Regression Project

```
#PART 1
library(ggplot2)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(readr)
library(car)
```

```
## Loading required package: carData
```

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:car':
##
##     recode
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(Matrix)
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```r
library(regclass)
```

```
## Loading required package: bestglm

## Loading required package: leaps

## Loading required package: VGAM

## Loading required package: stats4

## Loading required package: splines

##
## Attaching package: 'VGAM'

## The following object is masked from 'package:lmtest':
##
##     lrtest

## The following object is masked from 'package:car':
##
##     logit

## Loading required package: rpart

## Loading required package: randomForest

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin

## Important regclass change from 1.3:
## All functions that had a . in the name now have an _
## all.correlations -> all_correlations, cor.demo -> cor_demo, etc.
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'lattice'

## The following object is masked from 'package:regclass':
##
##     qq

##
## Attaching package: 'caret'

## The following object is masked from 'package:VGAM':
##
##     predictors
```

```r
columns <- c("Sex","Length", "Diameter", "Height", "Whole_wt", "Shuck_wt", "Visc_wt", "Shell_wt", "Rings

abalone <- read_csv("https://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data",col
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Sex = col_character(),
##   Length = col_double(),
##   Diameter = col_double(),
##   Height = col_double(),
##   Whole_wt = col_double(),
##   Shuck_wt = col_double(),
##   Visc_wt = col_double(),
##   Shell_wt = col_double(),
##   Rings = col_double()
## )
```

```r
abalone$Sex <- as.factor(abalone$Sex)
abalone
```

```
## # A tibble: 4,177 x 9
##    Sex   Length Diameter Height Whole_wt Shuck_wt Visc_wt Shell_wt Rings
##    <fct>  <dbl>    <dbl>  <dbl>    <dbl>    <dbl>   <dbl>    <dbl> <dbl>
##  1 M      0.455    0.365  0.095    0.514   0.224   0.101     0.15     15
##  2 M      0.35     0.265  0.09     0.226   0.0995  0.0485    0.07      7
##  3 F      0.53     0.42   0.135    0.677   0.256   0.142     0.21      9
##  4 M      0.44     0.365  0.125    0.516   0.216   0.114     0.155    10
##  5 I      0.33     0.255  0.08     0.205   0.0895  0.0395    0.055     7
##  6 I      0.425    0.3    0.095    0.352   0.141   0.0775    0.12      8
##  7 F      0.53     0.415  0.15     0.778   0.237   0.142     0.33     20
##  8 F      0.545    0.425  0.125    0.768   0.294   0.150     0.26     16
##  9 M      0.475    0.37   0.125    0.509   0.216   0.112     0.165     9
## 10 F      0.55     0.44   0.15     0.894   0.314   0.151     0.32     19
## # ... with 4,167 more rows
```

```
set.seed(42)
#Splitting dataset in train and test using 70/30 method
indexes <- sample(1:nrow(abalone), size = 0.3 * nrow(abalone))
abalone_train <- abalone[-indexes,]
abalone_test <- abalone[indexes,]
```

```
#Q1

rankMatrix(abalone[,2:8])[1]
```

```
## [1] 7
```

$$Age = \beta_0 + \beta_1 Height + \epsilon$$

with P1-P4 and full rank assumption

with the rank assumption and under P1-P4 which are: P1: Errors are centered P2: The model is homoskedastic. Variance of all the error terms are same. P3: Errors are uncorrelated. P4: Errors are Gaussian. We also assume that no high leverage outliers are present. In order to study those hypothesis, we'll be visualing the regression line and then the residuals graphically to observe if they satisfy our assumptions. We'll also build the following tests to further investigate the postulates 2,3 and 4: Breush-Pagan test for P2, Durbin-Watson test for P3, Shapiro-Wilks test for P4 We'll check if our full rank assumption is met. We will also be computing the Cook distances to detect if there are outliers that change too much our estimations for beta's. Finally, we'll build confidence intervals for the betas to study the efficiency of the model.

```
#Q2

summary(abalone)
```

```
##  Sex          Length         Diameter          Height          Whole_wt
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000   Min.   :0.0020
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415
##  M:1528   Median :0.545   Median :0.4250   Median :0.1400   Median :0.7995
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395   Mean   :0.8287
##           3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650   3rd Qu.:1.1530
##           Max.   :0.815   Max.   :0.6500   Max.   :1.1300   Max.   :2.8255
##     Shuck_wt         Visc_wt          Shell_wt          Rings
```

```
##  Min.   :0.0010   Min.   :0.0005   Min.   :0.0015   Min.   : 1.000
##  1st Qu.:0.1860   1st Qu.:0.0935   1st Qu.:0.1300   1st Qu.: 8.000
##  Median :0.3360   Median :0.1710   Median :0.2340   Median : 9.000
##  Mean   :0.3594   Mean   :0.1806   Mean   :0.2388   Mean   : 9.934
##  3rd Qu.:0.5020   3rd Qu.:0.2530   3rd Qu.:0.3290   3rd Qu.:11.000
##  Max.   :1.4880   Max.   :0.7600   Max.   :1.0050   Max.   :29.000
```

```
diag(var(abalone))
```

```
## Warning in var(abalone): NAs introduced by coercion
```

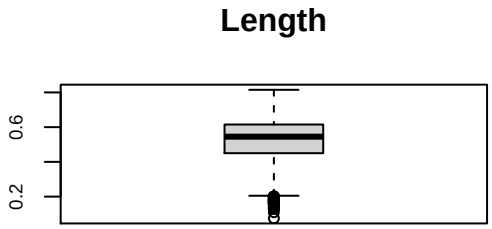
```
##          Sex       Length     Diameter       Height     Whole_wt     Shuck_wt
##           NA  0.014422308  0.009848551  0.001749503  0.240481389  0.049267551
##      Visc_wt     Shell_wt        Rings
##  0.012015284  0.019377383 10.395265947
```

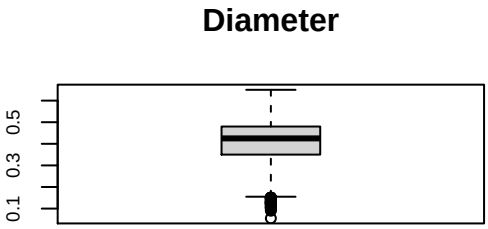
```
sqrt(diag(var(abalone)))
```

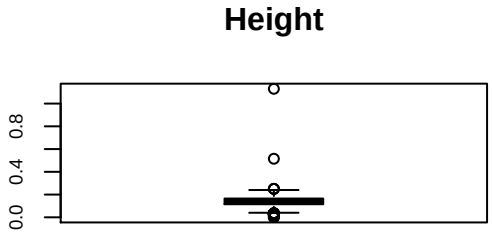
```
## Warning in var(abalone): NAs introduced by coercion
```

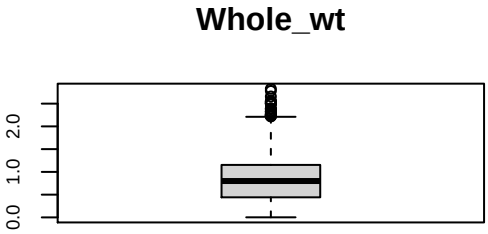
```
##        Sex     Length   Diameter     Height   Whole_wt   Shuck_wt    Visc_wt
##         NA 0.12009291 0.09923987 0.04182706 0.49038902 0.22196295 0.10961425
##   Shell_wt      Rings
## 0.13920267 3.22416903
```

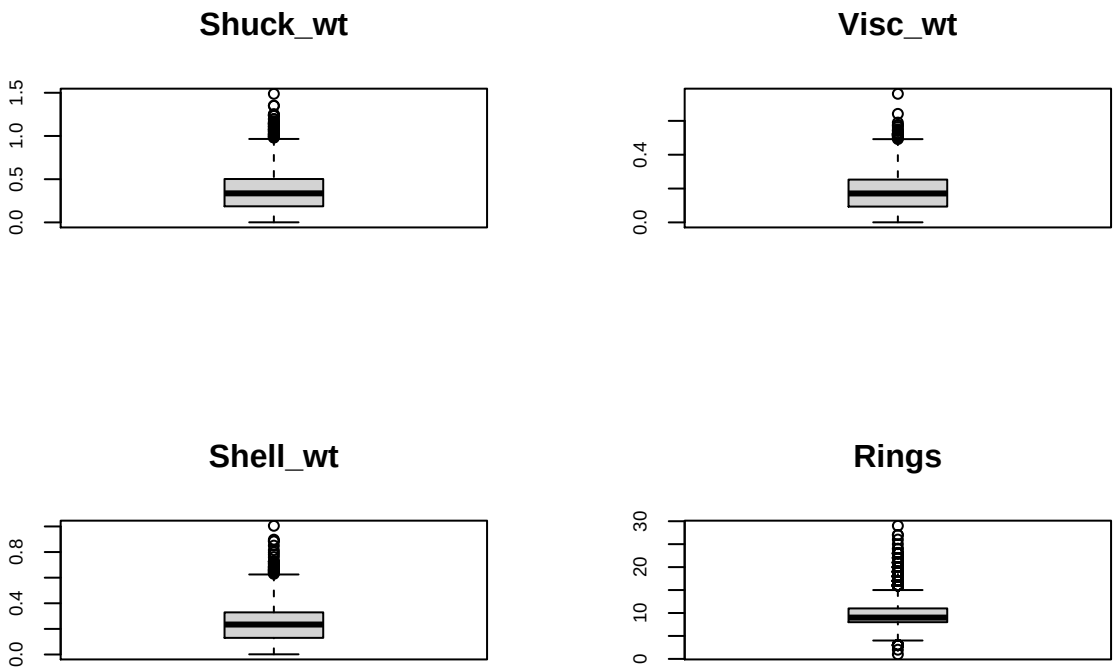
```
par(mfrow=c(2,2))
for (i in 2:ncol(abalone)){
  boxplot(abalone[i], boxwex=0.5, cex.axis=0.75, main=colnames(abalone[i]))
}
```


Length
0.2
0.6

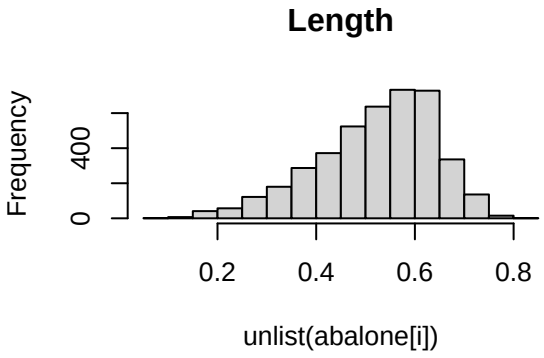

Diameter
0.1
0.3
0.5

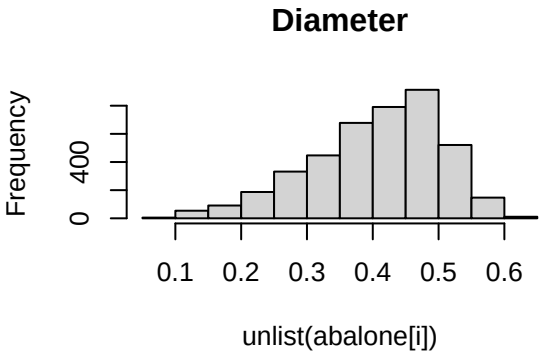

Height
0.0
0.4
0.8

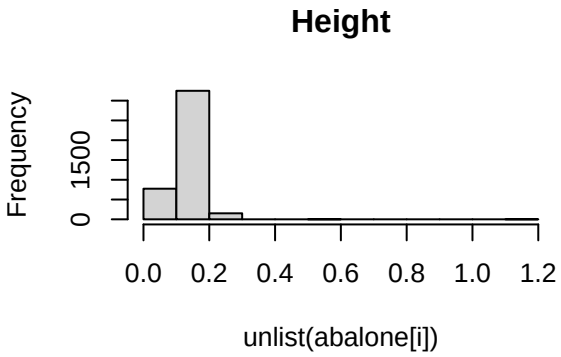

Whole_wt
0.0
1.0
2.0

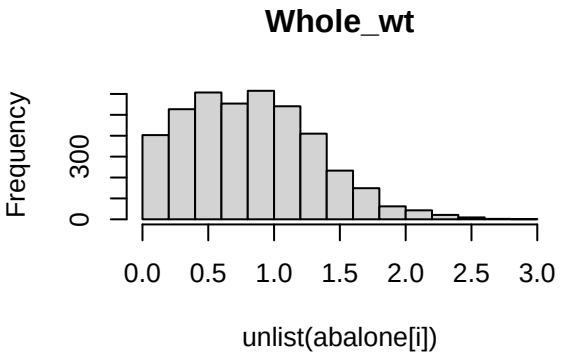


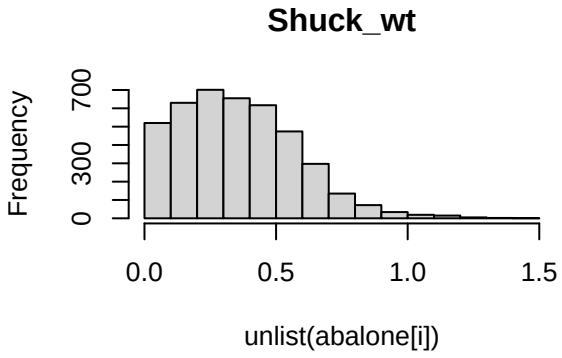

```
par(mfrow=c(2,2))
for (i in 2:ncol(abalone)){
  hist(unlist(abalone[i]), main=colnames(abalone[i]))
}
```

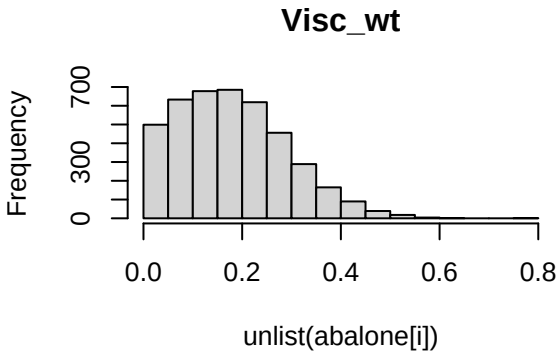

Length
Frequency
0
400
0.2
0.4
0.6
0.8
unlist(abalone[i])

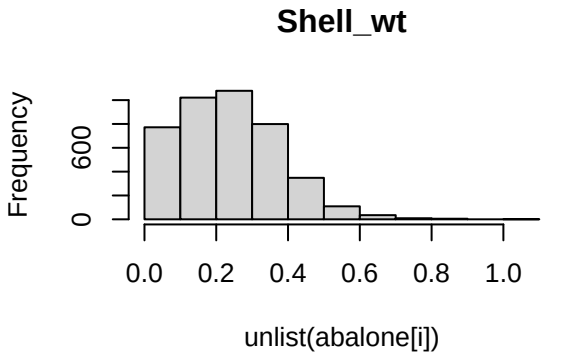

Diameter
Frequency
0
400
0.1
0.2
0.3
0.4
0.5
0.6
unlist(abalone[i])

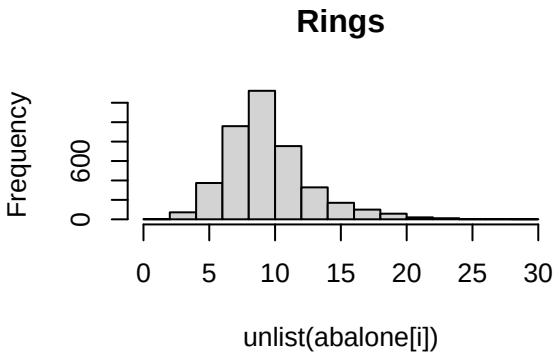

Height
Frequency
0
1500
0.0
0.2
0.4
0.6
0.8
1.0
1.2
unlist(abalone[i])


Whole_wt
Frequency
0
300
0.0
0.5
1.0
1.5
2.0
2.5
3.0
unlist(abalone[i])





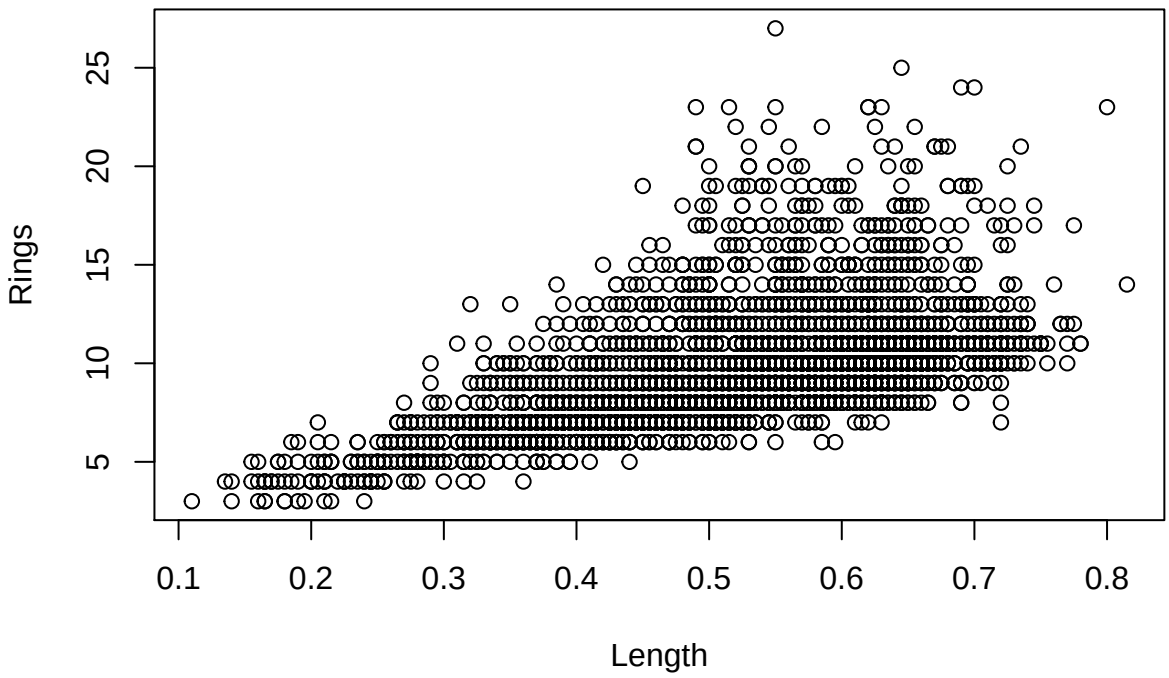


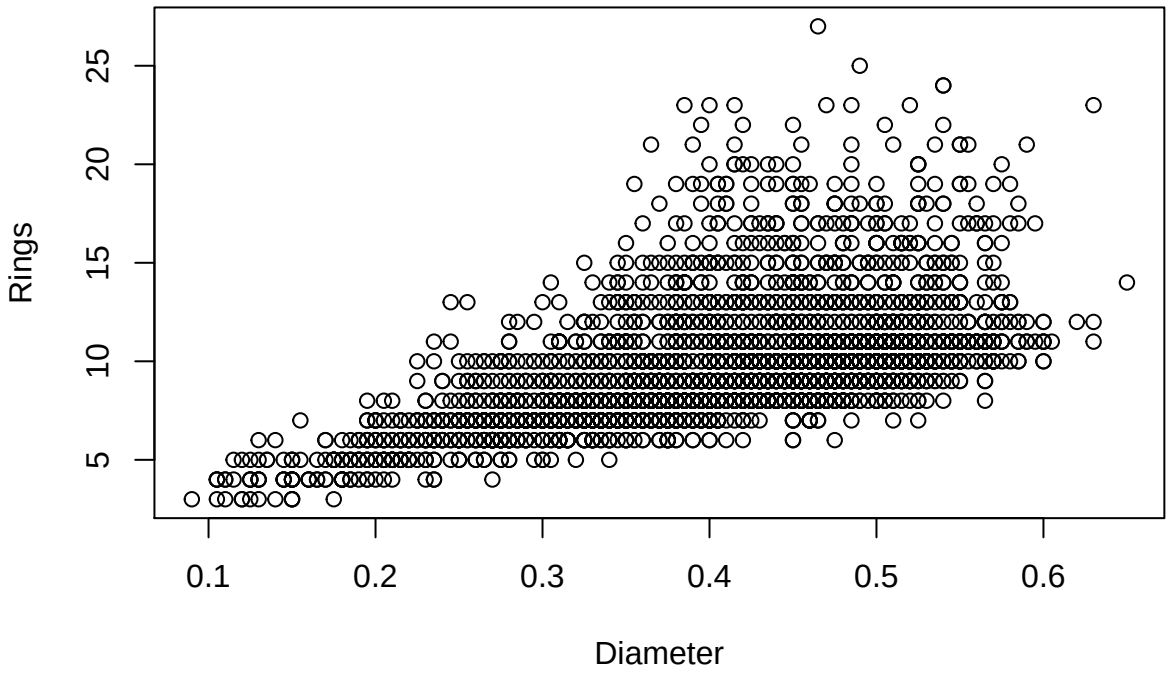


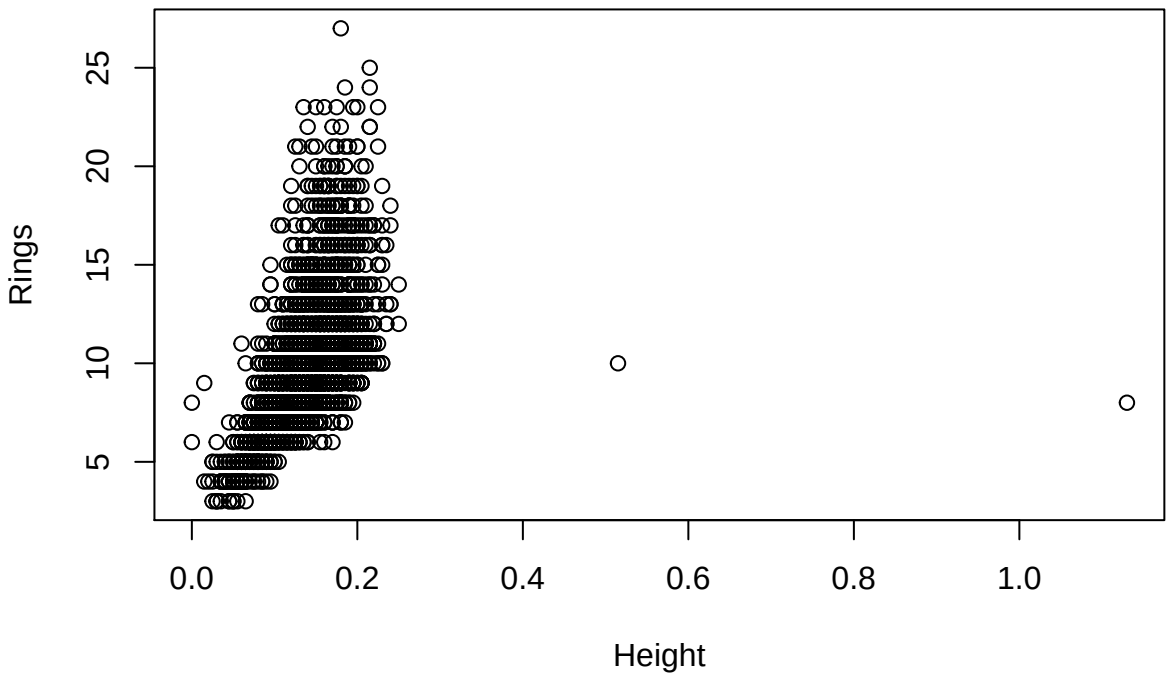
Considering the boxplots of all the features in the dataset, all the variables present outliers (by the definition of quantiles and interquantile range). Height has two significant outliers. In addition, The medians of the various different types of weights are more or less close to each other.

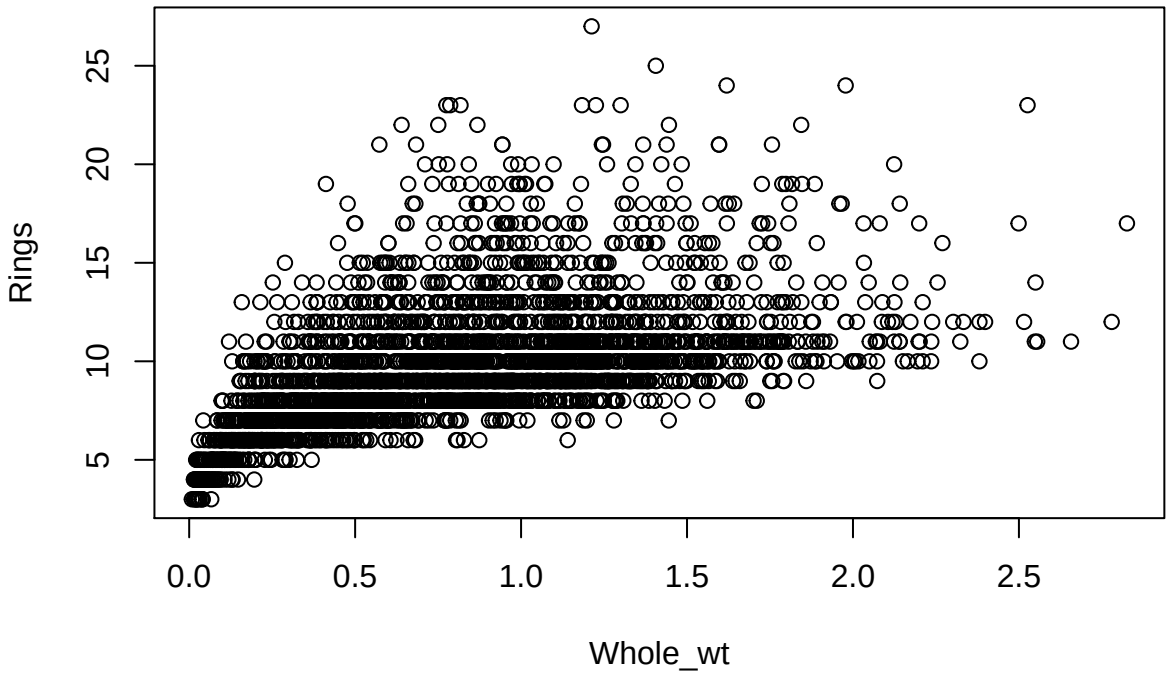
Considering the Histograms, It's easy to see how the distribution of Rings is more or less centered, the Length and the Diameter are left skewed (the frequency of larger values is bigger), while all the others present are right skewed (frequency of smaller values is bigger).

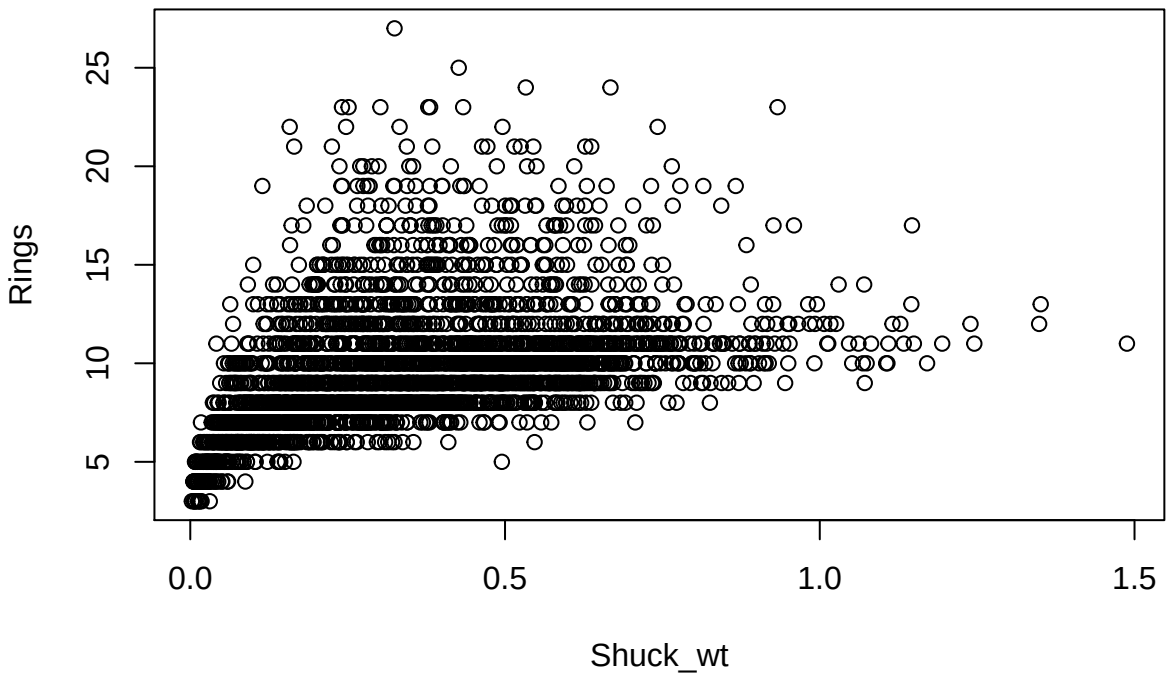
```
#Q3

