model4

Liancheng

2023-11-25

```
rm(list = ls())
gc()
```

```
##          used (Mb) gc trigger (Mb) max used (Mb)
## Ncells 472000 25.3    1018141 54.4   660860 35.3
## Vcells 895369  6.9    8388608 64.0  1800812 13.8
```

```
set.seed(123)
library(car)
```

```
## Loading required package: carData
```

```
library(ggplot2)
library(olsrr)
```

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

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

```
############### (1) Data cleaning ##########################################
## select variables
library(NHANES)
df0 <- NHANES
df <- NHANES[NHANES$Age >= 18 & NHANES$Age < 60,]
# colSums(is.na(df)) / nrow(df)
df <- df[, which(colSums(is.na(df)) / nrow(df) < 0.3)]
# exclude duplication
df <- df[!duplicated(df),]
names(df)
```

```
##  [1] "ID"              "SurveyYr"        "Gender"          "Age"
##  [5] "AgeDecade"       "Race1"           "Education"       "MaritalStatus"
##  [9] "HHIncome"        "HHIncomeMid"     "Poverty"         "HomeRooms"
## [13] "HomeOwn"         "Work"            "Weight"          "Height"
## [17] "BMI"             "BMI_WHO"         "Pulse"           "BPSysAve"
## [21] "BPDiaAve"        "BPSys1"          "BPDia1"          "BPSys2"
## [25] "BPDia2"          "BPSys3"          "BPDia3"          "DirectChol"
## [29] "TotChol"         "UrineVol1"       "UrineFlow1"      "Diabetes"
## [33] "HealthGen"       "DaysPhysHlthBad" "DaysMentHlthBad" "LittleInterest"
## [37] "Depressed"       "SleepHrsNight"   "SleepTrouble"    "PhysActive"
## [41] "Alcohol12PlusYr" "AlcoholYear"     "Smoke100"        "Smoke100n"
## [45] "Marijuana"       "RegularMarij"    "HardDrugs"       "SexEver"
```

```
## [49] "SexAge"          "SexNumPartnLife" "SexNumPartYear"  "SameSex"
## [53] "SexOrientation"
```

```
# df$BPSysAve
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:car':
##
##     recode
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

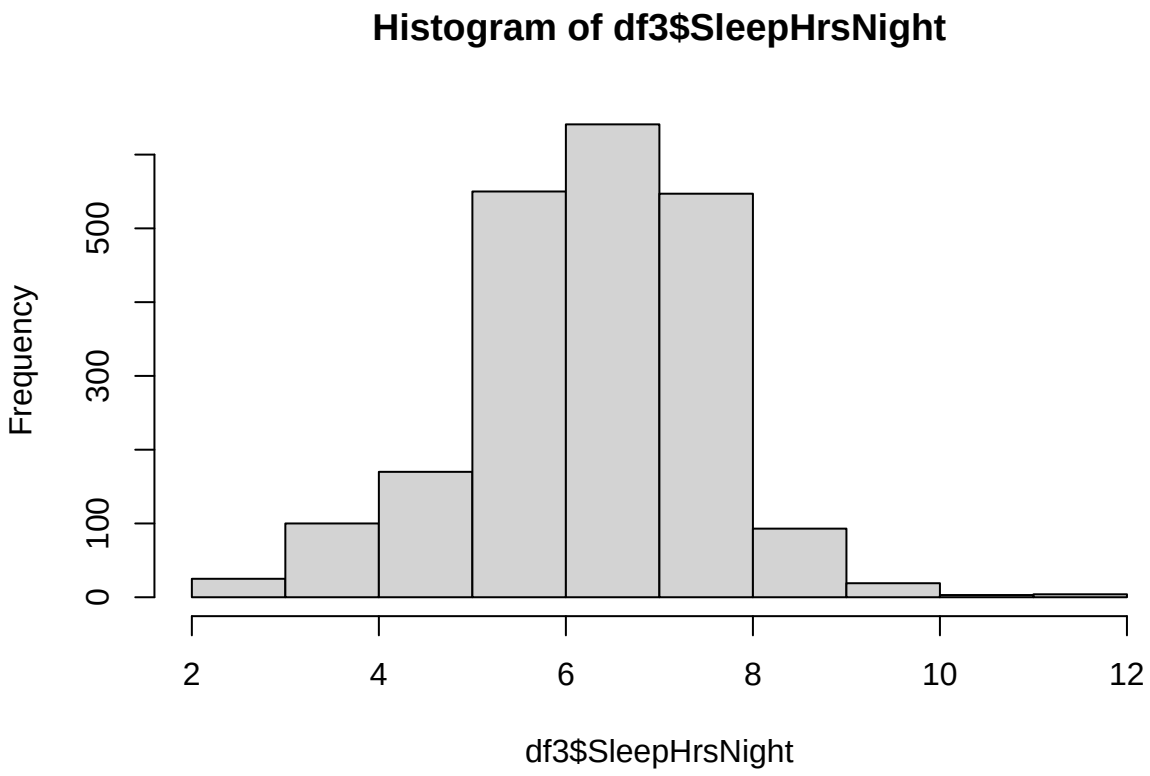
```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
df2 <- df %>% select(
  SleepHrsNight,
  BMI,
  DirectChol,
  Age,
  Gender,
  Race1,
  TotChol,
  BPDiaAve,
  BPSysAve,
  AlcoholYear,
  Poverty,
  SexNumPartnLife,
  SexNumPartYear,
  DaysMentHlthBad,
  UrineFlow1,
  PhysActive,
  DaysPhysHlthBad,
  Smoke100,
  Depressed,
  HealthGen,
  SexAge
)

df3 <- na.omit(df2)
#df3$SleepHrsNight <- df3$SleepHrsNight * 60
#df3 <- df3[, -which(names(df3) %in% "SleepHrsNight")]
# cor(df3$BPSysAve,df3$BPDiaAve)
psych::describe(df3)
```

```
##                vars    n   mean    sd median trimmed   mad   min     max
## SleepHrsNight     1 2152   6.78  1.31   7.00    6.85  1.48  2.00   12.00
## BMI               2 2152  28.77  6.75  27.60   28.09  5.78 15.02   69.00
## DirectChol        3 2152   1.35  0.41   1.29    1.31  0.39  0.39    3.83
## Age               4 2152  39.18 11.33  39.00   39.15 14.83 20.00   59.00
## Gender*           5 2152   1.53  0.50   2.00    1.54  0.00  1.00    2.00
```

```
## Race1*              6 2152   3.43  1.15   4.00    3.57  0.00  1.00    5.00
## TotChol             7 2152   5.07  1.05   4.99    5.01  1.04  1.53   13.65
## BPDiaAve            8 2152  71.19 11.84  71.00   71.28 10.38  0.00  116.00
## BPSysAve            9 2152 117.43 14.28 116.00  116.50 13.34 78.00  209.00
## AlcoholYear        10 2152  70.59 94.22  24.00   50.94 35.58  0.00  364.00
## Poverty            11 2152   2.84  1.69   2.78    2.89  2.49  0.00    5.00
## SexNumPartnLife    12 2152  16.73 66.13   7.00    8.91  5.93  0.00 2000.00
## SexNumPartYear     13 2152   1.38  2.59   1.00    1.04  0.00  0.00   69.00
## DaysMentHlthBad    14 2152   4.47  8.02   0.00    2.40  0.00  0.00   30.00
## UrineFlow1         15 2152   1.07  0.97   0.81    0.91  0.60  0.00   10.14
## PhysActive*        16 2152   1.58  0.49   2.00    1.60  0.00  1.00    2.00
## DaysPhysHlthBad    17 2152   3.16  7.19   0.00    1.12  0.00  0.00   30.00
## Smoke100*          18 2152   1.46  0.50   1.00    1.45  0.00  1.00    2.00
## Depressed*         19 2152   1.30  0.58   1.00    1.16  0.00  1.00    3.00
## HealthGen*         20 2152   2.64  0.94   3.00    2.65  1.48  1.00    5.00
## SexAge             21 2152  17.10  3.39  17.00   16.80  2.97  9.00   44.00
##                    range  skew kurtosis   se
## SleepHrsNight      10.00 -0.30     0.69 0.03
## BMI                53.98  1.28     2.96 0.15
## DirectChol          3.44  1.09     2.27 0.01
## Age                39.00  0.02    -1.15 0.24
## Gender*             1.00 -0.12    -1.99 0.01
## Race1*              4.00 -1.13     0.08 0.02
## TotChol            12.12  0.92     3.47 0.02
## BPDiaAve          116.00 -0.39     3.13 0.26
## BPSysAve          131.00  1.00     2.94 0.31
## AlcoholYear       364.00  1.66     1.98 2.03
## Poverty             5.00 -0.01    -1.47 0.04
## SexNumPartnLife  2000.00 18.82   456.62 1.43
## SexNumPartYear     69.00 14.07   293.16 0.06
## DaysMentHlthBad    30.00  2.16     3.76 0.17
## UrineFlow1         10.14  2.89    14.06 0.02
## PhysActive*         1.00 -0.32    -1.90 0.01
## DaysPhysHlthBad    30.00  2.80     7.06 0.15
## Smoke100*           1.00  0.15    -1.98 0.01
## Depressed*          2.00  1.83     2.21 0.01
## HealthGen*          4.00  0.11    -0.33 0.02
## SexAge             35.00  1.51     5.56 0.07
```

```
# psych::pairs.panels(df3)
hist(df3$SleepHrsNight)
```




```
# colSums(is.na(df2)) / nrow(df2)
fit0 <-
  lm(SleepHrsNight ~ .,
     data = df3)
#data type
df3$Gender <- ifelse(df3$Gender == "male", 0, 1)
df3$Smoke100 <- ifelse(df3$Smoke100 == "No", 0, 1)
df3$PhysActive <- ifelse(df3$PhysActive == "No", 0, 1)
df3 <- df3 %>%
  mutate(
    Race1 = case_when(
      Race1 == 'Black' ~ 1,
      Race1 == 'Hispanic' ~ 2,
      Race1 == 'Mexican' ~ 3,
      Race1 == 'White' ~ 4,
      Race1 == 'Other' ~ 5,
      TRUE ~ NA_integer_  # Default value if none of the conditions are met
    )
  )

## model_2 add known risk factors ##
m_full = lm(
  BMI ~ SleepHrsNight + Age + Gender + Race1  + Poverty + TotChol + BPDiaAve + BPSysAve + AlcoholYear +
    DaysPhysHlthBad + HealthGen + PhysActive + SleepHrsNight * Age + SleepHrsNight *
    Gender,
  df3
```

