sta302 final project

shimmy

6/2/2021

setting up the data and the proper libraries

```r
#libraries and given code
rm(list = ls())
setwd("C:/Users/shimm/OneDrive - University of Toronto/second_year/summer first semester/sta302/final pr
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------ tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.2     v dplyr   1.0.0
## v tidyr   1.1.1     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts --------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(NHANES)
```

```
## Warning: package 'NHANES' was built under R version 4.0.5
```

```r
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some
```

```r
library(olsrr)
```

```
##
## Attaching package: 'olsrr'

## The following object is masked from 'package:datasets':
##
##     rivers
```

```r
library(graphics)
library(psych)
```

```
##
## Attaching package: 'psych'

## The following object is masked from 'package:car':
##
##     logit

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
library(glmnet)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loaded glmnet 4.0-2
```

```r
library(rms)
```

```
## Loading required package: Hmisc

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##
## Attaching package: 'Hmisc'
```

```
## The following object is masked from 'package:psych':
##
##     describe
```

```
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
## Loading required package: SparseM
```

```
##
## Attaching package: 'SparseM'
```

```
## The following object is masked from 'package:base':
##
##     backsolve
```

```
##
## Attaching package: 'rms'
```

```
## The following objects are masked from 'package:car':
##
##     Predict, vif
```

```r
small.nhanes <- na.omit(NHANES[NHANES$SurveyYr=="2011_12"
& NHANES$Age > 17,c(1,3,4,8:11,13,17,20,21,25,46,50,51,52,61)])
small.nhanes <- as.data.frame(small.nhanes %>%
group_by(ID) %>% filter(row_number()==1) )
nrow(small.nhanes)
```

```
## [1] 743
```

```r
## Checking whether there are any ID that was repeated. If not ##
## then length(unique(small.nhanes$ID)) and nrow(small.nhanes) are same ##
length(unique(small.nhanes$ID))
```

```
## [1] 743
```

```r
1
```

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

```r
set.seed(1005476995)
train <- small.nhanes[sample(seq_len(nrow(small.nhanes)), size = 500),]
nrow(train)
```

```
## [1] 500
```

```r
length(which(small.nhanes$ID %in% train$ID))
```

```
## [1] 500
```

```r
test <- small.nhanes[!small.nhanes$ID %in% train$ID,]
nrow(test)
```

```
## [1] 243
```

```r
train_minus_id <- train[,2:17]
test_minus_id <- test[,2:17]
```

creating valuable functions

```r
f_multi_minus_vif <- function(m){
    h <- hatvalues(m)
   thresh_hold <- 2*(dim(model.matrix(m))[2])/nrow(m$model)
   w <- which(h > thresh_hold)
   print("leverage")
   print(w)

d <- cooks.distance(m)
cut <- which(d > qf(.5,
                    df1 =  ncol(m$model[, -c(1)]) + 1,
                    df2 = nrow(m$model[, -c(1)])-ncol(m$model[, -c(1)]) - 1)
                    )

print("cut_d")
print(cut)

dfits<- dffits(m)
cut_fits <- which(
  abs(dfits) > 2*sqrt((ncol(m$model[, -c(1)]) + 1)/nrow(m$model[, -c(1)]))
  )
print("cut_fits")
print(cut_fits)


df_b <- dfbetas(m)
cut_b <- which(
  abs(df_b[,1]) > 2/sqrt(nrow(m$model[, -c(1)]))

  )
print("cut_beta")
print(cut_b)
print("lev + cut_b")
lev_cut_b <- intersect(w, cut_b)
print(lev_cut_b)
print("lev + cut_fits")
```

```
 lev_cut_fits<- intersect(w, cut_fits)
print(lev_cut_fits)
print(" lev + cut_d")
w_cut <- intersect(w, cut)
print(w_cut )
print("b + fits")
b_cut_fits <- intersect(cut_b, cut_fits)
print(b_cut_fits)
print("d + b")
d_b <-intersect(cut_b, cut)
print(d_b)
print("d + fits")
d_fits <- intersect(cut, cut_fits)
print(d_fits)
print(" all outliers intersect")
all_intersection <- intersect(intersect(cut, cut_fits), cut_b)
print(all_intersection )
ls <- list(lev_cut_b, lev_cut_fits, w_cut, b_cut_fits, d_b, d_fits, all_intersection)
print(psych::pairs.panels(m$model[, -c(1)], density = TRUE))
print(plot(m))

print(anova(m))
return(ls)

}

f_multi_diagnostic<- function(m){

ls <- f_multi_minus_vif(m)
v <- car::vif(m)
print("VIF")
print(v)

return(ls)
}
```

setting up the model with all variables and getting the error

```
model.all <- lm(BPSysAve ~., data = train_minus_id)
model.all.error <- mean((model.all$model$BPSysAve - model.all$fitted.values)^2)
model.all.error
```

```
## [1] 207.816
```

variable selection

lasso method

variable selection

```r
set.seed(1005476995)
cv.out <- cv.glmnet(x = data.matrix(train_minus_id[, -which(names( train_minus_id) == "BPSysAve")]), y =
#plot(cv.out)
best.lambda <- cv.out$lambda.1se
best.lambda
```

```
## [1] 3.710926
```

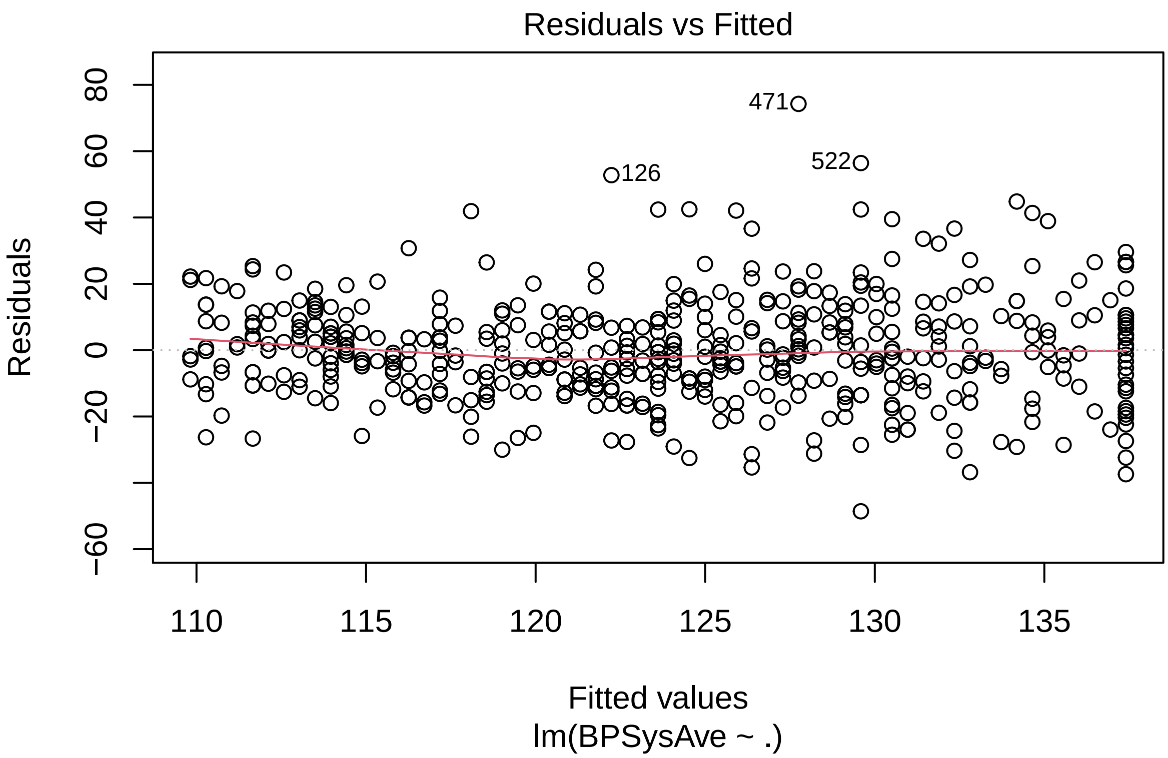
```r
co<-coef(cv.out, s = "lambda.1se")
co
```

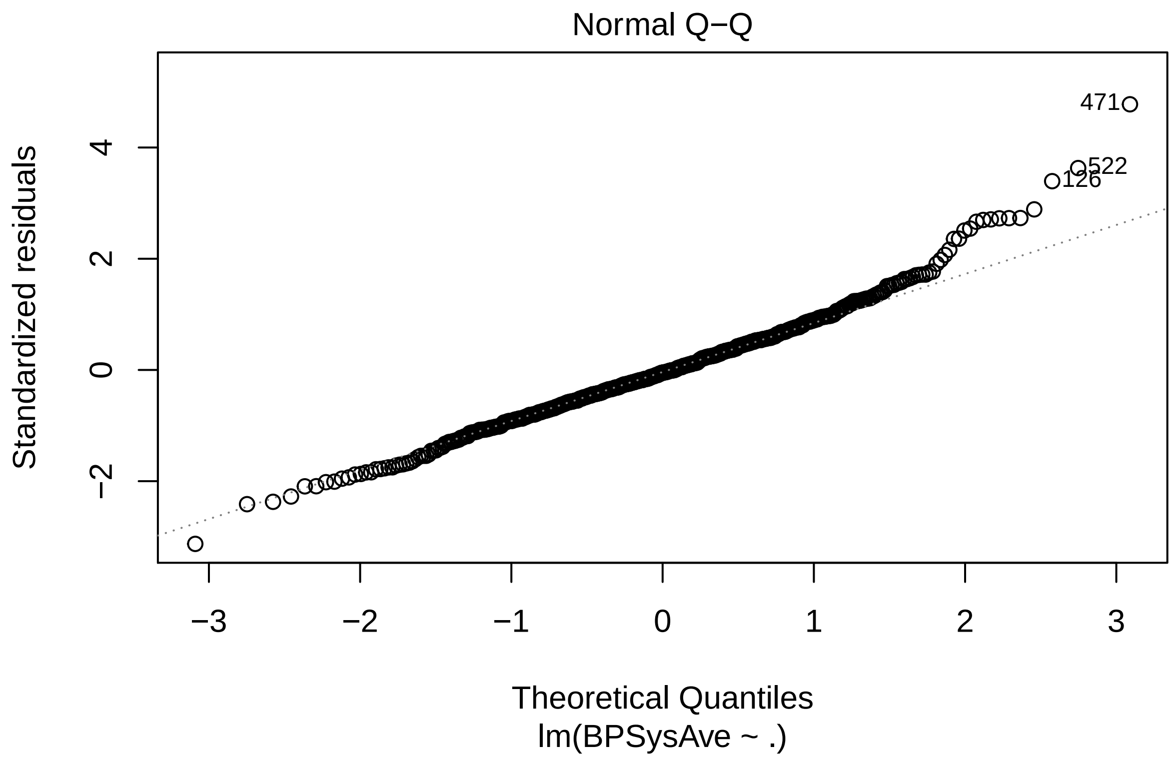
```
## 16 x 1 sparse Matrix of class "dgCMatrix"
##                         1
## (Intercept)   111.7508418
## Gender          .
## Age             0.2413249
## Race3           .
## Education       .
## MaritalStatus   .
## HHIncome        .
## Poverty         .
## Weight          .
## Height          .
## BMI             .
## Depressed       .
## SleepHrsNight   .
## SleepTrouble    .
## PhysActive      .
## SmokeNow        .
```

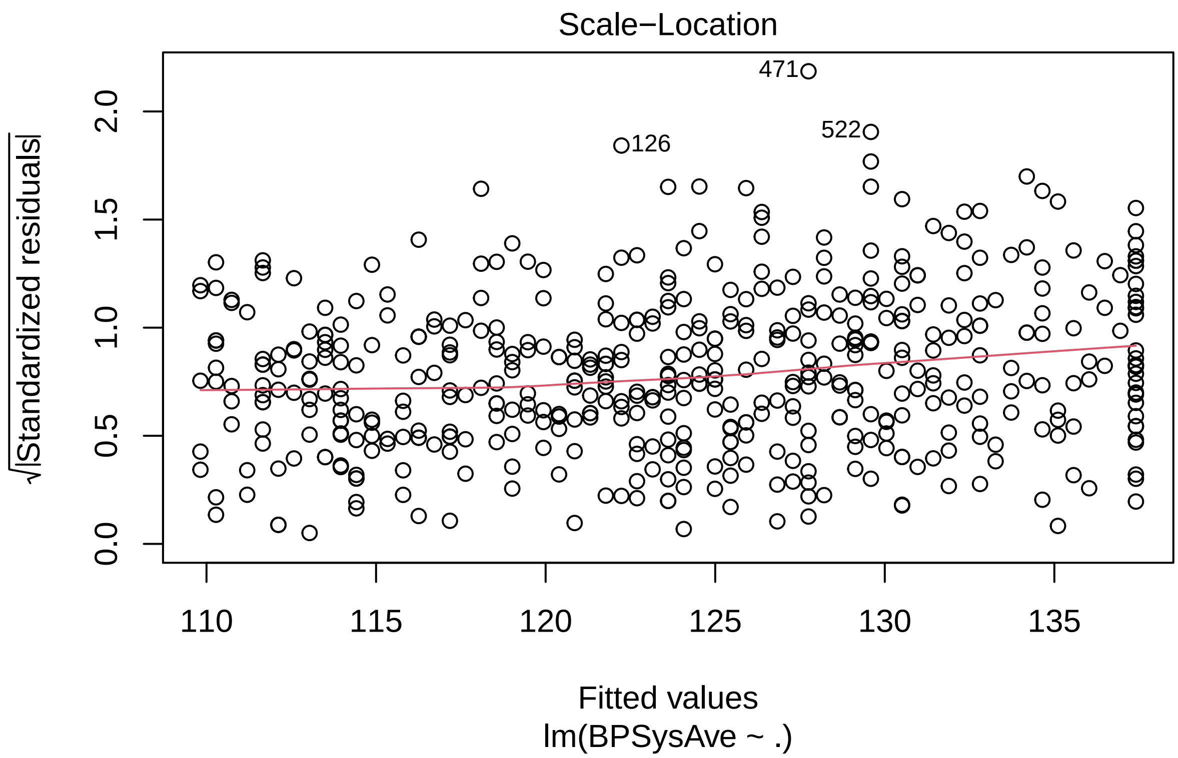
```r
#model.matrix
#wanna see how the preidction performance is in the trainng set
#choose use 10 for B, and find which is the best model. Lasso, Aic, BIC. There is no one answer.
#important to do cross validation.
# should only be used for accuract of final model


thresh <- 0.00
# select variables #
inds<-which(abs(co) > thresh )
variables<-row.names(co)[inds]
sel.var.lasso<-variables[!(variables %in% '(Intercept)')]
sel.var.lasso
```

```
## [1] "Age"
```

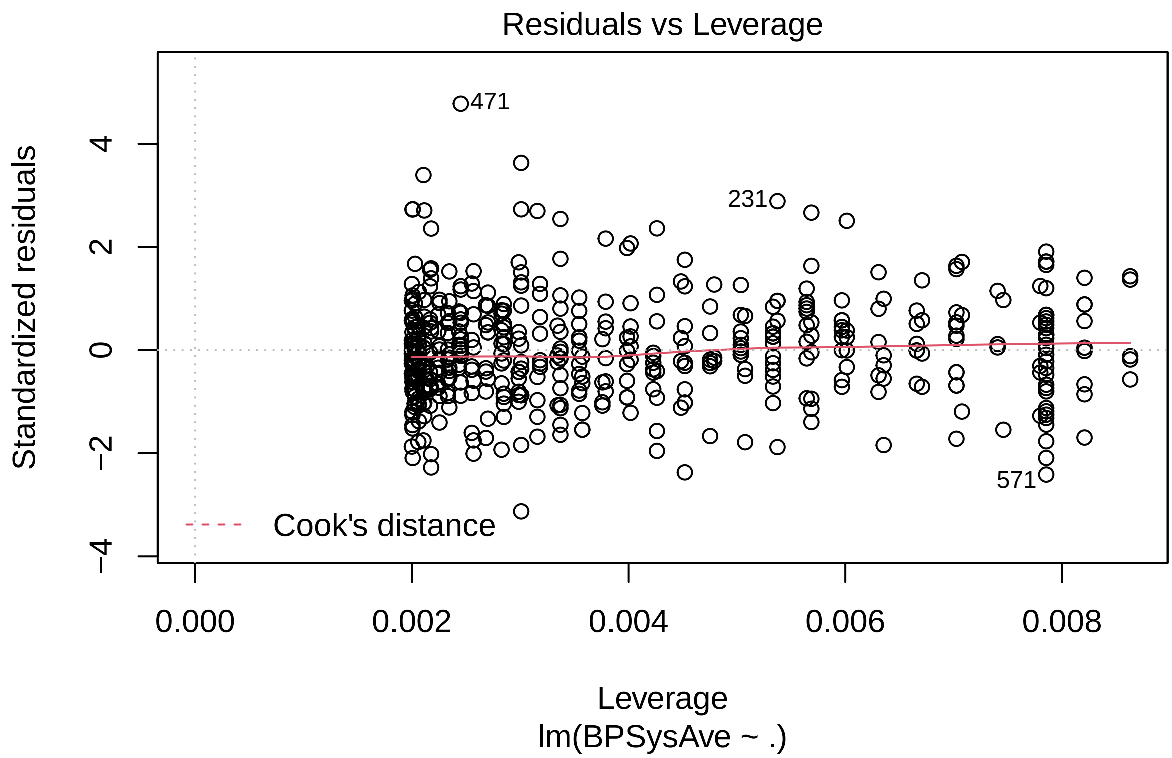
```r
model.lasso <- lm(BPSysAve ~., data = train_minus_id %>% select(Age, BPSysAve))
plot(model.lasso)
```


Residuals vs Fitted
Residuals
Fitted values
lm(BPSysAve ~ .)
80
60
40
20
0
−20
−60
110
115
120
125
130
135
471
522
126


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(BPSysAve ~ .)
471
522
126
4
2
0
−2
−3
−2
−1
0
1
2
3



Scale−Location
471
522
126
0.0
0.5
1.0
1.5
2.0
√|Standardized residuals|
110
115
120
125
130
135
Fitted values
lm(BPSysAve ~ .)

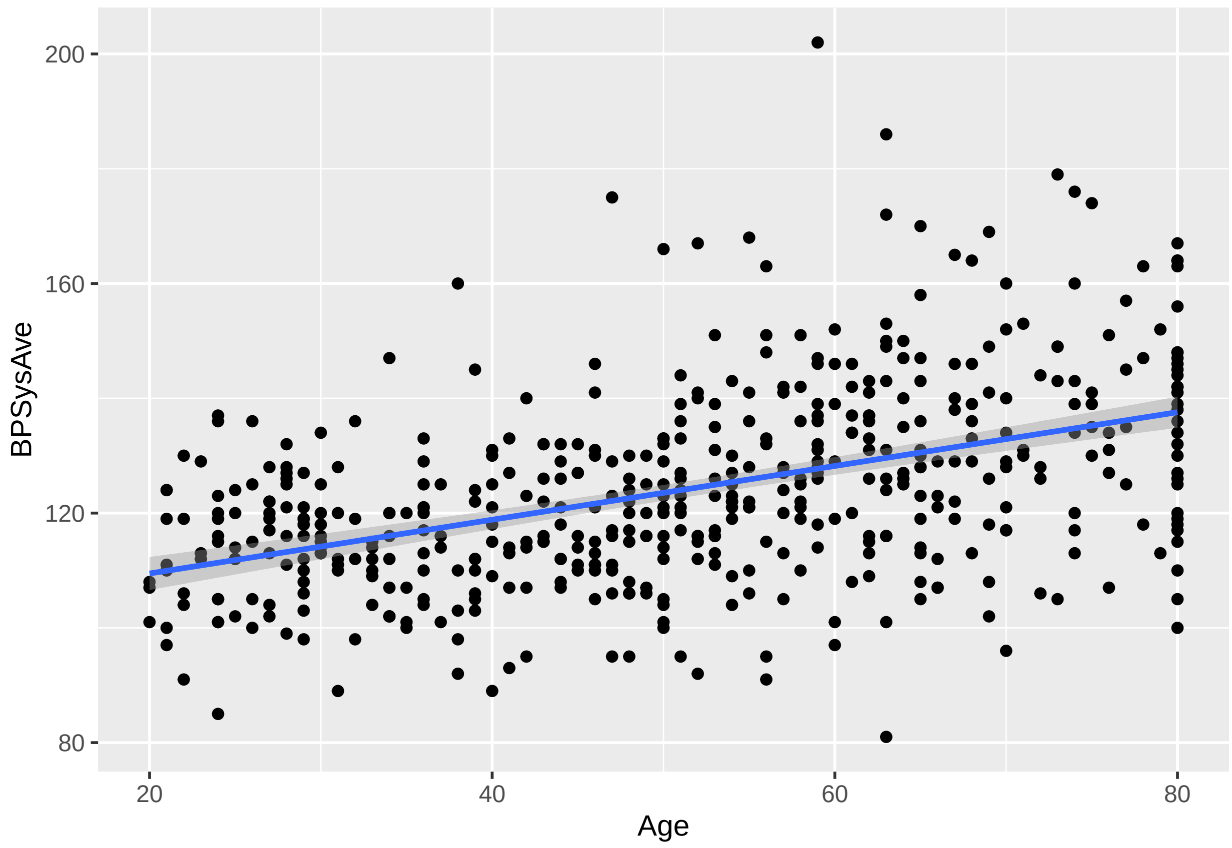



```r
summary(model.lasso)
```

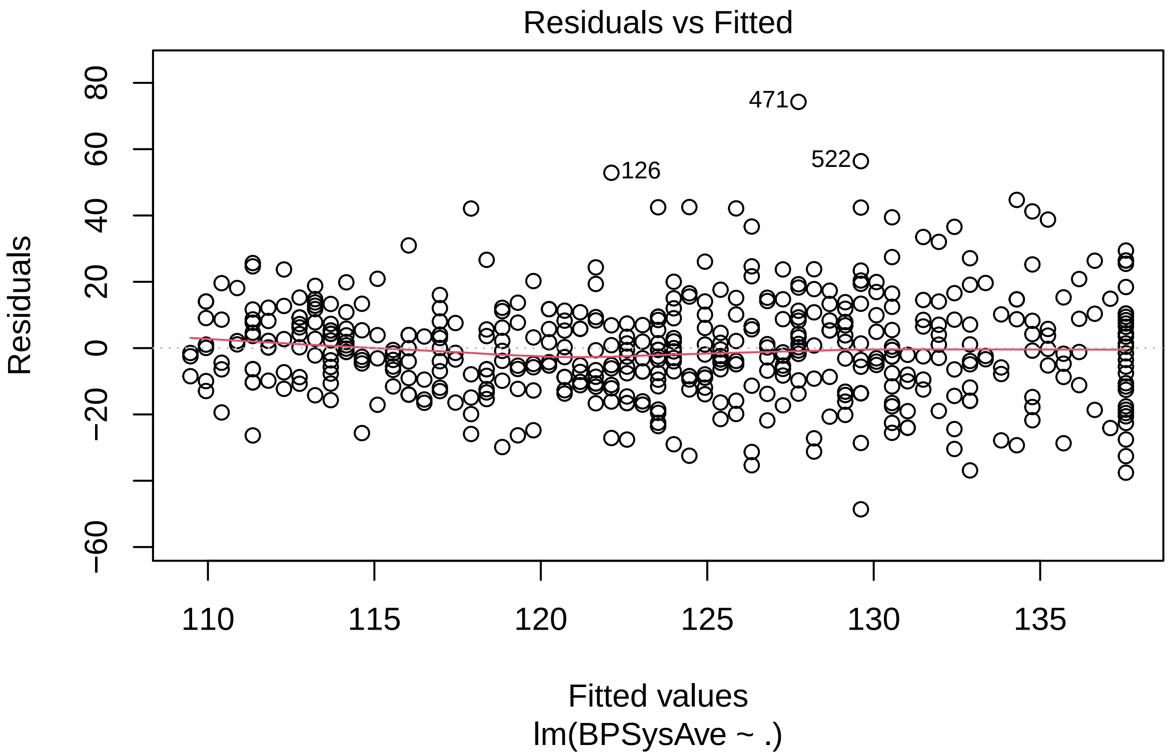
```
##
## Call:
## lm(formula = BPSysAve ~ ., data = train_minus_id %>% select(Age,
##     BPSysAve))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -48.590  -9.806  -0.763   8.666  74.249
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 100.62666    2.19910   45.76   <2e-16 ***
## Age           0.45974    0.04096   11.22   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.56 on 498 degrees of freedom
## Multiple R-squared:  0.2019, Adjusted R-squared:  0.2003
## F-statistic:   126 on 1 and 498 DF,  p-value: < 2.2e-16
```

