HOMEWORK 2 - Exercises in Faraway Introduction

STOR 590, FALL 2020

Rui Li

8/27/2020

Instructions

Question 1: Faraway, page 24, exercise 2 ("rock" dataset)

Question 2: Faraway, page 24, exercise 5 ("prostate" dataset)

In each case, I'd like you to conduct an analysis, following the six bullet points listed in the question. Lengthy answers are not required, but you should be sure to address each of these bullet points in your answer.

Due time and date: 1:00 pm Friday, August 28

Exercises

Since this is a review chapter, it is best to consult the recommended background texts for specific questions on linear models. However, it is worthwhile gaining some practice using R on some real data. Your data analysis should consist of:

1. An initial data analysis that explores the numerical and graphical characteristics of the data.

2. Variable selection to choose the best model.

3. An exploration of transformations to improve the fit of the model.

4. Diagnostics to check the assumptions of your model.

5. Some predictions of future observations for interesting values of the predictors.

6. An interpretation of the meaning of the model with respect to the particular area of application.

There is always some freedom in deciding which methods to use, in what order to apply them, and how to interpret the results. So there may not be one clear right answer and good analysts may come up with different models.

Exercise 2 The *rock* data - use *perm* as the response.

Load *rock* data

```
library(faraway)
```

```
## Warning: package 'faraway' was built under R version 3.6.3
```

```
head(rock)
```

```
##   area    peri     shape perm
## 1 4990 2791.90 0.0903296  6.3
## 2 7002 3892.60 0.1486220  6.3
## 3 7558 3930.66 0.1833120  6.3
## 4 7352 3869.32 0.1170630  6.3
## 5 7943 3948.54 0.1224170 17.1
## 6 7979 4010.15 0.1670450 17.1
```

```
str(rock)
```

```
## 'data.frame':    48 obs. of  4 variables:
##  $ area : int  4990 7002 7558 7352 7943 7979 9333 8209 8393 6425 ...
##  $ peri : num  2792 3893 3931 3869 3949 ...
##  $ shape: num  0.0903 0.1486 0.1833 0.1171 0.1224 ...
##  $ perm : num  6.3 6.3 6.3 6.3 17.1 17.1 17.1 17.1 119 119 ...
```

According to the information above, we know that *rock* is a data frame with 48 observations and 4 numeric columns.

area: Area of pores space, in pixels out of 256 by 256.

peri: Perimeter in pixels.

shape: Perimeter/sqrt(area).

perm: Permeability in milli-Darcies.

Initial Data Analysis

```
#Summarize the dataset
summary(rock)
```

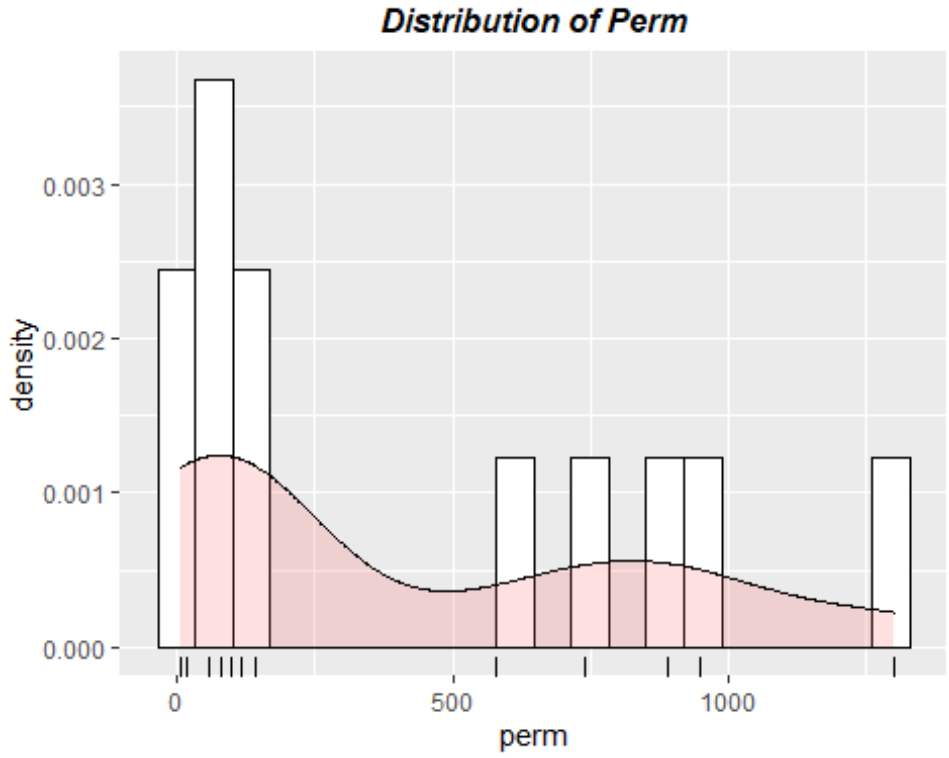
```
##       area            peri           shape             perm       
##  Min.   : 1016   Min.   : 308.6   Min.   :0.09033   Min.   :   6.30  
##  1st Qu.: 5305   1st Qu.:1414.9   1st Qu.:0.16226   1st Qu.:  76.45  
##  Median : 7487   Median :2536.2   Median :0.19886   Median : 130.50  
##  Mean   : 7188   Mean   :2682.2   Mean   :0.21811   Mean   : 415.45  
##  3rd Qu.: 8870   3rd Qu.:3989.5   3rd Qu.:0.26267   3rd Qu.: 777.50  
##  Max.   :12212   Max.   :4864.2   Max.   :0.46413   Max.   :1300.00
```

We have four numerical variables, and the six summary statistics show us the general distribution of the variables. I am going to analyze in depth with each variable, and the *perm* response.

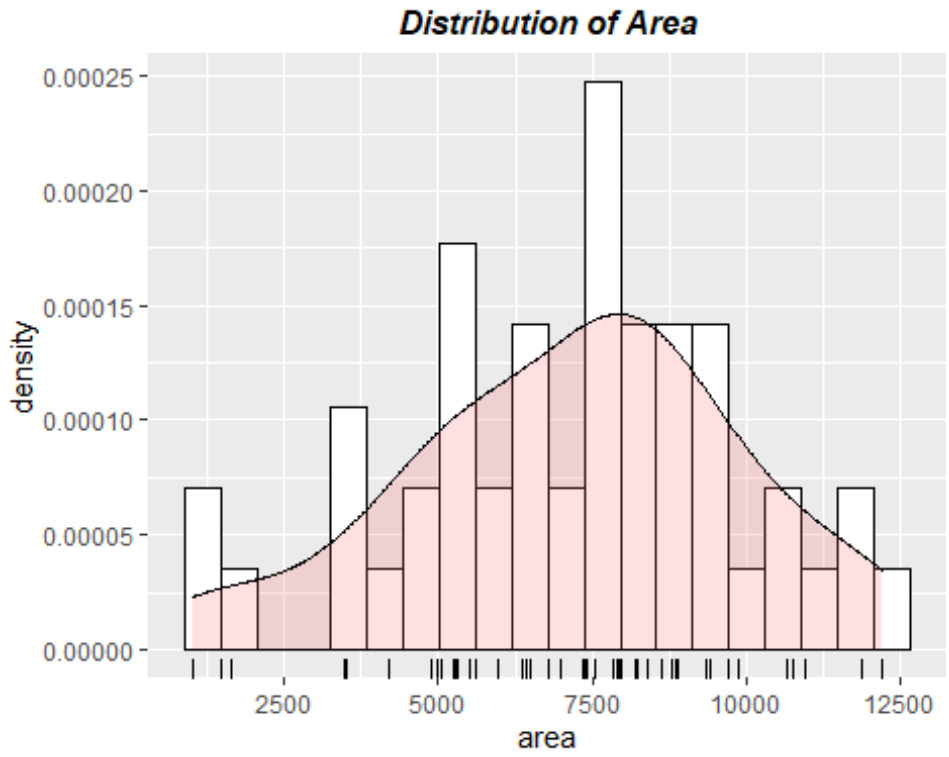
```
#Distribution of each variable
#Perm
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

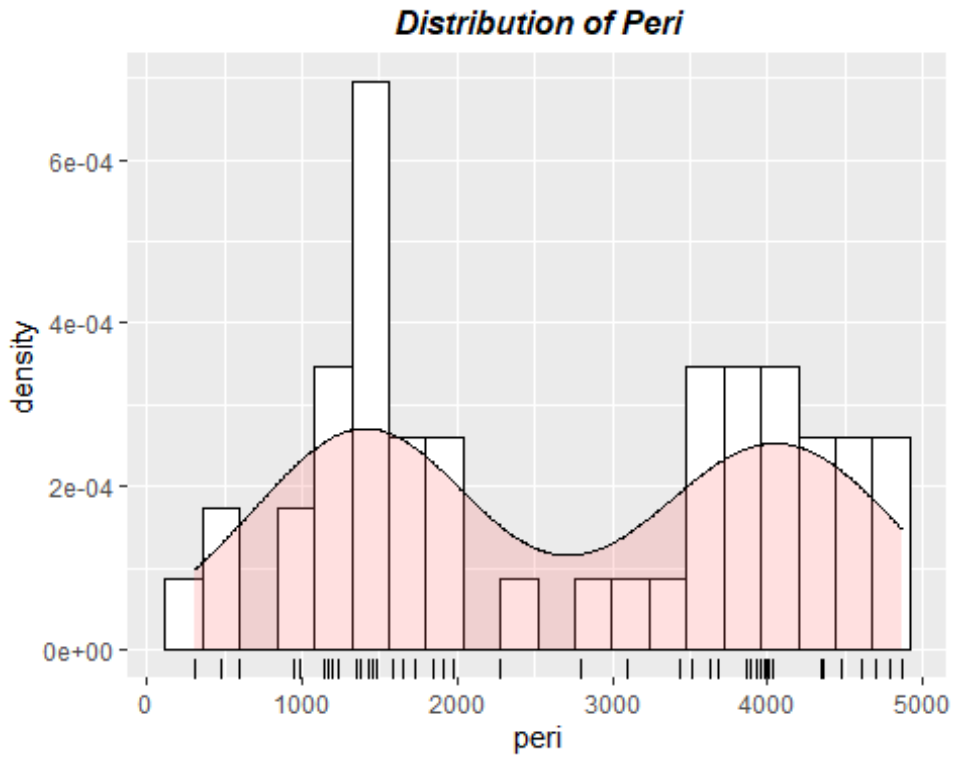
```
ggplot(rock, aes(x=perm)) +
  geom_histogram(aes(y=..density..), bins = 20,fill = "white", col = "black")
 +
  geom_density(alpha=.2, fill="#FF6666") +
  geom_rug() +
  labs(title = 'Distribution of Perm') +
  theme(plot.title = element_text(hjust = 0.5, size=12, face="bold.italic"))
```



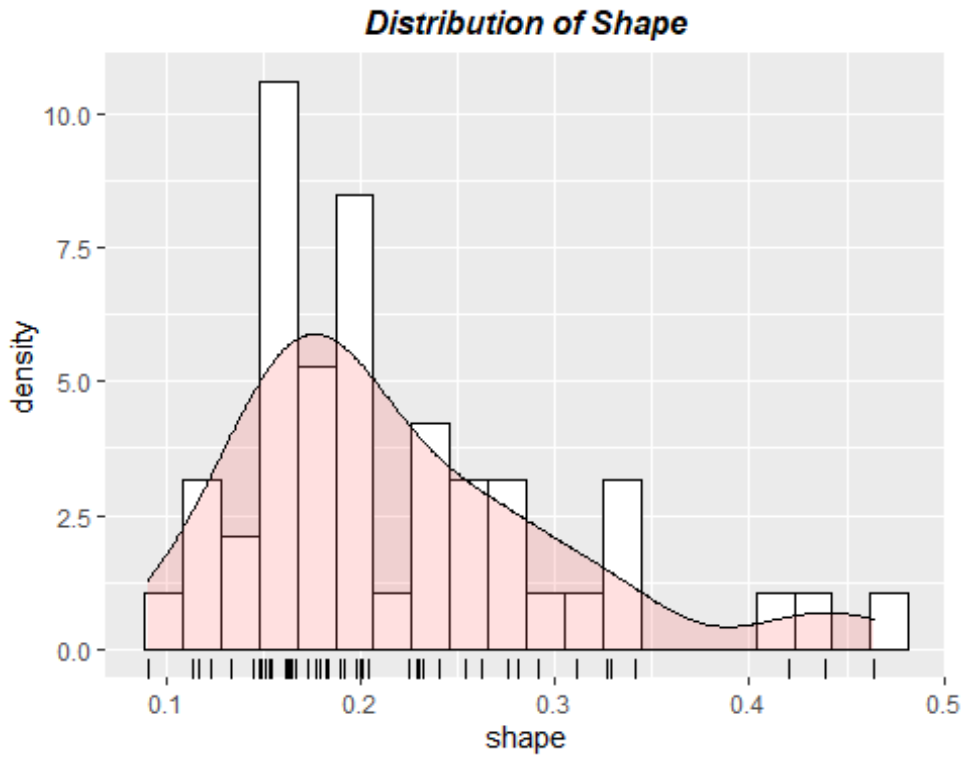

