DA 605 - Assignment 11

Dan Fanelli

LINEAR REGRESSION IN R

- Your submission should include the final linear fits, and their corresponding significance levels.
- In addition, you should clearly state what you concluded from looking at the fit and their significance levels.

```r
library(knitr)
library(ggplot2)
library(rpart)
```

Age and Max HR:

We hear that MaxHR = 220 - Age

```r
age <- c(18,23,25,35,65,54,34,56,72,19,23,42,18,39,37)
maxHR <- c(202,186,187,180,156,169,174,172,153,199,193,174,198,183,178)
hr_age_df <- data.frame(age, maxHR)
summary(hr_age_df)
```

```
##       age            maxHR      
##  Min.   :18.00   Min.   :153.0  
##  1st Qu.:23.00   1st Qu.:173.0  
##  Median :35.00   Median :180.0  
##  Mean   :37.33   Mean   :180.3  
##  3rd Qu.:48.00   3rd Qu.:190.0  
##  Max.   :72.00   Max.   :202.0  
```

```r
kable(hr_age_df, align = "l")
```

age	maxHR
18	202
23	186
25	187
35	180
65	156
54	169
34	174
56	172
72	153
19	199
23	193

age	maxHR
42	174
18	198
39	183
37	178

Using R's lm function:

1. Perform regression analysis

```
hr_age_model <- lm(maxHR ~ age)
summary(hr_age_model)
```

```
##
## Call:
## lm(formula = maxHR ~ age)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9258 -2.5383  0.3879  3.1867  6.6242
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 210.04846    2.86694   73.27  < 2e-16 ***
## age          -0.79773    0.06996  -11.40 3.85e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.578 on 13 degrees of freedom
## Multiple R-squared:  0.9091, Adjusted R-squared:  0.9021
## F-statistic:   130 on 1 and 13 DF,  p-value: 3.848e-08
```

2. Measure the signicance of the independent variables

```
cor(age, maxHR)
```

```
## [1] -0.9534656
```

The independent variable 'age' has a *** next to it, so its significance level is basically 0.

3. What is the resulting equation?

```
HR = 210.04846 + (-0.79773)*age
```

4. Is the effect of Age on Max HR significant?

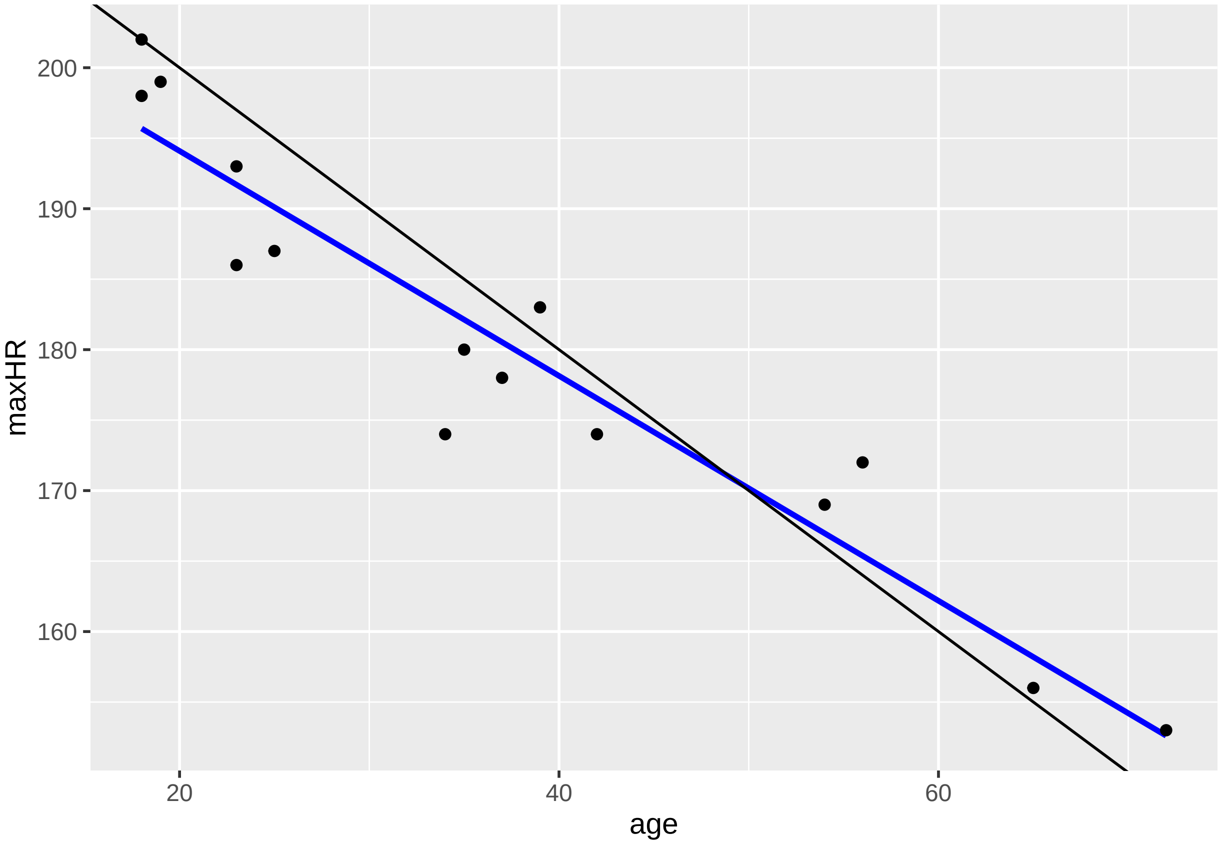
Its basically 0, so not significant.

5. What is the signicance level?

The significant level is *** which is basically zero.

6. Please also plot the fitted relationship between Max HR and Age.

```
ggplot(data = hr_age_df, aes(x = age, y = maxHR)) + geom_smooth(method = "lm", se=FALSE, color="blue",
```



Auto Data:

Using the Auto data set from Assignment 5 (also attached here) perform a Linear Regression analysis using mpg as the dependent variable and the other 4 (displacement, horsepower, weight, acceleration) as independent variables.

```
mpg_df <- read.table("auto-mpg.data", sep="")
names(mpg_df) <- c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "model_ye

mpg_df <- mpg_df[,c("mpg","displacement","horsepower","weight","acceleration")]
mpg_df <- subset(mpg_df, displacement != '?' && horsepower != '?' && weight != '?' && acceleration != '

# it was showing horsepower as categorical, so giving strange results
```

```r
mpg_df <- transform(mpg_df, horsepower = as.numeric(horsepower))

kable(head(mpg_df, n=10), align = "l")
```

mpg	displacement	horsepower	weight	acceleration
18	307	17	3504	12.0
15	350	35	3693	11.5
18	318	29	3436	11.0
16	304	29	3433	12.0
17	302	24	3449	10.5
15	429	42	4341	10.0
14	454	47	4354	9.0
14	440	46	4312	8.5
14	455	48	4425	10.0
15	390	40	3850	8.5

