SLR Assignment

Zander Bonnet

Apr 24, 2024

Multivariate Vs. Simple Regression

The difference in multivariate and simple linear regression is that simple linear regression uses one independent variable, and multivariate regression utilizes two or more independent variables. Multivariate regression also has an additional assumption that the independent variables can't be colinear, as this will violate the multicoliniarity assumption. In this example I will use weight of the vehicle to predict the mpg of it, so it is simple linear regression. If I utilized weight, horsepower, year made, and so on it would become multivariate linear regression as there are multiple factors used in the model.

Simple Regression Question

In this analysis I will create a simple regression model to answer the question of if you can use the weight of a vehicle to predict the mpg of that vehicle. If we can by how much?

```
mpg <- read.table("~/Desktop/GCU/DSC_520/Data/auto+mpg/auto-mpg.data")
colnames(mpg) <- c('mpg', 'cylinders','displacment',  'horsepower', 'weight', 'accelerat
ion', 'model.year', 'orgin', 'car.name')
mpg$horsepower <- as.integer(mpg$horsepower)
```

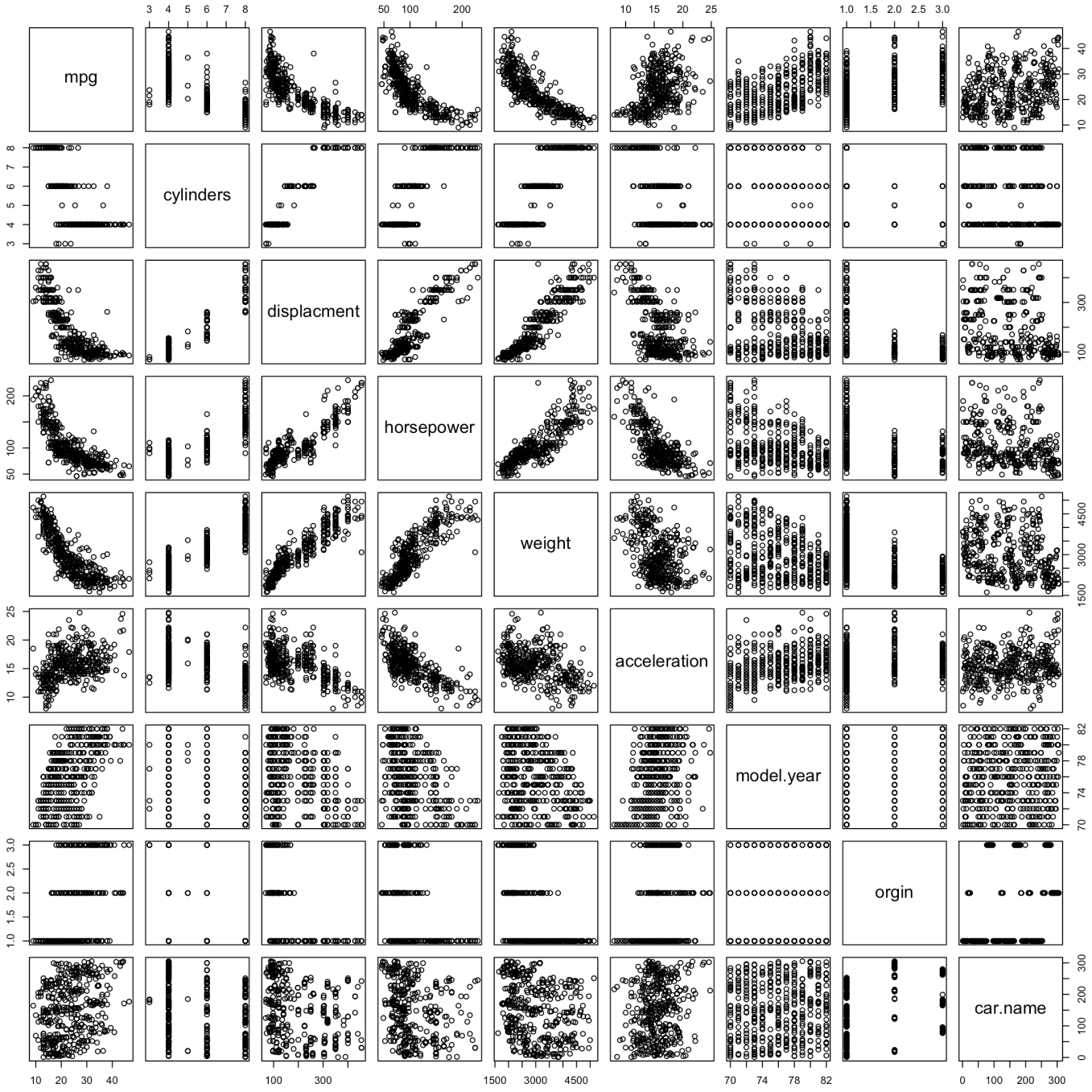
```
## Warning: NAs introduced by coercion
```

```
#Finds meaasures of central tendancy
summary(mpg)
```

```
##       mpg          cylinders      displacment      horsepower        weight
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0   Min.   :1613
##  1st Qu.:17.50   1st Qu.:4.000   1st Qu.:104.2   1st Qu.: 75.0   1st Qu.:2224
##  Median :23.00   Median :4.000   Median :148.5   Median : 93.5   Median :2804
##  Mean   :23.51   Mean   :5.455   Mean   :193.4   Mean   :104.5   Mean   :2970
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:262.0   3rd Qu.:126.0   3rd Qu.:3608
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0   Max.   :5140
##                                                  NA's   :6
##   acceleration     model.year        orgin         car.name
##  Min.   : 8.00   Min.   :70.00   Min.   :1.000   Length:398
##  1st Qu.:13.82   1st Qu.:73.00   1st Qu.:1.000   Class :character
##  Median :15.50   Median :76.00   Median :1.000   Mode  :character
##  Mean   :15.57   Mean   :76.01   Mean   :1.573
##  3rd Qu.:17.18   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :24.80   Max.   :82.00   Max.   :3.000
##
```