```
)
summary(m_full)
```

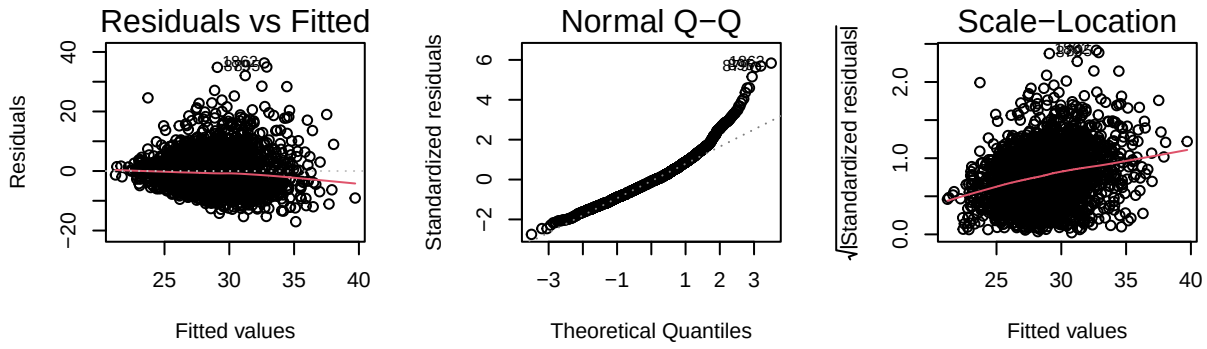
```
##
## Call:
## lm(formula = BMI ~ SleepHrsNight + Age + Gender + Race1 + Poverty +
##     TotChol + BPDiaAve + BPSysAve + AlcoholYear + Smoke100 +
##     UrineFlow1 + DaysMentHlthBad + DaysPhysHlthBad + HealthGen +
##     PhysActive + SleepHrsNight * Age + SleepHrsNight * Gender,
##     data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.019  -4.059  -0.648   3.165  36.301
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)          21.349305   2.996916   7.124 1.43e-12 ***
## SleepHrsNight        -0.542616   0.378594  -1.433  0.15194
## Age                  -0.105136   0.062783  -1.675  0.09416 .
## Gender                3.768696   1.435161   2.626  0.00870 **
## Race1                -0.503222   0.121964  -4.126 3.83e-05 ***
## Poverty               0.072729   0.090968   0.800  0.42409
## TotChol               0.014773   0.135905   0.109  0.91345
## BPDiaAve              0.058709   0.013701   4.285 1.91e-05 ***
## BPSysAve              0.054450   0.011792   4.617 4.12e-06 ***
## AlcoholYear          -0.008396   0.001513  -5.549 3.23e-08 ***
## Smoke100             -0.802999   0.286852  -2.799  0.00517 **
## UrineFlow1           -0.102218   0.142435  -0.718  0.47305
## DaysMentHlthBad      -0.030250   0.017962  -1.684  0.09230 .
## DaysPhysHlthBad       0.015142   0.020943   0.723  0.46975
## HealthGenVgood        1.928283   0.470249   4.101 4.28e-05 ***
## HealthGenGood         3.559316   0.468010   7.605 4.24e-14 ***
## HealthGenFair         5.299570   0.575060   9.216  < 2e-16 ***
## HealthGenPoor         7.640142   1.077494   7.091 1.81e-12 ***
## PhysActive           -0.837418   0.294615  -2.842  0.00452 **
## SleepHrsNight:Age     0.017092   0.009024   1.894  0.05837 .
## SleepHrsNight:Gender -0.477032   0.206903  -2.306  0.02123 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.242 on 2131 degrees of freedom
## Multiple R-squared:  0.1538, Adjusted R-squared:  0.1459
## F-statistic: 19.37 on 20 and 2131 DF,  p-value: < 2.2e-16
```

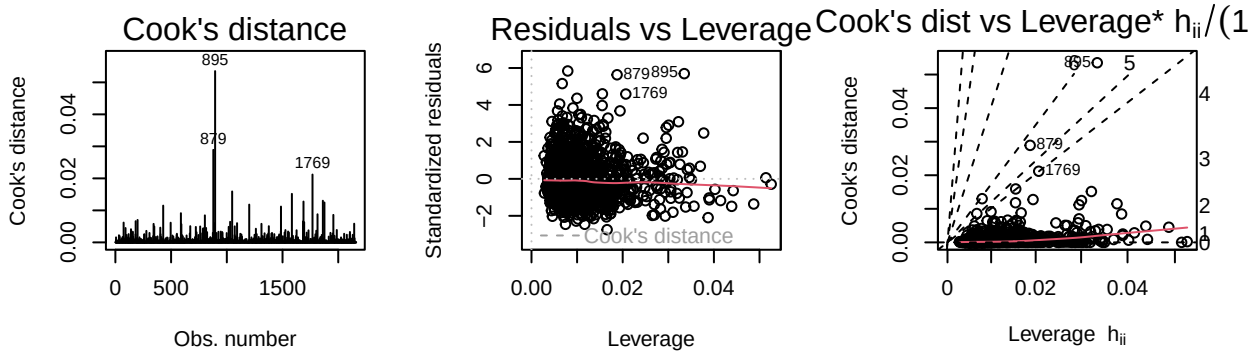
```
car::Anova(m_full, type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: BMI
##                       Sum Sq   Df F value    Pr(>F)
## (Intercept)             1977    1 50.7479 1.431e-12 ***
## SleepHrsNight             80    1  2.0542  0.151936
## Age                      109    1  2.8043  0.094161 .
```

```
## Gender                   269    1  6.8957  0.008702 **
## Race1                    663    1 17.0238 3.833e-05 ***
## Poverty                   25    1  0.6392  0.424089
## TotChol                    0    1  0.0118  0.913452
## BPDiaAve                 715    1 18.3611 1.909e-05 ***
## BPSysAve                 831    1 21.3201 4.118e-06 ***
## AlcoholYear             1200    1 30.7921 3.229e-08 ***
## Smoke100                 305    1  7.8363  0.005167 **
## UrineFlow1                20    1  0.5150  0.473054
## DaysMentHlthBad          110    1  2.8364  0.092299 .
## DaysPhysHlthBad           20    1  0.5228  0.469750
## HealthGen               4548    4 29.1844 < 2.2e-16 ***
## PhysActive               315    1  8.0793  0.004520 **
## SleepHrsNight:Age        140    1  3.5871  0.058366 .
## SleepHrsNight:Gender     207    1  5.3157  0.021230 *
## Residuals              83018 2131
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
########### model 2 diagnosis ###########
par(mfrow = c(2, 3)) #read more from ?plot.lm
plot(m_full, which = 1)
plot(m_full, which = 2)
plot(m_full, which = 3)
plot(m_full, which = 4)
plot(m_full, which = 5)
plot(m_full, which = 6)
```






```
par(mfrow = c(1, 1)) # reset

m_full.yhat = m_full$fitted.values
m_full.res = m_full$residuals
m_full.h = hatvalues(m_full)
m_full.r = rstandard(m_full)
m_full.rr = rstudent(m_full)
#which subject is most outlying with respect to the x space
Hmisc::describe(m_full.h)
```

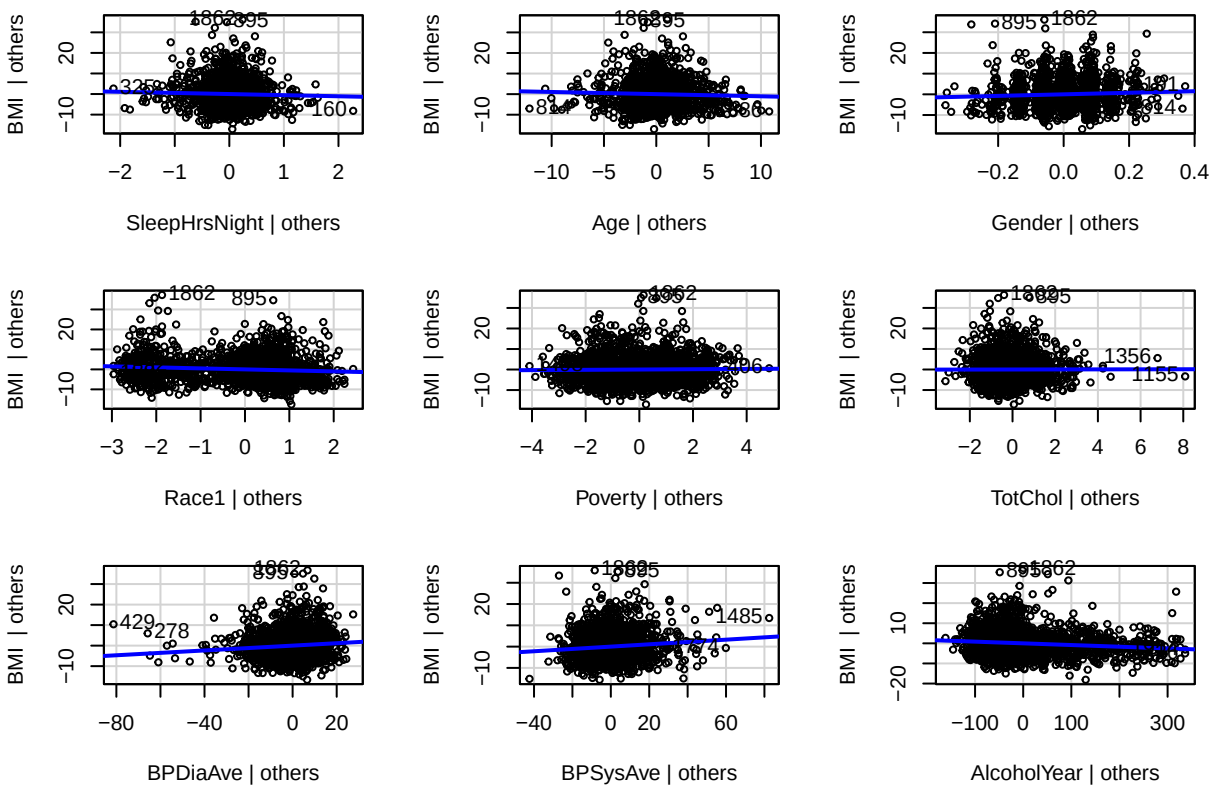
```
## m_full.h
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     2152        0     2152        1 0.009758 0.005425 0.004453 0.005041
##      .25      .50      .75      .90      .95
## 0.006148 0.008122 0.011389 0.016108 0.020491
##
## lowest : 0.002781565 0.002839218 0.003194507 0.003255413 0.003296775
## highest: 0.043894006 0.044237670 0.048729461 0.051407460 0.052560897
```

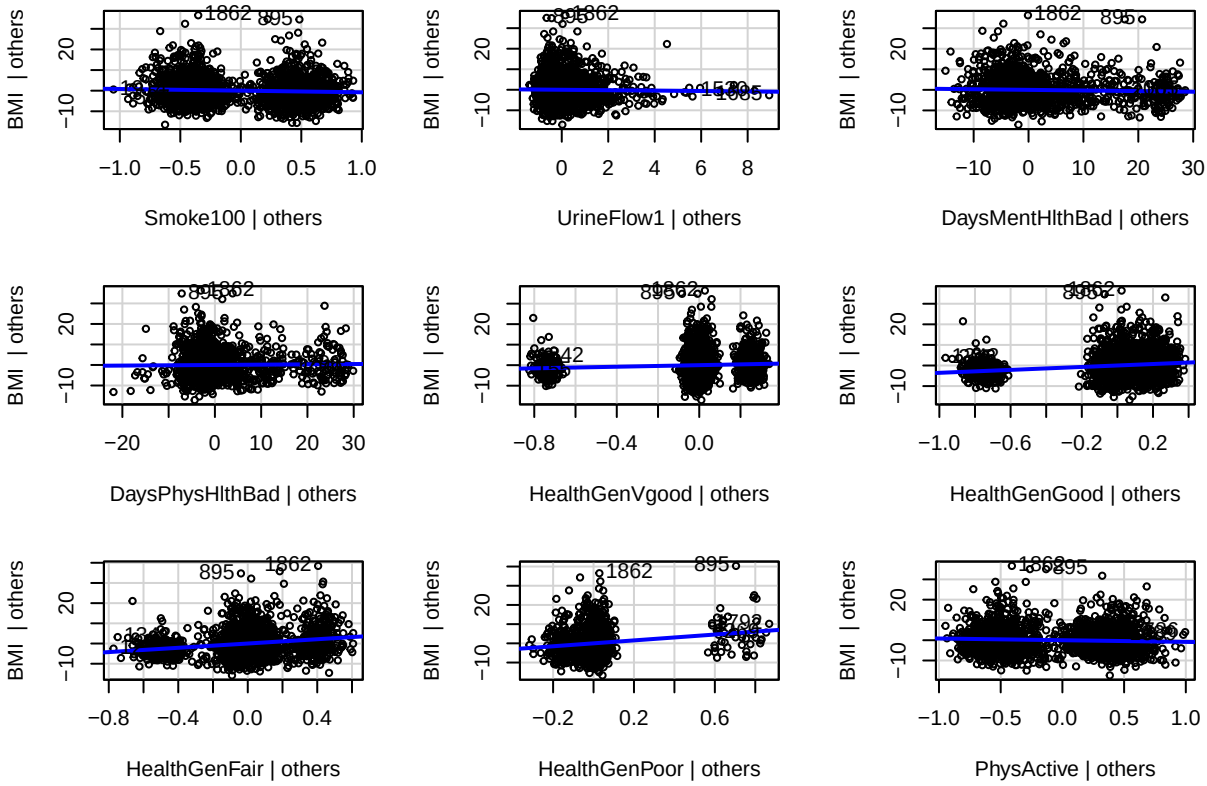
```
m_full.h[which.max(m_full.h)]
```

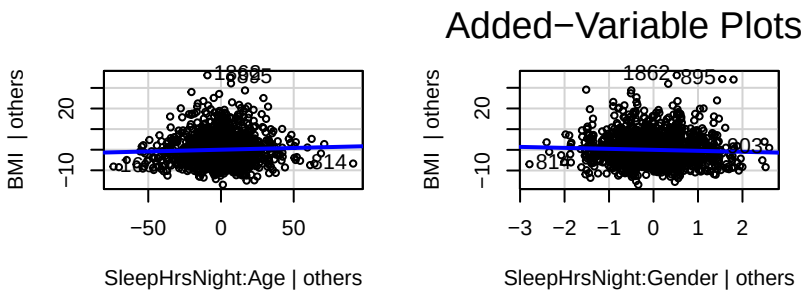
```
##      1685
## 0.0525609
```