plot(Rings ~ Length + Diameter + Height + Whole_wt + Shuck_wt + Visc_wt + Shell_wt, data=abalone_train)
```

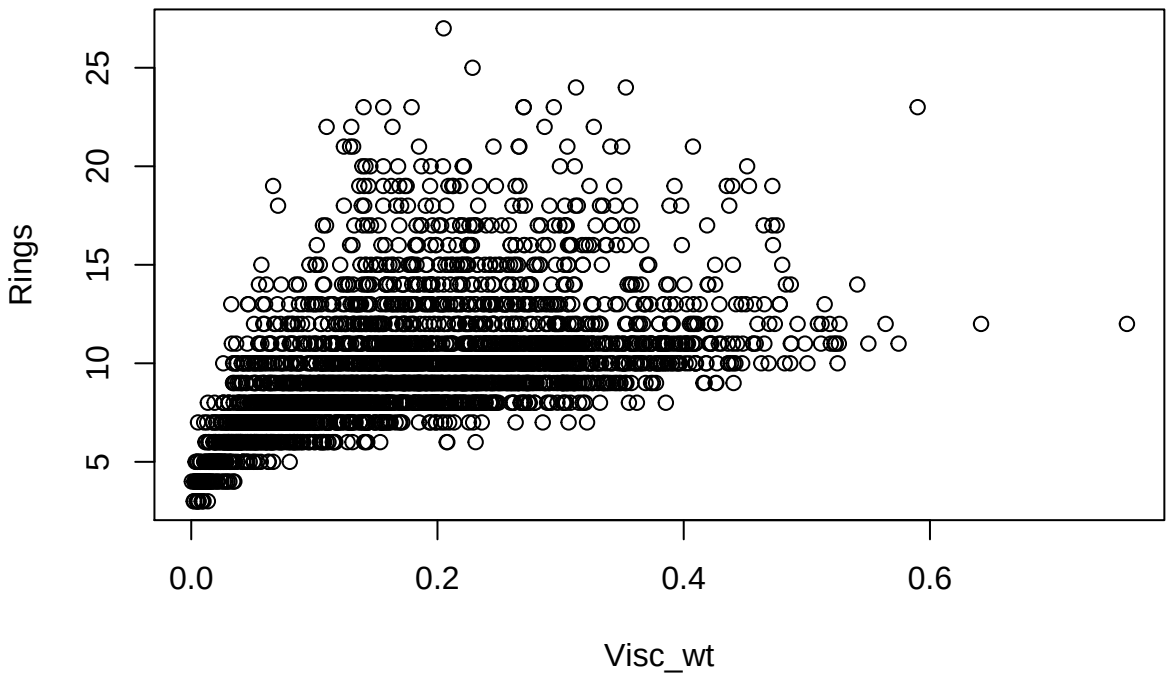

Rings
5
10
15
20
25
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Length

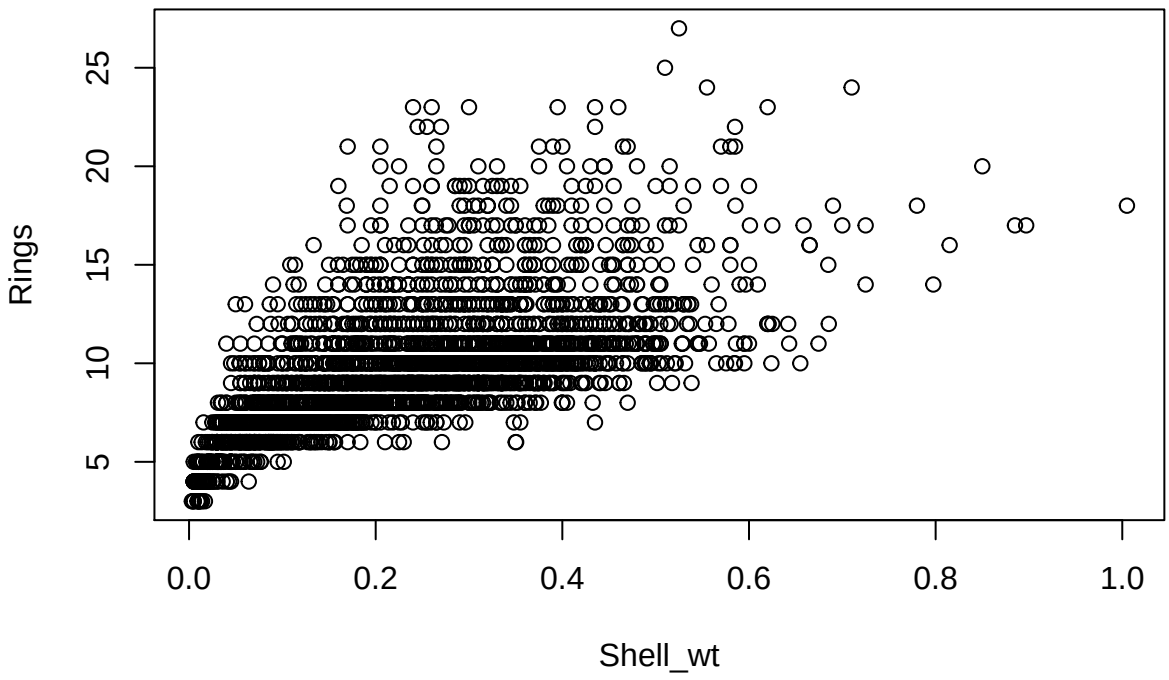

Rings
5
10
15
20
25
0.1
0.2
0.3
0.4
0.5
0.6
Diameter


Rings
5
10
15
20
25
0.0
0.2
0.4
0.6
0.8
1.0
Height


Rings
5
10
15
20
25
0.0
0.5
1.0
1.5
2.0
2.5
Whole_wt


Rings
5
10
15
20
25
0.0
0.5
1.0
1.5
Shuck_wt


Rings
5
10
15
20
25
0.0
0.2
0.4
0.6
Visc_wt



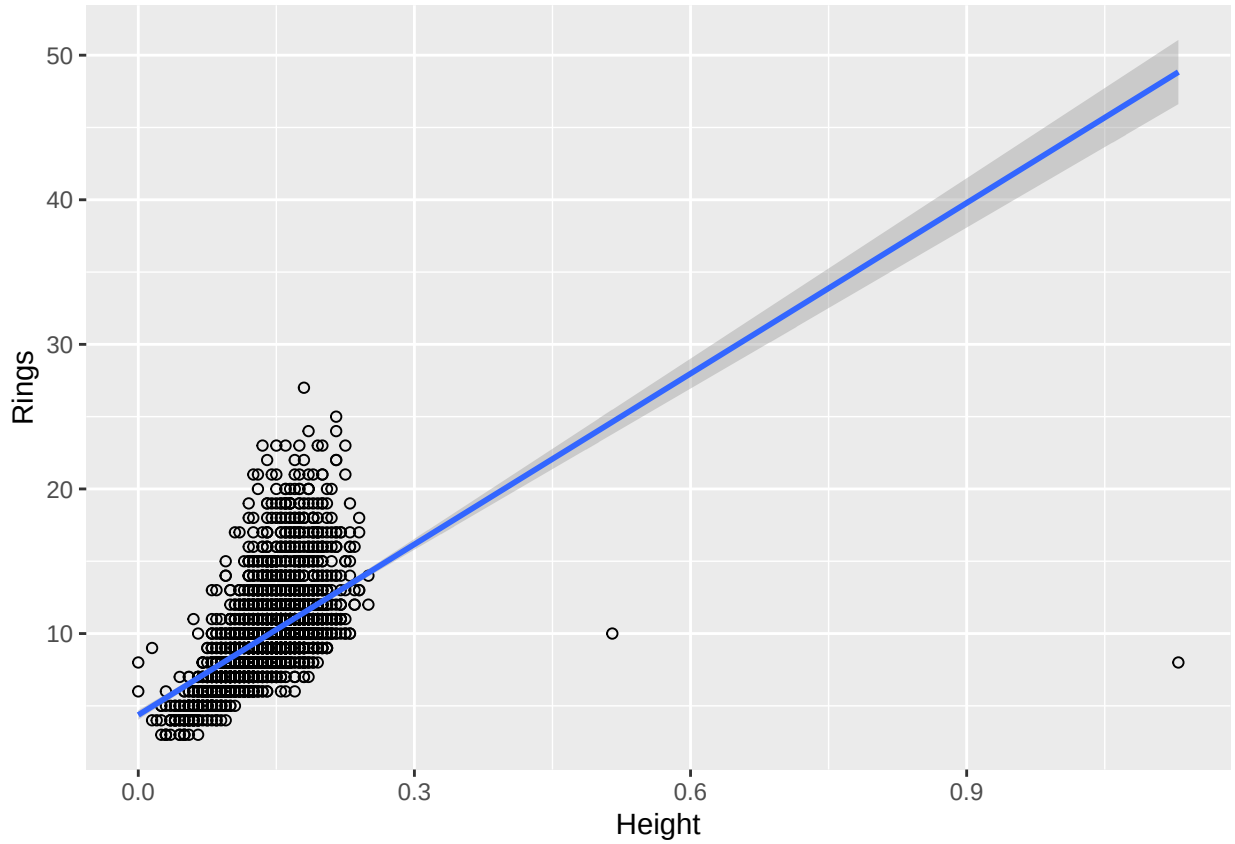
We can graphically see the positive correlation between Rings (and consequently, age of Abalone) and Height, confirming the biologists' hypothesis. In general, from the scatter plots, we can also see that there are linear correlations between Rings and other variables such as Length and Shell Weight.

```
#Q4

linear_mod = lm(Rings ~ Height, data=abalone_train)
summary(linear_mod)
```

```
##
## Call:
## lm(formula = Rings ~ Height, data = abalone_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -40.833  -1.663  -0.663   0.747  15.550
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.3665     0.1674   26.08   <2e-16 ***
## Height       39.3507     1.1426   34.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.678 on 2922 degrees of freedom
## Multiple R-squared:  0.2887, Adjusted R-squared:  0.2885
## F-statistic:  1186 on 1 and 2922 DF,  p-value: < 2.2e-16
```

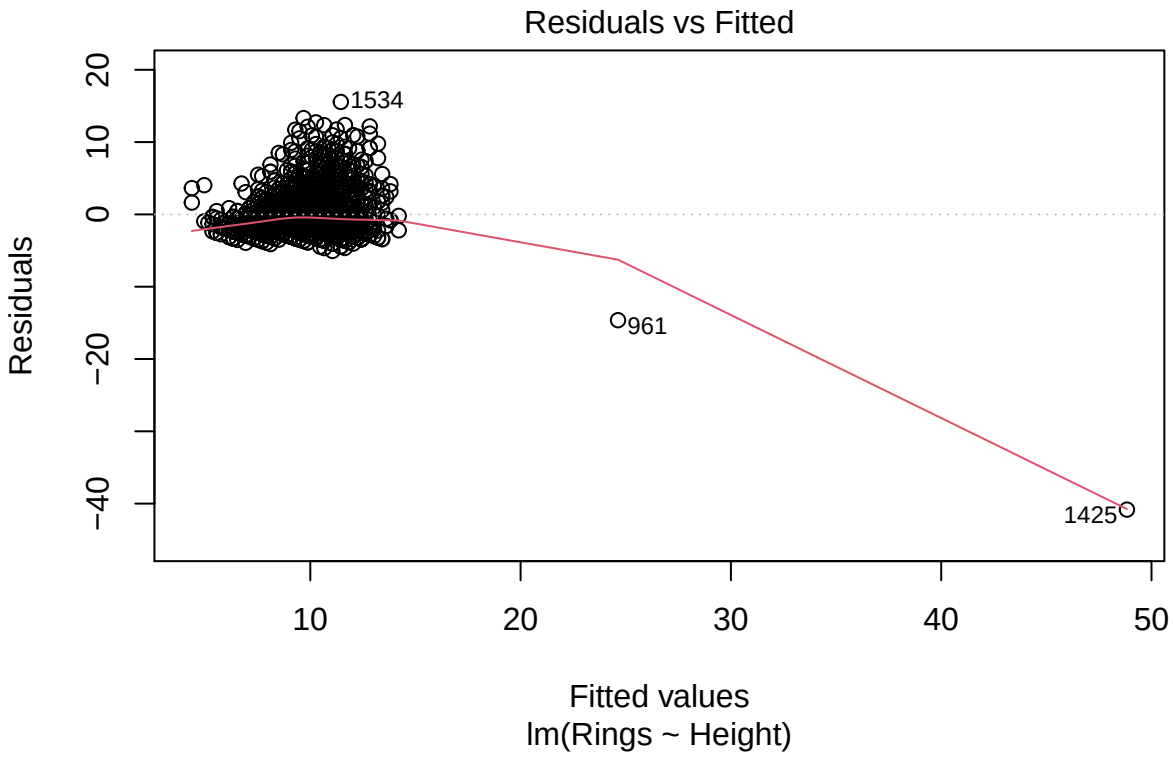
```
#Q5

ggplot(abalone_train, aes(x=Height, y=Rings)) + geom_point(shape=1) + geom_smooth(method=lm)
```

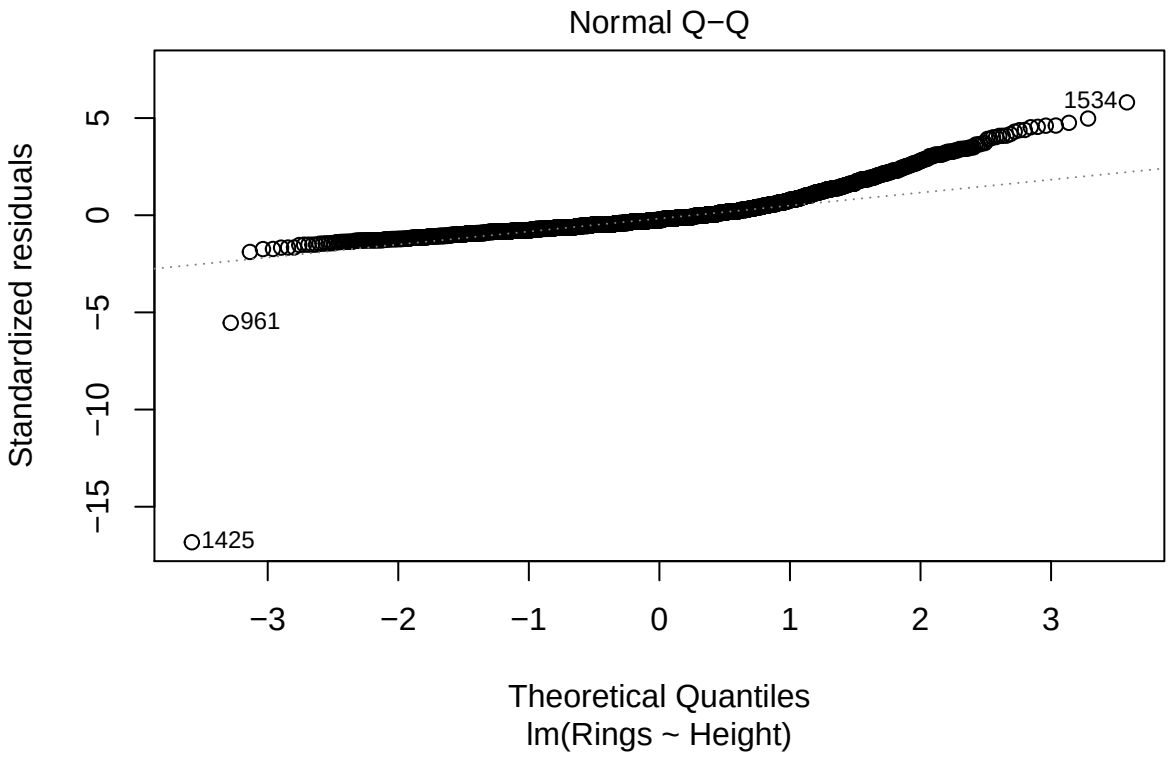
```
## `geom_smooth()` using formula 'y ~ x'
```

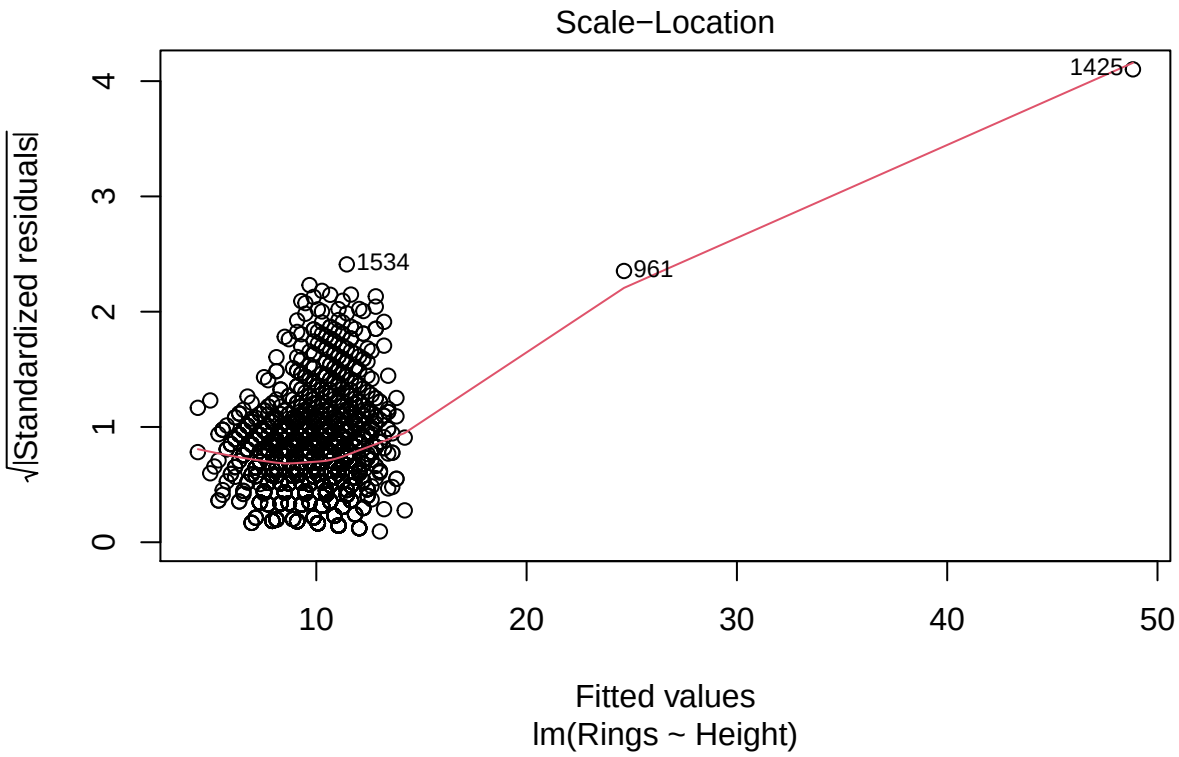


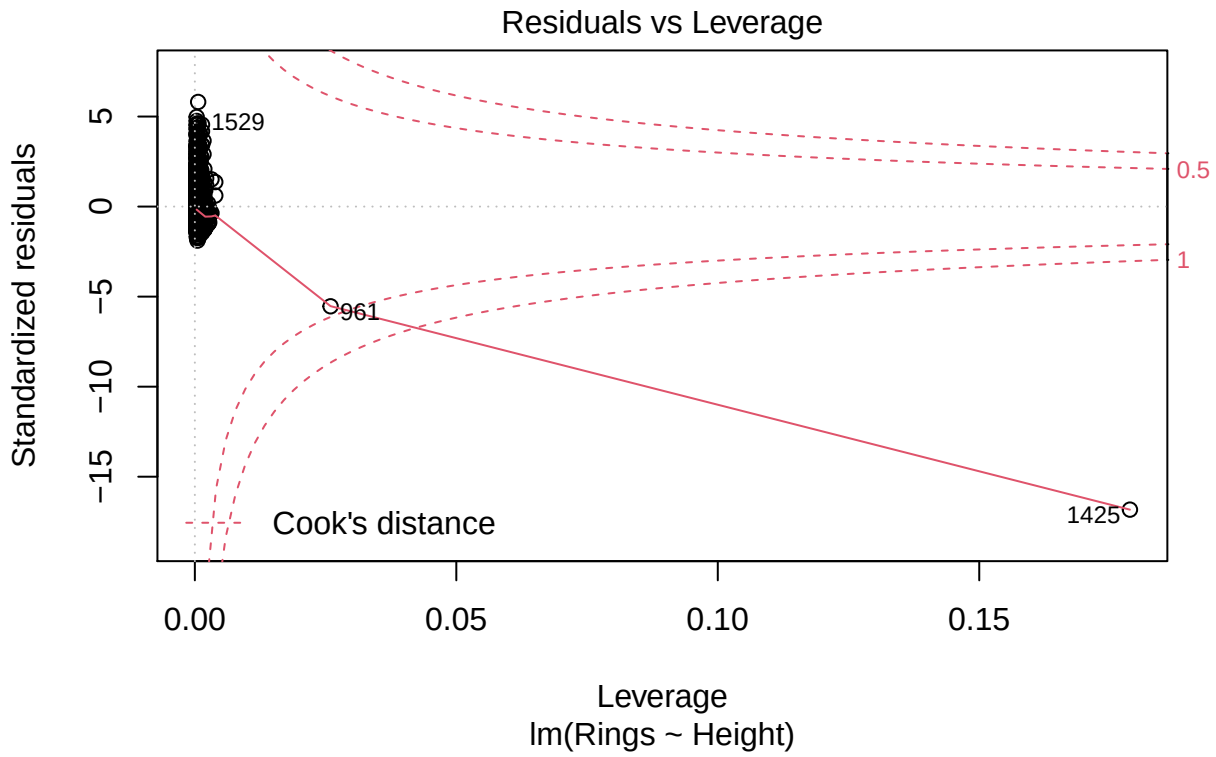
From the graph can be seen that are present two outliers of the predictor Height. Those two points are high leverage and are affecting the fit of the line. The line doesn't seem to be the best fit. Taking a polynomial or exponential function of Height might provide a better fit.

