Đồ Án Nhóm 1

Thanh Thảo - Bích Trâm

2024-07-23

Contents

ĐỌC DỮ LIỆU **3**

MÔ TẢ DỮ LIỆU **3**

KIỂM TRA DỮ LIỆU **3**
Nhận xét: . . . 4

LÀM SẠCH DỮ LIỆU **4**
Nhận xét: . . . 4

KIỂM TRA OUTLIER **4**
Nhận xét: . . . 8

DỮ LIỆU SAU KHI LÀM SẠCH **9**

CHIA DỮ LIỆU **12**

KIỂM TRA TƯƠNG QUAN **12**

KIỂM TRA ĐA CỘNG TUYẾN **13**
Mô hình đầy đủ biến . . . 13

Mô hình không có biến DISPLACEMENT **14**

Mô hình không có biến DISPLACEMENT, WEIGHT **14**

Mô hình không có biến DISPLACEMENT, WEIGHT, HORSEPOWER **14**

XÂY DỰNG MÔ HÌNH **14**
Nhận xét: . . . 15

SO SÁNH MÔ HÌNH XÂY DỰNG VỚI MÔ HÌNH TẠO BẰNG PHƯƠNG PHÁP STEPWISE **16**
Nhận xét: . . . 17
TÓM LẠI: . . . 17

KIỂM TRA PHẦN DƯ **17**
Nhận xét: . . . 17

DỰ ĐOÁN **17**

BẢO HIỂM **19**

Bộ dữ liệu: CSM **23**
MÔ TẢ DỮ LIỆU . . . 23

ĐỌC DỮ LIỆU 23
TIỀN XỬ LÝ DỮ LIỆU 23

ĐỌC DỮ LIỆU

```
autoMpgDataOrg = read.csv(here("data", "auto-mpg_data.csv"), header=FALSE)

autoMpgDataOrg = rename(autoMpgDataOrg,
                        "mpg"="V1", "cylinders" = "V2", "displacement" = "V3",
                        "horsepower" = "V4", "weight" = "V5", "acceleration" = "V6",
                        "model year" = "V7", "origin" = "V8", "car name" = "V9")

rowAmount = dim(autoMpgDataOrg)[1]
```

MÔ TẢ DỮ LIỆU

V1 - mpg: continuous
V2 - cylinders: multi-valued discrete - xi lanh
V3 - displacement: continuous - dung tích xi lanh
V4 - horsepower: continuous - mã lực
V5 - weight: continuous - trọng lượng
V6 - acceleration: continuous - tăng tốc
V7 - model year: multi-valued discrete - năm sx
V8 - origin: multi-valued discrete - nguồn gốc
V9 - car name: string (unique for each instance) - tên xe

KIỂM TRA DỮ LIỆU

```
str(autoMpgDataOrg)
```

```
## 'data.frame':    398 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : int  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : chr  "130" "165" "150" "150" ...
##  $ weight      : int  3504 3693 3436 3433 3449 4341 4354 4312 4425 3850 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ model year  : int  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ car name    : chr  "chevrolet chevelle malibu" "buick skylark 320" "plymouth satellite" "amc rebel s
```

```
unique(autoMpgDataOrg$horsepower)
```

```
##  [1] "130" "165" "150" "140" "198" "220" "215" "225" "190" "170" "160" "95"
## [13] "97"  "85"  "88"  "46"  "87"  "90"  "113" "200" "210" "193" "?"   "100"
## [25] "105" "175" "153" "180" "110" "72"  "86"  "70"  "76"  "65"  "69"  "60"
## [37] "80"  "54"  "208" "155" "112" "92"  "145" "137" "158" "167" "94"  "107"
## [49] "230" "49"  "75"  "91"  "122" "67"  "83"  "78"  "52"  "61"  "93"  "148"
## [61] "129" "96"  "71"  "98"  "115" "53"  "81"  "79"  "120" "152" "102" "108"
## [73] "68"  "58"  "149" "89"  "63"  "48"  "66"  "139" "103" "125" "133" "138"
## [85] "135" "142" "77"  "62"  "132" "84"  "64"  "74"  "116" "82"
```

```
isTRUE(duplicated(autoMpgDataOrg))
```

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

Nhận xét:

- horsepower: không đúng kiểu dữ liệu. Trong mô tả dữ liệu là biến liên tục, trong data là kiểu chuỗi
- horsepower: có dữ liệu bị thiếu (?)
- không có dòng dữ liệu trùng

LÀM SẠCH DỮ LIỆU

```
missingCounter = count(filter(autoMpgDataOrg, autoMpgDataOrg$horsepower == "?"))
autoMpgData = subset(autoMpgDataOrg, autoMpgDataOrg$horsepower!="?")[,-9]

autoMpgData$horsepower = as.integer(autoMpgData$horsepower)
autoMpgData$`model year` = as.factor(autoMpgData$`model year`)
autoMpgData$origin = as.factor(autoMpgData$origin)
```

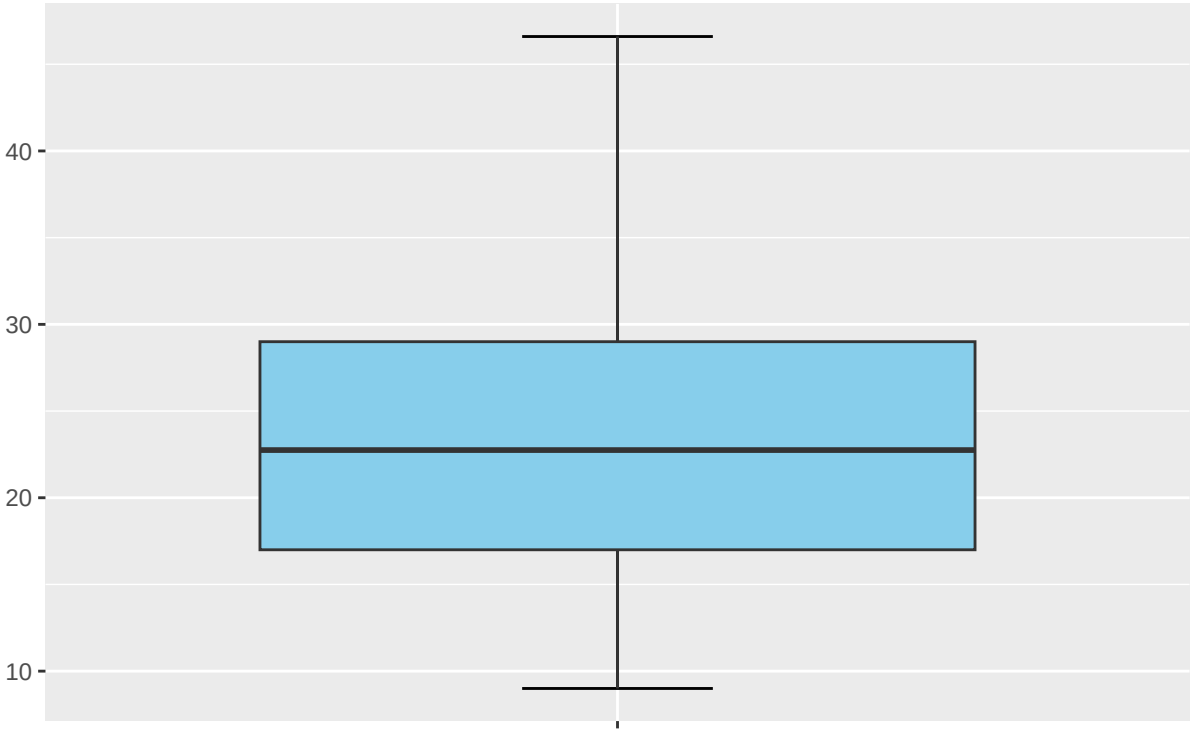
Nhận xét:

- Chuyển kiểu dữ liệu của biến horsepower sang integer do horsepower là biến liên tục nhưng trong bộ dữ liệu là kiểu chuỗi
- Các biến "model year", "origin" là biến định tính nên chuyển sang dạng factor
- Có 1 biến "car name" không có giá trị dử dụng trong thống kê => loại biến "car name" ra khỏi bộ dữ liệu

