STA 141 Project

Influence of risk factors and socioeconomic status on heart attack death rate at county level in the US

```
library(ggplot2)
library(lmtest)
library(tidyverse)
library(sandwich)
library(MASS)
library(usmap)
library(scales)
```

```
# read csv file with the cdc data
cdc <-  read.csv('cdc-report.csv')

# changing the variables names for better readability
names(cdc) <- c('cnty_fips','display_name','HeartAttack_rate','theme_range','obesity','diabetes','smoke
'no_college','no_internet','income', 'poverty')

#checking number of rows and columns
c(nrow(cdc),ncol(cdc))
```

```
## [1] 3226   11
```

```
head(cdc)
```

```
##   cnty_fips    display_name HeartAttack_rate         theme_range obesity
## 1      1001 "Autauga, (AL)"            143.8 101.9 - 501.3 (639)    32.7
## 2      1003 "Baldwin, (AL)"             37.2    0.0 - 40.5 (648)    30.1
## 3      1005 "Barbour, (AL)"             42.0   40.6 - 54.6 (639)    41.2
## 4      1007    "Bibb, (AL)"             53.0   40.6 - 54.6 (639)    37.4
## 5      1009  "Blount, (AL)"             32.5    0.0 - 40.5 (648)    32.5
## 6      1011 "Bullock, (AL)"             40.2    0.0 - 40.5 (648)    46.4
##   diabetes smoker no_college no_internet income poverty
## 1     11.4   19.4       73.4        19.4  58000    12.1
## 2      8.2   17.5       68.1        18.2  60000    10.1
## 3     15.1   24.5       88.4        39.5  36000    27.1
## 4     12.2   22.7       89.6        30.8  48000    20.3
## 5     12.2   22.1       86.9        27.0  53000    16.3
## 6     27.9   24.4       87.9        39.9  32000    30.0
```

```
# checking columns types
sapply(cdc, class)
```

```
##        cnty_fips     display_name HeartAttack_rate      theme_range
##        "integer"         "factor"        "numeric"         "factor"
##          obesity         diabetes           smoker       no_college
##        "numeric"        "numeric"        "numeric"        "numeric"
##      no_internet           income          poverty
##        "numeric"        "integer"        "numeric"
```

```
# example of '-1' values meaning insufficient information
tail(cdc)
```

```
##      cnty_fips                            display_name HeartAttack_rate
## 3221     72149                         "Villalba, (PR)"             68.6
## 3222     72151                          "Yabucoa, (PR)"             32.9
## 3223     72153                            "Yauco, (PR)"            103.8
## 3224     78010  "Saint Croix (County Equivalent), (VI)"             -1.0
## 3225     78020   "Saint John (County Equivalent), (VI)"             -1.0
## 3226     78030 "Saint Thomas (County Equivalent), (VI)"             -1.0
##              theme_range obesity diabetes smoker no_college no_internet income
## 3221   54.7 - 70.6 (645)      -1     14.6     -1       80.1        30.7     -1
## 3222    0.0 - 40.5 (648)      -1     14.3     -1       82.0        42.0     -1
## 3223 101.9 - 501.3 (639)      -1     14.9     -1       75.4        52.2     -1
## 3224                          -1     -1.0     -1       -1.0        -1.0     -1
## 3225                          -1     -1.0     -1       -1.0        -1.0     -1
## 3226                          -1     -1.0     -1       -1.0        -1.0     -1
##      poverty
## 3221      -1
## 3222      -1
## 3223      -1
## 3224      -1
## 3225      -1
## 3226      -1
```

