

STAT I512, Honglang Wang (hlwang)

Final Exam

Kyle McCrocklin

Problem 1:

```
library(MASS)
library(ggplot2)
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.4.2
```

```
## corrplot 0.95 loaded
```

```
library(splines)
set.seed(42)

load("G:/Other computers/My Laptop/Google Drive/STAT512 Projects/Final Exam/meps512.RData")

# Predictors:
# AGE42X, ADBMI42, MNHLTH42, FAMINC18, RACETHX, SEX
# age, BMI, perceived mental health status, family income, race and gender

# Response:
# TOTEXP18 = total adult medical expenditures.

# Convert to factors
data$MNHLTH42 = as.factor(data$MNHLTH42)
data$RACETHX = as.factor(data$RACETHX)
data$SEX = as.factor(data$SEX)

colSums(is.na(data))
```

```
## TOTEXP18  AGE42X  ADBMI42  MNHLTH42  FAMINC18  RACETHX  SEX
##          0         0         0         0         0         0         0
```

```
m1 = lm(TOTEXP18 ~ AGE42X, data=data)
cat('\nMODEL 1:')
```

```
##
## MODEL 1:
```

```
summary(m1)
```

```
##
## Call:
## lm(formula = TOTEXP18 ~ AGE42X, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14904  -6955  -3807   -538  801046
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2385.134    407.137  -5.858 4.76e-09 ***
## AGE42X       203.406      7.691   26.448 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18930 on 18250 degrees of freedom
## Multiple R-squared:  0.03691,    Adjusted R-squared:  0.03686
## F-statistic: 699.5 on 1 and 18250 DF,  p-value: < 2.2e-16
```

```
m2 = lm(TOTEXP18 ~ AGE42X + SEX, data=data)
cat('\nMODEL 2:')
```

```
##
## MODEL 2:
```

```
summary(m2)
```

```
##
## Call:
## lm(formula = TOTEXP18 ~ AGE42X + SEX, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15414  -6908  -3791   -465  800523
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1848.028    428.466  -4.313 1.62e-05 ***
## AGE42X       203.081      7.688   26.415 < 2e-16 ***
## SEXMALE     -1125.788    280.895  -4.008 6.15e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18920 on 18249 degrees of freedom
## Multiple R-squared:  0.03776,    Adjusted R-squared:  0.03765
## F-statistic: 358.1 on 2 and 18249 DF,  p-value: < 2.2e-16
```

```
m3 = lm(TOTEXP18 ~ AGE42X * SEX, data=data)
cat('\nMODEL 3:')
```

```
##
## MODEL 3:
```

```
summary(m3)
```

```
##
## Call:
## lm(formula = TOTEXP18 ~ AGE42X * SEX, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15308   -6963   -3830   -181  800381
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -643.38     554.06  -1.161  0.24557
## AGE42X           178.93      10.43  17.162 < 2e-16 ***
## SEXMALE       -3752.99     816.28  -4.598  4.3e-06 ***
## AGE42X:SEXMALE    52.89      15.43   3.428  0.00061 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18920 on 18248 degrees of freedom
## Multiple R-squared:  0.03838,    Adjusted R-squared:  0.03822
## F-statistic: 242.8 on 3 and 18248 DF,  p-value: < 2.2e-16
```

a)

In model 1, β_1 represents the change in medical expenditure for an increase in age of 1 year. This captures the relationship between age and medical expenditure without accounting for the potential influence of any other variables.

In model 2, β_1 represents the change in medical expenditure for an increase in age of 1 year while holding gender constant. This isolates the effects of each predictor from the other which gives a more precise understanding of each predictors contribution to the response.

b)

The change in expected response if age increases by 1 while sex held fixed for Model 2 is an increase of \$203.08

c)

The change in expected response if age increases by 1 while sex held fixed for Model 3 is an increase of... for females ($X_2=0$): $\beta_1 = \$178.93$ for males ($X_2=1$): $\beta_1 + \beta_3 = 178.93 + 52.89 = \231.82

Problem 2:

###Print summary statistics

```
# summary statistics
cat('\nContinuous variables:\n')
```

```
##
## Continuous variables:
```

```
summary(data[, c("TOTEXP18", "AGE42X", "ADBMI42", "FAMINC18")])
```

```
##      TOTEXP18      AGE42X      ADBMI42      FAMINC18
## Min.   :      0  Min.   :18.0  Min.   : 0.10  Min.   : -309948
## 1st Qu.:    400  1st Qu.:34.0  1st Qu.:23.70  1st Qu.:  27130
## Median :   2024  Median :50.0  Median :27.30  Median :   56838
## Mean   :   7725  Mean   :49.7  Mean   :28.15  Mean   :   75915
## 3rd Qu.:   7087  3rd Qu.:64.0  3rd Qu.:31.90  3rd Qu.: 104718
## Max.   :  807611  Max.   :85.0  Max.   :71.10  Max.   :  583219
```

```
cat('\nMental health:')
```

```
##
## Mental health:
```

```
table(data$MNHLTH42)
```

```
##
## EXCELLENT      FAIR      GOOD      INVALID      POOR VERY GOOD
##      5868      1405      4904          6      301      5768
```

```
cat('\nRace:')
```

```
##
## Race:
```

```
table(data$RACETHX)
```

```
##
##      ASIAN ONLY      BLACK ONLY      HISPANIC OTHER OR MULTIPLE
##      900      2633      3642      569
##      WHITE ONLY
##      10508
```

```
cat('\nGender:')

```

```
##
## Gender:

```

```
table(data$SEX)

```

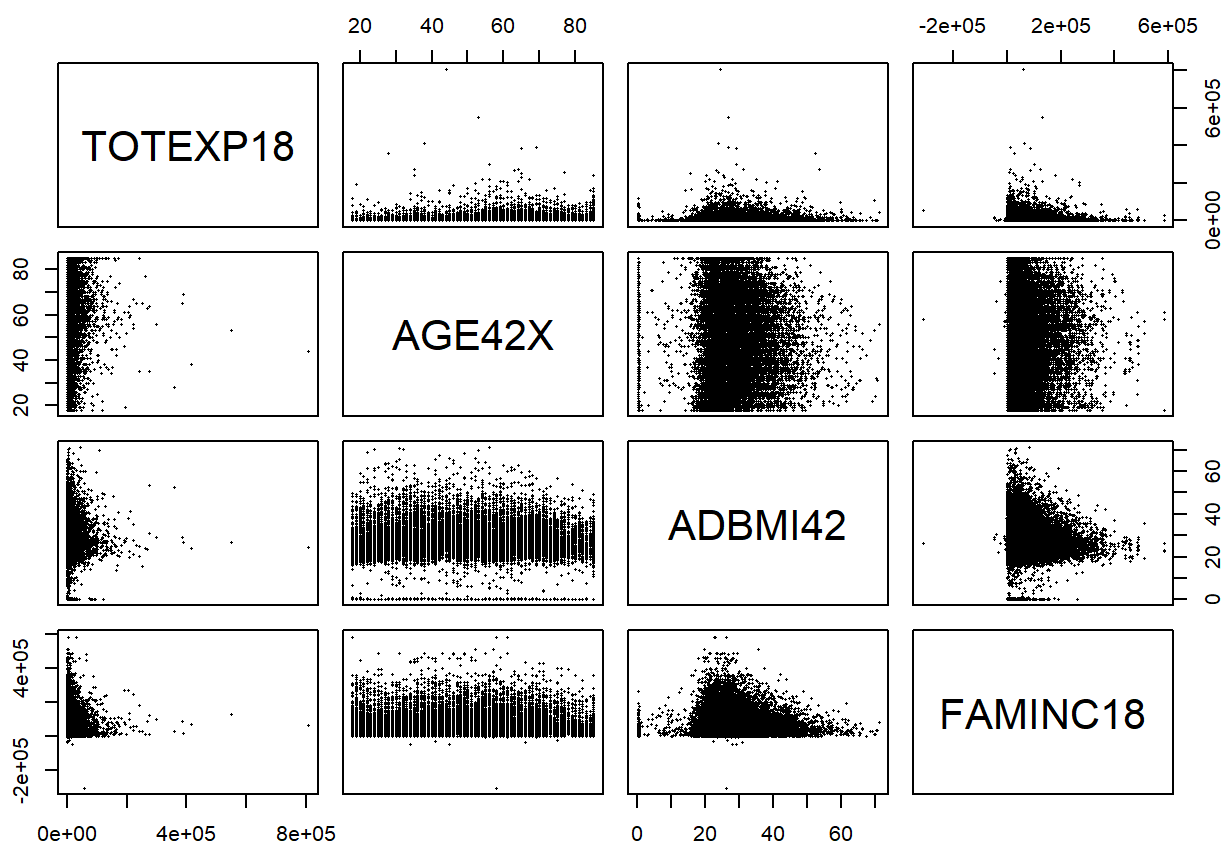
```
##
## FEMALE    MALE
##    9806    8446

```

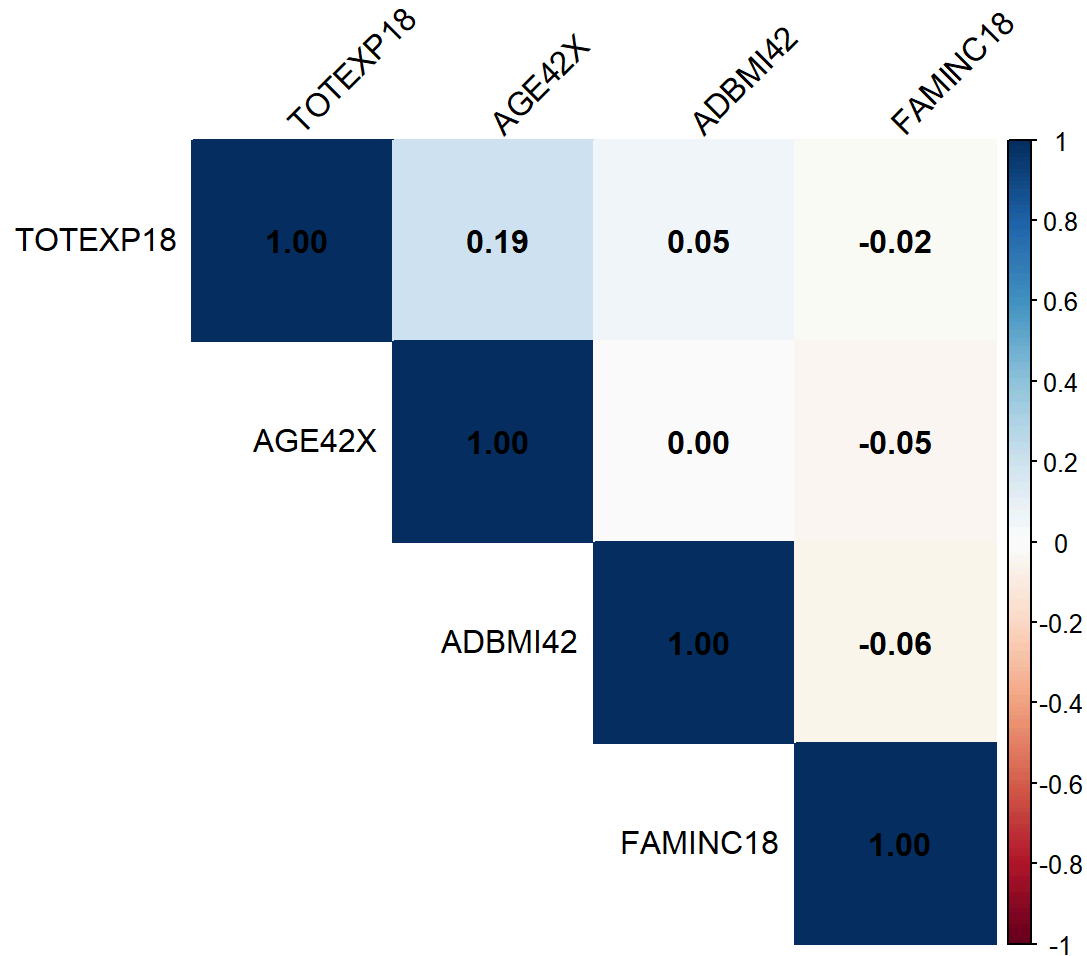
Investigate distribution of variables using plots

```
# pairwise scatterplot
pairs(data[, c("TOTEXP18", "AGE42X", "ADBMI42", "FAMINC18")],
      pch = 16,
      cex = 0.3)

```

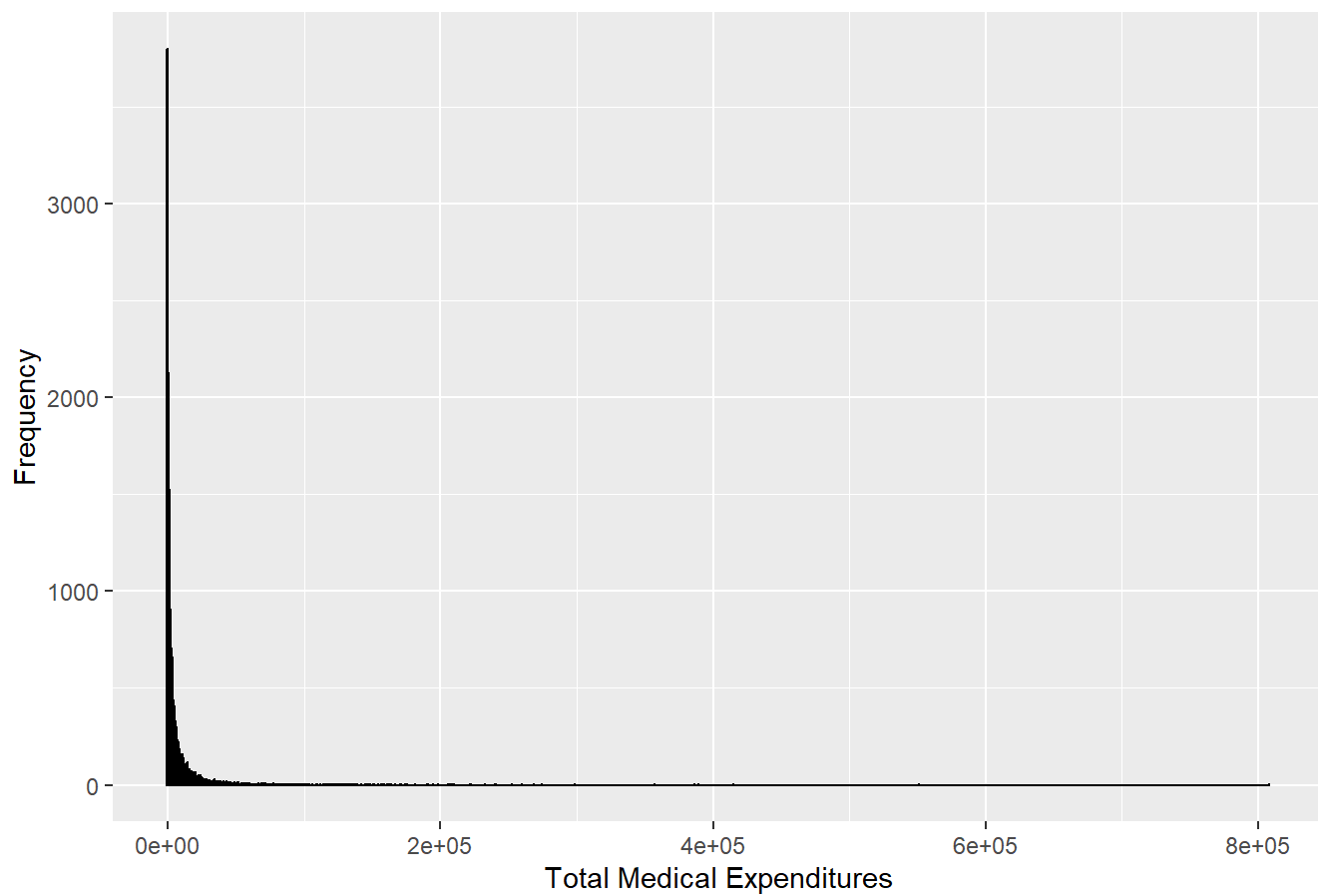


```
# correlation matrix
cor_matrix = cor(data[, c("TOTEXP18", "AGE42X", "ADBMI42", "FAMINC18")], use = "complete.obs")
corrplot(cor_matrix, method = "color", type = "upper", tl.col = "black", tl.srt = 45, addCoef.col = "black")
```



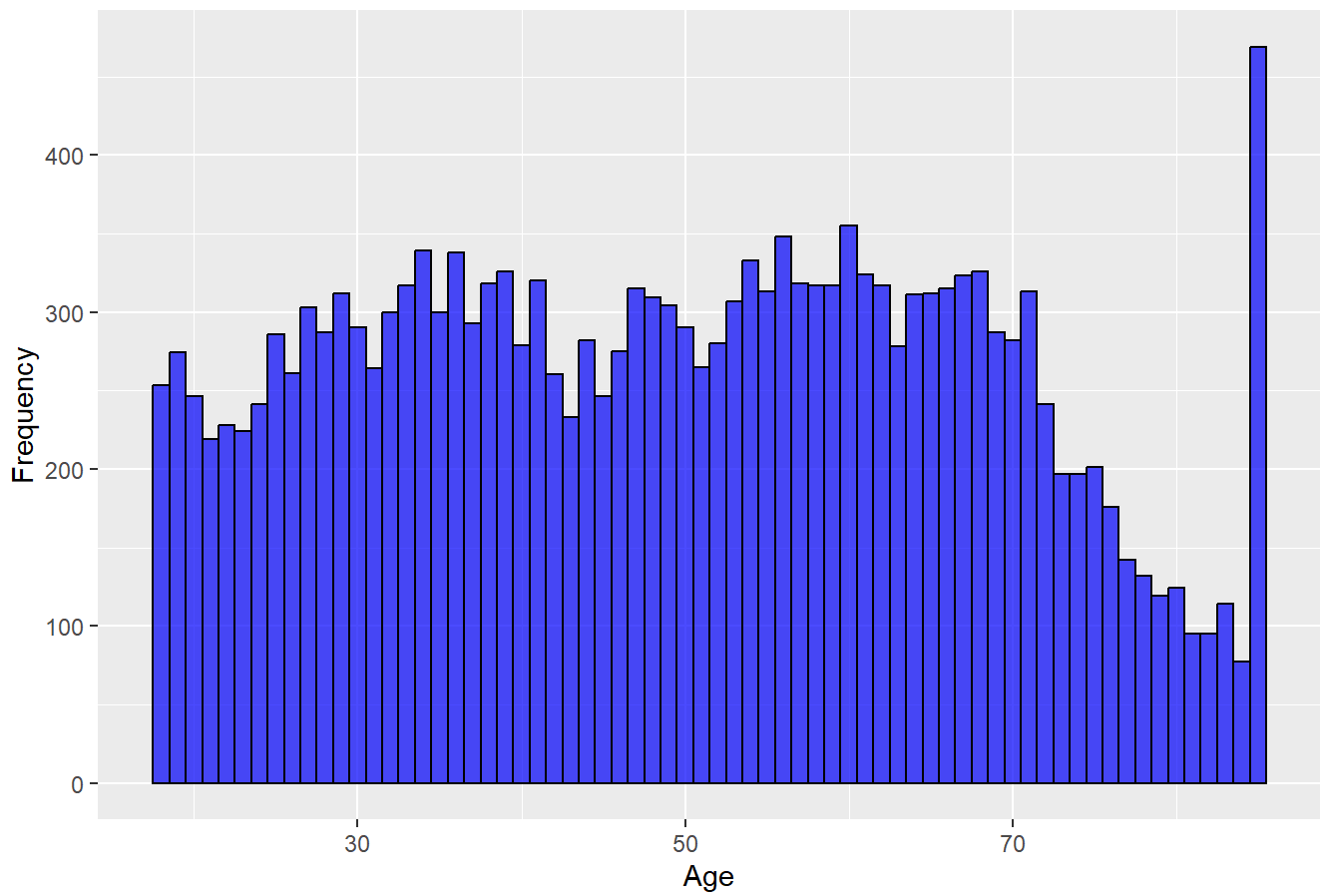
```
# histograms for continuous variables
ggplot(data, aes(x = TOTEXP18)) +
  geom_histogram(binwidth = 500, fill = "blue", color = "black", alpha = 0.7) +
  ggtitle("Histogram of Total Medical Expenditures (TOTEXP18)") +
  xlab("Total Medical Expenditures") +
  ylab("Frequency")
```

Histogram of Total Medical Expenditures (TOTEXP18)



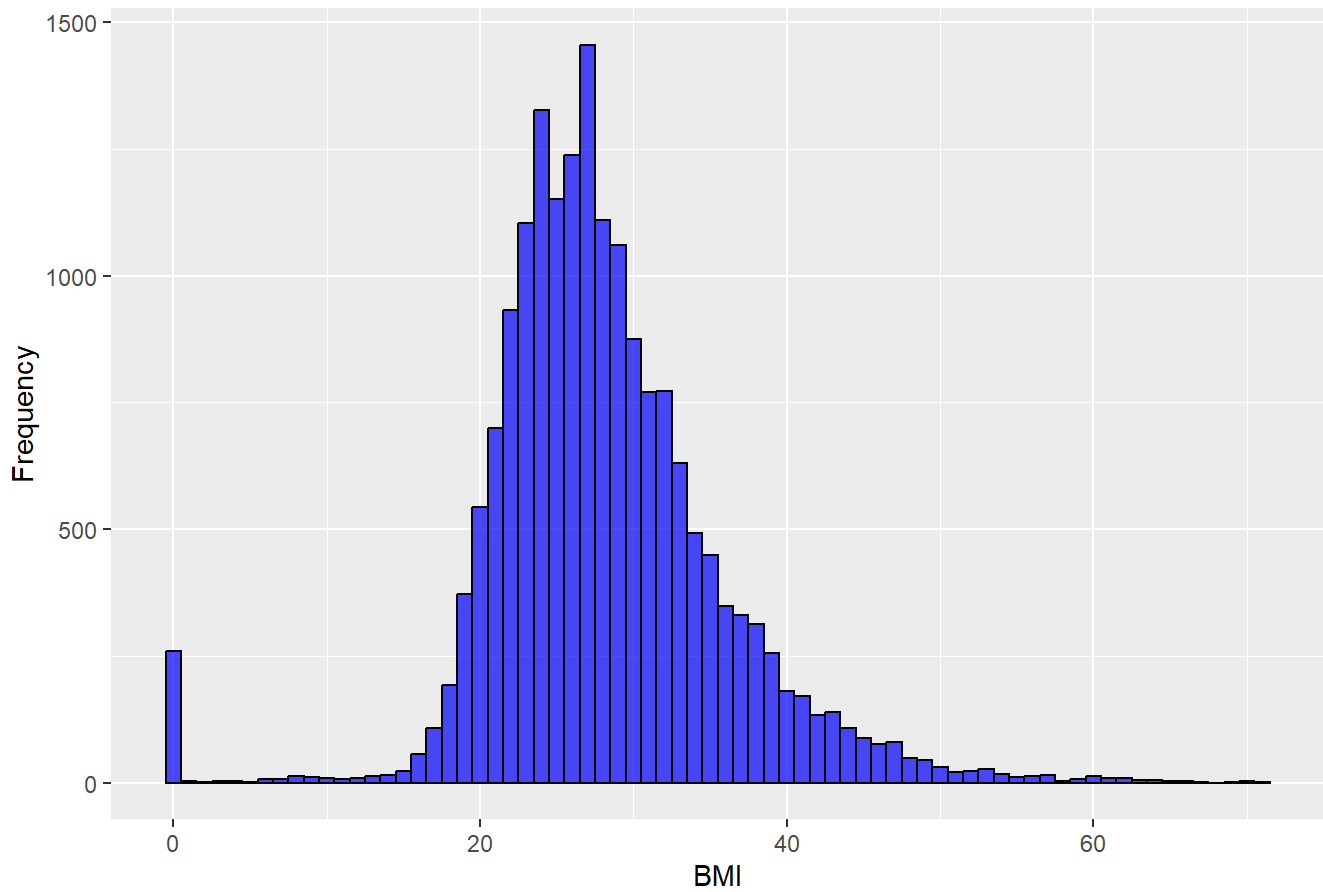
```
ggplot(data, aes(x = AGE42X)) +  
  geom_histogram(binwidth = 1, fill = "blue", color = "black", alpha = 0.7) +  
  ggtitle("Histogram of Age (AGE42X)") +  
  xlab("Age") +  
  ylab("Frequency")
```

Histogram of Age (AGE42X)



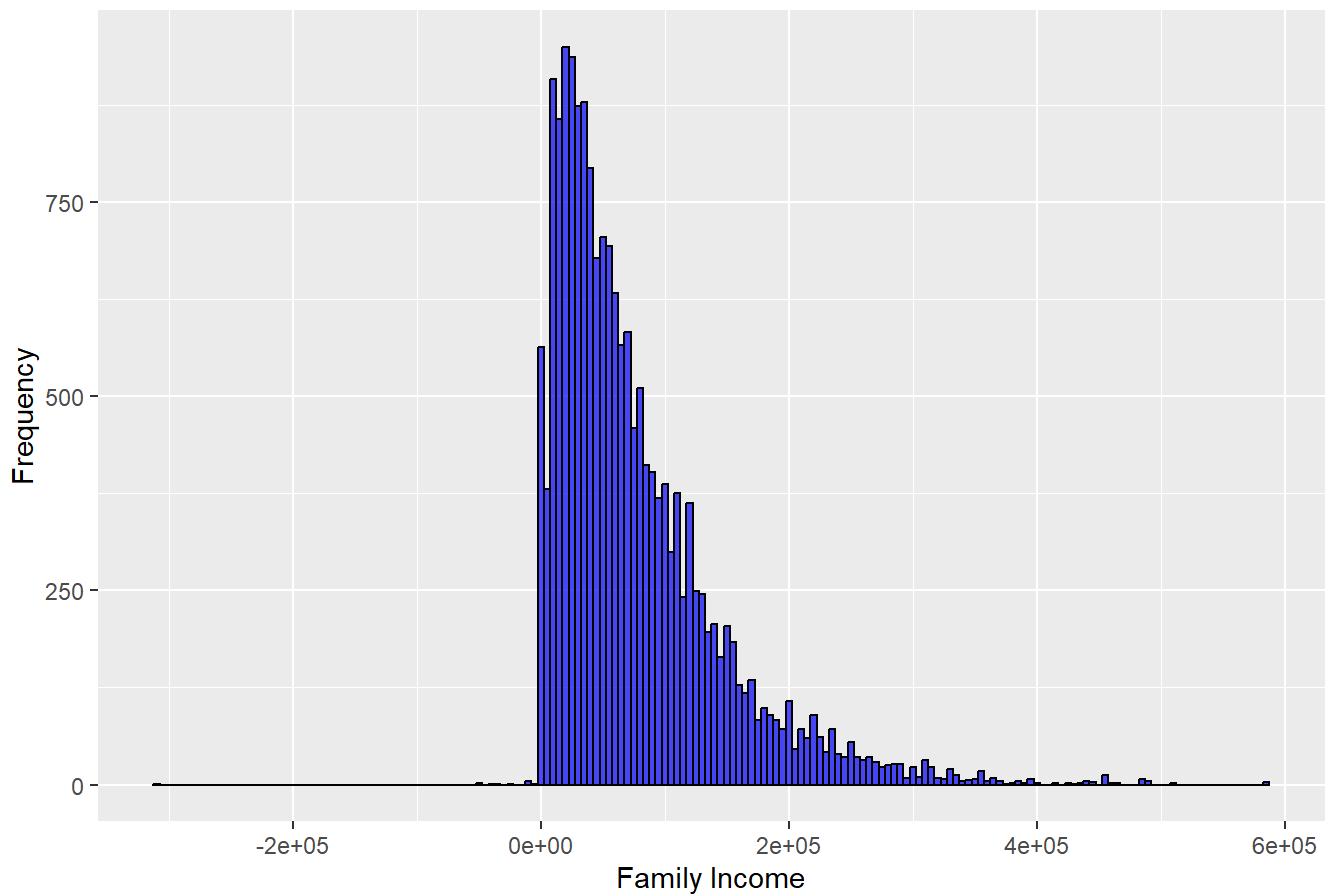
```
ggplot(data, aes(x = ADBMI42)) +  
  geom_histogram(binwidth = 1, fill = "blue", color = "black", alpha = 0.7) +  
  ggtitle("Histogram of BMI (ADBMI42)") +  
  xlab("BMI") +  
  ylab("Frequency")
```


Histogram of BMI (ADBMI42)



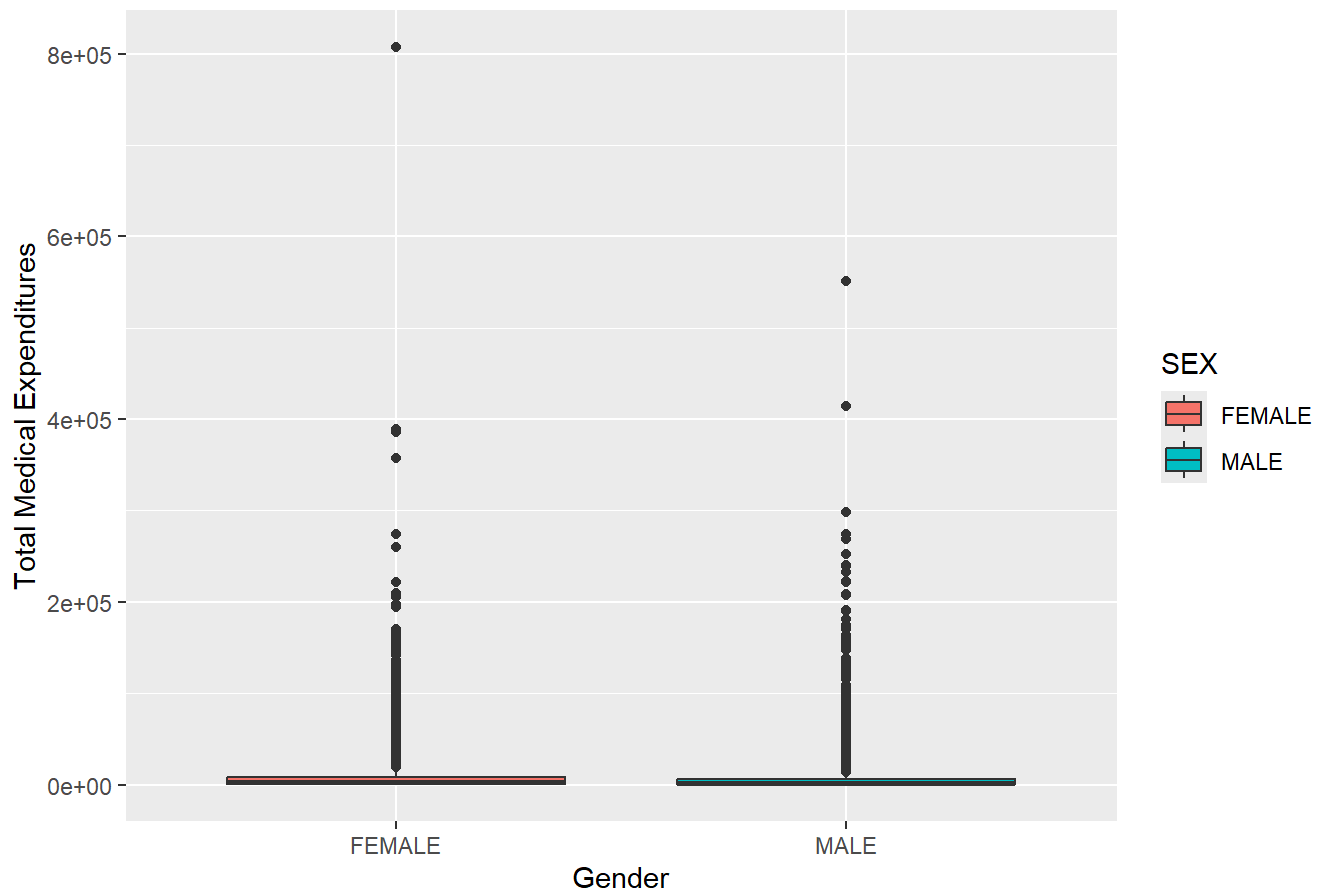
```
ggplot(data, aes(x = FAMINC18)) +  
  geom_histogram(binwidth = 5000, fill = "blue", color = "black", alpha = 0.7) +  
  ggtitle("Histogram of Family Income (FAMINC18)") +  
  xlab("Family Income") +  
  ylab("Frequency")
```

Histogram of Family Income (FAMINC18)



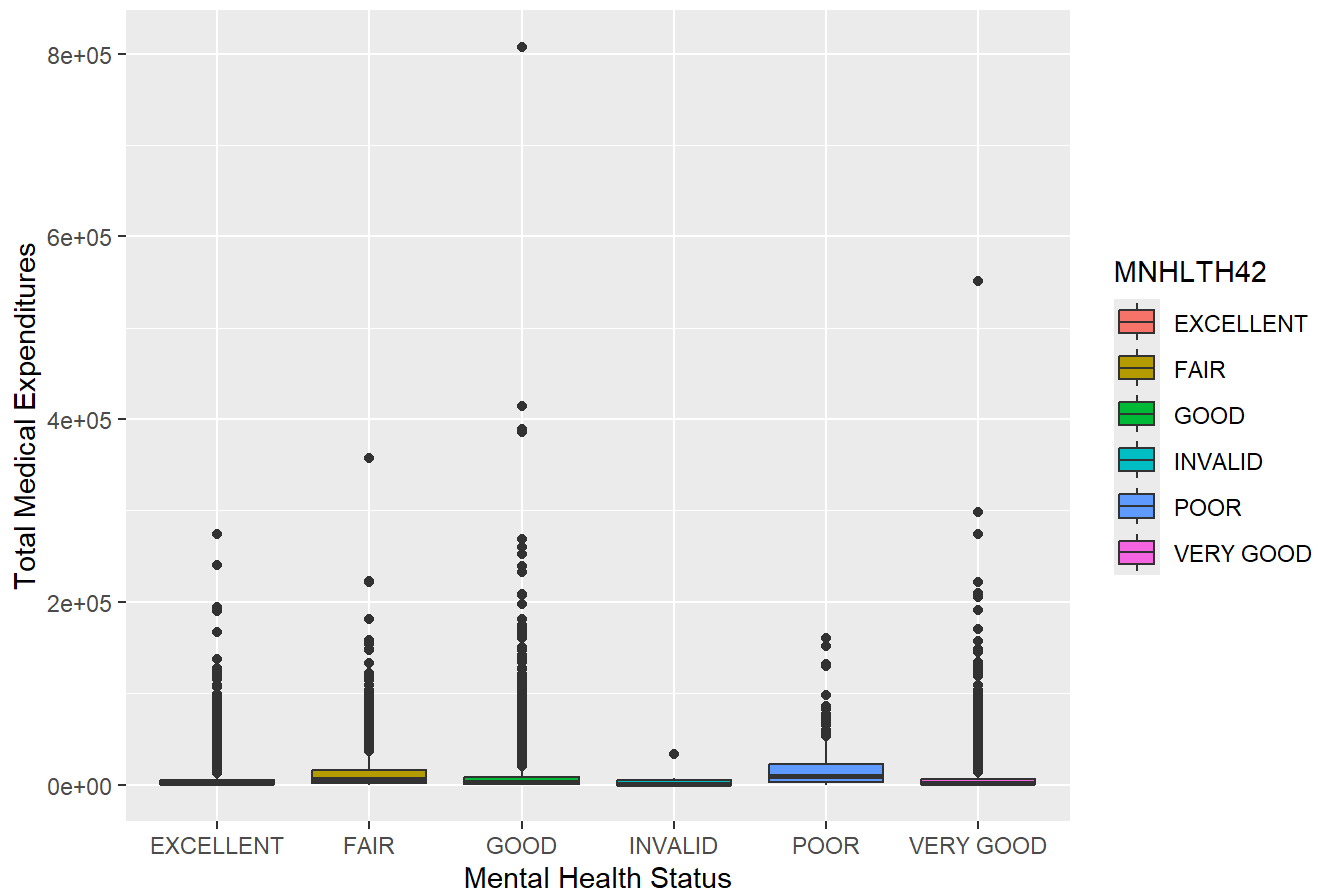
```
# boxplot for `TOTEXP18` by gender
ggplot(data, aes(x = SEX, y = TOTEXP18, fill = SEX)) +
  geom_boxplot() +
  ggtitle("Boxplot of Total Medical Expenditures by Gender") +
  xlab("Gender") +
  ylab("Total Medical Expenditures")
```