From looking at the measures of central tendency we can see that there is potential skewness in the displacement and horsepower factors as they have a large difference in the mean and median. We can even see that they appear to be positively skewed because the mean is larger than the median. MPG and acceleration of similar means and medians, so they appear to be the most uniformly distributed.

```
plot(mpg)
```



Variables

The independent variables are cylinders, displacement, horsepower, weight, acceleration, model year, and origin.The dependent variable is the mpg of the vehicle. The car name is just an identifier.

```
#identifies missing values and removes them
mpg[which(is.na(mpg$horsepower) == TRUE),]
```

```
##      mpg cylinders displacment horsepower weight acceleration model.year orgin
## 33  25.0         4          98         NA   2046         19.0         71     1
## 127 21.0         6         200         NA   2875         17.0         74     1
## 331 40.9         4          85         NA   1835         17.3         80     2
## 337 23.6         4         140         NA   2905         14.3         80     1
## 355 34.5         4         100         NA   2320         15.8         81     2
## 375 23.0         4         151         NA   3035         20.5         82     1
##                 car.name
## 33            ford pinto
## 127        ford maverick
## 331 renault lecar deluxe
## 337   ford mustang cobra
## 355          renault 18i
## 375       amc concord dl
```

```
mpg <- na.omit(mpg)
```

```
#finds any outliers
low <- quantile(mpg$mpg, .25)
up <- quantile(mpg$mpg, .75)
iqr <- IQR(mpg$mpg)

lowthresh <- low - (1.5 * iqr)
upthresh <- up + (1.5 * iqr)

which(mpg$mpg < lowthresh | mpg$mpg > upthresh)
```

```
## integer(0)
```

```
#identifies if any missing values remain
sum(is.na(iris))
```