```
# showing that those '-1' values are not NA
colSums(is.na(cdc))
```

```
##        cnty_fips     display_name HeartAttack_rate      theme_range
##                0                0                0                0
##          obesity         diabetes           smoker       no_college
##                0                0                0                0
##      no_internet           income          poverty
##                0                0                0
```

```
# replacing '-1' values for NA
cdc[cdc == -1] <- NA
colSums(is.na(cdc))
```

```
##        cnty_fips     display_name HeartAttack_rate      theme_range
##                0                0               11                0
##          obesity         diabetes           smoker       no_college
##               84                6               84                6
##      no_internet           income          poverty
##                6               85               85
```

```r
#Removing missing values
nrow(cdc)
```

```
## [1] 3226
```

```r
cdc <- na.omit(cdc)
nrow(cdc)
```

```
## [1] 3133
```

```r
#Removing 0 values from the outcome heart attack death rate
cdc <- cdc[cdc$HeartAttack_rate != 0,]

nrow(cdc)
```

```
## [1] 3131
```

```r
ncol(cdc)
```

```
## [1] 11
```

```r
# mean an median of the outcome heart attack death rate
c(mean(cdc$HeartAttack_rate),median(cdc$HeartAttack_rate))
```

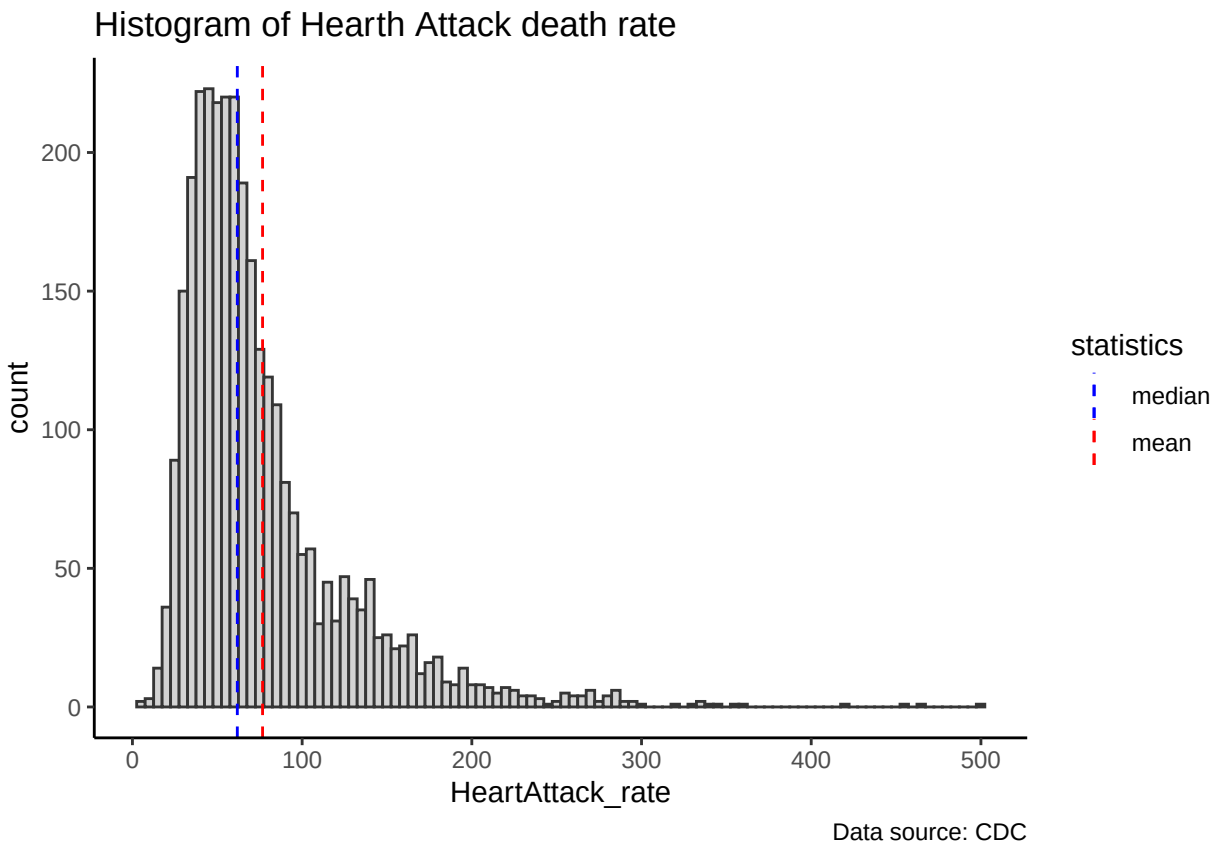
```
## [1] 76.68604 61.80000
```

```r
vars = c('HeartAttack_rate','obesity','diabetes','smoker',
'no_college','no_internet','income', 'poverty')

mean_table = as.data.frame(apply(cdc[,vars], MARGIN=2, FUN=mean))
colnames(mean_table) = c("Mean Values")
mean_table
```

```
##                  Mean Values
## HeartAttack_rate    76.68604
## obesity             33.43392
## diabetes            10.49080
## smoker              20.13108
## no_college          78.02839
## no_internet         24.59665
## income           55689.55605
## poverty             14.46694
```

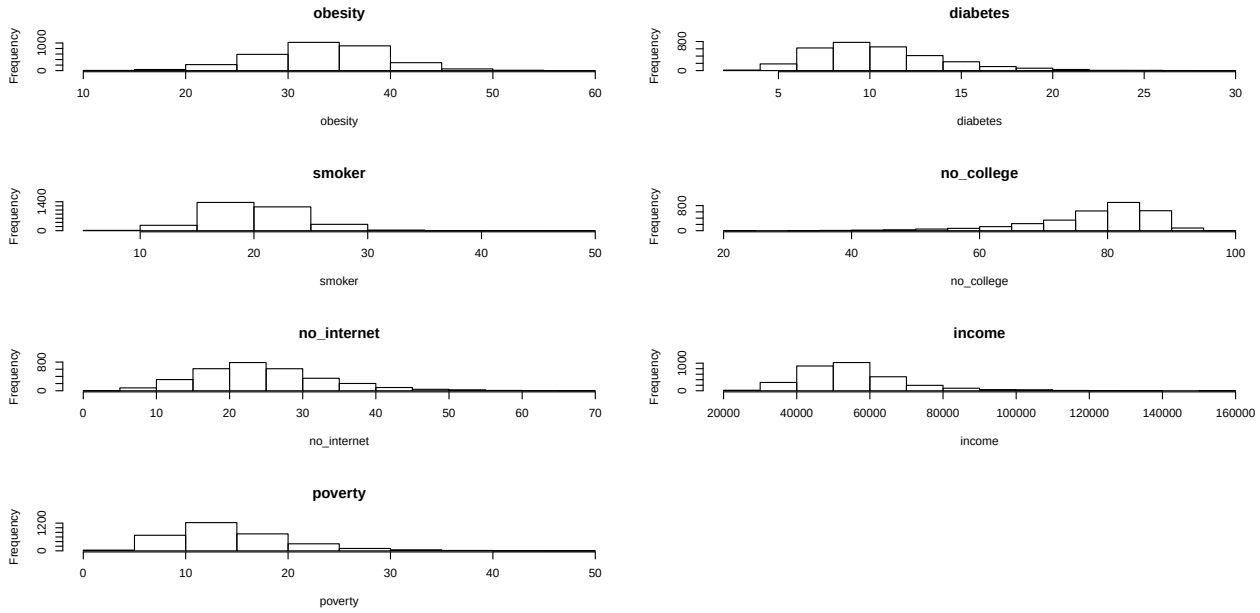
```r
# Creating a histogram for the outcome
p = ggplot(data=cdc, aes(x=HeartAttack_rate)) +
  geom_histogram(bins=100, fill="#D3D3D3", color='#353535') +
  geom_vline(aes(xintercept = mean(cdc$HeartAttack_rate), color='mean'), linetype="dashed") +
  geom_vline(aes(xintercept = median(cdc$HeartAttack_rate), color='median'), linetype="dashed") +
  scale_color_manual(name="statistics", values =c(median = "blue", mean = "red"))+
```

```
  theme_bw() +
  theme(axis.line = element_line(colour = "black"),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_blank(),
