Seaworld

Yifeng Huang

August 16, 2018

```
## Loading required package: carData
## corrplot 0.84 loaded
##
## Attaching package: 'psych'
## The following object is masked from 'package:car':
##
##     logit
##
## Attaching package: 'ggplot2'
## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

Data Clean

Overview

```
#data import
seawatch.ori<-read_excel("~/MSBA notes/Business Stats/Seawatch C w blanks-1.xlsx")

# we delete CNVHRS, Notes, City and Zip code.
seawatch<-seawatch.ori[,3:20]
seawatch<-seawatch[,-2]
# overview
head(seawatch)
```

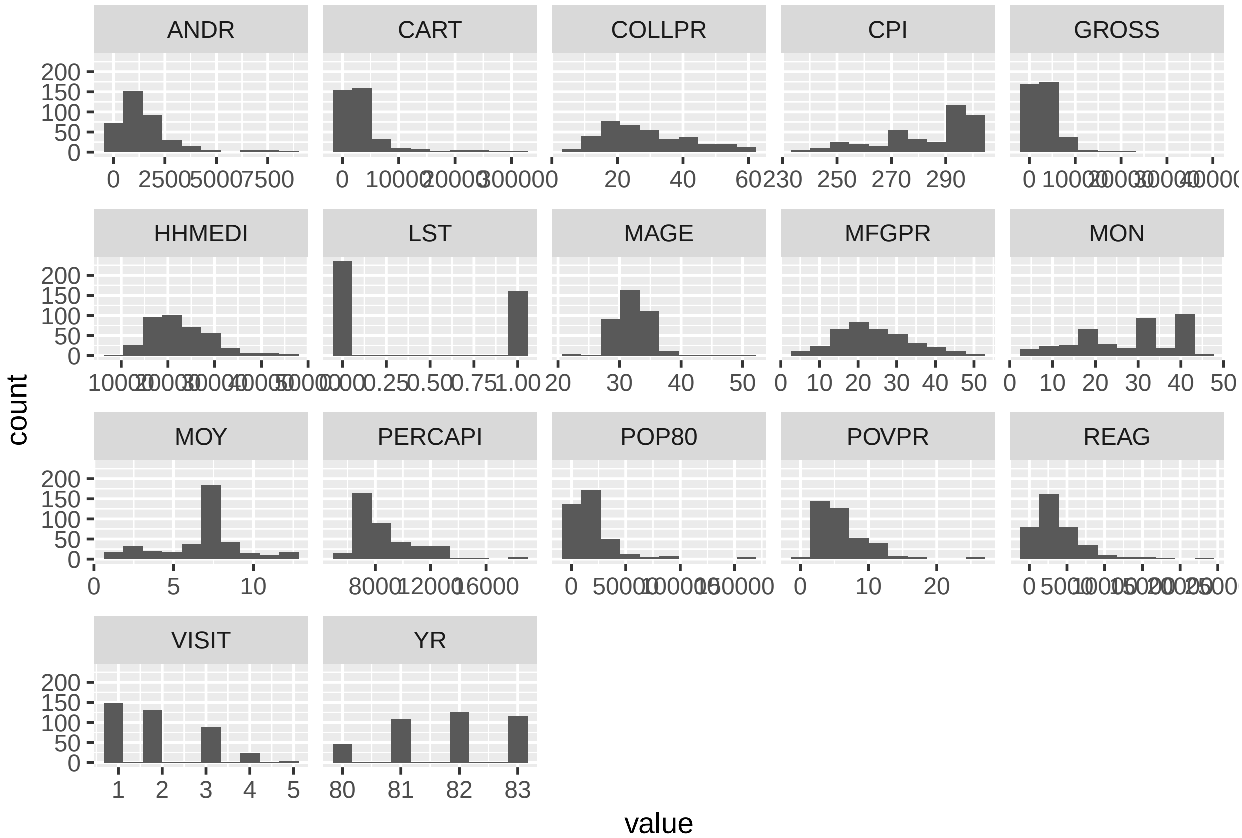
```
## # A tibble: 6 x 17
##   GROSS   MOY    YR   MON VISIT   LST   CPI POP80 HHMEDI PERCAPI POVPR
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>   <dbl> <dbl>
## 1  6222     9    80     9     1     0  249. 17544  27323   10522   3.8
## 2  8641     9    81    21     2     0  276. 17544  27323   10522   3.8
## 3 10687     9    82    33     3     1  293. 17544  27323   10522   3.8
## 4  2423     7    81    19     1     0  271. 10381  19387    7317   6.3
## 5  2609     7    82    31     2     0  291. 10381  19387    7317   6.3
## 6  1321     7    83    43     3     1  300. 10381  19387    7317   6.3
## # ... with 6 more variables: MFGPR <dbl>, COLLPR <dbl>, MAGE <dbl>,
## #   CART <dbl>, REAG <dbl>, ANDR <dbl>
```

```
describe(seawatch)
```

```
##          vars   n    mean      sd  median trimmed     mad    min
## GROSS       1 394 3441.30 4056.82  2419.5 2755.20 2227.61   43.0
## MOY         2 396    6.62    2.45     7.0    6.66    1.48    1.0
## YR          3 396   81.79    0.99    82.0   81.86    1.48   80.0
```