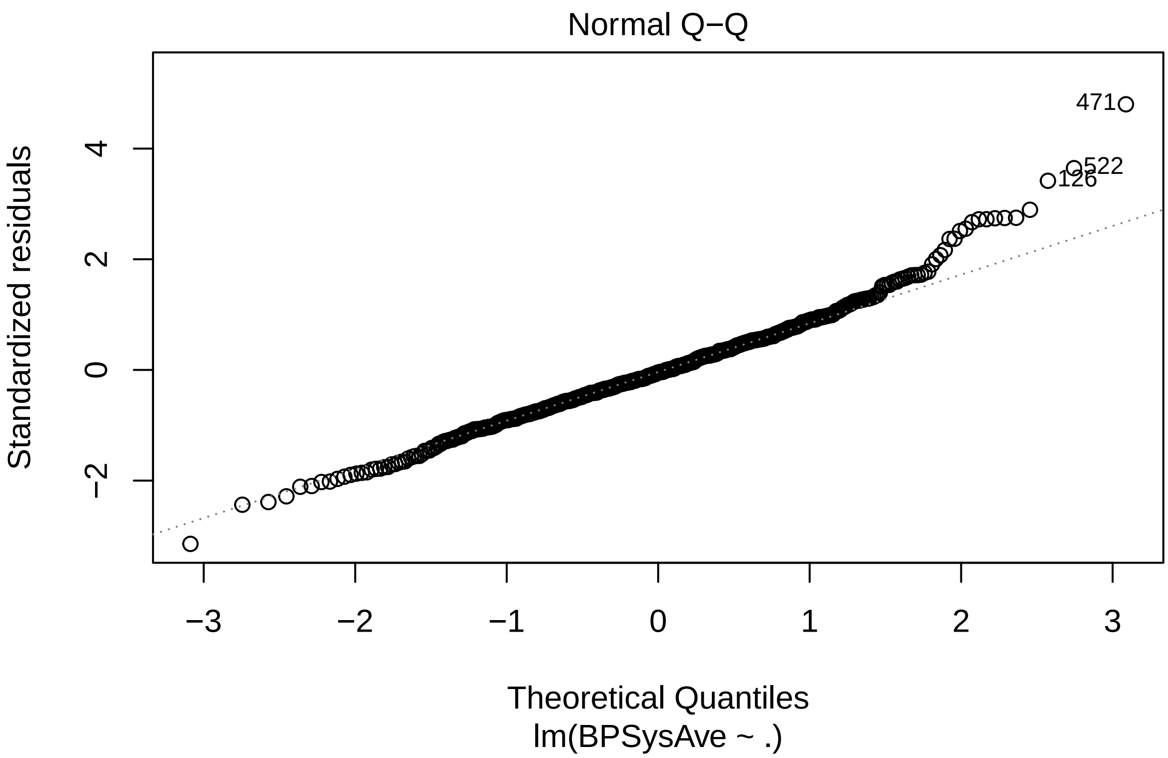
outliers

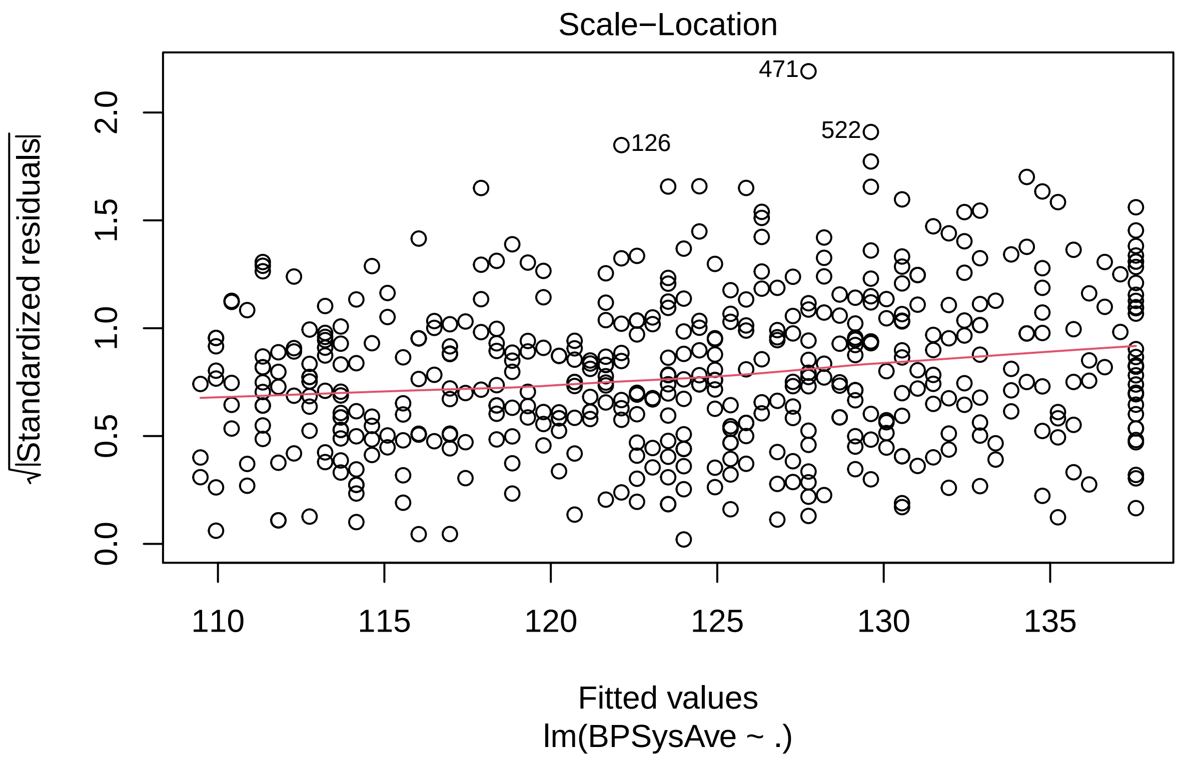
```
hii <- hatvalues(model.lasso)
leverage_point <- which(hii > 4/nrow(model.lasso$model))
cooks <- cooks.distance(model.lasso)
outliers <- which(cooks > 4/(nrow(model.lasso$model)-2))
lasso.outliers <- intersect(outliers, leverage_point)
ggplot(model.lasso$model[-lasso.outliers,],aes(y = BPSysAve, x = Age)) +
  geom_point() +
  geom_smooth(method='lm', formula= y~x)
```

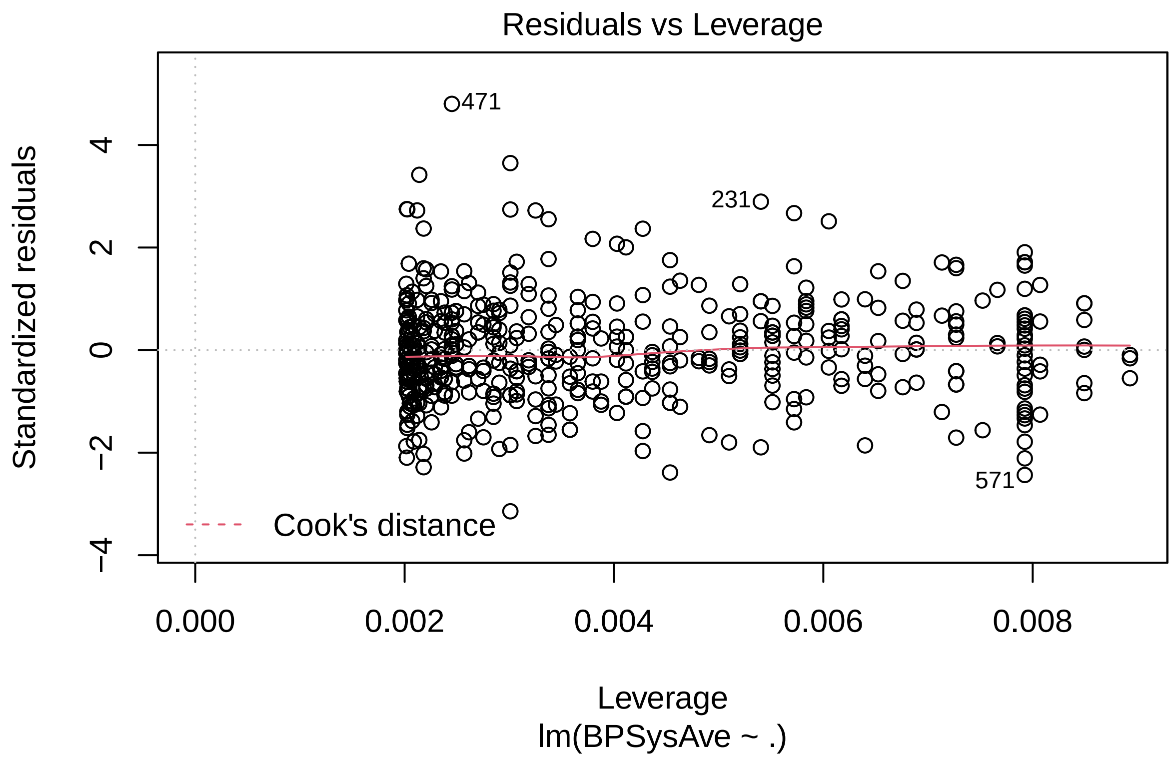



```
model.lasso.outliers <- lm(BPSysAve ~., data = train_minus_id[-lasso.outliers,] %>% select(Age, BPSysAve
plot(model.lasso.outliers)
```


Residuals vs Fitted
Residuals
80
60
40
20
0
−20
−60
471
522
126
110
115
120
125
130
135
Fitted values
lm(BPSysAve ~ .)

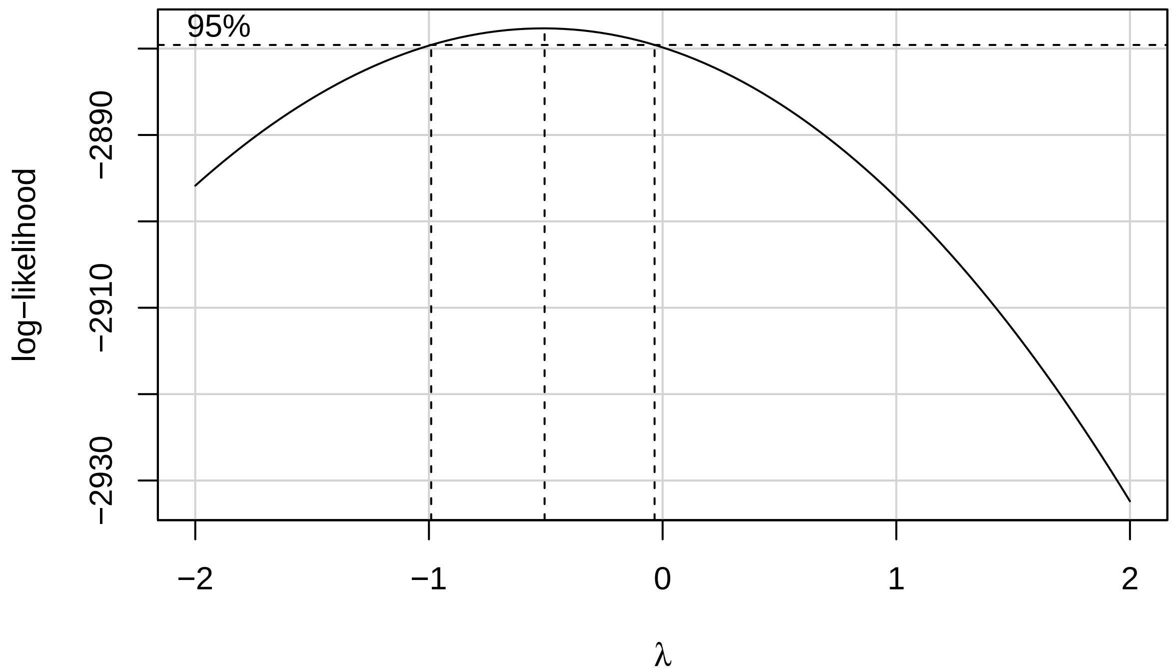

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(BPSysAve ~ .)
471
522
126
−3
−2
−1
0
1
2
3
4
2
0
−2


Scale−Location
471
522
126
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
110
115
120
125
130
135
Fitted values
lm(BPSysAve ~ .)

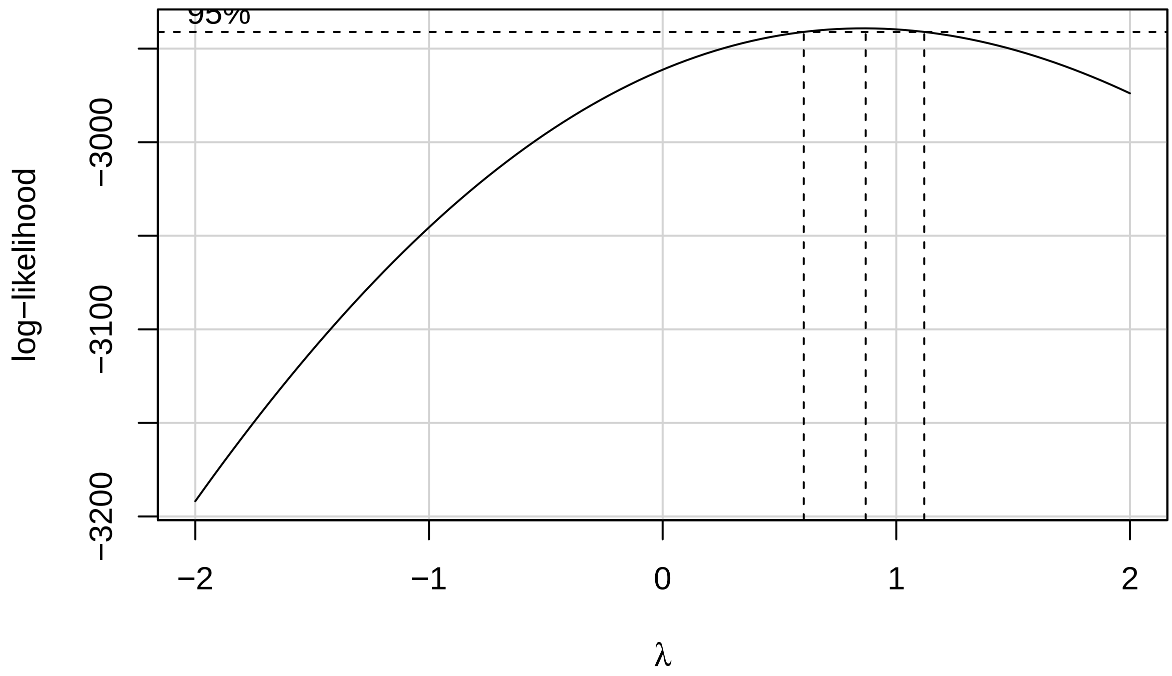


boxcox transformation

```
boxCox(model.lasso.outliers) # -.5
```

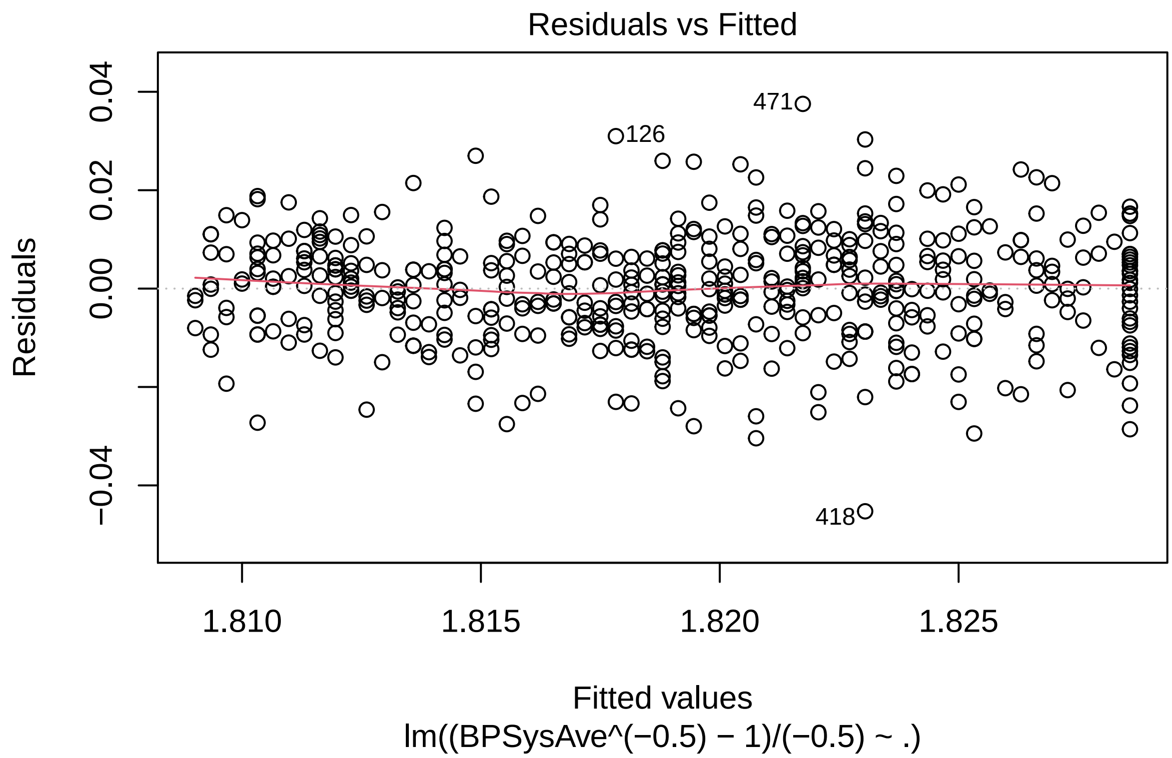


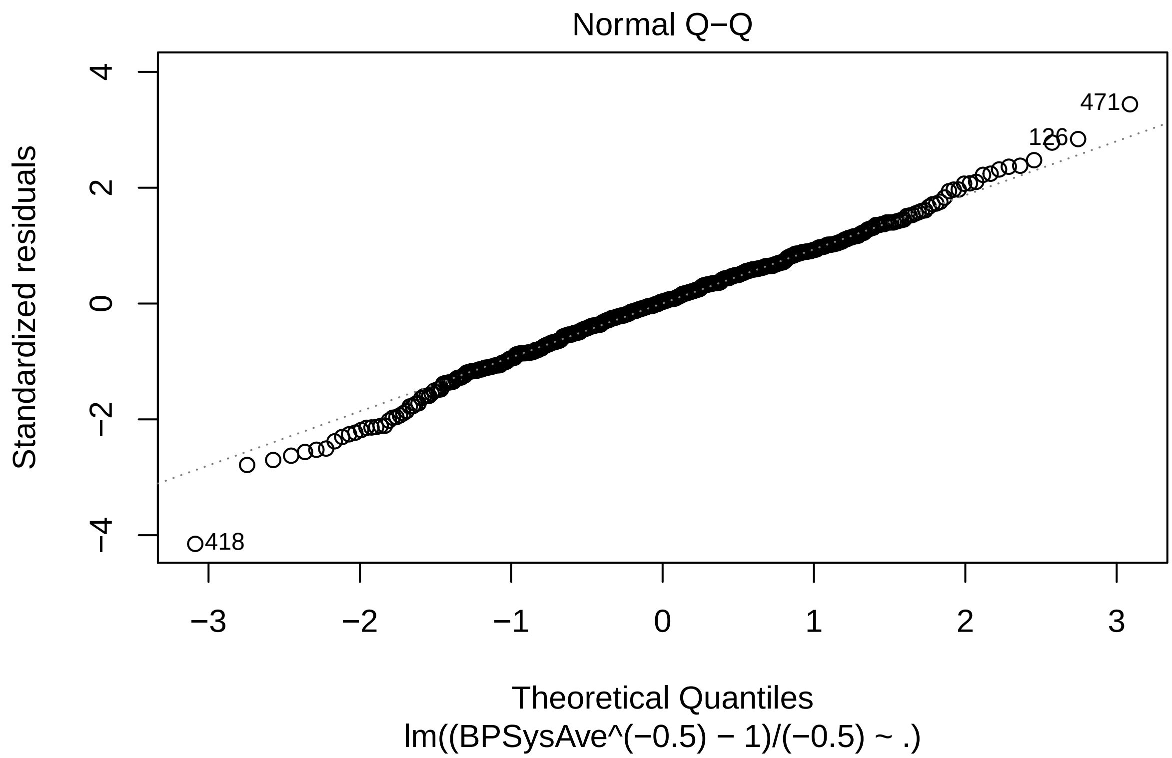

```r
boxCox(lm(Age ~ 1, data = train_minus_id[-lasso.outliers,] %>% select(Age, BPSysAve))) # approx 1
```