```
###################### Assumption:LINE ##############################

#(1)Linear: 2 approaches
```

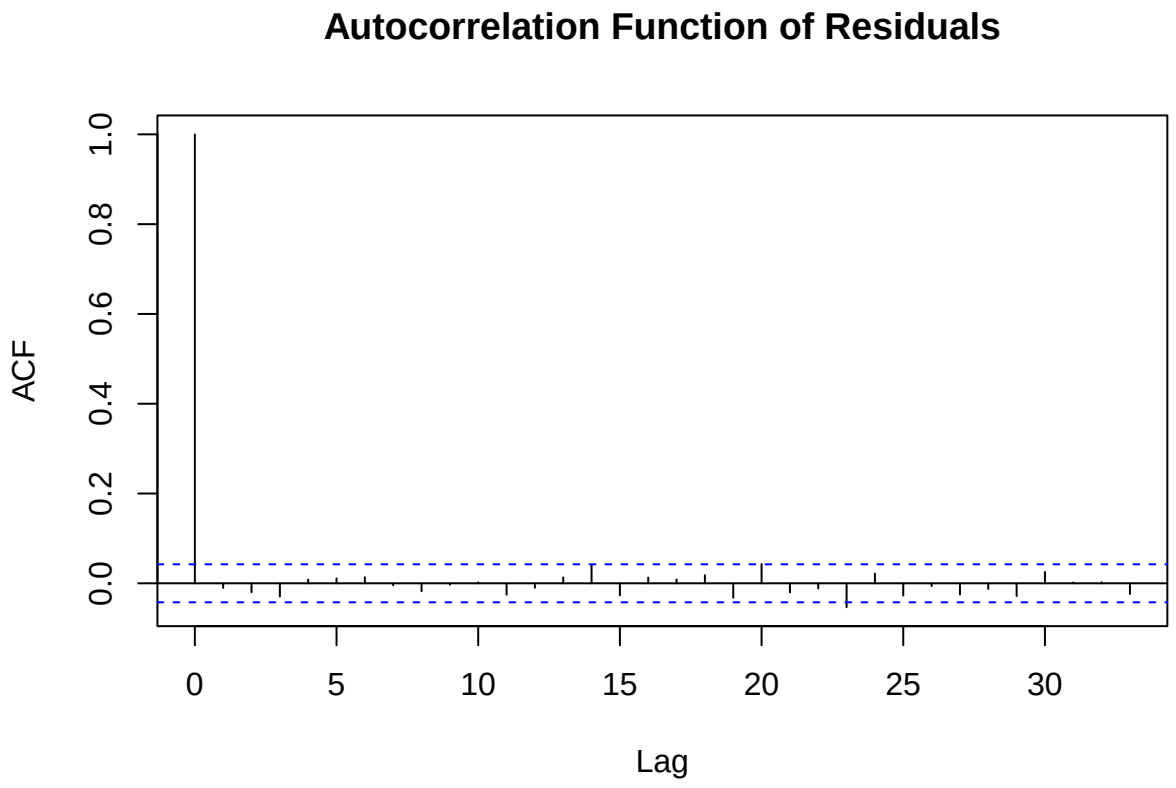
```
# partial regression plots
car::avPlots(m_full)
```



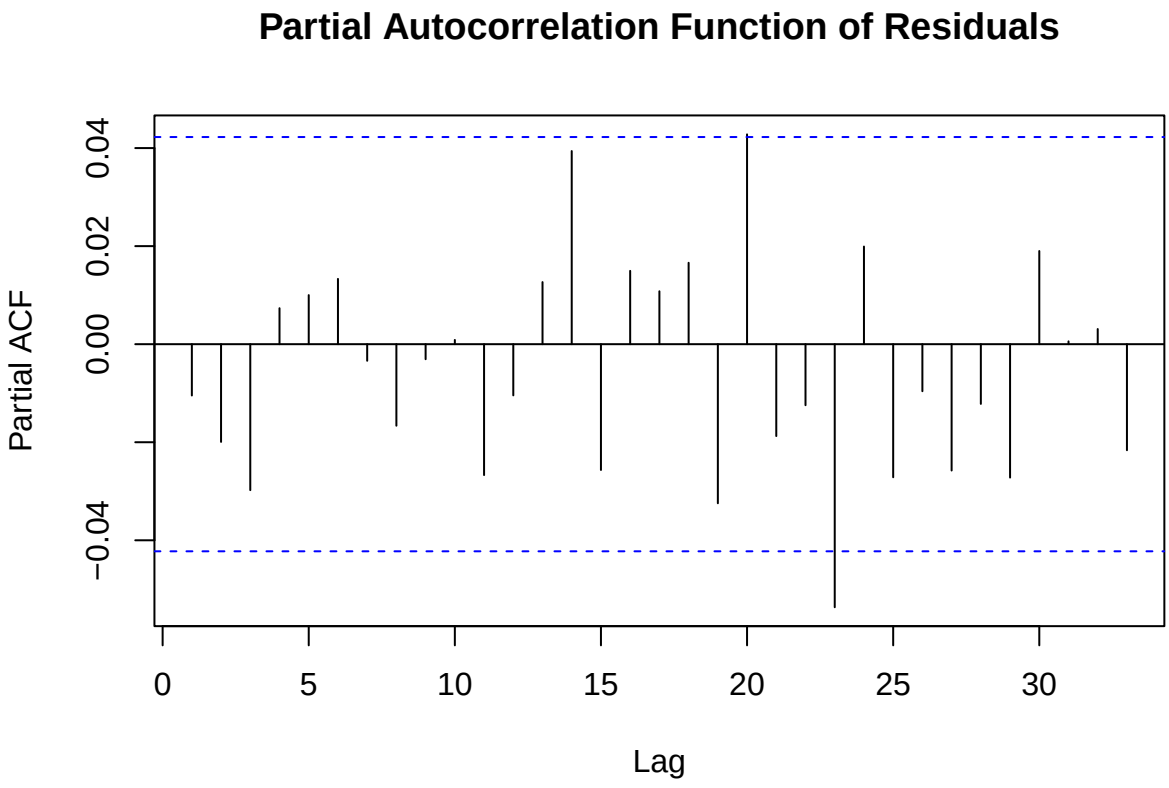

BMI | others
Smoke100 | others
UrineFlow1 | others
DaysMentHlthBad | others
DaysPhysHlthBad | others
HealthGenVgood | others
HealthGenGood | others
HealthGenFair | others
HealthGenPoor | others
PhysActive | others



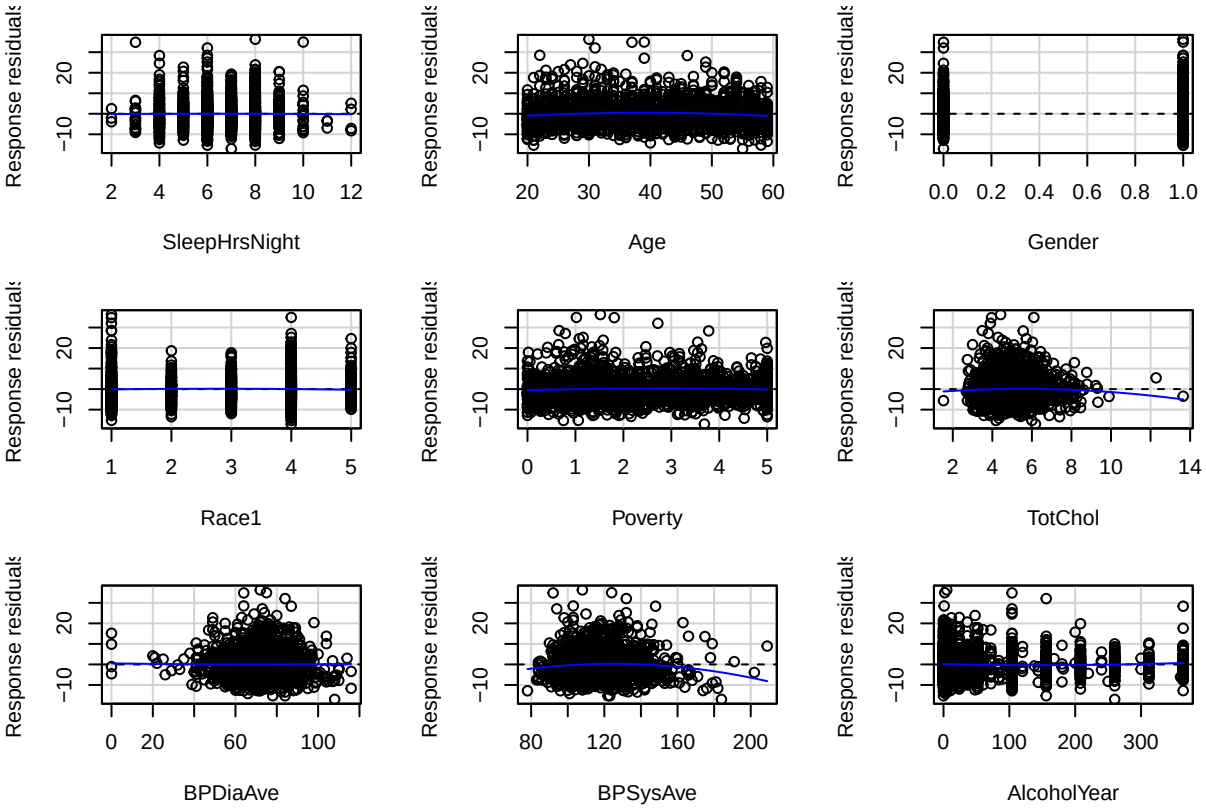

