Determining a Prediction Model For Abalone Age

Joseph Gonzalez
UC Davis

December 9, 2019

Abstract

The most prominent feature of the abalone, or *haliotis*, is its shell's inner lining, which features a captivating iridescent surface. While the inner layer can be used as an art piece or as an attraction, it can also be used to determine the age of an abalone by cutting its shell through the cone, staining it, and counting its rings through a microscope. In this paper, we examine other measurements, like length, height, and shell weight, to predict the abalone's age and avoid the tedious and time-consuming task of counting its rings. We will use R to test the significance of each measurement and, eventually, generate a model. We will determine whether this model is a "good" model for predicting abalone age.

1 Introduction

Before using models to predict abalone age, biologists used microscopes to count the number of rings within the abalone's shell. This task, which includes cutting and staining the shell's cone, is laborious and requires high attention to details. To avoid this procedure, biologists hope to use physical measurements to predict abalone age. The goal of this project is to determine whether certain measurements can be used to predict abalone age. In the analysis, we determine which measurements are more significant in determining abalone age and which are not. The underlying motivation for this project is to promote further scientific exploration in determining the age of marine life. This is important because, in recent years, climate change and over-fishing have impacted most aquatic populations. If the prediction of marine life age could reduce killing off younger specimen, this could possibly alleviate some of the problems associated with the declining populations.

The UCI machine learning Repository, a collection of databases used for machine learning, provided the abalone data set for this project. The data consists of 9 variables with 4177 observations for each variable. The response variable is the number of rings and adding 1.5 to this measurement will result in the abalone's age. For the eight predictor variables, there is one qualitative variable and seven quantitative variables. The predictors include sex(M, F, and I), length(mm), diameter(mm), height(mm), whole weight(grams), shucked weight(grams), viscera weight(grams) and shell weight(grams).

1.1 Methods and Results

1.1.1 Missing Values, Variable Types, and Summary Statistics

First, I searched missing values, analyzed variables classes, and obtained summary statistics for each variable. From my search, the data did not have any missing values. It is important to identify and fix missing values because they could affect later analysis. Something that I did find concerning was that there are observed values of 0 in the height data column. In real research, I would ask the data collector to explain the meaning of these zeros to confirm that they are not missing values. For this project, I assumed that these values were too small for measurement because they are located in rows corresponding to infant abalone.

Next, I found the classes of each variable(figure 1) and discover all except one were numeric. In the data set, sex is the only categorical variable with 3 classes. In addition, I obtained the summary statistics for these variables(figures 2 and 3). The summary stats provide a range of the data and an idea of how the data is behaving. Two details that stood out from these outputs were that there are more male abalone and the means and medians are fairly close to each other. To supplement the summary statistics, I used histograms and charts to obtain a visual of the data.

1.1.2 Histograms and Charts

Histograms provide a visual of how the data behaves and this image can suggest whether the variable should be transformed. Figure 4, shows the histogram of the number of abalone rings. From this graph, we can see that the data is slightly right-skewed and this suggests that the rings data may need a transformation. To test this indication, I transformed the rings data by $\frac{1}{rings}$, $log(rings)$, and $\sqrt{rings}$. Figure 5 shows all 3 transformations and the original histogram. The histograms that look the most normal are the $\sqrt{rings}$ and $log(rings)$. For this project, I will use the log transformation on rings because when data is skewed right a log transformation will usually suffice for correction and we can also see in figure 6 that the box cox(with all x variables) shows that $\lambda = 0$ indicating a log transformation is needed. As a result, I applied the log transformation to the rings data and used this transformation throughout the rest of the analysis.

Figure 7 shows the histograms of the quantitative predictor variables. The length and diameter variables both appear to be skewed left and the height, whole weight, shucked weight, viscera weight, and shell weight appear to be skewed right. Furthermore, figure 8 shows the pie chart and bar graph for the sex categorical variable. These charts are included because traditional histograms cannot properly display categorical variables in a meaningful way. We can see from both these images that males have the most number of samples and females has the least number of samples.

1.1.3 Correlation Matrix, Pairwise Scatter Plot, and Side-By-Side Boxplot

Figure 9 contains the pairwise correlation coefficients between each predictor variable and between the predictor variables and the response variable. This matrix provides us with information about the relationships between each variable. In the first row or first column, we can see that each predictor variable has a moderate correlation with the response variable Y. In the rest of the matrix, we see that the pairwise correlation coefficients between the predictors are very high(most above 0.7). This is a sign that multicollinearity may be present between the variables and, as a result, I decided to look at the variance inflation factors for each variable(figure 10).

Except for height, all VIFs are more than 10, which is an indication of high multicollinearity. The VIFs are also more than one(the extreme being 109.6 for weight) and this means that the variance for our regression coefficients will be inflated due to intercorrelation. This s important because high multicollinearity will lead to large sampling variability and conflicts between T and F tests. Another important detail to note is that, in the model, a regression coefficient may only reflect the marginal effect a predictor

variable has on the response variable given the other predictors are also in the model.

Figure 11 shows that pairwise scatter plots of the variables. This further supports the correlation matrix because this plot displays the positive trends between the predictor variables. This image also reveals a possible curvilinear relationship between the predictor variables length, diameter, whole weight, shucked weight, viscera weight, and shell weight(see model selection section for further analysis).

Figure 12 shows the box plot of rings with respect to sex. This reveals how the amount of rings varies between the different sex categories. The distribution of rings is more symmetric between males and females. Furthermore, the number of rings appears to be less in the infant category.

1.1.4 Model Selection

Before I started the model selection process, I split the data into two random subsets with 2089 and 2088 observations. The subset with 2089 observation is used as the training data set and figure 13 shows that the distribution of each variable's measurements is similar in each data set. In addition, the categorical variable in each data set had similar totals based on category. Overall, this step is important because it allows for a model to be built using the training data and to check its generalization using the validation data.

1.1.5 Best Subsets Regression

To initiate the best subsets regression, I first fit the model with all first-order terms(see figure 14 for R output and figure A for equation). In the R output, we can see the model has a moderate fit($R_a^2 = 0.6036$) data and relatively low standard error($S_e = 0.2052$). All variables, except for the male category(must keep in to maintain whole categorical variable), are significant for the t-test at a significance level of $\alpha = 0.05$. Overall, this model looks decently adequate. However, there appears to be non-constant variance in its residuals vs fitted values plot(figure 16). This indicates the presence of outliers or the need for a second-order term in the model. Also, its Q-Q plot is slightly heavy tail right. We can also see in this graph that there may be possible outliers in the data. It is important to identify outliers because they may overly influence our regression line.

Next, I used model 1 to conduct the best subset regression. For this procedure, I generated the 9 best models(subsets of model 1) for each model size and ranks them based on SSE. In figure 17, we see that the full model has the best SSE, R^2, R_a^2, and its C_p= 10, which means there is no in-sample bias. The model with 8 variables has the best aic and the model with 7 variables has the best bic. This table is important because it provides details on which models are adequate based on bias(C_p), variance, fit(SSE, R^2, R_a^2), and model complexity(aic ,bic). Ultimately, model 1 does appear to be the best model for this set in regards to finding a predictive model because it has the best fit and has relatively low indicators for model complexity.

1.1.6 Step-wise Regression

After using subsets regression, I used step-wise regression to further analyze the first-order model with respect to aic and bic. In this procedure, I used forward step-wise and

backward step-wise because forward selection and backward elimination do not work as well when multicollinearity is present.

Using the aic criterion, the forward step-wise process resulted in the full model(figure 18) with all first-order terms. The aic from this process matched the aic from the best subsets regression. In this case, forward selection, backward elimination, and backward step-wise resulted in the full model. This further solidifies model 1 being an adequate model. Next, I used bic criterion for the set of first-order variables. This process produced model 2 minus the length variable(see figure 19). The bic value for this model is slightly higher than the bic values from the best subsets regression(suboptimal). In figure 20, we can see that the errors are fairly normal and there is non-constant variance in these residuals vs fitted values graphs. This indicates that a higher-order term may be needed for the model.

After testing the first-order model, I next conducted model selection with interaction terms. I first used forward step-wise regression with respect to aic and generated model 3 in figure 21. The first issue with this model is that it is relatively large. The full with all interaction terms(used for conducting model selection) has a total of 45 coefficients and this model has 23. So, I included using the bic criterion for the previous step and this step to put more penalty on larger models(log(n)*p) because they are often complex and have larger sampling variability. Using this approach with bic, I generated models 4(figure 22) and model 5(figure 23) using forward step-wise and backward step-wise. The diagnostic plots for the full interaction model and models 3 through 5 are in figure 24. The plots almost identical and show that the errors are fairly normal(with some possible outliers) and that variance is more constant than the first-order models. I also explored quadratic models with no interaction terms. I included the full model with all second-order quadratic terms(figure 25) and a subset generated from backward step-wise under bic(figure 26).

Figure 27 shows models and their values for SSE, R^2, R_a^2, aic, bic, C_P and $Press_p$. To find the best predictive model, I looked for the models that had a relatively low $Press_p$, high R_a^2, and moderate values for the other criterion. The first model that I chose to use for validation was model 3 because it has the lowest C_P(low bias), lowest aic(goodness of fit and model complexity), lowest $Press_p$ and relatively low bic. It also has the best R^2 and R_a^2 compared to the other models(not full model with all interactions) in question. I also found that model 5 had the second lowest aic, lowest bic, a relatively low C_p and the second lowest $Press_p$. As a result, I used these models for validation.

1.1.7 Validation

For validation, I fitted models 3 and 5 to the validation set, compared its estimators to the training set regression estimators, analyzed the percentage differences between the estimators and their standard deviations, and found the $MSEP_v$(measures predictability). For model 3(figure 28), we can see that there many changes in the estimators' signs compared to the training set regression(figure 21), whereas model 5(figure 30) only has only change in sign compared to the training set regression(figure 23). I also compared the percentage changes in estimators and standard errors(figures 29 and 31) between both models and I discovered that model 3's changes in most cases were higher compared

to model 5. Lastly, the $MSEP_v$ for model 3 is 0.03678189 and the $MSEP_v$ for model 5 is 0.03650119(both close to their respective $\frac{SSE}{n}$ and $\frac{Press_p}{n}$, which means no severe over-fitting). Overall, model 5 has a smaller $MSEP_v$(good predictability), fewer coefficients(less complex), smaller changes in estimation parameters and standard errors, and fewer estimator sign changes compared to model 3. Therefore, I chose model 5 to be the final model and the equation, its assumptions and anova output can be found in figures 32 and 34. I also fitted the model to the entire data set(figure 33) and noticed that, compared to the training set output, no estimator changed signed and that the R^2, R_a^2, and MSE were consistent.

