Predicting the longevity of an Abalone species using Linear Regression Model

Author:

Mahdi Keshavarz

June 2022

Summary

In this project, a linear regression model is used to identify the factors contributing the longevity of an Abalone. Number of rings indicates the longevity and more rings is a sign of longer age.

The dataset includes 9 variables that most of them are numerical fields so it is suitable for Linear Regression Model. First, there is an overview on the data and we have used various charts to represent an insight into the data structure. Then, we have used the pre-processing methods to make the data clean and ready to create the model.

In the end, different methods have been used to create the best model. After that, significant factors have been introduced.

Table of Contents

Introduction

In the last decade, Machine Learning has emerged with greater applications in many different branches. Now, experts use this data science subject to obtain useful models by teaching machine (computer) with data. The results could help them to better predict, and also interprete variances in data. Machine Learning is becoming a necessary skill for every field that need to analyse data and also build data-driven models for prediction or interpreting.

Predicting longevity of animals has always been an interesting subject for biologists and other scientist. They want to know what factors attributes longer life in different specious in hope to apply them to prolong human lifes. Whatever the reason is, researchers will not be able to identify the key factors unless they use data.

In this projects, we have used a dataset with different columns that are also known as variables. one variable is the response variable and it means we want to identify the impact of other variables on that. The response variable here is number of rings that specify how much the Abalone has lived.

Most of the variables are numeric, so Linear Regression will be a suitable approach to serve our purpose. First, we clean the data which is known as data-preprocess or data preparation. After that, we create the model to predict the longevity of this type of Abalone.

The approach of the project

In this project we aim to use a linear regression model to predict the longevity of a particular abalone's life. So, we need to clean the data neatly, and create the model with extra cautious. The reason is simple. Any wromg attempt will cause inaccuracy in the model. The model could be very beneficial to biology students and experts. The model will help them estimate the duration of lifecycle of any sample using the model. In simple words, this model will provide us with two important insights:

1- predicting the life of abalone
2- identifying the main features which determines the longevity, and understand how much each factor will change the longevity.

Without any delay, we go strictly right to the main points. First, we will use simple statistical measures and graphical figures in order to perceive a general overview of the dataset. Then, we apply the data preparation process. After that, the data will be ready to create, and also test the model.

Overall view of the data

There are 9 variables (columns) in the dataset:

- Sex: M for male, F for female, and l for ……..
- Length: length of the sample
- Diameter: diameter of the sample
- Height: Height of the sample
- Whole weight: Whole weight of the sample
- Shucked weight: Shucked weight of the sample
- Viscera weight: Viscera weight of the sample
- Shell weight: Shell weight of the sample
- Rings: this means the longevity of the sample (more rings means more longevity). We want to predict this variable as the response variable. So, it will be the dependent variable in the model.

There are 4177 records (rows) in the dataset that each one represents a unique sample. There are enough records to both create, and test the model. Although the more data means more accuracy, but the data should suffice.

We will use R studio to make the programming easier. First, we have to enter the dataset into the R studio. The dataset is in CSV format which is suitable for R studio. All the headings are in the proper format so we don't have to be worried about misspelled and inaccurate words.

We use the "summary" function to perceive usefull information about the dataself. The code is as follow:

```
abalone=read.csv("Abalone.csv")

summary(abalone)
```

the result is as follow:

```
    Sex             Length         Diameter         Height        Whole.weight
Length:4177    Min.   :0.075   Min.   :0.0550   Min.   :0.0000   Min.   :0.0020
Class :character   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415
Mode  :character   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
 Shucked.weight   Viscera.weight    Shell.weight        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
```

Figure 1- dataset initial summary 1

It is better to convert the "Sex" into factor so that we will get better, and more detailed information:

abalone$Sex=as.factor(abalone$Sex)

summary(abalone)

```
 Sex          Length         Diameter         Height        Whole.weight
 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
 Shucked.weight   Viscera.weight    Shell.weight        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
> |
```

Figure 2- dataset initial summary 2

First, we examine the first column that is Sex. There are three unique values in this variable:

- M
- F
- L

All three values are large enough, so there is no need to merge them. As the figure below shows, M is the dominant sex of the samples with 1528 records. I, and F are the second, and the third with 1342, and 1307 records respectively.

