# UWDS2-Wk6-LinearModeling-Abalone-jms206

*Jim Stearns, NetID=jms206*

*Due 24 Feb 2015*

## Overview

**This assignment uses the abalone dataset from the UCI machine learning data repository. Each row in the dataset describes a different abalone, including its sex, linear dimensions, and weights. The last column contains the number of rings in an abalone's shell. This is a proxy for the abalone's age, just as tree rings tell us how old a tree is. The problem we face is coming up with an easy to apply model that predicts the number of rings (counting them is laborious and pretty unpleasant for the abalone, which gets sawed in half).**

**The data is available here: (http://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data)**

**The data description is here: (http://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.names)**

```
urlUciAboloneData <- "http://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data"
abo <- read.csv(urlUciAboloneData, header=FALSE)
abo_names=c("Sex","Length", "Diameter","Height", "Whole_weight", "Shucked_weight",
    "Viscera_weight","Shell_weight","Rings")
names(abo) <- abo_names
```

Put the data.frame into a data.table in order to simplify column references below.

```
library(data.table)
abodt <- data.table(abo)
summary(abodt)
```

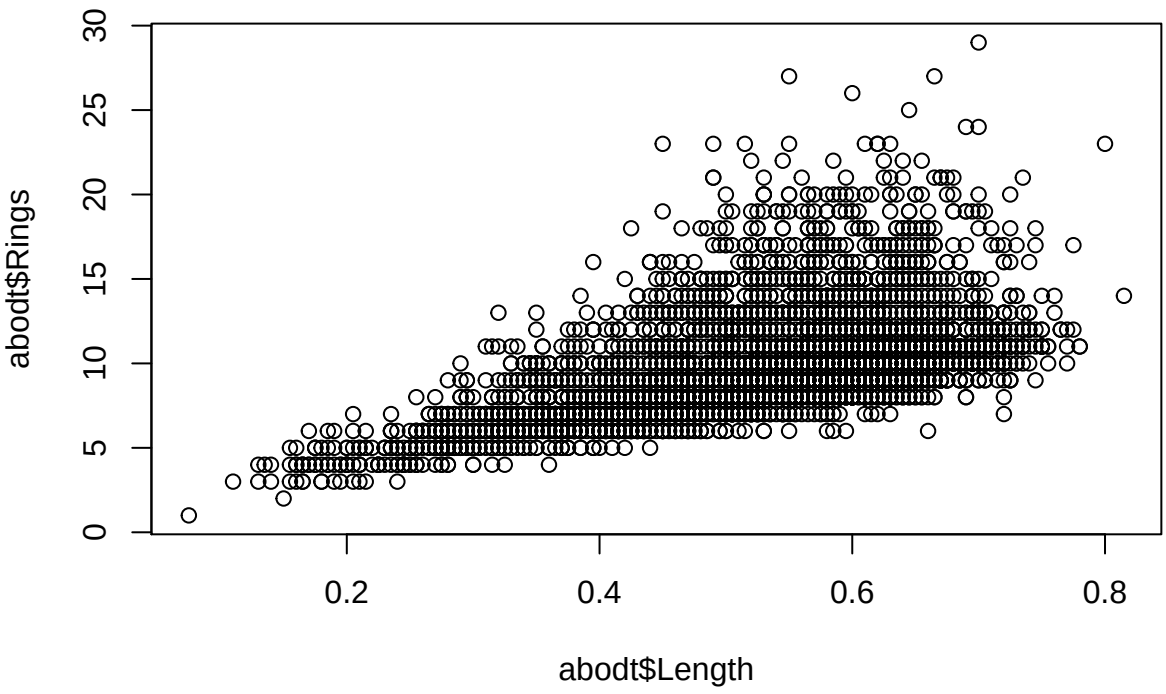
```
##  Sex          Length         Diameter          Height
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150
##  M:1528   Median :0.545   Median :0.4250   Median :0.1400
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395
##           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.   :0.0005   Min.   :0.0015
##  1st Qu.:0.4415   1st Qu.:0.1860   1st Qu.:0.0935   1st Qu.:0.1300
##  Median :0.7995   Median :0.3360   Median :0.1710   Median :0.2340
##  Mean   :0.8287   Mean   :0.3594   Mean   :0.1806   Mean   :0.2388
##  3rd Qu.:1.1530   3rd Qu.:0.5020   3rd Qu.:0.2530   3rd Qu.:0.3290
##  Max.   :2.8255   Max.   :1.4880   Max.   :0.7600   Max.   :1.0050
##      Rings
```

```
##  Min.   : 1.000
##  1st Qu.: 8.000
##  Median : 9.000
##  Mean   : 9.934
##  3rd Qu.:11.000
##  Max.   :29.000
```

