Machine Learning - Regression

Pratik Gandhi

June 21, 2016

Loading the libraries and the data:

```
library(GGally)
library(AppliedPredictiveModeling)

data(abalone)
```

Initial investigation:

```
# Dimensions of data:
dim(abalone)
```

```
## [1] 4177    9
```

```
# Head of dataset:
head(abalone)
```

```
##   Type LongestShell Diameter Height WholeWeight ShuckedWeight VisceraWeight
## 1    M        0.455    0.365  0.095      0.5140        0.2245        0.1010
## 2    M        0.350    0.265  0.090      0.2255        0.0995        0.0485
## 3    F        0.530    0.420  0.135      0.6770        0.2565        0.1415
## 4    M        0.440    0.365  0.125      0.5160        0.2155        0.1140
## 5    I        0.330    0.255  0.080      0.2050        0.0895        0.0395
## 6    I        0.425    0.300  0.095      0.3515        0.1410        0.0775
##   ShellWeight Rings
## 1       0.150    15
## 2       0.070     7
## 3       0.210     9
## 4       0.155    10
## 5       0.055     7
## 6       0.120     8
```

```
# Tail of dataset:
tail(abalone)
```

```
##      Type LongestShell Diameter Height WholeWeight ShuckedWeight VisceraWeight
## 4172    M        0.560    0.430  0.155      0.8675        0.4000        0.1720
## 4173    F        0.565    0.450  0.165      0.8870        0.3700        0.2390
## 4174    M        0.590    0.440  0.135      0.9660        0.4390        0.2145
## 4175    M        0.600    0.475  0.205      1.1760        0.5255        0.2875
## 4176    F        0.625    0.485  0.150      1.0945        0.5310        0.2610
## 4177    M        0.710    0.555  0.195      1.9485        0.9455        0.3765
##      ShellWeight Rings
## 4172      0.2290     8
## 4173      0.2490    11
## 4174      0.2605    10
## 4175      0.3080     9
```

```
## 4176      0.2960    10
## 4177      0.4950    12
```

```
# Checking variable types:
str(abalone)
```

```
## 'data.frame':    4177 obs. of  9 variables:
##  $ Type         : Factor w/ 3 levels "F","I","M": 3 3 1 3 2 2 1 1 3 1 ...
##  $ LongestShell : num  0.455 0.35 0.53 0.44 0.33 0.425 0.53 0.545 0.475 0.55 ...
##  $ Diameter     : num  0.365 0.265 0.42 0.365 0.255 0.3 0.415 0.425 0.37 0.44 ...
##  $ Height       : num  0.095 0.09 0.135 0.125 0.08 0.095 0.15 0.125 0.125 0.15 ...
##  $ WholeWeight  : num  0.514 0.226 0.677 0.516 0.205 ...
##  $ ShuckedWeight: num  0.2245 0.0995 0.2565 0.2155 0.0895 ...
##  $ VisceraWeight: num  0.101 0.0485 0.1415 0.114 0.0395 ...
##  $ ShellWeight  : num  0.15 0.07 0.21 0.155 0.055 0.12 0.33 0.26 0.165 0.32 ...
##  $ Rings        : int  15 7 9 10 7 8 20 16 9 19 ...
```

```
# Basic statistics summary:
summary(abalone)
```

```
##  Type      LongestShell      Diameter          Height        WholeWeight
##  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
##  ShuckedWeight    VisceraWeight     ShellWeight         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
```

