R lab 5.0

Пользуясь примером из лекции файл (5.0.R) проанализируйте данные о возрасте и физ. характеристиках молюсков https://archive.ics.uci.edu/ml/datasets/abalone (https://archive.ics.uci.edu/ml/datasets/abalone)

```
data <- read.csv("https://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.dat
a", header=TRUE, sep=",")
summary(data)
```

```
##  M            X0.455          X0.365           X0.095      
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000  
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150  
##  M:1527   Median :0.545   Median :0.4250   Median :0.1400  
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395  
##           3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650  
##           Max.   :0.815   Max.   :0.6500   Max.   :1.1300  
##      X0.514          X0.2245           X0.101            X0.15       
##  Min.   :0.0020   Min.   :0.0010   Min.   :0.00050   Min.   :0.0015  
##  1st Qu.:0.4415   1st Qu.:0.1860   1st Qu.:0.09337   1st Qu.:0.1300  
##  Median :0.7997   Median :0.3360   Median :0.17100   Median :0.2340  
##  Mean   :0.8288   Mean   :0.3594   Mean   :0.18061   Mean   :0.2389  
##  3rd Qu.:1.1533   3rd Qu.:0.5020   3rd Qu.:0.25300   3rd Qu.:0.3290  
##  Max.   :2.8255   Max.   :1.4880   Max.   :0.76000   Max.   :1.0050  
##       X15        
##  Min.   : 1.000  
##  1st Qu.: 8.000  
##  Median : 9.000  
##  Mean   : 9.932  
##  3rd Qu.:11.000  
##  Max.   :29.000  
```

```
colnames(data)
```

```
## [1] "M"       "X0.455"  "X0.365"  "X0.095"  "X0.514"  "X0.2245" "X0.101" 
## [8] "X0.15"   "X15"
```

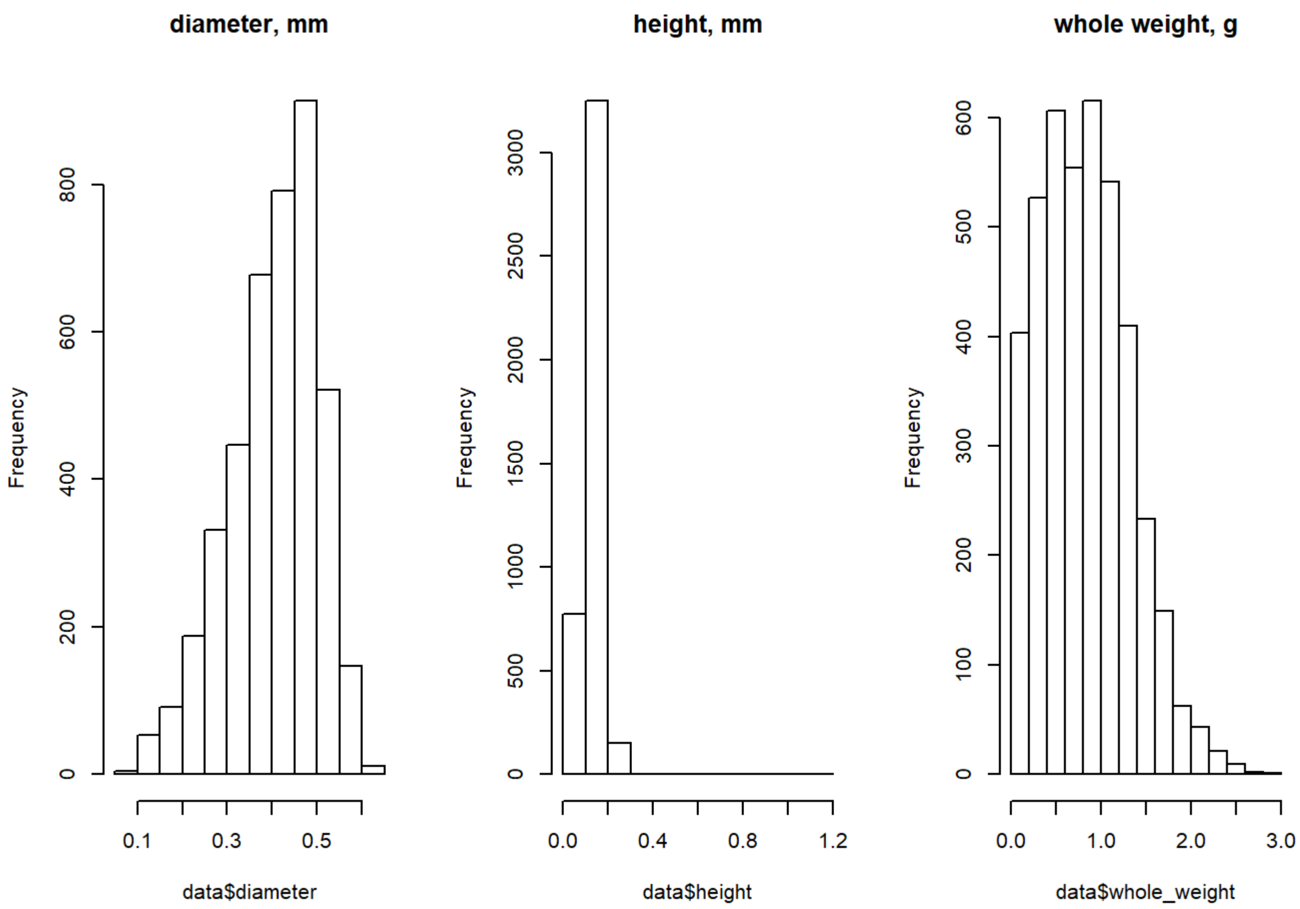
```
colnames(data) <- c("sex", "length", "diameter", "height", 
                "whole_weight", "shucked_weight",
                "viscera_weight", "shell_weight", "rings")

colnames(data)
```

```
## [1] "sex"            "length"         "diameter"       "height"        
## [5] "whole_weight"   "shucked_weight" "viscera_weight" "shell_weight"  
## [9] "rings"
```

```
summary(data)
```

```
##  sex          length         diameter          height
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150
##  M:1527   Median :0.545   Median :0.4250   Median :0.1400
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395
##           3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650
##           Max.   :0.815   Max.   :0.6500   Max.   :1.1300
##   whole_weight    shucked_weight   viscera_weight     shell_weight
##  Min.   :0.0020   Min.   :0.0010   Min.   :0.00050   Min.   :0.0015
##  1st Qu.:0.4415   1st Qu.:0.1860   1st Qu.:0.09337   1st Qu.:0.1300
##  Median :0.7997   Median :0.3360   Median :0.17100   Median :0.2340
##  Mean   :0.8288   Mean   :0.3594   Mean   :0.18061   Mean   :0.2389
##  3rd Qu.:1.1533   3rd Qu.:0.5020   3rd Qu.:0.25300   3rd Qu.:0.3290
##  Max.   :2.8255   Max.   :1.4880   Max.   :0.76000   Max.   :1.0050
##      rings
##  Min.   : 1.000
##  1st Qu.: 8.000
##  Median : 9.000
##  Mean   : 9.932
##  3rd Qu.:11.000
##  Max.   :29.000
```

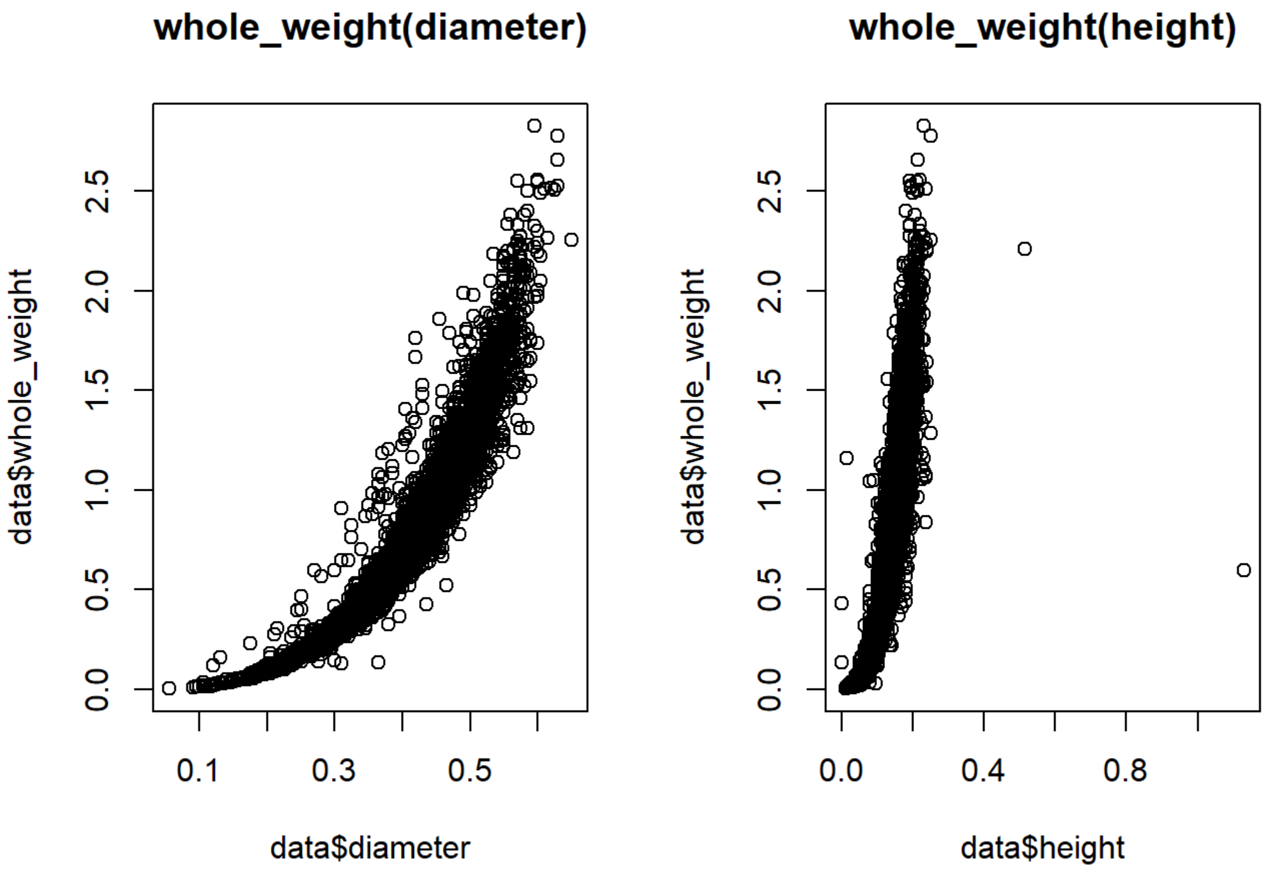
```
data$sex <- factor(c("Female", "Infant", "Male")[data$sex])
par(mfrow=c(1,3)) #Set or Query Graphical Parameters
hist(data$diameter, main = "diameter, mm")
hist(data$height, main = "height, mm")
hist(data$whole_weight, main = "whole weight, g")
```



Видим ассиметрию https://en.wikipedia.org/wiki/Skewness (https://en.wikipedia.org/wiki/Skewness) и выбросы (от них нужно избавиться)

Визулизируем возможные зависимости

```
par(mfrow=c(1,2))
plot(data$diameter, data$whole_weight,'p',main = "whole_weight(diameter)")
plot(data$height, data$whole_weight,'p',main = "whole_weight(height)")
```



Хорошо видна зависимость, нужно её исследовать построить линейные модели при помощи функции lm, посмотреть их характеристики избавиться от выборосов, построить ещё модели и проверить их разделить массив данных на 2 случайные части подогнать модель по первой части спрогнозировать (функция predict) значения во второй части проверить качесвто прогноза

Линейные модели

Зависимость веса от диаметра

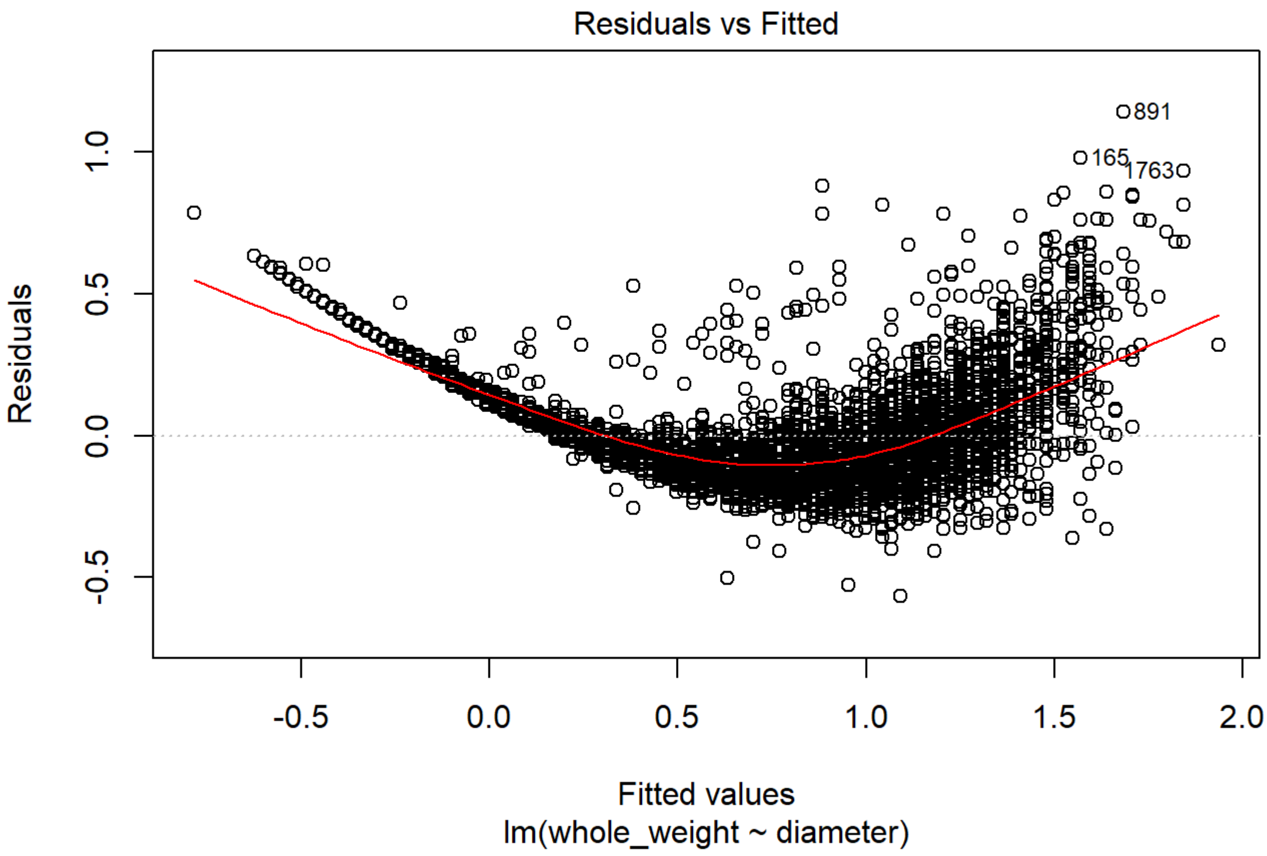
```
linear.model.wd<-lm(whole_weight~diameter, data=data)
linear.model.wd
```