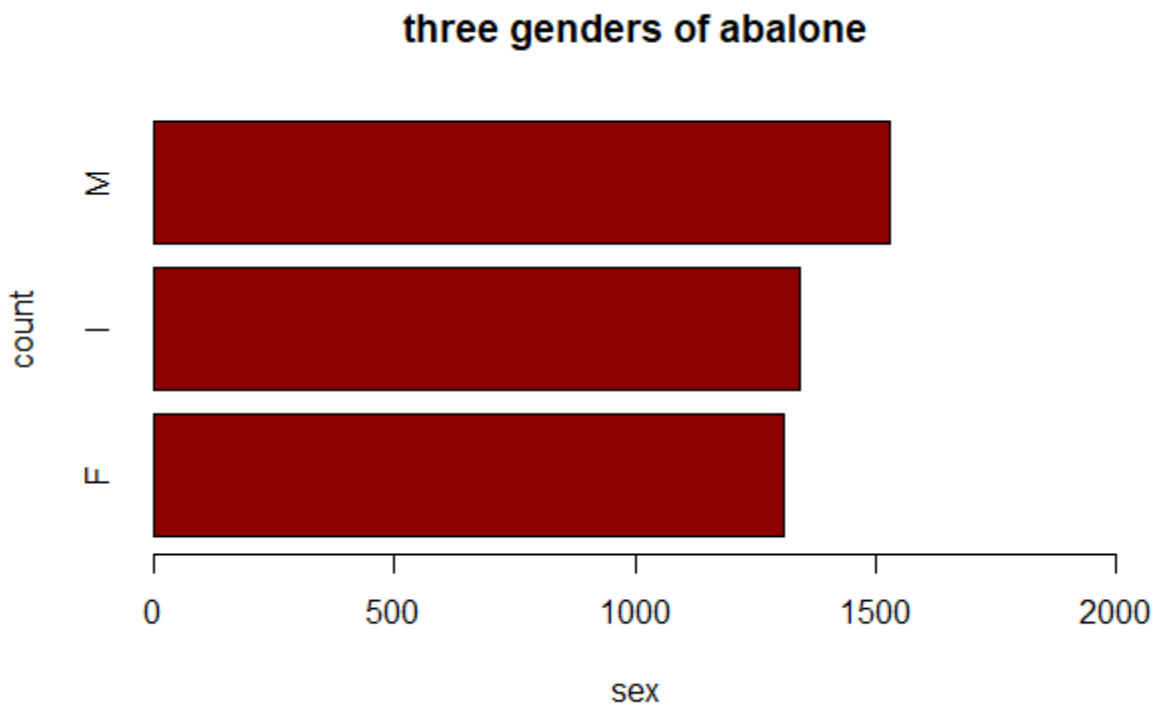

Figure 3- Abalone gender groups

The next variable is the “Length”, and as the figure below indicates, it seem the Rings increase with the length of the abalone at first. After that, there is no specific relationship, and we have to let the model decide.

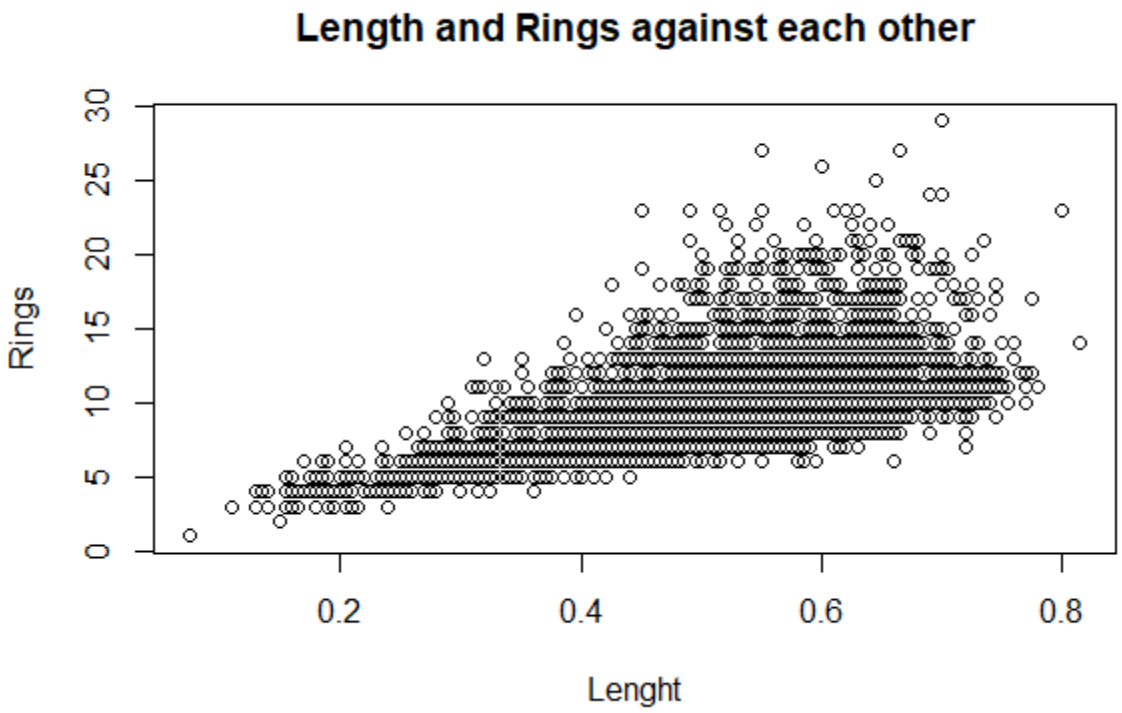

Figure 4- Length against Rings

The next variable is the “Diameter”. First, less diameter results in less rings, but after increasing the diameter the rings will be risen too. Similar to length, from some unclear point, more diameter does not necessarily results in more rings. So, we have to wait until the model determines the role of this variable.

