0.1 Appendix B RCodes and Outputs

0.1.1 Exploratory Data Analysis

```
#import data
abalone<- read.csv("abalone.txt", header=FALSE)
names(abalone)<-c("Sex","Length","Diameter","Height","Whole_weight","Shucked_weight",
                  "Viscera_weight","Shell_weight","Rings")
#calculate age using rings
abalone_age<-cbind(abalone[,-9],"Age"=abalone$Rings+1.5)
#check data type
sapply(abalone_age,class)
```

```
##           Sex        Length      Diameter        Height  Whole_weight Shucked_weight Viscera_weight
##   "character"     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##  Shell_weight           Age
##     "numeric"     "numeric"
```

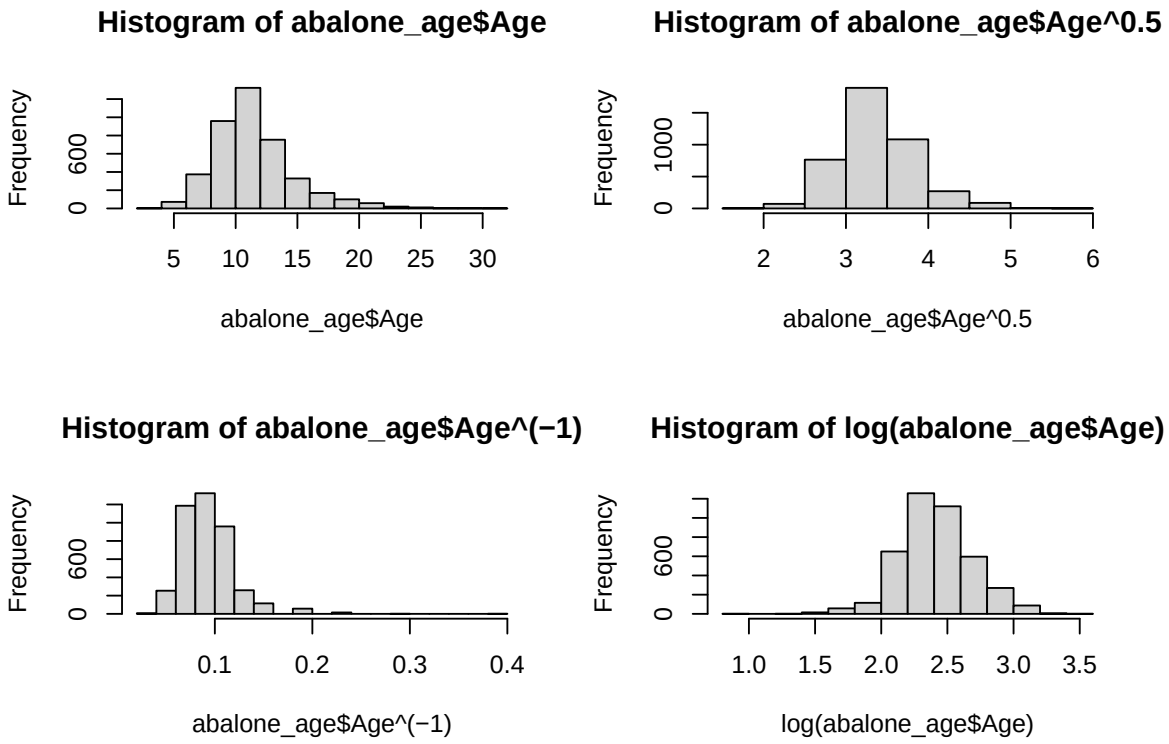
```
#check missing data
sum(is.na(abalone_age))
```

```
## [1] 0
```

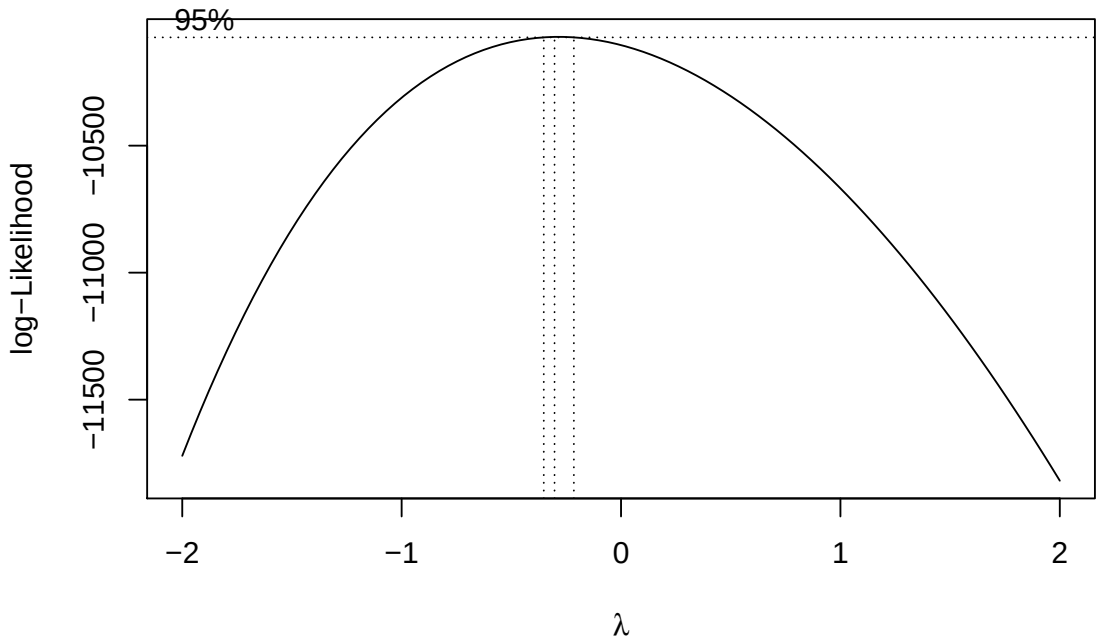
```
#summary data
summary(abalone_age)
```

```
##      Sex               Length         Diameter          Height        Whole_weight    Shucked_weight
##  Length:4177        Min.   :0.075   Min.   :0.0550   Min.   :0.0000   Min.   :0.0020   Min.   :0.0010
##  Class :character   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415   1st Qu.:0.1860
##  Mode  :character   Median :0.545   Median :0.4250   Median :0.1400   Median :0.7995   Median :0.3360
##                     Mean   :0.524   Mean   :0.4079   Mean   :0.1395   Mean   :0.8287   Mean   :0.3594
##                     3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650   3rd Qu.:1.1530   3rd Qu.:0.5020
##                     Max.   :0.815   Max.   :0.6500   Max.   :1.1300   Max.   :2.8255   Max.   :1.4880
##  Viscera_weight    Shell_weight        Age
##  Min.   :0.0005   Min.   :0.0015   Min.   : 2.50
##  1st Qu.:0.0935   1st Qu.:0.1300   1st Qu.: 9.50
##  Median :0.1710   Median :0.2340   Median :10.50
##  Mean   :0.1806   Mean   :0.2388   Mean   :11.43
##  3rd Qu.:0.2530   3rd Qu.:0.3290   3rd Qu.:12.50
##  Max.   :0.7600   Max.   :1.0050   Max.   :30.50
```

```
#distribution of response variable
par(mfrow=c(2,2))
hist(abalone_age$Age)
hist(abalone_age$Age^0.5)
hist(abalone_age$Age^(-1))
hist(log(abalone_age$Age))
```



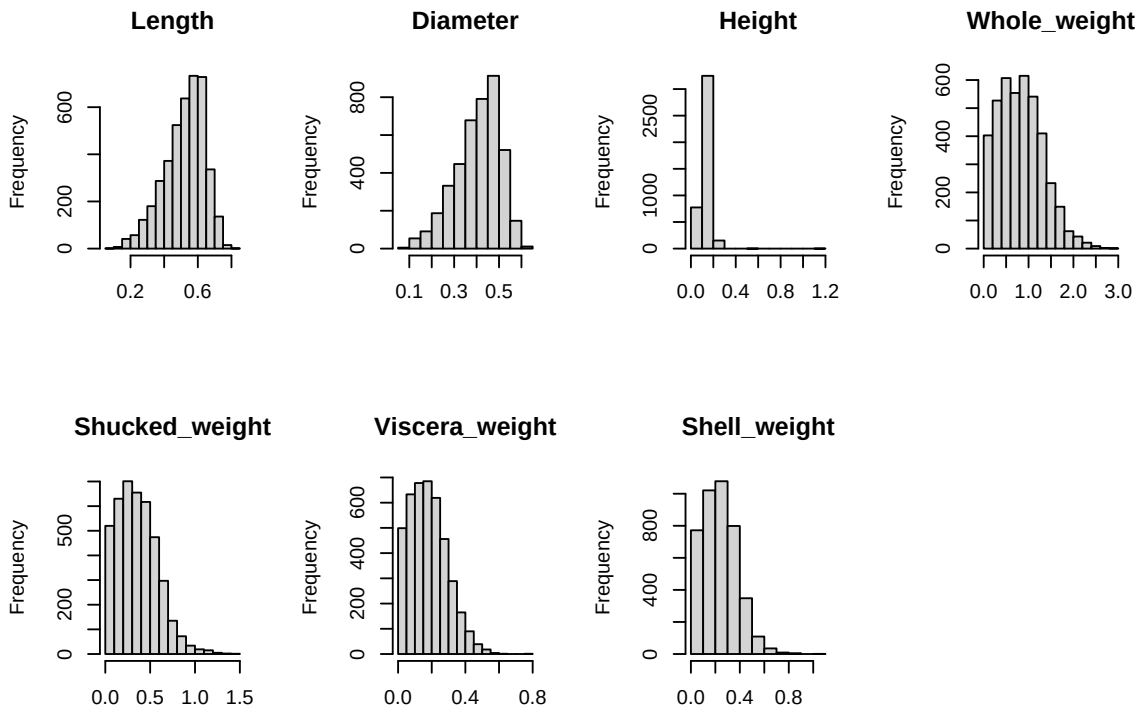

```
#distribution of response variable
par(mfrow=c(1,1))
boxcox(Age ~ ., data = abalone_age)
```



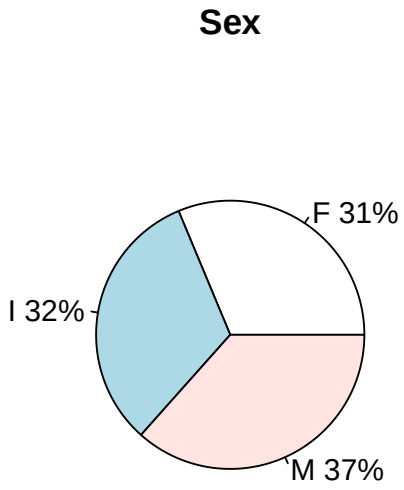

```
#use log transformation
abalone_log_age<-cbind(abalone_age[,-9],log_age=log(abalone_age$Age))
#distribution of quantitive variables
par(mfrow=c(2,4))
sapply(names(abalone_log_age)[2:8],function(x) hist(abalone_log_age[[x]],main = x,xlab =""))
```

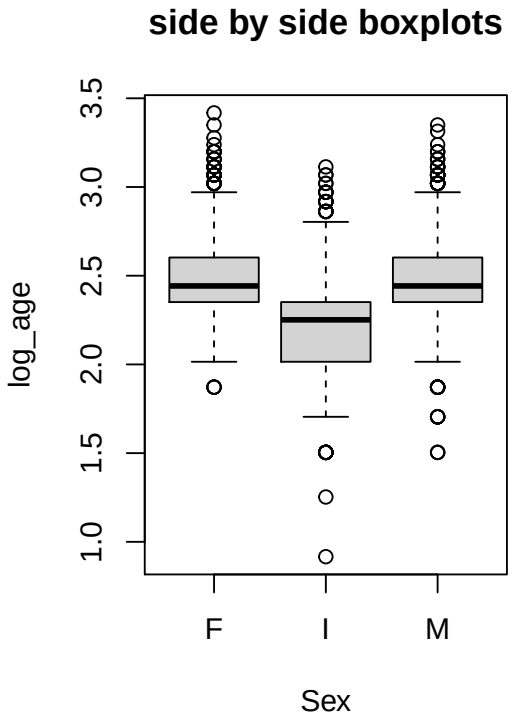
```
##          Length              Diameter             Height              Whole_weight
## breaks   Numeric,17          Numeric,13           Numeric,13          Numeric,16
## counts   Integer,16          Integer,12           Integer,12          Integer,15
## density  Numeric,16          Numeric,12           Numeric,12          Numeric,15
## mids     Numeric,16          Numeric,12           Numeric,12          Numeric,15
## xname    "abalone_log_age[[x]]" "abalone_log_age[[x]]" "abalone_log_age[[x]]" "abalone_log_age[[x]]"
## equidist TRUE                TRUE                 TRUE                TRUE
##          Shucked_weight      Viscera_weight       Shell_weight
## breaks   Numeric,16          Numeric,17           Numeric,12
## counts   Integer,15          Integer,16           Integer,11
## density  Numeric,15          Numeric,16           Numeric,11
## mids     Numeric,15          Numeric,16           Numeric,11
## xname    "abalone_log_age[[x]]" "abalone_log_age[[x]]" "abalone_log_age[[x]]"
## equidist TRUE                TRUE                 TRUE
```

```
#distribution of categorical variable
par(mfrow=c(1,2))
```