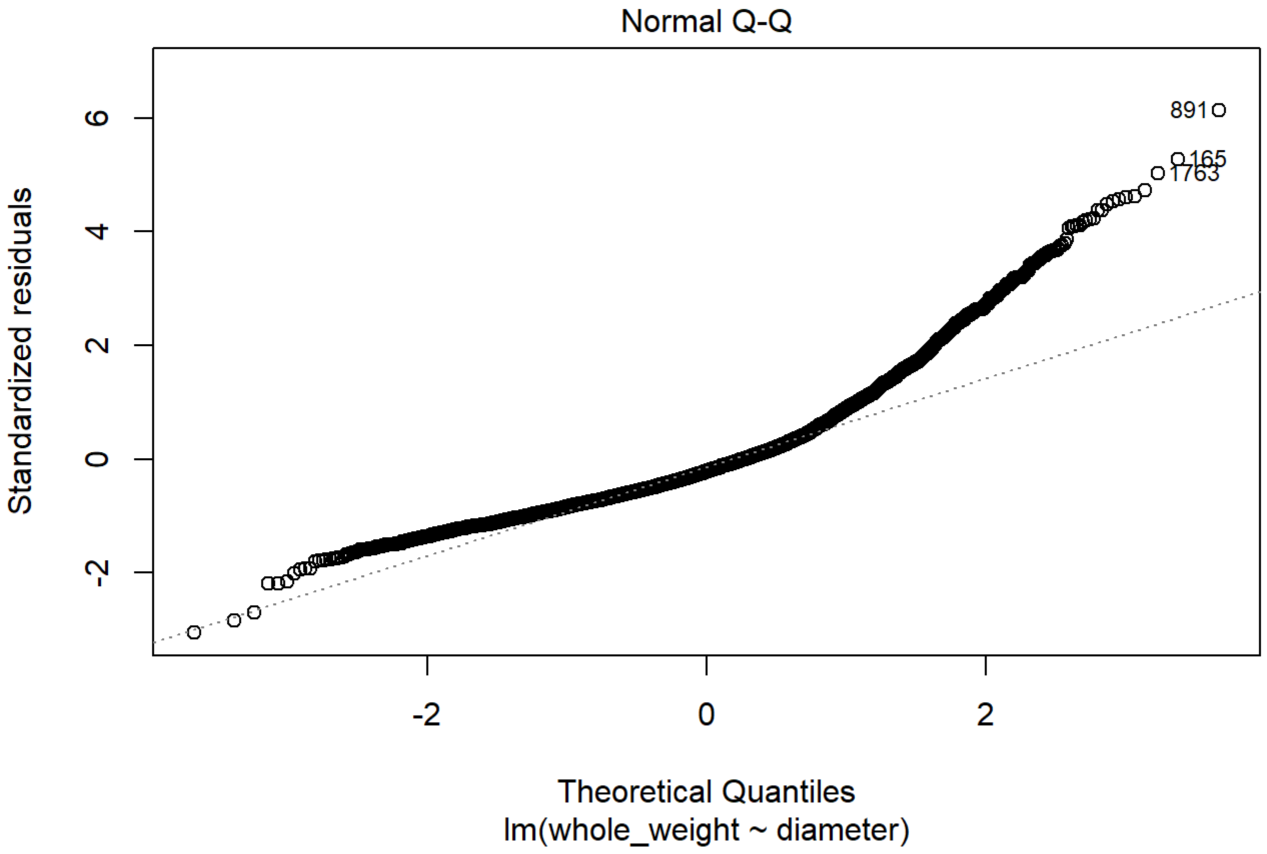
```
##
## Call:
## lm(formula = whole_weight ~ diameter, data = data)
##
## Coefficients:
## (Intercept)     diameter
##      -1.036        4.573
```

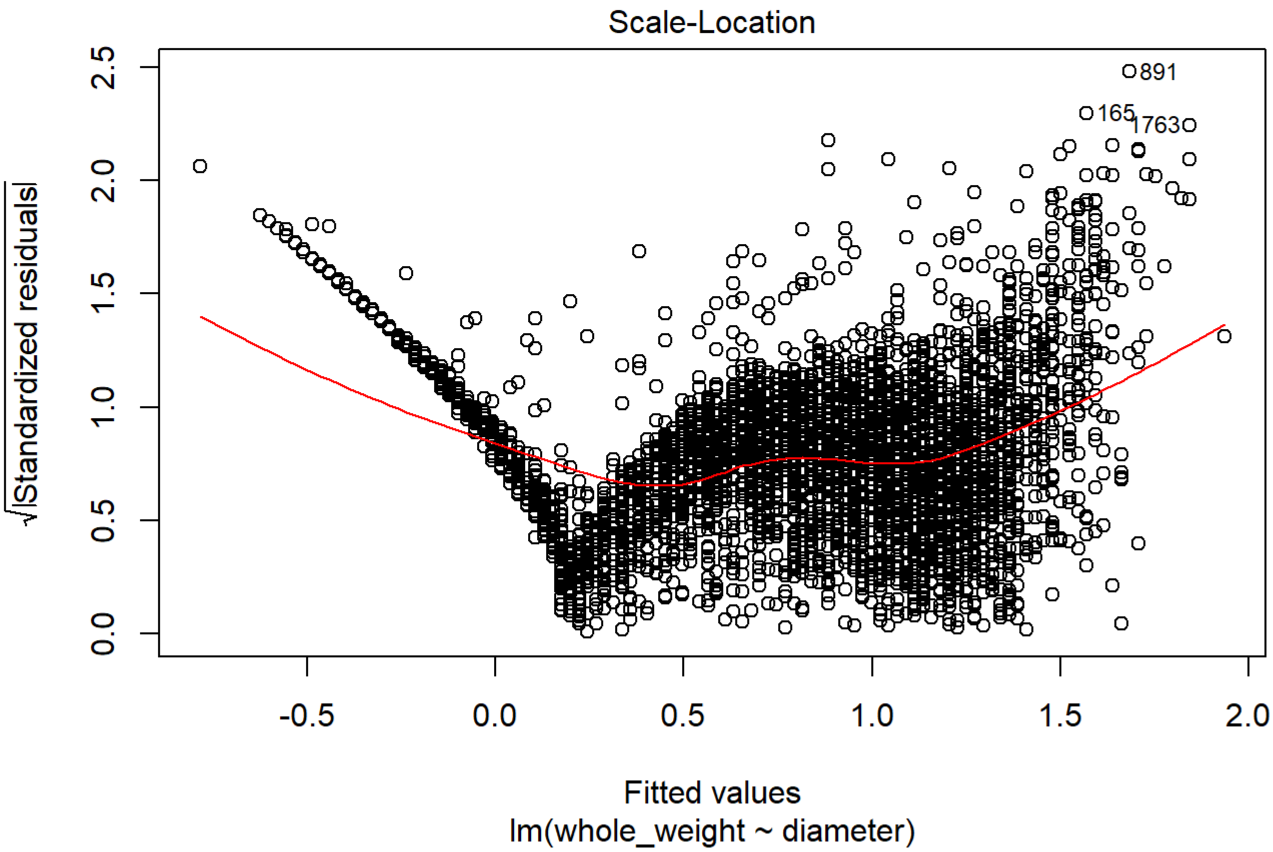
```
summary(linear.model.wd)
```

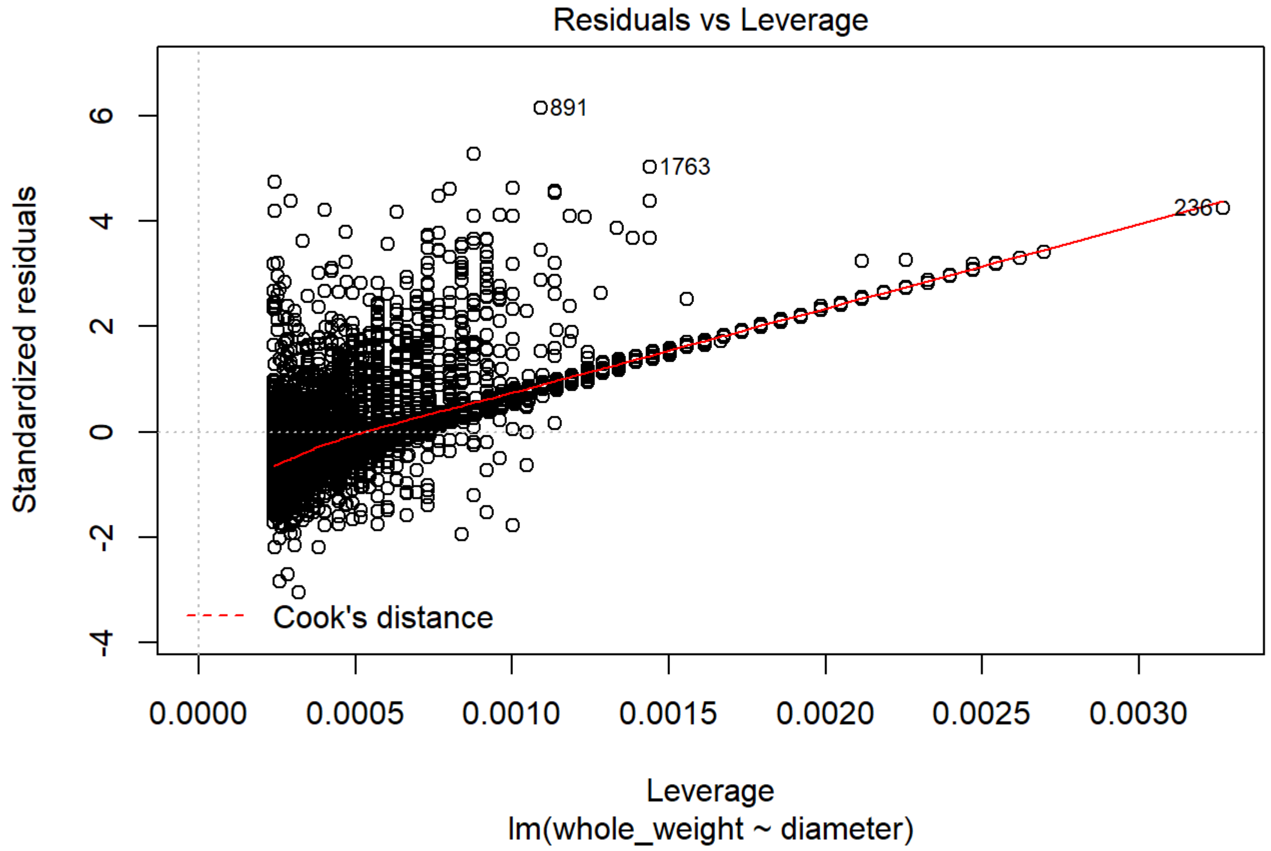
```
##
## Call:
## lm(formula = whole_weight ~ diameter, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.56747 -0.12310 -0.03997  0.07211  1.14104
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.03645    0.01216   -85.2   <2e-16 ***
## diameter     4.57295    0.02898   157.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1858 on 4174 degrees of freedom
## Multiple R-squared:  0.8565, Adjusted R-squared:  0.8564
## F-statistic: 2.491e+04 on 1 and 4174 DF,  p-value: < 2.2e-16
```

```
plot(linear.model.wd)
```


Residuals vs Fitted
Residuals
-0.5
0.0
0.5
1.0
Fitted values
lm(whole_weight ~ diameter)
-0.5
0.0
0.5
1.0
1.5
2.0
891
165
1763


Normal Q-Q
Standardized residuals
-2
0
2
4
6
Theoretical Quantiles
lm(whole_weight ~ diameter)
-2
0
2
891
165
1763