```
#Q6

plot(linear_mod)
```


Residuals vs Fitted
Residuals
20
10
0
−20
−40
1534
961
1425
10
20
30
40
50
Fitted values
lm(Rings ~ Height)


Normal Q−Q
Standardized residuals
5
0
−5
−10
−15
1534
961
1425
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(Rings ~ Height)

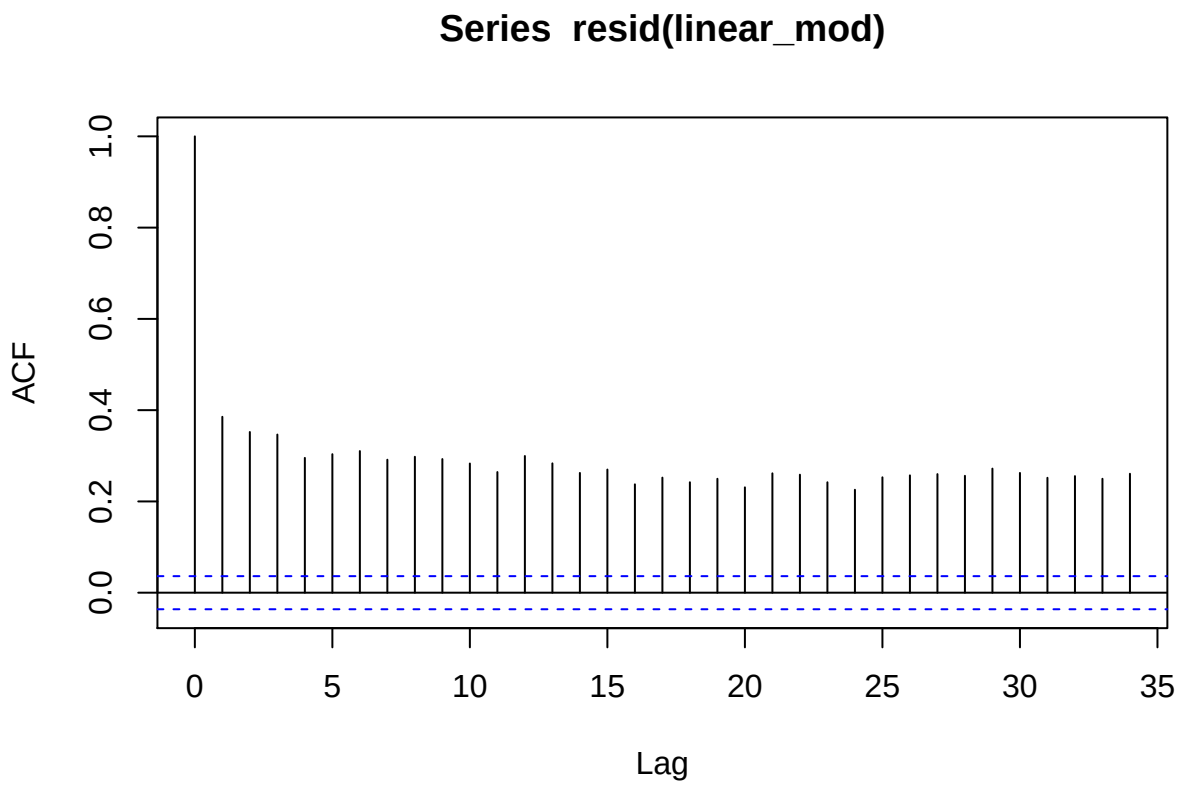

Scale−Location
√|Standardized residuals|
0
1
2
3
4
10
20
30
40
50
1425
961
1534
Fitted values
lm(Rings ~ Height)




```
durbinWatsonTest(linear_mod, max.lag=10)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.3854835      1.226760       0
##    2       0.3521707      1.292799       0
##    3       0.3465464      1.304034       0
##    4       0.2954202      1.405995       0
##    5       0.3035207      1.385262       0
##    6       0.3103434      1.371577       0
##    7       0.2914158      1.408542       0
##    8       0.2978661      1.395501       0
##    9       0.2928285      1.405469       0
##   10       0.2831512      1.424605       0
##  Alternative hypothesis: rho[lag] != 0
```

```
acf(resid(linear_mod))
```




```
bptest(linear_mod)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  linear_mod
## BP = 587.76, df = 1, p-value < 2.2e-16
```

```
shapiro.test(resid(linear_mod))
```

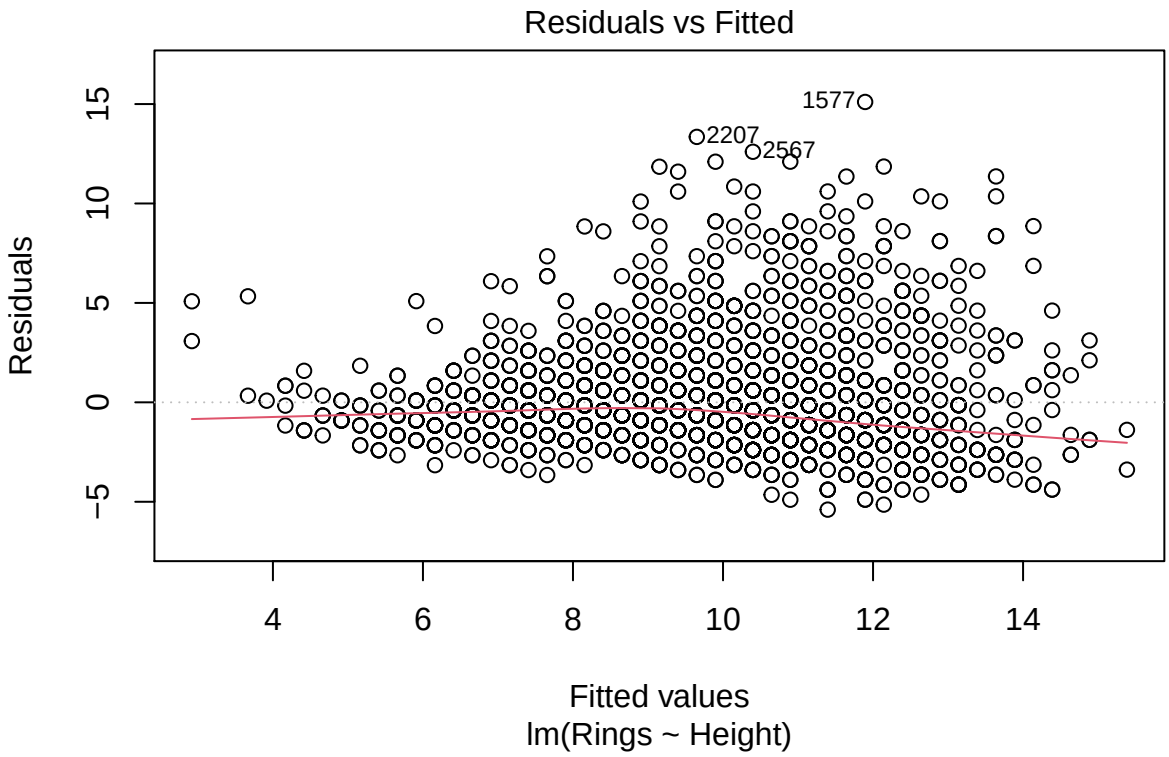
```
##
##  Shapiro-Wilk normality test
##
## data:  resid(linear_mod)
## W = 0.82351, p-value < 2.2e-16
```

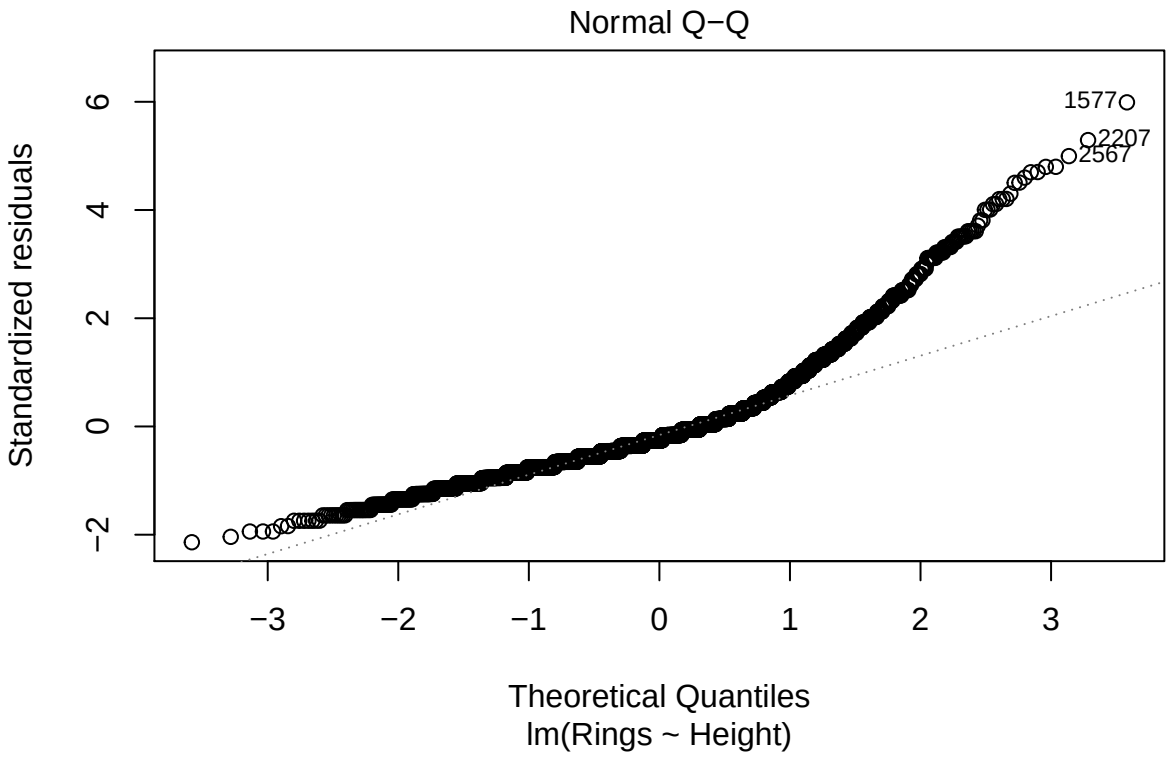
The errors are not centered since the Residuals-Fitted graph does not have a line which on average is zero due to the presence of two outliers in the data. The errors are Gaussian in the lower quantiles since in the Normal Q-Q plot more or less lies on the line that represent the quantiles of the standard normal. The plot diverges at higher quantiles, suggesting that we could perform feature engineering. The results of the Shapiro-Wilkes test also do not suggest Gaussian distribution of residuals. Possibly due to the presence of outliers, there is heteroskedasticity since the line in the Scale-Location plot is really far from being horizontal. In addition, the studentized Breusch-Pagan test has a very low p-value, so there is high probability of heteroskedasticity. The results of Durbin-Watson test suggest autocorrelation. This may be due to ordering in the data.

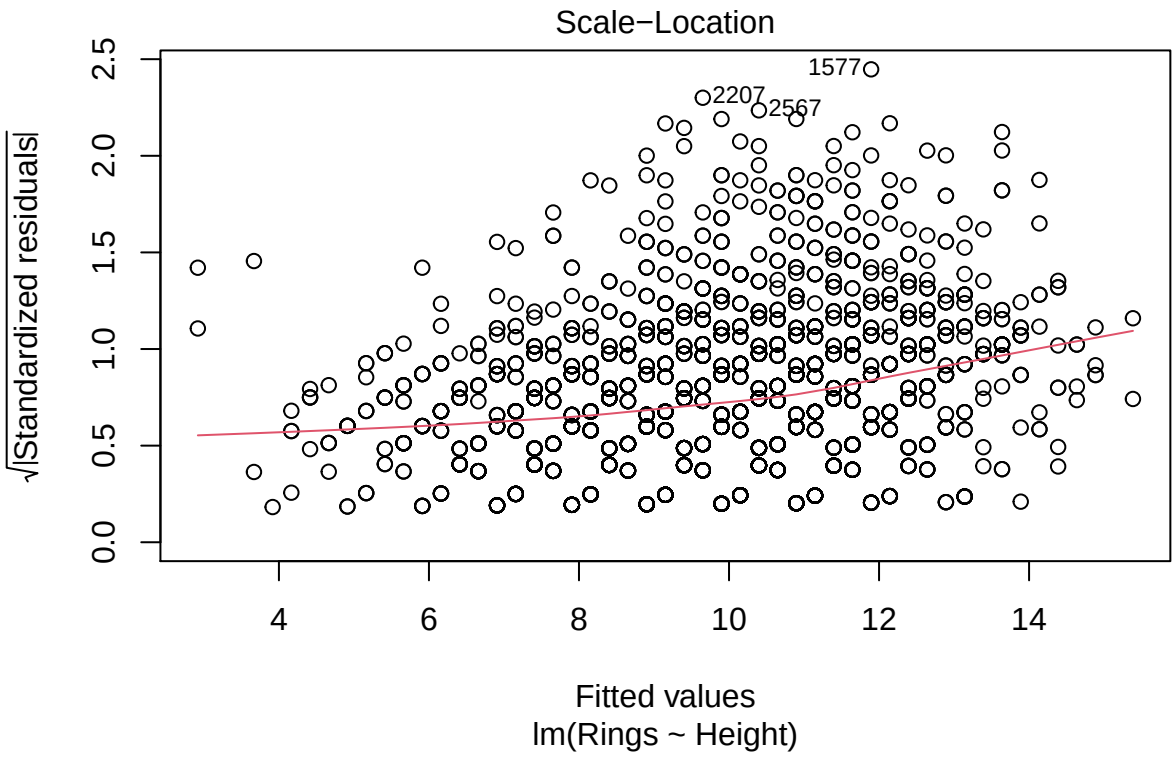
```r
#we remove the two outliers and sort the data randomly
new_abalone_train = abalone_train[-c(961, 1425),]

set.seed(1234)
new_abalone_train = new_abalone_train[sample(nrow(new_abalone_train)), ]

linear_mod_new = lm(Rings ~ Height, data=new_abalone_train)
summary(linear_mod_new)
```

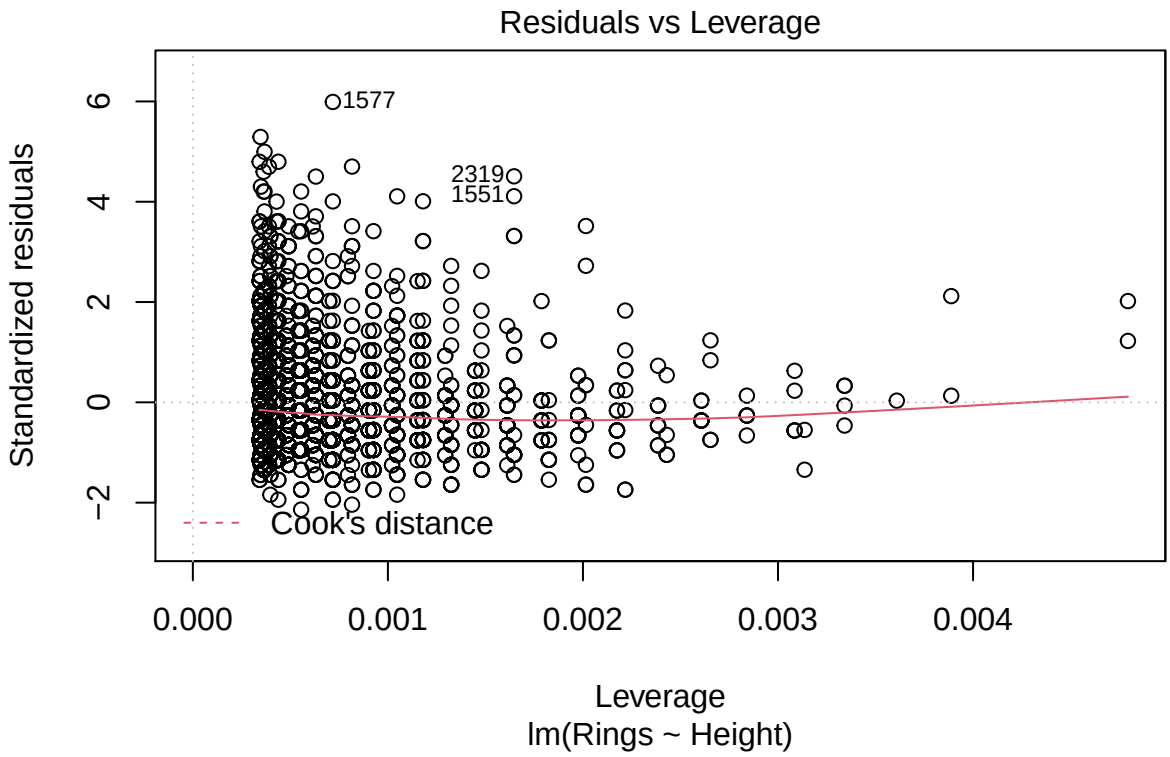
```
##
## Call:
## lm(formula = Rings ~ Height, data = new_abalone_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3958 -1.6451 -0.6451  0.8480 15.1056
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.9192     0.1747   16.71   <2e-16 ***
## Height       49.8620     1.2068   41.32   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.523 on 2920 degrees of freedom
## Multiple R-squared:  0.369,  Adjusted R-squared:  0.3687
## F-statistic:  1707 on 1 and 2920 DF,  p-value: < 2.2e-16
```

```r
plot(linear_mod_new)
```


Residuals vs Fitted
1577
2207
2567
Residuals
15
10
5
0
−5
4
6
8
10
12
14
Fitted values
lm(Rings ~ Height)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Rings ~ Height)
1577
2207
2567
−2
0
2
4
6
−3
−2
−1
0
1
2
3

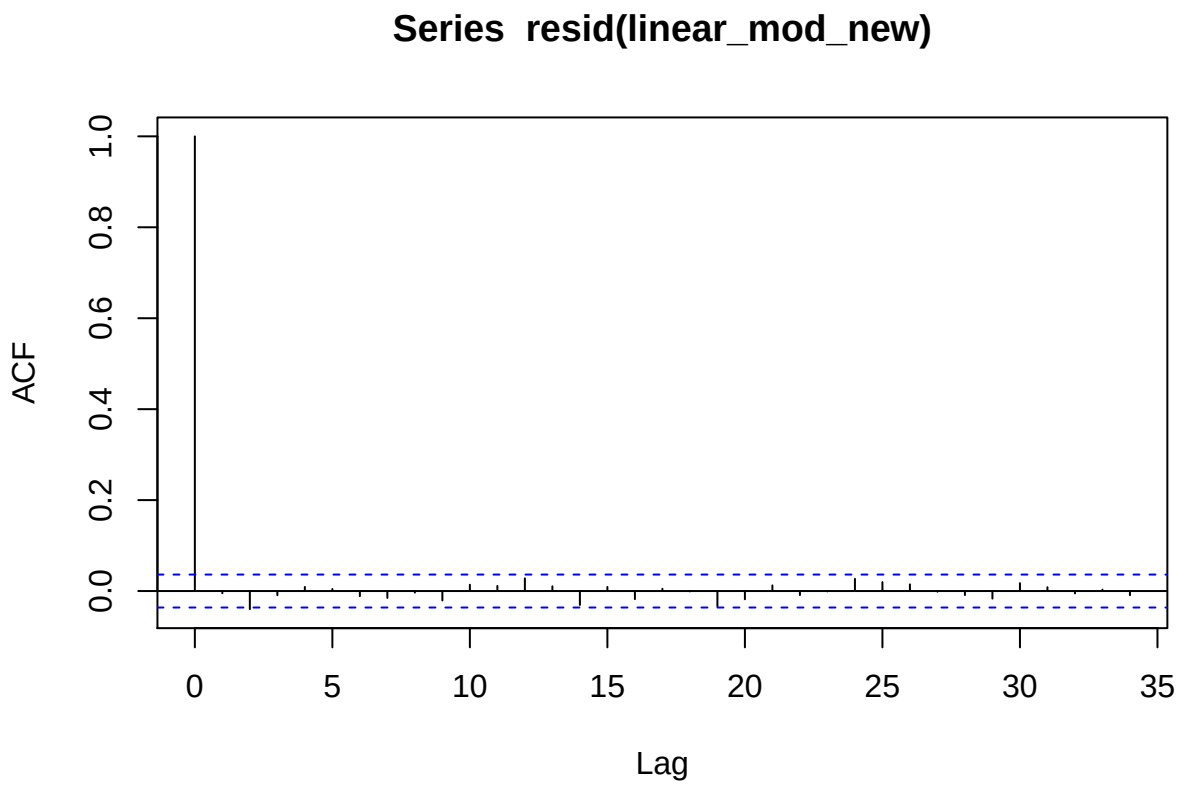

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
1577
2207
2567
4
6
8
10
12
14
Fitted values
lm(Rings ~ Height)




```
durbinWatsonTest(linear_mod_new, max.lag=10)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1    -0.004990487      2.009648   0.836
##    2    -0.040032692      2.079713   0.022
##    3    -0.009243499      2.018020   0.612
##    4     0.009085316      1.981345   0.632
##    5     0.004536839      1.990338   0.828
##    6    -0.011322523      2.021475   0.514
##    7    -0.015229135      2.028778   0.390
##    8    -0.003485785      2.005055   0.828
##    9    -0.020834181      2.038914   0.220
##   10     0.013903480      1.969394   0.490
##  Alternative hypothesis: rho[lag] != 0
```

```
acf(resid(linear_mod_new))
```




```
bptest(linear_mod_new)
```

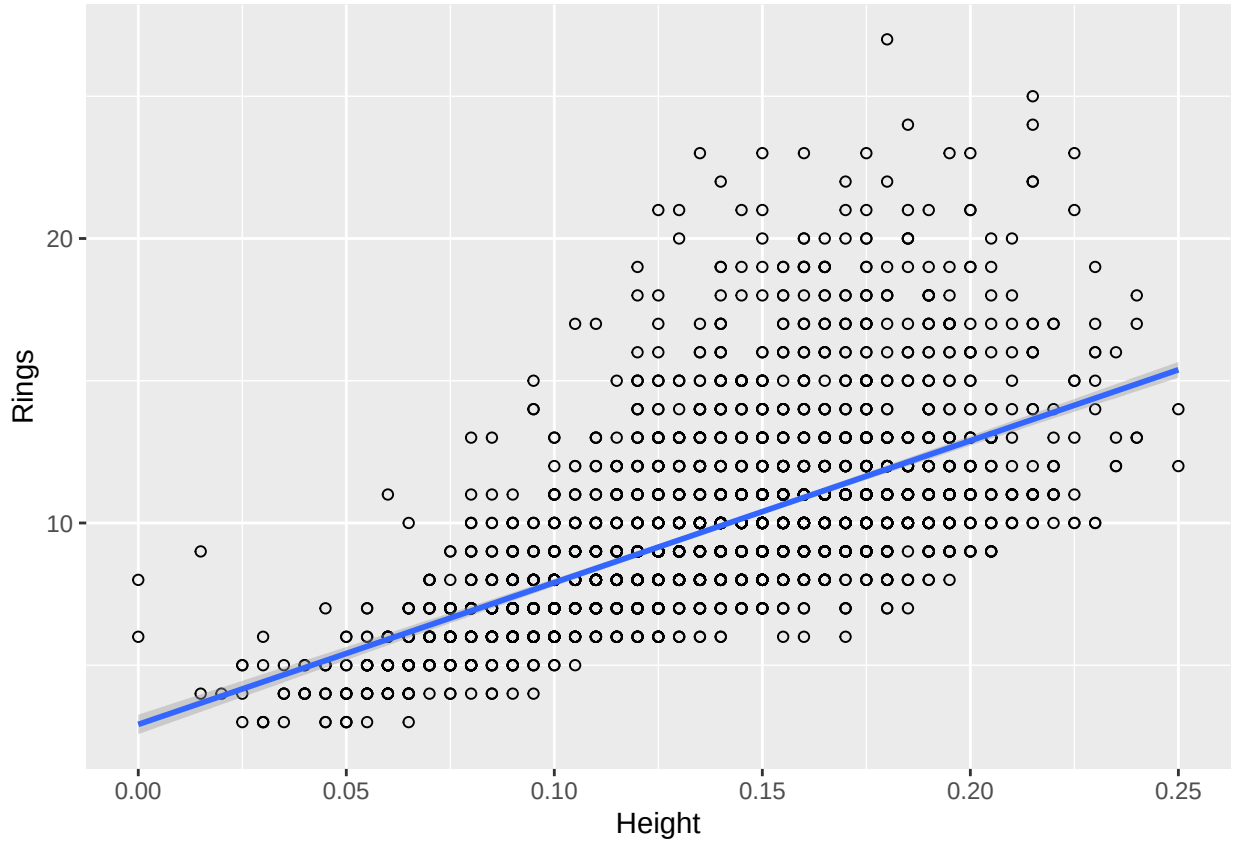
```
##
##  studentized Breusch-Pagan test
##
## data:  linear_mod_new
## BP = 77.169, df = 1, p-value < 2.2e-16
```

```
shapiro.test(resid(linear_mod_new))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(linear_mod_new)
## W = 0.87818, p-value < 2.2e-16
```

```
ggplot(new_abalone_train, aes(x=Height, y=Rings)) + geom_point(shape=1) + geom_smooth(method=lm)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



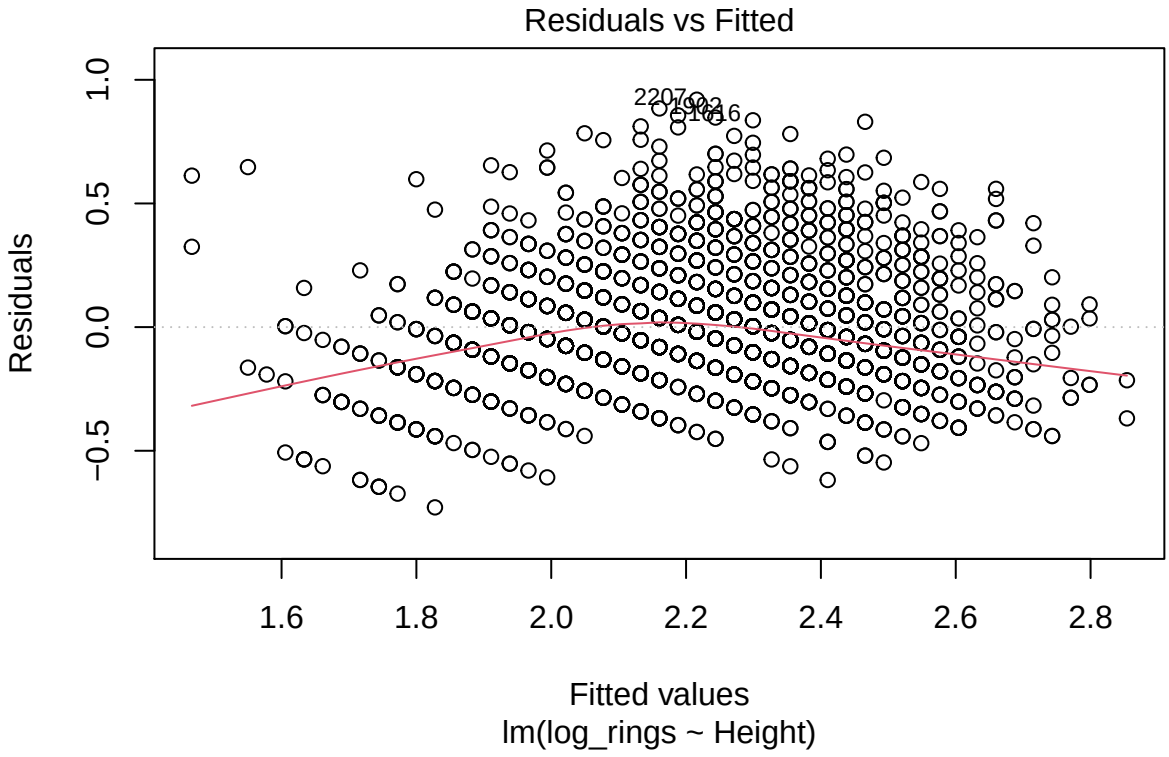
We removed the outliers sequentially till none of the points have Cook's distance greater than 1 From the new graph we can see that the elimination of the outliers allow us to better satisfy the postulates. The errors are more centered since the red line in the residuals vs fitted plot is on average more close to 0. However, we still see some trend in the variance of the residuals. Furthermore, the results of the B-P test also suggest that there exists heteroskedasticity. The results of the Q-Q plot and the S-W test suggest that the residuals do not follow a Gaussian Distribution. Sorting the data seems to have removed the apparent autocorrelation in the residual terms as seen from the results of the D-W test.