## 1. Plot Rings vs Length

**Plot the number of rings as a function of length.**

```
plot(abodt$Length, abodt$Rings)
```




## 2. Linear Model: Rings ~ Length

__ Fit a linear model to this data (rings = a*length + b) using R's lm command. Examine the output of the summary table for the fit. Is length a significant factor? __

```
abolm <- lm(Rings ~ Length, data=abodt)
abolm_summary <- summary(abolm)
abolm_summary
```

```
##
## Call:
## lm(formula = Rings ~ Length, data = abodt)
##
## Residuals:
```

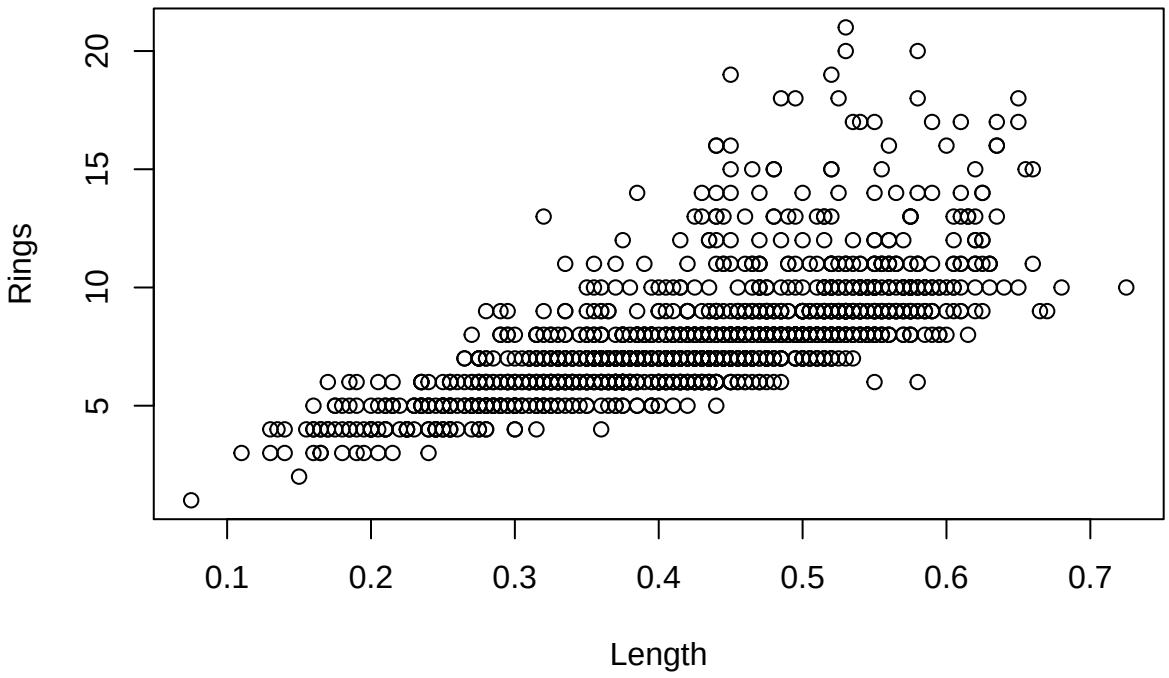
```
##     Min      1Q  Median      3Q     Max
## -5.9665 -1.6961 -0.7423  0.8733 16.6776
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1019     0.1855   11.33   <2e-16 ***
## Length       14.9464     0.3452   43.30   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.679 on 4175 degrees of freedom
## Multiple R-squared:  0.3099, Adjusted R-squared:  0.3098
## F-statistic:  1875 on 1 and 4175 DF,  p-value: < 2.2e-16
```

With a p-value < 2^15, length is a significant factor.

