

Denizkulağının yaşı(abalone), kabuğun koni boyunca kesilmesi, boyanması ve mikroskopta halka sayısının sayılmasıyla belirlenir - bu sıkıcı ve zaman alıcı bir iştir. Elde edilmesi daha kolay olan diğer ölçümler yaşı tahmin etmek için kullanılır.

NOT!! : Denizkulağının yaşı = *rings* + 1.5

Veri setindeki değişkenler ve özellikleri:

Name / Data Type / Measurement Unit / Description

Sex / nominal / – / M, F, and I (infant) Length / continuous / mm / Longest shell measurement Diameter / continuous / mm / perpendicular to length Height / continuous / mm / with meat in shell Whole weight / continuous / grams / whole abalone Shucked weight / continuous / grams / weight of meat Viscera weight / continuous / grams / gut weight (after bleeding) Shell weight / continuous / grams / after being dried Rings / integer / – / +1.5 gives the age in years

Gerekli olabilecek paketler:

```
options(warn=-1)

library(ggplot2)
library(dplyr)

##
## Attaching package: 'dplyr'

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

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

library(broom)
library(ggpubr)
library(ISLR)
library(caret)

## Zorunlu paket yükleniyor: lattice

library(car)

## Zorunlu paket yükleniyor: carData

##
## Attaching package: 'car'

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

library(lmtest)

## Zorunlu paket yükleniyor: zoo

##
## Attaching package: 'zoo'

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

library(olsrr)

##
## Attaching package: 'olsrr'
```

```
## The following object is masked from 'package:datasets':
##
## rivers
```

Çalışacağımız veriyi çekelim:

```
library(readr)
abalone <- read_csv("C:/Users/Ahmet/Desktop/abalone_edited.csv")
```

```
## Rows: 4177 Columns: 9
## — Column specification —————
## Delimiter: ","
## chr (1): sex
## dbl (8): length, diameter, height, whole_weight, shucked_weight, viscera_vei...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
View(abalone)
names(abalone)
```

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

Veride kayıp gözlem olup olmadığını test ediyoruz.:

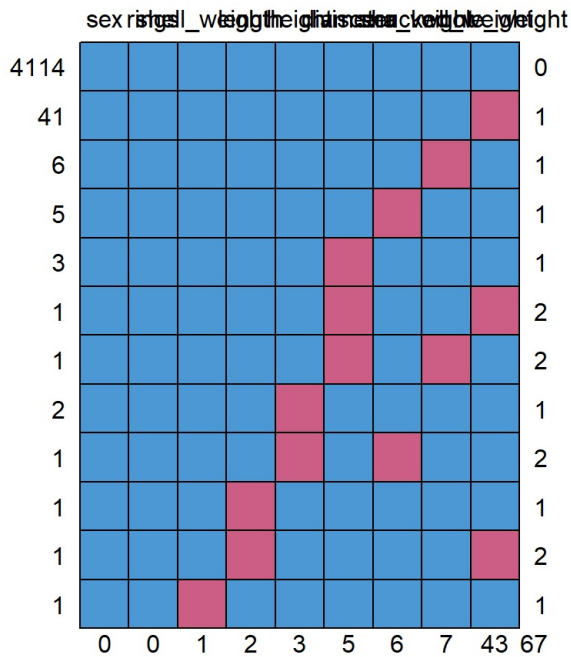
```
library(mice)
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##   filter
```

```
## The following objects are masked from 'package:base':
##
##   cbind, rbind
```

```
md.pattern(abalone)
```



```
##      sex rings shell_weight length height diameter viscera_veight
## 4114   1    1          1      1      1          1          1
## 41     1    1          1      1      1          1          1
## 6      1    1          1      1      1          1          1
## 5      1    1          1      1      1          1          0
## 3      1    1          1      1      1          0          1
## 1      1    1          1      1      1          0          1
## 1      1    1          1      1      1          0          1
## 2      1    1          1      1      0          1          1
## 1      1    1          1      1      0          1          0
## 1      1    1          1      0      1          1          1
## 1      1    1          1      0      1          1          1
## 1      1    1          0      1      1          1          1
##      0    0          1      2      3          5          6
##      shucked_weight whole_weight
## 4114          1          1 0
## 41           1          0 1
## 6            0          1 1
## 5            1          1 1
## 3            1          1 1
## 1            1          0 2
## 1            0          1 2
## 2            1          1 1
## 1            1          1 2
## 1            1          1 1
## 1            1          0 2
## 1            1          1 1
##      7          43 67
```

Tablodan da görüldüğü üzere veri setimiz içerisinde 67 tane eksik gözlem var bunu düzeltmek için doldurma işlemi uygulayacağız.

```
set.seed(123)
imputed<-mice(abalone,m=3)
```

```
##
## iter imp variable
## 1 1 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 1 2 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 1 3 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 2 1 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 2 2 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 2 3 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 3 1 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 3 2 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 3 3 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 4 1 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 4 2 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 4 3 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 5 1 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 5 2 length diameter height whole_weight shucked_weight viscera_veight shell_weight
## 5 3 length diameter height whole_weight shucked_weight viscera_veight shell_weight
```

```
imputed$imp
```

```
## $sex
## [1] 1 2 3
## <0 rows> (or 0-length row.names)
##
## $length
##      1      2      3
## 2069 0.555 0.56 0.575
## 3558 0.500 0.50 0.530
##
## $diameter
##      1      2      3
## 601  0.430 0.425 0.425
## 1276 0.385 0.385 0.370
## 1280 0.375 0.395 0.370
## 2067 0.435 0.410 0.415
## 4162 0.470 0.450 0.450
##
## $height
##      1      2      3
## 78   0.155 0.175 0.17
## 605  0.135 0.140 0.13
## 2065 0.140 0.130 0.13
```

```

##
## $whole_weight
##      1      2      3
## 10  0.8715 0.8570 0.8750
## 17  0.2655 0.2655 0.2790
## 18  0.4625 0.4170 0.4355
## 21  0.2500 0.2535 0.2500
## 72  0.3625 0.3610 0.3525
## 88  0.8320 0.8525 0.8380
## 99  0.5425 0.5215 0.5095
## 156 0.6965 0.6755 0.6875
## 161 1.1270 1.0935 1.0680
## 170 1.7100 1.6750 1.6750
## 174 0.7585 0.7200 0.7200
## 253 1.1085 1.0770 1.1420
## 256 1.0340 1.0175 1.0765
## 262 0.8675 0.8745 0.8635
## 270 0.5155 0.5220 0.4980
## 273 1.2085 1.1035 1.0530
## 424 0.1145 0.1110 0.1145
## 434 0.7200 0.7170 0.7365
## 598 0.9300 0.9285 0.9480
## 886 1.3830 1.3825 1.3520
## 890 1.7495 1.7925 1.7640
## 992 1.0980 1.0980 1.0835
## 1000 0.8325 0.8740 0.8020
## 1252 0.2500 0.2570 0.2535
## 1256 0.3140 0.3060 0.3030
## 1269 0.3620 0.3650 0.3465
## 1276 0.5600 0.5150 0.5130
## 1439 0.2715 0.2745 0.2875
## 1443 0.2845 0.2825 0.2725
## 1777 0.9550 0.9640 0.9205
## 2069 0.8525 0.8455 0.8655
## 2406 1.1145 1.2135 1.1835
## 2568 0.4555 0.4355 0.4410
## 2814 0.0740 0.0740 0.0625
## 3003 1.4640 1.5220 1.4535
## 3189 2.1190 2.1275 2.0370
## 3538 0.2865 0.2865 0.3140
## 3542 0.4980 0.5240 0.5160
## 3555 0.6185 0.5915 0.6255
## 3564 0.9270 0.9215 0.9270
## 4021 1.4940 1.4985 1.4985
## 4025 0.1460 0.1485 0.1460
## 4174 0.9555 0.9545 0.9545
##
## $shucked_weight
##      1      2      3
## 91  0.3245 0.3245 0.3315
## 426 0.4700 0.4870 0.4865
## 435 0.1255 0.1280 0.1150
## 1440 0.1695 0.1965 0.1695
## 2067 0.3620 0.3155 0.3605
## 3557 0.2565 0.2655 0.2790
## 4172 0.4435 0.4340 0.4125
##
## $viscera_veight
##      1      2      3
## 98  0.1340 0.1505 0.1395
## 269 0.1075 0.1200 0.1265
## 605 0.1410 0.1195 0.1420
## 1445 0.0780 0.0930 0.1015
## 2072 0.1715 0.1580 0.1580
## 3560 0.3150 0.3490 0.2385
##
## $shell_weight
##      1      2      3
## 92 0.2295 0.2705 0.25
##
## $rings
## [1] 1 2 3
## <0 rows> (or 0-length row.names)

```

verileri imput edilmiş 2. veriler ile dolduralım:

```
abalone_imp <- complete(imputed, 2)
View(abalone_imp)
md.pattern(abalone_imp)
```

```
##  /\      /\
## {  \----' }
## {  0  0  }
## ==> V <== No need for mice. This data set is completely observed.
##  \  \ /  /
##  \  \----'
```



```
##      sex length diameter height whole_weight shucked_weight viscera_weight
## 4177  1      1          1      1          1          1          1
##      0      0          0      0          0          0          0
##      shell_weight rings
## 4177          1      1 0
##          0      0 0
```

Veri setimizdeki eksik gözlemleri doldurduk artık model oluşturmaya geçebiliriz.