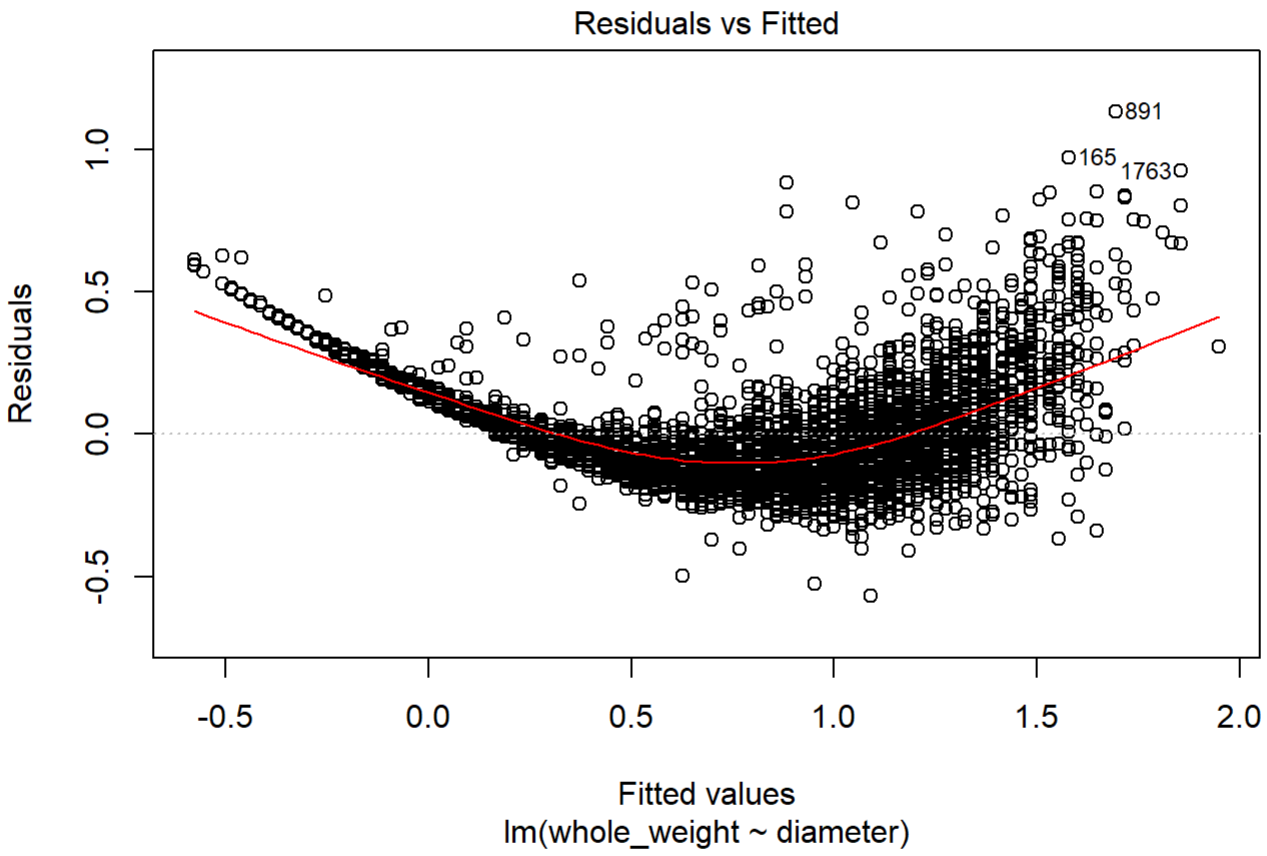


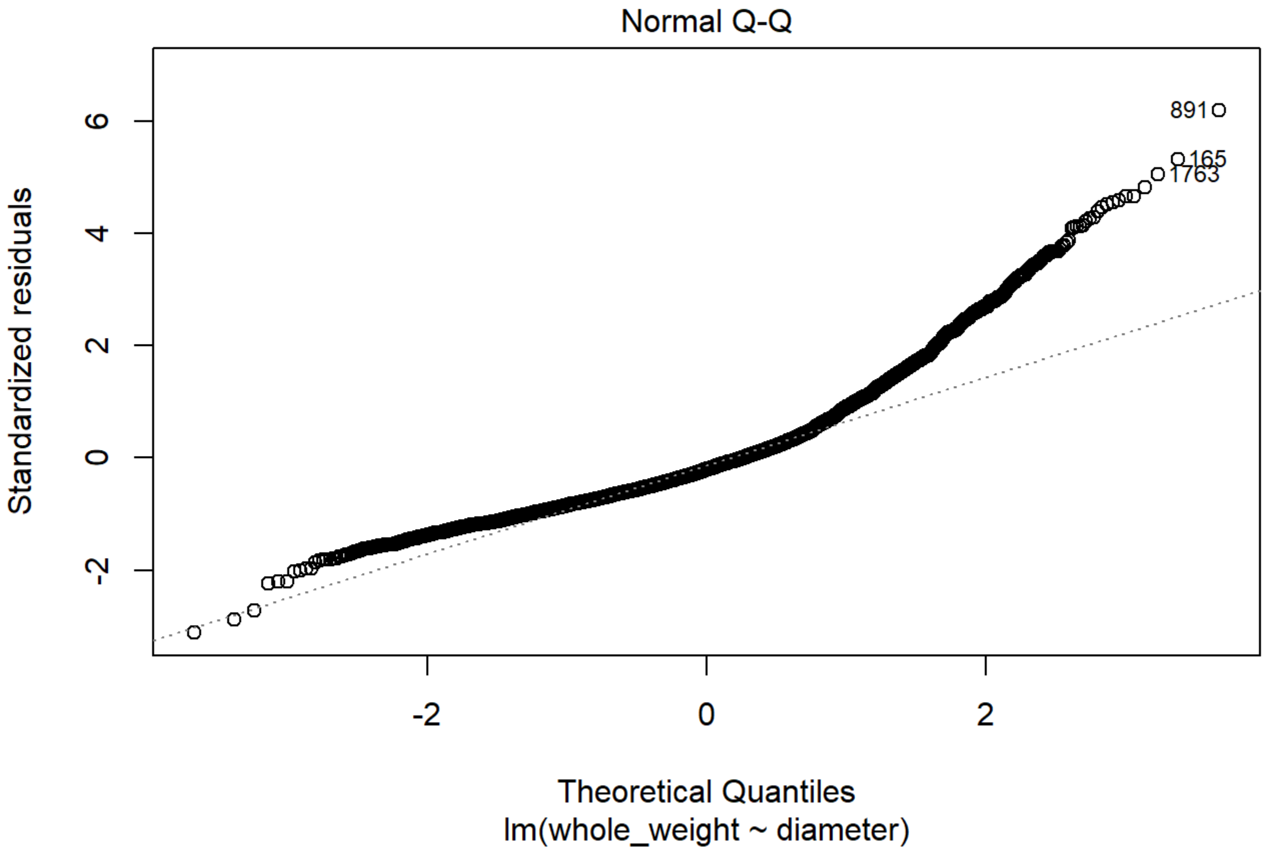
###Выбросы = outlier

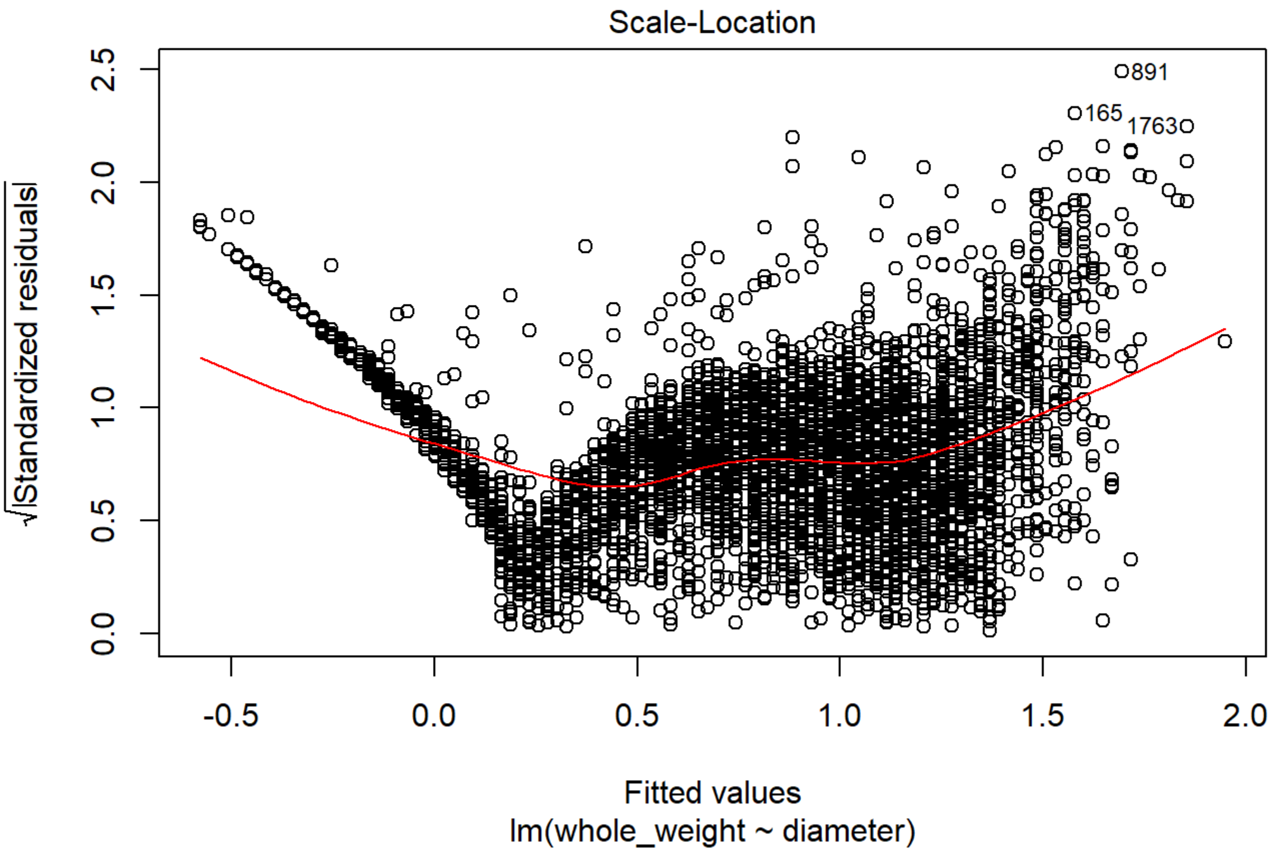
```
data.noout<-data[data$height<0.4&data$height>0.03&data$diameter>0.1,]
linear.model.wd.outlier<-lm(whole_weight~diameter,data=data.noout)
linear.model.wd.outlier
```

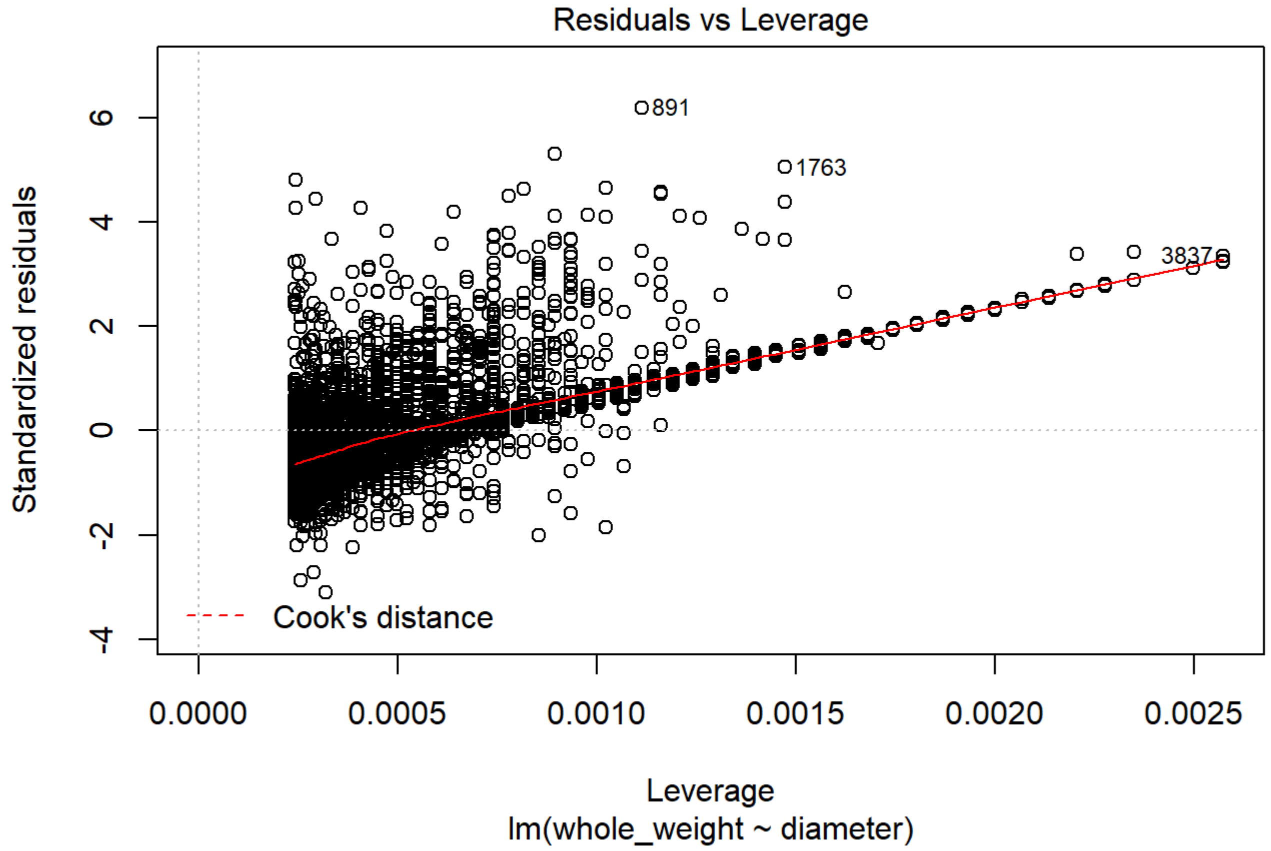
```
##
## Call:
## lm(formula = whole_weight ~ diameter, data = data.noout)
##
## Coefficients:
## (Intercept)     diameter
##      -1.065        4.636
```

```
plot(linear.model.wd.outlier)
```


Residuals vs Fitted
Residuals
1.0
0.5
0.0
-0.5
-0.5
0.0
0.5
1.0
1.5
2.0
891
165
1763
Fitted values
lm(whole_weight ~ diameter)


Normal Q-Q
Standardized residuals
6
4
2
0
-2
-2
0
2
891
165
1763
Theoretical Quantiles
lm(whole_weight ~ diameter)





##Зависимость веса от высоты

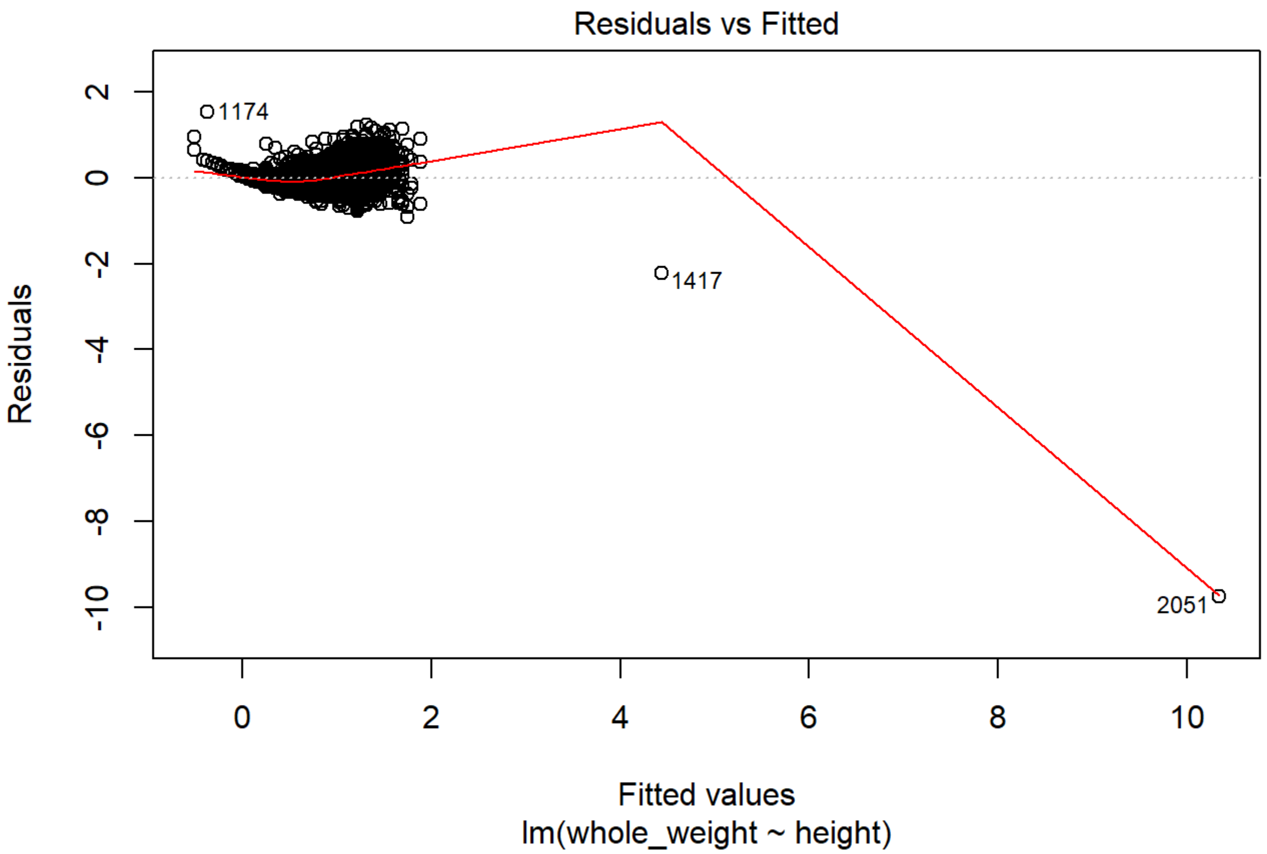
```
linear.model.wh<-lm(whole_weight~height, data=data)
linear.model.wh
```