## 3. Linear Model, Immature Abalone: Rings ~ Length

__There are three sexes of abalone: male, female, and immature. Filter the data so that only the immature abalone remain. Fit the same model to this data (rings = a*length + b). Examine the output of summary: is this model a better or worse than the model fit to all of the data?__

```
immaturedt <- abodt[abodt$Sex == "I",]
with (immaturedt, plot(Length, Rings))
```




```
immaturelm1 <- lm(Rings ~ Length, data=immaturedt)
immaturelm1_summary <- summary(immaturelm1)
immaturelm1_summary
```

```
##
## Call:
## lm(formula = Rings ~ Length, data = immaturedt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.3002 -1.0736 -0.3398  0.5271 11.4911
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.1204     0.2024   5.535 3.73e-08 ***
## Length       15.8273     0.4586  34.513  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.828 on 1340 degrees of freedom
## Multiple R-squared:  0.4706, Adjusted R-squared:  0.4702
## F-statistic:  1191 on 1 and 1340 DF,  p-value: < 2.2e-16
```

```
stopifnot(abolm_summary$adj.r.squared < immaturelm1_summary$adj.r.squared)
```

"Goodness of fit" is usually defined by $R^2$, or the fraction of variance in the data that's explained by the model.

The *adjusted* $R^2$ value is used here, rather than $R^2$, in preparation for comparing this value to that obtained when the number of predictors is increased (e.g. height and diameter). Unadjusted $R^2$ can be inflated by the mere presence of additional predictors. *Adjusted* $R^2$ takes the number of predictors into account.

The *adjusted* $R^2$ value for linear model of immature abalone of 0.47 is notably larger than the corresponding value of 0.31 for the linear model of the entire abolone sample (female, male, and immature).

In summary, a linear model using Length as a sole predictor of rings is a better predictor when applied to the immature (pre-adult) subset of abolone than when applied to the entire dataset.

## 4. Linear Model, Immature Abalone: Rings ~ Length + Height + Diameter

**Still working with the immature abalone only, add Height and Diameter to the model (rings = a*length* + *b*height + etc.). Examine the output of summary: what are the significant factors in this new model? Compare the result to the "length only" model and explain why the two results on consistent with each other.**

```
immaturelm2 <- lm(Rings ~ Length + Height + Diameter, data=immaturedt)
immaturelm2_summary <- summary(immaturelm2)
immaturelm2_summary
```

```
##
## Call:
## lm(formula = Rings ~ Length + Height + Diameter, data = immaturedt)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.4320 -1.0075 -0.2701  0.5536 11.0270
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.5223     0.1976   7.703 2.57e-14 ***
## Length       -2.9797     2.6292  -1.133  0.25728
## Height       40.8082     3.6467  11.190  < 2e-16 ***
## Diameter      9.9103     3.3316   2.975  0.00299 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.729 on 1338 degrees of freedom
## Multiple R-squared:  0.5274, Adjusted R-squared:  0.5263
## F-statistic: 497.7 on 3 and 1338 DF,  p-value: < 2.2e-16
```

```
# Assert expected p-values for the three predictors:
stopifnot(immaturelm2_summary$coefficients["Length","Pr(>|t|)"] > 0.25)
stopifnot(immaturelm2_summary$coefficients["Height","Pr(>|t|)"] < 0.05)
stopifnot(immaturelm2_summary$coefficients["Diameter","Pr(>|t|)"] < 0.05)
```

The significant predictors (significant: a p-value less than 0.05) are *Height* and *Diameter* (with p-values of 0.000 and 0.003 , respectively. *Length* is not significant (p-value of 0.26 is greater than 0.05).