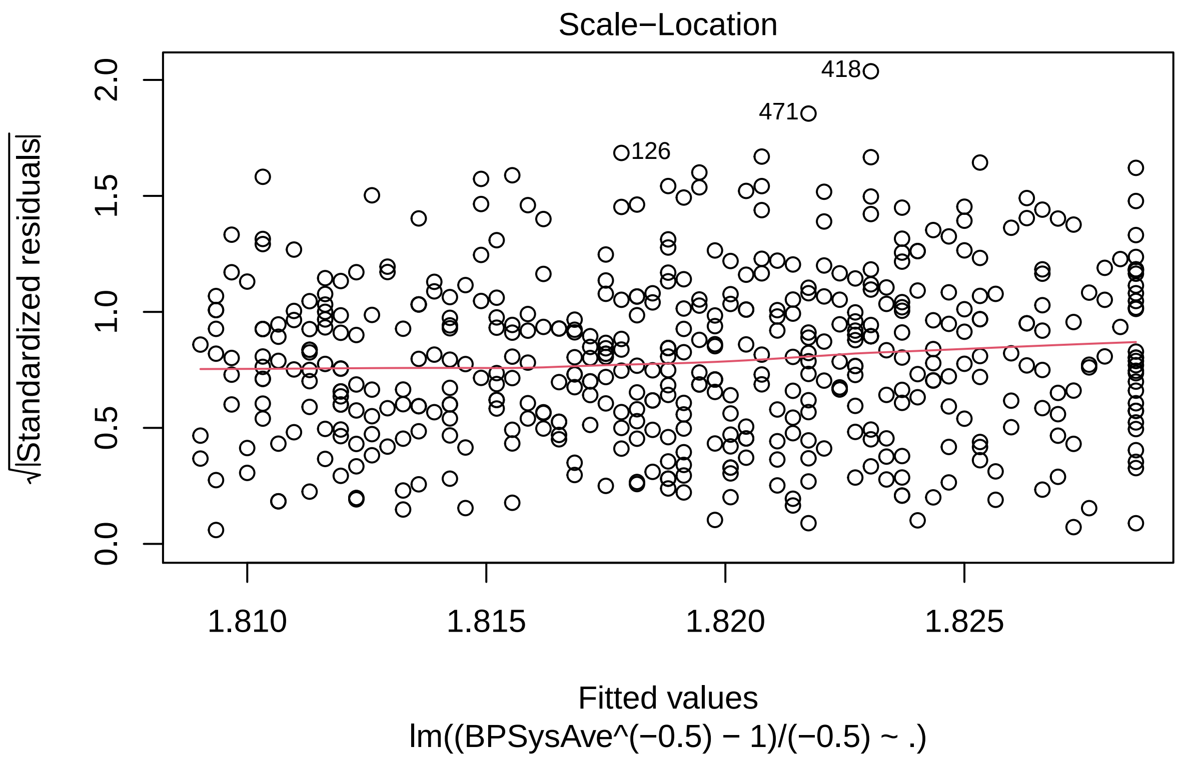


final model and calculations

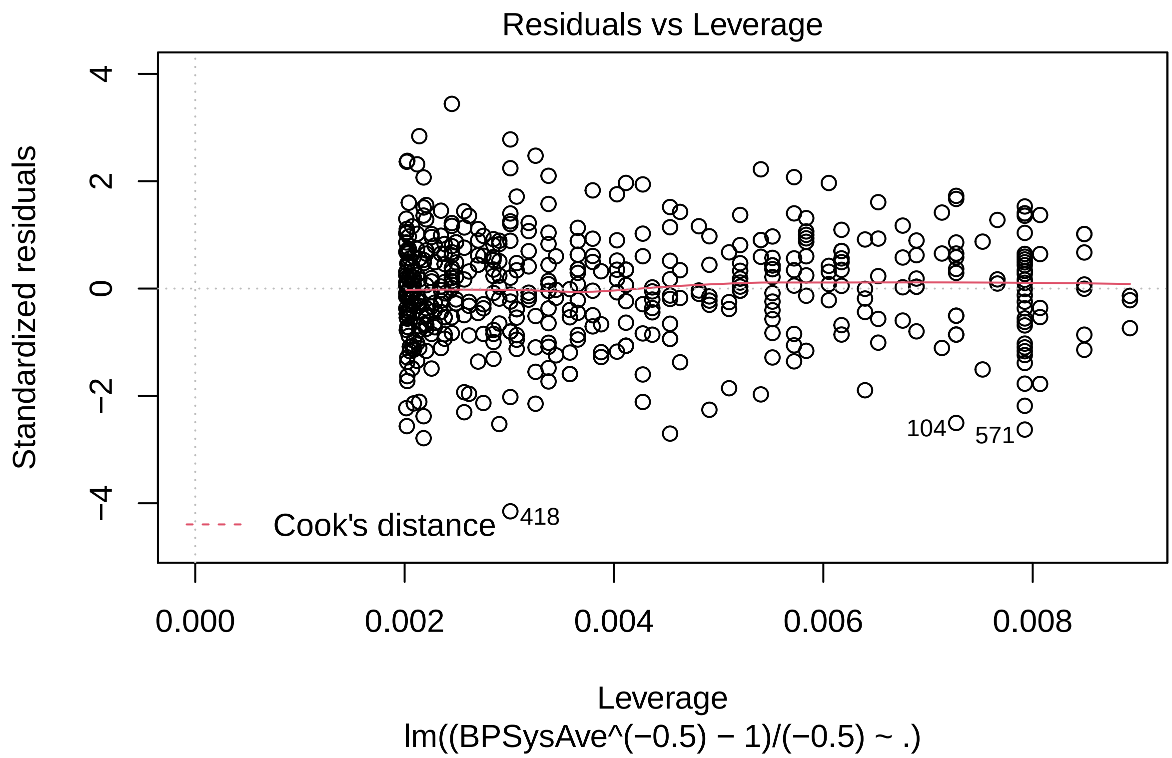
```
model.lasso.boxcox <- lm((BPSysAve^(-.5) - 1)/(-.5) ~., data = train_minus_id[-lasso.outliers,] %>% sel
plot(model.lasso.boxcox)
```


Residuals vs Fitted
Residuals
0.04
0.02
0.00
−0.04
471
126
418
1.810
1.815
1.820
1.825
Fitted values
lm((BPSysAve^(−0.5) − 1)/(−0.5) ~ .)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm((BPSysAve^(−0.5) − 1)/(−0.5) ~ .)
471
126
418
−3
−2
−1
0
1
2
3
4
2
0
−2
−4



Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
418
471
126
1.810
1.815
1.820
1.825
Fitted values
lm((BPSysAve^(−0.5) − 1)/(−0.5) ~ .)

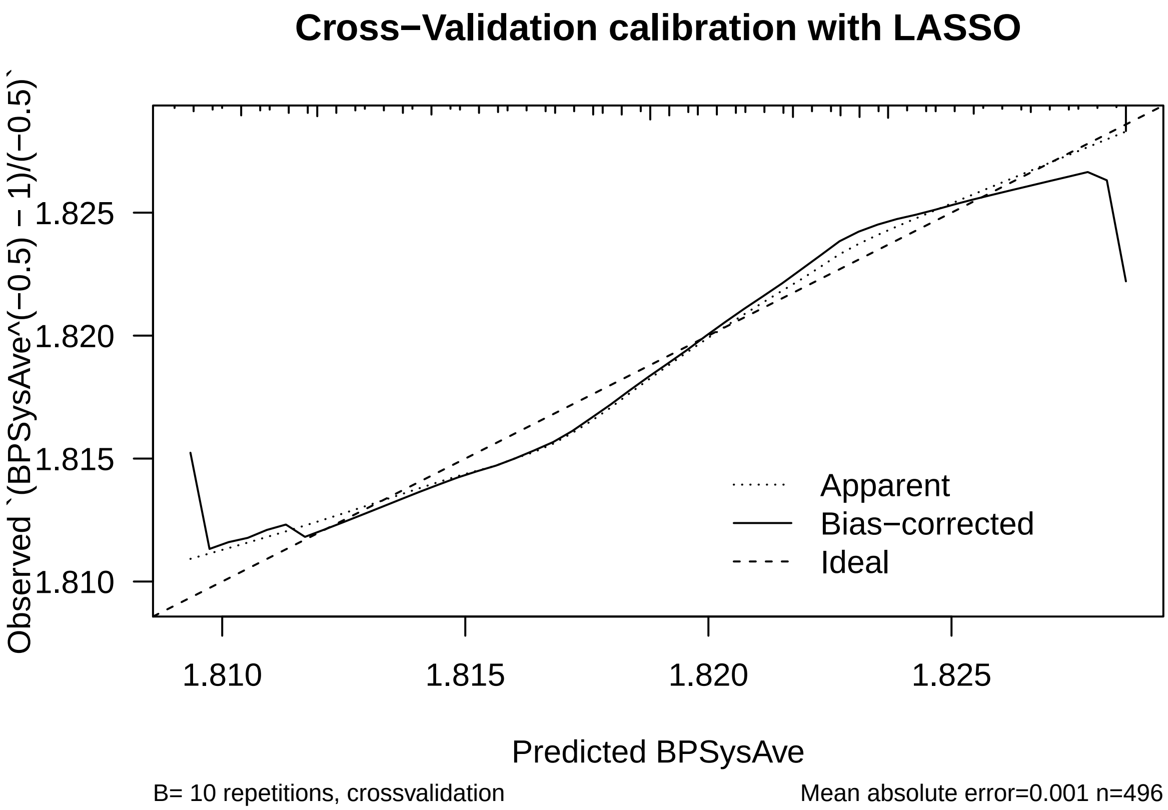



```
error.lasso <- mean((model.lasso.boxcox$model$`(BPSysAve^(-0.5) - 1)/(-0.5)` - model.lasso.boxcox$fitte
error.lasso
```

```
## [1] 0.000118855
```

cross validation and test error

```
### Cross Validation and prediction performance of lasso based selection ###
ols.lasso <- ols(`(BPSysAve^(-0.5) - 1)/(-0.5)` ~ ., data = model.lasso.boxcox$model,
                 x=T, y=T, model = T)

## 10 fold cross validation ##
lasso.cross <- calibrate(ols.lasso, method = "crossvalidation", B = 10)
## Calibration plot ##
#pdf("lasso_cross.pdf", height = 8, width = 16)
plot(lasso.cross, las = 1, xlab = "Predicted BPSysAve", main = "Cross-Validation calibration with LASSO
```




```
##
## n=496   Mean absolute error=0.001   Mean squared error=0
## 0.9 Quantile of absolute error=0.001
```

```
#dev.off()
test_minus_id.lasso.tranformation <- test_minus_id[-lasso.outliers,]
test_minus_id.lasso.tranformation$`(BPSysAve^(-0.5) - 1)/(-0.5)` <- (test_minus_id.lasso.tranformation$I




## Test Error ##
pred.lasso <- predict(ols.lasso, newdata = test_minus_id.lasso.tranformation)
## Prediction error ##
pred.error.lasso <- mean((test_minus_id.lasso.tranformation$`(BPSysAve^(-0.5) - 1)/(-0.5)` - pred.lasso)
```

aic model

variable selection

```
model.lm <- lm (BPSysAve~ ., data =  train_minus_id)
summary(model.lm)
```

```
##
## Call:
## lm(formula = BPSysAve ~ ., data = train_minus_id)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -44.816  -9.668  -0.754   9.019  61.150
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               175.32295   58.07097   3.019  0.00268 **
## Gendermale                  3.24539    2.01738   1.609  0.10836
## Age                         0.53500    0.05474   9.773  < 2e-16 ***
## Race3Black                  7.78296    4.18558   1.859  0.06360 .
## Race3Hispanic               4.84084    4.64456   1.042  0.29784
## Race3Mexican                4.39209    4.66721   0.941  0.34717
## Race3White                  2.25183    3.74627   0.601  0.54808
## Race3Other                 -3.16617    5.39096  -0.587  0.55728
## Education9 - 11th Grade    -1.52333    3.28156  -0.464  0.64272
## EducationHigh School       -0.28400    3.07475  -0.092  0.92645
## EducationSome College       0.93238    3.06363   0.304  0.76101
## EducationCollege Grad      -1.97907    3.36150  -0.589  0.55632
## MaritalStatusLivePartner   -1.42350    2.99910  -0.475  0.63527
## MaritalStatusMarried       -3.78705    2.26820  -1.670  0.09567 .
## MaritalStatusNeverMarried   3.10287    2.77621   1.118  0.26429
## MaritalStatusSeparated     -6.81544    5.04845  -1.350  0.17767
## MaritalStatusWidowed        1.12319    3.28249   0.342  0.73237
## HHIncome 5000-9999         -6.62464    6.28070  -1.055  0.29209
## HHIncome10000-14999        -3.94524    5.33768  -0.739  0.46020
## HHIncome15000-19999       -10.43443    5.36976  -1.943  0.05260 .
## HHIncome20000-24999        -6.02142    5.37616  -1.120  0.26329
## HHIncome25000-34999        -7.32342    5.28099  -1.387  0.16619
## HHIncome35000-44999        -8.41409    5.49167  -1.532  0.12617
## HHIncome45000-54999        -9.09756    5.81827  -1.564  0.11859
## HHIncome55000-64999        -7.49498    6.41055  -1.169  0.24294
## HHIncome65000-74999        -3.10158    6.35765  -0.488  0.62589
## HHIncome75000-99999         0.84807    6.32161   0.134  0.89334
## HHIncomemore 99999          0.13591    6.26647   0.022  0.98271
## Poverty                    -2.17460    0.92469  -2.352  0.01911 *
## Weight                      0.37465    0.33504   1.118  0.26406
## Height                     -0.42714    0.34297  -1.245  0.21362
## BMI                        -0.98037    0.96845  -1.012  0.31192
## DepressedSeveral           -0.42129    1.82545  -0.231  0.81758
## DepressedMost               2.15839    2.61450   0.826  0.40949
## SleepHrsNight               0.03708    0.52440   0.071  0.94367
## SleepTroubleYes            -2.67353    1.54819  -1.727  0.08486 .
## PhysActiveYes              -0.62633    1.55355  -0.403  0.68702
## SmokeNowYes                -0.67578    1.56942  -0.431  0.66696
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15 on 462 degrees of freedom
## Multiple R-squared:  0.3123, Adjusted R-squared:  0.2572
## F-statistic:  5.67 on 37 and 462 DF,  p-value: < 2.2e-16
```

```r
n <- nrow(train_minus_id)
sel.var.aic <- step(model.lm, trace = 0, k = 2, direction = "both")
sel.var.aic<-attr(terms(sel.var.aic), "term.labels")
sel.var.aic
```

```
## [1] "Gender"        "Age"           "Race3"         "MaritalStatus"
## [5] "HHIncome"      "Poverty"       "SleepTrouble"
```

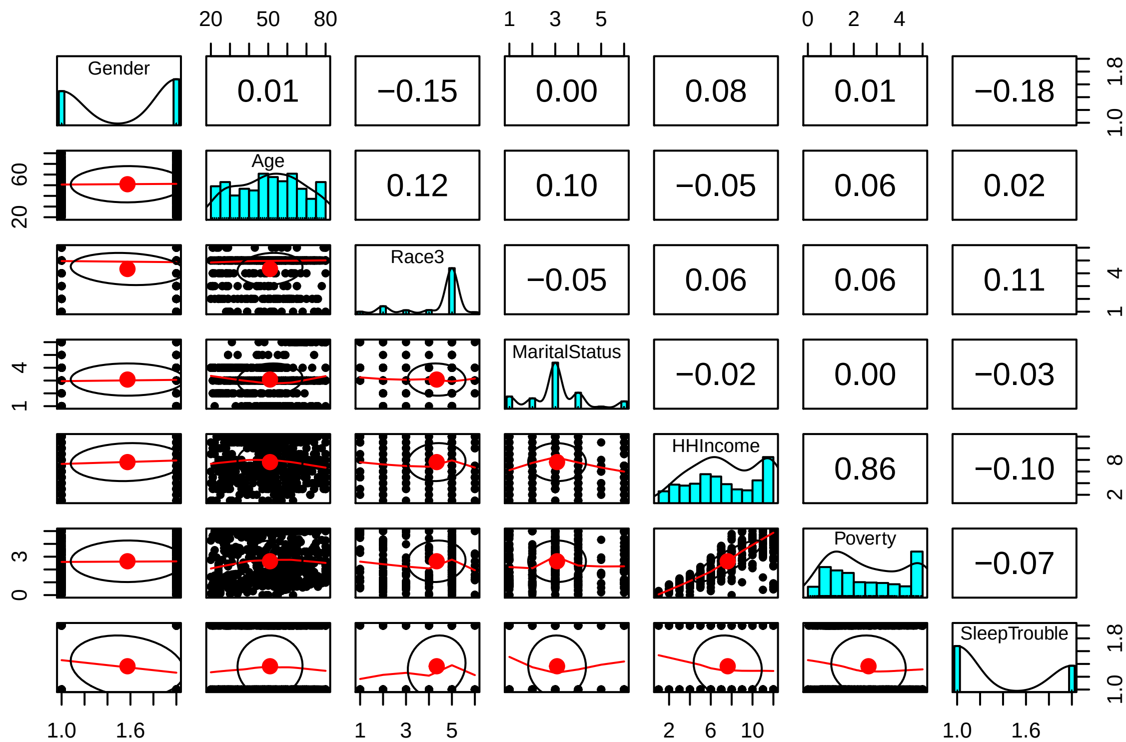
```r
model.aic <-  lm(BPSysAve ~., data = train_minus_id[,c(sel.var.aic, "BPSysAve")])
```

diagnostics

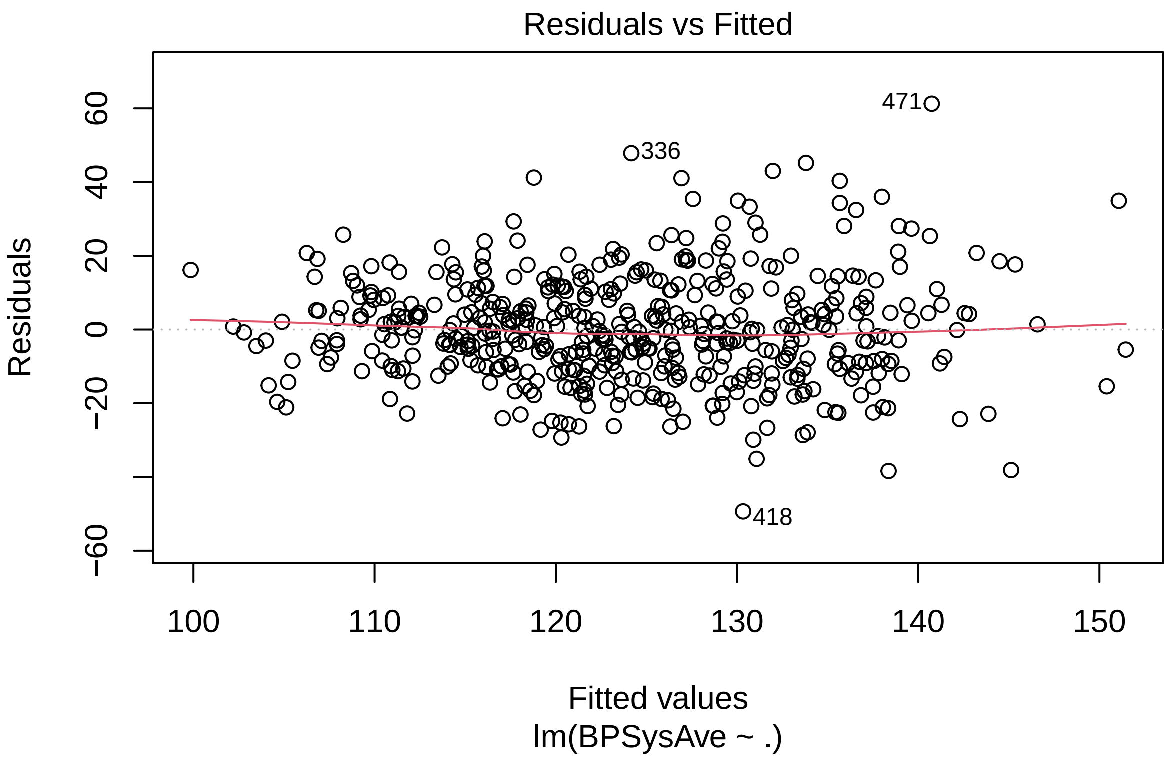
```r
r.aic <- f_multi_diagnostic(model.aic)
```