1.1.8 Identifying Outliers

After selecting a final model, I conducted model diagnostics(figure 35) and found potential outlying cases. First, I looked for outliers in Y by using standardized residuals and found that observations 237, 1217 2052, 2182 and 2628 were outlying in Y. Then, I used leverage points to find outliers in the X values and discover that all the values in figure 36 were outlying in X. In this figure, we can also see that observations 237, 1217, 2052, and 2628 are also outlying in X. To find the most influential cases, I used cooks distance on the observations(37) and found that observation 2052 was the most influential. It is important to identify outliers because they can cause non-normality and non-constant variance(Q-Q plot figure 35). Removing outliers, such as observation 2052, may improve the fit and also set constraints on the measurements that can be entered into the model(avoiding extrapolation).

1.2 Conclusion and Discussion

My model shows that all the predictor variables given with the data, and some of their interactions, are important in determining the number of abalone rings. This model can potentially help biologists predict abalone age and avoid counting the rings through a microscope. Considering the entire set of interaction terms and quadratic terms, this model is simplistic and, according to the results, has "good" predictability without overfitting. I believe my results are relatively general. The only concern for generalization is the estimator that changed signs from the training data set to the validation set. This concern could potentially affect predictions for some sample sets.

The limitations of this investigation was time, page constraints, lack of biological information, and accessibility to the data collectors. If there were more time and fewer page constraints, I would have more thoroughly analyzed the data, tried to investigation more models(interactions with quadratics), and possibly correct the change in estimator signs for the model. Furthermore, I would have tried to learn about the abalone's biological qualities to make more informed inferences about the problem and the importance of each variable. If I had access to the data collectors, I could have had some of my questions about the data answered and further supported my results on outlying values. Overall, I am content with the results given the amount of time permitted for this project. I hope that biologists were able to find a model to predict abalone age and make their work less strenuous.

A R Figures, tables, and outputs

```
          Sex        Length      Diameter        Height  Whole.weight Shucked.weight
     "factor"     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
Viscera.weight  Shell.weight         Rings
     "numeric"     "numeric"     "integer"
```

Figure 1: Variable Classes

```
$Sex
   F    I    M
1307 1342 1528

$Length
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.075   0.450   0.545   0.524   0.615   0.815

$Diameter
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0550  0.3500  0.4250  0.4079  0.4800  0.6500

$Height
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.1150  0.1400  0.1395  0.1650  1.1300
```

Figure 2: Summary Statistics 1

```
$Whole.weight
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0020  0.4415  0.7995  0.8287  1.1530  2.8255

$Shucked.weight
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0010  0.1860  0.3360  0.3594  0.5020  1.4880

$Viscera.weight
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0005  0.0935  0.1710  0.1806  0.2530  0.7600

$Shell.weight
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0015  0.1300  0.2340  0.2388  0.3290  1.0050

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

Figure 3: Summary Statistics 2

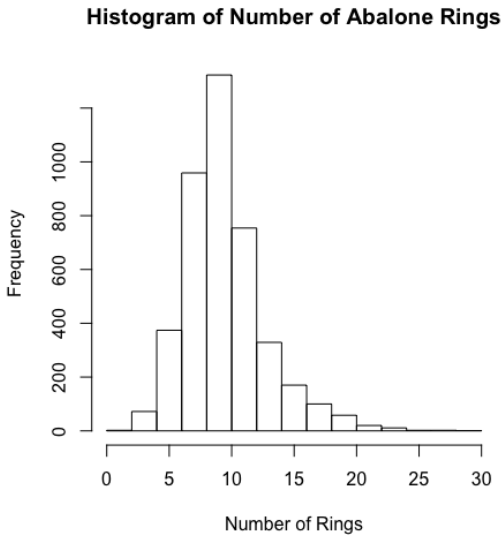


Figure 4: Rings Histogram

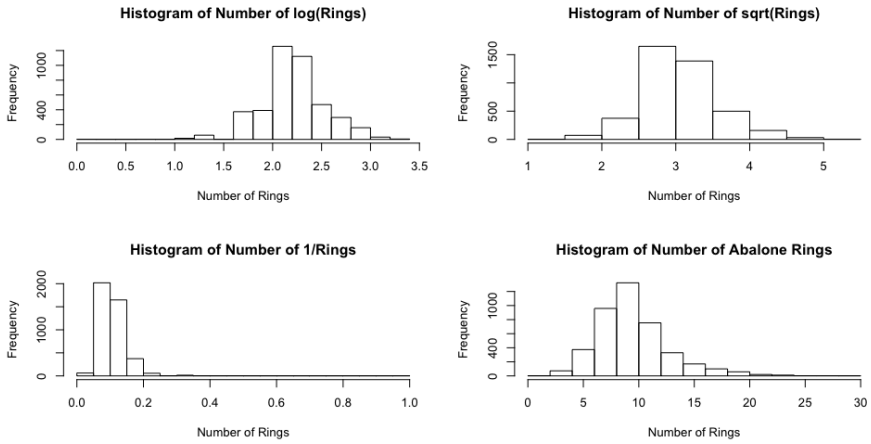


Figure 5: Rings Transformation Comparison

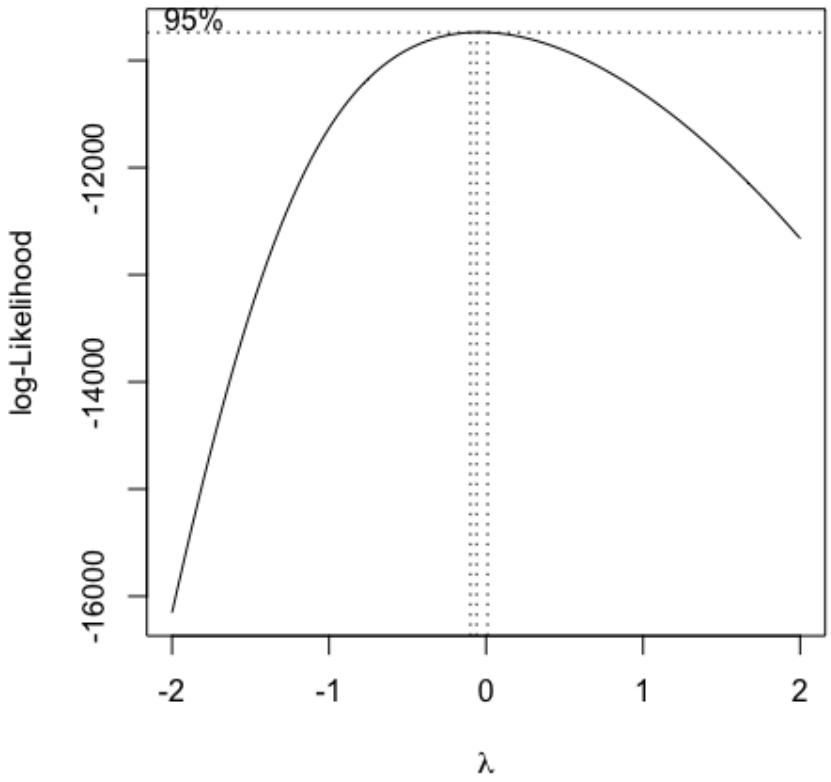


Figure 6: Box Cox Procedure With All First-order Predictors

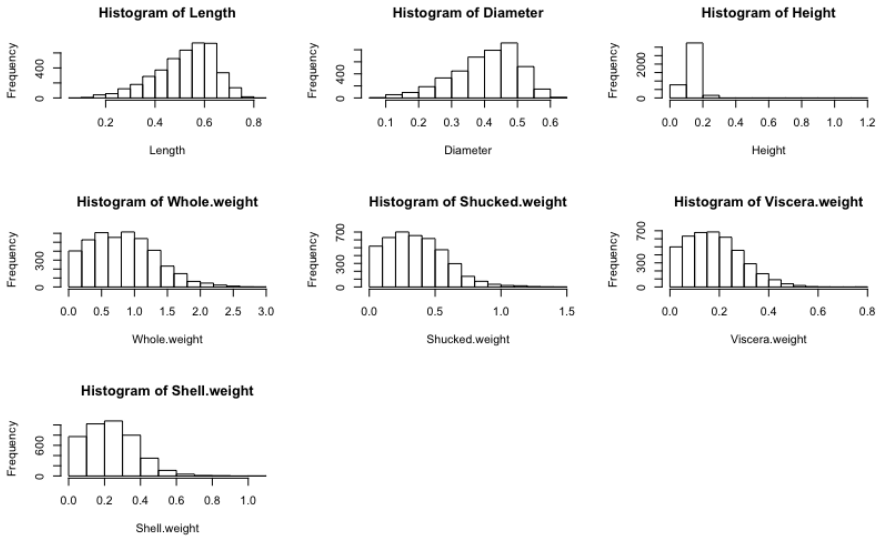


Figure 7: Histogram of Quantitative Predictor Variables

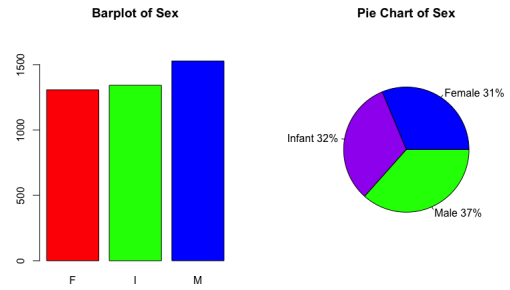


Figure 8: Pie Chart and Bar Graph For Sex

	Rings	Length	Diameter	Height	Whole.weight	Shucked.weight	Viscera.weight	Shell.weight
Rings	1.0000000	0.6537687	0.6683216	0.6253101	0.5959416	0.4912156	0.5659361	0.6664236
Length	0.6537687	1.0000000	0.9868116	0.8275536	0.9252612	0.8979137	0.9030177	0.8977056
Diameter	0.6683216	0.9868116	1.0000000	0.8336837	0.9254521	0.8931625	0.8997244	0.9053298
Height	0.6253101	0.8275536	0.8336837	1.0000000	0.8192208	0.7749723	0.7983193	0.8173380
Whole.weight	0.5959416	0.9252612	0.9254521	0.8192208	1.0000000	0.9694055	0.9663751	0.9553554
Shucked.weight	0.4912156	0.8979137	0.8931625	0.7749723	0.9694055	1.0000000	0.9319613	0.8826171
Viscera.weight	0.5659361	0.9030177	0.8997244	0.7983193	0.9663751	0.9319613	1.0000000	0.9076563
Shell.weight	0.6664236	0.8977056	0.9053298	0.8173380	0.9553554	0.8826171	0.9076563	1.0000000

Figure 9: Correlation Matrix

	VIF
Length	40.771813
Diameter	41.845452
Height	3.559939
Whole.weight	109.592750
Shucked.Weight	28.353191
Viscera.weight	17.346276
Shell.weight	21.258289

Figure 10: Variance Inflation Factors

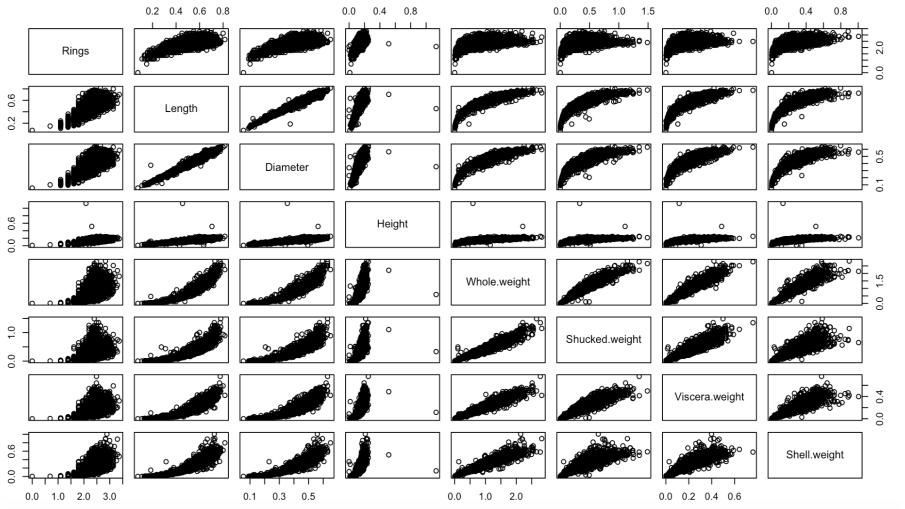


Figure 11: Pairwise Scatter Plot

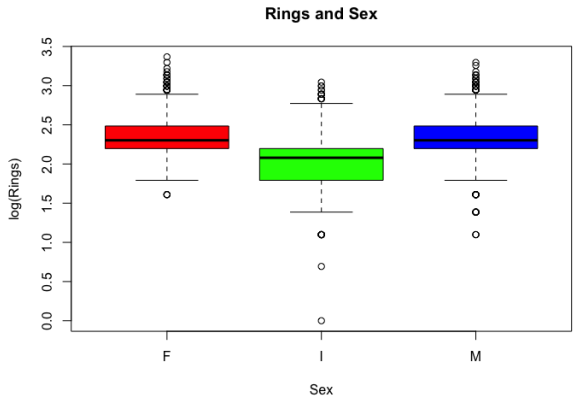


Figure 12: Side-by-Side Boxplot

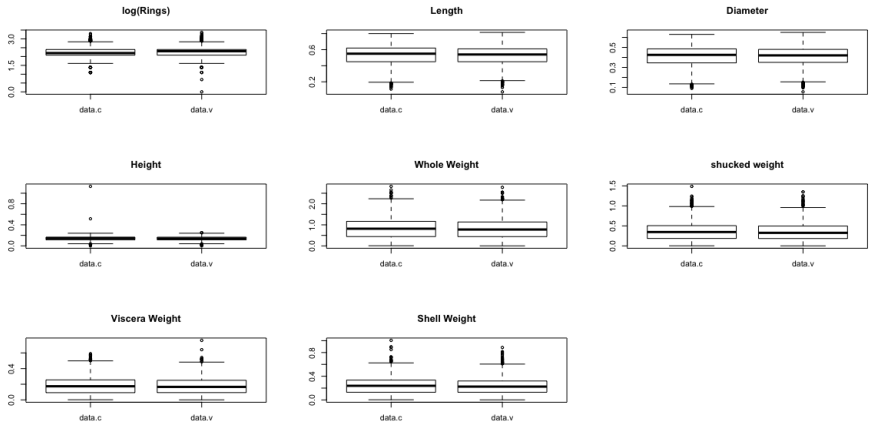


Figure 13: Training and Validation Boxplots

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.33426    0.03815  34.973  < 2e-16 ***
SexI          -0.08787    0.01372  -6.405 1.86e-10 ***
SexM           0.01501    0.01099   1.366  0.17207
Length         0.65979    0.24706   2.671  0.00763 **
Diameter       1.35648    0.30495   4.448 9.12e-06 ***
Height         0.78509    0.16512   4.755 2.12e-06 ***
Whole.weight   0.60132    0.10572   5.688 1.47e-08 ***
Shucked.weight -1.68705    0.11733 -14.379  < 2e-16 ***
Viscera.weight -0.82515    0.17574  -4.695 2.84e-06 ***
Shell.weight   0.74560    0.16435   4.537 6.04e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2054 on 2079 degrees of freedom
Multiple R-squared:  0.6054,    Adjusted R-squared:  0.6036
F-statistic: 354.3 on 9 and 2079 DF,  p-value: < 2.2e-16
```