Veri setimizi test ve eğitim olarak bölüceiz:

```
set.seed(123)
sampleIndex <- sample(1:nrow(abalone_imp), size=0.8*nrow(abalone_imp))

abtrain <- abalone_imp[sampleIndex,]
abtest <- abalone_imp[-sampleIndex,]
View(abtrain)
```

Model oluşturmadan önce değişkenler arasındaki korelasyona bakalım —>

```
abtrain_xsex <- abtrain[, -1] # sex değişkenimiz kategorik olduğu için burada çıkarıyorum
View(abtrain_xsex)
cor(abtrain_xsex)
```

```
##           length diameter    height whole_weight shucked_weight
## length      1.0000000 0.9868118 0.8188037    0.9234100    0.8976546
## diameter    0.9868118 1.0000000 0.8251334    0.9246736    0.8936983
## height      0.8188037 0.8251334 1.0000000    0.8089434    0.7664419
## whole_weight 0.9234100 0.9246736 0.8089434    1.0000000    0.9703658
## shucked_weight 0.8976546 0.8936983 0.7664419    0.9703658    1.0000000
## viscera_weight 0.9016530 0.8996008 0.7906647    0.9672826    0.9337856
## shell_weight 0.8962259 0.9050590 0.8074580    0.9555667    0.8844818
## rings       0.5598511 0.5787174 0.5538885    0.5396404    0.4223127
##           viscera_weight shell_weight    rings
## length      0.9016530    0.8962259 0.5598511
## diameter    0.8996008    0.9050590 0.5787174
## height      0.7906647    0.8074580 0.5538885
## whole_weight 0.9672826    0.9555667 0.5396404
## shucked_weight 0.9337856    0.8844818 0.4223127
## viscera_weight 1.0000000    0.9110390 0.5054206
## shell_weight 0.9110390    1.0000000 0.6268478
## rings       0.5054206    0.6268478 1.0000000
```

Korelasyon matrisi incelendiğinde bağımlı değişken (rings) ile bağımsız değişkenler arasındaki ilişkilerin çoğu pozitif yönlü görünüyor.

Matris incelendiğinde bağımsız değişkenler arasında ilişki olduğu görülmektedir. En çokta length ve diameter arasındaki yüksek oranda ilişkiyi görebiliyoruz

Bu da bize multicollinearity sorunu olabileceğini söyler. Bu sorunu inceleyeceğiz.

Model oluşturma

Tüm değişkenleri içeren bir modeli oluşturmakla başlıyoruz:

```
abtrain_model1 = lm(rings~.,data=abtrain) # Bütün değişkenleri içine aldığımız modeli oluşturuyoruz.
summary(abtrain_model1)
```

```
##
## Call:
## lm(formula = rings ~ ., data = abtrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.7986 -1.3138 -0.3230  0.8891 13.9282
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.70346    0.32087   11.542 < 2e-16 ***
## sexI           -0.86005    0.11403   -7.543 5.91e-14 ***
## sexM            0.06561    0.09278    0.707  0.479
## length         -0.50232    2.00773   -0.250  0.802
## diameter       12.19852    2.48327    4.912 9.44e-07 ***
## height         10.12712    1.65872    6.105 1.14e-09 ***
## whole_weight    8.94602    0.80889   11.060 < 2e-16 ***
## shucked_weight -20.26549    0.93070  -21.774 < 2e-16 ***
## viscera_weight -10.59531    1.45277   -7.293 3.76e-13 ***
## shell_weight    8.93500    1.24823    7.158 1.00e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.185 on 3331 degrees of freedom
## Multiple R-squared:  0.5421, Adjusted R-squared:  0.5408
## F-statistic: 438.1 on 9 and 3331 DF, p-value: < 2.2e-16
```

Hipotez testlerimiz için α yı 0.05 olarak alacağım

Test sonucundan $R^2 = 0.5408$ olduğunu, p değerinin $2.2 * 10^{-16}$ 'dan küçük olduğunu görebiliriz ve bu, α , 0.05'in normal anlamlı değerlerinden daha düşüktür. Yani sıfır hipotezini reddediyoruz.

Katkı modelini tüm tahmin edicilere uydurduktan sonra, Test istatistiklerinin length hariç tüm değişkenleri anlamlı olarak gösterdiğini görebiliriz.

Veri setinin korelasyonuna baktığımızda değişkenler arasında çoklu bağlantı sorunu olabileceğini götmüştük. Veri seti ile Çoklu bağlantı sorununun varlığını bulmak için VIF değerini hesaplayacağız;

```
vif(abtrain_model1)
```

##		GVIF	Df	GVIF^(1/(2*Df))
## sex	1.541269	2		1.114217
## length	40.687614	1		6.378684
## diameter	42.644985	1		6.530313
## height	3.399414	1		1.843750
## whole_weight	108.413482	1		10.412180
## shucked_weight	29.271778	1		5.410340
## viscera_weight	17.494967	1		4.182698
## shell_weight	20.935449	1		4.575527

Tüm değişkenler için VIF faktörüne baktık ve Cinsiyet ve Boy dışında tüm tahmin edicilerin çoklu bağlantı sorunu olduğu görülüyor, çoğu 10'dan büyük bunları kontrol edeceğiz. VE whole_weight değişkeninin en yüksek VIF değerine sahip olduğunu görebiliyoruz.

whole_weight i çıkardığımızda VIF değeri nasıl olacak gözlemleyelim;

```
abtrain_model1_without_whole_weight <- lm(rings~.-whole_weight,data=abtrain) # whole_weight i çıkardığımız model
vif(abtrain_model1_without_whole_weight)
```

##		GVIF	Df	GVIF^(1/(2*Df))
## sex	1.537832	2		1.113595
## length	40.678264	1		6.377951
## diameter	42.620151	1		6.528411
## height	3.399022	1		1.843644
## shucked_weight	9.184175	1		3.030540
## viscera_weight	11.230437	1		3.351184
## shell_weight	7.965833	1		2.822381

Görüldüğü üzere whole_weight i çıkardığımızda shucked_weight, viscera_weight ve shell_weight üzerindeki vif değerlerinin düşüşünü gözlemleyebiliyoruz ama hâlâ daha 10 dan yüksek degerde olan vif degerleri var o yüzden birdahaki modelde diameter değişkeninide çıkararak kontrol ediyoruz ;

```
abtrain_model1_without_ww_diameter <- lm(rings~.-whole_weight-diameter,data=abtrain) # whole_weight ve diameterin
olmadığı model
vif(abtrain_model1_without_ww_diameter)
```

##		GVIF	Df	GVIF^(1/(2*Df))
## sex	1.520748	2		1.110489
## length	7.874349	1		2.806127
## height	3.363630	1		1.834020
## shucked_weight	9.182537	1		3.030270
## viscera_weight	11.215823	1		3.349003
## shell_weight	7.572629	1		2.751841

Şuan değerlerden görüldüğü üzere VIF puanlarımda çok düştü fakat ne kadar düşse de R2 dede düşüş yaşandı bu bazı değişkenlerin modelde kalması gerektiğini gösteriyor.

Şimdi son bulduğumuz modele bakalım;

```
abtrain_model2 = lm(rings~.-whole_weight -diameter,data=abtrain) # karmaşık olmaması adına model2 olarak adlandır
ıyorum
summary(abtrain_model2)
```

```
##
## Call:
## lm(formula = rings ~ . - whole_weight - diameter, data = abtrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.9544  -1.3345  -0.3404   0.8777  16.1004
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.69221    0.32750   11.274 < 2e-16 ***
## sexI           -0.97756    0.11579   -8.443 < 2e-16 ***
## sexM            0.04191    0.09477    0.442  0.658
## length          8.50059    0.90253    9.419 < 2e-16 ***
## height         11.20086    1.68600    6.643 3.57e-11 ***
## shucked_weight -11.70258    0.53266  -21.970 < 2e-16 ***
## viscera_veight  -1.19843    1.18860   -1.008  0.313
## shell_weight   20.68692    0.76711   26.967 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.233 on 3333 degrees of freedom
## Multiple R-squared:  0.5216, Adjusted R-squared:  0.5206
## F-statistic: 519.1 on 7 and 3333 DF,  p-value: < 2.2e-16
```

Bu modele baktığımızda önceki modele göre lengthin anlamlı viscera_veight in ise anlamsız olduğunu görüyoruz.

Birde bu modelden viscevara_veighti çıkararak bakalım nasıl değiştiğini;

```
abtrain_model2.1 = lm(rings~.-whole_weight -diameter -viscera_veight,data=abtrain)
summary(abtrain_model2.1)
```

```
##
## Call:
## lm(formula = rings ~ . - whole_weight - diameter - viscera_veight,
##      data = abtrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.8049  -1.3403  -0.3466   0.8780  16.1378
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.73317    0.32497   11.488 < 2e-16 ***
## sexI           -0.96373    0.11497   -8.382 < 2e-16 ***
## sexM            0.04752    0.09460    0.502  0.616
## length          8.36026    0.89174    9.375 < 2e-16 ***
## height         11.08747    1.68225    6.591 5.06e-11 ***
## shucked_weight -12.00691    0.43888  -27.358 < 2e-16 ***
## shell_weight   20.41384    0.71772   28.443 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.233 on 3334 degrees of freedom
## Multiple R-squared:  0.5214, Adjusted R-squared:  0.5206
## F-statistic: 605.4 on 6 and 3334 DF,  p-value: < 2.2e-16
```

R2 ve residuallerde bi değişikliğin olmadığını görebiliyoruz. Bu 2 modelin AIC, BIC ve RMSE sine bakarak hangisini kullanacağımıza karar verelim.

```
AIC(abtrain_model2, k=8)
```

```
## [1] 14911.85
```

```
BIC(abtrain_model2)
```

```
## [1] 14912.88
```

```
sqrt(sum(abtrain_model2$residuals**2)/length(abtrain_model2$residuals)) # RMSE yi ölçüyoruz burada
```

```
## [1] 2.229844
```

```
AIC(abtrain_model2.1, k=7)
```



```
## [1] 14896.87
```

```
BIC(abtrain_model2.1)
```

```
## [1] 14905.79
```

```
sqrt(sum(abtrain_model2.1$residuals**2)/length(abtrain_model2.1$residuals))
```

```
## [1] 2.230184
```

sonuçlara göre abtrain_model2.1 in AIC ve BIC değerleri iyi abtrain_model2 nin de RMSE si iyi burada fazla bir fark yok o yüzden viscera_veightin olmadığı abtrain_model2.1 i kullanacağım.