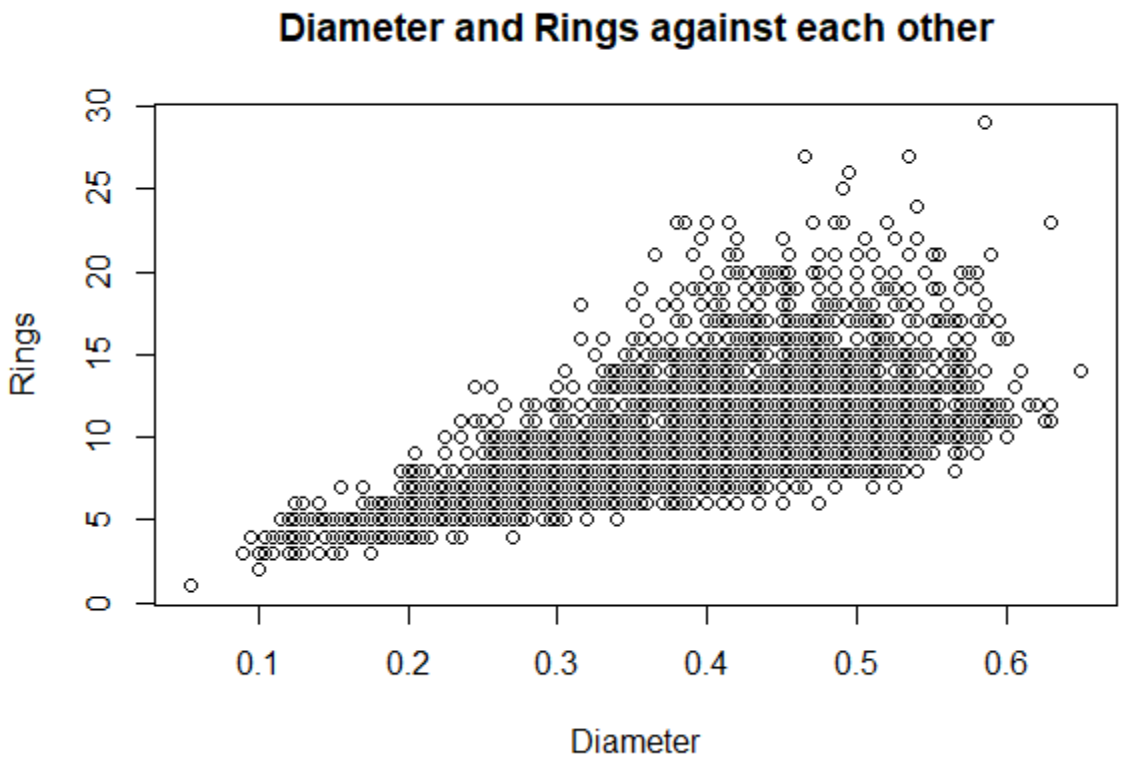

Figure 5- Diameter against Rings

Now, it is the “Height” that should be examined carefully. There are two outliers obviously and we will take care of them later in the ore-process stage. It seem the rings are less sensitive to height rather than two previous variables (Length and diamater).

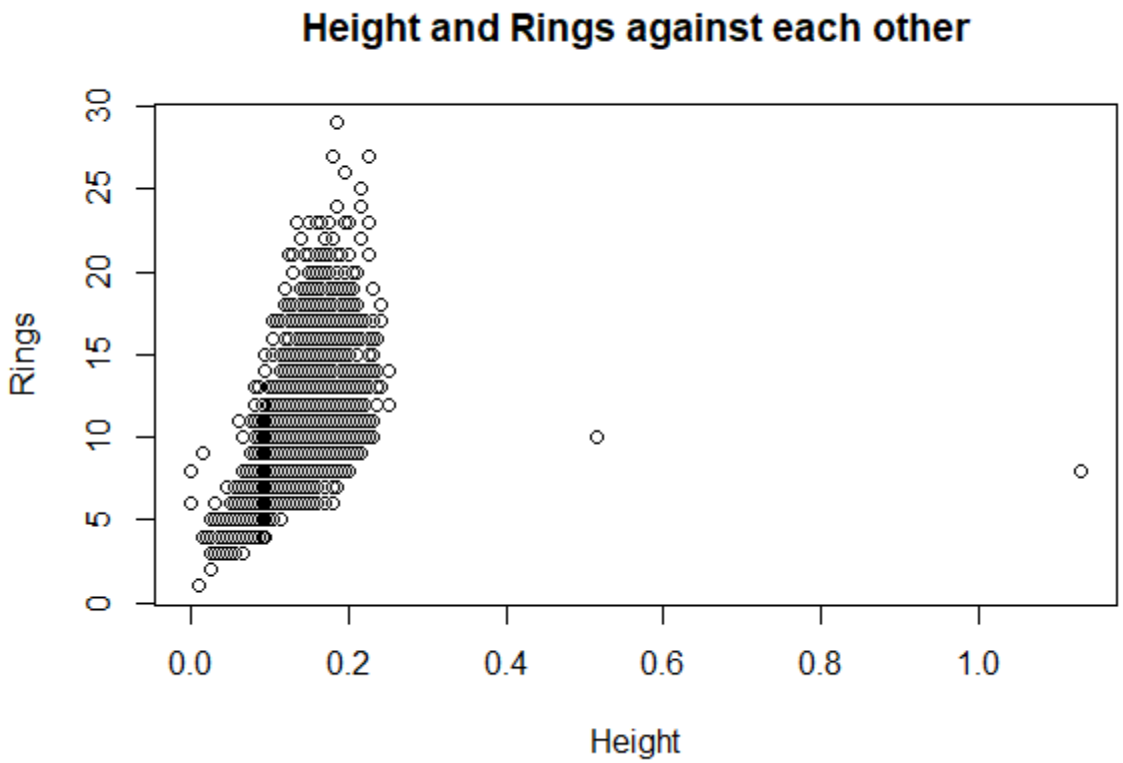

Figure 6- Height against Rings

The next variable is the “whole weight” of the sample. There is a logarithmic relationship between this variable, and the response variable (Rings).