    panel.background = element_blank())

# Add titles
p <- p + labs(title = "Histogram of Hearth Attack death rate",
              caption = "Data source: CDC")

p
```

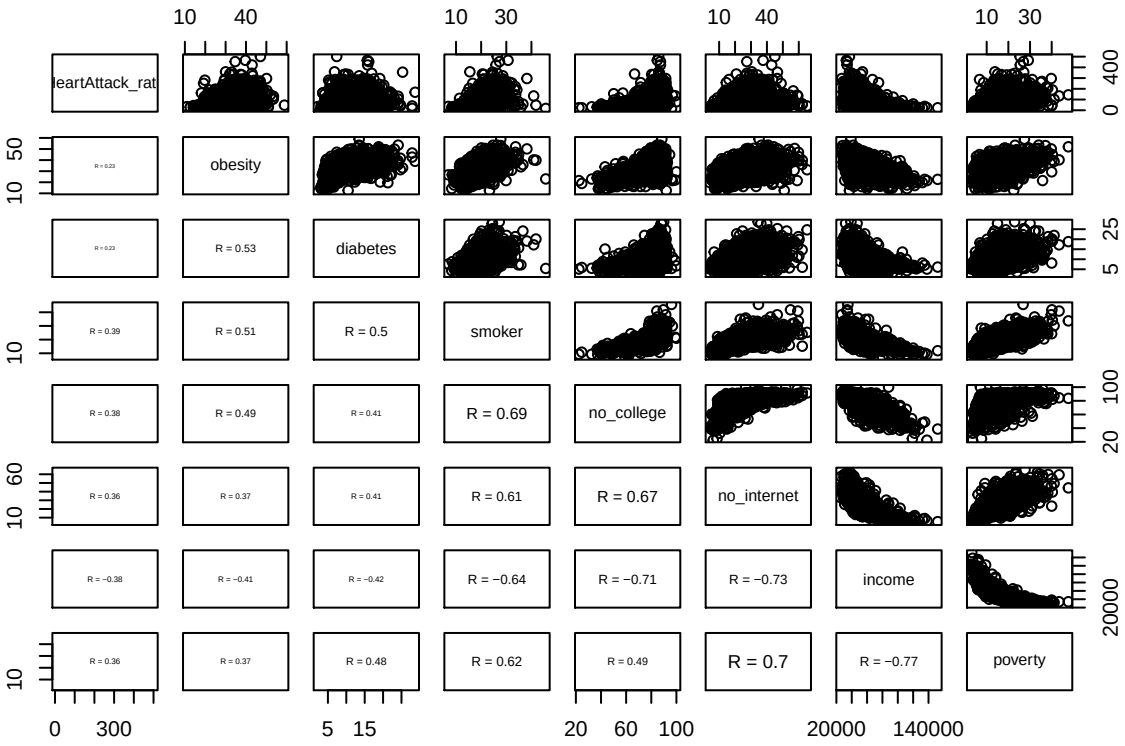



```
# function 1: create histograms for the selected variables in a dataframe

histo_maker = function(df, var_vector, dim_vector){
  # df: the dataframe used
  # var_vector: the vector of variables column index that we want to make a histogram for
  # dim_vector: dimension of the output graphs

  par(mfrow=dim_vector)

  for (col in colnames(df[, var_vector])) {
    hist(cdc[,col], main=col, xlab=col)
  }
}
```

```r
dim = c(4,2)
index = c(5,6,7,8,9,10,11)
histo_maker(cdc,index, dim)

# track the distribution of values in each variables
```




```r
# selecting variables for the analysis
vars = c('HeartAttack_rate','obesity','diabetes','smoker',
'no_college','no_internet','income', 'poverty')

panel.cor <- function(x, y) {
    par(usr = c(0, 1, 0, 1))
    r <- round(cor(x, y, use = "complete.obs"), 2)
    txt <- paste0("R = ", r)
    cex.cor <- 0.8/strwidth(txt)
    text(0.5, 0.5, txt, cex = cex.cor * abs(r))
}
# plotting paired plot with correlation
pairs(~HeartAttack_rate + obesity + diabetes + smoker + no_college +no_internet + income + poverty, data
```




```
# correlation matrix
mcor = round(cor(cdc[,vars]),2)
upper = mcor
upper[upper.tri(mcor)] = ""
upper = as.data.frame(upper)

upper
```

```
##                  HeartAttack_rate obesity diabetes smoker no_college
## HeartAttack_rate                1
## obesity                      0.23       1
## diabetes                     0.23    0.53        1
## smoker                       0.39    0.51      0.5      1
## no_college                   0.38    0.49     0.41   0.69          1
## no_internet                  0.36    0.37     0.41   0.61       0.67
## income                      -0.38   -0.41    -0.42  -0.64      -0.71
## poverty                      0.36    0.37     0.48   0.62       0.49
##                  no_internet income poverty
## HeartAttack_rate
## obesity
## diabetes
## smoker
## no_college
## no_internet                1
## income                 -0.73      1
## poverty                  0.7  -0.77       1
```

```
# dividing the dataset into training and test set
set.seed(10)
```

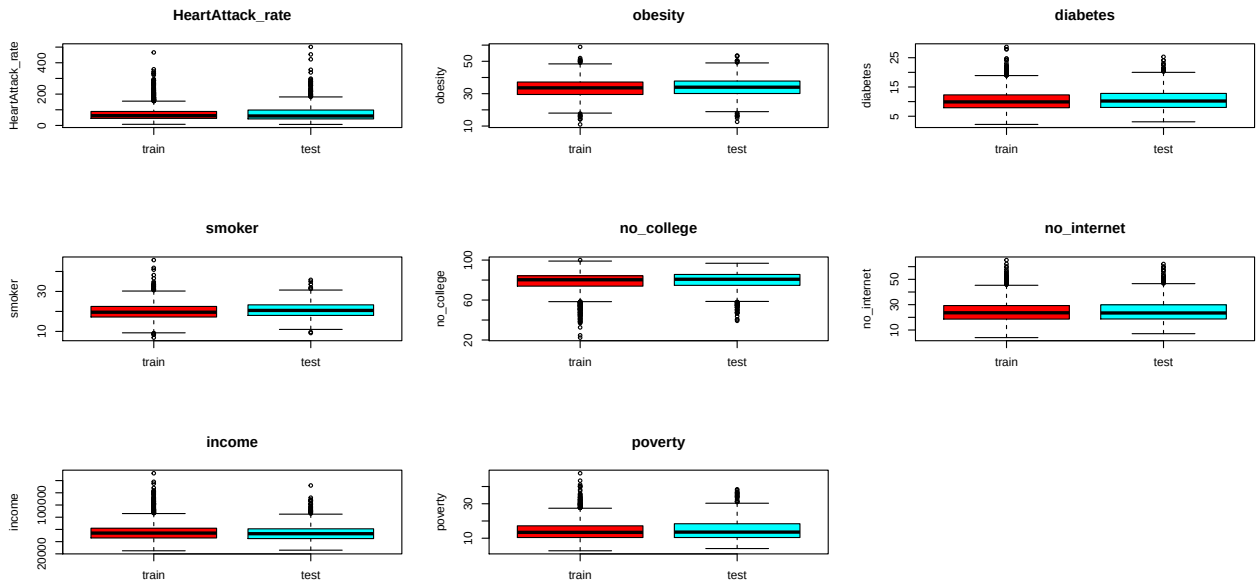
```
n <- nrow(cdc)/4
ind <- sample(1:(2*n), n, replace=FALSE)

cdc_train <- cdc[-ind, vars]
cdc_test <- cdc[ind, vars]

nrow(cdc_train)
```

```
## [1] 2349
```

```
nrow(cdc_test)
```

```
## [1] 782
```

```
# variables distribution in the training and test set
vars = c('HeartAttack_rate','obesity','diabetes','smoker',
'no_college','no_internet','income', 'poverty')
par(mfrow=c(3,3))
for(col in vars){
  boxplot(cdc_train[,col],cdc_test[,col],
          col=rainbow(2),ylab=col,main=col,
          names=c("train", "test"))
}
```

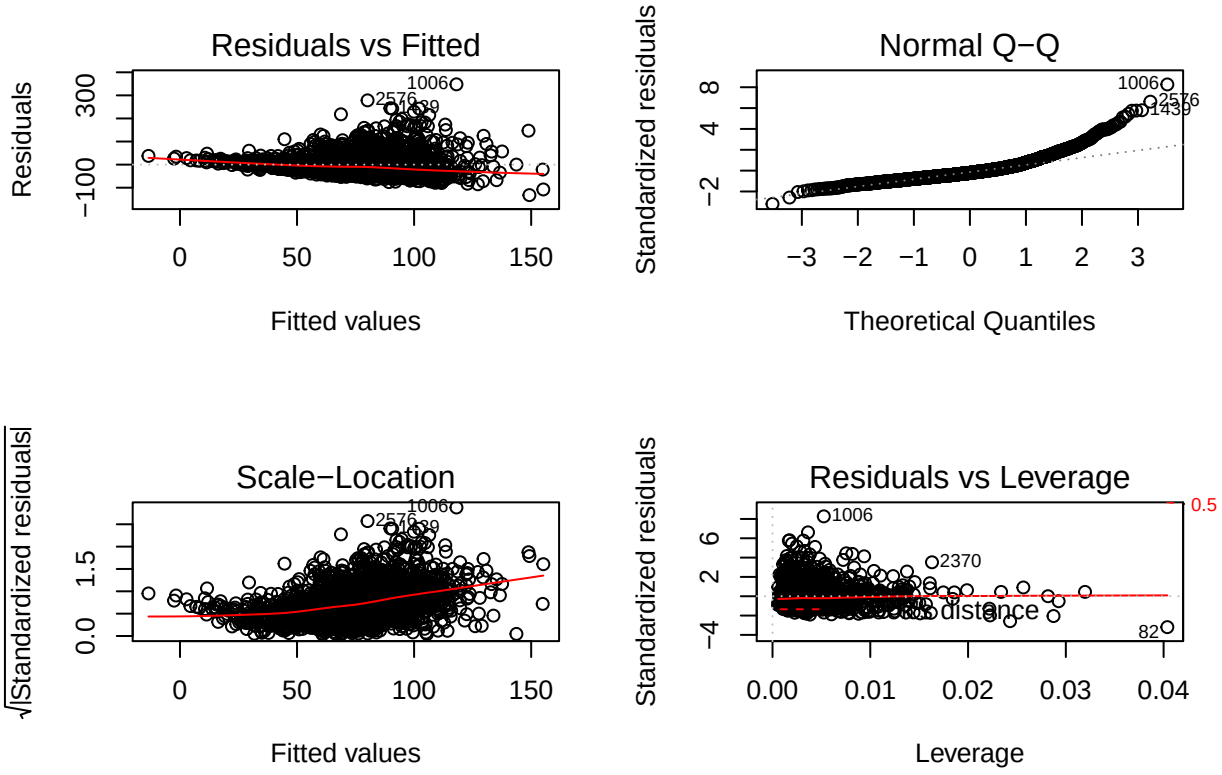



```
