HW5: Regression

Question 8.1

- I work within the field of financial investments. Specifically I work to model stock prices and try to understand which companies will perform well, and which will not

Although I work on equities, I often need to model their credit as well, specifically their yield on their debt. This is the amount of interest you would recieve if you were to give them a loan. It is an important metric to see how likely the firm may be to default, and how much they have to spend to fulfill their debt obligations.

However, For many companies, we do not have data for their debt yield, meaning we need to predict it usng linear regression. A few of the predictors we can use are the following:

- The credit rating, if provided
  - This is a categorization given by an agency which buckets firms based on their likelihood to default
- The amount of debt to assets the firm has (their leverage)
- Their profitability relative to their debt obligations (how much profit they earn vs how much interest they need to pay)
- The size of the firm, as a proxy for the liquidity of the debt obligation

Question 8.2

First load in libraries and data

```
library("ggplot2")
library("dplyr")
library("tidyr")
library("stats")
library("corrplot")

setwd("C:/Users/phene/OneDrive/R files")
df <- read.csv("uscrime.csv", stringsAsFactors = FALSE)
```

Taking a look at the dataset

We have 15 predictor variables, one of which is binary, along with the respnse

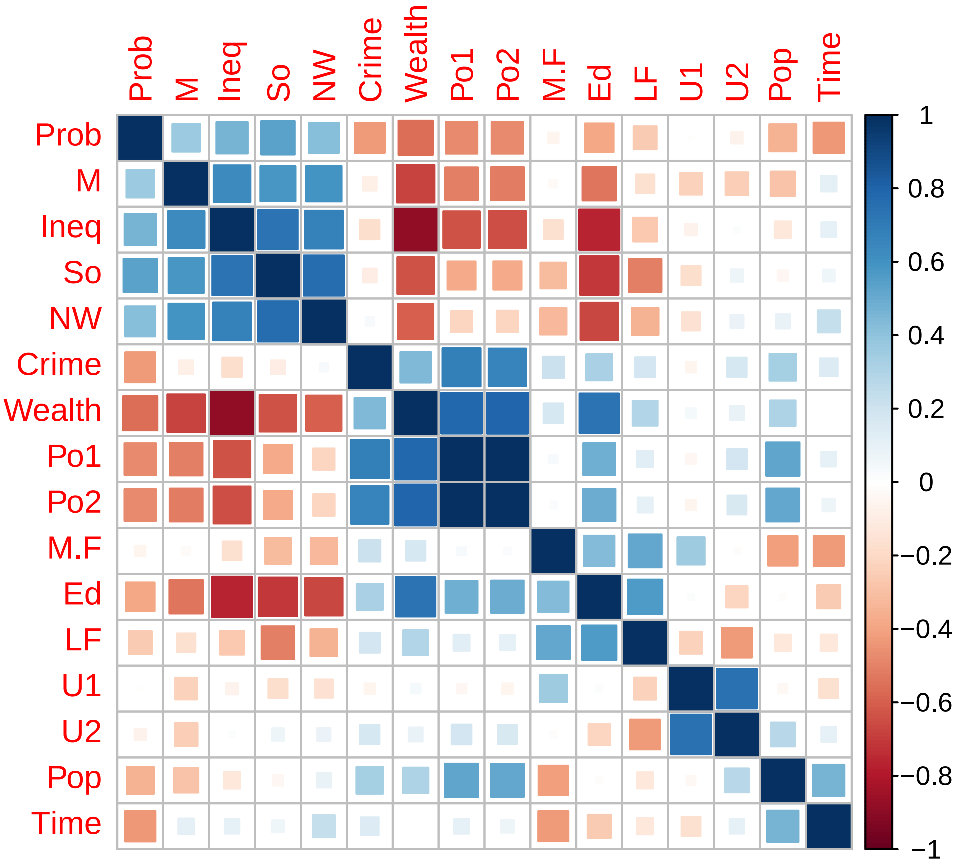
```
summary(df)
```

```
##        M               So               Ed             Po1
##  Min.   :11.90   Min.   :0.0000   Min.   : 8.70   Min.   : 4.50
##  1st Qu.:13.00   1st Qu.:0.0000   1st Qu.: 9.75   1st Qu.: 6.25
##  Median :13.60   Median :0.0000   Median :10.80   Median : 7.80
##  Mean   :13.86   Mean   :0.3404   Mean   :10.56   Mean   : 8.50
##  3rd Qu.:14.60   3rd Qu.:1.0000   3rd Qu.:11.45   3rd Qu.:10.45
##  Max.   :17.70   Max.   :1.0000   Max.   :12.20   Max.   :16.60
##       Po2               LF              M.F              Pop
##  Min.   : 4.100   Min.   :0.4800   Min.   : 93.40   Min.   :  3.00
##  1st Qu.: 5.850   1st Qu.:0.5305   1st Qu.: 96.45   1st Qu.: 10.00
##  Median : 7.300   Median :0.5600   Median : 97.70   Median : 25.00
##  Mean   : 8.023   Mean   :0.5612   Mean   : 98.30   Mean   : 36.62
##  3rd Qu.: 9.700   3rd Qu.:0.5930   3rd Qu.: 99.20   3rd Qu.: 41.50
##  Max.   :15.700   Max.   :0.6410   Max.   :107.10   Max.   :168.00
##        NW              U1                U2            Wealth
##  Min.   : 0.20   Min.   :0.07000   Min.   :2.000   Min.   :2880  
##  1st Qu.: 2.40   1st Qu.:0.08050   1st Qu.:2.750   1st Qu.:4595  
##  Median : 7.60   Median :0.09200   Median :3.400   Median :5370  
##  Mean   :10.11   Mean   :0.09547   Mean   :3.398   Mean   :5254  
##  3rd Qu.:13.25   3rd Qu.:0.10400   3rd Qu.:3.850   3rd Qu.:5915  
##  Max.   :42.30   Max.   :0.14200   Max.   :5.800   Max.   :6890  
##       Ineq            Prob              Time           Crime       
##  Min.   :12.60   Min.   :0.00690   Min.   :12.20   Min.   : 342.0  
##  1st Qu.:16.55   1st Qu.:0.03270   1st Qu.:21.60   1st Qu.: 658.5  
##  Median :17.60   Median :0.04210   Median :25.80   Median : 831.0  
##  Mean   :19.40   Mean   :0.04709   Mean   :26.60   Mean   : 905.1  
##  3rd Qu.:22.75   3rd Qu.:0.05445   3rd Qu.:30.45   3rd Qu.:1057.5  
##  Max.   :27.60   Max.   :0.11980   Max.   :44.00   Max.   :1993.0
```