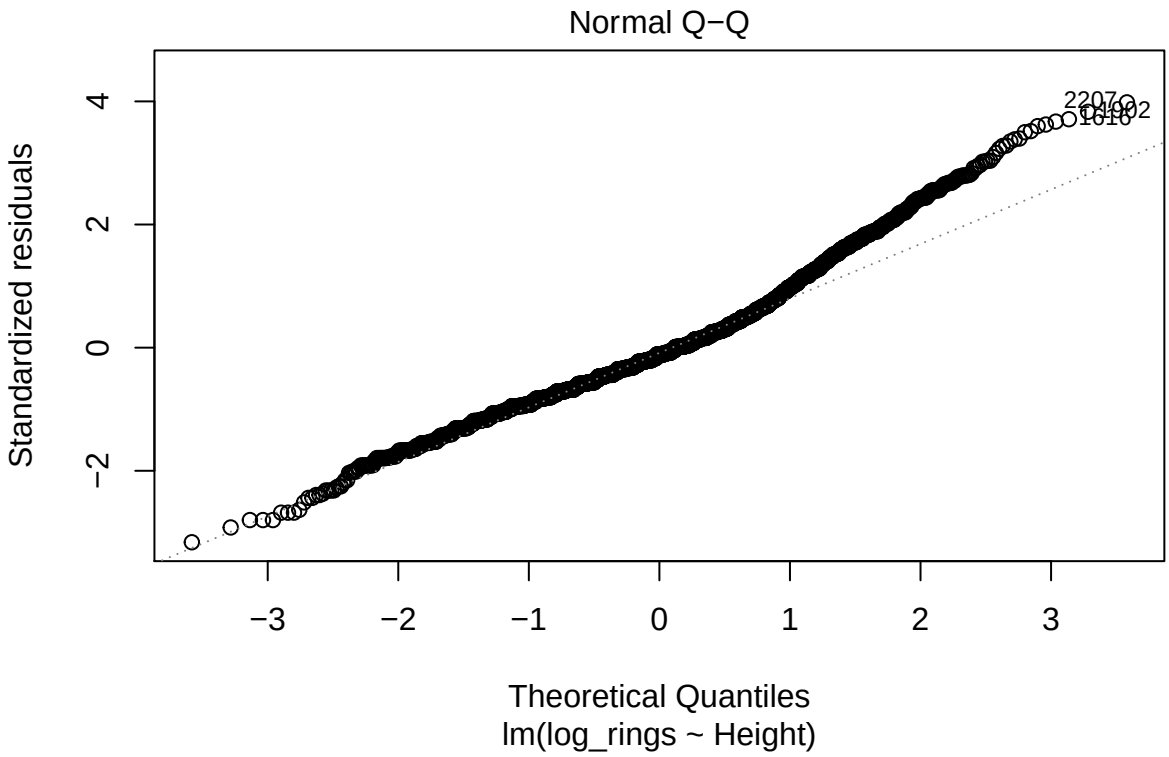
```
#here we used the logarithm of the number of Rings to get a better fit
new_abalone_train$log_rings = log(new_abalone_train$Rings)

linear_mod_log_simple = lm(log_rings ~ Height, data=new_abalone_train)
summary(linear_mod_log_simple)
```

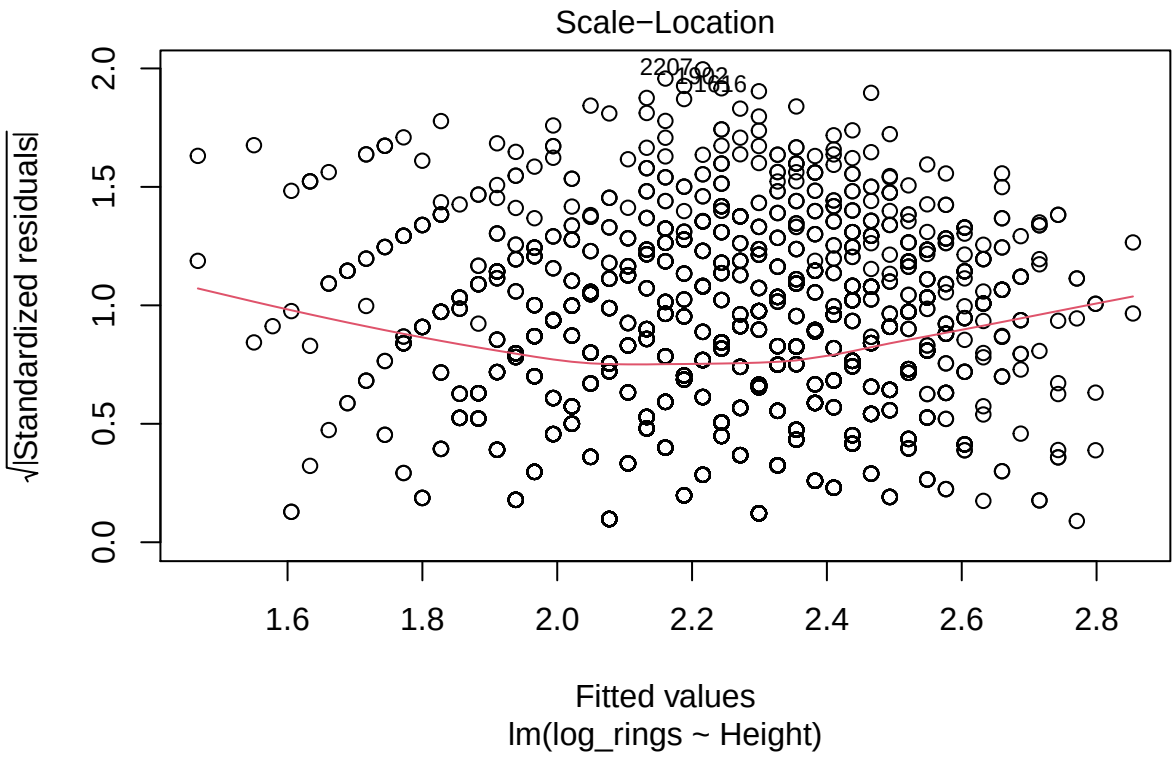
```
##
## Call:
## lm(formula = log_rings ~ Height, data = new_abalone_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72895 -0.15740 -0.02552  0.11799  0.91957
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.46694    0.01598   91.78   <2e-16 ***
## Height       5.54808    0.11041   50.25   <2e-16 ***
## ---
```

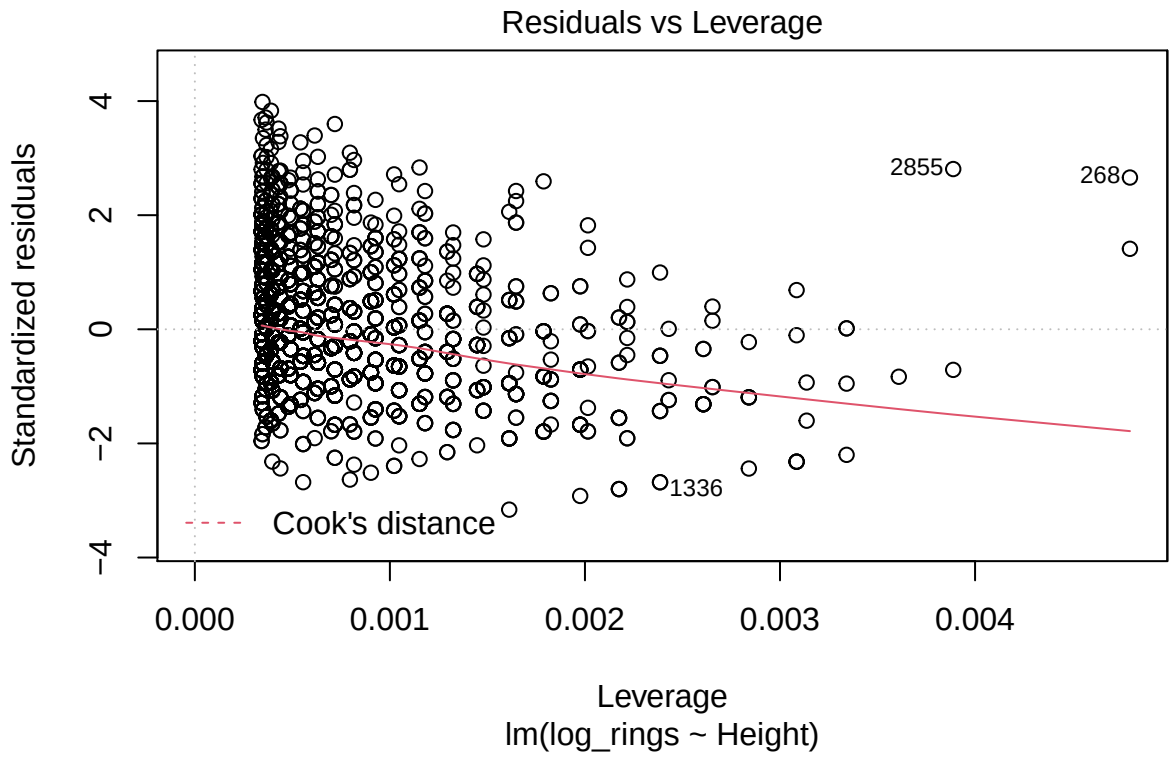
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2308 on 2920 degrees of freedom
## Multiple R-squared:  0.4637, Adjusted R-squared:  0.4635
## F-statistic:  2525 on 1 and 2920 DF,  p-value: < 2.2e-16
```

```
plot(linear_mod_log_simple)
```




Normal Q−Q
Standardized residuals
4
2
0
−2
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(log_rings ~ Height)

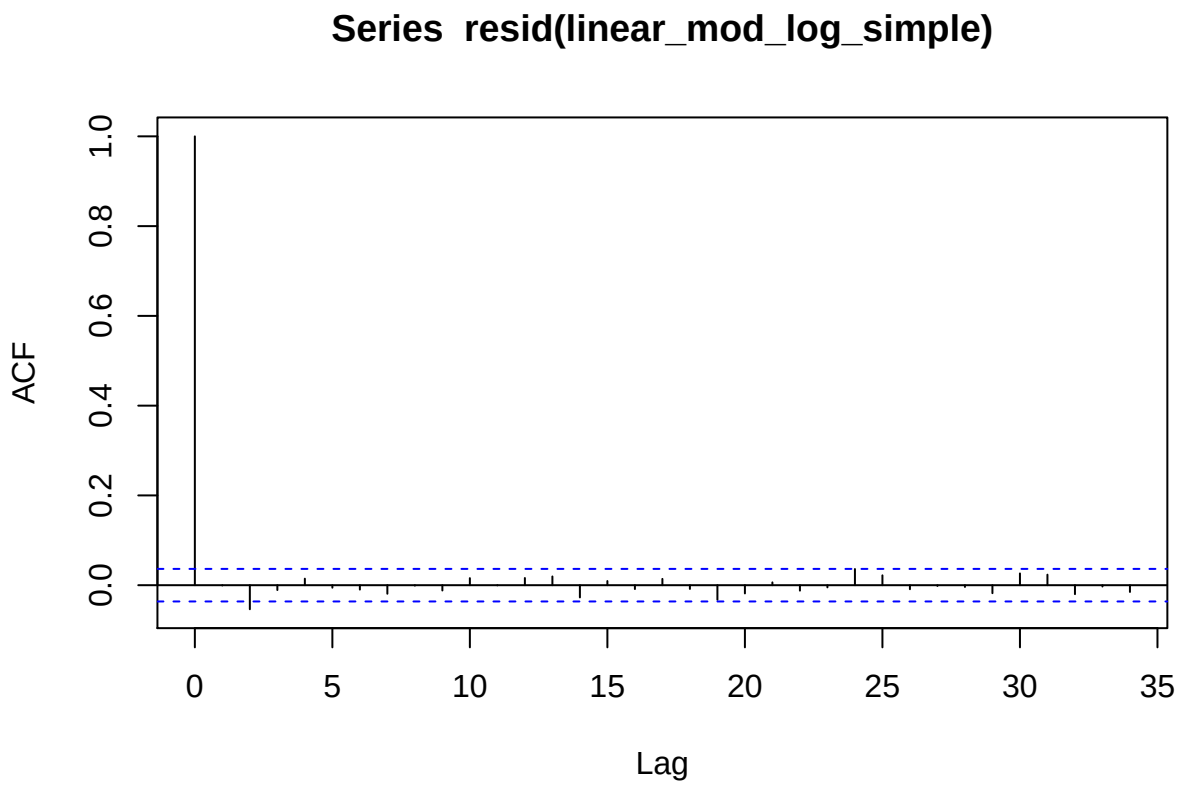

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
1.6
1.8
2.0
2.2
2.4
2.6
2.8
Fitted values
lm(log_rings ~ Height)




```
durbinWatsonTest(linear_mod_log_simple, max.lag=10)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1   -0.0006473558      2.000970   0.988
##    2   -0.0535919713      2.106797   0.000
##    3   -0.0108529869      2.021248   0.562
##    4    0.0145632230      1.970415   0.438
##    5   -0.0056258038      2.010500   0.728
##    6   -0.0098248195      2.018197   0.542
##    7   -0.0190783698      2.036133   0.264
##    8   -0.0010194703      1.999680   0.928
##    9   -0.0118225309      2.019745   0.462
##   10    0.0157142341      1.964656   0.424
##  Alternative hypothesis: rho[lag] != 0
```

```
acf(resid(linear_mod_log_simple))
```




```
bptest(linear_mod_log_simple)
```

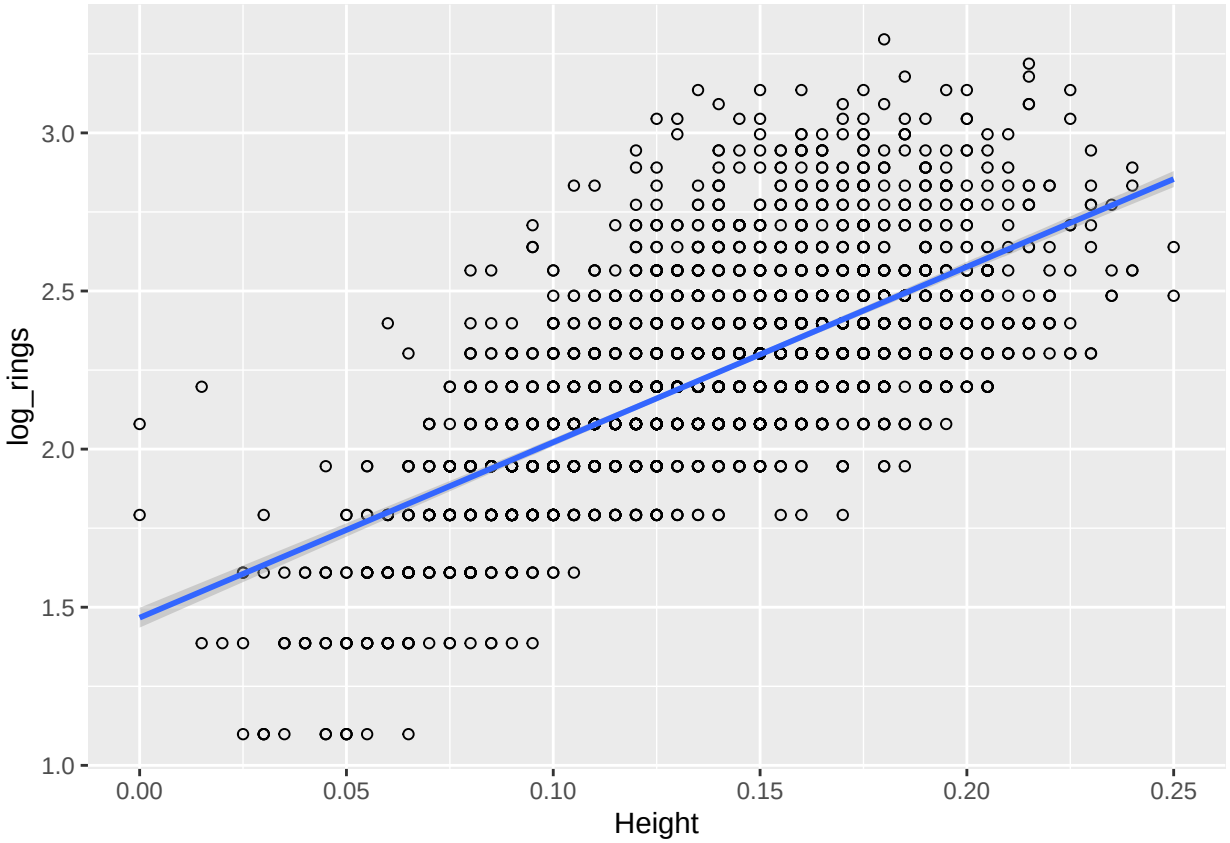
```
##
##  studentized Breusch-Pagan test
##
## data:  linear_mod_log_simple
## BP = 1.1934, df = 1, p-value = 0.2746
```

```
shapiro.test(resid(linear_mod_log_simple))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(linear_mod_log_simple)
## W = 0.97261, p-value < 2.2e-16
```

```
ggplot(new_abalone_train, aes(x=Height, y=log_rings)) + geom_point(shape=1) +geom_smooth(method=lm)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



We decided to use the logarithm of the Rings as the response variable. This appears to better satisfy the postulate of homoskedasticity as seen from the results of the results of the B-P test. It also seems to better satisfy the condition of Gaussian distribution of residuals as we get a better value of the S-W statistic. Lastly, we observe a better fit as seen from the graph.

```
#Q7

confint(linear_mod_log_simple, level=0.95)
```

```
##                2.5 %   97.5 %
## (Intercept) 1.435597 1.498273
## Height      5.331584 5.764575
```

In the context of the problem, these confidence intervals (of the coefficients) means that an additional unit change in Height will change the response variable (number of rings or its logarithm) by a value present in the confidence interval 95% of the times (so with 95% confidence).

```
#Q8
```

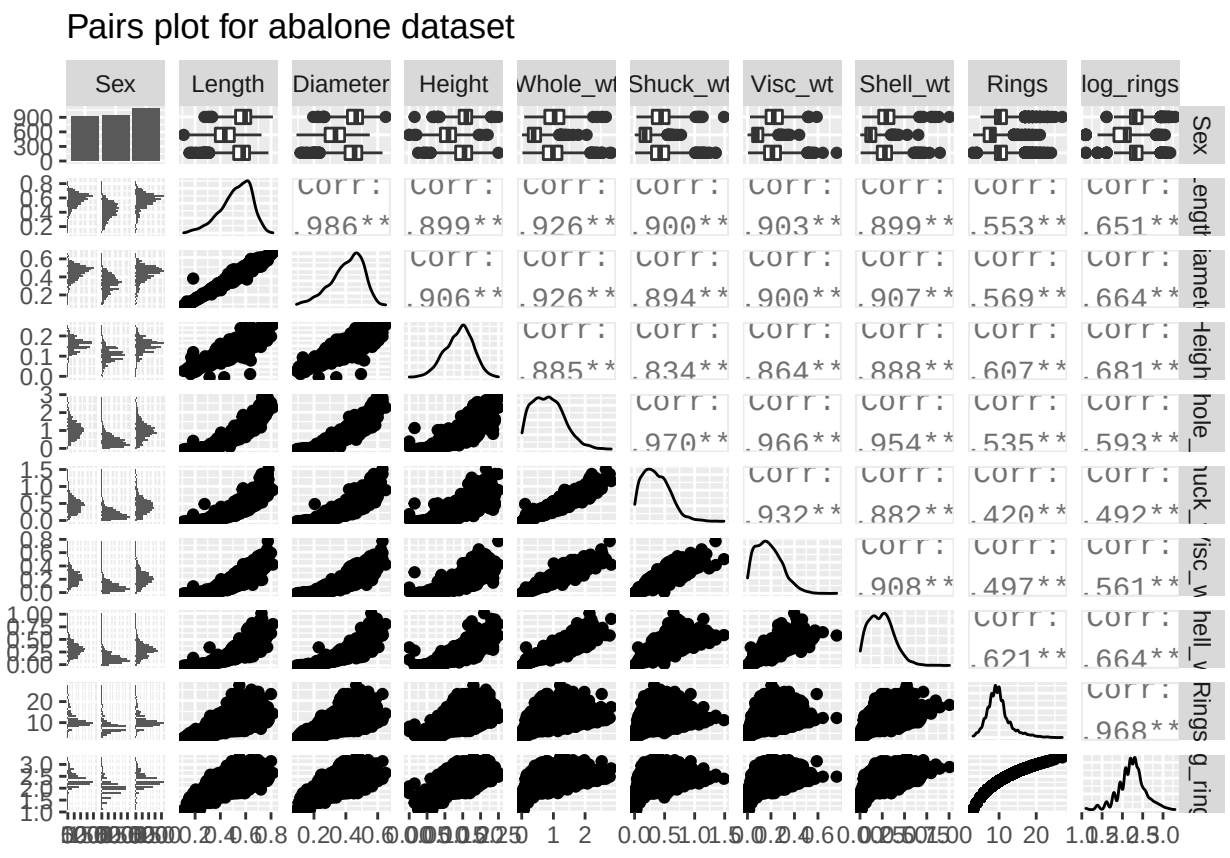
As the p-value is much less that an hypothetical 0.05 alpha, we reject the null hypothesis that $\beta_1 = 0$. Hence, there is a statistically significant relationship between the Height and the number of rings.

==========

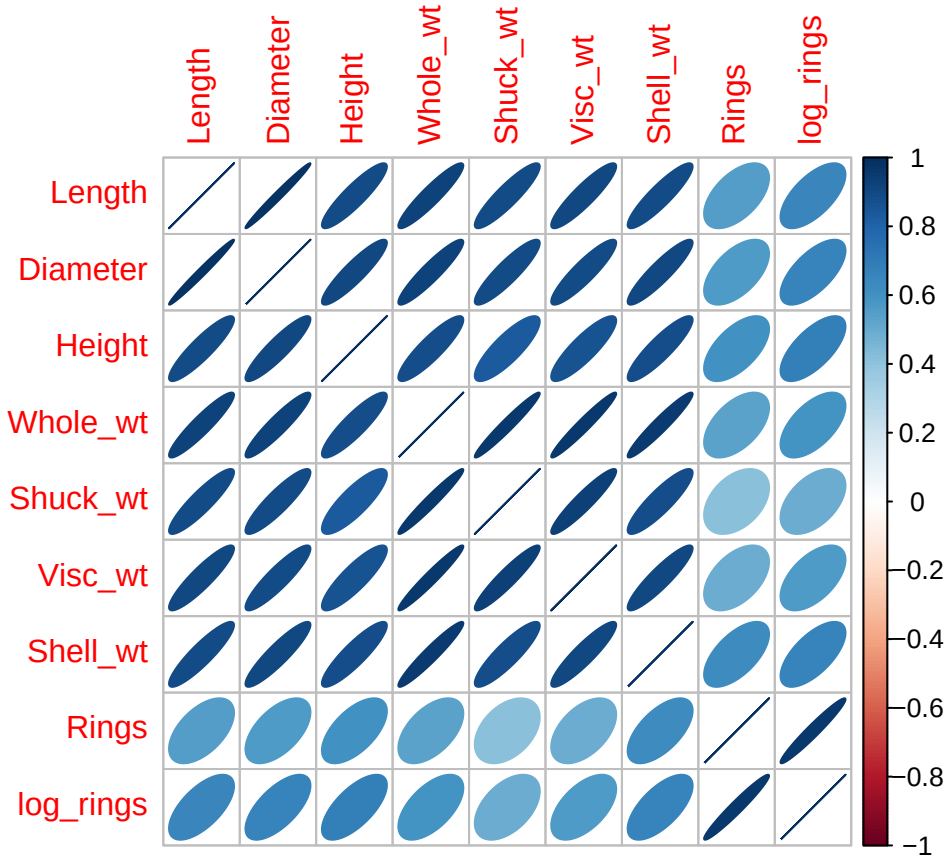
==========

```
#PART 2
#Q9

ggpairs(new_abalone_train, title="Pairs plot for abalone dataset",progress = F) #+ theme_grey(base_size
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
withoutSex = new_abalone_train[-1]
corrplot(cor(withoutSex), method = "ellipse")
```



We decided to keep as dependent variable the logarithm of the number of rings, since we saw that the postulate are more close to being met with this manipulation. First of all, we started fitting a model with all the variables to check those that we need to keep and those that we need to delete. However, we will first remove the point which has a Cook distance greater than 1 (which is the same outlier that we also removed before) and, in addition, we remove also another observation which has a Cook distance between 0.5 an 1. Even if this observation can be left in the model we decided to remove it since it has a big impact on the verification of the postulates. In addition, we need to remove it to have a dataset of a dimension that is the same of the one on which we performed the simple linear model.This last manipulation is necessary in order to perform the Anova and Ancova.

In addition to the Height, the features we have chosen to use in our first model are: Diameter and Whole_wt. As Diameter is highly correlated with Length (but have a better correlation with log_rings) and Whole_wt is highly correlated with the other three weight features.

```
new_abalone_scale = data.frame(rapply(new_abalone_train, scale, c("numeric","integer"), how="replace"))
new_abalone_scale = new_abalone_scale[-c(9)]
head(new_abalone_scale)
```

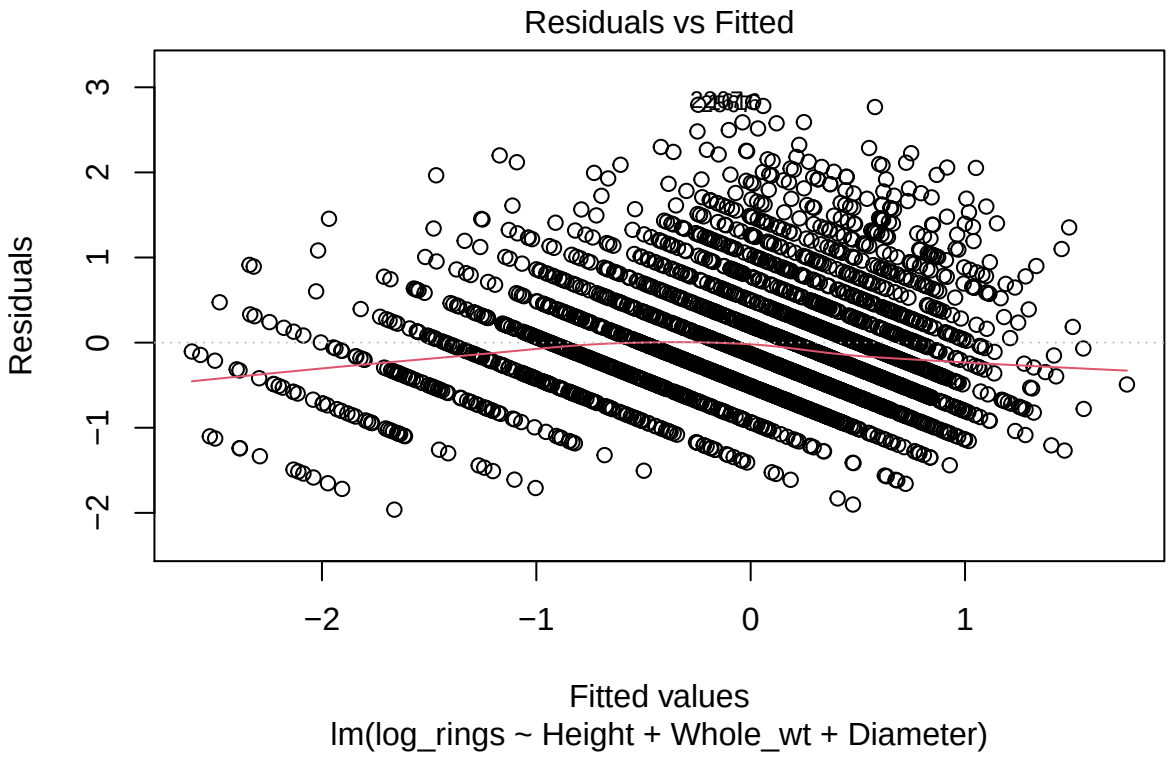
```
##   Sex     Length   Diameter      Height    Whole_wt    Shuck_wt    Visc_wt
## 1   F  0.2945845  0.2205879  0.27175029  0.08601802  0.41481788 -0.1854874
## 2   I -1.5308510 -1.5960627 -1.79643622 -1.23627747 -1.15112464 -1.2582231
## 3   M -0.0373129  0.1701253 -0.37455799 -0.08535310  0.05137563  0.2128280
## 4   F  0.7924305  0.9270631  0.27175029  0.93476136  1.05645048  1.1950375
## 5   F  0.7094561  0.6747505  0.01322698  0.61027167  0.74909501  0.4889330
## 6   F  1.2902765  1.5326133  1.43510521  1.68717183  0.98465942  1.8468264
##     Shell_wt  log_rings
## 1 -0.1400305 -0.5120953
```