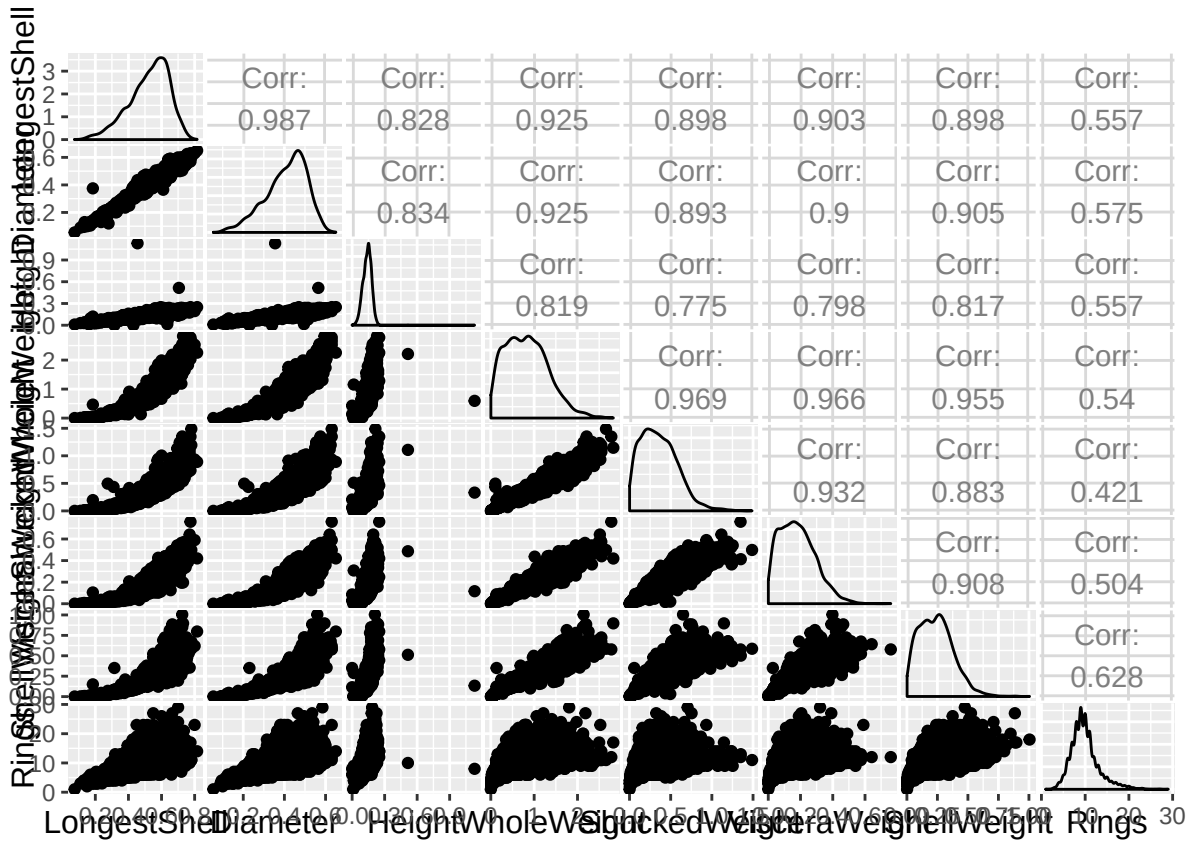
```
# Adding new variable age: abalone['Age'] <- 1.5*abalone$Rings
```

Overview of correlations:

```
# plot(abalone$LongestShell, abalone$Diameter) Positive correlation

# plot(abalone$LongestShell, abalone$Height) No correlation as such

con_abalone <- abalone[, 2:9]
ggpairs(con_abalone)
```



Shuffling the data into train and test set:

```
set.seed(1)

# Shuffling the dataset:
n <- nrow(abalone)
shuffled <- abalone[sample(n), ]

# Splitting the dataset into 70/30 ratio:
train_indices <- 1:round(0.7 * n)
train <- shuffled[train_indices, ]
test_indices <- (round(0.7 * n) + 1):n
test <- shuffled[test_indices, ]
```

