# Regression

Dillon Carter

02/18

The data for this notebook was provided from here (https://www.kaggle.com/datasets/mirbektoktogaraev/should-this-loan-be-approved-or-denied?select=SBAnational.csv).

*Linear Regression Explanation* Linear Regression seeks to find a line that trends from a predictor or a set of predictors to an output. As the predictors change in some fashion, the line will predict the supposed output. The relationship between the values can be described as the slope of the line w and the intercept b. The intercept tells where the line is expected to begin from and the slope describes how much the output changes with one-unit change in the input predictors. The returned model will have multiple methods of analyzing its accuracy. The predictors will have p and t values to describe how good a predictor they were in defining the slope of the line. The r-squared value will tell how close the predicted value from the found line is to the actual value. The closer the r-squared is to 1, the better the model.

```
if(!require('tidyverse')){
  install.packages('tidyverse')
}
```

```
## Loading required package: tidyverse
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
## ✓ ggplot2 3.4.1     ✓ purrr   1.0.1
## ✓ tibble  3.1.8     ✓ dplyr   1.1.0
## ✓ tidyr   1.3.0     ✓ stringr 1.5.0
## ✓ readr   2.1.4     ✓ forcats 1.0.0
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
```

```
library('tidyverse')
data <- read.csv("data/SBAnational.csv", header=TRUE)
set.seed(02222001)
str(data)
```

```
## 'data.frame':    899164 obs. of  27 variables:
##  $ LoanNr_ChkDgt    : num  1e+09 1e+09 1e+09 1e+09 1e+09 ...
##  $ Name             : chr  "ABC HOBBYCRAFT" "LANDMARK BAR & GRILLE (THE)" "WHITLOCK DDS, T
ODD M." "BIG BUCKS PAWN & JEWELRY, LLC" ...
##  $ City             : chr  "EVANSVILLE" "NEW PARIS" "BLOOMINGTON" "BROKEN ARROW" ...
##  $ State            : chr  "IN" "IN" "IN" "OK" ...
##  $ Zip              : int  47711 46526 47401 74012 32801 6062 7083 34491 32456 6073 ...
##  $ Bank             : chr  "FIFTH THIRD BANK" "1ST SOURCE BANK" "GRANT COUNTY STATE BANK"
"1ST NATL BK & TR CO OF BROKEN" ...
##  $ BankState        : chr  "OH" "IN" "IN" "OK" ...
##  $ NAICS            : int  451120 722410 621210 0 0 332721 0 811118 721310 0 ...
##  $ ApprovalDate     : chr  "28-Feb-97" "28-Feb-97" "28-Feb-97" "28-Feb-97" ...
##  $ ApprovalFY       : chr  "1997" "1997" "1997" "1997" ...
##  $ Term             : int  84 60 180 60 240 120 45 84 297 84 ...
##  $ NoEmp            : int  4 2 7 2 14 19 45 1 2 3 ...
##  $ NewExist         : int  2 2 1 1 1 1 2 2 2 2 ...
##  $ CreateJob        : int  0 0 0 0 7 0 0 0 0 0 ...
##  $ RetainedJob      : int  0 0 0 0 7 0 0 0 0 0 ...
##  $ FranchiseCode    : int  1 1 1 1 1 1 0 1 1 1 ...
##  $ UrbanRural       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ RevLineCr        : chr  "N" "N" "N" "N" ...
##  $ LowDoc           : chr  "Y" "Y" "N" "Y" ...
##  $ ChgOffDate       : chr  "" "" "" "" ...
##  $ DisbursementDate : chr  "28-Feb-99" "31-May-97" "31-Dec-97" "30-Jun-97" ...
##  $ DisbursementGross: chr  "$60,000.00 " "$40,000.00 " "$287,000.00 " "$35,000.00 " ...
##  $ BalanceGross     : chr  "$0.00 " "$0.00 " "$0.00 " "$0.00 " ...
##  $ MIS_Status       : chr  "P I F" "P I F" "P I F" "P I F" ...
##  $ ChgOffPrinGr     : chr  "$0.00 " "$0.00 " "$0.00 " "$0.00 " ...
##  $ GrAppv           : chr  "$60,000.00 " "$40,000.00 " "$287,000.00 " "$35,000.00 " ...
##  $ SBA_Appv         : chr  "$48,000.00 " "$32,000.00 " "$215,250.00 " "$28,000.00 " ...
```

Looking at the data, there are some columns that have too many individual factors to be be useful (City, Zip, Business Name). Each business is unique in its name so there really isn't an ability to predict based off name. Similarly, there are a large number of cities and zip codes in the data that don't have a lot of recurring values so training the data off them will likely not be accurate. The number of jobs created and retained are good information but are results based off the loan being approved. Similarly are the dates for the approval and disbursement. These are outside the scope of what I will be looking at.

Some interesting data to pull from would be the NAICS, indicating the type of business, the term of the loan, whether the business was older than 2 years (NewExist), the UrbanRural divide, whether the borrower is on a revolving line of credit, the gross amount approved by the bank and the amount approved by the SBA. Something that might be interesting to try excluding to see if it affects the accuracy of predictions is whether the borrower is enrolled in the LowDoc loan program. Inclusion in the program will cap the max size of the loan to 150000$.