Boxplot of Total Medical Expenditures by Gender

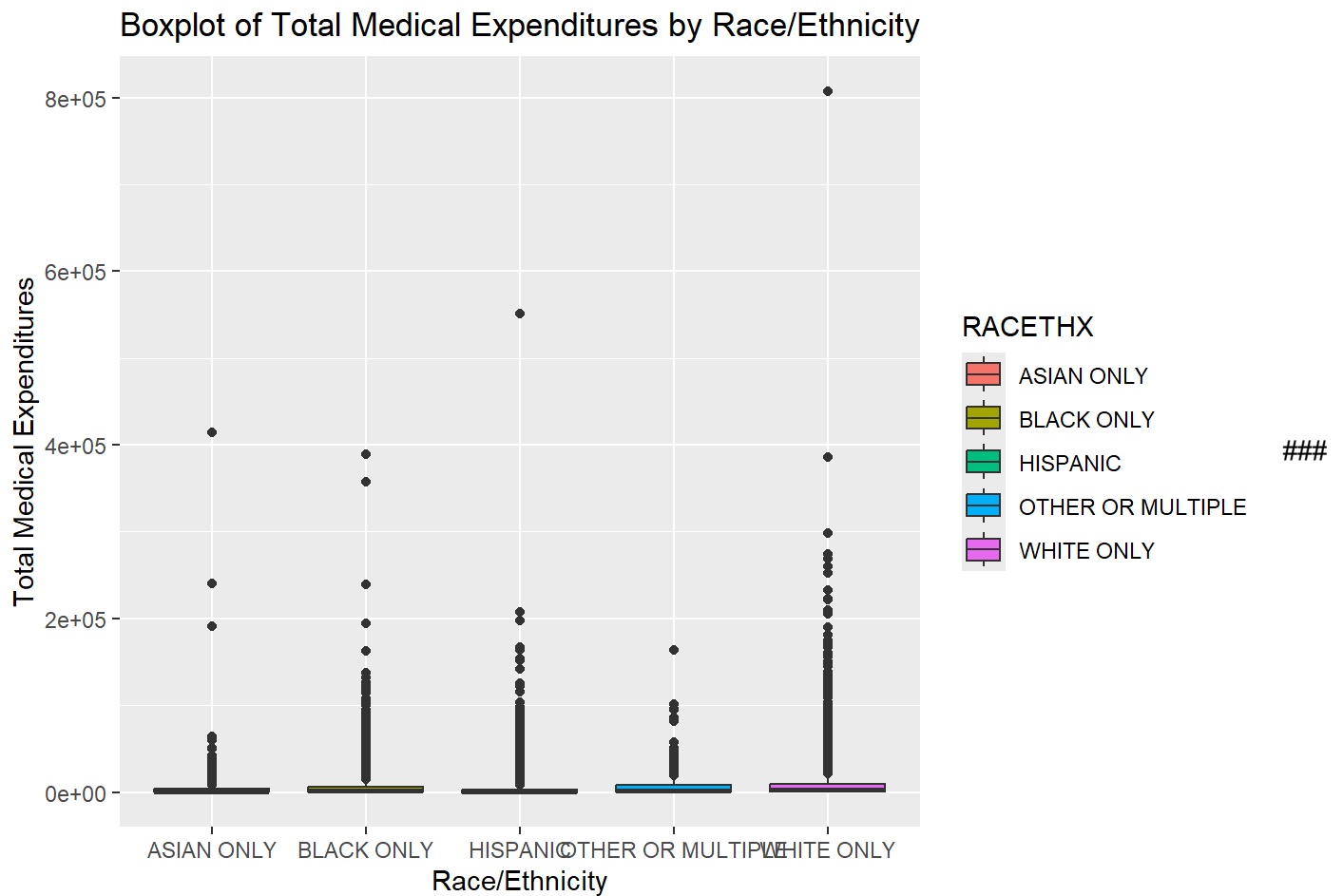


```
# boxplot for `TOTEXP18` by perceived mental health status
ggplot(data, aes(x = MNHLTH42, y = TOTEXP18, fill = MNHLTH42)) +
  geom_boxplot() +
  ggtitle("Boxplot of Total Medical Expenditures by Mental Health Status") +
  xlab("Mental Health Status") +
  ylab("Total Medical Expenditures")
```

Boxplot of Total Medical Expenditures by Mental Health Status



```
# boxplot for `TOTEXP18` by race
ggplot(data, aes(x = RACETHX, y = TOTEXP18, fill = RACETHX)) +
  geom_boxplot() +
  ggtitle("Boxplot of Total Medical Expenditures by Race/Ethnicity") +
  xlab("Race/Ethnicity") +
  ylab("Total Medical Expenditures")
```



BMI appears to have some invalid data. The histogram bar including BMIs less than 1 does not make sense.

```
data_raw = data
data = data[data$ADBMI42 >= 1, ]
```

Negative family income also does not make sense.

```
data = data[data$FAMINC18 >= 0, ]
```

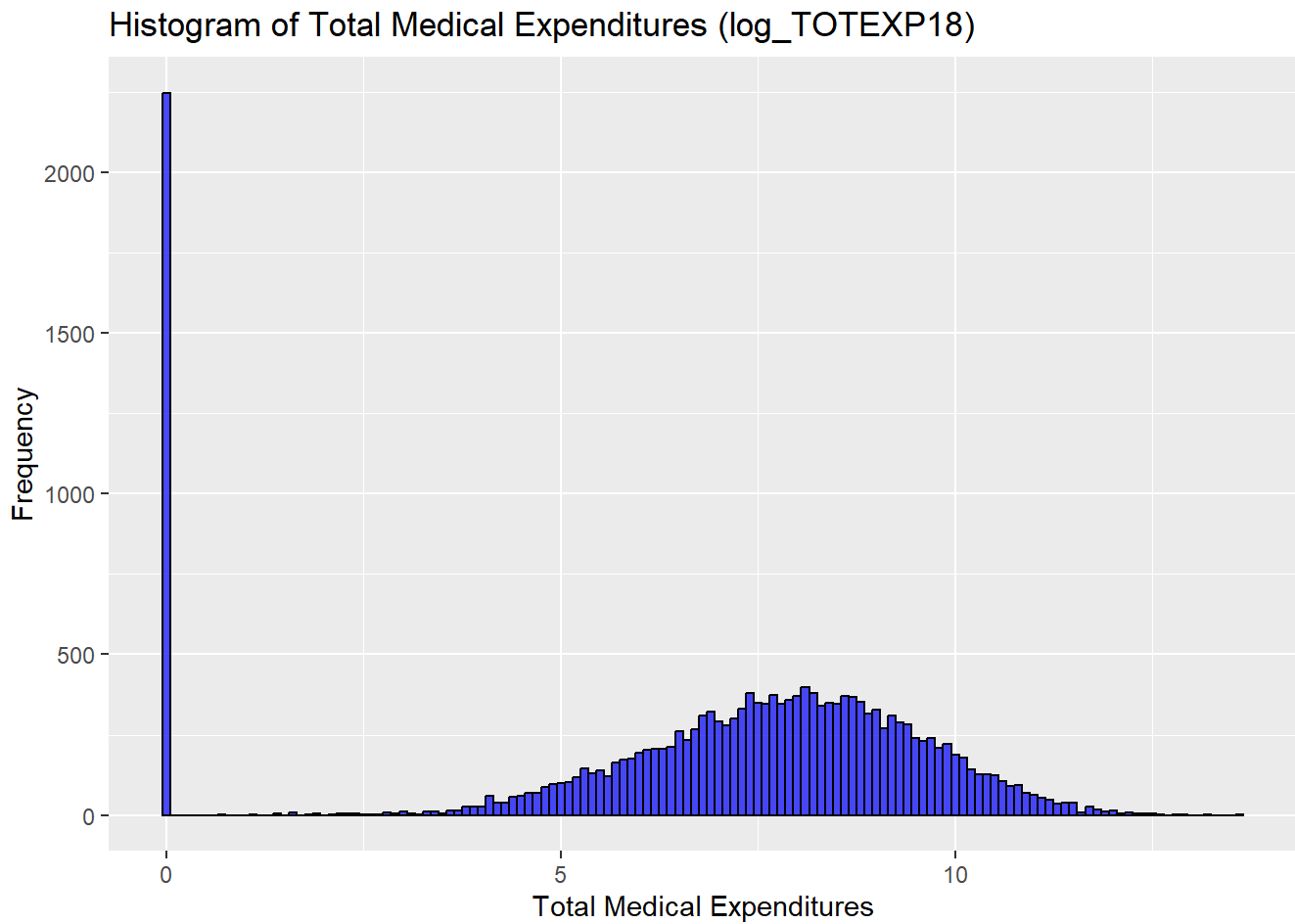
Also going to remove the 6 'invalid' mental health assessments.

```
data = data[data$MNHLTH42 != 'INVALID', ]
```

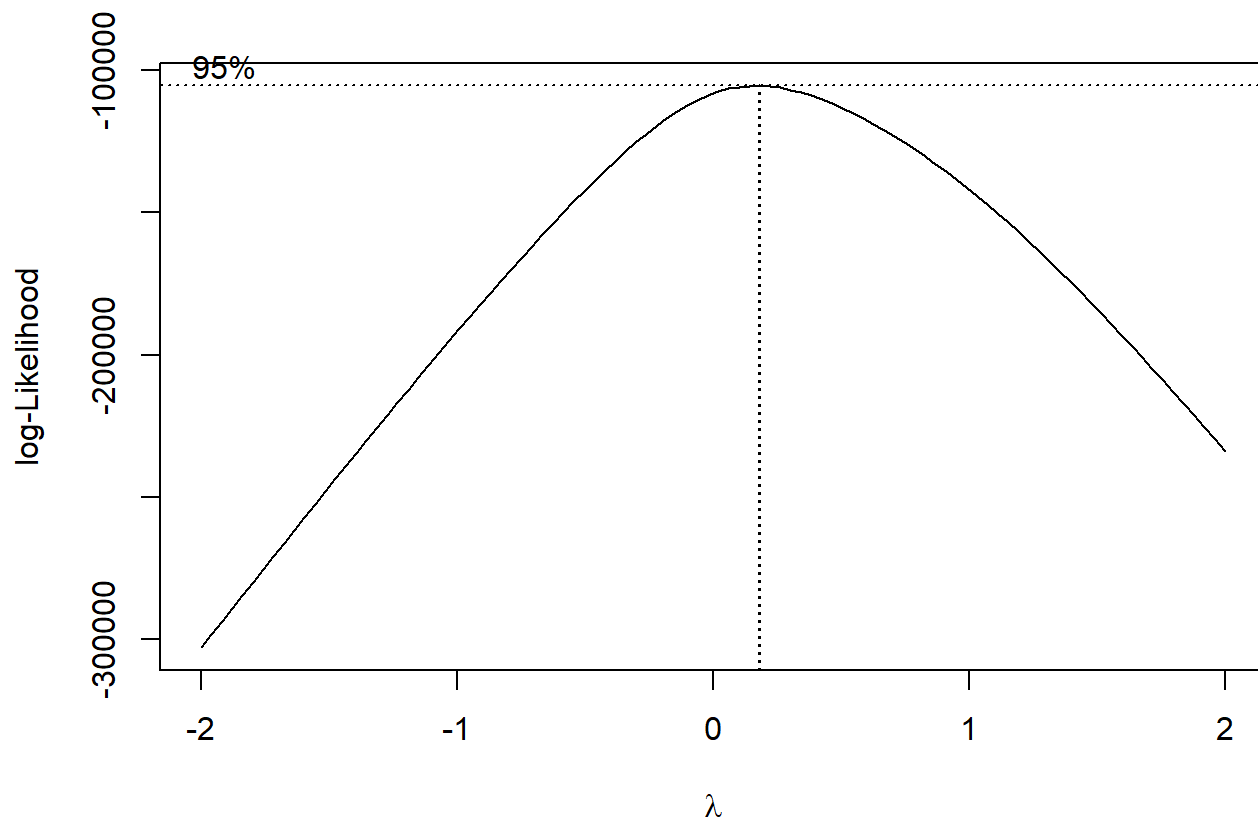
The response is heavily skewed. I want to see what it would look

like with a log or box-cox transformation.

```
# Log
data$log_TOTEXP18 = log(data$TOTEXP18+1)
ggplot(data, aes(x = log_TOTEXP18)) +
  geom_histogram(binwidth = .1, fill = "blue", color = "black", alpha = 0.7) +
  ggtitle("Histogram of Total Medical Expenditures (log_TOTEXP18)") +
  xlab("Total Medical Expenditures") +
  ylab("Frequency")
```



```
# box-cox
boxcox_result <- boxcox(TOTEXP18+1 ~ 1, data = data, lambda = seq(-2, 2, by = 0.1))
```

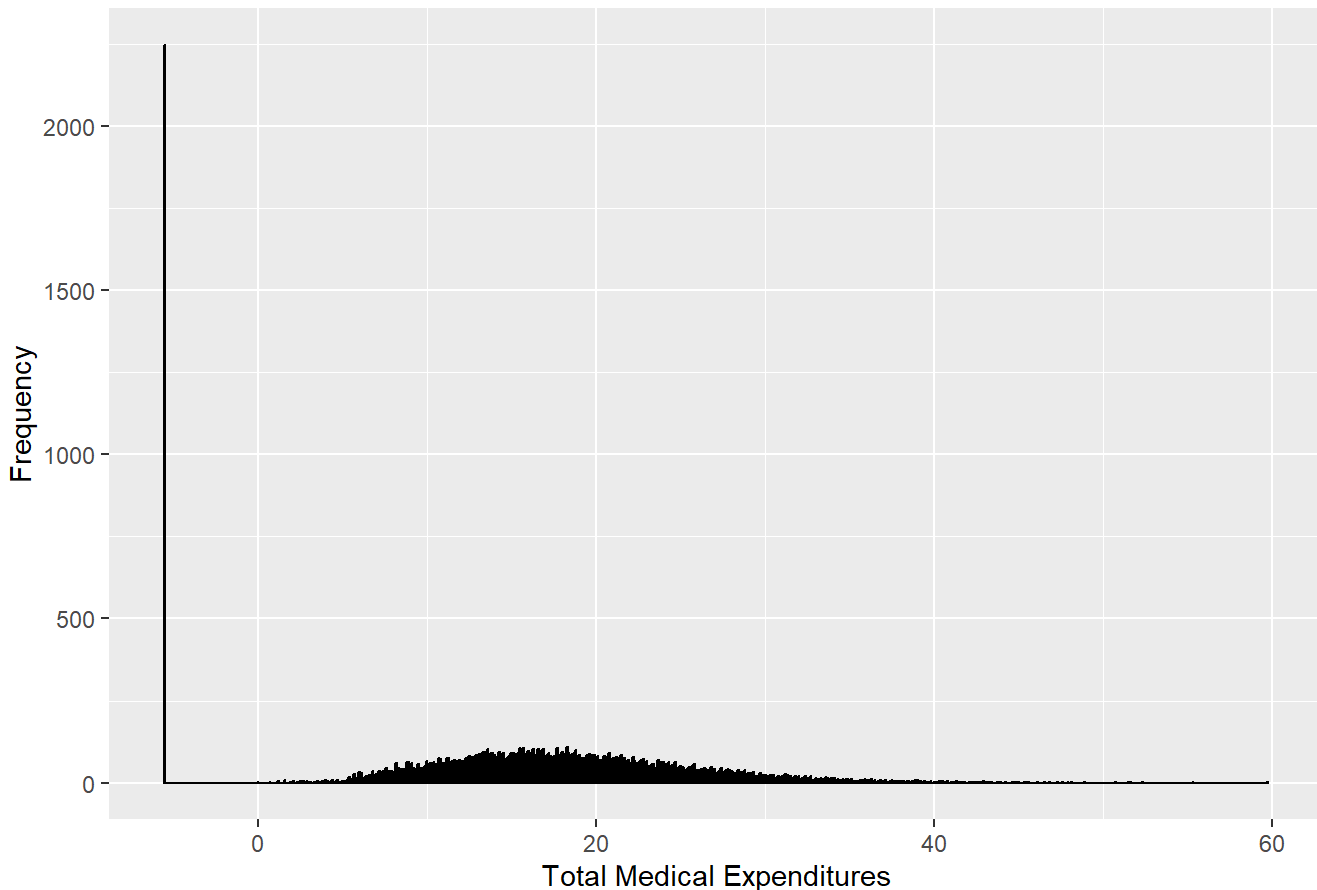


```
optimal_lambda <- boxcox_result$x[which.max(boxcox_result$y)]  
print(optimal_lambda)
```

```
## [1] 0.1818182
```

```
data$boxcox_TOTEXP18 <- (data$TOTEXP18^optimal_lambda - 1) / optimal_lambda  
ggplot(data, aes(x = boxcox_TOTEXP18)) +  
  geom_histogram(binwidth = .1, fill = "blue", color = "black", alpha = 0.7) +  
  ggtitle("Histogram of Total Medical Expenditures (boxcox_TOTEXP18)") +  
  xlab("Total Medical Expenditures") +  
  ylab("Frequency")
```

Histogram of Total Medical Expenditures (boxcox_TOTEXP18)



There are a lot of people with 0 expenditure. I am going to create a binary indicator variable to consider this.

```
# binary variable indicating whether expenditure is zero
# data$NO_EXP <- as.factor(ifelse(data$TOTEXP18 == 0, 1, 0))
```

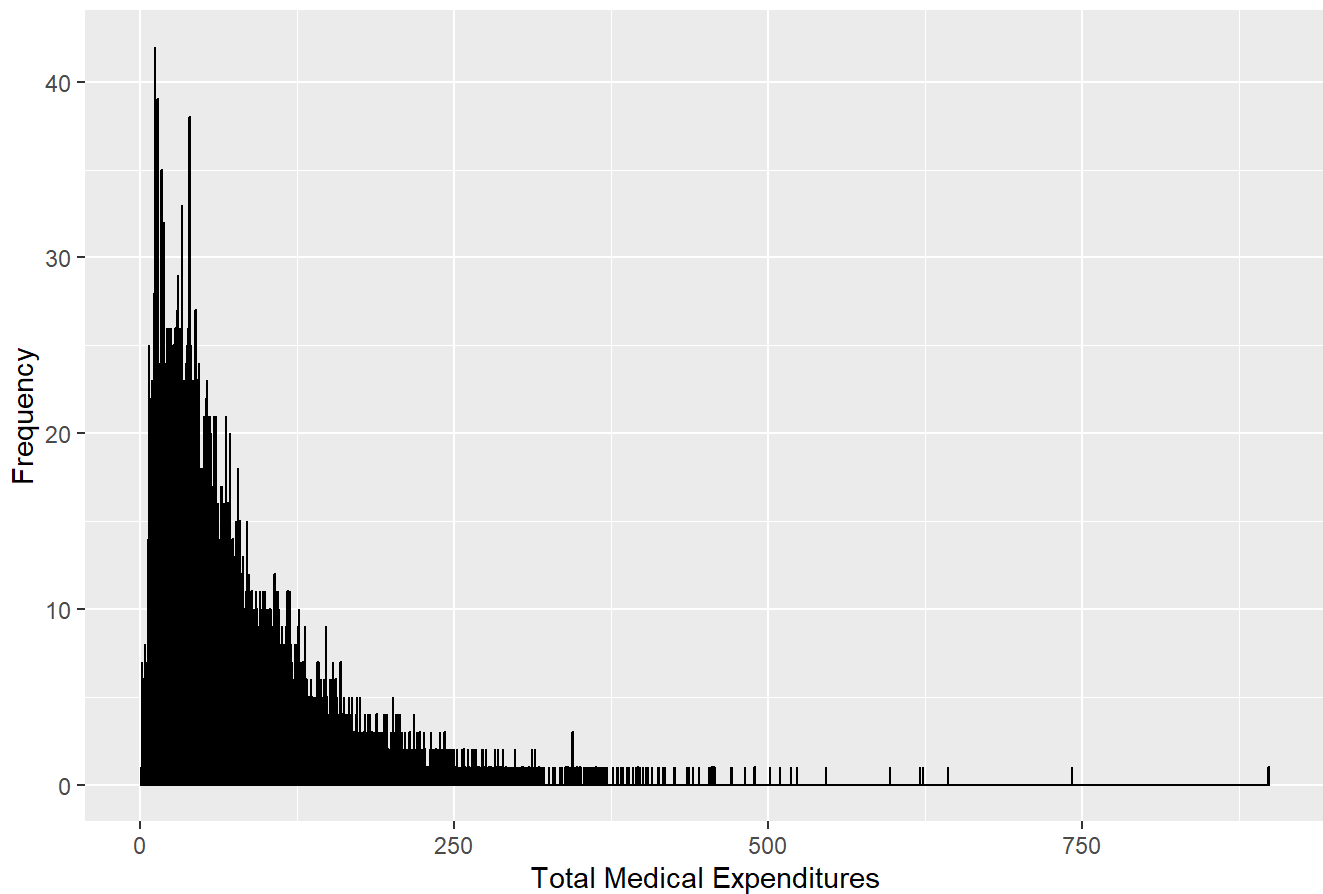
Actually, I am going to remove the 0 expenditure observations. I am only trying to predict non-zero expenditure amount.

```
data = data[data$TOTEXP18 != 0, ]
```

What about sqrt transform of response?

```
# sqrt
data$sqrt_TOTEXP18 = sqrt(data$TOTEXP18)
ggplot(data, aes(x = sqrt_TOTEXP18)) +
  geom_histogram(binwidth = .1, fill = "blue", color = "black", alpha = 0.7) +
  ggtitle("Histogram of Total Medical Expenditures (sqrt_TOTEXP18)") +
  xlab("Total Medical Expenditures") +
  ylab("Frequency")
```


Histogram of Total Medical Expenditures (sqrt_TOTEXP18)

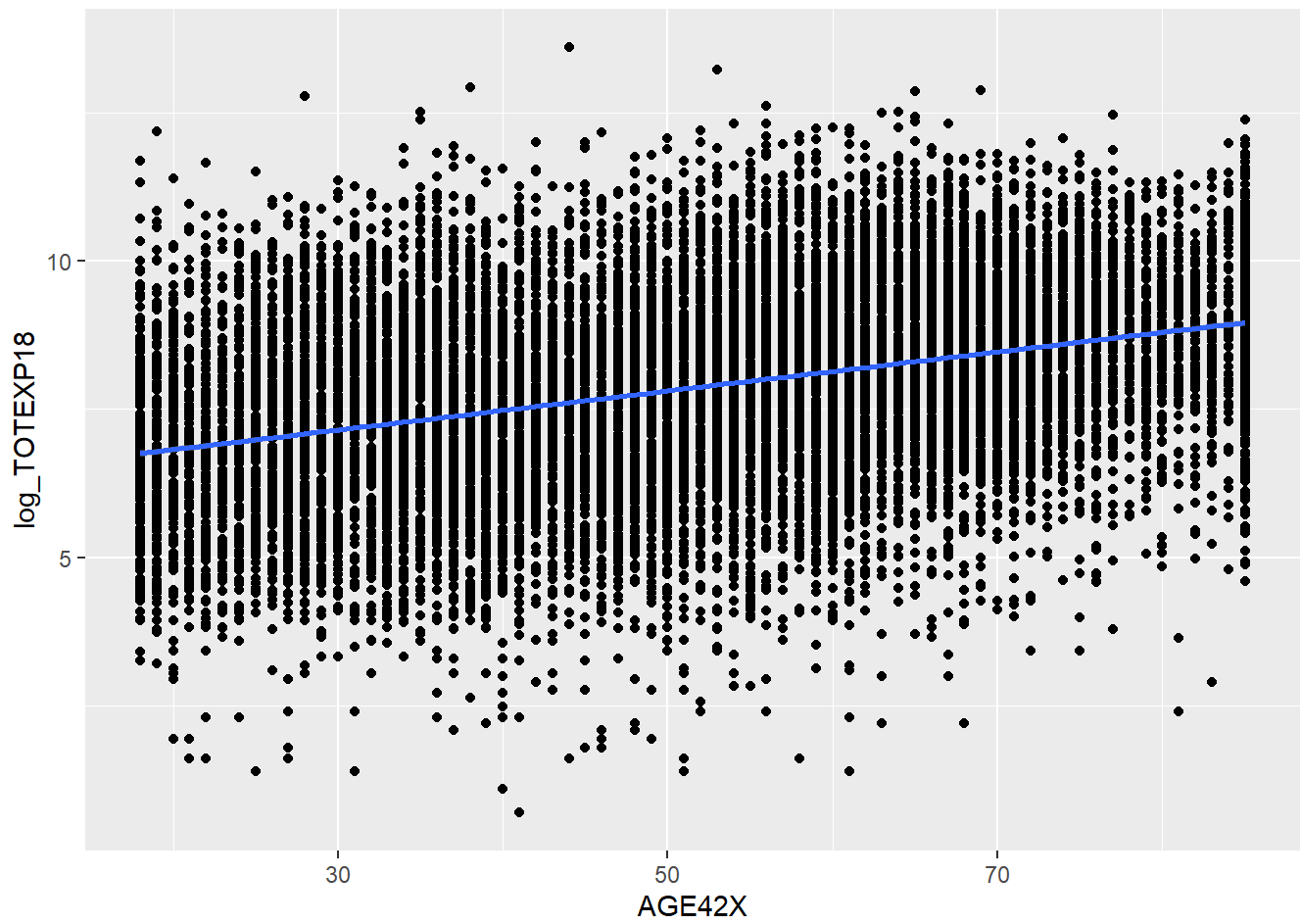


```
# Nope.  
data$sqrt_TOTEXP18 = NULL
```

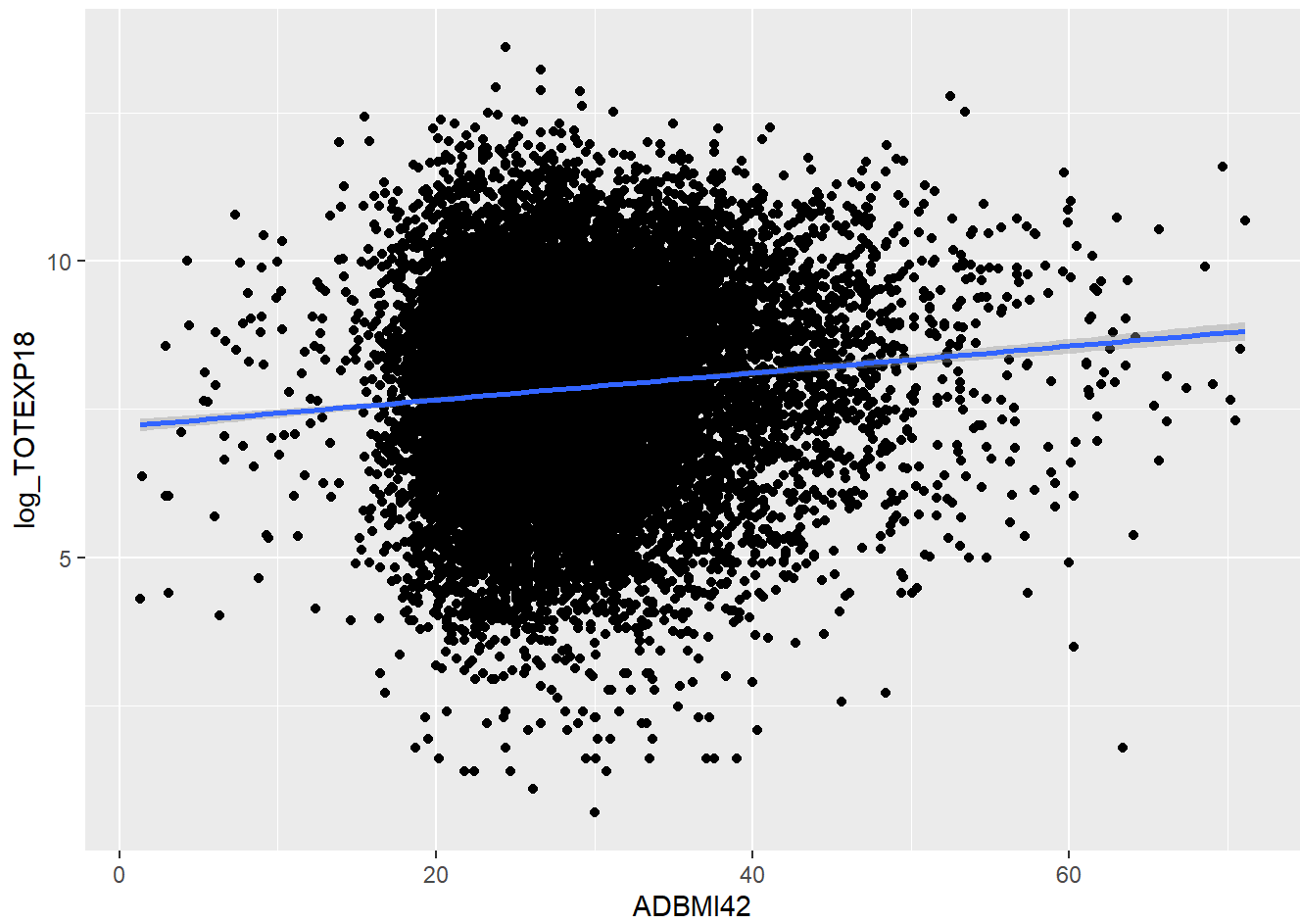
More pairwise scatter plots against log-transformed response

```
for (col in colnames(data)[!colnames(data) %in% c('log_TOTEXP18', 'boxcox_TOTEXP18', 'TOTEXP18')]) {  
  plot = ggplot(data, aes(x = !!sym(col), y = log_TOTEXP18)) + geom_point() + geom_smooth(method = "lm")  
  print(plot)  
}
```

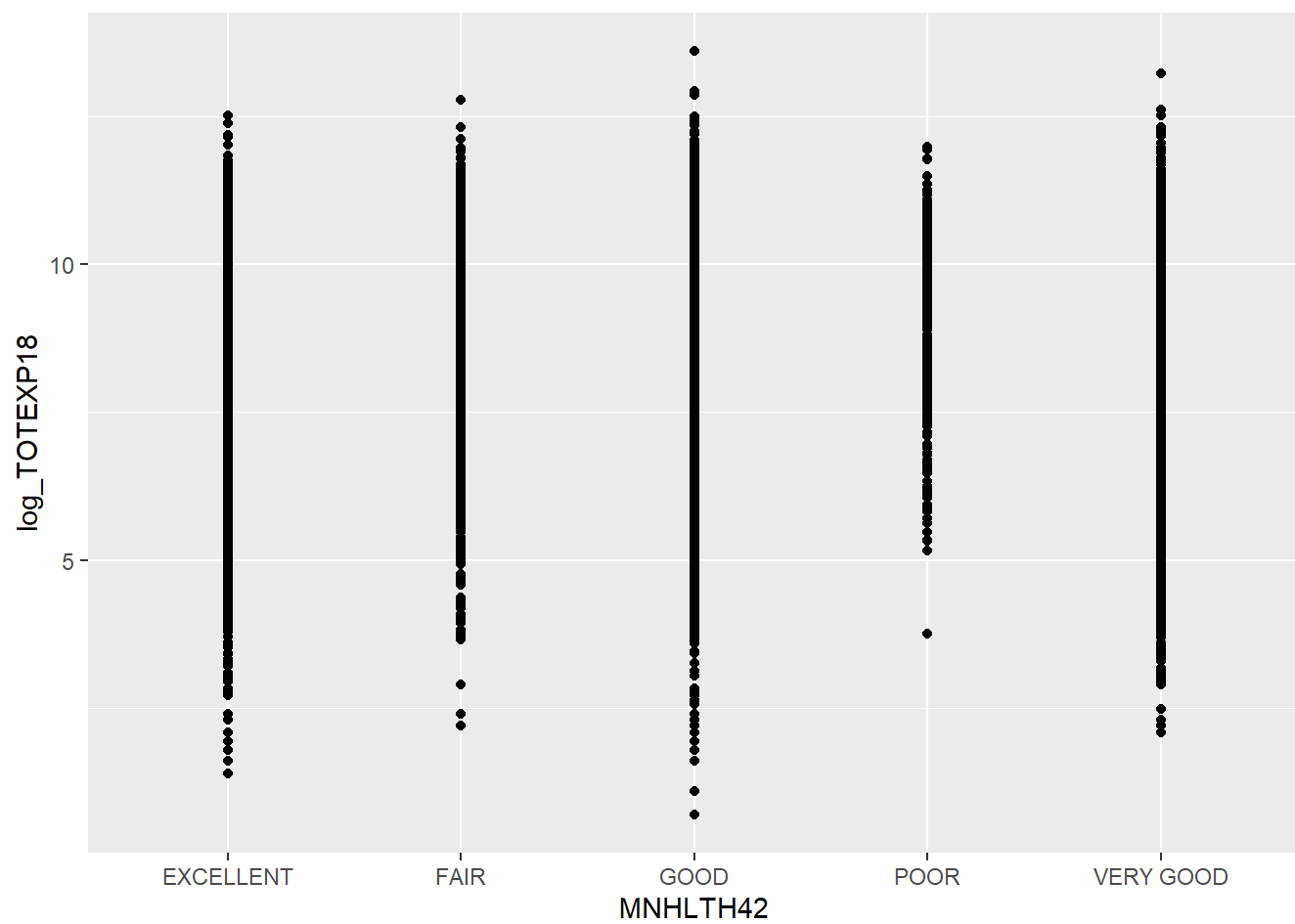
```
## `geom_smooth()` using formula = 'y ~ x'
```



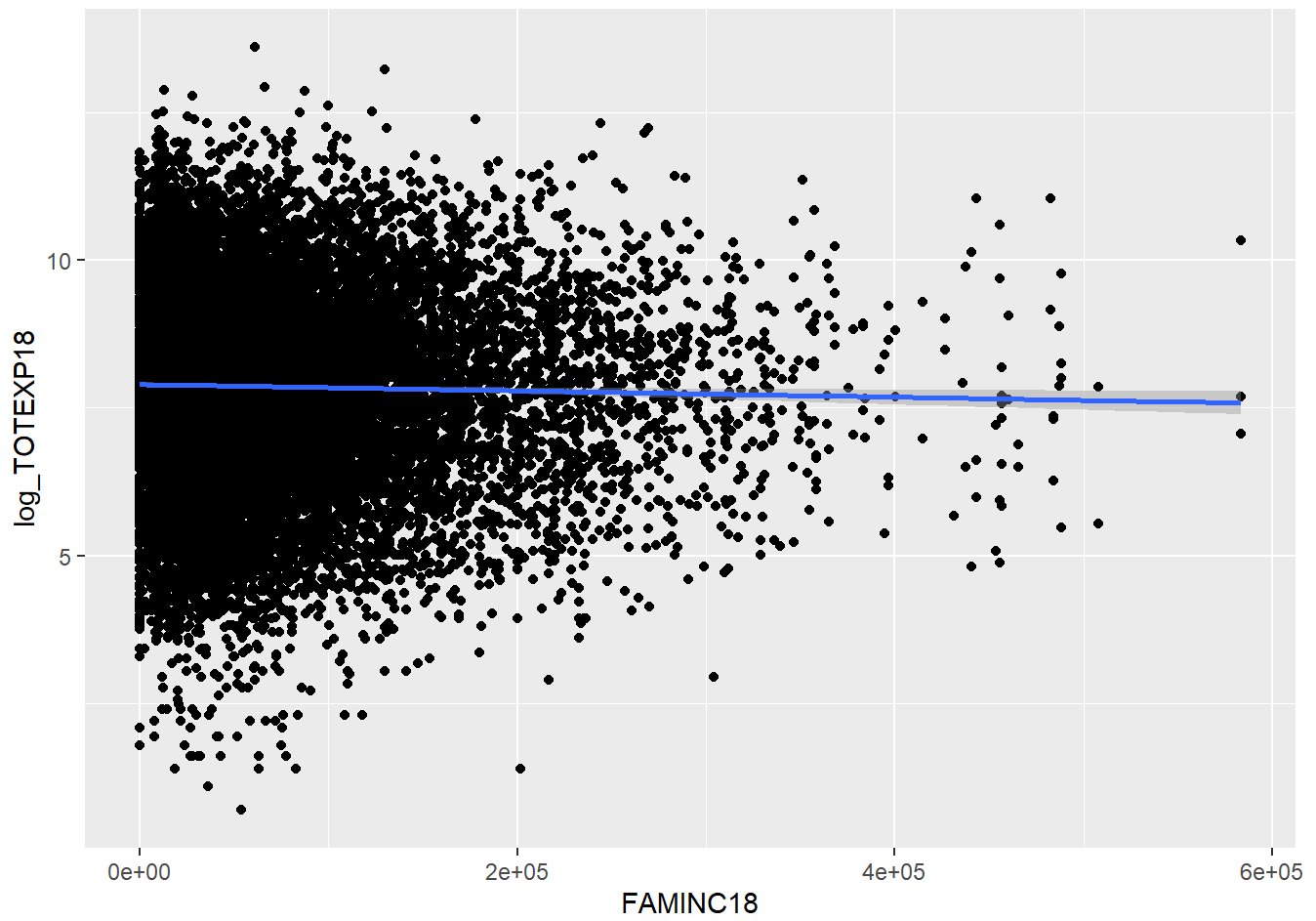
```
## `geom_smooth()` using formula = 'y ~ x'
```