Model selection on train dataset:

```
lm_abalone <- lm(Rings ~ ., train)

summary(lm_abalone)
```

```
##
## Call:
## lm(formula = Rings ~ ., data = train)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2679 -1.3056 -0.3429  0.8679 13.7223
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     3.99315    0.35095  11.378  < 2e-16 ***
## TypeI          -0.82156    0.12332  -6.662 3.21e-11 ***
## TypeM           0.06159    0.10055   0.612    0.540
## LongestShell   -1.38099    2.26178  -0.611    0.542
## Diameter       12.50885    2.75530   4.540 5.86e-06 ***
## Height          8.60411    1.65643   5.194 2.20e-07 ***
## WholeWeight     9.88003    0.87994  11.228  < 2e-16 ***
## ShuckedWeight -20.72722    1.00081 -20.710  < 2e-16 ***
## VisceraWeight -10.82232    1.55431  -6.963 4.11e-12 ***
## ShellWeight     7.54958    1.32660   5.691 1.39e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.203 on 2914 degrees of freedom
## Multiple R-squared:  0.5365, Adjusted R-squared:  0.5351
## F-statistic: 374.8 on 9 and 2914 DF,  p-value: < 2.2e-16
```

```
lm_abalone_1 <- lm(Rings ~ Type + Diameter + Height + WholeWeight + ShuckedWeight +
    VisceraWeight + ShellWeight, train)
summary(lm_abalone_1)
```

```
##
## Call:
## lm(formula = Rings ~ Type + Diameter + Height + WholeWeight +
##     ShuckedWeight + VisceraWeight + ShellWeight, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2303 -1.3069 -0.3448  0.8708 13.7643
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     3.92096    0.33040  11.867  < 2e-16 ***
## TypeI          -0.82824    0.12282  -6.744 1.85e-11 ***
## TypeM           0.06054    0.10053   0.602    0.547
## Diameter       10.98726    1.17519   9.349  < 2e-16 ***
## Height          8.57490    1.65556   5.179 2.38e-07 ***
## WholeWeight     9.89448    0.87953  11.250  < 2e-16 ***
## ShuckedWeight -20.78484    0.99624 -20.863  < 2e-16 ***
## VisceraWeight -10.92526    1.54498  -7.071 1.91e-12 ***
## ShellWeight     7.56374    1.32626   5.703 1.29e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.203 on 2915 degrees of freedom
## Multiple R-squared:  0.5365, Adjusted R-squared:  0.5352
## F-statistic: 421.7 on 8 and 2915 DF,  p-value: < 2.2e-16
```

```r
# Remove the type:
lm_abalone_2 <- lm(Rings ~ Diameter + Height + WholeWeight + ShuckedWeight + VisceraWeight +
    ShellWeight, train)
summary(lm_abalone_2)  # 0.5252
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + WholeWeight + ShuckedWeight +
##     VisceraWeight + ShellWeight, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7432 -1.3595 -0.3917  0.9028 13.5915
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.9365     0.2967   9.897  < 2e-16 ***
## Diameter       12.0835     1.1773  10.263  < 2e-16 ***
## Height          9.4897     1.6692   5.685 1.44e-08 ***
## WholeWeight    10.2098     0.8879  11.499  < 2e-16 ***
## ShuckedWeight -21.2909     1.0021 -21.246  < 2e-16 ***
## VisceraWeight -10.2400     1.5585  -6.570 5.92e-11 ***
## ShellWeight     7.3815     1.3403   5.507 3.96e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.226 on 2917 degrees of freedom
## Multiple R-squared:  0.5261, Adjusted R-squared:  0.5251
## F-statistic: 539.7 on 6 and 2917 DF,  p-value: < 2.2e-16
```

```r
# Remove other types of weights:
lm_abalone_3 <- lm(Rings ~ Diameter + Height + WholeWeight, train)
summary(lm_abalone_3)
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + WholeWeight, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.8684  -1.6426  -0.6670   0.9144  16.0104
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.4411     0.3424   7.130 1.26e-12 ***
## Diameter     12.6511     1.3554   9.334  < 2e-16 ***
## Height       15.8078     1.9394   8.151 5.31e-16 ***
## WholeWeight   0.1235     0.2651   0.466    0.641
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.61 on 2920 degrees of freedom
## Multiple R-squared:  0.3479, Adjusted R-squared:  0.3472
## F-statistic: 519.2 on 3 and 2920 DF,  p-value: < 2.2e-16
```

```r
lm_abalone_4 <- lm(Rings ~ Diameter + Height + ShellWeight, train)
summary(lm_abalone_4)
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + ShellWeight, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5040 -1.5546 -0.5508  0.9218 15.7842
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.8747     0.2956  19.872  < 2e-16 ***
## Diameter     -0.6334     1.1655  -0.544    0.587
## Height        8.9859     1.8711   4.802 1.65e-06 ***
## ShellWeight  12.7353     0.7912  16.096  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.502 on 2920 degrees of freedom
## Multiple R-squared:  0.401,  Adjusted R-squared:  0.4003
## F-statistic: 651.5 on 3 and 2920 DF,  p-value: < 2.2e-16
```

