351e-07  4.175154e-06  1.000000e+00
Weight             2.339998e-14 -1.093757e-15 -1.032925e-11
                              PC4           PC5           PC6
Currentprice       1.000000e+00  1.744690e-04  2.847096e-09
Marketcap         -8.458258e-10 -6.948045e-13 -2.342791e-14
Ebitda             6.411188e-09  1.607363e-11  3.834060e-17
Revenuegrowth     -1.744689e-04  1.000000e+00  2.503437e-08
Fulltimeemployees -2.356368e-04  4.077189e-07  9.669605e-12
Weight            -2.842726e-09 -2.503486e-08  1.000000e+00
Importance of components:
                            PC1       PC2    PC3   PC4
Standard deviation     2.326e+11 6.536e+09 127528 353.7
Proportion of Variance 9.992e-01 7.900e-04      0   0.0
Cumulative Proportion  9.992e-01 1.000e+00      1   1.0
                          PC5       PC6
Standard deviation     2.435 7.126e-06
Proportion of Variance 0.000 0.000e+00
Cumulative Proportion  1.000 1.000e+00
```

It was identified that these scaling issues were due to values of EBITDA and Marketcap being significantly larger than the rest of the variables, so we scaled them down. From there we were able to get a better reading:

```
Importance of components:
                            PC1    PC2    PC3    PC4     PC5
Standard deviation     1.5027 1.0002 0.9180 0.8752 0.36435
Proportion of Variance 0.4516 0.2001 0.1686 0.1532 0.02655
Cumulative Proportion  0.4516 0.6517 0.8203 0.9735 1.00000
> |
```