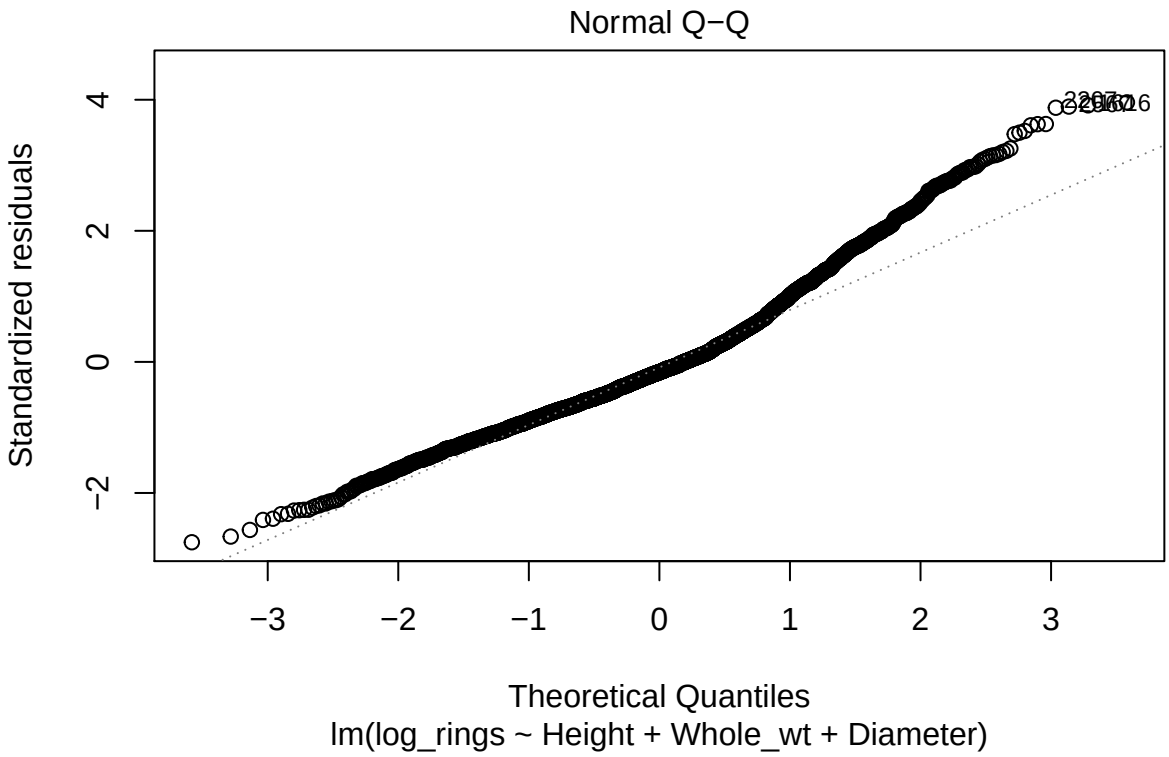
```
## 2 -1.3273946 -0.9358040
## 3 -0.3559149 -0.5120953
## 4  0.8242530  0.1959619
## 5  0.7954684  0.4983908
## 6  2.1987169  0.4983908
```

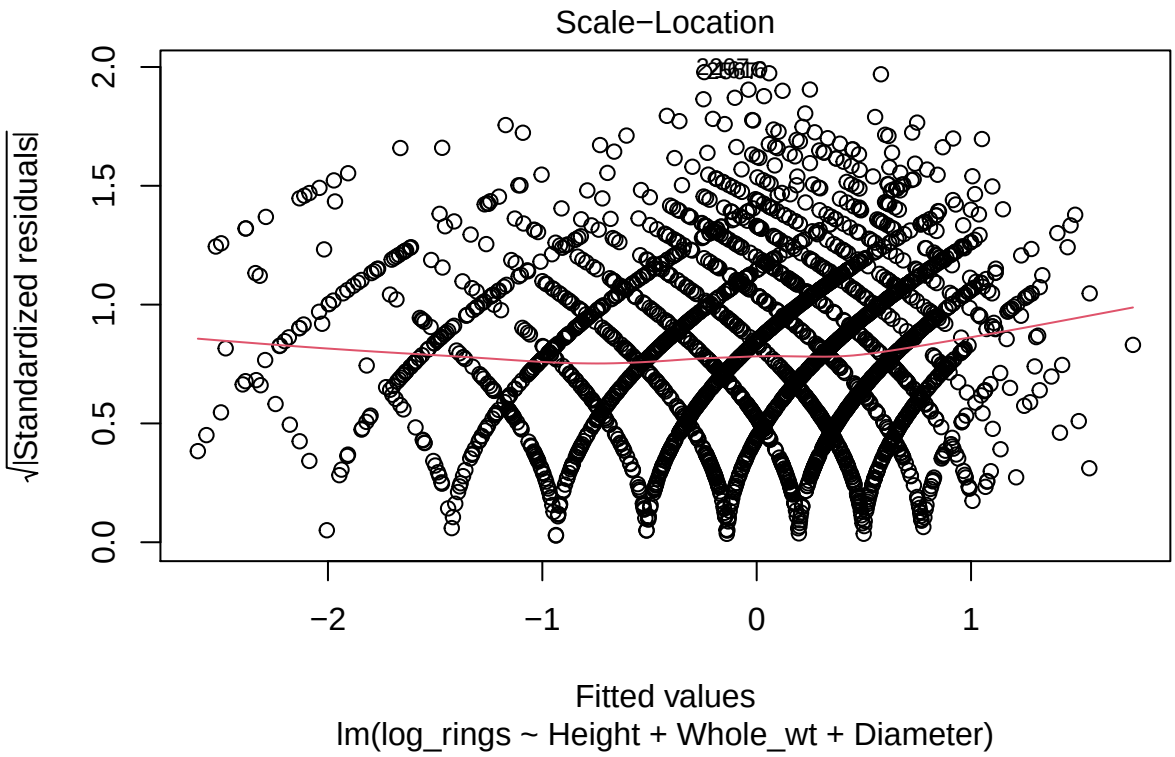
```
linear_mod_log = lm(log_rings ~ Height + Whole_wt + Diameter, data=new_abalone_scale)
summary(linear_mod_log)
```

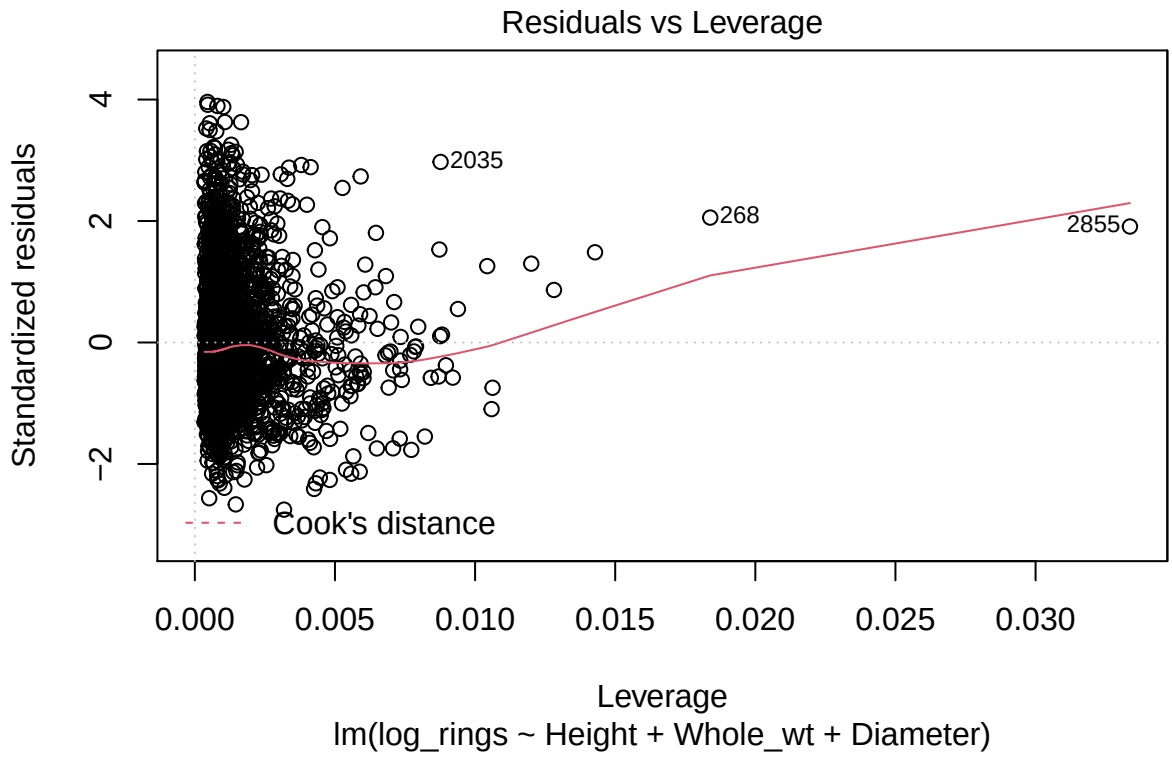
```
##
## Call:
## lm(formula = log_rings ~ Height + Whole_wt + Diameter, data = new_abalone_scale)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9621 -0.4831 -0.1116  0.3616  2.8275
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.045e-15  1.321e-02   0.000        1
## Height       5.265e-01  3.260e-02  16.152   <2e-16 ***
## Whole_wt    -3.307e-01  3.662e-02  -9.029   <2e-16 ***
## Diameter     4.938e-01  4.017e-02  12.293   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7141 on 2918 degrees of freedom
## Multiple R-squared:  0.4906, Adjusted R-squared:  0.4901
## F-statistic: 936.7 on 3 and 2918 DF,  p-value: < 2.2e-16
```

```
plot(linear_mod_log)
```


Residuals vs Fitted
Residuals
3
2
1
0
−1
−2
−2
−1
0
1
Fitted values
lm(log_rings ~ Height + Whole_wt + Diameter)


Normal Q−Q
Standardized residuals
4
2
0
−2
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(log_rings ~ Height + Whole_wt + Diameter)

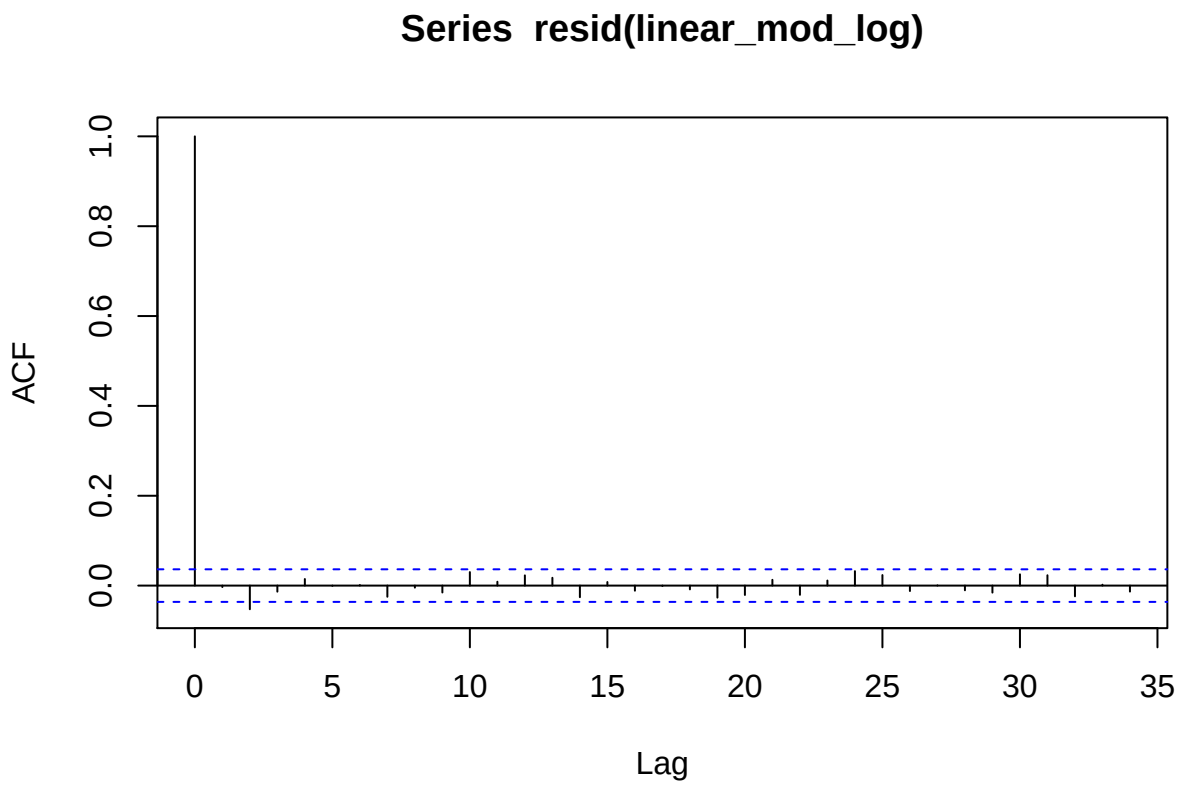

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
−2
−1
0
1
Fitted values
lm(log_rings ~ Height + Whole_wt + Diameter)




```
durbinWatsonTest(linear_mod_log, max.lag=10)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1   -0.0032319453      2.006100   0.816
##    2   -0.0526269409      2.104788   0.002
##    3   -0.0135309541      2.026467   0.484
##    4    0.0146476050      1.970091   0.434
##    5   -0.0007252133      2.000550   0.858
##    6    0.0013253677      1.995713   0.998
##    7   -0.0248565370      2.047635   0.156
##    8   -0.0045395589      2.006614   0.814
##    9   -0.0153453322      2.026734   0.368
##   10    0.0301266666      1.935710   0.114
##  Alternative hypothesis: rho[lag] != 0
```

```
acf(resid(linear_mod_log))
```




```
bptest(linear_mod_log)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  linear_mod_log
## BP = 23.295, df = 3, p-value = 3.505e-05
```

```
shapiro.test(resid(linear_mod_log))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(linear_mod_log)
## W = 0.96208, p-value < 2.2e-16
```

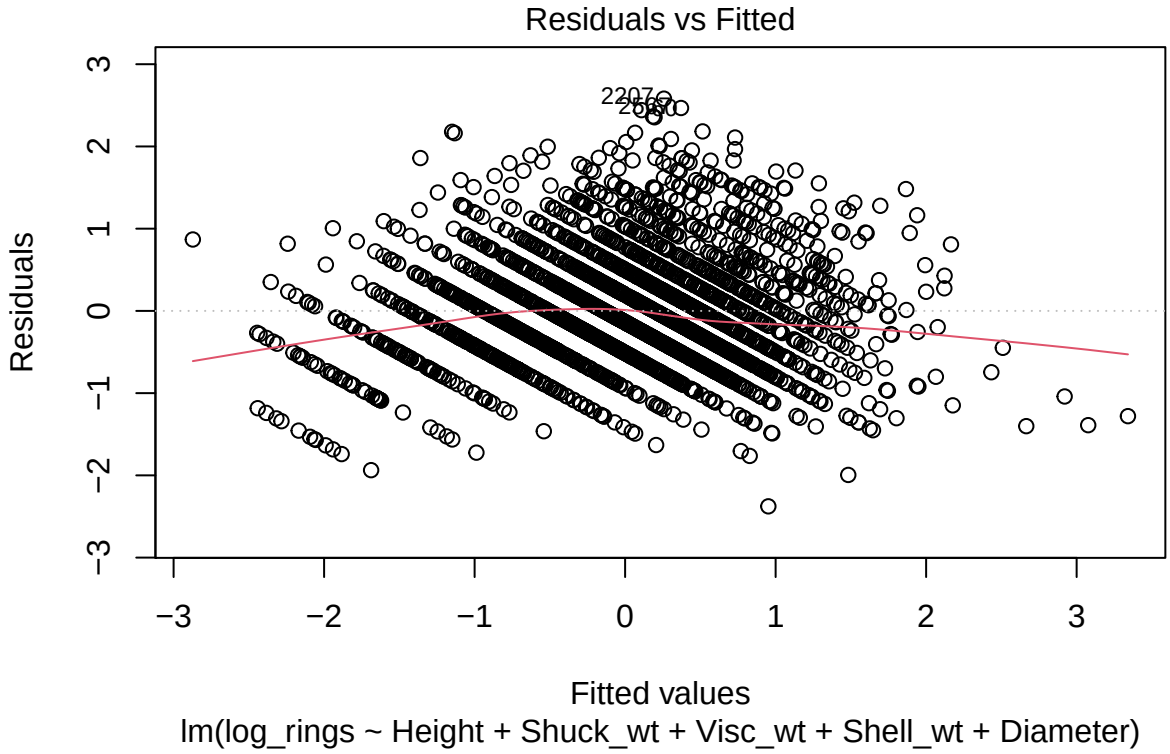
This models meets the validity postulates. However, we try another model where we replace Whole_wt by Shuck_wt , Visc_wt, and Shell_wt to see if this model better explains the number of rings and to see if one of these features has more impact than the other ones.

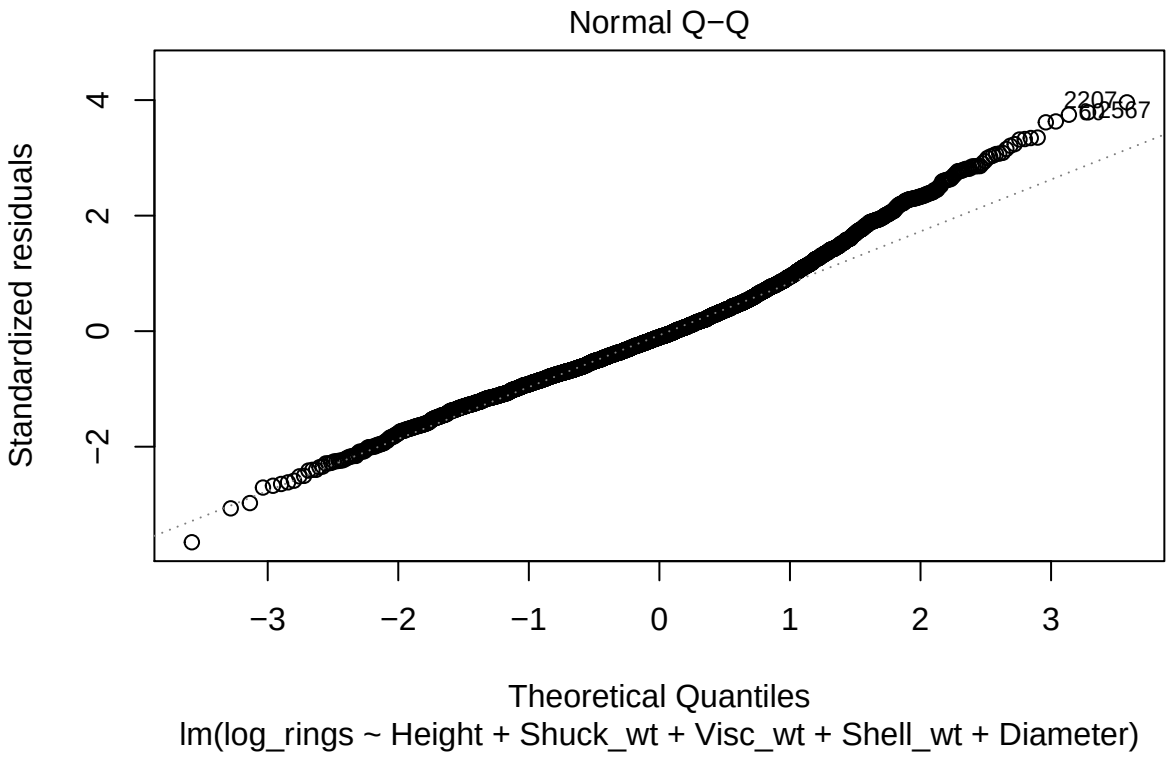
```
linear_mod_log = lm(log_rings ~ Height + Shuck_wt + Visc_wt + Shell_wt + Diameter, data=new_abalone_scal
summary(linear_mod_log)
```

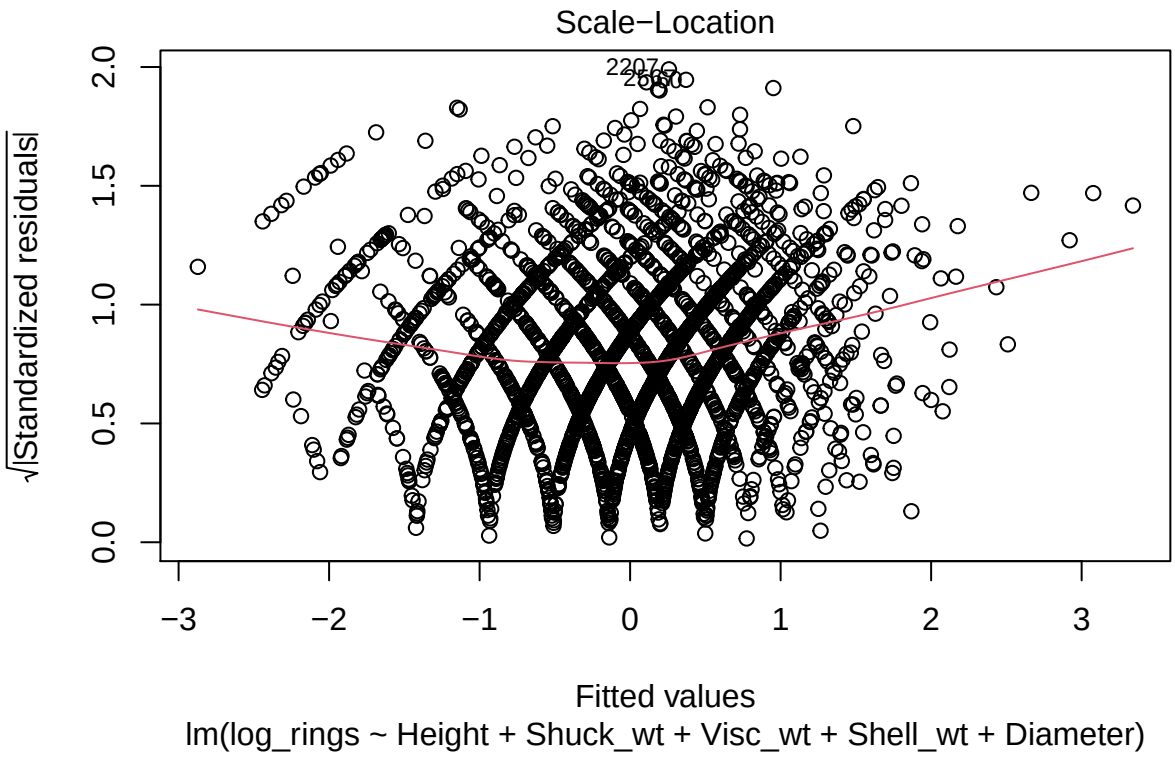
```
##
```

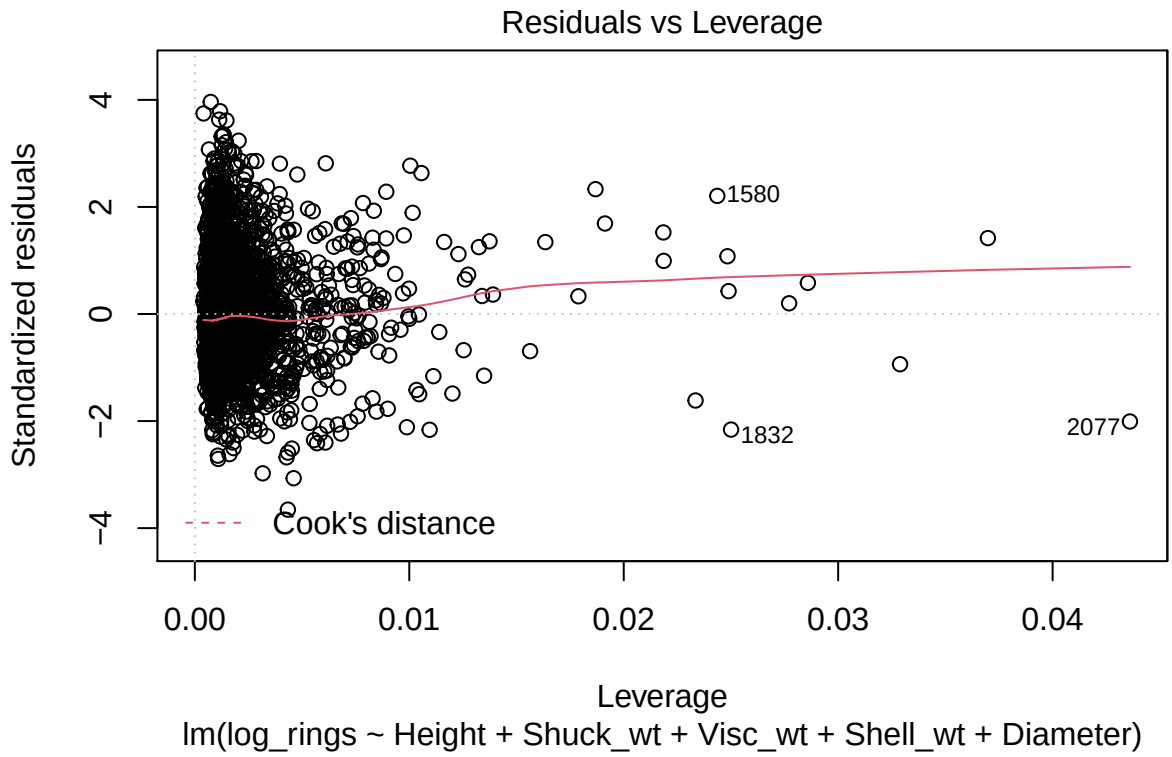
```
## Call:
## lm(formula = log_rings ~ Height + Shuck_wt + Visc_wt + Shell_wt +
##     Diameter, data = new_abalone_scale)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.37657 -0.43956 -0.06653  0.34915  2.58090
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.070e-15  1.206e-02   0.000   1.0000
## Height       3.602e-01  3.093e-02  11.648   <2e-16 ***
## Shuck_wt    -6.957e-01  3.584e-02 -19.412   <2e-16 ***
## Visc_wt     -7.075e-02  3.947e-02  -1.793   0.0731 .
## Shell_wt     5.237e-01  3.468e-02  15.101   <2e-16 ***
## Diameter     5.490e-01  3.719e-02  14.763   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6518 on 2916 degrees of freedom
## Multiple R-squared:  0.5759, Adjusted R-squared:  0.5751
## F-statistic: 791.8 on 5 and 2916 DF,  p-value: < 2.2e-16
```

```
plot(linear_mod_log)
```




Normal Q−Q
Standardized residuals
4
2
0
−2
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(log_rings ~ Height + Shuck_wt + Visc_wt + Shell_wt + Diameter)