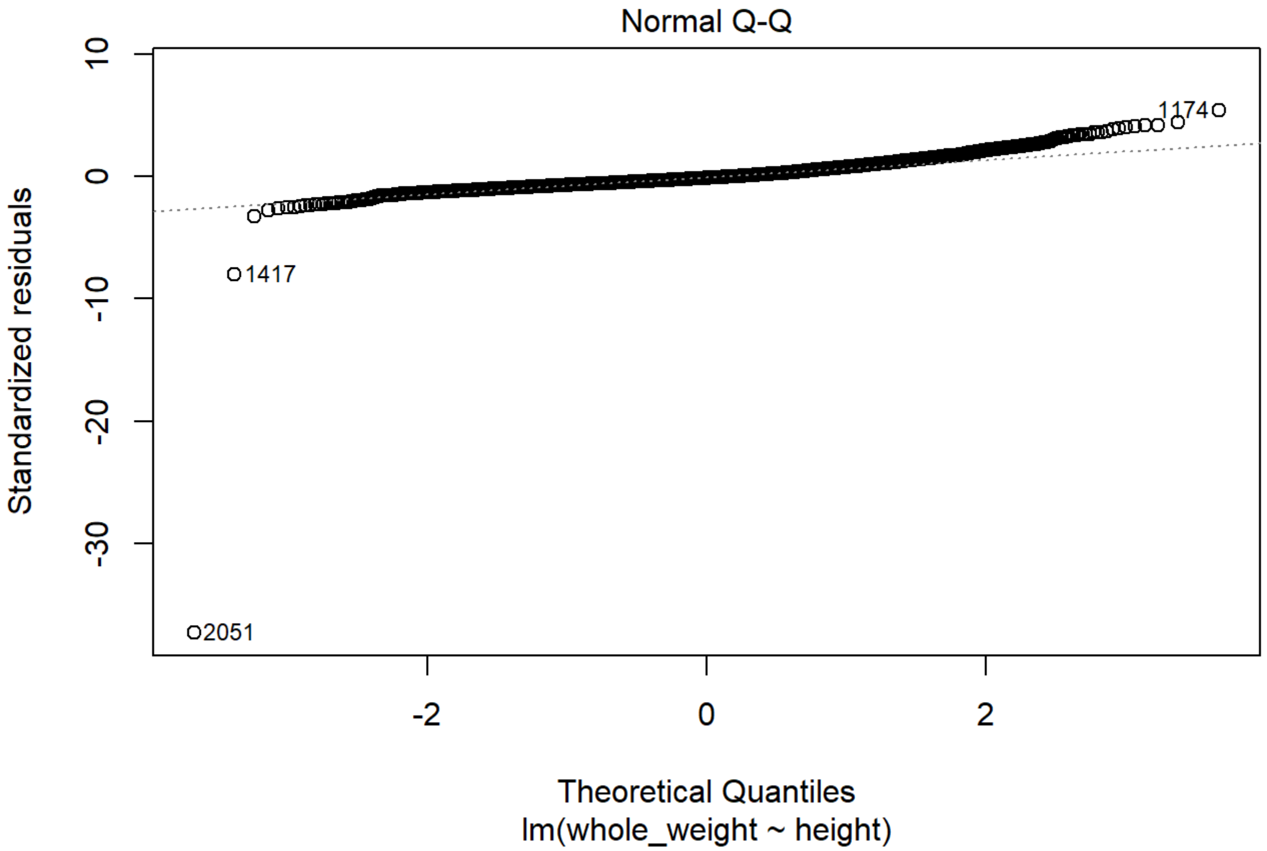
```
##
## Call:
## lm(formula = whole_weight ~ height, data = data)
##
## Coefficients:
## (Intercept)       height
##     -0.5114       9.6054
```

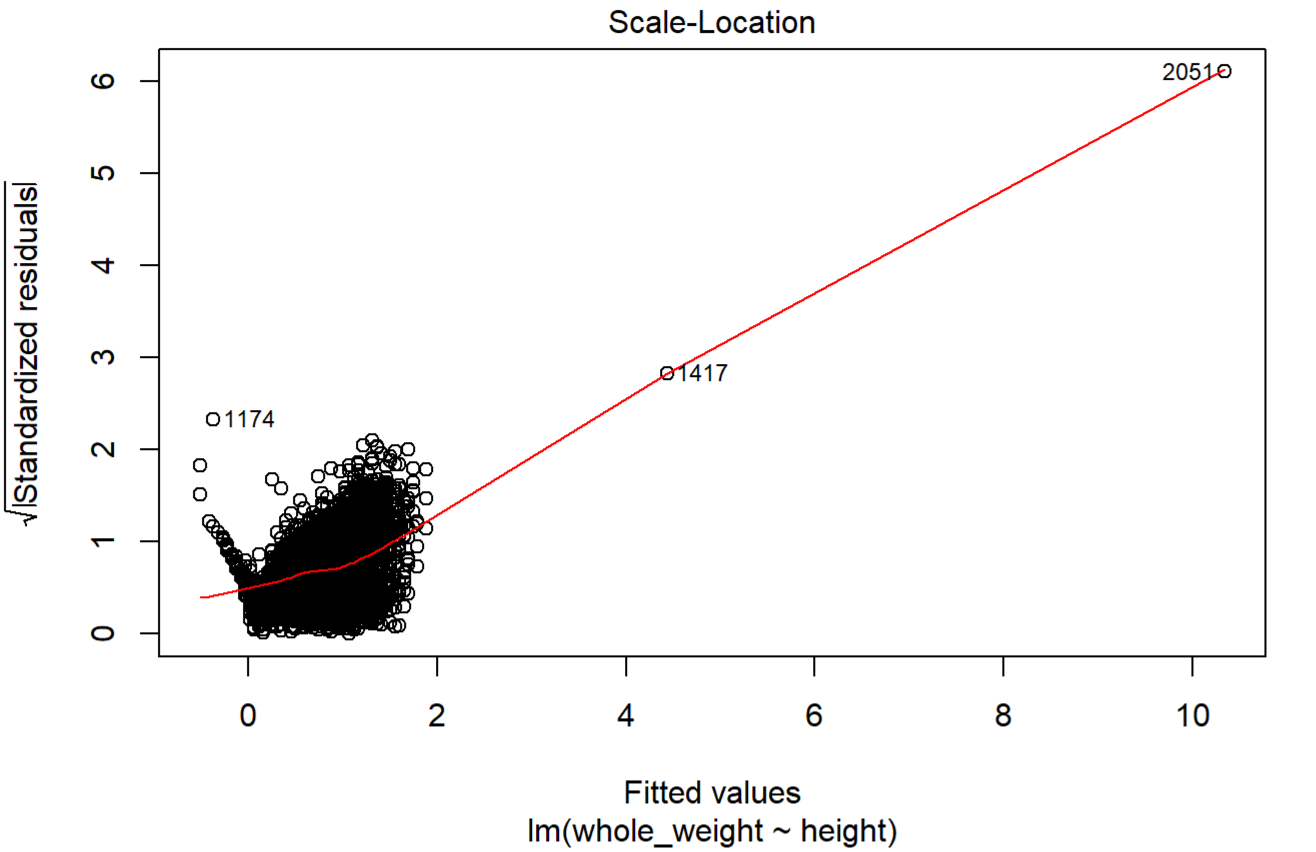
```
summary(linear.model.wh)
```

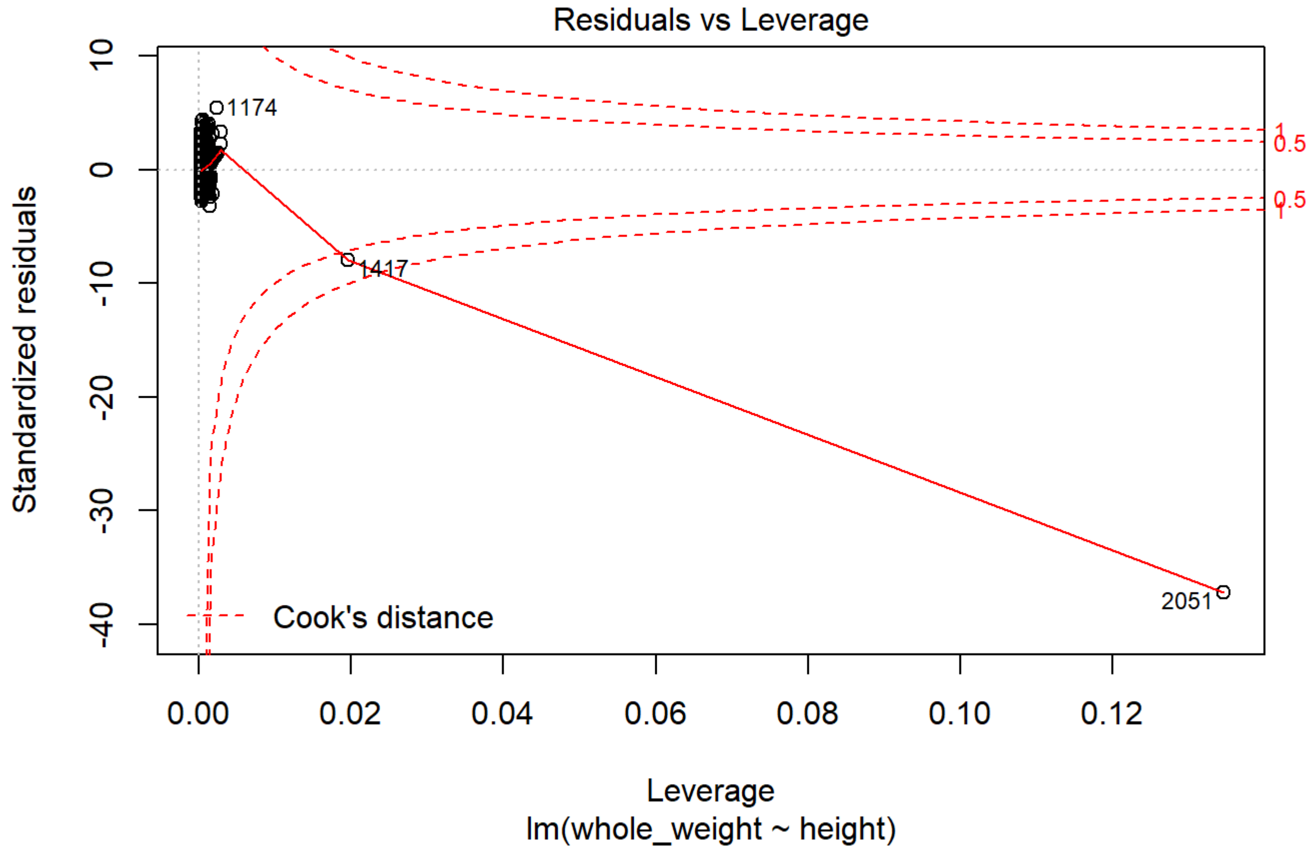
```
##
## Call:
## lm(formula = whole_weight ~ height, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.7487 -0.1488 -0.0346  0.1151  1.5238
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.51140    0.01516  -33.73   <2e-16 ***
## height       9.60540    0.10408   92.29   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2813 on 4174 degrees of freedom
## Multiple R-squared:  0.6711, Adjusted R-squared:  0.671
## F-statistic:  8517 on 1 and 4174 DF,  p-value: < 2.2e-16
```

```
plot(linear.model.wh)
```


Residuals vs Fitted
Residuals
2
0
-2
-4
-6
-8
-10
0
2
4
6
8
10
1174
1417
2051
Fitted values
lm(whole_weight ~ height)


Normal Q-Q
Standardized residuals
10
0
-10
-20
-30
-2
0
2
1174
1417
2051
Theoretical Quantiles
lm(whole_weight ~ height)


Scale-Location
√|Standardized residuals|
0
1
2
3
4
5
6
2051
1417
1174
0
2
4
6
8
10
Fitted values
lm(whole_weight ~ height)

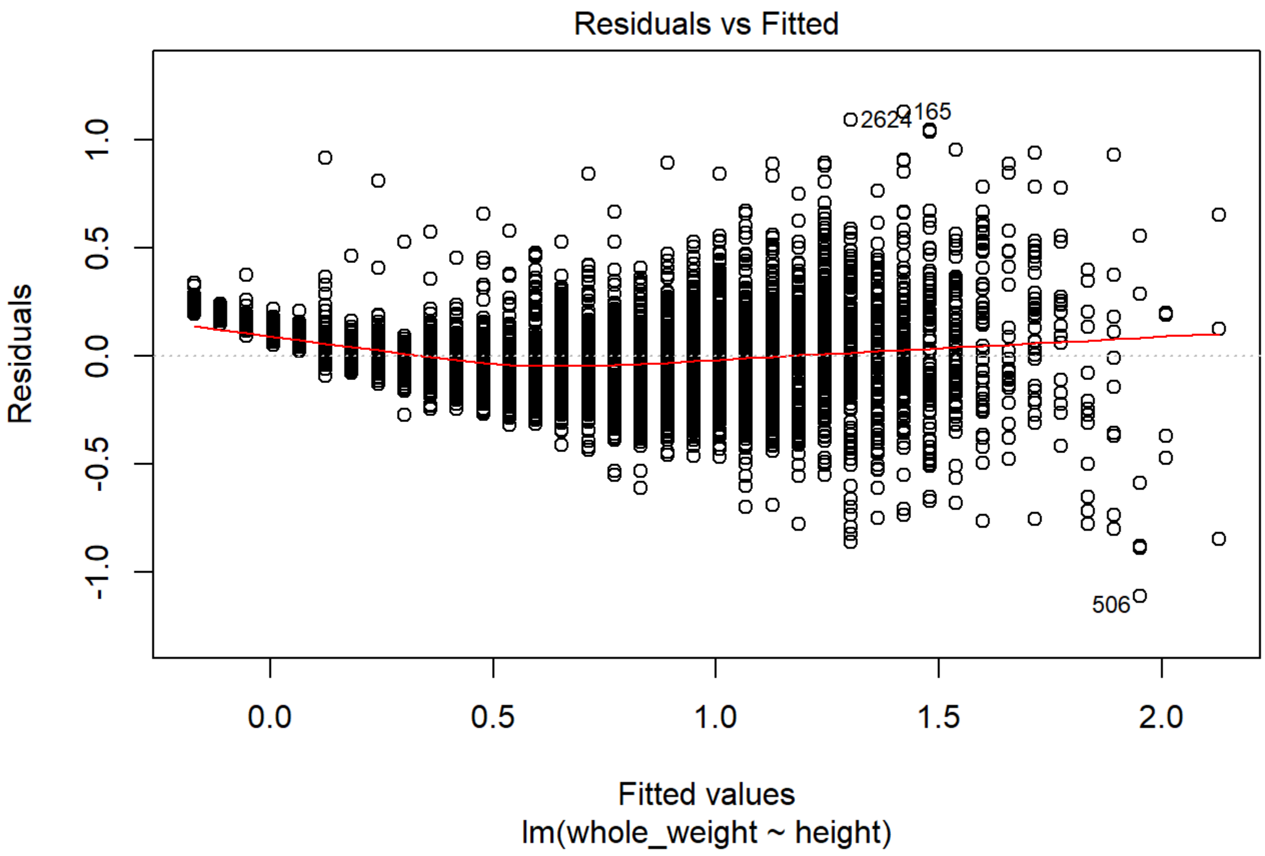

Residuals vs Leverage
Standardized residuals
10
0
-10
-20
-30
-40
1174
1417
2051
1
0.5
0.5
1
Cook's distance
0.00
0.02
0.04
0.06
0.08
0.10
0.12
Leverage
lm(whole_weight ~ height)