Bir de *StepWise* ile bulabileceğimiz modeli araştıralım;

```
# Burada bothside ile oluşturacağım
```

```
step(lm(rings~1, data=abtrain),direction = "both",  
      scope = ~sex+length+diameter+height+whole_weight+shucked_weight+viscera_veight+shell_weight)
```

```
## Start: AIC=7823.56  
## rings ~ 1  
##  
##           Df Sum of Sq  RSS    AIC  
## + shell_weight  1   13643.3 21078 6158.0  
## + diameter     1   11628.7 23093 6463.0  
## + length       1   10882.8 23839 6569.2  
## + height       1   10652.2 24069 6601.3  
## + whole_weight  1   10111.3 24610 6675.6  
## + viscera_veight 1    8869.6 25852 6840.0  
## + sex          2    6859.3 27862 7092.2  
## + shucked_weight 1    6192.5 28529 7169.3  
## <none>                34721 7823.6  
##  
## Step: AIC=6157.99  
## rings ~ shell_weight  
##  
##           Df Sum of Sq  RSS    AIC  
## + shucked_weight  1    2784.3 18294 5686.7  
## + whole_weight    1    1407.7 19670 5929.0  
## + viscera_veight  1     880.6 20197 6017.4  
## + sex             2     507.6 20570 6080.5  
## + height          1     227.3 20851 6123.8  
## + diameter        1      24.9 21053 6156.0  
## <none>                21078 6158.0  
## + length          1       0.7 21077 6159.9  
## - shell_weight     1   13643.3 34721 7823.6  
##  
## Step: AIC=5686.66  
## rings ~ shell_weight + shucked_weight  
##  
##           Df Sum of Sq  RSS    AIC  
## + diameter     1    1136.9 17157 5474.3  
## + length       1     905.9 17388 5519.0  
## + sex          2     752.9 17541 5550.3  
## + height       1     653.4 17640 5567.2  
## + whole_weight  1     643.7 17650 5569.0  
## + viscera_veight 1      52.9 18241 5679.0  
## <none>                18294 5686.7  
## - shucked_weight 1    2784.3 21078 6158.0  
## - shell_weight   1   10235.1 28529 7169.3  
##  
## Step: AIC=5474.29  
## rings ~ shell_weight + shucked_weight + diameter  
##  
##           Df Sum of Sq  RSS    AIC  
## + sex          2     486.7 16670 5382.1  
## + whole_weight  1     440.0 16717 5389.5  
## + height       1     233.0 16924 5430.6  
## <none>                17157 5474.3  
## + length       1       2.2 17155 5475.9  
## + viscera_veight 1       1.9 17155 5475.9  
## - diameter     1    1136.9 18294 5686.7  
## - shucked_weight 1    3896.3 21053 6156.0
```

```
## - shell_weight      1      4383.0 21540 6232.4
##
## Step:  AIC=5382.14
## rings ~ shell_weight + shucked_weight + diameter + sex
##
##              Df Sum of Sq  RSS    AIC
## + whole_weight  1      355.9 16314 5312.0
## + height        1      181.4 16489 5347.6
## <none>                                16670 5382.1
## + viscera_veight 1         0.9 16669 5384.0
## + length         1         0.3 16670 5384.1
## - sex            2      486.7 17157 5474.3
## - diameter       1      870.8 17541 5550.3
## - shucked_weight 1     3900.0 20570 6082.5
## - shell_weight   1     4087.7 20758 6112.8
##
## Step:  AIC=5312.03
## rings ~ shell_weight + shucked_weight + diameter + sex + whole_weight
##
##              Df Sum of Sq  RSS    AIC
## + viscera_veight 1      236.04 16078 5265.3
## + height         1      156.16 16158 5281.9
## <none>                                16314 5312.0
## + length         1         2.24 16312 5313.6
## - whole_weight   1      355.91 16670 5382.1
## - sex            2      402.63 16717 5389.5
## - shell_weight   1      516.90 16831 5414.2
## - diameter       1      736.74 17051 5457.6
## - shucked_weight 1     2094.45 18409 5713.6
##
## Step:  AIC=5265.34
## rings ~ shell_weight + shucked_weight + diameter + sex + whole_weight +
##          viscera_veight
##
##              Df Sum of Sq  RSS    AIC
## + height         1      177.64 15900 5230.2
## <none>                                16078 5265.3
## + length         1         0.01 16078 5267.3
## - viscera_veight 1      236.04 16314 5312.0
## - shell_weight   1      292.90 16371 5323.7
## - sex            2      435.39 16514 5350.6
## - whole_weight   1      591.06 16669 5384.0
## - diameter       1      818.38 16897 5429.2
## - shucked_weight 1     2314.09 18392 5712.6
##
## Step:  AIC=5230.22
## rings ~ shell_weight + shucked_weight + diameter + sex + whole_weight +
##          viscera_veight + height
##
##              Df Sum of Sq  RSS    AIC
## <none>                                15900 5230.2
## + length         1         0.30 15900 5232.2
## - height         1      177.64 16078 5265.3
## - shell_weight   1      245.59 16146 5279.4
## - viscera_veight 1      257.51 16158 5281.9
## - sex            2      392.75 16293 5307.7
## - diameter       1      541.85 16442 5340.2
## - whole_weight   1      584.39 16485 5348.8
## - shucked_weight 1     2281.87 18182 5676.3
```

```
##
## Call:
## lm(formula = rings ~ shell_weight + shucked_weight + diameter +
##     sex + whole_weight + viscera_veight + height, data = abtrain)
##
## Coefficients:
## (Intercept)      shell_weight  shucked_weight      diameter          sexI
##      3.67757         8.94693        -20.28407        11.64059        -0.86157
##          sexM    whole_weight  viscera_veight          height
##      0.06537         8.94909        -10.62778        10.11305
```

Modelden length çıktından sonra nihai model olarak shell_weight + shucked_weight + diameter + sex + whole_weight + viscera_veight + height olduğunu buluyoruz

Step wise tekniğinin oluşturduğu model üzerinde AIC BIC ve RMSE değerlerine bakalım;

```
# stepwise üzerinden çıkarılan modeli oluşturunuz
abtrain_model3 <- lm(rings ~ shell_weight + shucked_weight + diameter + sex + whole_weight + viscera_weight + height, data = abtrain)
summary(abtrain_model3)
```

```
##
## Call:
## lm(formula = rings ~ shell_weight + shucked_weight + diameter +
##     sex + whole_weight + viscera_weight + height, data = abtrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.7808 -1.3124 -0.3242  0.8859 13.9437
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.67757    0.30369   12.110 < 2e-16 ***
## shell_weight    8.94693    1.24715    7.174 8.94e-13 ***
## shucked_weight -20.28407    0.92760  -21.867 < 2e-16 ***
## diameter      11.64059    1.09241   10.656 < 2e-16 ***
## sexI           -0.86157    0.11385   -7.568 4.89e-14 ***
## sexM            0.06537    0.09276    0.705  0.481
## whole_weight    8.94909    0.80869   11.066 < 2e-16 ***
## viscera_weight -10.62778    1.44676   -7.346 2.56e-13 ***
## height         10.11305    1.65754    6.101 1.17e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.185 on 3332 degrees of freedom
## Multiple R-squared:  0.5421, Adjusted R-squared:  0.541
## F-statistic: 493 on 8 and 3332 DF,  p-value: < 2.2e-16
```

Bu bulduğumuz model en başta da oluşturduğumuz modelden anlamsız olan lenglhtin çıkarılmasıyla oluşmuş bir model ve bulduğumuz modeller arasında en yüksek R2 değerine sahip model.

BU model için birde AIC, BIC ve RMSE değerlerini analiz edilebilecek olan abtrain_model2.1 ile karşılaştıralım ;

```
AIC(abtrain_model3, k=9)
```

```
## [1] 14783.57
```

```
BIC(abtrain_model3)
```

```
## [1] 14774.71
```

```
sqrt(sum(abtrain_model3$residuals**2)/length(abtrain_model3$residuals))
```

```
## [1] 2.181558
```

```
AIC(abtrain_model2.1, k=7)
```

```
## [1] 14896.87
```

```
BIC(abtrain_model2.1)
```

```
## [1] 14905.79
```

```
sqrt(sum(abtrain_model2.1$residuals**2)/length(abtrain_model2.1$residuals))
```

```
## [1] 2.230184
```

Sonuca göre abtrain_model3 ün daha iyi değerlere sahip olduğunu görebiliyoruz.

Bir de şimdiye kadar oluşturduğumuz modellerden eğer var ise aykırı değerleri çıkararak modelleri güncelleyelim ve inceleyelim;

```
# abtrain_model1 için cook.distance

dist <- cooks.distance(abtrain_model1)
olcut1 <- mean(dist)*3
olcut2 <- 4/length(dist)
olcut1;olcut2
```

```
## [1] 0.007586995
```

```
## [1] 0.001197246
```

```
olcut1Index <- which(dist > olcut1)
olcut2Index <- which(dist > olcut2)
length(olcut1Index)
```

```
## [1] 24
```

```
length(olcut2Index)
```

```
## [1] 207
```

Burada iki ölçüt için birinde 24 değerinde ise 207 aykırı değer var burada fazla olanı seçiyorum.

```
abtrainrem_model1 <- abtrain[-olcut2Index,]
```

modeli güncelliyorum;

```
abtrain_model1rem = lm(rings~.-length,data=abtrainrem_model1)
summary(abtrain_model1rem)
```

```
##
## Call:
## lm(formula = rings ~ . - length, data = abtrainrem_model1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.6007 -1.1025 -0.2003  0.9110  7.0229
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.41720    0.24487   13.955 < 2e-16 ***
## sexI           -0.77127    0.08952   -8.616 < 2e-16 ***
## sexM            0.07195    0.07349    0.979  0.328
## diameter       9.54043    0.93966   10.153 < 2e-16 ***
## height        18.34597    2.07574    8.838 < 2e-16 ***
## whole_weight  10.28332    0.83700   12.286 < 2e-16 ***
## shucked_weight -20.79296    0.93189  -22.313 < 2e-16 ***
## viscera_weight -12.72319    1.30171   -9.774 < 2e-16 ***
## shell_weight   5.53420    1.25899    4.396 1.14e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.658 on 3125 degrees of freedom
## Multiple R-squared:  0.6183, Adjusted R-squared:  0.6174
## F-statistic: 632.9 on 8 and 3125 DF,  p-value: < 2.2e-16
```

Burada gördüğümüz üzere ilk abtrain_model1 den aykırıları çıkarınca R2 deki artışı görebiliyoruz, birde AIC, BIC ve RMSE leri karşılaştıralım;

```
AIC(abtrain_model1, k=10)
```

```
## [1] 14803.51
```

```
BIC(abtrain_model1)
```

```
## [1] 14782.76
```

```
sqrt(sum(abtrain_model1$residuals**2)/length(abtrain_model1$residuals))
```

```
## [1] 2.181537
```

```
AIC(abtrain_model1rem, k=10)
```

```
## [1] 12154.86
```

```
BIC(abtrain_model1rem)
```

```
## [1] 12135.36
```

```
sqrt(sum(abtrain_model1rem$residuals**2)/length(abtrain_model1rem$residuals))
```

```
## [1] 1.655826
```

Aykırı değerleri çıkardığımızda modelin ne kadar iyileştiğini gözlemleyebiliyoruz AIC, BIC ve RMSE de ki ciddi değişim fark edilebiliyor.