Figure 14: R Output of Model With All First-Order Variables

$$\hat{Y} = 1.33426 - 0.08787X_{i1,I} + 0.01501X_{i1,M} + 0.65979X_{i2} + 1.356489X_{i3}$$
$$+ 0.78509\mathrm{X}_{i4} + 0.60132X_{i5} - 1.68705X_{i6} - 0.82515X_{i7} + 0.74560X_{i8}$$

Figure 15: Model 1 Regression Equation

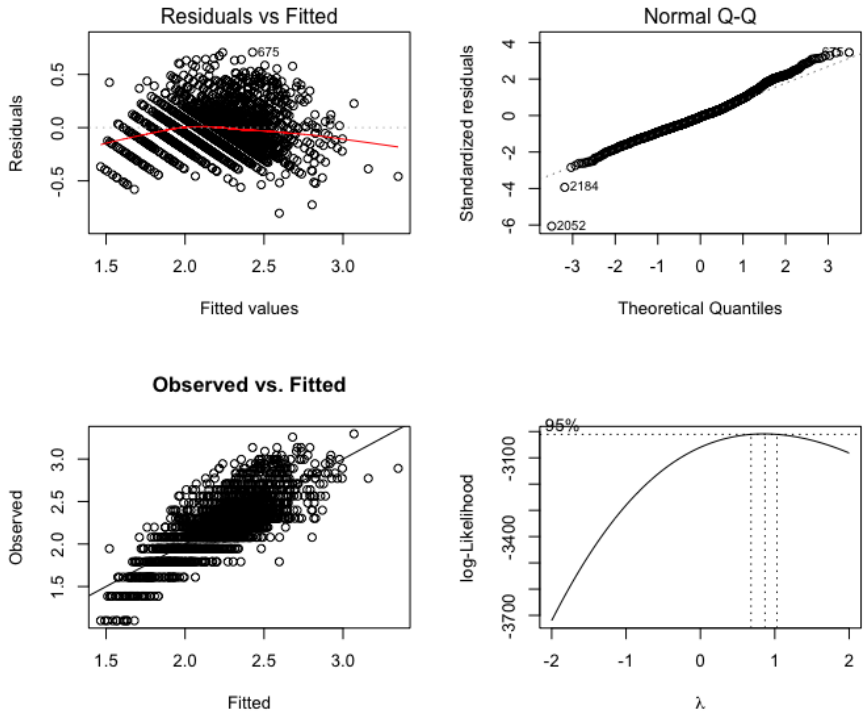


Figure 16: Model 1 Diagnostics

	(Intercept)	SexI	SexM	Length	Diameter	Height	Whole.weight	Shucked.weight	Viscera.weight	Shell.weight
none	1	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	1
2	1	0	0	0	1	0	0	1	0	0
3	1	0	0	0	1	0	0	1	0	1
4	1	1	0	0	1	0	0	1	0	1
5	1	1	0	0	1	1	0	1	0	1
6	1	1	0	0	1	1	1	1	1	0
7	1	1	0	0	1	1	1	1	1	1
8	1	1	0	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1

	sse	R^2	R^2_a	Cp	aic	bic
none	222.2156	0.0000	0.0000	158473.3777	-4673.372	-4679.016
1	120.3478	0.4584	0.4582	768.0699	-5958.120	-5946.832
2	109.2075	0.5086	0.5081	505.9673	-6159.038	-6142.105
3	93.4781	0.5793	0.5787	135.0723	-6481.925	-6459.348
4	90.4672	0.5929	0.5921	65.6947	-6548.318	-6520.095
5	89.5049	0.5972	0.5962	44.8819	-6568.657	-6534.790
6	88.9247	0.5998	0.5987	33.1262	-6580.244	-6540.733
7	88.0844	0.6036	0.6023	15.2054	-6598.078	-6552.922
8	87.7748	0.6050	0.6035	9.8661	-6603.432	-6552.632
9	87.6961	0.6054	0.6036	10.0000	-6603.307	-6546.862

Figure 17: Best Subsets Regression

```
Step:  AIC=-6603.31
data.c$Rings ~ Shell.weight + Shucked.weight + Diameter + Sex +
    Height + Whole.weight + Viscera.weight + Length
```

Figure 18: Forward Step-wise: Model and AIC

```
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     1.36672    0.03622  37.737  < 2e-16 ***
Shell.weight    0.73475    0.16454   4.466 8.41e-06 ***
Shucked.weight -1.66610    0.11724 -14.211  < 2e-16 ***
Diameter        2.09521    0.12855  16.299  < 2e-16 ***
SexI           -0.08495    0.01370  -6.202 6.70e-10 ***
SexM            0.01582    0.01100   1.438    0.151
Height          0.79403    0.16533   4.803 1.68e-06 ***
Whole.weight    0.59692    0.10586   5.639 1.95e-08 ***
Viscera.weight -0.78135    0.17523  -4.459 8.68e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2057 on 2080 degrees of freedom
Multiple R-squared:  0.604,     Adjusted R-squared:  0.6025
F-statistic: 396.6 on 8 and 2080 DF,  p-value: < 2.2e-16
```

```
Step:  AIC=-6547.35
data.c$Rings ~ Shell.weight + Shucked.weight + Diameter + Sex +
    Height + Whole.weight + Viscera.weight
```

Figure 19: Model 2(Forward Step-wise BIC)

