Handling-outliers.R

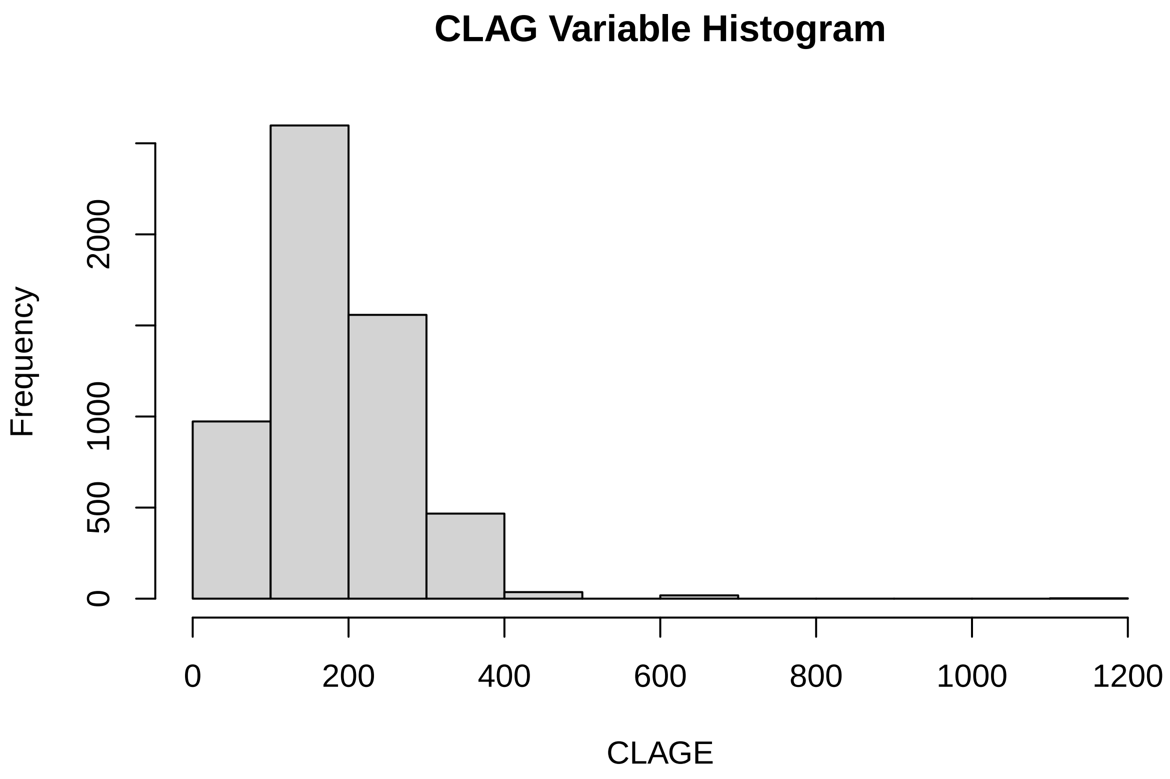
patriciamaya

2020-12-06

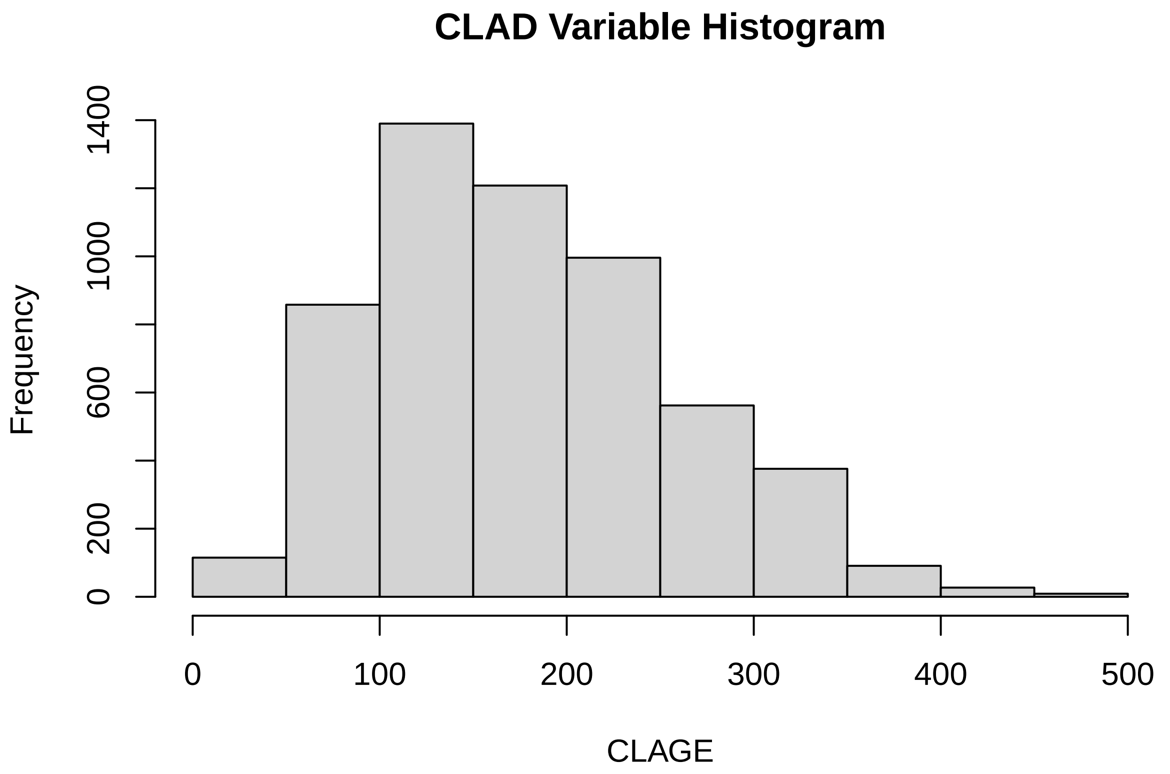
```
#HANDLING OUTLIERS

data <- read.csv("~/Downloads/hmeq.csv")
summary(data)
```

```
##       BAD              LOAN          MORTDUE           VALUE
##  Min.   :0.0000   Min.   : 1100   Min.   :  2063   Min.   :  8000
##  1st Qu.:0.0000   1st Qu.:11100   1st Qu.: 46276   1st Qu.: 66076
##  Median :0.0000   Median :16300   Median : 65019   Median : 89236
##  Mean   :0.1995   Mean   :18608   Mean   : 73761   Mean   :101776
##  3rd Qu.:0.0000   3rd Qu.:23300   3rd Qu.: 91488   3rd Qu.:119824
##  Max.   :1.0000   Max.   :89900   Max.   :399550   Max.   :855909
##                                   NA's   :518      NA's   :112
##     REASON              JOB                 YOJ             DEROG
##  Length:5960        Length:5960        Min.   : 0.000   Min.   : 0.0000
##  Class :character   Class :character   1st Qu.: 3.000   1st Qu.: 0.0000
##  Mode  :character   Mode  :character   Median : 7.000   Median : 0.0000
##                                        