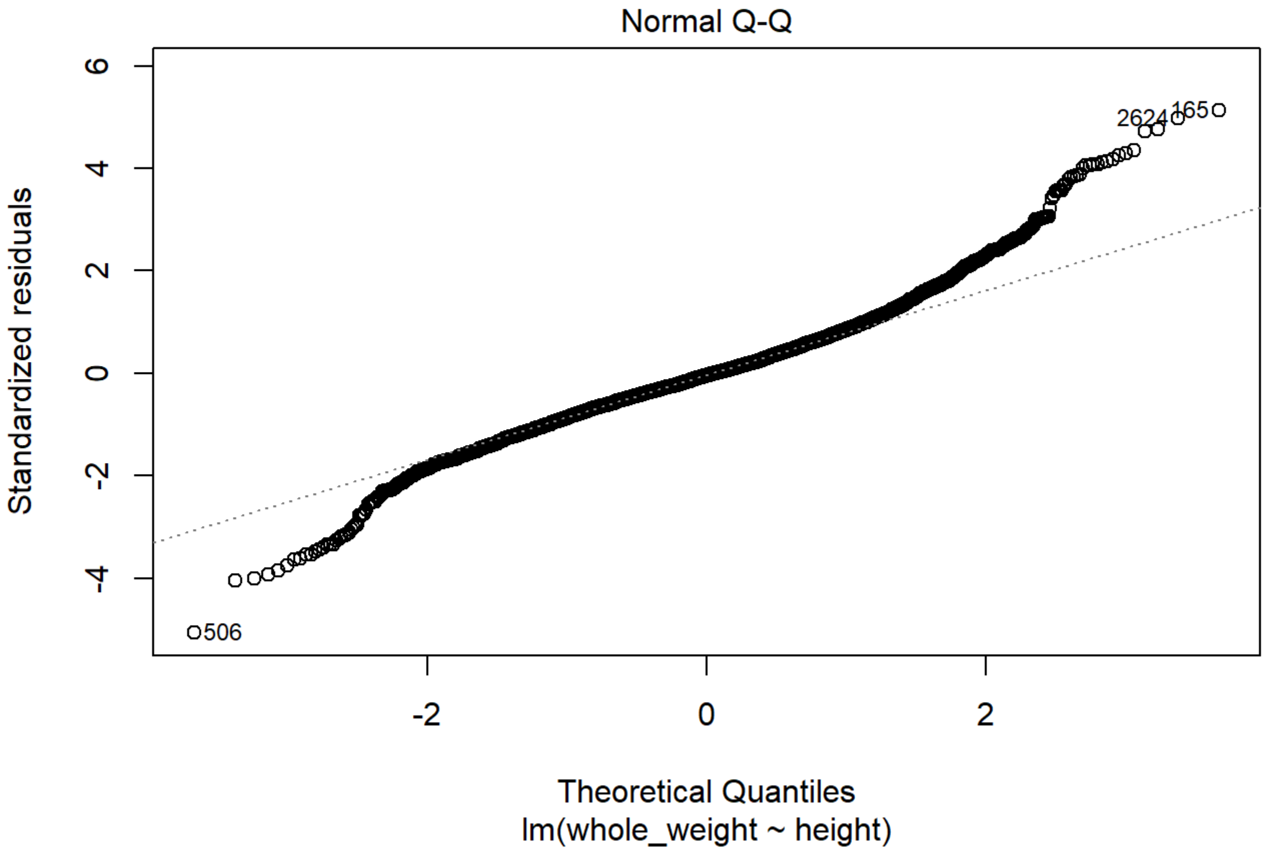
###Выбросы

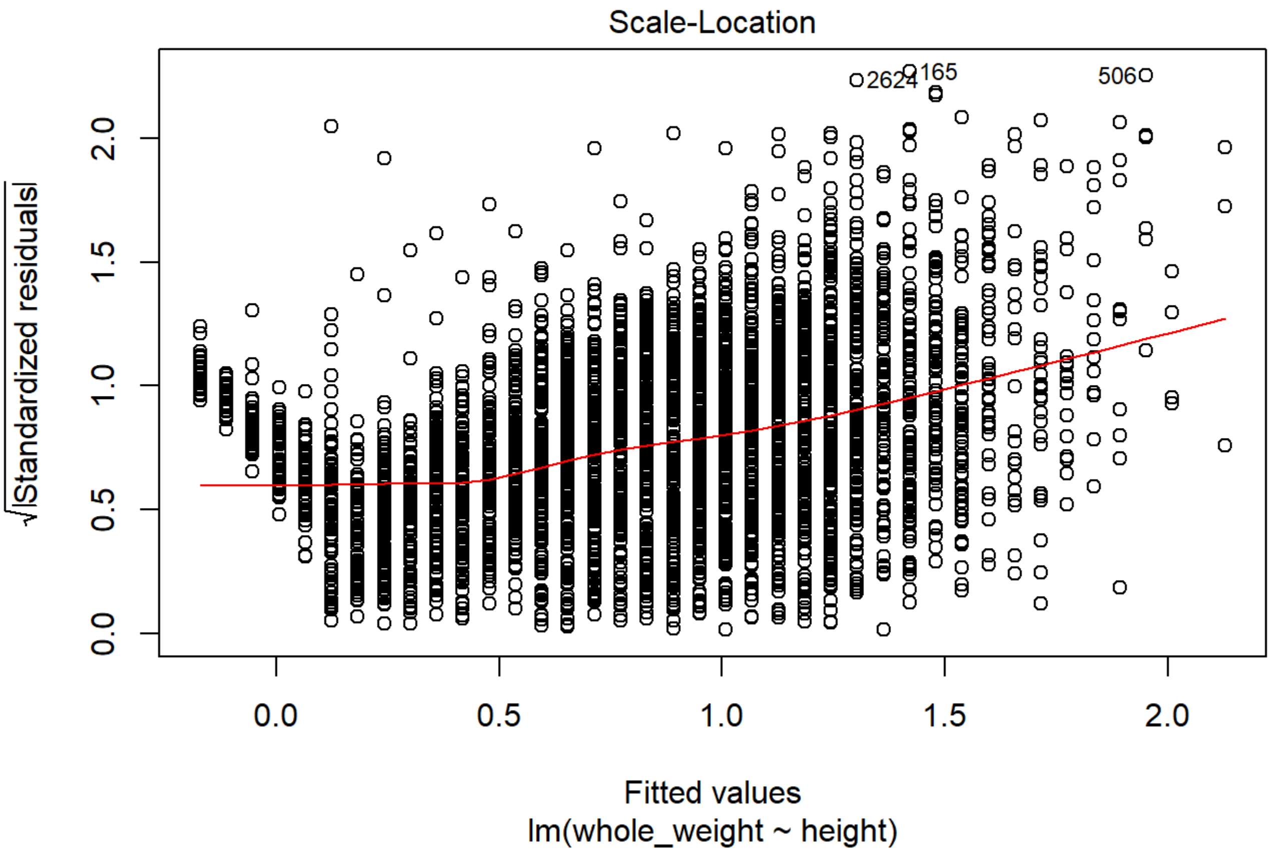
```
data.noout<-data[data$height<0.4&data$height>0.05,]
linear.model.wh.outlier<-lm(whole_weight~height,data=data.noout)
linear.model.wh.outlier
```

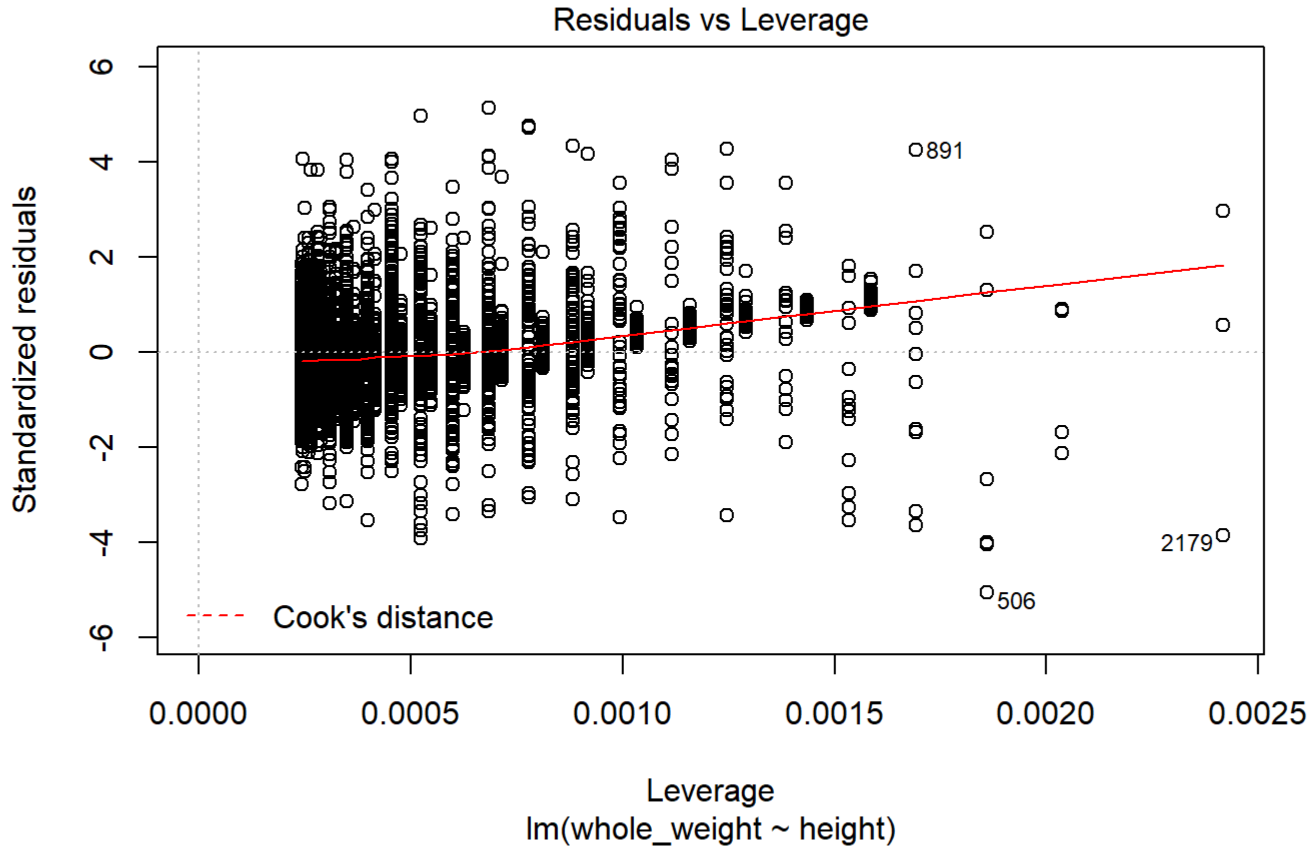
```
##
## Call:
## lm(formula = whole_weight ~ height, data = data.noout)
##
## Coefficients:
## (Intercept)       height
##     -0.8202      11.7954
```

```
plot(linear.model.wh.outlier)
```


Residuals vs Fitted
Residuals
1.0
0.5
0.0
-0.5
-1.0
0.0
0.5
1.0
1.5
2.0
Fitted values
lm(whole_weight ~ height)
2624
165
506


Normal Q-Q
Standardized residuals
6
4
2
0
-2
-4
-2
0
2
Theoretical Quantiles
lm(whole_weight ~ height)
2624
165
506





##Зависимость веса от высоты и диаметра

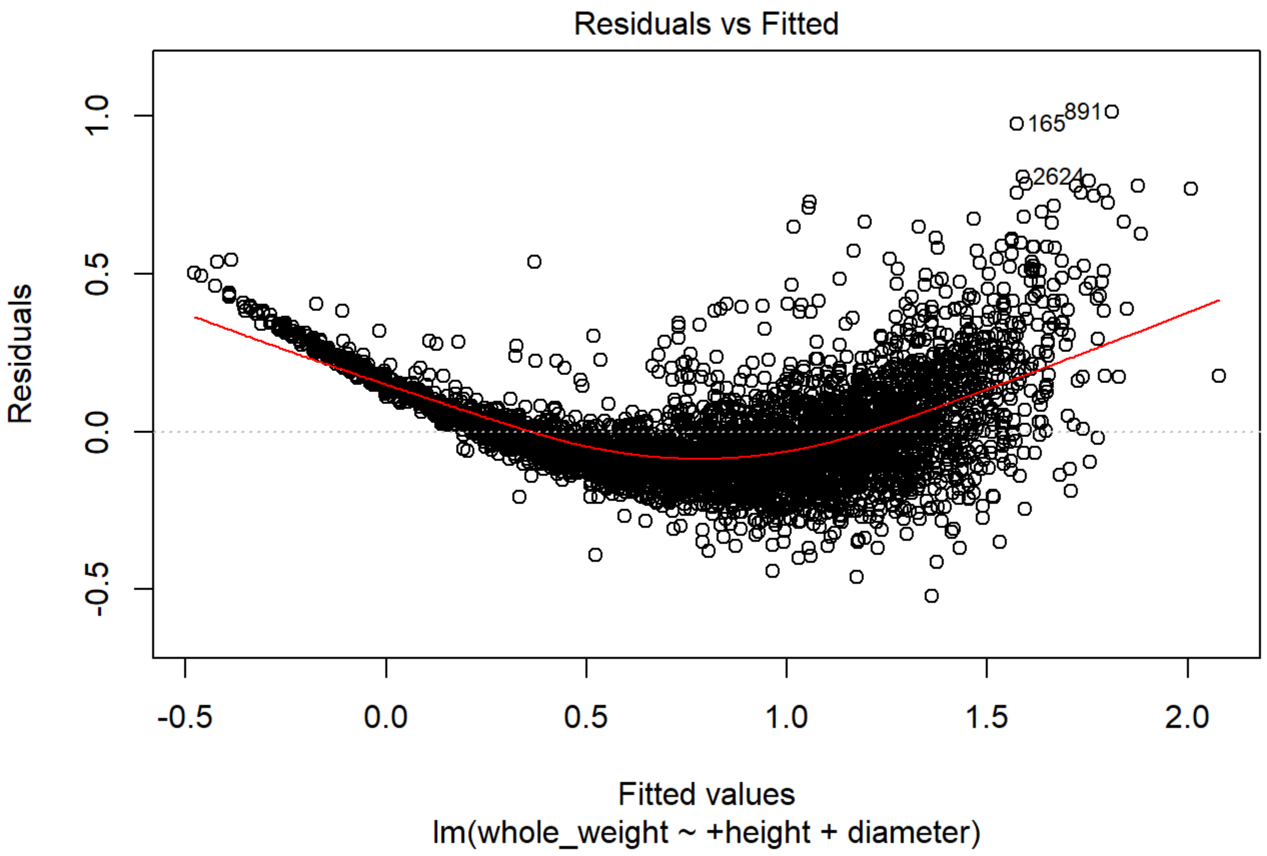
```
linear.model.w.hd<-lm(whole_weight~+height+diameter,data=data.noout)
linear.model.w.hd
```