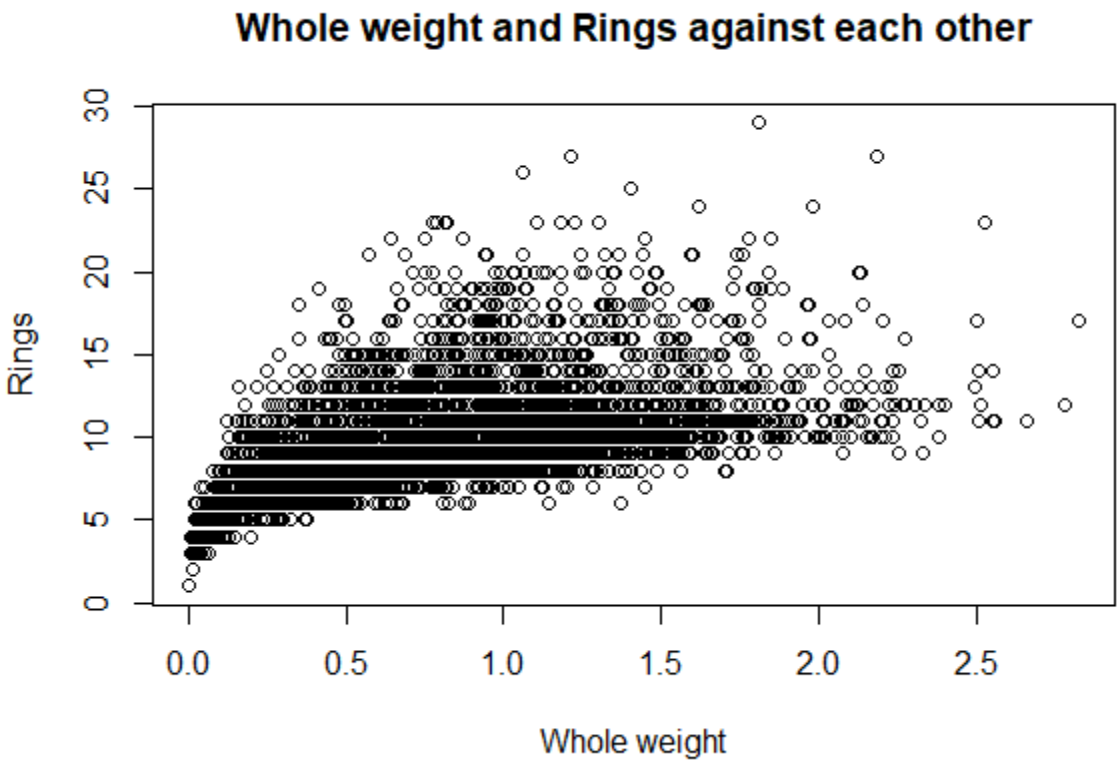

Figure 7- Whole Weight against Rings

Shucked weight is a little similar to the Ehole weight in terms of relationship to the response variable.

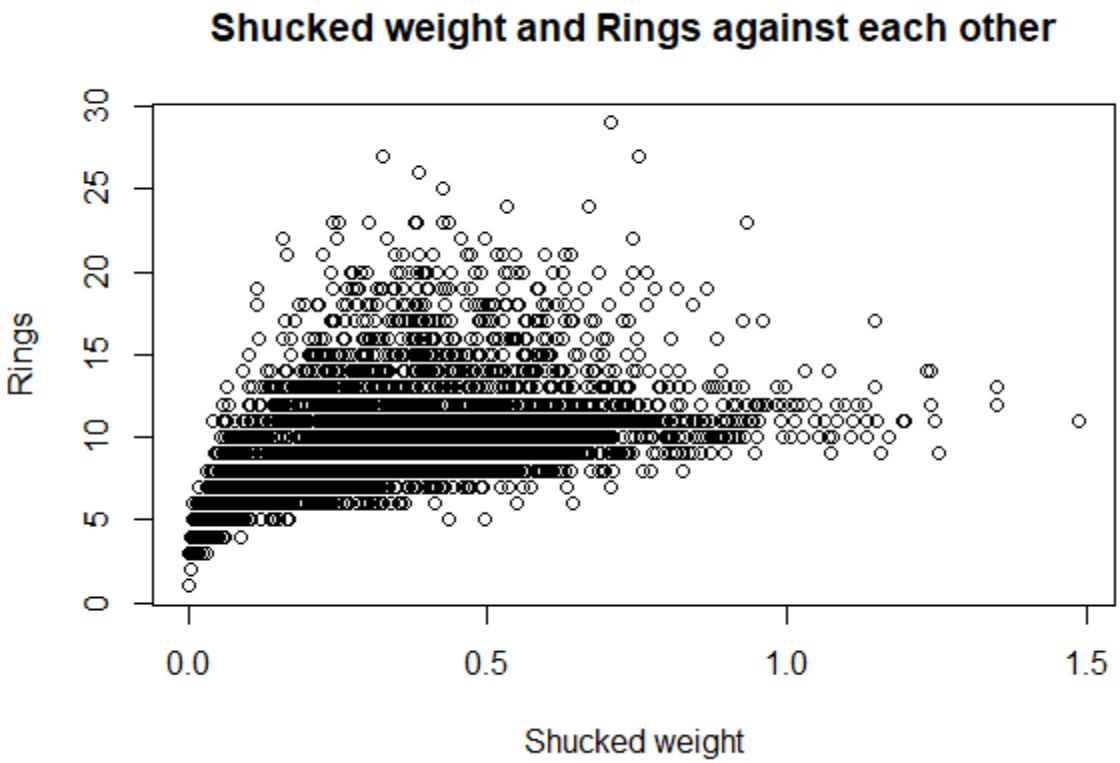

Figure 8- Shuckes Weight against Rings

There is similar situation for the Viscera wight variable.