```
## MON        4 396    28.13    11.72    31.0    28.85    17.79     3.0
## VISIT      5 396     2.00     0.97     2.0     1.90     1.48     1.0
## LST        6 396     0.41     0.49     0.0     0.38     0.00     0.0
## CPI        7 396   282.14    16.85   290.6   284.31    13.34   236.4
## POP80      8 387 19179.16 22616.29 13212.0 14863.73 11421.95   688.0
## HHMEDI     9 387 22643.48  6654.16 21304.0 21984.26  6862.96 10108.0
## PERCAPI   10 387  8706.69  2236.39  8060.0  8429.94  1885.87  5188.0
## POVPR     11 387     6.20     3.96     5.1     5.68     2.67     0.4
## MFGPR     12 371    23.89     9.65    22.3    23.41     8.60     3.0
## COLLPR    13 371    28.72    13.20    25.9    27.55    14.97     8.2
## MAGE      14 387    32.23     3.06    32.1    32.07     2.67    21.7
## CART      15 380  3784.08  5064.56  2171.5  2628.76  1965.19    97.0
## REAG      16 380  3881.71  3430.15  3067.0  3359.21  2604.93    84.0
## ANDR      17 380  1528.81  1463.64  1105.0  1262.62   919.21    63.0
##               max    range  skew kurtosis      se
## GROSS     38256.0  38213.0  4.12    25.26  204.38
## MOY          12.0     11.0 -0.24     0.52    0.12
## YR           83.0      3.0 -0.28    -1.03    0.05
## MON          44.0     41.0 -0.27    -1.06    0.59
## VISIT         5.0      4.0  0.67    -0.27    0.05
## LST           1.0      1.0  0.38    -1.86    0.02
## CPI         300.9     64.5 -0.85    -0.38    0.85
## POP80    161799.0 161111.0  3.60    16.78 1149.65
## HHMEDI    47646.0  37538.0  1.03     1.39  338.25
## PERCAPI   17850.0  12662.0  1.30     1.94  113.68
## POVPR        26.1     25.7  2.00     6.23    0.20
## MFGPR        48.4     45.4  0.41    -0.16    0.50
## COLLPR       61.7     53.5  0.65    -0.47    0.69
## MAGE         50.2     28.5  1.33     7.57    0.16
## CART      31225.0  31128.0  3.13    10.62  259.81
## REAG      23339.0  23255.0  2.17     7.07  175.96
## ANDR       8586.0   8523.0  2.28     6.40   75.08
```

```r
ggplot(gather(seawatch), aes(value)) +
    geom_histogram(bins = 10) +
    facet_wrap(~key, scales = 'free_x')
```




```
## Note that VISIT and LST are categorical variables

## Convert numberic variables to Categorical
seawatch$VISIT<-as.factor(seawatch$VISIT)
seawatch$LST<-as.factor(seawatch$LST)
```

Missing Values

```
# number of NA's
nrow(seawatch)-nrow(na.omit(seawatch))
```

```
## [1] 34
```

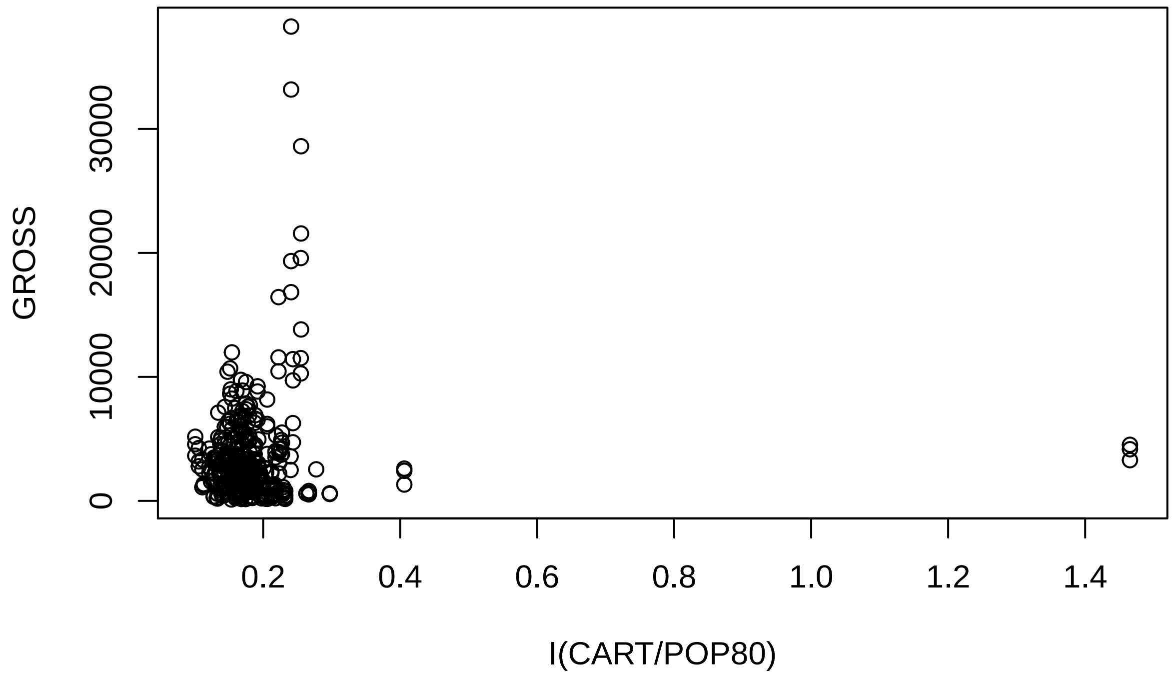
```
## Since there are only 34 rows containing na's we can simply delete it
seawatch<-na.omit(seawatch)
```

Other strange observation

CART, REAG, and ANDR

- For some observations, the number of votes is bigger than total population. Due to the high correlation (0.9533607) between POP80 and the sum of those there vote numbers, we can build a model to predict the right population.