```
#Area
ggplot(rock, aes(x=area)) +
  geom_histogram(aes(y=..density..), bins = 20,fill = "white", col = "black")
 +
  geom_density(alpha=.2, fill="#FF6666") +
  geom_rug() +
  labs(title = 'Distribution of Area') +
  theme(plot.title = element_text(hjust = 0.5, size=12, face="bold.italic"))
```




```
#Peri
ggplot(rock, aes(x=peri)) +
  geom_histogram(aes(y=..density..), bins = 20,fill = "white", col = "black")
 +
  geom_density(alpha=.2, fill="#FF6666") +
  geom_rug() +
  labs(title = 'Distribution of Peri') +
  theme(plot.title = element_text(hjust = 0.5, size=12, face="bold.italic"))
```




```r
#Shape
library(ggplot2)
ggplot(rock, aes(x=shape)) +
  geom_histogram(aes(y=..density..), bins = 20,fill = "white", col = "black")
 +
  geom_density(alpha=.2, fill="#FF6666") +
  geom_rug() +
  labs(title = 'Distribution of Shape') +
  theme(plot.title = element_text(hjust = 0.5, size=12, face="bold.italic"))
```



From the distribution plots above, we can see that there are some outliers in bothe *perm* response and *Shape* variable. We should be careful about them in the following analysis.

```
#Correlation between variables
cor(rock)
```

```
##              area       peri      shape       perm
## area   1.0000000  0.8225064 -0.1821611 -0.3966370
## peri   0.8225064  1.0000000 -0.4331255 -0.7387158
## shape -0.1821611 -0.4331255  1.0000000  0.5567208
## perm  -0.3966370 -0.7387158  0.5567208  1.0000000
```

```
#Pair Plots
library(ggplot2)
library(GGally)
```

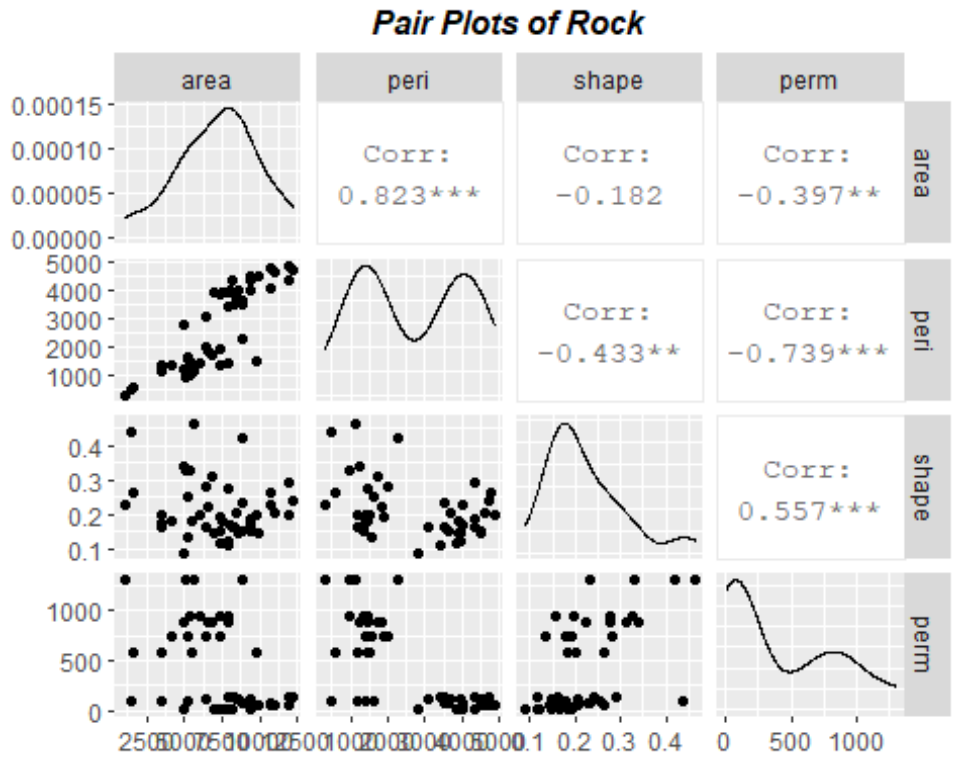
```
## Warning: package 'GGally' was built under R version 3.6.3
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
##
## Attaching package: 'GGally'
```

```
## The following object is masked from 'package:faraway':
##
##     happy
```

```
ggpairs(rock) +
  ggtitle("Pair Plots of Rock") +
  theme(plot.title = element_text(hjust = 0.5, size=12, face="bold.italic"))
```



From the results above, we can see that *area*, *peri*, and *shape* has some correlations with *perm*. We can continue to define a linear model, and find deeper relationship.

Variable Selection

```
#Build up full model
full.lm = lm(formula = perm ~ ., data = rock)
print(summary(full.lm))

##
## Call:
## lm(formula = perm ~ ., data = rock)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -750.26  -59.57   10.66  100.25  620.91
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 485.61797  158.40826   3.066 0.003705 **
## area          0.09133    0.02499   3.654 0.000684 ***
## peri         -0.34402    0.05111  -6.731 2.84e-08 ***
```

```
## shape       899.06926  506.95098   1.773 0.083070 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 246 on 44 degrees of freedom
## Multiple R-squared:  0.7044, Adjusted R-squared:  0.6843
## F-statistic: 34.95 on 3 and 44 DF,  p-value: 1.033e-11
```

```
#Apply backward selection model
full.backward = step(full.lm, direction = "backward")
```

```
## Start:  AIC=532.34
## perm ~ area + peri + shape
##
##         Df Sum of Sq     RSS    AIC
## <none>               2663023 532.34
## - shape  1    190360 2853383 533.66
## - area   1    808191 3471213 543.06
## - peri   1   2741707 5404730 564.32
```

```
print(summary(full.backward))
```

```
##
## Call:
## lm(formula = perm ~ area + peri + shape, data = rock)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -750.26  -59.57   10.66  100.25  620.91
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 485.61797  158.40826   3.066 0.003705 **
## area          0.09133    0.02499   3.654 0.000684 ***
## peri         -0.34402    0.05111  -6.731 2.84e-08 ***
## shape       899.06926  506.95098   1.773 0.083070 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 246 on 44 degrees of freedom
## Multiple R-squared:  0.7044, Adjusted R-squared:  0.6843
## F-statistic: 34.95 on 3 and 44 DF,  p-value: 1.033e-11
```

```
#Apply forward selection model
full.forward <- step(lm(perm ~ 1, data=rock), list(upper=full.lm), direction=
'forward')
```

```
## Start:  AIC=584.84
## perm ~ 1
##
##         Df Sum of Sq     RSS    AIC
```

```
## + peri   1   4916322 4092864 548.97
## + shape  1   2792290 6216896 569.04
## + area   1   1417333 7591852 578.63
## <none>               9009186 584.84
##
## Step:  AIC=548.97
## perm ~ peri
##
##         Df Sum of Sq     RSS    AIC
## + area   1   1239481 2853383 533.66
## + shape  1    621651 3471213 543.06
## <none>               4092864 548.97
##
## Step:  AIC=533.66
## perm ~ peri + area
##
##         Df Sum of Sq     RSS    AIC
## + shape  1    190360 2663023 532.34
## <none>               2853383 533.66
##
## Step:  AIC=532.34
## perm ~ peri + area + shape
```

```
print(summary(full.forward))
```

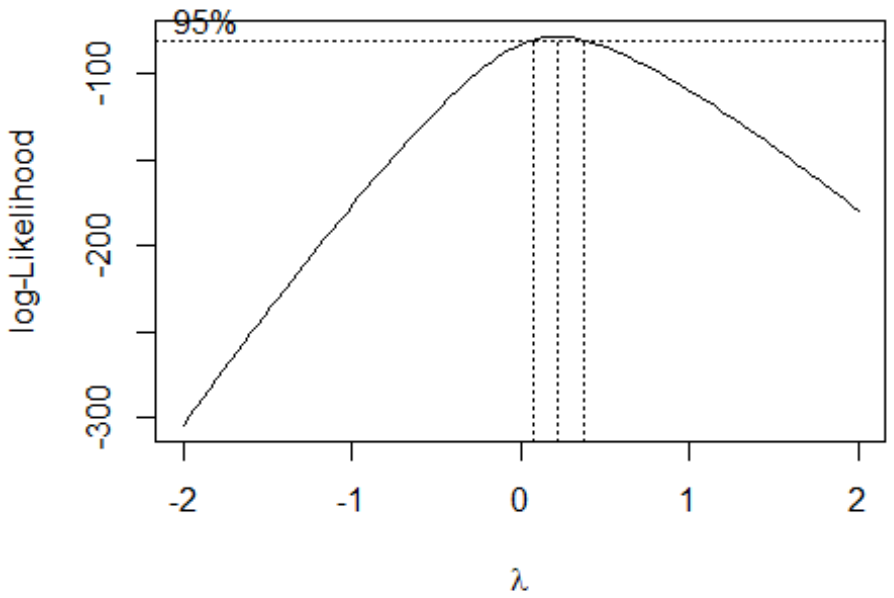
```
##
## Call:
## lm(formula = perm ~ peri + area + shape, data = rock)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -750.26  -59.57   10.66  100.25  620.91
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 485.61797  158.40826   3.066 0.003705 **
## peri         -0.34402    0.05111  -6.731 2.84e-08 ***
## area          0.09133    0.02499   3.654 0.000684 ***
## shape       899.06926  506.95098   1.773 0.083070 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 246 on 44 degrees of freedom
## Multiple R-squared:  0.7044, Adjusted R-squared:  0.6843
## F-statistic: 34.95 on 3 and 44 DF,  p-value: 1.033e-11
```

I apply both backward and forward selection to the model, and have the same optimal model. According to the report, the optimal model is exactly the same as the full model, *perm ~ area + peri + shape*. However, the significant importance of variable *shape* is mild. We may think of improving the model by transforming some of the variables.

Exploration of Transformations

Transform the response perm