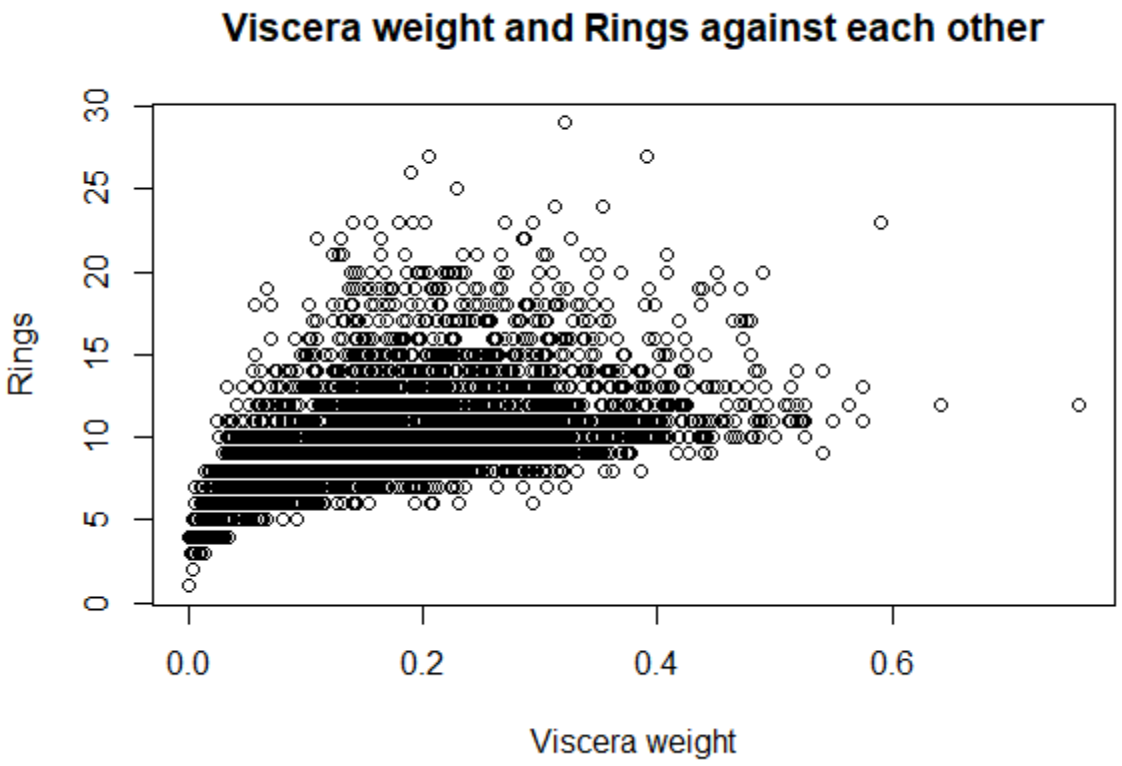

Figure 9- Viscera weight against Rings

At last, the shell weight scatterplot agains the response variable indicates similarities to the other variables that depict some weight.

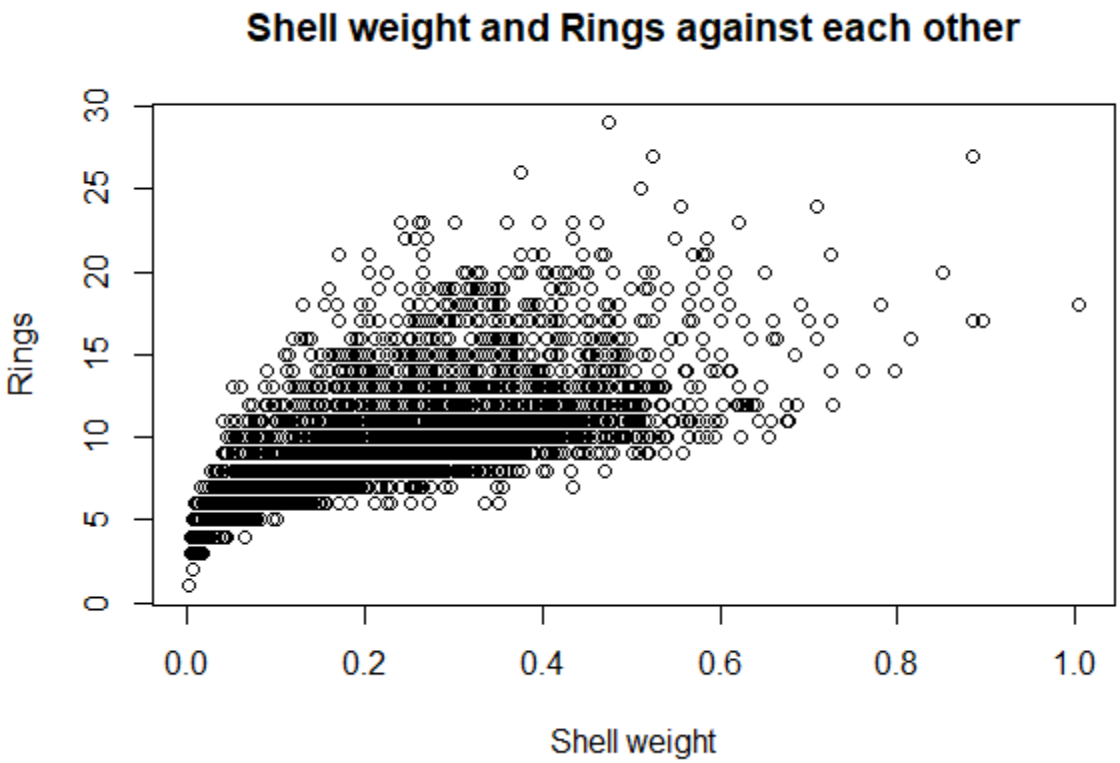

Figure 10- Shell against Rings

Pre-Processing

Pre-processing is an important part of every data-driven projects. Our machine learning will result inappropriate results without a reasonable pre-processing. In some cases like missing data the model will not be created at all, so we have to take this stage very serious.