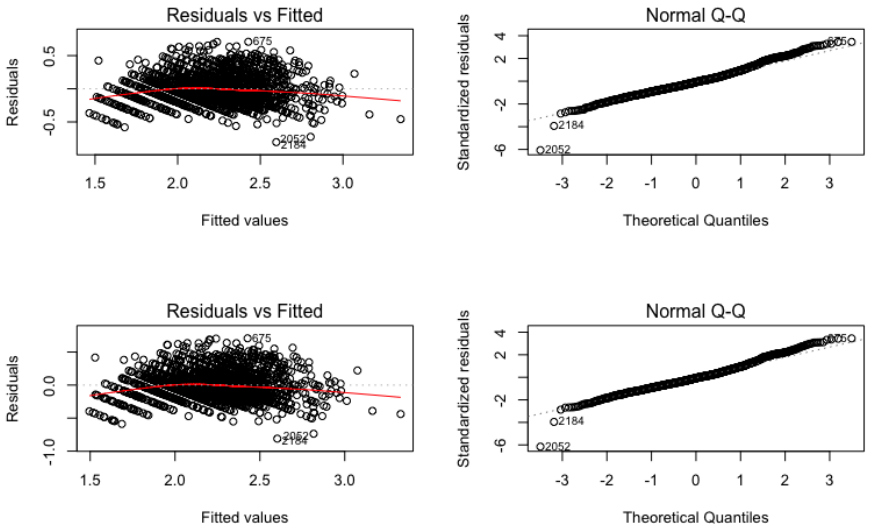


Figure 20: Model Diagnostics for Two First Order Step-wise models

```
Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                  0.24045    0.15375   1.564  0.11801
Shell.weight                 2.48250    0.32571   7.622 3.79e-14 ***
Shucked.weight              -7.93121    0.62323 -12.726  < 2e-16 ***
Diameter                    11.23761    1.46758   7.657 2.90e-14 ***
Whole.weight                 1.87104    0.38224   4.895 1.06e-06 ***
SexI                         0.06348    0.11572   0.549  0.58334
SexM                         0.11363    0.10290   1.104  0.26964
Viscera.weight              -0.63982    0.16411  -3.899 9.98e-05 ***
Height                      -1.43082    1.28540  -1.113  0.26578
Length                       0.34280    0.94927   0.361  0.71805
Shucked.weight:Diameter     15.89874    1.29128  12.312  < 2e-16 ***
Shucked.weight:SexI          1.04970    0.20125   5.216 2.01e-07 ***
Shucked.weight:SexM          0.11034    0.10120   1.090  0.27570
Shell.weight:SexI           -0.22316    0.32165  -0.694  0.48788
Shell.weight:SexM            0.32494    0.17063   1.904  0.05701 .
Diameter:Height            -42.29539    8.75277  -4.832 1.45e-06 ***
Diameter:Length            -15.30436    1.55072  -9.869  < 2e-16 ***
Height:Length               38.78603    7.30402   5.310 1.21e-07 ***
Diameter:SexI               -0.99104    0.43279  -2.290  0.02213 *
Diameter:SexM               -0.55407    0.34406  -1.610  0.10746
Shell.weight:Shucked.weight -2.85121    0.50669  -5.627 2.08e-08 ***
Diameter:Whole.weight       -2.29141    0.73743  -3.107  0.00191 **
Shucked.weight:Height       -2.62680    1.43785  -1.827  0.06786 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1881 on 2066 degrees of freedom
Multiple R-squared:  0.6712,    Adjusted R-squared:  0.6677
F-statistic: 191.7 on 22 and 2066 DF,  p-value: < 2.2e-16
```

Figure 21: Model 3 With Interaction(Forward-Stepwise AIC)

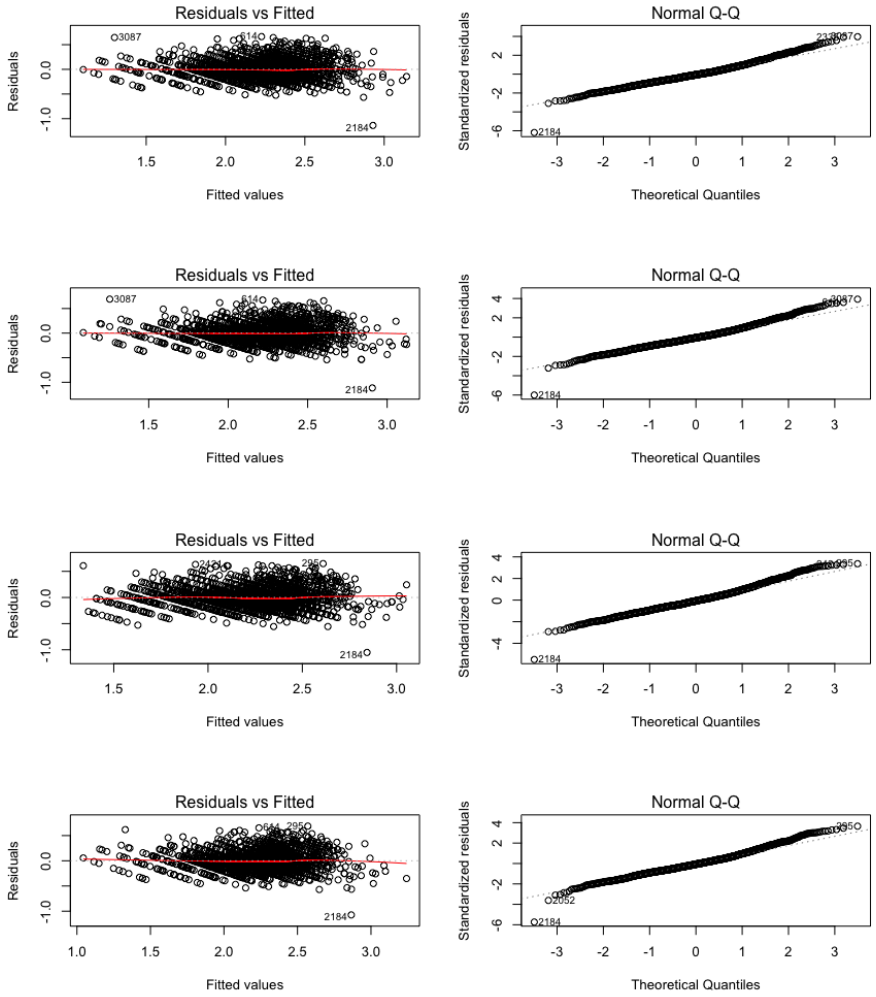
```
Step:  AIC=-6793.16
data.c$Rings ~ Shell.weight + Shucked.weight + Diameter + Whole.weight +
    Sex + Viscera.weight + Shell.weight:Diameter + Shucked.weight:Diameter +
    Shucked.weight:Whole.weight + Shucked.weight:Sex
```

Figure 22: Model 4(Forward Step-wise BIC)

```
Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                  0.40052    0.08630   4.641 3.68e-06 ***
SexI                        -0.21462    0.02750  -7.805 9.38e-15 ***
SexM                        -0.04667    0.02407  -1.939 0.052645 .
Length                       0.19539    0.94525   0.207 0.836259
Diameter                    10.93410    1.26025   8.676  < 2e-16 ***
Height                       0.06640    0.88663   0.075 0.940313
Whole.weight                 0.74966    0.09825   7.630 3.56e-14 ***
Shucked.weight              -6.23138    0.37445 -16.641  < 2e-16 ***
Viscera.weight              -0.59888    0.16434  -3.644 0.000275 ***
Shell.weight                 3.21431    0.26041  12.343  < 2e-16 ***
SexI:Shucked.weight          0.50304    0.08312   6.052 1.69e-09 ***
SexM:Shucked.weight          0.11801    0.04862   2.427 0.015306 *
Length:Diameter            -15.07903    1.02201 -14.754  < 2e-16 ***
Length:Height               38.53886    6.58826   5.850 5.71e-09 ***
Diameter:Height            -48.57024    7.84484  -6.191 7.17e-10 ***
Diameter:Shucked.weight     12.36194    0.95731  12.913  < 2e-16 ***
Shucked.weight:Shell.weight -3.85333    0.38806  -9.930  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.189 on 2072 degrees of freedom
Multiple R-squared:  0.667,     Adjusted R-squared:  0.6644
F-statistic: 259.4 on 16 and 2072 DF,  p-value: < 2.2e-16
```

```
Step:  AIC=-6848.23
data.c$Rings ~ Sex + Length + Diameter + Height + Whole.weight +
    Shucked.weight + Viscera.weight + Shell.weight + Sex:Shucked.weight +
    Length:Diameter + Length:Height + Diameter:Height + Diameter:Shucked.weight +
    Shucked.weight:Shell.weight
```

Figure 23: Model 5(Backward Step-wise BIC)



Order: Full Model, Model 3, Model 4 and Model 5

Figure 24: Interaction Models' Diagnostic Plots

```
lm(formula = polyfitdata$V1 ~ factor(polyfitdata$V2) + Lengthcent +
    Diametercent + heightcent + Whole.weightcent + Shucked.weightcent +
    Viscera.weightcent + Shell.weightcent + I(Lengthcent^2) +
    I(Diametercent^2) + I(heightcent^2) + I(Whole.weightcent^2) +
    I(Shucked.weightcent^2) + I(Viscera.weightcent^2) + I(Shell.weightcent^2),
    data = polyfitdata)
```

Figure 25: Model With All Quadratic Terms

```
Step:  AIC=-6831.55
polyfitdata$V1 ~ factor(polyfitdata$V2) + heightcent + Whole.weightcent +
    Shucked.weightcent + Viscera.weightcent + Shell.weightcent +
    I(Diametercent^2) + I(heightcent^2) + I(Whole.weightcent^2) +
    I(Shucked.weightcent^2) + I(Viscera.weightcent^2)
```

Figure 26: Model 6 (Backward Step-wise BIC)

	SSE	R^2	R_a^2	aic	bic	C_p	Press_p**
Model 1	87.696	0.6054	0.6036	-6603.309	-6546.865	414.85491	92.70924
Model 2	87.997	0.6040	0.6025	-6598.151	-6598.151	421.38028	93.04910
Full Model(Interactions)	72.202	0.6751	0.6681	-6939.429	-6685.429	46.01109	83.03287
Model 3	73.060	0.6712	0.6677	-6958.751	-6828.929	26.31263	77.60008
Model 4	77.092	0.6531	0.6511	-6866.533	-6793.155	120.51285	78.28871
Model 5	73.995	0.6670	0.6644	-6944.186	-6848.230	40.79507	77.98137
Full Model(Quadratics)	75.252	0.6614	0.6587	-6908.997	-6813.041	76.39767	90.35031
Model 6	75.501	0.6602	0.6581	-6908.096	-6829.074	77.45021	90.56377

Figure 27: Criteria Chart For All Models From Model Selection

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	0.36967	0.16061	2.302	0.021452	*
Shell.weight	1.64183	0.31858	5.154	2.80e-07	***
Shucked.weight	-4.49931	0.49374	-9.113	< 2e-16	***
Diameter	3.69482	1.49137	2.477	0.013311	*
Whole.weight	0.70748	0.27238	2.597	0.009461	**
SexI	0.05345	0.12274	0.436	0.663224	
SexM	0.09114	0.10493	0.869	0.385163	
Viscera.weight	-0.49922	0.15725	-3.175	0.001523	**
Height	6.30155	1.88851	3.337	0.000863	***
Length	3.48988	0.92910	3.756	0.000177	***
Shucked.weight:Diameter	6.50322	1.36061	4.780	1.88e-06	***
Shucked.weight:SexI	0.95195	0.19302	4.932	8.80e-07	***
Shucked.weight:SexM	0.19238	0.10611	1.813	0.069977	.
Shell.weight:SexI	0.55628	0.30416	1.829	0.067559	.
Shell.weight:SexM	0.08380	0.16938	0.495	0.620842	
Diameter:Height	-6.92753	8.52390	-0.813	0.416473	
Diameter:Length	-7.93244	1.88399	-4.210	2.66e-05	***
Height:Length	-5.26132	7.78272	-0.676	0.499100	
Diameter:SexI	-1.28782	0.45727	-2.816	0.004904	**
Diameter:SexM	-0.45831	0.35034	-1.308	0.190957	
Shell.weight:Shucked.weight	-1.85854	0.50422	-3.686	0.000234	***
Diameter:Whole.weight	0.07378	0.55330	0.133	0.893930	
Shucked.weight:Height	2.54334	2.54890	0.998	0.318484	

```
Residual standard error: 0.1866 on 2065 degrees of freedom
Multiple R-squared:  0.6477,    Adjusted R-squared:  0.6439
F-statistic: 172.6 on 22 and 2065 DF,  p-value: < 2.2e-16
```