*The columns to be analyzed are the following: +State (4) +NAICS (8) +Term (11) +NoEmp (12) +NewExist (13) - +UrbanRural (17) - +RevLinCr (18) +LowDoc (19) +GrAppv (26) +SBA_Appv(27)

With the chosen columns in place, refactoring them into better formats is done here. I'm changing stats, NAICS, NewExist, UrbanRural, RevLineCr, and LowDoc into factors as they have a set of known values that the tuple has to be a part of. Gross Approved and SBA Approved are both in non-numeric format, so I use the parse_number to get them into exclusively integer format. Then all NA data is omitted as there is plenty of data to use without it affecting the quality of results.The LowDoc column has some extraneous values outside of the "Y" or "N" desired so those rows are found and removed. Additionally, the NAICS column has some extraneous values in 0's. As this data is based off the 2012 system, there is no sector containing 0 as the leading values. This isn't an insignificant amount of the data though. Roughly a quarter, 200,000 entries, contain a 0. So I'm going to leave those in and see if it cannot be worked around without changing the data or removing them entirely.

```
#Why is there so much weird data in these fields? Who looks at a yes/no question and answers "Q"?
extraneous <- c("0","1","A","C","R","S", "T", "", "`", ",", "3", "2", "7", ".", "4", "-", "
Q")
no_zero <- c(0)
extraneous
```

```
##  [1] "0" "1" "A" "C" "R" "S" "T" ""  "`" "," "3" "2" "7" "." "4" "-" "Q"
```

```
no_zero
```

```
## [1] 0
```

```
str(data)
```

```
## 'data.frame':    899164 obs. of  27 variables:
##  $ LoanNr_ChkDgt    : num  1e+09 1e+09 1e+09 1e+09 1e+09 ...
##  $ Name             : chr  "ABC HOBBYCRAFT" "LANDMARK BAR & GRILLE (THE)" "WHITLOCK DDS, T
ODD M." "BIG BUCKS PAWN & JEWELRY, LLC" ...
##  $ City             : chr  "EVANSVILLE" "NEW PARIS" "BLOOMINGTON" "BROKEN ARROW" ...
##  $ State            : chr  "IN" "IN" "IN" "OK" ...
##  $ Zip              : int  47711 46526 47401 74012 32801 6062 7083 34491 32456 6073 ...
##  $ Bank             : chr  "FIFTH THIRD BANK" "1ST SOURCE BANK" "GRANT COUNTY STATE BANK"
"1ST NATL BK & TR CO OF BROKEN" ...
##  $ BankState        : chr  "OH" "IN" "IN" "OK" ...
##  $ NAICS            : int  451120 722410 621210 0 0 332721 0 811118 721310 0 ...
##  $ ApprovalDate     : chr  "28-Feb-97" "28-Feb-97" "28-Feb-97" "28-Feb-97" ...
##  $ ApprovalFY       : chr  "1997" "1997" "1997" "1997" ...
##  $ Term             : int  84 60 180 60 240 120 45 84 297 84 ...
##  $ NoEmp            : int  4 2 7 2 14 19 45 1 2 3 ...
##  $ NewExist         : int  2 2 1 1 1 1 2 2 2 2 ...
##  $ CreateJob        : int  0 0 0 0 7 0 0 0 0 0 ...
##  $ RetainedJob      : int  0 0 0 0 7 0 0 0 0 0 ...
##  $ FranchiseCode    : int  1 1 1 1 1 1 0 1 1 1 ...
##  $ UrbanRural       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ RevLineCr        : chr  "N" "N" "N" "N" ...
##  $ LowDoc           : chr  "Y" "Y" "N" "Y" ...
##  $ ChgOffDate       : chr  "" "" "" "" ...
##  $ DisbursementDate : chr  "28-Feb-99" "31-May-97" "31-Dec-97" "30-Jun-97" ...
##  $ DisbursementGross: chr  "$60,000.00 " "$40,000.00 " "$287,000.00 " "$35,000.00 " ...
##  $ BalanceGross     : chr  "$0.00 " "$0.00 " "$0.00 " "$0.00 " ...
##  $ MIS_Status       : chr  "P I F" "P I F" "P I F" "P I F" ...
##  $ ChgOffPrinGr     : chr  "$0.00 " "$0.00 " "$0.00 " "$0.00 " ...
##  $ GrAppv           : chr  "$60,000.00 " "$40,000.00 " "$287,000.00 " "$35,000.00 " ...
##  $ SBA_Appv         : chr  "$48,000.00 " "$32,000.00 " "$215,250.00 " "$28,000.00 " ...
```

```
sum(data$NAICS==0)
```

```
## [1] 201948
```