```r
# scatter plot
plot(GROSS~I(CART/POP80),data = seawatch)
```



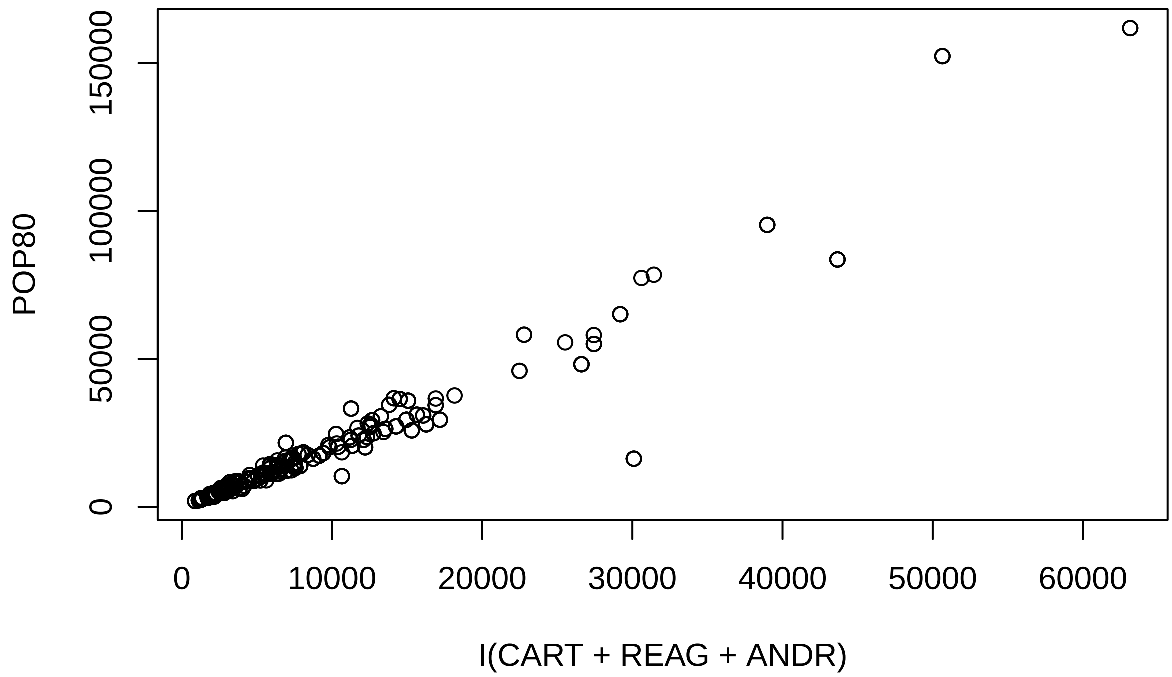

```r
# observations that CART or REAG or ANDR is larger than total population
ex.obs<-seawatch$CART>seawatch$POP80 | seawatch$REAG>seawatch$POP80 | seawatch$REAG>seawatch$POP80

seawatch[ex.obs,c("POP80",'REAG','ANDR')]
```

```
## # A tibble: 3 x 3
##   POP80  REAG  ANDR
##   <dbl> <dbl> <dbl>
## 1 16301  4638  1572
## 2 16301  4638  1572
## 3 16301  4638  1572
```

```r
# correlation
cor(seawatch$POP80,I(seawatch$CART+seawatch$REAG+seawatch$ANDR))
```

```
## [1] 0.9533607
```

```r
# plot pop80 and sum of those 3 vote numbers
plot(POP80~I(CART+REAG+ANDR),data = seawatch)
```




```r
# original sd
sd(seawatch[-ex.obs,]$POP80)
```

```
## [1] 23057.15
```

```r
# predictive model
## full model
pop.lm<-lm(POP80~.+I(CART+REAG+ANDR),data = seawatch[-ex.obs,])
## Predictors selection
step(pop.lm,direction = "backward",trace = 0)
```

```
##
## Call:
## lm(formula = POP80 ~ GROSS + MOY + CPI + HHMEDI + POVPR + MFGPR +
##     MAGE + CART + REAG + ANDR, data = seawatch[-ex.obs, ])
##
## Coefficients:
## (Intercept)        GROSS          MOY          CPI       HHMEDI
##  -1.515e+04   -8.491e-01   -1.790e+02    5.996e+01    1.132e-01
##       POVPR        MFGPR         MAGE         CART         REAG
##   7.209e+02    1.418e+02   -4.204e+02    1.183e+00    2.222e+00
##        ANDR
##   7.499e+00
```

```r
## update model
pop.lm<-lm(formula = POP80 ~ GROSS + MOY + CPI + HHMEDI + POVPR + MFGPR +
    MAGE + CART + REAG + ANDR,data = seawatch[-ex.obs,])
```

```r
## check multicolinearity
vif(pop.lm)
```

```
##     GROSS       MOY       CPI    HHMEDI     POVPR     MFGPR      MAGE
##  4.153757  1.049295  1.275316  2.534493  2.855746  1.470281  1.322949
##      CART      REAG      ANDR
##  5.099883 15.065419 29.754043
```

```r
## drop ANDR
pop.lm<-update(pop.lm,.~.-ANDR)
vif(pop.lm)
```

```
##    GROSS      MOY      CPI   HHMEDI    POVPR    MFGPR     MAGE     CART
## 1.729190 1.048175 1.144997 2.474438 2.671959 1.445496 1.307185 4.138947
##     REAG
## 3.836808
```