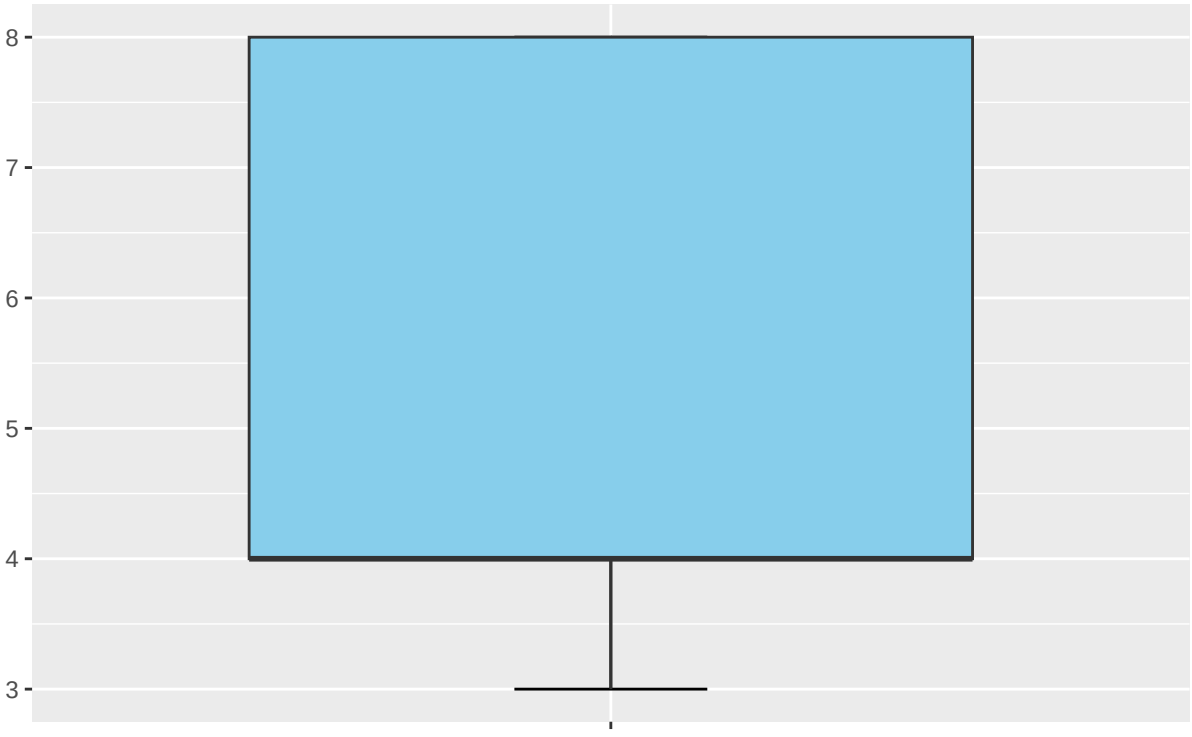
KIỂM TRA OUTLIER

```
drawBoxPlot <- function(){
  ouliersIndexList = list()

  for (i in 1:length(autoMpgData)){
    if(names(autoMpgData[i]) != "model year" && names(autoMpgData[i]) != "origin"){
      boxPlot =
        ggplot(autoMpgData, aes(x="", y=autoMpgData[[i]])) +
        stat_boxplot(geom="errorbar", width=0.2) +
        xlab("") +
        ylab("") +
        ggtitle(paste("OUTLIERS OF", toupper(names(autoMpgData[i])))) +
        geom_boxplot(fill="skyblue", outlier.colour = "red")

      print(boxPlot)
    }
  }
}

drawBoxPlot()
```