# Fitting the first order model with all the variables
fit.1 <- lm(HeartAttack_rate ~ obesity + diabetes + smoker
 + no_college + no_internet + income + poverty, data=cdc_train)

summary(fit.1)
```

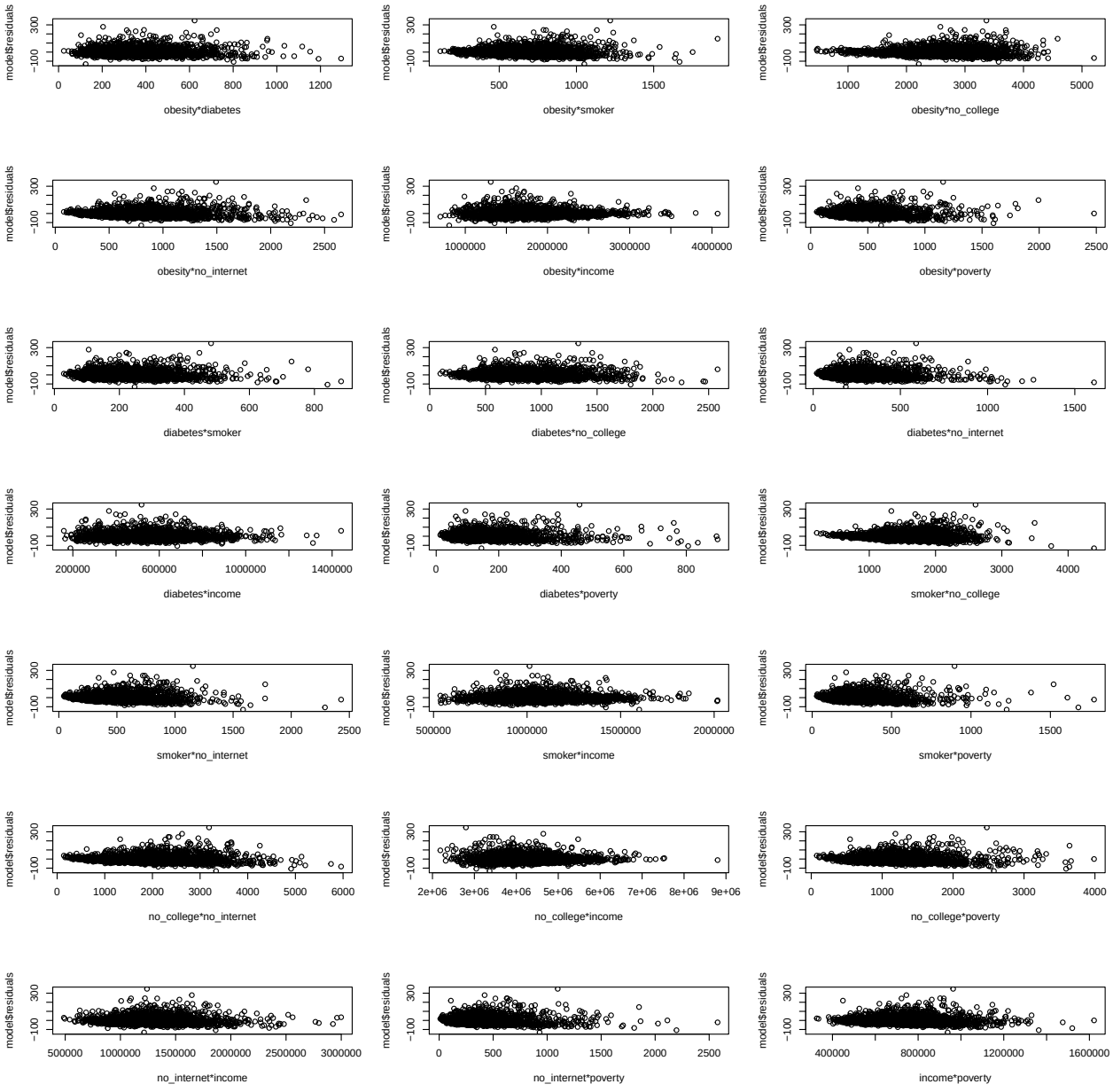
```
##
## Call:
```

```
## lm(formula = HeartAttack_rate ~ obesity + diabetes + smoker +
##     no_college + no_internet + income + poverty, data = cdc_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -132.15  -25.56   -9.13   13.78  347.81
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.511e+01  1.698e+01  -2.657  0.00794 **
## obesity     -1.027e-01  1.919e-01  -0.535  0.59267
## diabetes    -2.235e-01  3.222e-01  -0.694  0.48793
## smoker       1.325e+00  3.369e-01   3.932 8.67e-05 ***
## no_college   9.911e-01  1.597e-01   6.205 6.46e-10 ***
## no_internet  3.056e-01  1.653e-01   1.848  0.06466 .
## income      -4.170e-05  1.196e-04  -0.349  0.72737
## poverty      1.231e+00  2.830e-01   4.351 1.41e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.17 on 2341 degrees of freedom
## Multiple R-squared:  0.1912, Adjusted R-squared:  0.1888
## F-statistic: 79.05 on 7 and 2341 DF,  p-value: < 2.2e-16
```

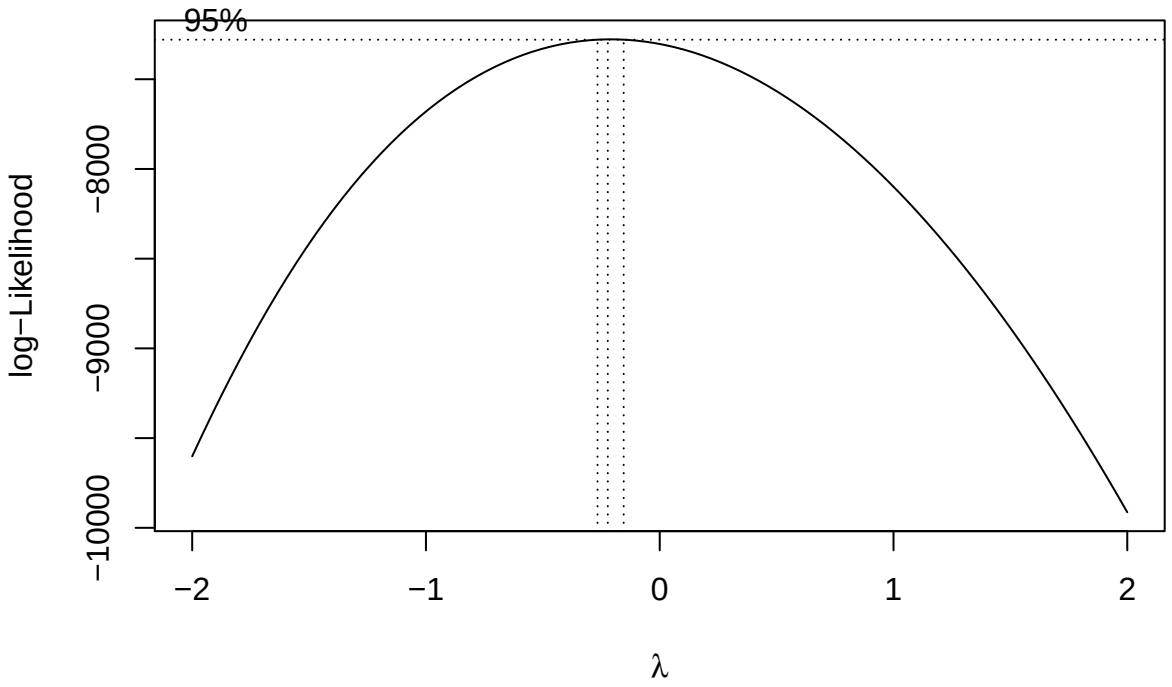
```
# residuals plots for fit.1
par(mfrow=c(2,2))
plot(fit.1)
```



```r
# function 2: create scatterplots of interaction terms and residuals from fit.1

inter_scatter = function(df, dim_vector, model){
  # df: the dataframe we are using, assuming that the responds variable is the first column
  # dim_vector: dimension of the output graphs
  # model: the model we created, where we are getting the residuals from

  par(mfrow = dim_vector)

  for (i in 2:(length(names(df))-1)){
    for (j in (i+1):length(names(df))){
      label = paste(names(df)[i],names(df)[j], sep="*")
      plot(df[,i]*df[,j], model$residuals, xlab = label)
    }
  }
}


dim2 = c(7,3)
inter_scatter(cdc_train, dim2, fit.1)
```



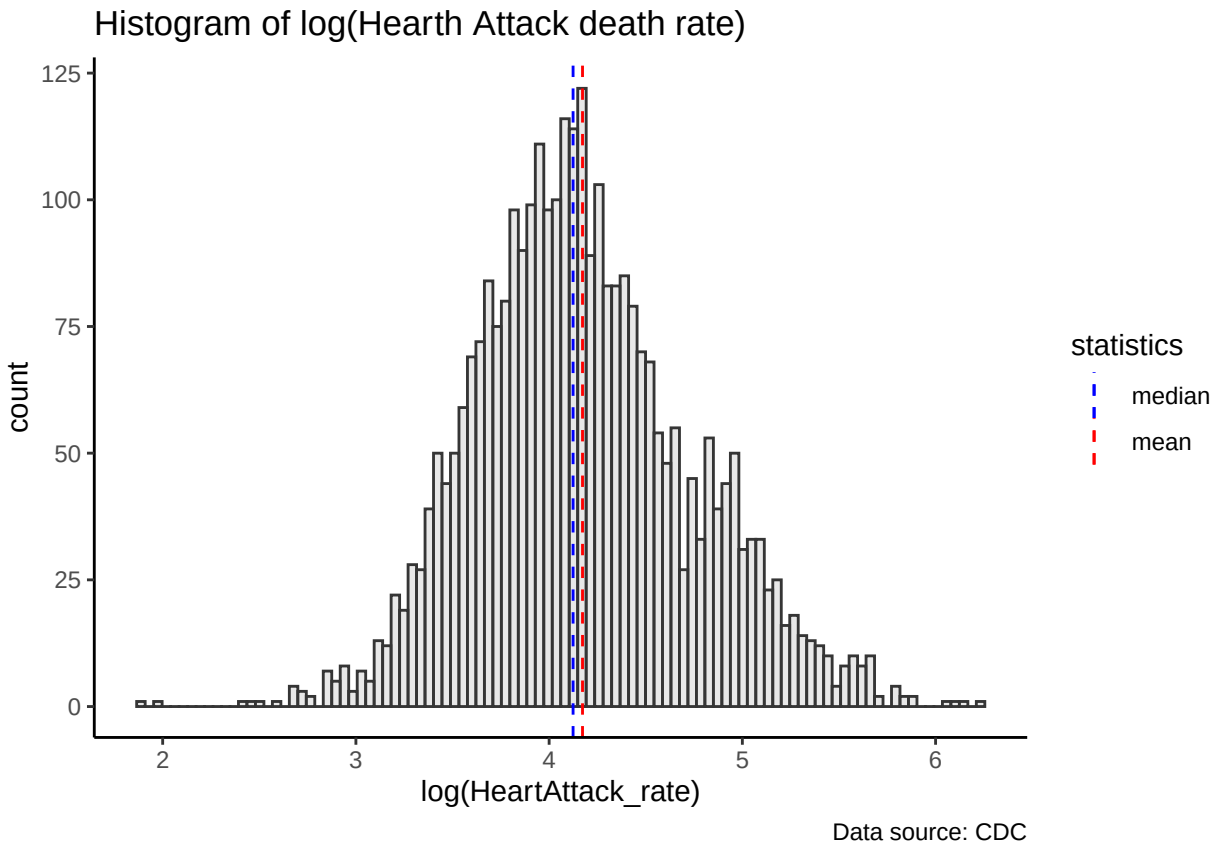

```
# BoxCox procedure
bc = boxcox(fit.1)
```




```
lambda = bc$x[which.max(bc$y)]
lambda
```

```
## [1] -0.2222222
```

```
# Creating a histogram for the log transformation of the response variable
p <- ggplot(data=cdc, aes(x=log(HeartAttack_rate))) +
  geom_histogram(bins=100, fill="#D3D3D3", color='#353535', alpha=0.5) +
  geom_vline(aes(xintercept = mean(log(cdc$HeartAttack_rate)), color='mean'), linetype="dashed") +
  geom_vline(aes(xintercept = median(log(cdc$HeartAttack_rate)), color='median'), linetype="dashed") +
  scale_color_manual(name="statistics", values =c(median = "blue", mean = "red"))+
  theme_bw() +
  theme(axis.line = element_line(colour = "black"),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_blank(),
    panel.background = element_blank())

# Add titles
p <- p + labs(title = "Histogram of log(Hearth Attack death rate)",
              caption = "Data source: CDC")

p
```




```
# step AIC procedure on traning data
full_model <- lm(log(HeartAttack_rate) ~ obesity + diabetes + smoker
 + no_college + no_internet + income + poverty, data=cdc_train)