```r
## summary and plot of the model
summary(pop.lm)
```

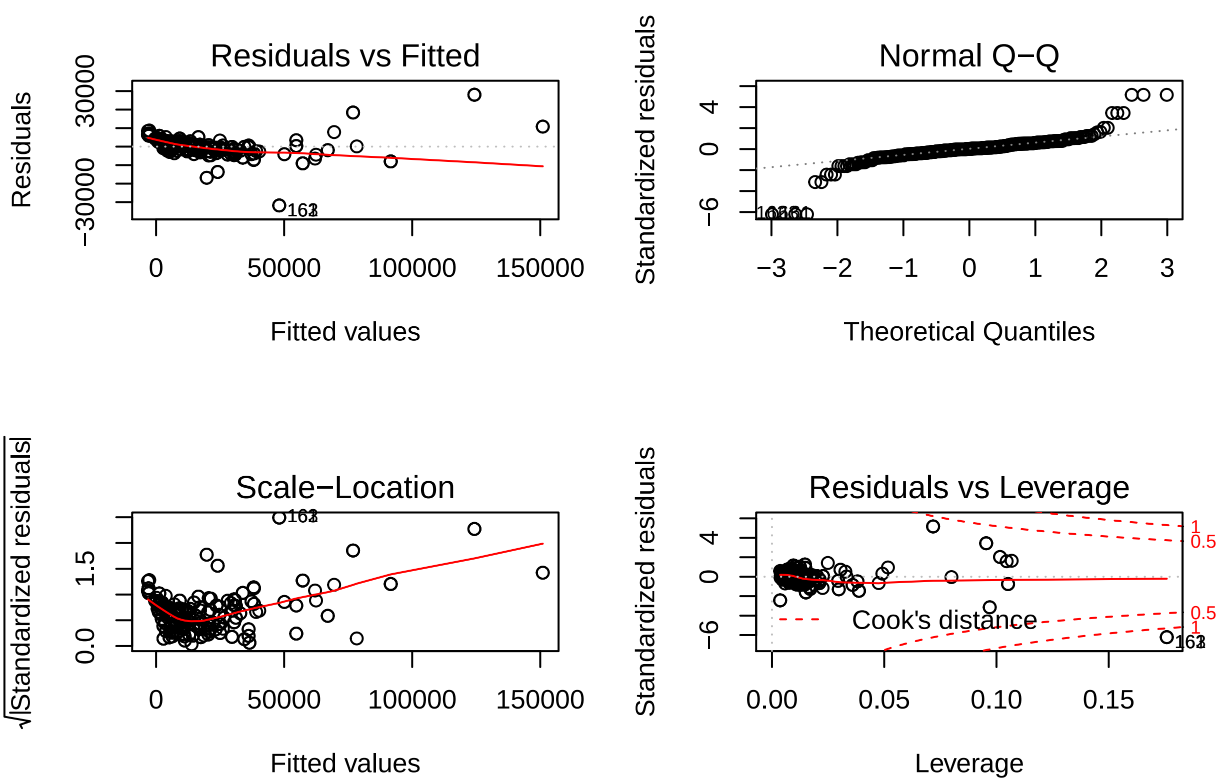
```
##
## Call:
## lm(formula = POP80 ~ GROSS + MOY + CPI + HHMEDI + POVPR + MFGPR +
##     MAGE + CART + REAG, data = seawatch[-ex.obs, ])
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32491  -1937    114   2064  27325
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  855.88035 7263.72590   0.118  0.90627
## GROSS         -0.09501    0.09354  -1.016  0.31047
## MOY         -206.87222  125.20657  -1.652  0.09938 .
## CPI           16.50456   18.91713   0.872  0.38355
## HHMEDI         0.03962    0.06937   0.571  0.56829
## POVPR        938.76317  122.01284   7.694 1.46e-13 ***
## MFGPR        108.83178   36.95402   2.945  0.00344 **
## MAGE        -502.76916  110.16717  -4.564 6.96e-06 ***
## CART           1.56789    0.11747  13.347  < 2e-16 ***
## REAG           4.19580    0.16951  24.753  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5626 on 351 degrees of freedom
## Multiple R-squared:  0.942,  Adjusted R-squared:  0.9405
## F-statistic: 632.9 on 9 and 351 DF,  p-value: < 2.2e-16
```

```r
## drop GROSS, MOY, CPI and HHMEDI
pop.lm<-update(pop.lm,.~.-GROSS-MOY-CPI-HHMEDI)
summary(pop.lm)
```

```
##
## Call:
## lm(formula = POP80 ~ POVPR + MFGPR + MAGE + CART + REAG, data = seawatch[-ex.obs,
##     ])
```