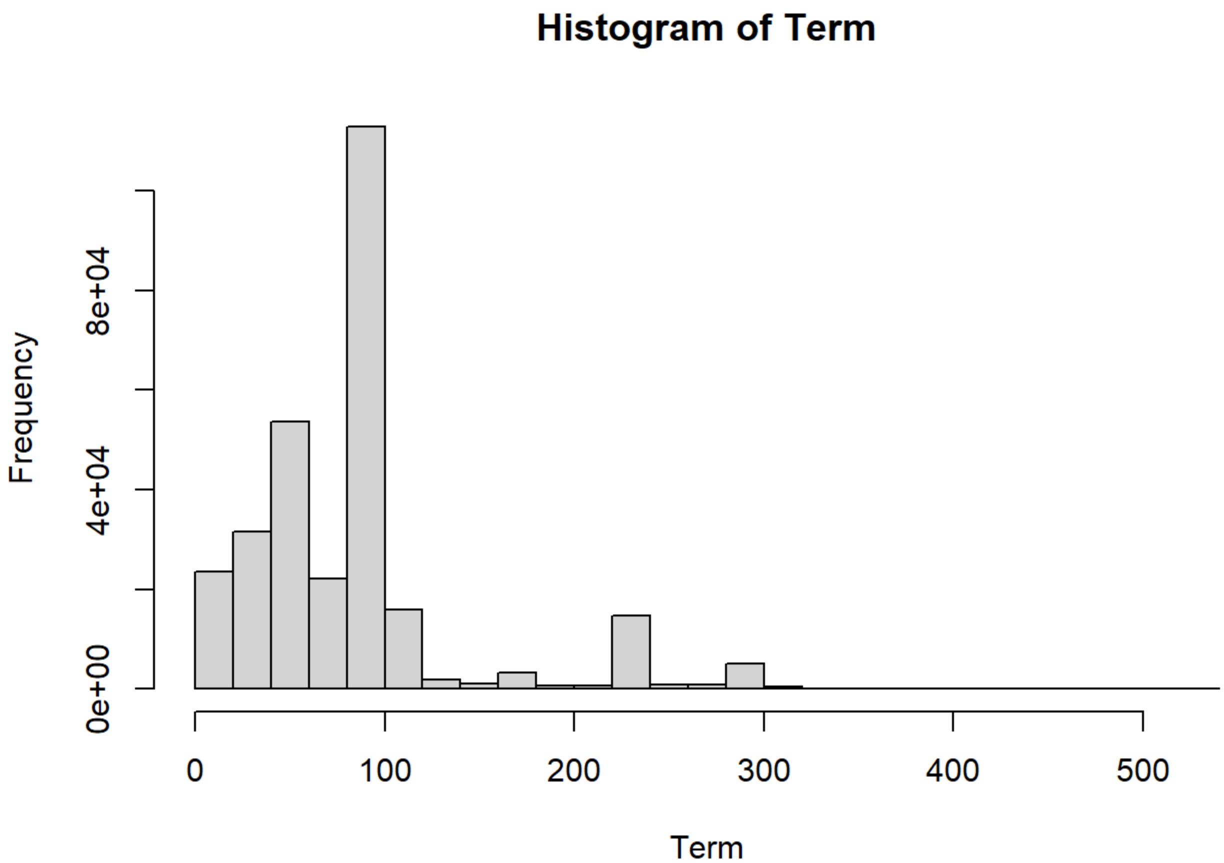
```
data <- data[-c(which(data$LowDoc %in% extraneous)),]
data <- data[-c(which(data$RevLineCr %in% extraneous)),]
data <- data[-c(which(data$NewExist %in% no_zero)),]
data <- data[-c(which(data$UrbanRural %in% no_zero)),]
data <- na.omit(data)
```

```
data$State <- factor(data$State)
data$NAICS <- as.numeric(substring(data$NAICS, 1, 2))
data$NAICS <- factor(data$NAICS)
data$NewExist <- factor(data$NewExist)
data$UrbanRural <- factor(data$UrbanRural)
data$RevLineCr <- factor(data$RevLineCr)
data$LowDoc <- factor(data$LowDoc)
data$GrAppv <- parse_number(data$GrAppv)
data$SBA_Appv <- parse_number(data$SBA_Appv)
data <- data[,c(4,8,11,12,13,17,18,19,26,27)]

#Separate training and test data
i <- sample(1:nrow(data), 0.8*nrow(data), replace=FALSE)
train <- data[i,]
test <- data[-i,]

attach(train)
```

Now, before I get onto actually looking at predictors in a linear regression model, let's look at some relationships between the predictors and the amounts we want to predict, i.e. the gross amount approved by the bank and the gross amount approved by the SBA.

One predictor that can lend some information about the loan is the term. Longer term loans will take a while for the lender to see back the money they lent in addition to the interest. So, they typically give them to businesses they are more sure about and can be confident in the fact that they will likely get their money back.

```
hist(Term)
```




The vast majority of the loans fall under 100 months, which is about 8 and a half years. This is a relatively short loan and gives an idea that the loans given out under the SBA program are typically on the smaller side. There are peaks around 240 and 280 months, Which are around 20 years. These are likely larger loans that have smaller interest payments but the lender can be almost assured they will be paid back.

```
summary(Term)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   50.00   84.00   82.73   84.00  527.00
```