```
#Box-Cox Transformation of the response
library(MASS)
full.bc = boxcox(perm ~ area + peri + shape, data=rock)
```

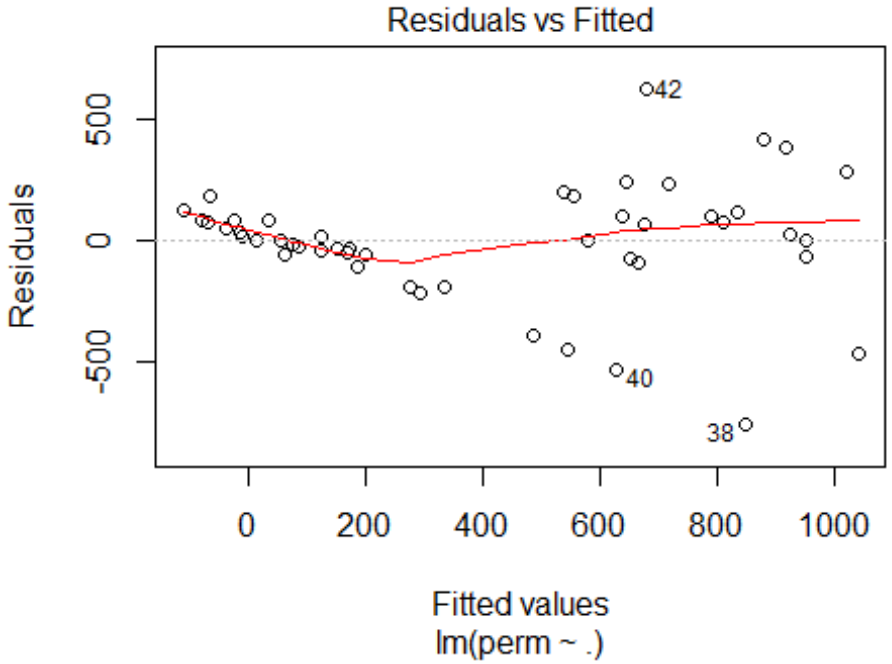


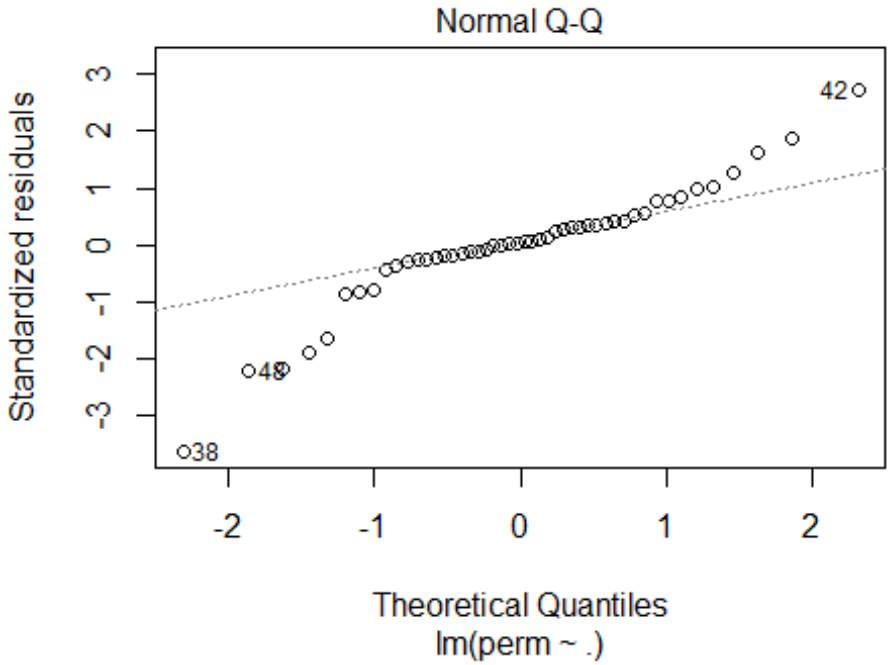

```
#Get the lamda of maximum log-Likelihood
lamda.max = full.bc$x[full.bc$y==max(full.bc$y)]

#Set up new Model
bc.lm = lm(perm^lamda.max ~ area + peri + shape, data=rock)
summary(bc.lm)

##
## Call:
## lm(formula = perm^lamda.max ~ area + peri + shape, data = rock)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.30235 -0.32020  0.09305  0.32367  1.06157
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.577e+00  3.451e-01  10.365 2.18e-13 ***
## area         3.074e-04  5.445e-05   5.645 1.12e-06 ***
```

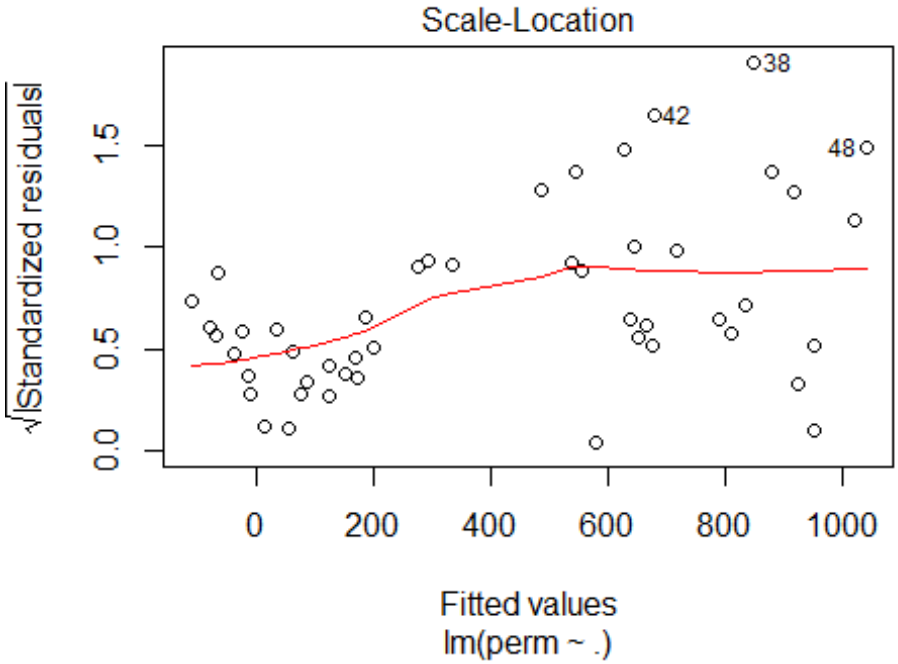
```
## peri        -1.025e-03  1.113e-04  -9.207 8.02e-12 ***
## shape        1.242e+00  1.104e+00   1.125    0.267
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5359 on 44 degrees of freedom
## Multiple R-squared:  0.7802, Adjusted R-squared:  0.7652
## F-statistic: 52.06 on 3 and 44 DF,  p-value: 1.601e-14
```

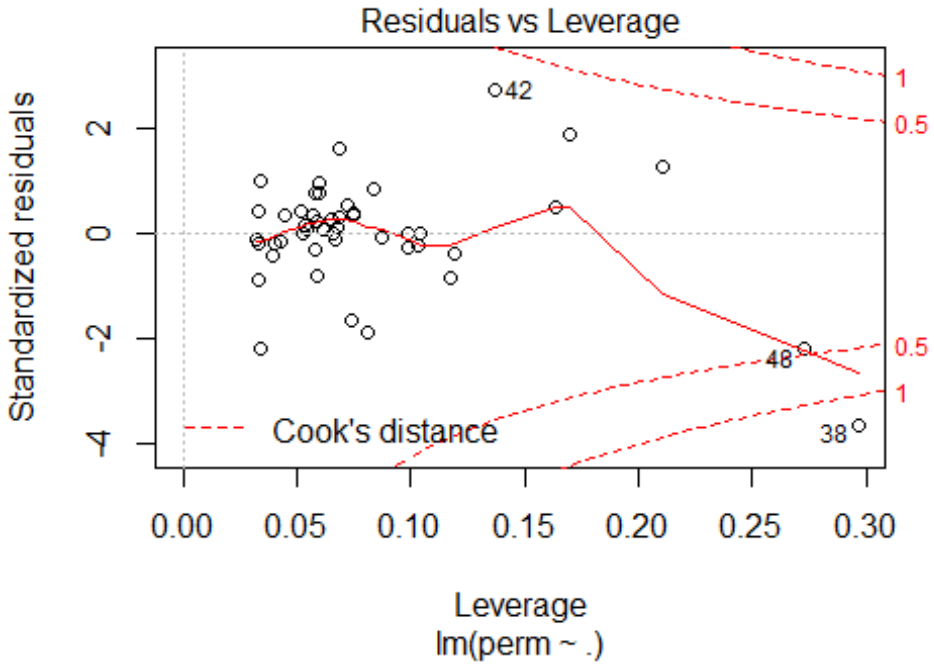
Diagnostics of models before and after Box-Cox Transformation