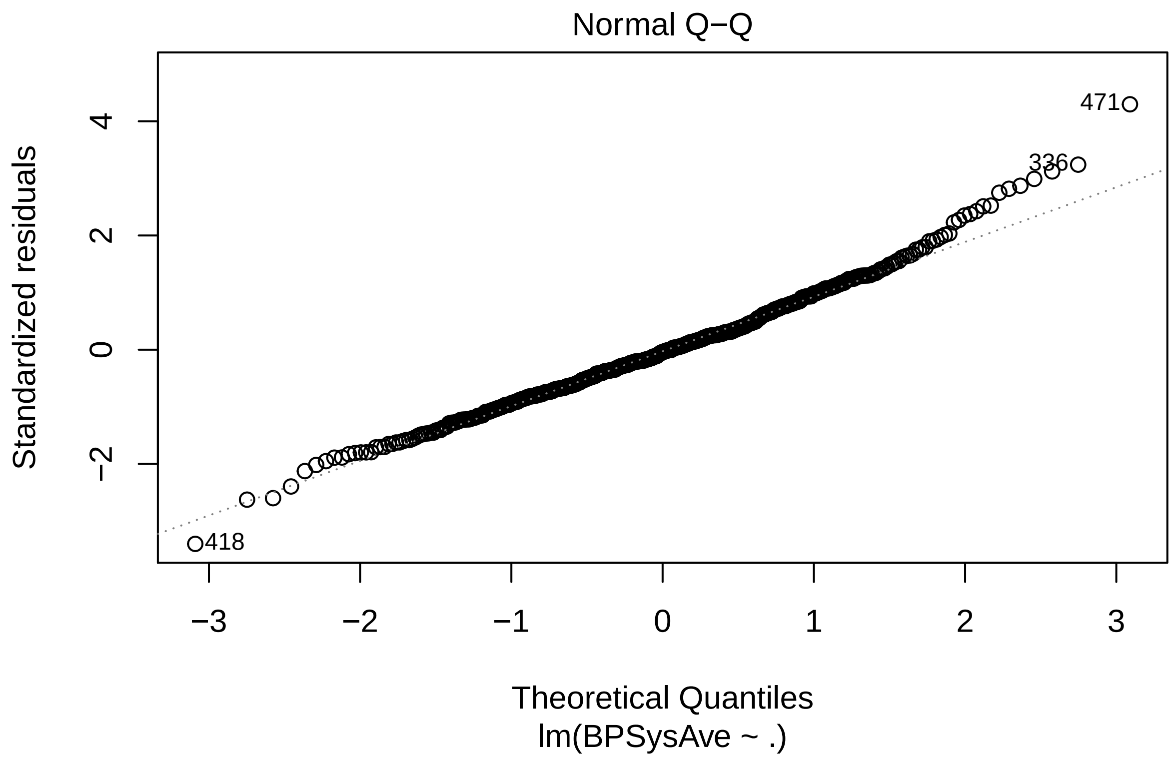
```
## [1] "leverage"
## 420 675 311 206 262 190 677 583  94 531 436 594 696 552 290 403 630 275 725 102
##   9  16  34  39  50  53  60  68  73 166 180 246 275 282 291 297 298 355 365 390
## 522 351 619  54 135 179 474
## 393 396 401 447 466 473 488
## [1] "cut_d"
## named integer(0)
## [1] "cut_fits"
## 310 420 196 631  86 107 311 319 206  90  20 362 142 348 260 343 677 443 231 583
##   2   9  11  12  18  23  34  38  39  41  42  44  48  49  51  56  60  66  67  68
## 246  94 167 704 424 488 428 545 126 665 617  36 209 487 225 476 279 564 516  95
##  72  73  77  78  80  85  89  95  96  97  98  99 101 102 106 109 117 120 122 125
## 604 106 104 171 641 674 177 111 720 131 245 646 363 526 413 599 436 471  47 154
## 129 131 138 140 143 152 154 155 156 157 164 167 169 175 177 178 180 184 186 187
## 625  23 383 695 731  15 426 283 276 423 687 418 108 315 457 594 557 606 338 512
## 192 193 194 201 204 205 208 210 211 212 217 233 235 236 240 246 250 257 260 265
## 251  43 552 403 630 632 324 723 492 530 300 671 125  72 214 541 373 610 239 243
## 267 281 282 297 298 315 328 333 337 347 349 356 357 364 372 375 376 384 386 387
## 522 619 571 511 647  79 314 129 432 226  78 513  76 444  54 603 303 281 664 135
## 393 401 408 409 410 415 422 433 436 439 441 442 443 444 447 449 452 457 462 466
## 336 274 375 650 694 474  37 626
## 476 477 482 483 486 488 491 500
## [1] "cut_beta"
## 691 420 675 348 262 167 476 268 171 177 471 283 338 696 632 492  72 243 522 314
##   3   9  16  49  50  77 109 110 140 154 184 210 260 275 315 337 364 387 393 422
## 513  54
## 442 447
## [1] "lev + cut_b"
## [1]   9  16  50 275 393 447
## [1] "lev + cut_fits"
##  [1]   9  34  39  60  68  73 180 246 282 297 298 393 401 447 466 488
## [1] " lev + cut_d"
## integer(0)
## [1] "b + fits"
##  [1]   9  49  77 109 140 154 184 210 260 315 337 364 387 393 422 442 447
## [1] "d + b"
```

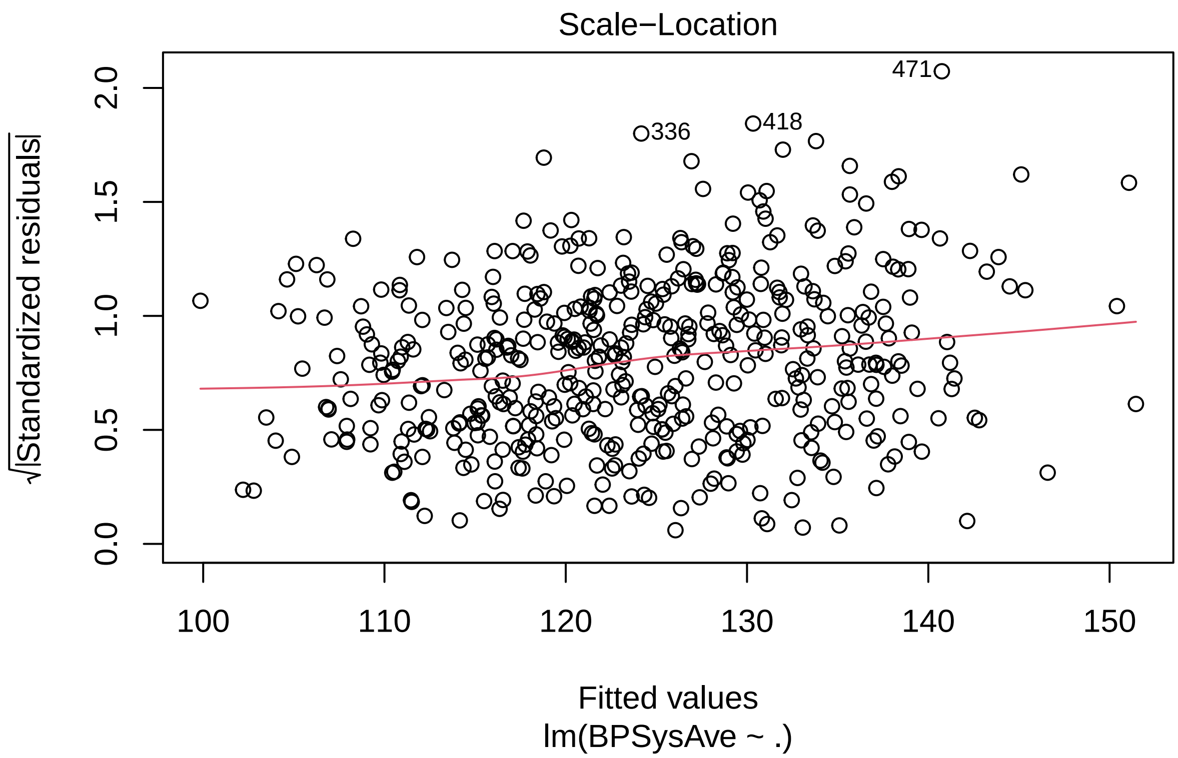
```
## integer(0)
## [1] "d + fits"
## integer(0)
## [1] " all outliers intersect"
## integer(0)
```

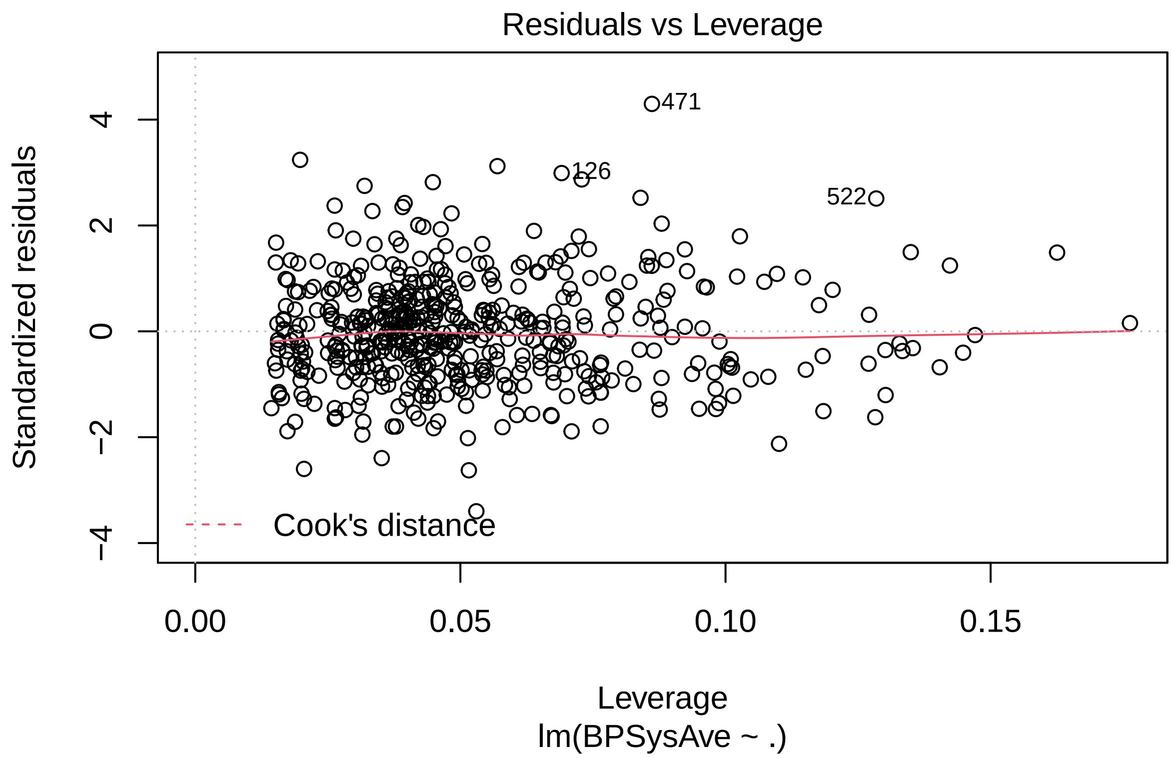



```
## NULL
```


Residuals vs Fitted
471
336
418
Residuals
60
40
20
0
−20
−60
100
110
120
130
140
150
Fitted values
lm(BPSysAve ~ .)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(BPSysAve ~ .)
471
336
418
4
2
0
−2
−3
−2
−1
0
1
2
3


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
100
110
120
130
140
150
471
336
418
Fitted values
lm(BPSysAve ~ .)




```
## NULL
## Analysis of Variance Table
##
## Response: BPSysAve
##                Df Sum Sq Mean Sq  F value    Pr(>F)
## Gender          1   1348  1347.9   6.0629 0.0141599 *
## Age             1  30323 30323.1 136.3984 < 2.2e-16 ***
## Race3           5   2125   425.0   1.9115 0.0909606 .
## MaritalStatus   5   4881   976.2   4.3913 0.0006428 ***
## HHIncome       11   4768   433.5   1.9497 0.0316128 *
## Poverty         1   1639  1638.8   7.3716 0.0068681 **
## SleepTrouble    1    632   631.9   2.8425 0.0924571 .
## Residuals     474 105376   222.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## [1] "VIF"
##                   GVIF Df GVIF^(1/(2*Df))
## Gender        1.160952  1        1.077475
## Age           1.527480  1        1.235913
## Race3         1.336965  5        1.029466
## MaritalStatus 2.007030  5        1.072150
## HHIncome      6.383621 11        1.087912
## Poverty       4.458362  1        2.111483
## SleepTrouble  1.096534  1        1.047155
```

```
model.aic.vif.outliers.df <- model.aic$model[-union(union(r.aic[[1]], r.aic[[2]]), r.aic[[4]]), -which(n
model.aic.vif.outliers <- lm(BPSysAve ~., data = model.aic.vif.outliers.df)
mult <- lm(cbind(BPSysAve, Age, Poverty)~1, data = model.aic.vif.outliers.df
           %>% filter(Poverty > 0)) # this allows us to ensure that we can do the boxcox tranformation f
summary(powerTransform(mult))
```

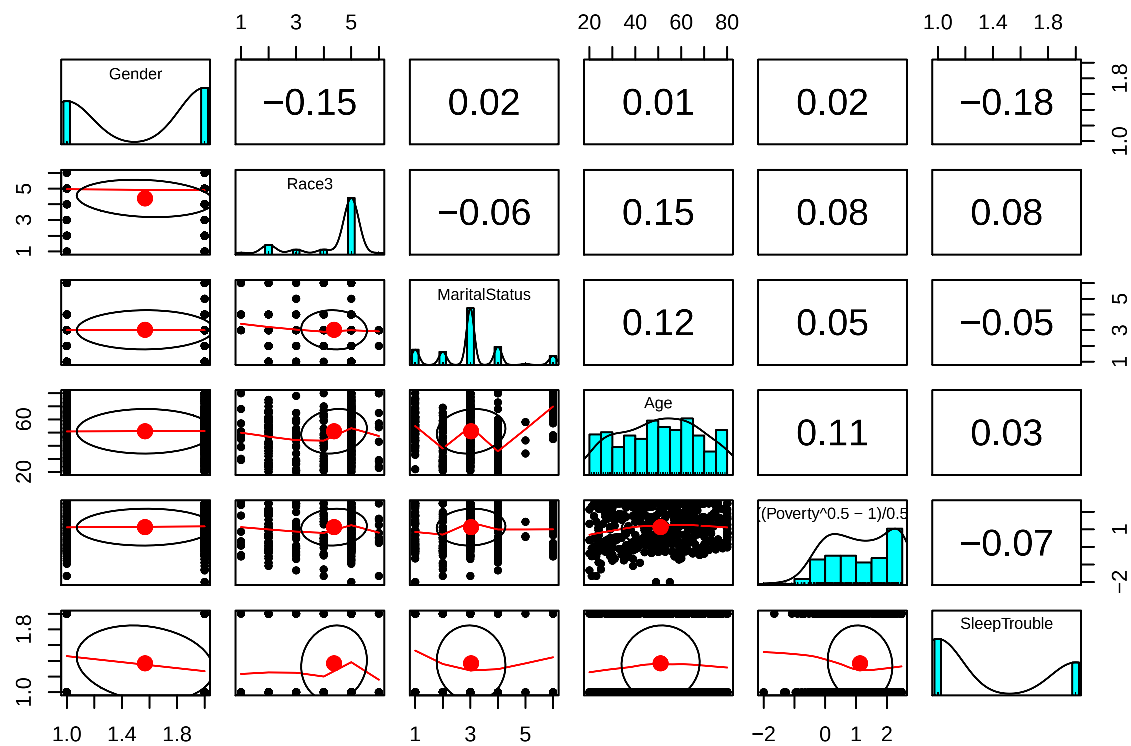
```
## bcPower Transformations to Multinormality
##          Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## BPSysAve   -0.1837         0.0      -0.7101       0.3426
## Age         0.8826         1.0       0.6287       1.1366
## Poverty     0.5171         0.5       0.3920       0.6422
##
## Likelihood ratio test that transformation parameters are equal to 0
##  (all log transformations)
##                                LRT df       pval
## LR test, lambda = (0 0 0) 140.2223  3 < 2.22e-16
##
## Likelihood ratio test that no transformations are needed
##                                LRT df       pval
## LR test, lambda = (1 1 1) 66.84866  3 2.0206e-14
```

boxcox

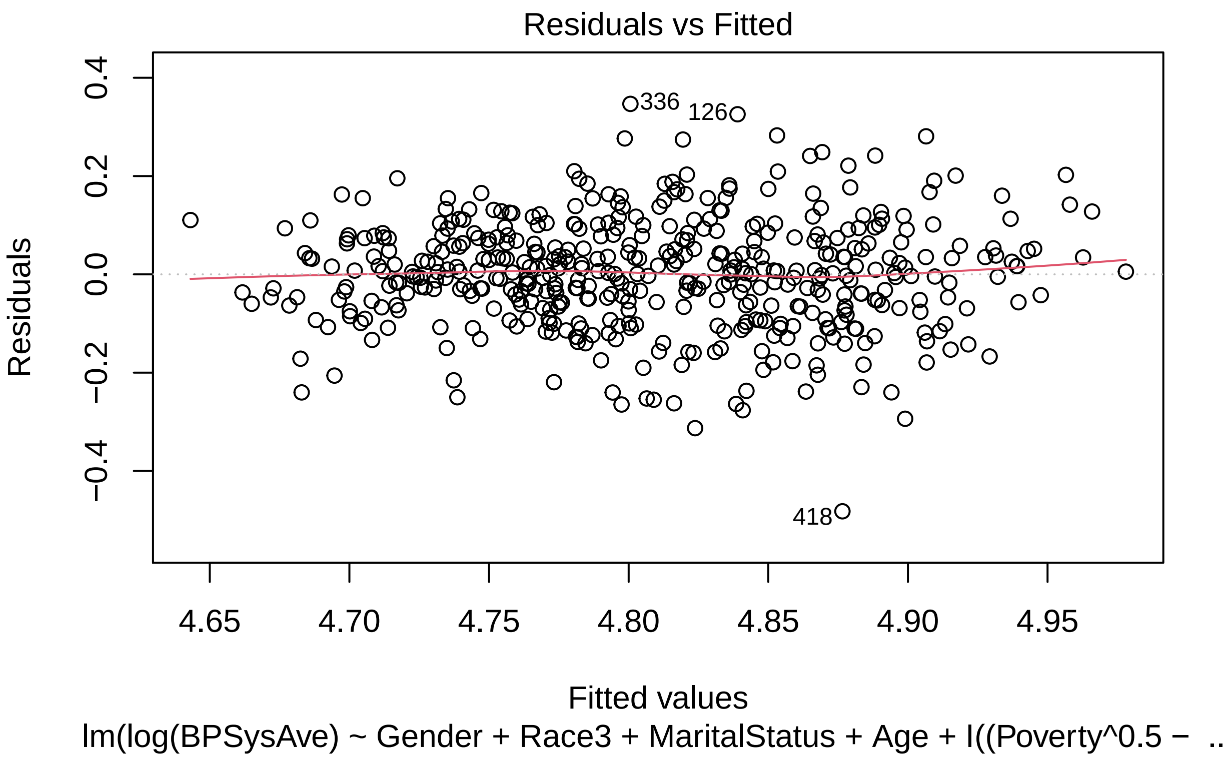
```
model.aic.vif.outliers.boxcox <- lm(log(BPSysAve) ~
                                      Gender +
                                      Race3 +
                                      MaritalStatus +
                                      Age +
                                      I((Poverty^.5 - 1)/.5) +
                                      SleepTrouble
                                    ,data = model.aic.vif.outliers.df)

#model.aic.vif.outliers.boxcox$model$`I(geometric.mean(Poverty)^(1 - 0.5) * (Poverty^0.5 - 1)/0.5)`
f_multi_minus_vif(model.aic.vif.outliers.boxcox)
```