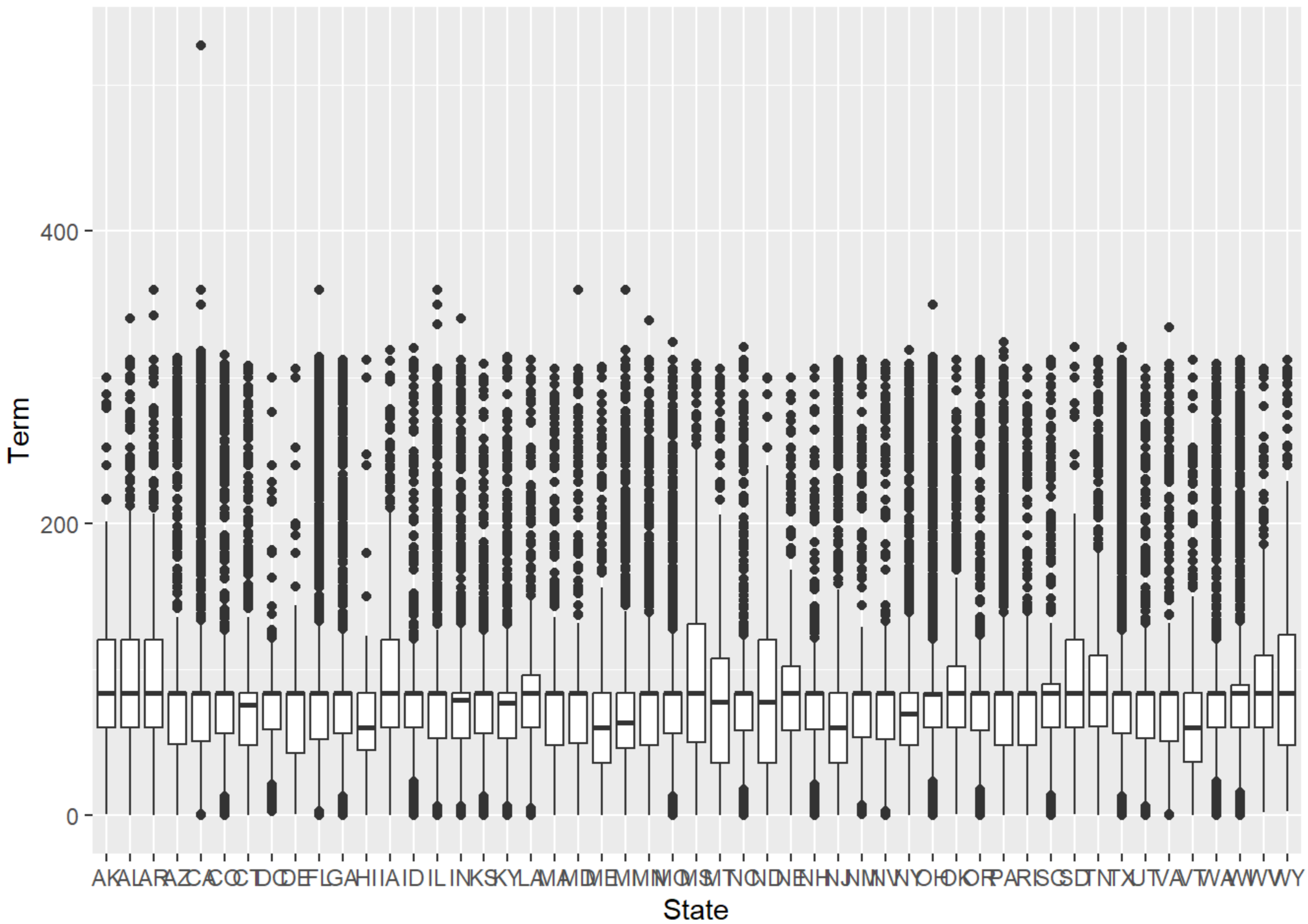
The summary confirms the brief look at the histogram above, with a weird outlier in the max at 527 months or 40 years. This could be an outlier where the borrower kept delaying loan payments and the term kept being extended. Luckily, it seems to be an extreme outlier based on the other summary factors.

Something I am interested in is the correlation between loan terms and the states that business are incorporated in. I would expect that since the SBA is a national program, loan terms stay roughly the same between the states, but also for wealthier states to have a higher max than less wealthy states.

```
levels(State)
```

```
##  [1] "AK" "AL" "AR" "AZ" "CA" "CO" "CT" "DC" "DE" "FL" "GA" "HI" "IA" "ID" "IL"
## [16] "IN" "KS" "KY" "LA" "MA" "MD" "ME" "MI" "MN" "MO" "MS" "MT" "NC" "ND" "NE"
## [31] "NH" "NJ" "NM" "NV" "NY" "OH" "OK" "OR" "PA" "RI" "SC" "SD" "TN" "TX" "UT"
## [46] "VA" "VT" "WA" "WI" "WV" "WY"
```

```
options(repr.plot.width=25, repr.plot.height=9)
ggplot(data, aes(x=State, y=Term)) +
  geom_boxplot(notch=FALSE)
```




As suspected, the line for the mean loan term is almost a horizontal line across all the states. Hawai'i, Vermont, New York and New Jersey are exceptions with means dipping below the general trend. Vermont is understandable as it has the lowest GDP of all states, but the other 3 seem somewhat strange. Maybe there is a better correlation between the term and the number of employees? Let's first examine the statistics of the number of employees.

```
summary(NoEmp)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    0.000    2.000    3.000    8.328    8.000 5000.000
```

The size of businesses seem to be grouped around less than 10 employees, with a few outliers with thousands of employees. So how many business are there that have more than 10, 100, or 1000 employees and how do those stats compare with the whole pie?

```
#More than 10
length(which(NoEmp > 10))
```

```
## [1] 52461
```

```
summary(NoEmp[NoEmp > 10])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    11.0    14.0    20.0    30.5    30.0  5000.0
```

```
#More than 100
length(which(NoEmp > 100))
```

```
## [1] 1502
```

```
summary(NoEmp[NoEmp > 100])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   101.0   120.0   150.0   210.3   200.0  5000.0
```

```
#More than 1000
length(which(NoEmp > 1000))
```

```
## [1] 22
```

```
summary(NoEmp[NoEmp > 1000])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1003    1455    1664    2324    2426    5000
```

Looking at the results, it is clear that the vast majority of the borrowers in the dataset have 10 or less employees. A larger minority have 100 or less and the rest are larger businesses. Now looking at a general correlation between the term and the number of employees.

```
cor(Term, NoEmp)
```

```
## [1] 0.08194305
```

So that would seem to suggest that the size of the borrower doesn’t necessarily relate to the stability of the borrower in the lender’s eyes.