Aykırı değerlerin modeli ne kadar etkilediğini gördüğümüze göre öbür modellerimizden de aykırı değerleri çıkaralım;

abtrain_model2.1 i güncelliyoruz:

```
dist <- cooks.distance(abtrain_model2.1)
olcut1 <- mean(dist)*3
olcut2 <- 4/length(dist)
olcut1;olcut2
```

```
## [1] 0.0109705
```

```
## [1] 0.001197246
```

```
olcut1Index <- which(dist > olcut1)
olcut2Index <- which(dist > olcut2)
length(olcut1Index)
```

```
## [1] 9
```

```
length(olcut2Index)
```

```
## [1] 192
```

```
abtrainrem_model2.1 <- abtrain[-olcut2Index,] # aykırı değeri fazla olanı seçiyorum
```

```
abtrain_model2.1rem = lm(rings~.-whole_weight -diameter -viscera_veight,data=abtrainrem_model2.1)
summary(abtrain_model2.1rem)
```

```
##
## Call:
## lm(formula = rings ~ . - whole_weight - diameter - viscera_veight,
##     data = abtrainrem_model2.1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.4946 -1.1297 -0.2108  0.9025  6.0201
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.72336    0.25846   14.406 < 2e-16 ***
## sexI           -0.89067    0.08930   -9.974 < 2e-16 ***
## sexM            0.04007    0.07363    0.544  0.586
## length         6.02771    0.75566    7.977 2.08e-15 ***
## height        19.16849    2.07075    9.257 < 2e-16 ***
## shucked_weight -11.21975    0.37351 -30.039 < 2e-16 ***
## shell_weight   18.58540    0.68739   27.037 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.675 on 3142 degrees of freedom
## Multiple R-squared:  0.6042, Adjusted R-squared:  0.6035
## F-statistic: 799.4 on 6 and 3142 DF,  p-value: < 2.2e-16
```

abtrain_model3 ü de güncelleyelim:

```
dist <- cooks.distance(abtrain_model3)
olcut1 <- mean(dist)*3
olcut2 <- 4/length(dist)
olcut1;olcut2
```

```
## [1] 0.008230917
```

```
## [1] 0.001197246
```

```
olcut1Index <- which(dist > olcut1)
olcut2Index <- which(dist > olcut2)
length(olcut1Index)
```

```
## [1] 22
```

```
length(olcut2Index)
```

```
## [1] 213
```

```
abtrainrem_model3 <- abtrain[-olcut2Index,] # aykırı değeri fazla olanı seçiyorum
```

```
abtrain_model3rem = lm(rings~.-length,data=abtrainrem_model3)
summary(abtrain_model3rem)
```

```
##
## Call:
## lm(formula = rings ~ . - length, data = abtrainrem_model3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.7704 -1.0939 -0.1853  0.9164  5.8275
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.44466    0.24326   14.160 < 2e-16 ***
## sexI           -0.79157    0.08855   -8.940 < 2e-16 ***
## sexM            0.07283    0.07267    1.002  0.316
## diameter       9.29409    0.92911   10.003 < 2e-16 ***
## height        18.84257    2.05896    9.152 < 2e-16 ***
## whole_weight   9.93066    0.83606   11.878 < 2e-16 ***
## shucked_weight -20.46462    0.92552  -22.111 < 2e-16 ***
## viscera_weight -12.17487    1.29261   -9.419 < 2e-16 ***
## shell_weight   5.82127    1.27666    4.560 5.32e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.639 on 3119 degrees of freedom
## Multiple R-squared:  0.6216, Adjusted R-squared:  0.6207
## F-statistic: 640.6 on 8 and 3119 DF,  p-value: < 2.2e-16
```

Aykırı değerler çıkarıldığın da bütün modellerde oluşan değişimi görebiliyoruz, R2 ler artmış durumda. Şimdi aykırı değerleri atılmış olan modellerde karşılaştırma yapıp birini seçelim.

```
cat("abtrain_model1rem ---> AIC:", AIC(abtrain_model1rem, k=10)," BIC:", BIC(abtrain_model1rem), " RMSE:",
", sqrt(sum(abtrain_model1rem$residuals**2)/length(abtrain_model1rem$residuals))), "\n")
```

```
## abtrain_model1rem ---> AIC: 12154.86 BIC: 12135.36 RMSE: 1.655826
```

```
cat("abtrain_model2.1rem ---> AIC:", AIC(abtrain_model2.1rem, k=7)," BIC:", BIC(abtrain_model2.1rem), " RMSE:",
", sqrt(sum(abtrain_model2.1rem$residuals**2)/length(abtrain_model2.1rem$residuals))), "\n")
```

```
## abtrain_model2.1rem ---> AIC: 12235.25 BIC: 12243.69 RMSE: 1.673455
```

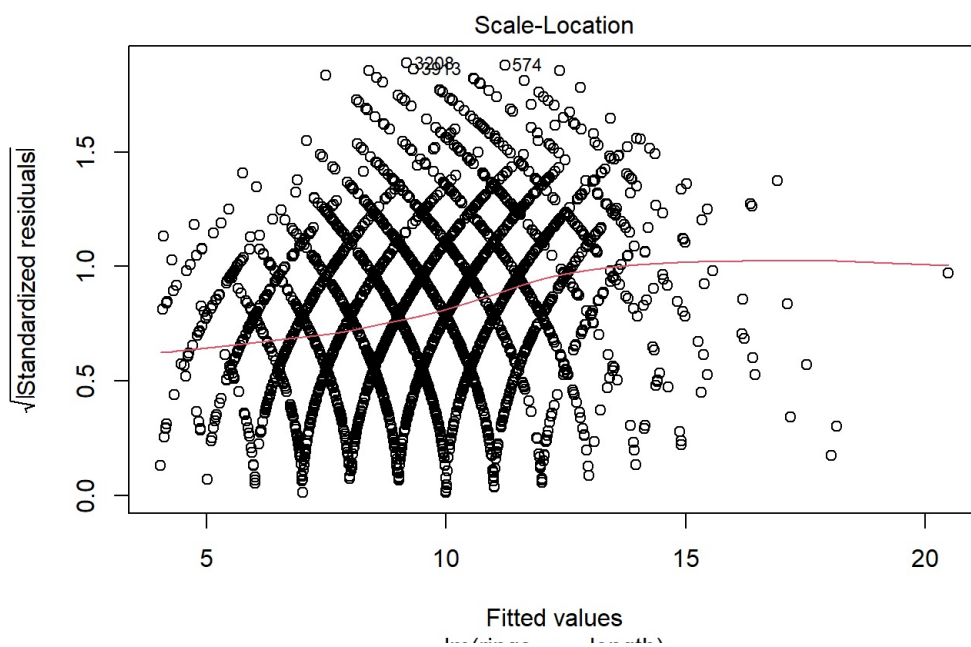
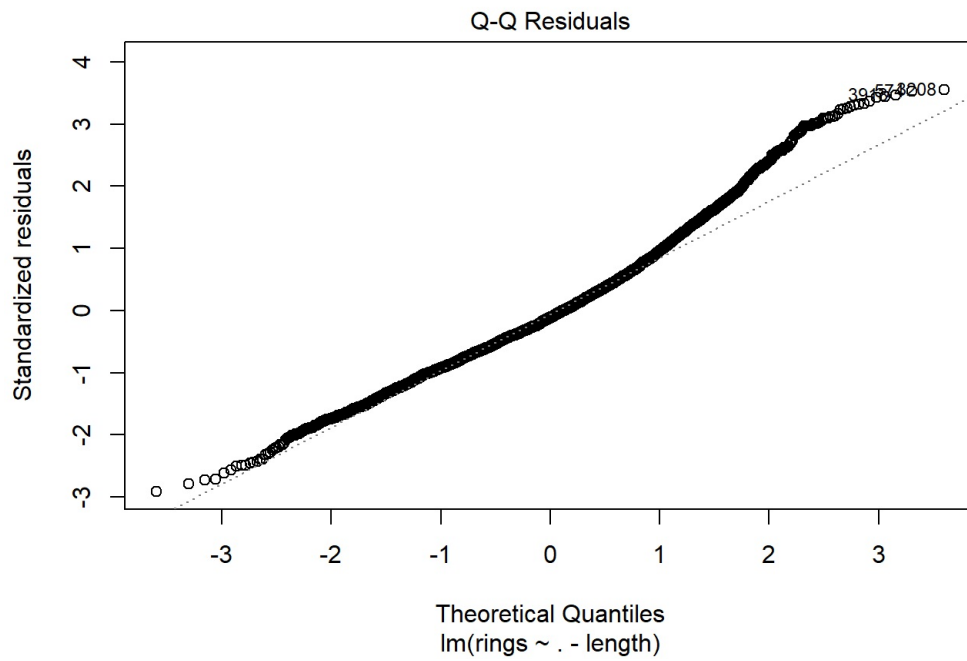
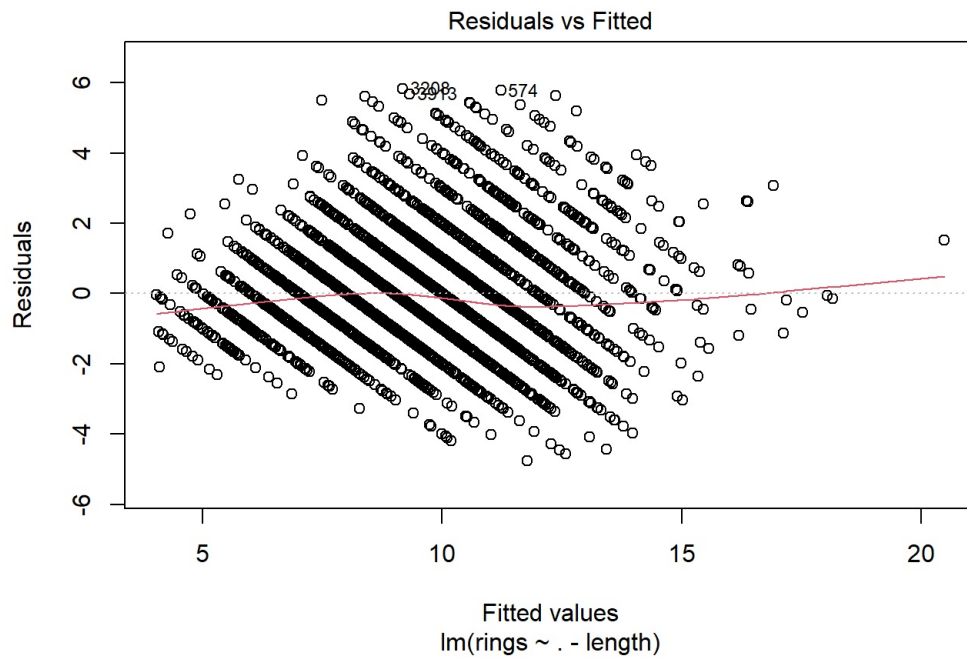
```
cat("abtrain_model3rem ---> AIC:", AIC(abtrain_model3rem, k=10)," BIC:", BIC(abtrain_model3rem), " RMSE:",
", sqrt(sum(abtrain_model3rem$residuals**2)/length(abtrain_model3rem$residuals)))
```

```
## abtrain_model3rem    --->    AIC: 12057.11    BIC: 12037.6    RMSE: 1.636181
```