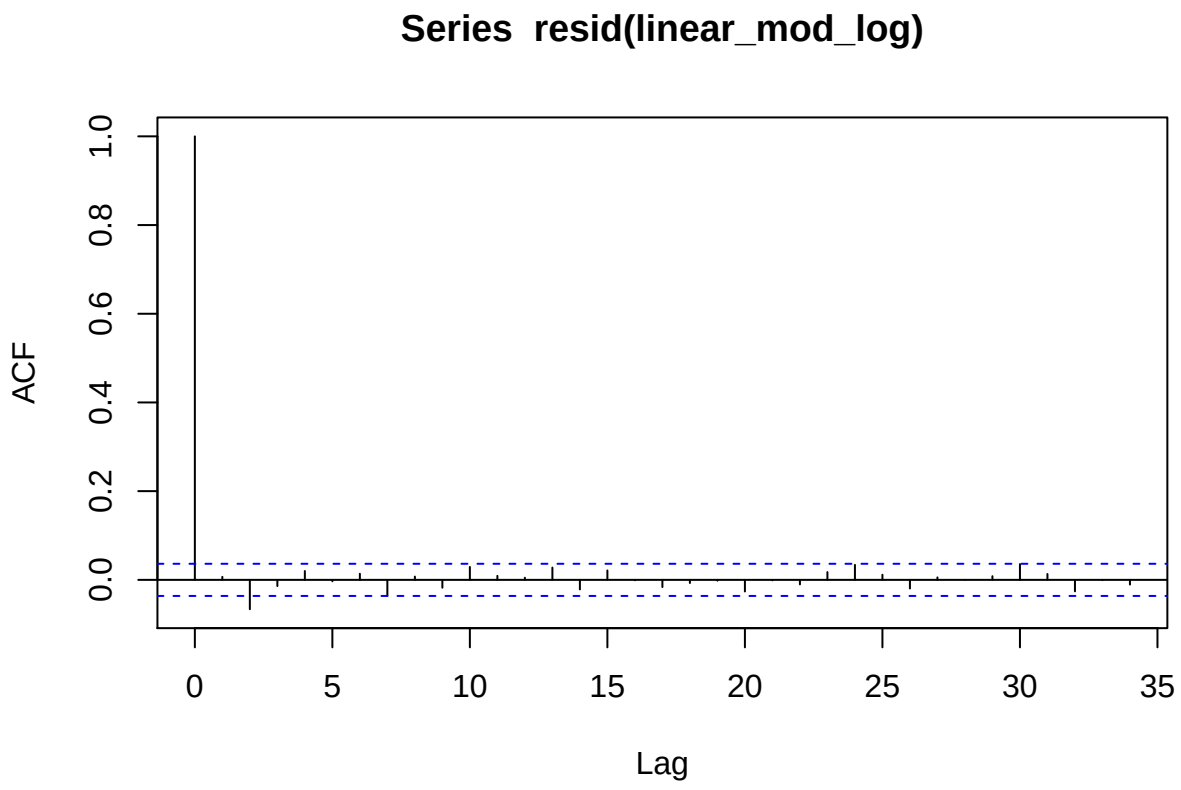

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
−3
−2
−1
0
1
2
3
Fitted values
lm(log_rings ~ Height + Shuck_wt + Visc_wt + Shell_wt + Diameter)




```
durbinWatsonTest(linear_mod_log, max.lag=10)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1     0.006844693      1.986193   0.650
##    2    -0.066253624      2.132264   0.000
##    3    -0.013935093      2.027580   0.440
##    4     0.019936761      1.959796   0.298
##    5    -0.003112105      2.005638   0.850
##    6     0.013838175      1.970100   0.462
##    7    -0.036076597      2.069735   0.070
##    8     0.007440448      1.982378   0.778
##    9    -0.017677624      2.031188   0.312
##   10     0.029154254      1.937136   0.136
##  Alternative hypothesis: rho[lag] != 0
```

```
acf(resid(linear_mod_log))
```




```
bptest(linear_mod_log)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  linear_mod_log
## BP = 103.23, df = 5, p-value < 2.2e-16
```

```
shapiro.test(resid(linear_mod_log))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(linear_mod_log)
## W = 0.98076, p-value < 2.2e-16
```

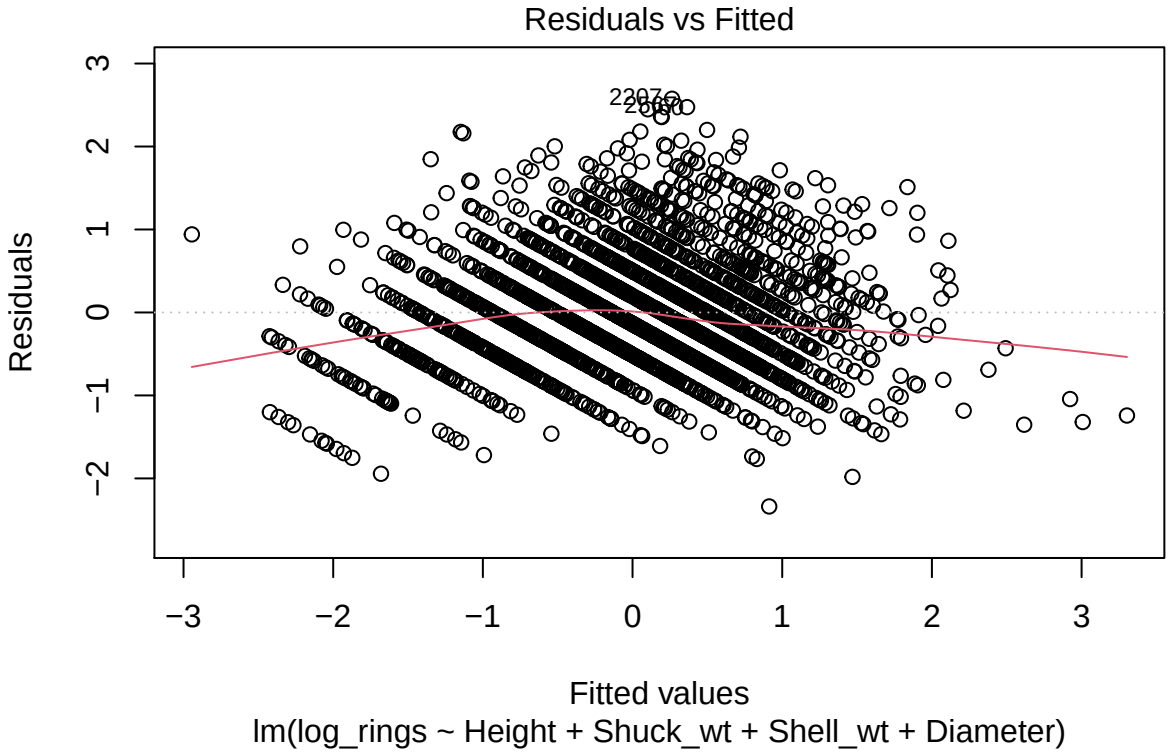
This models meets the validity postulates. We can reject the null hypothesis for Visc_wt as its p_value= $0.0731 > 0.05$, so we make a new model without it.

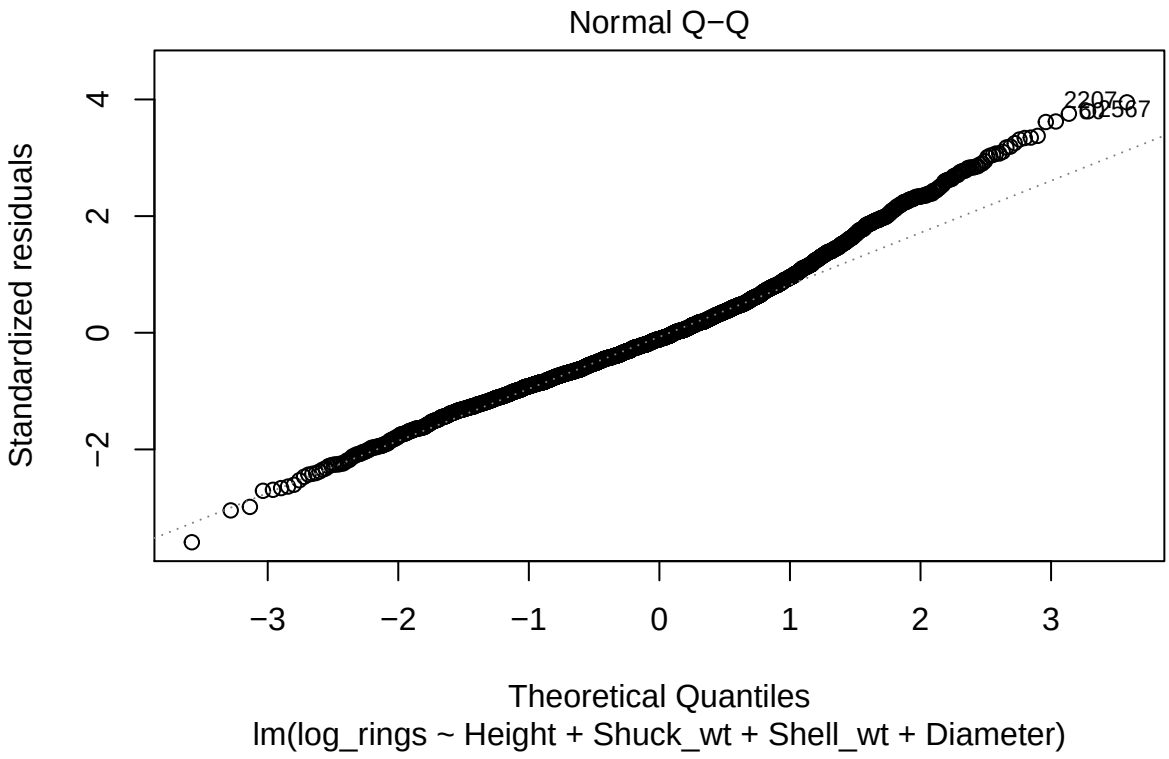
```
linear_mod_log = lm(log_rings ~ Height + Shuck_wt + Shell_wt + Diameter, data=new_abalone_scale)
summary(linear_mod_log)
```

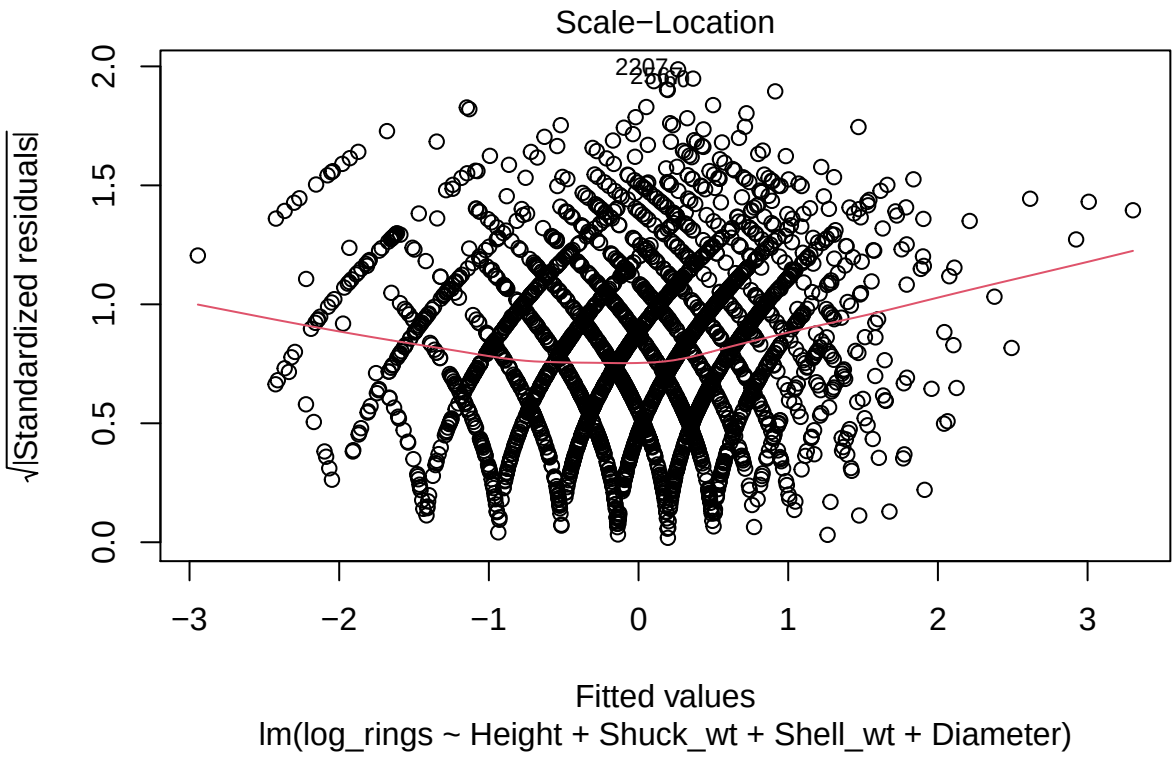
```
##
```

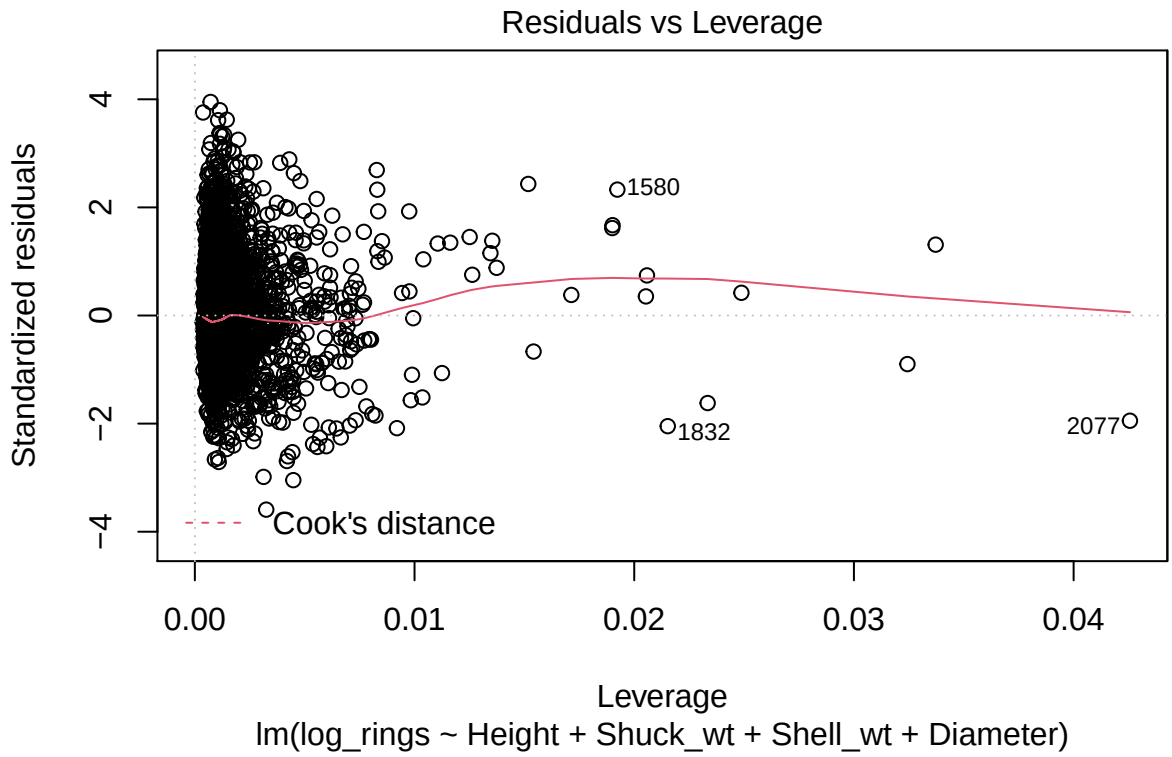
```
## Call:
## lm(formula = log_rings ~ Height + Shuck_wt + Shell_wt + Diameter,
##     data = new_abalone_scale)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.33799 -0.43744 -0.07189  0.34680  2.57505
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.072e-15  1.206e-02    0.00        1
## Height       3.530e-01  3.067e-02   11.51   <2e-16 ***
## Shuck_wt    -7.332e-01  2.911e-02  -25.19   <2e-16 ***
## Shell_wt     5.047e-01  3.303e-02   15.28   <2e-16 ***
## Diameter     5.428e-01  3.704e-02   14.65   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6521 on 2917 degrees of freedom
## Multiple R-squared:  0.5754, Adjusted R-squared:  0.5748
## F-statistic: 988.2 on 4 and 2917 DF,  p-value: < 2.2e-16
```

```
plot(linear_mod_log)
```




Normal Q−Q
Standardized residuals
4
2
0
−2
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(log_rings ~ Height + Shuck_wt + Shell_wt + Diameter)

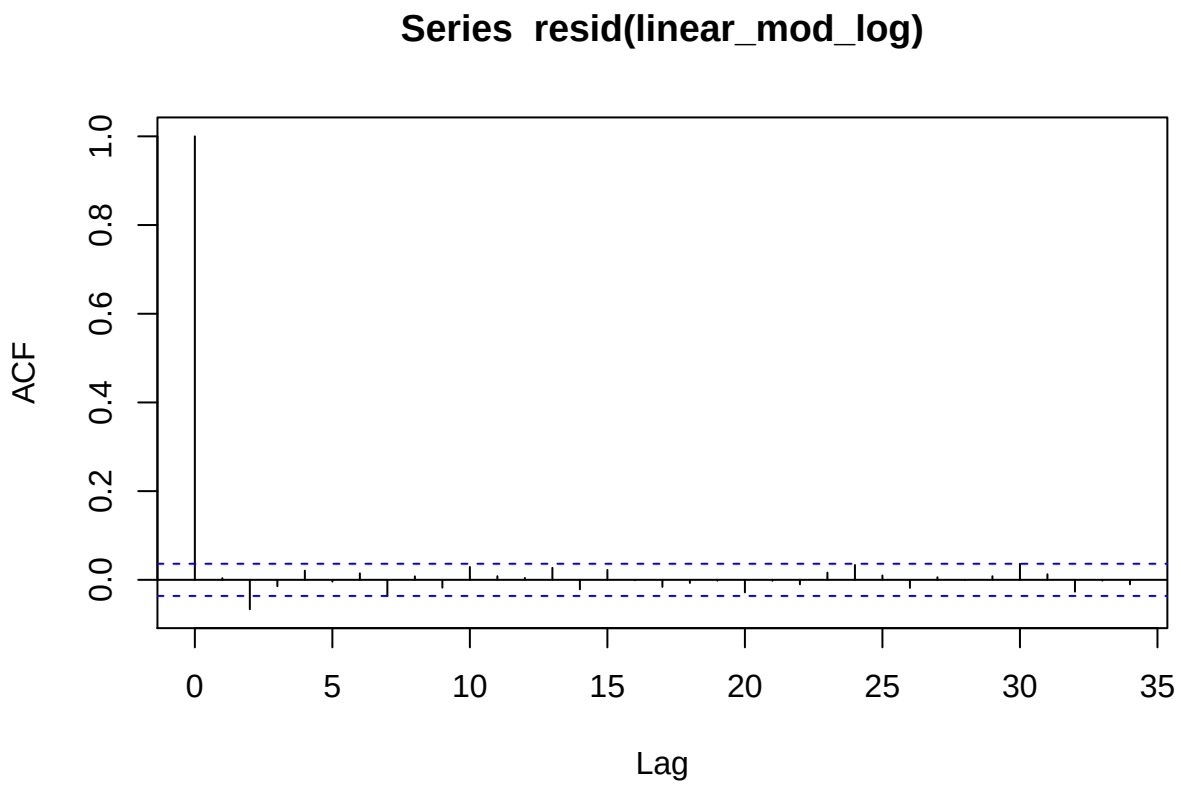

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
−3
−2
−1
0
1
2
3
Fitted values
lm(log_rings ~ Height + Shuck_wt + Shell_wt + Diameter)




```
durbinWatsonTest(linear_mod_log, max.lag=10)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1     0.003739155      1.992421   0.850
##    2    -0.066297859      2.132372   0.002
##    3    -0.014304187      2.028328   0.456
##    4     0.020684120      1.958314   0.280
##    5    -0.003964743      2.007349   0.770
##    6     0.014673944      1.968389   0.418
##    7    -0.036342435      2.070226   0.046
##    8     0.007863753      1.981503   0.724
##    9    -0.017563319      2.030944   0.360
##   10     0.029114534      1.937249   0.134
##  Alternative hypothesis: rho[lag] != 0
```

```
acf(resid(linear_mod_log))
```




```
bptest(linear_mod_log)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  linear_mod_log
## BP = 102.16, df = 4, p-value < 2.2e-16
```

```
shapiro.test(resid(linear_mod_log))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(linear_mod_log)
## W = 0.98077, p-value < 2.2e-16
```

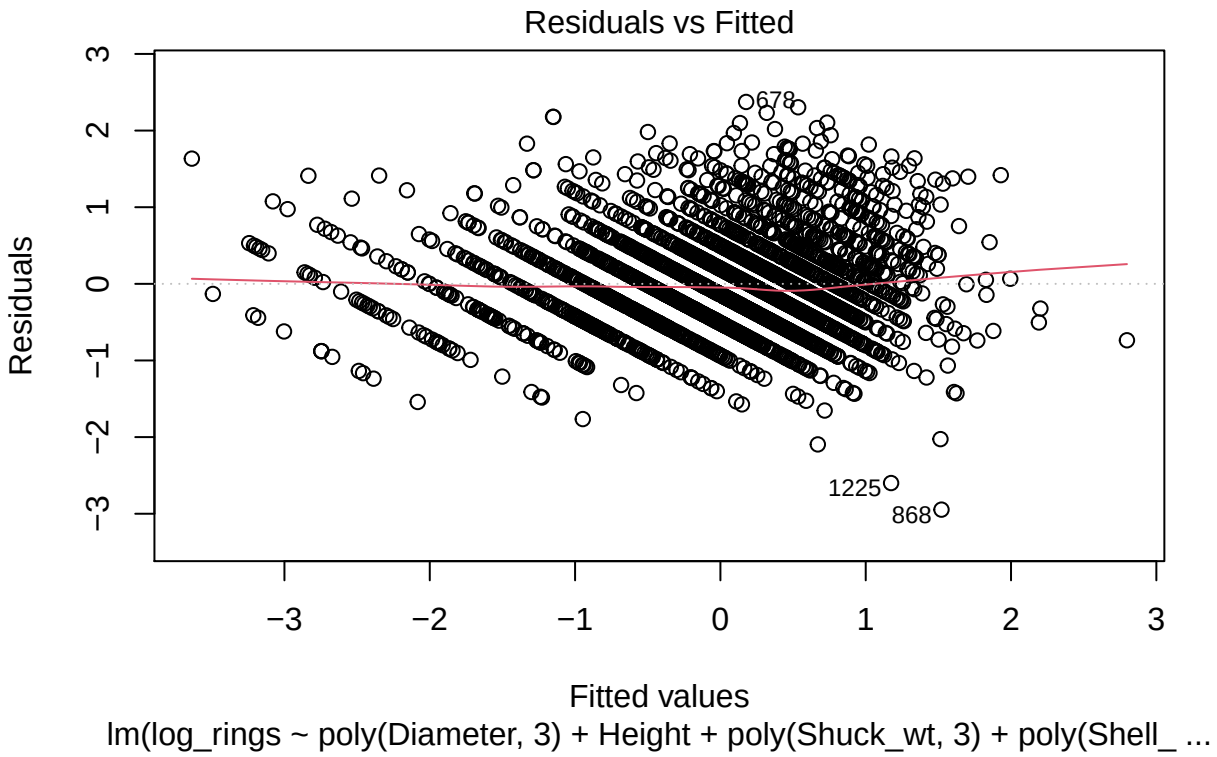
This models meets the validity postulates. The Adjusted R-squared of this model is $0.5748 > 0.4901$ (Adjusted R-squared of the first model).

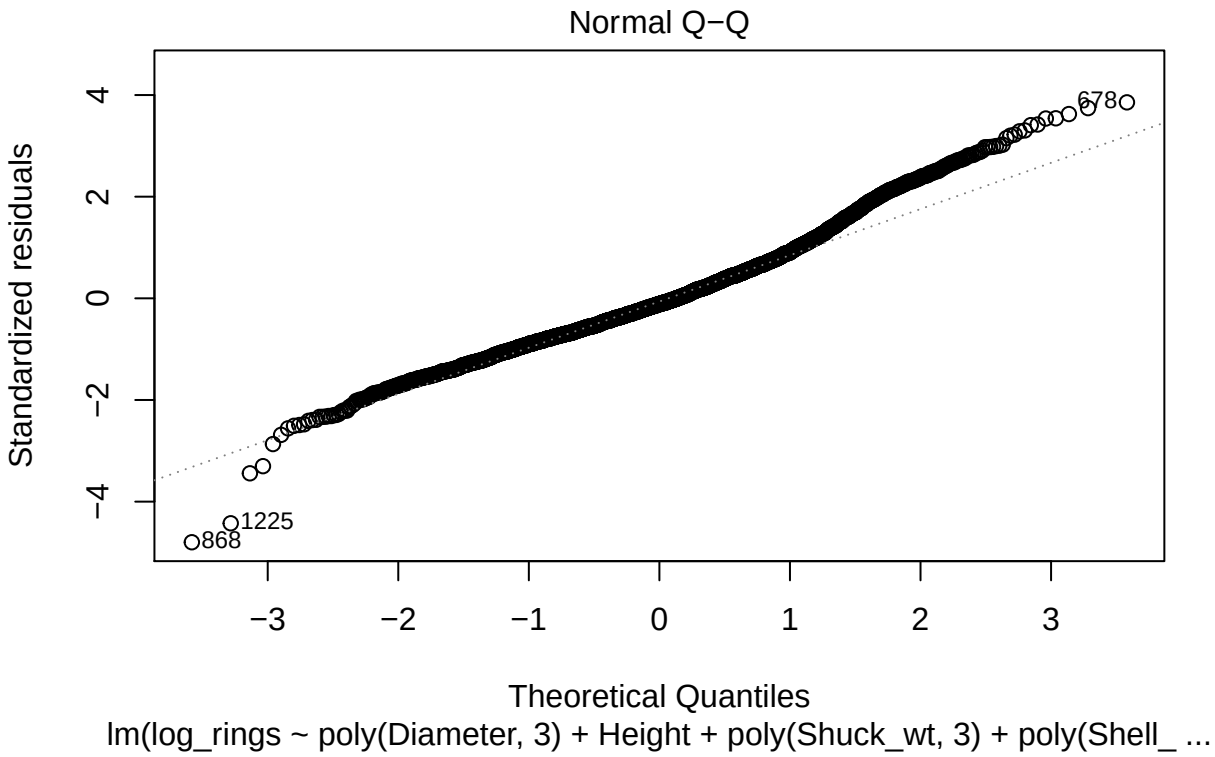
After many tests we decided to implement a model with polynomial features