```r
lm_abalone_5 <- lm(Rings ~ Diameter + Height + WholeWeight + VisceraWeight, train)
summary(lm_abalone_5)
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + WholeWeight + VisceraWeight,
##     data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.0940  -1.6338  -0.6585   0.9243  15.3743
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.3779     0.3411   6.971 3.87e-12 ***
## Diameter       12.8825     1.3502   9.541  < 2e-16 ***
## Height         15.9136     1.9310   8.241 2.55e-16 ***
## WholeWeight     2.0013     0.4481   4.466 8.27e-06 ***
## VisceraWeight  -8.8641     1.7095  -5.185 2.31e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.599 on 2919 degrees of freedom
## Multiple R-squared:  0.3538, Adjusted R-squared:  0.3529
## F-statistic: 399.6 on 4 and 2919 DF,  p-value: < 2.2e-16
```

```r
lm_abalone_6 <- lm(Rings ~ Diameter + Height + WholeWeight + ShellWeight, train)
summary(lm_abalone_6)
```

```
##
## Call:
```

```
## lm(formula = Rings ~ Diameter + Height + WholeWeight + ShellWeight,
##     data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.4961  -1.4690  -0.5118   0.8738  16.6516
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.4952     0.3169  11.030  < 2e-16 ***
## Diameter      8.8568     1.2523   7.073 1.89e-12 ***
## Height       10.5032     1.7912   5.864 5.03e-09 ***
## WholeWeight  -5.7888     0.3486 -16.605  < 2e-16 ***
## ShellWeight  25.6677     1.0857  23.641  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.392 on 2919 degrees of freedom
## Multiple R-squared:  0.4527, Adjusted R-squared:  0.4519
## F-statistic: 603.5 on 4 and 2919 DF,  p-value: < 2.2e-16
```

```
lm_abalone_7 <- lm(Rings ~ Diameter + Height + WholeWeight + ShuckedWeight, train)
summary(lm_abalone_7)  # 0.5076
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + WholeWeight + ShuckedWeight,
##     data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.4231 -1.3675 -0.4262  0.9429 13.8876
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.7425     0.2975   9.218  < 2e-16 ***
## Diameter       12.9423     1.1772  10.994  < 2e-16 ***
## Height         10.2240     1.6940   6.035 1.79e-09 ***
## WholeWeight    11.0497     0.4223  26.164  < 2e-16 ***
## ShuckedWeight -24.1887     0.7838 -30.861  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.267 on 2919 degrees of freedom
## Multiple R-squared:  0.5083, Adjusted R-squared:  0.5076
## F-statistic: 754.4 on 4 and 2919 DF,  p-value: < 2.2e-16
```

```
lm_abalone_8 <- lm(Rings ~ Diameter + Height + WholeWeight + ShuckedWeight + VisceraWeight,
    train)
summary(lm_abalone_8)  # Highest - 0.5203
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + WholeWeight + ShuckedWeight +
##     VisceraWeight, data = train)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5669 -1.3610 -0.4166  0.9098 12.8947
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.6569     0.2938   9.043  < 2e-16 ***
## Diameter       13.2912     1.1625  11.433  < 2e-16 ***
## Height         10.2382     1.6720   6.123 1.04e-09 ***
## WholeWeight    14.0997     0.5407  26.075  < 2e-16 ***
## ShuckedWeight -24.8039     0.7767 -31.934  < 2e-16 ***
## VisceraWeight -13.0853     1.4778  -8.855  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.238 on 2918 degrees of freedom
## Multiple R-squared:  0.5212, Adjusted R-squared:  0.5203
## F-statistic: 635.2 on 5 and 2918 DF,  p-value: < 2.2e-16
```

```
lm_abalone_9 <- lm(Rings ~ Diameter + Height + WholeWeight + ShuckedWeight + ShellWeight,
    train)
summary(lm_abalone_9)  # 0.5182
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + WholeWeight + ShuckedWeight +
##     ShellWeight, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3173 -1.3744 -0.4069  0.9421 14.5588
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     3.1068     0.2977  10.435  < 2e-16 ***
## Diameter       11.3628     1.1807   9.624  < 2e-16 ***
## Height          9.1838     1.6806   5.465 5.03e-08 ***
## WholeWeight     6.5469     0.6960   9.406  < 2e-16 ***
## ShuckedWeight -19.4730     0.9701 -20.074  < 2e-16 ***
## ShellWeight    10.3008     1.2736   8.088 8.82e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.243 on 2918 degrees of freedom
## Multiple R-squared:  0.5191, Adjusted R-squared:  0.5182
## F-statistic: 629.9 on 5 and 2918 DF,  p-value: < 2.2e-16
```

```
lm_abalone_10 <- lm(Rings ~ Diameter + Height + WholeWeight + ShellWeight, train)
summary(lm_abalone_10)
```

```
##
## Call:
## lm(formula = Rings ~ Diameter + Height + WholeWeight + ShellWeight,
##     data = train)
```