```
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -31778  -2049    114   2375  28029
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5141.8484  4034.3412   1.275   0.2033
## POVPR        914.1322    84.1918  10.858  < 2e-16 ***
## MFGPR        115.1605    33.6754   3.420   0.0007 ***
## MAGE        -508.0712   107.4020  -4.731 3.24e-06 ***
## CART           1.5629     0.1140  13.711  < 2e-16 ***
## REAG           4.1345     0.1648  25.096  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5632 on 355 degrees of freedom
## Multiple R-squared:  0.9412, Adjusted R-squared:  0.9403
## F-statistic:  1136 on 5 and 355 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(pop.lm)
```

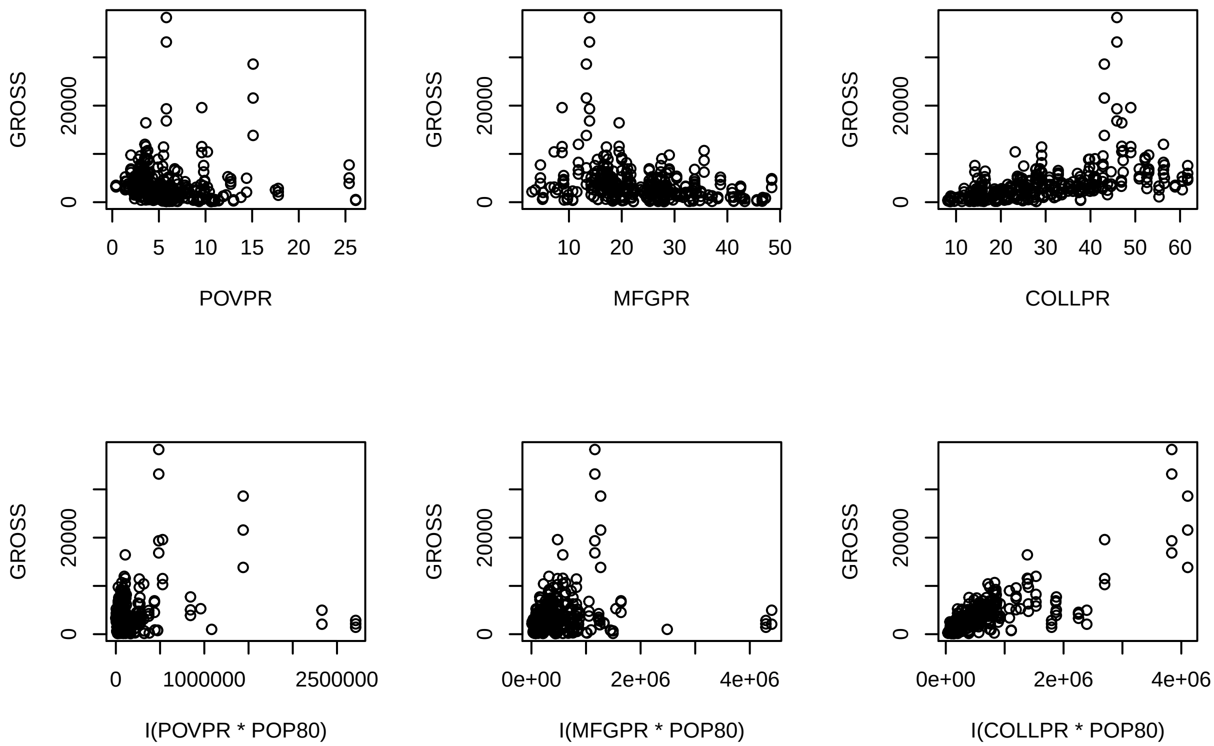



```
## predict the total population
pred<-predict(pop.lm,seawatch[ex.obs,])
seawatch[ex.obs,"POP80"]<-pred
```

POVPR,COLLPR and MFGPR

- Due to the increasing variance, instead of percentage, we transform those varibles to exact number by mutiplying total population. As a result, the linear correlations is more obvious.

```
#Compare variance before and after transformation
par(mfrow=c(2,3))
plot(GROSS~POVPR,data = seawatch)
plot(GROSS~MFGPR,data = seawatch)
plot(GROSS~COLLPR,data = seawatch)
plot(GROSS~I(POVPR*POP80),data = seawatch)
plot(GROSS~I(MFGPR*POP80),data = seawatch)
plot(GROSS~I(COLLPR*POP80),data = seawatch)
```




```
# Add POVPP,MFGPP,COLLPP to data
seawatch$POVPP<-seawatch$POVPR*seawatch$POP80
seawatch$MFGPP<-seawatch$MFGPR*seawatch$POP80
seawatch$COLLPP<-seawatch$COLLPR*seawatch$POP80
```

Modeling

Training and Testing subsets split

```
set.seed(1024)
train.num<-sample(1:dim(seawatch)[1],round(nrow(seawatch)*0.75))
```

```r
seawatch.train<-seawatch[train.num,]
seawatch.test<-seawatch[-train.num,]
```

Predictors Selections

```r
# full model
full.lm<-lm(data = seawatch.train,GROSS~.)

# predictors selection
step(full.lm,direction = "backward",trace = 0)
```

```
##
## Call:
## lm(formula = GROSS ~ MOY + YR + VISIT + LST + HHMEDI + CART +
##     REAG + ANDR + POVPP + COLLPP, data = seawatch.train)
##
## Coefficients:
## (Intercept)          MOY           YR       VISIT2       VISIT3
##  -5.801e+04    9.315e+01    6.935e+02   -1.625e+02    4.430e+02
##      VISIT4       VISIT5         LST1       HHMEDI         CART
##   2.590e+03    1.700e+04   -7.933e+02    6.632e-02   -3.096e-01
##        REAG         ANDR        POVPP       COLLPP
##  -4.408e-01    2.561e+00   -1.915e-03    3.714e-03
```

```r
# update model
fit.lm<-lm(formula = GROSS ~ MOY + YR + VISIT + LST + HHMEDI + CART +
    REAG + ANDR + POVPP + COLLPP, data = seawatch.train)

# check multicolinearity
vif(fit.lm)
```

```
##             GVIF Df GVIF^(1/(2*Df))
## MOY     1.182588  1        1.087469
## YR      3.647111  1        1.909741
## VISIT   3.691875  4        1.177351
## LST     2.093257  1        1.446809
## HHMEDI  2.119586  1        1.455880
## CART   11.627003  1        3.409839
## REAG   19.918715  1        4.463039
## ANDR   41.908423  1        6.473672
## POVPP   4.223934  1        2.055221
## COLLPP 18.224195  1        4.268981
```

```r
# drop ANDR
fit.lm<-update(fit.lm,.~.-ANDR)
vif(fit.lm)
```

```
##            GVIF Df GVIF^(1/(2*Df))
## MOY    1.169804  1        1.081575
## YR     3.386926  1        1.840360
## VISIT  3.483543  4        1.168833
## LST    2.057111  1        1.434263
## HHMEDI 1.656580  1        1.287082
## CART   9.243068  1        3.040241
```

```
## REAG   3.974517  1        1.993619
## POVPP  3.560529  1        1.886937
## COLLPP 5.917441  1        2.432579
```