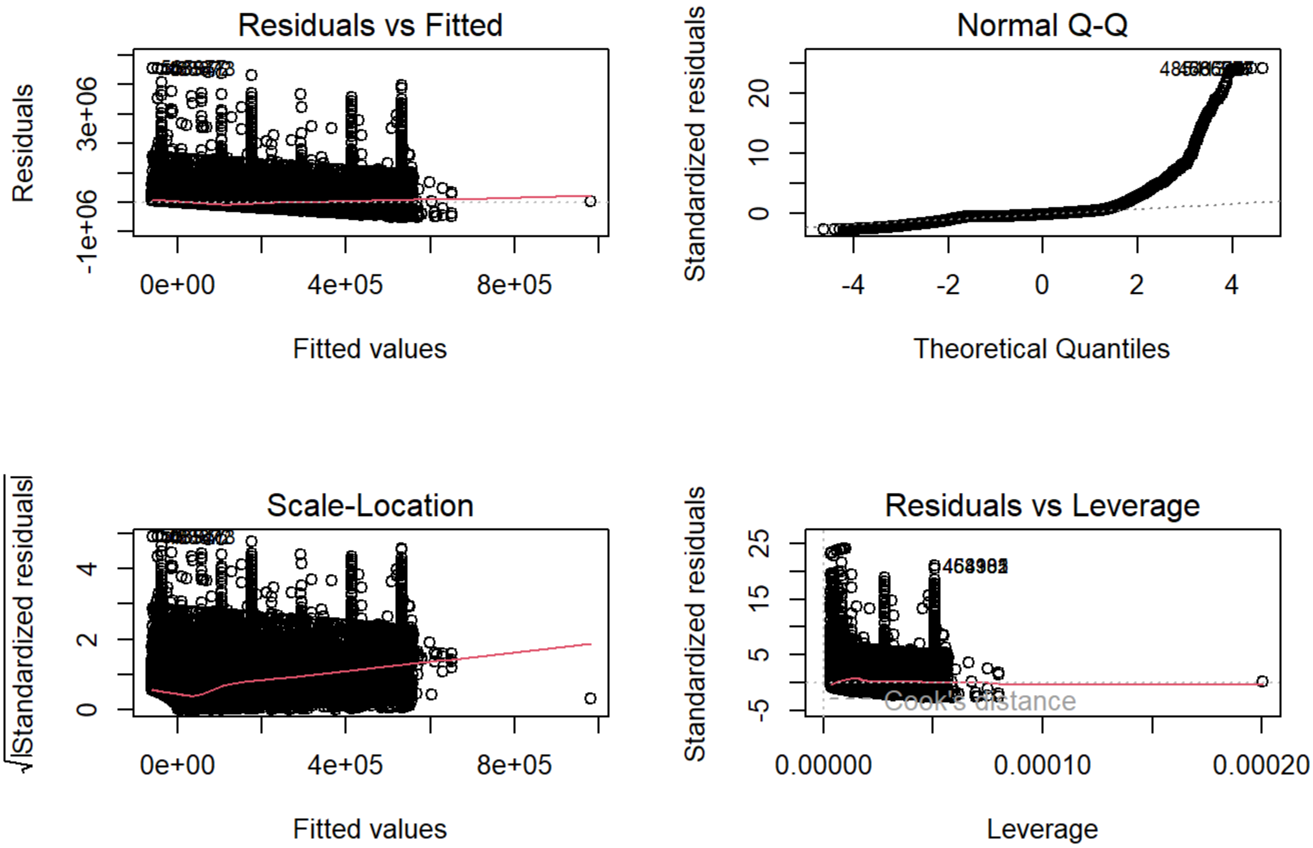
Let’s plot the term against the SBA approved

```
lm1 <- lm(SBA_Appv~Term, data=train)
summary(lm1)
```

```
##
## Call:
## lm(formula = SBA_Appv ~ Term, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -512028  -86616  -39161   25682 4558024
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -61978.40     606.42  -102.2   <2e-16 ***
## Term          1977.32       5.97   331.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 189000 on 288642 degrees of freedom
## Multiple R-squared:  0.2754, Adjusted R-squared:  0.2754
## F-statistic: 1.097e+05 on 1 and 288642 DF,  p-value: < 2.2e-16
```

The term and the SBA Approved amount don’t correlate all that well. There is a tiny p value, meaning the null hypothesis that the two values are unrelated can be rejected. So the model was good but the actual data didn’t track well onto each other. The adjusted R-squared being just .27 means that changes in the term really don’t necessarily affect changes in the expected SBA approved. Plotting the residuals.

```
par(mfrow=c(2,2))
plot(lm1)
```