null_model <- lm(log(HeartAttack_rate)~ 1, data = cdc_train)

fit.2 <- stepAIC(null_model, scope = list(upper = full_model, lower = ~1), direction = "both", k = 2, t
```

```
summary(fit.2)
```

```
##
## Call:
## lm(formula = log(HeartAttack_rate) ~ no_college + poverty + smoker +
##     income + no_internet, data = cdc_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.06896 -0.30231 -0.03513  0.28619  1.59572
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.661e+00  1.848e-01  14.399  < 2e-16 ***
## no_college   1.663e-02  1.717e-03   9.686  < 2e-16 ***
## poverty      8.627e-03  3.052e-03   2.827  0.00474 **
## smoker       9.160e-03  3.534e-03   2.592  0.00960 **
## income      -2.806e-06  1.310e-06  -2.142  0.03233 *
```

```
## no_internet  2.802e-03  1.810e-03   1.548  0.12184
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4628 on 2343 degrees of freedom
## Multiple R-squared:  0.2529, Adjusted R-squared:  0.2513
## F-statistic: 158.6 on 5 and 2343 DF,  p-value: < 2.2e-16
```

```
# Nonconstant error variance test
bptest(fit.2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  fit.2
## BP = 63.237, df = 5, p-value = 2.6e-12
```

```
# standard errors after correction
fit.2 %>%
    vcovHC() %>%
    diag() %>%
    sqrt()
```

```
##  (Intercept)   no_college      poverty       smoker       income  no_internet
## 1.818838e-01 1.693004e-03 3.155920e-03 4.264769e-03 1.234733e-06 1.981551e-03
```

```
# new standard errors and pvalues
coeftest(fit.2, vcov = vcovHC(fit.2))
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  2.6610e+00  1.8188e-01 14.6304 < 2.2e-16 ***
## no_college   1.6630e-02  1.6930e-03  9.8229 < 2.2e-16 ***
## poverty      8.6269e-03  3.1559e-03  2.7336  0.006312 **
## smoker       9.1602e-03  4.2648e-03  2.1479  0.031826 *
## income      -2.8056e-06  1.2347e-06 -2.2723  0.023161 *
## no_internet  2.8016e-03  1.9816e-03  1.4139  0.157538
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# fit the final model with 4 variables
fit.final <- lm(log(HeartAttack_rate) ~ no_college + poverty + smoker + income, data = cdc_train)

s<- summary(fit.final)
s
```

```
##
## Call:
## lm(formula = log(HeartAttack_rate) ~ no_college + poverty + smoker +
```

```
##     income, data = cdc_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.08341 -0.30534 -0.03385  0.28676  1.60606
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.656e+00  1.848e-01  14.371  < 2e-16 ***
## no_college   1.749e-02  1.626e-03  10.758  < 2e-16 ***
## poverty      1.025e-02  2.868e-03   3.573  0.00036 ***
## smoker       9.193e-03  3.535e-03   2.601  0.00936 **
## income      -3.113e-06  1.295e-06  -2.403  0.01634 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.463 on 2344 degrees of freedom
## Multiple R-squared:  0.2521, Adjusted R-squared:  0.2509
## F-statistic: 197.6 on 4 and 2344 DF,  p-value: < 2.2e-16
```

```
#fit the model with the validation set
valid <-lm(fit.final, cdc_test)