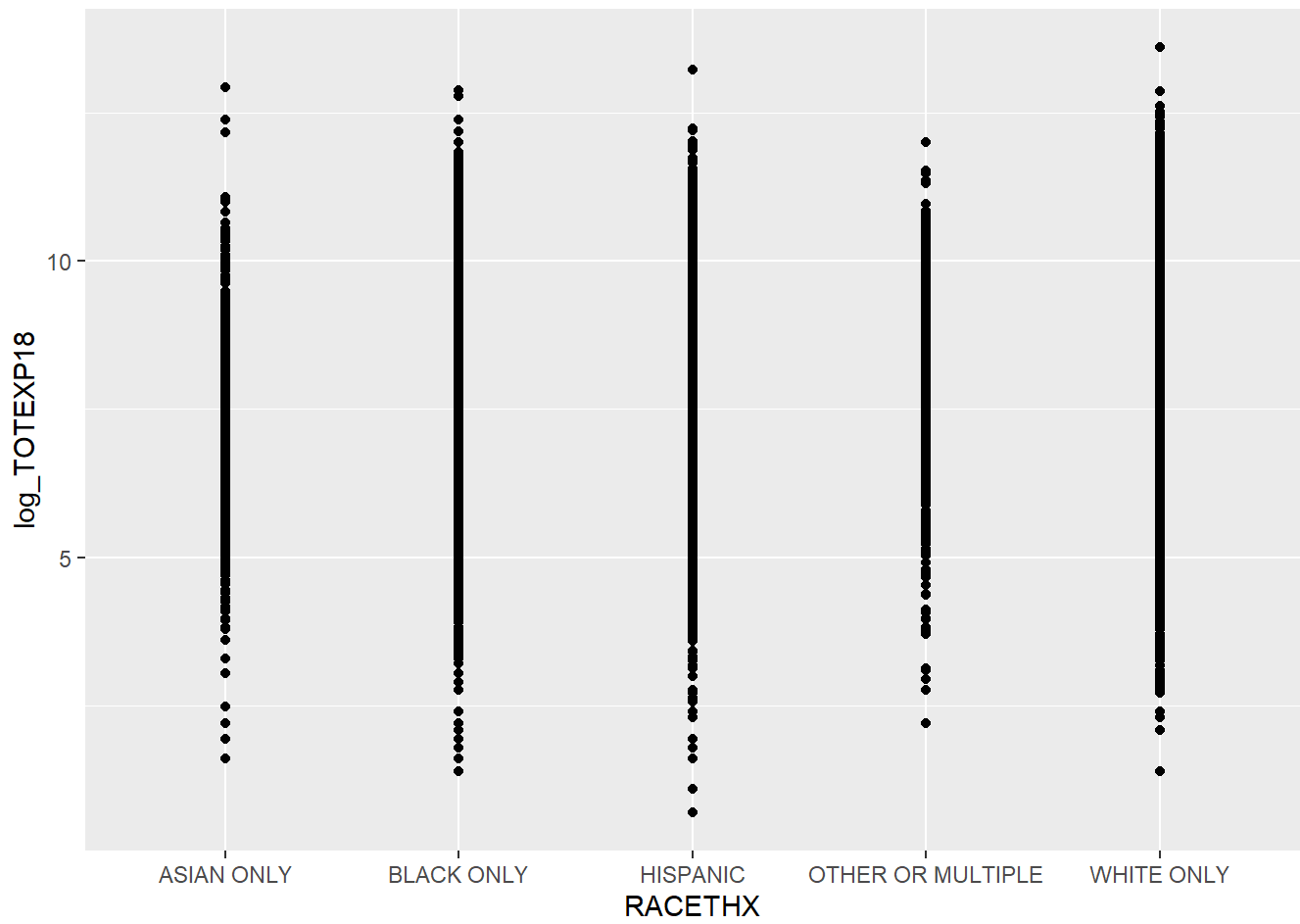
```
## `geom_smooth()` using formula = 'y ~ x'
```



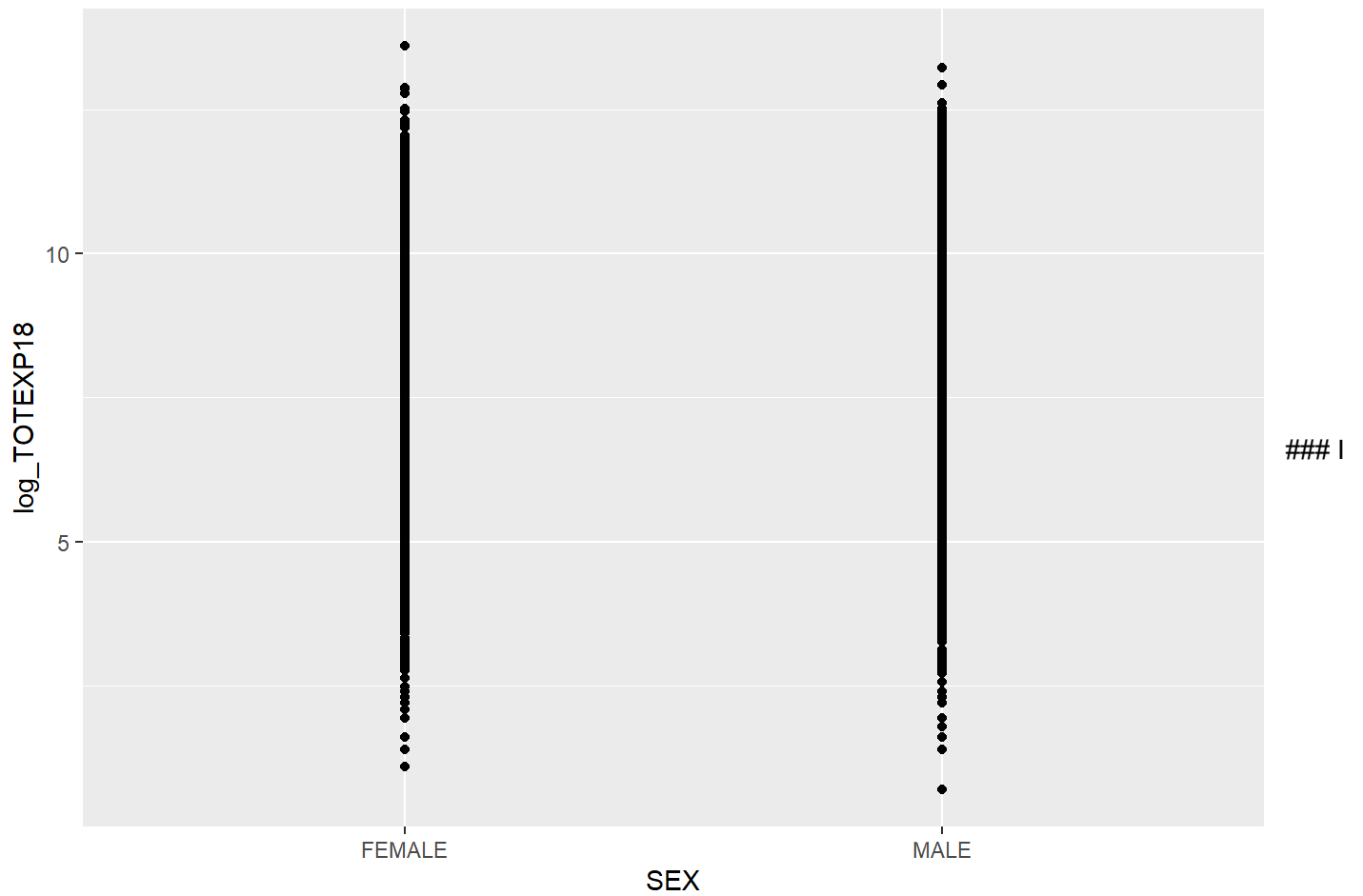
```
## `geom_smooth()` using formula = 'y ~ x'
```



```
## `geom_smooth()` using formula = 'y ~ x'
```



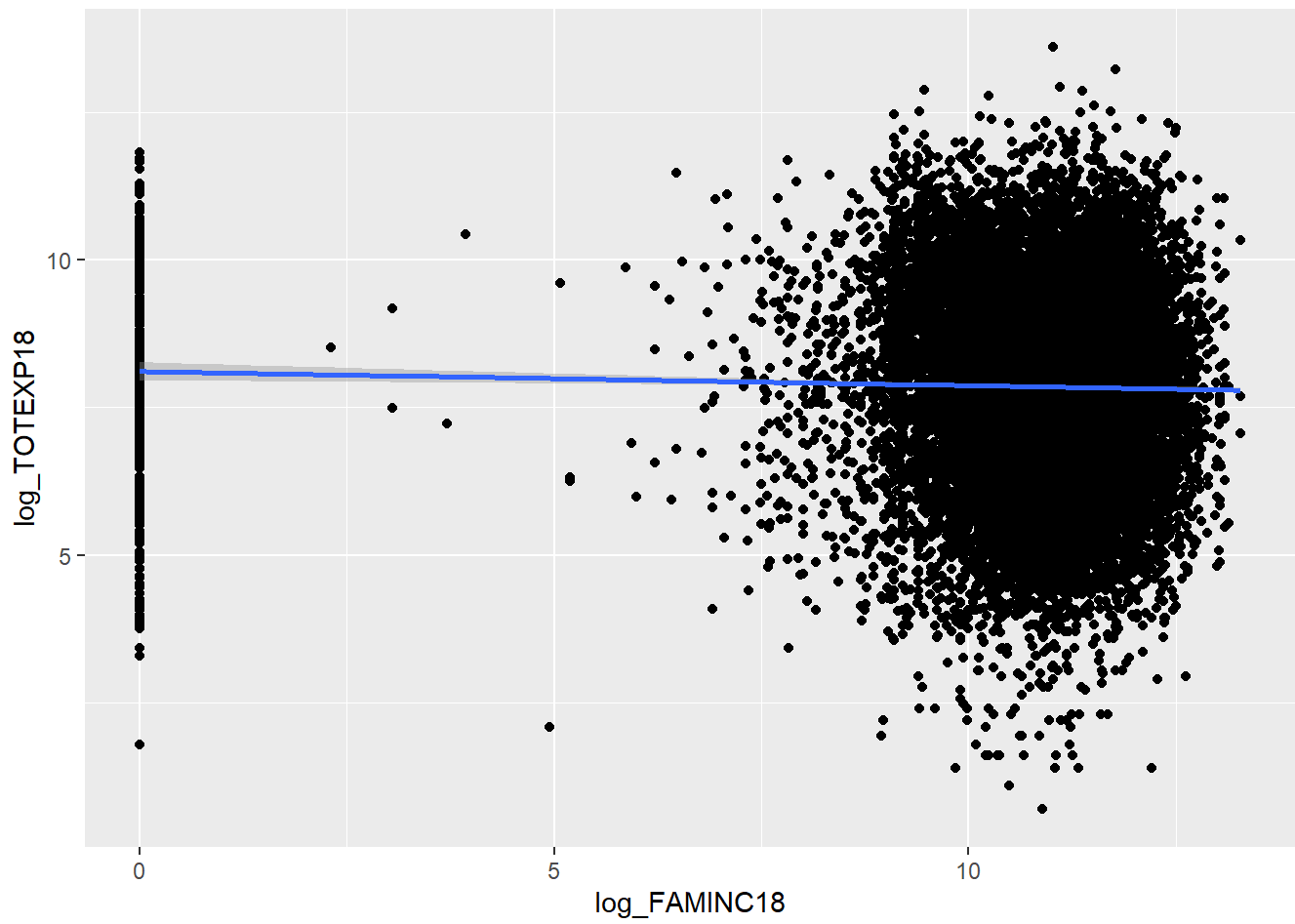
```
## `geom_smooth()` using formula = 'y ~ x'
```



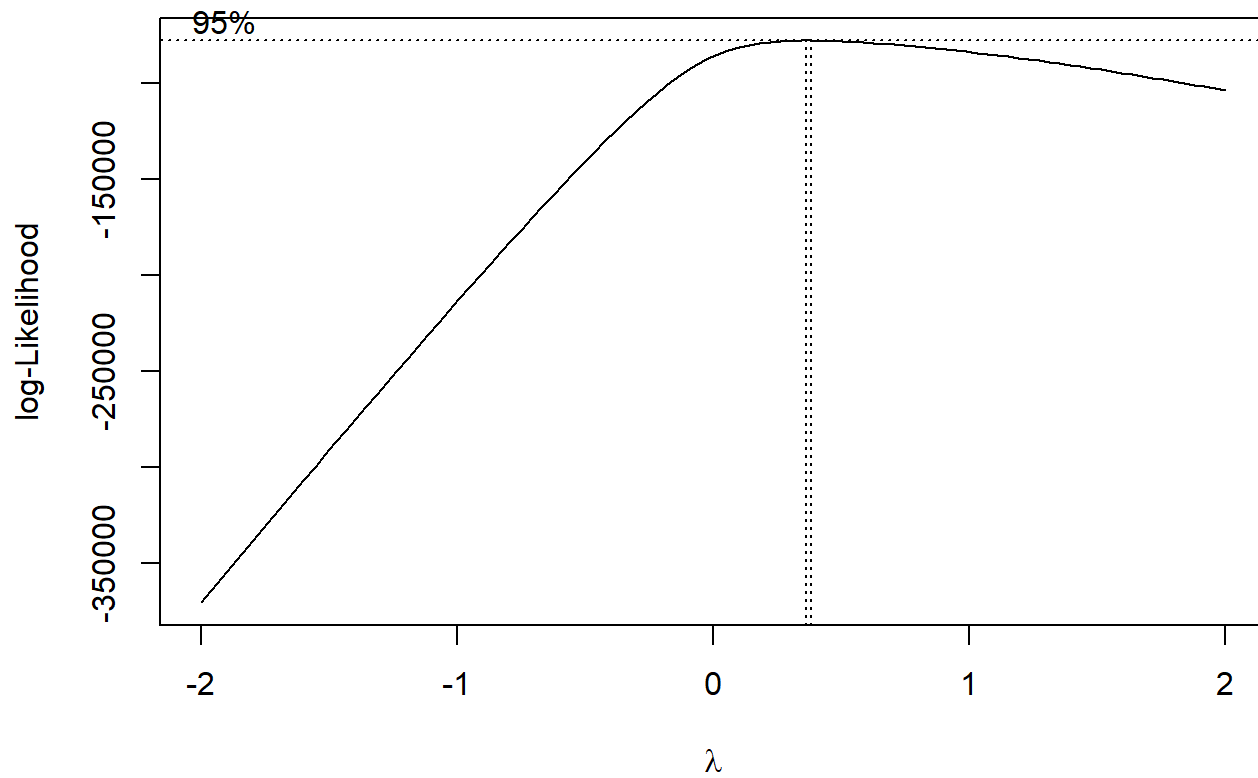
will log transform and box-cox transform family income

```
# Log
data$log_FAMINC18 = log(data$FAMINC18+1)
ggplot(data, aes(x = log_FAMINC18, y = log_TOTEXP18)) + geom_point() + geom_smooth(method = "lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
# box-cox  
boxcox_result <- boxcox(FAMINC18+1 ~ 1, data = data, lambda = seq(-2, 2, by = 0.1))
```

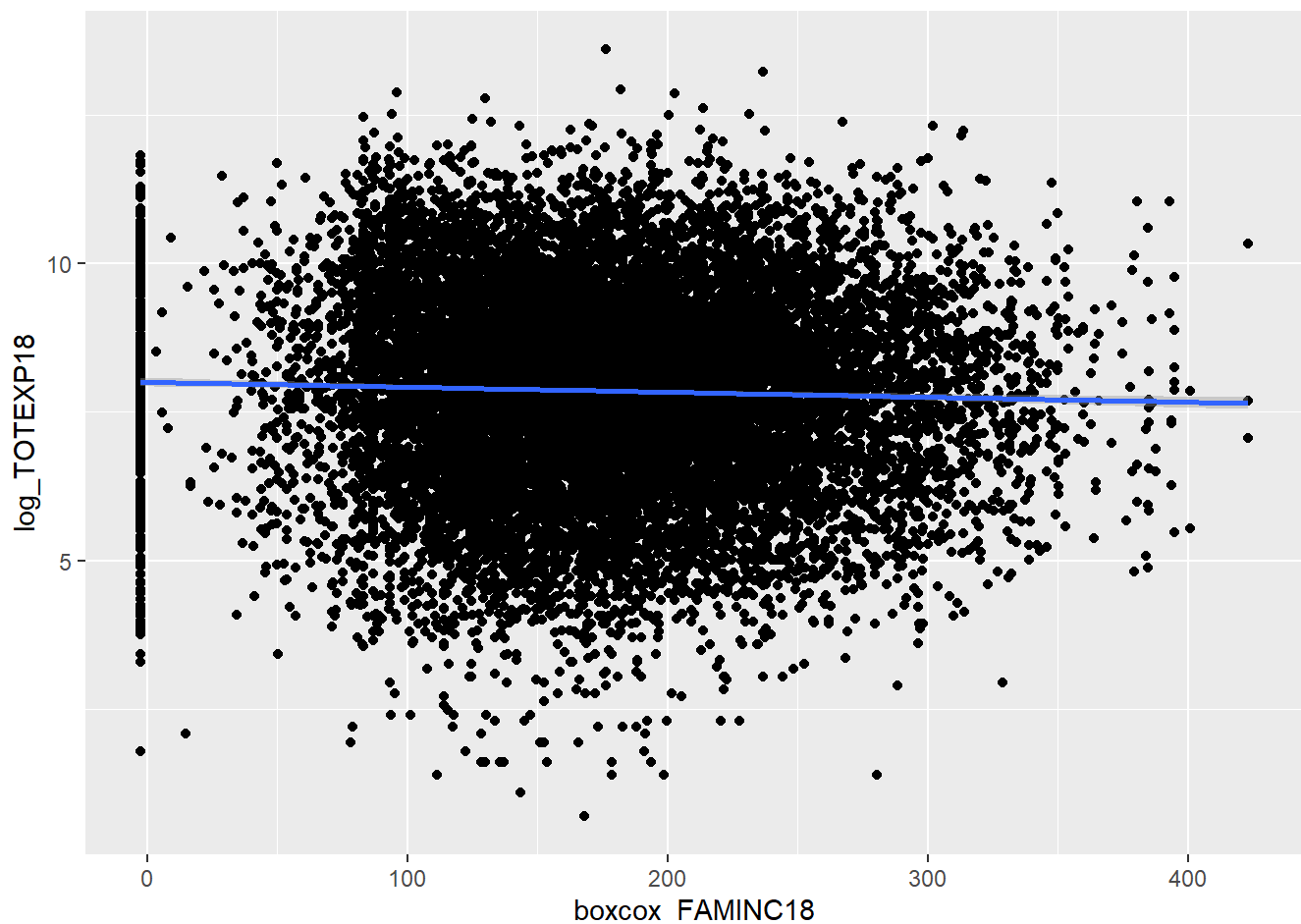



```
optimal_lambda <- boxcox_result$x[which.max(boxcox_result$y)]  
print(optimal_lambda)
```

```
## [1] 0.3838384
```

```
data$boxcox_FAMINC18 <- (data$FAMINC18^optimal_lambda - 1) / optimal_lambda  
ggplot(data, aes(x = boxcox_FAMINC18, y = log_TOTEXP18)) + geom_point() + geom_smooth(method =  
"lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
# box-cox looks better
data$log_FAMINC18 = NULL
# data$FAMINC18 = NULL
```

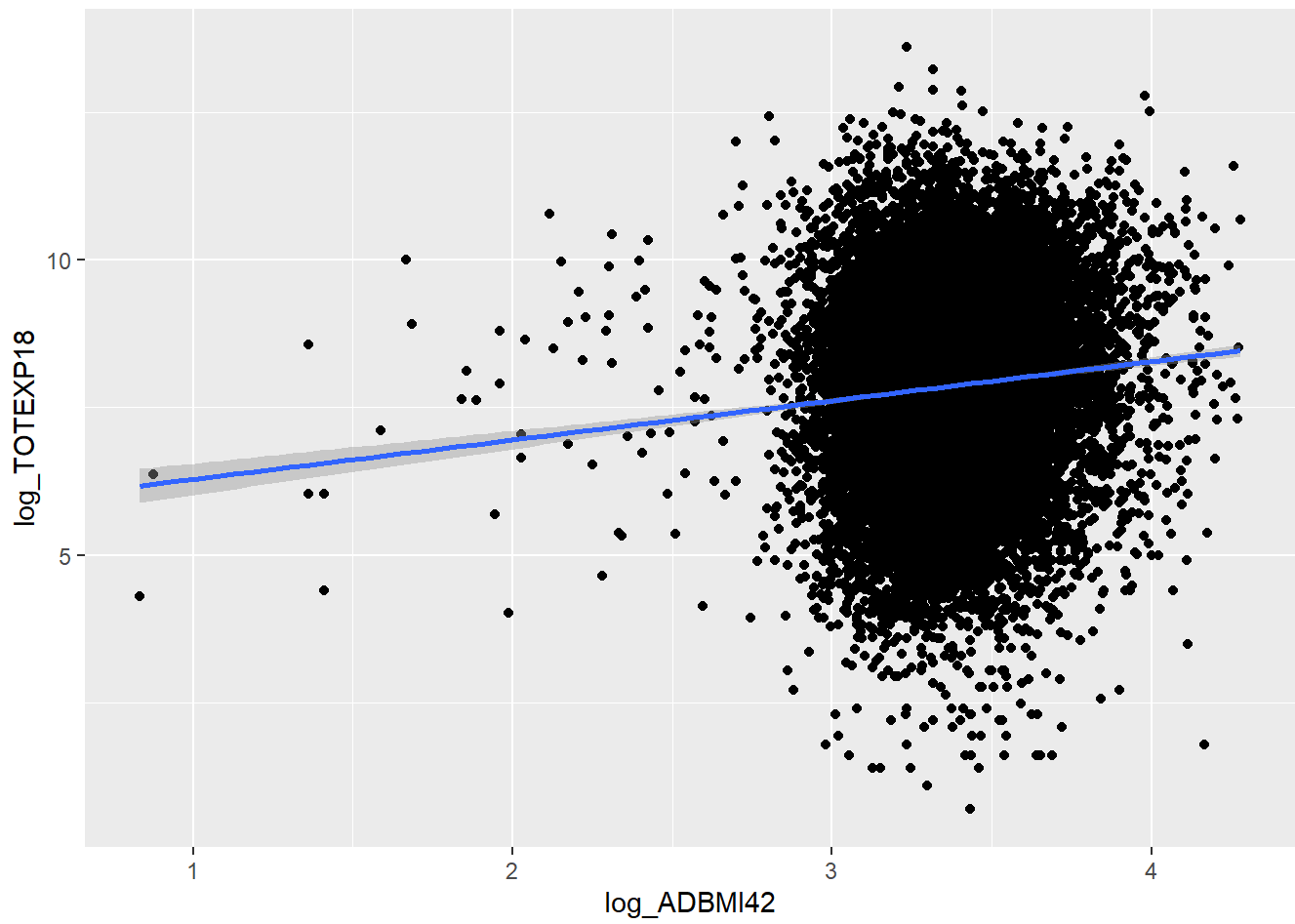
There are a lot of people with very low/no family income. I am going to create a binary indicator variable to consider this.

```
data$NO_INC <- as.factor(ifelse(data$boxcox_FAMINC18 < 0, 1, 0))
```

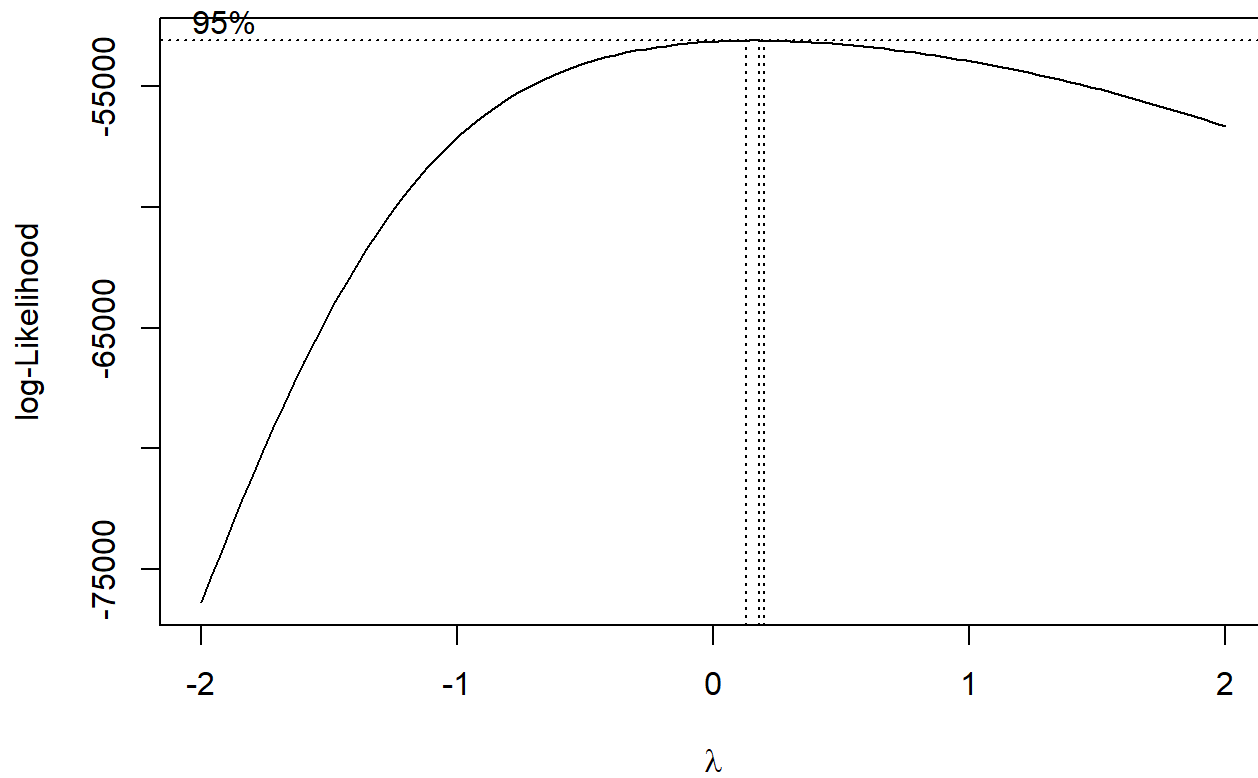
I will log transform and box-cox transform BMI

```
# Log
data$log_ADBMI42 = log(data$ADBMI42+1)
ggplot(data, aes(x = log_ADBMI42, y = log_TOTEXP18)) + geom_point() + geom_smooth(method = "lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
# box-cox  
boxcox_result <- boxcox(ADBMI42+1 ~ 1, data = data, lambda = seq(-2, 2, by = 0.1))
```

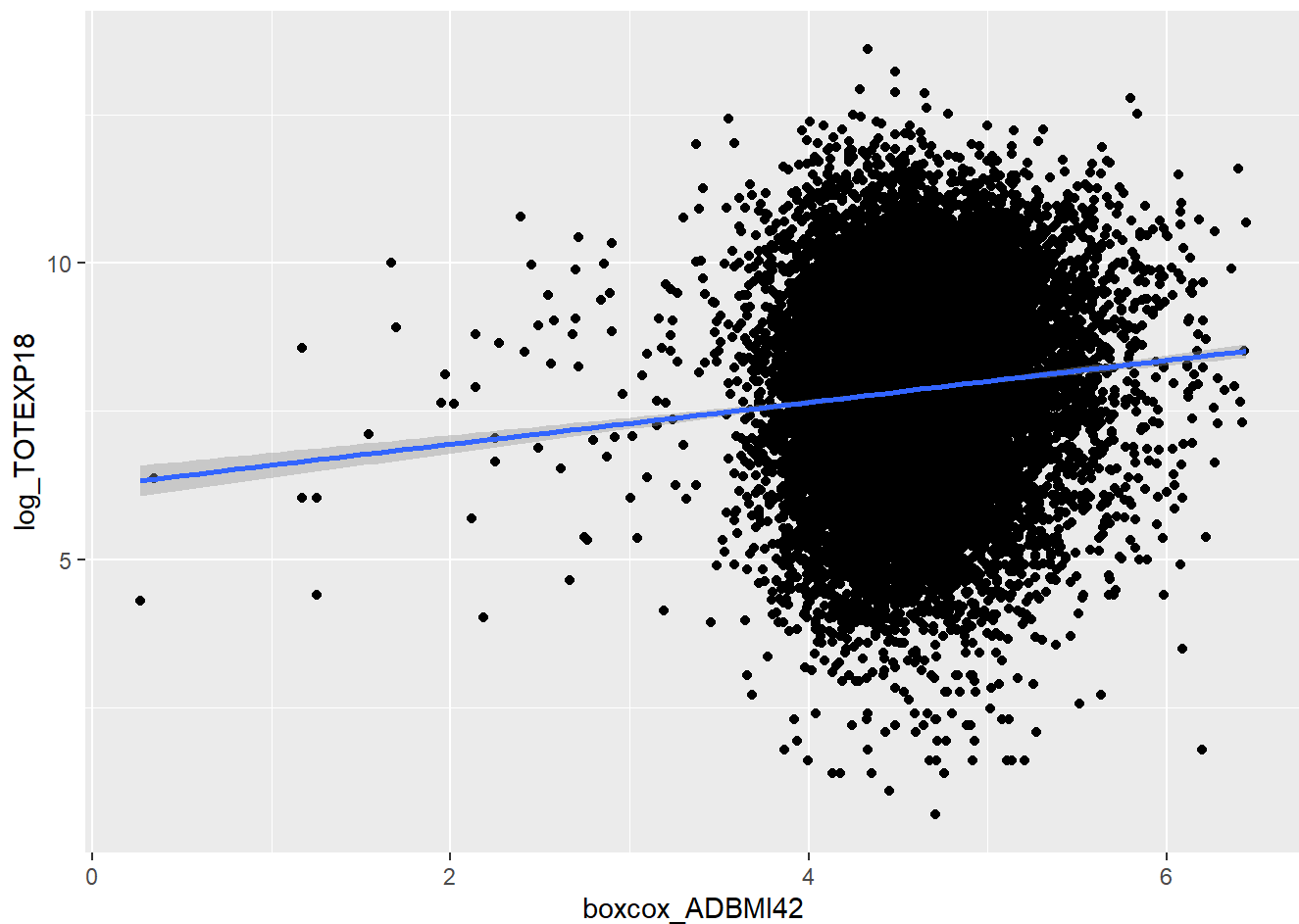


```
optimal_lambda <- boxcox_result$x[which.max(boxcox_result$y)]  
print(optimal_lambda)
```

```
## [1] 0.1818182
```

```
data$boxcox_ADBMI42 <- (data$ADBMI42^optimal_lambda - 1) / optimal_lambda  
ggplot(data, aes(x = boxcox_ADBMI42, y = log_TOTEXP18)) + geom_point() + geom_smooth(method = "lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



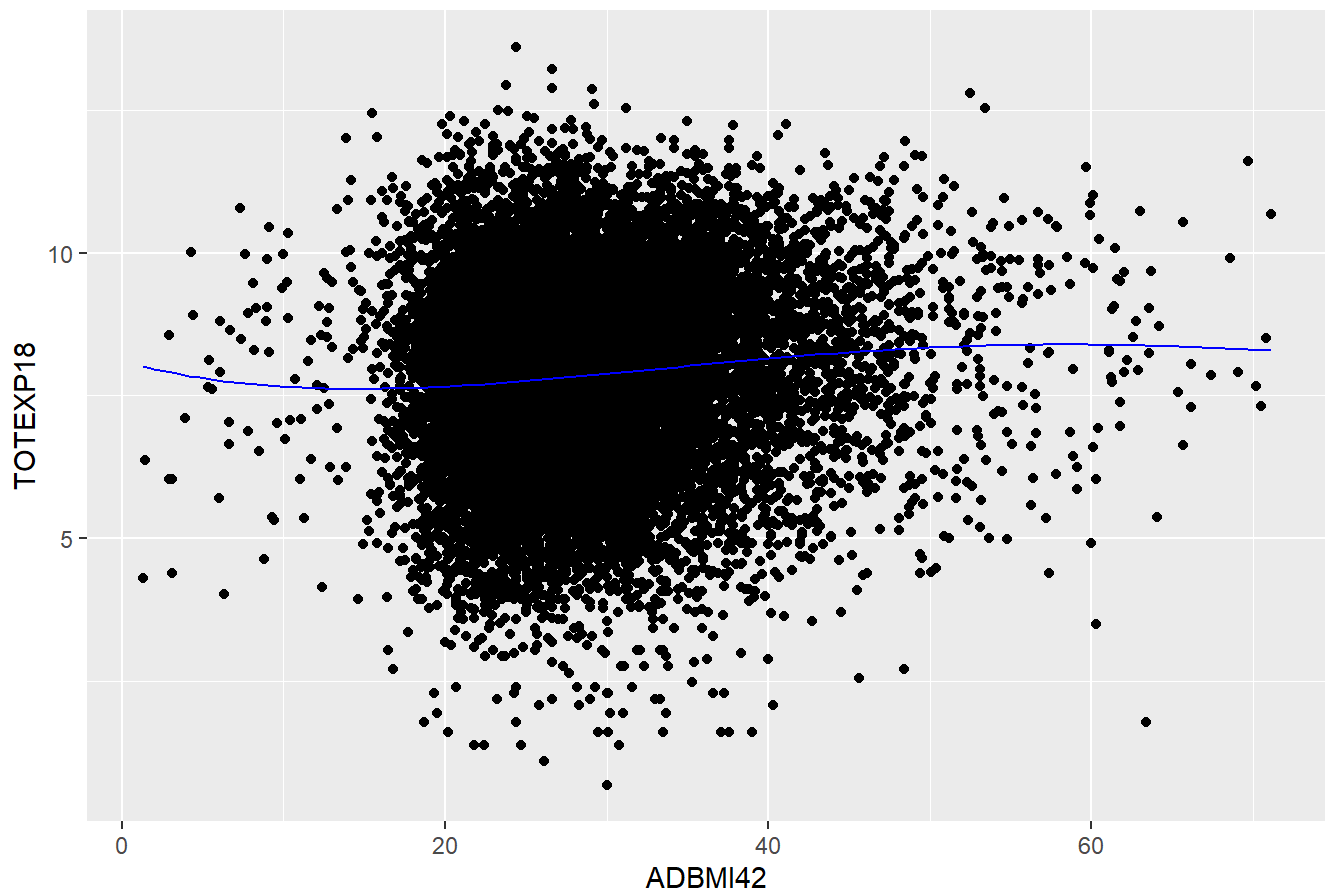
```
# don't love either
data$log_ADBMI42 = NULL
data$boxcox_ADBMI42 = NULL
```

I will try to b-spline transform the BMI, Age, and Income predictors

```
# BMI
bs_model = lm(log_TOTEXP18 ~ bs(ADBMI42, degree = 4), data = data) # degree 4 looks good
new_data = data.frame(ADBMI42 = seq(min(data$ADBMI42), max(data$ADBMI42), length.out = 100))
new_data$log_TOTEXP18_pred = predict(bs_model, newdata = new_data)