```r
# drop CART
fit.lm<-update(fit.lm,.~.-CART)
vif(fit.lm)
```

```
##            GVIF Df GVIF^(1/(2*Df))
## MOY    1.169804  1        1.081575
## YR     3.268160  1        1.807805
## VISIT  3.211092  4        1.156995
## LST    2.048469  1        1.431247
## HHMEDI 1.651053  1        1.284933
## REAG   3.521195  1        1.876485
## POVPP  2.650217  1        1.627949
## COLLPP 2.842762  1        1.686049
```

```r
# summary
summary(fit.lm)
```

```
##
## Call:
## lm(formula = GROSS ~ MOY + YR + VISIT + LST + HHMEDI + REAG +
##     POVPP + COLLPP, data = seawatch.train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8740.9  -747.8  -177.7   700.2 10321.6
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -9.377e+03  1.717e+04  -0.546  0.58539
## MOY          6.628e+01  5.079e+01   1.305  0.19304
## YR           1.128e+02  2.081e+02   0.542  0.58823
## VISIT2       3.916e+02  3.193e+02   1.227  0.22106
## VISIT3       1.459e+03  4.404e+02   3.313  0.00105 **
## VISIT4       4.772e+03  6.767e+02   7.052 1.59e-11 ***
## VISIT5       2.079e+04  2.163e+03   9.611  < 2e-16 ***
## LST1        -7.261e+02  3.343e+02  -2.172  0.03074 *
## HHMEDI       1.641e-02  2.220e-02   0.739  0.46042
## REAG         7.518e-02  6.352e-02   1.184  0.23763
## POVPP       -2.824e-03  5.347e-04  -5.282 2.70e-07 ***
## COLLPP       4.558e-03  2.829e-04  16.114  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1898 on 260 degrees of freedom
## Multiple R-squared:  0.8164, Adjusted R-squared:  0.8086
## F-statistic: 105.1 on 11 and 260 DF,  p-value: < 2.2e-16
```

```r
# drop YR,POVPR,PERCAPI
fit.lm<-update(fit.lm,.~.-YR-MOY-HHMEDI-REAG)
summary(fit.lm)
```

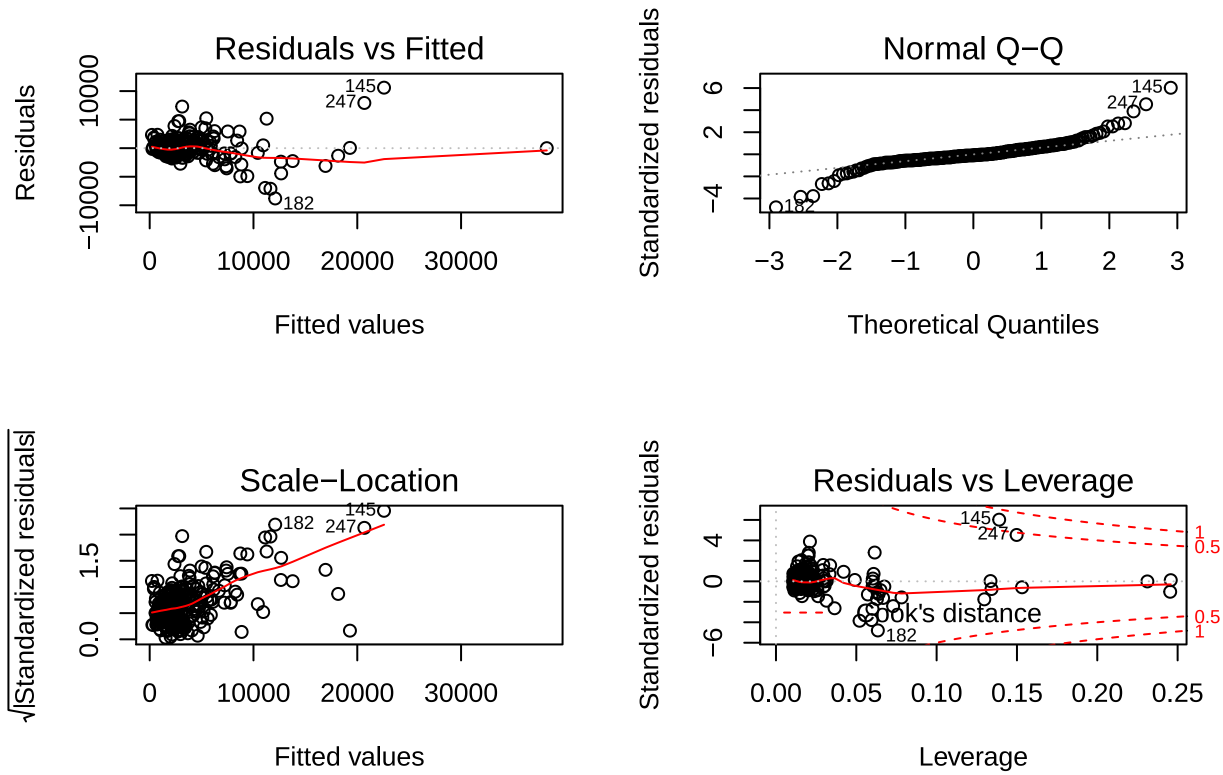
```
##
```

```
## Call:
## lm(formula = GROSS ~ VISIT + LST + POVPP + COLLPP, data = seawatch.train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8804.8  -785.8  -157.8   762.9 10598.9
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.253e+02  2.156e+02   3.364 0.000882 ***
## VISIT2       5.104e+02  2.767e+02   1.845 0.066217 .
## VISIT3       1.644e+03  3.577e+02   4.597 6.64e-06 ***
## VISIT4       4.929e+03  5.648e+02   8.726 3.03e-16 ***
## VISIT5       2.128e+04  2.068e+03  10.292  < 2e-16 ***
## LST1        -6.765e+02  2.774e+02  -2.439 0.015398 *
## POVPP       -2.618e-03  4.053e-04  -6.460 5.01e-10 ***
## COLLPP       4.740e-03  2.299e-04  20.617  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1896 on 264 degrees of freedom
## Multiple R-squared:  0.8141, Adjusted R-squared:  0.8091
## F-statistic: 165.1 on 7 and 264 DF,  p-value: < 2.2e-16
```