1. What is the final linear regression fit equation?

```r
mpg_model <- lm(mpg ~ mpg_df$displacement + mpg_df$horsepower + mpg_df$weight + mpg_df$acceleration, da

summary(mpg_model)
```

```
##
## Call:
## lm(formula = mpg ~ mpg_df$displacement + mpg_df$horsepower +
##     mpg_df$weight + mpg_df$acceleration, data = mpg_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.8331  -2.8735  -0.3164   2.4449  16.2079
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         40.8838025  1.9966258  20.476  < 2e-16 ***
## mpg_df$displacement -0.0106291  0.0065254  -1.629   0.1041
## mpg_df$horsepower    0.0047774  0.0082597   0.578   0.5633
## mpg_df$weight       -0.0061405  0.0007449  -8.243 2.54e-15 ***
## mpg_df$acceleration  0.1722165  0.0976340   1.764   0.0785 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.298 on 393 degrees of freedom
## Multiple R-squared:  0.7006, Adjusted R-squared:  0.6976
## F-statistic:   230 on 4 and 393 DF,  p-value: < 2.2e-16
```

So the final linear regression fit equation is:

mpg = 40 - (displacement * -0.0106291) + (horsepower * 0.0047774) + (weight * -0.0061405) + (acceleration * 0.1722165)

2. Which of the 4 independent variables have a significant impact on mpg?

Based on the significance codes, only weight seems to have zero impact on mpg.

3. What are their corresponding significance levels?

- displacement: 0.1041
- horsepower: 0.5633
- weight: (zero…)
- acceleration: 0.0785

4. What are the standard errors on each of the coeficients?

- displacement: 1.9966258
- horsepower: 0.0065254
- weight: 0.0007449
- acceleration: 0.0976340

Please perform this experiment in two ways.

1. First take any random 40 data points from the entire auto data sample and perform the linear regression fit and measure the 95% confidence intervals.

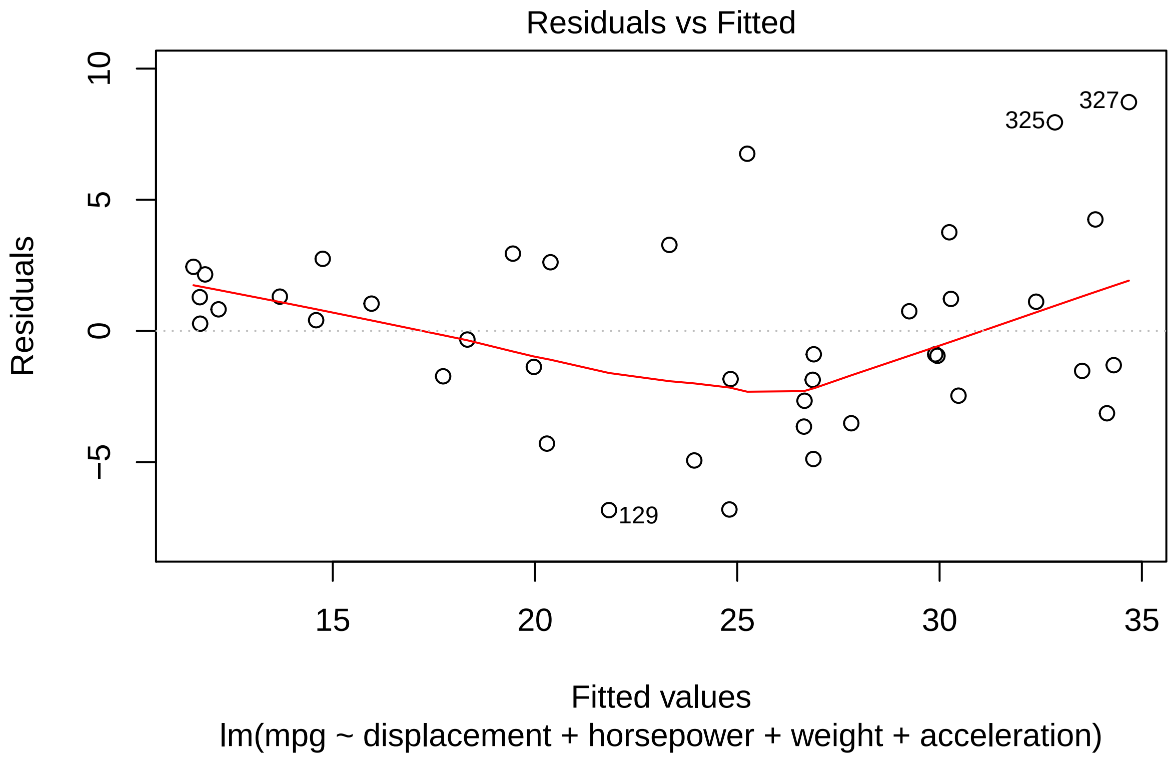
```
mpg_df_40_sample <- mpg_df[sample(nrow(mpg_df), 40), ]
mpg_df_40_sample_fit <- lm(mpg ~ displacement + horsepower + weight + acceleration, data=mpg_df_40_sampl
summary(mpg_df_40_sample_fit)
```