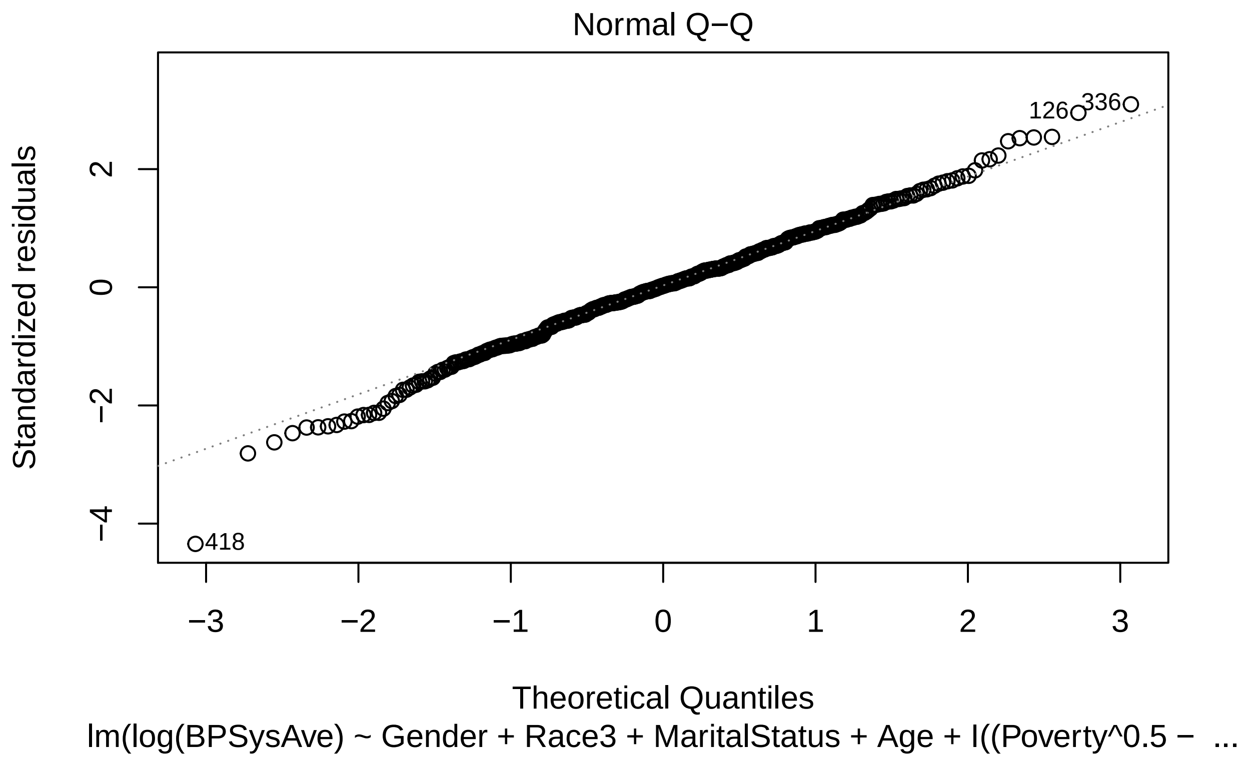
```
## [1] "leverage"
## 310 691 190 268  30 641 531  66  84  82 457 702 557 398 290 501 275 194 502 102
##   2   3  47  99 117 131 153 192 206 208 224 228 233 237 271 330 331 345 360 364
## 409 692 414 511 444 664 179  37 302 298
## 367 378 379 381 414 431 441 458 461 466
## [1] "cut_d"
## named integer(0)
## [1] "cut_fits"
## 196 631 107 319  20 362 260 343 443 231 424 488 545 126 665 617 209 225 268 279
##  10  11  21  35  38  40  45  50  59  60  70  75  85  86  87  88  91  96  99 106
## 564 104 641 674 111 245 531 646 413 599  47 154 625 695  15 687  84 418 108 315
## 109 127 131 140 142 151 153 154 164 165 171 172 177 186 190 201 206 217 219 220
## 457 557 606 523 251 199  43 723 300 501 275 671 125 610 239 409 414 571 511  79
## 224 233 240 244 249 252 262 310 325 330 331 332 333 359 361 367 379 380 381 387
## 444 303 281 664 336 375 694  37
## 414 421 426 431 444 450 454 458
```

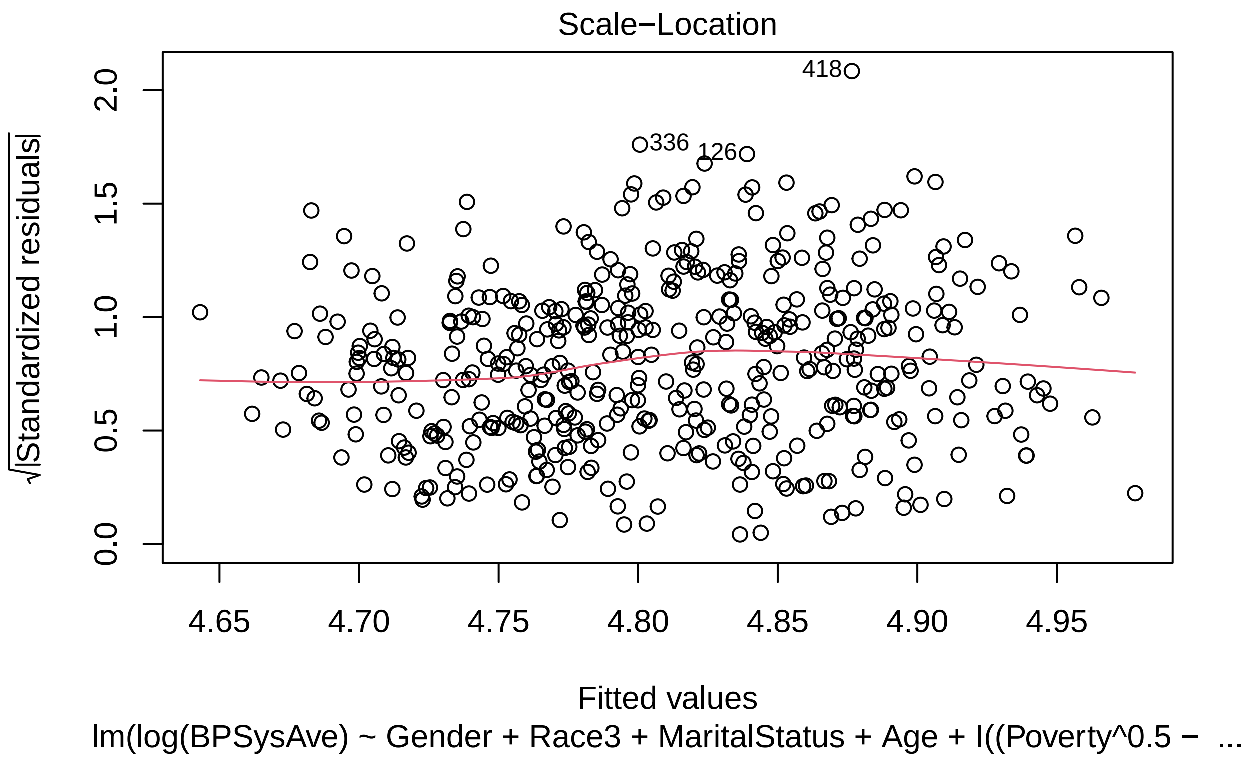
```
## [1] "cut_beta"
## 310 704 268 564  84 418 290 692 414 571 444 302
##   2  68  99 109 206 217 271 378 379 380 414 461
## [1] "lev + cut_b"
## [1]   2  99 206 271 378 379 414 461
## [1] "lev + cut_fits"
##  [1]  99 131 153 206 224 233 330 331 367 379 381 414 431 458
## [1] " lev + cut_d"
## integer(0)
## [1] "b + fits"
## [1]  99 109 206 217 379 380 414
## [1] "d + b"
## integer(0)
## [1] "d + fits"
## integer(0)
## [1] " all outliers intersect"
## integer(0)
```

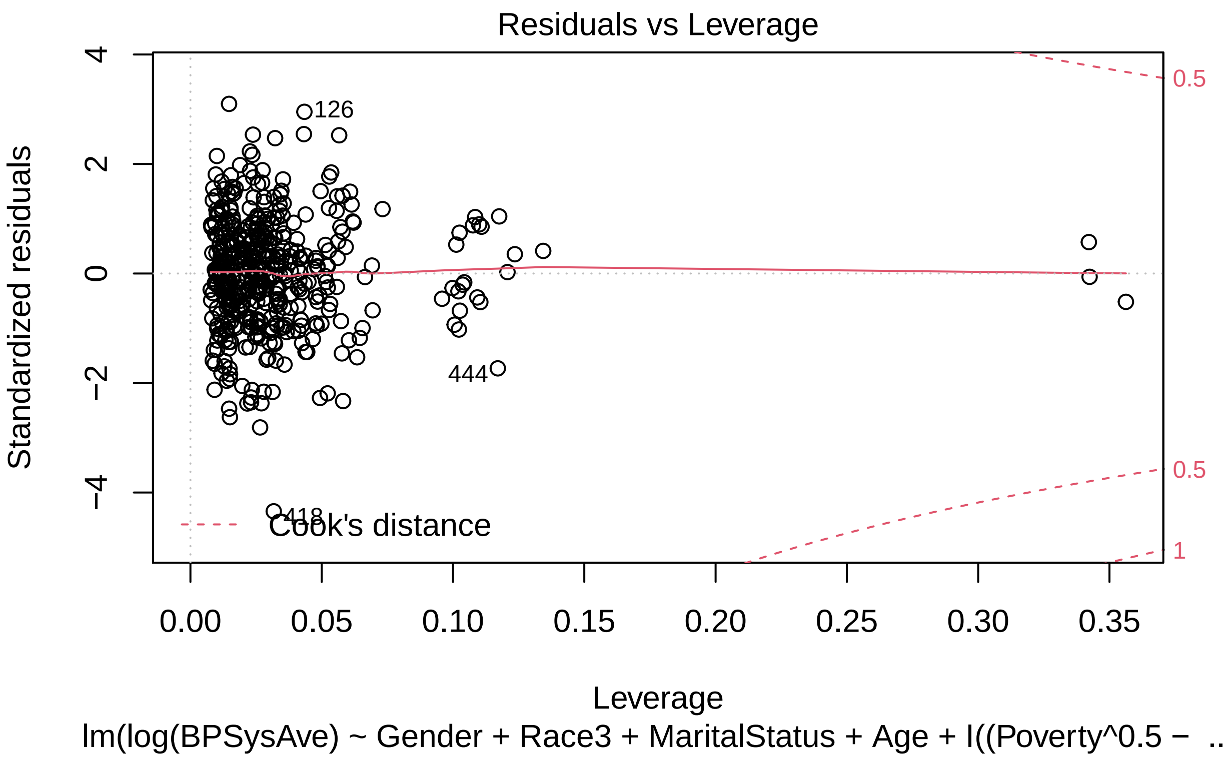



```
## NULL
```


Residuals vs Fitted
Residuals
0.4
0.2
0.0
−0.2
−0.4
336
126
418
4.65
4.70
4.75
4.80
4.85
4.90
4.95
Fitted values
lm(log(BPSysAve) ~ Gender + Race3 + MaritalStatus + Age + I((Poverty^0.5 − ...


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(log(BPSysAve) ~ Gender + Race3 + MaritalStatus + Age + I((Poverty^0.5 − ...
126
336
418
2
0
−2
−4
−3
−2
−1
0
1
2
3


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
418
336
126
4.65
4.70
4.75
4.80
4.85
4.90
4.95
Fitted values
lm(log(BPSysAve) ~ Gender + Race3 + MaritalStatus + Age + I((Poverty^0.5 − ...




```
## NULL
## Analysis of Variance Table
##
## Response: log(BPSysAve)
##                            Df Sum Sq Mean Sq  F value    Pr(>F)
## Gender                      1 0.0679 0.06791   5.3362   0.02134 *
## Race3                       5 0.0962 0.01924   1.5122   0.18455
## MaritalStatus               5 0.5338 0.10676   8.3887 1.335e-07 ***
## Age                         1 1.3769 1.37686 108.1882 < 2.2e-16 ***
## I((Poverty^0.5 - 1)/0.5)    1 0.0396 0.03956   3.1086   0.07855 .
## SleepTrouble                1 0.0168 0.01685   1.3239   0.25051
## Residuals                 452 5.7524 0.01273
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## [[1]]
## [1]   2  99 206 271 378 379 414 461
##
## [[2]]
##  [1]  99 131 153 206 224 233 330 331 367 379 381 414 431 458
##
## [[3]]
## integer(0)
##
## [[4]]
```

```
## [1]  99 109 206 217 379 380 414
##
## [[5]]
## integer(0)
##
## [[6]]
## integer(0)
##
## [[7]]
## integer(0)
```

```
model.aic.vif.outliers
```

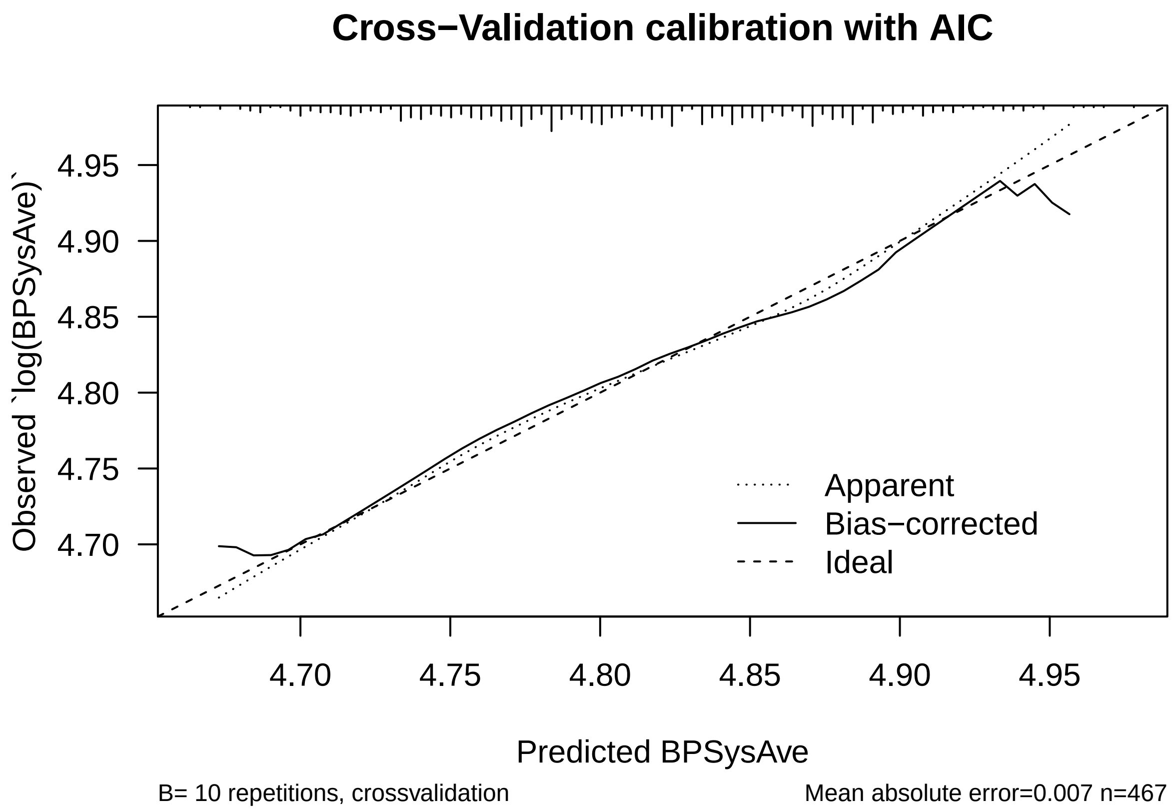
```
##
## Call:
## lm(formula = BPSysAve ~ ., data = model.aic.vif.outliers.df)
##
## Coefficients:
##              (Intercept)                Gendermale
##                  93.2780                    1.8123
##                      Age                Race3Black
##                   0.4957                   11.4762
##            Race3Hispanic               Race3Mexican
##                   9.6355                    8.0931
##               Race3White                Race3Other
##                   5.9953                    3.6812
##  MaritalStatusLivePartner      MaritalStatusMarried
##                   0.3960                   -2.3712
## MaritalStatusNeverMarried    MaritalStatusSeparated
##                   4.7795                  -10.5305
##      MaritalStatusWidowed                   Poverty
##                   3.2403                   -0.7665
##           SleepTroubleYes
##                  -1.5409
```

```
error.aic <- mean((model.aic.vif.outliers.boxcox$fitted.values - model.aic.vif.outliers.boxcox$model$`lo
error.aic
```

```
## [1] 0.01231778
```

cross validation and testing

```
ols.aic <- ols(`log(BPSysAve)` ~ ., data = model.aic.vif.outliers.boxcox$model,
               x=T, y=T, model = T)

## 10 fold cross validation ##
aic.cross <- calibrate(ols.aic, method = "crossvalidation", B = 10)
## Calibration plot ##
plot(aic.cross, las = 1, xlab = "Predicted BPSysAve", main = "Cross-Validation calibration with AIC")
```




```
##
## n=467   Mean absolute error=0.007   Mean squared error=7e-05
## 0.9 Quantile of absolute error=0.013
```

```
g <- test_minus_id$BPSysAve
gsub(",", "", g)
```

```
##   [1] "103" "110" "122" "121" "123" "125" "130" "141" "217" "128" "108" "107"
##  [13] "131" "143" "86"  "126" "124" "132" "133" "115" "102" "138" "137" "166"
##  [25] "119" "119" "122" "157" "124" "116" "105" "132" "122" "165" "154" "128"
##  [37] "121" "115" "99"  "121" "126" "114" "117" "117" "104" "101" "119" "97"
##  [49] "170" "115" "120" "122" "140" "136" "116" "121" "110" "108" "118" "143"
##  [61] "130" "107" "128" "127" "156" "135" "102" "128" "130" "133" "124" "109"
##  [73] "98"  "109" "115" "125" "125" "136" "137" "100" "140" "132" "137" "114"
##  [85] "138" "122" "117" "105" "118" "113" "119" "128" "141" "130" "123" "116"
##  [97] "152" "110" "124" "116" "191" "114" "114" "161" "118" "124" "153" "129"
## [109] "116" "128" "145" "130" "98"  "113" "112" "106" "123" "117" "188" "142"
## [121] "116" "116" "124" "95"  "123" "134" "129" "112" "160" "136" "122" "124"
## [133] "126" "133" "114" "152" "105" "116" "169" "102" "98"  "135" "140" "116"
## [145] "104" "119" "122" "179" "122" "110" "131" "149" "110" "133" "112" "112"
## [157] "127" "123" "123" "117" "124" "121" "118" "141" "126" "103" "128" "114"
## [169] "123" "123" "104" "112" "150" "141" "132" "120" "131" "119" "100" "153"
## [181] "145" "111" "138" "107" "118" "114" "112" "105" "133" "112" "114" "111"
## [193] "121" "118" "128" "132" "118" "113" "129" "114" "113" "119" "130" "133"
## [205] "143" "126" "107" "108" "114" "129" "112" "114" "123" "161" "105" "103"
```

```
## [217] "117" "110" "173" "112" "99"  "105" "133" "121" "127" "179" "121" "147"
## [229] "109" "103" "147" "165" "131" "113" "159" "147" "136" "118" "114" "107"
## [241] "142" "137" "112"
```

```
g2 <-gsub(",", "", g)


test_minus_id.aic.transformation <- test_minus_id[-union(union(r.aic[[1]], r.aic[[2]]), r.aic[[4]]),]
g <- test_minus_id.aic.transformation$BPSysAve
gsub(",", "", g)
```

```
##   [1] "103" "110" "122" "121" "123" "125" "130" "141" "128" "108" "107" "131"
##  [13] "143" "86"  "124" "132" "133" "115" "102" "138" "137" "166" "119" "119"
##  [25] "122" "157" "124" "116" "105" "132" "122" "154" "128" "121" "115" "121"
##  [37] "126" "114" "117" "117" "104" "101" "119" "97"  "120" "122" "140" "136"
##  [49] "116" "121" "110" "108" "118" "130" "107" "128" "127" "156" "135" "102"
##  [61] "130" "133" "124" "109" "109" "115" "125" "136" "137" "100" "140" "132"
##  [73] "137" "114" "138" "122" "117" "105" "118" "113" "119" "128" "141" "130"
##  [85] "123" "116" "152" "110" "124" "116" "191" "114" "114" "161" "118" "124"
##  [97] "153" "129" "128" "145" "130" "98"  "113" "112" "106" "123" "117" "188"
## [109] "142" "116" "116" "124" "95"  "123" "134" "129" "112" "160" "136" "122"
## [121] "124" "126" "133" "114" "152" "105" "116" "169" "98"  "135" "140" "116"
## [133] "104" "119" "122" "179" "122" "110" "131" "149" "110" "112" "112" "127"
## [145] "123" "123" "117" "124" "121" "118" "141" "126" "103" "128" "114" "123"
## [157] "123" "104" "112" "150" "141" "132" "120" "131" "119" "100" "145" "111"
## [169] "138" "118" "114" "112" "105" "133" "112" "114" "111" "121" "118" "128"
## [181] "132" "118" "113" "129" "114" "113" "119" "130" "133" "143" "126" "107"
## [193] "108" "114" "112" "114" "123" "161" "105" "103" "117" "110" "173" "112"
## [205] "99"  "105" "133" "121" "127" "179" "121" "147" "109" "103" "147" "165"
## [217] "131" "113" "159" "147" "136" "118" "114" "107" "142" "137" "112"
```

```
g2 <-gsub(",", "", g)
test_minus_id.aic.transformation$`log(BPSysAve)` <- log(as.numeric(g2))

test_minus_id.aic.transformation$`I((Poverty^0.5 - 1)/0.5)` <- I((test_minus_id.aic.transformation$Pover
## Test Error ##
pred.aic <- predict(ols.aic, newdata = test_minus_id.aic.transformation[,c( "Gender",        "Age"   ,
"I((Poverty^0.5 - 1)/0.5)", "SleepTrouble", "log(BPSysAve)")])
## Prediction error ##
pred.error.AIC <- mean((test_minus_id.aic.transformation$`log(BPSysAve)` - pred.aic)^2)
pred.error.AIC
```

```
## [1] 0.01540345
```