# Plot the data and the fitted curve
library(ggplot2)
ggplot(data, aes(x = ADBMI42, y = log_TOTEXP18)) +
  geom_point() +
  geom_line(data = new_data, aes(x = ADBMI42, y = log_TOTEXP18_pred), color = "blue") +
  labs(title = "B-Splines Transformation of ADBMI42 on TOTEXP18",
       x = "ADBMI42", y = "TOTEXP18")
```

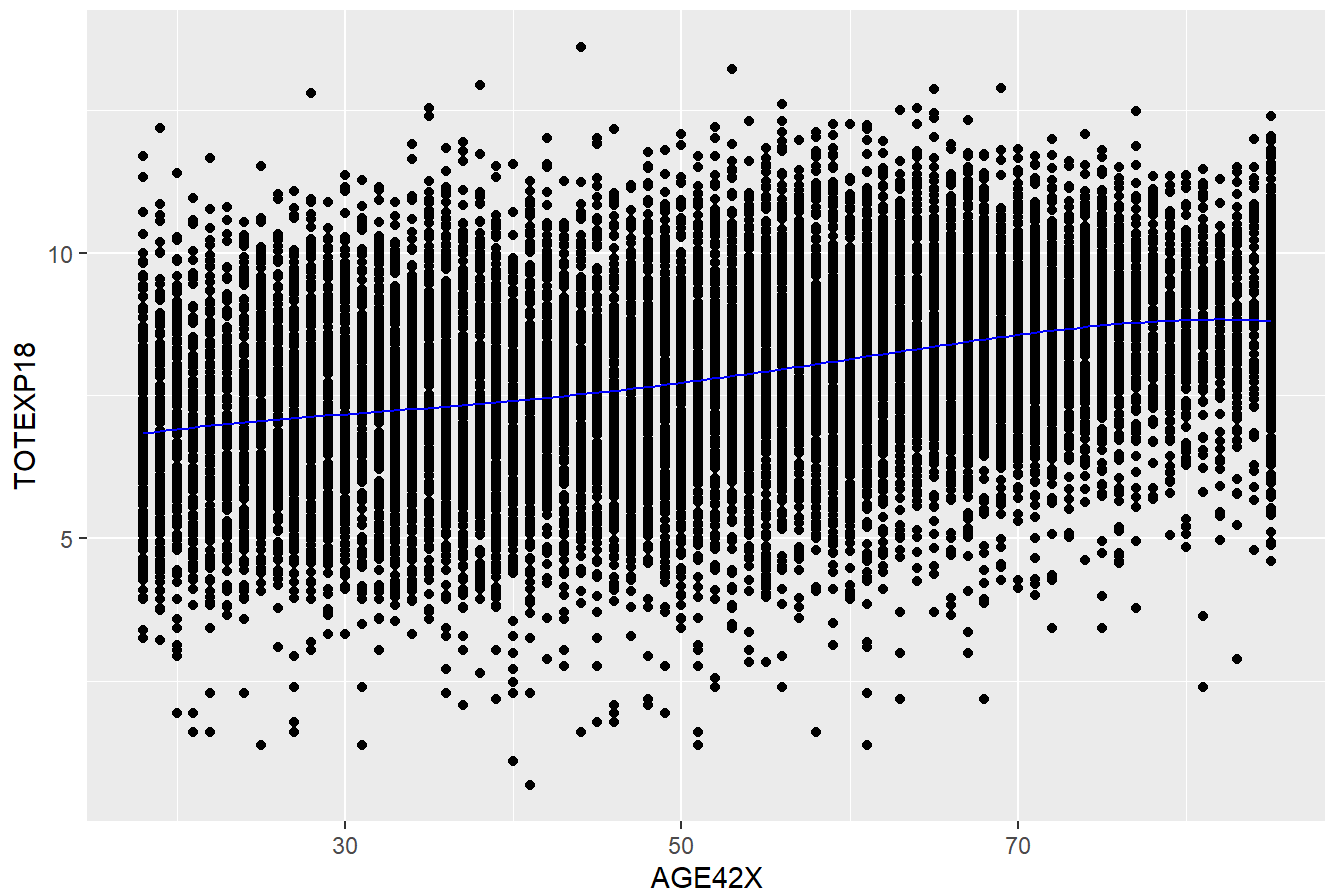
B-Splines Transformation of ADBMI42 on TOTEXP18



```
# AGE
bs_model = lm(log_TOTEXP18 ~ bs(AGE42X, degree = 4), data = data) # degree 4 Looks good
new_data = data.frame(AGE42X = seq(min(data$AGE42X), max(data$AGE42X), length.out = 100))
new_data$log_TOTEXP18_pred = predict(bs_model, newdata = new_data)

# Plot the data and the fitted curve
library(ggplot2)
ggplot(data, aes(x = AGE42X, y = log_TOTEXP18)) +
  geom_point() +
  geom_line(data = new_data, aes(x = AGE42X, y = log_TOTEXP18_pred), color = "blue") +
  labs(title = "B-Splines Transformation of AGE42X on TOTEXP18",
       x = "AGE42X", y = "TOTEXP18")
```

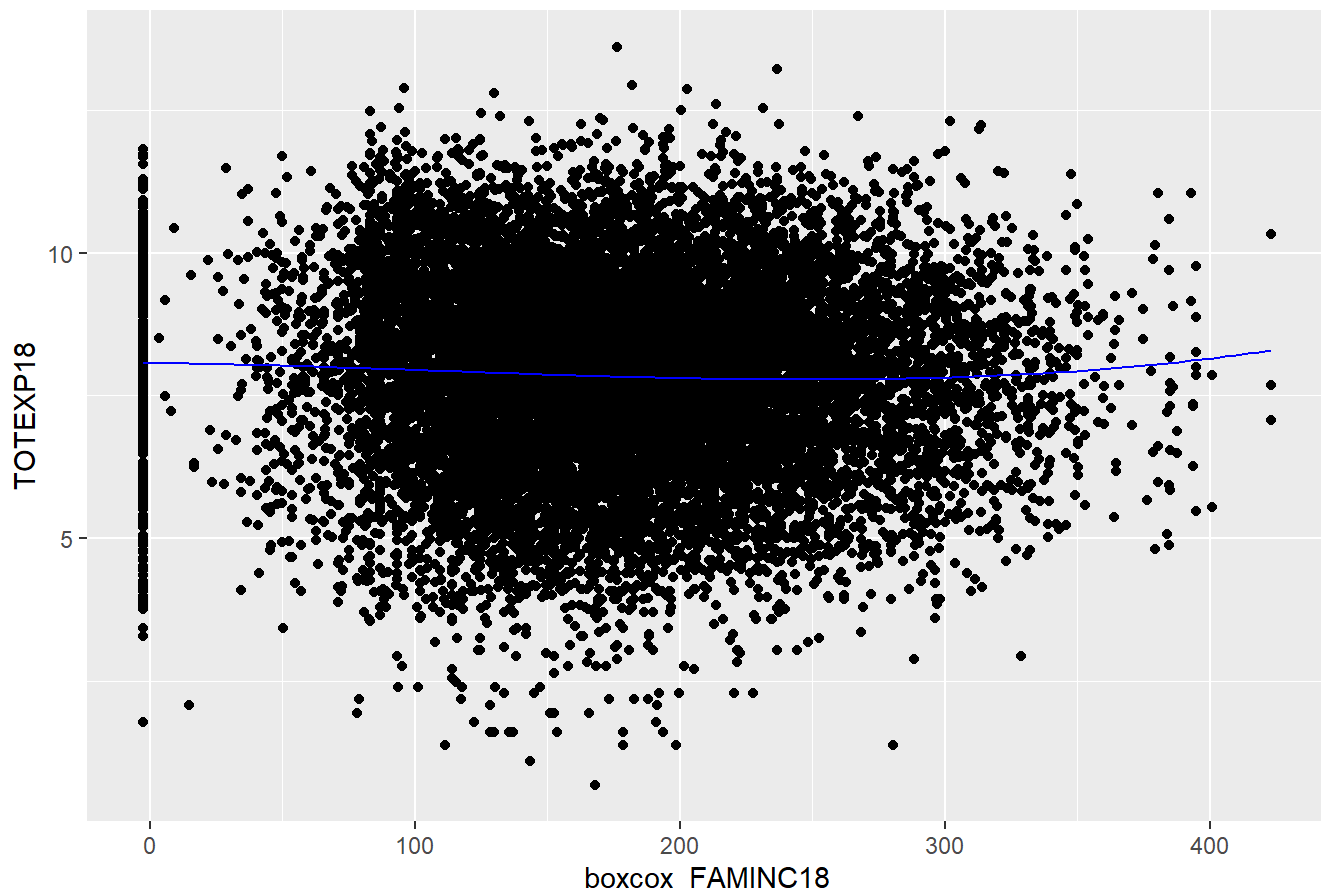
B-Splines Transformation of AGE42X on TOTEXP18



```
# INCOME
bs_model = lm(log_TOTEXP18 ~ bs(boxcox_FAMINC18, degree = 3), data = data) # degree 3 Looks good
new_data = data.frame(boxcox_FAMINC18 = seq(min(data$boxcox_FAMINC18), max(data$boxcox_FAMINC18), length.out = 100))
new_data$log_TOTEXP18_pred = predict(bs_model, newdata = new_data)

# Plot the data and the fitted curve
library(ggplot2)
ggplot(data, aes(x = boxcox_FAMINC18, y = log_TOTEXP18)) +
  geom_point() +
  geom_line(data = new_data, aes(x = boxcox_FAMINC18, y = log_TOTEXP18_pred), color = "blue") +
  labs(title = "B-Splines Transformation of boxcox_FAMINC18 on TOTEXP18",
        x = "boxcox_FAMINC18", y = "TOTEXP18")
```

B-Splines Transformation of boxcox_FAMINC18 on TOTEXP18

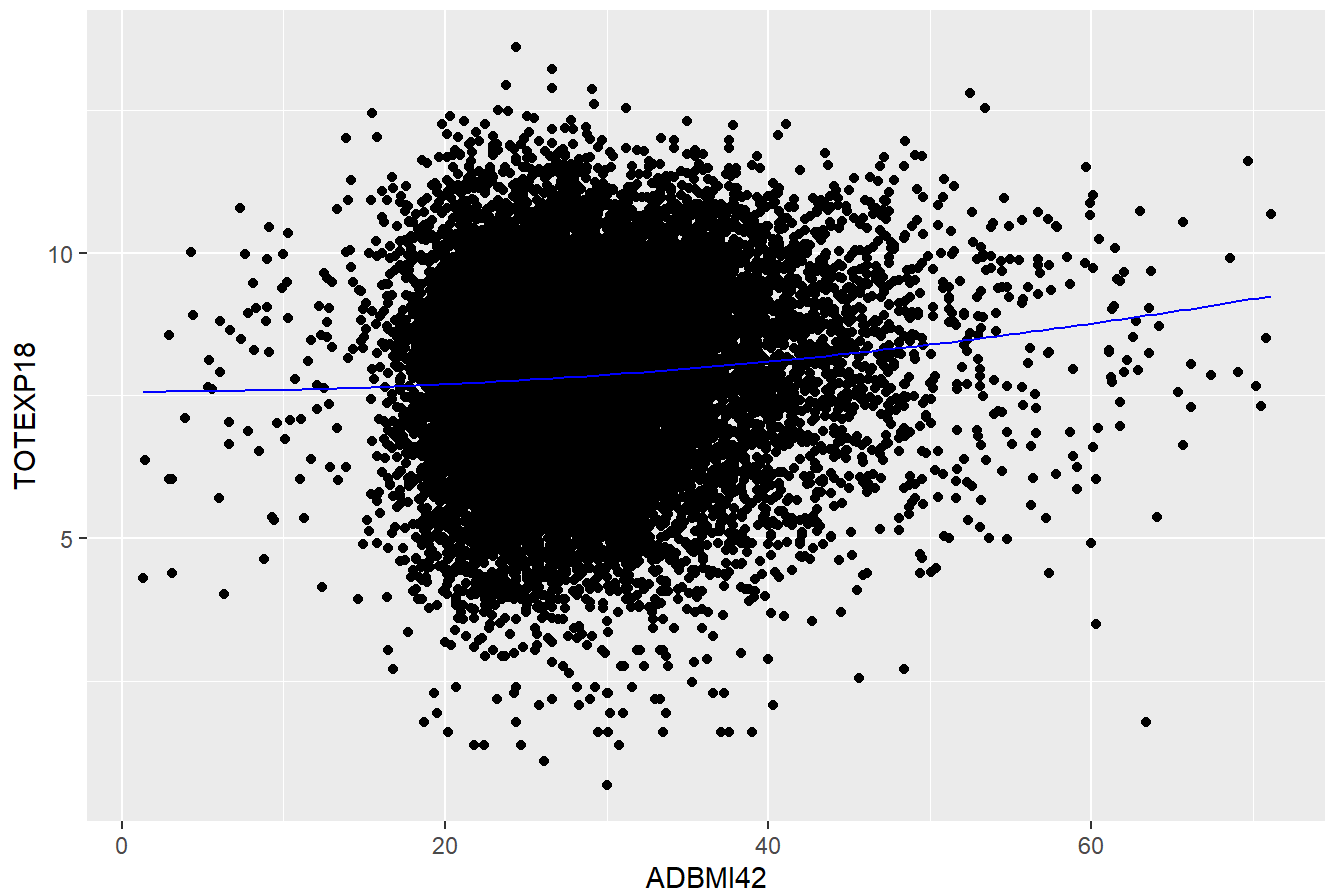


I will also try to quadratic transform the BMI predictor

```
bs_model = lm(log_TOTEXP18 ~ I(ADBMI42^2), data = data)
new_data = data.frame(ADBMI42 = seq(min(data$ADBMI42), max(data$ADBMI42), length.out = 100))
new_data$log_TOTEXP18_pred = predict(bs_model, newdata = new_data)

# Plot the data and the fitted curve
library(ggplot2)
ggplot(data, aes(x = ADBMI42, y = log_TOTEXP18)) +
  geom_point() +
  geom_line(data = new_data, aes(x = ADBMI42, y = log_TOTEXP18_pred), color = "blue") +
  labs(title = "Quadratic of ADBMI42 on TOTEXP18",
       x = "ADBMI42", y = "TOTEXP18")
```


Quadratic of ADBMI42 on TOTEXP18

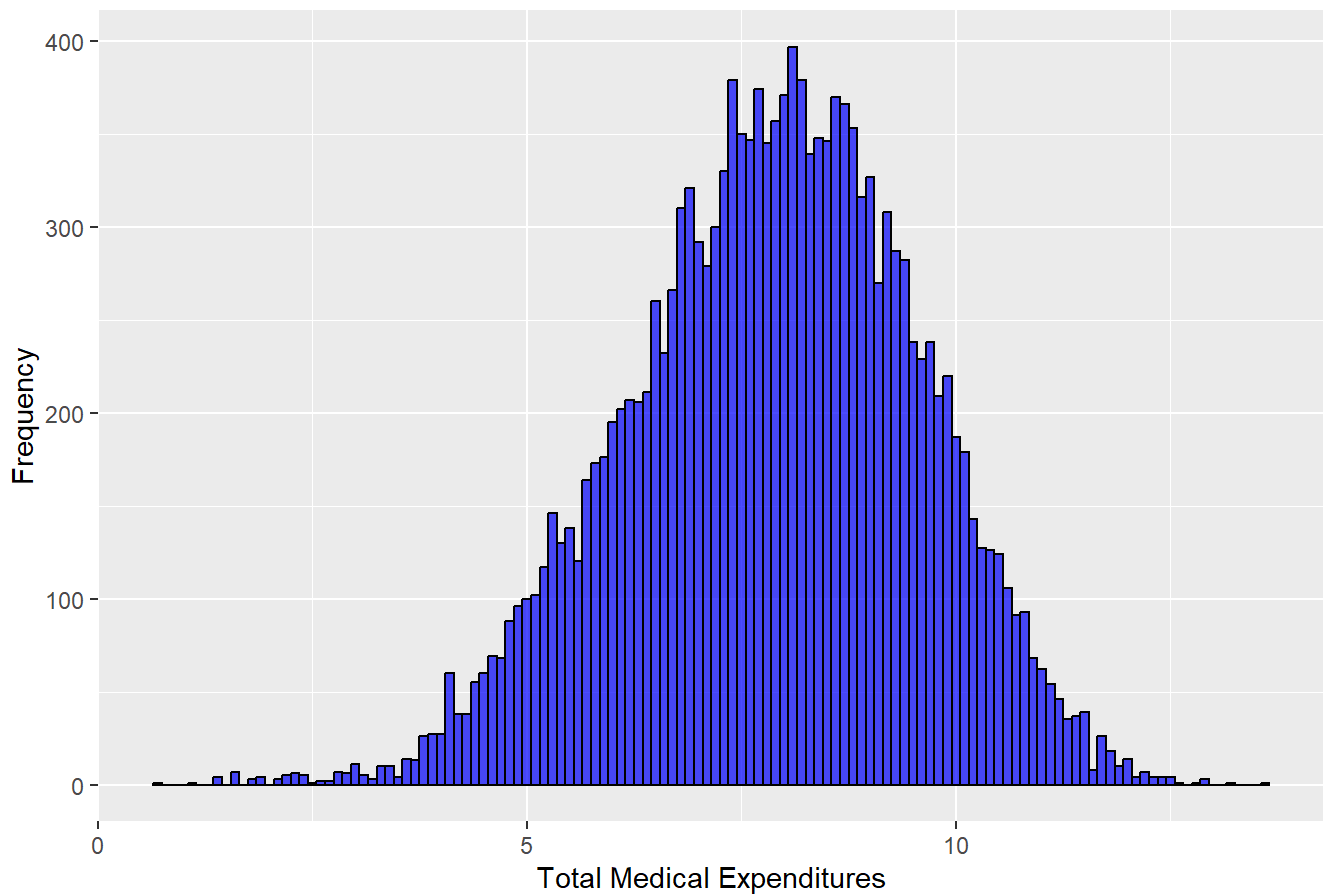


```
# This might be better than the b-spline
```

Final plots of transformed variables

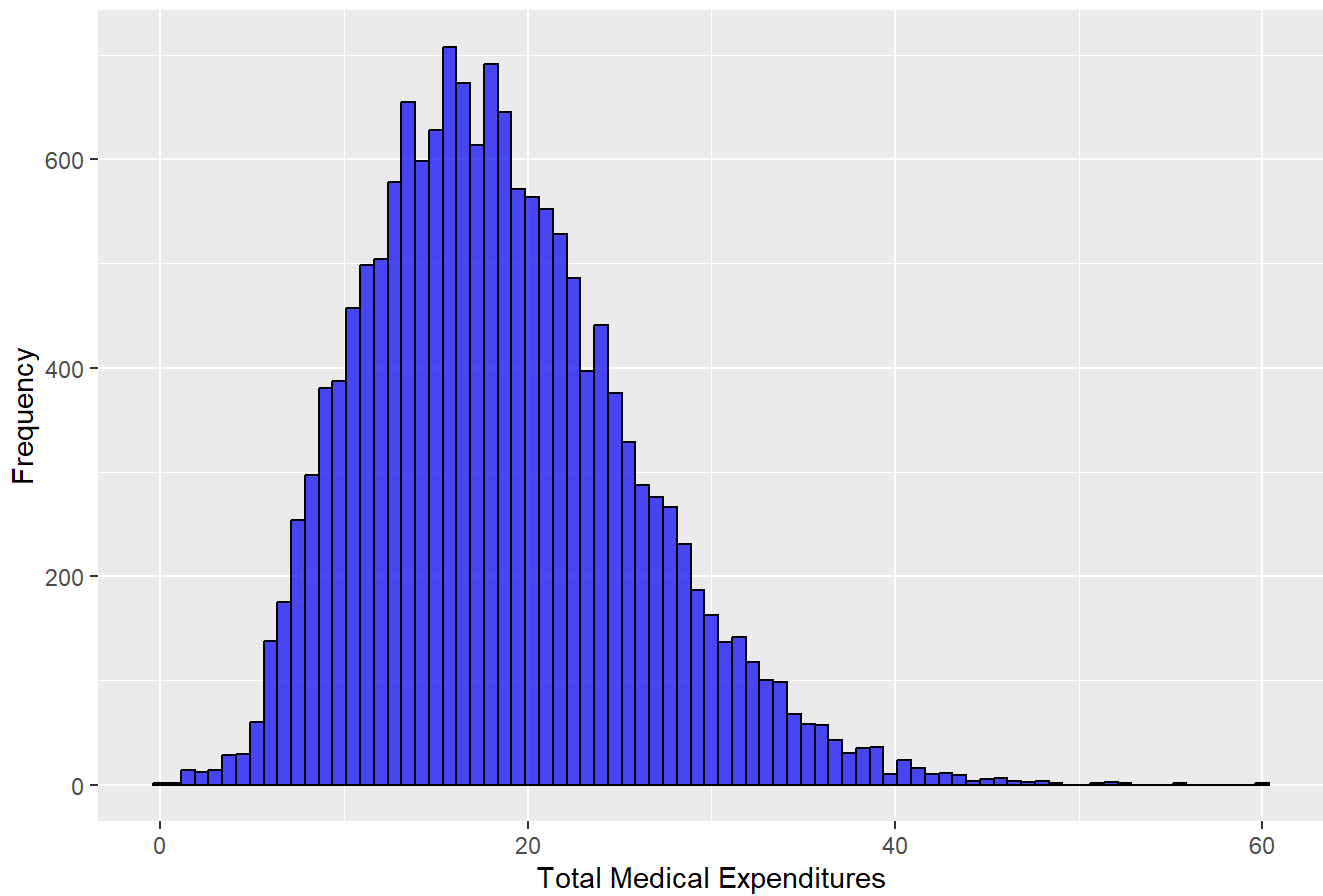
```
ggplot(data, aes(x = log_TOTEXP18)) +  
  geom_histogram(binwidth = .1, fill = "blue", color = "black", alpha = 0.7) +  
  ggtitle("Histogram of Total Medical Expenditures (log_TOTEXP18)") +  
  xlab("Total Medical Expenditures") +  
  ylab("Frequency")
```

Histogram of Total Medical Expenditures (log_TOTEXP18)



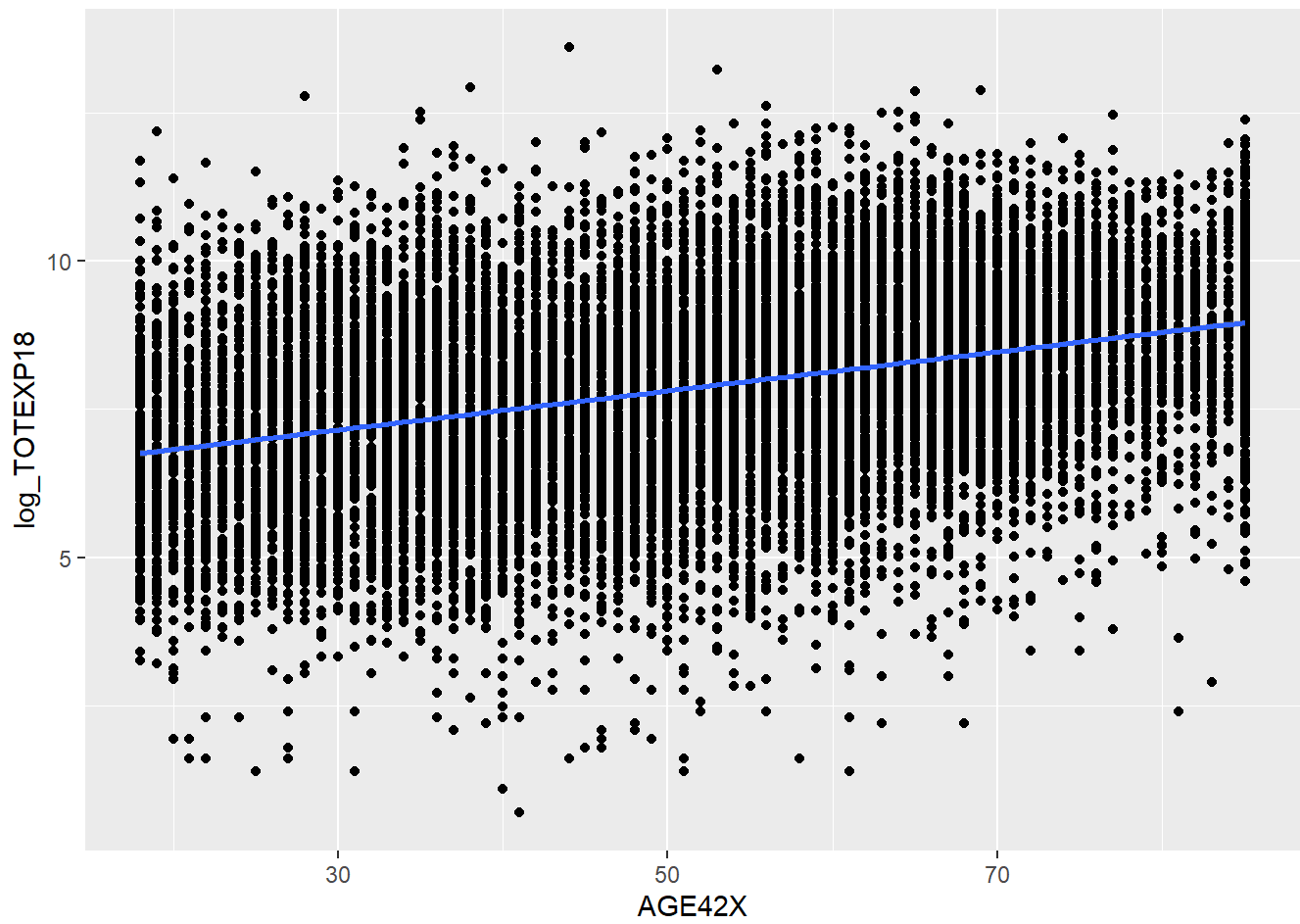
```
ggplot(data, aes(x = boxcox_TOTEXP18)) +  
  geom_histogram(binwidth = .75, fill = "blue", color = "black", alpha = 0.7) +  
  ggtitle("Histogram of Total Medical Expenditures (boxcox_TOTEXP18)") +  
  xlab("Total Medical Expenditures") +  
  ylab("Frequency")
```

Histogram of Total Medical Expenditures (boxcox_TOTEXP18)

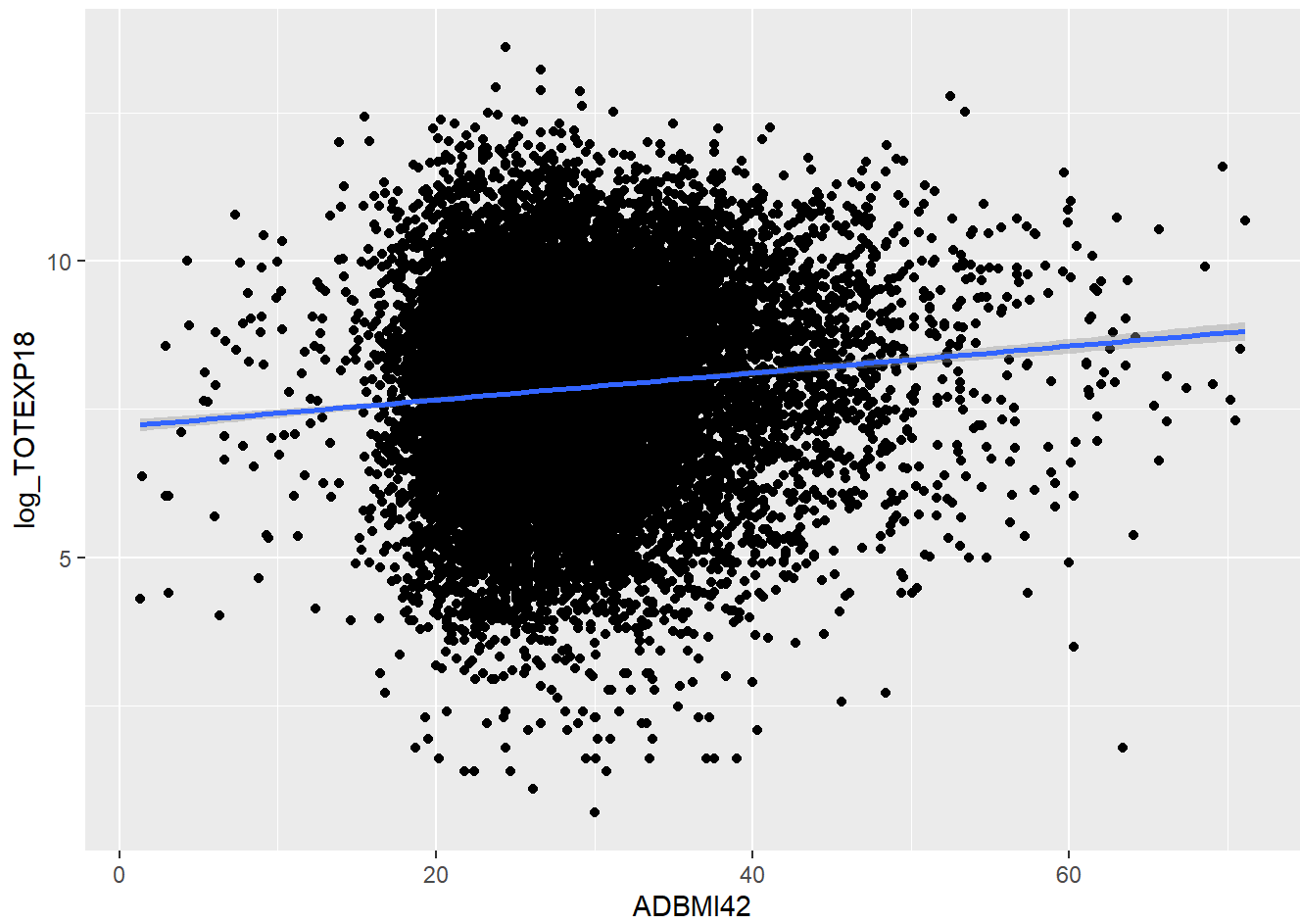