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

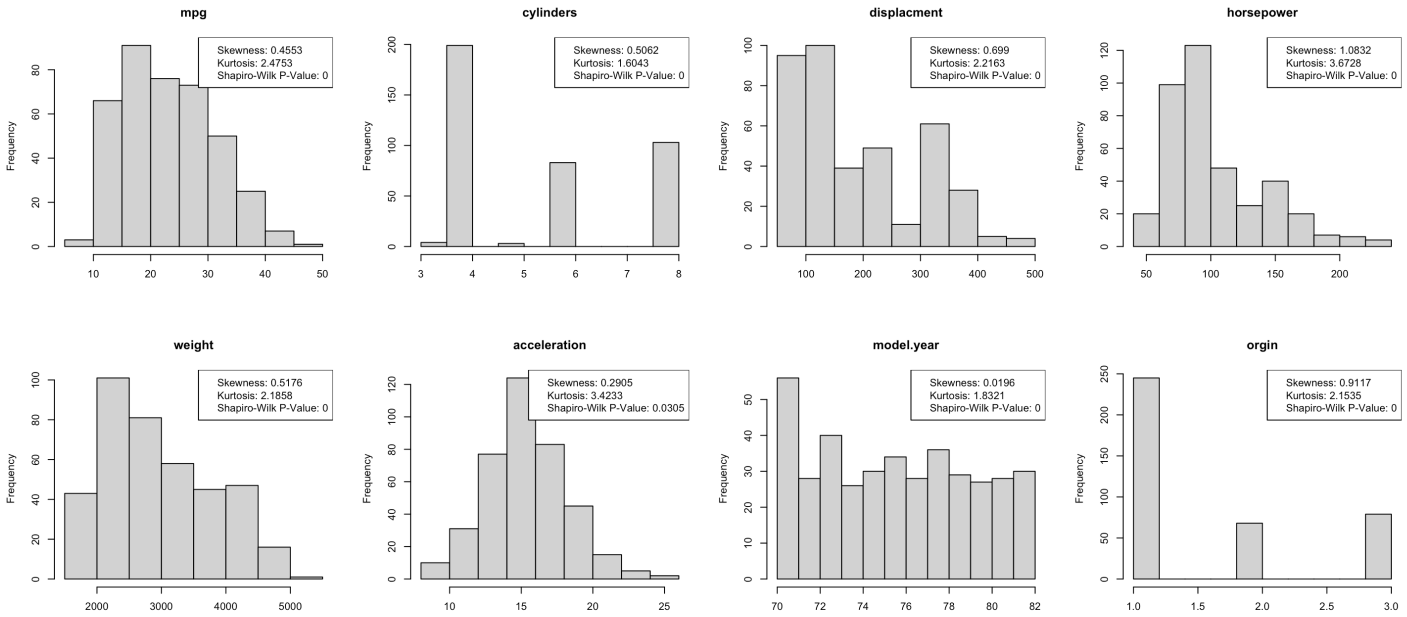
There are no outliers in the dependent variable, and the missing values were all in the horsepower factor. I chose to remove the data with missing values because there were only a handful.

```
#gets measure of spread of the data
var(mpg[-9])
```

```
##                      mpg    cylinders displacment  horsepower      weight
## mpg             60.918142  -10.3529281  -657.58521  -233.85793  -5517.4407
## cylinders      -10.352928    2.9096965   169.72195    55.34824   1300.4244
## displacment   -657.585207  169.7219486 10950.36755  3614.03374  82929.1001
## horsepower    -233.857926   55.3482436  3614.03374  1481.56939  28265.6202
## weight       -5517.440704 1300.4243632 82929.10014 28265.62023 721484.7090
## acceleration     9.115514   -2.3750522  -156.99444   -73.18697   -976.8153
## model.year      16.691477   -2.1719296  -142.57213   -59.03643   -967.2285
## orgin            3.553510   -0.7817344   -51.80079   -14.11274   -400.2660
##              acceleration   model.year        orgin
## mpg             9.1155144   16.6914766    3.5535101
## cylinders      -2.3750522   -2.1719296   -0.7817344
## displacment  -156.9944354 -142.5721332  -51.8007921
## horsepower    -73.1869670  -59.0364320  -14.1127407
## weight       -976.8152526 -967.2284566 -400.2660499
## acceleration    7.6113312    2.9504619    0.4727882
## model.year      2.9504619   13.5699149    0.5386502
## orgin           0.4727882    0.5386502    0.6488595
```

If you look at the diagnal of the matrix you can see the variances for each variable. The other values represent the covariances for the corresponding variables. Displacement appears to have the largest spread of data as it has a very large variance comparative to its mean. Weight also appears to have a lot of variation in the data. Acceleration appears to have the smallest relative variance.

```
library('moments')
par(mfrow = c(2,4))
for(i in names(mpg[1:9])){
  if(is.numeric(mpg[1,i])) {
    hist(mpg[,i], xlab = NULL, main = i)
    legend('topright', legend = c(paste('Skewness:',round(skewness(mpg[,i]),4))
                              ,paste('Kurtosis:',round(kurtosis(mpg[,i]),4)),
                              paste('Shapiro-Wilk P-Value:', round(shapiro.test(mpg[,i])
$p.value, 4))))
  }
}
```



All of the factors have a slight to slightly significant positive skew. The horsepower factor is the most skewed, and model year has the least skew. All of the data also has a positive kurtosis value, so the data is 'too pointy' compared to a true normal distribution. We can also see if the data fits the normal distribution by looking at the p-value of the Shapiro-Wilks test. We can see that none of the the data appears to follow a normal distribution since all the p-values are less then .05.