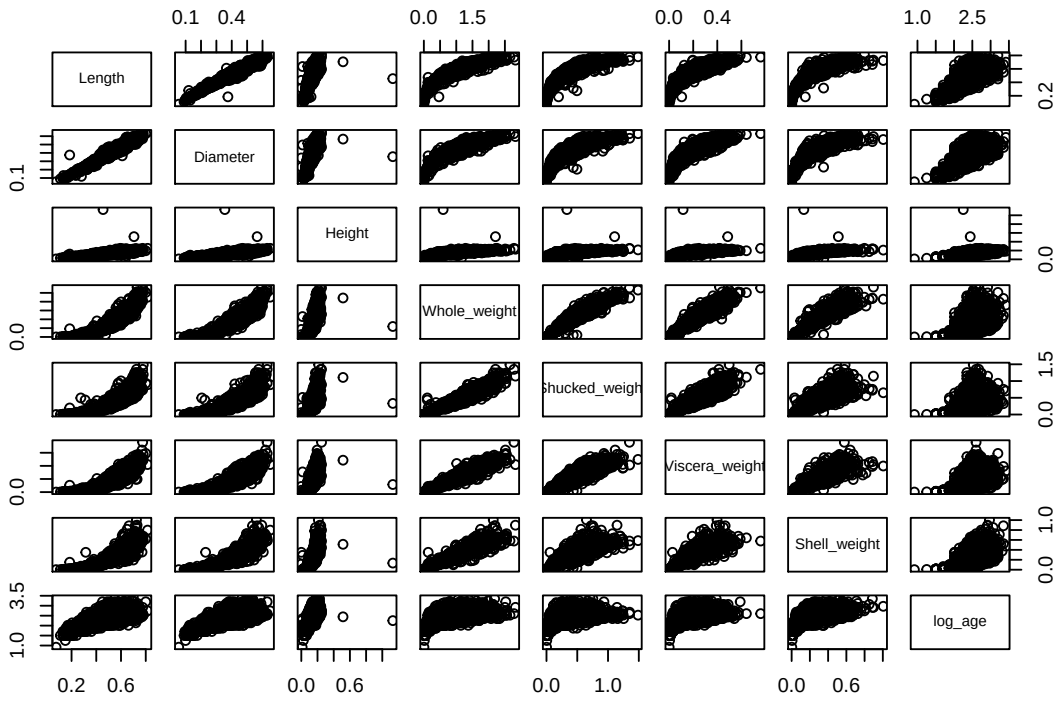



```
lbls=c("F","I","M")
pct=round(100*table(abalone_log_age$Sex)/4177)
lab=paste(lbls,pct)
lab=paste(lab,"%",sep="")
pie(table(abalone_log_age$Sex),labels=lab,main="Sex")
boxplot(abalone_log_age$log_age~abalone_log_age$Sex,main='side by side boxplots',
      xlab='Sex',ylab='log_age')
```

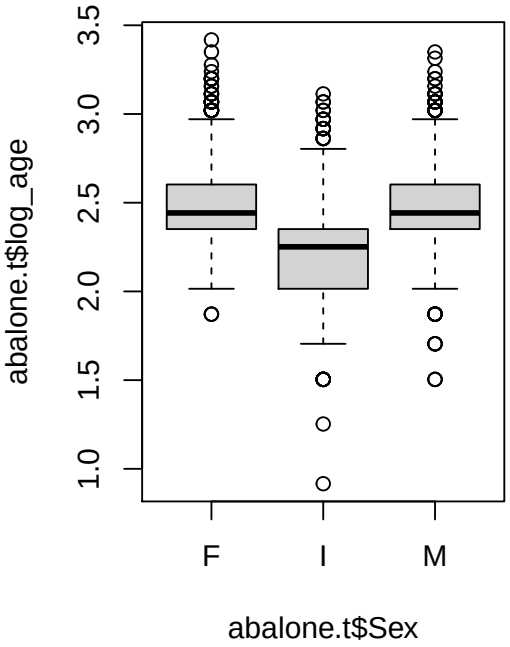




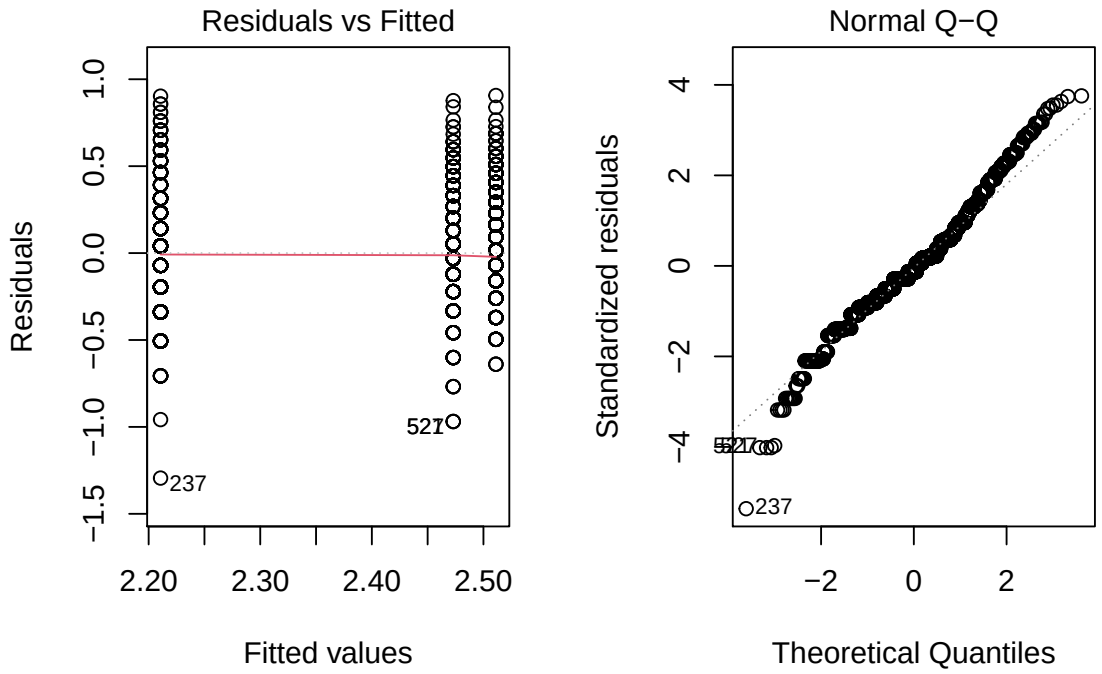

```
#scatter plot matrix
pairs(abalone_log_age[!sapply(abalone_log_age,class)=='character'])
```




```
#split the data into train and test
abalone.s<-abalone_log_age
set.seed(1234)
n.s=nrow(abalone.s)
index.s=sample(1: n.s, size=n.s/5, replace=FALSE)#randomly sample 80% cases
abalone.v=abalone.s[index.s,]  ## training data set
abalone.t=abalone.s[-index.s,] ## validation set.
n=nrow(abalone.t)
abaloneo.t <- abalone_age[index.s,]
abaloneo.v <- abalone_age[-index.s,]
#simple factor anova on Sex
boxplot(abalone.t$log_age~abalone.t$Sex)
abnovamodel <- lm(log_age~Sex, data = abalone.t)
par(mfrow=c(1,2))
```



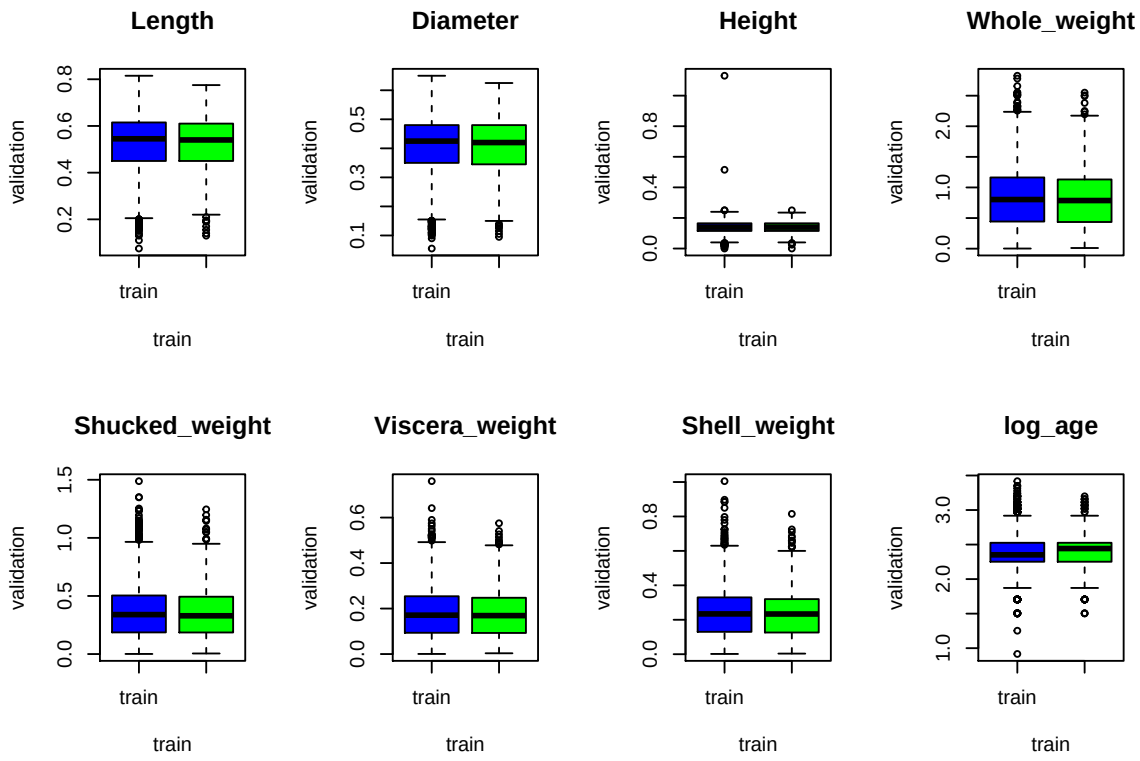

```
plot(abnovamodel,which = 1)
plot(abnovamodel,which = 2)
```




```
summary(abnovamodel)
```

```
##
## Call:
## lm(formula = log_age ~ Sex, data = abalone.t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2944 -0.1598 -0.0306  0.1407  0.9066
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.511170   0.007425 338.208  < 2e-16 ***
## SexI        -0.300494   0.010449 -28.758  < 2e-16 ***
## SexM        -0.038219   0.010169  -3.758 0.000174 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2414 on 3339 degrees of freedom
## Multiple R-squared:   0.23,  Adjusted R-squared:  0.2296
## F-statistic: 498.7 on 2 and 3339 DF,  p-value: < 2.2e-16
```

```
#plot training data set againt validation data set
dt = abalone.t
dv = abalone.v
dt$tv="train"
dv$tv="validation"
dall=rbind(dt,dv)
names = names(abalone.s)
names = names[names!= "Sex"]
par(mfrow=c(2,4))
for(name in names)
{
 boxplot(dall[[name]]~dall$tv,main=name,
       xlab='train',ylab='validation',col=c("blue","green"))
}
```




```
#Correlationa and VIF
cor(abaloneo.t[2:9])
```

```
##                Length  Diameter    Height Whole_weight Shucked_weight Viscera_weight Shell_weight
## Length       1.0000000 0.9879276 0.8922750    0.9245302      0.8946757      0.9039078    0.9013089
## Diameter     0.9879276 1.0000000 0.8949310    0.9249926      0.8903638      0.9014630    0.9088631
## Height       0.8922750 0.8949310 1.0000000    0.8795695      0.8229852      0.8651495    0.8941262
```

```
## Whole_weight   0.9245302 0.9249926 0.8795695    1.0000000      0.9701887      0.9718170    0.9591222
## Shucked_weight 0.8946757 0.8903638 0.8229852    0.9701887      1.0000000      0.9347480    0.8863436
## Viscera_weight 0.9039078 0.9014630 0.8651495    0.9718170      0.9347480      1.0000000    0.9189635
## Shell_weight   0.9013089 0.9088631 0.8941262    0.9591222      0.8863436      0.9189635    1.0000000
## Age            0.5429988 0.5575955 0.5881608    0.5137147      0.4018165      0.4876196    0.5975764
##                      Age
## Length         0.5429988
## Diameter       0.5575955
## Height         0.5881608
## Whole_weight   0.5137147
## Shucked_weight 0.4018165
## Viscera_weight 0.4876196
## Shell_weight   0.5975764
## Age            1.0000000
```

```
rxx<-cor(abaloneo.t[2:8])
rxxI<-solve(rxx)
(vif<-diag(rxxI))
```

```
##       Length     Diameter       Height Whole_weight Shucked_weight Viscera_weight   Shell_weight
##    44.693902    45.446812     6.450176   148.326779    33.922719      22.345792      28.259461
```