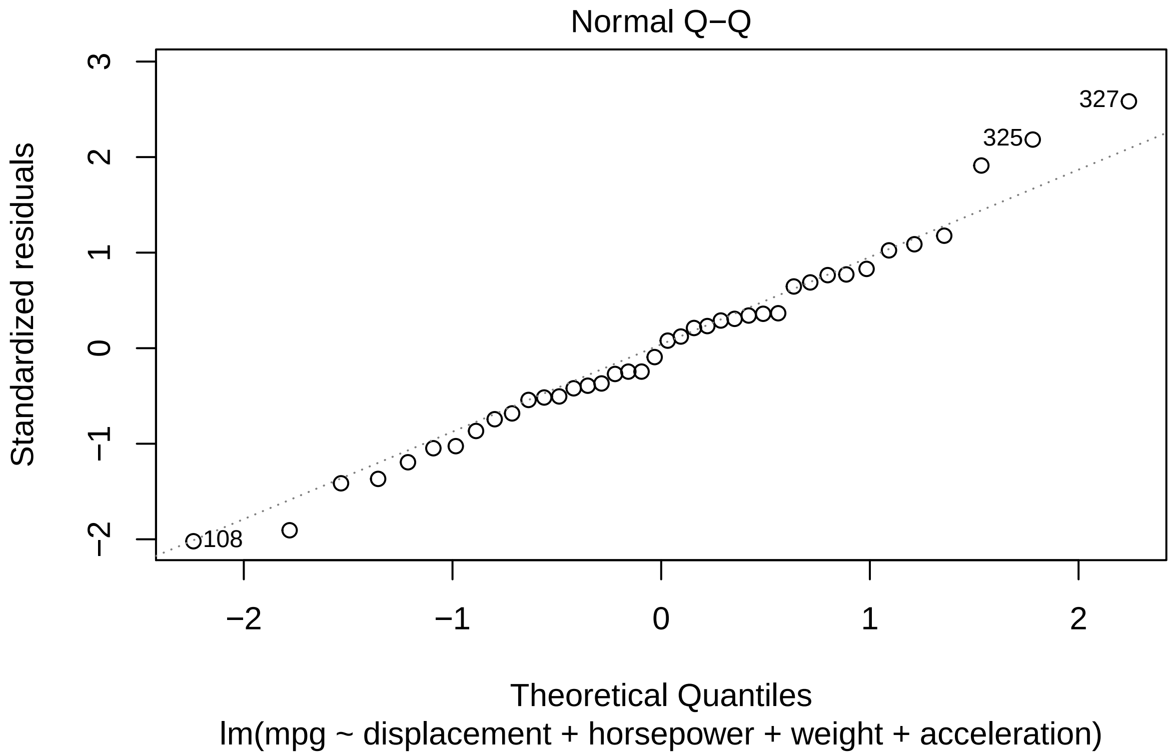
```
##
## Call:
## lm(formula = mpg ~ displacement + horsepower + weight + acceleration,
##     data = mpg_df_40_sample)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.8277 -2.0141 -0.0255  2.2268  8.7200
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30.55892    6.83873   4.469  7.9e-05 ***
## displacement  0.03381    0.02297   1.472 0.149994
## horsepower    0.03583    0.02679   1.337 0.189786
## weight       -0.01006    0.00257  -3.916 0.000398 ***
## acceleration  0.95999    0.29367   3.269 0.002425 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.767 on 35 degrees of freedom
## Multiple R-squared:  0.8132, Adjusted R-squared:  0.7919
## F-statistic:  38.1 on 4 and 35 DF,  p-value: 2.699e-12
```

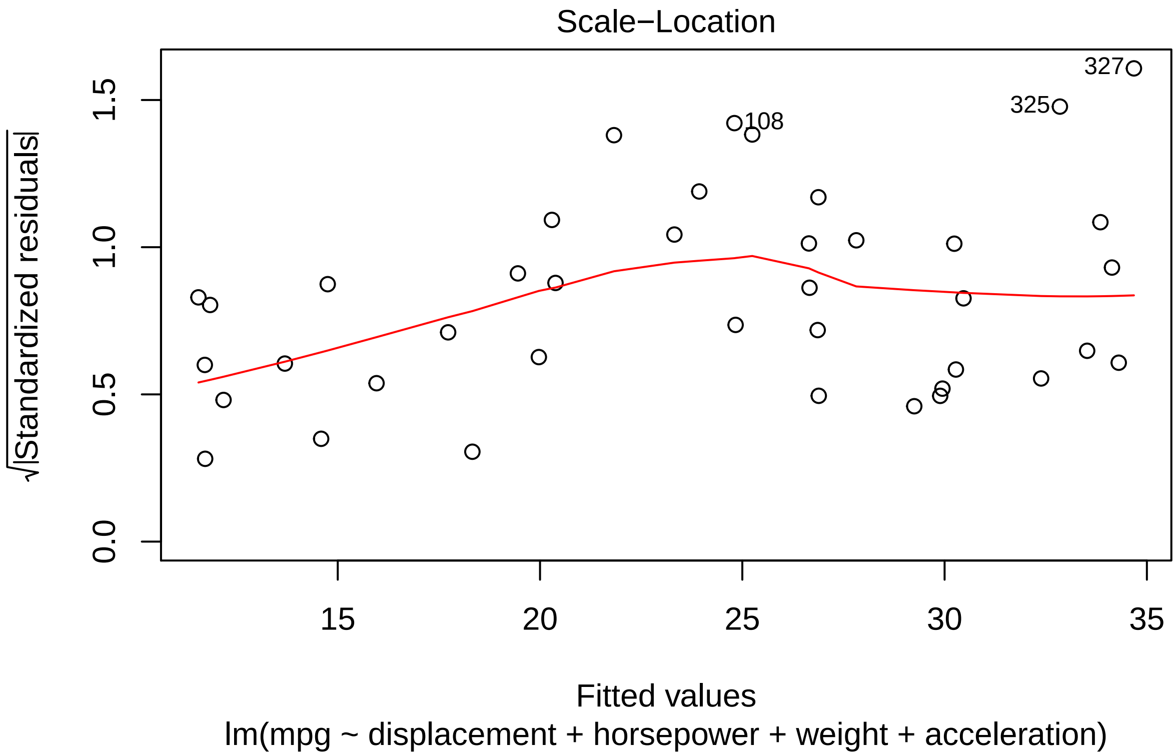
```
confint(mpg_df_40_sample_fit, level=0.95)
```