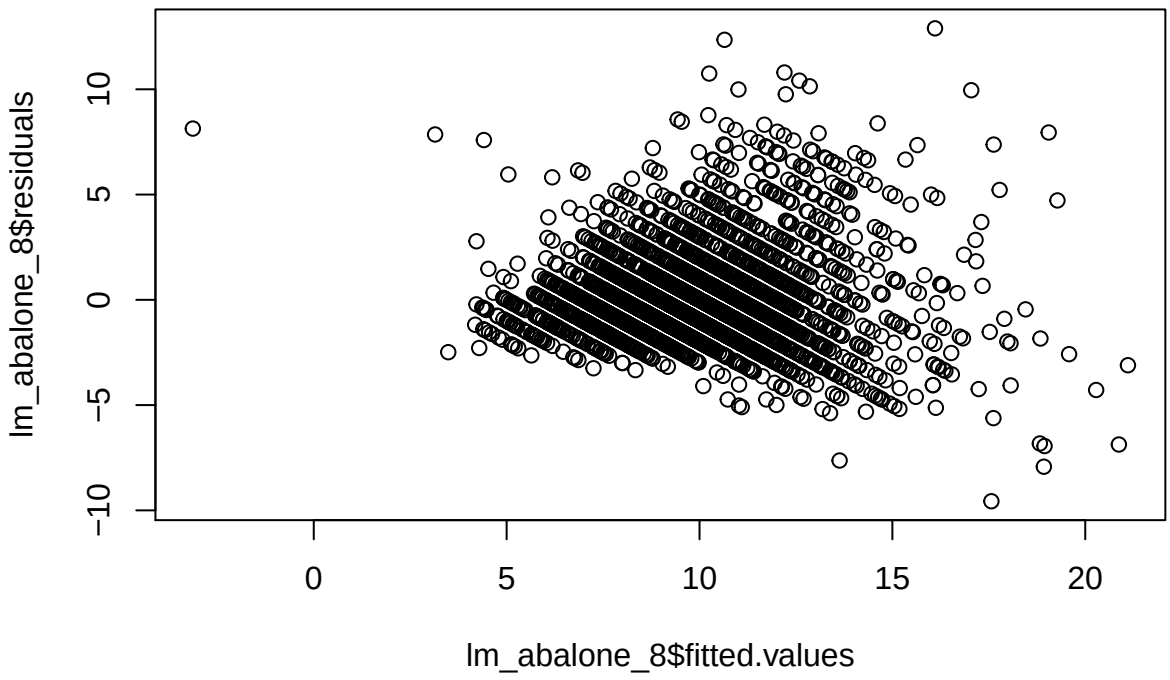
```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.4961  -1.4690  -0.5118   0.8738  16.6516
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.4952     0.3169  11.030  < 2e-16 ***
## Diameter      8.8568     1.2523   7.073 1.89e-12 ***
## Height       10.5032     1.7912   5.864 5.03e-09 ***
## WholeWeight  -5.7888     0.3486 -16.605  < 2e-16 ***
## ShellWeight  25.6677     1.0857  23.641  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.392 on 2919 degrees of freedom
## Multiple R-squared:  0.4527, Adjusted R-squared:  0.4519
## F-statistic: 603.5 on 4 and 2919 DF,  p-value: < 2.2e-16
```