There are several task that evey machine learning project involves them:

- Filter data: deleting unnecessary fields or records
- Vggregate values: aggregating data horizontally or vertically if necessary
- Missing value treatment: how to deal with empty values
- Outlier treatment: how to deal with values that do not obey the general behavior of the data
- Variable transformation: transforming variables (columns) into a new one that will have better uses.
- Variable reduction: it is also called dimension reduction, and it is done when there are some fields with no use (like phone number in many cases)

Now, we should carefully examine the report of the "summary" function in R Studio.

```
 Sex        Length          Diameter         Height        Whole.weight
 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
 Shucked.weight   Viscera.weight    Shell.weight        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
>
```

Figure 11- database secondary summary 1

- Sex seems to fine with ni problem.
- Min length is 0.075 and a lot less than the first quantile which is 0.45. there seems to be a outlier because this difference is considerable and causes skeness. Less skeness means data behaves like "Normal Distribution" that is the best form for analysin and creating models.
- Diamter seems to have outlier for the same reason

- Height is similar to the two previous fields, and it also seems to have outliers from the opposite side too.
- Whole weight seems to have similar outlier
- Shucked weight also have a lot difference between the first quantile and the min value.
- Viscera weight also has outlier
- Shell weight has outlier
- Ring seems to be fine and because it is the response value we will not change it yet

There are no missing values, and no variable seems to be useless. So, we only have to deal with outliers. After that, we check whether there is correlation between variables.

Outlier treatment approach is called 3 sigma or capping and flooring which is very popular in heavy industries like car manufacturing factories. We will replace every value less than 0.3*p1[1] with itself, and every value larger than 3*p99[2] with itself.

*length_LL=0.3*quantile(abalone$Length, 0.01)*

*length_UL=3*quantile(abalone$Length, 0.99)*

length_LL

length_UL

abalone$Length[abalone$Length<length_LL]=length_LL

abalone$Length[abalone$Length>length_UL]=length_UL

*diameter_LL=0.3*quantile(abalone$Diameter, 0.01)*

*diameter_UL=3*quantile(abalone$Diameter, 0.99)*

diameter_LL

diameter_UL

abalone$Diameter[abalone$Diameter<diameter_LL]=diameter_LL

[1] The first quantile
[2] The 99th quantile

```
abalone$Diameter[abalone$Diameter>diameter_UL]=diameter_UL

height_LL=0.3*quantile(abalone$Height, 0.01)
height_UL=3*quantile(abalone$Height, 0.99)
height_LL
height_UL
abalone$Height[abalone$Height<height_LL]=height_LL
abalone$Height[abalone$Height>height_UL]=height_UL

whole_weight_LL=0.3*quantile(abalone$Whole.weight, 0.01)
whole_weight_UL=3*quantile(abalone$Whole.weight, 0.99)
whole_weight_LL
whole_weight_UL
abalone$Whole.weight[abalone$Whole.weight<whole_weight_LL]=whole_weight_LL
abalone$Whole.weight[abalone$Whole.weight>whole_weight_UL]=whole_weight_UL

shucked_weight_LL=0.3*quantile(abalone$Shucked.weight, 0.01)
shucked_weight_UL=3*quantile(abalone$Shucked.weight, 0.99)
shucked_weight_LL
shucked_weight_UL
abalone$Shell.weight[abalone$Shucked.weight<shucked_weight_LL]=shucked_weight_LL
abalone$Shell.weight[abalone$Shell.weight>shucked_weight_UL]=shucked_weight_UL

viscera_LL=0.3*quantile(abalone$Viscera.weight, 0.01)
viscera_UL=3*quantile(abalone$Viscera.weight, 0.99)
viscera_LL
viscera_UL
```

abalone$Viscera.weight[abalone$Viscera.weight<viscera_LL]=viscera_LL

abalone$Viscera.weight[abalone$Viscera.weight>viscera_UL]=viscera_UL

*shell_weight_LL=0.3*quantile(abalone$Shell.weight, 0.01)*

*shell_weight_UL=3*quantile(abalone$Shell.weight, 0.99)*

shell_weight_LL

shell_weight_UL

abalone$Shell.weight[abalone$Shell.weight<shell_weight_LL]=shell_weight_LL

abalone$Shell.weight[abalone$Shell.weight>shell_weight_UL]=shell_weight_UL

the sex variable is categorical which is not acceptable in linear regression. So, we use dummy variables to solve this problem. There are three unique values in this variable (M, F, L), and we will make 3-1=2 new variables. if a record is F, the F variable is 1, otherwise, it is zero. This approach applies to M also. For L, both M, and F variables will be zoro. As discussed earlier, there are enough number of each sex unique values, so there is no need to merge some records.

The code is as follows:

abalone=dummy.data.frame(abalone) #dummy variable

abalone=abalone[,-2] #useless variable reduction

Now, we only have to use the “cor” function to check correlation between any pairs of the variables. if there is a significant correlation, we will delete one of the variables. the variable which is less correlated to the response variable will be the one deleted.