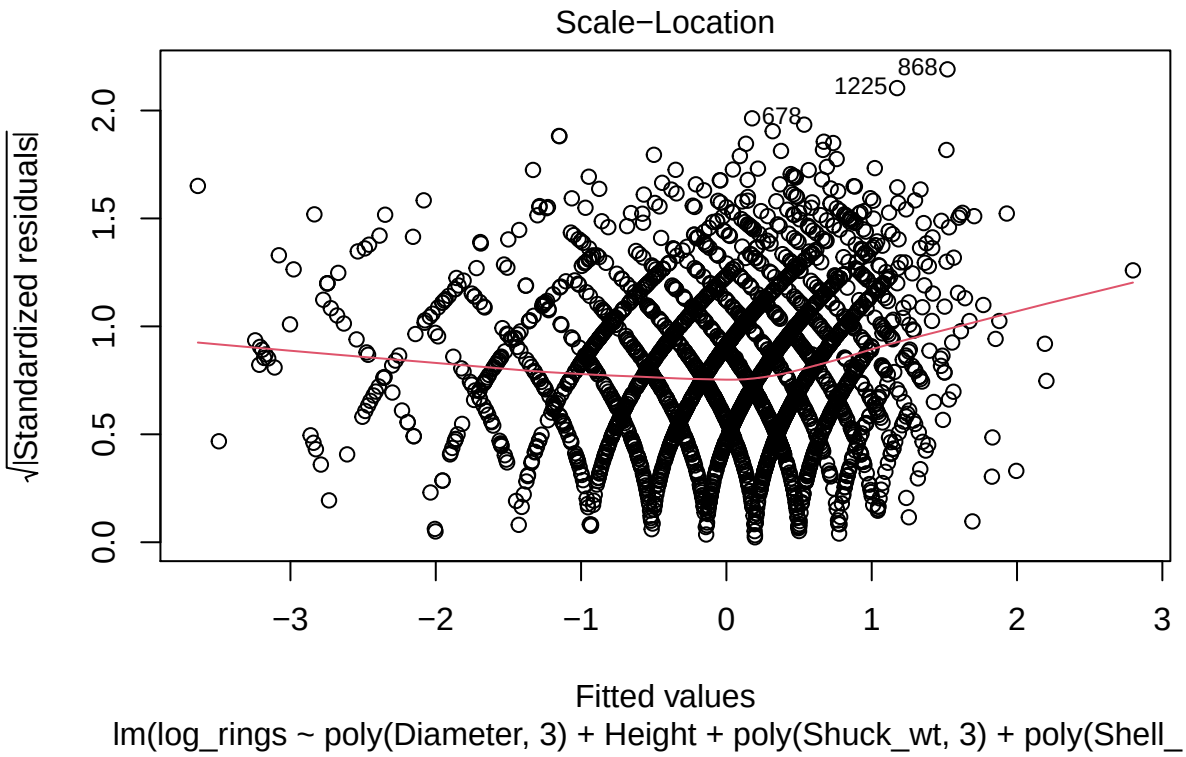
```
linear_mod_log = lm(log_rings ~ poly(Diameter, 3) + Height + poly(Shuck_wt, 3) + poly(Shell_wt, 3), data
summary(linear_mod_log)
```

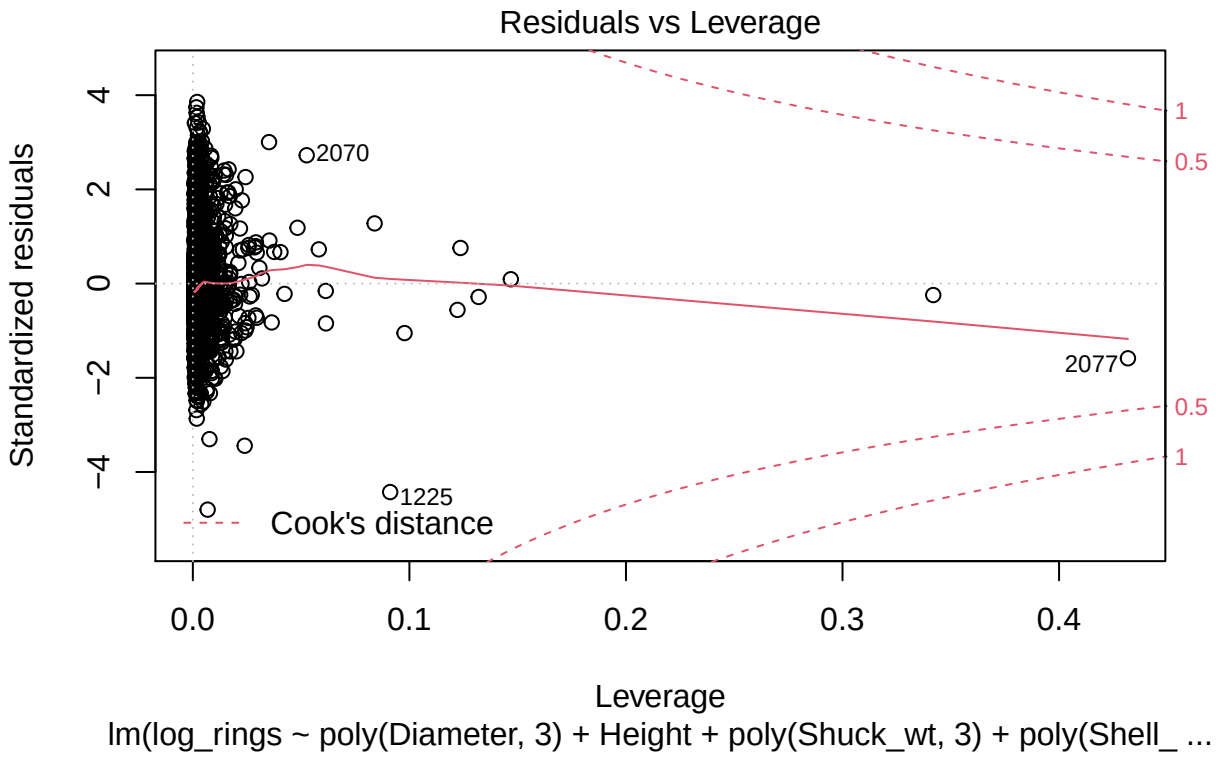
```
##
## Call:
## lm(formula = log_rings ~ poly(Diameter, 3) + Height + poly(Shuck_wt,
##     3) + poly(Shell_wt, 3), data = new_abalone_scale)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.94633 -0.41542 -0.06899  0.33964  2.37325
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.228e-15  1.140e-02   0.000  1.00000
## poly(Diameter, 3)1  7.491e+00  3.270e+00   2.291  0.02203 *
## poly(Diameter, 3)2 -8.800e+00  1.470e+00  -5.986 2.41e-09 ***
## poly(Diameter, 3)3  3.783e+00  9.564e-01   3.956 7.82e-05 ***
## Height               2.192e-01  3.032e-02   7.229 6.17e-13 ***
## poly(Shuck_wt, 3)1 -3.984e+01  2.162e+00 -18.431  < 2e-16 ***
## poly(Shuck_wt, 3)2  1.133e+01  1.429e+00   7.934 2.99e-15 ***
## poly(Shuck_wt, 3)3 -2.395e+00  9.179e-01  -2.609  0.00912 **
## poly(Shell_wt, 3)1  5.729e+01  2.616e+00  21.897  < 2e-16 ***
## poly(Shell_wt, 3)2 -1.545e+01  1.594e+00  -9.696  < 2e-16 ***
## poly(Shell_wt, 3)3  4.873e+00  1.050e+00   4.642 3.60e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.616 on 2911 degrees of freedom
## Multiple R-squared:  0.6218, Adjusted R-squared:  0.6205
## F-statistic: 478.6 on 10 and 2911 DF,  p-value: < 2.2e-16
```

```
plot(linear_mod_log)
```


Residuals vs Fitted
Residuals
Fitted values
lm(log_rings ~ poly(Diameter, 3) + Height + poly(Shuck_wt, 3) + poly(Shell_ ...
678
1225
868
−3
−2
−1
0
1
2
3


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(log_rings ~ poly(Diameter, 3) + Height + poly(Shuck_wt, 3) + poly(Shell_ ...
678
1225
868
4
2
0
−2
−4
−3
−2
−1
0
1
2
3

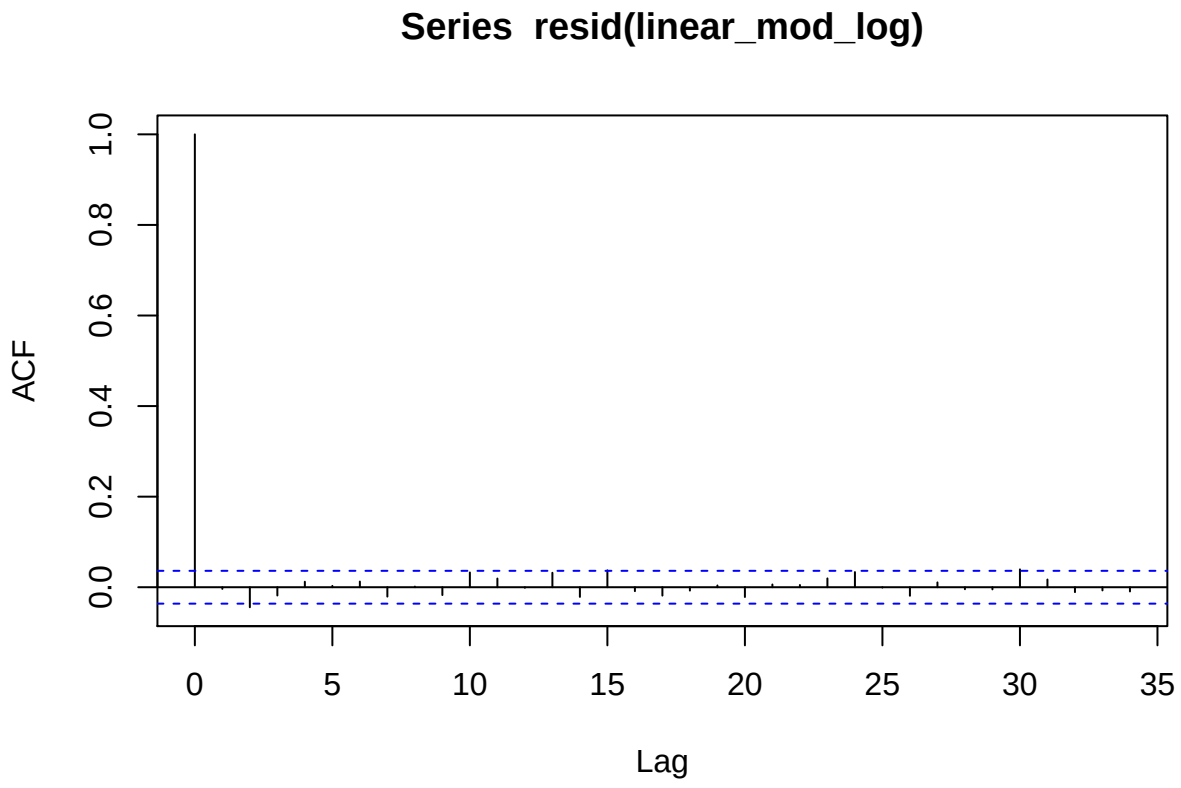

Scale−Location
868
1225
678
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
−3
−2
−1
0
1
2
3
Fitted values
lm(log_rings ~ poly(Diameter, 3) + Height + poly(Shuck_wt, 3) + poly(Shell_ ...




```
durbinWatsonTest(linear_mod_log, max.lag=10)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1    -0.003698396      2.007325   0.840
##    2    -0.044171382      2.088112   0.018
##    3    -0.018287963      2.036323   0.302
##    4     0.012138405      1.975372   0.538
##    5     0.002969778      1.993597   0.918
##    6     0.012567041      1.973618   0.562
##    7    -0.020722931      2.040007   0.212
##    8     0.001420639      1.995576   0.954
##    9    -0.017264590      2.031818   0.342
##   10     0.032638787      1.931525   0.102
##  Alternative hypothesis: rho[lag] != 0
```

```
acf(resid(linear_mod_log))
```




```
bptest(linear_mod_log)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  linear_mod_log
## BP = 131.63, df = 10, p-value < 2.2e-16
```

```
shapiro.test(resid(linear_mod_log))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(linear_mod_log)
## W = 0.97782, p-value < 2.2e-16
```

```
#Q10
```

To check which variables have a significant impact on the target variable (number of rings), we look at the p-values of the t-tests. If the p-value of the single variable is low it means that at a specific significance level the variable has a significant impact on the number of rings. In our final model, we observe that all the covariates are significant at a significance level of 0.05.

Through the use of polynomial features we slightly improved all the postulates. The errors are centered, normal, and less correlated than before and the model is homokedastic.

```r
#Q11

linear_mod_log_simple = lm(log_rings ~ Height, data=new_abalone_scale)
anova(linear_mod_log_simple, linear_mod_log)
```

```
## Analysis of Variance Table
##
## Model 1: log_rings ~ Height
## Model 2: log_rings ~ poly(Diameter, 3) + Height + poly(Shuck_wt, 3) +
##     poly(Shell_wt, 3)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   2920 1566.5
## 2   2911 1104.8  9    461.73 135.18 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

we can see that it has a really small p-value and so the removal of the other other covariates results in a worse model. We are testing that all the variables, apart from Height, are not correlated with log_rings. Since the p-value is very low we reject this hypothesis, and conclude that increasing the number of covariates improved our model.

We can also prove that our last model is better using mean of the squared residuals.

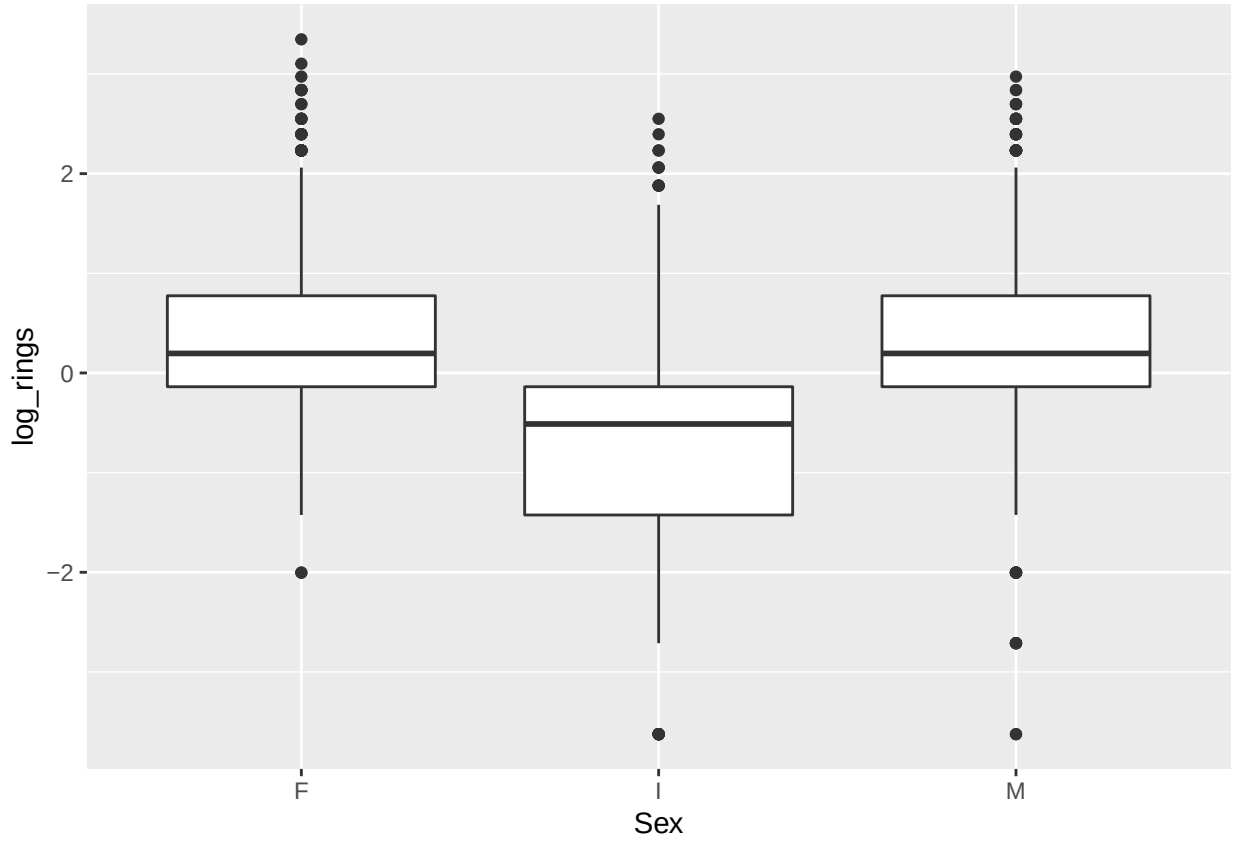
```r
mean(linear_mod_log$residuals^2)
```

```
## [1] 0.3780799
```

```r
mean(linear_mod_log_simple$residuals^2)
```

```
## [1] 0.5360985
```

```r
#Q12

ggplot(new_abalone_scale, aes(x=Sex, y=log_rings)) + geom_boxplot()
```




```
new_abalone_scale['Infant'] = ifelse(new_abalone_scale$Sex == 'I','I','NI')
new_abalone_scale$Infant = as.factor(new_abalone_scale$Infant)
new_abalone_scale = new_abalone_scale[-c(1)]
linear_mod_log_sex = lm(log_rings ~ poly(Diameter, 3) + Height + poly(Shuck_wt, 3) + poly(Shell_wt, 3) +

anova(linear_mod_log_sex)
```

```
## Analysis of Variance Table
##
## Response: log_rings
##                     Df  Sum Sq Mean Sq  F value    Pr(>F)
## poly(Diameter, 3)    3 1387.76  462.59 1248.986 < 2.2e-16 ***
## Height               1  108.96  108.96  294.202 < 2.2e-16 ***
## poly(Shuck_wt, 3)    3  122.19   40.73  109.971 < 2.2e-16 ***
## poly(Shell_wt, 3)    3  197.34   65.78  177.603 < 2.2e-16 ***
## Infant               1   26.97   26.97   72.828 < 2.2e-16 ***
## Residuals         2910 1077.78    0.37
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The null hypothesis is that the variance are the same for the two groups (Infants and no infants). Therefore, as the p-value is very small we reject the null hypothesis indicating that the variance in the two groups is different. And so an hypothetical regression line between the two groups (infant and non infants) is very similar, meaning that there is no interaction between infant and the other covariates.

==========

```
#PART 3

#Q13
new_abalone_scale$Shell_wt2=new_abalone_scale$Shell_wt^2
new_abalone_scale$Shell_wt3=new_abalone_scale$Shell_wt^3
new_abalone_scale$Shuck_wt2=new_abalone_scale$Shuck_wt^2
new_abalone_scale$Shuck_wt3=new_abalone_scale$Shuck_wt^3
new_abalone_scale$Diameter2=new_abalone_scale$Diameter^2
new_abalone_scale$Diameter3=new_abalone_scale$Diameter^3

abalone_test$log_rings = log(abalone_test$Rings)
new_abalone_scale_test = data.frame(rapply(abalone_test, scale, c("numeric","integer"), how="replace"))
new_abalone_scale_test = new_abalone_scale_test[-c(9)]
new_abalone_scale_test['Infant'] = ifelse(new_abalone_scale_test$Sex == 'I','I','NI')
new_abalone_scale_test$Infant = as.factor(new_abalone_scale_test$Infant)
new_abalone_scale_test = new_abalone_scale_test[-c(1)]
new_abalone_scale_test$Shell_wt2=new_abalone_scale_test$Shell_wt^2
new_abalone_scale_test$Shell_wt3=new_abalone_scale_test$Shell_wt^3
new_abalone_scale_test$Shuck_wt2=new_abalone_scale_test$Shuck_wt^2
new_abalone_scale_test$Shuck_wt3=new_abalone_scale_test$Shuck_wt^3
new_abalone_scale_test$Diameter2=new_abalone_scale_test$Diameter^2
new_abalone_scale_test$Diameter3=new_abalone_scale_test$Diameter^3

linear_mod_log_simple = lm(log_rings ~ Height, data=new_abalone_scale)
y.hat = predict(linear_mod_log_simple, newdata=new_abalone_scale_test)
error.lse = mean((new_abalone_scale_test$log_rings-y.hat)^2)
error.lse
```

```
## [1] 0.5254822
```

```
AIC_error = nrow(new_abalone_scale_test)*log(error.lse) + 2*1
BIC_error = nrow(new_abalone_scale_test)*log(error.lse) + 1*log(nrow(new_abalone_scale_test))

AIC_error
```

```
## [1] -804.2291
```

```
BIC_error
```

```
## [1] -799.0958
```

```
linear_mod_log_sex = lm(log_rings ~ Diameter + Diameter2 + Diameter3 + Height +
                          Shuck_wt + Shuck_wt2 + Shuck_wt3 + Shell_wt + Shell_wt2 +
                          Shell_wt3 + Infant, data=new_abalone_scale)

y.hat = predict(linear_mod_log_sex, newdata=new_abalone_scale_test)
error.mult = mean((new_abalone_scale_test$log_rings-y.hat)^2)
error.mult
```

```
## [1] 0.3399011
```

```
AIC_error = nrow(new_abalone_scale_test)*log(error.mult) + 2*11
BIC_error = nrow(new_abalone_scale_test)*log(error.mult) + 11*log(nrow(new_abalone_scale_test))

AIC_error
```

```
## [1] -1330.113
```

```
BIC_error
```

```
## [1] -1273.647
```