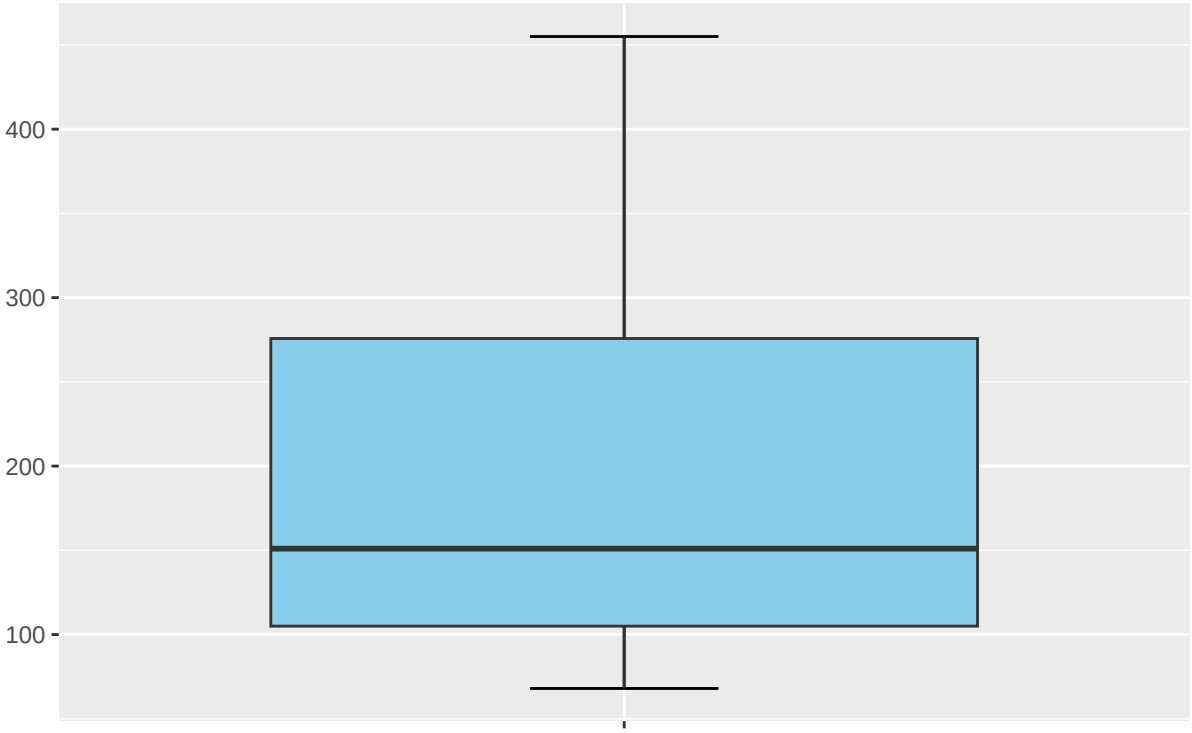
OUTLIERS OF MPG



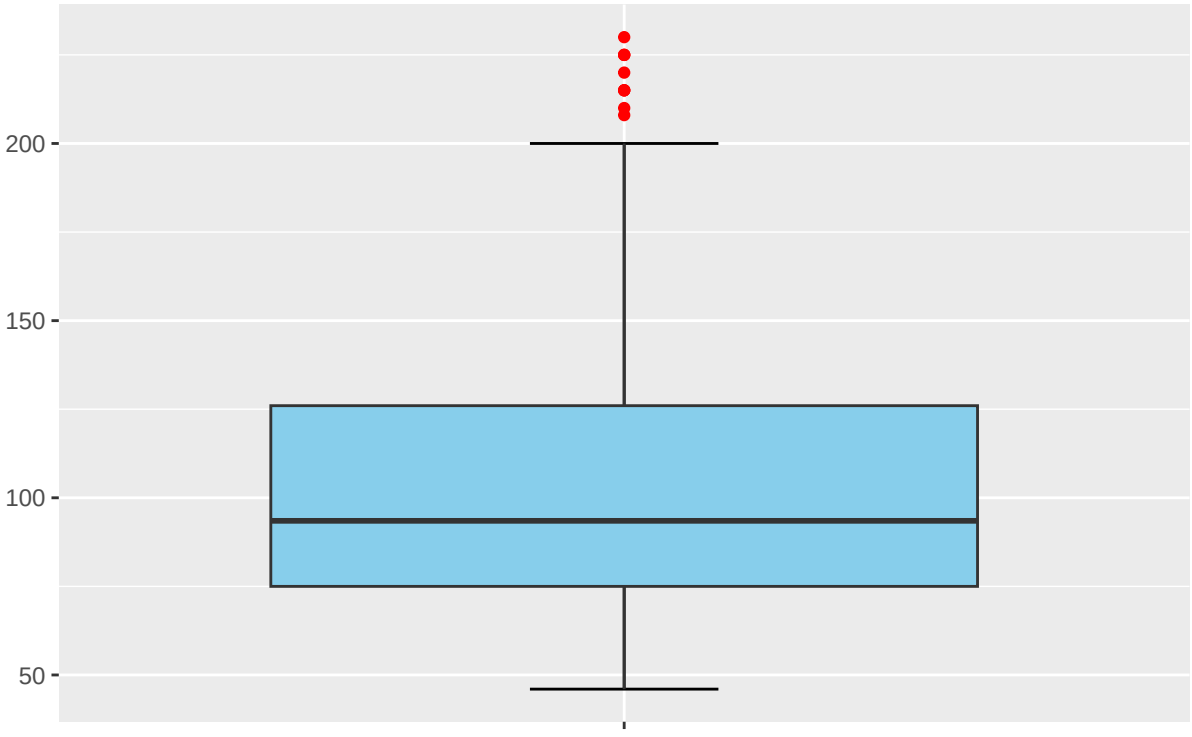
OUTLIERS OF CYLINDERS



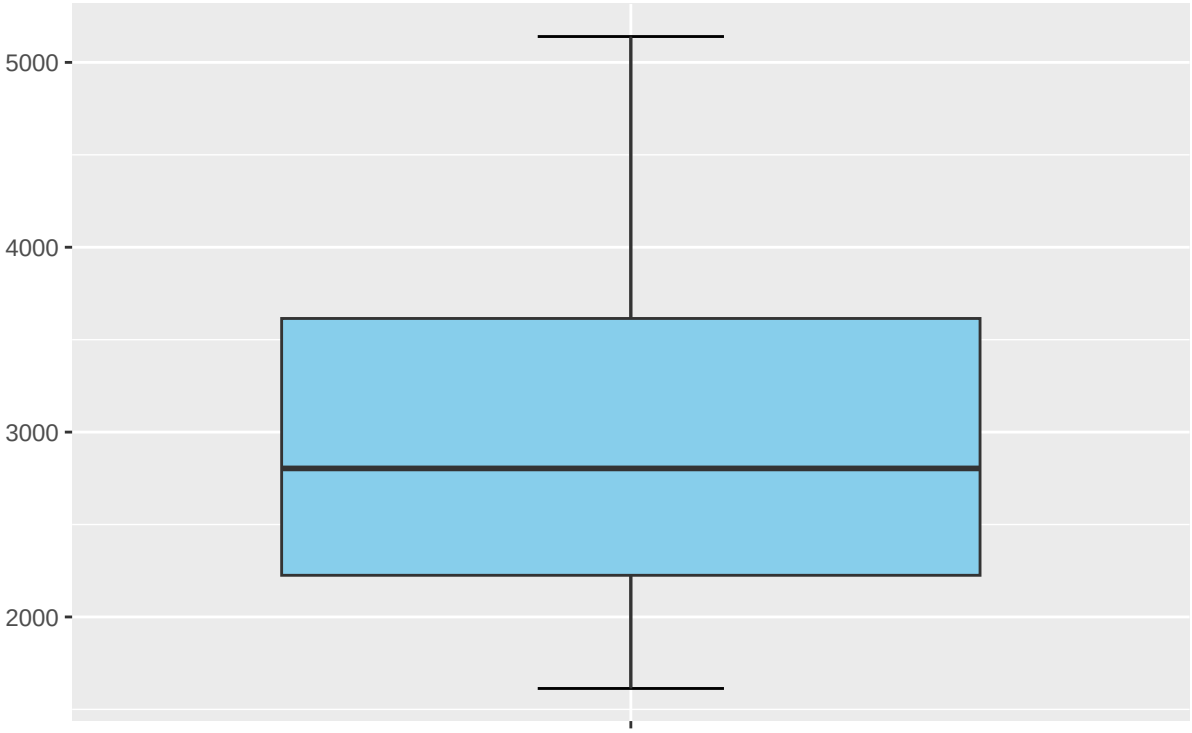
OUTLIERS OF DISPLACEMENT



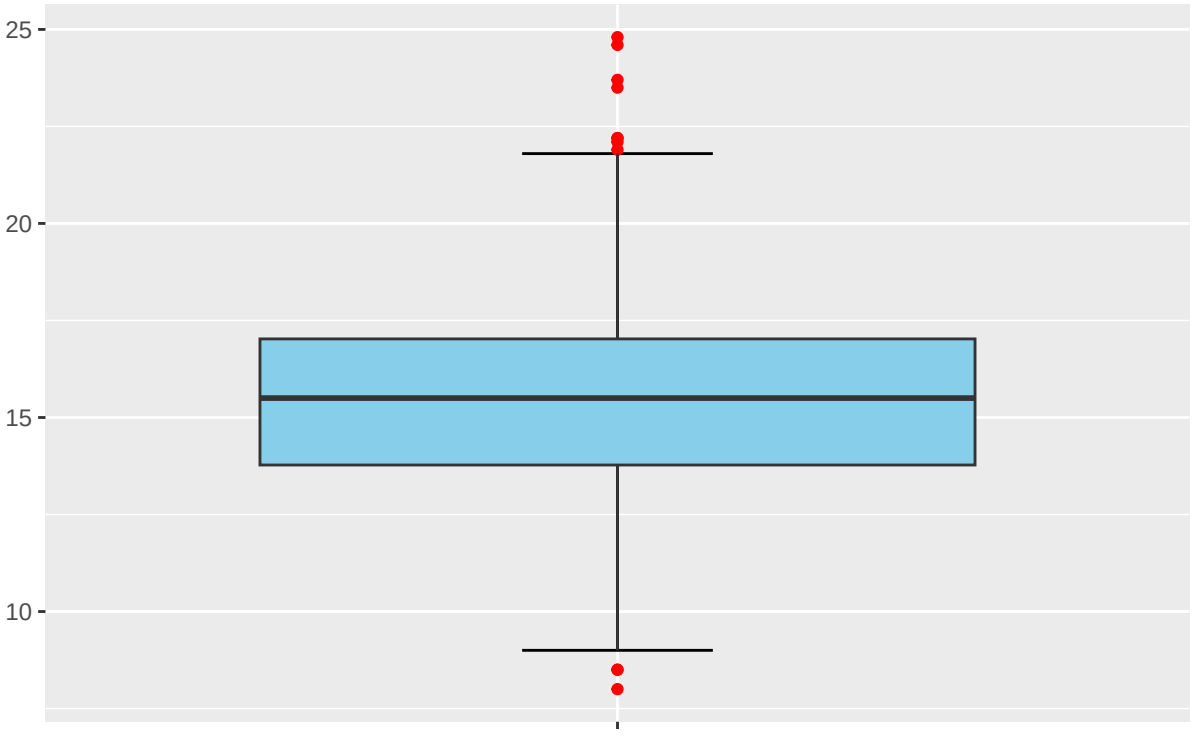
OUTLIERS OF HORSEPOWER



OUTLIERS OF WEIGHT



OUTLIERS OF ACCELERATION



```
findOutliersIndexList <- function(){
  outlierIndexesList = list()
  outlierVariableCounter = 0
  for (i in 1:length(autoMpgData)){
    if(names(autoMpgData[i]) != "model year" && names(autoMpgData[i]) != "origin"){
      quantitleValue = quantile(autoMpgData[[i]])
      upperValue = quantitleValue[4] + (quantitleValue[4]-quantitleValue[2])*1.5
      lowerValue = quantitleValue[2] - (quantitleValue[4]-quantitleValue[2])*1.5

      indexOutlier = which(autoMpgData[[i]] > upperValue |
                             autoMpgData[[i]] < lowerValue)
      if(length(indexOutlier) > 0){
        outlierVariableCounter = outlierVariableCounter + 1
        outlierIndexesList = append(outlierIndexesList, indexOutlier)
      }
    }
  }

  return(outlierIndexesList)
}

findDuplicatedOutlierRow <- function(){
  outliersIndexesList = findOutliersIndexList()

  duplicatedOutlierRow = list()
  for(i in 2:length(outliersIndexesList)){
    for(j in 1:(i-1)){
      if(outliersIndexesList[i] %in% outliersIndexesList[j]){
        duplicatedOutlierRow =
          append(duplicatedOutlierRow, outliersIndexesList[i])
      }
    }
  }

  return(duplicatedOutlierRow)
}

duplicatedOutlierRow = findDuplicatedOutlierRow()
totalRemovingRow = length(duplicatedOutlierRow) + missingCounter
removingPercentage = round(totalRemovingRow*100/rowAmount, 2)

autoMpgData = autoMpgData[-as.vector(unlist(duplicatedOutlierRow)),]
```

Nhận xét:

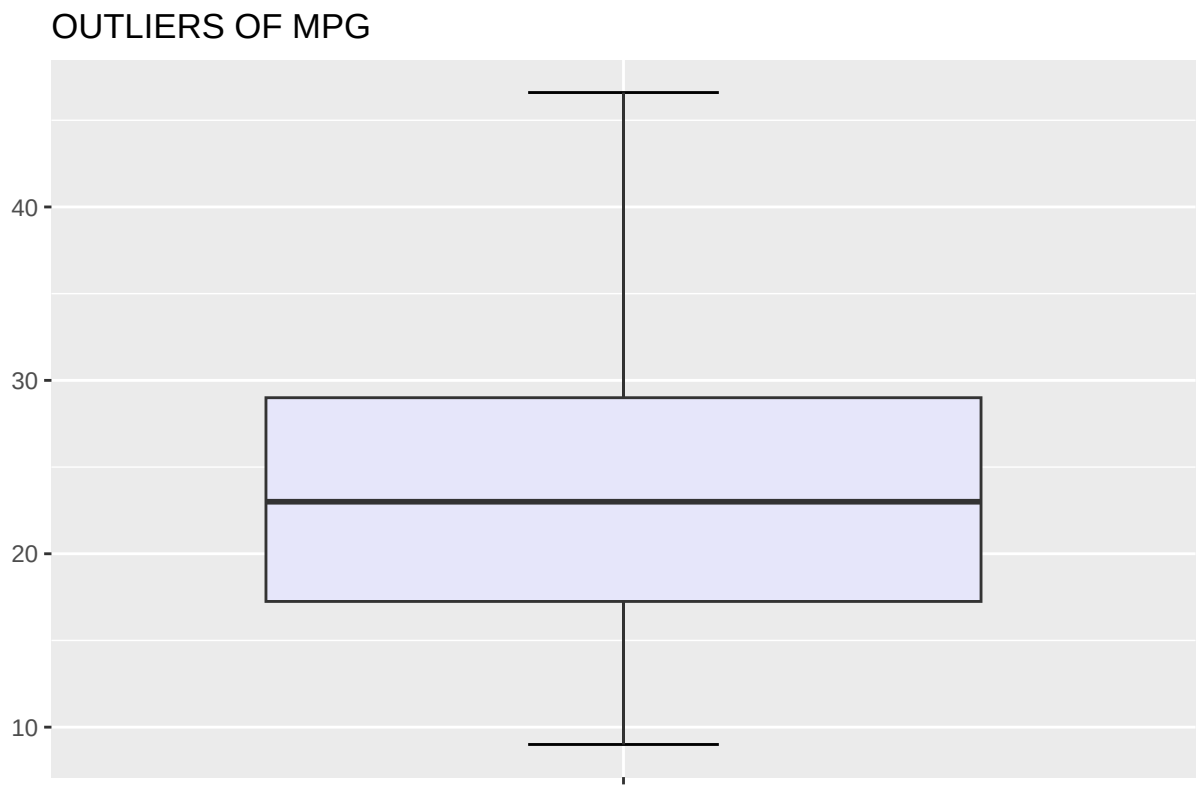
- Có 6 dòng dữ liệu có biến mã lực bị thiếu giá trị. Xử lý bằng cách thay giá trị thiếu bằng trung vị hoặc xóa dòng các dòng dữ liệu. Do số lượng dòng thiếu dữ liệu khá nhỏ so với bộ dữ liệu nên xóa dữ liệu cũng không ảnh hưởng đến kết quả xây dựng mô hình

- Có 1 ví trí dòng mà các biến có giá trị ngoại lai đều có giá trị nằm ở dòng đó

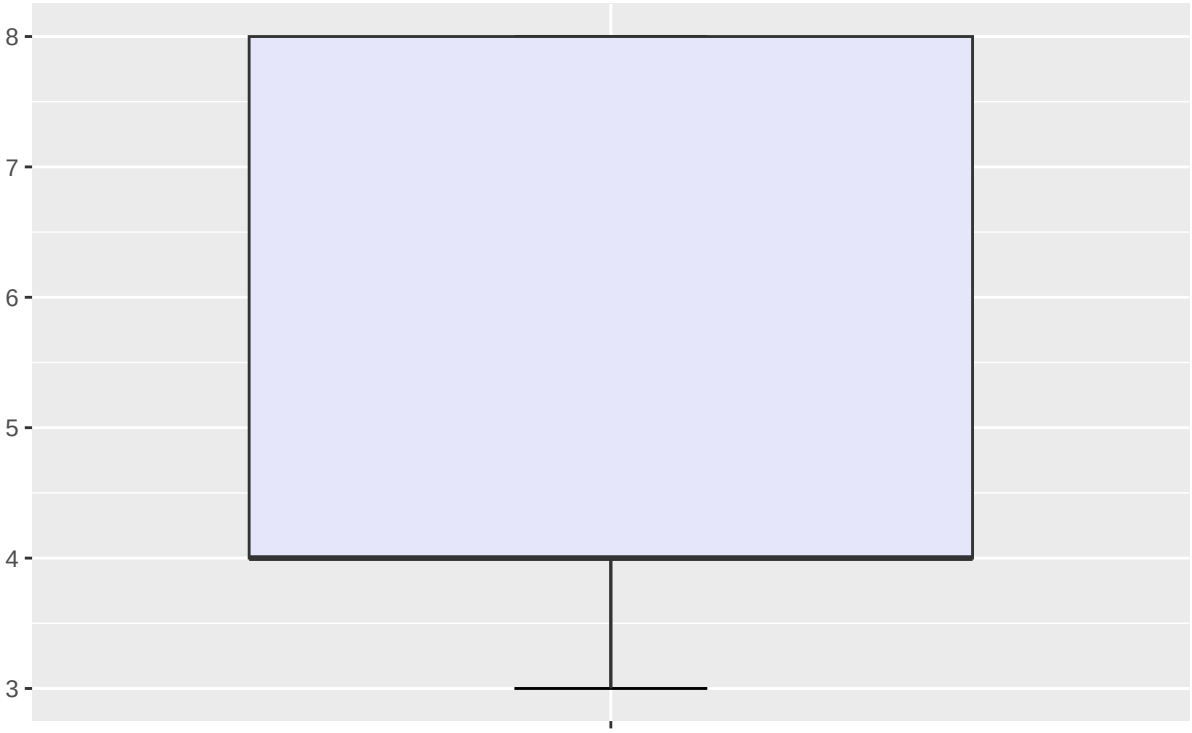
- Tổng số lượng dòng cần xóa chiếm 7 *100/ 398 = 1.76 % bao gồm các dòng dữ liệu bị khuyết và dòng dữ liệu có giá trị ngoại lai ở các biến. Vị trí các dòng đã xóa là: 8 , 33, 127, 331, 337, 355, 375

DỮ LIỆU SAU KHI LÀM SẠCH

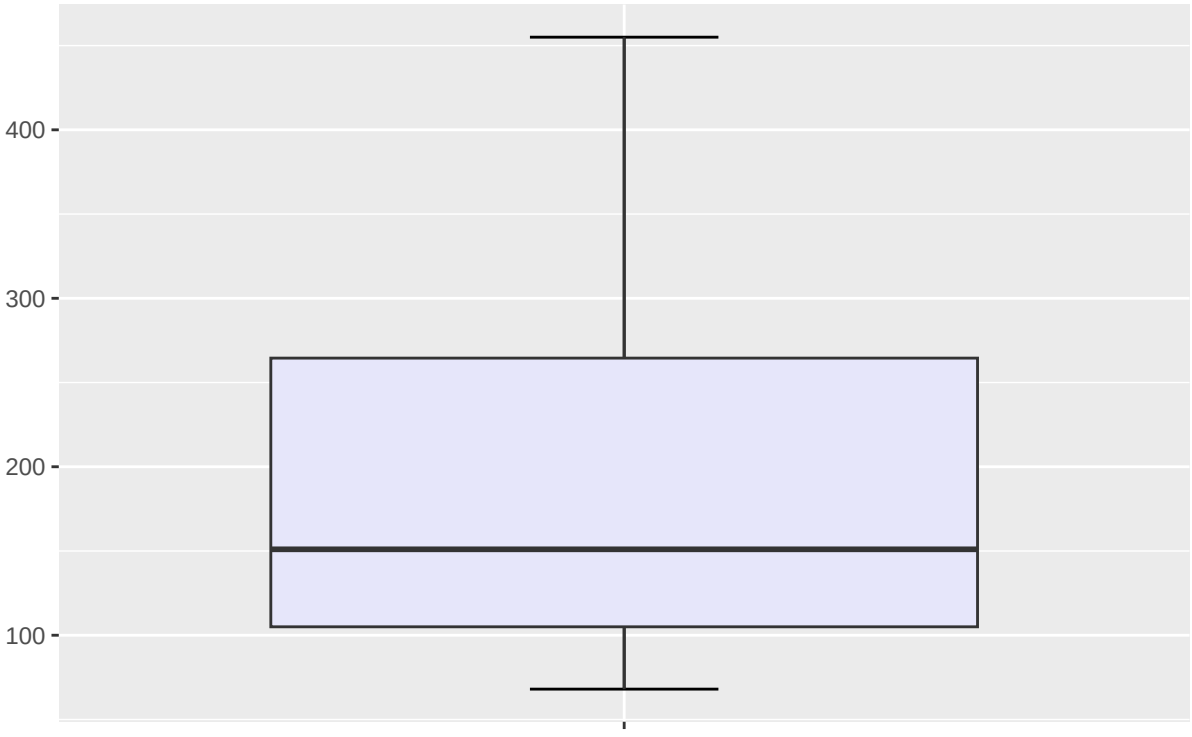
```
drawBoxPlot <- function(){
  ouliersIndexList = list()

  for (i in 1:length(autoMpgData)){
    if(names(autoMpgData[i]) != "model year" && names(autoMpgData[i]) != "origin"){
      boxPlot =
        ggplot(autoMpgData, aes(x="", y=autoMpgData[[i]])) +
        stat_boxplot(geom="errorbar", width=0.2) +
        xlab("") +
        ylab("") +
        ggtitle(paste("OUTLIERS OF", toupper(names(autoMpgData[i])))) +
        geom_boxplot(fill="lavender", outlier.colour = "red")

      print(boxPlot)
    }
  }
}

drawBoxPlot()
```