In order to create the model we need to make sure that we are not violating the assumptions of linear regression.

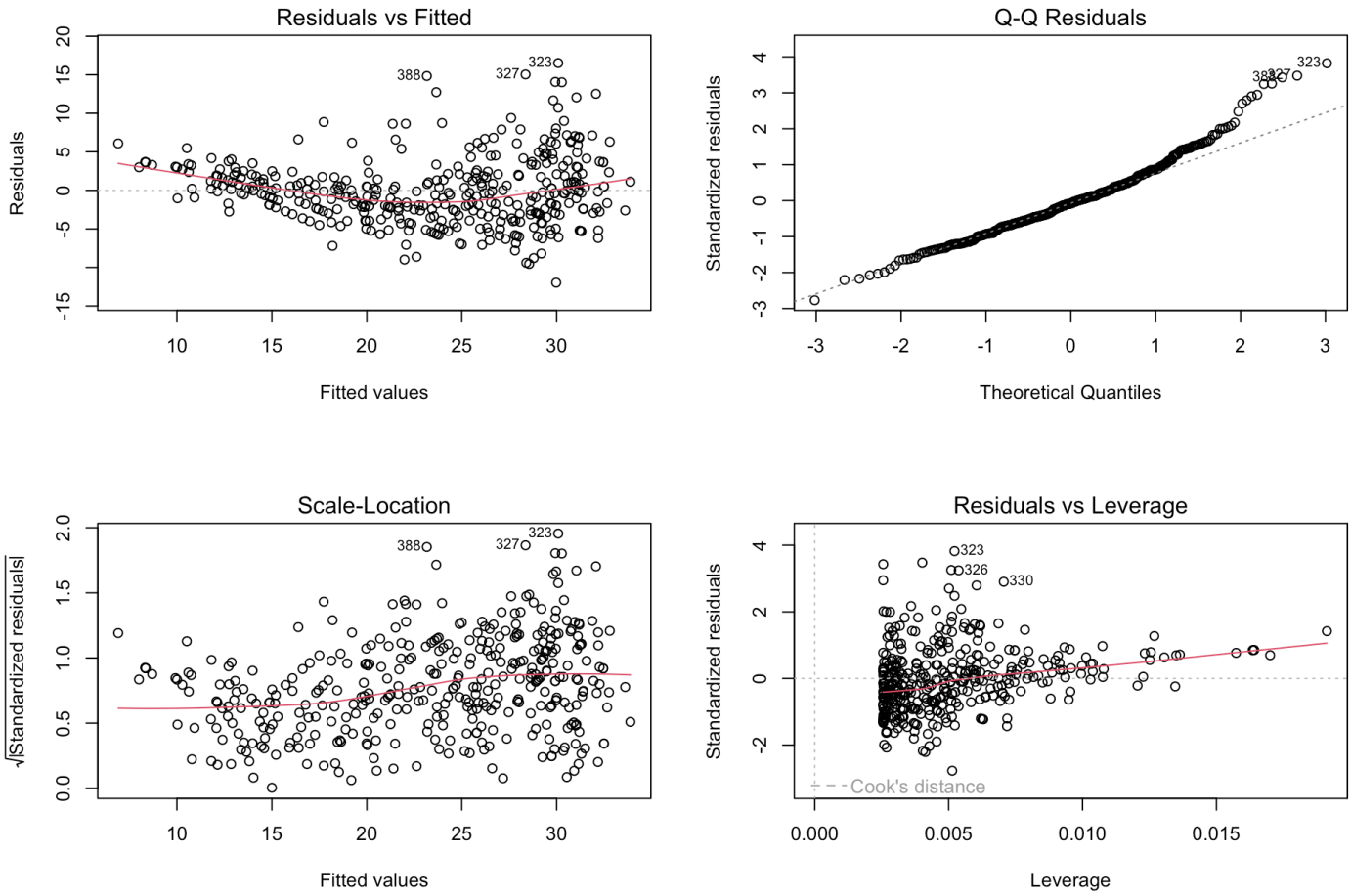
The weight and the MPG are independent data points and they do have a linear relationship. We will check for normality and equal variance of the residuals.

```
mod <- lm(mpg ~ weight,data = mpg)
summary(mod)
```

```
##
## Call:
## lm(formula = mpg ~ weight, data = mpg)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.9736  -2.7556  -0.3358   2.1379  16.5194
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 46.216524   0.798673   57.87   <2e-16 ***
## weight      -0.007647   0.000258  -29.64   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.333 on 390 degrees of freedom
## Multiple R-squared:  0.6926, Adjusted R-squared:  0.6918
## F-statistic: 878.8 on 1 and 390 DF,  p-value: < 2.2e-16
```