Looking at correlation between the variables

We see that some of our variables are highly correlated, which could lead to an issue with collinearity Also notice that at least on a univirate basis, some variables have little relationship with crime

```
df %>% cor(use = "pairwise.complete.obs") %>% corrplot(method = "square", order = "hclust")
```



Run the regression assuming a linear fit

- We can see thar quite a few of our variables are not significant and have high p values (> .05), which may at least partially be a result of our small sample size
- We return an adjusted R-Squared of 70%
- Our significant F score indicates that the model as a whole has a relationship

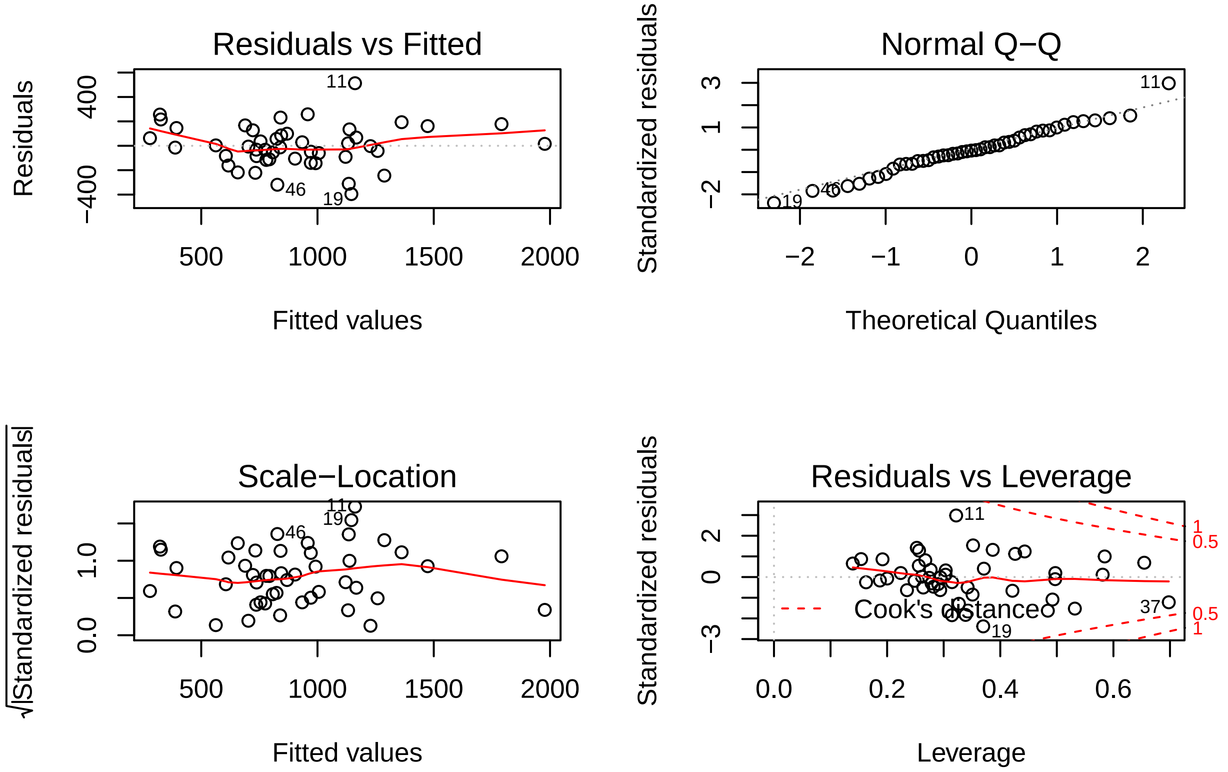
```r
fit <- lm(Crime ~.,data = df)
summary(fit)
```

```
##
## Call:
## lm(formula = Crime ~ ., data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -395.74  -98.09   -6.69  112.99  512.67
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.984e+03  1.628e+03  -3.675 0.000893 ***
## M            8.783e+01  4.171e+01   2.106 0.043443 *
## So          -3.803e+00  1.488e+02  -0.026 0.979765
## Ed           1.883e+02  6.209e+01   3.033 0.004861 **
## Po1          1.928e+02  1.061e+02   1.817 0.078892 .
## Po2         -1.094e+02  1.175e+02  -0.931 0.358830
## LF          -6.638e+02  1.470e+03  -0.452 0.654654
## M.F          1.741e+01  2.035e+01   0.855 0.398995
## Pop         -7.330e-01  1.290e+00  -0.568 0.573845
## NW           4.204e+00  6.481e+00   0.649 0.521279
## U1          -5.827e+03  4.210e+03  -1.384 0.176238
## U2           1.678e+02  8.234e+01   2.038 0.050161 .
## Wealth       9.617e-02  1.037e-01   0.928 0.360754
## Ineq         7.067e+01  2.272e+01   3.111 0.003983 **
## Prob        -4.855e+03  2.272e+03  -2.137 0.040627 *
## Time        -3.479e+00  7.165e+00  -0.486 0.630708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 209.1 on 31 degrees of freedom
## Multiple R-squared:  0.8031, Adjusted R-squared:  0.7078
## F-statistic: 8.429 on 15 and 31 DF,  p-value: 3.539e-07
```

Now let us take a closer look at the model fit, using the handy plot function with lm objects

