Assignment_STAT6180

Alyssa Lim

5/6/2020

Question 1 - Companies Data

```
companies = read.table("companies.dat", header = TRUE)
```

Correlation Matrix

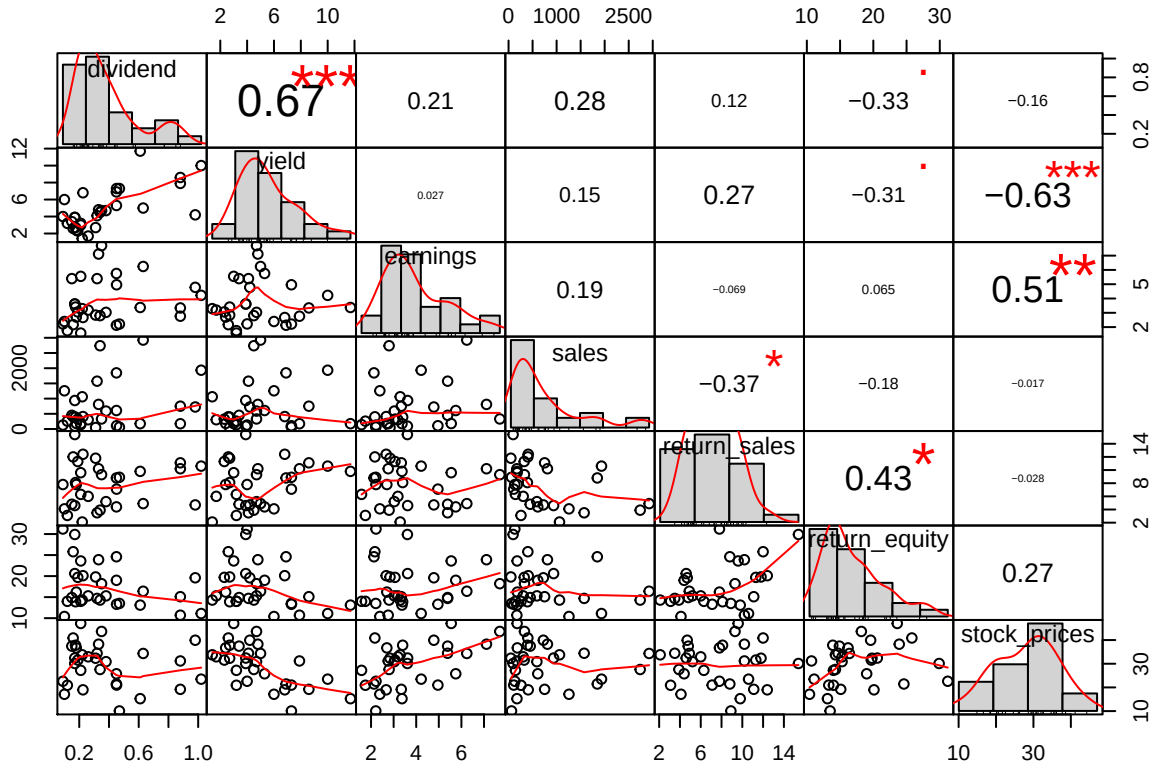
The Correlation Matrix indicates the linear relationship between the variables

- The upper right of the matrix contains the **correlation coefficients**
- The lower left of the matrix illustrate the linear relationship using **scatter plot**
- The diagonal of the matrix shows the **distribution of the variables**

```
library("PerformanceAnalytics")
```

```
## Warning: package 'PerformanceAnalytics' was built under R version 3.6.3

## Loading required package: xts

## Warning: package 'xts' was built under R version 3.6.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 3.6.3

##
## 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
```

```
companies.cor <- companies[, c(2,3,4,5,6,7,8)]
chart.Correlation(companies.cor, histogram = TRUE, pch = 19)
```