```
for (col in colnames(data)[!colnames(data) %in% c('log_TOTEXP18', 'boxcox_TOTEXP18', 'TOTEXP18', 'F
AMINC18')]) {
  plot = ggplot(data, aes(x = !!sym(col), y = log_TOTEXP18)) + geom_point() + geom_smooth(method
= "lm")
  print(plot)
}
```

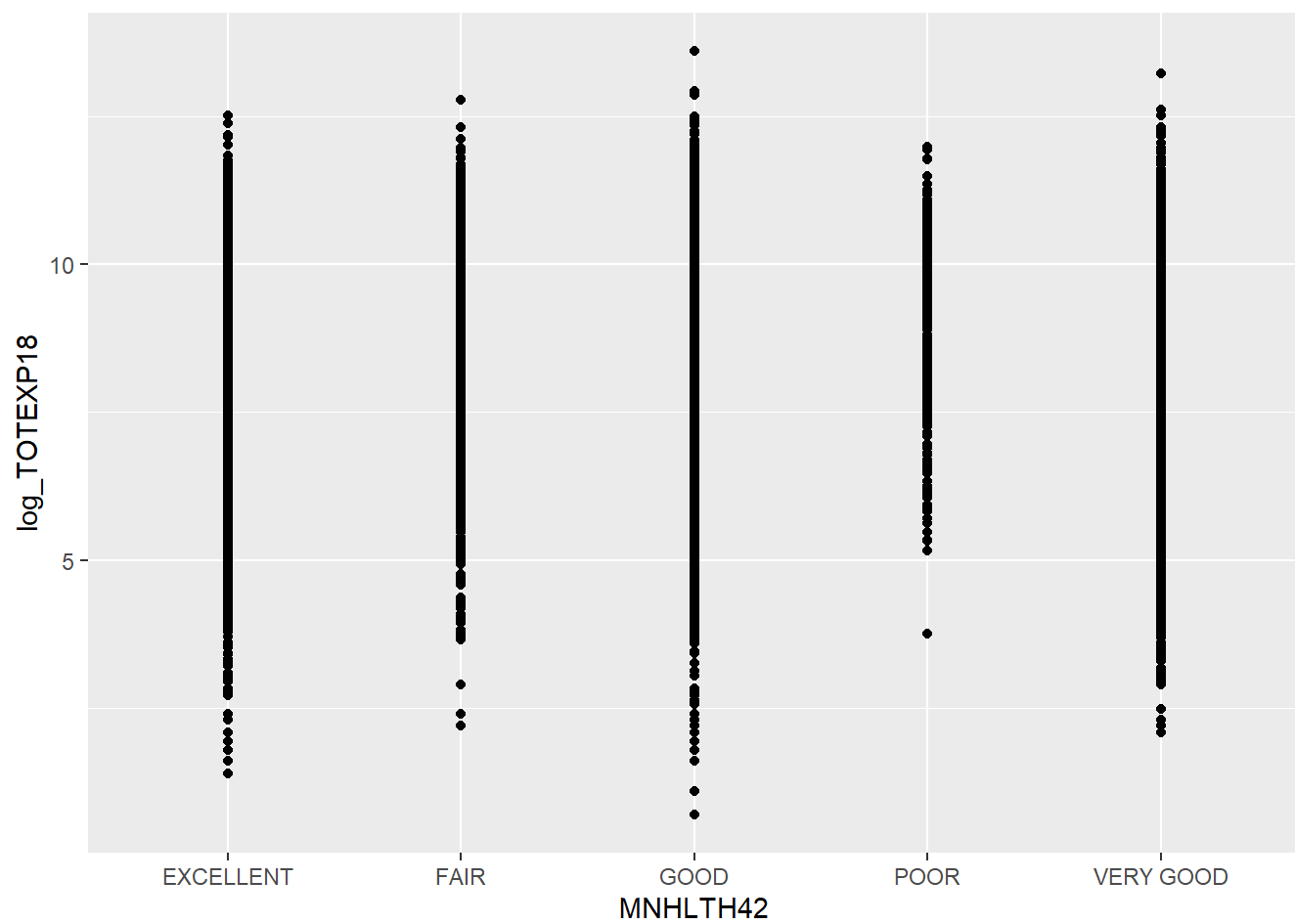
```
## `geom_smooth()` using formula = 'y ~ x'
```



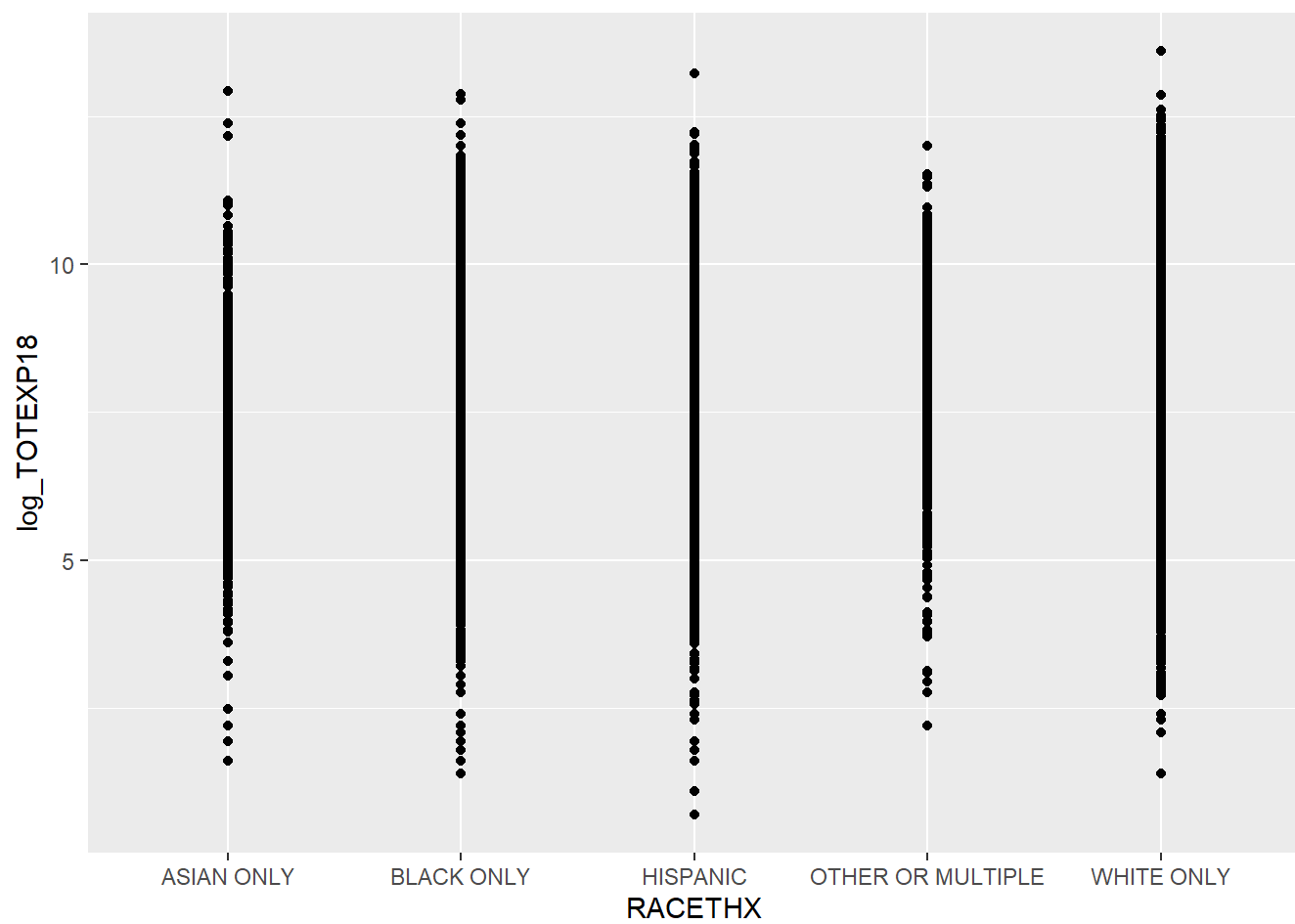
```
## `geom_smooth()` using formula = 'y ~ x'
```



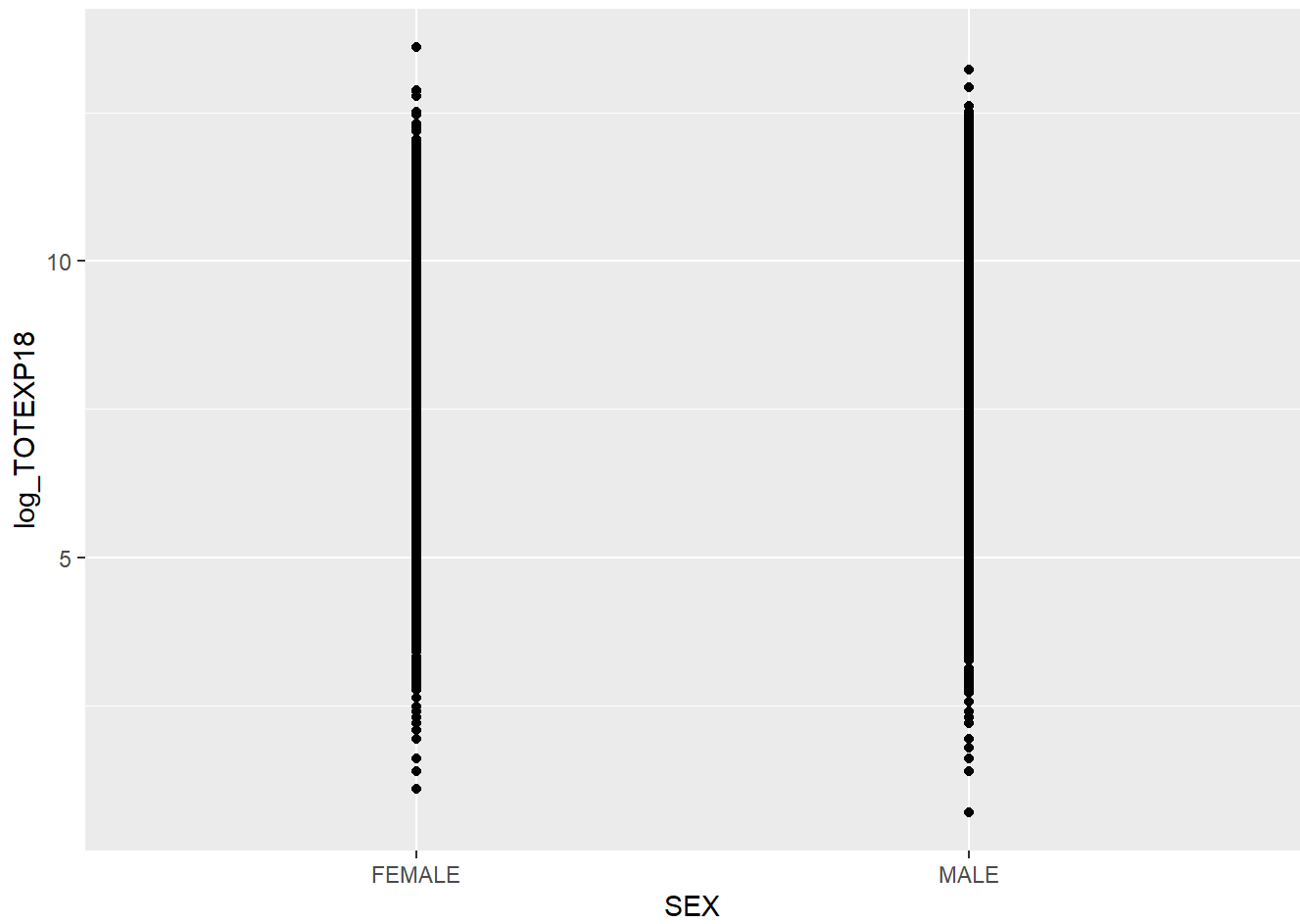
```
## `geom_smooth()` using formula = 'y ~ x'
```



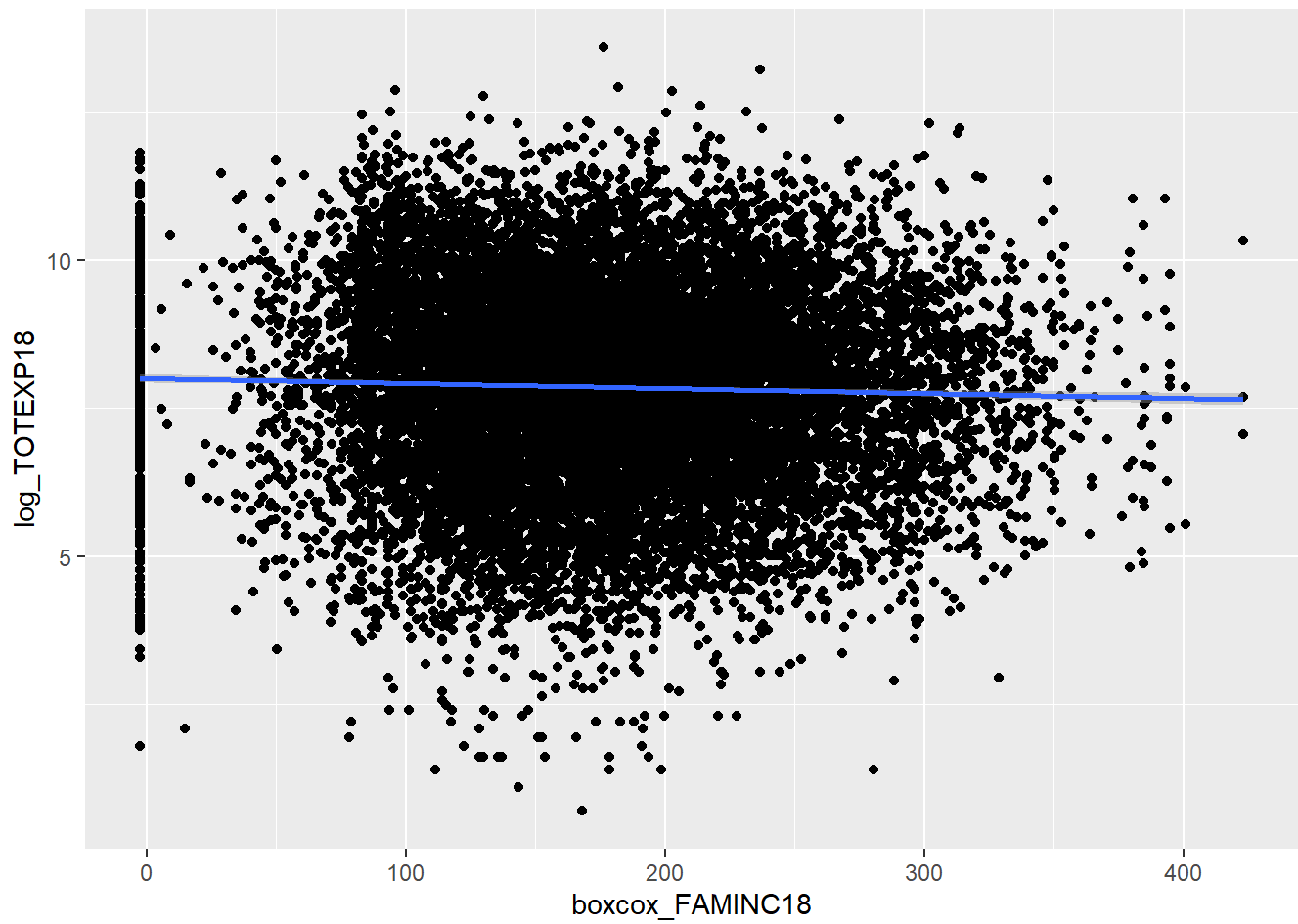
```
## `geom_smooth()` using formula = 'y ~ x'
```



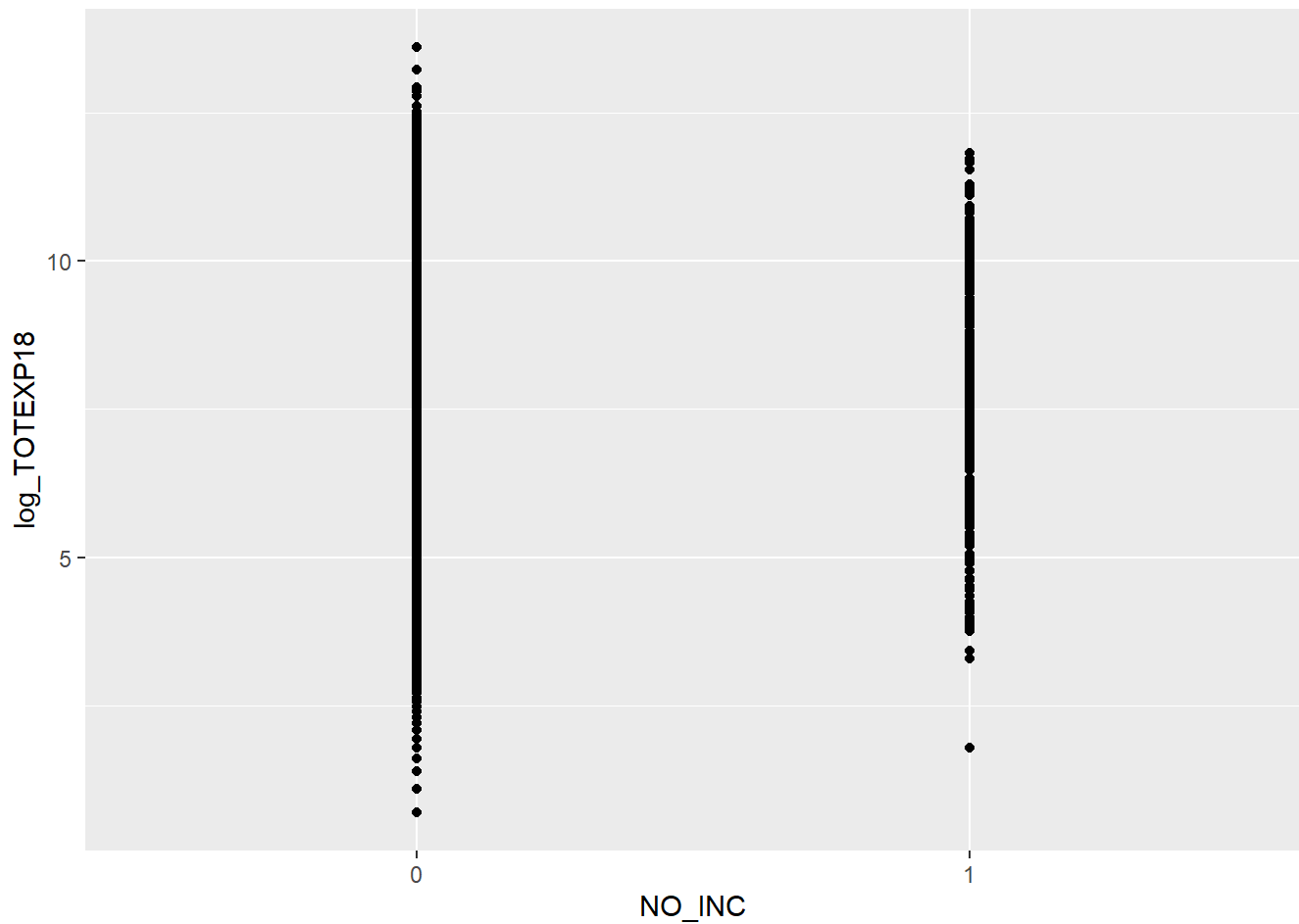
```
## `geom_smooth()` using formula = 'y ~ x'
```



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
cat('\nObservations in original data = ', nrow(data_raw))
```

```
##  
## Observations in original data = 18252
```

```
cat('\nObservations in cleaned data = ', nrow(data))
```

```
##  
## Observations in cleaned data = 15724
```

```
cat('\nObservations in removed = ', nrow(data_raw)-nrow(data))
```

```
##  
## Observations in removed = 2528
```

```
# I will use log transformed total expenditure
```

Time for model building and variable selection. I will create all first order terms, interaction terms, and b-spline transformed BMI to choose from.

```
response = "log_TOTEXP18"
first_order_terms = setdiff(colnames(data), c("log_TOTEXP18", "boxcox_TOTEXP18", "TOTEXP18", "ADB
MI42", "AGE42X", "FAMINC18", "boxcox_FAMINC18"))
interaction_terms = combn(first_order_terms, 2, FUN = function(x) paste(x[1], x[2], sep = ":"))
b_spline_terms = "bs(ADBMI42, degree = 4) + bs(AGE42X, degree = 4) + bs(boxcox_FAMINC18, degree
= 3)"
quadratic_terms = "I(ADBMI42^2)"

# combine
all_terms = c(first_order_terms, interaction_terms, b_spline_terms)

# formula string
# formula_string = paste(all_terms, collapse = " + ")
formula_string <- paste(response, " ~", paste(all_terms, collapse = " + "))
model_formula <- as.formula(formula_string)
print(model_formula)
```

```
## log_TOTEXP18 ~ MNHLTH42 + RACETHX + SEX + NO_INC + MNHLTH42:RACETHX +
##      MNHLTH42:SEX + MNHLTH42:NO_INC + RACETHX:SEX + RACETHX:NO_INC +
##      SEX:NO_INC + bs(ADBMI42, degree = 4) + bs(AGE42X, degree = 4) +
##      bs(boxcox_FAMINC18, degree = 3)
```

Full model

```
full_model = lm(model_formula, data = data)
summary(full_model)
```

```
##
## Call:
## lm(formula = model_formula, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.6441 -0.9752  0.0374  1.0128  6.2243
##
## Coefficients:
##                                Estimate Std. Error t value
## (Intercept)                   7.424503   0.449424  16.520
## MNHLTH42FAIR                  1.106175   0.270893   4.083
## MNHLTH42GOOD                   0.106991   0.149790   0.714
## MNHLTH42POOR                   1.448945   0.506252   2.862
## MNHLTH42VERY GOOD              0.083000   0.136102   0.610
## RACETHXBLACK ONLY              0.194320   0.126138   1.541
## RACETHXHISPANIC                0.151690   0.123454   1.229
## RACETHXOTHER OR MULTIPLE       0.749220   0.181473   4.129
## RACETHXWHITE ONLY              0.590796   0.112044   5.273
## SEXMALE                       -0.108379   0.117448  -0.923
## NO_INC1                       -1.528080   0.502635  -3.040
## bs(ADBMI42, degree = 4)1       -1.385167   0.920936  -1.504
## bs(ADBMI42, degree = 4)2       -0.096251   0.465587  -0.207
## bs(ADBMI42, degree = 4)3        0.547807   0.921680   0.594
## bs(ADBMI42, degree = 4)4        0.158928   0.513603   0.309
## bs(AGE42X, degree = 4)1         0.531884   0.203791   2.610
## bs(AGE42X, degree = 4)2       -0.001142   0.184626  -0.006
## bs(AGE42X, degree = 4)3        2.173903   0.178098  12.206
## bs(AGE42X, degree = 4)4        1.769197   0.079462  22.265
## bs(boxcox_FAMINC18, degree = 3)1 -1.556297   0.409934  -3.796
## bs(boxcox_FAMINC18, degree = 3)2  0.258861   0.182317   1.420
## bs(boxcox_FAMINC18, degree = 3)3 -0.476589   0.371924  -1.281
## MNHLTH42FAIR:RACETHXBLACK ONLY -0.212087   0.292241  -0.726
## MNHLTH42GOOD:RACETHXBLACK ONLY  0.326341   0.168951   1.932
## MNHLTH42POOR:RACETHXBLACK ONLY -0.096960   0.561638  -0.173
## MNHLTH42VERY GOOD:RACETHXBLACK ONLY 0.145177   0.158738   0.915
## MNHLTH42FAIR:RACETHXHISPANIC -0.234372   0.288709  -0.812
## MNHLTH42GOOD:RACETHXHISPANIC  0.133100   0.165514   0.804
## MNHLTH42POOR:RACETHXHISPANIC  0.234972   0.539383   0.436
## MNHLTH42VERY GOOD:RACETHXHISPANIC -0.020070   0.155023  -0.129
## MNHLTH42FAIR:RACETHXOTHER OR MULTIPLE -0.705688   0.361502  -1.952
## MNHLTH42GOOD:RACETHXOTHER OR MULTIPLE 0.150082   0.234402   0.640
## MNHLTH42POOR:RACETHXOTHER OR MULTIPLE -0.331102   0.766448  -0.432
## MNHLTH42VERY GOOD:RACETHXOTHER OR MULTIPLE 0.067125   0.227216   0.295
## MNHLTH42FAIR:RACETHXWHITE ONLY -0.297397   0.273400  -1.088
## MNHLTH42GOOD:RACETHXWHITE ONLY  0.297005   0.152649   1.946
## MNHLTH42POOR:RACETHXWHITE ONLY -0.092467   0.505602  -0.183
## MNHLTH42VERY GOOD:RACETHXWHITE ONLY 0.008667   0.138745   0.062
## MNHLTH42FAIR:SEXMALE          0.177868   0.096213   1.849
## MNHLTH42GOOD:SEXMALE          0.029688   0.064592   0.460
## MNHLTH42POOR:SEXMALE          -0.184738   0.190752  -0.968
## MNHLTH42VERY GOOD:SEXMALE      0.022143   0.061486   0.360
```

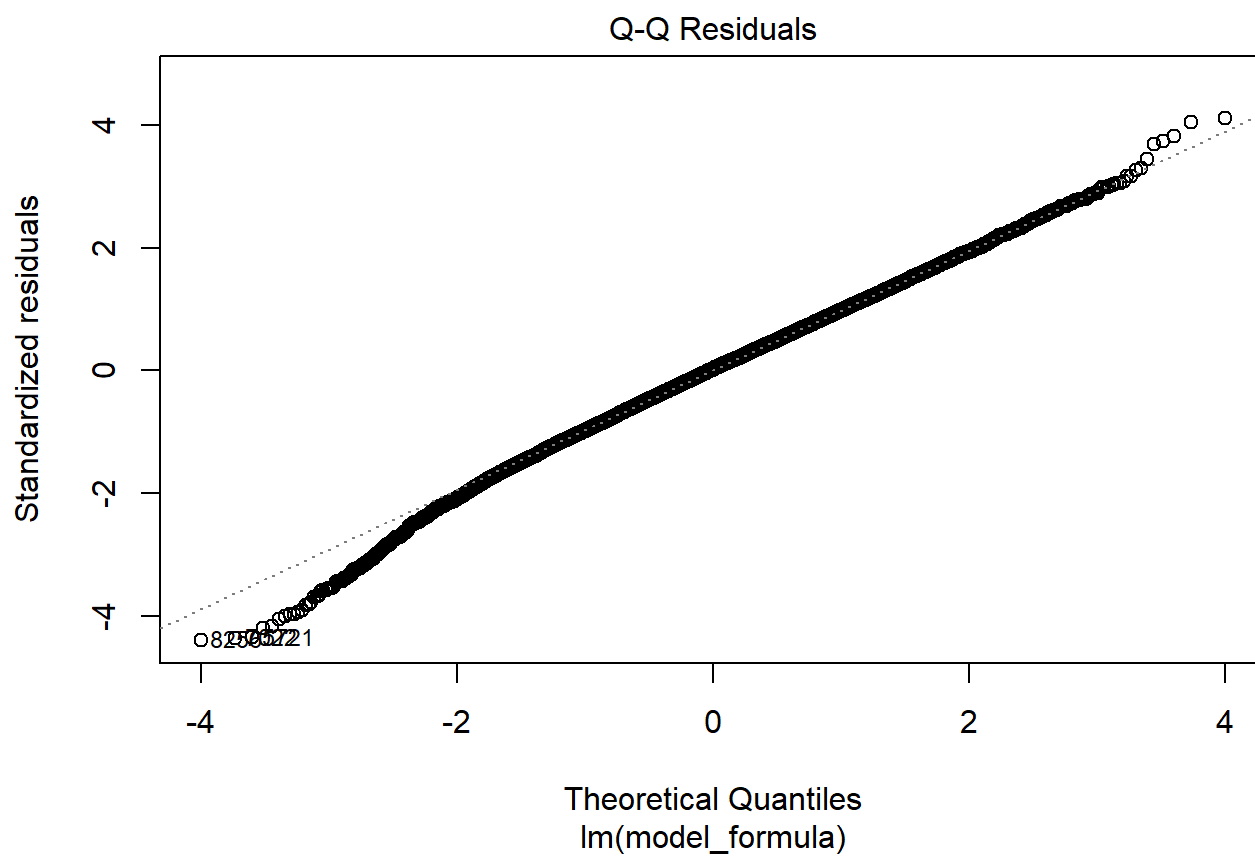
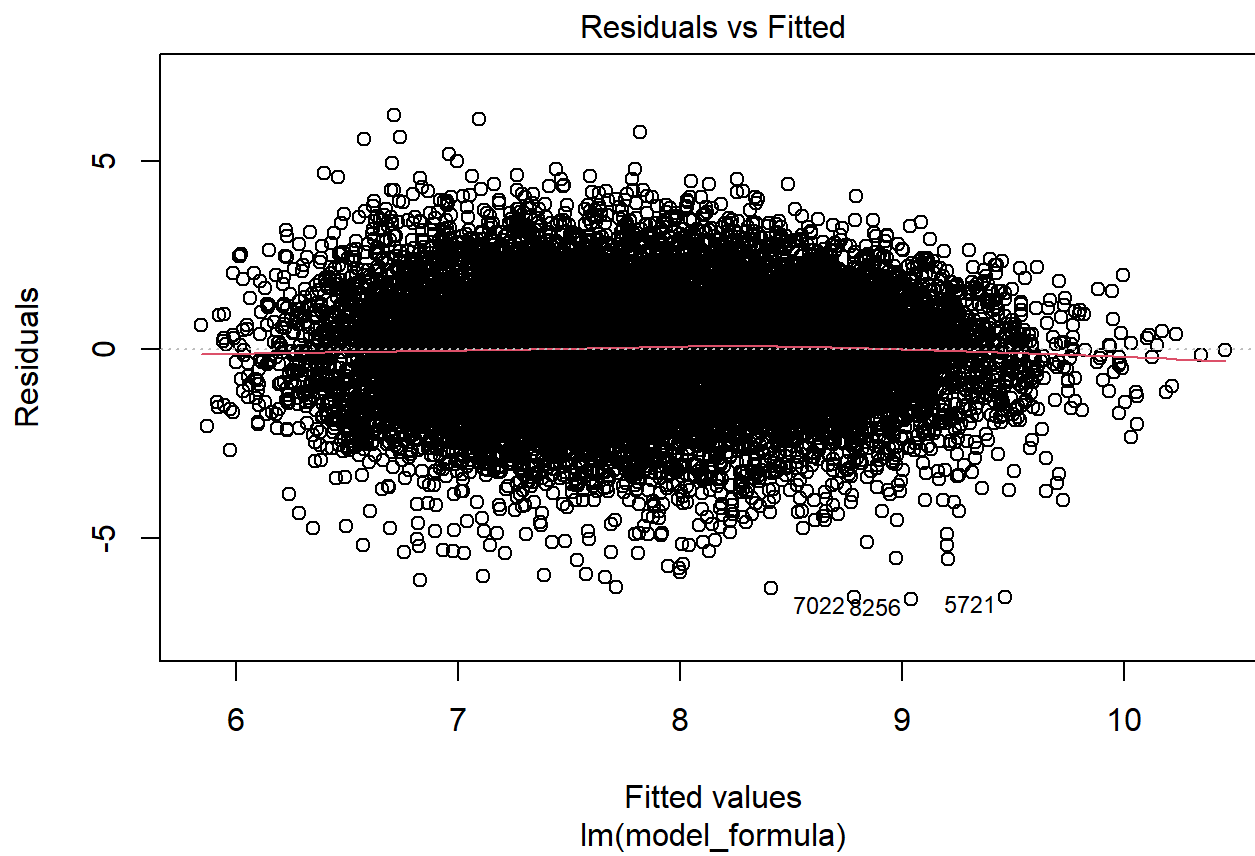
## MNHLTH42FAIR:NO_INC1	0.065985	0.268993	0.245
## MNHLTH42GOOD:NO_INC1	0.070205	0.230382	0.305
## MNHLTH42POOR:NO_INC1	-0.169123	0.414535	-0.408
## MNHLTH42VERY GOOD:NO_INC1	0.305629	0.252060	1.213
## RACETHXBLACK ONLY:SEXMALE	-0.180018	0.131504	-1.369
## RACETHXHISPANIC:SEXMALE	-0.337166	0.128102	-2.632
## RACETHXOTHER OR MULTIPLE:SEXMALE	-0.549151	0.179272	-3.063
## RACETHXWHITE ONLY:SEXMALE	-0.225013	0.116643	-1.929
## RACETHXBLACK ONLY:NO_INC1	0.783551	0.457895	1.711
## RACETHXHISPANIC:NO_INC1	0.443921	0.462084	0.961
## RACETHXOTHER OR MULTIPLE:NO_INC1	0.168311	0.650042	0.259
## RACETHXWHITE ONLY:NO_INC1	1.005707	0.451101	2.229
## SEXMALE:NO_INC1	0.120962	0.179272	0.675
##	Pr(> t)		
## (Intercept)	< 2e-16	***	
## MNHLTH42FAIR	4.46e-05	***	
## MNHLTH42GOOD	0.475068		
## MNHLTH42POOR	0.004214	**	
## MNHLTH42VERY GOOD	0.541979		
## RACETHXBLACK ONLY	0.123451		
## RACETHXHISPANIC	0.219197		
## RACETHXOTHER OR MULTIPLE	3.67e-05	***	
## RACETHXWHITE ONLY	1.36e-07	***	
## SEXMALE	0.356135		
## NO_INC1	0.002369	**	
## bs(ADBMI42, degree = 4)1	0.132579		
## bs(ADBMI42, degree = 4)2	0.836224		
## bs(ADBMI42, degree = 4)3	0.552282		
## bs(ADBMI42, degree = 4)4	0.756994		
## bs(AGE42X, degree = 4)1	0.009064	**	
## bs(AGE42X, degree = 4)2	0.995063		
## bs(AGE42X, degree = 4)3	< 2e-16	***	
## bs(AGE42X, degree = 4)4	< 2e-16	***	
## bs(boxcox_FAMINC18, degree = 3)1	0.000147	***	
## bs(boxcox_FAMINC18, degree = 3)2	0.155674		
## bs(boxcox_FAMINC18, degree = 3)3	0.200067		
## MNHLTH42FAIR:RACETHXBLACK ONLY	0.468017		
## MNHLTH42GOOD:RACETHXBLACK ONLY	0.053430	.	
## MNHLTH42POOR:RACETHXBLACK ONLY	0.862938		
## MNHLTH42VERY GOOD:RACETHXBLACK ONLY	0.360433		
## MNHLTH42FAIR:RACETHXHISPANIC	0.416924		
## MNHLTH42GOOD:RACETHXHISPANIC	0.421317		
## MNHLTH42POOR:RACETHXHISPANIC	0.663111		
## MNHLTH42VERY GOOD:RACETHXHISPANIC	0.896994		
## MNHLTH42FAIR:RACETHXOTHER OR MULTIPLE	0.050944	.	
## MNHLTH42GOOD:RACETHXOTHER OR MULTIPLE	0.522004		
## MNHLTH42POOR:RACETHXOTHER OR MULTIPLE	0.665751		
## MNHLTH42VERY GOOD:RACETHXOTHER OR MULTIPLE	0.767674		
## MNHLTH42FAIR:RACETHXWHITE ONLY	0.276713		
## MNHLTH42GOOD:RACETHXWHITE ONLY	0.051712	.	
## MNHLTH42POOR:RACETHXWHITE ONLY	0.854890		
## MNHLTH42VERY GOOD:RACETHXWHITE ONLY	0.950194		

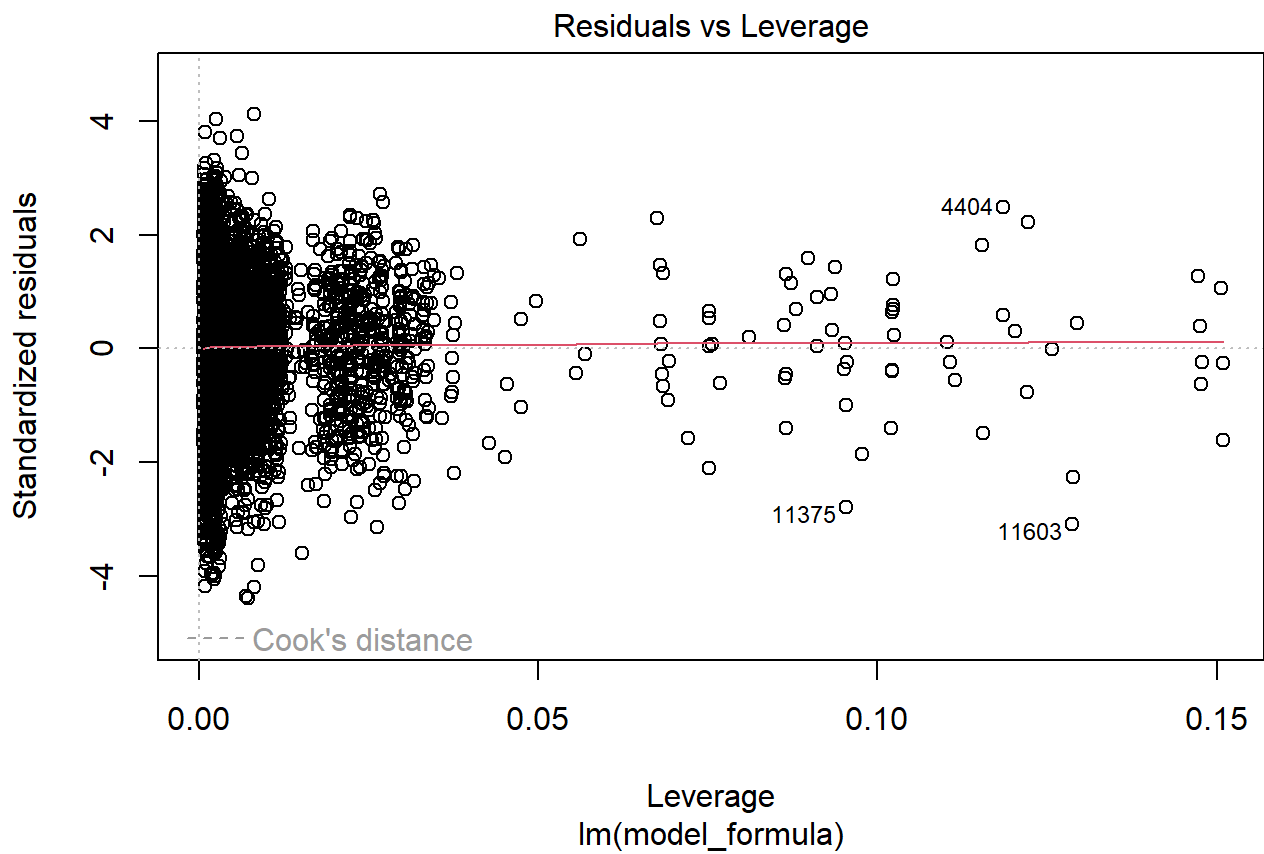
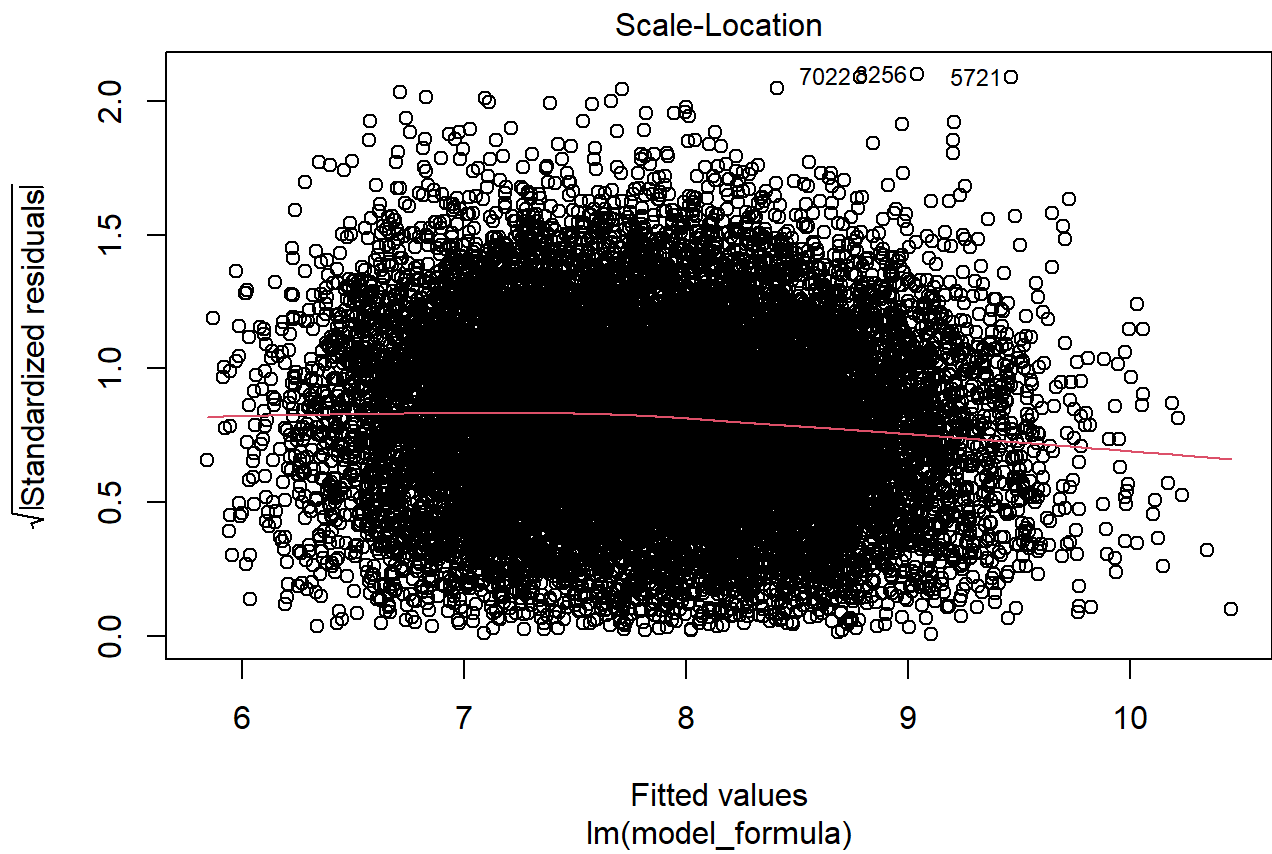
```

## MNHLTH42FAIR:SEXMALE          0.064522 .
## MNHLTH42GOOD:SEXMALE          0.645793
## MNHLTH42POOR:SEXMALE          0.332825
## MNHLTH42VERY GOOD:SEXMALE     0.718756
## MNHLTH42FAIR:NO_INC1          0.806224
## MNHLTH42GOOD:NO_INC1          0.760575
## MNHLTH42POOR:NO_INC1          0.683293
## MNHLTH42VERY GOOD:NO_INC1     0.225329
## RACETHXBLACK ONLY:SEXMALE     0.171043
## RACETHXHISPANIC:SEXMALE       0.008496 **
## RACETHXOTHER OR MULTIPLE:SEXMALE 0.002193 **
## RACETHXWHITE ONLY:SEXMALE     0.053740 .
## RACETHXBLACK ONLY:NO_INC1     0.087063 .
## RACETHXHISPANIC:NO_INC1       0.336722
## RACETHXOTHER OR MULTIPLE:NO_INC1 0.795697
## RACETHXWHITE ONLY:NO_INC1     0.025798 *
## SEXMALE:NO_INC1              0.499852
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.515 on 15669 degrees of freedom
## Multiple R-squared:  0.196, Adjusted R-squared:  0.1933
## F-statistic: 70.75 on 54 and 15669 DF, p-value: < 2.2e-16

```

```
plot(full_model)
```





Stepwise variable selection (AIC)