Table 1- Correlation between variables 1

	SexF	SexM	Length	Diameter	Height	Whole.weight	Shucked.weight	Viscera.weight	Shell.weight	Rings
SexF	1	- 0.51	0.30	0.31	0.30	0.29	0.26	0.30	0.30	0.25
SexM	-0.51	1	0.23	0.24	0.22	0.25	0.25	0.24	0.23	0.18
Length	0.30	0.23	1	0.98	0.87	0.92	0.89	0.90	0.89	0.55
Diameter	0.31	0.24	0.98	1	0.87	0.92	0.89	0.89	0.90	0.57
Height	0.30	0.22	0.87	0.87	1	0.86	0.81	0.84	0.86	0.58
Whole.weight	0.29	0.25	0.92	0.92	0.86	1	0.96	0.96	0.95	0.54
Shucked.weight	0.26	0.25	0.89	0.89	0.81	0.96	1	0.93	0.88	0.42
Viscera.weight	0.30	0.24	0.89	0.84	0.96	0.96	0.93	1	0.90	0.50
Shell.weight	0.30	0.23	0.89	0.90	0.86	0.95	0.88	0.90	1	0.62
Rings	0.25	0.18	0.55	0.57	0.58	0.54	0.42	0.50	0.62	1

There are many variables that are correlated significantly. So, we have to delete some of them because of two reason:

1- if two variables are highly correlated, then they depict similar information. So, only one of them will be sufficient.

2- correlation causes inaccuracy in linear regression models.

Diameter and length are highly correlated, so we have to delete one of them. Length is less corelated to the rings (0.55) rather than diameter (0.57), so we will delete it.

Shucked weight, viscera weight, shell weight and whole weight are in similar situation. They are 95 or 95 percent corelated which are very close numbers. So, it is suitable to delete Shucked weight that is less corelated to the response variable.

- Length
- Shucked weight

Table 2- Correlation between variables 2

	Sex F	Sex M	Diameter	Height	Whole.weight	Shucked.weight	Viscera.weight	Shell.weight	Rings
SexF	1	-0.51	0.31	0.30	0.29	0.26	0.30	0.30	0.25
SexM	-0.51	1	0.24	0.22	0.25	0.25	0.24	0.23	0.18
Diameter	0.31	0.24	1	0.87	0.92	0.89	0.89	0.90	0.57
Height	0.30	0.22	0.87	1	0.86	0.81	0.84	0.86	0.58
Whole.weight	0.29	0.25	0.92	0.86	1	0.96	0.96	0.95	0.54
Viscera.weight	0.30	0.24	0.84	0.96	0.96	0.93	1	0.90	0.50
Shell.weight	0.30	0.23	0.90	0.86	0.95	0.88	0.90	1	0.62
Rings	0.25	0.18	0.57	0.58	0.54	0.42	0.50	0.62	1

Viscera weight and Whole weight, Whole weight and Height, and Shell weight and Whole weight are corelated with close numbers. Viscera weight is less corelated to the response variable, so we will delete it.

- Viscera weight

Table 3- Correlation between variables 3

	SexF	SexM	Diameter	Height	Whole.weight	Shucked.weight	Shell.weight	Rings
SexF	1	- 0.51	0.31	0.30	0.29	0.26	0.30	0.25
SexM	- 0.51	1	0.24	0.22	0.25	0.25	0.23	0.18
Diameter	0.31	0.24	1	0.87	0.92	0.89	0.90	0.57
Height	0.30	0.22	0.87	1	0.86	0.81	0.86	0.58
Whole.weight	0.29	0.25	0.92	0.86	1	0.96	0.95	0.54
Shell.weight	0.30	0.23	0.90	0.86	0.95	0.88	1	0.62
Rings	0.25	0.18	0.57	0.58	0.54	0.42	0.62	1

Between whole weight and shell weight, the first one is less corelated to the response variable, so it will be deleted too.

- whole weight

Table 4- Correlation between variables 4

	SexF	SexM	Diameter	Height	Shucked.weight	Shell.weight	Rings
SexF	1	- 0.51	0.31	0.30	0.26	0.30	0.25
SexM	-0.51	1	0.24	0.22	0.25	0.23	0.18
Diameter	0.31	0.24	1	0.87	0.89	0.90	0.57
Height	0.30	0.22	0.87	1	0.81	0.86	0.58
Shell.weight	0.30	0.23	0.90	0.86	0.88	1	0.62
Rings	0.25	0.18	0.57	0.58	0.42	0.62	1

We create the model with, and without correlated variables, so we weill have a chance to compare both results.

Train-Test Split

Now we have the dataset ready to use, but there is one important step before creating the linear regression model. We need to devide data into two sets:

- Train set: it is used to train the model. In fact, this set is the source that makes Machine to learn
- Test set: this set is used to test the accuracy of the model after in is trained by trainig set.

There are some approaches for train-test split, but we have chosen the 80-20 way. In this method, 80 percent of the records will be the trainig set, and the remaining 20 percent are the test set. This is a simple but efficient way that is being used in many data-driven projects.

```
set.seed(0)

x=sample.split(abalone, SplitRatio=0.8)

training_set= subset(abalone, x==TRUE)

test_set= subset(abalone, x==FALSE)
```

Linear Regression Model

Now, it is time to create our model. We use the “lm” function wchich is used to create linear models:

linear_model=lm(Price.USD.~ . ,data=training_set)

summary(linear_model)

The result is as follow:

Table 5- LRM[3]1 Residuals

Residuals:				
Min	1Q	Median	3Q	Max
- 8.6220	- 1.3212	- 0.3379	0.8635	14.0345

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)     2.9121     0.2678  10.875  < 2e-16 ***
SexF            0.8062     0.1021   7.894 3.72e-15 ***
SexM            0.8641     0.0955   9.048  < 2e-16 ***
Length         -0.7093     1.8042  -0.393    0.694
Diameter       10.1924     2.2271   4.577 4.86e-06 ***
Height         16.6478     1.9151   8.693  < 2e-16 ***
Whole.weight    8.9280     0.7232  12.345  < 2e-16 ***
Shucked.weight -19.6556    0.8152 -24.112  < 2e-16 ***
Viscera.weight -10.8459    1.2907  -8.403  < 2e-16 ***
Shell.weight    8.2482     1.1254   7.329 2.76e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 12- LRM1 summary

Table 6- F-Test results of the LRM1

Residual standard error:	2.187 on 4167 degrees of freedom
Multiple R-squared:	0.5408
Adjusted R-squared:	0.5408
F-statistic:	93.88 on 9 and 4167 DF
p-value:	< 2.2e-16

[3] Linear Regression Model

The results show all the variables are significant with 99 percent confidence. Length is the only variable that seems to have no significant impact on the number of the rings (and therefore on the longevity).