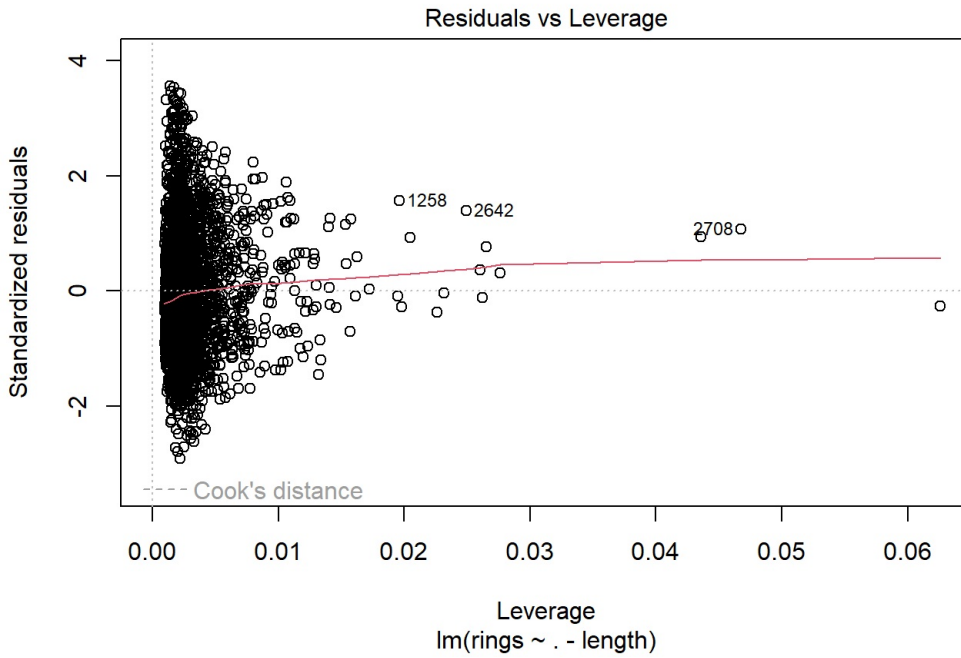
Burada AIC, BIC ve RMSE olarak abtrain_model3rem in daha iyi olduğunu gördük. Şuanda abtrain_model3rem önceki modellere göre bulduğumuz en iyi modelimiz.

abtrain_model3rem modelimiz grafiğine bakalım :

```
plot(abtrain_model3rem)
```



lm(rings ~ . - length)



İlk grafikte, modelin sabit değişimi ihlal ettiğini ve uygun değerler arttıkça artıkların da arttığı bir modele sahip olduğunu görüyoruz. Değişen varyans sorunu olabileceğini düşünüyoruz.

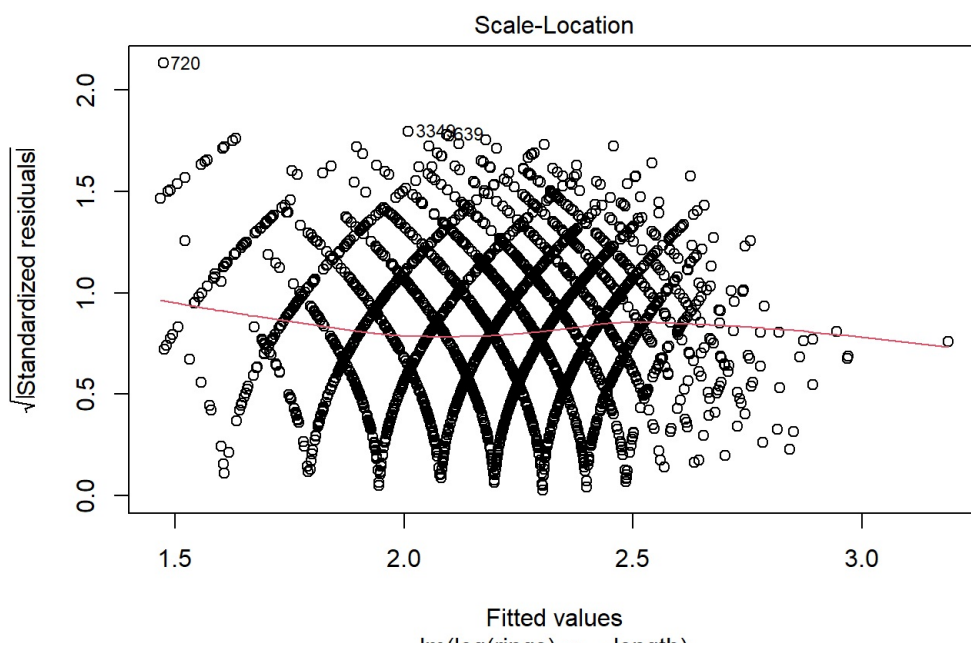
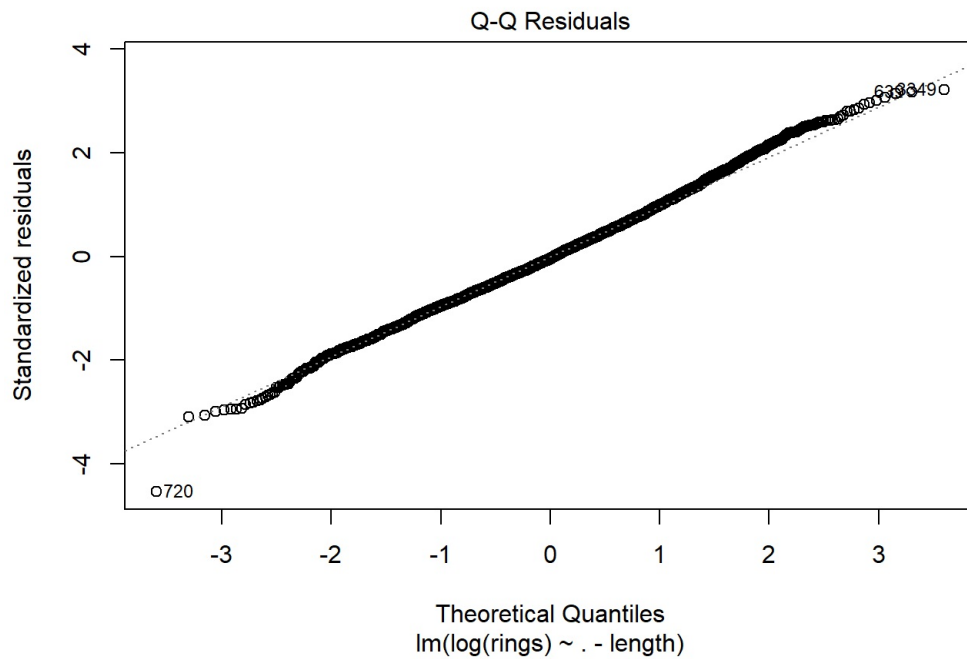
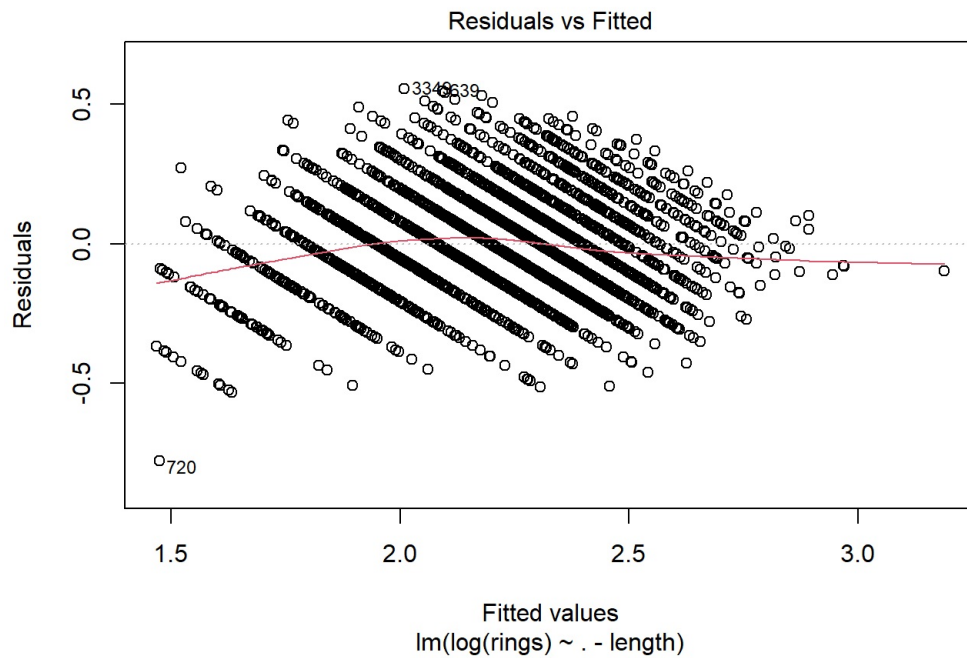
Normal QQ çiziminde ayrıca, hataların normal olarak dağılmayabileceğinin bir göstergesi olan kalın kuyruk vardır.