Figure 28: Validation Output For Model 3

(Intercept)	Shell.weight	Shucked.weight	Diameter
53.744	33.864	43.271	67.121
Whole.weight	SexI	SexM	Viscera.weight
62.188	15.799	19.791	21.975
Height	Length	Shucked.weight:Diameter	Shucked.weight:SexI
540.417	918.057	59.096	9.313
Shucked.weight:SexM	Shell.weight:SexI	Shell.weight:SexM	Diameter:Height
74.359	349.271	74.212	83.621
Diameter:Length	Height:Length	Diameter:SexI	Diameter:SexM
48.169	113.565	29.946	17.284
Shell.weight:Shucked.weight	Diameter:Whole.weight	Shucked.weight:Height	
34.816	103.220	196.823	

(Intercept)	Shell.weight	Shucked.weight	Diameter
4.458	2.191	20.777	1.621
Whole.weight	SexI	SexM	Viscera.weight
28.740	6.061	1.964	4.179
Height	Length	Shucked.weight:Diameter	Shucked.weight:SexI
46.920	2.125	5.369	4.089
Shucked.weight:SexM	Shell.weight:SexI	Shell.weight:SexM	Diameter:Height
4.858	5.438	0.737	2.615
Diameter:Length	Height:Length	Diameter:SexI	Diameter:SexM
21.492	6.554	5.656	1.825
Shell.weight:Shucked.weight	Diameter:Whole.weight	Shucked.weight:Height	
0.487	24.970	77.271	

Figure 29: Model 3: Percent Changes in Estimators(above) and Standard Errors(below)

```
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)                 0.71200    0.08483   8.394  < 2e-16 ***
SexI                       -0.27501    0.02742 -10.031  < 2e-16 ***
SexM                       -0.05381    0.02341  -2.299 0.021624 *
Length                      3.61567    0.92564   3.906 9.68e-05 ***
Diameter                    2.37389    1.23185   1.927 0.054105 .
Height                      4.94086    1.35640   3.643 0.000276 ***
Whole.weight                0.72689    0.08272   8.788  < 2e-16 ***
Shucked.weight             -4.28525    0.33824 -12.669  < 2e-16 ***
Viscera.weight             -0.46861    0.15597  -3.004 0.002692 **
Shell.weight                1.87449    0.24617   7.615 4.00e-14 ***
SexI:Shucked.weight         0.70903    0.08608   8.237 3.09e-16 ***
SexM:Shucked.weight         0.10734    0.04888   2.196 0.028211 *
Length:Diameter            -8.30737    1.16632  -7.123 1.45e-12 ***
Length:Height              -5.06382    6.51697  -0.777 0.437235
Diameter:Height            -1.80099    7.66176  -0.235 0.814184
Diameter:Shucked.weight     7.30256    0.92657   7.881 5.19e-15 ***
Shucked.weight:Shell.weight -2.06315    0.38973  -5.294 1.32e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1867 on 2071 degrees of freedom
Multiple R-squared:  0.6461,    Adjusted R-squared:  0.6434
F-statistic: 236.3 on 16 and 2071 DF,  p-value: < 2.2e-16
```

Figure 30: Validation Output For Model 5

(Intercept)	SexI
77.766	28.135
SexM	Length
15.306	1750.496
Diameter	Height
78.289	7341.445
Whole.weight	Shucked.weight
3.038	31.231
Viscera.weight	Shell.weight
21.752	41.683
SexI:Shucked.weight	SexM:Shucked.weight
40.949	9.036
Length:Diameter	Length:Height
44.908	113.140
Diameter:Height	Diameter:Shucked.weight
96.292	40.927
Shucked.weight:Shell.weight	
46.458	

(Intercept)	SexI
1.704	0.301
SexM	Length
2.738	2.075
Diameter	Height
2.253	52.983
Whole.weight	Shucked.weight
15.813	9.672
Viscera.weight	Shell.weight
5.092	5.469
SexI:Shucked.weight	SexM:Shucked.weight
3.559	0.541
Length:Diameter	Length:Height
14.121	1.082
Diameter:Height	Diameter:Shucked.weight
2.334	3.211
Shucked.weight:Shell.weight	
0.428	

Figure 31: Model 5: Percent Changes in Estimators(above) and Standard Errors(below)

Model Assumptions:
1. Error terms are i.i.d, normal and have constant variance.
2. Each observation is independent and randomly obtained.

$$Y = \beta_0 + \beta_1 X_{i1,I} + \beta_2 X_{i1,M} + \beta_3 X_{i2} + \beta_4 X_{i3} + \beta_5 X_{i4} +$$

$$\beta_6 X_{i5} + \beta_7 X_{i6} + \beta_8 X_{i7} + \beta_9 X_{i8} + \beta_{10} X_{i1,I} X_{i6} + \beta_{11} X_{i1,M} X_{i6} + \beta_{12} X_{i2} X_{i3} + \beta_{13} X_{i2} X_{i4} + \beta_{14} X_{i3} X_{i4} + \beta_{15} X_{i3} X_{i6} + \beta_{16} X_{i6} X_{i8} + \xi$$

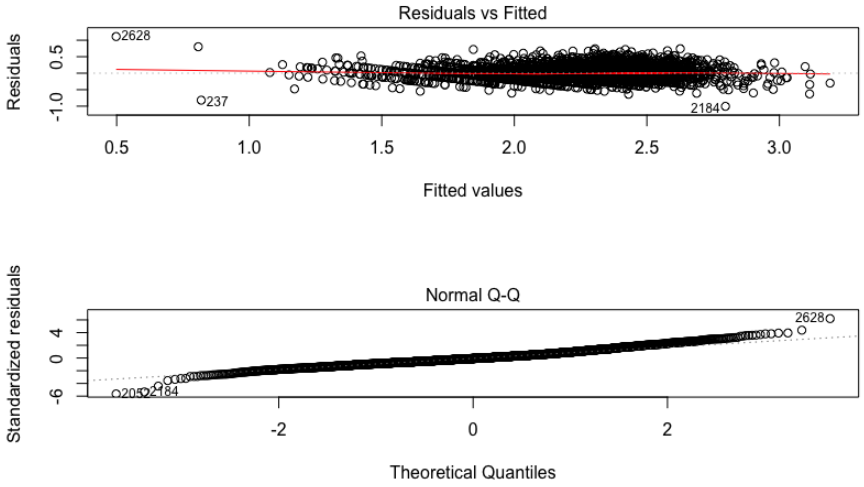
Figure 32: Final Model Assumptions and Equation

```
Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                  0.57613    0.06050   9.522  < 2e-16 ***
SexI                        -0.24529    0.01946 -12.606  < 2e-16 ***
SexM                        -0.04857    0.01682  -2.888  0.00389 **
Length                       1.89617    0.66265   2.861  0.00424 **
Diameter                     6.97232    0.87419   7.976 1.94e-15 ***
Height                       1.15801    0.71849   1.612  0.10710
Whole.weight                 0.75024    0.06320  11.870  < 2e-16 ***
Shucked.weight              -5.16064    0.25002 -20.641  < 2e-16 ***
Viscera.weight              -0.51470    0.11337  -4.540 5.78e-06 ***
Shell.weight                 2.53341    0.17809  14.226  < 2e-16 ***
SexI:Shucked.weight          0.61221    0.05993  10.215  < 2e-16 ***
SexM:Shucked.weight          0.10892    0.03447   3.160  0.00159 **
Length:Diameter            -12.15960    0.74054 -16.420  < 2e-16 ***
Length:Height               18.10974    4.60966   3.929 8.68e-05 ***
Diameter:Height            -24.66462    5.48892  -4.494 7.20e-06 ***
Diameter:Shucked.weight      9.60007    0.66166  14.509  < 2e-16 ***
Shucked.weight:Shell.weight -2.93124    0.27395 -10.700  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1888 on 4160 degrees of freedom
Multiple R-squared:  0.6521,    Adjusted R-squared:  0.6507
F-statistic: 487.3 on 16 and 4160 DF,  p-value: < 2.2e-16
```

Figure 33: Final Model Fitted to Entire Data Set

```
Analysis of Variance Table

Response: Rings
                             Df  Sum Sq Mean Sq   F value    Pr(>F)
Sex                           2 103.228  51.614 1447.5038 < 2.2e-16 ***
Length                        1  89.301  89.301 2504.4275 < 2.2e-16 ***
Diameter                      1   6.238   6.238  174.9484 < 2.2e-16 ***
Height                        1   5.277   5.277  147.9957 < 2.2e-16 ***
Whole.weight                  1   3.942   3.942  110.5625 < 2.2e-16 ***
Shucked.weight                1  42.344  42.344 1187.5334 < 2.2e-16 ***
Viscera.weight                1   3.666   3.666  102.8147 < 2.2e-16 ***
Shell.weight                  1   1.401   1.401   39.2995 4.005e-10 ***
Sex:Shucked.weight            2   5.827   2.914   81.7133 < 2.2e-16 ***
Length:Diameter               1   8.661   8.661  242.8972 < 2.2e-16 ***
Length:Height                 1   0.060   0.060    1.6848    0.1944
Diameter:Height               1   0.002   0.002    0.0624    0.8028
Diameter:Shucked.weight       1   3.969   3.969  111.3078 < 2.2e-16 ***
Shucked.weight:Shell.weight   1   4.082   4.082  114.4897 < 2.2e-16 ***
Residuals                  4160 148.334   0.036
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 34: Final Model Anova Table



Variance looks constant and errors appear to be slightly heavy-tailed

Figure 35: Final Model Diagnostic Plots