BIC

```
n <- nrow(train)
sel.var.bic <- step(model.lm, trace = 0, k = log(n), direction = "both")
sel.var.bic<-attr(terms(sel.var.bic), "term.labels")
sel.var.bic
```

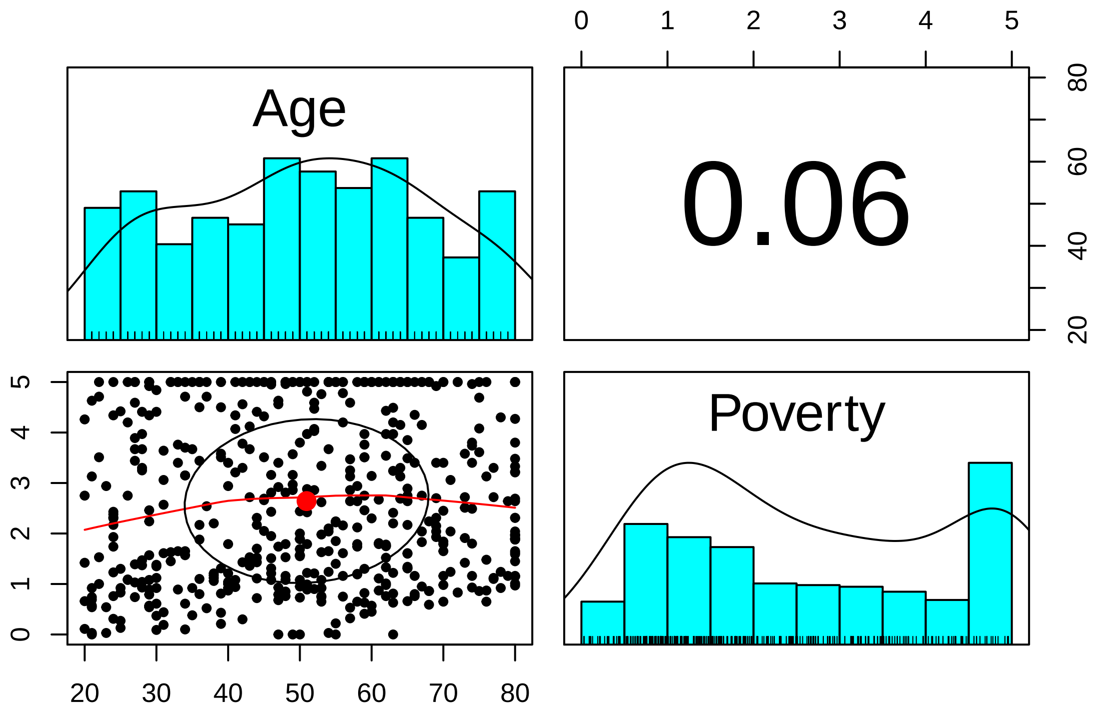
```
## [1] "Age"     "Poverty"
```

```
model.bic<-  lm(BPSysAve ~., data = train_minus_id[,c(sel.var.bic, "BPSysAve")])
```

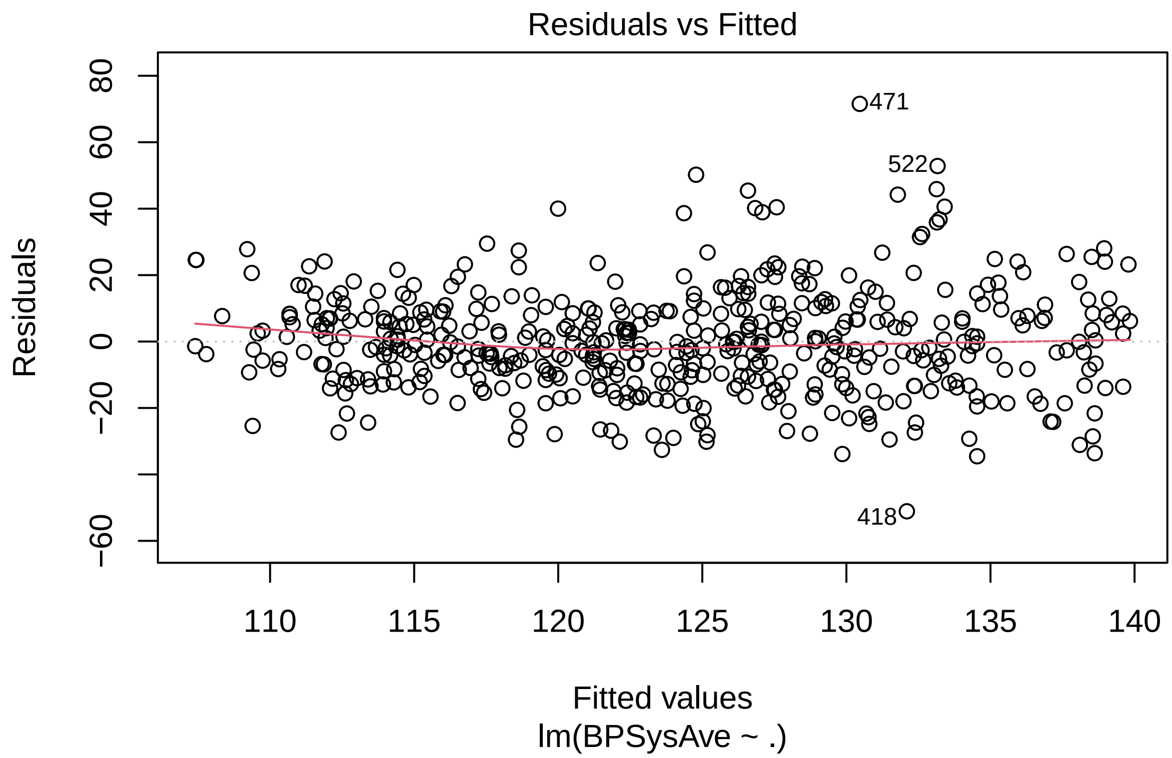
###diagnostics

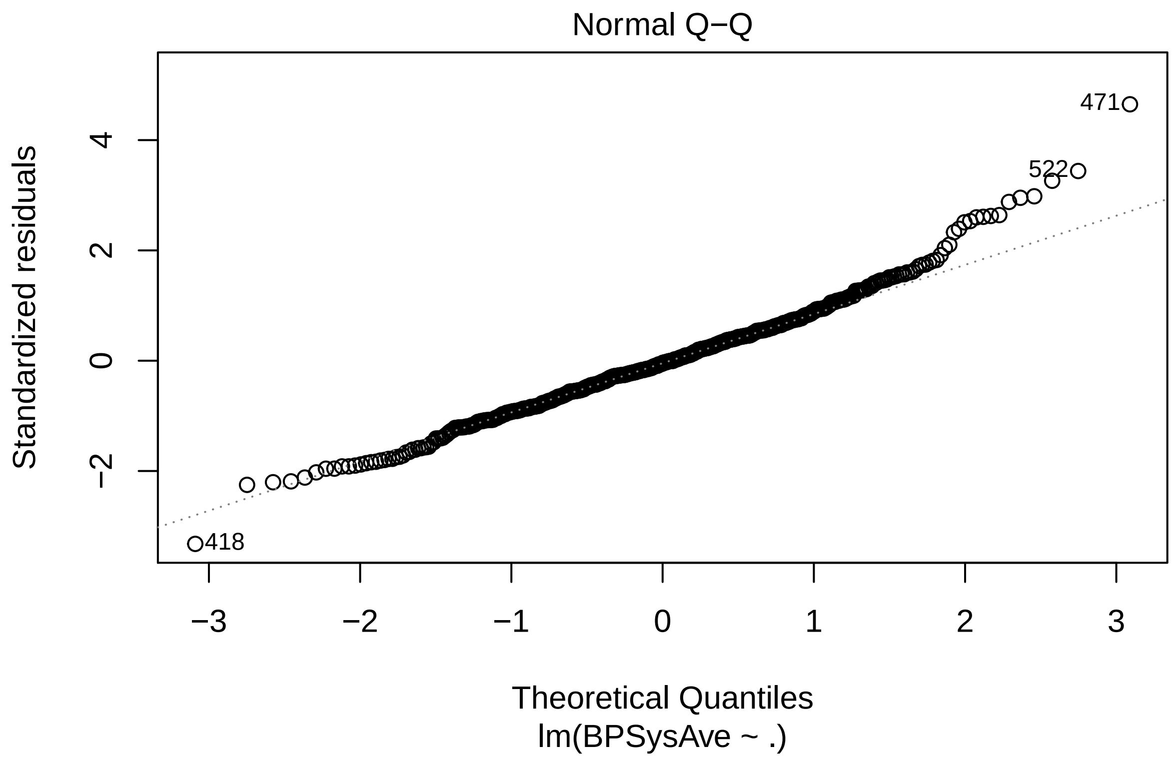
```
r.bic <- f_multi_diagnostic(model.bic)
```

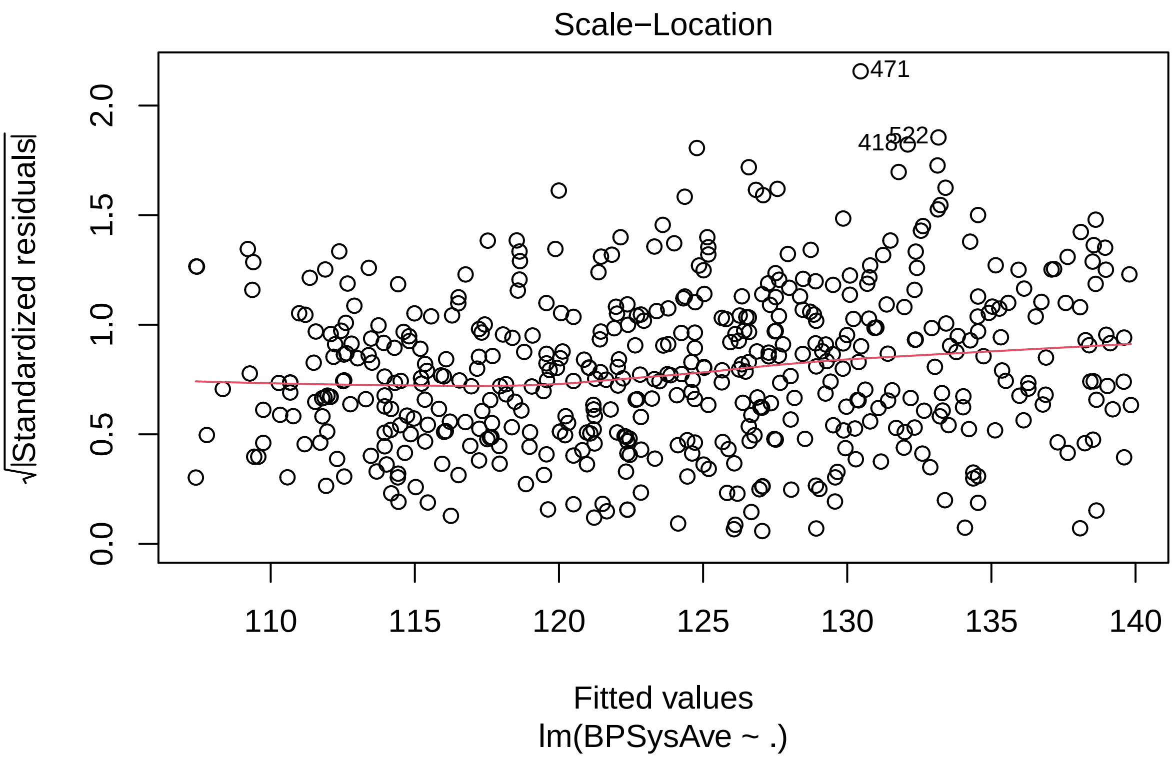
```
## [1] "leverage"
## 206 357 102 351
##  39 316 390 396
## [1] "cut_d"
## named integer(0)
## [1] "cut_fits"
## 196 348 231 167 573 126 209 564  95 471 193 687 418 108 315 251 723 300 610 522
##  11  49  67  77  92  96 101 120 125 184 185 217 233 235 236 267 333 349 384 393
## 571 513 444 664 336
## 408 442 444 462 476
## [1] "cut_beta"
## 196 631 682 260 343 231 167 573 126 104 413 193 154 687 315 594 606 523 241 579
##  11  12  47  51  56  67  77  92  96 138 177 185 187 217 236 246 257 262 283 325
## 723 610 448 351 571 432 226 513 135 336
## 333 384 389 396 408 436 439 442 466 476
## [1] "lev + cut_b"
## [1] 396
## [1] "lev + cut_fits"
## integer(0)
## [1] " lev + cut_d"
## integer(0)
## [1] "b + fits"
##  [1]  11  67  77  92  96 185 217 236 333 384 408 442 476
## [1] "d + b"
## integer(0)
## [1] "d + fits"
## integer(0)
## [1] " all outliers intersect"
## integer(0)
```

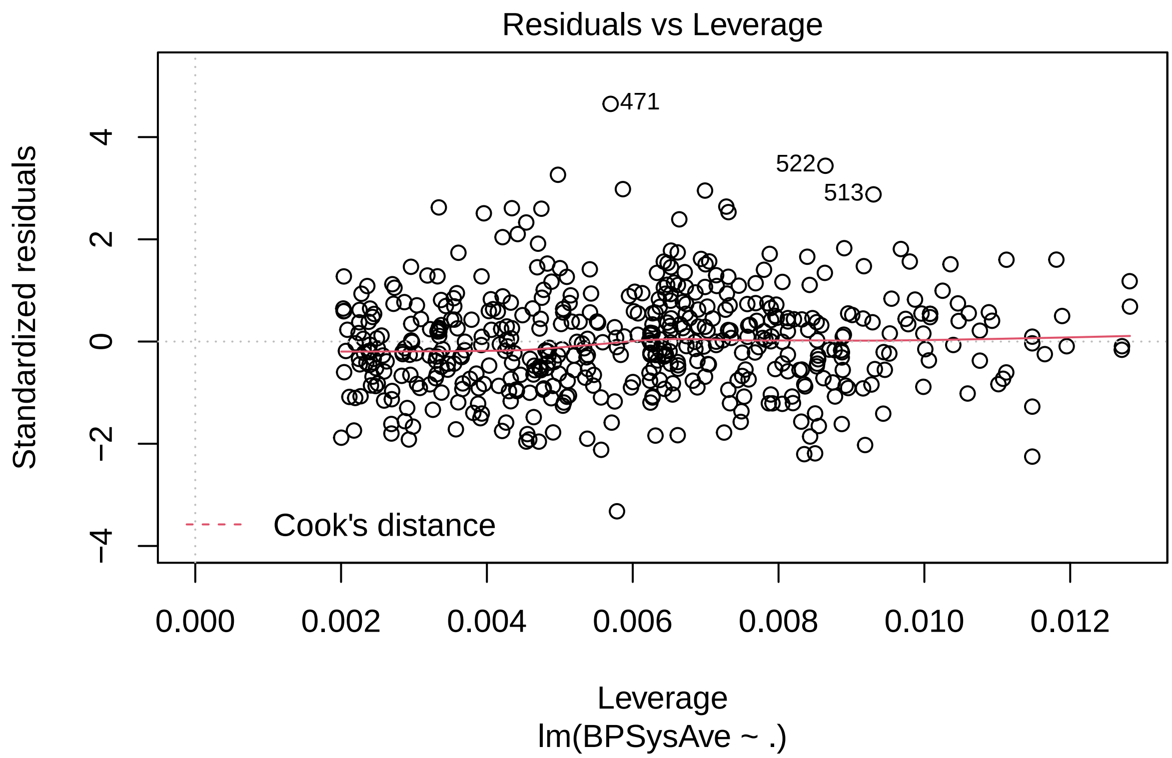

Age
0.06
Poverty
0
1
2
3
4
5
20
40
60
80
20
30
40
50
60
70
80
0
1
2
3
4
5
20
40
60
80

```
## NULL
```


Residuals vs Fitted
Residuals
80
60
40
20
0
−20
−60
110
115
120
125
130
135
140
471
522
418
Fitted values
lm(BPSysAve ~ .)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(BPSysAve ~ .)
471
522
418
4
2
0
−2
−3
−2
−1
0
1
2
3


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
110
115
120
125
130
135
140
Fitted values
lm(BPSysAve ~ .)
471
522
418




```
## NULL
## Analysis of Variance Table
##
## Response: BPSysAve
##            Df Sum Sq Mean Sq  F value    Pr(>F)
## Age         1  30507 30507.0 128.1465 < 2.2e-16 ***
## Poverty     1   2268  2267.7   9.5255  0.002139 **
## Residuals 497 118317   238.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## [1] "VIF"
##      Age  Poverty
## 1.004177 1.004177
```

outliers

```
model.bic.vif.outliers.df <- model.bic$model[-union(r.bic[[1]], r.bic[[4]]),]
model.bic.vif.outliers <- lm(BPSysAve ~., data = model.bic.vif.outliers.df)
```

#boxcox

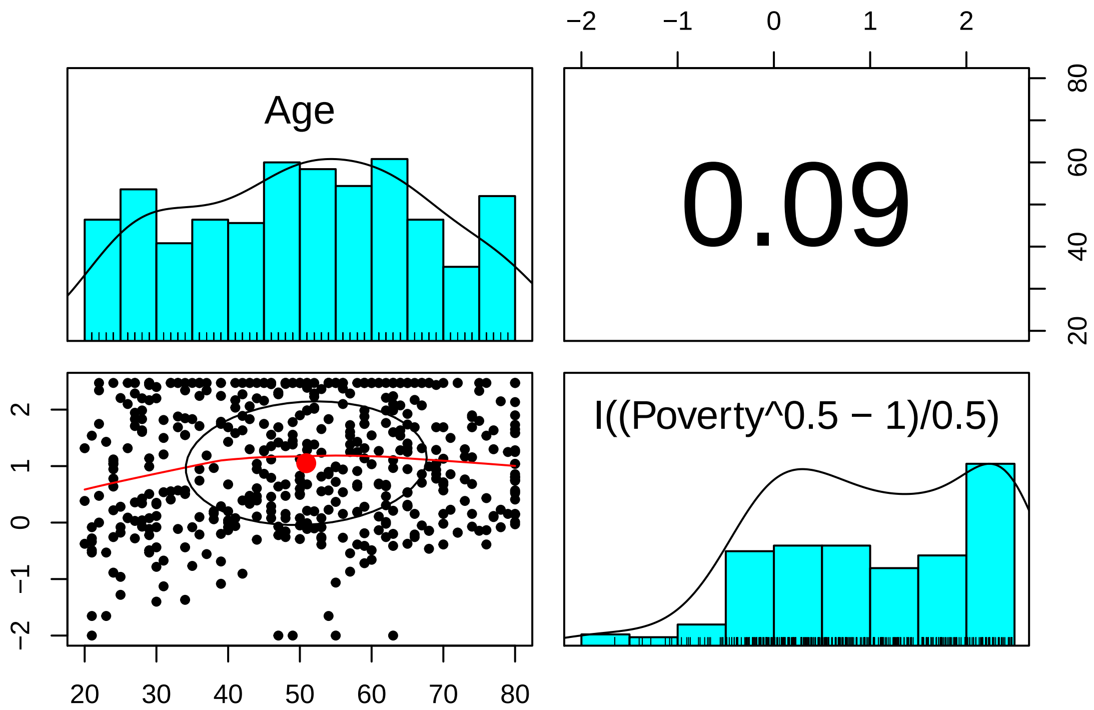
```
mult <- lm(cbind(BPSysAve, Age, Poverty)~1, data = model.bic.vif.outliers.df %>% filter(Poverty > 0))
summary(powerTransform(mult))
```

```
## bcPower Transformations to Multinormality
##          Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## BPSysAve   -0.1050         0.0      -0.6004       0.3904
## Age         0.8617         1.0       0.6093       1.1142
## Poverty     0.5140         0.5       0.4013       0.6266
##
## Likelihood ratio test that transformation parameters are equal to 0
##  (all log transformations)
##                               LRT df       pval
## LR test, lambda = (0 0 0) 160.6255  3 < 2.22e-16
##
## Likelihood ratio test that no transformations are needed
##                               LRT df       pval
## LR test, lambda = (1 1 1) 77.36574  3 < 2.22e-16
```