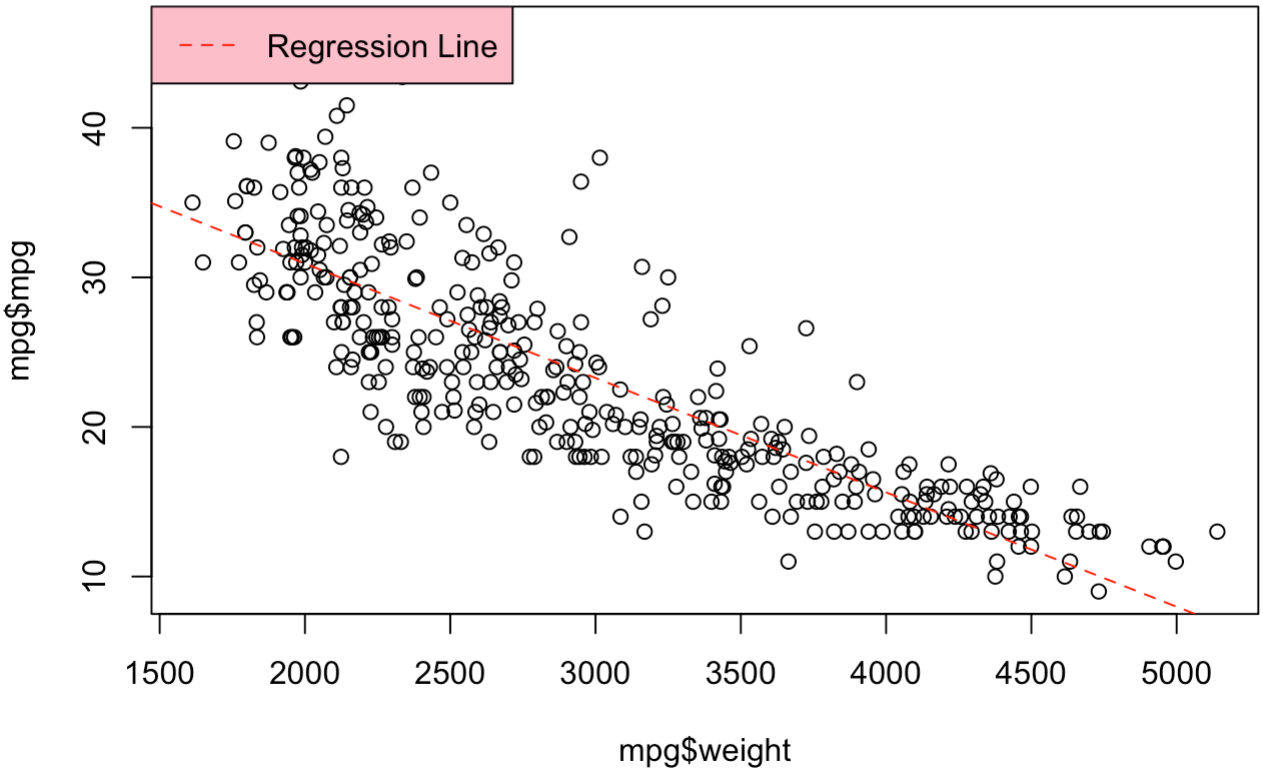
The model is found to be a significant with a F-stat of 878, and both the intercept and the independent variable (weight) have a very small p-value. The model says that if the vehicle weighed 0 zero units it would have an MPG of 46.2. Then for every 1 unit of weight the car will lose .007 MPG.