```
#Diagnostics befor transformation
plot(full.lm)
```

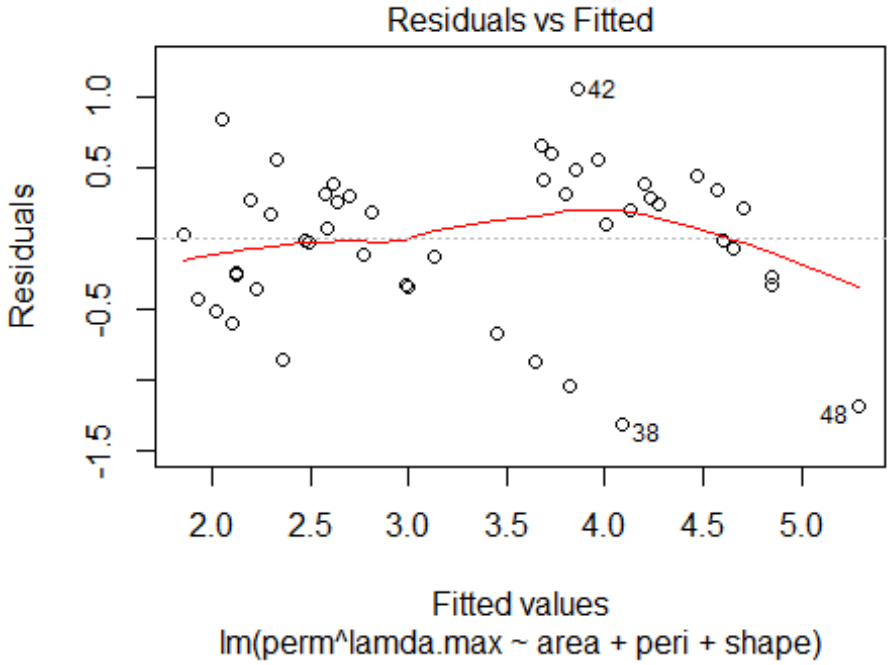

Residuals vs Fitted
Residuals
500
0
-500
42
40
38
0
200
400
600
800
1000
Fitted values
lm(perm ~ .)

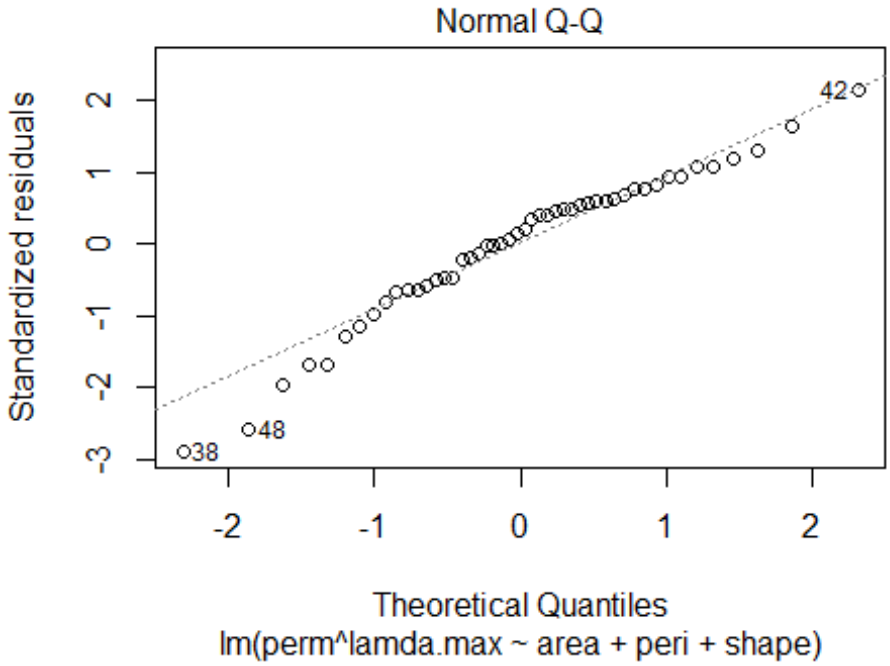

Normal Q-Q
Standardized residuals
3
2
1
0
-1
-2
-3
42
48
38
-2
-1
0
1
2
Theoretical Quantiles
lm(perm ~ .)

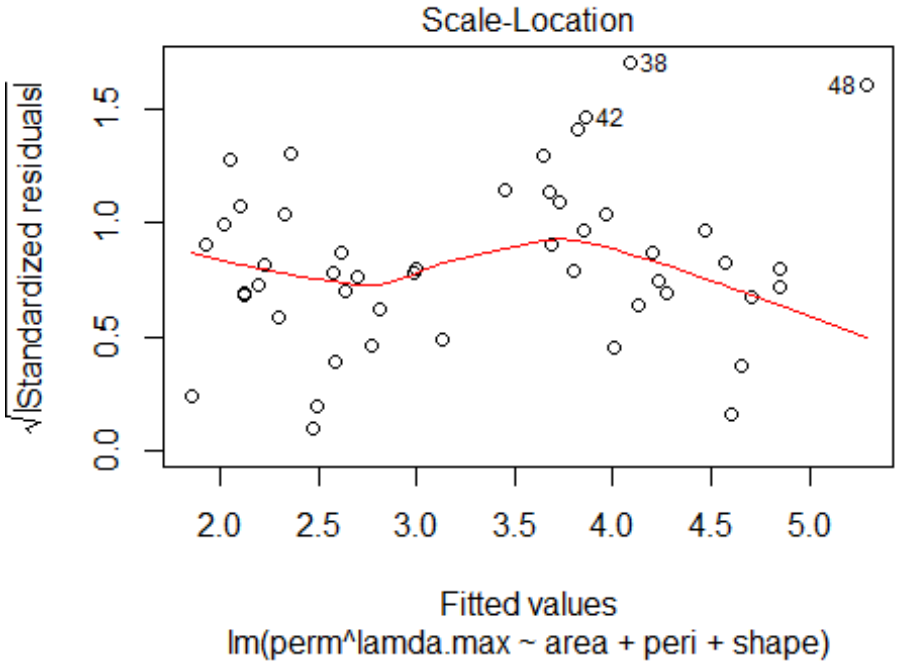


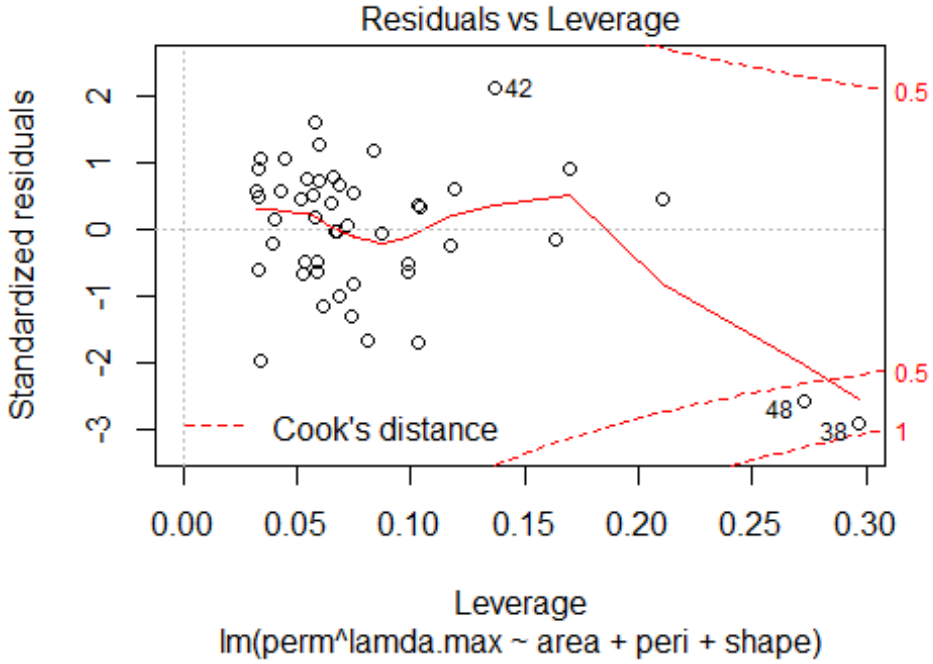



```
#Diagnostics after transformation
plot(bc.lm)
```


Residuals vs Fitted
Residuals
1.0
0.5
-0.5
-1.5
42
38
48
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Fitted values
lm(perm^lamda.max ~ area + peri + shape)


Normal Q-Q
Standardized residuals
2
1
0
-1
-2
-3
42
48
38
-2
-1
0
1
2
Theoretical Quantiles
lm(perm^lamda.max ~ area + peri + shape)

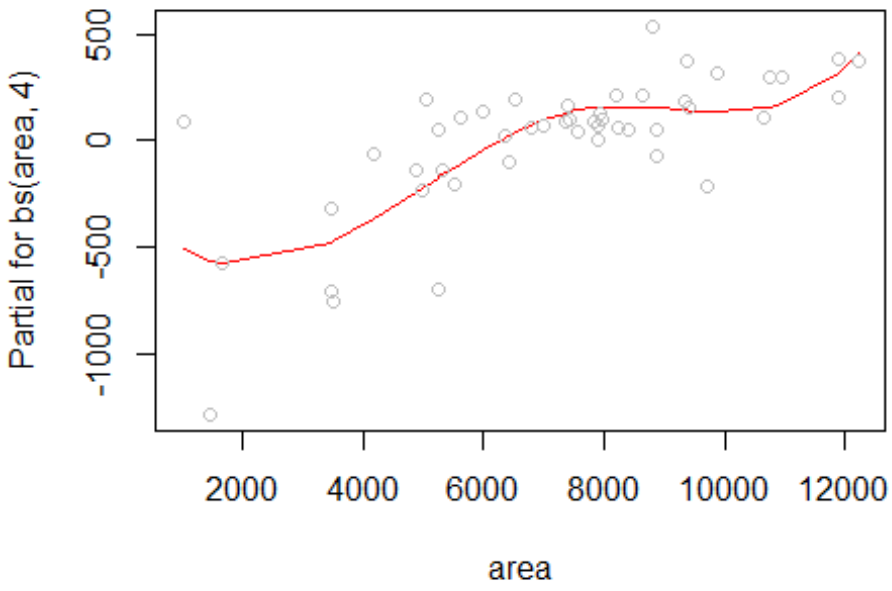




Compared to the old model, the new model does not perform better in the significat importance of *shape*. Also, the QQ-plot of the new model is less linear than the one of old model. Therefore, I will still keep the old model as the optimal one. I will continue to improve the model by transforming the predictions.

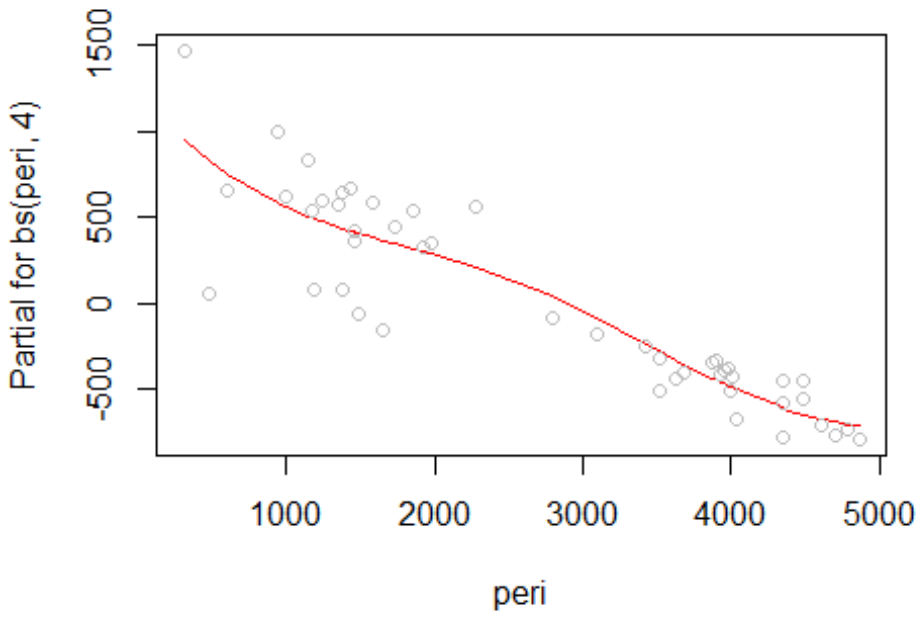
Transform the predictors

```
library(splines)
#Spline Transformations on area
spli.lm.area = lm(formula = perm ~ bs(area,4) + peri + shape, data = rock)
summary(spli.lm.area)

##
## Call:
## lm(formula = perm ~ bs(area, 4) + peri + shape, data = rock)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -719.81  -97.11  -14.58  122.83  596.98
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   558.42225  239.10061   2.336 0.024489 *
## bs(area, 4)1 -324.41847  355.51810  -0.913 0.366828
## bs(area, 4)2 1140.63681  296.04926   3.853 0.000403 ***
## bs(area, 4)3  455.24838  343.87564   1.324 0.192879
## bs(area, 4)4  919.45714  304.15143   3.023 0.004302 **
## peri           -0.33123    0.05122  -6.466  9.4e-08 ***
## shape        1072.78900  513.61243   2.089 0.042984 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 236.8 on 41 degrees of freedom
## Multiple R-squared:  0.7449, Adjusted R-squared:  0.7076
## F-statistic: 19.95 on 6 and 41 DF,  p-value: 9.523e-11