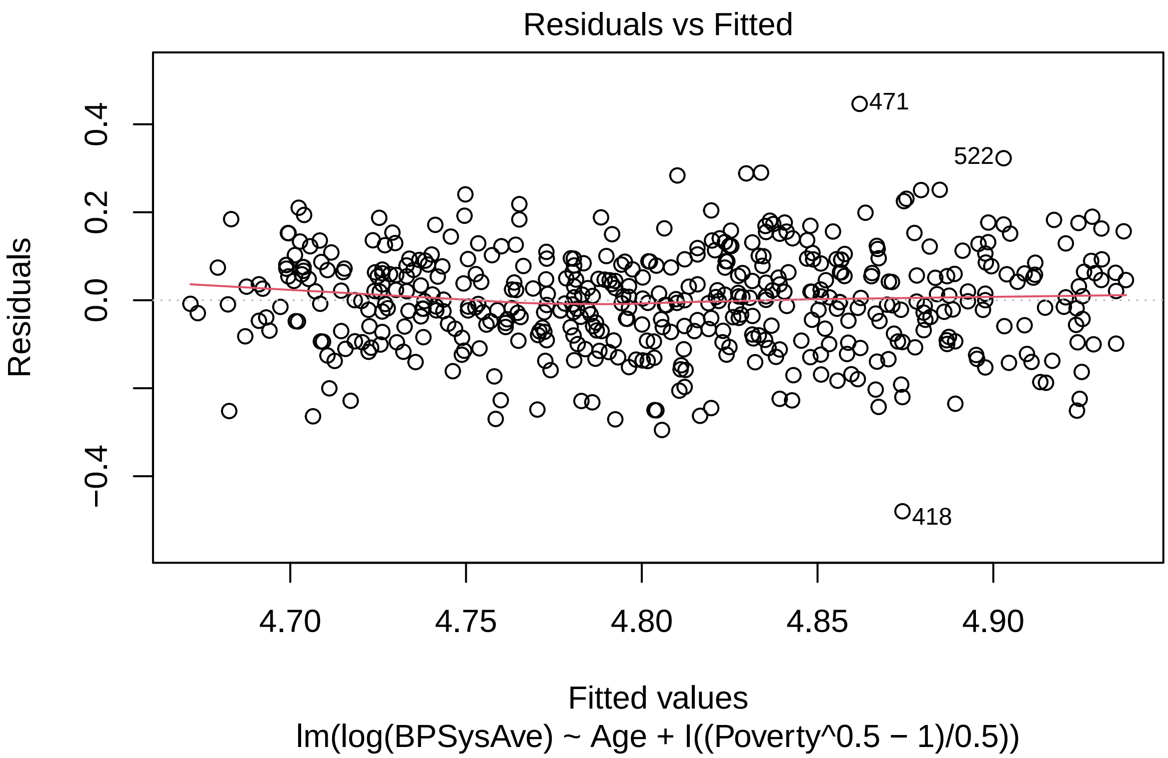
```
model.bic.vif.outliers.boxcox <- lm(log(BPSysAve) ~
                                    #Gender +
                                    #Race3 +
                                    #MaritalStatus +
                                    Age +
                                    I((Poverty^.5 - 1)/.5)
                                    #SleepTrouble
                                  ,data = model.bic.vif.outliers.df)

f_multi_diagnostic(model.bic.vif.outliers.boxcox)
```

```
## [1] "leverage"
## 420 675 206 262 435 232 177 667 696 357 510 725 102 522 179
##   9  15  38  49  88 116 149 248 267 308 321 356 380 383 460
## [1] "cut_d"
## named integer(0)
## [1] "cut_fits"
## 107 319 348 361 209 564  95 104 171 413 471 154 418 594 338 251 300 448 522 592
##  22  37  48  83  96 115 120 133 135 172 179 181 226 238 252 259 340 379 383 385
## 226 444
## 427 431
## [1] "cut_beta"
## 631  86  20 682 260 343 218 488 361 353 104 413 154 594 606 338 523 241 121 579
##  11  17  41  46  50  55  64  82  83 113 133 172 181 238 249 252 254 275 297 317
## 448 432 226 444 135
## 379 424 427 431 453
## [1] "lev + cut_b"
## integer(0)
## [1] "lev + cut_fits"
## [1] 383
## [1] " lev + cut_d"
## integer(0)
## [1] "b + fits"
## [1]  83 133 172 181 238 252 379 427 431
## [1] "d + b"
## integer(0)
## [1] "d + fits"
## integer(0)
## [1] " all outliers intersect"
```

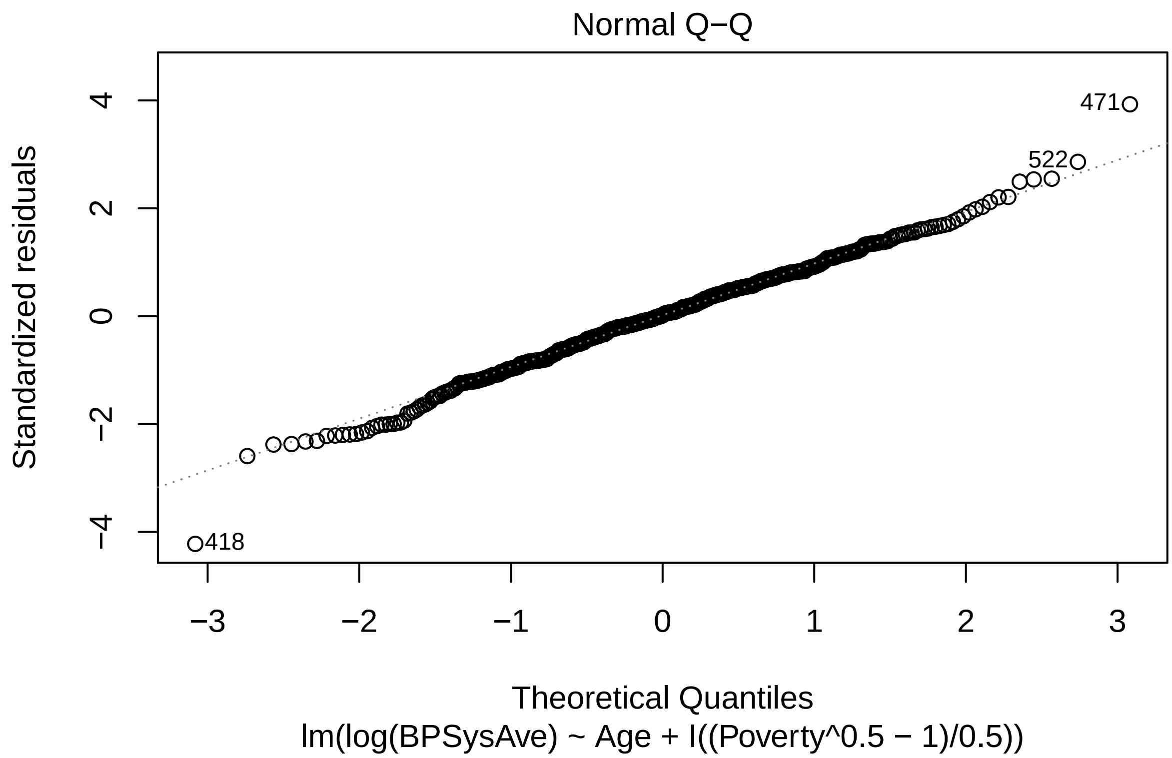
```
## integer(0)
```

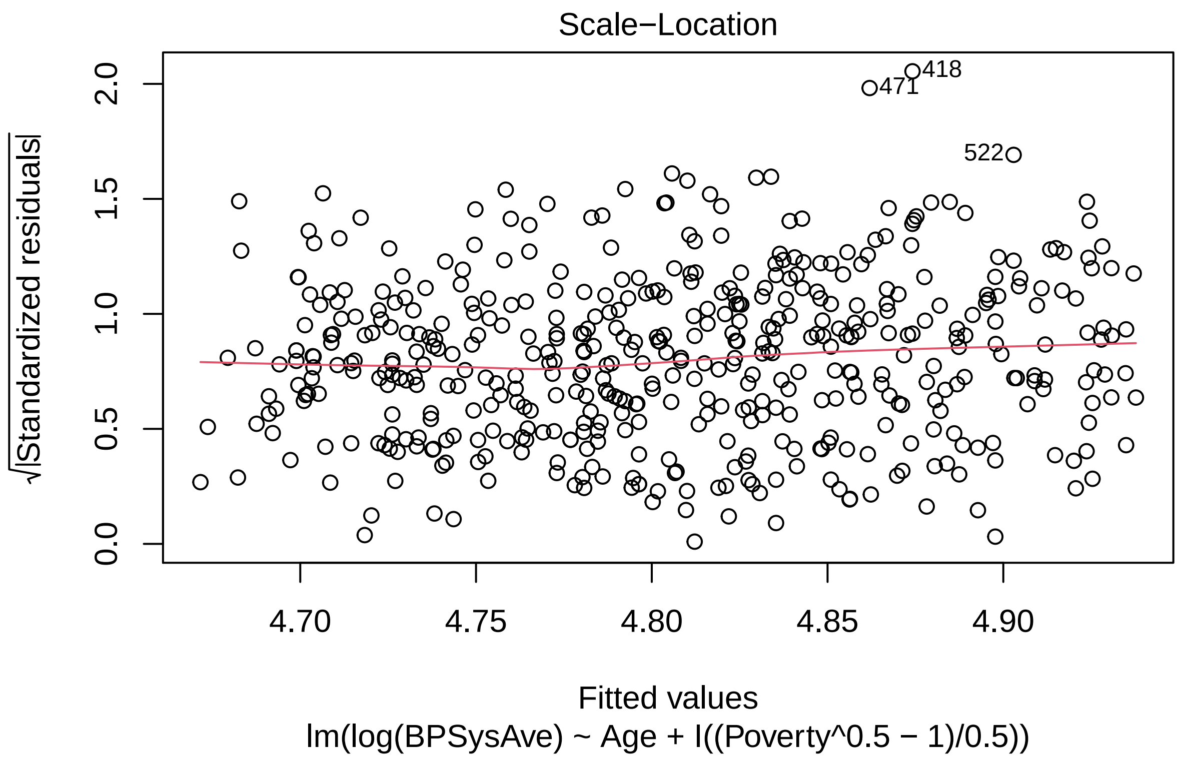



```
## NULL
```

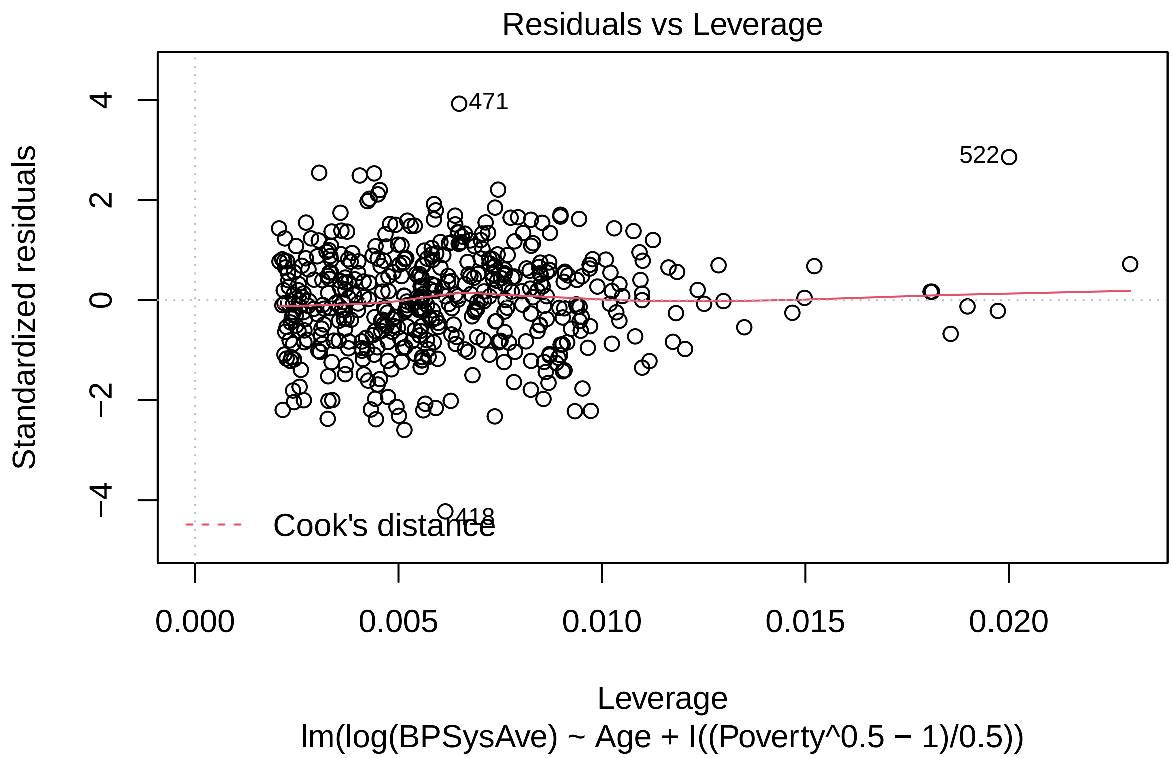


Residuals vs Fitted
Residuals
0.4
0.2
0.0
−0.4
471
522
418
4.70
4.75
4.80
4.85
4.90
Fitted values
lm(log(BPSysAve) ~ Age + I((Poverty^0.5 − 1)/0.5))


Normal Q−Q
471
522
418
Standardized residuals
4
2
0
−2
−4
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(log(BPSysAve) ~ Age + I((Poverty^0.5 − 1)/0.5))



Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
4.70
4.75
4.80
4.85
4.90
Fitted values
lm(log(BPSysAve) ~ Age + I((Poverty^0.5 − 1)/0.5))
418
471
522




```
## NULL
## Analysis of Variance Table
##
## Response: log(BPSysAve)
##                           Df Sum Sq Mean Sq F value    Pr(>F)
## Age                        1 1.9841 1.98407 152.719 < 2.2e-16 ***
## I((Poverty^0.5 - 1)/0.5)   1 0.1474 0.14737  11.344 0.0008175 ***
## Residuals                483 6.2750 0.01299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## [1] "VIF"
##                      Age I((Poverty^0.5 - 1)/0.5)
##                 1.007638                 1.007638
```

```
## [[1]]
## integer(0)
##
## [[2]]
## [1] 383
##
## [[3]]
## integer(0)
##
## [[4]]
## [1]  83 133 172 181 238 252 379 427 431
```

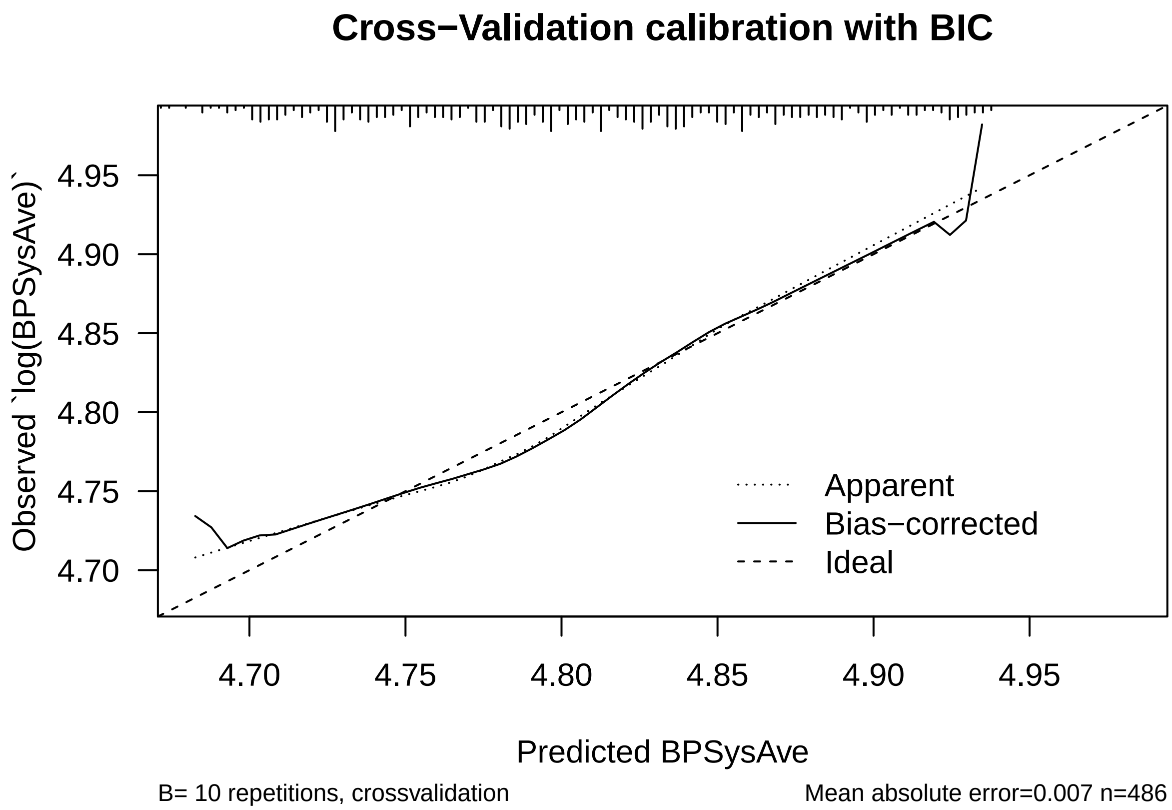
```
##
## [[5]]
## integer(0)
##
## [[6]]
## integer(0)
##
## [[7]]
## integer(0)
```

```
error.bic <- mean((model.bic.vif.outliers.boxcox$fitted.values - model.bic.vif.outliers.boxcox$model$`lo
error.bic
```

```
## [1] 0.01291148
```

testing and cross validation

```
ols.bic <- ols(`log(BPSysAve)`  ~ ., data = model.bic.vif.outliers.boxcox$model,
               x=T, y=T, model = T)

## 10 fold cross validation ##
bic.cross <- calibrate(ols.bic, method = "crossvalidation", B = 10)
#bic.boot <- calibrate(ols.bic, method = "boot", B = 10)
## Calibration plot ##
#pdf("bic_cross.pdf", height = 8, width = 16)
plot(bic.cross, las = 1, xlab = "Predicted BPSysAve", main = "Cross-Validation calibration with BIC")
```




```
##
## n=486   Mean absolute error=0.007   Mean squared error=1e-04
## 0.9 Quantile of absolute error=0.014
```

```
#plot(bic.boot,las = 1, xlab = "Predicted LPSA", main = "Bootstrapping calibration with BIC")
#dev.off()
test_minus_id.bic.transformation <-  test_minus_id[-union(r.bic[[1]],r.bic[[4]]),]

g <- test_minus_id.bic.transformation$BPSysAve
#gsub(",", "", g)
g2 <-gsub(",", "", g)
test_minus_id.bic.transformation$`log(BPSysAve)` <- log(as.numeric(g2))
test_minus_id.bic.transformation$`I((Poverty^0.5 - 1)/0.5)` <- I((test_minus_id.bic.transformation$Pove
## Test Error ##
pred.bic <- predict(ols.bic, newdata = test_minus_id.bic.transformation[,c("Age","I((Poverty^0.5 - 1)/0
## Prediction error ##
pred.error.BIC <- mean((test_minus_id.bic.transformation$`log(BPSysAve)` - pred.bic)^2)
pred.error.BIC
```

```
## [1] 0.01713172
```

model choice

```r
print(c(pred.error.AIC, pred.error.BIC, pred.error.lasso, min(c(pred.error.AIC, pred.error.BIC, pred.er
```

```
## [1]   0.01540345   0.01713172 344.23522691   0.01540345
```

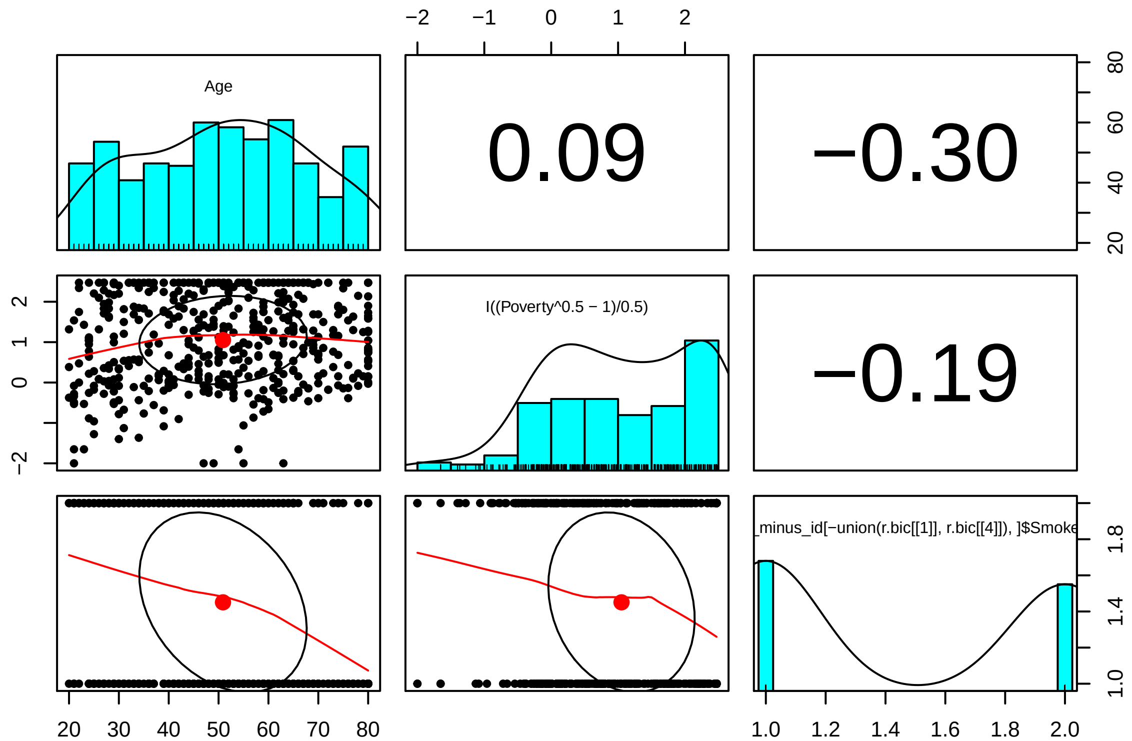
adding in smokenow to BIc mdel

```r
model.bic.vif.outliers.boxcox.smokeNow <- lm(`log(BPSysAve)` ~., data = cbind(model.bic.vif.outliers.bo
```