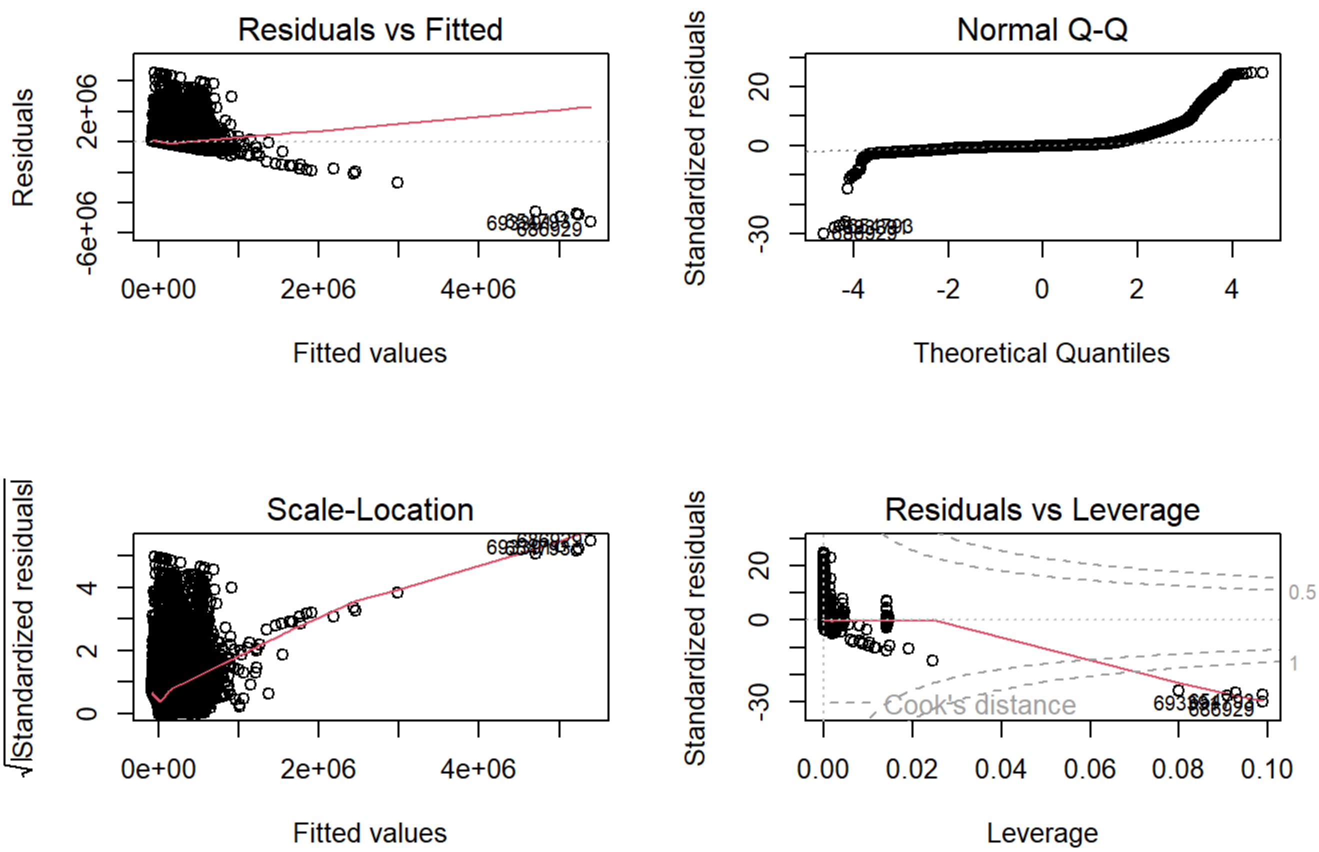

Looking at the trend line of the residuals against fitted graph, it’s smooth but doesn’t follow a discernible line. The Normal Q-Q is good for theoretical quantiles up to 2 where it deviates greatly. The Scale-Location graph is similar to the 1st residuals graph. It is mostly linear with a non-linear bump near 0. Finally the residuals vs leverage graph has an outlier with a much higher influence than the bulk of the data. This could have skewed results. So building off the simple linear regression, let’s look at one with more predictors, including NoEmp, NAICS, and Term.

```
lm2 <- lm(SBA_Appv~Term+NAICS+NoEmp, data=train)
summary(lm2)
```

```
##
## Call:
## lm(formula = SBA_Appv ~ Term + NAICS + NoEmp, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5253647   -72946   -33229    28019  4557352
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -57269.340   2285.493 -25.058  < 2e-16 ***
## Term          1904.060      5.973 318.790  < 2e-16 ***
## NAICS11      73351.803   4049.245  18.115  < 2e-16 ***
## NAICS21      95743.711   7784.251  12.300  < 2e-16 ***
## NAICS22      -1140.606  12267.730  -0.093 0.925923
## NAICS23     -18566.484   2424.148  -7.659 1.88e-14 ***
## NAICS31      18347.129   3514.948   5.220 1.79e-07 ***
## NAICS32      44043.091   3148.520  13.989  < 2e-16 ***
## NAICS33      52369.829   2680.743  19.536  < 2e-16 ***
## NAICS42      40205.578   2545.714  15.793  < 2e-16 ***
## NAICS44      -8634.434   2402.291  -3.594 0.000325 ***
## NAICS45     -30986.597   2661.514 -11.642  < 2e-16 ***
## NAICS48     -18407.875   2829.736  -6.505 7.77e-11 ***
## NAICS49      -8572.690   6240.703  -1.374 0.169544
## NAICS51      -9878.059   3422.400  -2.886 0.003898 **
## NAICS52     -30684.254   3406.647  -9.007  < 2e-16 ***
## NAICS53     -18272.182   3146.278  -5.808 6.35e-09 ***
## NAICS54     -27115.144   2409.405 -11.254  < 2e-16 ***
## NAICS55      24116.571  22130.866   1.090 0.275835
## NAICS56     -36837.792   2627.523 -14.020  < 2e-16 ***
## NAICS61     -36067.364   3946.420  -9.139  < 2e-16 ***
## NAICS62     -13177.672   2555.004  -5.158 2.50e-07 ***
## NAICS71      -6959.914   3330.644  -2.090 0.036649 *
## NAICS72      11084.690   2452.256   4.520 6.18e-06 ***
## NAICS81     -33875.991   2461.253 -13.764  < 2e-16 ***
## NAICS92     -44701.793  22453.090  -1.991 0.046493 *
## NoEmp         1046.013     11.599  90.183  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 184300 on 288617 degrees of freedom
## Multiple R-squared:  0.3115, Adjusted R-squared:  0.3114
## F-statistic:  5022 on 26 and 288617 DF,  p-value: < 2.2e-16
```