#Nature of the fit
termplot(spli.lm.area, partial=TRUE,terms = 1)
```



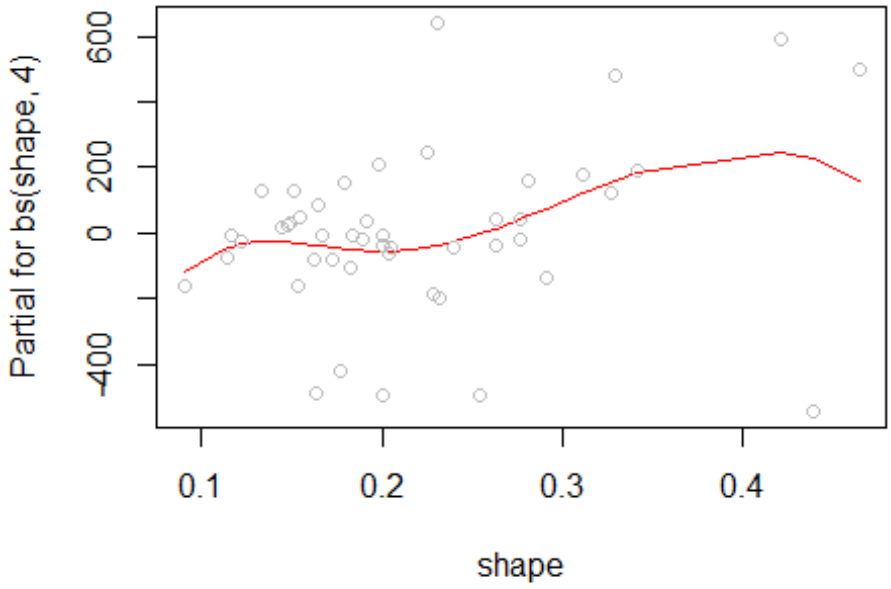

```
#Polynomial Transformations on peir
spli.lm.peri = lm(formula = perm ~ area + bs(peri,4) + shape, data = rock)
summary(spli.lm.peri)

##
## Call:
## lm(formula = perm ~ area + bs(peri, 4) + shape, data = rock)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -776.42  -82.72   26.21  112.62  521.38
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.913e+02  2.612e+02   1.881 0.067084 .
## area          9.602e-02  3.228e-02   2.974 0.004903 **
## bs(peri, 4)1 -5.614e+02  4.172e+02  -1.346 0.185758
## bs(peri, 4)2 -5.681e+02  4.604e+02  -1.234 0.224297
## bs(peri, 4)3 -1.595e+03  4.383e+02  -3.639 0.000758 ***
## bs(peri, 4)4 -1.660e+03  4.586e+02  -3.619 0.000805 ***
## shape         8.246e+02  5.687e+02   1.450 0.154686
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 252.4 on 41 degrees of freedom
## Multiple R-squared:  0.7101, Adjusted R-squared:  0.6677
## F-statistic: 16.74 on 6 and 41 DF,  p-value: 1.196e-09
```

```r
#Nature of the fit
termplot(spli.lm.peri, partial=TRUE,terms = 2)
```




```r
#Polynomial Transformations on shape
spli.lm.shape = lm(formula = perm ~ area + peri + bs(shape,4), data = rock)
summary(spli.lm.shape)

##
## Call:
## lm(formula = perm ~ area + peri + bs(shape, 4), data = rock)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -767.97  -47.91   14.56   82.50  678.38
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    555.01714  236.55311   2.346  0.02388 *
## area             0.09499    0.02710   3.505  0.00112 **
## peri            -0.34963    0.05464  -6.399 1.17e-07 ***
## bs(shape, 4)1  153.22661  320.04422   0.479  0.63465
## bs(shape, 4)2 -159.56884  339.78225  -0.470  0.64111
## bs(shape, 4)3  578.39233  473.13013   1.222  0.22851
## bs(shape, 4)4  274.23832  299.20377   0.917  0.36473
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

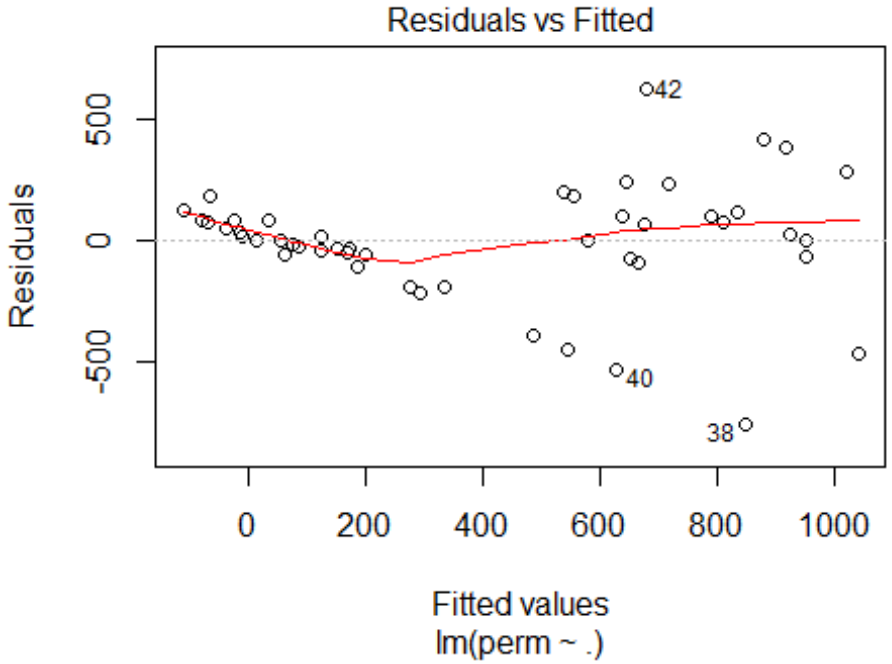
```
## Residual standard error: 251.8 on 41 degrees of freedom
## Multiple R-squared:  0.7115, Adjusted R-squared:  0.6693
## F-statistic: 16.85 on 6 and 41 DF,  p-value: 1.091e-09

#Nature of the fit
termplot(spli.lm.shape, partial=TRUE,terms = 3)
```

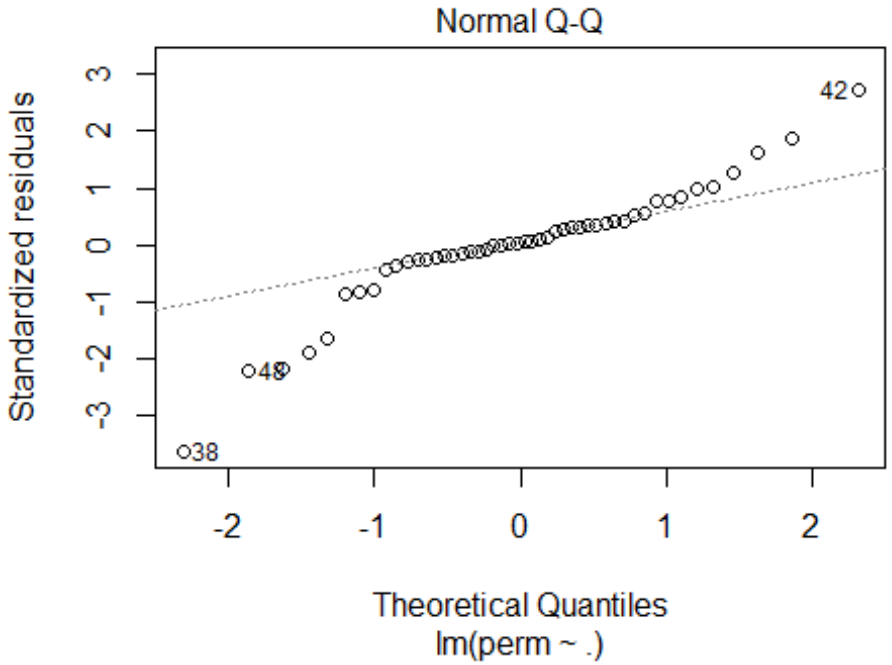


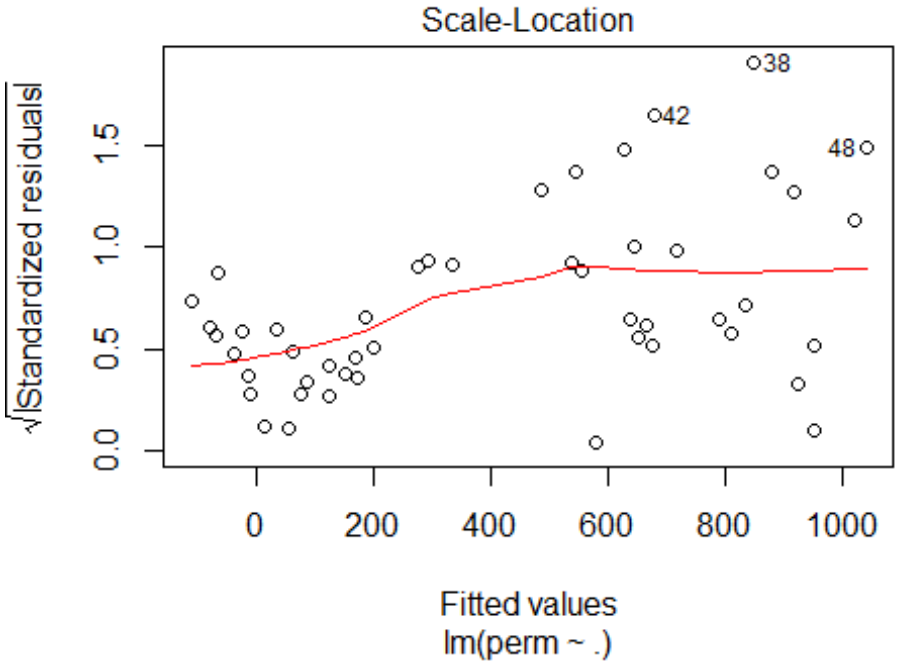
From the result, we can see that the partial for *peri* has negatively related to the constant fill, but do not add any significant, or improve the performance of the model. Thus, I will still choose my final model as *perm ~ area + peri + shape*.

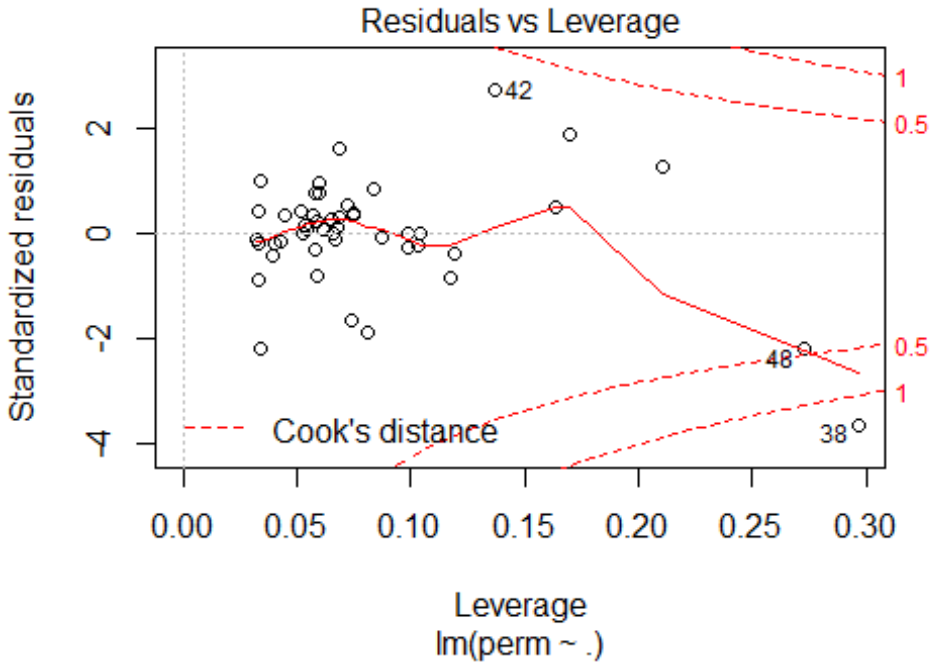
Diagnostics of the Choosing Model

```
plot(full.lm)
```


Residuals vs Fitted
Residuals
500
0
-500
42
40
38
0
200
400
600
800
1000
Fitted values
lm(perm ~ .)


Normal Q-Q
Standardized residuals
3
2
1
0
-1
-2
-3
42
48
38
-2
-1
0
1
2
Theoretical Quantiles
lm(perm ~ .)