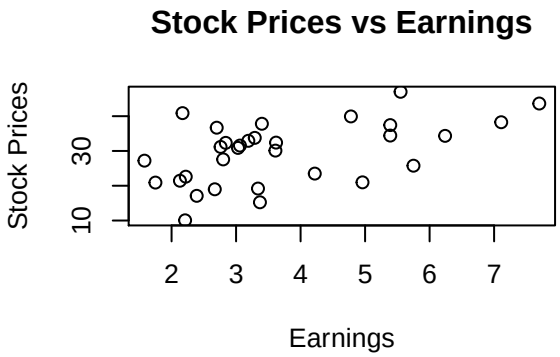


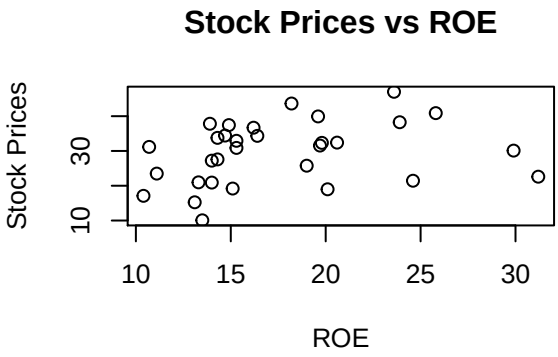
Scatter Plots and Variables Relationship

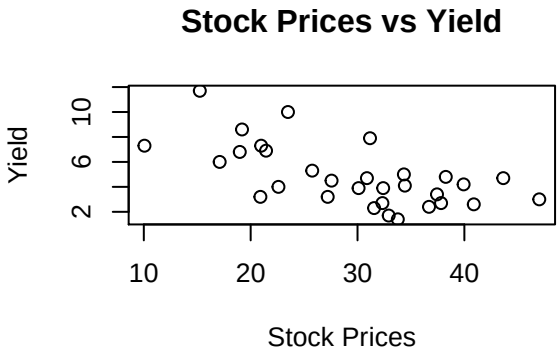
a. Stock Prices Plots

- The **stock price depends on the company earnings**, which means when the earnings is high the stock price will also rise
- The stock price affects yield inversely proportional, **if the stock price is high the yield will decrease**

```
par(mfrow = c(2,2))

plot(companies$earnings, companies$stock_prices, main = "Stock Prices vs Earnings",
     xlab = "Earnings", ylab = "Stock Prices")

plot(companies$return_equity, companies$stock_prices, main = "Stock Prices vs ROE",
     xlab = "ROE", ylab = "Stock Prices")

plot(companies$stock_prices, companies$yield, main = "Stock Prices vs Yield",
     xlab = "Stock Prices", ylab = "Yield")
```