Modele log dönüşümü uygulayarak tekrar kontrol edelim ;

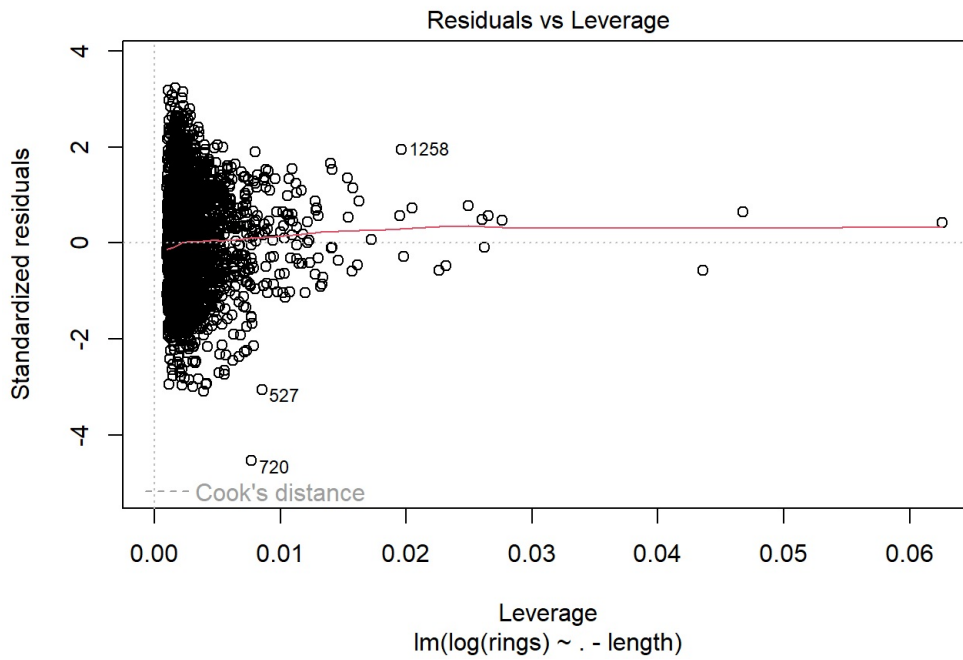
```
abtrain_model3remlogged = lm(log(rings)~.-length,data=abtrainrem_model3)
summary(abtrain_model3remlogged)
```

```
##
## Call:
## lm(formula = log(rings) ~ . - length, data = abtrainrem_model3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.78050 -0.11425 -0.00729  0.10964  0.55567
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.325187   0.025601  51.763 < 2e-16 ***
## sexI          -0.090906   0.009319  -9.755 < 2e-16 ***
## sexM           0.009298   0.007648   1.216  0.2242
## diameter      1.801074   0.097779  18.420 < 2e-16 ***
## height        2.341003   0.216685  10.804 < 2e-16 ***
## whole_weight  0.742155   0.087987   8.435 < 2e-16 ***
## shucked_weight -1.798961  0.097402 -18.469 < 2e-16 ***
## viscera_weight -1.057876  0.136035  -7.776 1.01e-14 ***
## shell_weight  0.385963   0.134356   2.873  0.0041 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1724 on 3119 degrees of freedom
## Multiple R-squared:  0.6517, Adjusted R-squared:  0.6508
## F-statistic: 729.6 on 8 and 3119 DF, p-value: < 2.2e-16
```

```
plot(abtrain_model3remlogged)
```



```
lm(log(rings) ~ . - length)
```



Önceki modele kıyasla daha iyi sabit varyasyona ve Q-Q grafiklerine sahip olduğunu görüyoruz.

```
sqrt(mean((abtrainrem_model3$rings - exp(fitted(abtrain_model3remlogged))) ^2))
```

```
## [1] 1.687585
```

```
sqrt(sum(abtrain_model3rem$residuals**2)/length(abtrain_model3rem$residuals))
```

```
## [1] 1.636181
```

değişen varyans sorununuz için Ağırlıklı en küçük kareler parametre tahminini deneyelim:

```
library(quantreg)
```

```
## Zorunlu paket yükleniyor: SparseM
```

```
##  
## Attaching package: 'SparseM'
```

```
## The following object is masked from 'package:base':  
##  
## backsolve
```

```
resid<-residuals(abtrain_model3rem)  
kareresid<-resid^2  
pred<-predict(abtrain_model3rem)  
mod<-lm(abs(rings)~.-length,data=abtrainrem_model3)  
weights<-1/predict(mod)^2  
weightedmodel<-lm(rings~.-length,data=abtrainrem_model3,weight=weights)  
  
bptest(weightedmodel)
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: weightedmodel  
## BP = 9.5396, df = 8, p-value = 0.2988
```

```
summary(weightedmodel)
```

```
##
## Call:
## lm(formula = rings ~ . - length, data = abtrainrem_model3, weights = weights)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42029 -0.12011 -0.01952  0.09640  0.72380
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.84002    0.17742   16.007 < 2e-16 ***
## sexI           -0.79672    0.08337   -9.556 < 2e-16 ***
## sexM            0.02921    0.07606    0.384  0.701
## diameter       11.68277    0.80545   14.505 < 2e-16 ***
## height         17.78980    1.96765    9.041 < 2e-16 ***
## whole_weight    8.39848    0.89606    9.373 < 2e-16 ***
## shucked_weight -18.60141    0.99725  -18.653 < 2e-16 ***
## viscera_veight -10.55958    1.38887   -7.603 3.8e-14 ***
## shell_weight    6.25348    1.34384    4.653 3.4e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1699 on 3119 degrees of freedom
## Multiple R-squared:  0.686, Adjusted R-squared:  0.6852
## F-statistic: 851.9 on 8 and 3119 DF, p-value: < 2.2e-16
```

Breusch-Pagan test ten de görüldüğü üzere değişen varyans sorununuz Weighted_Least_Squares ile çözülmüştür.

şimdi weightedmodelimizi karşılaştıralım:

```
AIC(weightedmodel,k=9)
```

```
## [1] 11809.73
```

```
BIC(weightedmodel)
```

```
## [1] 11800.21
```

```
sqrt(sum(weightedmodel$residuals**2)/length(weightedmodel$residuals))
```

```
## [1] 1.641606
```

```
AIC(abtrain_model3rem,k=9)
```

```
## [1] 12047.11
```

```
BIC(abtrain_model3rem)
```

```
## [1] 12037.6
```

```
sqrt(sum(abtrain_model3rem$residuals**2)/length(abtrain_model3rem$residuals))
```

```
## [1] 1.636181
```

weightedmodelin AIC ve BIC değerleri iyi olsada abtrain_model3rem in rmse si bir tık daha iyi.

Bir de modelleri testset üzerinden test edelim:

```
predictionswm<-predict(weightedmodel,abtest)
R2(predictionswm,abtest$rings)
```

```
## [1] 0.5189778
```

```
RMSE(predictionswm,abtest$rings)
```

```
## [1] 2.250359
```

```
MAE(predictionswm,abtest$rings)
```

```
## [1] 1.572282
```

```
predictionsrm<-predict(abtrain_model3rem,abtest)  
R2(predictionsrm,abtest$rings)
```

```
## [1] 0.5201435
```

```
RMSE(predictionsrm,abtest$rings)
```

```
## [1] 2.24798
```