```
  [1]   82   84  110  130  150  158  164  165  166  167
 [11]  169  171  225  237  238  239  240  271  278  292
 [21]  307  335  356  359  373  502  524  526  527  548
 [31]  612  637  647  659  661  695  697  720  721  761
 [41]  883  892  899 1000 1035 1042 1050 1052 1053 1099
 [51] 1146 1169 1175 1185 1194 1200 1202 1203 1205 1207
 [61] 1208 1210 1211 1217 1258 1359 1386 1395 1401 1412
 [71] 1417 1418 1419 1420 1427 1428 1429 1430 1525 1528
 [81] 1529 1638 1692 1700 1738 1749 1751 1755 1757 1760
 [91] 1761 1762 1763 1764 1787 1813 1822 1824 1981 1983
[101] 1985 1987 1988 2007 2052 2085 2090 2091 2109 2115
[111] 2158 2162 2170 2181 2202 2209 2210 2213 2251 2266
[121] 2275 2335 2369 2372 2381 2382 2398 2408 2435 2529
[131] 2535 2538 2540 2543 2545 2624 2625 2626 2628 2642
[141] 2676 2708 2710 2711 2729 2791 2802 2811 2812 2855
[151] 2863 2864 2930 2957 2963 2971 2983 2988 2994 3008
[161] 3009 3035 3051 3082 3083 3087 3129 3141 3142 3149
[171] 3150 3152 3162 3189 3319 3328 3338 3389 3397 3428
[181] 3519 3543 3600 3629 3714 3716 3717 3733 3801 3815
[191] 3828 3838 3861 3900 3903 3904 3929 3962 3963 3993
[201] 3994 3997 4018 4053 4083 4090 4113 4149
```

Figure 36: Leverage Values

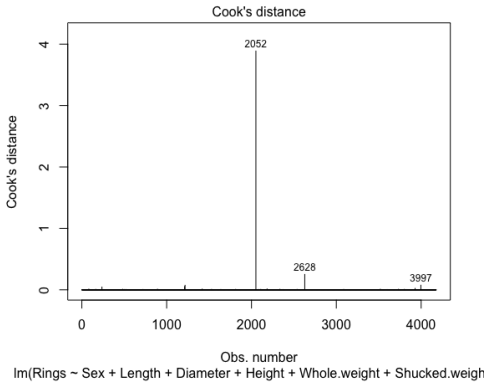


Figure 37: Cooks Distance Plot

B R Code

```
abaloneData=read.table("abalone.txt",sep=",",header=FALSE)
colnames(abaloneData)=c("Sex","Length","Diameter","Height","Whole.weight",
"Shucked.weight","Viscera.weight","Shell.weight", "Rings")

#Confirm no missing values:
sapply(abaloneData,class)
any(is.na(abaloneData)) ##doesn't appear to be any NA in data
any(abaloneData=="?")
any(abaloneData=="")
any(abaloneData==0)
which(abaloneData$Height==0)
which(abaloneData$Rings==0)
##do have a few entries that come up as zero, will confirm which ones
##I will keep these zeros in assuming that the measurement
was too small to calculate
##and this may have to do with the abalone being categorized as infants.
##In the real world, I would ask what the 0 meant from
the data collectors perspective.
##No NAs or other conflicting values, so I will continue with investigation.
##Next,I will look at some summary stats for the variables:
sapply(abaloneData, summary)
##Next, I will look at the histograms for the variables:
##First is the ring variable(the response variable)
hist(abaloneData$Rings, xlab="Number of Rings",
main="Histogram of Number of Abalone Rings")
##The distribution of the response variables is a bit right scewed.
##Test some transformations for this variable to see if we can adjust it:

logrings=log(abaloneData$Rings)
sqrtrings=sqrt(abaloneData$Rings)
fracrings=1/abaloneData$Rings
par(mfrow = c(2, 2))
hist(logrings, xlab="Number of Rings",
main="Histogram of Number of log(Rings)")
##looks normal, sqrt in sec, original in third
hist(sqrtrings, xlab="Number of Rings",
main="Histogram of Number of sqrt(Rings)")
hist(fracrings, xlab="Number of Rings",
main="Histogram of Number of 1/Rings")

##Boxplot comparison:
par(mfrow = c(2, 2))
```

```
boxplot(abaloneData$Rings, main="Boxplot of Rings",
xlab="Rings", horizontal=TRUE)
boxplot(sqrtrings, main="Boxplot of sqrt(Rings)",
xlab="sqrt(Rings)", horizontal= TRUE)
boxplot(logrings, main="Boxplot of sqrt(Rings)",
xlab="sqrt(Rings)", horizontal= TRUE)
```

```
##Histograms for the quantative variables:
par(mfrow = c(3, 3))
index=c(2:8)
for(i in index) {hist(abaloneData[, i],
main=paste("Histogram of", names(abaloneData)[i]),xlab=names(abaloneData[i]))}

##box plot for quantative variables
par(mfrow = c(3, 3))
index=c(2:8)
for(i in index) {boxplot(abaloneData[, i], main=paste("Boxplot of",
names(abaloneData)[i]),xlab=names(abaloneData[i]), horizontal = TRUE)}


##Bar graph and Pie chart for gender:
par(mfrow = c(1, 2))
table(abaloneData$Sex)
barplot(table(abaloneData$Sex),col=rainbow(3),main='Barplot of Sex', ylim=c(0,1700))


n=4177
##pie for sex
lbls=c('Female','Infant','Male')
pct=round(100*table(abaloneData$Sex)/n)
lab=paste(lbls,pct)
lab=paste(lab,'%',sep='')
pie(table(abaloneData$Sex),
labels=lab,col=c('blue','purple','green'),main='Pie Chart of Sex')

##box plot for variables with respect to sex:
par(mfrow = c(3, 3))
index=c(2:9)
for(i in index) {boxplot(abaloneData[, i]~abaloneData$Sex, main=
paste("Boxplot of", names(abaloneData)[i],"Vs Sex"),
```

```
xlab=names(abaloneData)[i], ylab="Sex",col=rainbow(3),horizontal = TRUE)}


##test to confirm transormation
library(MASS)
boxcox(abaloneData$Rings~., data=abaloneData)


##transform rings before moving on
abaloneData$Rings=log(abaloneData$Rings)

##Scatter plot and correlation matrix for quantative variabnles
quandata=data.frame(cbind(abaloneData$Rings,
abaloneData$Length,abaloneData$Diameter,abaloneData$Height,
abaloneData$Whole.weight,
abaloneData$Shucked.weight, abaloneData$Viscera.weight, abaloneData$Shell.weight))
colnames(quandata)=c("Rings","Length",
"Diameter","Height","Whole.weight",
"Shucked.weight","Viscera.weight","Shell.weight")
pairs(quandata)
cor(quandata)


##finding VIF
lenglm=lm(abaloneData$Length~abaloneData$Diameter+
abaloneData$Height+abaloneData$Whole.weight+
abaloneData$Shucked.weight+
abaloneData$Viscera.weight+
abaloneData$Shell.weight, data= abaloneData)

diameterlm=lm(abaloneData$Diameter~abaloneData$Length+
abaloneData$Height+abaloneData$Whole.weight+
abaloneData$Shucked.weight+
abaloneData$Viscera.weight+
abaloneData$Shell.weight, data= abaloneData)

Heightlm=lm(abaloneData$Height~abaloneData$Length+a
baloneData$Diameter+abaloneData$Whole.weight+
abaloneData$Shucked.weight+abaloneData$Viscera.weight+
abaloneData$Shell.weight, data= abaloneData)

wholeweightlm=lm(abaloneData$Whole.weight~abaloneData$Length+
abaloneData$Diameter+abaloneData$Height+
abaloneData$Shucked.weight+
abaloneData$Viscera.weight+abaloneData$Shell.weight, data= abaloneData)
```

```
shuckedlm=lm(abaloneData$Shucked.weight~abaloneData$Length+
abaloneData$Diameter+abaloneData$Height+
abaloneData$Whole.weight+
abaloneData$Viscera.weight+
abaloneData$Shell.weight, data= abaloneData)

Visceralm=lm(abaloneData$Viscera.weight~abaloneData$Length+
abaloneData$Diameter+abaloneData$Height+
abaloneData$Whole.weight+abaloneData$Shucked.weight+
abaloneData$Shell.weight, data= abaloneData)

shellweightlm=lm(abaloneData$Shell.weight~abaloneData$Length+
abaloneData$Diameter+abaloneData$Height+
abaloneData$Whole.weight+abaloneData$Shucked.weight+
abaloneData$Viscera.weight, data= abaloneData)

VIFmatrix=rbind(1/(1-summary(lenglm)$r.squared),
1/(1-summary(diameterlm)$r.squared),
1/(1-summary(Heightlm)$r.squared),
1/(1-summary(wholeweightlm)$r.squared),
1/(1-summary(shuckedlm)$r.squared), 1/(1-summary(Visceralm)$r.squared),
1/(1-summary(shellweightlm)$r.squared))

colnames(VIFmatrix)= "VIF"
rownames(VIFmatrix)= c("Length", "Diameter",
"Height", "Whole.weight", "Shucked.Weight",
"Viscera.weight", "Shell.weight")
VIFmatrix


##boxplot for rings and categorical variable
boxplot(abaloneData$Rings~abaloneData$Sex,
main='Rings and Sex',xlab='Sex',
ylab='log(Rings)',col=rainbow(3))


##Creating the training and validation data sets
set.seed(10)
n.s=nrow(abaloneData)
index.s=sample(1: n.s, size=2089, replace=FALSE)
data.c=abaloneData[index.s,]  ##training set
data.v=abaloneData[-index.s,] ##validation set
```

```
##test to see that the training and
validation data look alike for the quantitative data
par(mfrow=c(3,3))
boxplot(data.c$Rings,data.v$Rings,
main='log(Rings)',names=c('data.c','data.v'))

boxplot(data.c$Length,data.v$Length,
main='Length',names=c('data.c','data.v'))

boxplot(data.c$Diameter,data.v$Diameter,
main='Diameter',names=c('data.c','data.v'))

boxplot(data.c$Height,data.v$Height,
main='Height',names=c('data.c','data.v'))

boxplot(data.c$Whole.weight,data.v$Whole.weight,
main='Whole Weight',names=c('data.c','data.v'))

boxplot(data.c$Shucked.weight,data.v$Shucked.weight,
main='shucked weight',names=c('data.c','data.v'))

boxplot(data.c$Viscera.weight,data.v$Viscera.weight,
main='Viscera Weight',names=c('data.c','data.v'))

boxplot(data.c$Shell.weight,data.v$Shell.weight,
main='Shell Weight',names=c('data.c','data.v'))


##summary of categorical variables from each set
summary(data.c$Sex)  #F=673   I=657  M=759
summary(data.v$Sex)  #F=634  I=685  M=769

#-------------------------------------------------------------------------


#Model with all first order coefficients on training data
basemodelfit=lm(data.c$Rings~., data= data.c) ##model has all firt order variables
summary(basemodelfit)
par(mfrow=c(2,2))
plot(basemodelfit, which=1)
plot(basemodelfit, which=2)
plot(basemodelfit$fitted.values,data.c$Rings,
main="Observed vs. Fitted", xlab="Fitted", ylab="Observed")
abline(-1.214e-14, 1)
boxcox(data.c$Rings~.,data= data.c)
```

```
PRESS_full = sum((basemodelfit$residuals/
(1-influence(basemodelfit)$hat))^2)
PRESS_full