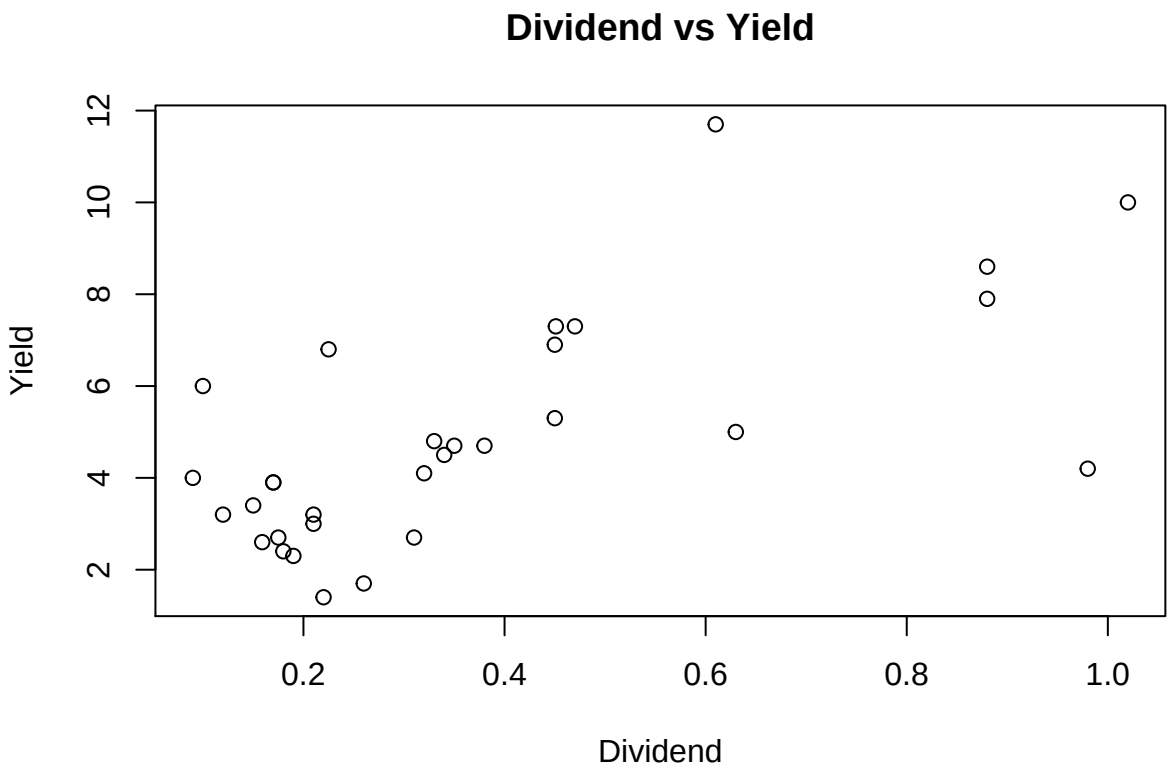




b. Dividend vs Yield Plot

- The yield depends on the dividend, **yield's behaviour is based on the increase or decrease of the dividend issued by the company**

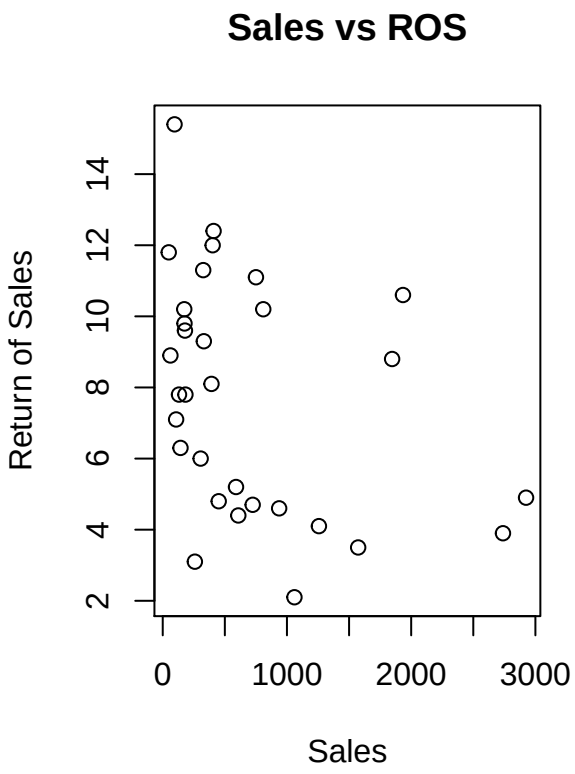
```
plot(companies$dividend, companies$yield, main = "Dividend vs Yield",
     xlab = "Dividend", ylab = "Yield")
```