For some of the NAICS predictors, the model is good; others not so much. But the R-square is still not good. .3098 is far from 1 but at least a little bit better than .27.

```
par(mfrow=c(2,2))
plot(lm2)
```



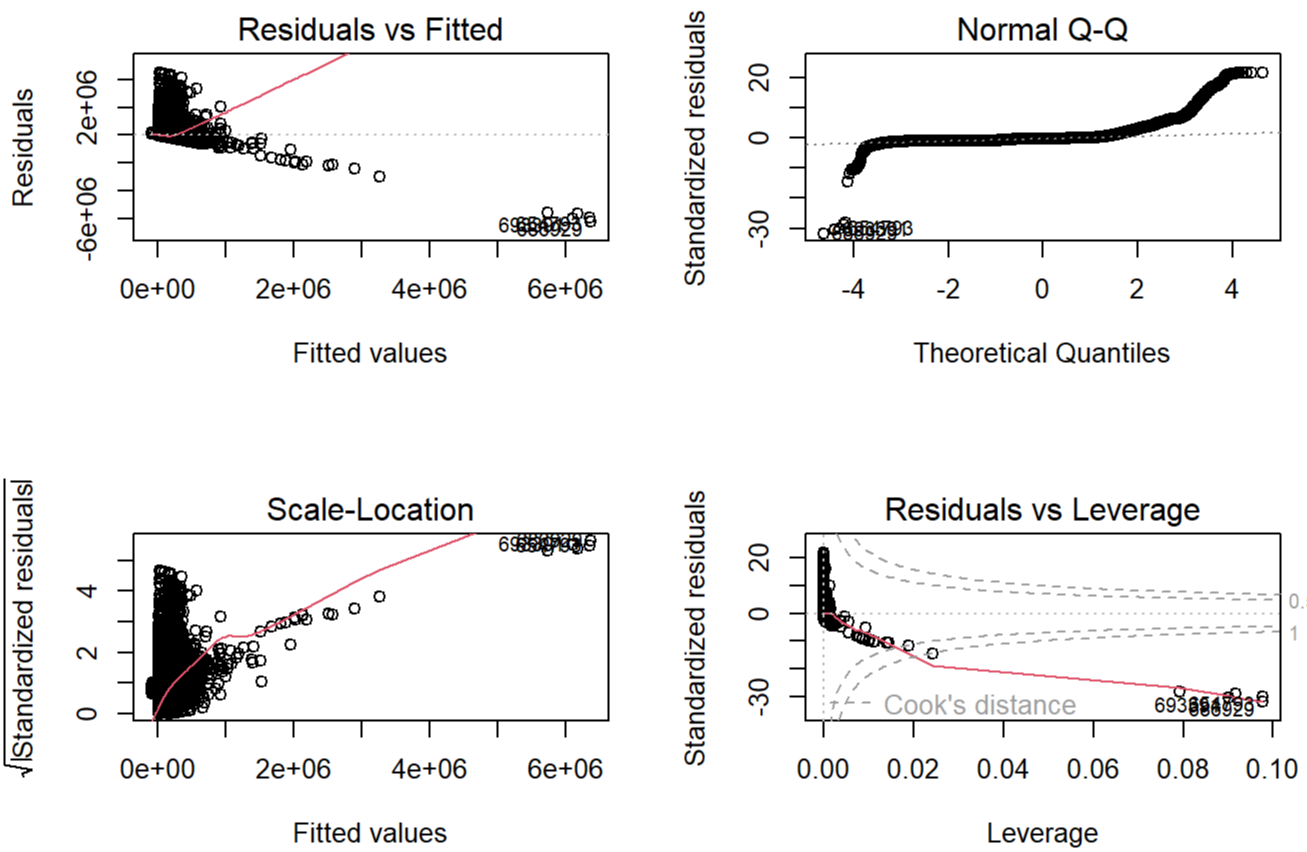

Maybe plotting against a polynomial model would be better.

```
lm3 <- lm(SBA_Appv~poly(State, NoEmp, LowDoc, RevLineCr), data=train)
summary(lm3)
```

```
##
## Call:
## lm(formula = SBA_Appv ~ poly(State, NoEmp, LowDoc, RevLineCr),
##     data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6222940  -102338   -17594     2610  4474564
##
## Coefficients:
##                                              Estimate Std. Error t value
## (Intercept)                                    101605        383  265.27
## poly(State, NoEmp, LowDoc, RevLineCr)1.0.0.0  -3364760     205796  -16.35
## poly(State, NoEmp, LowDoc, RevLineCr)0.1.0.0  19807163     206255   96.03
## poly(State, NoEmp, LowDoc, RevLineCr)0.0.1.0  -3722444     206244  -18.05
## poly(State, NoEmp, LowDoc, RevLineCr)0.0.0.1 -38855702     206709 -187.97
##                                             Pr(>|t|)
## (Intercept)                                   <2e-16 ***
## poly(State, NoEmp, LowDoc, RevLineCr)1.0.0.0   <2e-16 ***
## poly(State, NoEmp, LowDoc, RevLineCr)0.1.0.0   <2e-16 ***
## poly(State, NoEmp, LowDoc, RevLineCr)0.0.1.0   <2e-16 ***
## poly(State, NoEmp, LowDoc, RevLineCr)0.0.0.1   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 205800 on 288639 degrees of freedom
## Multiple R-squared:  0.1413, Adjusted R-squared:  0.1413
## F-statistic: 1.187e+04 on 4 and 288639 DF,  p-value: < 2.2e-16
```

No that definitely made it worse. R-squared went down to .14.

```
par(mfrow=c(2,2))
plot(lm3)
```




```
#Compare models
anova(lm1, lm2)
```

| | Res.Df <dbl> | RSS <dbl> | Df <dbl> | Sum of Sq <dbl> | F <dbl> | Pr(>F) <dbl> |
|---|---|---|---|---|---|---|
| 1 | 288642 | 1.031407e+16 | NA | NA | NA | NA |
| 2 | 288617 | 9.800095e+15 | 25 | 5.139743e+14 | 605.4706 | 0 |

2 rows

```
anova(lm1, lm3)
```

| | Res.Df <dbl> | RSS <dbl> | Df <dbl> | Sum of Sq <dbl> | F <dbl> | Pr(>F) <dbl> |
|---|---|---|---|---|---|---|
| 1 | 288642 | 1.031407e+16 | NA | NA | NA | NA |
| 2 | 288639 | 1.222266e+16 | 3 | -1.908589e+15 | NA | NA |

2 rows

```
anova(lm2, lm3)
```

| | Res.Df <dbl> | RSS <dbl> | Df <dbl> | Sum of Sq <dbl> | F <dbl> | Pr(>F) <dbl> |
|---|---|---|---|---|---|---|