```
                     PC1   PC2   PC3   PC4   PC5
Currentprice        0.36 -0.01  0.84  0.40  0.11
Marketcap           0.61 -0.06  0.01 -0.36 -0.71
EBITDA.scaled       0.60 -0.04 -0.18 -0.35  0.70
Revenuegrowth      -0.04 -1.00 -0.03  0.09  0.01
Fulltimeemployees   0.38  0.07 -0.51  0.76 -0.08
> |
```

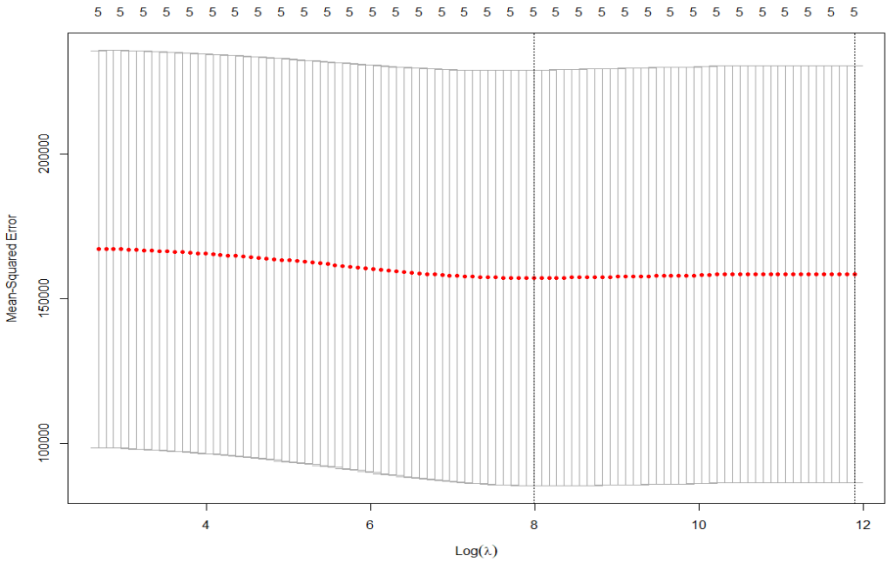
PC1 was accounting for roughly 45.16% of the variance, and it was influenced by marketcap and EBITDA relatively evenly with revenue growth having a negative influence. The correlation between marketcap and EBITDA was also seen in our correlation plot to the right.



Based on our scree plot, we could infer that 1 principal component was substantial enough. We then decided to rotate the factors. When we rotated the factors it was determined that 3 principal components were substantial enough as they accounted for 100% of the variance in our data as opposed to 82.03% of our variance pre-rotation. In summary, marketcap and EBITDA are correlated, despite the observation earlier regarding Tesla not having the same EBITDA to market cap relationship as the other S&P shares. In order to get the above outputs, we needed to remove the companies that did not report EBITDA or their total employee count. Lastly, the PC4 was heavily influenced by the fulltimeemployee variable in addition to the current price.

Ridge Regression:

As we can see based on the visualizations of our Ridge Regression below and to the right, we get a few very important pieces of information. It would be best to look at our original training and testing RMSE. As we can see, our training RMSE was 364.6091 and our testing RMSE was 312.7322. This is very alarming because of the fact that there is such a significant difference in the two RMSEs. This



can be our first sign of overfitting. As for the R^2 value, it is around 15% so that may not be an issue. As for the RMSEs, we decided to run a Ridge Regression to minimize overfitting. In doing so, we got a minimum lambda value of 2940 which seems extremely high. The R^2 also goes down to about 5% (seen at the bottom of the screenshot below). This seems worse; however, we can see that the Ridge RMSE is 300.8339. This means we are at least a little closer to our testing RMSE of 312.7322. This made us want to try Lasso Regression next.

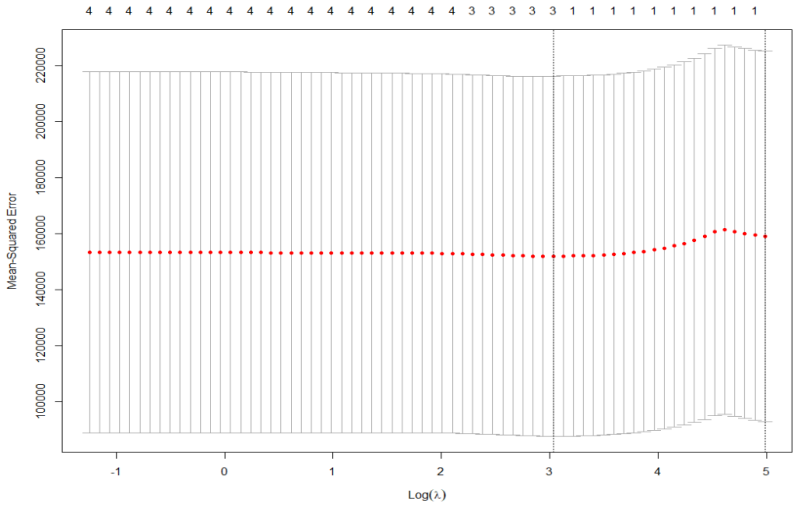
```
[1] "Training set RMSE:"
[1] 364.6091
[1] "Testing set RMSE:"
[1] 312.7322
[1] "Min Lambda:"
[1] 2940.453
[1] "Lambda.1se:"
[1] 146347

Call:  cv.glmnet(x = xTrain, y = yTrain, nfolds = 7, alpha = 0)

Measure: Mean-Squared Error

    Lambda Index Measure    SE Nonzero
min   2940    43  157112 71840       5
1se 146347     1  158415 72085       5
[1] "Ridge RMSE:"
[1] 300.8339
[1] "Testing set RMSE:"
[1] 312.7322

Call:
lm(formula = Currentprice ~ Marketcap + Ebitda + Revenuegrowth +
    Fulltimeemployees + Weight, data = df_companies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-1286.7  -115.4   -62.4    16.1  5323.9