**Compare the result to the 'length only' model and explain why the two results on consistent with each other.**

The fit with three predictors (two, effectively, because of $Length$'s high p-value) is better than with one predictor: 0.53 vs 0.47.

But why has *Length* been displaced? By itself, it's a pretty good predictor, yet when *Diameter* and *Length* are added, it becomes insignificant (high p-value)?

Reason: Because *Length* is strongly correlated with *Diameter* (0.986).

Sanity Check: Use *Length* with *Diameter* should yield a similar good fit as using *Height* with *Diameter*:

```
immaturelm3 <- lm(Rings ~ Length + Height, data=immaturedt)
immaturelm3_summary <- summary(immaturelm3)
immaturelm3_summary
```

```
##
## Call:
## lm(formula = Rings ~ Length + Height, data = immaturedt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5666 -1.0215 -0.2717  0.5584 11.0690
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   1.3914     0.1932   7.201 9.92e-13 ***
## Length        4.2061     1.0408   4.041 5.62e-05 ***
## Height       43.5199     3.5413  12.289  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.734 on 1339 degrees of freedom
## Multiple R-squared:  0.5243, Adjusted R-squared:  0.5235
## F-statistic: 737.8 on 2 and 1339 DF,  p-value: < 2.2e-16
```

```
stopifnot(abs(immaturelm3_summary$adj.r.squared - immaturelm2_summary$adj.r.squared) < 0.01)
```

The $adjustedR^2$ values are within 0.01 of each other.

## 5. Linear Model, Immature Abalone: Rings ~ . -Sex

**Still working with the immature abalone only, add all of the factors to the model (except Sex: since we only have immature abalone, this value is the same for every data point) (rings = a*length* + *b*height + etc.).**

```
# The following use of "." caused this error:
# Error in `contrasts<-`(`*tmp*`, value = contr.funs[1 + isOF[nn]]) :
# contrasts can be applied only to factors with 2 or more levels
##mmaturelm4 <- lm(Rings ~ .-Sex, data=immaturedt)
# Explicitly specifying all the predictors works:
immaturelm4 <- lm(Rings ~ Length + Diameter + Height + Whole_weight + Shucked_weight + Viscera_weigh
immaturelm4_summary <- summary(immaturelm4)
immaturelm4_summary
```

```
##
## Call:
## lm(formula = Rings ~ Length + Diameter + Height + Whole_weight +
##     Shucked_weight + Viscera_weight + Shell_weight, data = immaturedt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.0870 -0.9375 -0.2514  0.5947 10.0078
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.9380     0.3113   9.437  < 2e-16 ***
## Length          -2.7461     2.4851  -1.105 0.269347
## Diameter         5.9345     3.1786   1.867 0.062116 .
## Height          28.8253     3.6232   7.956 3.77e-15 ***
## Whole_weight     8.2272     1.4910   5.518 4.12e-08 ***
## Shucked_weight -14.6625     1.6283  -9.005  < 2e-16 ***
## Viscera_weight -11.2791     3.2068  -3.517 0.000451 ***
```

```
## Shell_weight    10.6606     2.3282   4.579 5.11e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.616 on 1334 degrees of freedom
## Multiple R-squared:  0.5879, Adjusted R-squared:  0.5858
## F-statistic: 271.9 on 7 and 1334 DF,  p-value: < 2.2e-16
```

```
# Using all predictors yields a somewhat higher adjusted R^2:
stopifnot((immaturelm4_summary$adj.r.squared - immaturelm2_summary$adj.r.squared) > 0.05)
stopifnot(abs(immaturelm4_summary$adj.r.squared - immaturelm2_summary$adj.r.squared) < 0.1)
```