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



Looking at the residuals we can see that the residuals might violate the homoscedasticity assumption at the higher values. To combat this we can try and transform the data so that we can have normally distributed residuals.

```
plot(mpg$weight, mpg$mpg)
abline(mod, col='red', lty = 2)
legend('topleft', legend = 'Regression Line', lty = 2, col = 'red', bg = 'pink')
```

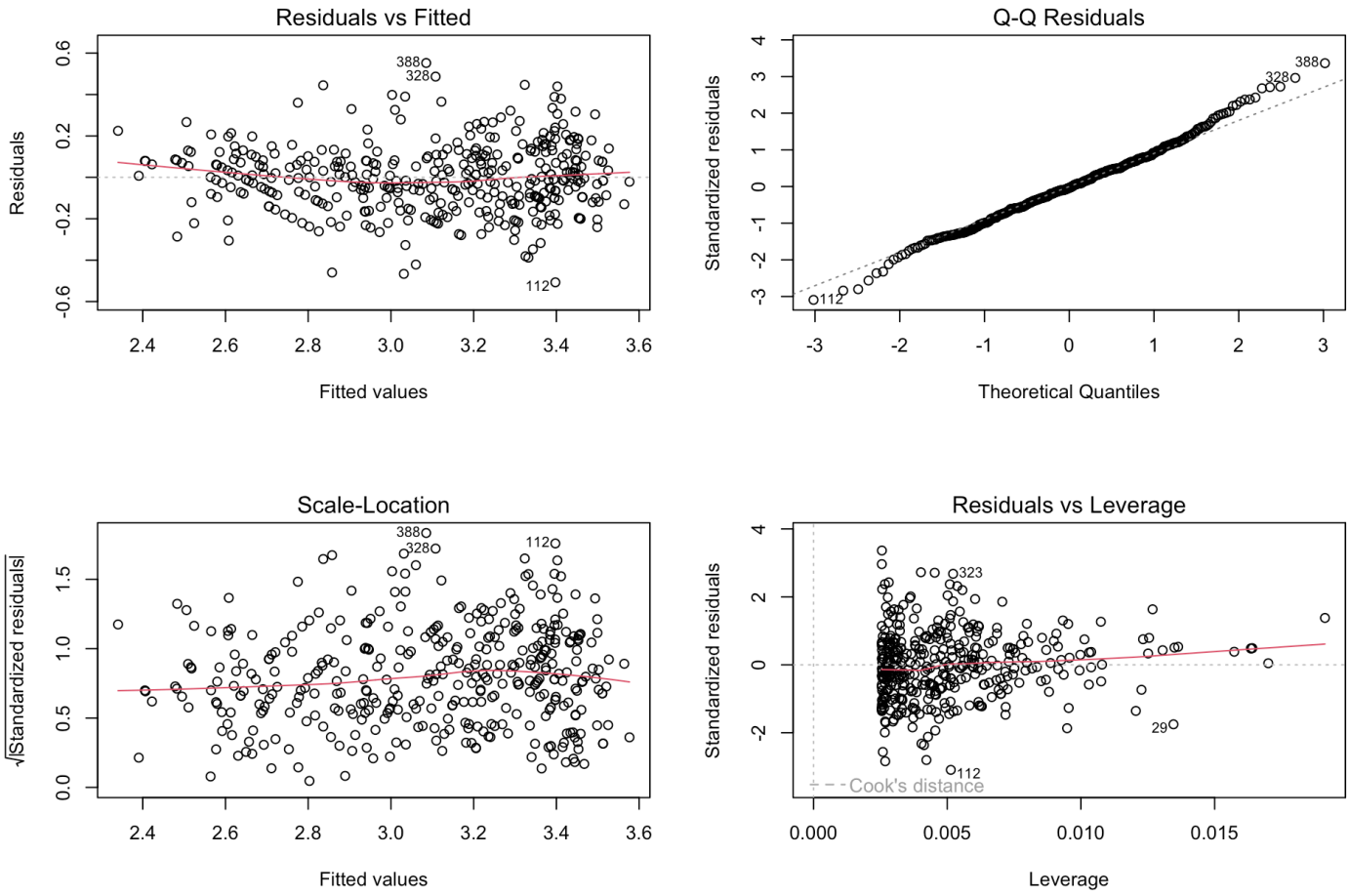



```
mod2 <- lm(log(mpg) ~ weight,data = mpg)
summary(mod2)
```

```
##
## Call:
## lm(formula = log(mpg) ~ weight, data = mpg)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50716 -0.09966 -0.00621  0.09973  0.55239
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.142e+00  3.031e-02  136.66   <2e-16 ***
## weight      -3.505e-04  9.790e-06  -35.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1644 on 390 degrees of freedom
## Multiple R-squared:  0.7668, Adjusted R-squared:  0.7662
## F-statistic:  1282 on 1 and 390 DF,  p-value: < 2.2e-16
```