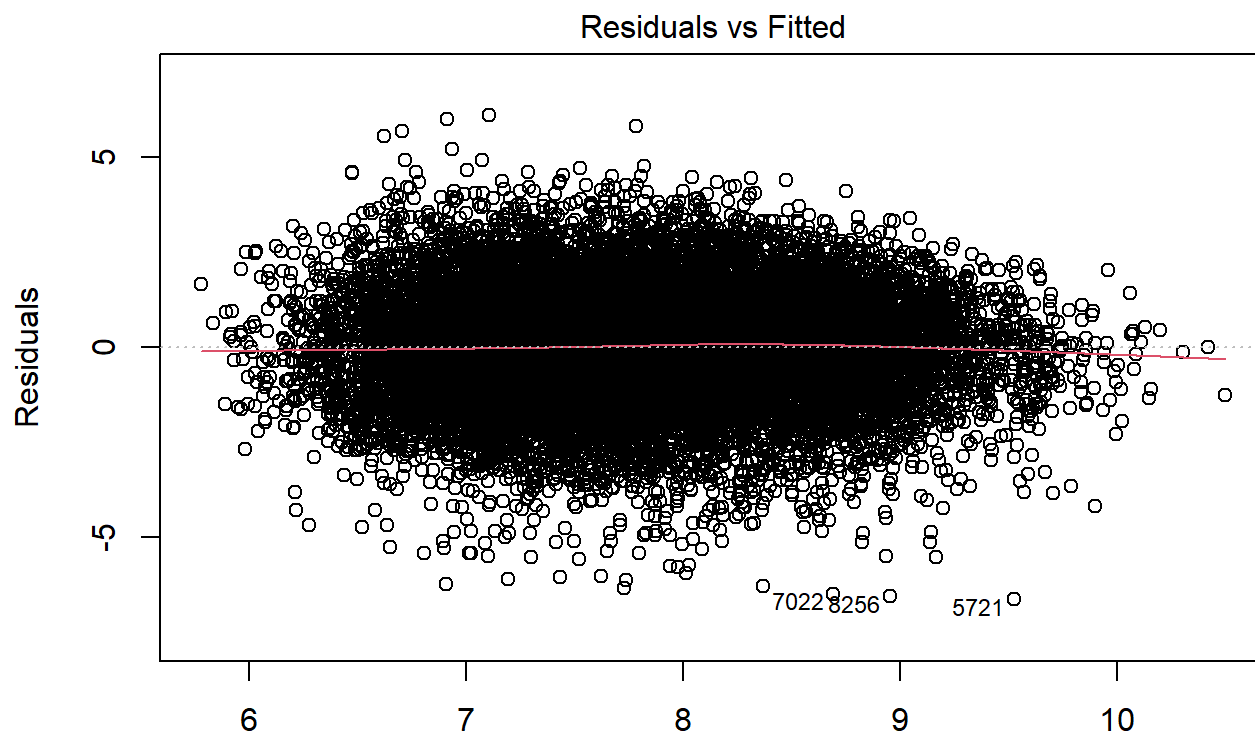
```
null_model = lm(log_TOTEXP18~1, data=data)
stepwise = step(null_model, scope = list(lower = null_model, upper = full_model), direction = "both", k = 2, trace = FALSE, test = "F", steps = 1000, add = 0.05 , drop = 0.05)
summary(stepwise)
```

```
##
## Call:
## lm(formula = log_TOTEXP18 ~ bs(AGE42X, degree = 4) + MNHLTH42 +
##     RACETHX + SEX + bs(ADBMI42, degree = 4) + bs(boxcox_FAMINC18,
##     degree = 3) + NO_INC + RACETHX:SEX + RACETHX:NO_INC, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.6353 -0.9784  0.0346  1.0071  6.1133
##
## Coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.36057    0.44328  16.605 < 2e-16 ***
## bs(AGE42X, degree = 4)1      0.54598    0.20372   2.680 0.007368 **
## bs(AGE42X, degree = 4)2     -0.01287    0.18451  -0.070 0.944405
## bs(AGE42X, degree = 4)3      2.18346    0.17788  12.275 < 2e-16 ***
## bs(AGE42X, degree = 4)4      1.77610    0.07939  22.373 < 2e-16 ***
## MNHLTH42FAIR      0.90514    0.04875  18.567 < 2e-16 ***
## MNHLTH42GOOD      0.37487    0.03267  11.476 < 2e-16 ***
## MNHLTH42POOR      1.32665    0.09390  14.129 < 2e-16 ***
## MNHLTH42VERY GOOD    0.12009    0.03062   3.922 8.83e-05 ***
## RACETHXBLACK ONLY    0.31155    0.08731   3.568 0.000361 ***
## RACETHXHISPANIC     0.16812    0.08491   1.980 0.047738 *
## RACETHXOTHER OR MULTIPLE 0.73528    0.11749   6.258 4.00e-10 ***
## RACETHXWHITE ONLY    0.65255    0.07784   8.384 < 2e-16 ***
## SEXMALE           -0.07344    0.11198  -0.656 0.511972
## bs(ADBMI42, degree = 4)1    -1.41210    0.92067  -1.534 0.125103
## bs(ADBMI42, degree = 4)2    -0.09423    0.46521  -0.203 0.839489
## bs(ADBMI42, degree = 4)3     0.50761    0.92138   0.551 0.581692
## bs(ADBMI42, degree = 4)4     0.16976    0.51330   0.331 0.740863
## bs(boxcox_FAMINC18, degree = 3)1 -1.55115    0.40921  -3.791 0.000151 ***
## bs(boxcox_FAMINC18, degree = 3)2  0.26138    0.18219   1.435 0.151409
## bs(boxcox_FAMINC18, degree = 3)3 -0.47465    0.37160  -1.277 0.201509
## NO_INC1           -1.36304    0.45830  -2.974 0.002943 **
## RACETHXBLACK ONLY:SEXMALE   -0.18516    0.13078  -1.416 0.156832
## RACETHXHISPANIC:SEXMALE    -0.34240    0.12747  -2.686 0.007238 **
## RACETHXOTHER OR MULTIPLE:SEXMALE -0.55270    0.17861  -3.095 0.001975 **
## RACETHXWHITE ONLY:SEXMALE  -0.23362    0.11609  -2.012 0.044186 *
## RACETHXBLACK ONLY:NO_INC1    0.75310    0.45230   1.665 0.095927 .
## RACETHXHISPANIC:NO_INC1     0.41835    0.45409   0.921 0.356916
## RACETHXOTHER OR MULTIPLE:NO_INC1 0.15877    0.64410   0.247 0.805296
## RACETHXWHITE ONLY:NO_INC1    0.99382    0.44541   2.231 0.025678 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.515 on 15694 degrees of freedom
## Multiple R-squared:  0.1946, Adjusted R-squared:  0.1931
## F-statistic: 130.7 on 29 and 15694 DF, p-value: < 2.2e-16
```

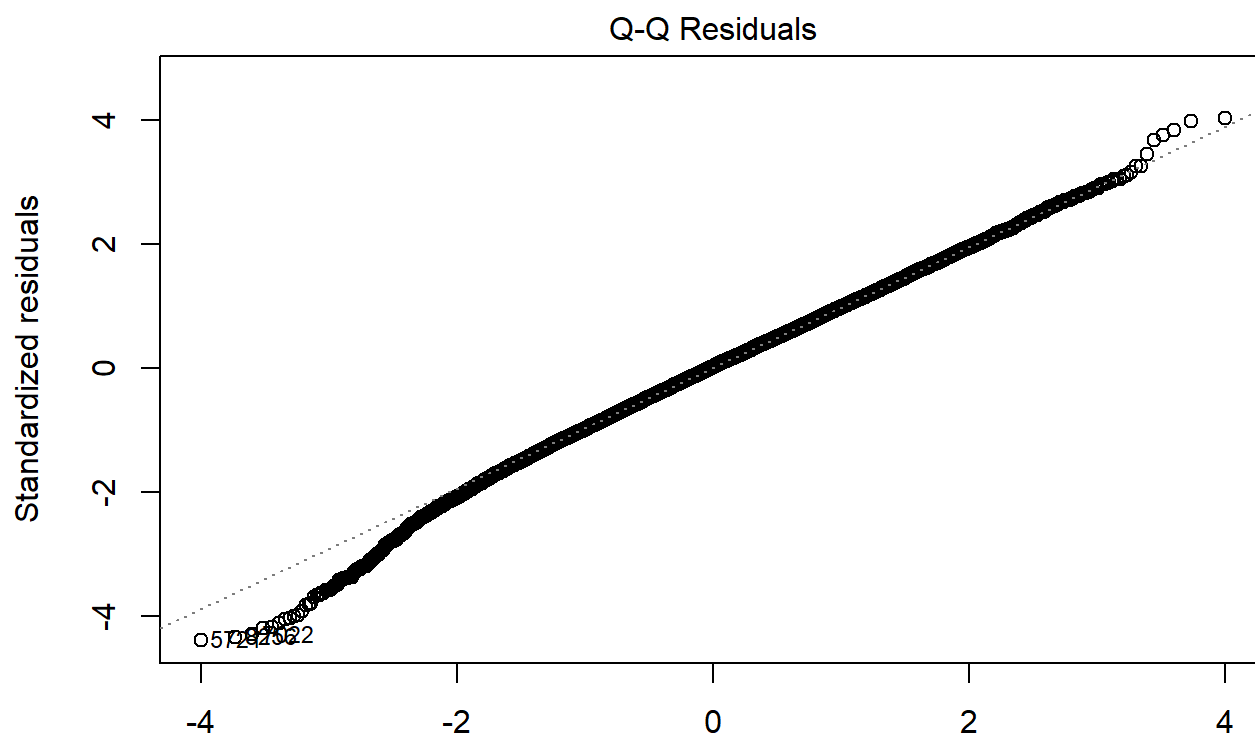
```
cat('\nNumber of terms = ',length(coef(stepwise))-1)
```

```
##  
## Number of terms = 29
```

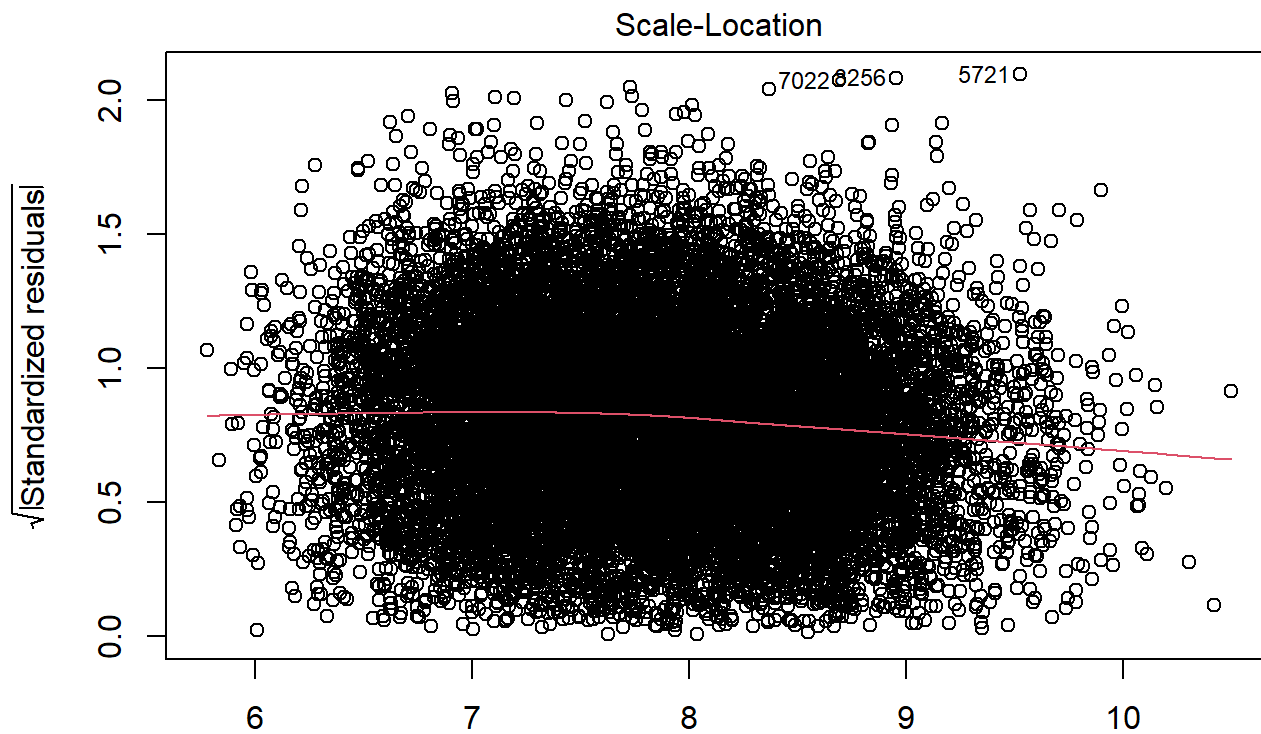
```
plot(stepwise)
```



Fitted values
 $\text{lm}(\log_TOTEXP18 \sim \text{bs}(\text{AGE42X}, \text{degree} = 4) + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{bs}(\text{AD} .$

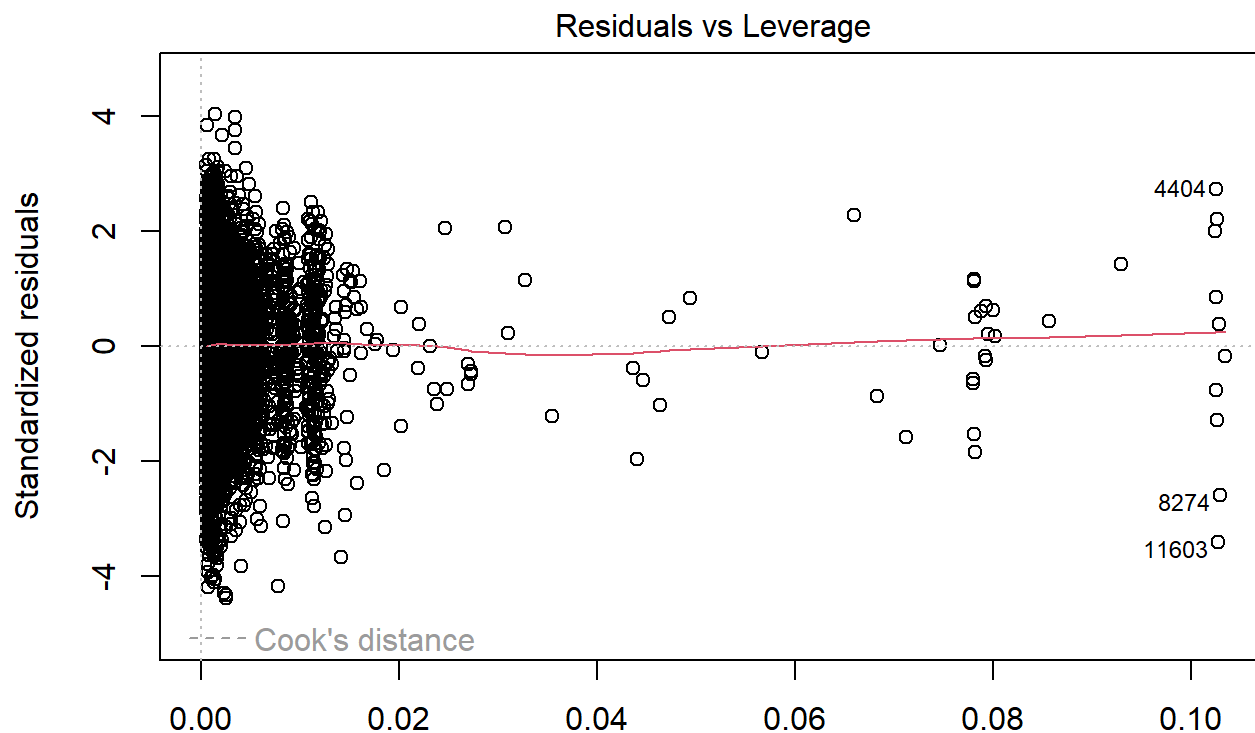


Theoretical Quantiles
 $\text{lm}(\log_TOTEXP18 \sim \text{bs}(\text{AGE42X}, \text{degree} = 4) + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{bs}(\text{AD} .$



Fitted values

$\text{lm}(\log_TOTEXP18 \sim \text{bs}(\text{AGE42X}, \text{degree} = 4) + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{bs}(\text{AD} .$



Leverage

$\text{lm}(\log_TOTEXP18 \sim \text{bs}(\text{AGE42X}, \text{degree} = 4) + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{bs}(\text{AD} .$

Lasso variable selection

```
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 4.4.2
```

```
## Loading required package: Matrix
```

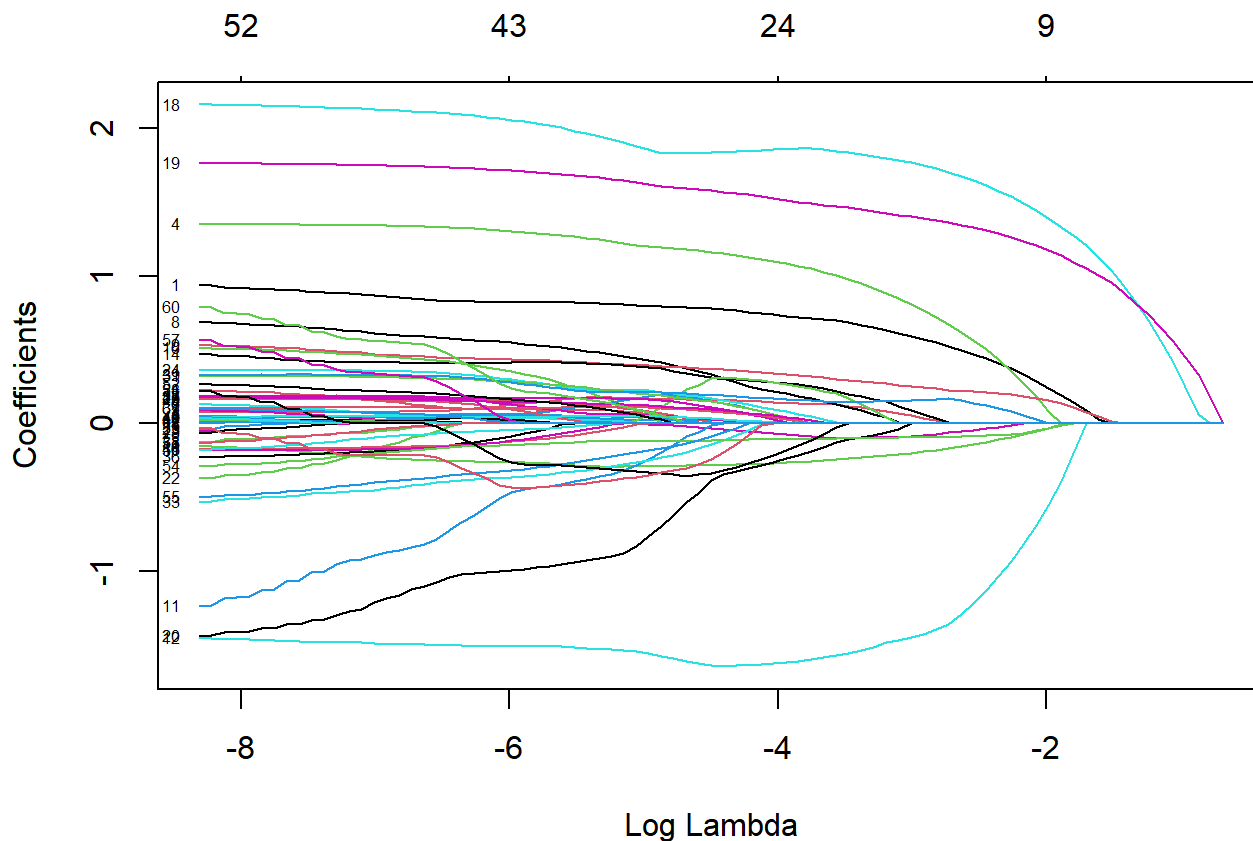
```
## Loaded glmnet 4.1-8
```

```
X = model.matrix(model_formula, data = data)[, -1]
```

```
# Lasso model
```

```
lasso = glmnet(x = X, y = data$log_TOTEXP18, alpha = 1)
```

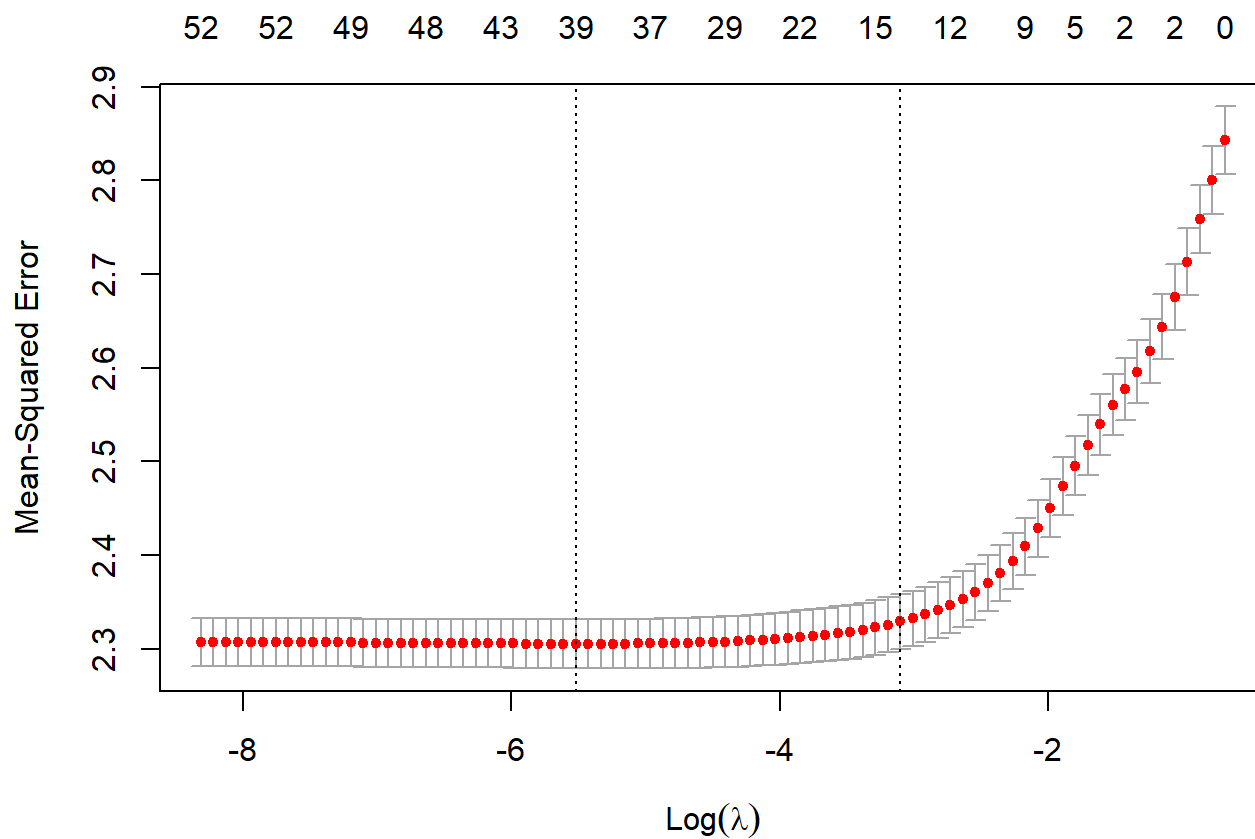
```
plot(lasso, xvar = "lambda", label = TRUE)
```



```
# cv to find optimal lambda
```

```
cv_lasso = cv.glmnet(x = X, y = data$log_TOTEXP18, alpha = 1, nfolds = 10)
```

```
plot(cv_lasso)
```



```
best_lambda = cv_lasso$lambda.min
cat("Best lambda from cross-validation:", best_lambda, "\n")
```

```
## Best lambda from cross-validation: 0.00400272
```

```
# use optimal lambda to fit best model
best_lasso = glmnet(x = X, y = data$log_TOTEXP18, alpha = 1, lambda = best_lambda)
coefficients_best_lasso = coef(best_lasso)
print(coefficients_best_lasso)
```

```

## 62 x 1 sparse Matrix of class "dgCMatrix"
##
## (Intercept) 7.5194231528
## MNHLTH42FAIR 0.8197911964
## MNHLTH42GOOD 0.1725829549
## MNHLTH42INVALID .
## MNHLTH42POOR 1.2667233038
## MNHLTH42VERY GOOD 0.0403719872
## RACETHXBLACK ONLY 0.0307027836
## RACETHXHISPANIC .
## RACETHXOTHER OR MULTIPLE 0.5068335078
## RACETHXWHITE ONLY 0.4199159745
## SEXMALE -0.2765963034
## NO_INC1 -0.3921935415
## bs(ADBMI42, degree = 4)1 -1.6029960736
## bs(ADBMI42, degree = 4)2 .
## bs(ADBMI42, degree = 4)3 0.3460983496
## bs(ADBMI42, degree = 4)4 0.0993183580
## bs(AGE42X, degree = 4)1 0.2552658983
## bs(AGE42X, degree = 4)2 0.1014457386
## bs(AGE42X, degree = 4)3 1.9942629548
## bs(AGE42X, degree = 4)4 1.6837310491
## bs(boxcox_FAMINC18, degree = 3)1 -0.9418546330
## bs(boxcox_FAMINC18, degree = 3)2 0.0912136047
## bs(boxcox_FAMINC18, degree = 3)3 .
## MNHLTH42FAIR:RACETHXBLACK ONLY 0.0507873675
## MNHLTH42GOOD:RACETHXBLACK ONLY 0.2409113824
## MNHLTH42INVALID:RACETHXBLACK ONLY .
## MNHLTH42POOR:RACETHXBLACK ONLY .
## MNHLTH42VERY GOOD:RACETHXBLACK ONLY 0.1684019325
## MNHLTH42FAIR:RACETHXHISPANIC 0.0005834217
## MNHLTH42GOOD:RACETHXHISPANIC 0.0239758147
## MNHLTH42INVALID:RACETHXHISPANIC .
## MNHLTH42POOR:RACETHXHISPANIC 0.2544373459
## MNHLTH42VERY GOOD:RACETHXHISPANIC .
## MNHLTH42FAIR:RACETHXOTHER OR MULTIPLE -0.3250564148
## MNHLTH42GOOD:RACETHXOTHER OR MULTIPLE 0.0752525446
## MNHLTH42INVALID:RACETHXOTHER OR MULTIPLE .
## MNHLTH42POOR:RACETHXOTHER OR MULTIPLE .
## MNHLTH42VERY GOOD:RACETHXOTHER OR MULTIPLE 0.1072200695
## MNHLTH42FAIR:RACETHXWHITE ONLY .
## MNHLTH42GOOD:RACETHXWHITE ONLY 0.2264406047
## MNHLTH42INVALID:RACETHXWHITE ONLY .
## MNHLTH42POOR:RACETHXWHITE ONLY .
## MNHLTH42VERY GOOD:RACETHXWHITE ONLY 0.0482764569
## MNHLTH42FAIR:SEXMALE 0.1380564852
## MNHLTH42GOOD:SEXMALE .
## MNHLTH42INVALID:SEXMALE .
## MNHLTH42POOR:SEXMALE -0.0764411182
## MNHLTH42VERY GOOD:SEXMALE .
## MNHLTH42FAIR:NO_INC1 .
## MNHLTH42GOOD:NO_INC1 .

```



```
## MNHLTH42INVALID:NO_INC1 .
## MNHLTH42POOR:NO_INC1 -0.0543075334
## MNHLTH42VERY GOOD:NO_INC1 0.1269345588
## RACETHXBLACK ONLY:SEXMALE .
## RACETHXHISPANIC:SEXMALE -0.1282326266
## RACETHXOTHER OR MULTIPLE:SEXMALE -0.2657238557
## RACETHXWHITE ONLY:SEXMALE -0.0220612748
## RACETHXBLACK ONLY:NO_INC1 .
## RACETHXHISPANIC:NO_INC1 -0.2906400232
## RACETHXOTHER OR MULTIPLE:NO_INC1 -0.4104835460
## RACETHXWHITE ONLY:NO_INC1 0.1777055987
## SEXMALE:NO_INC1 .
```

```
summary(best_lasso)
```

```
##          Length Class      Mode
## a0          1    -none-   numeric
## beta        61    dgMatrix S4
## df           1    -none-   numeric
## dim          2    -none-   numeric
## lambda       1    -none-   numeric
## dev.ratio    1    -none-   numeric
## nulldev      1    -none-   numeric
## npasses      1    -none-   numeric
## jerr         1    -none-   numeric
## offset       1    -none-   logical
## call         5    -none-   call
## nobs         1    -none-   numeric
```

```
# num non zero coeff
non_zero_coefs = sum(coefficients_best_lasso != 0)
cat("\nNumber of non-zero coefficients:", non_zero_coefs, "\n")
```

```
##
## Number of non-zero coefficients: 40
```

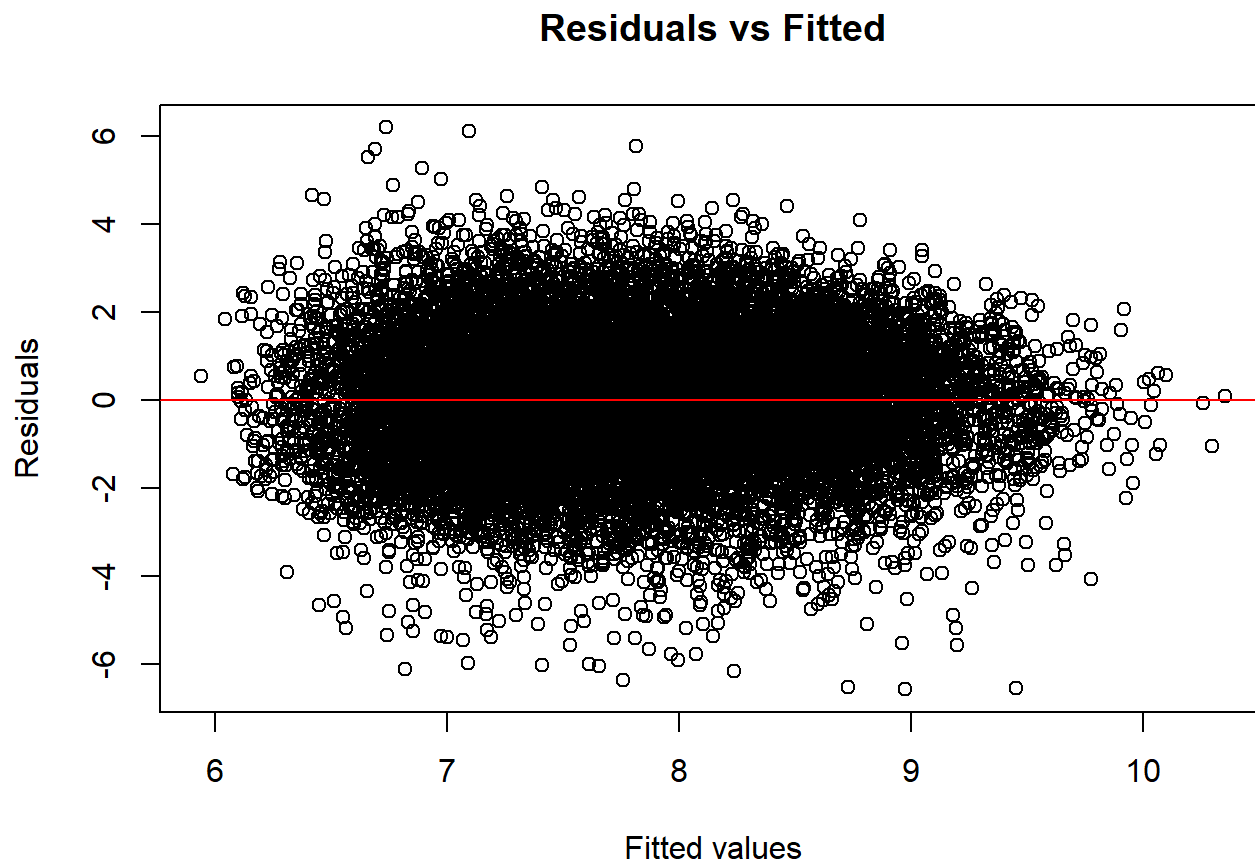
```
# DIAGNOSTICS:
# Predict using the best Lasso model
y_pred = predict(best_lasso, X)

# Compute residuals
residuals = data$log_TOTEXP18 - y_pred

# Compute R-squared (1 - RSS/TSS)
rss = sum(residuals^2)
tss = sum((data$log_TOTEXP18 - mean(data$log_TOTEXP18))^2)
r_squared = 1 - rss / tss
cat("\nR-squared:", r_squared, "\n")
```

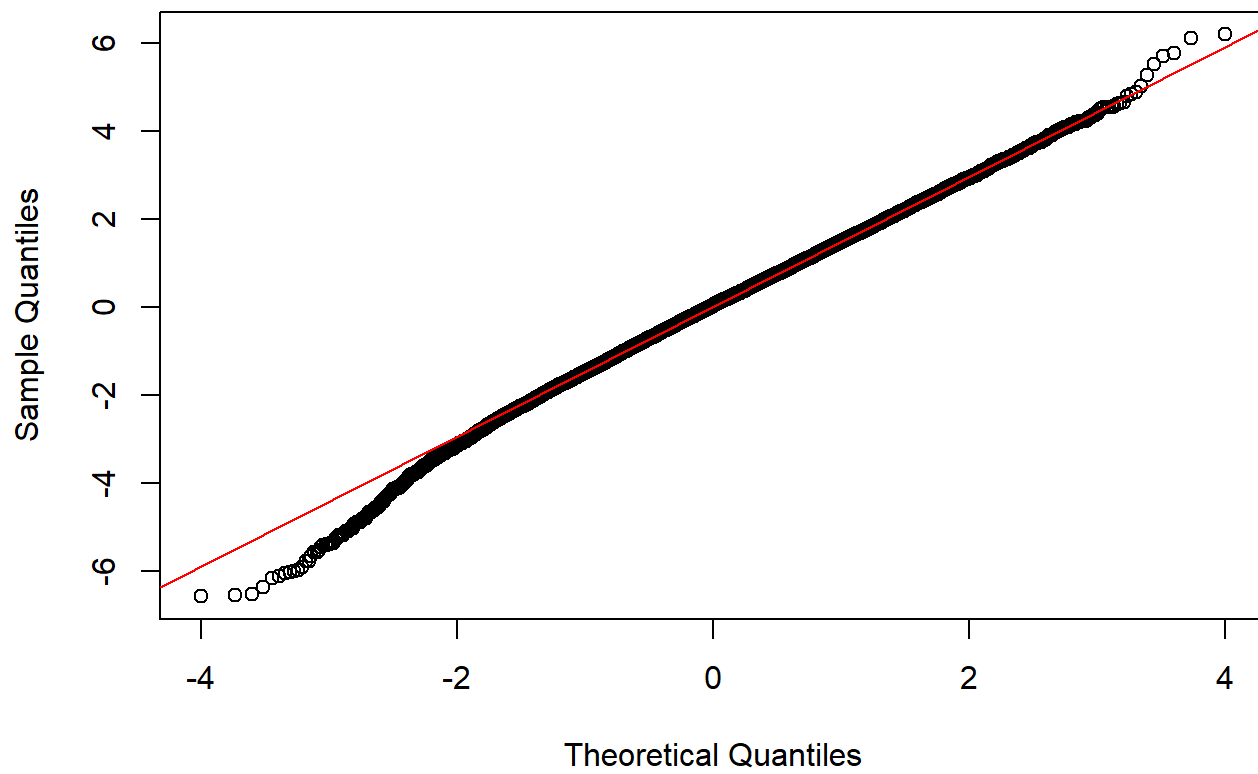
```
##  
## R-squared: 0.1950217
```

```
# Plot 1: Residuals vs Fitted Values  
plot(y_pred, residuals, xlab = "Fitted values", ylab = "Residuals", main = "Residuals vs Fitted")  
abline(h = 0, col = "red")
```



```
# Plot 2: Q-Q plot of residuals  
qqnorm(residuals, main = "Q-Q Plot")  
qqline(residuals, col = "red")
```

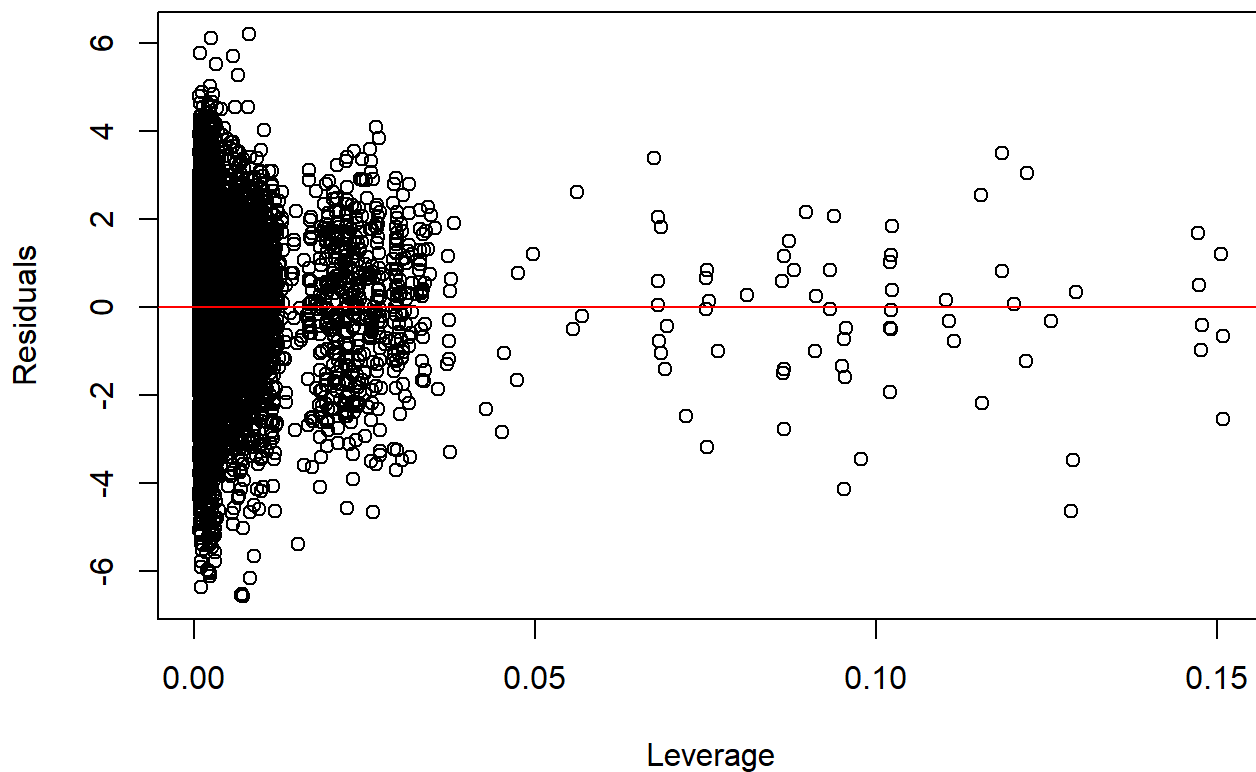
Q-Q Plot



```
# Plot 3: Residuals vs Leverage plot
leverage = hatvalues(lm(data$log_TOTEXP18 ~ X))

plot(leverage, residuals, xlab = "Leverage", ylab = "Residuals", main = "Residuals vs Leverage")
abline(h = 0, col = "red")
```

Residuals vs Leverage



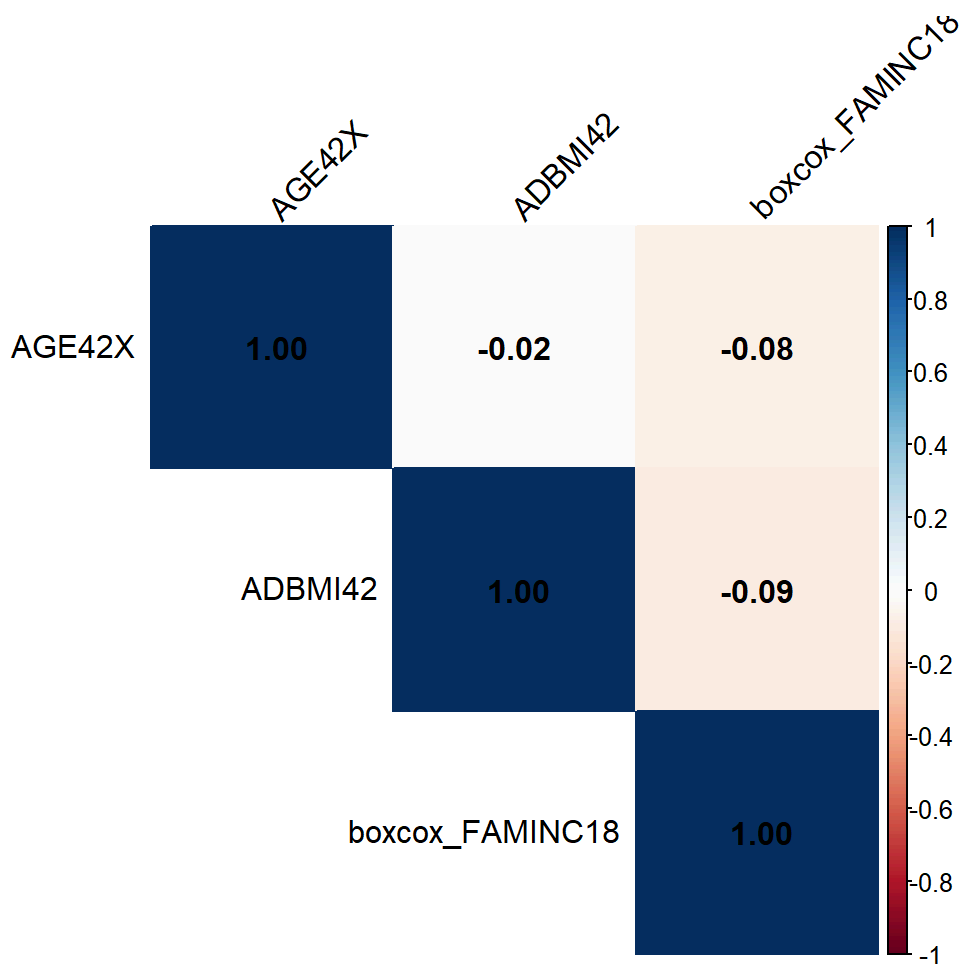
Problem 3:

The forward-backward stepwise (AIC) selected model explains almost the exact same amount of variance as the LASSO selected model but with only 29 predictors instead of 41. For this reason, I will choose the stepwise model and perform diagnostics on it.