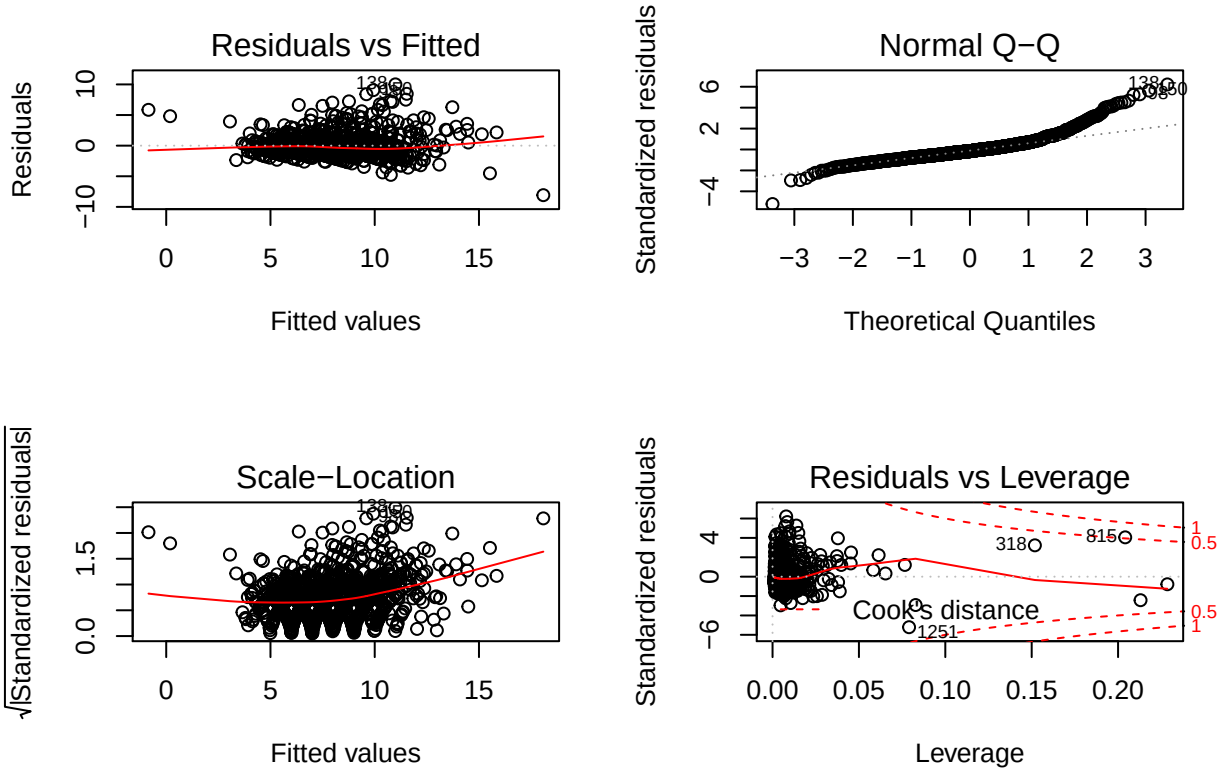
**Examine the residuals and summarize your observations.**

All predictors were significant except $Length$ and $Diameter$.

The $adjustedR^2$ with all predictors of 0.586 is somewhat but not hugely better than the $adjustedR^2$ for $Length + Height + Diameter$ of 0.526.