```r
# residual analysis
par(mfrow=c(2,2))
plot(fit.lm)
```

```
## Warning: not plotting observations with leverage one:
##   152
```

```
## Warning: not plotting observations with leverage one:
##   152
```



- There exists clear non-constant variance. Also predictors are not normally distributed. As a result, we use the power transformation to modify the model.

Power Transformation model

```r
powerTransform(cbind(seawatch.train$GROSS,seawatch$POVPR,seawatch$COLLPP)~1)
```

```
## Warning in cbind(seawatch.train$GROSS, seawatch$POVPR, seawatch$COLLPP):
## number of rows of result is not a multiple of vector length (arg 1)
```

```
## Estimated transformation parameters
##          Y1          Y2          Y3
##  0.12458545  0.10717096 -0.05498497
```

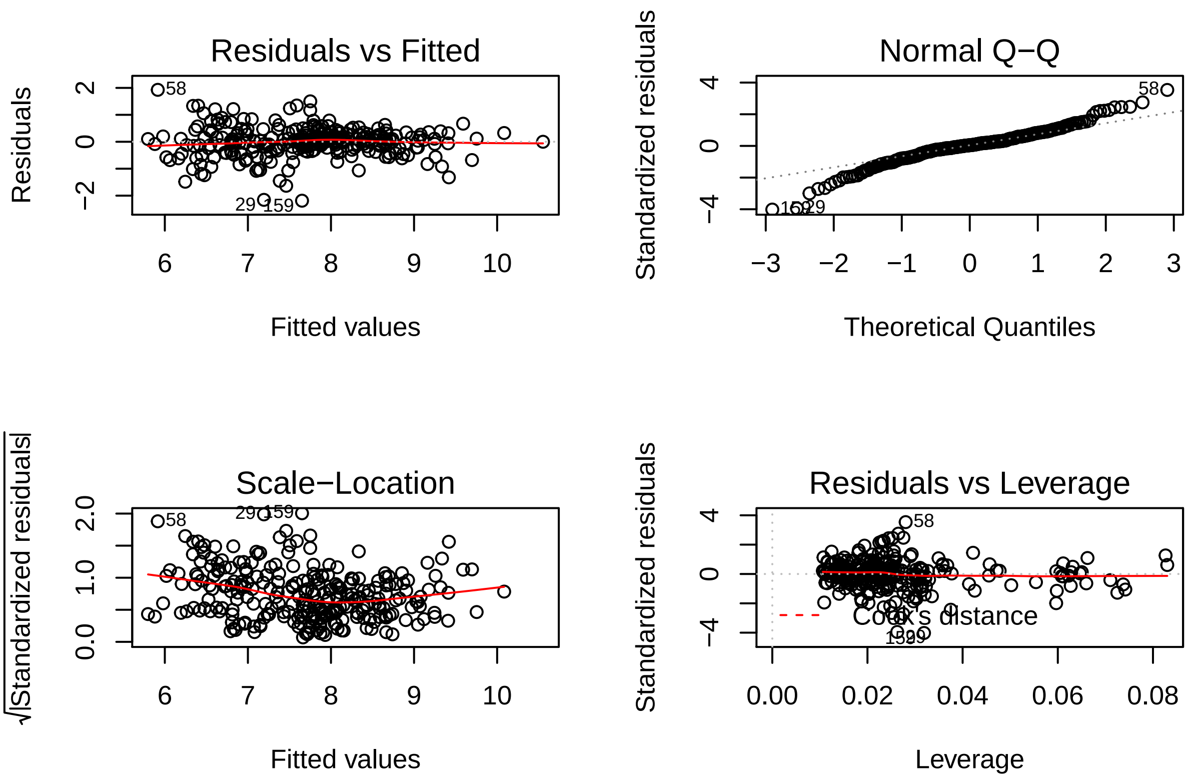
```r
# Thus, we build another model by taking natural log on both sides
new.fit.lm<-lm(formula = log(GROSS) ~ VISIT + LST + log(POVPP) + log(COLLPP), data = seawatch.train)
summary(new.fit.lm)
```

```
##
## Call:
## lm(formula = log(GROSS) ~ VISIT + LST + log(POVPP) + log(COLLPP),
##     data = seawatch.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.18775 -0.22711  0.02269  0.27381  1.92732
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.93422    0.47035  -1.986  0.04804 *
## VISIT2       0.17483    0.08055   2.170  0.03087 *
## VISIT3       0.51467    0.10682   4.818 2.45e-06 ***
## VISIT4       0.84580    0.16582   5.101 6.47e-07 ***
## VISIT5       1.57043    0.57016   2.754  0.00629 **
## LST1        -0.25719    0.08260  -3.114  0.00205 **
## log(POVPP)  -0.27453    0.03869  -7.097 1.18e-11 ***
## log(COLLPP)  0.90809    0.04800  18.919  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5531 on 264 degrees of freedom
## Multiple R-squared:  0.7195, Adjusted R-squared:  0.7121
## F-statistic: 96.76 on 7 and 264 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(new.fit.lm)
```