Normality & Independent Tests:

We have made some implicit assumptions that needs to be verified.

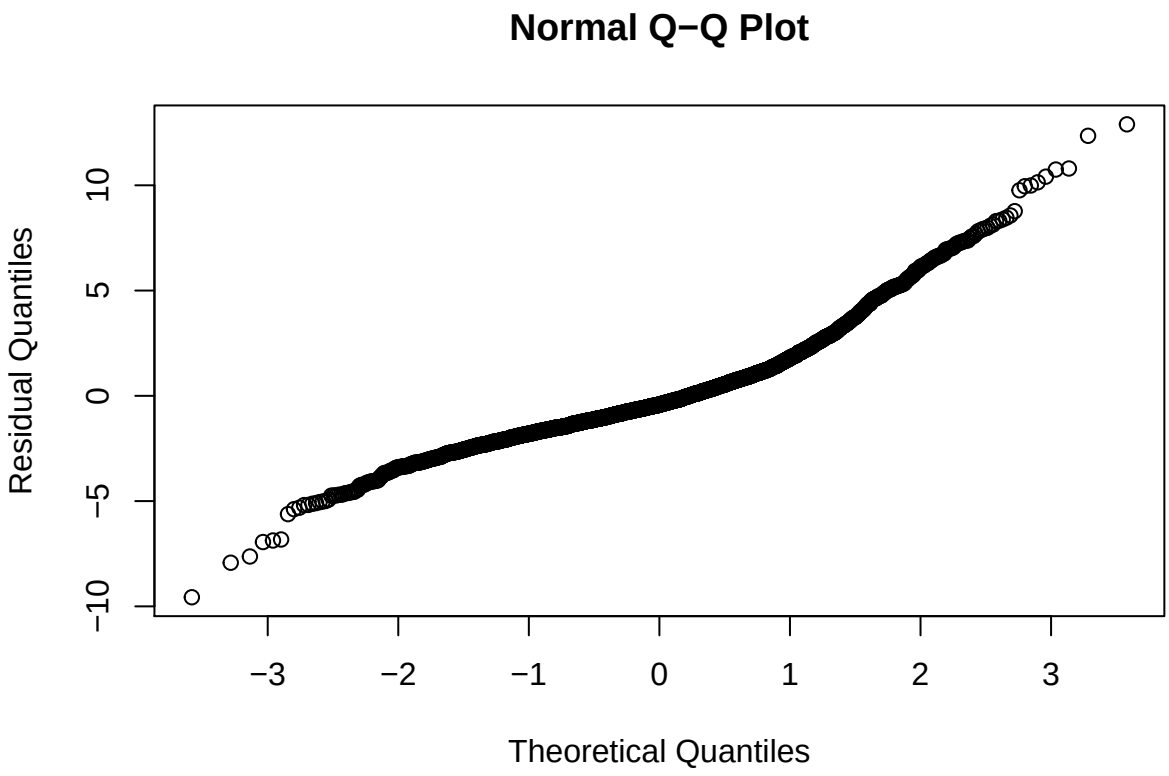
1. Observations must be independent.
2. Error must be normally distributed with mean=0.

```
# PLotting the residuals:
plot(lm_abalone_8$fitted.values, lm_abalone_8$residuals)
```




```
# No pattern as such ==> mutually independent

# Q-Q norm plot: (Compares quantiles of residuals to quantiles of normal
# distribution)
qqnorm(lm_abalone_8$residuals, ylab = "Residual Quantiles")
```




```
# Quantiles are approximately in one line
```

Predicting on Test Set:

```
pred <- predict(lm_abalone_8, test)

test_table <- table(pred, test$Rings)
```

Performance of the model:

```
rmse_test = sqrt((sum((test$Rings - pred)^2))/nrow(test))
# test$Rings = True Response pred = Estimated Response nrow(test) = Observations

# Sum of Squared Errors:
SSE <- sum((test$Rings - pred)^2)

# Sum of Squared Variation from mean:
SST <- sum((test$Rings - mean(test$Rings))^2)

# Coefficient of Determination:
R2 = 1 - SSE/SST
R2
```

```
## [1] 0.518848
```

```r
# Better fit if close to 1

# RMSE for train set
rmse_train <- sqrt(mean(lm_abalone_8$residuals^2))

# Ration of test to train set:
rmse_test/rmse_train
```

```
## [1] 0.9954316
```

Cross Validation:

```r
accs <- rep(0, 6)

for (i in 1:6) {
    # These indices indicate the interval of the test set
    indices <- (((i - 1) * round((1/6) * nrow(shuffled))) + 1):((i * round((1/6) *
        nrow(shuffled))))

    # Exclude them from the train set
    cv_train <- shuffled[-indices, ]

    # Include them in the test set
    cv_test <- shuffled[indices, ]

    model <- lm(Rings ~ Diameter + Height + WholeWeight + ShuckedWeight + VisceraWeight,
        cv_train)

    cv_pred <- predict(model, cv_test)

    cv_test_rmse = sqrt((sum((test$Rings - pred)^2))/nrow(test))

    accs[i] <- cv_test_rmse

}

mean(accs)
```

```
## [1] 2.225128
```