Diameter and height both have positive coefficients. If these two variables increase, so will the response variable. Whole weight and shell weight also are positively significant. Shcked weight and Viscera weight are two significant variables with large and negative coefficints. If we increase these two variables, the response will decrease.

It seems being male means more longevity because the coefficient is 0.86, but for SexF it is 0.80. Intercept is also significant, and it means we are losing some important variables. R-squared is 0.534 and it means the model defines 53 percent of variation in the response variable. Adjusted R-squared is 0.54 and it may seems not enough, but considering some missing variables in gathering the dataset and also scientific concept of the subject, it is good enough.

Now, we should examine the F-test result. This is done due to ensure us that the results are accurate and there is no proof no decline them. P-value is small and it means there is large confidence interval for the test and the results are accurate.

We should use the test set to chack the accuracy of the model:

```
z1=predict(linear_model1, training_set)

z2=predict(linear_model1, test_set)

MSE1=mean((training_set$Rings-z1)^2)  #training_set MSE

MSE2=mean((test_set$Rings-z2)^2)  #test_set MSE
```

MSE1 which is mean squared error for the training set is 4.61, and this number fot rhe test set is 5.17, so the results are good and the test set indicates the model can predict the response variable properly. We still have to do additional steps to get better results.

Linear Regression Model 2

Now, we create the model without significant correlation between variables. this means some of the column that were discussed earlier should be deleted. The code and the results are as follows:

```
Residuals:
    Min      1Q  Median      3Q     Max
-7.5601 -1.5688 -0.5367  0.8953 15.8046

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    5.2906     0.1717  30.821  < 2e-16 ***
SexF           0.8929     0.1134   7.873 4.39e-15 ***
SexM           0.7463     0.1066   7.000 2.97e-12 ***
Height        12.1806     1.9356   6.293 3.43e-10 ***
Shell.weight  10.0177     0.5533  18.105  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.473 on 4172 degrees of freedom
Multiple R-squared:  0.4122,    Adjusted R-squared:  0.4116
F-statistic: 731.3 on 4 and 4172 DF,  p-value: < 2.2e-16
```

Figure 13- LRM2 summary

As the figure indicates, the results have been exacerbated, so it is better not to delete any variable.

Alternative approaches

We have seen that the linear regression model lack some accuracy, so we need to look for substitution approaches. The previous Linear Regression model applies OLS method to obtain the best coefficients. There are some alternative ways to create a Linear Regression model that may be usefull to build a better model.

We have chosen Ridge Regression which is a shrinkage method. This approach is suitable for our model because it tries to make coefficient close to zero. In this way, the model will become more accurate especially in our case that the OLS method is not accurate enough.

The code is as follows:

```
abalone3=read.csv("Abalone.csv")

summary(abalone3)

abalone3$Sex=as.factor(abalone3$Sex)

summary(abalone3)


x1=model.matrix(Rings~ . , data=abalone3)[,-10]

y1=abalone3$Rings


grid=10^seq(10,-2,lenght=100)

install.packages("glmnet")

ridge_model=glmnet(x1, y1, alpha=1, lambda=grid)

summary(ridge_model)


CriticalValue_fit = cv.glmnet(x1, y1, alpha=1, lambda=grid)

plot(CriticalValue_fit)  #ploting Lambda against MSE

optimum_lambda=CriticalValue_fit$lambda.min  #value of Lambda for with minimum MSE


tss=sum((y1-min(y1))^2)  #total sum of square

y_a=predict(ridge_model, s=optimum_lambda, newx=x1)  #predicted values of y
```

rss=sum((y_a-y1)^2) #residual sum of square

RSquared=1-rss/tss #R^2

RSquared

Now the R squared value is about 94 percent which is considerably larger than the previous model.

Results and findings

The best Linear Regression model with OLS method resulted as follows:

```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      2.9121     0.2678  10.875  < 2e-16 ***
SexF             0.8062     0.1021   7.894 3.72e-15 ***
SexM             0.8641     0.0955   9.048  < 2e-16 ***
Length          -0.7093     1.8042  -0.393    0.694
Diameter        10.1924     2.2271   4.577 4.86e-06 ***
Height          16.6478     1.9151   8.693  < 2e-16 ***
Whole.weight     8.9280     0.7232  12.345  < 2e-16 ***
Shucked.weight -19.6556     0.8152 -24.112  < 2e-16 ***
Viscera.weight -10.8459     1.2907  -8.403  < 2e-16 ***
Shell.weight     8.2482     1.1254   7.329 2.76e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 14- Results of the best LRM with OLS method

All the parameter were significant except the diameter. The linear regression formula is as follows:

$$Response = 2.9121 + 0.8062SexF + 0.8641\ SexM - 0.7093\ Length + 10.1924\ Diameter + 16.6478\ Height + 8.9280\ Whole.Weight - 19.6556\ Shucked.Weight - 10.8459 Viscerra.weight + 8.2182\ Shell.weight$$

This can be used to predict the response variable but notice that there is mean squared error with value of 5.17 which was obtained from the test set. Although more variables are necessary to create a more accurate model. But still the results seems to be acceptable.