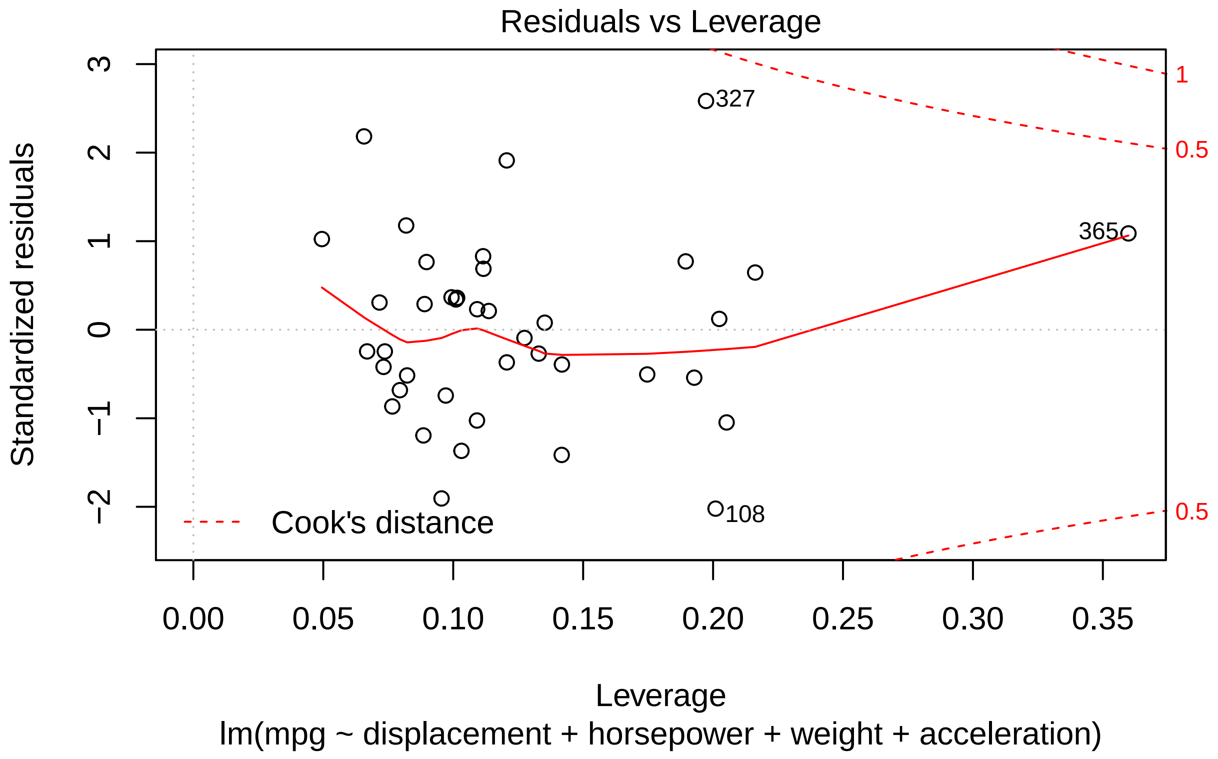
```
##                    2.5 %       97.5 %
## (Intercept)  16.67554995 44.442287432
## displacement -0.01282461  0.080450512
## horsepower   -0.01856633  0.090225147
## weight       -0.01528242 -0.004847084
## acceleration  0.36380861  1.556174382
```

```
plot(mpg_df_40_sample_fit)
```




Normal Q−Q
Standardized residuals
3
2
1
0
−1
−2
327
325
108
−2
−1
0
1
2
Theoretical Quantiles
lm(mpg ~ displacement + horsepower + weight + acceleration)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
15
20
25
30
35
Fitted values
lm(mpg ~ displacement + horsepower + weight + acceleration)
108
325
327



```r
cor(mpg_df_40_sample$displacement, mpg_df_40_sample$mpg)
```

```
## [1] -0.8437184
```

```r
cor(mpg_df_40_sample$horsepower, mpg_df_40_sample$mpg)
```

```
## [1] 0.482075
```

```r
cor(mpg_df_40_sample$weight, mpg_df_40_sample$mpg)
```

```
## [1] -0.8676221
```

```r
cor(mpg_df_40_sample$acceleration, mpg_df_40_sample$mpg)
```

```
## [1] 0.5922994
```

2. Then, take the entire data set (all 392 points) and perform linear regression and measure the 95% confidence intervals.

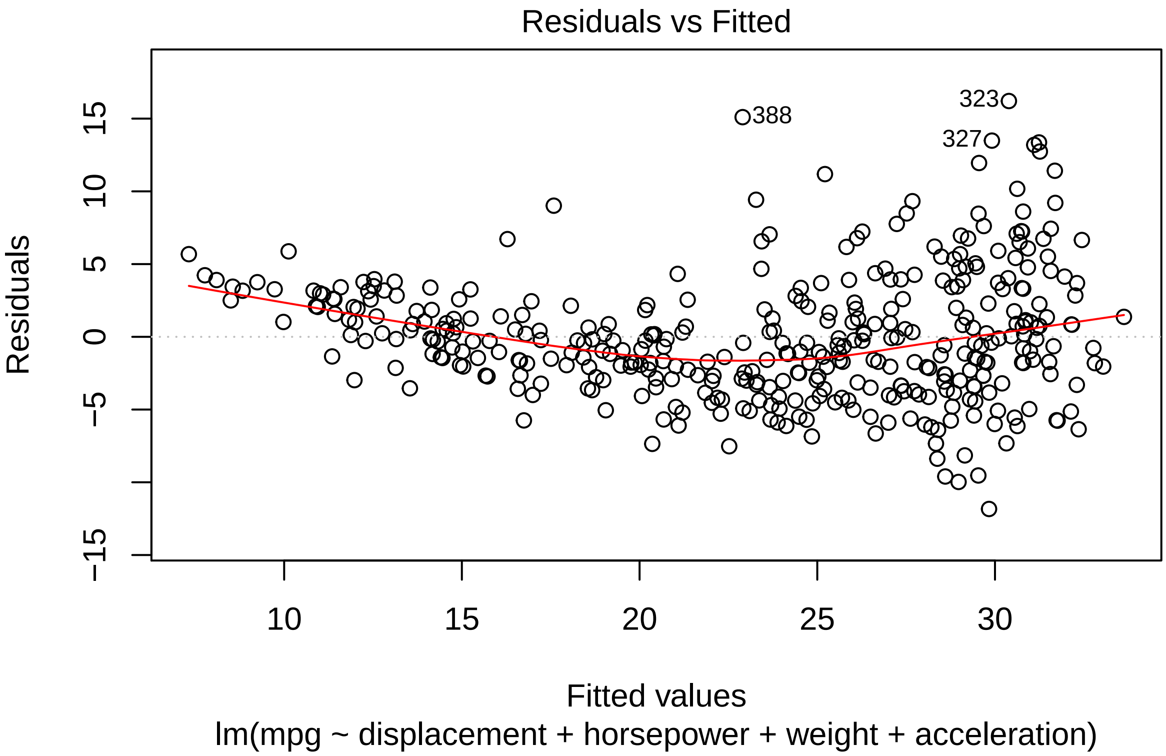
```r
mpg_df_all <- mpg_df
mpg_df_all_fit <- lm(mpg ~ displacement + horsepower + weight + acceleration, data=mpg_df_all)
summary(mpg_df_all_fit)
```