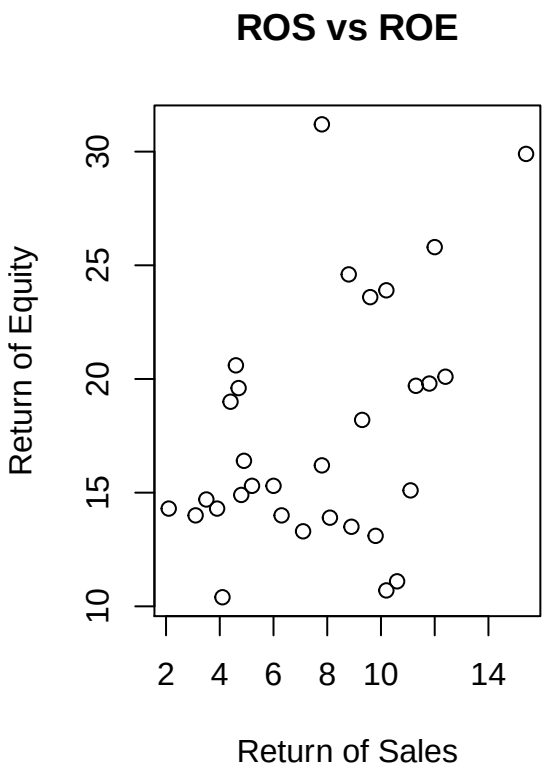


c. ROS Plots

- The **Return of Sales is dependent on sales**, but inversely proportional with each other.
- Return of Sales measures company's efficiency, while Return of Equity measure performance. **Performance increases as company efficiency also increases**.

```
par(mfrow = c(1,2))

plot(companies$sales, companies$return_sales, main = "Sales vs ROS",
     xlab = "Sales", ylab = "Return of Sales")

plot(companies$return_sales, companies$return_equity,
     main = "ROS vs ROE", xlab = "Return of Sales",
     ylab = "Return of Equity")
```





Regression Model

Regression Summary

RESPONSE = Stock Prices

PREDICTORS = All

- $R^2 = 0.7717$ indicates that the model is strong, this is due to all predictors are used in fitting the model
- $P - Value = 1.139e^{-06}$ which is less than 0.05 significance level, which indicates linear relationship between the response and all predictors

```
fitAll = lm(stock_prices ~ ., data = companies.cor)
summary(fitAll)
```

```
##
## Call:
## lm(formula = stock_prices ~ ., data = companies.cor)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9692 -3.4477  0.3714  3.0018  8.0128
##
```

```
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   28.0223916  4.4664430   6.274 1.74e-06 ***
## dividend      10.3797828  4.8012590   2.162   0.0408 *
## yield         -3.3987596  0.5179006  -6.563 8.69e-07 ***
## earnings       2.7203359  0.5791694   4.697 8.97e-05 ***
## sales          0.0003916  0.0013411   0.292   0.7728
## return_sales   0.6787534  0.3695837   1.837   0.0787 .
## return_equity -0.0842791  0.2165979  -0.389   0.7006
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.757 on 24 degrees of freedom
## Multiple R-squared:  0.7717, Adjusted R-squared:  0.7146
## F-statistic: 13.52 on 6 and 24 DF,  p-value: 1.139e-06
```