0.1.2 Model Building and Diagnostic

```
#first-order model
fit1 = lm(log_age ~., data = abalone.t)
summary(fit1)
```

```
##
## Call:
## lm(formula = log_age ~ ., data = abalone.t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.90125 -0.11471 -0.01730  0.09249  0.70817
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.653184   0.025750  64.202  < 2e-16 ***
## SexI          -0.068570   0.009053  -7.574 4.65e-14 ***
## SexM           0.009963   0.007411   1.344   0.1789
## Length         0.317279   0.158557   2.001   0.0455 *
## Diameter       1.280300   0.195716   6.542 7.02e-11 ***
## Height         0.940768   0.128236   7.336 2.75e-13 ***
## Whole_weight   0.559886   0.062414   8.971  < 2e-16 ***
## Shucked_weight -1.485720   0.071299 -20.838  < 2e-16 ***
```

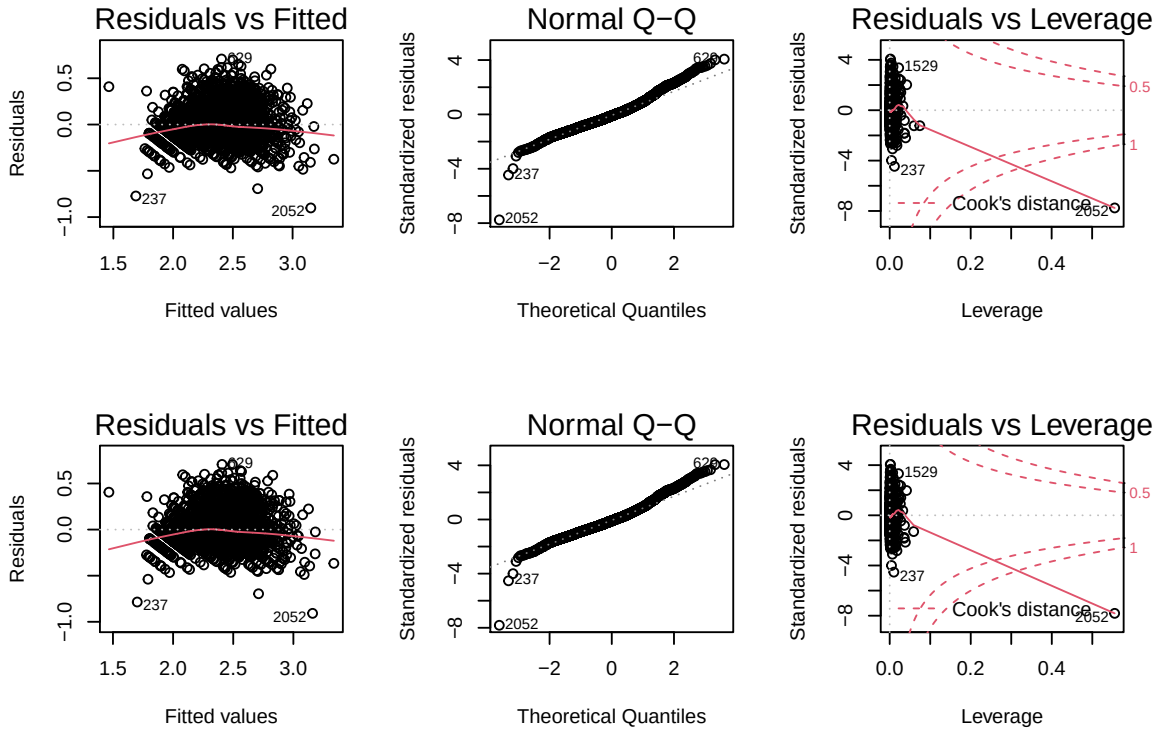
```
## Viscera_weight -0.691850   0.112061  -6.174 7.47e-10 ***
## Shell_weight    0.587010   0.097145   6.043 1.68e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1742 on 3332 degrees of freedom
## Multiple R-squared:  0.5997, Adjusted R-squared:  0.5986
## F-statistic: 554.7 on 9 and 3332 DF,  p-value: < 2.2e-16
```

```
#first-order subset
library(leaps)
sub_set <- regsubsets(log_age~., data=abalone.t, nbest=1,nvmax=9, method="exhaustive")
sum_sub <- summary(sub_set)
p.m <- as.integer(rownames(sum_sub$which))+1
ssto <- sum((abalone.t$log_age-mean(abalone.t$log_age))^2)
sse <- (1-sum_sub$rsq)*ssto
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, bic, aic)
colnames(res_sub) <- c(colnames(sum_sub$which),"sse", "R^2", "R^2_a", "Cp","bic", "aic")
res_sub
```

```
##   (Intercept) SexI SexM Length Diameter Height Whole_weight Shucked_weight Viscera_weight Shell_weight
## 1           1    0    0      0        0      0            0              0              0            1
## 2           1    0    0      0        0      0            0              1              0            1
## 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      bic      aic
## 1 138.7007 0.4511156 0.4509512 1230.929111 -10618.04 -10630.27
## 2 126.7682 0.4983365 0.4980361  839.860300 -10910.56 -10928.91
## 3 108.6915 0.5698719 0.5694853  246.398401 -11416.60 -11441.06
## 4 105.5093 0.5824650 0.5819645  143.572632 -11507.80 -11538.37
## 5 103.7660 0.5893637 0.5887483   88.147507 -11555.36 -11592.05
## 6 102.4200 0.5946901 0.5939609   45.810838 -11590.88 -11633.68
## 7 101.3283 0.5990104 0.5981685   11.848226 -11618.58 -11667.49
## 8 101.2056 0.5994959 0.5985346    9.807109 -11614.51 -11669.54
## 9 101.1508 0.5997130 0.5986318   10.000000 -11608.21 -11669.36
```

```
#Stepwise
library(MASS)
fit0<-lm(log_age~1,data=abalone.t)
model_1_fs_aic<-stepAIC(fit0,scope=list(upper=fit1,lower=~1),direction = "both",k=2,trace = FALSE)
```

```
model_1_fs_bic<-stepAIC(fit0,scope=list(upper=fit1,lower=~1),direction = "both",k=log(n),trace = FALSE)
model_1_f_aic<-stepAIC(fit0,scope=list(upper=fit1,lower=~1),direction = "forward",k=2,trace = FALSE)
model_1_f_bic<-stepAIC(fit0,scope=list(upper=fit1,lower=~1),direction = "forward",k=log(n),trace = FALSE)
model_1_b_aic<-stepAIC(fit1,scope=list(upper=fit1,lower=~1),direction = "back",k=2,trace = FALSE)
model_1_b_bic<-stepAIC(fit1,scope=list(upper=fit1,lower=~1),direction = "back",k=log(n),trace = FALSE)
model_1_bs_aic<-stepAIC(fit1,scope=list(upper=fit1,lower=~1),direction = "both",k=2,trace = FALSE)
model_1_bs_bic<-stepAIC(fit1,scope=list(upper=fit1,lower=~1),direction = "both",k=log(n),trace = FALSE)
#diagnostic
par(mfrow=c(2,3))
plot(fit1,which=1)
plot(fit1,which=2)
plot(fit1,which=5)
plot(model_1_b_bic,which=1)
plot(model_1_b_bic,which=2)
plot(model_1_b_bic,which=5)
```




```
#summary
fmodel1t <- model_1_b_bic
fmodel2t <- fit1
summary(fmodel1t)
```

##

```
## Call:
## lm(formula = log_age ~ Sex + Diameter + Height + Whole_weight +
##     Shucked_weight + Viscera_weight + Shell_weight, data = abalone.t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.90851 -0.11350 -0.01759  0.09438  0.70670
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.669887   0.024370  68.522  < 2e-16 ***
## SexI           -0.067341   0.009036  -7.452 1.16e-13 ***
## SexM            0.010064   0.007414   1.357    0.175
## Diameter        1.630663   0.087492  18.638  < 2e-16 ***
## Height          0.945975   0.128267   7.375 2.06e-13 ***
## Whole_weight    0.558587   0.062439   8.946  < 2e-16 ***
## Shucked_weight -1.474758   0.071120 -20.736  < 2e-16 ***
## Viscera_weight -0.669948   0.111575  -6.004 2.13e-09 ***
## Shell_weight    0.581355   0.097147   5.984 2.40e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1743 on 3333 degrees of freedom
## Multiple R-squared:  0.5992, Adjusted R-squared:  0.5983
## F-statistic: 622.9 on 8 and 3333 DF,  p-value: < 2.2e-16
```

```
summary(fmodel2t)
```

```
##
## Call:
## lm(formula = log_age ~ ., data = abalone.t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.90125 -0.11471 -0.01730  0.09249  0.70817
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.653184   0.025750  64.202  < 2e-16 ***
## SexI           -0.068570   0.009053  -7.574 4.65e-14 ***
## SexM            0.009963   0.007411   1.344   0.1789
## Length          0.317279   0.158557   2.001   0.0455 *
## Diameter        1.280300   0.195716   6.542 7.02e-11 ***
## Height          0.940768   0.128236   7.336 2.75e-13 ***
## Whole_weight    0.559886   0.062414   8.971  < 2e-16 ***
## Shucked_weight -1.485720   0.071299 -20.838  < 2e-16 ***
```

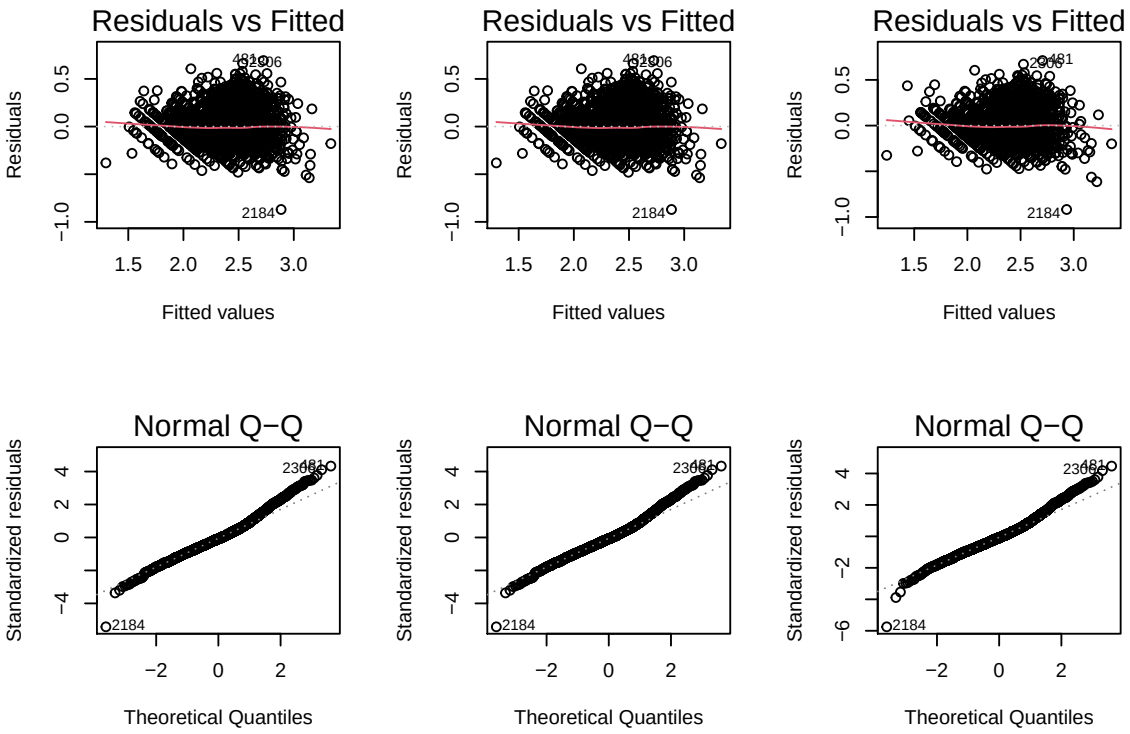
```
## Viscera_weight -0.691850   0.112061  -6.174 7.47e-10 ***
## Shell_weight    0.587010   0.097145   6.043 1.68e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1742 on 3332 degrees of freedom
## Multiple R-squared:  0.5997, Adjusted R-squared:  0.5986
## F-statistic: 554.7 on 9 and 3332 DF,  p-value: < 2.2e-16
```

```
#Second order: add interaction and quadratic terms
#center
abalone.s_c<-cbind(Sex=abalone.s[,1],as.data.frame(
  sapply(abalone.s[,2:8],function(x) x-mean(x))),log_age=abalone.s[,9])
abalone.v_c=abalone.s_c[index.s,]
abalone.t_c=abalone.s_c[-index.s,]
#fit with all terms
fit2 = lm(log_age ~ .+.^2+I(Length^2)+I(Diameter^2)+I(Height^2)+I(Whole_weight^2)+
          I(Shucked_weight^2)+I(Viscera_weight^2)+I(Shell_weight^2), data =abalone.t_c)
summary(fit2)
```

```
##
## Call:
## lm(formula = log_age ~ . + .^2 + I(Length^2) + I(Diameter^2) +
##     I(Height^2) + I(Whole_weight^2) + I(Shucked_weight^2) + I(Viscera_weight^2) +
##     I(Shell_weight^2), data = abalone.t_c)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.91879 -0.10326 -0.01245  0.08632  0.71898
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             2.456e+00  7.528e-03 326.216  < 2e-16 ***
## SexI                   -8.619e-04  1.211e-02  -0.071  0.94326
## SexM                   -7.104e-04  8.104e-03  -0.088  0.93015
## Length                 -5.061e-01  2.919e-01  -1.734  0.08309 .
## Diameter                9.374e-01  3.612e-01   2.595  0.00949 **
## Height                  8.943e-01  3.536e-01   2.530  0.01147 *
## Whole_weight            8.417e-01  1.277e-01   6.591 5.08e-11 ***
## Shucked_weight         -2.108e+00  1.430e-01 -14.748  < 2e-16 ***
## Viscera_weight         -4.605e-01  2.396e-01  -1.922  0.05471 .
## Shell_weight            9.855e-01  2.082e-01   4.733 2.30e-06 ***
## I(Length^2)            -6.395e+00  1.981e+00  -3.228  0.00126 **
## I(Diameter^2)          -1.030e+01  4.505e+00  -2.287  0.02224 *
## I(Height^2)            -8.936e-01  5.660e-01  -1.579  0.11446
## I(Whole_weight^2)       1.537e-01  2.493e-01   0.616  0.53775
```