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        1.600e+02  2.130e+01   7.509 3.99e-13 ***
Marketcap          1.452e-07  5.896e-08   2.463   0.0142 *
Ebitda            -4.049e-09  2.680e-09  -1.511   0.1316
Revenuegrowth     -6.447e+00  1.580e+01  -0.408   0.6834
Fulltimeemployees -2.009e-04  1.481e-04  -1.356   0.1757
Weight            -6.158e+06  2.516e+06  -2.448   0.0148 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 367.4 on 396 degrees of freedom
Multiple R-squared:  0.1581,    Adjusted R-squared:  0.1475
F-statistic: 14.88 on 5 and 396 DF,  p-value: 2.185e-13


Call:  glmnet(x = xTrain, y = yTrain, alpha = 0, lambda = 2940.453)

  Df %Dev Lambda
1  5 5.69   2940
[1] "------------------------------------------------------------"
```

Lasso Regression:

```
[1] "Min Lambda:"
[1] 20.74432
[1] "Lambda.1se:"
[1] 146.347
[1] "Lasso RMSE:"
[1] 297.7286
[1] "Testing set RMSE:"
[1] 312.7322

Call:
lm(formula = Currentprice ~ Marketcap + Ebitda + Revenuegrowth +
    Fulltimeemployees + Weight, data = df_companies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-1286.7  -115.4   -62.4    16.1  5323.9

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        1.600e+02  2.130e+01   7.509 3.99e-13 ***
Marketcap          1.452e-07  5.896e-08   2.463   0.0142 *
Ebitda            -4.049e-09  2.680e-09  -1.511   0.1316
Revenuegrowth     -6.447e+00  1.580e+01  -0.408   0.6834
Fulltimeemployees -2.009e-04  1.481e-04  -1.356   0.1757
Weight            -6.158e+06  2.516e+06  -2.448   0.0148 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 367.4 on 396 degrees of freedom
Multiple R-squared:  0.1581,    Adjusted R-squared:  0.1475
F-statistic: 14.88 on 5 and 396 DF,  p-value: 2.185e-13


Call:  glmnet(x = xTrain, y = yTrain, alpha = 1, lambda = 20.74432)

  Df %Dev Lambda
1  3 13.4  20.74
```

As we can see from our screenshot of Lasso above, we got a much more reasonable minimum lambda value of 20.74432. This seems significantly better. Another nice instance that we got from our lasso is that the R^2 value went up to 13.4% which is much closer to our original of 15%. Interestingly enough, we actually got a lower RMSE value with our Lasso regression than we did with Ridge. The Lasso RMSE value we got was 297.7286. Another interesting piece of information we got was that there were only 3 of the 5 variables used by the end of our Lasso Regression. Although the RMSE value is slightly less for Lasso, it seems like there could be a better alpha value in between 0 and 1 that we could use.

Lasso Lambda plot:



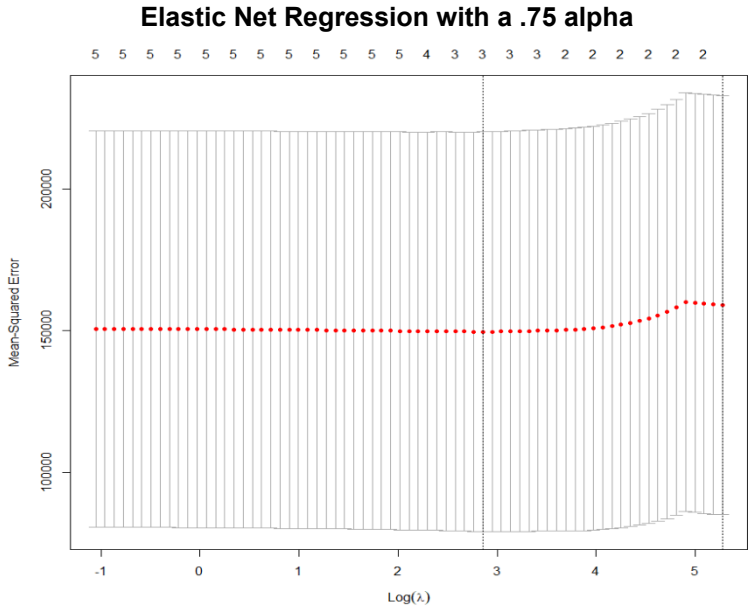
Elastic Net Regression:

```
> rmseElastic25
[1] 297.8684
> rmseElastic5
[1] 320.5059
> rmseElastic75
[1] 307.6612
> rmseCompanyTest
[1] 312.7322
>
```

Elastic Net Regression with .75 alpha:

```
    Lambda Index Measure    SE Nonzero
min  17.37    27  149645 70422       3
1se 195.13     1  159102 73823       0
```

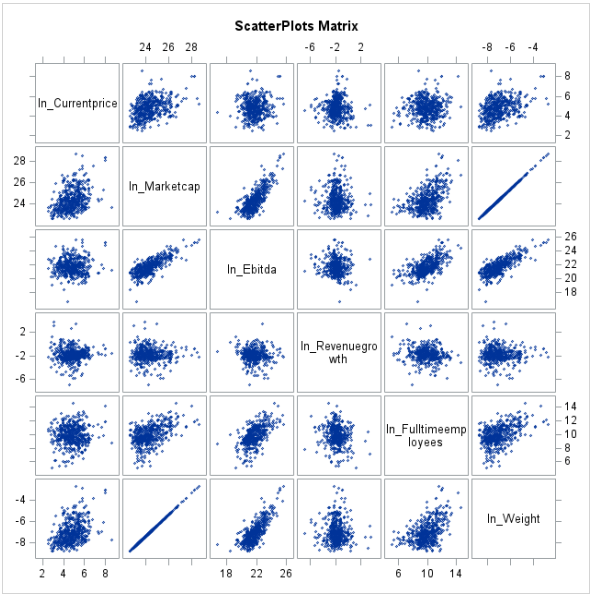
Based on these, we can see that an alpha value of .75 has an rmse of 307.6612 which is the closest value of our elastic nets to our testing RMSE of 312.7322. With our Ridge RMSE having a value of 300 and our Lasso RMSE having a value of 297, we can see that it may be better to use an alpha value of .75 to get closer to our testing RMSE. Elastic Net with an alpha value of .50 is also noted, but we decided to choose the alpha that was the closest to our testing RMSE. Similarly to the Lasso, all three Net Regressions keep 3 variables. This furthers our reasoning as to why an elastic net regression with an alpha value of .75 would be better to use for our model. We can also see that our minimum lambda drops all the way down to 17.37. This gives balance between the alpha being in between the Ridge and Lasso regressions above. The graph of the lambda are below:



Conclusions:

The data told several stories throughout our investigative analysis. Centralizing our look at the companies that made up the S&P, we found unique situations abound from our regression models and other methods of statistical analysis. Starting with goog and googl, who shared almost direct correlation in price over the duration of its existence on the S&P but was determined to have some variations that were statistically unique. The volume would change for a variety of reasons, but it was unique to discover that the Class C shares, which don't come with voting rights and are historically deemed less valuable, closed higher than the Class A shares with more frequency of recent.

Additionally we saw one outlier in particular that ended up coming up twice, which was tesla. As we saw from our initial look, it appeared to be an outlier in our EBITDA vs Market Cap. When we ran our PCA and our correlation plot, we were able to see that EBITDA and MarketCap were in fact correlated.



As we continued forward with our company data, we were able to see that utilizing three of the variables within the company data analysis was recommended amongst elastic net and lasso regression techniques.

Lastly, as we viewed our scatter plots and scatter plot matrix we were able to visualize events such as COVID and their impact on the S&P 500 value, in addition to EBITDA and full time employees shared a positive correlation, but there was no significant correlation between the stocks current price and the quantity of employees they had.