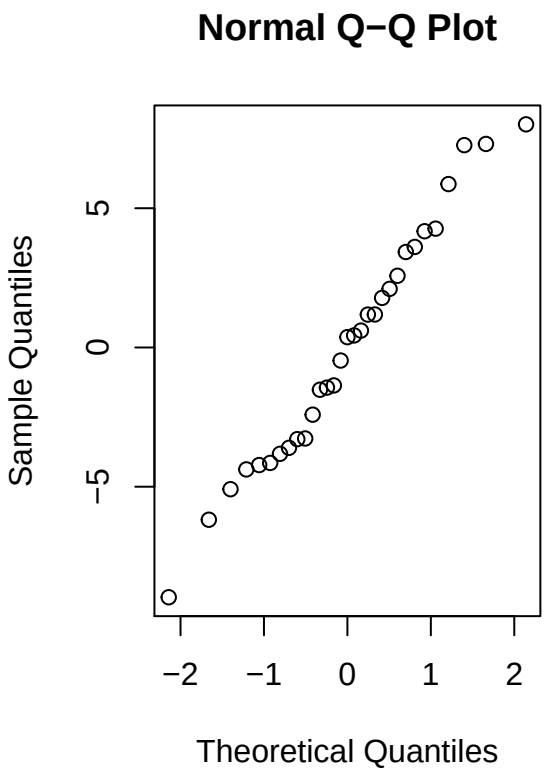
Validating the Full Regression Model

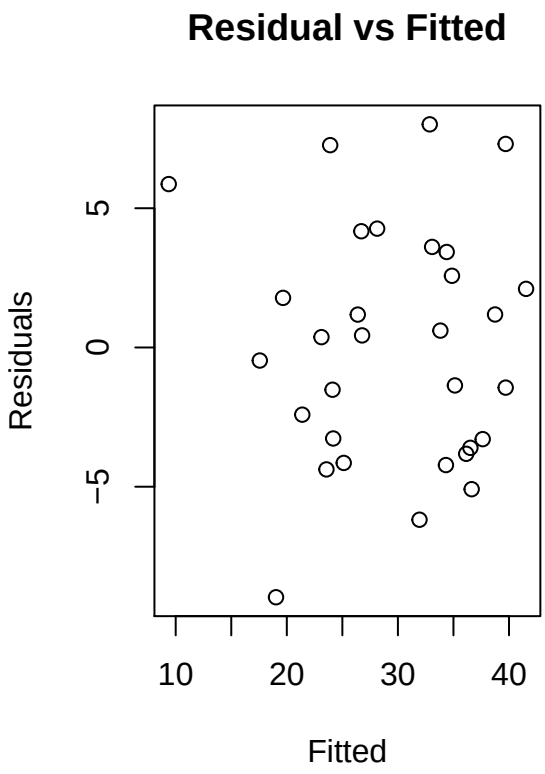
$\varepsilon \sim N(0, \sigma^2)$

- Q-Q plot indicates normal residual as it follows a straight line
- Residual vs Fitted plot doesn't contain any pattern which confirms constant variance

Therefore, Full Regression Analysis is Valid

```r
par(mfrow = c(1, 2))

qqnorm(fitAll$residuals)
plot(fitAll$fitted, fitAll$residuals, main = "Residual vs Fitted",
     xlab = "Fitted", ylab = "Residuals")
```





95% Confidence Interval of the Slope

The confidence interval shows the range, where the possible true value of the slope, for Earnings lies.

```
confint(fitAll, 'earnings', level=0.95)
```

```
##              2.5 %   97.5 %
## earnings 1.524989 3.915683
```

Multiple Regression

Multiple Regression Summary

RESPONSE : Stock Prices

PREDICTORS : All

```
summary(fitAll)
```

```
##
## Call:
## lm(formula = stock_prices ~ ., data = companies.cor)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9692 -3.4477  0.3714  3.0018  8.0128
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   28.0223916  4.4664430   6.274 1.74e-06 ***
## dividend      10.3797828  4.8012590   2.162   0.0408 *
## yield         -3.3987596  0.5179006  -6.563 8.69e-07 ***
## earnings       2.7203359  0.5791694   4.697 8.97e-05 ***
## sales          0.0003916  0.0013411   0.292   0.7728
## return_sales   0.6787534  0.3695837   1.837   0.0787 .
## return_equity -0.0842791  0.2165979  -0.389   0.7006
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.757 on 24 degrees of freedom
## Multiple R-squared:  0.7717, Adjusted R-squared:  0.7146
## F-statistic: 13.52 on 6 and 24 DF,  p-value: 1.139e-06
```

Hypothesis:

$H_0 : \beta_1 + \beta_2 + \beta_3 + \beta_4 + \beta_5 + \beta_6 = 0$

$H_1 : \text{not all } \beta_i = 0$

```
aov.fitAll = anova(fitAll)
aov.fitAll
```

```
## Analysis of Variance Table
##
## Response: stock_prices
##               Df  Sum Sq Mean Sq F value    Pr(>F)
## dividend       1   62.26   62.26  2.7509  0.110214
## yield          1 1182.77 1182.77 52.2623 1.806e-07 ***
## earnings       1  488.63  488.63 21.5908  0.000102 ***
## sales          1   10.25   10.25  0.4530  0.507366
## return_sales   1   88.31   88.31  3.9023  0.059829 .
## return_equity  1    3.43    3.43  0.1514  0.700630
## Residuals     24  543.15   22.63
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Test Statistic:

```
MS = aov.fitAll$`Mean Sq`
Full_regSS = MS[1] + MS[2] + MS[3] + MS[4] + MS[5] + MS[6]
RegMS = Full_regSS/6
Fobs = RegMS/MS[7]
cat('Fobs = ', Fobs)
```

```
## Fobs =  13.51843
```

Null Distribution:

$Pr(F_{6,24} \geq F_{obs} |$ all $\beta_i = 0)$

P-Value:

```
cat('F6,24 = ', qf(.95, df1 = 6, df2 = 24))
```

```
## F6,24 =  2.508189
```

$Pr(F_{6,24} \geq F_{obs}) = 2.51 < 13.52$

```
Pval = pf(q = 13.51843, df1 = 6, df2 = 24, lower.tail = FALSE)
cat('P-value =', Pval)
```

```
## P-value = 1.138712e-06
```

Conclusion:

- $P(F_{6,24} < F_{obs})$, significant evidence to reject the null hypothesis
- P-value is less than 0.05 significance level, which indicates that our model is a regression model
- Though looking at the F-test output of each variables, we can see that Sales and Return seems to be insignificant predictors for these model

Backward Model Selection

1. All Predictors

Sales gives the highest t-test output of **0.7728**, which mean it is the most insignicant predictor in the model.

```
summary(fitAll)
```

```
##
## Call:
## lm(formula = stock_prices ~ ., data = companies.cor)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9692 -3.4477  0.3714  3.0018  8.0128
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   28.0223916  4.4664430   6.274 1.74e-06 ***
## dividend      10.3797828  4.8012590   2.162   0.0408 *
## yield         -3.3987596  0.5179006  -6.563 8.69e-07 ***
## earnings       2.7203359  0.5791694   4.697 8.97e-05 ***
## sales          0.0003916  0.0013411   0.292   0.7728
## return_sales   0.6787534  0.3695837   1.837   0.0787 .
```

```
## return_equity -0.0842791  0.2165979  -0.389   0.7006
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.757 on 24 degrees of freedom
## Multiple R-squared:  0.7717, Adjusted R-squared:  0.7146
## F-statistic: 13.52 on 6 and 24 DF,  p-value: 1.139e-06
```

2. Remove Sales in predictors

Return on Equity (ROE) now gives the highest t-test output of **0.7281**, which means it is also a insignificant predictor in the model.

```
fitFive = lm(stock_prices ~ dividend + yield + earnings + return_sales + return_equity,
             data = companies.cor)
summary(fitFive)
```

```
##
## Call:
## lm(formula = stock_prices ~ dividend + yield + earnings + return_sales +
##     return_equity, data = companies.cor)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.1809 -3.5997  0.2969  2.9546  8.1417
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   28.21292    4.33693   6.505 8.18e-07 ***
## dividend      10.69800    4.58960   2.331   0.0281 *
## yield         -3.37930    0.50411  -6.703 5.03e-07 ***
## earnings       2.73474    0.56641   4.828 5.82e-05 ***
## return_sales   0.63141    0.32600   1.937   0.0641 .
## return_equity -0.07367    0.20959  -0.352   0.7281
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.669 on 25 degrees of freedom
## Multiple R-squared:  0.7709, Adjusted R-squared:  0.725
## F-statistic: 16.82 on 5 and 25 DF,  p-value: 2.689e-07
```

3. Remove Return on Equity in predictors