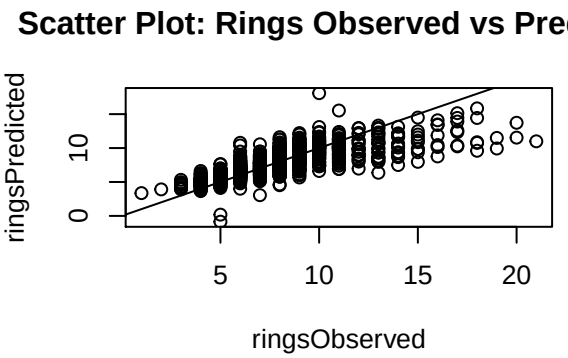
**Use graphical methods (histogram, qqplot, etc.)**

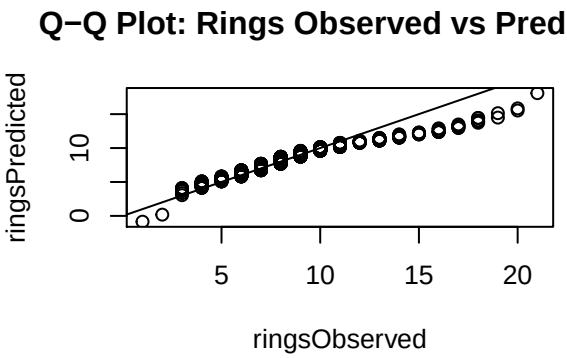
```
par(mfrow=c(2,2))
plot(immaturelm4)
```




```
ringsPredicted <- predict(immaturelm4)
ringsObserved <- immaturedt$Rings
plot(ringsObserved, ringsPredicted, main="Scatter Plot: Rings Observed vs Predicted")
abline(0, 1)

qqplot(ringsObserved, ringsPredicted, main="Q-Q Plot: Rings Observed vs Predicted")
abline(0, 1)
```