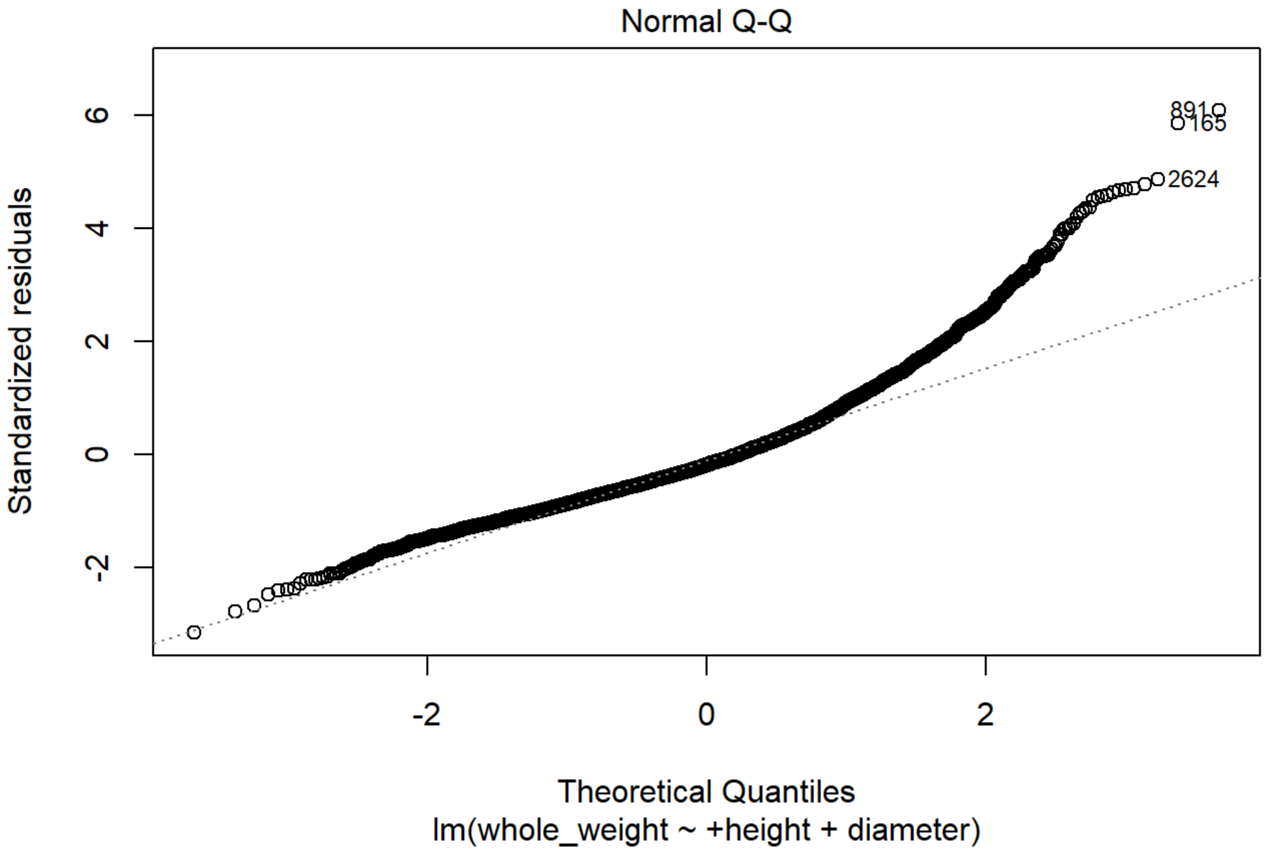
```
##
## Call:
## lm(formula = whole_weight ~ +height + diameter, data = data.noout)
##
## Coefficients:
## (Intercept)       height     diameter
##      -1.120        3.763        3.473
```

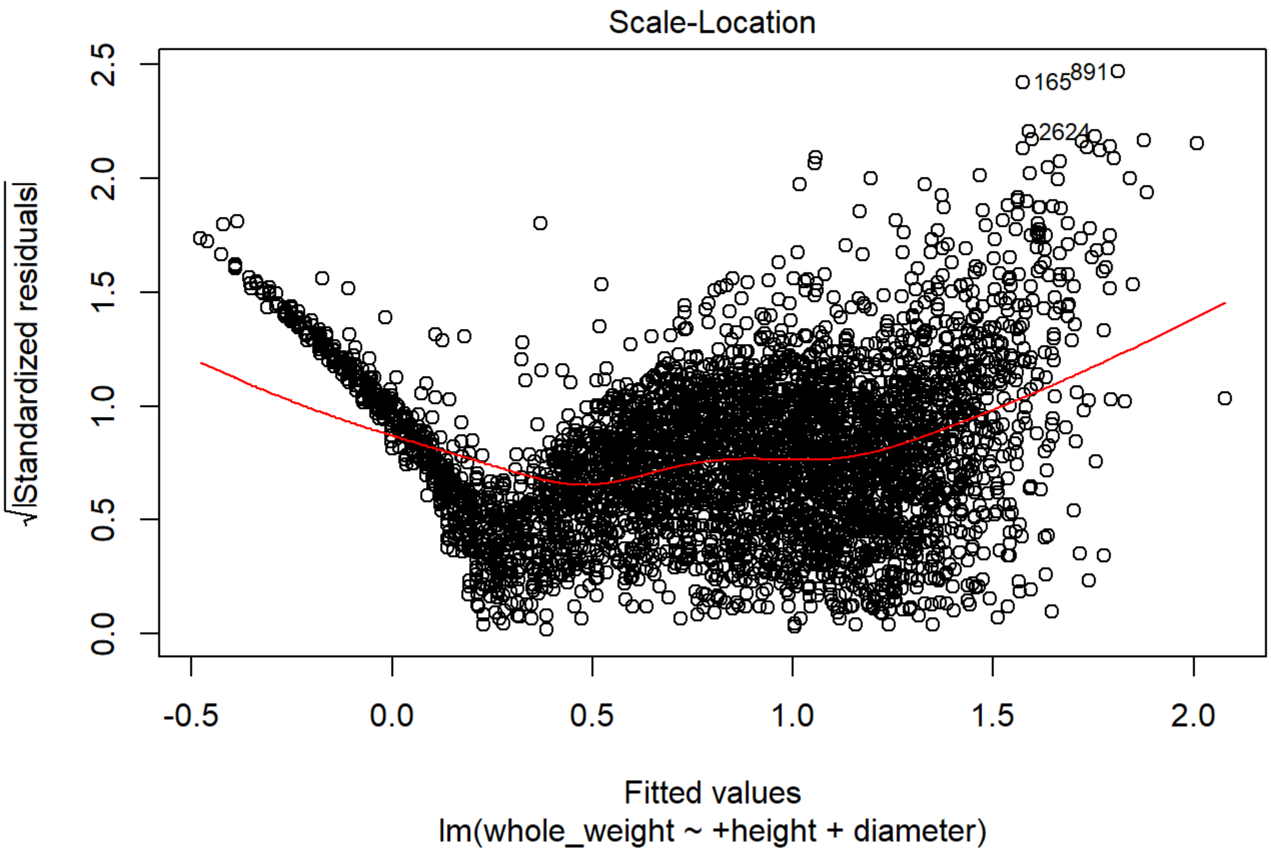
```
summary(linear.model.w.hd)
```

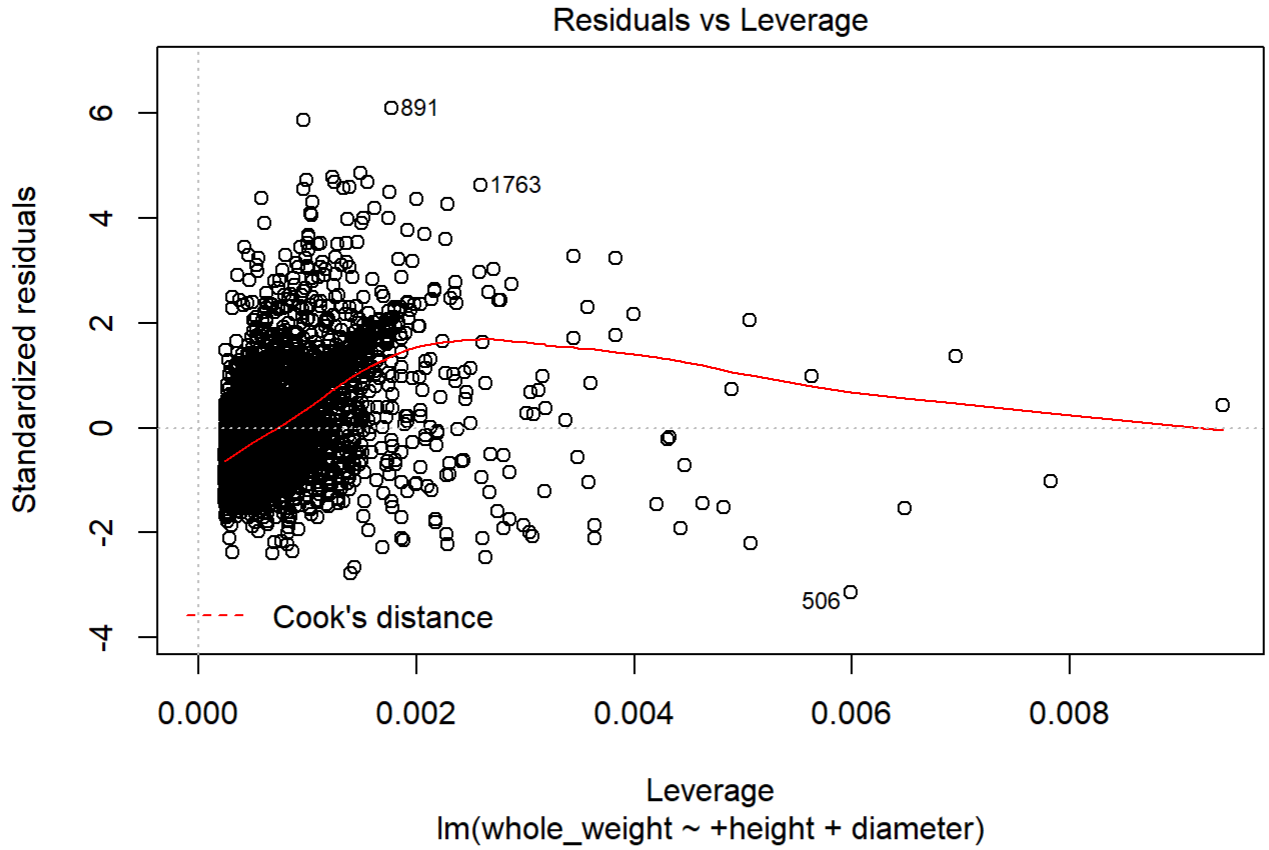
```
##
## Call:
## lm(formula = whole_weight ~ +height + diameter, data = data.noout)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52231 -0.10868 -0.03049  0.07438  1.01366
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.12005    0.01168  -95.91   <2e-16 ***
## height       3.76302    0.16194   23.24   <2e-16 ***
## diameter     3.47294    0.06292   55.20   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1664 on 4105 degrees of freedom
## Multiple R-squared:  0.8817, Adjusted R-squared:  0.8817
## F-statistic: 1.53e+04 on 2 and 4105 DF,  p-value: < 2.2e-16
```

```
plot(linear.model.w.hd)
```


Residuals vs Fitted
Residuals
1.0
0.5
0.0
-0.5
-0.5
0.0
0.5
1.0
1.5
2.0
165
891
2624
Fitted values
lm(whole_weight ~ +height + diameter)


Normal Q-Q
Standardized residuals
6
4
2
0
-2
-2
0
2
891
165
2624
Theoretical Quantiles
lm(whole_weight ~ +height + diameter)


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
-0.5
0.0
0.5
1.0
1.5
2.0
165
891
2624
Fitted values
lm(whole_weight ~ +height + diameter)


Residuals vs Leverage
Standardized residuals
-4
-2
0
2
4
6
0.000
0.002
0.004
0.006
0.008
891
1763
506
Cook's distance
Leverage
lm(whole_weight ~ +height + diameter)

##Всё

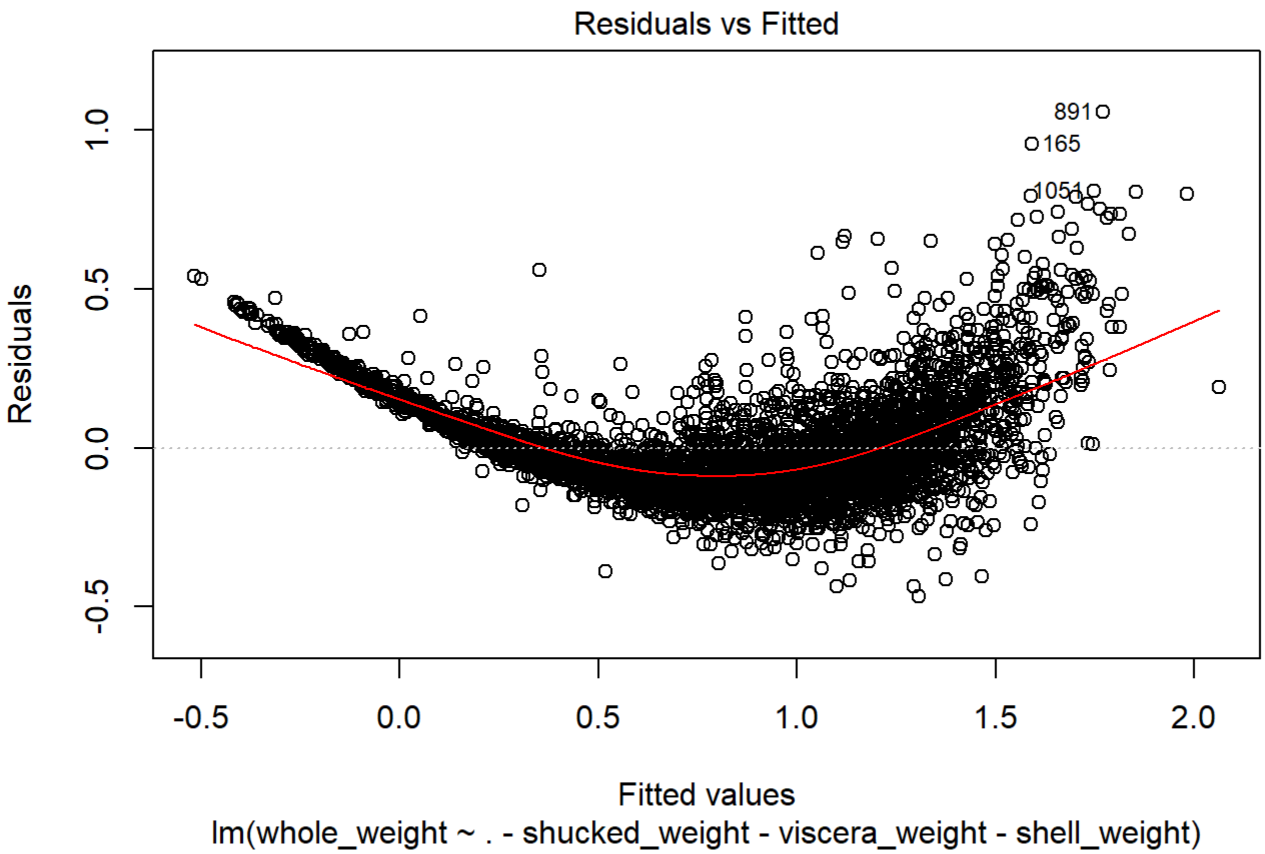
```
linear.model.all<-lm(whole_weight~.-shucked_weight-viscera_weight-shell_weight,data=data.noout)
linear.model.all
```