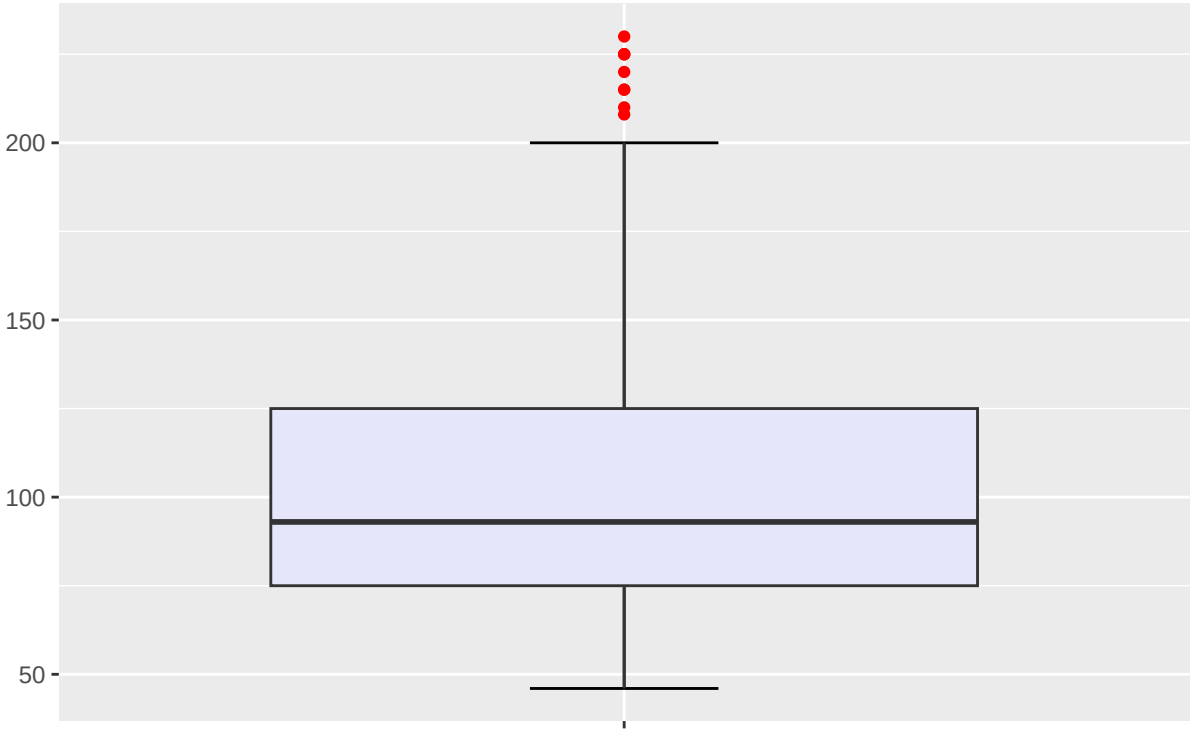
OUTLIERS OF CYLINDERS



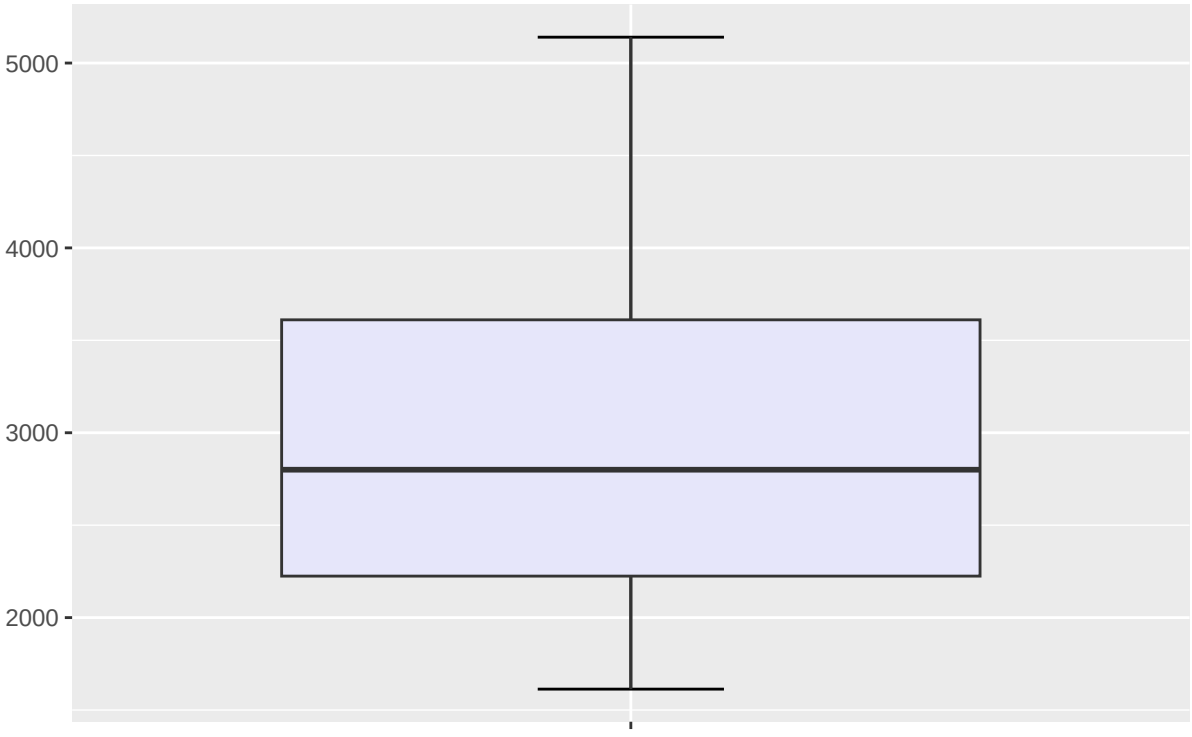
OUTLIERS OF DISPLACEMENT

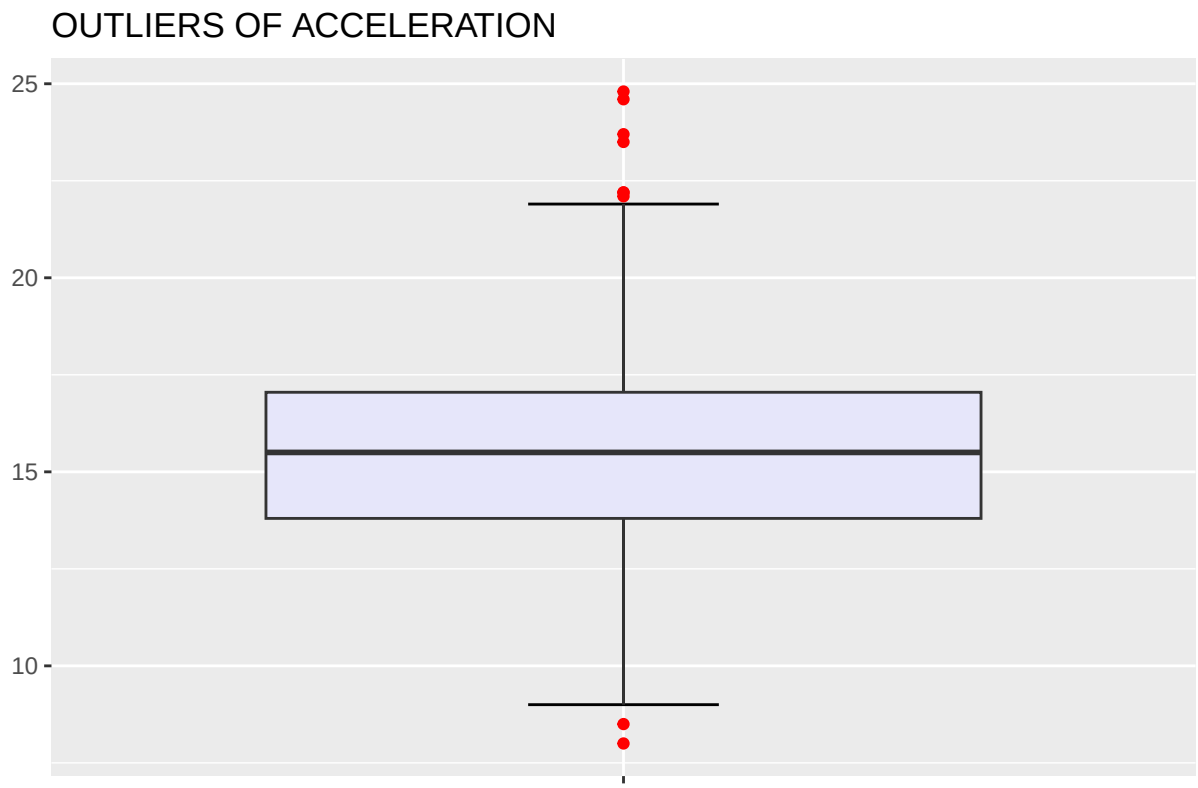


OUTLIERS OF HORSEPOWER



OUTLIERS OF WEIGHT





CHIA DỮ LIỆU

- 80% dữ liệu được chọn ngẫu nhiên dùng để xây dựng mô hình, 20% dữ liệu còn lại dùng để kiểm tra lại mô hình

```
set.seed(123)
trainingSamples = autoMpgData$mpg  %>%  createDataPartition(p = 0.8, list = FALSE)
trainingData  = autoMpgData[trainingSamples, ]
testData = autoMpgData[-trainingSamples, ]
```

KIỂM TRA TƯƠNG QUAN

- Hai biến "model year", "origin" là hai biến định tính nên không sử dụng để kiểm tra tính tương quan

```
cor(trainingData[,1:6])
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7792048   -0.8102289 -0.7706496 -0.8319118
## cylinders    -0.7792048  1.0000000    0.9496186  0.8373671  0.8992735
## displacement -0.8102289  0.9496186    1.0000000  0.8899384  0.9414345
## horsepower   -0.7706496  0.8373671    0.8899384  1.0000000  0.8651272
## weight       -0.8319118  0.8992735    0.9414345  0.8651272  1.0000000
## acceleration  0.4059557 -0.4938165   -0.5269226 -0.6855951 -0.4049508
##              acceleration
## mpg             0.4059557
## cylinders      -0.4938165
## displacement   -0.5269226
## horsepower     -0.6855951
```

```
## weight         -0.4049508
## acceleration    1.0000000
```

```
corrplot.mixed(cor(trainingData[,1:6]), order = 'AOE')
```

##

Nhận xét:
- Các biến có mối tương quan khá mạnh với nhau. Phần lớn đều trên 0.7 nên ta cần phải kiểm tra xem các biến có xảy ra hiện tượng đa cộng tuyến hay không

KIỂM TRA ĐA CỘNG TUYẾN

Mô hình đầy đủ biến

```
model = lm(mpg~., data = trainingData)
vif(model)
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## cylinders    11.212327  1        3.348481
## displacement 24.789802  1        4.978936
## horsepower   11.624247  1        3.409435
## weight       14.815844  1        3.849136
## acceleration  2.872558  1        1.694862
## `model year`  2.113889 12        1.031680
## origin        2.298261  2        1.231260
```

Nhận xét:

- Có hiện tượng đa cộng tuyến xảy ra, mạnh nhất ở biến DISPLACEMENT (GVIF = 24.789802) nên ta loại biến DISPLACEMENT ra khỏi mô hình

Mô hình không có biến DISPLACEMENT

```
model2 = lm(mpg~cylinders+horsepower+weight+acceleration+`model year`+origin, data = trainingData)
vif(model2)
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## cylinders     6.746116  1        2.597329
## horsepower   11.257807  1        3.355266
## weight       11.547966  1        3.398230
## acceleration  2.801580  1        1.673792
## `model year`  2.012512 12        1.029570
## origin        1.996183  2        1.188639
```

Nhận xét:

- Vẫn còn hiện tượng đa cộng tuyến xảy ra, mạnh nhất ở WEIGHT (GVIF = 11.547966) nên ta loại WEIGHT ra khỏi mô hình

Mô hình không có biến DISPLACEMENT, WEIGHT

```
model3 = lm(mpg~cylinders+horsepower+acceleration+`model year`+origin, data = trainingData)
vif(model3)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## cylinders    4.589249  1        2.142253
## horsepower   5.775981  1        2.403327
## acceleration 2.026722  1        1.423630
## `model year` 1.714730 12        1.022723
## origin       1.818132  2        1.161198
```

Nhận xét:

- Vẫn có thể còn hiện tượng đa cộng tuyến xảy ra ở HORSEPOWER (GVIF = 5.775981) nên ta loại HORSEPOWER ra khỏi mô hình

Mô hình không có biến DISPLACEMENT, WEIGHT, HORSEPOWER

```
model4 = lm(mpg~cylinders+acceleration+`model year`+origin, data = trainingData)
vif(model4)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## cylinders    2.227309  1        1.492417
## acceleration 1.422211  1        1.192565
## `model year` 1.480651 12        1.016488
## origin       1.817055  2        1.161026
```

Nhận xét:

- Tất cả các giá trị GVIF đều nhỏ hơn 5 nên ta dừng kiểm tra đa cộng tuyến

XÂY DỰNG MÔ HÌNH

```
summary(model4)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + acceleration + `model year` +
##     origin, data = trainingData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4175  -2.3087  -0.2217   2.2323  14.1194
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    32.753916   2.242091  14.609  < 2e-16 ***
## cylinders      -2.349631   0.182884 -12.848  < 2e-16 ***
## acceleration   -0.009037   0.091799  -0.098 0.921646
## `model year`71  0.624626   1.128044   0.554 0.580184
## `model year`72 -1.221205   1.121612  -1.089 0.277127
## `model year`73 -1.522439   0.998846  -1.524 0.128523
## `model year`74  1.317371   1.131838   1.164 0.245390
## `model year`75 -0.488419   1.111800  -0.439 0.660760
## `model year`76  1.111766   1.083431   1.026 0.305654
## `model year`77  2.726123   1.118326   2.438 0.015368 *
## `model year`78  3.288479   1.055336   3.116 0.002012 **
## `model year`79  5.506420   1.080577   5.096 6.19e-07 ***
## `model year`80  7.904532   1.182340   6.686 1.14e-10 ***
## `model year`81  6.368719   1.141365   5.580 5.42e-08 ***
## `model year`82  8.010543   1.163068   6.887 3.39e-11 ***
## origin2         2.531052   0.673433   3.758 0.000206 ***
## origin3         3.620897   0.658657   5.497 8.31e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.689 on 297 degrees of freedom
## Multiple R-squared:  0.7849, Adjusted R-squared:  0.7733
## F-statistic: 67.75 on 16 and 297 DF,  p-value: < 2.2e-16
```

Nhận xét:

- Các biến acceleration, model year71, model year72, model year74, model year75, model year76 không có ý nghĩa trong thống kê (vì Pr > 0.05) nên ta không đưa vào mô hình