Diagnosis Results: From the *Residuals vs. fitted values*, we can see the scatterplot distributes randomly, with some exception outliers. From the *Normal Probability plot*, the points are generally follow the straight line. From the *Scale Lotion*, the plot shows radom pattern. From the *Cook's distance*, we can see there are some outliers affecting the model.

Thus, according to the results above, I will pick the full model,*perm ~ area + peri + shape*, as my optimal model for the future steps.

Predictions of Future Observations

```
#New observations
new.obs <- data.frame(
  area = c(8, 800, 800, 800, 800, 8000),
  peri = c(3, 3, 30, 300, 3000, 3000),
  shape = c(5, 50, 5, 500, 0.05, 0.5)
)

#Predict based on 99% prediction interval
full.pre = predict(full.lm, se.fit=T, newdata=new.obs, interval='prediction',
 level = .99)
full.pre$fit
```

```
##           fit          lwr          upr
## 1   4980.6629   -1562.8912   11524.2169
## 2  45511.1159  -22396.7351  113418.9670
## 3   5043.7106   -1482.4446   11569.8657
## 4 449990.1075 -232119.0217 1132099.2368
## 5   -428.4353   -1229.4000     372.5293
## 6    633.7491    -140.5311    1408.0293
```

From the results above, we can see that the shape has a great effect on the results.

Interpretation of the Model

```
summary(full.lm)
```

```
##
## Call:
## lm(formula = perm ~ ., data = rock)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -750.26  -59.57   10.66  100.25  620.91
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 485.61797  158.40826   3.066 0.003705 **
## area          0.09133    0.02499   3.654 0.000684 ***
## peri         -0.34402    0.05111  -6.731 2.84e-08 ***
## shape       899.06926  506.95098   1.773 0.083070 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 246 on 44 degrees of freedom
```

```
## Multiple R-squared:  0.7044, Adjusted R-squared:  0.6843
## F-statistic: 34.95 on 3 and 44 DF,  p-value: 1.033e-11
```

According to our model, *perm = 899.07shape + 0.09area - 0.34peri + 485.62*. The permeability of a petroleum rock has been greatly affecting by its shape, and it is postively related with its area and shape, but negatively related with its perimeter.

Exercise 5 The *prostate* data - use *lpsa* as the response.

Load *prostate* data

```
library(faraway)
head(prostate)
```

```
##       lcavol lweight age      lbph svi      lcp gleason pgg45     lpsa
## 1 -0.5798185  2.7695  50 -1.386294   0 -1.38629       6     0 -0.43078
## 2 -0.9942523  3.3196  58 -1.386294   0 -1.38629       6     0 -0.16252
## 3 -0.5108256  2.6912  74 -1.386294   0 -1.38629       7    20 -0.16252
## 4 -1.2039728  3.2828  58 -1.386294   0 -1.38629       6     0 -0.16252
## 5  0.7514161  3.4324  62 -1.386294   0 -1.38629       6     0  0.37156
## 6 -1.0498221  3.2288  50 -1.386294   0 -1.38629       6     0  0.76547
```

```
str(prostate)
```

```
## 'data.frame':    97 obs. of  9 variables:
##  $ lcavol : num  -0.58 -0.994 -0.511 -1.204 0.751 ...
##  $ lweight: num  2.77 3.32 2.69 3.28 3.43 ...
##  $ age    : int  50 58 74 58 62 50 64 58 47 63 ...
##  $ lbph   : num  -1.39 -1.39 -1.39 -1.39 -1.39 ...
##  $ svi    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ lcp    : num  -1.39 -1.39 -1.39 -1.39 -1.39 ...
##  $ gleason: int  6 6 7 6 6 6 6 6 6 6 ...
##  $ pgg45  : int  0 0 20 0 0 0 0 0 0 0 ...
##  $ lpsa   : num  -0.431 -0.163 -0.163 -0.163 0.372 ...
```

According to the information above, we know that *prostate* is a data frame with 97 observations and 9 columns.

lcavol: log(cancer volume).

lweight: log(prostate weight).

age: age.

lbph: log(benign prostatic hyperplasia amount)

svi: seminal vesicle invasion.

lcp: log(capsular penetration).

gleason: Gleason score.

pgg45: percentage Gleason scores 4 or 5.

lpsa: log(prostate specific antigen).

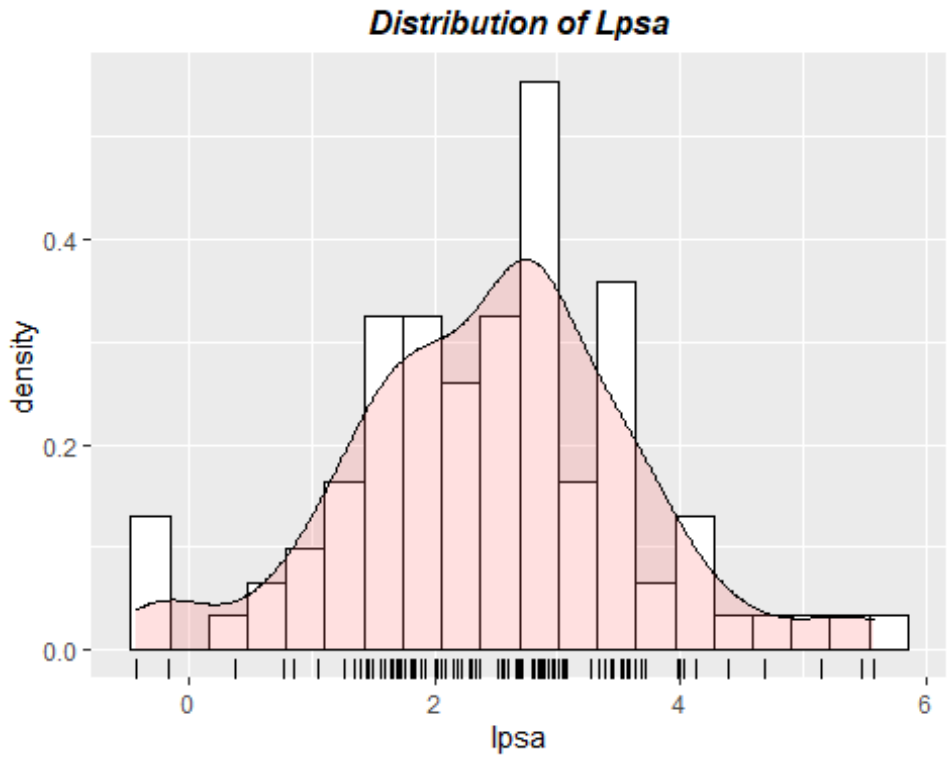
Initial Data Analysis

```
#Summarize the dataset
summary(prostate)
```

```
##      lcavol           lweight           age              lbph        
##  Min.   :-1.3471   Min.   :2.375   Min.   :41.00   Min.   :-1.3863  
##  1st Qu.: 0.5128   1st Qu.:3.376   1st Qu.:60.00   1st Qu.:-1.3863  
##  Median : 1.4469   Median :3.623   Median :65.00   Median : 0.3001  
##  Mean   : 1.3500   Mean   :3.653   Mean   :63.87   Mean   : 0.1004  
##  3rd Qu.: 2.1270   3rd Qu.:3.878   3rd Qu.:68.00   3rd Qu.: 1.5581  
##  Max.   : 3.8210   Max.   :6.108   Max.   :79.00   Max.   : 2.3263  
##       svi              lcp             gleason          pgg45       
##  Min.   :0.0000   Min.   :-1.3863   Min.   :6.000   Min.   :  0.00  
##  1st Qu.:0.0000   1st Qu.:-1.3863   1st Qu.:6.000   1st Qu.:  0.00  
##  Median :0.0000   Median :-0.7985   Median :7.000   Median : 15.00  
##  Mean   :0.2165   Mean   :-0.1794   Mean   :6.753   Mean   : 24.38  
##  3rd Qu.:0.0000   3rd Qu.: 1.1786   3rd Qu.:7.000   3rd Qu.: 40.00  
##  Max.   :1.0000   Max.   : 2.9042   Max.   :9.000   Max.   :100.00  
##       lpsa        
##  Min.   :-0.4308  
##  1st Qu.: 1.7317  
##  Median : 2.5915  
##  Mean   : 2.4784  
##  3rd Qu.: 3.0564  
##  Max.   : 5.5829
```

We have nine numerical variables, and the six summary statistics show us the general distribution of the variables. I am going to analyze in depth with the *lpsa* response.

```
#Distribution of response
library(ggplot2)
ggplot(prostate, aes(x=lpsa)) +
  geom_histogram(aes(y=..density..), bins = 20,fill = "white", col = "black")
 +
  geom_density(alpha=.2, fill="#FF6666") +
  geom_rug() +
  labs(title = 'Distribution of Lpsa') +
  theme(plot.title = element_text(hjust = 0.5, size=12, face="bold.italic"))
```

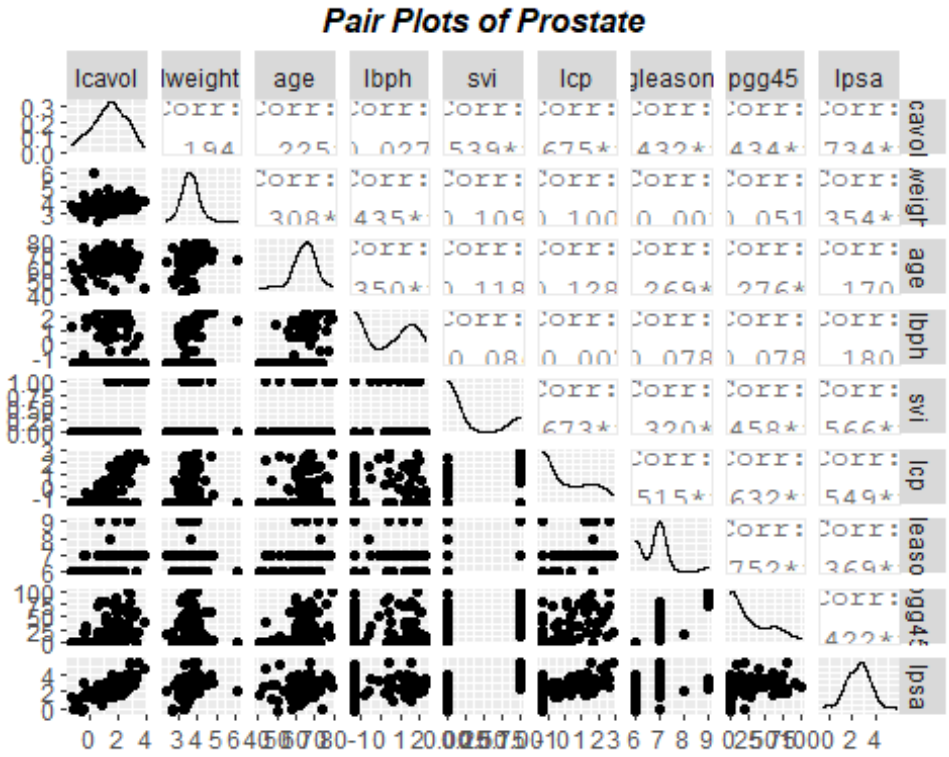


From the distribution plot above, we can see that the data are generally normal distributed, and there are some outliers less than zero

```
#Correlation between variables
cor(prostate)
```

```
##              lcavol      lweight       age        lbph         svi
lcp
## lcavol  1.00000000  0.194128387 0.2249999  0.02734971  0.53884500  0.67531
058
## lweight 0.19412839  1.000000000 0.3075247  0.43493174  0.10877818  0.10023
889
## age     0.22499988  0.307524741 1.0000000  0.35018592  0.11765804  0.12766
778
## lbph    0.02734971  0.434931744 0.3501859  1.00000000 -0.08584327 -0.00699
944
## svi     0.53884500  0.108778185 0.1176580 -0.08584327  1.00000000  0.67311
122
## lcp     0.67531058  0.100238891 0.1276678 -0.00699944  0.67311122  1.00000
000
## gleason 0.43241705 -0.001283003 0.2688916  0.07782044  0.32041222  0.51482
991
## pgg45   0.43365224  0.050846195 0.2761124  0.07846000  0.45764762  0.63152
807
## lpsa    0.73446028  0.354121818 0.1695929  0.17980950  0.56621818  0.54881
316
##              gleason     pgg45      lpsa
```

```
## lcavol   0.432417052 0.4336522 0.7344603
## lweight -0.001283003 0.0508462 0.3541218
## age      0.268891599 0.2761124 0.1695929
## lbph     0.077820444 0.0784600 0.1798095
## svi      0.320412221 0.4576476 0.5662182
## lcp      0.514829912 0.6315281 0.5488132
## gleason  1.000000000 0.7519045 0.3689867
## pgg45    0.751904512 1.0000000 0.4223157
## lpsa     0.368986693 0.4223157 1.0000000

#Pair Plots
library(ggplot2)
library(GGally)
ggpairs(prostate) +
  ggtitle("Pair Plots of Prostate") +
  theme(plot.title = element_text(hjust = 0.5, size=12, face="bold.italic"))
```



From the results above, we can see that *lcavol*, *lweight*, and *svi* has some correlations with *lpsa*. We can continue to define a linear model, and find deeper relationship.