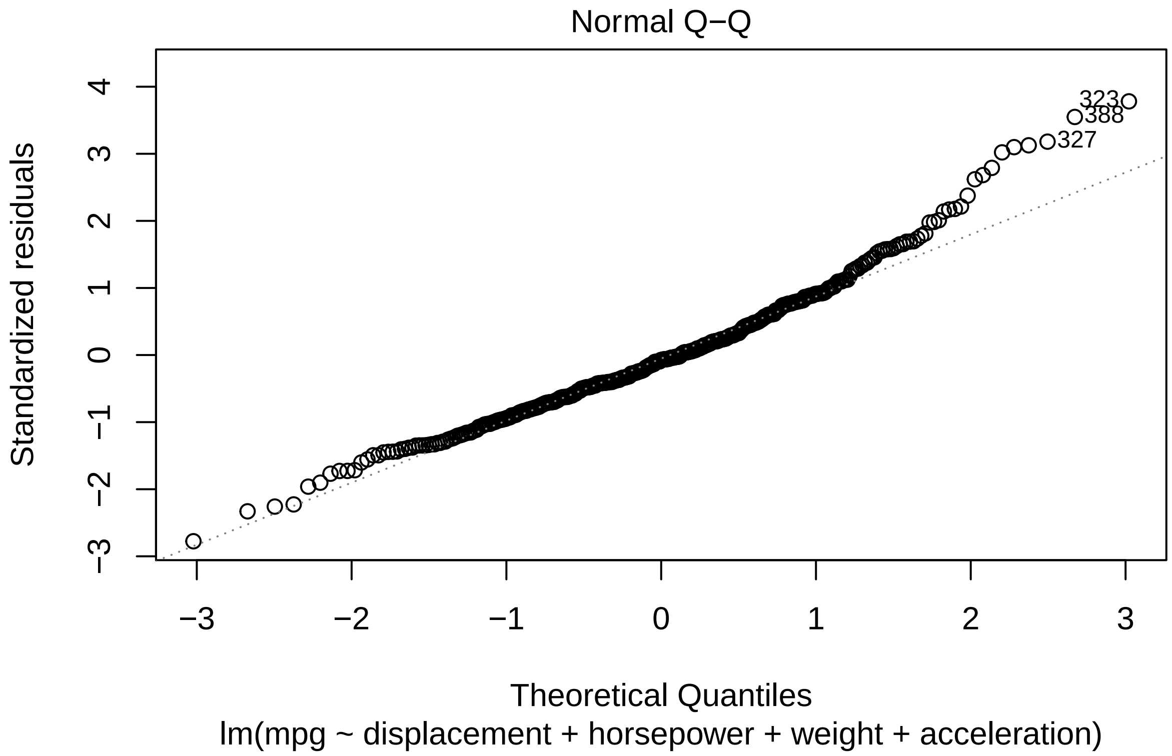
```
##
## Call:
## lm(formula = mpg ~ displacement + horsepower + weight + acceleration,
##     data = mpg_df_all)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.8331  -2.8735  -0.3164   2.4449  16.2079
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  40.8838025  1.9966258  20.476  < 2e-16 ***
## displacement -0.0106291  0.0065254  -1.629   0.1041
## horsepower    0.0047774  0.0082597   0.578   0.5633
## weight       -0.0061405  0.0007449  -8.243 2.54e-15 ***
## acceleration  0.1722165  0.0976340   1.764   0.0785 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.298 on 393 degrees of freedom
## Multiple R-squared:  0.7006, Adjusted R-squared:  0.6976
## F-statistic:   230 on 4 and 393 DF,  p-value: < 2.2e-16
```

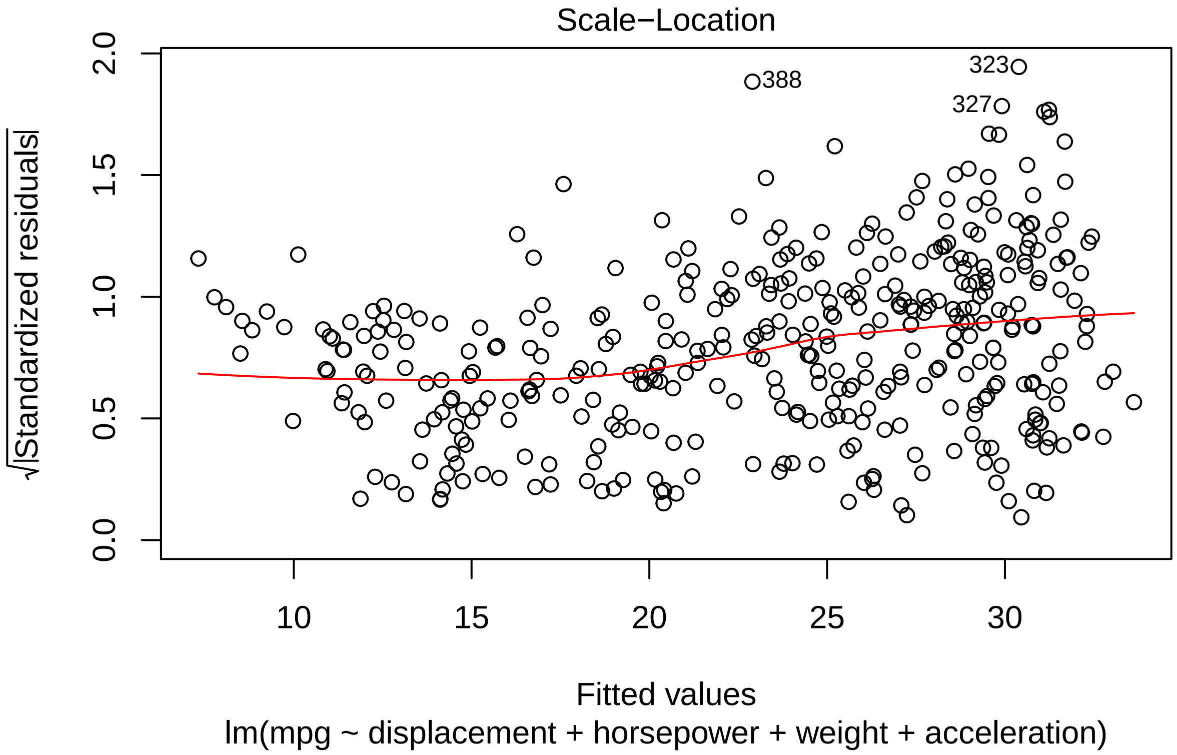
```r
confint(mpg_df_all_fit, level=0.95)
```