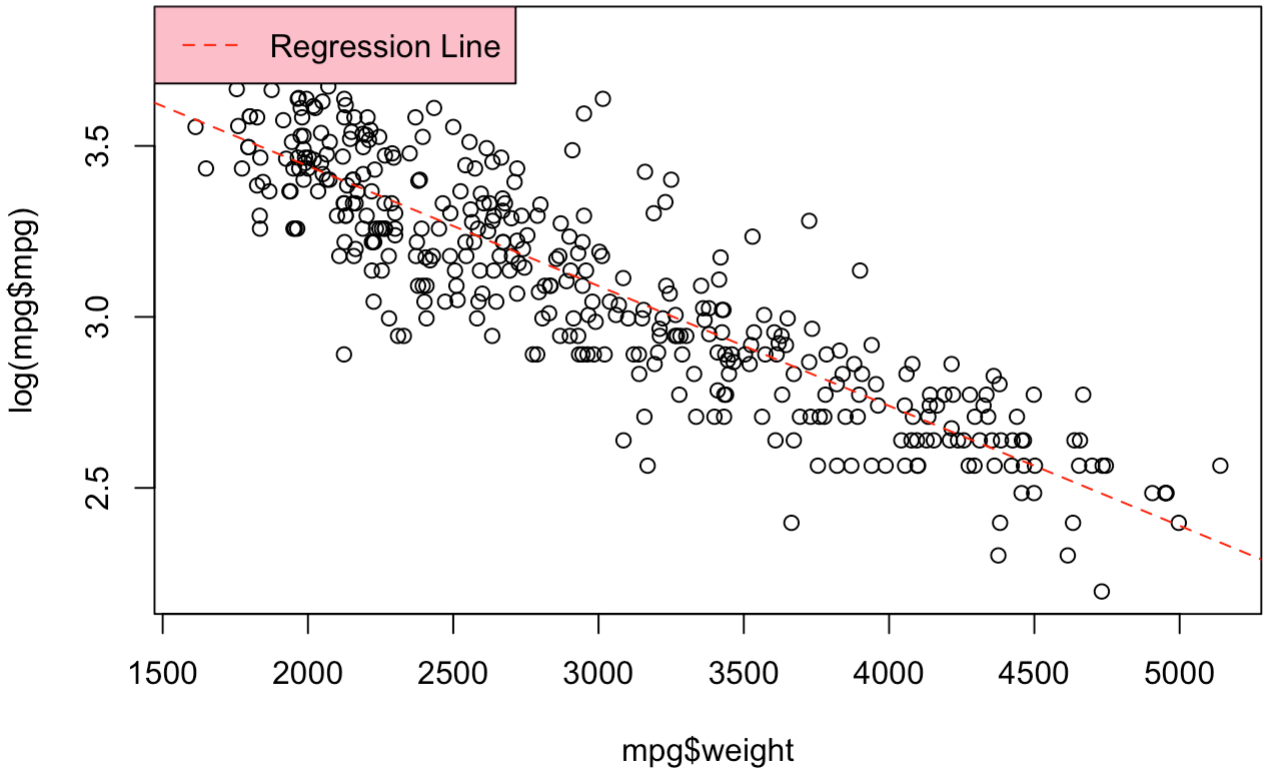
By conducting a log transformation on the dependent variable (weight) we able to improve the significance of the model, and increase the R-Squared so we know that it is better at explaining the variation of the data. In this model for every one unit of weight there is a .0003 decrease in log mpg, and if the weight was zero then the vehicle would have a log mpg of 4.14.

```
par(mfrow = c(2,2))
plot(mod2)
```



The residual plots show that the assumption of homoscedasticity is satisfied as the residuals are basically normal.

```
plot(mpg$weight, log(mpg$mpg))
abline(mod2, col='red', lty = 2)
legend('topleft', legend = 'Regression Line', lty = 2, col = 'red', bg = 'pink')
```



By plotting the regression line over the true data shows how well the model can follow the trend of the data.

Reference

Quinlan,R.. (1993). Auto MPG. UCI Machine Learning Repository. https://doi.org/10.24432/C5859H. (https://doi.org/10.24432/C5859H.)