sv <- summary(valid)
sv
```

```
##
## Call:
## lm(formula = fit.final, data = cdc_test)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.19571 -0.34357 -0.02429  0.35208  1.65550
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.021e+00  3.843e-01  10.464  < 2e-16 ***
## no_college   2.138e-03  3.549e-03   0.602   0.5471
## poverty     -1.402e-02  5.457e-03  -2.569   0.0104 *
## smoker       5.136e-02  8.732e-03   5.882 6.02e-09 ***
## income      -1.587e-05  2.760e-06  -5.749 1.29e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5359 on 777 degrees of freedom
## Multiple R-squared:  0.2973, Adjusted R-squared:  0.2937
## F-statistic: 82.18 on 4 and 777 DF,  p-value: < 2.2e-16
```

```
anova(fit.final)
```

```
## Analysis of Variance Table
##
## Response: log(HeartAttack_rate)
##              Df Sum Sq Mean Sq  F value  Pr(>F)
```

```
## no_college    1 150.13 150.127 700.4169 < 2e-16 ***
## poverty       1  16.68  16.682  77.8293 < 2e-16 ***
## smoker        1   1.34   1.344   6.2700 0.01235 *
## income        1   1.24   1.238   5.7745 0.01634 *
## Residuals  2344 502.41   0.214
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Data only from california
cdc_ca <- filter(cdc, grepl("(CA)", display_name, ignore.case = FALSE))

nrow(cdc_ca)
```

```
## [1] 58
```

```
head(cdc_ca)
```

```
##   cnty_fips      display_name HeartAttack_rate       theme_range obesity
## 1      6001   "Alameda, (CA)"             37.2  0.0 - 40.5 (648)    22.0
## 2      6003    "Alpine, (CA)"             44.5 40.6 - 54.6 (639)    16.8
## 3      6005    "Amador, (CA)"             50.5 40.6 - 54.6 (639)    29.8
## 4      6007     "Butte, (CA)"             45.0 40.6 - 54.6 (639)    30.3
## 5      6009 "Calaveras, (CA)"             47.0 40.6 - 54.6 (639)    28.1
## 6      6011    "Colusa, (CA)"             45.2 40.6 - 54.6 (639)    38.8
##   diabetes smoker no_college no_internet income poverty
## 1      6.7   11.3       52.6        11.1 108000     8.9
## 2      4.6   16.9       65.5        17.4  58000    17.2
## 3     14.2   14.6       80.7        15.4  63000     9.8
## 4      9.3   15.3       72.8        14.6  58000    16.1
## 5      7.4   13.9       81.7        17.7  68000    12.1
## 6     14.7   15.3       85.0        19.0  59000    12.0
```

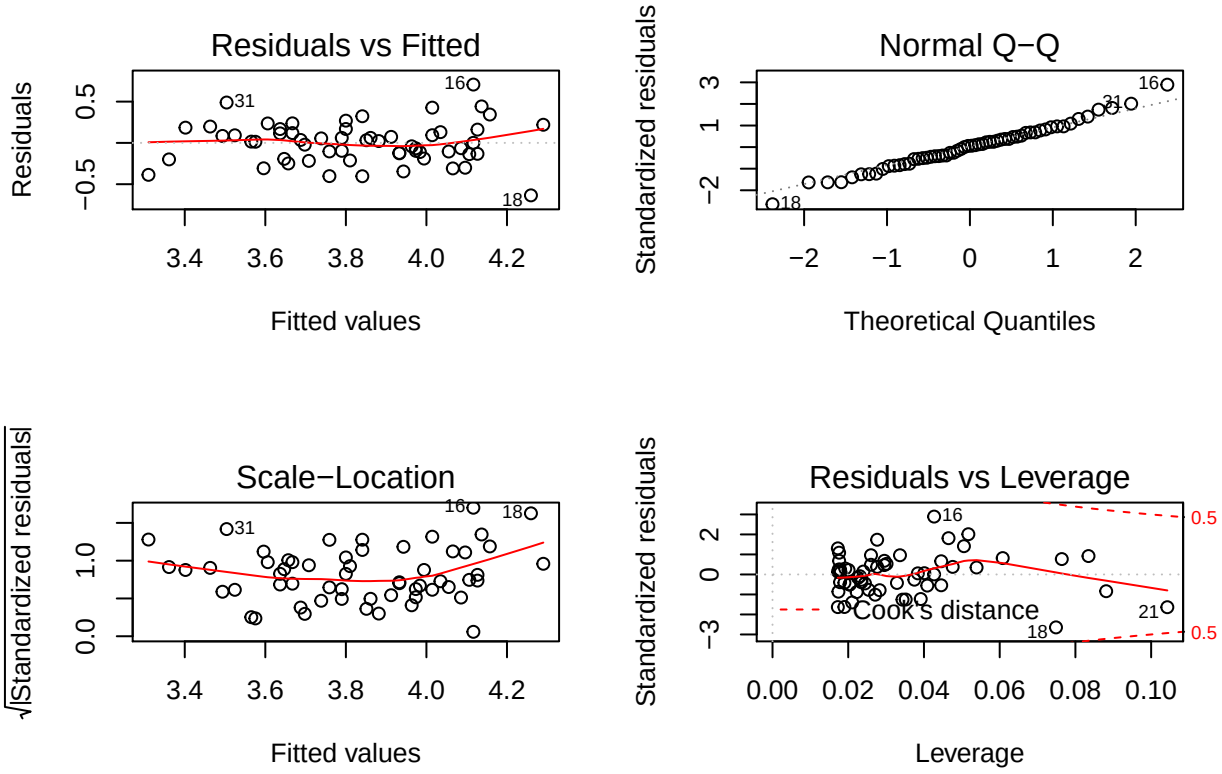
```
# finding the optimal model for the counties in california
full_model.ca =
  lm(log(HeartAttack_rate) ~
    obesity + diabetes + smoker + no_college + no_internet + income + poverty,
    data=cdc_ca)

null_model.ca <- lm(log(HeartAttack_rate)~ 1, data = cdc_ca)

fit.ca <- stepAIC(null_model.ca, scope = list(upper = full_model.ca, lower = ~1), direction = "both", k
```

```
summary(fit.ca)
```

```
##
## Call:
## lm(formula = log(HeartAttack_rate) ~ smoker, data = cdc_ca)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6352 -0.1351  0.0146  0.1533  0.7061
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.37054    0.20832  11.380 3.44e-16 ***