```
#(2)Independence:

residuals <- resid(m_full)
acf(residuals, main = "Autocorrelation Function of Residuals")
```

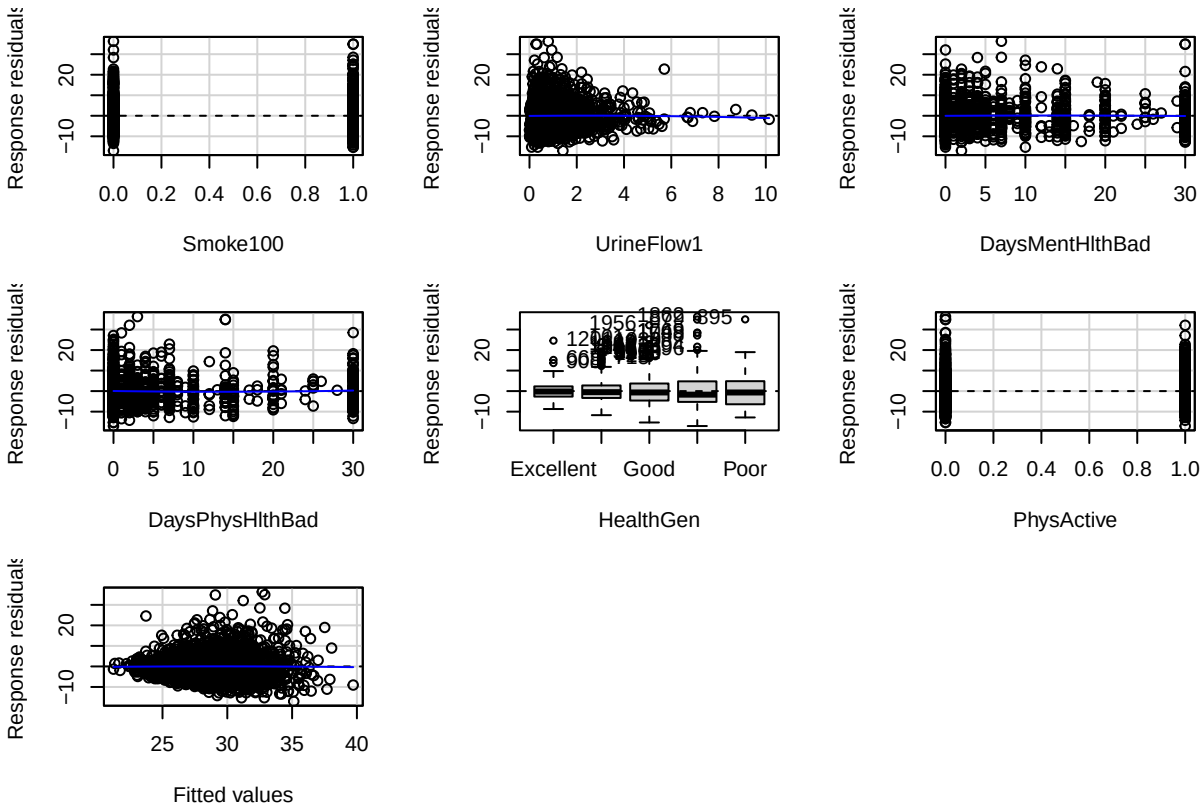



```
pacf(residuals, main = "Partial Autocorrelation Function of Residuals")
```



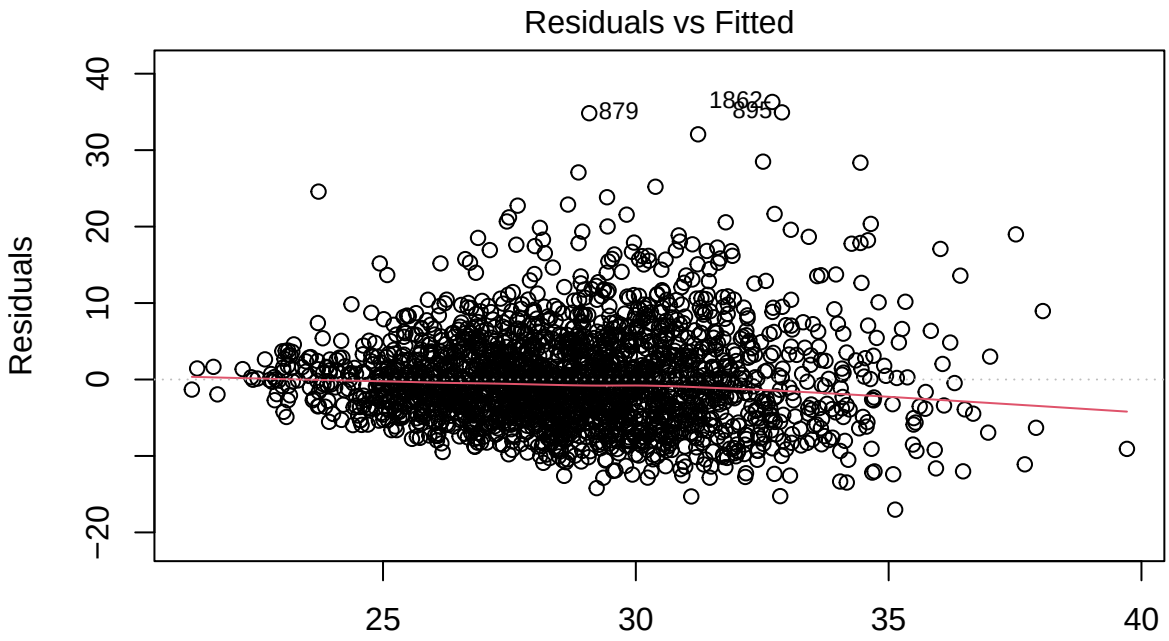

```
#(3)E: constant var: residuals-fitted values; transform for variance-stable...(total: 4 solutions)

car::residualPlots(m_full, type = "response")
```


Response residuals
SleepHrsNight
Age
Gender
Race1
Poverty
TotChol
BPDiaAve
BPSysAve
AlcoholYear

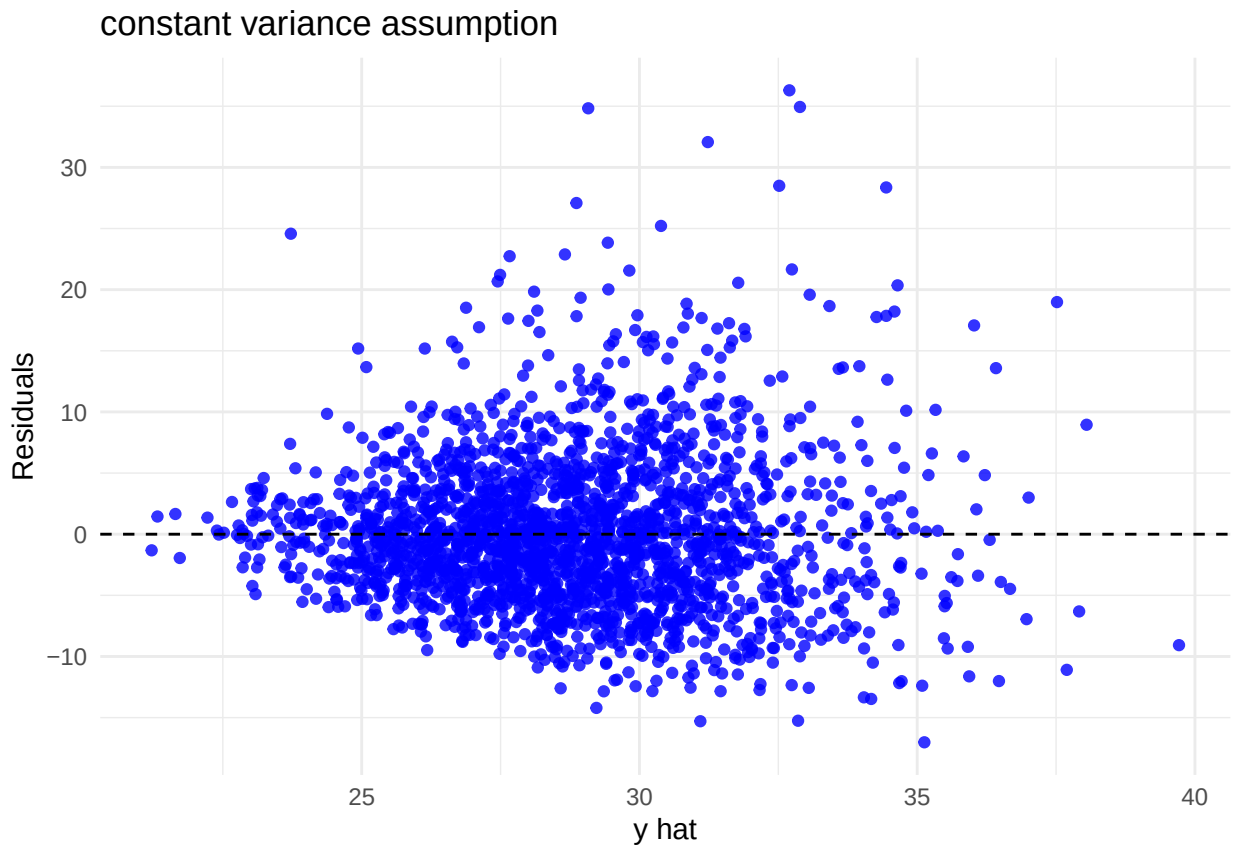



```
##                  Test stat Pr(>|Test stat|)
## SleepHrsNight      -0.1682        0.8664592
## Age                -3.6818        0.0002374 ***
## Gender              0.3790        0.7047535
## Race1              -0.9121        0.3618251
## Poverty            -1.4371        0.1508401
## TotChol            -1.3948        0.1632108
## BPDiaAve            0.2900        0.7718192
## BPSysAve           -3.6373        0.0002820 ***
## AlcoholYear         1.7583        0.0788482 .
## Smoke100            0.6279        0.5301459
## UrineFlow1         -0.3979        0.6907472
## DaysMentHlthBad    -0.4167        0.6769121
## DaysPhysHlthBad     0.9004        0.3680075
## HealthGen
## PhysActive         -0.0387        0.9691385
## Tukey test         -0.3128        0.7544028
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(m_full, which = 1)
```