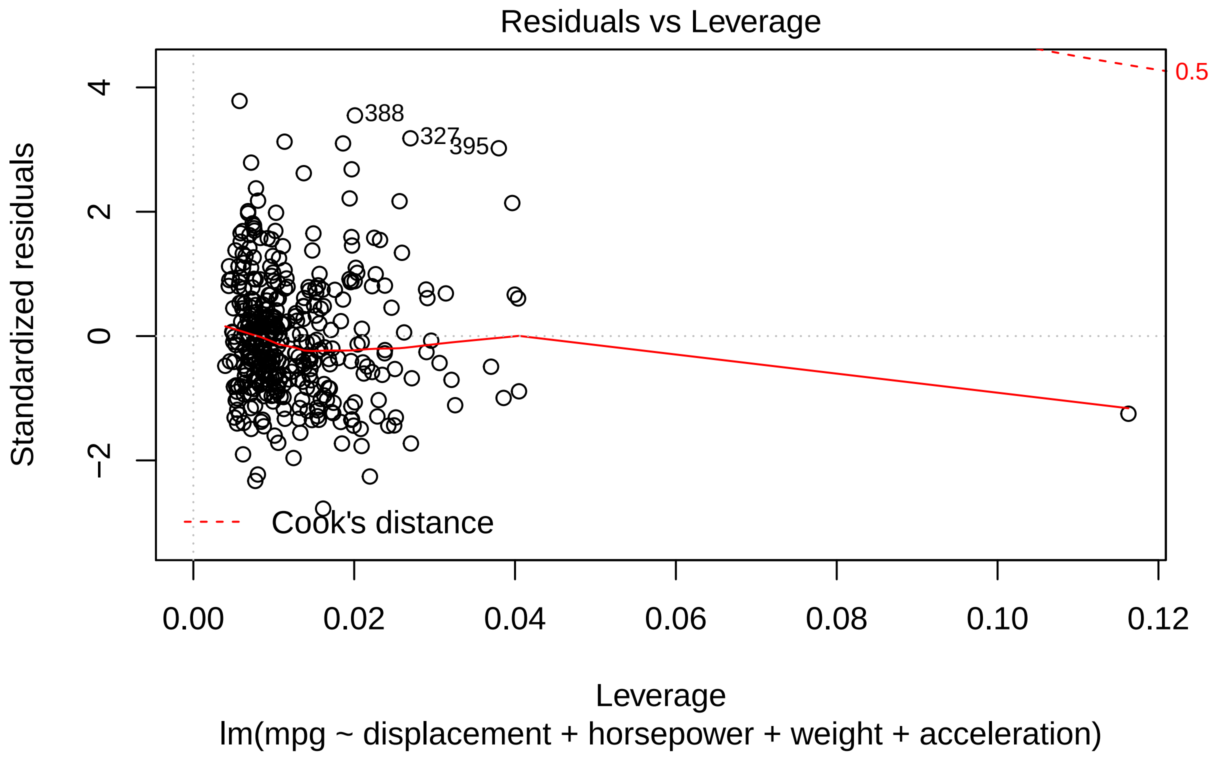
```
##                     2.5 %       97.5 %
## (Intercept)  36.958399051 44.809205992
## displacement -0.023458057  0.002199922
## horsepower   -0.011461281  0.021016071
## weight       -0.007605082 -0.004675912
## acceleration -0.019733841  0.364166819
```

```r
plot(mpg_df_all_fit)
```


Residuals vs Fitted
Residuals
Fitted values
lm(mpg ~ displacement + horsepower + weight + acceleration)
−15
−5
0
5
10
15
10
15
20
25
30
388
323
327


Normal Q−Q
Standardized residuals
4
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3
323
388
327
Theoretical Quantiles
lm(mpg ~ displacement + horsepower + weight + acceleration)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
388
323
327
10
15
20
25
30
Fitted values
lm(mpg ~ displacement + horsepower + weight + acceleration)



```
cor(mpg_df_all$displacement, mpg_df_all$mpg)
```

```
## [1] -0.8042028
```

```
cor(mpg_df_all$horsepower, mpg_df_all$mpg)
```

```
## [1] 0.4215846
```

```
cor(mpg_df_all$weight, mpg_df_all$mpg)
```

```
## [1] -0.8317409
```

```
cor(mpg_df_all$acceleration, mpg_df_all$mpg)
```

```
## [1] 0.4202889
```

3. Please report the resulting fit equation, their significance values and confidence intervals for each of the two runs.

(done, see above…)

Some Help Analysis / Debugging:

from: http://www.statmethods.net/stats/regression.html

```
# Other useful functions
coefficients(mpg_df_all_fit) # model coefficients
```

```
##  (Intercept) displacement   horsepower       weight acceleration
## 40.883802522 -0.010629067  0.004777395 -0.006140497  0.172216489
```

```
confint(mpg_df_all_fit, level=0.95) # CIs for model parameters
```

```
##                     2.5 %       97.5 %
## (Intercept)  36.958399051 44.809205992
## displacement -0.023458057  0.002199922
## horsepower   -0.011461281  0.021016071
## weight       -0.007605082 -0.004675912
## acceleration -0.019733841  0.364166819
```