| 1 | 288617 | 9.800095e+15 | NA | NA | NA | NA |
| 2 | 288639 | 1.222266e+16 | -22 | -2.422564e+15 | 3242.979 | 0 |

2 rows

With all three models plotted alongside with their residuals, the best model provided thus far was the secondary multiple predictor linear model. All the models had low p and t values meaning that the linear regression algorithm was at least mostly accurate. But, the r-squared for all was low. It was highest for the multiple predictor model which makes it the best case when trying to find an appropriate SBA_Appr from the predictors.

```
pred1 <- predict(lm1, newdata=test)
cor1 <- cor(pred1, test$SBA_Appv)
mse1 <- mean((pred1-test$SBA_Appv)^2)
rmse1 <- sqrt(mse1)
```

```
pred2 <- predict(lm2, newdata=test)
cor2 <- cor(pred2, test$SBA_Appv)
mse2 <- mean((pred2-test$SBA_Appv)^2)
rmse2 <- sqrt(mse2)
```

```
pred3 <- predict(lm3, newdata=test)
cor3 <- cor(pred3, test$SBA_Appv)
mse3 <- mean((pred3-test$SBA_Appv)^2)
rmse3 <- sqrt(mse3)
```

```
print(paste("Model 1: Correlation: ", cor1))
```

```
## [1] "Model 1: Correlation:  0.529689466713622"
```

```
print(paste("mse: ", mse1))
```

```
## [1] "mse:  34281492548.4086"
```

```
print(paste("rmse: ", rmse1))
```

```
## [1] "rmse:  185152.619609901"
```

```
print(paste("Model 2: Correlation: ", cor2))
```

```
## [1] "Model 2: Correlation:  0.549908607051546"
```

```
print(paste("mse: ", mse2))
```

```
## [1] "mse:  33275465826.8794"
```

```
print(paste("rmse: ", rmse2))
```

```
## [1] "rmse:  182415.640302249"
```

```
print(paste("Model 3: Correlation: ", cor3))
```

```
## [1] "Model 3: Correlation:  0.356415054363853"
```

```
print(paste("mse: ", mse3))
```

```
## [1] "mse:  41711499768.0784"
```

```
print(paste("rmse: ", rmse3))
```

```
## [1] "rmse:  204233.93392891"
```

All in total, the correlation of the 1st 2 models are much closer than the 3rd model. It is likely not a polynomial line then. The higher correlation of the multiple predictor model seems to indicate that the method to predict the approved amount by the SBA will be through as many of the predictors as possible. This would make sense with the understanding behind the dataset. The lender will want as much information as possible about who they are lending to before they start giving out larger amounts of money. Some locations and types of businesses will be considered more reliable and will be given longer terms and allowed to borrow more.