SO SÁNH MÔ HÌNH XÂY DỰNG VỚI MÔ HÌNH TẠO BẰNG PHƯƠNG PHÁP STEPWISE

```
modelComparison = stepAIC(model, direction = "both",trace = FALSE)
summary(modelComparison)
```

```
##
## Call:
## lm(formula = mpg ~ displacement + horsepower + weight + `model year` +
##     origin, data = trainingData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.4403 -1.8445  0.0137  1.6653 11.8703
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)    37.4138347  1.2925765  28.945  < 2e-16 ***
## displacement    0.0133566  0.0062217   2.147 0.032622 *
## horsepower     -0.0198318  0.0116072  -1.709 0.088577 .
## weight         -0.0061908  0.0006615  -9.358  < 2e-16 ***
## `model year`71  0.8447420  0.9585351   0.881 0.378879
## `model year`72  0.1711217  0.9440919   0.181 0.856291
## `model year`73 -0.7863200  0.8384164  -0.938 0.349079
## `model year`74  1.9058457  0.9858504   1.933 0.054165 .
## `model year`75  0.6926657  0.9763151   0.709 0.478592
## `model year`76  1.5980866  0.9467495   1.688 0.092470 .
## `model year`77  3.2765532  0.9531081   3.438 0.000671 ***
## `model year`78  3.1877580  0.9043637   3.525 0.000491 ***
## `model year`79  5.1919267  0.9402891   5.522 7.35e-08 ***
## `model year`80  9.2282220  1.0216315   9.033  < 2e-16 ***
## `model year`81  6.5869794  0.9955703   6.616 1.72e-10 ***
## `model year`82  8.9562255  0.9866275   9.078  < 2e-16 ***
## origin2         2.6666858  0.5899407   4.520 8.94e-06 ***
## origin3         2.5282001  0.5871199   4.306 2.26e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.044 on 296 degrees of freedom
## Multiple R-squared:  0.8541, Adjusted R-squared:  0.8457
## F-statistic: 101.9 on 17 and 296 DF,  p-value: < 2.2e-16
```

```
anova(model4, modelComparison)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ cylinders + acceleration + `model year` + origin
## Model 2: mpg ~ displacement + horsepower + weight + `model year` + origin
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    297 4042.7
## 2    296 2742.5  1    1300.1 140.32 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nhận xét:

1. Mô hình 2 có Pr < 0.05 nên ta chọn mô hinh 2: mpg ~ displacement + weight + model year + origin

2. Ở mô hình 2 có:

- Các biến horsepower, model year71, model year72, model year73, model year74, model year75, model year76 không có nhiều ý nghĩa trong thống kê nên ta không đưa vào mô hình

- Adjusted R-squared = 0.8457 => giải thích được 83.87% sự phụ thuộc của biến mpg vào các biến displacement, weight, model year77, model year78, model year79, model year80, model year81, model year82, origin2, origin3

3. Sự phụ thuộc của mpg vào các biến theo tỉ lệ như sau:
mpg = 37.4138347 + 0.0133566*(displacement) -0.0061908*(weight) + 3.2765532*(model year77) + 3.1877580*(model year78) + 5.1919267*(model year79) + 9.2282220*(model year80) + 6.5869794*(model year81) + 8.9562255*(model year82) + 2.6666858*(origin2) + 2.5282001*(origin3)

TÓM LẠI:

- Mức độ hao xăng phụ thuộc vào dung tích xi lanh, trọng lượng xe, năm sản xuất và nguồn gốc xe

- Khi trọng lượng xe tăng 1 đơn vị thì mức độ hao xăng giảm 0.0056894 đơn vị

- Với những xe sản xuất từ năm 1971 đến 1976 không ảnh hưởng đến mức độ hao xăng

- Những xe sản xuất từ năm 1977 đến 1982 có ảnh hưởng đến mức độ hao xăng theo tỉ lệ tương ứng: 3.2765532, 3.1877580, 5.1919267, 9.2282220, 6.5869794, 8.9562255

- Khi nguồn gốc xe là 2 hoặc 3 cũng ảnh hưởng đến mức độ hao xăng, tỉ lệ ảnh hưởng tương ứng là 2.0228451 và 1.7243036. Những xe có nguồn gốc là 1 thì không ảnh hưởng

KIỂM TRA PHẦN DƯ

```
residus = residuals(modelComparison)
shapiro.test(residus)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residus
## W = 0.98204, p-value = 0.0005727
```

Nhận xét:

- p-value = 0.0005727 < 0.05 => phần dư có phân phối chuẩn nên ta không cần phải xử lý phần dư để đưa về dạng chuẩn

DỰ ĐOÁN

```
predictions = modelComparison %>% predict(testData)
rmse = RMSE(predictions, testData$mpg)
r2 = R2(predictions, testData$mpg)
```

- Giá trị của RMSE = 3.1243252 cho biết độ lệch trung bình giữa các giá trị dự đoán và các giá trị thực tế là 3.1243252

- Giá trị của R2 = 0.8484403 cho biết 84.84% biến thiên của biến phụ thuộc có thể được giải thích bởi các biến độc lập được sử dụng trong mô hình. Từ đây cho thấy mô hình phù hợp chặt chẽ với dữ liệu.

```r
insurance = read.csv(here("data", "insurance.csv"),  header=TRUE, sep=",")
names(insurance)
```

```
## [1] "age"      "sex"      "bmi"      "children" "smoker"   "region"   "charges"
```

```r
is.factor(insurance$smoker)
```

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

```r
insurance$smoker = as.factor(insurance$smoker)
```

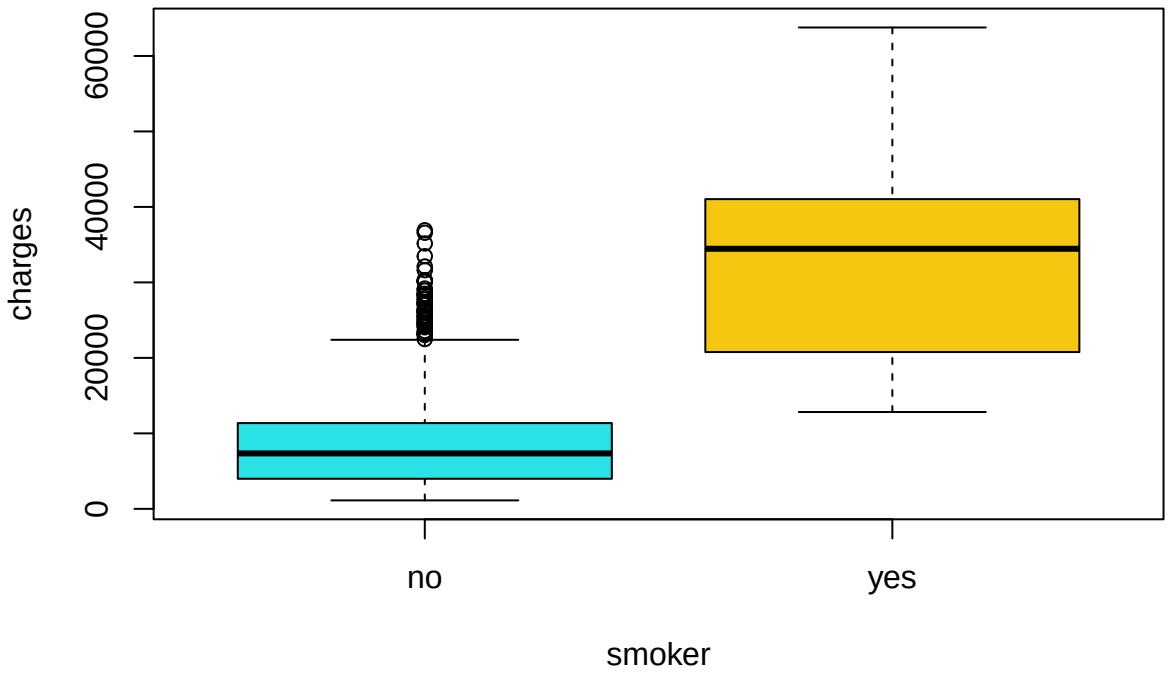
BẢO HIỂM

Mục tiêu chi phí bảo hiểm y tế có bị ảnh hưởng bởi người sử dụng bảo hiểm có hút thuốc hay không

```r
insurance1 <- lm(charges ~ smoker, data = insurance, col = c(5,7))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'col' will be disregarded
```

```r
plot(charges ~ smoker, data = insurance, col = c(5,7))
```



- Từ biểu đồ boxplot, ta thấy rõ ràng việc sử dụng thuốc lá có ảnh hưởng rõ ràng đến số tiền họ chi tiêu cho bảo hiểm

```r
# aov() function:
insurance_aov = aov(charges ~ smoker, data = insurance)
insurance_aov
```