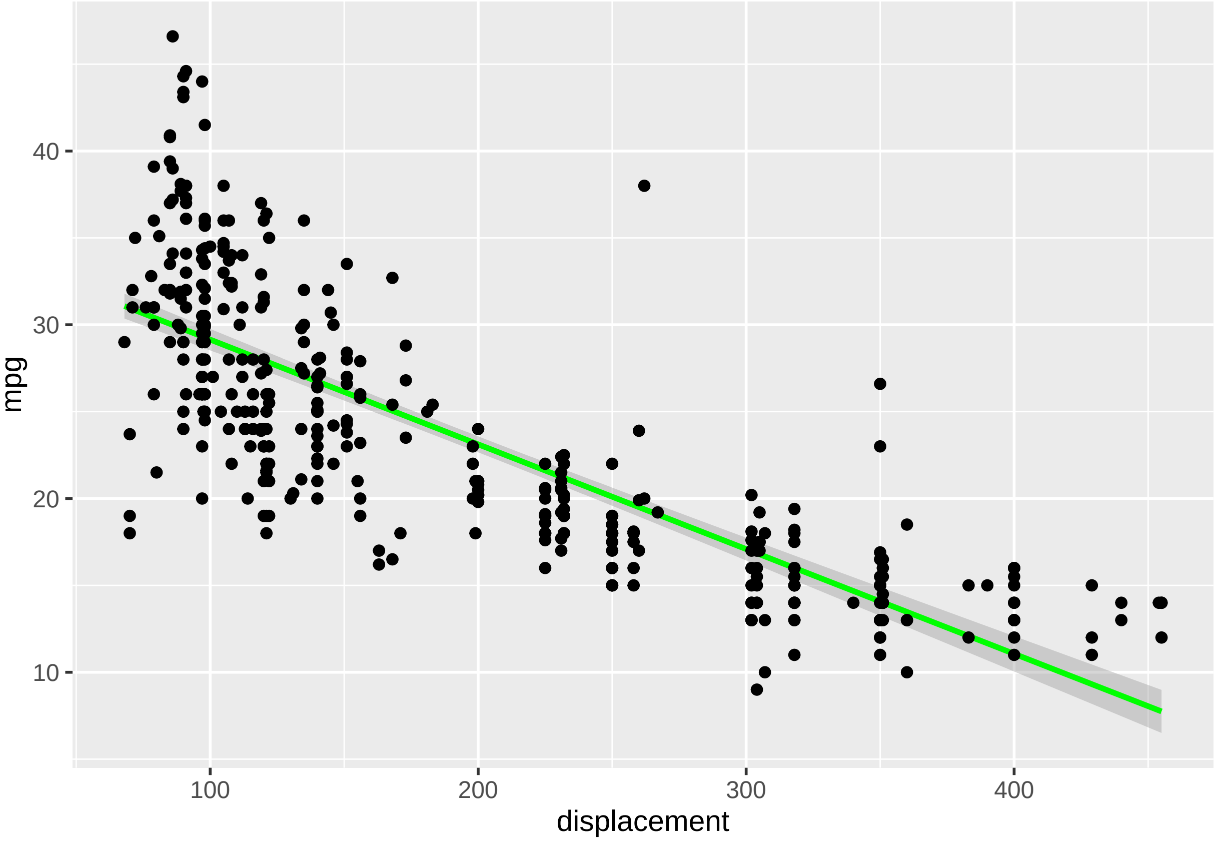
```
#fitted(mpg_df_all_fit) # predicted values
#residuals(mpg_df_all_fit) # residuals
anova(mpg_df_all_fit) # anova table
```

```
## Analysis of Variance Table
##
## Response: mpg
##               Df  Sum Sq Mean Sq  F value    Pr(>F)
## displacement   1 15685.2 15685.2 849.0675 < 2.2e-16 ***
## horsepower     1    42.5    42.5   2.3013   0.13007
## weight         1  1207.4  1207.4  65.3575 7.808e-15 ***
## acceleration   1    57.5    57.5   3.1113   0.07853 .
## Residuals    393  7260.0    18.5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

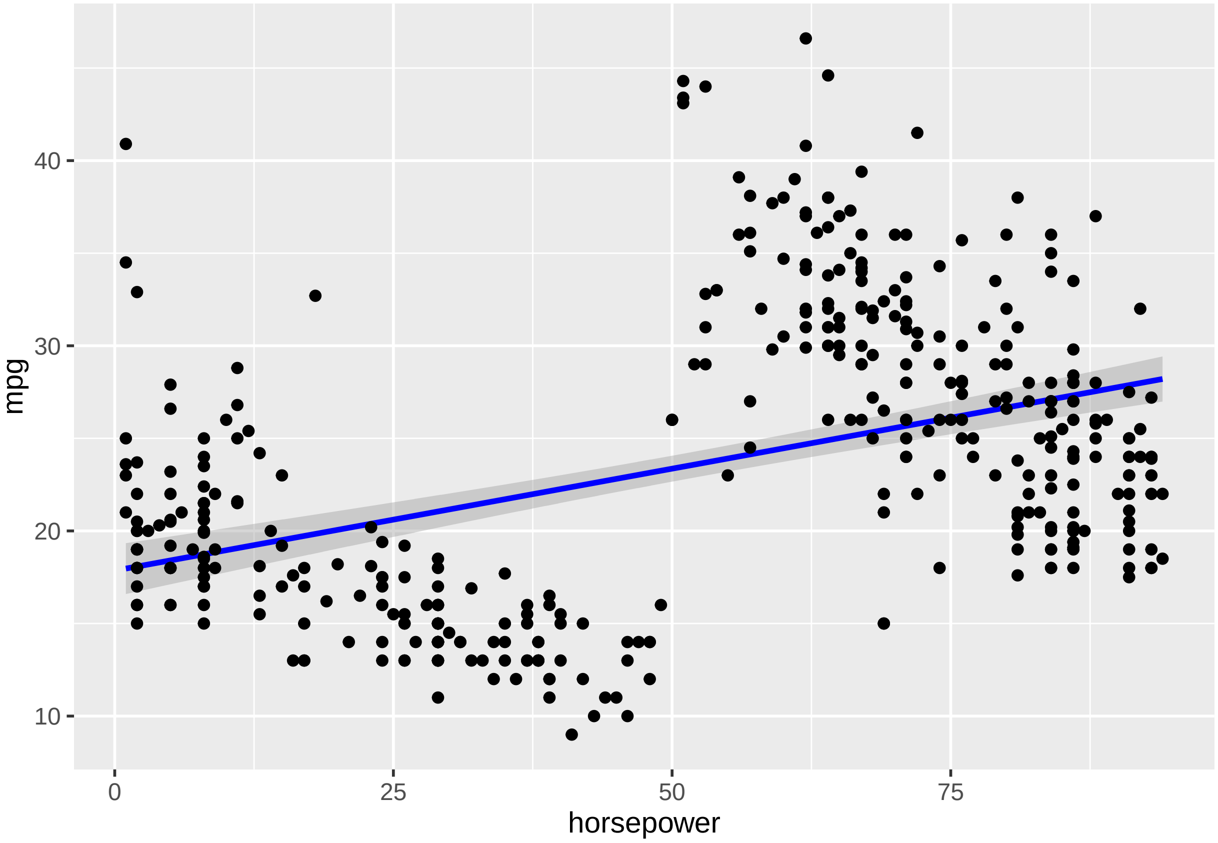
```
vcov(mpg_df_all_fit) # covariance matrix for model parameters
```