Independence assumption:

Satisfied. The 3 continuous variables are not correlated.

```
# correlation matrix
cor_matrix = cor(data[, c("AGE42X", "ADBMI42", "boxcox_FAMINC18")], use = "complete.obs")
corrplot(cor_matrix, method = "color", type = "upper", tl.col = "black", tl.srt = 45, addCoef.col = "black")
```

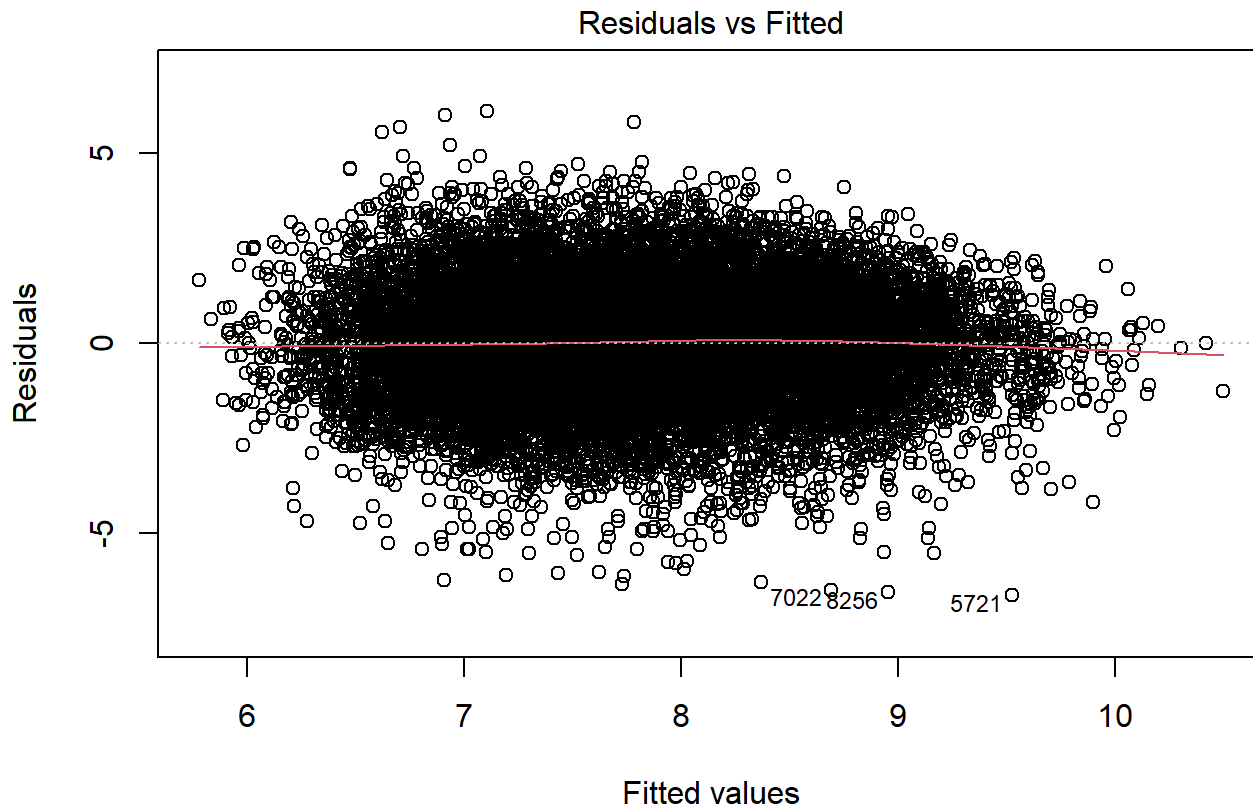


Linearity and constant variance of residuals assumptions:

Linearity is satisfied. The residuals vs fitted plot does not show any non-linear pattern.

The residuals also have mostly constant variance but there is a slight football shape. Also a trend of decreasing variance as fitted value increases.

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



`lm(log_TOTEXP18 ~ bs(AGE42X, degree = 4) + MNHLTH42 + RACETHX + SEX + bs(AD .`

I will try a weighted least squares model to resolve the slight heteroscedasticity.

I tried several methods and neither improved the heteroscedasticity. They seemed to make everything else worse though.

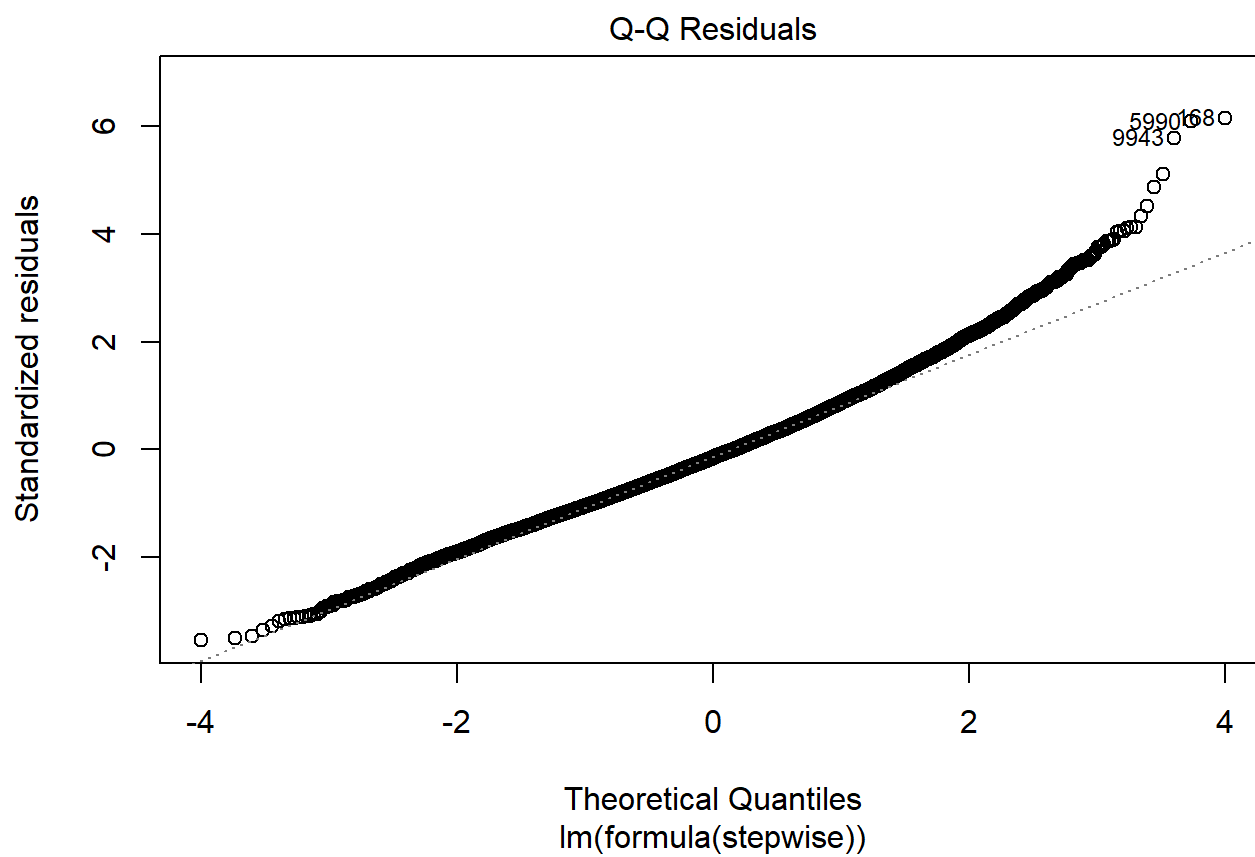
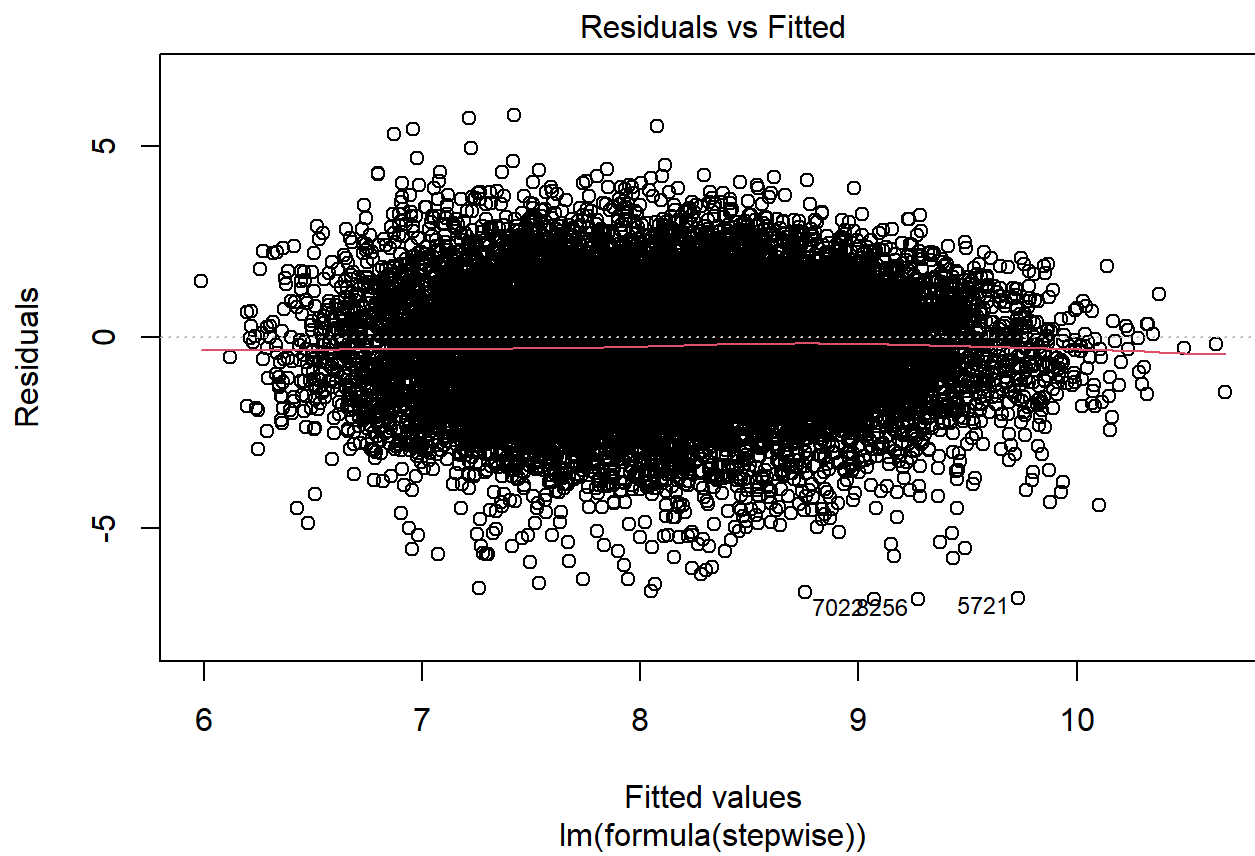
```
residuals = resid(stepwise)
variance_model = lm(I(residuals^2) ~ data$log_TOTEXP18)
fitted_variance = predict(variance_model)
weights = 1 / fitted_variance

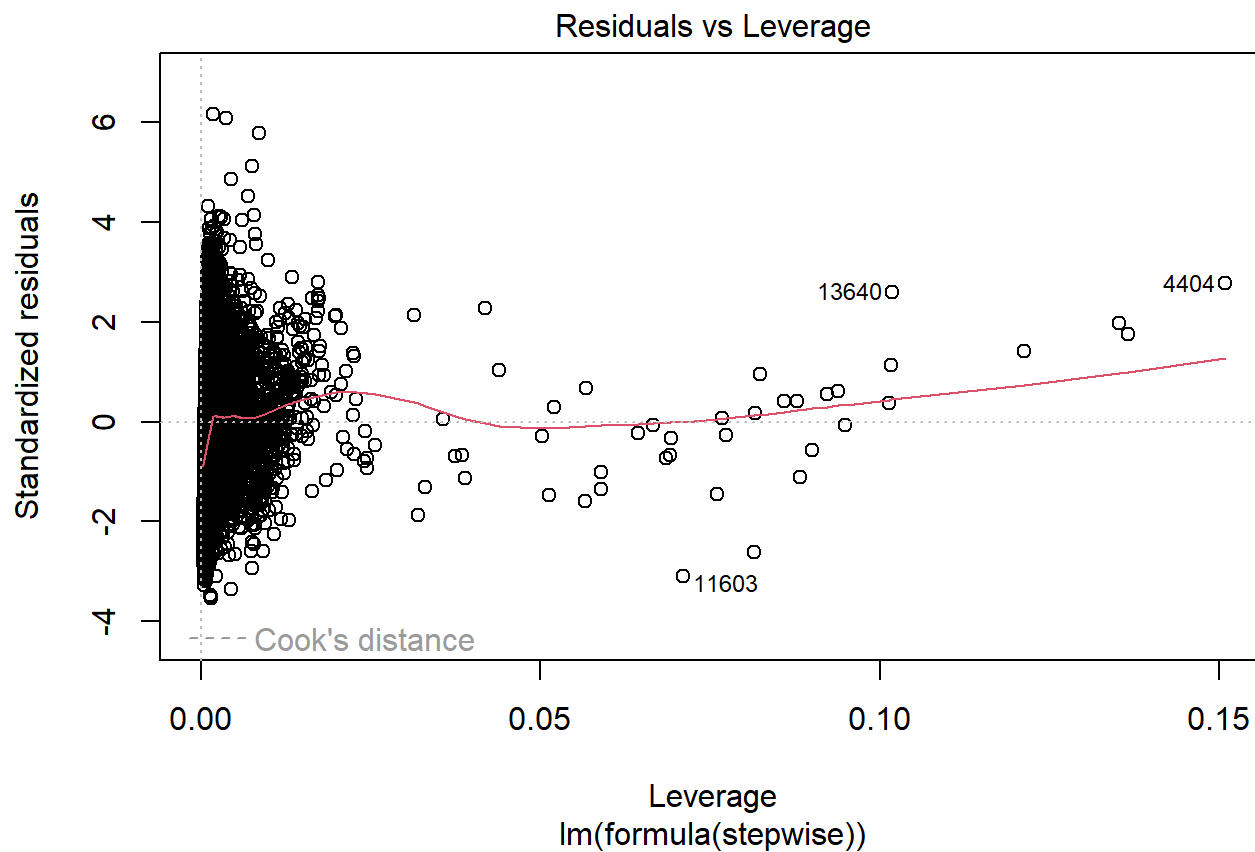
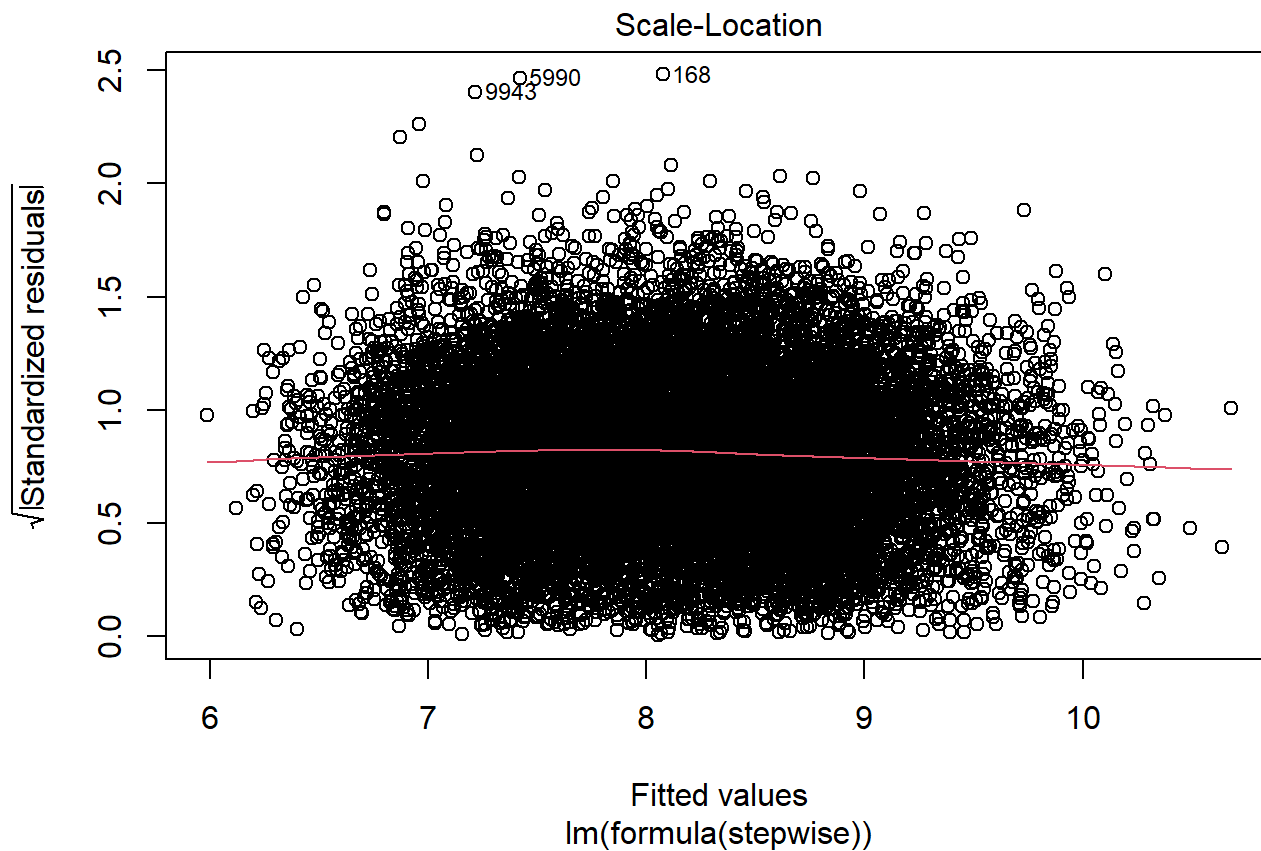
# weights = 1 / fitted(stepwise)^2

wls_model = lm(formula(stepwise), data = data, weights = weights)
summary(wls_model)
```

```
##
## Call:
## lm(formula = formula(stepwise), data = data, weights = weights)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -3.6068 -0.7845 -0.1526  0.5189  6.2711
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   7.69940     0.45233  17.022 < 2e-16 ***
## bs(AGE42X, degree = 4)1        0.47624     0.20990   2.269 0.023285 *
## bs(AGE42X, degree = 4)2        0.28899     0.18340   1.576 0.115103
## bs(AGE42X, degree = 4)3        2.01852     0.17622  11.455 < 2e-16 ***
## bs(AGE42X, degree = 4)4        1.74308     0.07976  21.853 < 2e-16 ***
## MNHLTH42FAIR                   0.90703     0.04722  19.207 < 2e-16 ***
## MNHLTH42GOOD                   0.39262     0.03271  12.003 < 2e-16 ***
## MNHLTH42POOR                   1.29946     0.08818  14.736 < 2e-16 ***
## MNHLTH42VERY GOOD              0.11402     0.03097   3.681 0.000233 ***
## RACETHXBLACK ONLY              0.36759     0.08967   4.099 4.16e-05 ***
## RACETHXHISPANIC               0.23649     0.08763   2.699 0.006967 **
## RACETHXOTHER OR MULTIPLE       0.78202     0.11792   6.632 3.42e-11 ***
## RACETHXWHITE ONLY             0.67418     0.08034   8.392 < 2e-16 ***
## SEXMALE                       -0.02627     0.11599  -0.227 0.820803
## bs(ADBMI42, degree = 4)1       -1.42523     0.93734  -1.521 0.128403
## bs(ADBMI42, degree = 4)2       -0.18771     0.45368  -0.414 0.679070
## bs(ADBMI42, degree = 4)3        0.50017     0.91665   0.546 0.585313
## bs(ADBMI42, degree = 4)4        0.19636     0.50423   0.389 0.696973
## bs(boxcox_FAMINC18, degree = 3)1 -1.66603     0.40825  -4.081 4.51e-05 ***
## bs(boxcox_FAMINC18, degree = 3)2  0.10235     0.18214   0.562 0.574190
## bs(boxcox_FAMINC18, degree = 3)3 -0.57693     0.37173  -1.552 0.120679
## NO_INC1                       -1.50957     0.48564  -3.108 0.001884 **
## RACETHXBLACK ONLY:SEXMALE      -0.16338     0.13450  -1.215 0.224495
## RACETHXHISPANIC:SEXMALE        -0.37048     0.13193  -2.808 0.004989 **
## RACETHXOTHER OR MULTIPLE:SEXMALE -0.61475     0.18169  -3.384 0.000717 ***
## RACETHXWHITE ONLY:SEXMALE      -0.23112     0.11987  -1.928 0.053871 .
## RACETHXBLACK ONLY:NO_INC1       0.88810     0.47956   1.852 0.064061 .
## RACETHXHISPANIC:NO_INC1         0.55156     0.48279   1.142 0.253281
## RACETHXOTHER OR MULTIPLE:NO_INC1 0.62457     0.67018   0.932 0.351376
## RACETHXWHITE ONLY:NO_INC1       1.08176     0.47244   2.290 0.022051 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.019 on 15694 degrees of freedom
## Multiple R-squared:  0.1868, Adjusted R-squared:  0.1853
## F-statistic: 124.3 on 29 and 15694 DF, p-value: < 2.2e-16
```

```
plot(wls_model)
```





I

am willing to accept the level of heteroscedasticity in the stepwise model. It does not seem extreme enough to me

Normality of residuals assumption:
Satisfied.

```
# stepwise formula: log_TOTEXP18 ~ bs(AGE42X, degree = 4) + MNHLTH42 + RACETHX + SEX + bs(ADBMI42, degree = 4) + bs(boxcox_FAMINC18, degree = 3) + NO_INC + RACETHX:SEX + RACETHX:NO_INC

linear_model = lm(log_TOTEXP18 ~ AGE42X + MNHLTH42 + RACETHX + SEX + ADBMI42 + boxcox_FAMINC18 + NO_INC, data=data)
vif(linear_model)
```

##	GVIF	Df	GVIF^(1/(2*Df))
## AGE42X	1.042501	1	1.021030
## MNHLTH42	1.105761	4	1.012646
## RACETHX	1.131593	4	1.015573
## SEX	1.012918	1	1.006438
## ADBMI42	1.046438	1	1.022955
## boxcox_FAMINC18	1.362076	1	1.167080
## NO_INC	1.192080	1	1.091824

Problem 4:

95% confidence intervals for coefficients.

Looks like a good chunk of the confidence intervals include 0.

```
conf_intervals = confint(stepwise, level = 0.95)
print(conf_intervals)
```

##	2.5 %	97.5 %
## (Intercept)	6.491689857	8.229443464
## bs(AGE42X, degree = 4)1	0.146668481	0.945296092
## bs(AGE42X, degree = 4)2	-0.374536073	0.348801792
## bs(AGE42X, degree = 4)3	1.834782365	2.532129379
## bs(AGE42X, degree = 4)4	1.620493765	1.931710871
## MNHLTH42FAIR	0.809582836	1.000698195
## MNHLTH42GOOD	0.310838975	0.438894666
## MNHLTH42POOR	1.142607206	1.510701410
## MNHLTH42VERY GOOD	0.060064235	0.180107234
## RACETHXBLACK ONLY	0.140405538	0.482696662
## RACETHXHISPANIC	0.001675799	0.334561157
## RACETHXOTHER OR MULTIPLE	0.504978921	0.965580226
## RACETHXWHITE ONLY	0.499979154	0.805117477
## SEXMALE	-0.292938649	0.146063810
## bs(ADBMI42, degree = 4)1	-3.216720595	0.392511205
## bs(ADBMI42, degree = 4)2	-1.006092692	0.817635734
## bs(ADBMI42, degree = 4)3	-1.298404263	2.313632098
## bs(ADBMI42, degree = 4)4	-0.836368260	1.175881409
## bs(boxcox_FAMINC18, degree = 3)1	-2.353244129	-0.749061011
## bs(boxcox_FAMINC18, degree = 3)2	-0.095735491	0.618488232
## bs(boxcox_FAMINC18, degree = 3)3	-1.203024455	0.253726553
## NO_INC1	-2.261346928	-0.464723373
## RACETHXBLACK ONLY:SEXMALE	-0.441495518	0.071172956
## RACETHXHISPANIC:SEXMALE	-0.592269009	-0.092538335
## RACETHXOTHER OR MULTIPLE:SEXMALE	-0.902787329	-0.202612313
## RACETHXWHITE ONLY:SEXMALE	-0.461162421	-0.006079689
## RACETHXBLACK ONLY:NO_INC1	-0.133468569	1.639667986
## RACETHXHISPANIC:NO_INC1	-0.471723640	1.308416768
## RACETHXOTHER OR MULTIPLE:NO_INC1	-1.103731506	1.421276062
## RACETHXWHITE ONLY:NO_INC1	0.120770538	1.866869191

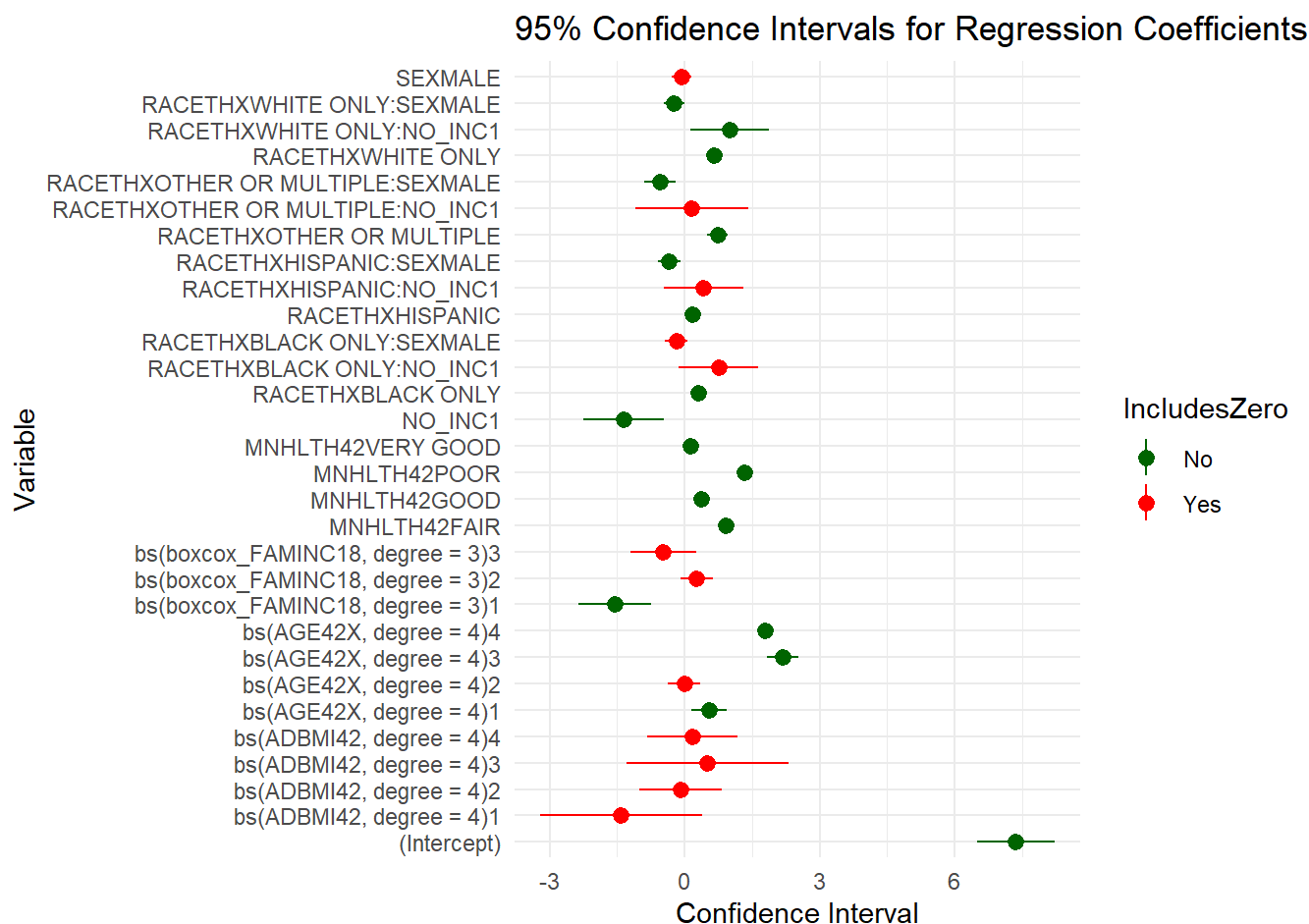
```

# Convert to data frame for plotting
conf_intervals_df <- as.data.frame(conf_intervals)
conf_intervals_df$Variable <- rownames(conf_intervals_df) # Add variable names as a column

# Create a new column to indicate if the confidence interval includes 0
conf_intervals_df$IncludesZero <- ifelse(conf_intervals_df$`2.5 %` > 0 | conf_intervals_df$`97.5 %` < 0, "No", "Yes")

# Plot the confidence intervals
ggplot(conf_intervals_df, aes(x = Variable, ymin = `2.5 %`, ymax = `97.5 %`, color = IncludesZero)) +
  geom_pointrange(aes(y = ( `2.5 %` + `97.5 %` ) / 2)) + # Use midpoint for 'y'
  coord_flip() + # Flip coordinates to make the plot horizontal
  theme_minimal() +
  labs(
    title = "95% Confidence Intervals for Regression Coefficients",
    x = "Variable",
    y = "Confidence Interval"
  ) +
  scale_color_manual(values = c("Yes" = "red", "No" = "darkgreen"))

```

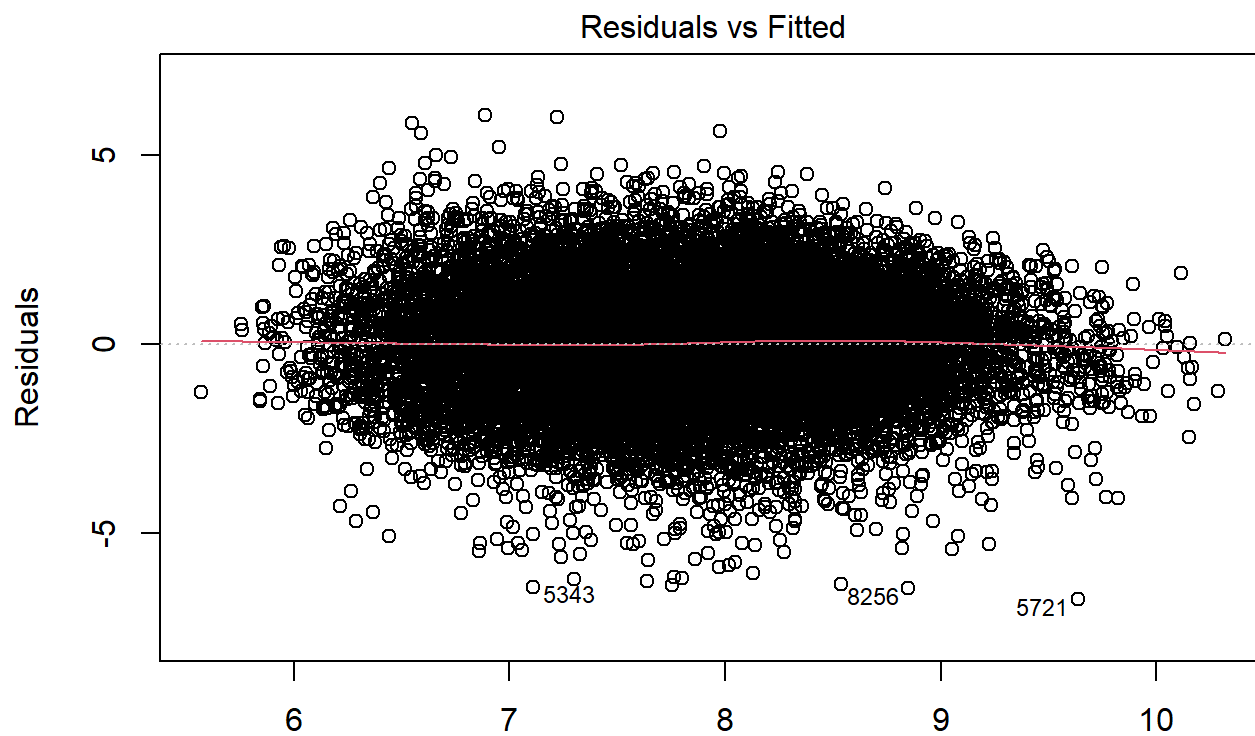


I will also fit a simple model just to see how it performs.