```
fitFour = lm(stock_prices ~ dividend + yield + earnings + return_sales,
             data = companies.cor)
summary(fitFour)
```

```
##
## Call:
## lm(formula = stock_prices ~ dividend + yield + earnings + return_sales,
##     data = companies.cor)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0146 -3.4573  0.3586  3.2298  7.9402
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   27.1990     3.1837   8.543 5.05e-09 ***
## dividend      11.0291     4.4155   2.498   0.0192 *
## yield         -3.3294     0.4755  -7.002 1.96e-07 ***
## earnings       2.6959     0.5461   4.937 3.97e-05 ***
## return_sales   0.5661     0.2633   2.150   0.0411 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.59 on 26 degrees of freedom
## Multiple R-squared:  0.7697, Adjusted R-squared:  0.7343
## F-statistic: 21.73 on 4 and 26 DF,  p-value: 5.634e-08
```

Therefore the best regression model for the data is:

$Y = 27.2 + (11.03 * dividend) + (-3.33 * yield) + (2.7 * earnings) + (0.57 * ROS)$

- All remaining predictors have T-test output less than 0.05 significance level, which indicates they are signicant predictors
- $R^2 = 0.7697$ indicates strong model, due to the model captures 77% of the variability in the companies data

Question 2 - Prof 2020 Data

Added square and cube of the predictor variety

```
prof = read.table("prof_2020.dat", header = TRUE)
prof$variety2 = prof$variety^2
prof$variety3 = prof$variety^3
head(prof)
```

```
##         state variety mathprof variety2 variety3
## 1     Alabama      78      252     6084   474552
## 2     Arizona      73      259     5329   389017
## 3    Arkansas      77      256     5929   456533
## 4  California      68      256     4624   314432
## 5    Colorado      85      267     7225   614125
## 6 Connecticut      86      270     7396   636056
```

Fitting the Data

Linear Model

```
lr = lm(mathprof ~ variety, data = prof)
summary(lr)
```

```
##
## Call:
## lm(formula = mathprof ~ variety, data = prof)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3623  -3.8687   0.1061   3.6503   8.6629
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 162.1225    12.0493  13.455 2.14e-15 ***
## variety       1.2532     0.1482   8.454 5.66e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.188 on 35 degrees of freedom
## Multiple R-squared:  0.6713, Adjusted R-squared:  0.6619
## F-statistic: 71.47 on 1 and 35 DF,  p-value: 5.659e-10
```

Quadratic Model

```
qr = lm(mathprof ~ variety + variety2, data = prof)
summary(qr)
```

```
##
## Call:
## lm(formula = mathprof ~ variety + variety2, data = prof)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.8465 -2.0569 -0.2999  2.0535  8.9431
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 772.22271  111.60653   6.919 5.66e-08 ***
## variety     -14.23086    2.82561  -5.036 1.54e-05 ***
## variety2      0.09767    0.01781   5.484 4.04e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.834 on 34 degrees of freedom
## Multiple R-squared:  0.8256, Adjusted R-squared:  0.8153
## F-statistic: 80.45 on 2 and 34 DF,  p-value: 1.282e-13
```

Cubic Model