diagnostic of BIC with smoek now

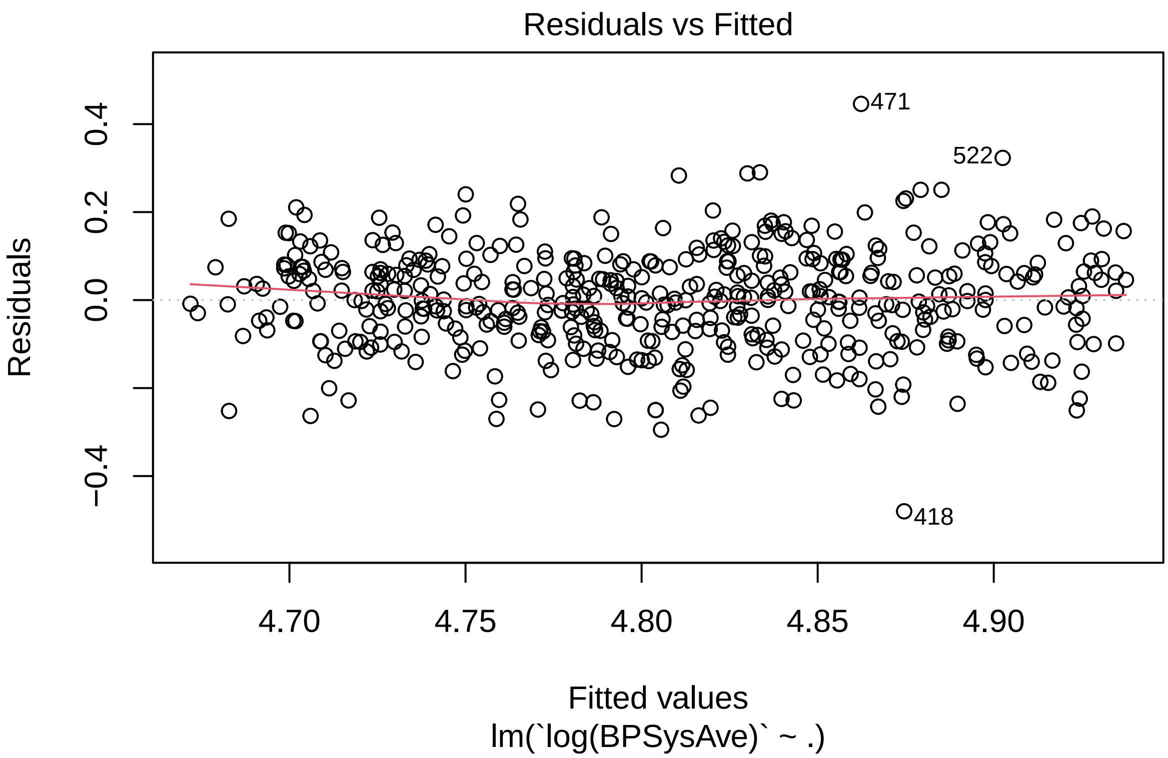
```r
model.bic.vif.outliers.boxcox.smokeNow %>% f_multi_diagnostic()
```

```
## [1] "leverage"
## 675 206 262 435 232 607 667 696 102 522 179
##  15  38  49  88 116 196 248 267 380 383 460
## [1] "cut_d"
## named integer(0)
## [1] "cut_fits"
## 107 348 361 209 564 104 471 154  15 426 418 594 251 300 239 243 522 226 444 213
##  22  48  83  96 115 133 179 181 199 202 226 238 259 340 376 377 383 427 431 448
## 474
## 474
## [1] "cut_beta"
## 107 260 443 361 104 646 413 154 607  15 426 418 594 743 523 251  59 630 239 522
##  22  50  65  83 133 162 172 181 196 199 202 226 238 251 254 259 271 290 376 383
## 297 226 303 213 664 474
## 425 427 439 448 449 474
## [1] "lev + cut_b"
## [1] 196 383
## [1] "lev + cut_fits"
## [1] 383
## [1] " lev + cut_d"
## integer(0)
## [1] "b + fits"
##  [1]  22  83 133 181 199 202 226 238 259 376 383 427 448 474
## [1] "d + b"
## integer(0)
## [1] "d + fits"
## integer(0)
## [1] " all outliers intersect"
## integer(0)
```

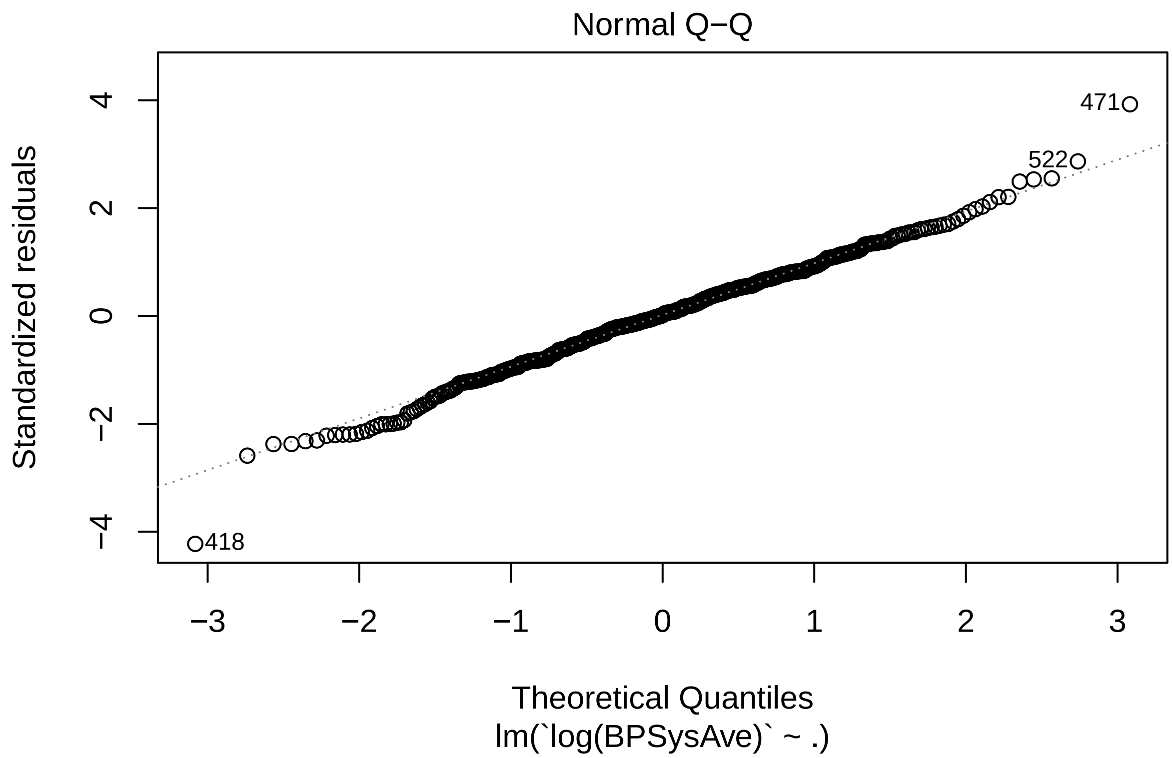


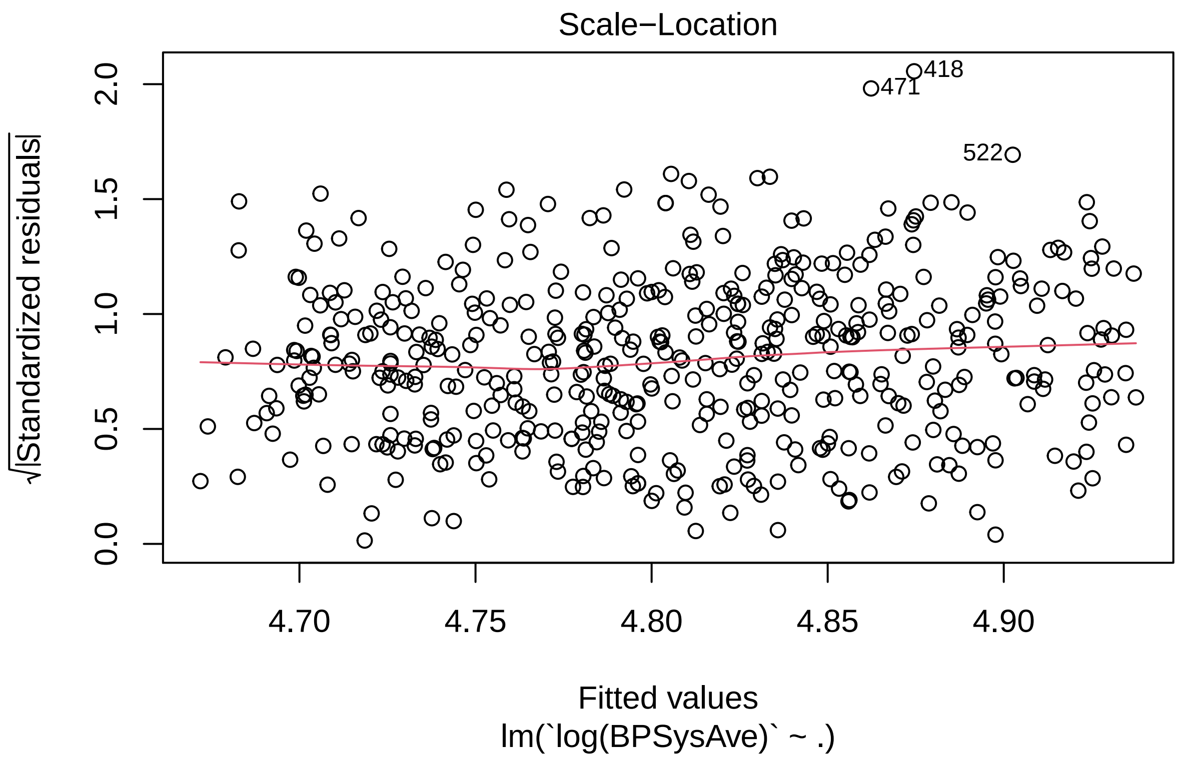
Age
0.09
−0.30
I((Poverty^0.5 − 1)/0.5)
−0.19
_minus_id[−union(r.bic[[1]], r.bic[[4]]),]$Smoke

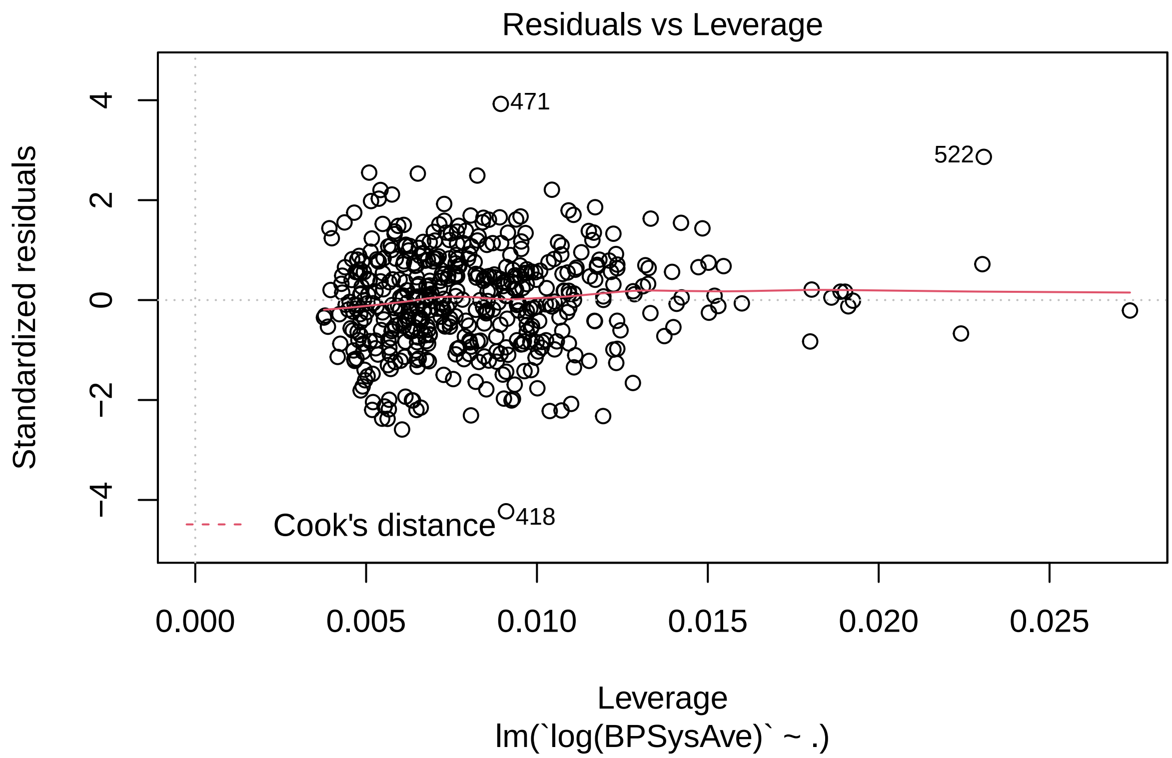

```
## NULL
```



Residuals vs Fitted
Residuals
0.4
0.2
0.0
−0.4
471
522
418
4.70
4.75
4.80
4.85
4.90
Fitted values
lm(`log(BPSysAve)` ~ .)


Normal Q−Q
Standardized residuals
4
2
0
−2
−4
471
522
418
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(`log(BPSysAve)` ~ .)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
4.70
4.75
4.80
4.85
4.90
Fitted values
lm(`log(BPSysAve)` ~ .)
418
471
522




```
## NULL
## Analysis of Variance Table
##
## Response: log(BPSysAve)
##                                                               Df Sum Sq Mean Sq
## Age                                                            1 1.9841 1.98407
## `I((Poverty^0.5 - 1)/0.5)`                                     1 0.1474 0.14737
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`    1 0.0001 0.00005
## Residuals                                                    482 6.2749 0.01302
##                                                               F value    Pr(>F)
## Age                                                          152.4037 < 2.2e-16
## `I((Poverty^0.5 - 1)/0.5)`                                    11.3203 0.0008278
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`    0.0039 0.9501031
## Residuals
##
## Age                                                          ***
## `I((Poverty^0.5 - 1)/0.5)`                                   ***
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## [1] "VIF"
##                                                          Age
##                                                     1.099263
##                                   `I((Poverty^0.5 - 1)/0.5)`
##                                                     1.036887
```

```
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`
##                                                     1.129889
```

```
## [[1]]
## [1] 196 383
##
## [[2]]
## [1] 383
##
## [[3]]
## integer(0)
##
## [[4]]
##  [1]  22  83 133 181 199 202 226 238 259 376 383 427 448 474
##
## [[5]]
## integer(0)
##
## [[6]]
## integer(0)
##
## [[7]]
## integer(0)
```

```
summary(model.bic.vif.outliers.boxcox.smokeNow)
```

```
##
## Call:
## lm(formula = `log(BPSysAve)` ~ ., data = cbind(model.bic.vif.outliers.boxcox$model,
##     train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.48010 -0.07132  0.00198  0.07570  0.44595
##
## Coefficients:
##                                                             Estimate
## (Intercept)                                                4.6246833
## Age                                                        0.0039046
## `I((Poverty^0.5 - 1)/0.5)`                                -0.0159351
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`Yes  0.0006923
##                                                           Std. Error
## (Intercept)                                                0.0199582
## Age                                                        0.0003234
## `I((Poverty^0.5 - 1)/0.5)`                                 0.0048197
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`Yes  0.0110570
##                                                           t value Pr(>|t|)
## (Intercept)                                               231.718  < 2e-16
## Age                                                        12.073  < 2e-16
## `I((Poverty^0.5 - 1)/0.5)`                                 -3.306  0.00102
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`Yes   0.063  0.95010
##
## (Intercept)                                               ***
```

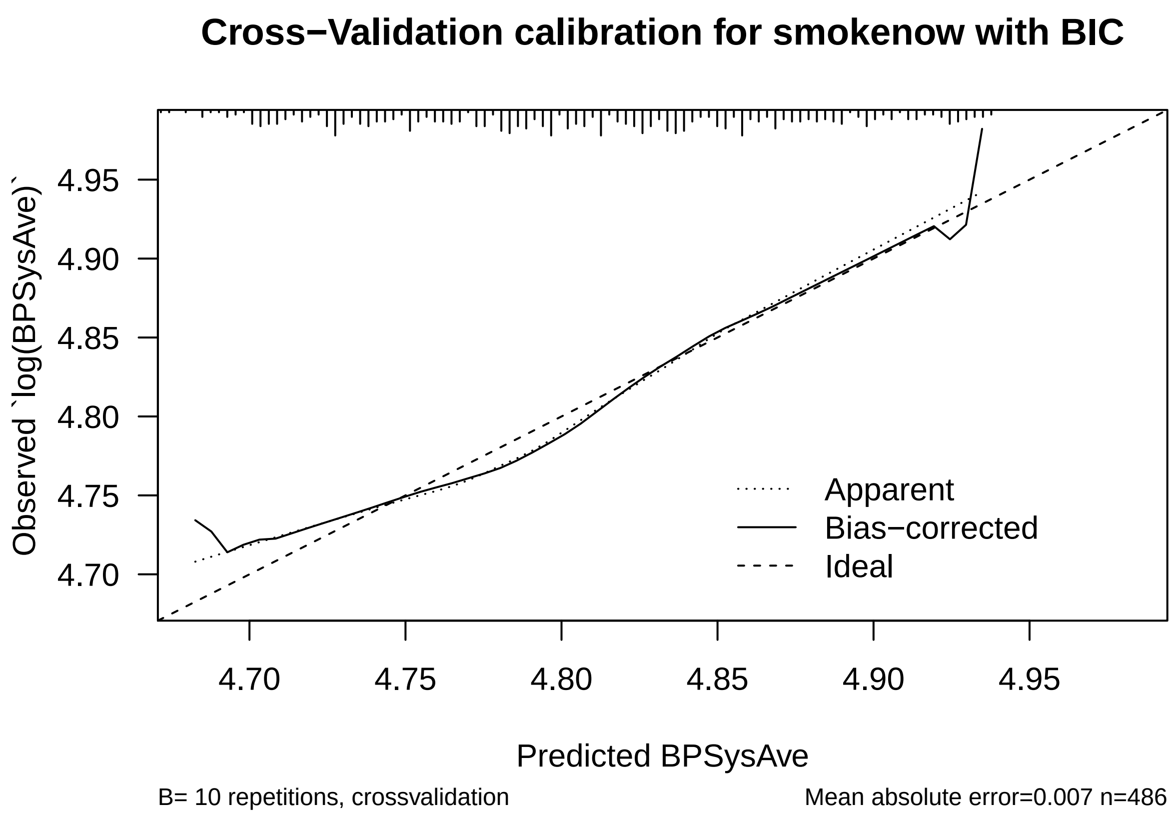
```
## Age                                                        ***
## `I((Poverty^0.5 - 1)/0.5)`                                 **
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`Yes
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1141 on 482 degrees of freedom
## Multiple R-squared:  0.2536, Adjusted R-squared:  0.2489
## F-statistic: 54.58 on 3 and 482 DF,  p-value: < 2.2e-16
```

```
anova(model.bic.vif.outliers.boxcox.smokeNow)
```

```
## Analysis of Variance Table
##
## Response: log(BPSysAve)
##                                                              Df Sum Sq Mean Sq
## Age                                                           1 1.9841 1.98407
## `I((Poverty^0.5 - 1)/0.5)`                                    1 0.1474 0.14737
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`   1 0.0001 0.00005
## Residuals                                                   482 6.2749 0.01302
##                                                              F value    Pr(>F)
## Age                                                         152.4037 < 2.2e-16
## `I((Poverty^0.5 - 1)/0.5)`                                   11.3203 0.0008278
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`   0.0039 0.9501031
## Residuals
##
## Age                                                         ***
## `I((Poverty^0.5 - 1)/0.5)`                                  ***
## `train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow`
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

calibration and training

```
ols.bic.smokenow <- ols(`log(BPSysAve)`  ~ ., data = as.data.frame(data.matrix(model.bic.vif.outliers.bo
               x=T, y=T, model = T)

## 10 fold cross validation ##
bic.cross.smoke.now <- calibrate(ols.bic.smokenow, method = "crossvalidation", B = 10)
#bic.boot <- calibrate(ols.bic, method = "boot", B = 10)
## Calibration plot ##
#pdf("bic_cross.pdf", height = 8, width = 16)
plot(bic.cross, las = 1, xlab = "Predicted BPSysAve", main = "Cross-Validation calibration for smokenow
```




```
##
## n=486   Mean absolute error=0.007   Mean squared error=1e-04
## 0.9 Quantile of absolute error=0.014
```

```
#plot(bic.boot,las = 1, xlab = "Predicted LPSA", main = "Bootstrapping calibration with BIC")
#dev.off()
test_minus_id.bic.transformation.smokenow <-  test_minus_id[-union(r.bic[[1]],r.bic[[4]]),]

g <- test_minus_id.bic.transformation.smokenow$BPSysAve
#gsub(",", "", g)
g2 <-gsub(",", "", g)
test_minus_id.bic.transformation.smokenow$`log(BPSysAve)` <- log(as.numeric(g2))
test_minus_id.bic.transformation.smokenow$`I((Poverty^0.5 - 1)/0.5)` <- I((test_minus_id.bic.transforma

`train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow` <- test_minus_id[-union(r.bic[[1]],r.bic[[4

## Test Error ##
train_minus_id <- train_minus_id %>% mutate('train_minus_id[-union(r.bic[[1]], r.bic[[4]]), ]$SmokeNow'
pred.bic.smokenow <- predict(ols.bic.smokenow, newdata = test_minus_id.bic.transformation.smokenow[,c(2
                                                                                                     18
                                                                                                     17
                                                                                                     16

                                                                                                     #
```

```
## Prediction error ##
```

prediction error

```
pred.error.BIC.smokenow <- mean((test_minus_id.bic.transformation.smokenow$`log(BPSysAve)` - pred.bic.sm
pred.error.BIC.smokenow
```

```
## [1] 0.01715312
```