Variable Selection

```
#Build up full model
full.lm = lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gle
ason +
```

```
    pgg45, data = prostate)
print(summary(full.lm))

##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp +
##     gleason + pgg45, data = prostate)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7331 -0.3713 -0.0170  0.4141  1.6381
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.669337   1.296387   0.516  0.60693
## lcavol       0.587022   0.087920   6.677 2.11e-09 ***
## lweight      0.454467   0.170012   2.673  0.00896 **
## age         -0.019637   0.011173  -1.758  0.08229 .
## lbph         0.107054   0.058449   1.832  0.07040 .
## svi          0.766157   0.244309   3.136  0.00233 **
## lcp         -0.105474   0.091013  -1.159  0.24964
## gleason      0.045142   0.157465   0.287  0.77503
## pgg45        0.004525   0.004421   1.024  0.30886
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7084 on 88 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.6234
## F-statistic: 20.86 on 8 and 88 DF,  p-value: < 2.2e-16

#Apply backward selection model
full.backward = step(full.lm, direction = "backward")

## Start:  AIC=-58.32
## lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##
##           Df Sum of Sq    RSS     AIC
## - gleason  1    0.0412 44.204 -60.231
## - pgg45    1    0.5258 44.689 -59.174
## - lcp      1    0.6740 44.837 -58.853
## <none>                 44.163 -58.322
## - age      1    1.5503 45.713 -56.975
## - lbph     1    1.6835 45.847 -56.693
## - lweight  1    3.5861 47.749 -52.749
## - svi      1    4.9355 49.099 -50.046
## - lcavol   1   22.3721 66.535 -20.567
##
## Step:  AIC=-60.23
## lpsa ~ lcavol + lweight + age + lbph + svi + lcp + pgg45
```

```
##
##           Df Sum of Sq    RSS     AIC
## - lcp      1    0.6623 44.867 -60.789
## <none>                  44.204 -60.231
## - pgg45    1    1.1920 45.396 -59.650
## - age      1    1.5166 45.721 -58.959
## - lbph     1    1.7053 45.910 -58.560
## - lweight  1    3.5462 47.750 -54.746
## - svi      1    4.8984 49.103 -52.037
## - lcavol   1   23.5039 67.708 -20.872
##
## Step:  AIC=-60.79
## lpsa ~ lcavol + lweight + age + lbph + svi + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - pgg45    1    0.6590 45.526 -61.374
## <none>                  44.867 -60.789
## - age      1    1.2649 46.131 -60.092
## - lbph     1    1.6465 46.513 -59.293
## - lweight  1    3.5647 48.431 -55.373
## - svi      1    4.2503 49.117 -54.009
## - lcavol   1   25.4189 70.285 -19.248
##
## Step:  AIC=-61.37
## lpsa ~ lcavol + lweight + age + lbph + svi
##
##           Df Sum of Sq    RSS     AIC
## <none>                  45.526 -61.374
## - age      1    0.9592 46.485 -61.352
## - lbph     1    1.8568 47.382 -59.497
## - lweight  1    3.2251 48.751 -56.735
## - svi      1    5.9517 51.477 -51.456
## - lcavol   1   28.7665 74.292 -15.871
```

```
print(summary(full.backward))
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.83505 -0.39396  0.00414  0.46336  1.57888
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.95100    0.83175   1.143 0.255882
## lcavol       0.56561    0.07459   7.583 2.77e-11 ***
## lweight      0.42369    0.16687   2.539 0.012814 *
## age         -0.01489    0.01075  -1.385 0.169528
```

```
## lbph         0.11184    0.05805   1.927 0.057160 .
## svi          0.72095    0.20902   3.449 0.000854 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7073 on 91 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6245
## F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16
```

```
#Apply forward selection model
full.forward <- step(lm(lpsa ~ 1, data=prostate), list(upper=full.lm), direct
ion='forward')
```

```
## Start:  AIC=28.84
## lpsa ~ 1
##
##           Df Sum of Sq     RSS     AIC
## + lcavol   1    69.003  58.915 -44.366
## + svi      1    41.011  86.907  -6.658
## + lcp      1    38.528  89.389  -3.926
## + pgg45    1    22.814 105.103  11.783
## + gleason  1    17.416 110.501  16.641
## + lweight  1    16.041 111.876  17.840
## + lbph     1     4.136 123.782  27.650
## + age      1     3.679 124.238  28.007
## <none>                127.918  28.837
##
## Step:  AIC=-44.37
## lpsa ~ lcavol
##
##           Df Sum of Sq    RSS     AIC
## + lweight  1    5.9485 52.966 -52.690
## + svi      1    5.2375 53.677 -51.397
## + lbph     1    3.2658 55.649 -47.898
## + pgg45    1    1.6980 57.217 -45.203
## <none>                58.915 -44.366
## + lcp      1    0.6562 58.259 -43.453
## + gleason  1    0.4156 58.499 -43.053
## + age      1    0.0025 58.912 -42.370
##
## Step:  AIC=-52.69
## lpsa ~ lcavol + lweight
##
##           Df Sum of Sq    RSS     AIC
## + svi      1    5.1814 47.785 -60.676
## + pgg45    1    1.9489 51.017 -54.327
## <none>                52.966 -52.690
## + lcp      1    0.8371 52.129 -52.236
## + gleason  1    0.7810 52.185 -52.131
## + lbph     1    0.6751 52.291 -51.935
```

```
## + age      1    0.4200 52.546 -51.463
##
## Step:  AIC=-60.68
## lpsa ~ lcavol + lweight + svi
##
##           Df Sum of Sq    RSS     AIC
## + lbph     1   1.30006 46.485 -61.352
## <none>                 47.785 -60.676
## + pgg45    1   0.57347 47.211 -59.847
## + age      1   0.40251 47.382 -59.497
## + gleason  1   0.38901 47.396 -59.469
## + lcp      1   0.06412 47.721 -58.806
##
## Step:  AIC=-61.35
## lpsa ~ lcavol + lweight + svi + lbph
##
##           Df Sum of Sq    RSS     AIC
## + age      1   0.95924 45.526 -61.374
## <none>                 46.485 -61.352
## + pgg45    1   0.35332 46.131 -60.092
## + gleason  1   0.21256 46.272 -59.796
## + lcp      1   0.10230 46.383 -59.565
##
## Step:  AIC=-61.37
## lpsa ~ lcavol + lweight + svi + lbph + age
##
##           Df Sum of Sq    RSS     AIC
## <none>                 45.526 -61.374
## + pgg45    1   0.65896 44.867 -60.789
## + gleason  1   0.45601 45.070 -60.351
## + lcp      1   0.12927 45.396 -59.650
```

```
print(summary(full.forward))
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + svi + lbph + age, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.83505 -0.39396  0.00414  0.46336  1.57888
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.95100    0.83175   1.143 0.255882
## lcavol       0.56561    0.07459   7.583 2.77e-11 ***
## lweight      0.42369    0.16687   2.539 0.012814 *
## svi          0.72095    0.20902   3.449 0.000854 ***
## lbph         0.11184    0.05805   1.927 0.057160 .
## age         -0.01489    0.01075  -1.385 0.169528
```

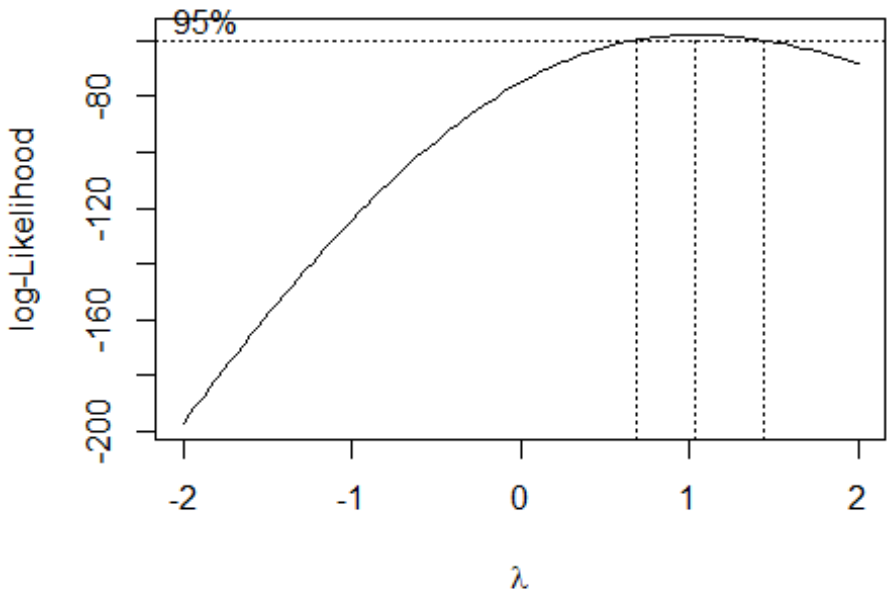
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7073 on 91 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6245
## F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16
```

I apply both backward and forward selection to the model, and have the same optimal model. According to the report, the optimal model is *lpsa ~ lcavol + lweight + age + lbph + svi*. However, the significant importance of some variables are mild. We may think of improving the model by transforming the variables.

Exploration of Transformations

Transform the response lpsa

```
#Box-Cox Transformation of the response
#Shift lpsa to all postive value
prostate$'shift_lpsa' = prostate$lpsa+(-min(prostate$lpsa))+1

library(MASS)
full.bc = boxcox(shift_lpsa ~ lcavol + lweight + age + lbph + svi, data=prost
ate)
```




```
#Get the lamda of maximum log-Likelihood
lamda.max = full.bc$x[full.bc$y==max(full.bc$y)]
```

```r
#Set up new Model
bc.lm = lm(shift_lpsa^lamda.max ~ lcavol + lweight + age + lbph + svi, data=p
rostate)
summary(bc.lm)
```