```
## Call:
##    aov(formula = charges ~ smoker, data = insurance)
##
## Terms:
```

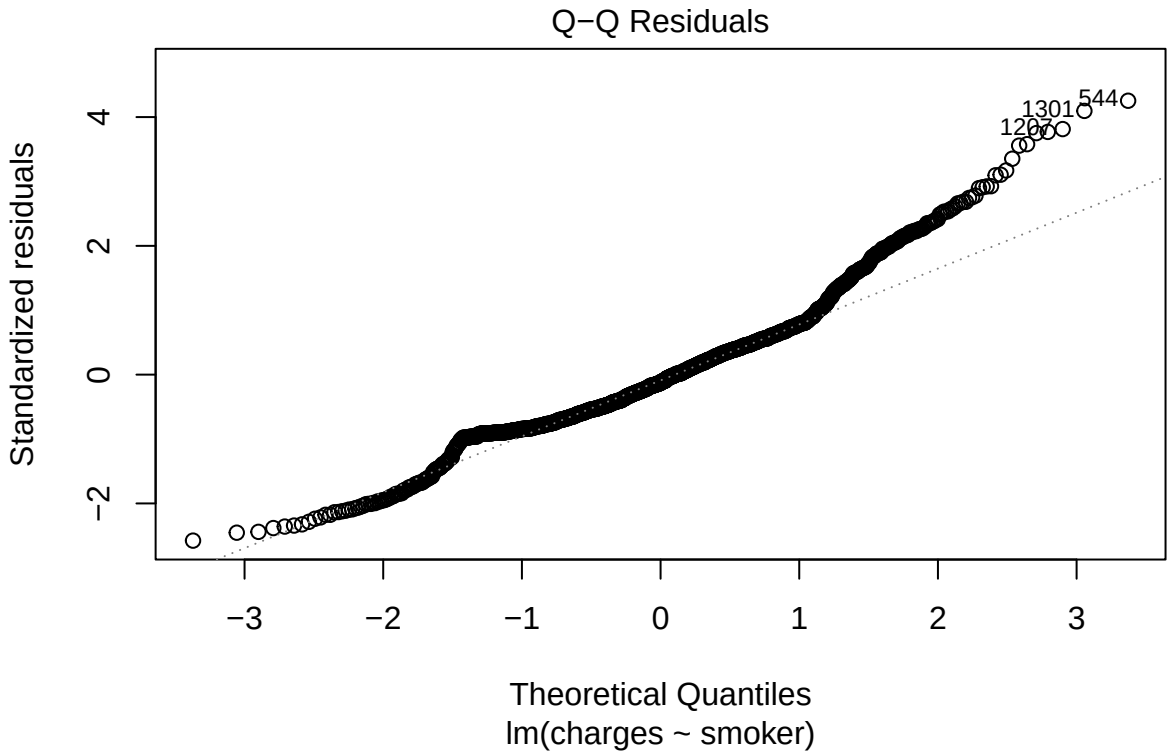
```
##                         smoker    Residuals
## Sum of Squares  121519903622  74554317947
## Deg. of Freedom             1         1336
##
## Residual standard error: 7470.216
## Estimated effects may be unbalanced
```

```
summary(insurance_aov)
```

```
##               Df    Sum Sq   Mean Sq F value Pr(>F)
## smoker         1 1.215e+11 1.215e+11    2178 <2e-16 ***
## Residuals   1336 7.455e+10 5.580e+07
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- vì giá trị p-value < 0.05 => không đủ cơ sở bác bỏ giả thuyết là việc hút thuốc hay không sẽ không ảnh hưởng tới số tiền mua bảo hiểm.

```
plot(insurance1,2)
```




```
shapiro.test(residuals(insurance1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(insurance1)
## W = 0.96084, p-value < 2.2e-16
```

nhận xét - vì p < 0.05 -> ta không bác bỏ Giả thuyết không (H0) của kiểm định Shapiro-Wilk cho rằng dữ liệu có phân phối chuẩn.

```r
smoker = data.frame(smoker = unique(insurance$smoker))
data.frame(smoker, insurance = predict(insurance_aov, smoker))
```

```
##   smoker insurance
## 1    yes 32050.232
## 2     no  8434.268
```

nhận xét - Người hút thuốc (smoker = yes) có chi phí bảo hiểm dự đoán cao hơn rất nhiều so với người không hút thuốc (smoker = no). Cụ thể, chi phí bảo hiểm dự đoán cho người hút thuốc là 32,050.232, trong khi đối với người không hút thuốc là 8,434.268.

```r
#Post Hoc Testing
#test all possible comparisons of two means.
with(insurance, pairwise.t.test(charges, smoker, p.adj = "none"))
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  charges and smoker
##
##     no
## yes <2e-16
##
## P value adjustment method: none
```

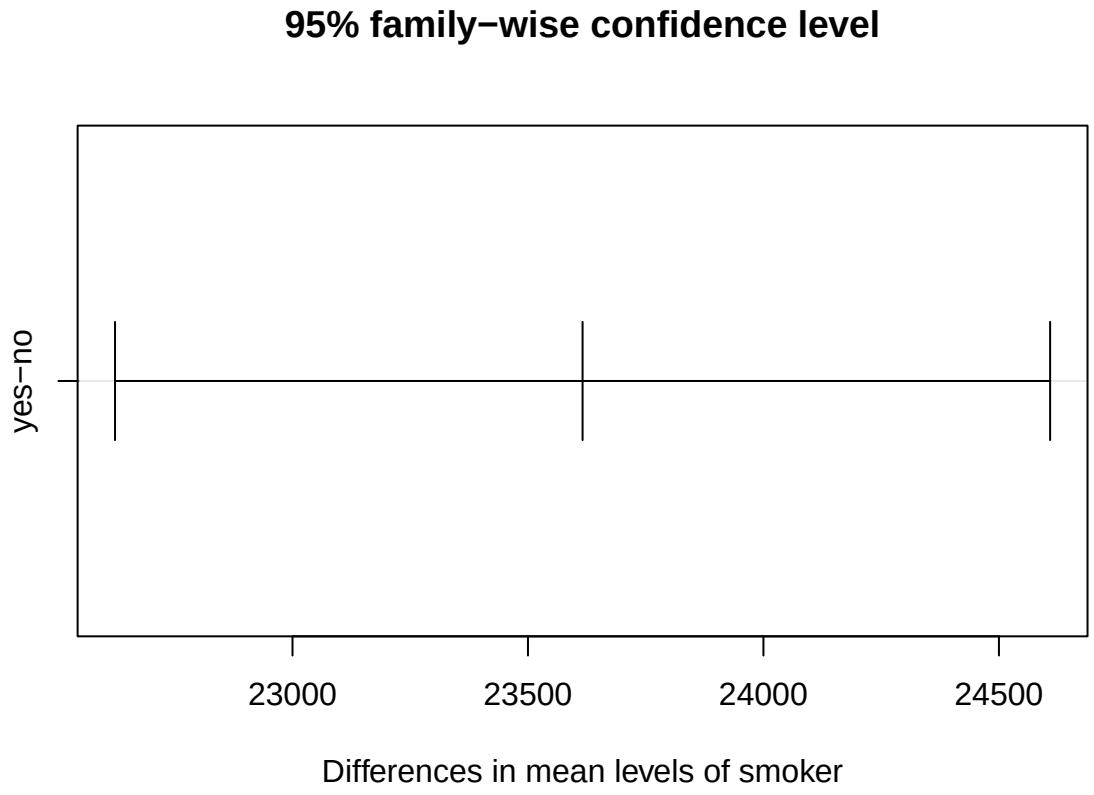
```r
#the Bonferroni correction
with(insurance, pairwise.t.test(charges, smoker, p.adj = "bonferroni"))
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  charges and smoker
##
##     no
## yes <2e-16
##
## P value adjustment method: bonferroni
```

```r
##Tukey's Honest Significance difference
TukeyHSD(insurance_aov, conf.level = 0.95)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = charges ~ smoker, data = insurance)
##
## $smoker
##            diff      lwr      upr p adj
## yes-no 23615.96 22623.17 24608.75     0
```

```r
# produce a plot of these confidence intervals
plot(TukeyHSD(insurance_aov, conf.level = 0.95))
```



nhận xét - Giá trị p < 2e-16 cho thấy rằng sự khác biệt về chi phí bảo hiểm giữa người hút thuốc (yes) và không hút thuốc (no) là rất có ý nghĩa thống kê. Cụ thể, xác suất để sự khác biệt này xảy ra do ngẫu nhiên là cực kỳ thấp.

Bộ dữ liệu: CSM

MÔ TẢ DỮ LIỆU

ĐỌC DỮ LIỆU

```
csmOrg = read_excel(here("data", "csm.xlsx"))
(dim(csmOrg))
```

```
## [1] 231  14
```

TIỀN XỬ LÝ DỮ LIỆU

Kiểm tra dữ liệu

Nhận xét

Loại bỏ dữ liệu khuyết

Loại bỏ outlier

Loại bỏ dữ liệu trùng

Chuẩn hóa dữ liệu

Chia dữ liệu