```
#or
ggplot(m_full, aes(x = m_full.yhat, y = m_full.res)) +
  geom_point(color = "blue", alpha = 0.8) +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "black") +
  labs(title = "constant variance assumption",
       x = "y hat",
       y = "Residuals") +
  theme_minimal()
```



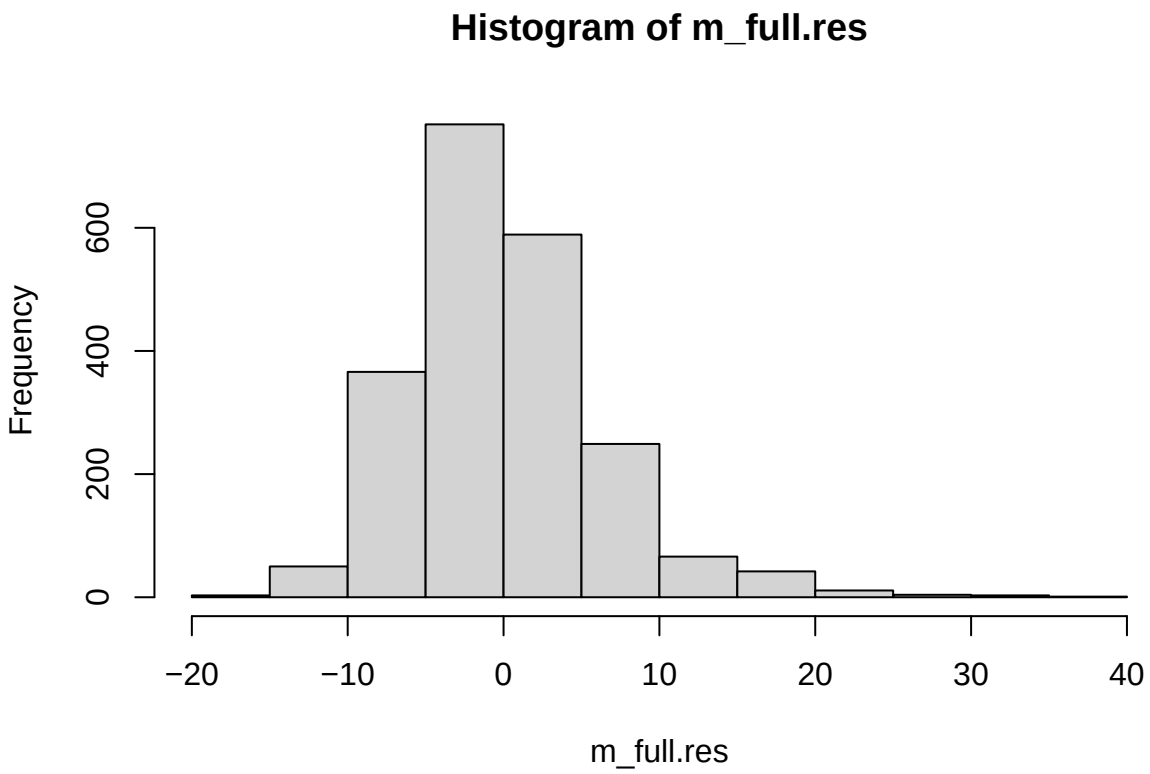

```
#conclusion: the constant variance assumption is basically not violated. The spread of the residuals ap


#(4)Normality: residuals freq - residuals (4 plots: his, box, Q-Q, shapiro); transform

#exam quartiles of the residuals
Hmisc::describe(m_full.res)
```

```
## m_full.res
##          n    missing   distinct       Info       Mean        Gmd        .05
##       2152          0       2152          1 -1.446e-16      6.685    -8.5512
##        .10        .25        .50        .75        .90        .95
##    -7.1412    -4.0587    -0.6485     3.1655     7.5311    10.6716
##
## lowest : -17.01907 -15.29660 -15.25576 -14.20568 -13.47199
## highest:  28.49374  32.06884  34.83171  34.94008  36.30097
```

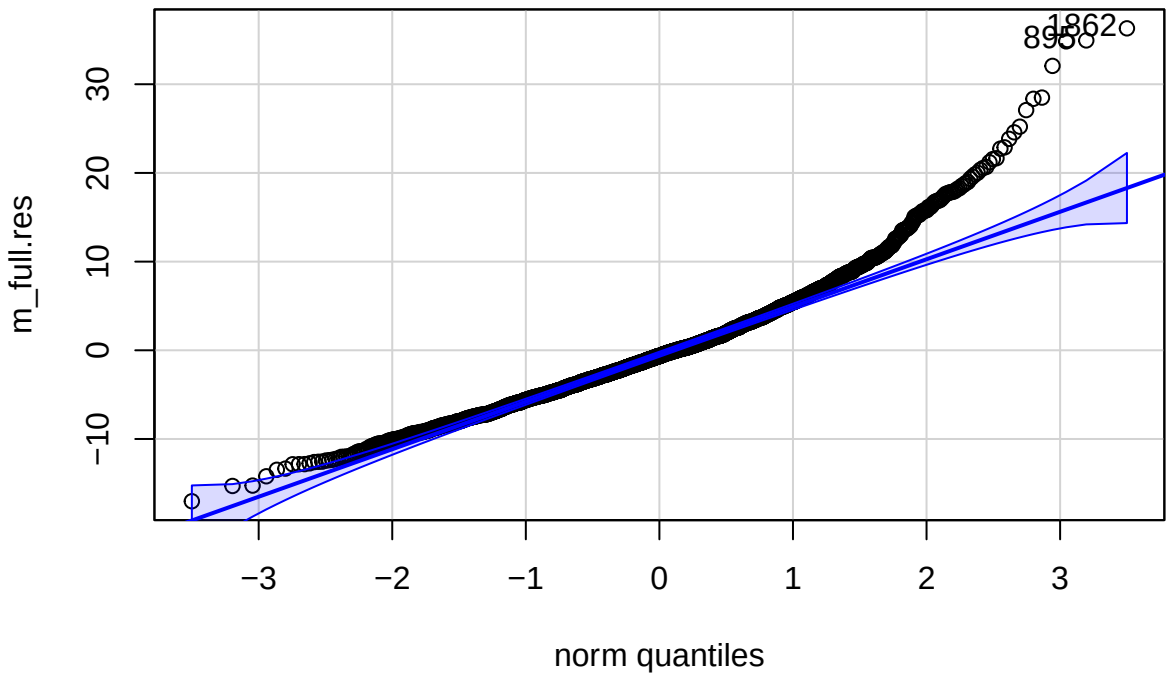
```
Hmisc::describe(m_full.res)$counts[c(".25", ".50", ".75")] #not symmetric
```

```
##       .25       .50       .75
## "-4.0587" "-0.6485" " 3.1655"
```

```
#histogram
par(mfrow = c(1, 1))
hist(m_full.res, breaks = 15)
```



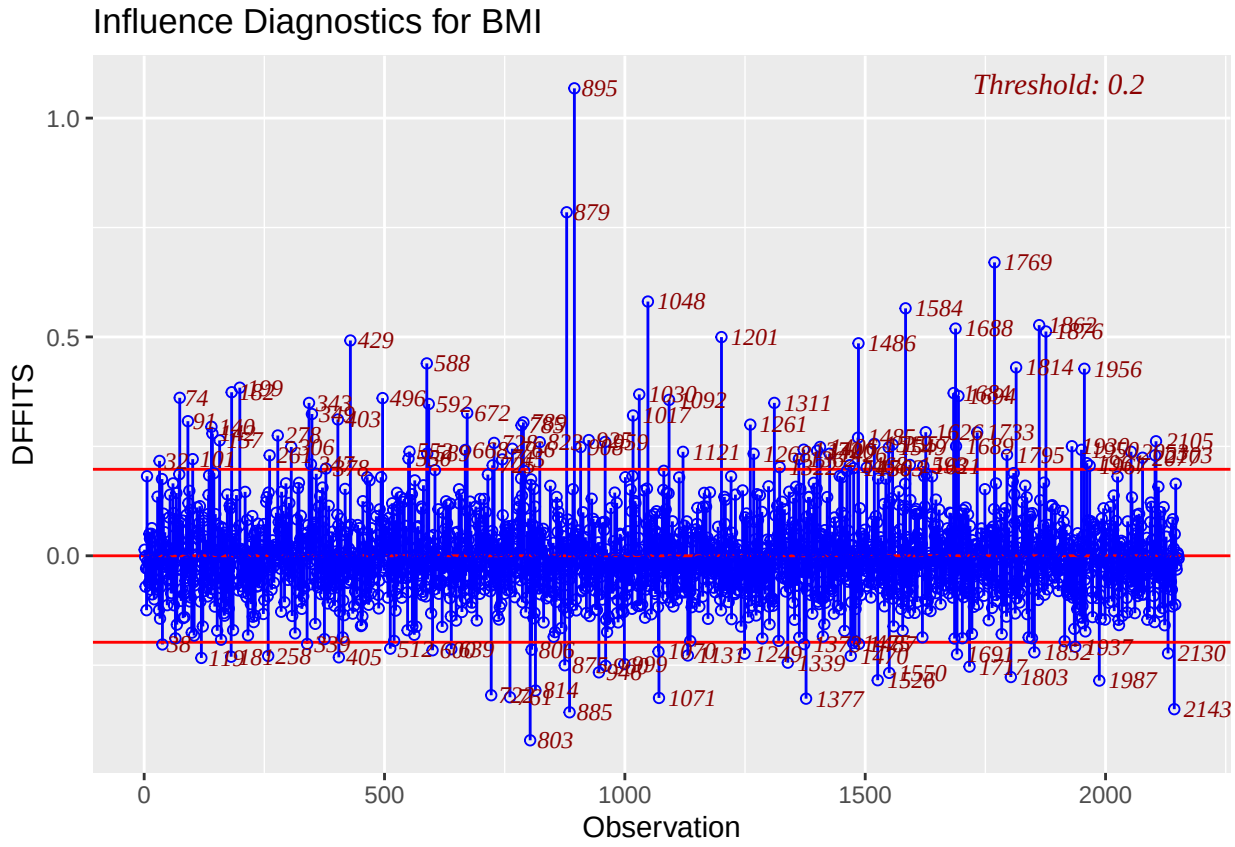

```
# Q-Q plot
qq.m_full.res = car::qqPlot(m_full.res)
```




```
m_full.res[qq.m_full.res]
```

```
##     1862      895
## 36.30097 34.94008
```

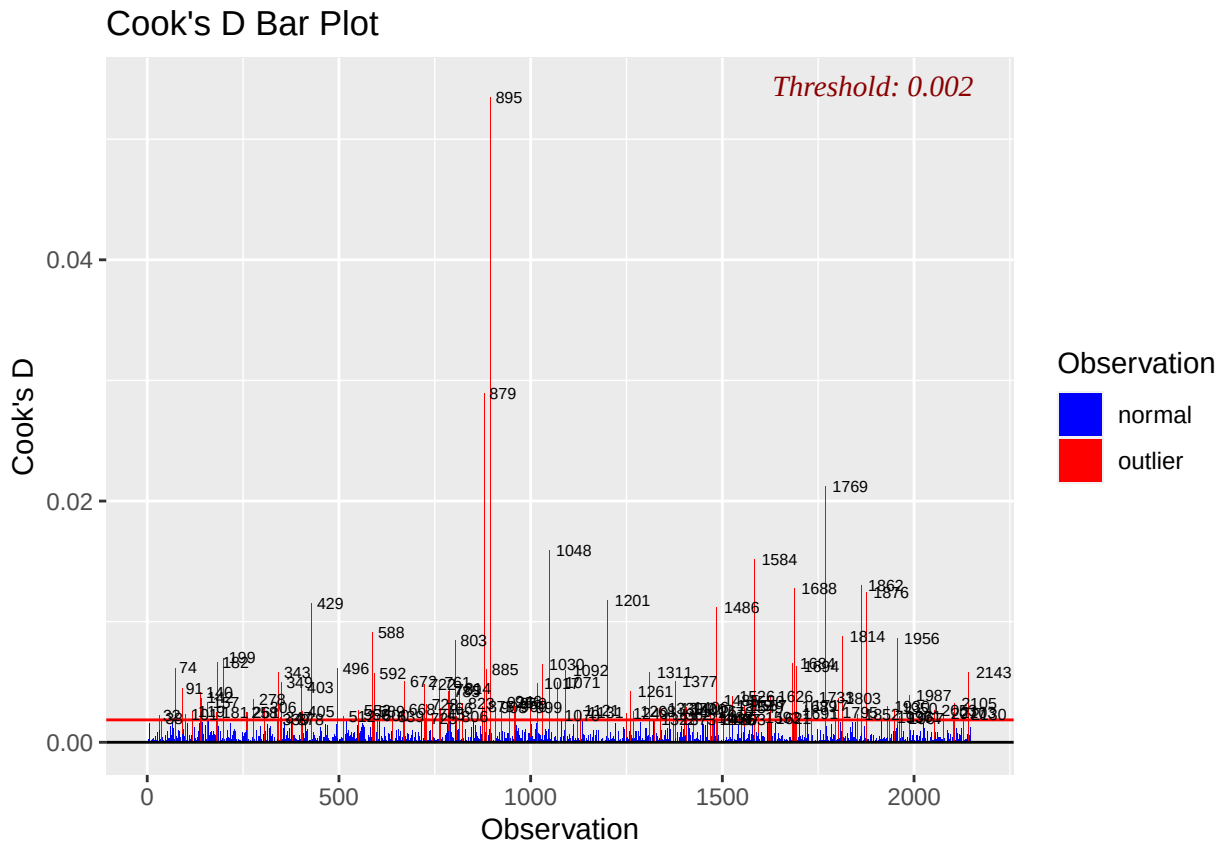
```
################ influential observations  ##################