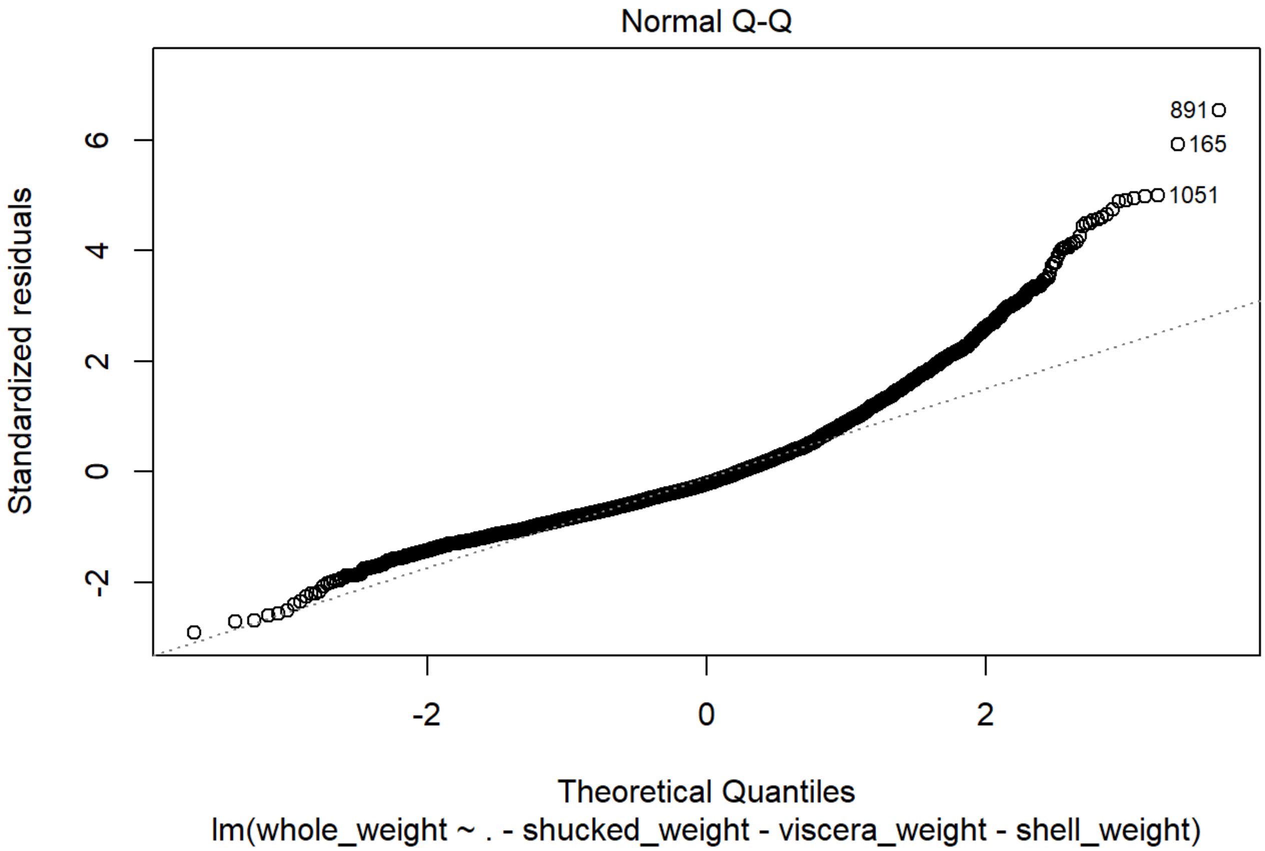
```
##
## Call:
## lm(formula = whole_weight ~ . - shucked_weight - viscera_weight -
##     shell_weight, data = data.noout)
##
## Coefficients:
## (Intercept)    sexInfant      sexMale       length     diameter
##   -1.157326    -0.021696     0.015360     1.911435     1.229664
##      height        rings
##    3.580197    -0.002294
```

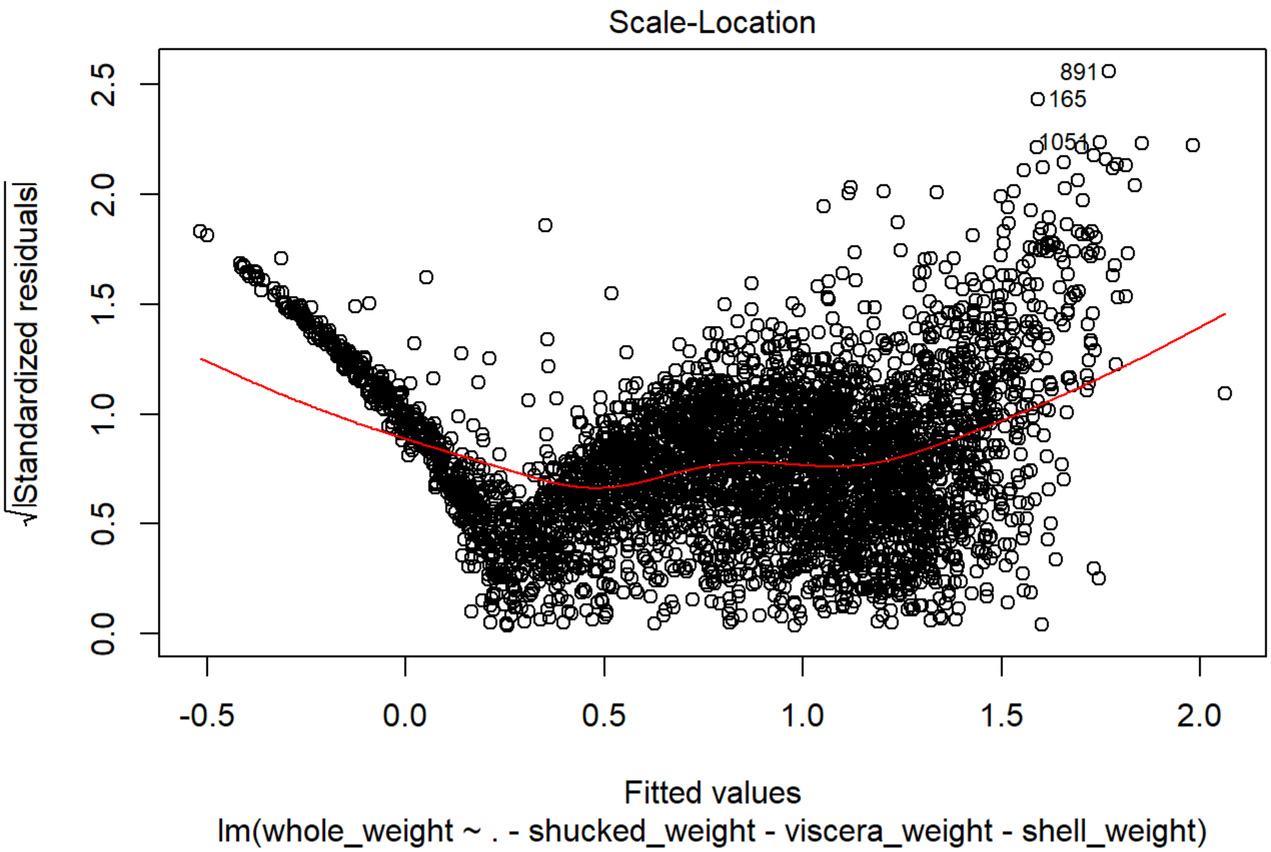
```
summary(linear.model.all)
```

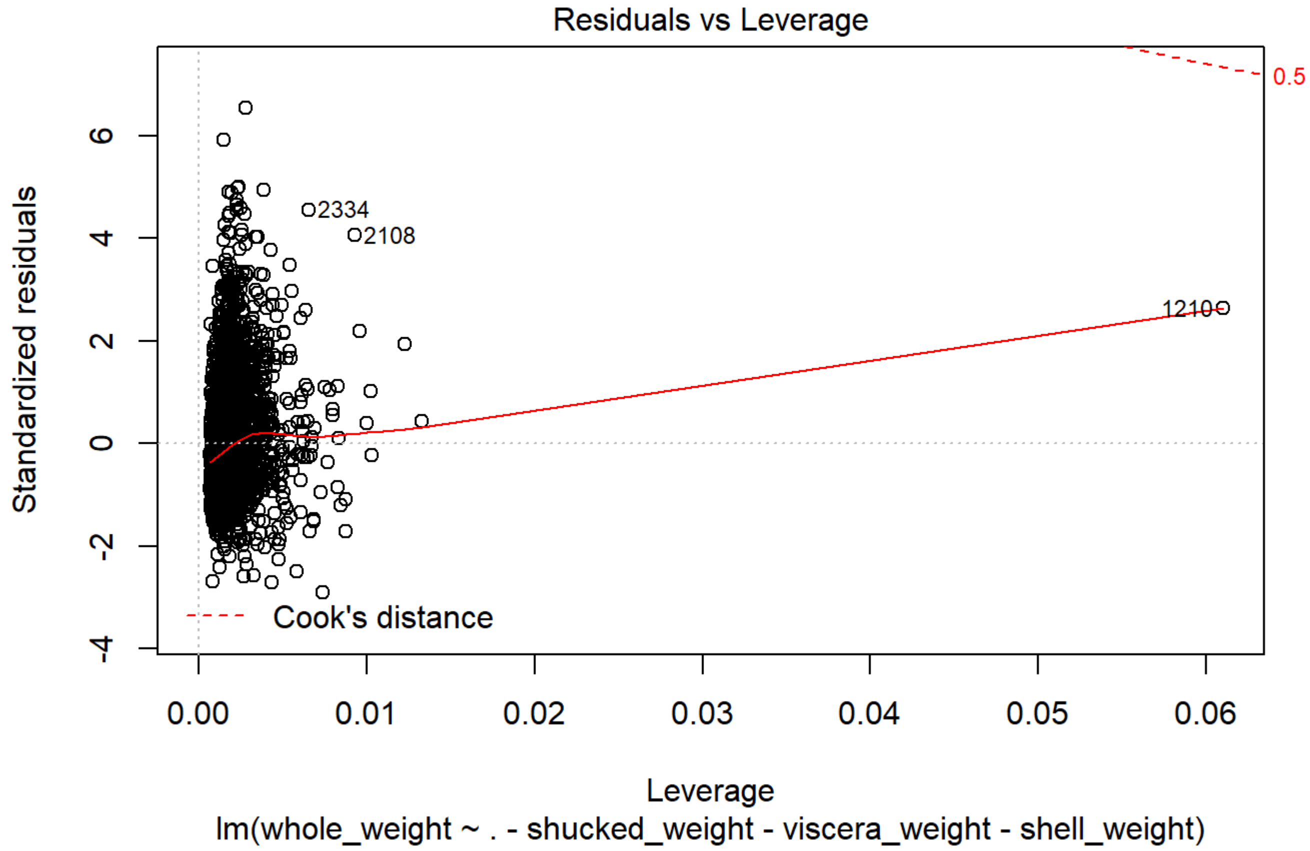
```
##
## Call:
## lm(formula = whole_weight ~ . - shucked_weight - viscera_weight -
##     shell_weight, data = data.noout)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46840 -0.10704 -0.03456  0.06938  1.05602
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.1573263  0.0167308 -69.174  < 2e-16 ***
## sexInfant   -0.0216956  0.0075909  -2.858  0.00428 **
## sexMale      0.0153597  0.0061246   2.508  0.01219 *
## length       1.9114347  0.1307500  14.619  < 2e-16 ***
## diameter     1.2296643  0.1636835   7.512 7.08e-14 ***
## height       3.5801973  0.1647054  21.737  < 2e-16 ***
## rings       -0.0022938  0.0009993  -2.295  0.02176 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1617 on 4101 degrees of freedom
## Multiple R-squared:  0.8885, Adjusted R-squared:  0.8883
## F-statistic:  5446 on 6 and 4101 DF,  p-value: < 2.2e-16
```

```
plot(linear.model.all)
```


Residuals vs Fitted
Residuals
1.0
0.5
0.0
-0.5
-0.5
0.0
0.5
1.0
1.5
2.0
891
165
1051
Fitted values
lm(whole_weight ~ . - shucked_weight - viscera_weight - shell_weight)


Normal Q-Q
Standardized residuals
6
4
2
0
-2
-2
0
2
891
165
1051
Theoretical Quantiles
lm(whole_weight ~ . - shucked_weight - viscera_weight - shell_weight)





##Разделение данных на две случайных части

```
odds <- seq(1, nrow(data.noout), by=2)
data.in <- data.noout[odds,]
data.out <- data.noout[-odds,]
```

Подгон модели по первой части

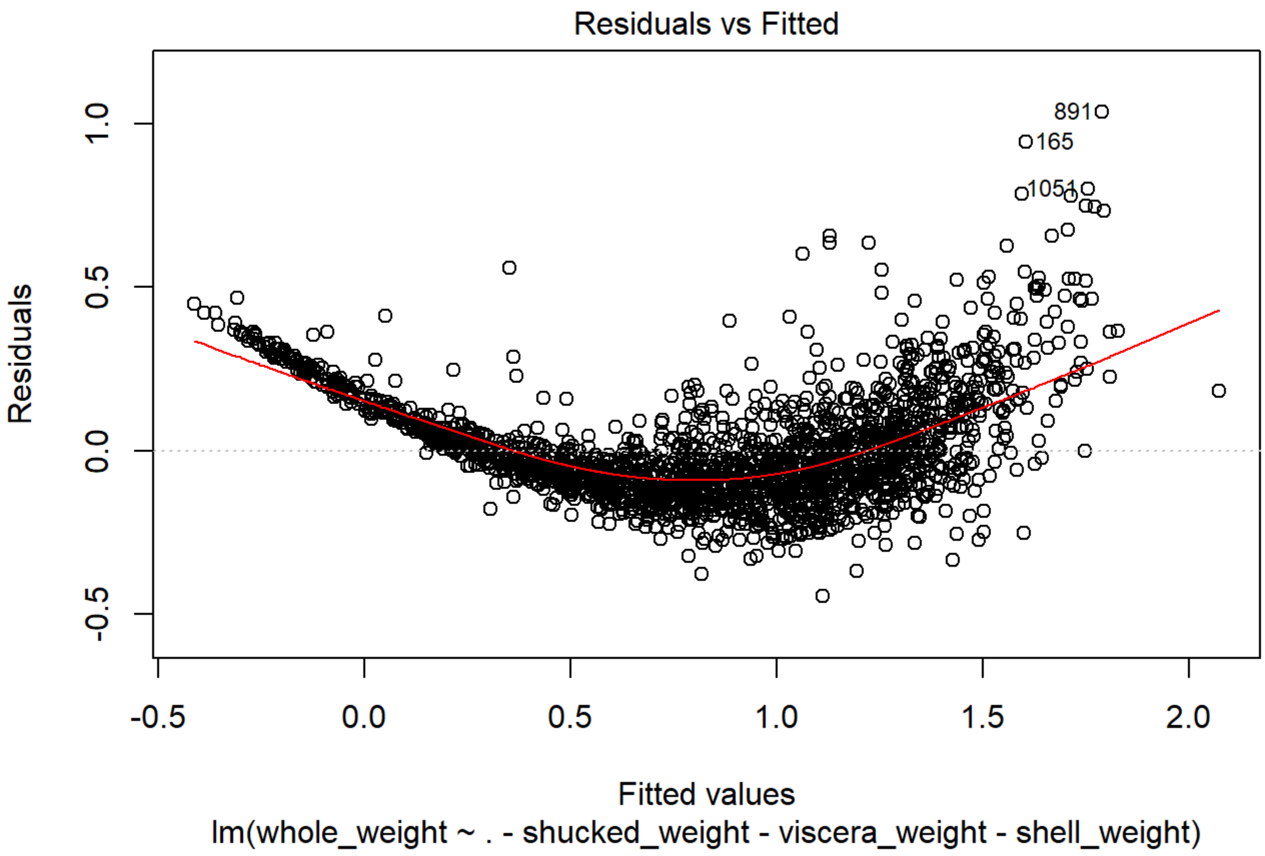
```
linear.model.all.half<-lm(whole_weight~.-shucked_weight-viscera_weight-shell_weight,data=data.i
n)
linear.model.all.half
```