- Residuals vs fitted: our residuals do not show any pattern with respect to our fitted, which is what we want to see. This indicates that our choice of linear may be the correct one.
- Normal Q-Q: residuals tend to follow normal distribution with some divergence at the tails, especially the lower end
- Scale - Location: No evidence of heteroskedasticity
- Residuals vs Leverage: No overwhelming outlier points

```r
# Lets us view all 4 graphs
par(mfrow = c(2,2))
plot(fit)
```



```r
# Resets graphing structure to normal
par(mfrow = c(1,1))
```

Now that we have our model fit, we need to predict upon the provided point

```r
pred_df <- df[1,]
pred_df[1,] <- NA
pred_df$M <- 14
pred_df$So <- 0
pred_df$Ed <- 10
pred_df$Po1 <- 12
pred_df$Po2 <- 15.5
pred_df$LF <- .640
pred_df$M.F <- 94
pred_df$Pop <- 150
pred_df$NW <- 1.1
pred_df$U1 <- .12
pred_df$U2 <- 3.6
pred_df$Wealth <- 3200
pred_df$Ineq <- 20.1
pred_df$Prob <- .04
pred_df$Time <- 39

pred_df <- pred_df[,1:15]

predict(object = fit, newdata =  pred_df)
```

```
##        1
## 155.4349
```

Let us remember that we likely overfit our data. We likely did not need all of the variables we used a predictors. In addition, the r- squared value is likely inflated, as we did not cross validate, or seperate out a training set.

We have obtained a prediction value of 155.4349

Summary / Conclusion:

Question 8.1

- We explained a use case for linear regression within the finance industry
- Using measures of credit quality, we can attempt to predict the yield of a company's debt

Question 8.2

- We fit the 15 predictors to the Crime response using a purely linear fit with untransformed variables
- Looking at the output coefficents, we see that many of them are not significant
- We looked at the overall model fit and confirmed issues such as non-normality and heteroskedasticity in the residuals were unlikely
- We realize that we likely overfit our data, and our errors are understated as we did not cross validate our results
    - Sticking with this process makes our results easier to understand
- Using our model fit, we predicted the new point and returned a value of **155**