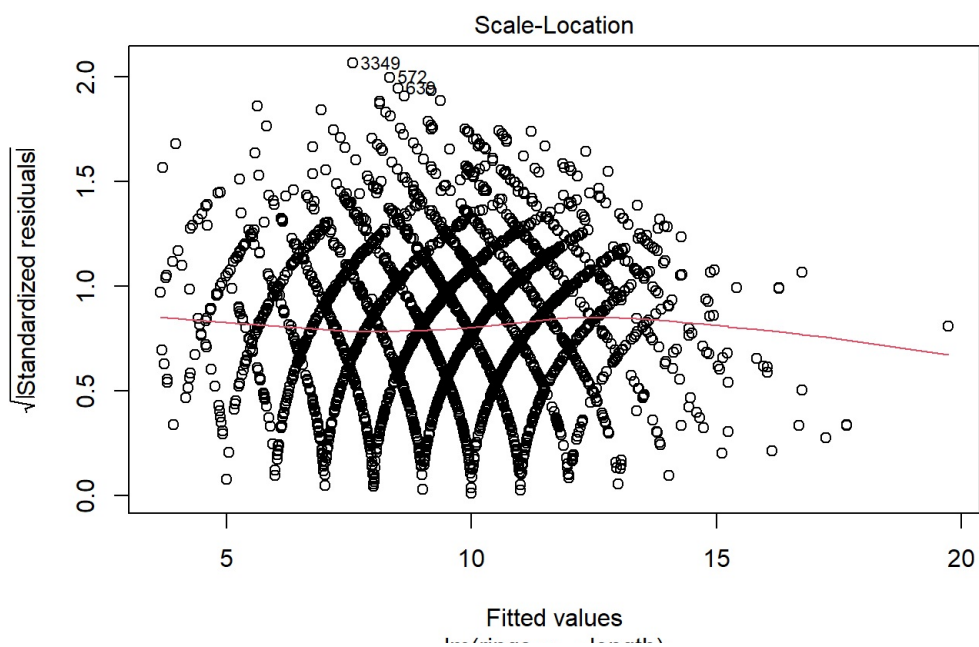
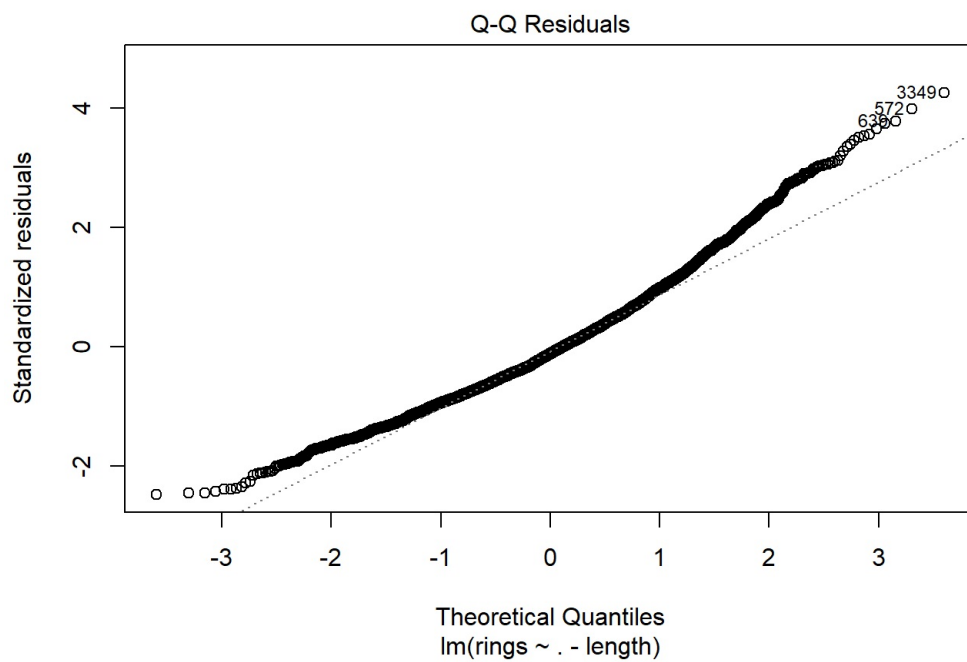
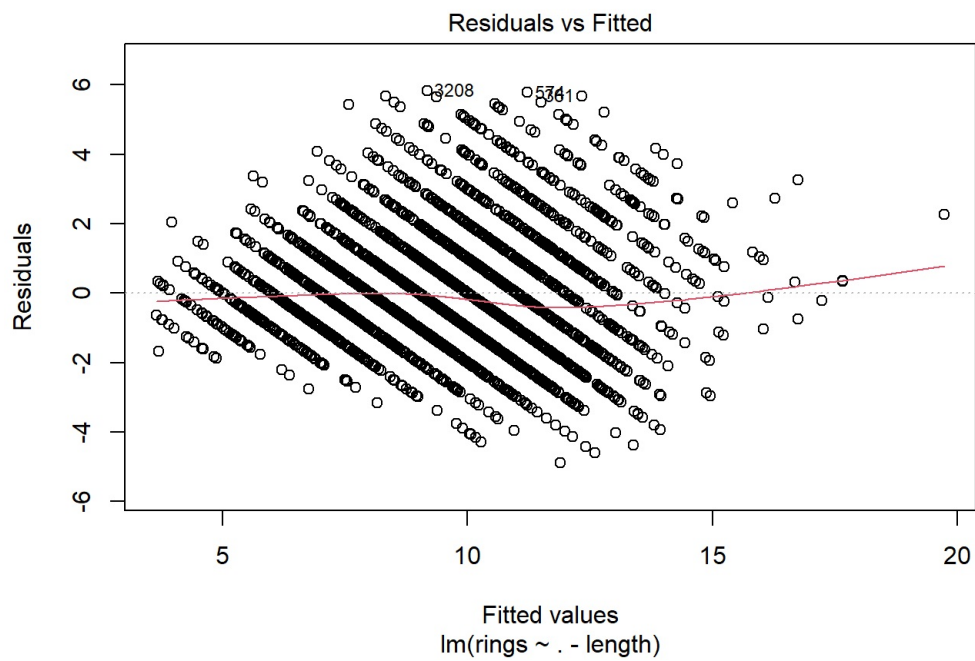
```
MAE(predictionsrm,abtest$rings)
```

```
## [1] 1.574947
```

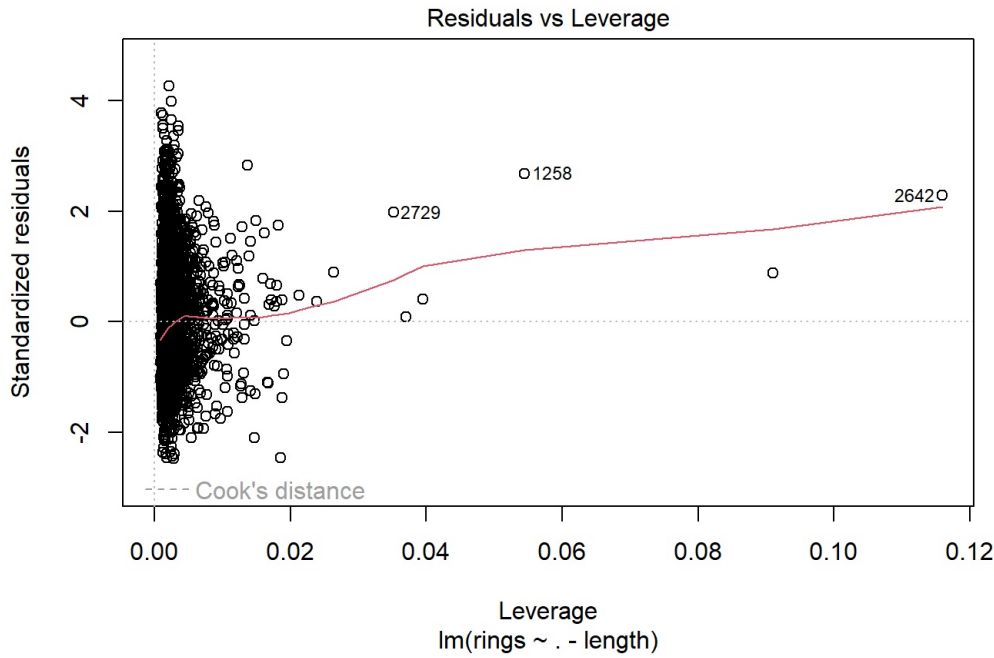
MAE de az da olsa bir düşüş yaşanmıştır fakat iki model arasında pek te bir fark yoktur.

weightedmodelimizin grafiklerine bakalım:

```
plot(weightedmodel)
```



lm(rings ~ . - length)