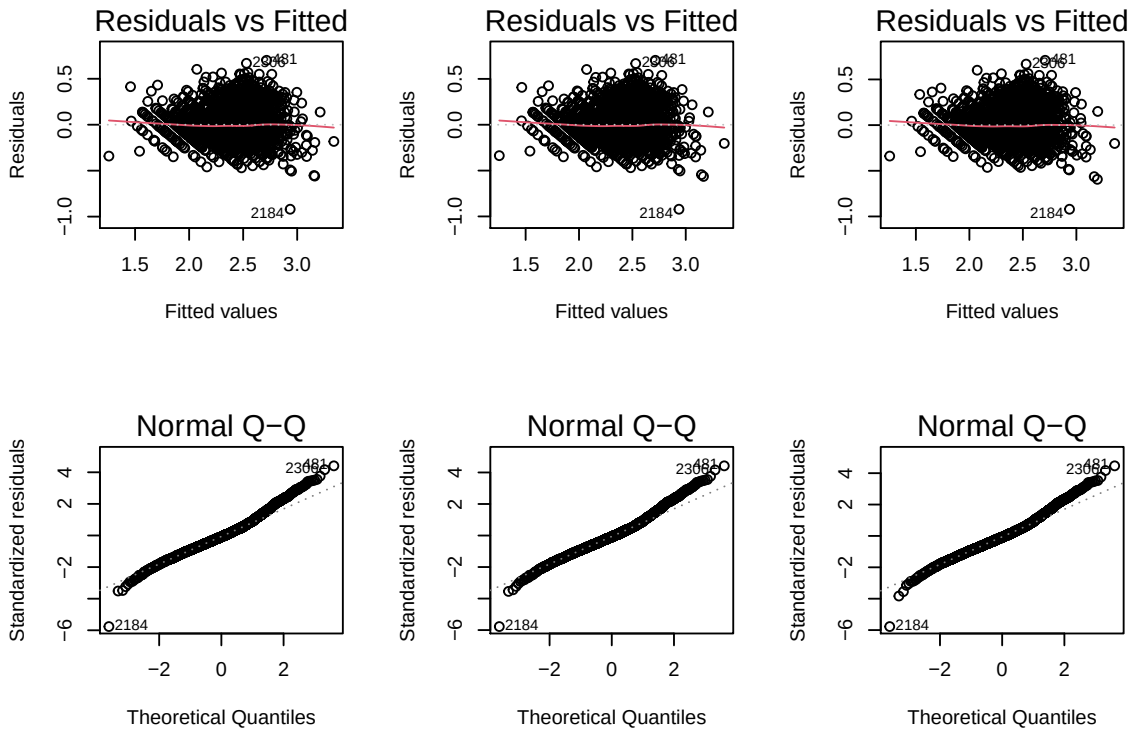
```
## I(Shucked_weight^2)          3.605e+00  5.161e-01   6.986 3.41e-12 ***
## I(Viscera_weight^2)          7.297e-01  1.643e+00   0.444  0.65688
## I(Shell_weight^2)           -3.713e-01  8.336e-01  -0.445  0.65600
## SexI:Length                 -4.951e-01  4.743e-01  -1.044  0.29664
## SexM:Length                  4.053e-01  3.712e-01   1.092  0.27499
## SexI:Diameter               -3.418e-01  5.848e-01  -0.584  0.55895
## SexM:Diameter               -6.120e-01  4.489e-01  -1.363  0.17282
## SexI:Height                  1.178e+00  6.078e-01   1.938  0.05270 .
## SexM:Height                  4.089e-02  4.304e-01   0.095  0.92432
## SexI:Whole_weight           -4.255e-02  2.406e-01  -0.177  0.85965
## SexM:Whole_weight            3.043e-02  1.304e-01   0.233  0.81553
## SexI:Shucked_weight          7.602e-01  2.765e-01   2.749  0.00601 **
## SexM:Shucked_weight          4.240e-02  1.485e-01   0.286  0.77527
## SexI:Viscera_weight         -1.272e-02  4.412e-01  -0.029  0.97700
## SexM:Viscera_weight         -1.454e-01  2.315e-01  -0.628  0.53008
## SexI:Shell_weight            5.840e-02  3.851e-01   0.152  0.87949
## SexM:Shell_weight            1.087e-01  2.084e-01   0.522  0.60191
## Length:Diameter              7.009e+00  4.919e+00   1.425  0.15426
## Length:Height               -9.636e+00  1.058e+01  -0.910  0.36267
## Length:Whole_weight         -1.727e-01  3.098e+00  -0.056  0.95555
## Length:Shucked_weight        4.959e+00  3.547e+00   1.398  0.16227
## Length:Viscera_weight       -4.178e+00  5.459e+00  -0.765  0.44413
## Length:Shell_weight          1.618e+00  4.910e+00   0.330  0.74168
## Diameter:Height              1.348e+01  1.286e+01   1.048  0.29449
## Diameter:Whole_weight        2.087e+00  3.784e+00   0.551  0.58136
## Diameter:Shucked_weight      1.364e-01  4.371e+00   0.031  0.97510
## Diameter:Viscera_weight      5.897e-01  6.837e+00   0.086  0.93127
## Diameter:Shell_weight       -4.729e+00  5.859e+00  -0.807  0.41962
## Height:Whole_weight          8.979e-01  3.172e+00   0.283  0.77715
## Height:Shucked_weight       -1.765e+00  3.598e+00  -0.491  0.62381
## Height:Viscera_weight       -5.186e+00  6.401e+00  -0.810  0.41789
## Height:Shell_weight          5.531e-01  4.085e+00   0.135  0.89229
## Whole_weight:Shucked_weight  -3.173e+00  6.628e-01  -4.787 1.77e-06 ***
## Whole_weight:Viscera_weight   1.635e+00  1.238e+00   1.321  0.18663
## Whole_weight:Shell_weight    -1.306e-01  9.356e-01  -0.140  0.88901
## Shucked_weight:Viscera_weight -1.261e-01  1.467e+00  -0.086  0.93152
## Shucked_weight:Shell_weight   1.652e+00  1.293e+00   1.278  0.20138
## Viscera_weight:Shell_weight  -2.805e+00  2.053e+00  -1.366  0.17202
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1608 on 3290 degrees of freedom
## Multiple R-squared:  0.6634, Adjusted R-squared:  0.6582
## F-statistic: 127.2 on 51 and 3290 DF,  p-value: < 2.2e-16
```

```
#Stepwise
model_2_fs_aic<-stepAIC(fit0,scope=list(upper=fit2,lower=~1),direction = "both",k=2,trace = FALSE)
model_2_fs_bic<-stepAIC(fit0,scope=list(upper=fit2,lower=~1),direction ="both",k=log(n),trace = FALSE)
model_2_f_aic<-stepAIC(fit0,scope=list(upper=fit2,lower=~1),direction = "forward",k=2,trace = FALSE)
model_2_f_bic<-stepAIC(fit0,scope=list(upper=fit2,lower=~1),direction = "forward",k=log(n),trace = FALSE)
model_2_b_aic<-stepAIC(fit2,scope=list(upper=fit2,lower=~1),direction = "back",k=2,trace = FALSE)
model_2_b_bic<-stepAIC(fit2,scope=list(upper=fit2,lower=~1),direction = "back",k=log(n),trace = FALSE)
model_2_bs_aic<-stepAIC(fit2,scope=list(upper=fit2,lower=~1),direction = "both",k=2,trace = FALSE)
model_2_bs_bic<-stepAIC(fit2,scope=list(upper=fit2,lower=~1),direction = "both",k=log(n),trace = FALSE)
#diagnostic
par(mfrow=c(2,3))
plot(model_2_fs_bic,which=1)
plot(model_2_f_bic,which=1)
plot(model_2_b_bic,which=1)
plot(model_2_fs_bic,which=2)
plot(model_2_f_bic,which=2)
plot(model_2_b_bic,which=2)
```




```
plot(model_2_fs_aic,which=1)
plot(model_2_f_aic,which=1)
plot(model_2_b_aic,which=1)
plot(model_2_fs_aic,which=2)
```

```
plot(model_2_f_aic,which=2)
plot(model_2_b_aic,which=2)
```




```
fmodel3t <- model_2_fs_bic
fmodel4t <- model_2_f_bic
fmodel5t <- model_2_b_bic
fmodel6t <- model_2_fs_aic
fmodel7t <- model_2_f_aic
fmodel8t <- model_2_b_aic
```