influence2 = data.frame(
  Residual = resid(m_full),
  Rstudent = rstudent(m_full),
  HatDiagH = hat(model.matrix(m_full)),
  CovRatio = covratio(m_full),
  DFFITS = dffits(m_full),
  COOKsDistance = cooks.distance(m_full)
)
# DFFITS
ols_plot_dffits(m_full)
```




```
influence2[order(abs(influence2$DFFITS), decreasing = T), ] %>% head()
```

```
##      Residual Rstudent    HatDiagH  CovRatio    DFFITS COOKsDistance
## 895  34.94008 5.736619 0.033498837 0.7572647 1.0679961    0.05351373
## 879  34.83171 5.674807 0.018766799 0.7510219 0.7848020    0.02890599
## 1769 28.35783 4.612895 0.020681924 0.8369471 0.6703585    0.02119736
## 1048 28.49374 4.622993 0.015543867 0.8318213 0.5809043    0.01591687
## 1584 18.98322 3.097879 0.032232760 0.9495341 0.5653635    0.01515960
## 1862 36.30097 5.885164 0.007952567 0.7255062 0.5269216    0.01301582
```

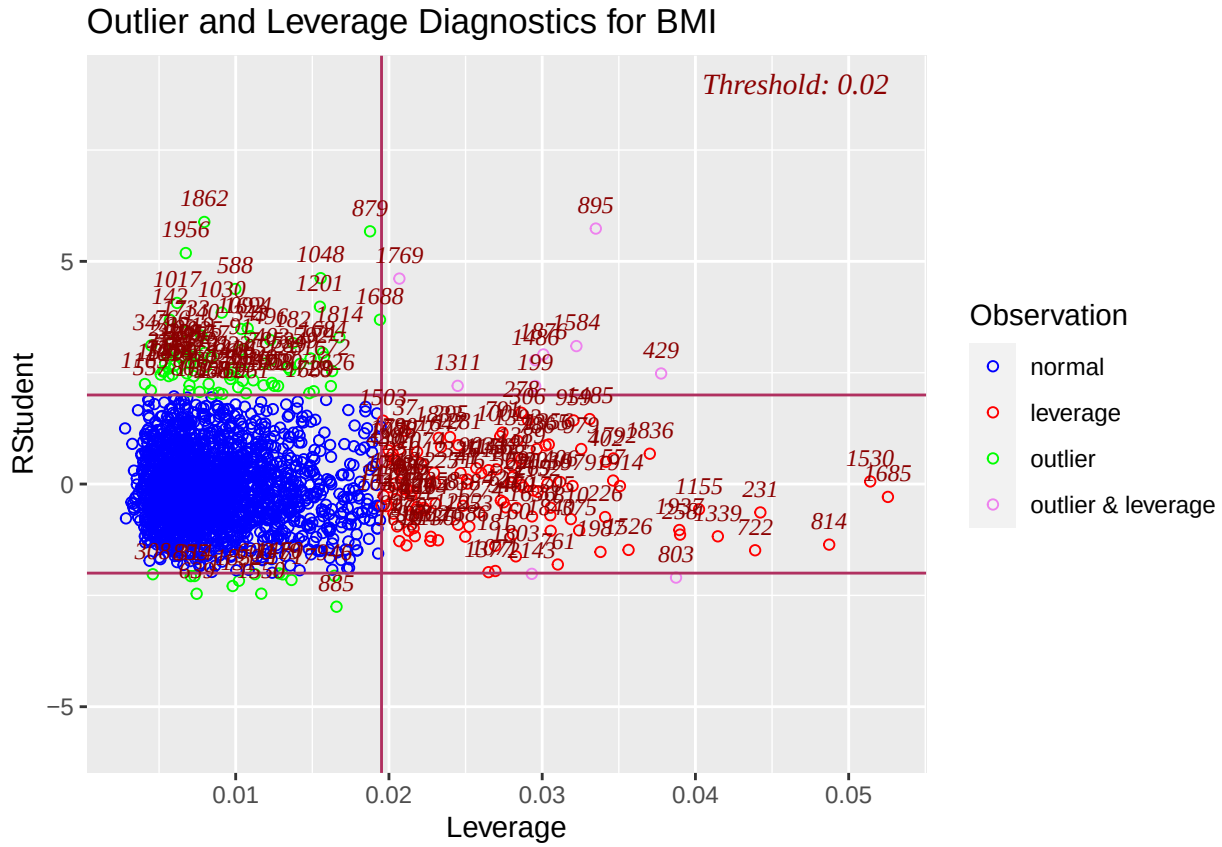
```
#From the plot above, we can see 2 observations with the largest (magnitude) of DFFITS, observation 879

# Cook's D
ols_plot_cooksd_bar(m_full)
```




```
influence2[order(influence2$COOKsDistance, decreasing = T), ] %>% head()
```

```
##      Residual Rstudent    HatDiagH  CovRatio    DFFITS COOKsDistance
## 895  34.94008 5.736619 0.033498837 0.7572647 1.0679961    0.05351373
## 879  34.83171 5.674807 0.018766799 0.7510219 0.7848020    0.02890599
## 1769 28.35783 4.612895 0.020681924 0.8369471 0.6703585    0.02119736
## 1048 28.49374 4.622993 0.015543867 0.8318213 0.5809043    0.01591687
## 1584 18.98322 3.097879 0.032232760 0.9495341 0.5653635    0.01515960
## 1862 36.30097 5.885164 0.007952567 0.7255062 0.5269216    0.01301582
```

```
#From the plot above, we can see that the observation 879 and 1769 also have the largest Cook's Distanc

#leverage
ols_plot_resid_lev(m_full)
```




```
#high leverage
influence2[order(influence2$HatDiagH, decreasing = T), ] %>% head()
```

```
##        Residual    Rstudent   HatDiagH CovRatio      DFFITS COOKsDistance
## 1685 -1.7567114 -0.28909280 0.05256090 1.065054 -0.06809152  2.208785e-04
## 1530  0.3556433  0.05848965 0.05140746 1.064600  0.01361609  8.832597e-06
## 814  -8.2792096 -1.36028339 0.04872946 1.042455 -0.30787411  4.511841e-03
## 231  -3.8895732 -0.63734177 0.04423767 1.052426 -0.13711771  8.955479e-04
## 722  -9.0732124 -1.48709041 0.04389401 1.033501 -0.31863022  4.831787e-03
## 1339 -7.1745361 -1.17417737 0.04146227 1.039370 -0.24420562  2.839323e-03
```

```
#high studentized residual
influence2[order(influence2$Rstudent, decreasing = T), ] %>% head()
```

```
##      Residual Rstudent    HatDiagH  CovRatio    DFFITS COOKsDistance
## 1862 36.30097 5.885164 0.007952567 0.7255062 0.5269216   0.013015818
## 895  34.94008 5.736619 0.033498837 0.7572647 1.0679961   0.053513733
## 879  34.83171 5.674807 0.018766799 0.7510219 0.7848020   0.028905986
## 1956 32.06884 5.186586 0.006742466 0.7811915 0.4273267   0.008591205
## 1048 28.49374 4.622993 0.015543867 0.8318213 0.5809043   0.015916873
## 1769 28.35783 4.612895 0.020681924 0.8369471 0.6703585   0.021197359
```

```
#From the plot above, we can see that the observation 1155 has the largest leverage (0.0368). Observati


#From the plot above, there is 7 observations(1048,1769,1684, 74, 72, 1689, 1311) located in the inters
#The thresholds for the externally studentized residual are -2 and 2, i.e. 2 in magnitude. The threshol
```

```r
#From (DFFITS), observations 879 and 1769 appear to be influential observations. Observation 1155 has es

rm2.df3 = df3[-c(879, 1769, 1155, 1048, 1769, 1684, 74, 72, 1689, 1311), ]
rm.m_full =  lm(
  BMI ~ SleepHrsNight + Age + Gender + Race1 + TotChol + BPDiaAve + BPSysAve + AlcoholYear + Smoke100 +
    DaysPhysHlthBad + PhysActive,
  rm2.df3
)
## Before removing these observations, the estimated coefficients are:
summary(m_full)$coef
```

```
##                          Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)          21.349305448 2.996916146  7.1237580 1.430603e-12
## SleepHrsNight        -0.542616261 0.378593774 -1.4332414 1.519356e-01
## Age                  -0.105135734 0.062782895 -1.6745920 9.416104e-02
## Gender                3.768696209 1.435160571  2.6259753 8.701945e-03
## Race1                -0.503221762 0.121963771 -4.1259938 3.833266e-05
## Poverty               0.072728633 0.090967555  0.7995008 4.240892e-01
## TotChol               0.014772748 0.135905287  0.1086988 9.134516e-01
## BPDiaAve              0.058709170 0.013701124  4.2849894 1.908793e-05
## BPSysAve              0.054449503 0.011792323  4.6173687 4.117883e-06
## AlcoholYear          -0.008395832 0.001513018 -5.5490621 3.229358e-08
## Smoke100             -0.802999282 0.286852351 -2.7993470 5.166730e-03
## UrineFlow1           -0.102217843 0.142434820 -0.7176465 4.730540e-01
## DaysMentHlthBad      -0.030249912 0.017961509 -1.6841521 9.229873e-02
## DaysPhysHlthBad       0.015141809 0.020942592  0.7230150 4.697500e-01
## HealthGenVgood        1.928282830 0.470248557  4.1005609 4.276195e-05
## HealthGenGood         3.559315708 0.468009737  7.6052172 4.238019e-14
## HealthGenFair         5.299570135 0.575060245  9.2156781 7.192101e-20
## HealthGenPoor         7.640141955 1.077494210  7.0906571 1.808115e-12
## PhysActive           -0.837417930 0.294615309 -2.8424115 4.519990e-03
## SleepHrsNight:Age     0.017091879 0.009024443  1.8939539 5.836644e-02
## SleepHrsNight:Gender -0.477031856 0.206903401 -2.3055776 2.122985e-02
```

```r
## After removing these observations, the estimated coefficients are:
summary(rm.m_full)$coef
```