We observe that the more complex model has a lower RMSE and lower AIC and BIC. Hence, the complex model is a better predictor for the data.

```
#Q14
reg0 <- lm(log_rings ~ 1, data=new_abalone_scale)
reg1 <- lm(log_rings ~ ., data=new_abalone_scale)
step(reg0, scope=list(lower=reg0, upper=reg1), data=new_abalone_scale, direction="both", k=log(nrow(new
```

```
## Start:  AIC=6.98
## log_rings ~ 1
##
##              Df Sum of Sq    RSS      AIC
## + Height      1   1354.52 1566.5 -1805.72
## + Diameter    1   1289.52 1631.5 -1686.92
## + Shell_wt    1   1287.12 1633.9 -1682.62
## + Length      1   1236.12 1684.9 -1592.82
## + Diameter3   1   1113.92 1807.1 -1388.22
## + Whole_wt    1   1026.61 1894.4 -1250.36
## + Visc_wt     1    920.39 2000.6 -1090.94
## + Shuck_wt    1    706.56 2214.4  -794.21
## + Infant      1    697.53 2223.5  -782.33
## + Diameter2   1    588.80 2332.2  -642.82
## + Shell_wt3   1    342.60 2578.4  -349.59
## + Shuck_wt3   1    152.41 2768.6  -141.63
## <none>                   2921.0     6.98
## + Shell_wt2   1      6.58 2914.4     8.37
## + Shuck_wt2   1      6.51 2914.5     8.44
##
## Step:  AIC=-1805.72
## log_rings ~ Height
##
##              Df Sum of Sq    RSS      AIC
## + Diameter2   1    123.77 1442.7 -2038.25
## + Diameter3   1    106.81 1459.7 -2004.10
## + Shuck_wt2   1     59.50 1507.0 -1910.89
## + Shuck_wt    1     55.98 1510.5 -1904.09
## + Infant      1     48.75 1517.7 -1890.13
## + Shell_wt    1     48.34 1518.1 -1889.34
## + Diameter    1     36.88 1529.6 -1867.36
## + Length      1     22.64 1543.8 -1840.28
## + Shuck_wt3   1     15.21 1551.3 -1826.26
```

```
## + Shell_wt3  1     10.75 1555.7 -1817.87
## + Visc_wt    1      8.35 1558.1 -1813.36
## <none>                   1566.5 -1805.72
## + Shell_wt2  1      4.25 1562.2 -1805.68
## + Whole_wt   1      1.39 1565.1 -1800.33
## - Height     1   1354.52 2921.0     6.98
##
## Step:  AIC=-2038.25
## log_rings ~ Height + Diameter2
##
##              Df Sum of Sq    RSS      AIC
## + Shell_wt   1    137.96 1304.7 -2323.97
## + Infant     1     44.02 1398.7 -2120.81
## + Shell_wt2  1     37.21 1405.5 -2106.62
## + Shell_wt3  1     33.59 1409.1 -2099.12
## + Shuck_wt   1     16.60 1426.1 -2064.08
## + Diameter3  1     14.06 1428.6 -2058.89
## + Whole_wt   1     13.02 1429.7 -2056.77
## + Diameter   1     12.95 1429.8 -2056.63
## + Length     1      4.03 1438.7 -2038.44
## <none>                   1442.7 -2038.25
## + Shuck_wt2  1      1.37 1441.3 -2033.06
## + Visc_wt    1      1.35 1441.4 -2033.01
## + Shuck_wt3  1      0.62 1442.1 -2031.54
## - Diameter2  1    123.77 1566.5 -1805.72
## - Height     1    889.50 2332.2  -642.82
##
## Step:  AIC=-2323.97
## log_rings ~ Height + Diameter2 + Shell_wt
##
##              Df Sum of Sq    RSS     AIC
## + Shuck_wt   1   130.718 1174.0 -2624.5
## + Whole_wt   1    62.740 1242.0 -2460.0
## + Length     1    57.449 1247.3 -2447.6
## + Visc_wt    1    40.228 1264.5 -2407.5
## + Diameter   1    33.838 1270.9 -2392.8
## + Infant     1    23.128 1281.6 -2368.2
## + Shuck_wt3  1     7.542 1297.2 -2332.9
## + Shuck_wt2  1     7.316 1297.4 -2332.4
## <none>                   1304.7 -2324.0
## - Height     1     6.032 1310.8 -2318.5
## + Diameter3  1     0.185 1304.6 -2316.4
## + Shell_wt3  1     0.045 1304.7 -2316.1
## + Shell_wt2  1     0.045 1304.7 -2316.1
## - Shell_wt   1   137.962 1442.7 -2038.2
## - Diameter2  1   213.394 1518.1 -1889.3
##
## Step:  AIC=-2624.46
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt
##
##              Df Sum of Sq    RSS     AIC
## + Infant     1    31.983 1142.0 -2697.2
## + Whole_wt   1    31.474 1142.5 -2695.9
## + Shuck_wt2  1    18.437 1155.6 -2662.7
```

```
## + Shuck_wt3  1    14.255 1159.8 -2652.2
## + Shell_wt2  1    10.240 1163.8 -2642.1
## + Diameter3  1     8.323 1165.7 -2637.3
## + Shell_wt3  1     7.880 1166.2 -2636.2
## <none>                   1174.0 -2624.5
## + Diameter   1     2.259 1171.8 -2622.1
## + Visc_wt    1     1.231 1172.8 -2619.6
## + Length     1     0.033 1174.0 -2616.6
## - Height     1    31.320 1205.3 -2555.5
## - Shuck_wt   1   130.718 1304.7 -2324.0
## - Diameter2  1   157.571 1331.6 -2264.4
## - Shell_wt   1   252.084 1426.1 -2064.1
##
## Step:  AIC=-2697.19
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt + Infant
##
##             Df Sum of Sq    RSS     AIC
## + Shuck_wt2  1    25.667 1116.4 -2755.6
## + Whole_wt   1    24.980 1117.1 -2753.8
## + Shuck_wt3  1    19.371 1122.7 -2739.2
## + Diameter3  1    13.411 1128.6 -2723.7
## + Shell_wt3  1     4.510 1137.5 -2700.8
## + Shell_wt2  1     4.487 1137.6 -2700.7
## <none>                   1142.0 -2697.2
## + Diameter   1     1.430 1140.6 -2692.9
## + Visc_wt    1     0.192 1141.8 -2689.7
## + Length     1     0.034 1142.0 -2689.3
## - Height     1    25.068 1167.1 -2641.7
## - Infant     1    31.983 1174.0 -2624.5
## - Shuck_wt   1   139.573 1281.6 -2368.2
## - Diameter2  1   142.367 1284.4 -2361.9
## - Shell_wt   1   230.956 1373.0 -2167.0
##
## Step:  AIC=-2755.63
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt + Infant +
##     Shuck_wt2
##
##             Df Sum of Sq    RSS     AIC
## + Whole_wt   1    27.465 1088.9 -2820.4
## + Shell_wt2  1    24.823 1091.5 -2813.3
## + Shell_wt3  1    14.640 1101.7 -2786.2
## + Diameter   1    10.605 1105.8 -2775.5
## + Length     1     4.395 1112.0 -2759.2
## <none>                   1116.4 -2755.6
## + Diameter3  1     2.202 1114.2 -2753.4
## + Visc_wt    1     0.799 1115.6 -2749.7
## + Shuck_wt3  1     0.063 1116.3 -2747.8
## - Shuck_wt2  1    25.667 1142.0 -2697.2
## - Height     1    29.006 1145.4 -2688.7
## - Infant     1    39.214 1155.6 -2662.7
## - Shuck_wt   1   159.550 1275.9 -2373.3
## - Diameter2  1   161.573 1278.0 -2368.7
## - Shell_wt   1   251.362 1367.7 -2170.2
##
```

```
## Step:  AIC=-2820.44
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt + Infant +
##     Shuck_wt2 + Whole_wt
##
##             Df Sum of Sq    RSS     AIC
## + Shell_wt2  1    22.386 1066.5 -2873.2
## + Shell_wt3  1    11.873 1077.0 -2844.5
## + Visc_wt    1     7.950 1081.0 -2833.9
## + Diameter   1     4.778 1084.1 -2825.3
## <none>                   1088.9 -2820.4
## + Length     1     0.503 1088.4 -2813.8
## + Diameter3  1     0.364 1088.5 -2813.4
## + Shuck_wt3  1     0.000 1088.9 -2812.5
## - Height     1    18.452 1107.4 -2779.3
## - Whole_wt   1    27.465 1116.4 -2755.6
## - Shuck_wt2  1    28.153 1117.1 -2753.8
## - Infant     1    32.080 1121.0 -2743.6
## - Shell_wt   1    35.917 1124.8 -2733.6
## - Shuck_wt   1   115.885 1204.8 -2532.9
## - Diameter2  1   178.664 1267.6 -2384.5
##
## Step:  AIC=-2873.15
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt + Infant +
##     Shuck_wt2 + Whole_wt + Shell_wt2
##
##             Df Sum of Sq    RSS     AIC
## + Visc_wt    1    11.107 1055.4 -2895.8
## + Diameter3  1     3.627 1062.9 -2875.1
## <none>                   1066.5 -2873.2
## + Shell_wt3  1     2.692 1063.8 -2872.6
## + Diameter   1     0.882 1065.6 -2867.6
## + Shuck_wt3  1     0.725 1065.8 -2867.2
## + Length     1     0.148 1066.4 -2865.6
## - Height     1    11.868 1078.4 -2848.8
## - Shell_wt2  1    22.386 1088.9 -2820.4
## - Infant     1    23.805 1090.3 -2816.6
## - Whole_wt   1    25.028 1091.5 -2813.3
## - Shuck_wt2  1    47.306 1113.8 -2754.3
## - Shell_wt   1    57.273 1123.8 -2728.3
## - Diameter2  1   120.512 1187.0 -2568.3
## - Shuck_wt   1   133.480 1200.0 -2536.6
##
## Step:  AIC=-2895.76
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt + Infant +
##     Shuck_wt2 + Whole_wt + Shell_wt2 + Visc_wt
##
##             Df Sum of Sq    RSS     AIC
## + Diameter3  1     4.805 1050.6 -2901.1
## + Shell_wt3  1     3.010 1052.4 -2896.1
## <none>                   1055.4 -2895.8
## + Diameter   1     1.404 1054.0 -2891.7
## + Shuck_wt3  1     0.781 1054.6 -2889.9
## + Length     1     0.009 1055.4 -2887.8
## - Visc_wt    1    11.107 1066.5 -2873.2
```

```
## - Height     1    14.089 1069.5 -2865.0
## - Infant     1    25.006 1080.4 -2835.3
## - Shell_wt2  1    25.543 1081.0 -2833.9
## - Whole_wt   1    36.024 1091.4 -2805.7
## - Shell_wt   1    43.101 1098.5 -2786.8
## - Shuck_wt2  1    46.327 1101.8 -2778.2
## - Diameter2  1   111.868 1167.3 -2609.4
## - Shuck_wt   1   144.166 1199.6 -2529.6
##
## Step:  AIC=-2901.12
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt + Infant +
##     Shuck_wt2 + Whole_wt + Shell_wt2 + Visc_wt + Diameter3
##
##             Df Sum of Sq    RSS     AIC
## + Shell_wt3  1     5.024 1045.6 -2907.1
## <none>                   1050.6 -2901.1
## - Diameter3  1     4.805 1055.4 -2895.8
## + Length     1     0.839 1049.8 -2895.5
## + Shuck_wt3  1     0.410 1050.2 -2894.3
## + Diameter   1     0.002 1050.6 -2893.1
## - Visc_wt    1    12.285 1062.9 -2875.1
## - Height     1    12.963 1063.6 -2873.3
## - Diameter2  1    17.106 1067.7 -2861.9
## - Infant     1    26.342 1077.0 -2836.7
## - Shell_wt2  1    29.779 1080.4 -2827.4
## - Shuck_wt2  1    31.351 1082.0 -2823.2
## - Whole_wt   1    33.008 1083.6 -2818.7
## - Shell_wt   1    44.382 1095.0 -2788.2
## - Shuck_wt   1   139.565 1190.2 -2544.6
##
## Step:  AIC=-2907.14
## log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt + Infant +
##     Shuck_wt2 + Whole_wt + Shell_wt2 + Visc_wt + Diameter3 +
##     Shell_wt3
##
##             Df Sum of Sq    RSS     AIC
## <none>                   1045.6 -2907.1
## + Length     1     2.403 1043.2 -2905.9
## + Shuck_wt3  1     1.880 1043.7 -2904.4
## - Diameter2  1     4.488 1050.1 -2902.6
## + Diameter   1     0.821 1044.8 -2901.5
## - Shell_wt3  1     5.024 1050.6 -2901.1
## - Diameter3  1     6.819 1052.4 -2896.1
## - Height     1    12.862 1058.5 -2879.4
## - Visc_wt    1    12.999 1058.6 -2879.0
## - Shell_wt2  1    20.106 1065.7 -2859.5
## - Infant     1    23.653 1069.2 -2849.8
## - Whole_wt   1    34.228 1079.8 -2821.0
## - Shuck_wt2  1    34.980 1080.6 -2819.0
## - Shell_wt   1    44.728 1090.3 -2792.7
## - Shuck_wt   1   144.575 1190.2 -2536.7

##
## Call:
```

```
## lm(formula = log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt +
##     Infant + Shuck_wt2 + Whole_wt + Shell_wt2 + Visc_wt + Diameter3 +
##     Shell_wt3, data = new_abalone_scale)
##
## Coefficients:
## (Intercept)       Height    Diameter2     Shell_wt     Shuck_wt     InfantNI
##     0.01532      0.17961     -0.09408      0.64185     -1.30013      0.24473
##   Shuck_wt2     Whole_wt    Shell_wt2      Visc_wt    Diameter3    Shell_wt3
##     0.12937      1.14174     -0.20944     -0.27935      0.03435      0.02196
```

By doing step-wise variable selection using BIC criteria, we have eliminated 'Length', 'Shuck_wt3', and 'Diameter' from the ANOVA model in Part 2.

```
regnew <- lm(formula = log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt +
    Infant + Shuck_wt2 + Whole_wt + Shell_wt2 + Visc_wt + Diameter3 +
    Shell_wt3, data = new_abalone_scale)

reganova <- lm(formula = log_rings ~ Height + Diameter2 + Shell_wt + Shuck_wt +
    Infant + Shuck_wt2 + Whole_wt + Shell_wt2 + Visc_wt + Diameter3 + Diameter +
    Shell_wt3 + Shuck_wt3 + Length, data = new_abalone_scale)

error_new = mean((predict(regnew, newdata=new_abalone_scale_test) - new_abalone_scale_test$log_rings)^2)
error_anova = mean((predict(reganova, newdata=new_abalone_scale_test) - new_abalone_scale_test$log_rings
bicnew = nrow(new_abalone_scale_test)*log(error_new) + 11*log(nrow(new_abalone_scale_test))
bicanova = nrow(new_abalone_scale_test)*log(error_anova) + 14*log(nrow(new_abalone_scale_test))
error_new
```

```
## [1] 0.3274448
```

```
error_anova
```

```
## [1] 0.3253462
```

```
bicnew
```

```
## [1] -1320.428
```

```
bicanova
```

```
## [1] -1307.084
```

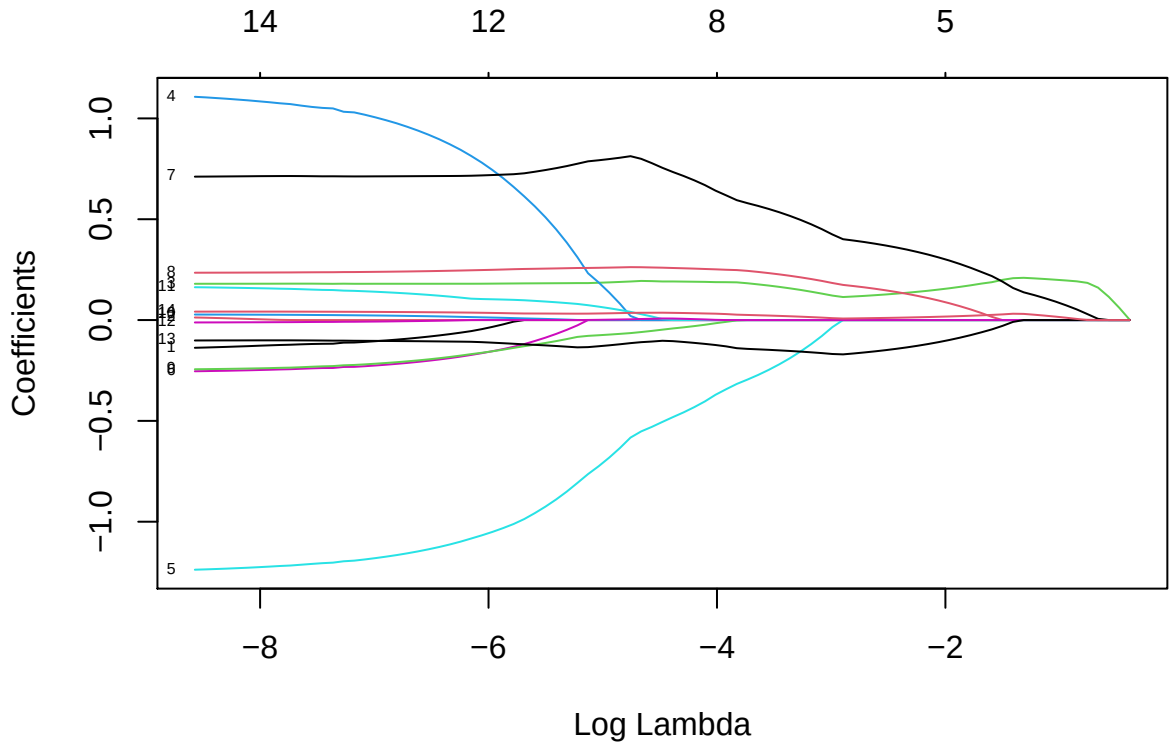
The prediction error on the new model is 0.327, which is roughly equal to the ANOVA model prediction error of 0.325. However, the BIC criterion for our new model is lower after we use step-wise feature selection. Thus, our new model doesn't have a better RMSE as we focus on minimizing BIC rather than RMSE while performing model selection.

```
#Q15
custom = trainControl(method='repeatedcv', number=3, repeats=10)
set.seed(1012)

#LASSO Regression
```

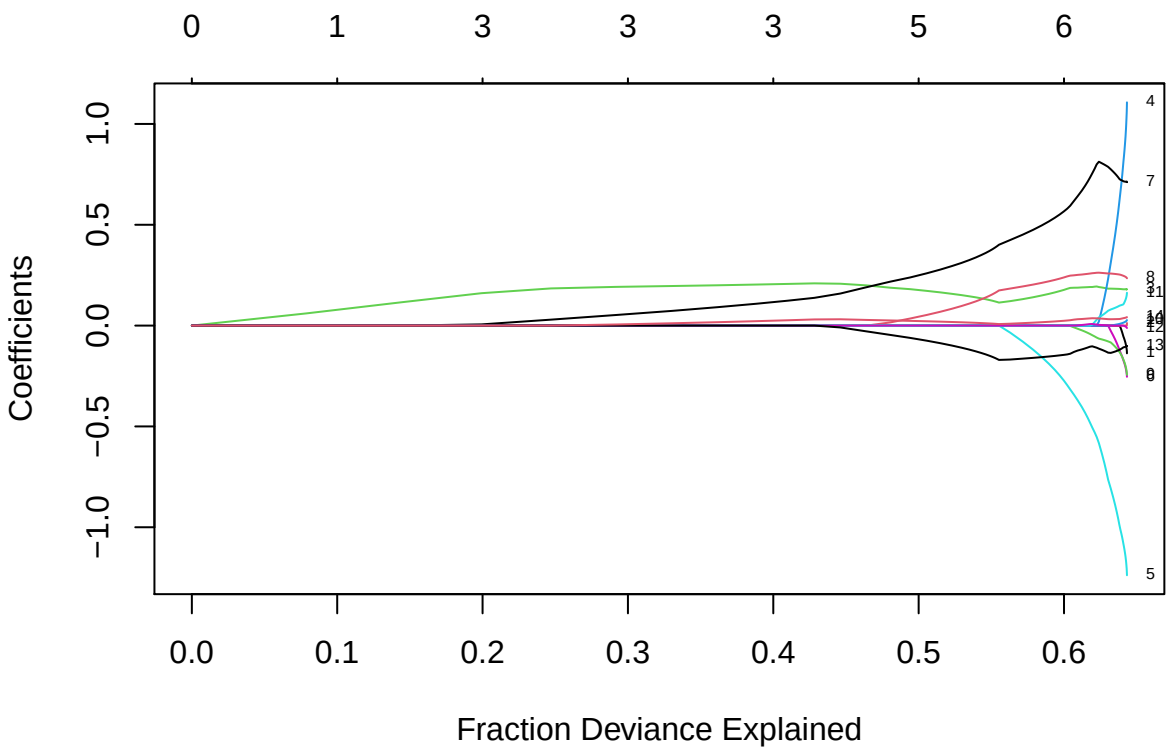
```
lasso = train(log_rings ~ ., new_abalone_scale, method='glmnet',
              tuneGrid=expand.grid(alpha=1, lambda=seq(0.001, 0.1, length=100)),
              trControl=custom)

lasso$bestTune
```

```
##   alpha lambda
## 1     1  0.001
```

```
plot(lasso$finalModel, xvar="lambda", label=T)
```




```
plot(lasso$finalModel, xvar="dev", label=T)
```

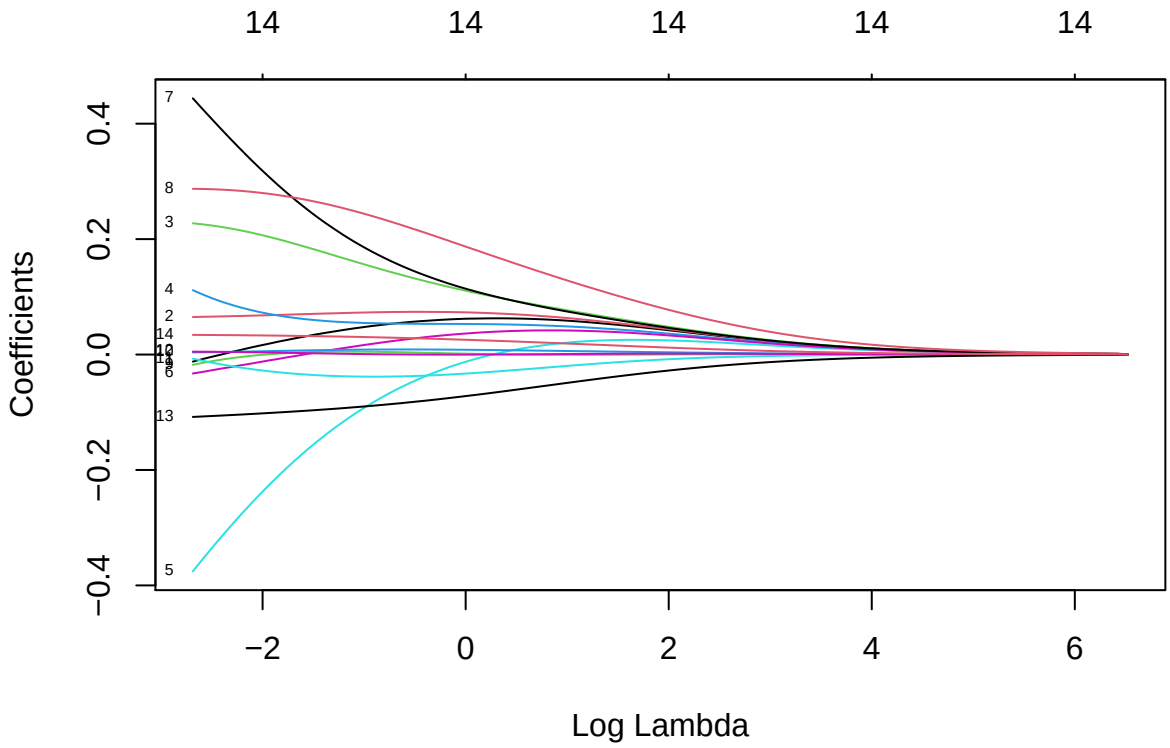


We observe that optimal lambda which minimizes RMSE is very close to 0. Thus, LASSO might not be the best regularization in this case.

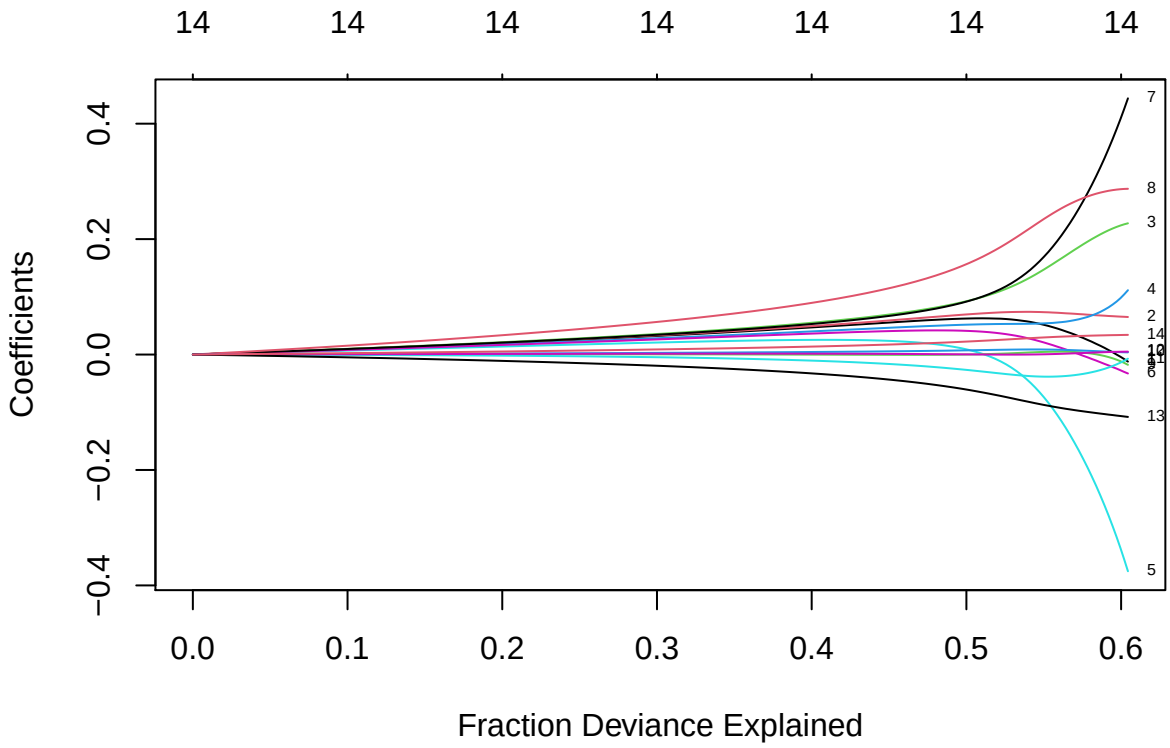
```
#Ridge Regression
ridge = train(log_rings ~ ., new_abalone_scale, method='glmnet',
              tuneGrid=expand.grid(alpha=0, lambda=seq(0.001, 0.1, length=100)),
              trControl=custom)

ridge$bestTune
```

```
##    alpha lambda
## 66     0  0.066
```

```
plot(ridge$finalModel, xvar="lambda", label=T)
```




```
plot(ridge$finalModel, xvar="dev", label=T)
```

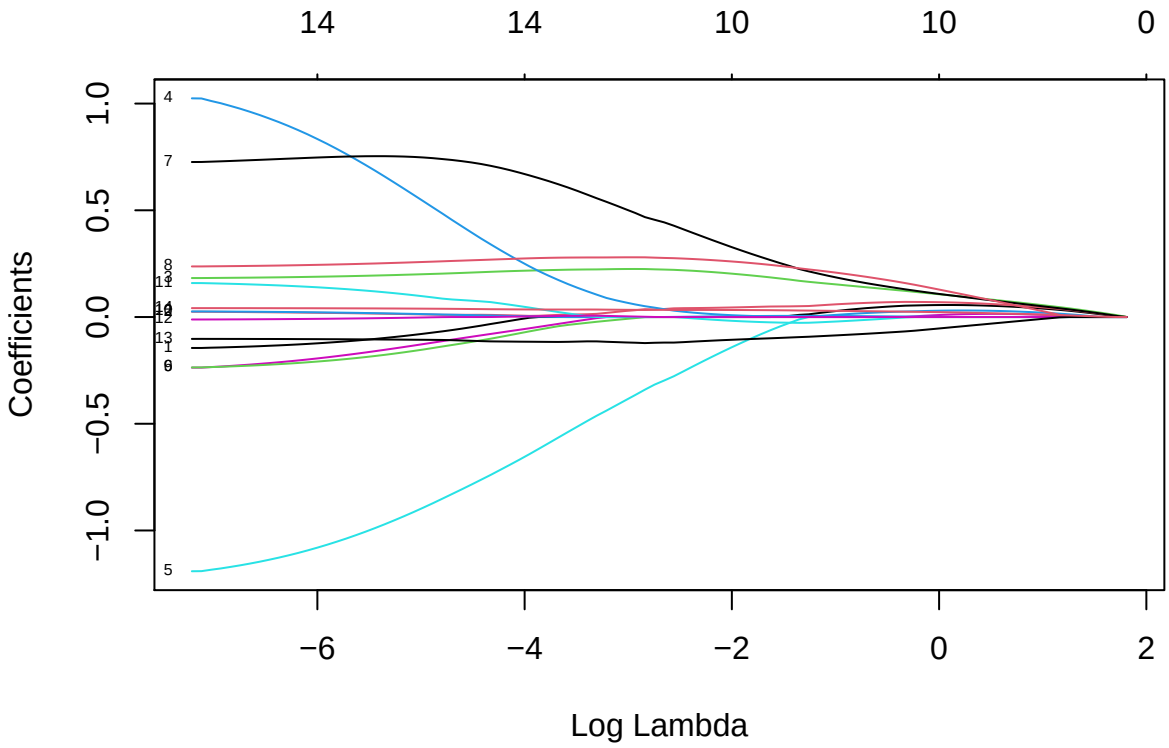


We observe that optimal lambda which minimizes RMSE is very close to 0. Thus, Ridge might not be the best regularization in this case.

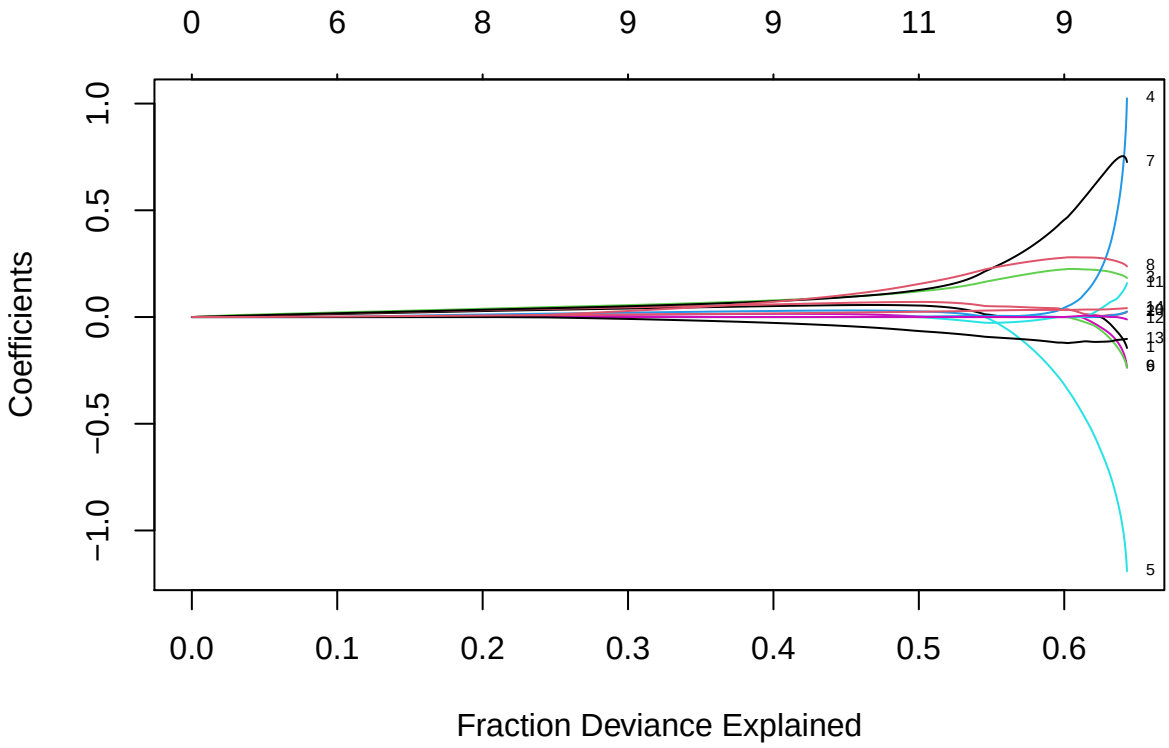
```
#Elastic Net
ela=train(log_rings~.,new_abalone_scale,
                  method='glmnet',
                   tuneGrid=expand.grid(alpha=seq(0, 1, length=10),lambda=seq(0.001, 0.1, length=100)),
                    trControl=custom)

ela$bestTune
```

```
##         alpha lambda
## 101 0.1111111  0.001
```

```
plot(ela$finalModel, xvar="lambda", label=T)
```




```
plot(ela$finalModel, xvar="dev", label=T)
```



The optimal alpha and lambda are given by 0.111 and 0.001. All 3 models have RMSE between 0.6 and 0.63, which is greater than the RMSE of our model 0.327. Hence, we choose the model specified after step-wise model selection as the best model for this problem.