0.1.3 Model Validation

```
#fit the model with validation data
fmodel1v <- lm(formula(model_1_b_bic),data = abalone.v)
fmodel2v <- lm(formula(fit1),data = abalone.v)
fmodel3v <- lm(formula(model_2_fs_bic),data = abalone.v_c)
fmodel4v <- lm(formula(model_2_f_bic),data = abalone.v_c)
fmodel5v <- lm(formula(model_2_b_bic),data = abalone.v_c)
fmodel6v <- lm(formula(model_2_fs_aic),data = abalone.v_c)
fmodel7v <- lm(formula(model_2_f_aic),data = abalone.v_c)
fmodel8v <- lm(formula(model_2_b_aic),data = abalone.v_c)
#check criterions
```

```r
sigma <- anova(fit2)['Residuals',3]
sigma
```

```
## [1] 0.02584985
```

```r
for(i in 1:8){
  print(paste("fmodel",i,sep=''))
  modelname_t <- get(paste("fmodel",i,'t',sep=''))
  modelname_v <- get(paste("fmodel",i,'v',sep=''))

  #compare coefficient estimator
  est <- cbind(summary(modelname_t)$coefficients[,1:2],summary(modelname_v)$coefficients[,1:2])
  colnames(est) <- c("training_est","training_std","validation_est","validation_std")
  print(est)

  #various criteria
  vali_summary<-matrix(NA,nrow=2,ncol=8)
  rownames(vali_summary)<-c("training","validation")
  colnames(vali_summary)<-c("SSE","MSE","Cp","Pressp","SSE/n","Pressp/n","MSPE","p")

  vali_summary[1,2] <- anova(modelname_t)['Residuals',3] #mse train
  vali_summary[1,1] <- anova(modelname_t)['Residuals',2] #sse train
  vali_summary[1,3] <- vali_summary[1,1]/sigma-n+2*length(modelname_t$coefficients) #cp train
  vali_summary[1,8] <- length(modelname_t$coefficients) #p
  vali_summary[1,4] <- sum(modelname_t$residuals^2/(1-influence(modelname_t)$hat)^2) #pressp train
  vali_summary[1,5] <- vali_summary[1,1]/n
  vali_summary[1,6] <- vali_summary[1,4]/n
  vali_summary[2,2] <- anova(modelname_v)['Residuals',3] #mse vali
  vali_summary[2,1] <- anova(modelname_v)['Residuals',2] #sse vali
  if(i==1|i==2){
    vali_summary[2,7] <- mean((predict.lm(modelname_t,abalone.v)-abalone.v$log_age)^2) #mspe
  }
  else{
    vali_summary[2,7] <- mean((predict.lm(modelname_t,abalone.v_c)-abalone.v$log_age)^2) #mspe
  }
  print(vali_summary)
}
```

```
## [1] "fmodel1"
##                training_est training_std validation_est validation_std
## (Intercept)     1.66988656  0.024370208    1.751133659     0.05018661
## SexI           -0.06734077  0.009036181   -0.118174613     0.01844963
## SexM            0.01006364  0.007414179   -0.001797965     0.01468073
## Diameter        1.63066331  0.087492113    1.300606234     0.18063647
## Height          0.94597526  0.128266987    1.749015635     0.39687751
## Whole_weight    0.55858738  0.062438585    0.592267641     0.14982528
```

```
## Shucked_weight  -1.47475838  0.071120493   -1.350455722     0.15782406
## Viscera_weight  -0.66994753  0.111575053   -0.932347479     0.25907863
## Shell_weight     0.58135472  0.097147435    0.305939186     0.23026362
##                SSE       MSE      Cp   Pressp      SSE/n   Pressp/n       MSPE  p
## training   101.27233 0.03038474 593.7145 105.2581 0.03030291 0.03149554         NA  9
## validation  24.59242 0.02977291       NA       NA         NA         NA 0.03024175 NA
## [1] "fmodel2"
##                training_est training_std validation_est validation_std
## (Intercept)     1.653184466  0.025749557    1.729763446     0.05313794
## SexI           -0.068569746  0.009052968   -0.118956159     0.01845527
## SexM            0.009962531  0.007411012   -0.002024406     0.01467755
## Length          0.317279449  0.158557210    0.413612573     0.33881031
## Diameter        1.280299806  0.195715882    0.850504433     0.41054864
## Height          0.940767992  0.128235626    1.695106685     0.39920969
## Whole_weight    0.559885742  0.062413838    0.593244394     0.14978297
## Shucked_weight -1.485719840  0.071299205   -1.363901537     0.15816121
## Viscera_weight -0.691849745  0.112060626   -0.954280427     0.25962416
## Shell_weight    0.587010356  0.097144807    0.322561501     0.23059766
##                SSE       MSE      Cp   Pressp      SSE/n   Pressp/n       MSPE  p
## training   101.15077 0.03035738 591.0121 105.1621 0.03026654 0.03146681         NA 10
## validation  24.54808 0.02975525       NA       NA         NA         NA 0.03015866 NA
## [1] "fmodel3"
##                           training_est training_std validation_est validation_std
## (Intercept)                1.14146297   0.06744187    2.480303145     0.01374599
## Shell_weight               1.46725644   0.20779014    1.146512165     0.28679836
## I(Shucked_weight^2)        3.14492155   0.29675186    2.855456298     0.62425589
## I(Shell_weight^2)          -0.95944458   0.22481518   -1.604966639     0.54779548
## Shucked_weight             -2.77719699   0.14536158   -1.859483845     0.19062712
## Diameter                    6.10548098   0.57040205   -0.290477018     0.27126355
## I(Diameter^2)              -9.93119248   1.16434834   -8.387542618     3.05559766
## Whole_weight                0.32434105   0.18004565    0.842139741     0.15836407
## SexI                       -0.17224662   0.01853949   -0.030997644     0.02176138
## SexM                       -0.03232796   0.01618672   -0.009240565     0.01465494
## Viscera_weight             -0.52579493   0.10588884   -0.627023905     0.24753736
## I(Length^2)                -0.50931254   0.14749201   -2.195219066     1.37214530
## Height                      2.48301322   0.39754127    0.510621503     0.39470430
## I(Height^2)                -1.46623558   0.28302562    5.234226614     7.86411527
## Shucked_weight:SexI         0.44201707   0.05636833    0.569348556     0.12006364
## Shucked_weight:SexM         0.08149439   0.03332149    0.023850305     0.06570030
## Shucked_weight:Whole_weight  -1.76547438   0.21381349   -1.564613586     0.40898466
## Diameter:Whole_weight        3.14902765   0.47939912    3.378712532     0.88642825
## Whole_weight:Height         -0.99085549   0.26936714   -1.808485295     1.40034315
##                SSE       MSE      Cp   Pressp      SSE/n  Pressp/n     MSPE  p
## training   86.35825 0.02598804 36.76401 109.7867 0.02584029 0.0328506       NA 19
## validation 20.71414 0.02538498       NA       NA         NA         NA 2.077675 NA
```

```
## [1] "fmodel4"
##                          training_est training_std validation_est validation_std
## (Intercept)                1.14146297   0.06744187    2.480303145     0.01374599
## Shell_weight               1.46725644   0.20779014    1.146512165     0.28679836
## I(Shucked_weight^2)        3.14492155   0.29675186    2.855456298     0.62425589
## I(Shell_weight^2)         -0.95944458   0.22481518   -1.604966639     0.54779548
## Shucked_weight            -2.77719699   0.14536158   -1.859483845     0.19062712
## Diameter                   6.10548098   0.57040205   -0.290477018     0.27126355
## I(Diameter^2)             -9.93119248   1.16434834   -8.387542618     3.05559766
## Whole_weight               0.32434105   0.18004565    0.842139741     0.15836407
## SexI                      -0.17224662   0.01853949   -0.030997644     0.02176138
## SexM                      -0.03232796   0.01618672   -0.009240565     0.01465494
## Viscera_weight            -0.52579493   0.10588884   -0.627023905     0.24753736
## I(Length^2)               -0.50931254   0.14749201   -2.195219066     1.37214530
## Height                     2.48301322   0.39754127    0.510621503     0.39470430
## I(Height^2)               -1.46623558   0.28302562    5.234226614     7.86411527
## Shucked_weight:SexI        0.44201707   0.05636833    0.569348556     0.12006364
## Shucked_weight:SexM        0.08149439   0.03332149    0.023850305     0.06570030
## Shucked_weight:Whole_weight -1.76547438   0.21381349   -1.564613586     0.40898466
## Diameter:Whole_weight      3.14902765   0.47939912    3.378712532     0.88642825
## Whole_weight:Height       -0.99085549   0.26936714   -1.808485295     1.40034315
##                SSE       MSE      Cp   Pressp     SSE/n  Pressp/n     MSPE  p
## training   86.35825 0.02598804 36.76401 109.7867 0.02584029 0.0328506       NA 19
## validation 20.71414 0.02538498       NA       NA        NA        NA 2.077675 NA
## [1] "fmodel5"
##                          training_est training_std validation_est validation_std
## (Intercept)              2.456742779  0.006849167    2.475742716     0.01386032
## SexI                    -0.010618313  0.010305020   -0.035934748     0.02256273
## SexM                    -0.004400265  0.007367586   -0.008371266     0.01470203
## Length                  -0.530264871  0.162205580   -0.392501793     0.35674541
## Diameter                 0.632906516  0.189840839    0.294358329     0.42111976
## Height                   1.253617883  0.206150790    0.588288689     0.39612530
## Whole_weight             0.872021433  0.074999184    0.912638612     0.17063721
## Shucked_weight          -2.032276858  0.092148258   -1.917249736     0.19490911
## Viscera_weight          -0.590700492  0.139359862   -0.627969397     0.30801559
## Shell_weight             0.979467142  0.115454533    0.835463694     0.27678130
## I(Length^2)             -4.396671502  0.950869880   -1.989826989     1.97628009
## I(Diameter^2)           -4.929443351  1.661323593   -6.123858038     3.47129672
## I(Height^2)             -1.493089640  0.289428764   -0.550809809     7.27736593
## I(Shucked_weight^2)      3.002747063  0.334960776    3.646421589     0.68858092
## SexI:Shucked_weight      0.459965804  0.056929560    0.542919973     0.12610736
## SexM:Shucked_weight      0.081543384  0.033243167    0.021187964     0.06634309
## Length:Shucked_weight    3.542631743  0.896973001    0.340755500     1.45355733
## Diameter:Whole_weight    1.652297105  0.579668037    1.978518070     0.98026881
## Height:Viscera_weight   -4.208513310  1.199260909   -4.242209023     5.92071970
```

```
## Whole_weight:Shucked_weight -2.176163504  0.232126202   -2.220641645     0.51644507
## Whole_weight:Viscera_weight  1.233286255  0.238522709    1.204413583     0.59578497
## Viscera_weight:Shell_weight -2.962813439  0.710681162   -2.542727750     1.61698729
##               SSE       MSE       Cp   Pressp      SSE/n   Pressp/n      MSPE  p
## training   85.74129 0.02582569 18.89707 103.0709 0.02565568 0.03084109        NA 22
## validation 20.84122 0.02563496       NA       NA         NA         NA 0.0258421 NA
## [1] "fmodel6"
##                          training_est training_std validation_est validation_std
## (Intercept)                0.91801604   0.10474064      2.4797676     0.01461226
## Shell_weight               2.73645794   0.53782516      0.9973287     0.30579058
## I(Shucked_weight^2)        2.84416775   0.32823342      3.4773049     0.66402043
## Shucked_weight            -4.40922696   0.44627748     -1.8720488     0.21302923
## Diameter                   3.83129980   1.04523046      0.0759256     0.48890521
## I(Diameter^2)             -5.08073925   1.59307428     -5.6911853     3.23147517
## Whole_weight               0.51081529   0.29097400      0.8948299     0.17150127
## SexI                      -0.02876044   0.07067694     -0.0288631     0.02490890
## SexM                      -0.01262559   0.06190402     -0.0155195     0.01599065
## Viscera_weight            -0.51252343   0.30632645     -0.7110516     0.31236506
## I(Length^2)               -4.26320518   0.95001874     -2.3534138     1.93148404
## Height                     2.54276815   0.39713518      0.5624234     0.39416112
## I(Height^2)               -1.54733122   0.28914479      1.5888968     7.16973449
## I(Viscera_weight^2)        1.30225018   0.48529877      2.0567783     1.42070837
## Length                     2.69706716   0.78833212     -0.4026694     0.35756445
## Shucked_weight:SexI        0.73002182   0.12053195      0.7367788     0.28179690
## Shucked_weight:SexM        0.08924868   0.06938470     -0.1453384     0.13534260
## Shucked_weight:Whole_weight -1.93017931   0.21641807     -2.0135655     0.44329234
## Diameter:Whole_weight      2.60442333   0.67680914      3.0092262     1.18863243
## Shell_weight:Diameter     -4.24197910   1.09631621     -4.4145610     2.58150841
## Diameter:SexI             -0.57686956   0.24865327     -0.2134635     0.54176712
## Diameter:SexM             -0.05009933   0.19265371      0.5002647     0.38533194
## Shucked_weight:Length      3.50673420   0.90851009      0.4645443     1.42822788
## Viscera_weight:Height     -4.43513424   1.19267822     -6.2835859     5.93070106
##               SSE       MSE       Cp   Pressp      SSE/n   Pressp/n     MSPE  p
## training   85.57542 0.02579127 16.48054 104.6812 0.02560605 0.03132291       NA 24
## validation 20.64471 0.02545587       NA       NA         NA         NA 2.599098 NA
## [1] "fmodel7"
##                          training_est training_std validation_est validation_std
## (Intercept)                0.91139977   0.10536978      2.48512848     0.01482421
## Shell_weight               2.48705692   0.60936453      1.04839552     0.30676318
## I(Shucked_weight^2)        2.82787764   0.34653818      3.26130654     0.68996461
## I(Shell_weight^2)         -0.32942937   0.34008325     -1.97287593     0.93545707
## Shucked_weight            -4.38961778   0.51284380     -1.83472968     0.21420945
## Diameter                   3.89970578   1.20699689      0.01076845     0.48997615
## I(Diameter^2)             -5.21157223   2.07081003    -11.91614806     5.22562950
## Whole_weight               0.43571780   0.33330574      0.91305779     0.17301216
```

```
## SexI                    -0.03241222   0.07164793    -0.03291017     0.02495236
## SexM                    -0.01496696   0.06224089    -0.01609048     0.01600901
## Viscera_weight           0.04719520   0.63528629    -0.77562144     0.31796075
## I(Length^2)             -4.35206283   1.22738121     0.22556174     3.25181473
## Height                   2.45379174   0.40347866     0.48472412     0.40053993
## I(Height^2)             -1.57632682   0.29576264     5.30602932     8.06589581
## I(Viscera_weight^2)      1.83832930   0.75720385     2.32947799     1.86115924
## Length                   2.74048461   0.90692583    -0.44494461     0.35852619
## Shucked_weight:SexI      0.72308044   0.12256811     0.68312032     0.28291927
## Shucked_weight:SexM      0.08557479   0.06982399    -0.14813988     0.13598691
## Shucked_weight:Whole_weight -1.94849410   0.23459871    -1.78296405     0.48942837
## Diameter:Whole_weight    2.35875398   0.78472406     2.19802283     1.36526175
## Whole_weight:Height      1.20897506   1.22214715    -0.56257124     2.77304774
## Shell_weight:Diameter   -3.18674210   1.59471311     3.00223848     4.19343987
## Diameter:SexI           -0.56231401   0.25292361    -0.11355704     0.54355001
## Diameter:SexM           -0.04119888   0.19389259     0.52099196     0.38651894
## Shucked_weight:Diameter -0.10580098   1.87228473     2.78043910     3.96633855
## Viscera_weight:Height   -9.58508690   5.36221972    -7.03153625    11.46794841
## Shucked_weight:Length    3.63492488   1.33911583    -2.18890584     3.25808246
##                SSE       MSE       Cp   Pressp      SSE/n   Pressp/n     MSPE  p
## training   85.53364 0.02580200 20.86426 104.2825 0.02559355 0.03120361       NA 27
## validation 20.50644 0.02537926       NA       NA         NA         NA 2.625934 NA
## [1] "fmodel8"
##                         training_est training_std validation_est validation_std
## (Intercept)              2.454815850  0.007073816     2.48228469     0.01483286
## SexI                    -0.001026534  0.011186706    -0.02909460     0.02514899
## SexM                    -0.002084350  0.007892842    -0.01578273     0.01614776
## Length                  -0.350069722  0.194657013    -0.59663421     0.45767433
## Diameter                 0.615254584  0.189901467     0.21889923     0.42358098
## Height                   1.253931944  0.204810273     0.55829205     0.39669370
## Whole_weight             0.898287996  0.078180014     0.91086457     0.18457642
## Shucked_weight          -2.130810145  0.103769809    -1.89134960     0.23425537
## Viscera_weight          -0.684308085  0.143092682    -0.65767602     0.31120312
## Shell_weight             0.989790865  0.124256901     1.00236417     0.31188042
## I(Length^2)             -4.835611025  1.015568282    -2.58277536     2.13290858
## I(Diameter^2)           -4.435725484  1.679407970    -5.01935108     3.54395839
## I(Height^2)             -1.248536155  0.269922037    -0.27857954     6.72980265
## I(Shucked_weight^2)      3.337286122  0.406010409     3.79294711     0.97597518
## SexI:Length             -0.468132059  0.208386756    -0.06952039     0.46174597
## SexM:Length              0.019005655  0.166140989     0.46192997     0.35014440
## SexI:Shucked_weight      0.738341489  0.124837623     0.67595150     0.28436632
## SexM:Shucked_weight      0.060468139  0.073262079    -0.16259440     0.14769604
## Length:Shucked_weight    3.509556521  0.944038860     0.35549392     1.59459615
## Diameter:Whole_weight    2.481021040  0.808840070     3.42416225     1.47505652
## Diameter:Shell_weight   -3.728522847  2.120333645    -6.21735093     4.18018760
```

```
## Height:Shucked_weight       -1.962553014  0.565650921    -2.19283285     2.75580178
## Whole_weight:Shucked_weight -2.643479291  0.403719460    -2.49549980     1.01915793
## Whole_weight:Viscera_weight  1.277761663  0.428705563     0.68525776     1.08020875
## Shucked_weight:Shell_weight  1.723959137  0.826278012     1.50250178     2.02634304
## Viscera_weight:Shell_weight -3.517673246  1.435827889    -1.48405126     3.25667921
##                SSE       MSE       Cp   Pressp     SSE/n   Pressp/n       MSPE  p
## training   85.41985 0.02575991 14.46205 112.5313 0.0255595 0.03367184         NA 26
## validation 20.66847 0.02554817       NA       NA        NA         NA 0.02563501 NA
```