## smoker       0.10211    0.01436   7.111 2.27e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2496 on 56 degrees of freedom
## Multiple R-squared:  0.4745, Adjusted R-squared:  0.4651
## F-statistic: 50.57 on 1 and 56 DF,  p-value: 2.266e-09
```

```
par(mfrow=c(2,2))
plot(fit.ca)
```

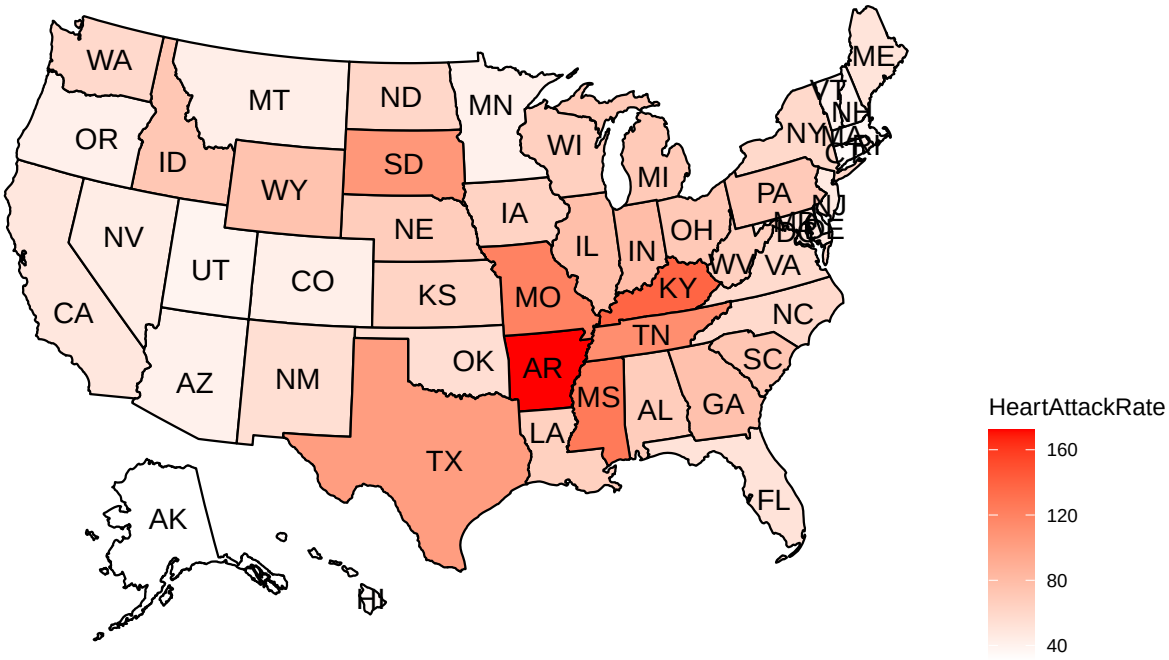



```
# Creating column State and aggregating states by the mean
cdc <- cdc %>%
  extract(display_name, c('State'), regex = "([(]\\w+[)])", remove = FALSE)

cdc$State <- gsub("[(]","",as.character(cdc$State))
cdc$State <- gsub("[)]","",as.character(cdc$State))

ha_by_state <- aggregate(cdc[,c(4,6:12)], list(cdc$State), mean)

names(ha_by_state) <- c('state', 'HeartAttackRate', 'obesity', 'diabetes','smoker','no_college','no_int

head(ha_by_state)
```

```
##   state HeartAttackRate  obesity  diabetes   smoker no_college no_internet
```

```
## 1    AK        31.37619 32.56190  8.147619 25.06190   76.64286    22.92857
## 2    AL        68.19851 38.20299 14.868657 21.75075   81.94776    30.88657
## 3    AR       172.08400 36.42667 13.084000 23.55200   83.48267    32.77733
## 4    AZ        41.84000 31.29333 10.266667 17.90000   79.66000    25.69333
## 5    CA        49.07931 27.08966  8.674138 14.32759   72.70862    16.55690
## 6    CO        43.47656 24.46562  6.885937 16.75781   68.07812    19.02188
##     income  poverty
## 1 67000.00 13.59048
## 2 46179.10 18.66418
## 3 44133.33 18.86000
## 4 51533.33 17.94667
## 5 71086.21 13.02069
## 6 62656.25 12.88906
```

```r
# plotting usmap colored by heart attack death rate
plot_usmap(data = ha_by_state, values = "HeartAttackRate", labels = T) +
  scale_fill_continuous(low = "white", high = "red", name = "HeartAttackRate") +
  theme(legend.position = "right")
```




```r
# A little interaction game

# rank the dataframe by the heart attack death rate
state_ordered = ha_by_state[order(ha_by_state$HeartAttackRate),]

# average death rate in your state

checkratebystate = function(state_abrv){
  if (state_abrv %in% state_ordered$state){
    rank_number = which(state_ordered$state == state_abrv)
    percentage = percent((51 - rank_number)/51, accuracy = 1)

    str = "In terms of low heart attack rate, your state ranks number"
```

```
    str2 = "out of all 50 states and the District of Columbia. It is better than"
    str3 = "of the states."

    print(paste(str,rank_number,str2,percentage,str3,sep = " "))

  } else {
    print("That's not an abbreviation of state. Make sure you only input 2 capital letters as strings.")
  }

}

# examples:
checkratebystate("CA")
```

```
## [1] "In terms of low heart attack rate, your state ranks number 19 out of all 50 states and the Distr
```

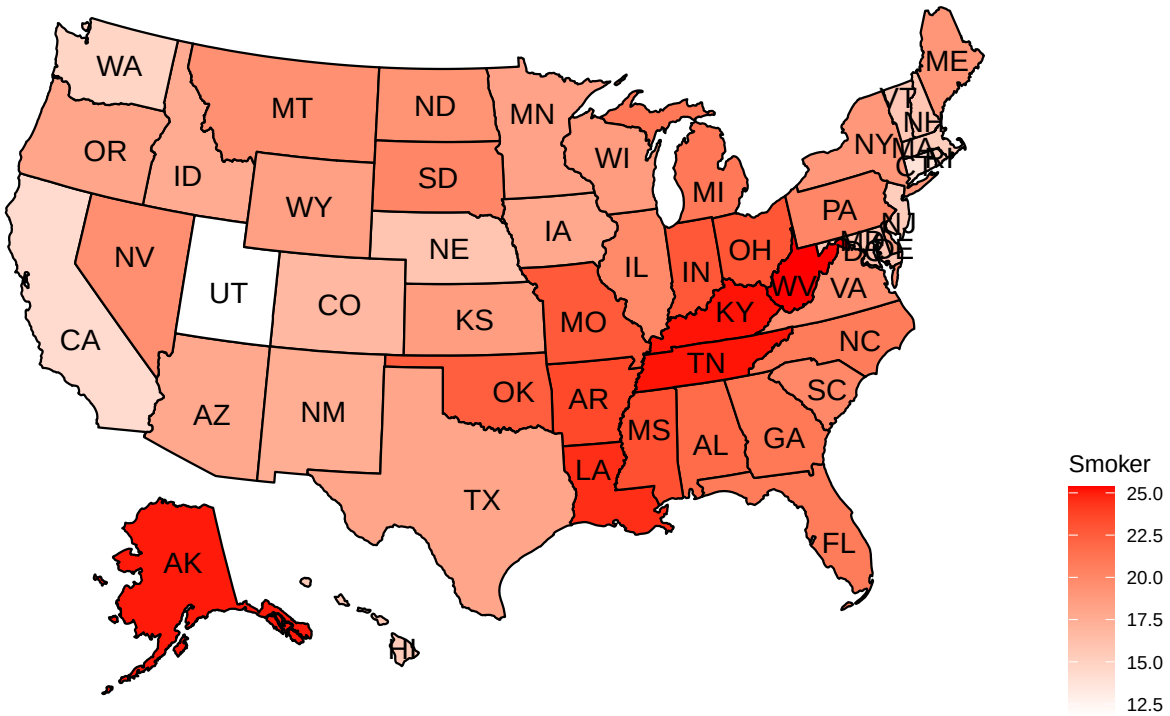
```
checkratebystate("AR")
```

```
## [1] "In terms of low heart attack rate, your state ranks number 51 out of all 50 states and the Distr
```

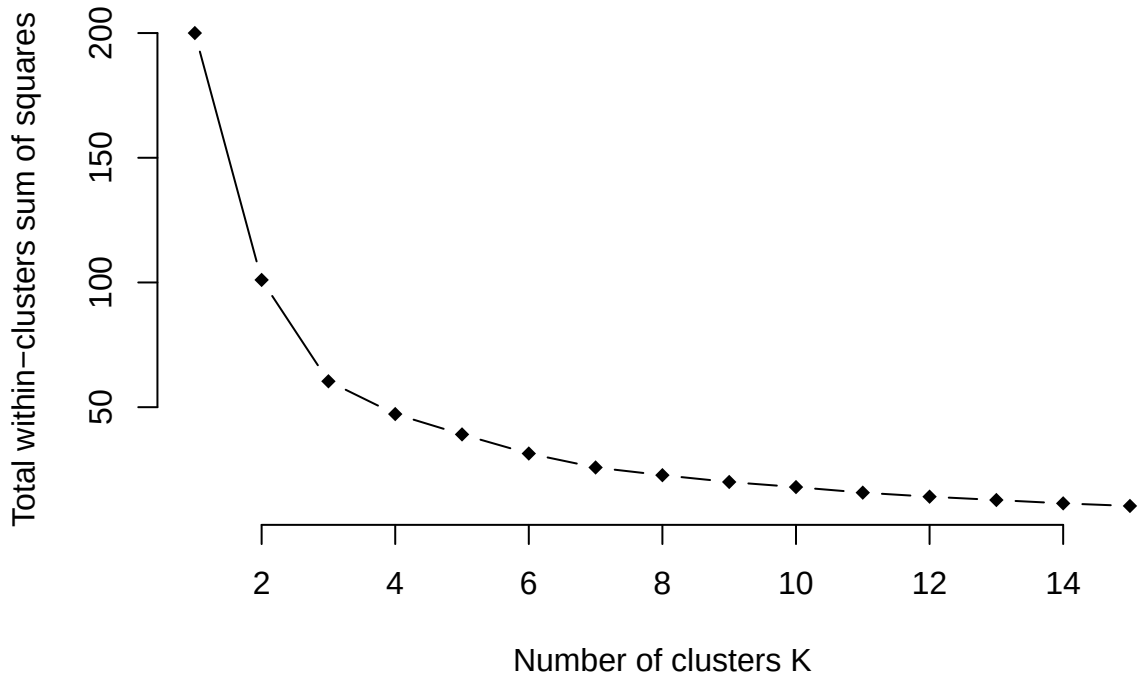
```
checkratebystate("wa")
```