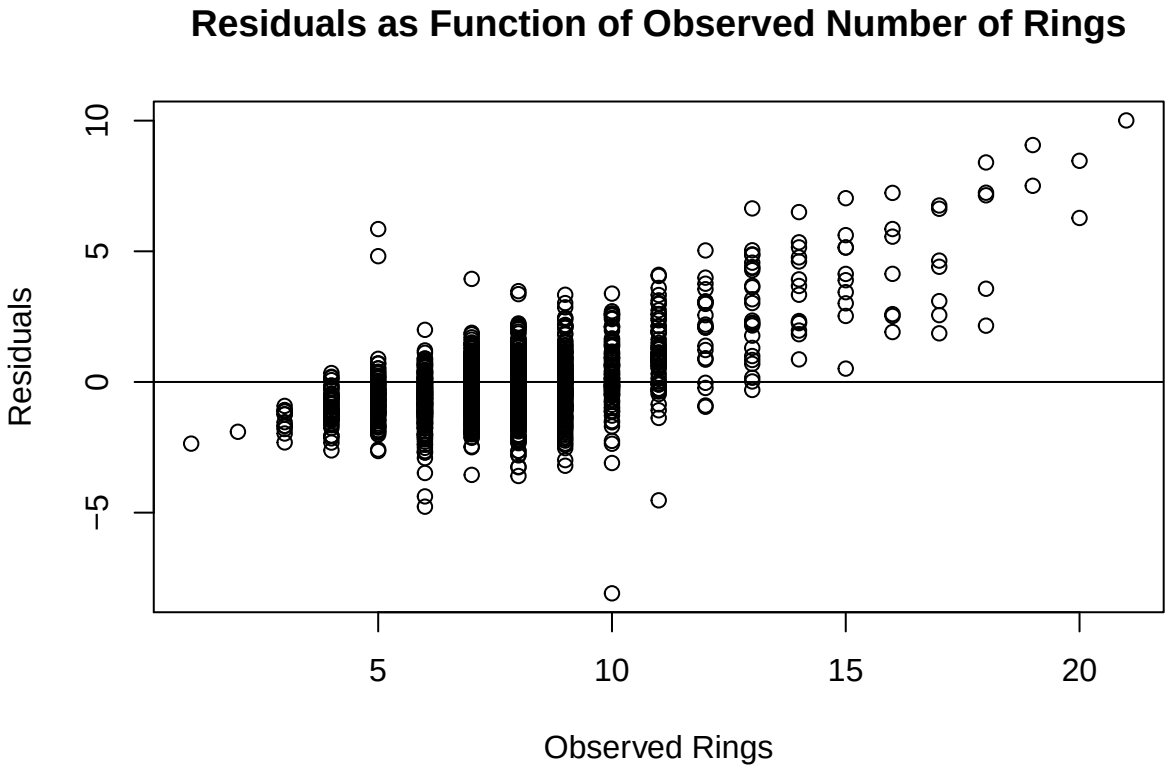


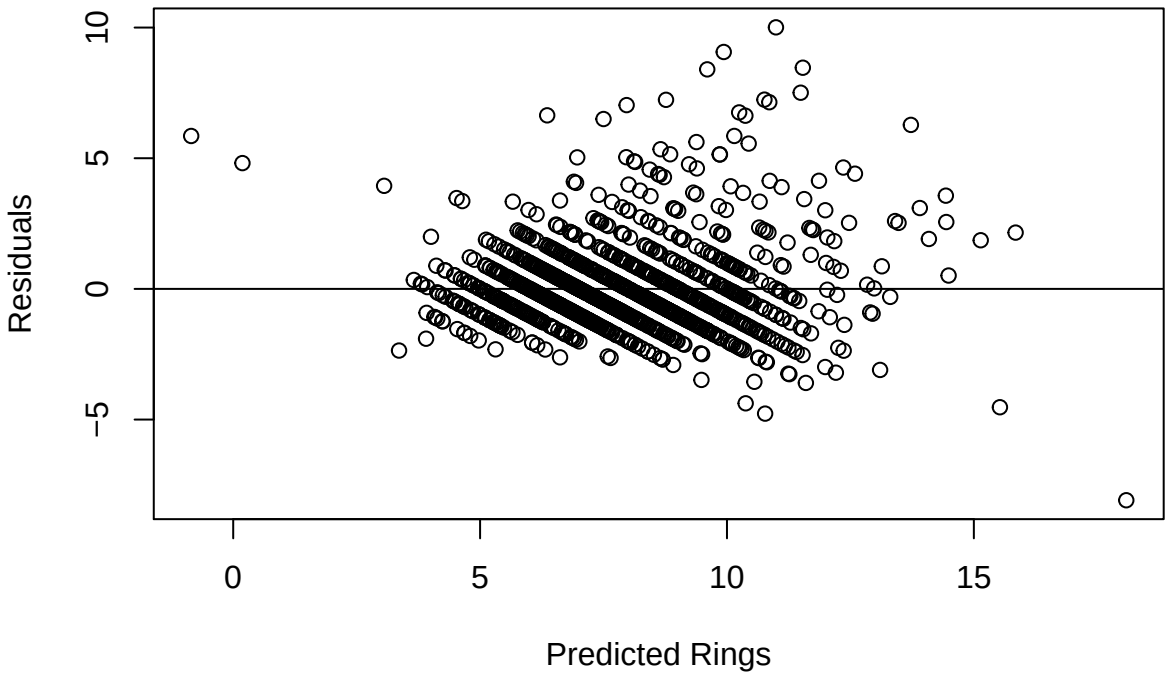

**Also plot the residuals as a function of the number of rings.**

Assumption: 'number of rings' means 'observed (not predicted) number of rings'"

```
immaturelm4_residuals <- resid(immaturelm4)
stopifnot(length(immaturelm4_residuals) == length(immaturedt$Rings))
plot(immaturedt$Rings, immaturelm4_residuals, ylab="Residuals", xlab="Observed Rings",
     main="Residuals as Function of Observed Number of Rings")
abline(0,0) # Ideally, all the values would be on this line.
```




```
plot(ringsPredicted, immaturelm4_residuals, ylab="Residuals", xlab="Predicted Rings",
     main="Residuals as Function of Predicted Number of Rings")
abline(0,0) # Ideally, all the values would be on this line.
```


Residuals as Function of Predicted Number of Rings
Residuals
10
5
0
−5
0
5
10
15
Predicted Rings