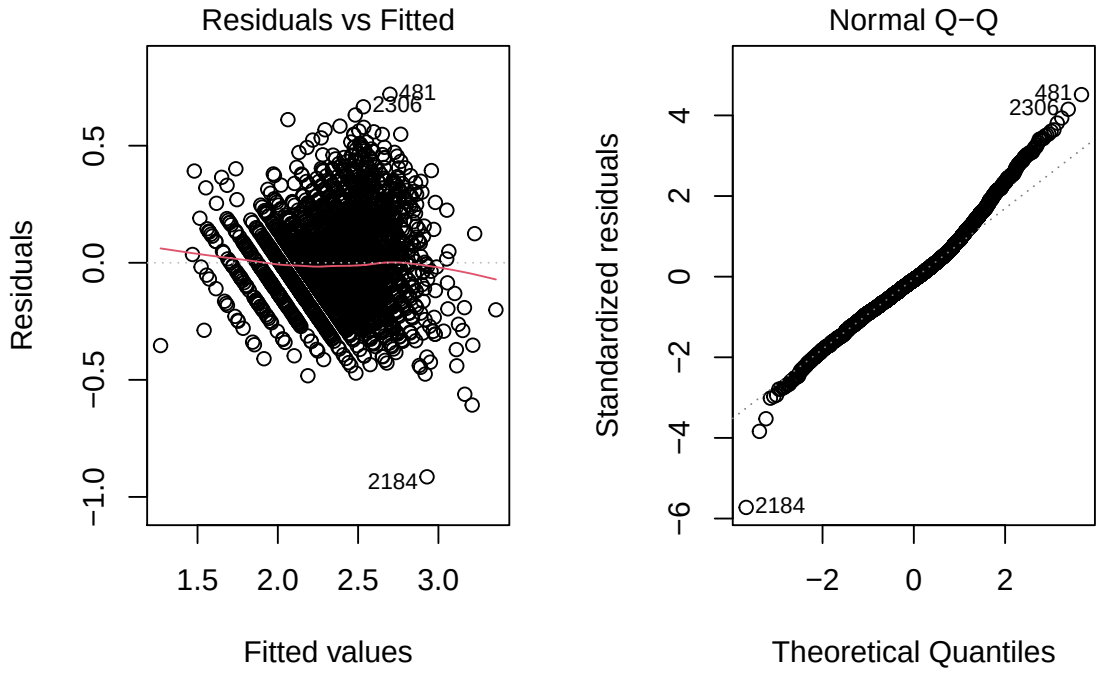
0.1.4 Model Finalization

```
#choose model 5 and regress on whole data set
model_chosen<-lm(formula(fmodel5t),data = abalone.s_c)
summary(model_chosen)
```

```
##
## Call:
## lm(formula = formula(fmodel5t), data = abalone.s_c)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.91414 -0.10252 -0.01258  0.08597  0.71974
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              2.461586   0.006086 404.474  < 2e-16 ***
## SexI                    -0.015597   0.009350  -1.668 0.095373 .
## SexM                    -0.004996   0.006573  -0.760 0.447314
## Length                  -0.505445   0.146828  -3.442 0.000582 ***
## Diameter                 0.534373   0.171494   3.116 0.001846 **
## Height                   1.100699   0.181558   6.063 1.46e-09 ***
## Whole_weight             0.868535   0.067787  12.813  < 2e-16 ***
## Shucked_weight          -1.976635   0.081619 -24.218  < 2e-16 ***
## Viscera_weight          -0.592712   0.126011  -4.704 2.64e-06 ***
## Shell_weight             0.967130   0.105968   9.127  < 2e-16 ***
## I(Length^2)             -3.454574   0.821988  -4.203 2.69e-05 ***
## I(Diameter^2)           -5.629486   1.455775  -3.867 0.000112 ***
## I(Height^2)             -1.346125   0.265300  -5.074 4.07e-07 ***
## I(Shucked_weight^2)      3.274705   0.290078  11.289  < 2e-16 ***
## SexI:Shucked_weight      0.474469   0.051691   9.179  < 2e-16 ***
## SexM:Shucked_weight      0.064557   0.029582   2.182 0.029143 *
## Length:Shucked_weight    2.291850   0.725461   3.159 0.001594 **
## Diameter:Whole_weight    1.914155   0.481242   3.978 7.08e-05 ***
## Height:Viscera_weight   -4.141755   1.123627  -3.686 0.000231 ***
## Whole_weight:Shucked_weight -2.226832   0.205193 -10.852  < 2e-16 ***
```

```
## Whole_weight:Viscera_weight  1.246636   0.217629   5.728 1.09e-08 ***
## Viscera_weight:Shell_weight -2.960237   0.644028  -4.596 4.43e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1606 on 4155 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.653
## F-statistic: 375.3 on 21 and 4155 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(1,2))
plot(model_chosen,which=1)
plot(model_chosen,which=2)
```




```
#Y outliers
stu.res.del <- studres(model_chosen)
a <- 0.05
p <- length(model_chosen$coefficients)
(bonthre <- qt(1-a/(2*n.s),n.s-p-1))
```

```
## [1] 4.383448
```

```
for (i in 1:length(stu.res.del)) {
  if(abs(stu.res.del[i]) > bonthre){
    print(i)
```

```
  }
}
```

```
## [1] 481
## [1] 2184
```

```
#X outliers
hh <- influence(model_chosen)$hat
(hth <- 2*p/n.s)
```

```
## [1] 0.01053388
```

```
(xout <- as.vector(which(hh > hth)))
```

```
##   [1]   47   82   84   86  110  129  130  149  150  158  160  164  165  166  167  168  169  170  171  237  238
##  [22]  239  240  271  278  307  308  335  356  358  359  373  451  466  479  481  511  521  524  525  526  527
##  [43]  548  612  637  647  648  659  661  688  695  697  719  720  721  747  761  763  883  886  889  892  899
##  [64] 1000 1035 1046 1049 1050 1052 1053 1099 1146 1163 1175 1185 1194 1198 1199 1200 1201 1202 1203 1205
##  [85] 1207 1208 1209 1210 1211 1217 1222 1258 1265 1324 1345 1359 1386 1395 1401 1412 1417 1418 1419 1420
## [106] 1426 1427 1428 1429 1430 1525 1528 1529 1531 1575 1638 1692 1700 1738 1748 1749 1751 1755 1757 1758
## [127] 1761 1762 1763 1764 1765 1778 1787 1789 1791 1796 1813 1822 1824 1875 1935 1959 1978 1981 1983 1985
## [148] 1987 1988 2007 2052 2085 2087 2089 2090 2091 2108 2109 2115 2128 2158 2161 2162 2170 2178 2181 2184
## [169] 2202 2209 2210 2211 2251 2266 2275 2335 2344 2369 2372 2381 2382 2395 2398 2408 2435 2454 2529 2535
## [190] 2540 2542 2543 2545 2621 2624 2625 2626 2628 2642 2676 2701 2707 2708 2710 2711 2729 2791 2802 2811
## [211] 2855 2857 2863 2864 2930 2952 2963 2971 2973 2974 2975 2983 2985 2988 2994 3008 3009 3035 3051 3081
## [232] 3083 3087 3129 3141 3142 3149 3150 3152 3162 3189 3217 3302 3319 3320 3328 3338 3339 3389 3396 3397
## [253] 3470 3472 3473 3519 3543 3594 3600 3601 3628 3629 3714 3716 3717 3733 3798 3801 3815 3828 3838 3861
## [274] 3878 3897 3900 3903 3929 3962 3963 3993 3994 3997 4018 4053 4083 4090 4093 4111 4113 4149
```

```
#influential cases
stu.res <- model_chosen$residuals/(1-hh)
d <- stu.res^2*hh/(p*(1-hh))
dth <- 4/(n.s-p)
(dout <- as.vector(which(d > dth)))
```