```
## [1] "That's not an abbreviation of state. Make sure you only input 2 capital letters as strings."
```

```
# plotting US map colored by current smoker prevalence
plot_usmap(data = ha_by_state, values = "smoker", labels = T) +
  scale_fill_continuous(
    low = "white", high = "red", name = "Smoker", label = scales::comma
  ) + theme(legend.position = "right")
```



```r
# clustering analysis - elbow method
set.seed(10)
cluster_vars = c('smoker','no_college','income', 'poverty')
k.max <- 15
data <- scale(ha_by_state[,cluster_vars])
wss <- sapply(1:k.max,
              function(k){kmeans(data, k, nstart=50,iter.max = 50)$tot.withinss})
plot(1:k.max, wss,
     type="b", pch = 18, frame = FALSE,
     xlab="Number of clusters K",
     ylab="Total within-clusters sum of squares")
```



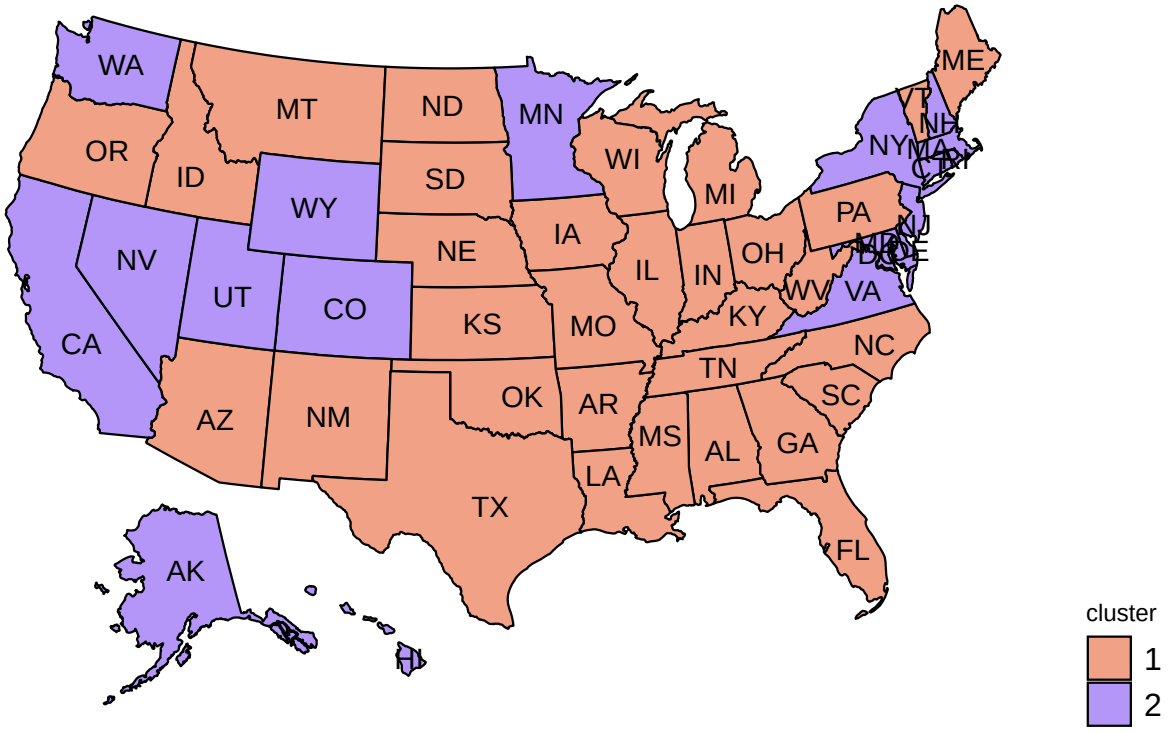

```r
# Evaluation number of states on each cluster
km2 <- kmeans(ha_by_state[,cluster_vars],2,nstart=50,iter.max = 50)
km3 <- kmeans(ha_by_state[,cluster_vars],3,nstart=50,iter.max = 50)


table(km2$cluster)
```

```
##
##  1  2
## 19 32
```

```r
table(km3$cluster)
```

```
##
##  1  2  3
##  7 21 23
```

```
# plotting the US map clustered
set.seed(10)
k = 2
km.out <- kmeans(ha_by_state[,cluster_vars],k,nstart=30)

k_colors = c('1' = "#F1A286", '2' = "#B396F8",'3' = "#75FA4C", '4' = "#F9DAAC")
ha_by_state['cluster'] <- factor(km.out$cluster)
plot_usmap(data = ha_by_state, values = "cluster", labels = T) +
  scale_fill_manual(values = k_colors[1:k], name = "cluster") + theme(legend.position = "right", legend
```




```
# Cluster centroids, heart attack rate added after the clustering process
aggregate(ha_by_state[,c('HeartAttackRate',cluster_vars)], list(ha_by_state$cluster), mean)
```

```
##   Group.1 HeartAttackRate   smoker no_college   income  poverty
## 1       1        75.95239 20.54512   78.70732 53051.65 14.99312
## 2       2        45.45372 16.75011   68.34345 71253.82 11.16937
```

```
# bootstraping mean of cluster 1
set.seed(10)
n=nrow(ha_by_state[ha_by_state$cluster==1,])
c=ncol(ha_by_state[,c('HeartAttackRate',cluster_vars)])
B=1000
mean.c1=matrix(0, B,c)
for (b in 1:B){
  index=sample(n, replace = TRUE)
  mean.c1[b, ]= apply(ha_by_state[ha_by_state$cluster==1,c('HeartAttackRate',cluster_vars)][index,], MA
}
```

```r
# bootstraping mean of cluster 2
set.seed(10)
n=nrow(ha_by_state[ha_by_state$cluster==2,])
c=ncol(ha_by_state[,c('HeartAttackRate',cluster_vars)])
B=1000
mean.c2=matrix(0, B,c)
for (b in 1:B){
  index=sample(n, replace = TRUE)
  mean.c2[b, ]= apply(ha_by_state[ha_by_state$cluster==2,c('HeartAttackRate',cluster_vars)][index,], MA
}
```

```r
# generating confidence interval
mean_confint <- function(x){
  return(paste(round(mean(x),2), " [", round(quantile(x, 0.025),2),',',round(quantile(x, 0.975),2),']',
}
table.c <- aggregate(ha_by_state[,c('HeartAttackRate',cluster_vars)], list(ha_by_state$cluster), mean)

table.c <- as.data.frame(as.matrix(t(table.c[,-1])))
names(table.c) <- c('Cluster 1', 'Cluster 2')

table.c$'Cluster 1'<- apply(mean.c1, MARGIN=2, FUN=mean_confint)
table.c$'Cluster 2' <- apply(mean.c2, MARGIN=2, FUN=mean_confint)

table.c
```

```
##                                  Cluster 1                   Cluster 2
## HeartAttackRate          76.11 [66.53,86.35]           45.53 [40.8,50.68]
## smoker                    20.55 [19.68,21.5]          16.75 [15.53,18.12]
## no_college               78.72 [77.41,79.84]             68.33 [64,71.94]
## income          53049.57 [51208.49,54792.72] 71282.89 [67882.11,74975.22]
## poverty                  14.99 [13.92,16.21]          11.17 [10.37,11.97]
```