#-------------------------------------------------------------------------------

##best subset approach
library(leaps)
sub_set=regsubsets(data.c$Rings~.,data=data.c,
nbest=1,nvmax=11,method="exhaustive")
sum_sub=summary(sub_set)
n=nrow(data.c)
p.m=as.integer(as.numeric(rownames(sum_sub$which))+1)
sse=sum_sub$rss
aic=n*log(sse/n)+2*p.m
bic=n*log(sse/n)+log(n)*p.m
res_sub=cbind(sum_sub$which,sse,sum_sub$rsq,sum_sub$adjr2,sum_sub$cp,aic, bic)
fit0=lm(data.c$Rings~1,data=data.c)
sse1=sum(fit0$residuals^2)
p=1
c1=sse1/0.001384-(n-2*p)
aic1=n*log(sse1/n)+2*p
bic1=n*log(sse1/n)+log(n)*p
none=c(1,rep(0,9),sse1,0,0,c1,bic1,aic1)
res_sub=rbind(none,res_sub)
colnames(res_sub)=c(colnames(sum_sub$which),"sse",
"R^2", "R^2_a", "Cp","aic", "bic")
round(res_sub, 4)

#-------------------------------------------------------------------------------
##Model Selection: By AIC
#I will mainly consider stepwise
because it is not recommended to do forward selection/backward
elmination with high multicollinearity present
##Forward stepwise
stepAIC(fit0, scope=list(upper=basemodelfit,
lower=~1), direction="both", k=2)
fs1_model1=lm(formula = data.c$Rings ~ Shell.weight + Shucked.weight + Diameter +
Sex + Height + Whole.weight + Viscera.weight + Length, data = data.c)
summary(fs1_model1) ##FULL MODEL WITH FIRST ORDER ONLY

##forward stepwise using interaction terms
##stepAIC(fit0, scope=list(upper=basemodelfit, lower=~1), direction="forward", k=2)
##ends up that forward stepwise choses same first order model
```

```
##testing backwards
##stepAIC(basemodelfit, scope=list(upper=basemodelfit, lower=~1),
direction="backward", k=2)
##also agrees with full first order model.

##backward stepwise
##stepAIC(basemodelfit, scope=list(lower=~1),
direction="both", k=2)
##fs2_model2=lm(formula = data.c$Rings ~ Sex + Length + Diameter + Height +
Whole.weight + Shucked.weight + Viscera.weight + Shell.weight, data = data.c)
##summary(fs2_model2)## SAME MODEL FOR BACKWARD STEPWISE

##stepwise with BIC test
stepAIC(fit0, scope=list(upper=basemodelfit,
lower=~1), direction="both", k=log(n))
fs2_model2=lm(formula = data.c$Rings ~ Shell.weight + Shucked.weight +
Diameter + Sex + Height + Whole.weight +
Viscera.weight, data = data.c)
summary(fs2_model2)
##does not have length

##BIC stepwise backwards
##stepAIC(basemodelfit, scope=list(lower=~1),
direction="both", k=log(n))
##results up same model

##BIC Forward
##stepAIC(fit0, scope=list(upper=basemodelfit,
lower=~1), direction="forward", k=log(n))
##same model

##BIC Backward
##stepAIC(basemodelfit, scope=list(lower=~1),
direction="backward", k=log(n))
##same model

##Diagnostics of the two first order only models:
par(mfrow=c(2,2))
plot(basemodelfit, which=1)
plot(basemodelfit, which=2)
plot(fs2_model2, which=1)
plot(fs2_model2, which=2)
##plots look similar not much difference

##------------------------------------------------------------------------
```

```
##Test the 2nd order interactions


##forward stepwise: by AIC
##full model with all interaions:
fitfullinters=lm(data.c$Rings~.^2,
data=data.c) ##FULL MODEL WITH INTERACTIONS ONLY
length(fitfullinters$coefficients)
## 45 interaction terms
##Forward stepwise
stepAIC(fit0, scope=list(upper=fitfullinters, lower=~1), direction="both", k=2)
fs3_model3inters= lm(formula = data.c$Rings ~ Shell.weight +
Shucked.weight + Diameter + Whole.weight + Sex +
Viscera.weight + Height + Length +
Shucked.weight:Diameter + Shucked.weight:Sex + Shell.weight:Sex
+ Diameter:Height + Diameter:Length + Height:Length +
Diameter:Sex + Shell.weight:Shucked.weight +
Diameter:Whole.weight + Shucked.weight:Height, data = data.c)

summary(fs3_model3inters)
length(fs3_model3inters$coefficients)
stepAIC(fit0, scope=list(upper=fitfullinters, lower=~1), direction="forward", k=2)
##use for reference

 ###From here on out I decided to focus on using BIC
##predictive models are large and BIC will help keep them condense
##Forward stepwise:
stepAIC(fit0, scope=list(upper=fitfullinters,
lower=~1), direction="both", k=log(n))
fs4_model4inters=lm(formula = data.c$Rings ~ Shell.weight + Shucked.weight +
Diameter + Whole.weight + Sex +
Viscera.weight + Shell.weight:Diameter + Shucked.weight:Diameter +
Shucked.weight:Whole.weight + Shucked.weight:Sex, data = data.c)
summary(fs4_model4inters)

##foward: agrees with forward stepwise
#stepAIC(fit0, scope=list(upper=fitfullinters,
lower=~1), direction="forward", k=log(n))

##Backwardstepwise
stepAIC(fitfullinters, scope=list(lower=~1),
direction="both", k=log(n))
fs5_model5inters=lm(formula = data.c$Rings ~ Sex + Length + Diameter +
Height + Whole.weight +
```

```
Shucked.weight + Viscera.weight +
Shell.weight + Sex:Shucked.weight +
Length:Diameter +
Length:Height + Diameter:Height +
Diameter:Shucked.weight + Shucked.weight:Shell.weight, data = data.c)
summary(fs5_model5inters)

##backward: Agrees with Backward stepwise
##stepAIC(fitfullinters, scope=list(lower=~1), direction="backward", k=log(n))

##diagnostics for models with interaction terms:

par(mfrow=c(2,2))
plot(fitfullinters, which=1)
plot(fitfullinters, which=2)
plot(fs3_model3inters, which=1)
plot(fs3_model3inters, which=2)
plot(fs4_model4inters, which=1)
plot(fs4_model4inters, which=2)
plot(fs5_model5inters, which=1)
plot(fs5_model5inters, which=2)
##looks as though non-linearity was resolved, Q-Q right tail heavy

##Test for polynomials

center=function(x) x-mean(x)
Lengthcent=center(data.c$Length)
Diametercent=center(data.c$Diameter)
heightcent=center(data.c$Height)
Whole.weightcent=center(data.c$Whole.weight)
Shucked.weightcent=center(data.c$Shucked.weight)
Viscera.weightcent=center(data.c$Viscera.weight)
Shell.weightcent=center(data.c$Shell.weight)
Lengthcent2=center(data.c$Length^2)
Diametercent2=center(data.c$Diameter^2)
heightcent2=center(data.c$Height^2)
Whole.weightcent2=center(data.c$Whole.weight^2)
Shucked.weightcent2=center(data.c$Shucked.weight^2)
Viscera.weightcent2=center(data.c$Viscera.weight^2)
Shell.weightcent2=center(data.c$Shell.weight^2)
polyfitdata=data.frame(cbind(data.c$Rings,
factor(data.c$Sex), Lengthcent,Diametercent,heightcent,
Whole.weightcent,Shucked.weightcent,
Viscera.weightcent, Shell.weightcent,Lengthcent2,
Diametercent2,heightcent2,Whole.weightcent2,Shucked.weightcent2
```

```
,Viscera.weightcent2,Shell.weightcent2))

par(mfrow=c(2,4))
plot(Lengthcent2,basemodelfit$residuals,
main="Residuals Vs Length", xlab="Length", ylab="Residuals")
plot(Diametercent2,basemodelfit$residuals,
main="Residuals Vs Diameter", xlab="Diameter", ylab="Residuals")
plot(heightcent2,basemodelfit$residuals,
main="Residuals Vs Height", xlab="Height", ylab="Residuals")
plot(Whole.weightcent2,basemodelfit$residuals,
main="Residuals Vs Whole Weight", xlab="Whole Weight", ylab="Residuals")
plot(Shucked.weightcent2, basemodelfit$residuals,
main="Residuals Vs Shucked Weight", xlab="Shucked Weight", ylab="Residuals")
plot(Viscera.weightcent2,basemodelfit$residuals,
main="Residuals Vs Viscera Weight", xlab="Viscera Weight", ylab="Residuals")
plot(Shell.weightcent2,basemodelfit$residuals,
main="Residuals Vs  Shell Weight",
xlab="Shell Weight", ylab="Residuals")

## no sign of a relationship here,
but we still saw a bit in the pairwise scatter

##take a look at full model with all just to be sure
polyfit=lm(polyfitdata$V1~factor(polyfitdata$V2)+Lengthcent+
Diametercent+heightcent+Whole.weightcent+
Shucked.weightcent+ Viscera.weightcent+ Shell.weightcent+Lengthcent2+Diametercent2+
heightcent2+Whole.weightcent2+
Shucked.weightcent2+
Viscera.weightcent2+ Shell.weightcent2, data=polyfitdata)
summary(polyfit)
plot(polyfit,which=1)




stepAIC(polyfit, scope=list( lower=~1), direction="both", k=log(n))
pollyfitmodel6=lm(formula = polyfitdata$V1 ~ factor(polyfitdata$V2) +
Diametercent +
             heightcent + Whole.weightcent + Shucked.weightcent +
             Viscera.weightcent +
        Shell.weightcent + Diametercent2 + heightcent2 + Whole.weightcent2 +
             Shucked.weightcent2 + Viscera.weightcent2, data = polyfitdata)
summary(pollyfitmodel6)
plot(pollyfitmodel6,which=1)
```

```
##selecting 2 models based on best criteria values

aicfs1=length(fs1_model1$fitted.values)*
log(87.696/length(fs1_model1$fitted.values))+
2*length(fs1_model1$coefficients)
aicfs2=length(fs2_model2$fitted.values)*
log(87.997/length(fs2_model2$fitted.values))+
2*length(fs2_model2$coefficients)
aicfullinters=length(fitfullinters$fitted.values)*
log(72.202/length(fitfullinters$fitted.values))+
2*length(fitfullinters$coefficients)
aicfs3=length(fs3_model3inters$fitted.values)*
log(73.060/length(fs3_model3inters$fitted.values))+
2*length(fs3_model3inters$coefficients)
aicfs4=length(fs4_model4inters$fitted.values)*
log(77.092/length(fs4_model4inters$fitted.values))+
2*length(fs4_model4inters$coefficients)
aicfs5=length(fs5_model5inters$fitted.values)*
log(73.995/length(fs5_model5inters$fitted.values))+
2*length(fs5_model5inters$coefficients)
aicfspoly=length(polyfit$fitted.values)*
log(75.252/length(polyfit$fitted.values))+
2*length(polyfit$coefficients)
aicfspolyfit6=length(pollyfitmodel6$fitted.values)*
log(75.501/length(pollyfitmodel6$fitted.values))+
2*length(pollyfitmodel6$coefficients)

bicfs1=length(fs1_model1$fitted.values)*
log(87.696/length(fs1_model1$fitted.values))+
log(length(fs1_model1$fitted.values))*length(fs1_model1$coefficients)
bicfs2=length(fs2_model2$fitted.values)*
log(87.997/length(fs2_model2$fitted.values))+
log(length(fs2_model2$fitted.values))*length(fs2_model2$coefficients)
bicfullinters=length(fitfullinters$fitted.values)*
log(72.202/length(fitfullinters$fitted.values))+
log(length(fitfullinters$fitted.values))*length(fitfullinters$coefficients)
bicfs3=length(fs3_model3inters$fitted.values)*
log(73.060/length(fs3_model3inters$fitted.values))+
log(length(fs3_model3inters$fitted.values))*length(fs3_model3inters$coefficients)
bicfs4=length(fs4_model4inters$fitted.values)*
log(77.092/length(fs4_model4inters$fitted.values))+
log(length(fs4_model4inters$fitted.values))*length(fs4_model4inters$coefficients)
bicfs5=length(fs5_model5inters$fitted.values)*
log(73.995/length(fs5_model5inters$fitted.values))
```