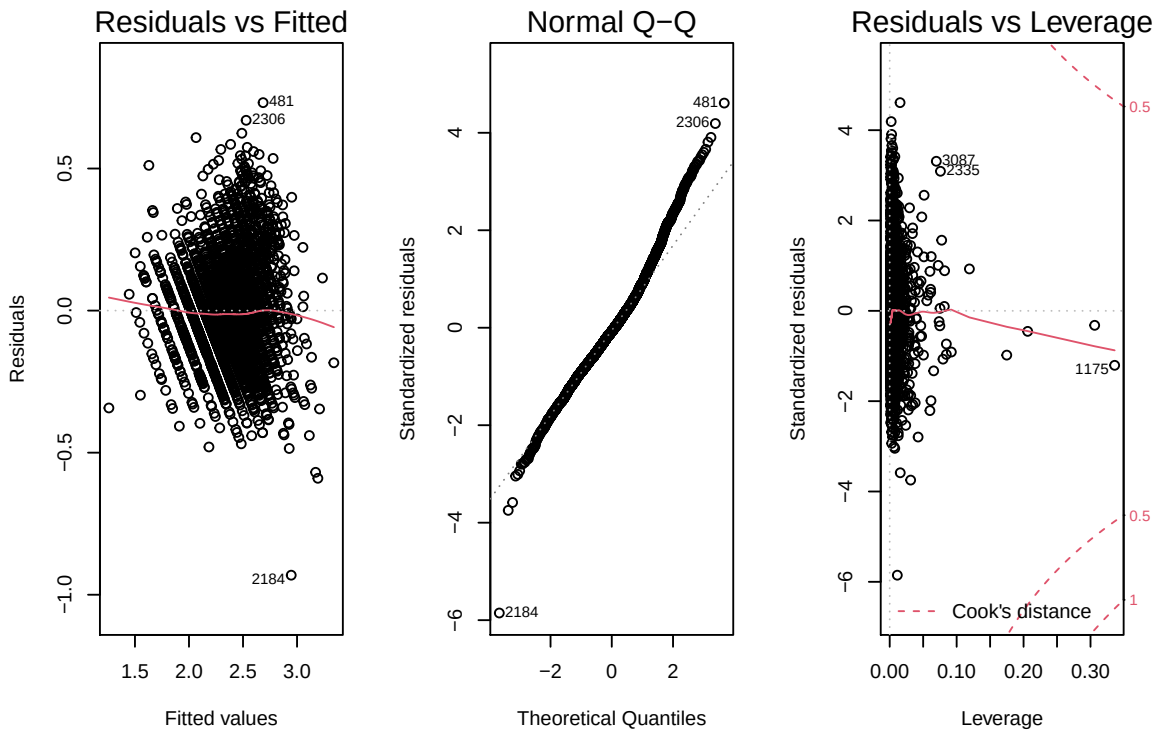
```
## [1]  892 1210 1217 1418 2052 2628 3997
```

```
#drop the influential cases and refit the model
data <- abalone.s_c[-dout,]
model_final<-lm(formula(fmodel5t),data = data)
summary(model_final)
```

```
##
## Call:
## lm(formula = formula(fmodel5t), data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -0.93124 -0.10240 -0.01239  0.08521  0.73190
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  2.459427   0.006169 398.658  < 2e-16 ***
## SexI                        -0.013168   0.009330  -1.411 0.158187
## SexM                        -0.004302   0.006556  -0.656 0.511789
## Length                      -0.519320   0.146706  -3.540 0.000405 ***
## Diameter                     0.585465   0.171851   3.407 0.000664 ***
## Height                       0.991439   0.184192   5.383 7.75e-08 ***
## Whole_weight                 0.948149   0.073669  12.870  < 2e-16 ***
## Shucked_weight              -2.123982   0.091794 -23.138  < 2e-16 ***
## Viscera_weight              -0.584255   0.128152  -4.559 5.29e-06 ***
## Shell_weight                 0.932841   0.114518   8.146 4.94e-16 ***
## I(Length^2)                 -5.659873   0.952654  -5.941 3.06e-09 ***
## I(Diameter^2)               -2.988029   1.632419  -1.830 0.067257 .
## I(Height^2)                  5.232278   3.377147   1.549 0.121381
## I(Shucked_weight^2)          2.923515   0.341106   8.571  < 2e-16 ***
## SexI:Shucked_weight          0.456799   0.051721   8.832  < 2e-16 ***
## SexM:Shucked_weight          0.051559   0.029882   1.725 0.084529 .
## Length:Shucked_weight        4.872624   0.932178   5.227 1.81e-07 ***
## Diameter:Whole_weight        0.740081   0.584037   1.267 0.205161
## Height:Viscera_weight       -7.969077   2.364059  -3.371 0.000756 ***
## Whole_weight:Shucked_weight -2.151461   0.233954  -9.196  < 2e-16 ***
## Whole_weight:Viscera_weight  1.393335   0.246083   5.662 1.60e-08 ***
## Viscera_weight:Shell_weight -2.739758   0.698879  -3.920 8.99e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.16 on 4148 degrees of freedom
## Multiple R-squared:  0.6566, Adjusted R-squared:  0.6548
## F-statistic: 377.6 on 21 and 4148 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(1,3))
plot(model_final,which=1)
plot(model_final,which=2)
plot(model_final,which=5)
```



0.1.5 Additional Methods: Data Preparation

```
#first-order
abalone_ridge.s <- cbind(SexI=as.numeric(abalone.s$Sex=="I"),SexM=as.numeric(abalone.s$Sex=="M"),abalone.s[,2
abalone_ridge.t <- abalone_ridge.s[-index.s,]
abalone_ridge.v <- abalone_ridge.s[index.s,]

#second-order
abalone_ridge.s_second<-abalone_ridge.s[,-10]
for (i in 1:9) {
  name1<-names(abalone_ridge.s)[i]
  for(j in i:9){
    name2<-names(abalone_ridge.s)[j]
    newdf<-data.frame(abalone_ridge.s[,i]*abalone_ridge.s[,j])
    names(newdf)<-paste(name1,name2,sep = ":")
    abalone_ridge.s_second<-cbind(abalone_ridge.s_second,newdf)
    names(newdf)
  }
}
drop <- c("SexI:SexI","SexM:SexM","SexI:SexM")
abalone_ridge.s_second = abalone_ridge.s_second[,!(names(abalone_ridge.s_second) %in% drop)]
abalone_ridge.s_second = cbind(abalone_ridge.s_second,abalone_ridge.s[,10])
```

```
abalone_ridge.t_second <- abalone_ridge.s_second[-index.s,]
abalone_ridge.v_second <- abalone_ridge.s_second[index.s,]
```

0.1.6 Additional Methods: PCR

```
#first order
pca<-prcomp(abalone_ridge.t[,1:9],center=TRUE,scale. = TRUE)
abalone_pca_quant.t <- as.data.frame(as.matrix(abalone_ridge.t[,1:9])%*%pca$rotation)
abalone_pca.t<-cbind(abalone_pca_quant.t,log_age=abalone.t[,9])
model.pca <- lm(log_age ~ .,data = abalone_pca.t[,-c(8,9)])
summary(model.pca)
```

```
##
## Call:
## lm(formula = log_age ~ ., data = abalone_pca.t[, -c(8, 9)])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.91855 -0.11379 -0.01997  0.09430  0.73983
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.66406    0.02580  64.503  < 2e-16 ***
## PC1         -0.96107    0.03728 -25.783  < 2e-16 ***
## PC2          0.20304    0.01144  17.743  < 2e-16 ***
## PC3         -0.24125    0.01713 -14.086  < 2e-16 ***
## PC4          1.06251    0.11685   9.093  < 2e-16 ***
## PC5          0.67633    0.07150   9.459  < 2e-16 ***
## PC6         -1.40561    0.06202 -22.664  < 2e-16 ***
## PC7          0.27576    0.09014   3.059  0.00224 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1763 on 3334 degrees of freedom
## Multiple R-squared:  0.5897, Adjusted R-squared:  0.5888
## F-statistic: 684.5 on 7 and 3334 DF,  p-value: < 2.2e-16
```

```
tranmatrix <- as.matrix(summary(model.pca)$coefficient[-1,1])
beta.pca <- t(tranmatrix)%*%t(pca$rotation[,c(1:7)])
beta.pca
```

```
##             SexI       SexM    Length  Diameter    Height Whole_weight Shucked_weight Viscera_weight
## [1,] -0.07398442 0.007936825 0.6520073 0.7762378 0.9551872   0.02007008     -0.9718863     -0.1047208
##      Shell_weight
## [1,]     1.326525
```

```r
abalone_pca_quant.v <- as.data.frame(as.matrix(abalone_ridge.v[,1:9])%*%pca$rotation)
mspe_pcr_first<-mean((predict(model.pca,abalone_pca_quant.v )-abalone_ridge.v[,10])^2)


#second order
pca_second<-prcomp(abalone_ridge.t_second[,1:51],center=TRUE,scale. = TRUE)
abalone_pca_quant.t_second <- as.data.frame(as.matrix(abalone_ridge.t_second[,1:51])%*%pca_second$rotation)
abalone_pca.t_second<-cbind(abalone_pca_quant.t_second,log_age=abalone.t$log_age)
model.pca_second <- lm(abalone.t$log_age ~., data=abalone_pca.t_second[,-c(23:51)])
summary(model.pca_second)
```

```
##
## Call:
## lm(formula = abalone.t$log_age ~ ., data = abalone_pca.t_second[,
##     -c(23:51)])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.84946 -0.10744 -0.01462  0.08959  0.71153
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.64257    0.03469  47.355  < 2e-16 ***
## PC1          0.18523    0.06064   3.054 0.002273 **
## PC2          0.08298    0.08989   0.923 0.356001
## PC3          0.20377    0.05761   3.537 0.000410 ***
## PC4         -1.05878    0.10406 -10.175  < 2e-16 ***
## PC5         -1.06617    0.10756  -9.913  < 2e-16 ***
## PC6         -0.90682    0.04175 -21.719  < 2e-16 ***
## PC7          0.21063    0.05200   4.051 5.22e-05 ***
## PC8          0.30659    0.08646   3.546 0.000397 ***
## PC9          1.42990    0.20973   6.818 1.09e-11 ***
## PC10        -0.13248    0.07107  -1.864 0.062398 .
## PC11         2.11864    0.15252  13.891  < 2e-16 ***
## PC12        -0.31886    0.07697  -4.143 3.52e-05 ***
## PC13         0.50366    0.13926   3.617 0.000303 ***
## PC14        -0.71568    0.15696  -4.560 5.31e-06 ***
## PC15        -1.48588    0.16809  -8.840  < 2e-16 ***
## PC16         0.26492    0.23026   1.151 0.250003
## PC17        -0.80186    0.14204  -5.645 1.79e-08 ***
## PC18         1.31239    0.29551   4.441 9.24e-06 ***
## PC19        -0.46141    0.20014  -2.305 0.021205 *
## PC20         3.04806    0.23021  13.240  < 2e-16 ***
## PC21        -0.45344    0.36138  -1.255 0.209653
## PC22         0.01453    0.31600   0.046 0.963331
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1644 on 3319 degrees of freedom
## Multiple R-squared:  0.645,  Adjusted R-squared:  0.6426
## F-statistic:   274 on 22 and 3319 DF,  p-value: < 2.2e-16
```

```
tranmatrix_second <- as.matrix(summary(model.pca_second)$coefficient[-1,1])
beta.pca_second <- t(tranmatrix_second)%*%t(pca_second$rotation[,c(1:22)])
beta.pca_second
```

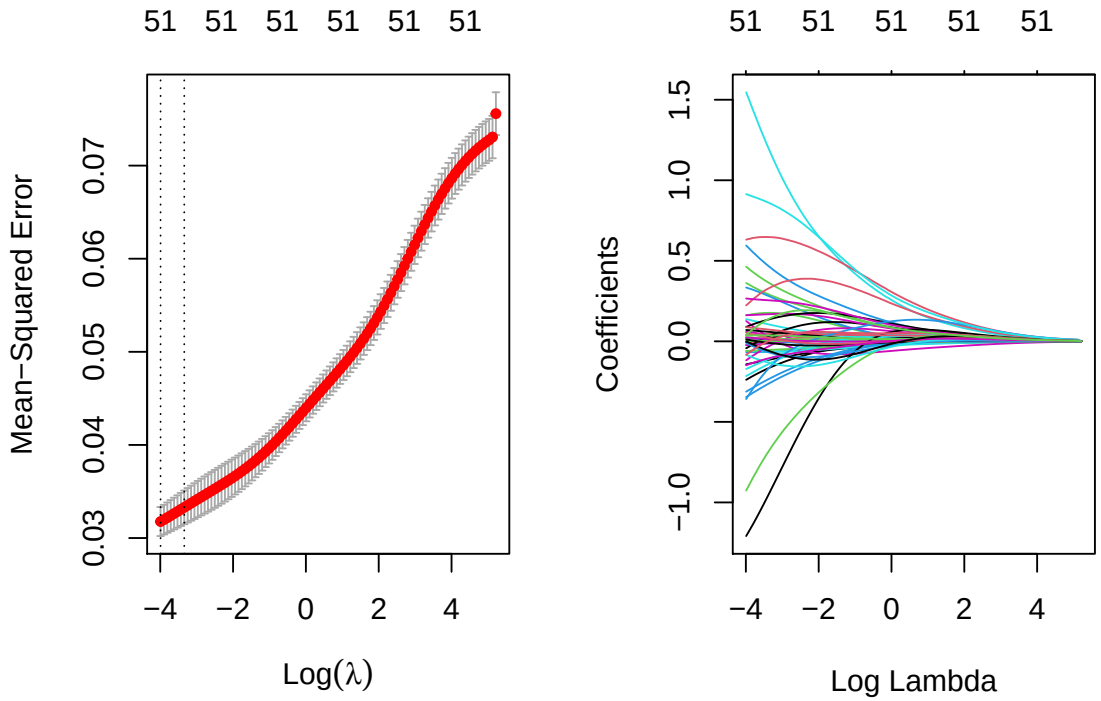
```
##           SexI       SexM    Length  Diameter   Height Whole_weight Shucked_weight Viscera_weight
## [1,] -0.2125719 -0.09702247 0.4554224 0.8307307 2.370532    0.4521742     -0.998566     -0.2393665
##      Shell_weight  SexI:Length SexI:Diameter SexI:Height SexI:Whole_weight SexI:Shucked_weight
## [1,]    2.006034 0.0008510313    0.00206615   0.5943287        0.08008002          0.2305946
##      SexI:Viscera_weight SexI:Shell_weight SexM:Length SexM:Diameter SexM:Height SexM:Whole_weight
## [1,]          0.3724936       -0.4943779   0.0804521     0.1206608  0.02573324      -0.001994674
##      SexM:Shucked_weight SexM:Viscera_weight SexM:Shell_weight Length:Length Length:Diameter Length:Heigh
## [1,]        -0.02150741          0.04888532       -0.01881517    -0.3864095     -0.2078835     0.4798621
##      Length:Whole_weight Length:Shucked_weight Length:Viscera_weight Length:Shell_weight Diameter:Diameter
## [1,]         -0.1569698          -0.9620117          -0.5742596          0.6149706       -0.02445224
##      Diameter:Height Diameter:Whole_weight Diameter:Shucked_weight Diameter:Viscera_weight
## [1,]       0.5786341          -0.1140828           -0.9145076           -0.5200575
##      Diameter:Shell_weight Height:Height Height:Whole_weight Height:Shucked_weight Height:Viscera_weight
## [1,]           0.6480579     -1.477068         -0.5789817          -1.557491          -0.9180177
##      Height:Shell_weight Whole_weight:Whole_weight Whole_weight:Shucked_weight Whole_weight:Viscera_weigl
## [1,]           0.4115238                0.1279334                 0.2825292                 0.2083684
##      Whole_weight:Shell_weight Shucked_weight:Shucked_weight Shucked_weight:Viscera_weight
## [1,]              -0.4553167                   0.6013727                  0.3414187
##      Shucked_weight:Shell_weight Viscera_weight:Viscera_weight Viscera_weight:Shell_weight
## [1,]               -0.3794393                  0.2167408               -0.2979337
##      Shell_weight:Shell_weight
## [1,]               -1.175155
```

```
abalone_pca_quant.v_second <- as.data.frame(as.matrix(abalone_ridge.v_second[,1:51])%*%pca_second$rotation)
mspe_pcr_second<-mean((predict(model.pca_second,abalone_pca_quant.v_second )-abalone_ridge.v[,10])^2)
```