```
##                     Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)     20.35681885 1.591402458 12.7917478 3.758113e-36
## SleepHrsNight   -0.16992002 0.105487052 -1.6108140 1.073684e-01
## Age              0.01603432 0.013256274  1.2095650 2.265800e-01
## Gender           0.38567462 0.286107176  1.3480075 1.777992e-01
## Race1           -0.49166686 0.120767754 -4.0711767 4.848398e-05
## TotChol          0.09239091 0.138702544  0.6661082 5.054140e-01
## BPDiaAve         0.05080608 0.013773193  3.6887658 2.309889e-04
## BPSysAve         0.06676170 0.011857120  5.6305153 2.034003e-08
## AlcoholYear     -0.01071753 0.001509265 -7.1011608 1.678800e-12
## Smoke100        -0.57220143 0.282360003 -2.0264961 4.283829e-02
## DaysPhysHlthBad  0.04676614 0.019585316  2.3878164 1.703545e-02
## PhysActive      -1.18819428 0.286816208 -4.1427027 3.566379e-05
```

```r
#### change percent
abs((rm.m_full$coefficients - m_full$coefficients) / (m_full$coefficients) * 100)
```

```
## Warning in rm.m_full$coefficients - m_full$coefficients: longer object length
```

```
## is not a multiple of shorter object length
```

```
##          (Intercept)        SleepHrsNight                  Age
##             4.648800            68.685049           115.251070
##               Gender                Race1              Poverty
##            89.766365             2.296184            27.035122
##              TotChol             BPDiaAve             BPSysAve
##           243.917613            13.715961           119.683431
##          AlcoholYear             Smoke100           UrineFlow1
##          6715.303845           105.823933          1062.413764
##      DaysMentHlthBad      DaysPhysHlthBad       HealthGenVgood
##         67395.464736          1222.191001            99.168466
##        HealthGenGood        HealthGenFair        HealthGenPoor
##            89.164360           109.277486            98.790717
##           PhysActive    SleepHrsNight:Age SleepHrsNight:Gender
##           106.066993           290.604779            97.753288
```

```
#The estimated regression coefficients doesn't change slightly after removing these observations. 5 of

##################   multicollinearity   ######################
#Pearson correlations
var2 = c(
  "BMI",
  "SleepHrsNight",
  "Age",
  "Gender",
  "Race1",
  "TotChol",
  "BPDiaAve",
  "BPSysAve",
  "AlcoholYear",
  "Smoke100",
  "DaysPhysHlthBad",
  "PhysActive"
)
newData2 = df3[, var2]
library("corrplot")
```

```
## corrplot 0.92 loaded
```

```
par(mfrow = c(1, 2))
cormat2 = cor(as.matrix(newData2[, -c(1)], method = "pearson"))
p.mat2 = cor.mtest(as.matrix(newData2[, -c(1)]))$p
corrplot(
  cormat2,
  method = "color",
  type = "upper",
  number.cex = 1,
  diag = FALSE,
  addCoef.col = "black",
  tl.col = "black",
  tl.srt = 90,
  p.mat = p.mat2,
  sig.level = 0.05,
  insig = "blank",
)
```

```r
#None of the covariates seem strongly correlated.There is no evidence of collinearity from the pair-wis

# collinearity diagnostics (VIF)
car::vif(m_full)
```

```
## there are higher-order terms (interactions) in this model
## consider setting type = 'predictor'; see ?vif
```

```
##                            GVIF Df GVIF^(1/(2*Df))
## SleepHrsNight         13.583457  1        3.685574
## Age                   27.952318  1        5.286995
## Gender                28.340488  1        5.323579
## Race1                  1.092306  1        1.045135
## Poverty                1.307058  1        1.143266
## TotChol                1.127465  1        1.061822
## BPDiaAve               1.453678  1        1.205686
## BPSysAve               1.565996  1        1.251398
## AlcoholYear            1.122185  1        1.059332
## Smoke100               1.130221  1        1.063119
## UrineFlow1             1.045986  1        1.022734
## DaysMentHlthBad        1.146283  1        1.070646
## DaysPhysHlthBad        1.251072  1        1.118513
## HealthGen              1.447335  4        1.047300
## PhysActive             1.168763  1        1.081093
## SleepHrsNight:Age     37.541993  1        6.127152
## SleepHrsNight:Gender  29.940850  1        5.471823
```

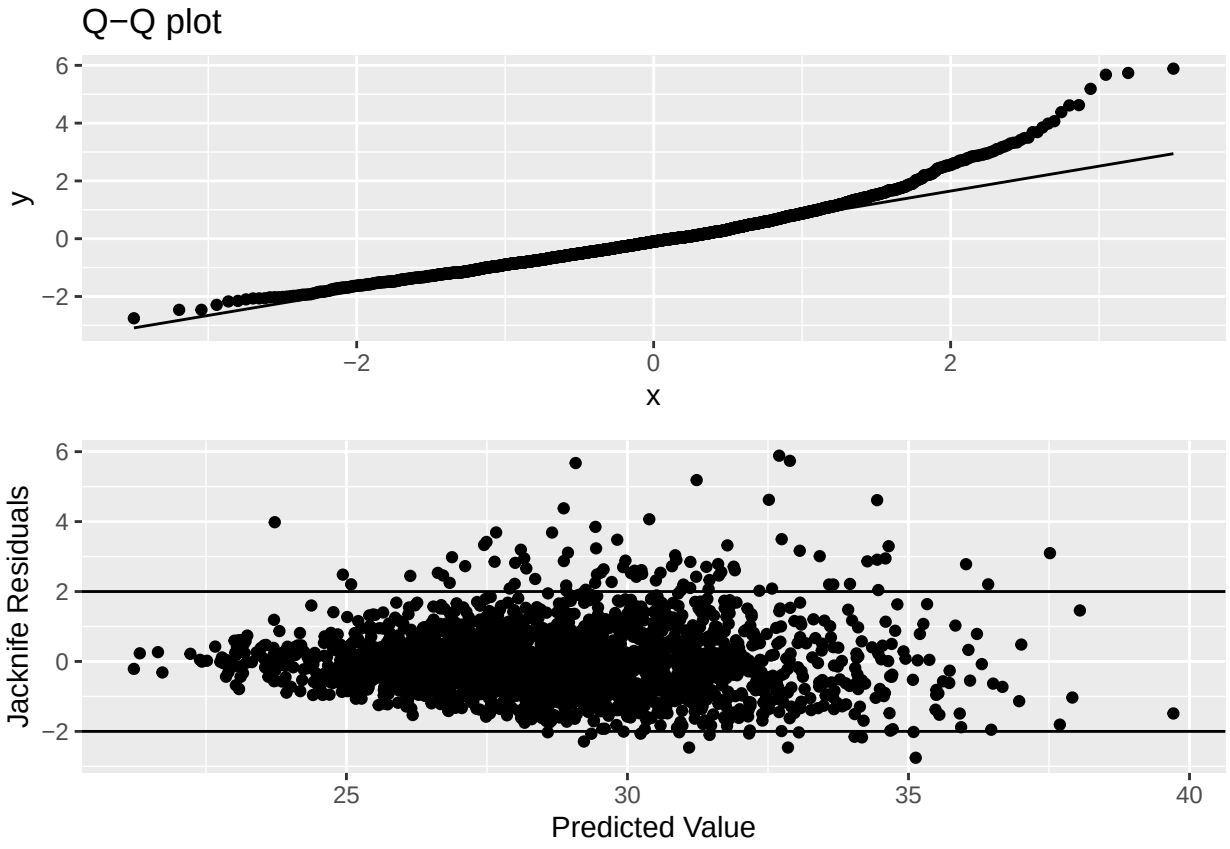
```r
#From the VIF values in the output above, once again we do not observe any potential collinearity issue

################ using log-transformed BMI  ##################
# log BMI
df3$logBMI = log(df3$BMI + 1)
m_full.log = lm(
  logBMI ~ SleepHrsNight + Age + Gender + Race1 + TotChol + BPDiaAve + BPSysAve + AlcoholYear + Smoke10
    DaysPhysHlthBad + PhysActive,
  df3
)
p21.log = ols_plot_resid_lev(m_full.log)
p22.log = ols_plot_cooksd_bar(m_full.log)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

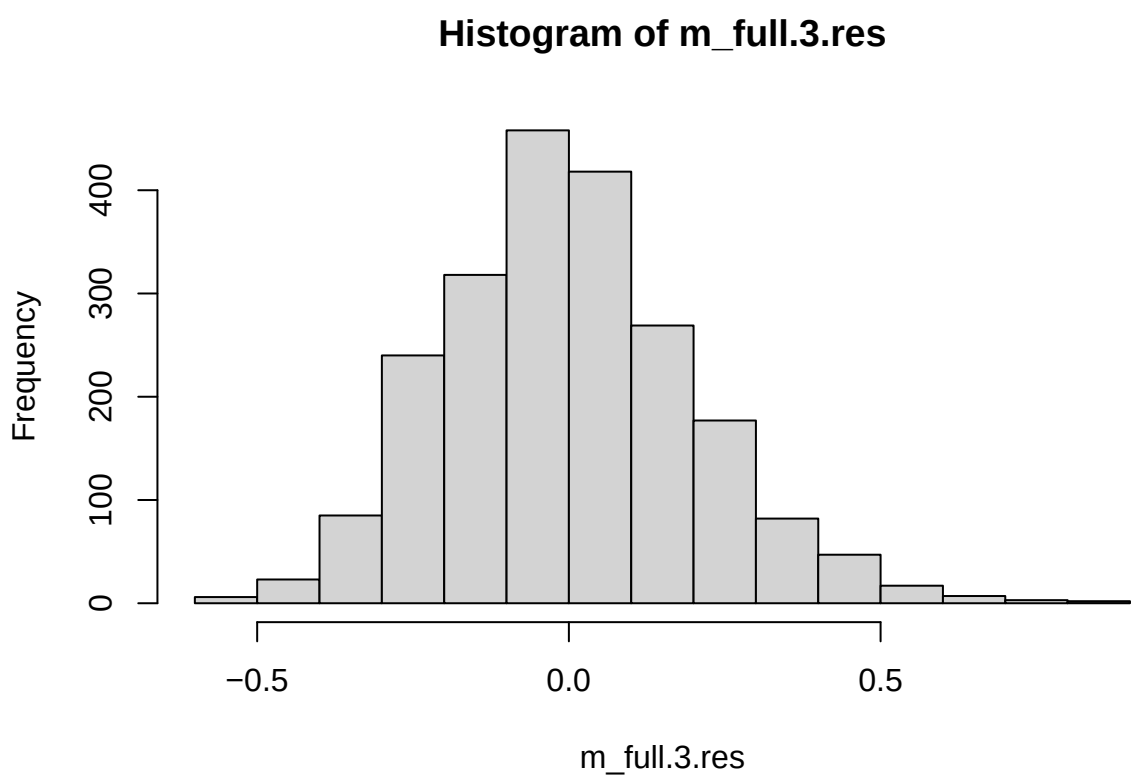
```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
p23.log = ggplot(m_full.log, aes(sample = rstudent(m_full.log))) + geom_qq() + stat_qq_line() +
  labs(title = "Q-Q plot")
p24.log = ggplot() + geom_point(aes(y = rstudent(m_full.log), x = m_full.log$fitted.values)) + labs(x =
  geom_hline(yintercept = c(-2, 2))
grid.arrange(p23.log, p24.log, nrow = 2)
```