```
##
## Call:
## lm(formula = shift_lpsa^lamda.max ~ lcavol + lweight + age +
##     lbph + svi, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.97557 -0.42195  0.00096  0.49966  1.68867
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.45702    0.89141   2.756 0.007062 **
## lcavol       0.60494    0.07994   7.567 2.98e-11 ***
## lweight      0.45233    0.17884   2.529 0.013155 *
## age         -0.01608    0.01153  -1.395 0.166320
## lbph         0.11908    0.06222   1.914 0.058780 .
## svi          0.78126    0.22402   3.487 0.000753 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.758 on 91 degrees of freedom
## Multiple R-squared:  0.6443, Adjusted R-squared:  0.6247
## F-statistic: 32.96 on 5 and 91 DF,  p-value: < 2.2e-16
```

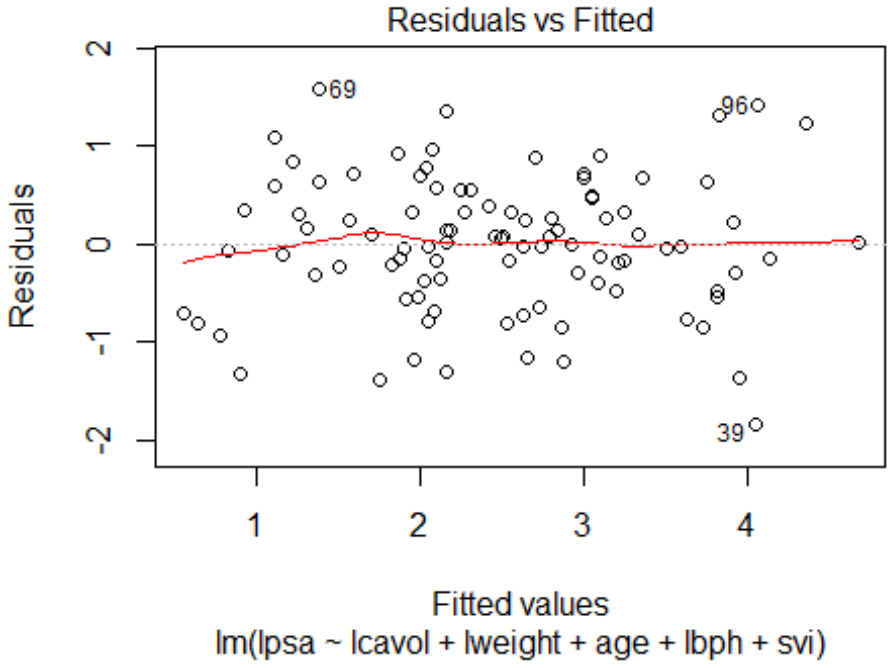
From the result, we can see that the lamda of maximum log-Likelihood is very closed to 1. And the new model's performance does not have too much change compared to the old one's. Thus, I will still choose my final model as *lpsa ~ lcavol + lweight + age + lbph + svi*.

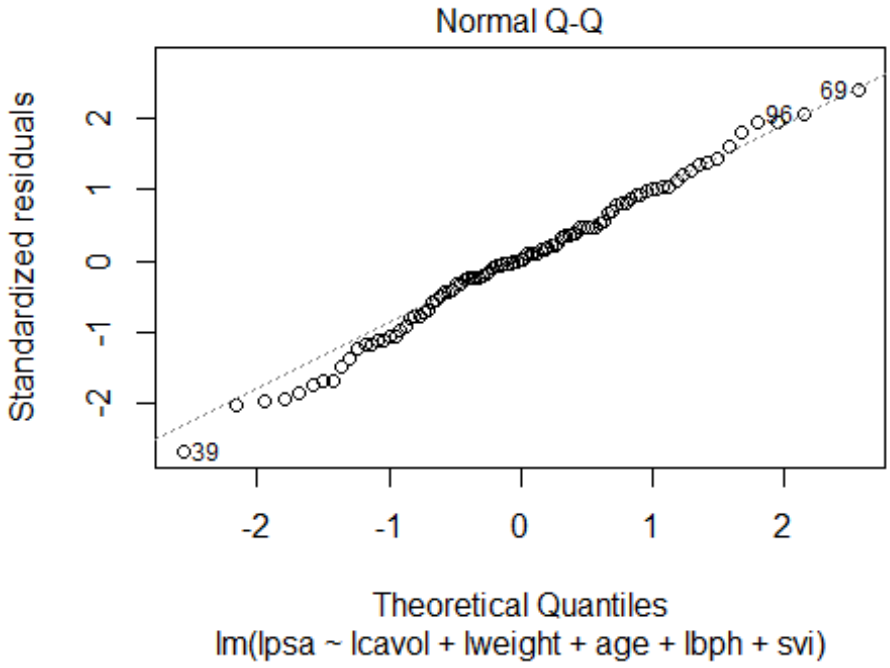
Diagnostics of the Choosing Model

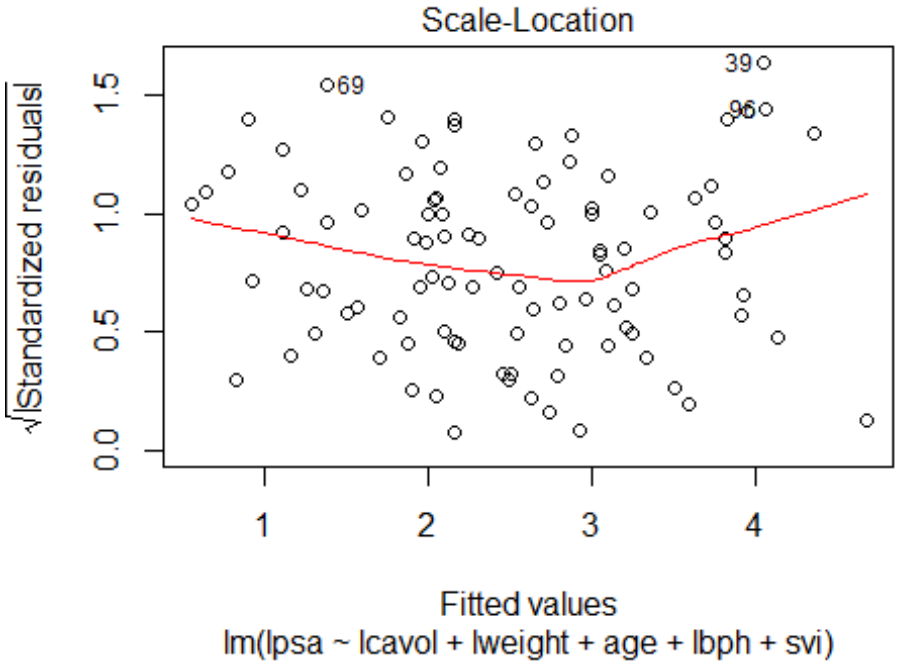
```r
my.lm = lm(lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)
summary(my.lm)
```

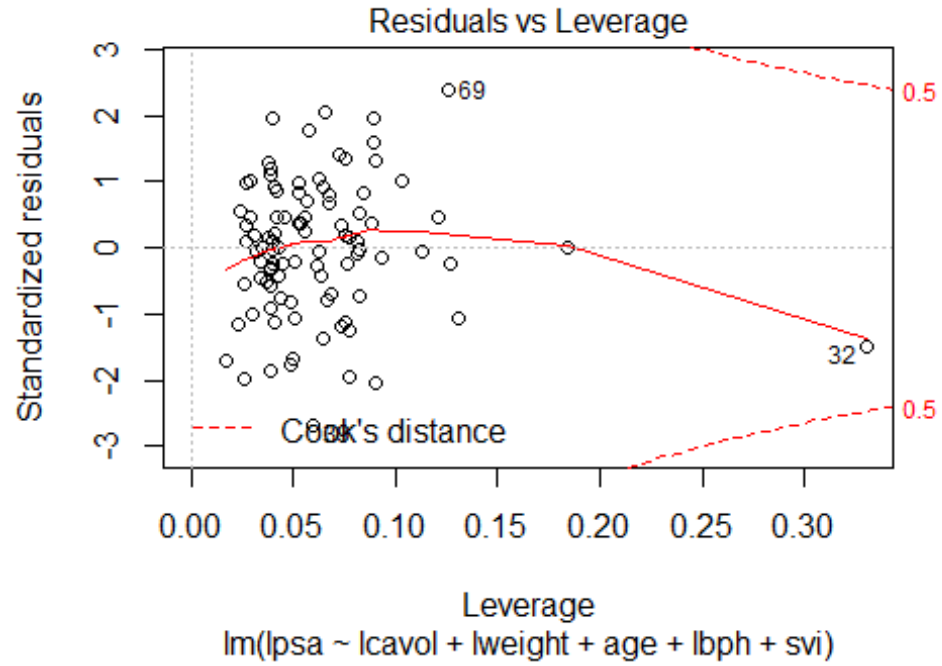
```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.83505 -0.39396  0.00414  0.46336  1.57888
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.95100    0.83175   1.143 0.255882
```

```
## lcavol       0.56561    0.07459   7.583 2.77e-11 ***
## lweight      0.42369    0.16687   2.539 0.012814 *
## age         -0.01489    0.01075  -1.385 0.169528
## lbph         0.11184    0.05805   1.927 0.057160 .
## svi          0.72095    0.20902   3.449 0.000854 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7073 on 91 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6245
## F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16

plot(my.lm)
```


Residuals vs Fitted
Residuals
Fitted values
lm(lpsa ~ lcavol + lweight + age + lbph + svi)
69
96
39


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(lpsa ~ lcavol + lweight + age + lbph + svi)
69
96
39





Diagnosis Results: From the *Residuals vs. fitted values*, we can see the scatterplot distributes randomly, with some exception outliers. From the *Normal Probability plot*, the points are generally follow the straight line. From the *Scale Lotion*, the plot shows a little downward pattern, but generally random. From the *Cook's distance*, we can see there are

some outliers affecting the model. Thus, according to the results above, I will pick the model, *lpsa ~ lcavol + lweight + age + lbph + svi*, as my optimal model for the future steps.

Predictions of Future Observations

```
#New observations
new.obs <- data.frame(
  lcavol = runif(5, min = -2, max = 4),
  lweight = runif(5, min = 2, max = 6),
  age = floor(runif(5, min = 41, max = 79)),
  lbph = runif(5, min = -2, max = 3),
  svi = c(0,1,0,1,0),
  lpsa = runif(5, min = -1, max = 6)
)

#Predict based on 99% prediction interval
my.pre = predict(my.lm, se.fit=T, newdata=new.obs, interval='prediction', lev
el = .99)
my.pre$fit
```

```
##        fit        lwr      upr
## 1 2.461497  0.4563694 4.466625
## 2 2.605737  0.5879052 4.623568
## 3 1.625393 -0.5156662 3.766451
## 4 5.265622  3.2216395 7.309604
## 5 0.955462 -0.9850258 2.895950
```

Interpretation of the Model

```
summary(my.lm)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.83505 -0.39396  0.00414  0.46336  1.57888
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.95100    0.83175   1.143 0.255882
## lcavol       0.56561    0.07459   7.583 2.77e-11 ***
## lweight      0.42369    0.16687   2.539 0.012814 *
## age         -0.01489    0.01075  -1.385 0.169528
## lbph         0.11184    0.05805   1.927 0.057160 .
## svi          0.72095    0.20902   3.449 0.000854 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.7073 on 91 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6245
## F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16
```

According to our model, *lpsa = 0.72svi + 0.11lbph - 0.01age + 0.42lweight + 0.57lcavol + 0.95*. The prostate specific antigen is postively related to the seminal vesicle invasion, cancer volume, prostate weight, and benign prostatic hyperplasia amount, but negatively related to age. The prostate specific antigen is greatly affect by cancer volume and seminal vesicle invasion.