```
##                (Intercept)  displacement    horsepower        weight
## (Intercept)   3.986515e+00  6.137168e-05 -5.930190e-03 -4.915394e-04
## displacement  6.137168e-05  4.258039e-05  3.801308e-06 -4.420947e-06
## horsepower   -5.930190e-03  3.801308e-06  6.822224e-05  6.909209e-07
## weight       -4.915394e-04 -4.420947e-06  6.909209e-07  5.549501e-07
## acceleration -1.404886e-01  2.979955e-04 -2.333706e-05 -2.166473e-05
##               acceleration
## (Intercept)  -1.404886e-01
## displacement  2.979955e-04
## horsepower   -2.333706e-05
## weight       -2.166473e-05
## acceleration  9.532405e-03
```

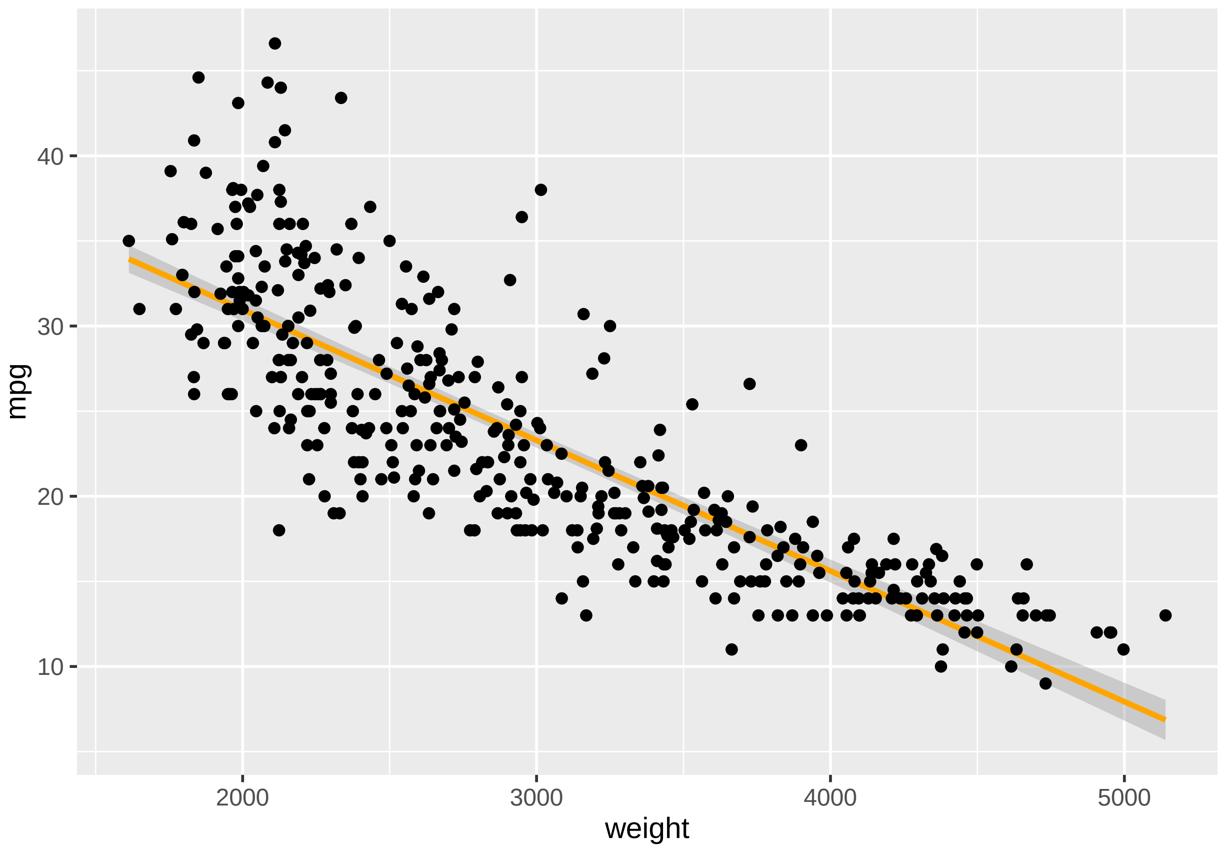
```
#influence(mpg_df_all_fit) # regression diagnostics

###########
ggplot(data = mpg_df, aes(y = mpg, x = displacement)) +      geom_smooth(method = "lm", color="green", fo
```



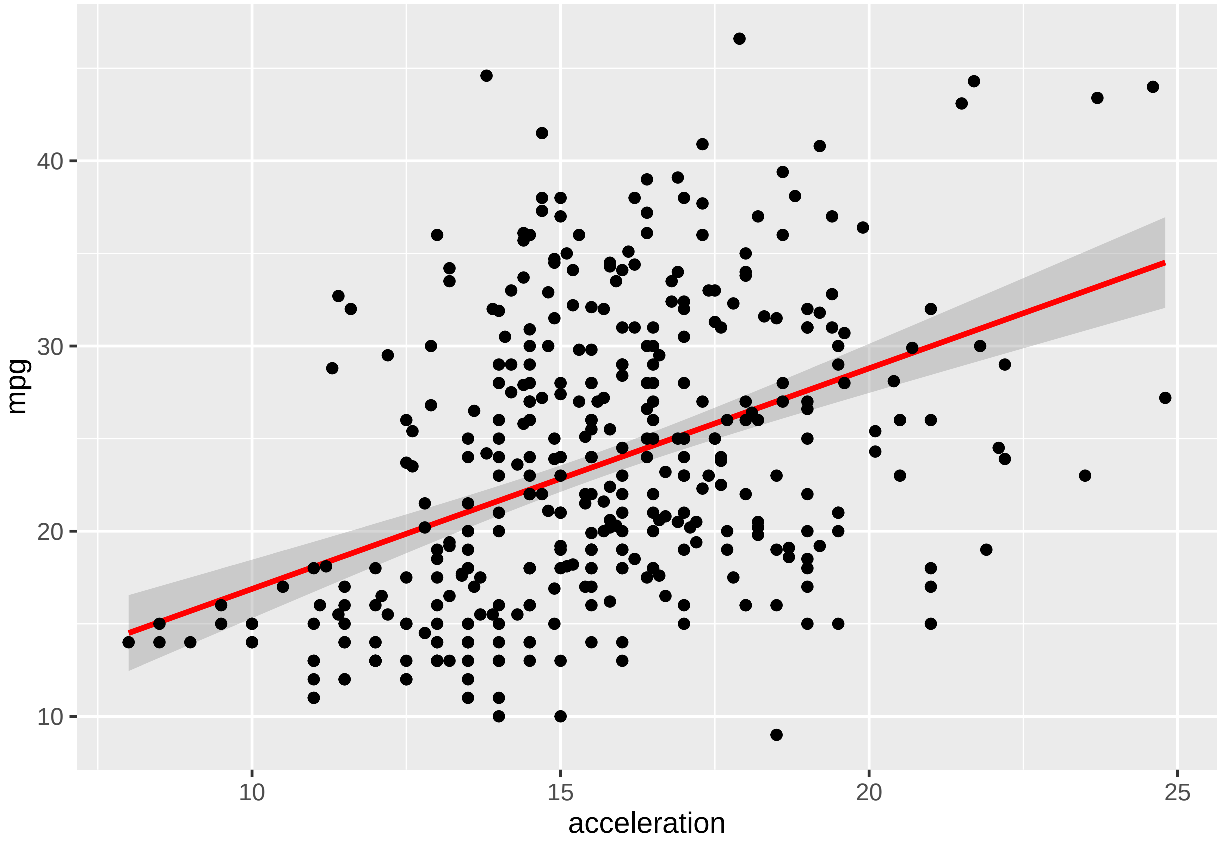
```
ggplot(mpg_df, aes(y=mpg, x=horsepower)) + geom_smooth(method = "lm", color="blue", formula = y ~ x) + g
```



```
ggplot(mpg_df, aes(y=mpg, x=weight)) + geom_smooth(method = "lm", color="orange", formula = y ~ x) + geo
```



```
ggplot(mpg_df, aes(y=mpg, x=acceleration)) + geom_smooth(method = "lm", color="red", formula = y ~ x) +
```


mpg
40
30
20
10
10
15
20
25
acceleration