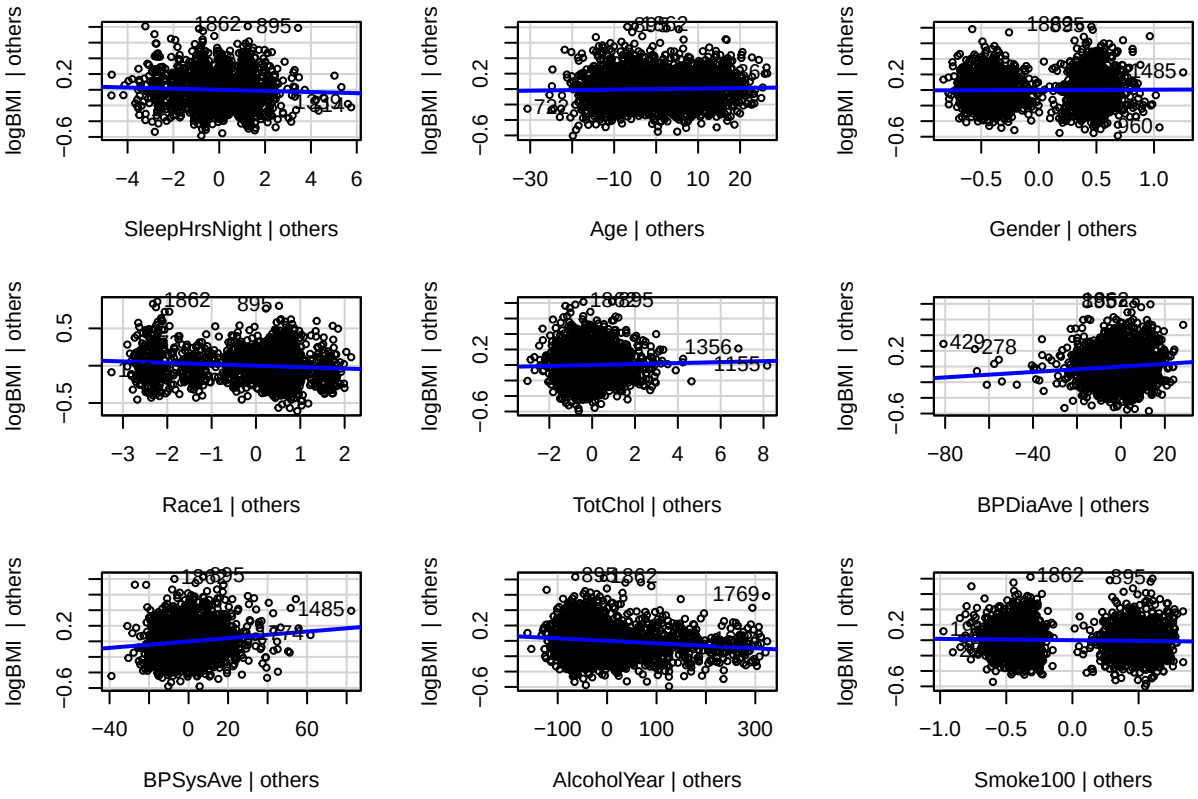
```
p23 = ggplot(m_full, aes(sample = rstudent(m_full))) + geom_qq() + stat_qq_line() +
  labs(title = "Q-Q plot")
p24 = ggplot() + geom_point(aes(y = rstudent(m_full), x = m_full$fitted.values)) + labs(x = "Predicted V
  geom_hline(yintercept = c(-2, 2))
grid.arrange(p23, p24, nrow = 2)


m_full.3.yhat = m_full.log$fitted.values
m_full.3.res = m_full.log$residuals
m_full.3.h = hatvalues(m_full.log)
m_full.3.r = rstandard(m_full.log)
m_full.3.rr = rstudent(m_full.log)

par(mfrow = c(1, 1))
```

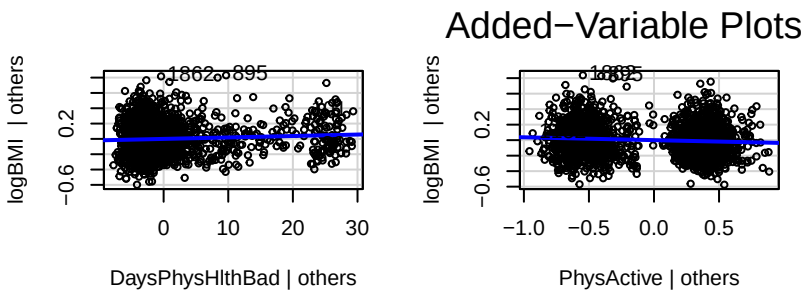



```
hist(m_full.3.res, breaks = 15)
```




```
car::avPlots(m_full.log)
```


logBMI | others
SleepHrsNight | others
Age | others
Gender | others
Race1 | others
TotChol | others
BPDiaAve | others
BPSysAve | others
AlcoholYear | others
Smoke100 | others

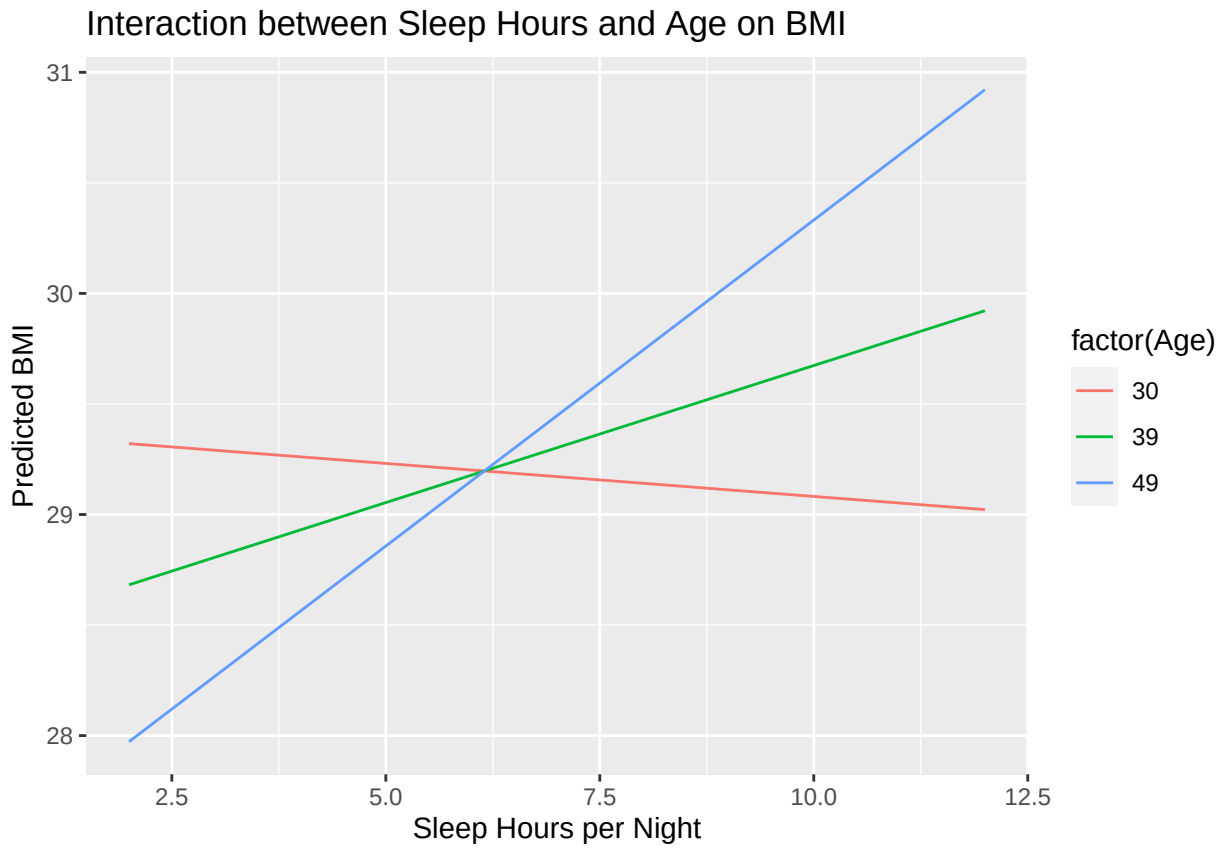



```
#After looking at residuals from models using the log-transformed (natural log scale) BMI adjusted for

#collinearity diagnostics

car::vif(m_full.log)
```

```
##   SleepHrsNight            Age         Gender          Race1       TotChol
##        1.035419       1.223319       1.106167       1.045711       1.122357
##       BPDiaAve       BPSysAve    AlcoholYear       Smoke100 DaysPhysHlthBad
##        1.447702       1.542999       1.091195       1.078534       1.057582
##     PhysActive
##        1.093222
```

```
#The VIF from both the models are the same. None of the VIF values are greater than 10. So there are no

getMode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}

new_data <- expand.grid(SleepHrsNight = seq(min(df3$SleepHrsNight), max(df3$SleepHrsNight), length.out =
                        Age = quantile(df3$Age, probs = c(0.25, 0.5, 0.75)),
                        Gender = median(df3$Gender, na.rm = TRUE),
                        Race1 = median(df3$Race1, na.rm = TRUE),
                        Poverty = median(df3$Poverty, na.rm = TRUE),
                        TotChol = median(df3$TotChol, na.rm = TRUE),
                        BPDiaAve = median(df3$BPDiaAve, na.rm = TRUE),
```

```
                        BPSysAve = median(df3$BPSysAve, na.rm = TRUE),
                        AlcoholYear = median(df3$AlcoholYear, na.rm = TRUE),
                        Smoke100 = getMode(df3$Smoke100),
                        UrineFlow1 = median(df3$UrineFlow1, na.rm = TRUE),
                        DaysMentHlthBad = median(df3$DaysMentHlthBad, na.rm = TRUE),
                        DaysPhysHlthBad = median(df3$DaysPhysHlthBad, na.rm = TRUE),
                        HealthGen = getMode(df3$HealthGen),
                        PhysActive = getMode(df3$PhysActive)
)

# predict
new_data$predicted_BMI <- predict(m_full, newdata = new_data)
# interaction
library(ggplot2)
ggplot(new_data, aes(x = SleepHrsNight, y = predicted_BMI, group = factor(Age))) +
  geom_line(aes(color = factor(Age))) +
  labs(title = "Interaction between Sleep Hours and Age on BMI",
       x = "Sleep Hours per Night",
       y = "Predicted BMI")
```




```
# cross validation
library(caret)
```

```
## Loading required package: lattice
```

```
splitIndex <-
  createDataPartition(df3$SleepHrsNight, p = 0.7, list = FALSE)
```

```r
trainData <- df3[splitIndex, ]
testData <- df3[-splitIndex, ]
predictions <- predict(m_full, newdata = testData)
mse <- mean((testData$SleepHrsNight - predictions) ^ 2)
control <-
  trainControl(method = "cv", number = 10)  # 10-fold cross-validation
cv_model <-
  train(
    SleepHrsNight ~ .,
    data = df3,
    method = "lm",
    trControl = control
  )
cv_model
```

```
## Linear Regression
##
## 2152 samples
##   21 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1937, 1938, 1936, 1937, 1937, 1937, ...
## Resampling results:
##
##   RMSE      Rsquared    MAE
##   1.280493  0.05032481  0.9944286
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```r
(cv_results <- cv_model$results)
```

```
##   intercept     RMSE   Rsquared       MAE     RMSESD RsquaredSD      MAESD
## 1      TRUE 1.280493 0.05032481 0.9944286 0.04858825 0.02601371 0.02959426
```