```
cr = lm(mathprof ~ variety + variety2 + variety3, data = prof)
summary(cr)
```

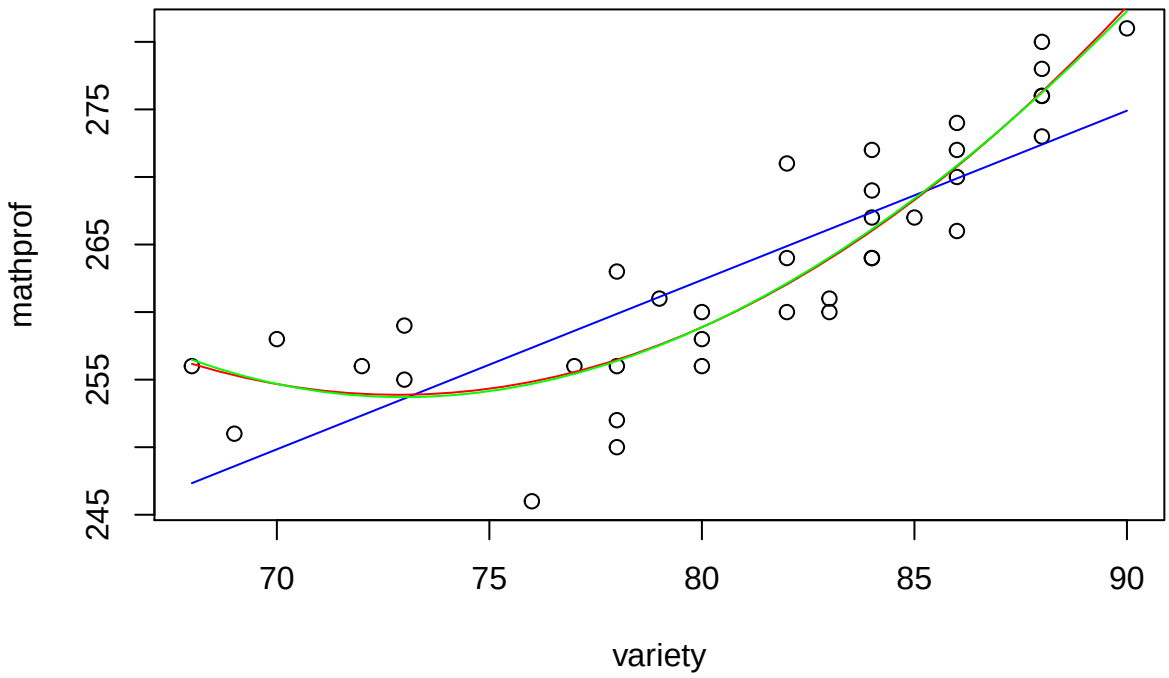
```
##
```

```
## Call:
## lm(formula = mathprof ~ variety + variety2 + variety3, data = prof)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6854 -2.1340 -0.2368  2.1942  8.8665
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.069e+03  1.603e+03   0.667    0.509
## variety     -2.559e+01  6.117e+01  -0.418    0.678
## variety2     2.418e-01  7.756e-01   0.312    0.757
## variety3    -6.070e-04  3.266e-03  -0.186    0.854
##
## Residual standard error: 3.89 on 33 degrees of freedom
## Multiple R-squared:  0.8257, Adjusted R-squared:  0.8099
## F-statistic: 52.12 on 3 and 33 DF,  p-value: 1.293e-12
```

Data Plot

- Linear Fit = BLUE
- Quadratic Fit = RED
- Cubic Fit = GREEN

```
linear = aov(mathprof ~ variety, data = prof)
quad = aov(mathprof ~ variety + I(variety^2), data = prof)
cub = aov(mathprof ~ variety + I(variety^2) + I(variety^3), data = prof)

plot(mathprof ~ variety, data = prof)
varietyDF = data.frame(variety = seq(from = min(prof$variety), to = max(prof$variety),
                                     length = 37))
fitted1 = predict(linear, varietyDF)
fitted2 = predict(quad, varietyDF)
fitted3 = predict(cub, varietyDF)

lines(varietyDF$variety, fitted1, col = 'blue')
lines(varietyDF$variety, fitted2, col = 'red')
lines(varietyDF$variety, fitted3, col = 'green')
```



Conclusion

The best model is Quadratic Model, due to:

- It gives the highest Adjusted R^2 value of 0.8153
- It has the lowest P-value of $1.282e^{-13}$
- In cubic fit, variety3 is insignificant in the model, with the highest T-test output of 0.854
- In quadratic fit all predictors are significant, all predictors has T-test output of less than 0.05 significance level