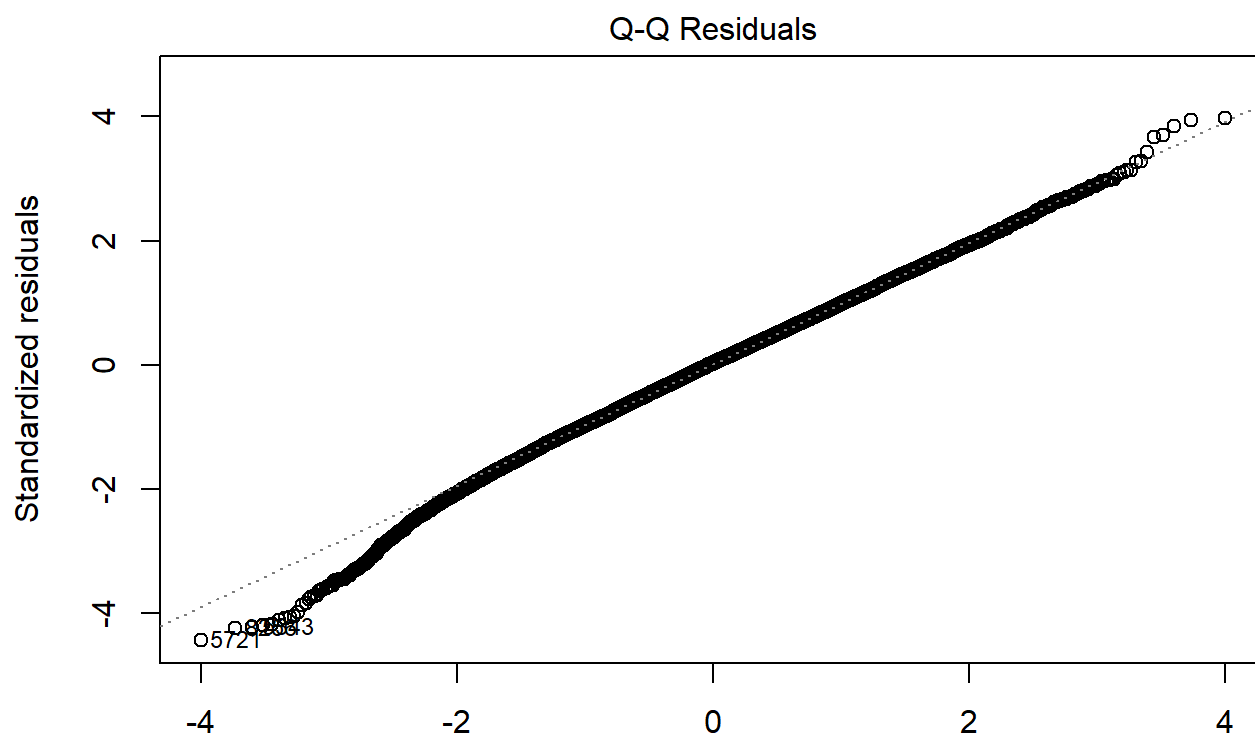
```
simple_model = lm(log_TOTEXP18 ~ AGE42X + ADBMI42 + MNHLTH42 + RACETHX + SEX + boxcox_FAMINC18 +  
NO_INC, data=data)  
summary(simple_model)
```

```
##  
## Call:  
## lm(formula = log_TOTEXP18 ~ AGE42X + ADBMI42 + MNHLTH42 + RACETHX +  
##     SEX + boxcox_FAMINC18 + NO_INC, data = data)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -6.7450 -0.9841  0.0423  1.0168  6.0460   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)      5.0269769   0.0934662   53.784 < 2e-16 ***  
## AGE42X            0.0305724   0.0006848   44.647 < 2e-16 ***  
## ADBMI42           0.0206268   0.0017402   11.853 < 2e-16 ***  
## MNHLTH42FAIR      0.9107021   0.0487434   18.684 < 2e-16 ***  
## MNHLTH42GOOD      0.3582367   0.0327063   10.953 < 2e-16 ***  
## MNHLTH42POOR      1.3140116   0.0939775   13.982 < 2e-16 ***  
## MNHLTH42VERY GOOD 0.1082867   0.0306654    3.531 0.000415 ***  
## RACETHXBLACK ONLY  0.2429771   0.0662492    3.668 0.000246 ***  
## RACETHXHISPANIC   0.0149461   0.0640367    0.233 0.815455   
## RACETHXOTHER OR MULTIPLE 0.5121827   0.0892229    5.740 9.61e-09 ***  
## RACETHXWHITE ONLY  0.5723745   0.0582530    9.826 < 2e-16 ***  
## SEXMALE           -0.3249948   0.0245677  -13.229 < 2e-16 ***  
## boxcox_FAMINC18    0.0009672   0.0002110    4.585 4.58e-06 ***  
## NO_INC1           0.1072883   0.0907089    1.183 0.236916   
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 1.52 on 15710 degrees of freedom  
## Multiple R-squared:  0.1887, Adjusted R-squared:  0.188   
## F-statistic: 281 on 13 and 15710 DF, p-value: < 2.2e-16
```

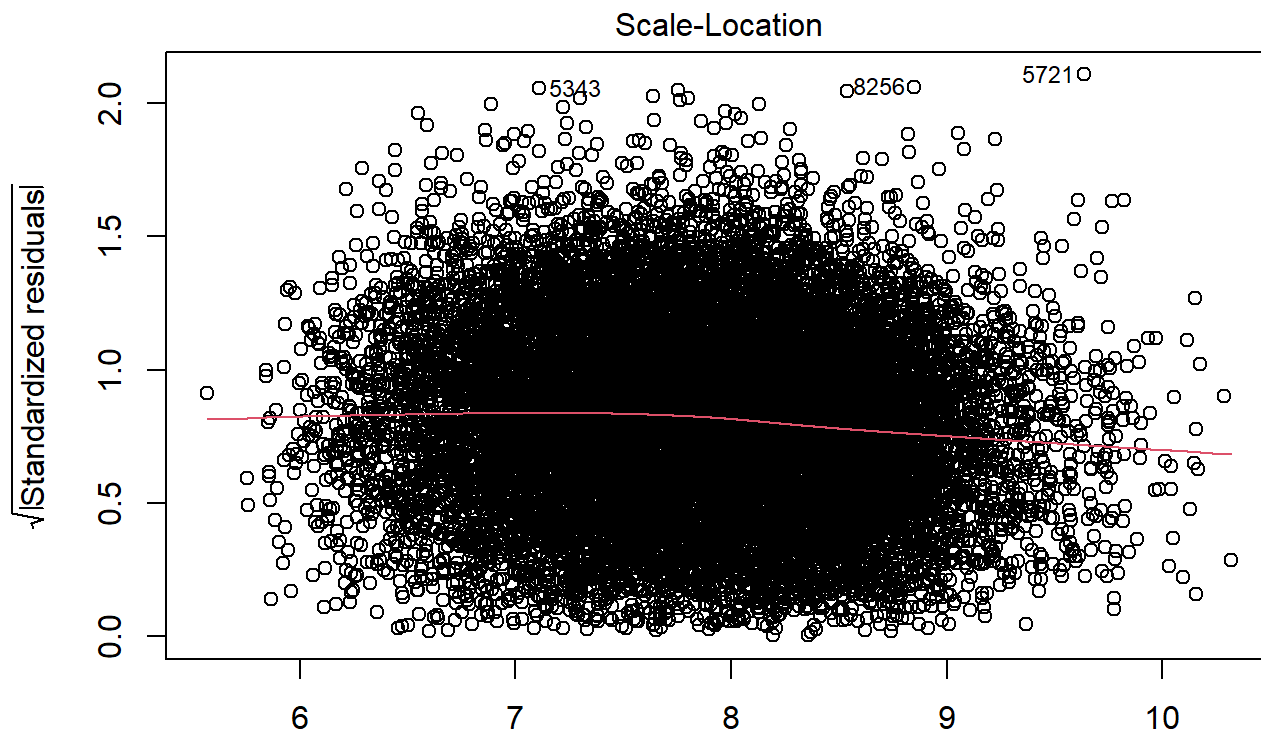
```
plot(simple_model)
```



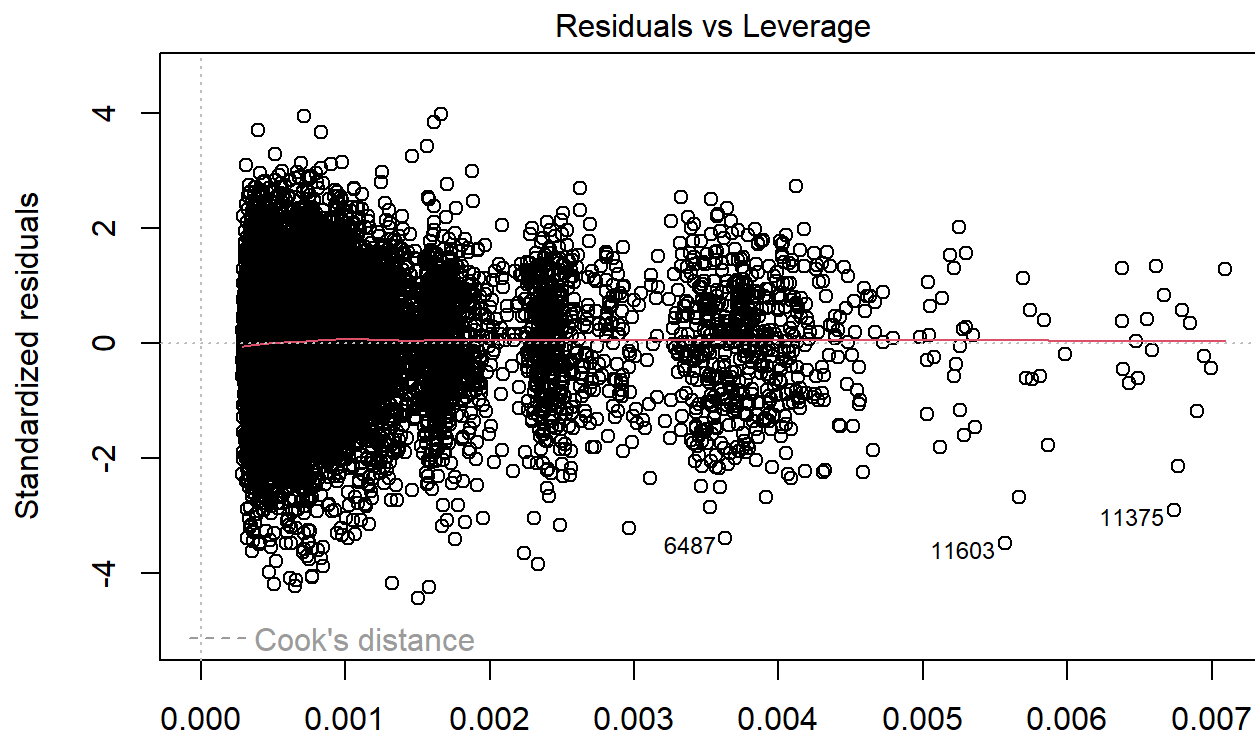
Fitted values
 $\text{lm}(\log_TOTEXP18 \sim \text{AGE42X} + \text{ADBMI42} + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{boxcox_FAM})$



Theoretical Quantiles
 $\text{lm}(\log_TOTEXP18 \sim \text{AGE42X} + \text{ADBMI42} + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{boxcox_FAM})$



Fitted values
 $\text{lm}(\log_TOTEXP18 \sim \text{AGE42X} + \text{ADBMI42} + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{boxcox_FAM})$



Leverage
 $\text{lm}(\log_TOTEXP18 \sim \text{AGE42X} + \text{ADBMI42} + \text{MNHLTH42} + \text{RACETHX} + \text{SEX} + \text{boxcox_FAM})$


```
# conf intervals
conf_intervals = confint(simple_model, level = 0.95)
print(conf_intervals)
```

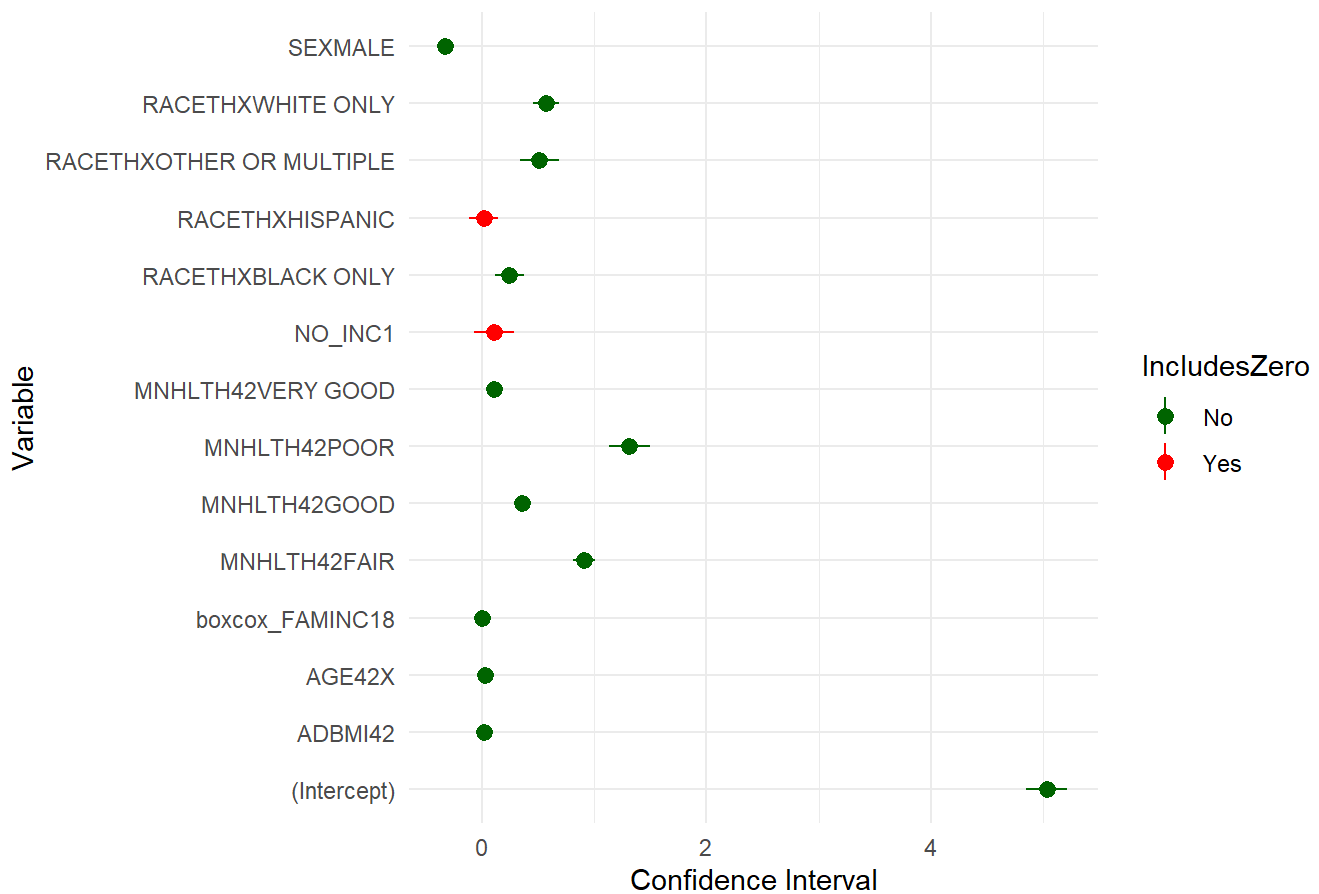
```
##              2.5 %      97.5 %
## (Intercept)    4.843772354  5.210181466
## AGE42X         0.029230181  0.031914593
## ADBMI42        0.017215715  0.024037802
## MNHLTH42FAIR   0.815159298  1.006244823
## MNHLTH42GOOD   0.294128575  0.422344887
## MNHLTH42POOR   1.129804960  1.498218253
## MNHLTH42VERY GOOD 0.048178958  0.168394346
## RACETHXBLACK ONLY 0.113121051  0.372833154
## RACETHXHISPANIC -0.110573285  0.140465430
## RACETHXOTHER OR MULTIPLE 0.337295623  0.687069788
## RACETHXWHITE ONLY 0.458191984  0.686557096
## SEXMALE        -0.373150181 -0.276839323
## boxcox_FAMINC18 0.000553698  0.001380761
## NO_INC1        -0.070511489  0.285088115
```

```
# Convert to data frame for plotting
conf_intervals_df <- as.data.frame(conf_intervals)
conf_intervals_df$Variable <- rownames(conf_intervals_df) # Add variable names as a column

# Create a new column to indicate if the confidence interval includes 0
conf_intervals_df$IncludesZero <- ifelse(conf_intervals_df$`2.5 %` > 0 | conf_intervals_df$`97.5 %` < 0, "No", "Yes")

# Plot the confidence intervals
ggplot(conf_intervals_df, aes(x = Variable, ymin = `2.5 %`, ymax = `97.5 %`, color = IncludesZero)) +
  geom_pointrange(aes(y = ( `2.5 %` + `97.5 %` ) / 2)) + # Use midpoint for 'y'
  coord_flip() + # Flip coordinates to make the plot horizontal
  theme_minimal() +
  labs(
    title = "95% Confidence Intervals for Regression Coefficients",
    x = "Variable",
    y = "Confidence Interval"
  ) +
  scale_color_manual(values = c("Yes" = "red", "No" = "darkgreen"))
```

95% Confidence Intervals for Regression Coefficients



Train/test split:

```
train_size = nrow(data) - 2000
train_data = data[1:train_size, ]
test_data = data[(train_size + 1):nrow(data), ]
```

Train/test prediction error for simple model:

```
simple_model_ = lm(log_TOTEXP18 ~ AGE42X + ADBMI42 + MNHLTH42 + RACETHX + SEX + boxcox_FAMINC18
+ NO_INC, data=train_data)
simple_model_pred = predict(simple_model_, newdata = test_data)
simple_model_pred = exp(simple_model_pred) - 1 # undo log transformation
simple_model_mse = mean((test_data$TOTEXP18 - simple_model_pred)^2)
print(paste("MSE for simple model = ", simple_model_mse))
```

```
## [1] "MSE for simple model = 337167774.117127"
```

```
print(paste("RMSE for simple model = ", sqrt(simple_model_mse)))
```

```
## [1] "RMSE for simple model = 18362.1288013434"
```

Train/test prediction error for full model:

```
full_model_ = lm(model_formula, data=train_data)
full_model_pred = predict(full_model_, newdata = test_data)
```

```
## Warning in bs(boxcox_FAMINC18, degree = 3L, knots = numeric(0), Boundary.knots
## = c(-2.60526315789473, : some 'x' values beyond boundary knots may cause
## ill-conditioned bases
```

```
full_model_pred = exp(full_model_pred) - 1 # undo log transformation
full_model_mse = mean((test_data$TOTEXP18 - full_model_pred)^2)
print(paste("MSE for full model = ", full_model_mse))
```

```
## [1] "MSE for full model = 336947320.599903"
```

```
print(paste("RMSE for full model = ", sqrt(full_model_mse)))
```

```
## [1] "RMSE for full model = 18356.1248797207"
```

Train/test prediction error for stepwise model:

```
null_model_ = lm(log_TOTEXP18~1, data=train_data)
stepwise_model_ = step(null_model, scope = list(lower = null_model, upper = full_model_), direct
ion = "both", k = 2, trace = FALSE, test = "F", steps = 1000, add = 0.05 , drop = 0.05)

stepwise_model_pred = predict(stepwise_model_, newdata = test_data)
stepwise_model_pred = exp(stepwise_model_pred) - 1 # undo log transformation
stepwise_model_mse = mean((test_data$TOTEXP18 - stepwise_model_pred)^2)
print(paste("MSE for stepwise model = ", stepwise_model_mse))
```

```
## [1] "MSE for stepwise model = 336966631.013725"
```

```
print(paste("RMSE for stepwise model = ", sqrt(stepwise_model_mse)))
```

```
## [1] "RMSE for stepwise model = 18356.6508659321"
```

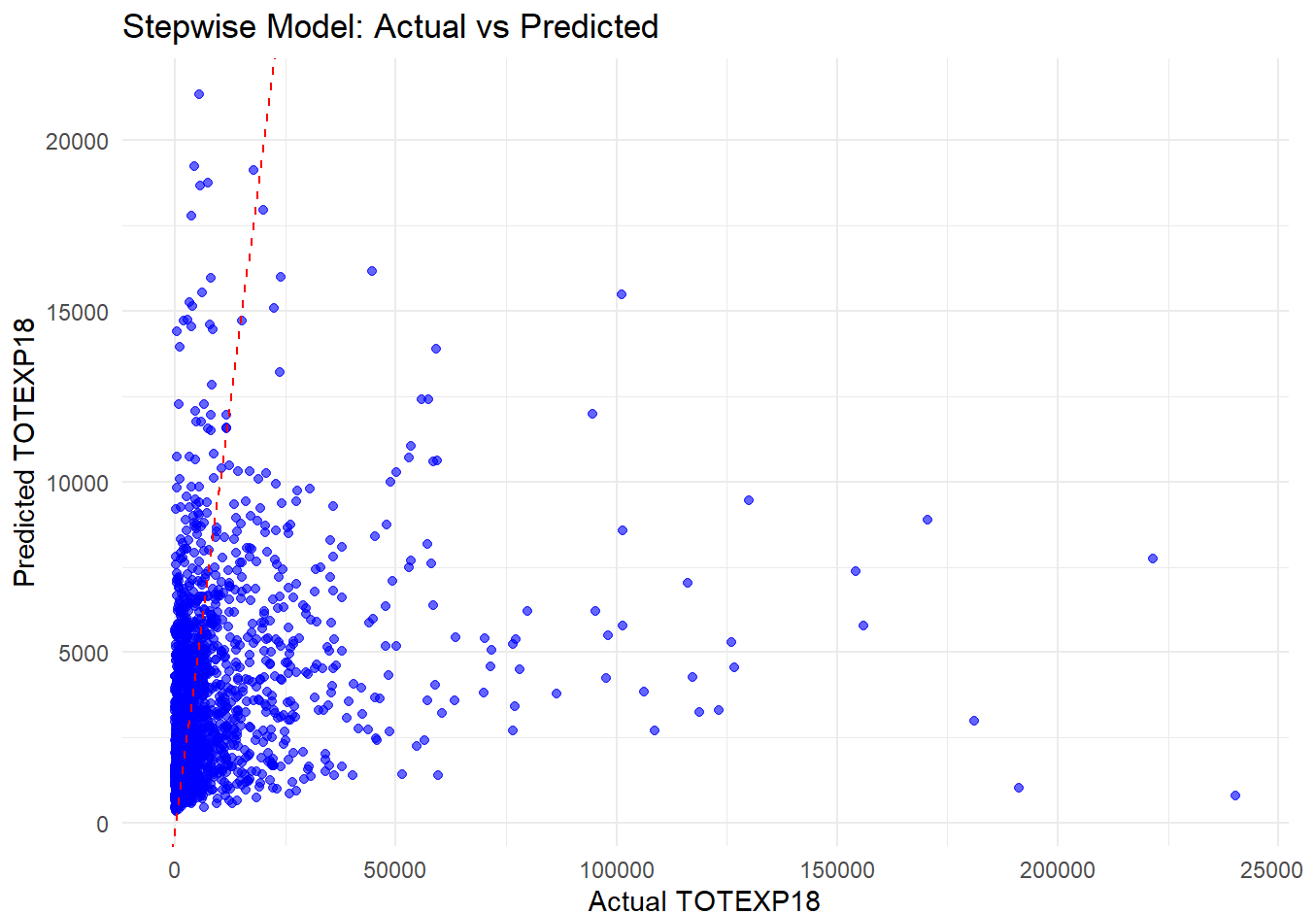
```

test = test_data
test = cbind(stepwise_model_pred, test)
test$diff = (test$stepwise_model_pred - test$TOTEXP18)^2

# Plot actual vs predicted
plot_data = data.frame(
  Actual = test_data$TOTEXP18,
  Predicted = stepwise_model_pred
)

ggplot(plot_data, aes(x = Actual, y = Predicted)) +
  geom_point(alpha = 0.6, color = 'blue') + # scatter plot of actual vs predicted
  geom_abline(slope = 1, intercept = 0, color = "red", linetype = "dashed") + # diagonal line for reference
  labs(
    title = "Stepwise Model: Actual vs Predicted",
    x = "Actual TOTEXP18",
    y = "Predicted TOTEXP18"
  ) +
  theme_minimal()

```



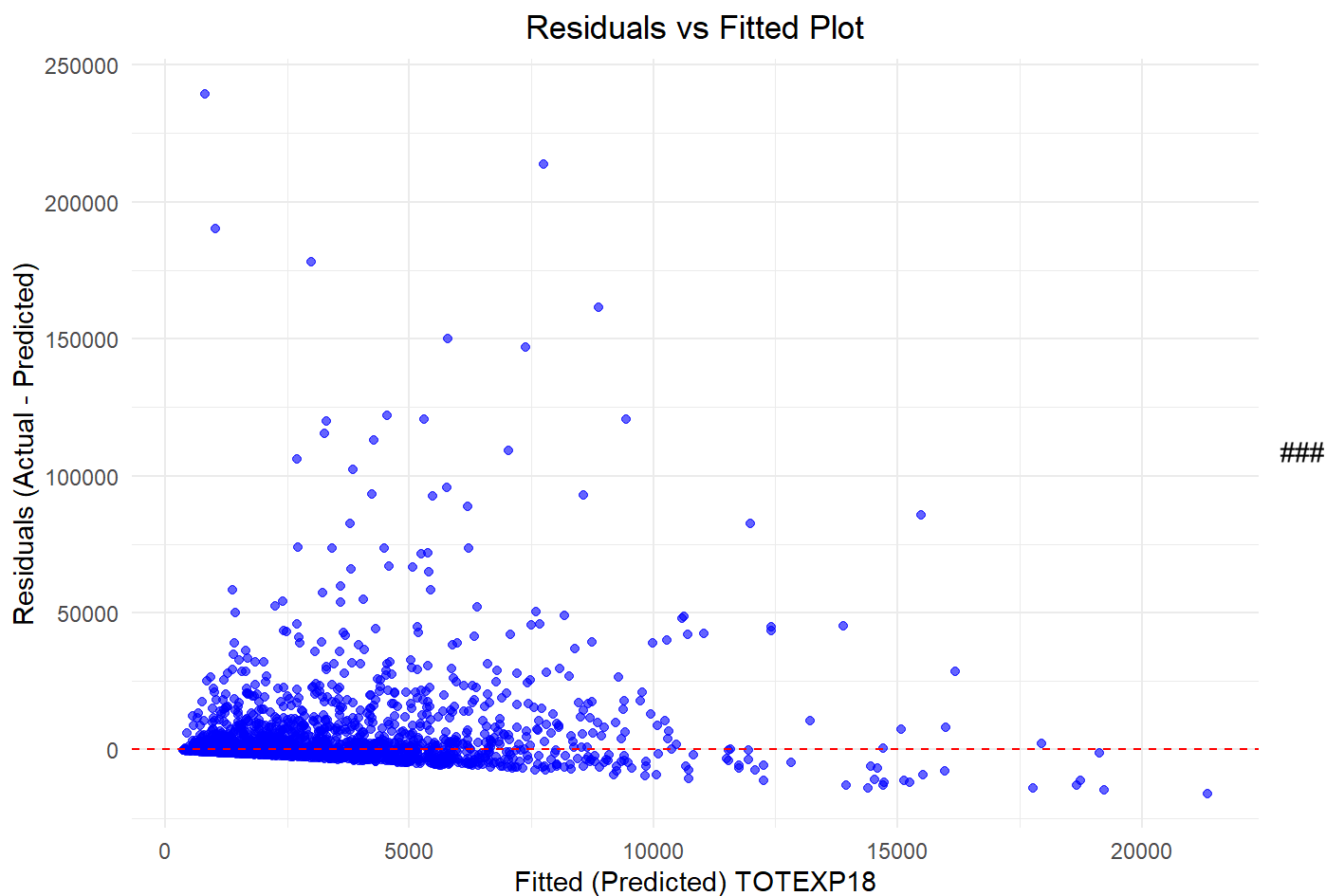
```

# Plot residuals vs fitted for test data
residuals = test_data$TOTEXP18 - stepwise_model_pred

residuals_data = data.frame(
  Fitted = stepwise_model_pred, # Predicted values (fitted)
  Residuals = residuals        # Residuals (errors)
)

ggplot(residuals_data, aes(x = Fitted, y = Residuals)) +
  geom_point(alpha = 0.6, color = 'blue') + # Scatter plot of residuals vs fitted
  geom_hline(yintercept = 0, color = "red", linetype = "dashed") + # Horizontal line at 0
  labs(
    title = "Residuals vs Fitted Plot",
    x = "Fitted (Predicted) TOTEXP18",
    y = "Residuals (Actual - Predicted)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5) # Center the title
  )

```



From the QC plots I made, it looks like a lot of my error comes from under estimating total expenditure for observations with some relatively high expenditures.

Train/test prediction error for lasso model:

```
X = model.matrix(model_formula, data = train_data)[, -1]
X_test = model.matrix(model_formula, data = test_data)[, -1]

# Lasso model
lasso_ = glmnet(x = X, y = train_data$log_TOTEXP18, alpha = 1)

# cv to find optimal lambda
cv_lasso_ = cv.glmnet(x = X, y = train_data$log_TOTEXP18, alpha = 1, nfolds = 10)
best_lambda_ = cv_lasso_$lambda.min
best_lasso_ = glmnet(x = X, y = train_data$log_TOTEXP18, alpha = 1, lambda = best_lambda_)
coefficients_best_lasso_ = coef(best_lasso_)

# Make predictions using the best_lasso model
best_lasso_pred <- predict(best_lasso_, newx = X_test, s = best_lambda_)

# Inverse transform the predictions (since the target was log-transformed)
best_lasso_pred <- exp(best_lasso_pred) - 1

# Calculate MSE (Mean Squared Error)
best_lasso_mse <- mean((test_data$TOTEXP18 - best_lasso_pred)^2)

# Calculate RMSE (Root Mean Squared Error)
best_lasso_rmse <- sqrt(best_lasso_mse)

# Print the results
print(paste("MSE for Lasso model = ", best_lasso_mse))
```

```
## [1] "MSE for Lasso model = 339513246.296453"
```

```
print(paste("RMSE for Lasso model = ", best_lasso_rmse))
```

```
## [1] "RMSE for Lasso model = 18425.8852242288"
```

So it looks like all 4 models I fit don't perform amazingly well. RMSE error is about \$18,000 for all models. Lowest error is full model, then stepwise model, then simple model, then lasso model.

This doesn't totally make sense to me because the full model should be more prone to overfitting. But I've already spent 10hrs on this and I have other classes.

Problem 5:

I found that mental health was a good indicator of how total medical expenditure. People with poor mental health spend the most, followed by those with fair, followed by those with good, then very good.

I found that older people have higher total expenditure.

I found that the "black only" ethnicity had higher expenditure. "Hispanic" also had higher expenditure but not as high as "black only".

I found that people with no income have less expenditure.

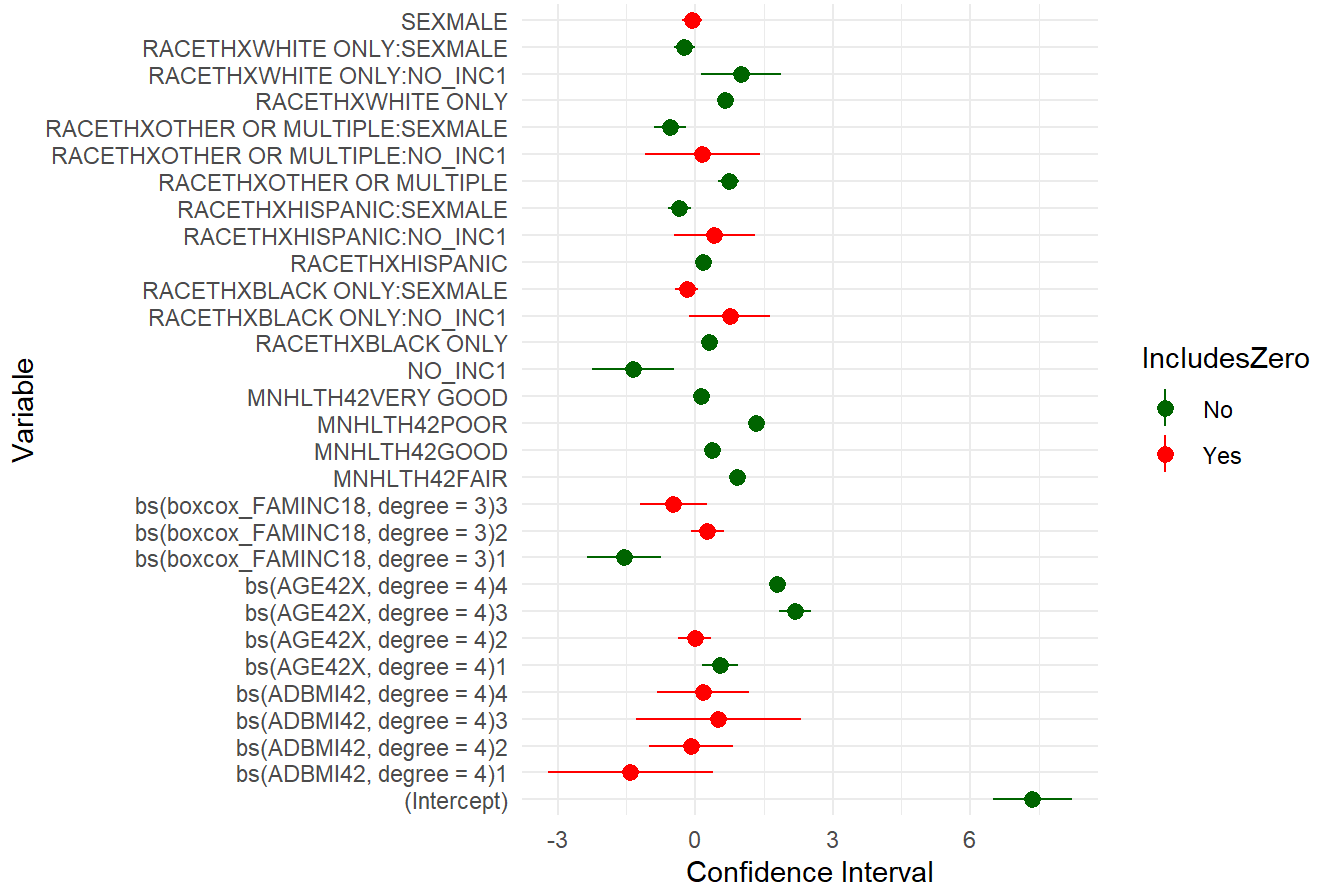
```
conf_intervals = confint(stepwise, level = 0.95)

# Convert to data frame for plotting
conf_intervals_df <- as.data.frame(conf_intervals)
conf_intervals_df$Variable <- rownames(conf_intervals_df) # Add variable names as a column

# Create a new column to indicate if the confidence interval includes 0
conf_intervals_df$IncludesZero <- ifelse(conf_intervals_df$`2.5 %` > 0 | conf_intervals_df$`97.5 %` < 0, "No", "Yes")

# Plot the confidence intervals
ggplot(conf_intervals_df, aes(x = Variable, ymin = `2.5 %`, ymax = `97.5 %`, color = IncludesZero)) +
  geom_pointrange(aes(y = ( `2.5 %` + `97.5 %` ) / 2)) + # Use midpoint for 'y'
  coord_flip() + # Flip coordinates to make the plot horizontal
  theme_minimal() +
  labs(
    title = "95% Confidence Intervals for Regression Coefficients",
    x = "Variable",
    y = "Confidence Interval"
  ) +
  scale_color_manual(values = c("Yes" = "red", "No" = "darkgreen"))
```

95% Confidence Intervals for Regression Coefficients



The paper “Racial and Ethnic Disparities in Health Care Use and Expenditures: Evidence from the MEPS” by R. E. Schoen, B. M. Stokes, & A. P. Ko (Health Affairs, 2013) also found that black and hispanic populations had higher total expenditure.

The paper “The Relationship Between Age and Health Care Expenditures in the United States” by H. A. Skinner, J. M. Staiger, & E. F. P. Rosenthal (The Journal of Health Economics, 2010) found that healthcare expenditure increases with age.

These support my findings.