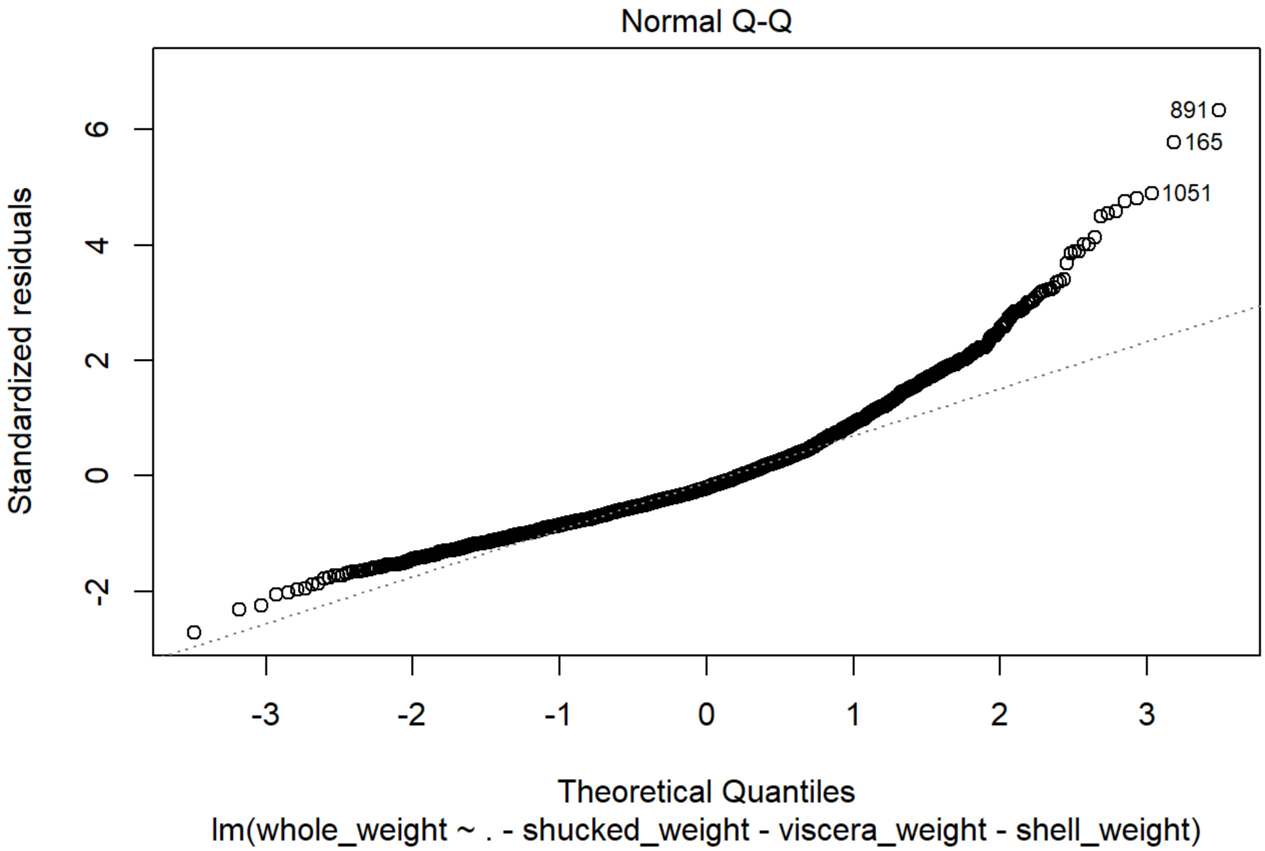
```
##
## Call:
## lm(formula = whole_weight ~ . - shucked_weight - viscera_weight -
##     shell_weight, data = data.in)
##
## Coefficients:
## (Intercept)    sexInfant      sexMale       length     diameter
##   -1.158210    -0.024309     0.024274     1.899176     1.165536
##      height        rings
##    3.812492    -0.001947
```

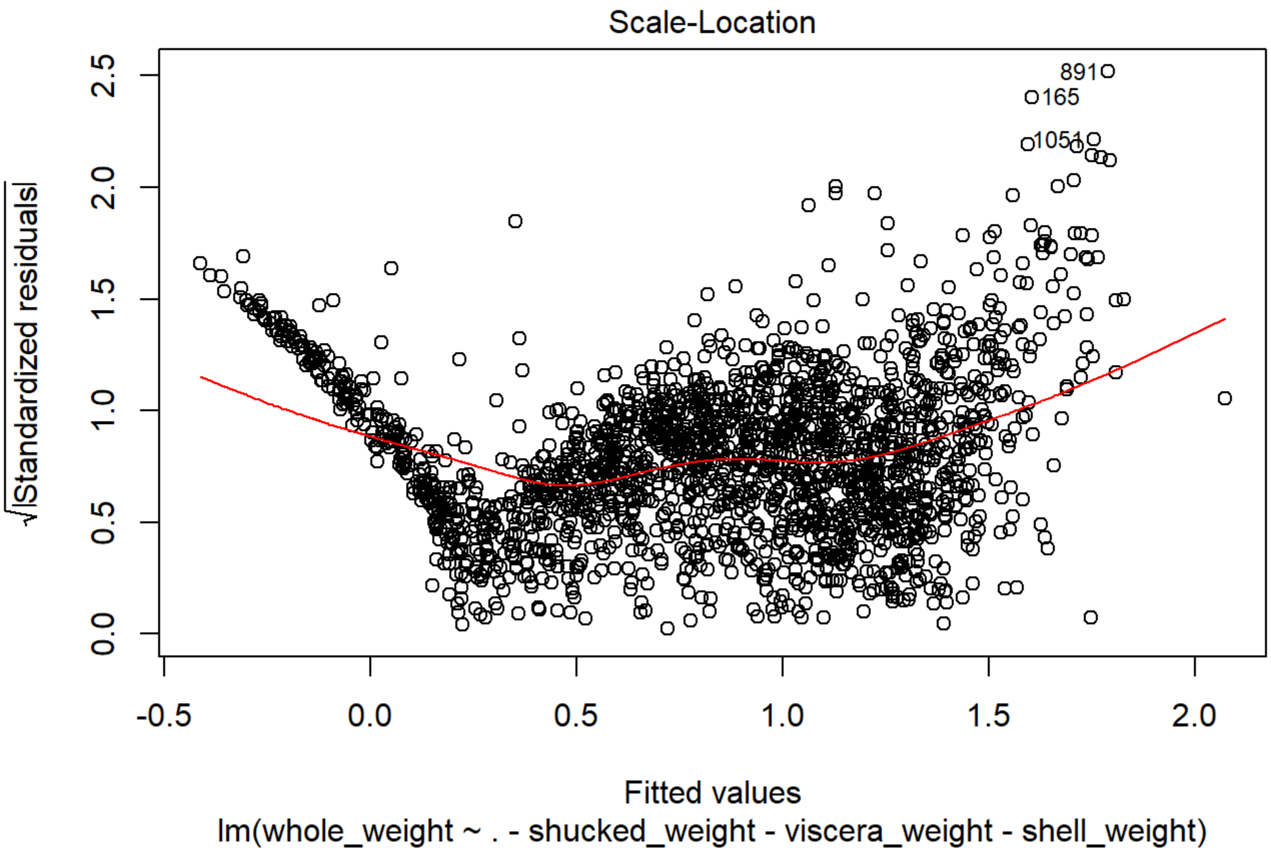
```
summary(linear.model.all.half)
```

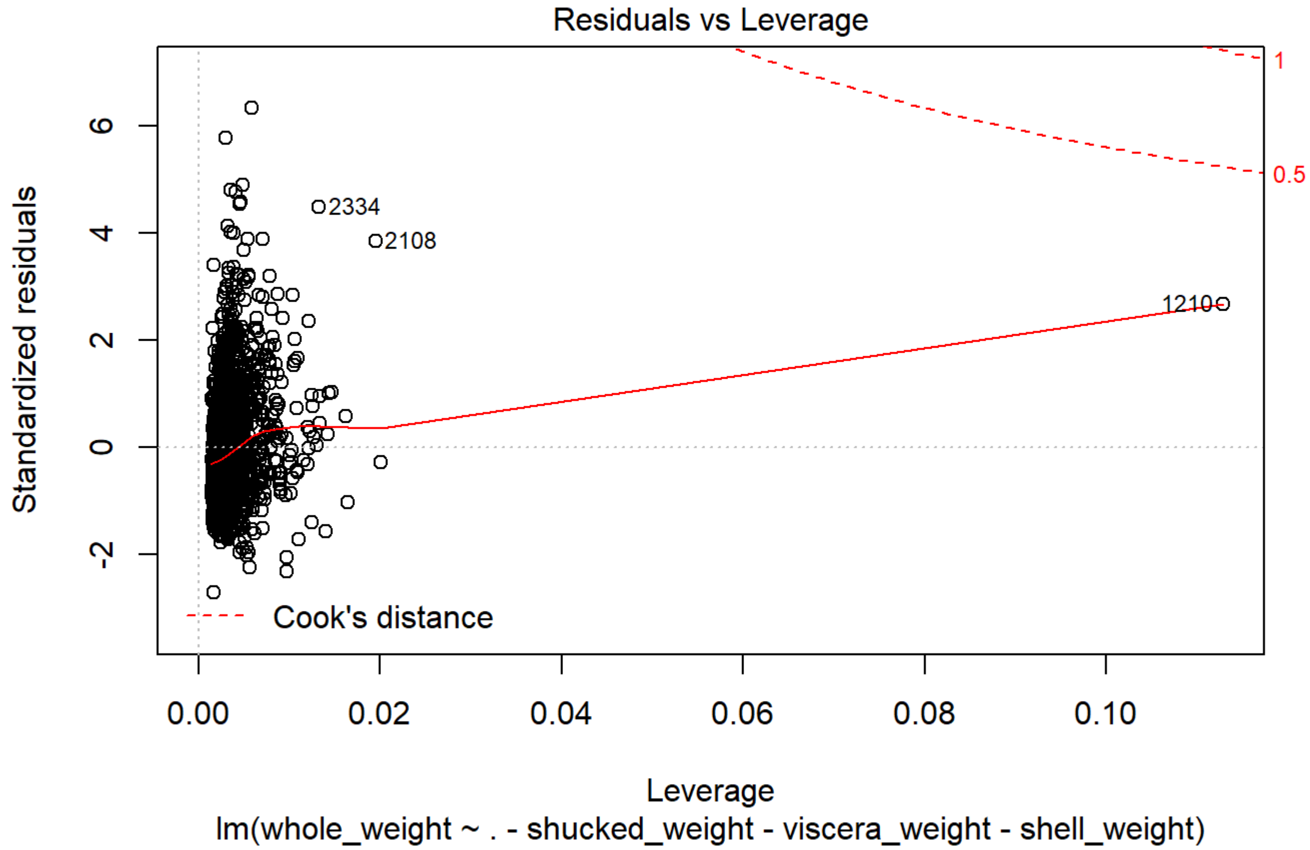
```
##
## Call:
## lm(formula = whole_weight ~ . - shucked_weight - viscera_weight -
##     shell_weight, data = data.in)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44573 -0.10895 -0.03478  0.07045  1.03577
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.158210   0.023804 -48.656  < 2e-16 ***
## sexInfant   -0.024309   0.010828  -2.245  0.02488 *
## sexMale      0.024274   0.008793   2.761  0.00582 **
## length       1.899176   0.180661  10.512  < 2e-16 ***
## diameter     1.165536   0.227570   5.122 3.31e-07 ***
## height       3.812492   0.239939  15.889  < 2e-16 ***
## rings       -0.001947   0.001461  -1.332  0.18288
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1639 on 2047 degrees of freedom
## Multiple R-squared:  0.8888, Adjusted R-squared:  0.8885
## F-statistic:  2727 on 6 and 2047 DF,  p-value: < 2.2e-16
```

```
plot(linear.model.all.half)
```


Residuals vs Fitted
Residuals
1.0
0.5
0.0
-0.5
-0.5
0.0
0.5
1.0
1.5
2.0
891
165
1051
Fitted values
lm(whole_weight ~ . - shucked_weight - viscera_weight - shell_weight)

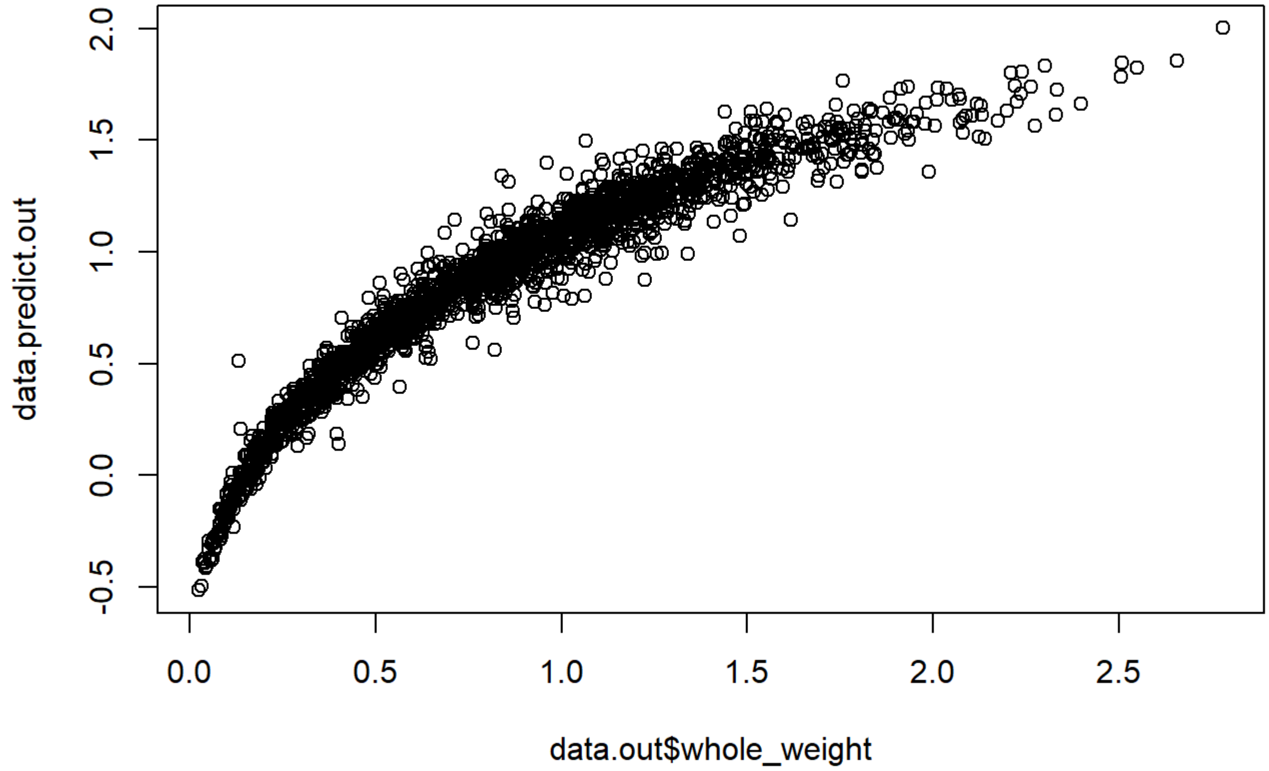

Normal Q-Q
Standardized residuals
6
4
2
0
-2
-3
-2
-1
0
1
2
3
891
165
1051
Theoretical Quantiles
lm(whole_weight ~ . - shucked_weight - viscera_weight - shell_weight)





##Прогноз значений во второй части

```
data.predict.out <- predict (linear.model.all.half, data.out)
plot (data.out$whole_weight, data.predict.out)
```



##Проверка качества прогноза Предсказание значений на известном наборе данных - in

```
data.predict.in <- predict (linear.model.all.half)
cor (data.in$whole_weight, data.predict.in) #почти 1
```

```
## [1] 0.9427599
```

Предсказание значений на неизвестном наборе данных - out

```
cor (data.out$whole_weight, data.predict.out) #почти 1 немного хуже
```

```
## [1] 0.9424124
```