```
## Warning: not plotting observations with leverage one:
##   152

## Warning: not plotting observations with leverage one:
##   152
```



- Note that the variance of error is more constant and the predictors are distributed better than the original linear model

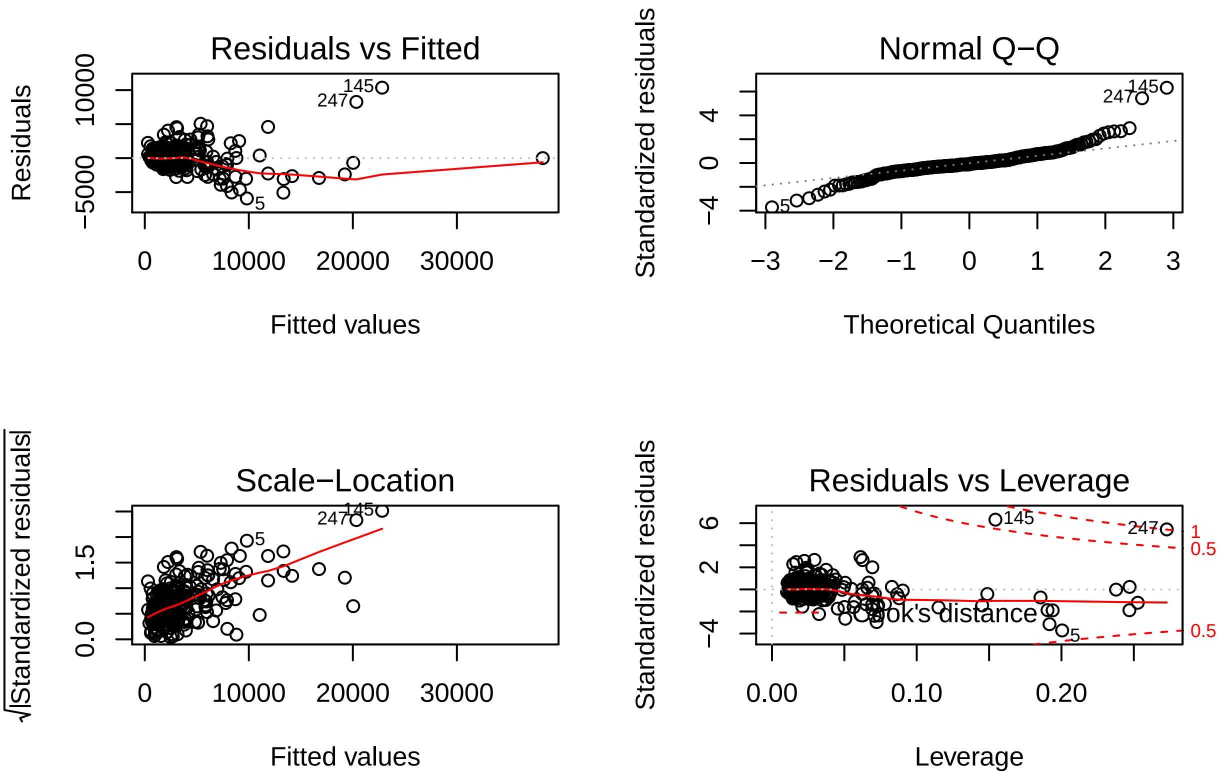
Model based on correlation

```r
model2<- lm(GROSS ~ VISIT + POP80 + PERCAPI + MFGPR + REAG + POVPP + COLLPP,data = seawatch.train)
summary(model2)
```

```
##
## Call:
## lm(formula = GROSS ~ VISIT + POP80 + PERCAPI + MFGPR + REAG +
##     POVPP + COLLPP, data = seawatch.train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5930.0  -750.4  -166.7   698.2 10360.4
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.502e+02  6.698e+02  -0.224 0.822673
## VISIT2       3.314e+02  2.591e+02   1.279 0.202013
## VISIT3       1.137e+03  2.996e+02   3.795 0.000184 ***
## VISIT4       3.919e+03  5.029e+02   7.792 1.57e-13 ***
## VISIT5       1.936e+04  1.918e+03  10.092  < 2e-16 ***
## POP80       -2.365e-01  3.756e-02  -6.296 1.29e-09 ***
## PERCAPI      8.191e-03  6.120e-02   0.134 0.893638
## MFGPR        2.078e+01  1.296e+01   1.603 0.110085
## REAG         7.606e-01  1.248e-01   6.094 3.94e-09 ***
## POVPP        4.770e-03  1.284e-03   3.714 0.000249 ***
## COLLPP       6.293e-03  3.888e-04  16.186  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1784 on 261 degrees of freedom
## Multiple R-squared:  0.8372, Adjusted R-squared:  0.831
## F-statistic: 134.3 on 10 and 261 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(model2)
```

```
## Warning: not plotting observations with leverage one:
##   152

## Warning: not plotting observations with leverage one:
##   152
```




```
#multicolinearity check
vif(model2)
```

```
##                GVIF Df GVIF^(1/(2*Df))
## VISIT      1.343200  4        1.037570
## POP80     65.917474  1        8.118958
## PERCAPI    1.590806  1        1.261272
## MFGPR      1.324058  1        1.150677
## REAG      15.398087  1        3.924040
## POVPP     17.316278  1        4.161283
## COLLPP     6.080990  1        2.465966
```

Cross Validation

```
#MSE function
MSE<-function(pred,actual){
  return(mean((pred-actual)^2))
}

#predictions based on each model
pred.fit<-predict(fit.lm,newdata = seawatch.test)
pred.fit.new<-predict(new.fit.lm,newdata = seawatch.test)
pred.model2<-predict(model2,newdata = seawatch.test)

#MSE table
```

```
data.frame(
  Model=c("fit.lm","fit.powertrans","model2"),
  MSE=c(MSE(pred.fit,seawatch.test$GROSS),MSE(exp(pred.fit.new),seawatch.test$GROSS),MSE(pred.model2,sea
)
```

```
##             Model     MSE
## 1          fit.lm 8248999
## 2 fit.powertrans 5771242
## 3          model2 8407178
```

- The model with lowest MSE so far is the log-transformation model.