0.1.7 Additional Methods: Ridge

```
par(mfrow=c(1,2))
XS <- as.matrix(abalone_ridge.t_second[,-52])
X <- as.matrix(abalone_ridge.t[,1:9])
Y <- abalone_ridge.t[,10]
modelc1 <- cv.glmnet(XS,Y,alpha = 0,family = "gaussian",type.measure="mse")
plot(modelc1)
model1 <- glmnet(XS,Y,alpha = 0,family = "gaussian")
```

```
plot(model1,xvar="lambda",label=TRUE)
```




```
#1se
model_ridge <- glmnet(XS,Y,alpha = 0,family = "gaussian",lambda = modelc1$lambda.1se)
betahat_ridge <- coef(model_ridge)
Yhat_ridge <- predict(model_ridge,as.matrix(abalone_ridge.v_second[,-52]))
Y.v <- as.matrix(abalone_ridge.v[,10])
e_ridge <- Y.v-Yhat_ridge
mspe_ridge <- mean(e_ridge^2)
#min
model_ridge_min <- glmnet(XS,Y,alpha = 0,family = "gaussian",lambda = modelc1$lambda.min)
betahat_min <- coef(model_ridge_min)
Yhat_ridge_min <- predict(model_ridge_min,as.matrix(abalone_ridge.v_second[,-52]))
e_ridge_min <- Y.v-Yhat_ridge_min
mspe_ridge_min <- mean(e_ridge_min^2)
mspe_ls <- mean((predict.lm(model_chosen,abalone.v_c)-abalone.v_c$log_age)^2)
```

0.1.8 Additional Methods: Lasso

```
#test whether variable is significant individually
tvalue <- {}
i=2
```

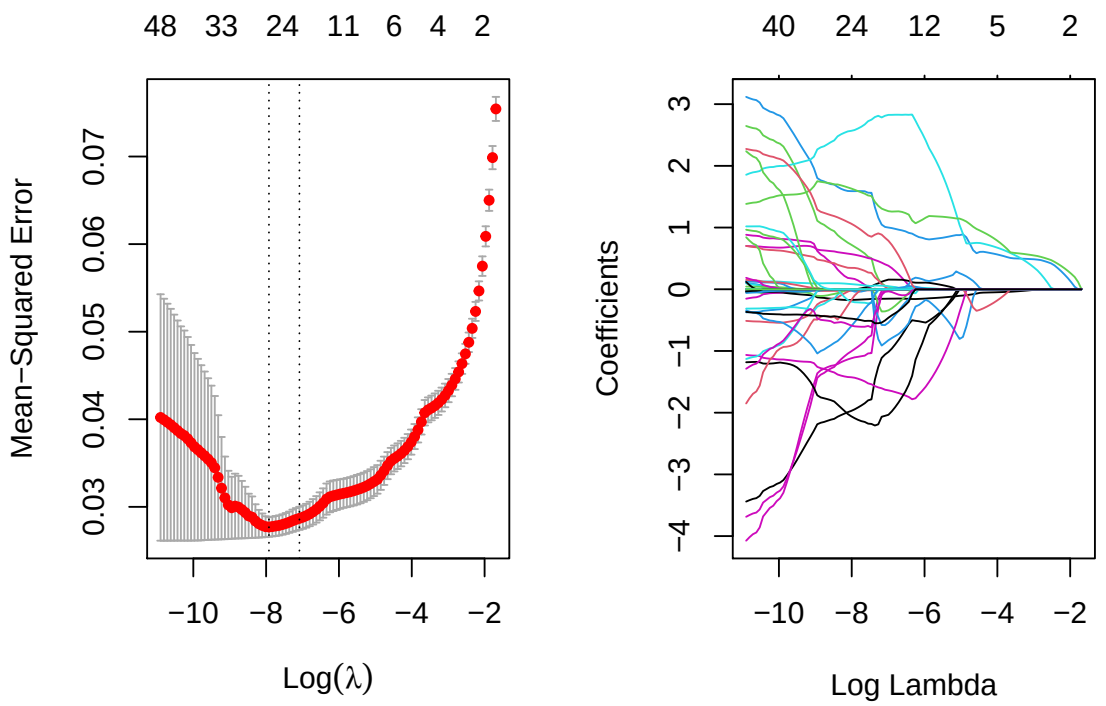
```r
par(mfrow=c(1,1))
for (i in 1:51) {
  tmodel <- lm(abalone.t$log_age~abalone_ridge.t_second[,i])
  stmodel <- summary(tmodel)
  stmodel$coefficients[2,4]
  tvalue[i] <- stmodel$coefficients[2,4]
}
tvalue
```

```
##  [1] 8.747430e-189  5.636428e-31  0.000000e+00  0.000000e+00  0.000000e+00 9.881313e-324 4.201259e-201
##  [8] 1.223692e-281  0.000000e+00  1.798208e-88  4.208013e-83  6.434002e-74  1.085719e-13  2.140858e-17
## [15]  6.916627e-14  2.318065e-12  3.765104e-47  7.875741e-49  6.027775e-53  6.098363e-64  1.634045e-48
## [22]  5.183090e-59  2.039384e-79  0.000000e+00  0.000000e+00  0.000000e+00 4.647453e-274 3.994588e-185
## [29] 5.089224e-245  0.000000e+00  0.000000e+00  0.000000e+00 4.543004e-279 5.155076e-190 9.588866e-251
## [36]  0.000000e+00  6.680806e-56 4.125332e-277 5.497270e-189 3.740469e-250  0.000000e+00 2.825978e-196
## [43] 1.096482e-144 3.741488e-183 1.004421e-248 8.951919e-102 2.655051e-136 7.235020e-193 7.666064e-160
## [50] 1.536177e-236 1.374228e-276
```

```r
which(tvalue >= 0.05)
```

```
## integer(0)
```

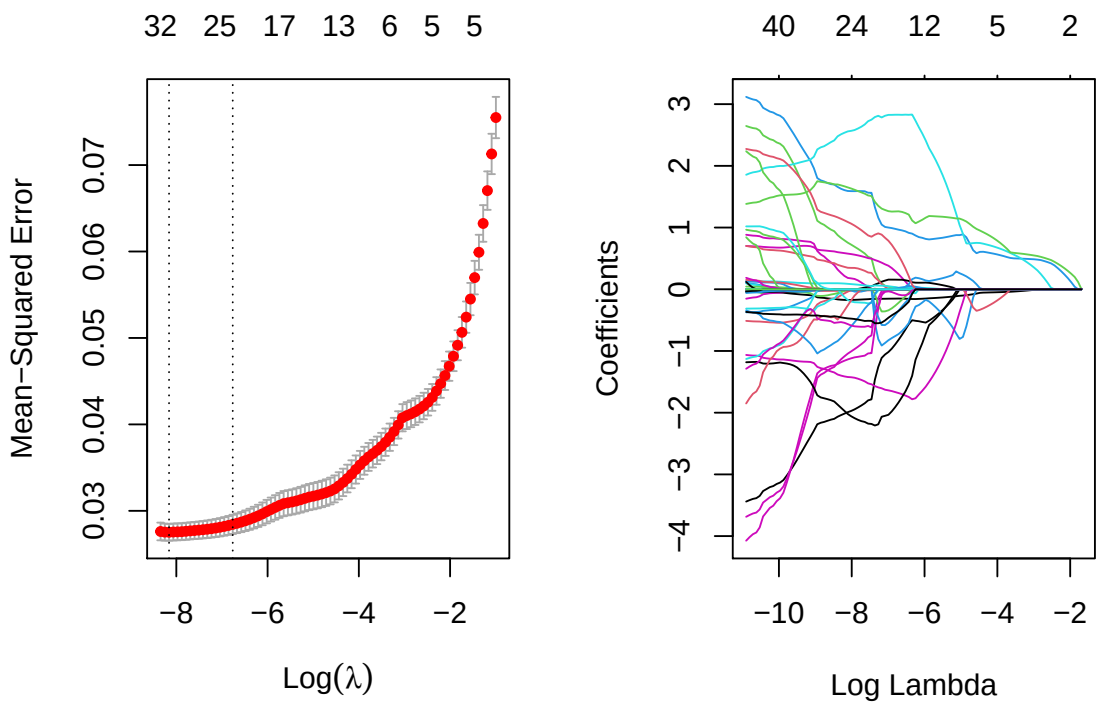
```r
#lasso
par(mfrow=c(1,2))
modelc2 <- cv.glmnet(XS,Y,alpha = 1,family = "gaussian",type.measure="mse")
plot(modelc2)
model2 <- glmnet(XS,Y,alpha = 1,family = "gaussian")
plot(model2,xvar="lambda",label=TRUE)
```




```
#1se
model_lasso.S <- glmnet(XS,Y,alpha = 1,family = "gaussian",lambda = modelc2$lambda.1se)
betahat_lasso.S <- coef(model_lasso.S)
Yhat_lasso.S <- predict(model_lasso.S,as.matrix(abalone_ridge.v_second[,-52]))
e_lasso.S <- Y.v-Yhat_lasso.S
mspe_lasso.S <- mean(e_lasso.S^2)
#min
model_lasso_min.S <- glmnet(XS,Y,alpha = 1,family = "gaussian",lambda = modelc2$lambda.min)
betahat_lasso_min.S <- coef(model_lasso_min.S)
Yhat_lasso_min.S <- predict(model_lasso_min.S,as.matrix(abalone_ridge.v_second[,-52]))
e_lasso_min <- Y.v-Yhat_lasso_min.S
mspe_lasso_min <- mean(e_lasso_min^2)
```

0.1.9 Additional Methods: Elastic Net Regularization

```
par(mfrow=c(1,2))
modelc3 <- cv.glmnet(XS,Y,alpha = 0.5,family = "gaussian",type.measure="mse")
plot(modelc3)
model3 <- glmnet(XS,Y,alpha = 0.5,family = "gaussian")
plot(model2,xvar="lambda",label=TRUE)
```




```
#1se
model_en.S <- glmnet(XS,Y,alpha = 0.5,family = "gaussian",lambda = modelc3$lambda.1se)
betahat_en.S <- coef(model_en.S)
Yhat_en.S <- predict(model_en.S,as.matrix(abalone_ridge.v_second[,-52]))
e_en.S <- Y.v-Yhat_en.S
mspe_en.S <- mean(e_en.S^2)
#min
model_en_min.S <- glmnet(XS,Y,alpha = 1,family = "gaussian",lambda = modelc3$lambda.min)
betahat_en_min.S <- coef(model_en_min.S)
Yhat_en_min.S <- predict(model_en_min.S,as.matrix(abalone_ridge.v_second[,-52]))
e_en_min <- Y.v-Yhat_en_min.S
mspe_en_min <- mean(e_en_min^2)
```

0.1.10 Conclusion and Discussion

```
cbind(mspe_ls,mspe_pcr_first,mspe_pcr_second,mspe_ridge_min,mspe_lasso_min,mspe_en_min)
```

```
##         mspe_ls mspe_pcr_first mspe_pcr_second mspe_ridge_min mspe_lasso_min mspe_en_min
## [1,] 0.02541876     0.03059731      0.02629352     0.02909562     0.02599769  0.02590794
```