değişen varyans sorunumuz geçmişti

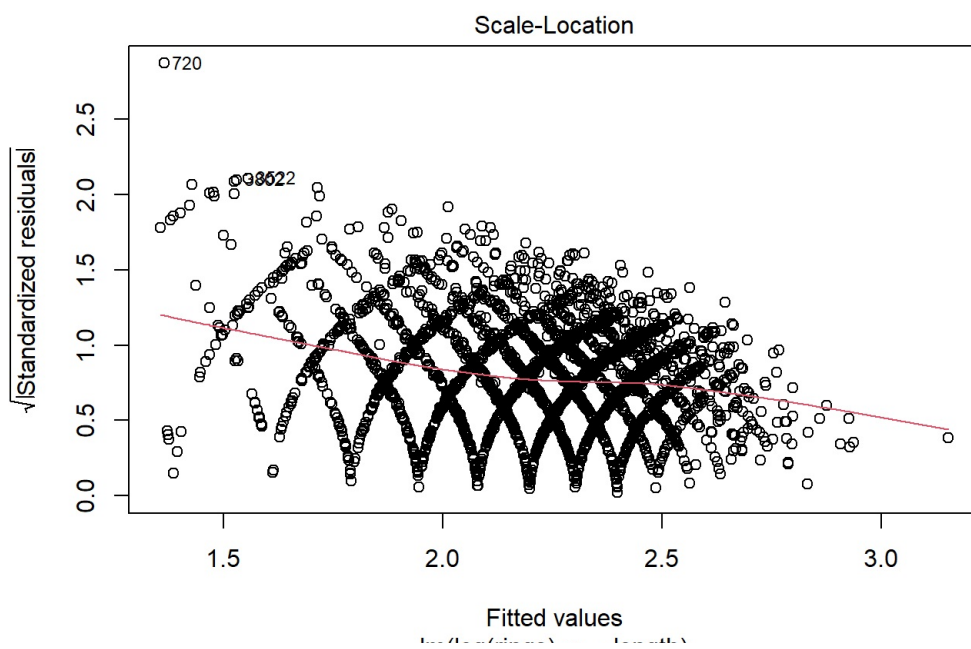
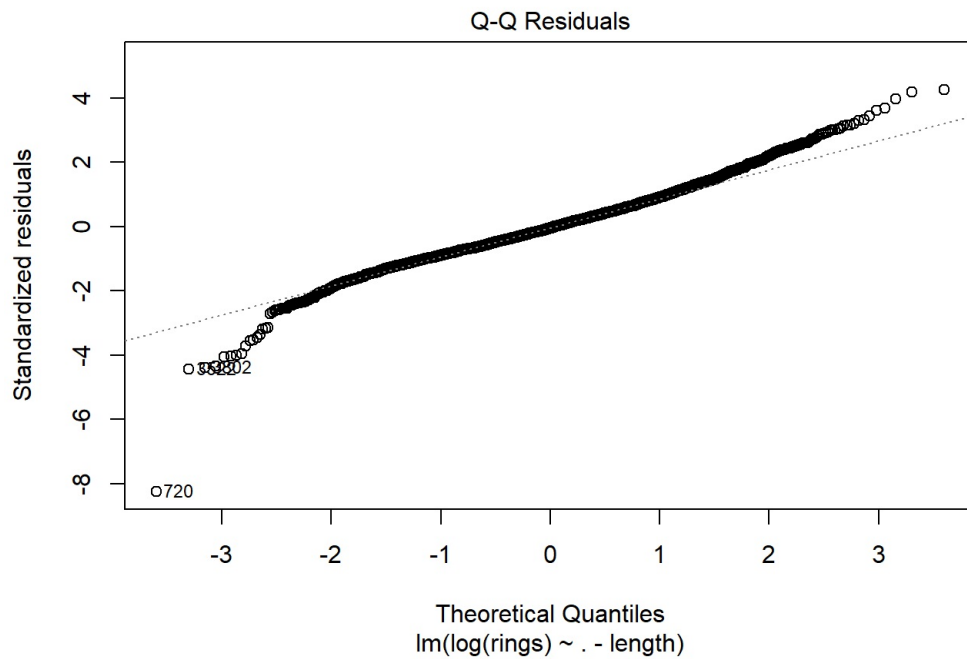
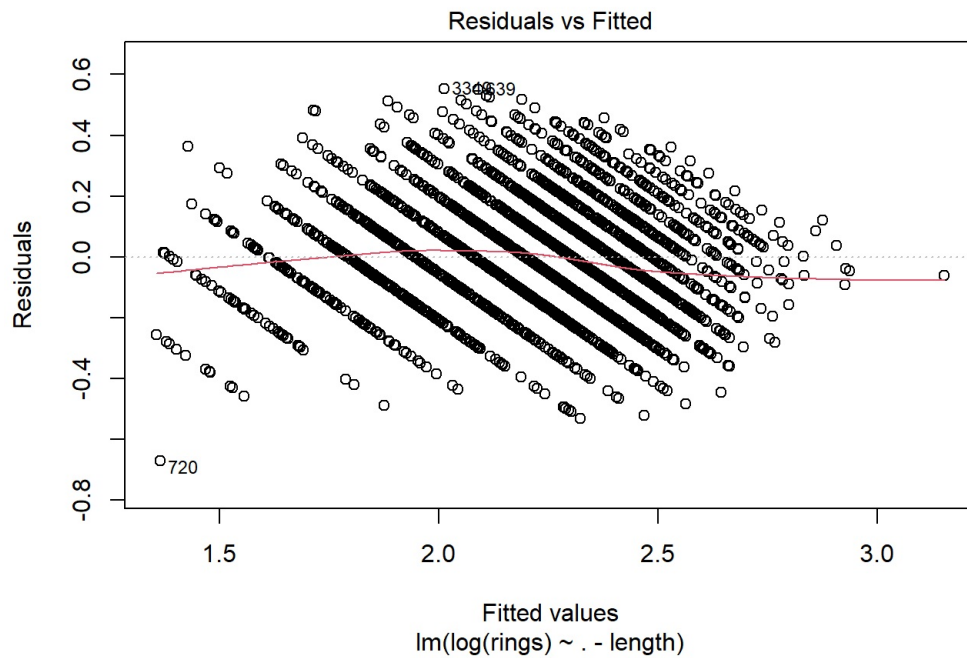
zaten bptestde de görmüştük fakat 2. grafikte normallikte sıkıntı var gibi gözüküyor

weightedmodel üzerinde log dönüşümü deneyelim:

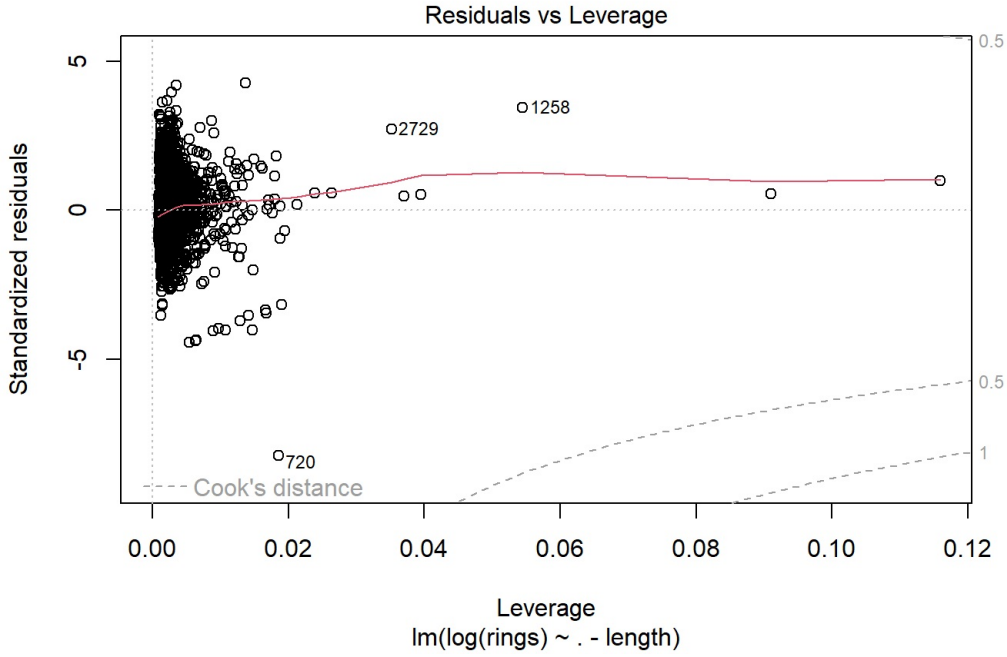
```
weightedmodellog<-lm(log(rings)~.-length,data=abtrainrem_model3,weight=weights)
summary(weightedmodellog)
```

```
##
## Call:
## lm(formula = log(rings) ~ . - length, data = abtrainrem_model3,
##     weights = weights)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -0.164071 -0.012837 -0.000901  0.011542  0.085164
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.156497   0.020962  55.172 < 2e-16 ***
## sexI          -0.089677   0.009850  -9.104 < 2e-16 ***
## sexM           0.008007   0.008986   0.891  0.3730
## diameter      2.316758   0.095160  24.346 < 2e-16 ***
## height        2.672324   0.232468  11.495 < 2e-16 ***
## whole_weight  0.633427   0.105865   5.983 2.43e-09 ***
## shucked_weight -1.745468   0.117820 -14.815 < 2e-16 ***
## viscera_veight -1.018690   0.164089  -6.208 6.07e-10 ***
## shell_weight  0.291536   0.158768   1.836  0.0664 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02008 on 3119 degrees of freedom
## Multiple R-squared:  0.7229, Adjusted R-squared:  0.7222
## F-statistic: 1017 on 8 and 3119 DF, p-value: < 2.2e-16
```

```
plot(weightedmodellog)
```



lm(log(rings) ~ . - length)



modelin R2 sinin iyi derece arttığını ve q-q plotta da normmalleştğini gözlemleyebiliyoruz

modelleri karşılaştıralım:

```
rmse <- function(actual, predicted) {  
  sqrt(mean((actual - predicted) ^ 2))  
}
```

```
cat(sqrt(mean((abtrainrem_model3$rings - exp(fitted(weightedmodellog))) ^ 2)), "wmlog", "\n")
```

```
## 1.717146 wmlog
```

```
cat(sqrt(mean((abtest$rings - exp(predict(weightedmodellog, newdata=abtest))) ^ 2)), "wmlog", "\n")
```

```
## 2.338202 wmlog
```

```
cat(rmse(abtrain$rings, predict(weightedmodel, abtrain)), "wm", "\n")
```

```
## 2.205996 wm
```

```
cat(rmse(abtest$rings, predict(weightedmodel, abtest)), "wm")
```

```
## 2.250359 wm
```

Test set üzerinde weightedmodel biraz daha önde olsada train set üzerinde log dönüşümünü uyguladığımız weightedmodellog modelimiz çok daha öndedir ve R2 açıklama oranı daha fazladır.

Sonuç olarak

İlk model için sabit değişim ve normallik varsayımlarını ihlal etmiştik. Ağırlıklı en küçük kareler parametre tahmini ve dönüşüm kullanarak iyi bir model bulmayı başardık. Seçilen bu aday modeller yanıt değişkeni için log dönüşümüne sahipti.

Yüksek çoklu bağlantının varlığından sonra bile, model seçimi sırasında 8 parametrenin tümünü içeren modeli seçtik. yani tüm seçim teknikleri, modeldeki tüm bağımsız değişkenler kullanıldığında en düşük AIC'nin geldiğini öne sürüyordu.

Modelden length değişkeninin denizkulağının yaşını tahmin etmede muhtemelen önemli bir yeri olmayacağını görebiliriz. bunu modelimizde seçmediğimizi görebiliriz. length, Diameter ile aşağı yukarı ilişkili olacağından (yani, length(uzunluk) arttıkça diameter(çap) artar gibi) bu tür bir anlam ifade eder ve modelimize zaten diameter aldığımız için, abalone veri setinde length değişkenine ihtiyacımız yoktur.

Tahmin yaparken gördüğümüz başka bir şey de, whole_weightin tek başına yaşı tahmin etmek için kullanılamayacağıdır. yani bütün weightlerin(ağırlıkların) birlikte anlamlı olduğudur.

Model üzerinde farklı teknikler ve dönüşümler polinom vb. kullanarak geliştirilebilir.

Elde ettiğimiz sonuçlara göre son nihai modelimiz Ağırlıklı en küçük kareler parametre tahmini ve log dönüşümü kullandığımız *weightedmodellog* dur.

$$\text{rings} = 1.156497 + (-0.089677x_1) + (0.008007x_2) + (2.316758x_3) + (2.672324x_4) + (0.633427x_5) + (-1.745468x_6) + (-1.018690x_7) + (0.291536x_8)$$

Processing math: 100%