```
+
log(length(fs5_model5inters$fitted.values))*length(fs5_model5inters$coefficients)
bicfspoly=length(polyfit$fitted.values)*
log(75.252/length(polyfit$fitted.values))+
log(length(polyfit$fitted.values))*length(polyfit$coefficients)
bicfspolyfit6=length(pollyfitmodel6$fitted.values)*
log(75.501/length(pollyfitmodel6$fitted.values))+
log(length(pollyfitmodel6$fitted.values))*length(pollyfitmodel6$coefficients)

cpfs1=87.696/0.03530641-(length(fs1_model1$fitted.values)-
2*length(fs1_model1$coefficients))
cpfs2=87.997/0.03530641-(length(fs2_model2$fitted.values)-
2*length(fs2_model2$coefficients))
cpfsfullinters=72.202/0.03530641-(length(fitfullinters$fitted.values)-
2*length(fitfullinters$coefficients))
cpfs3=73.060/0.03530641-
(length(fs3_model3inters$fitted.values)-
2*length(fs3_model3inters$coefficients))
cpfs4=77.092/0.03530641-(length(fs4_model4inters$fitted.values)-
2*length(fs4_model4inters$coefficients))
cpfs5=73.995/0.03530641-(length(fs5_model5inters$fitted.values)-
2*length(fs5_model5inters$coefficients))
cpfspolyfull=75.252/0.03530641-(length(polyfit$fitted.values)-
2*length(polyfit$coefficients))
cpfs6=75.501/0.03530641-(length(pollyfitmodel6$fitted.values)-
2*length(pollyfitmodel6$coefficients))

PRESS_fullfs1 = sum((fs1_model1$residuals
/(1-influence(fs1_model1)$hat))^2)
PRESS_fullfs2 = sum((fs2_model2$residuals
/(1-influence(fs2_model2)$hat))^2)
PRESS_fullfsfullinters = sum((fitfullinters$residuals
/(1-influence(fitfullinters)$hat))^2)
PRESS_fullfs3 = sum((fs3_model3inters$residuals
/(1-influence(fs3_model3inters)$hat))^2)
PRESS_fullfs4 = sum((fs4_model4inters$residuals
/(1-influence(fs4_model4inters)$hat))^2)
PRESS_fullfs5 = sum((fs5_model5inters$residuals
/(1-influence(fs5_model5inters)$hat))^2)
PRESS_fullfsfullpoly = sum((polyfit$residuals
/(1-influence(polyfit)$hat))^2)
PRESS_fullfs6 = sum((pollyfitmodel6$residuals/(1-influence(pollyfitmodel6)$hat))^2)
```

```
Criteria.values.matrix= data.frame(cbind(
c(87.696,87.997,72.202,73.060,77.092,73.995,75.252,75.501),
c(0.6054,0.604,0.6751,0.6712,0.6531,0.667,0.6614,0.6602),
c(0.6036,0.6025, 0.6681,0.6677,0.6511, 0.6644, 0.6587,0.6581),
c(aicfs1,aicfs2,aicfullinters,aicfs3,aicfs4,aicfs5, aicfspoly,aicfspolyfit6),
c(bicfs1,aicfs2,bicfullinters,bicfs3,bicfs4,bicfs5, bicfspoly,bicfspolyfit6),
c(cpfs1,cpfs2,cpfsfullinters, cpfs3,cpfs4, cpfs5,cpfspolyfull,cpfs6),
c(PRESS_fullfs1,PRESS_fullfs2,
PRESS_fullfsfullinters,PRESS_fullfs3, PRESS_fullfs4,
PRESS_fullfs5,PRESS_fullfsfullpoly,PRESS_fullfs6)))
colnames(Criteria.values.matrix)=c("SSE", "R^2",
"R_a^2", "aic", "bic","C_p", "Press_p**")
row.names(Criteria.values.matrix)=c("Model 1", "Model 2",
"Full Model(Interactions)","Model 3", "Model 4", "Model 5",
"Full Model(Quadratics)","Model 6")



##Looking at the table Model 3 has the lowest Press_p,
Lowest C_P(Bias and variance), lowest aic and relatively low bic
##in addition, it has the best R^2 and R_a^2 compared to the other models sum
##model 3 generated by AIC criteria

##Model 5 looks to be the second best model:
second lowest aic, lowest bic, relatively low Cp and second lowest Press_p
##generated by BIC criteria


##Validation for Model 3 and model 5
##we will see which model has the best MSPE_v
##for model 3 Cp is approximately p(little bias), and the press_p is
not much larger than SSE(no overfitting)\
##Model 5's Cp more than p indicating bias, press_p is also close to SSE
##now we will fit these two models to the validation data
fs3_model3inters
fs5_model5inters
modelevalfs3=lm(data.v$Rings ~ Shell.weight + Shucked.weight +
Diameter
+ Whole.weight + Sex + Viscera.weight +
Height
+ Length + Shucked.weight:Diameter +
Shucked.weight:Sex
+ Shell.weight:Sex + Diameter:Height +
Diameter:Length + Height:Length + Diameter:Sex
```

```
+ Shell.weight:Shucked.weight
+ Diameter:Whole.weight + Shucked.weight:Height,data=data.v)

modelevalfs5=lm(data.v$Rings ~ Sex + Length +
Diameter + Height + Whole.weight
+ Shucked.weight + Viscera.weight +
Shell.weight
+ Sex:Shucked.weight + Length:Diameter +
Length:Height + Diameter:Height
+ Diameter:Shucked.weight + Shucked.weight:Shell.weight,data=data.v)
summary(fs3_model3inters)
summary(modelevalfs3)
##multiple sign changes
##changes in estimation
###For Fist model--Estimation and standard error percent changes:
round(abs(coef(fs3_model3inters)-coef(modelevalfs3))
/abs(coef(fs3_model3inters))*100,3)
sd.fs3= summary(fs3_model3inters)$coefficients[,"Std. Error"]
sd.fs3.v= summary(modelevalfs3)$coefficients[,"Std. Error"]
round(abs(sd.fs3-sd.fs3.v)/sd.fs3*100,3)
##mspe
newdata1=data.v[,-9]
pred.fs3 = predict.lm(fs3_model3inters, newdata1)
mspe.fs3=mean((pred.fs3-data.v[,9])^2)
mspe.fs3
##MSPEv = 0.03678189
PRESS_fullfs3/2089 ##0.037147
73.060/2089 ##0.03497367
##no severe over-fitting


#MODEL 5
summary(fs5_model5inters)
summary(modelevalfs5)
##one sign change, some changes in estimation and standard errors
###For second model--Estimation and standard error percent changes:
round(abs(coef(fs5_model5inters)-coef(modelevalfs5))
/abs(coef(fs5_model5inters))*100,3)
sd.fs5= summary(fs5_model5inters)$coefficients[,"Std. Error"]
sd.fs5.v= summary(modelevalfs5)$coefficients[,"Std. Error"]
round(abs(sd.fs5-sd.fs5.v)/sd.fs5*100,3)
##mspe
newdata2=data.v[,-9]
pred.fs5 = predict.lm(fs5_model5inters, newdata2)
mspe.fs5=mean((pred.fs5-data.v[,9])^2)
```

```
mspe.fs5
##MSPEv = 0.03650119
PRESS_fullfs5/2089 ##0.03732952
73.995/2089 ##0.03542125
##No severe over fitting
##Model 5 has a smaller mspev than model 3.
This model has better predictability than model 3.
##In addition, it has less coefficients, so it's simpler.
##Futhermore, the changes in its estimation parameters and standard errors is
##less in magnitude. It only has one change in sign. Whereas, model 3 has multiple
##changes in sign from the training data fit to the validation data fit.
##I believe model 5 will be the best model to use for the abalone problem.

##fit model to entire data set.
fit.fs1.final=lm(Rings ~ Sex + Length + Diameter +
Height + Whole.weight + Shucked.weight +
Viscera.weight + Shell.weight + Sex:Shucked.weight +
Length:Diameter + Length:Height + Diameter:Height +
Diameter:Shucked.weight + Shucked.weight:Shell.weight, data=abaloneData)
summary(fit.fs1.final)

anova(fit.fs1.final)


##Model Diagnostics
par(mfrow=c(2,1))
plot(fit.fs1.final, which=1)
plot(fit.fs1.final, which=2)

##Check outliers in Y
res=residuals(fit.fs1.final)
n = nrow(abaloneData)
p = 23
h1 = influence(fit.fs1.final)$hat
d.res.std=studres(fit.fs1.final)
max(abs(d.res.std))
sort(abs(d.res.std),decreasing=T)
qt(1-0.1/(2*n),n-p-1)
idx.Y = as.vector(which(abs(
d.res.std)>=qt(1-0.1/(2*n),n-p-1)))
idx.Y
##outliers in Y 237,1217,2052,2184,2628

##leverages and outlying X
idx.X = as.vector(which(h1>(2*23/n)))
```

```
idx.X
plot(h1,res,xlab="leverage",ylab="residuals")
##many leverage points in x


##influenceplot
plot(fit.fs1.final, which=4)
cook.d = res^2*h1/(p*1.293*(1-h1)^2)
cook.max = cook.d[which(cook.d==max(cook.d))]
pf(cook.max,p,n-p)
idx = c(idx.X,idx.Y)
cook.d[idx]
pf(cook.d[idx],p,n-p)

##2052 has biggest cooks distance


##end of code
```

References

[1] Warwick J Nash, Tracy L Sellers, Simon R Talbot, Andrew J Cawthorn and Wes B Ford (1994)"The Population Biology of Abalone (Haliotis species) in Tasmania. I. Blacklip Abalone (H.rubra) from the North Coast and Islands of Bass Strait", Sea Fisheries Division, TechnicalReport No. 48 .

Data Source

UCI machine learning Repository.