Mean   : 8.922   Mean   : 0.2546
##                                        3rd Qu.:13.000   3rd Qu.: 0.0000
##                                        Max.   :41.000   Max.   :10.0000
##                                        NA's   :515      NA's   :708
##      DELINQ            CLAGE             NINQ             CLNO
##  Min.   : 0.0000   Min.   :   0.0   Min.   : 0.000   Min.   : 0.0
##  1st Qu.: 0.0000   1st Qu.: 115.1   1st Qu.: 0.000   1st Qu.:15.0
##  Median : 0.0000   Median : 173.5   Median : 1.000   Median :20.0
##  Mean   : 0.4494   Mean   : 179.8   Mean   : 1.186   Mean   :21.3
##  3rd Qu.: 0.0000   3rd Qu.: 231.6   3rd Qu.: 2.000   3rd Qu.:26.0
##  Max.   :15.0000   Max.   :1168.2   Max.   :17.000   Max.   :71.0
##  NA's   :580       NA's   :308      NA's   :510      NA's   :222
##     DEBTINC
##  Min.   :  0.5245
##  1st Qu.: 29.1400
##  Median : 34.8183
##  Mean   : 33.7799
##  3rd Qu.: 39.0031
##  Max.   :203.3121
##  NA's   :1267
```

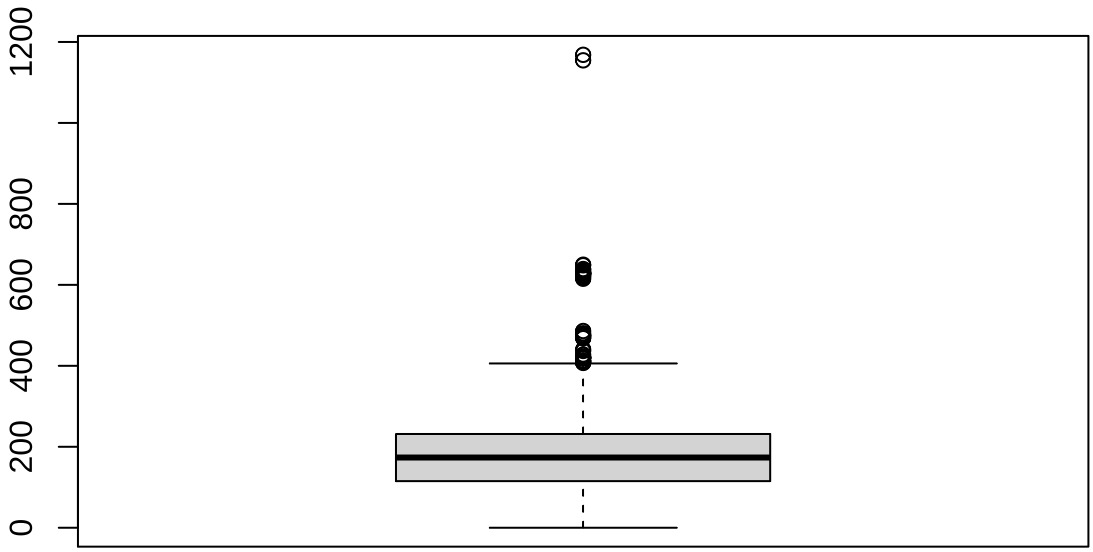
```
# METHOD 1:
#To detect the outliers, you can first draw the histogram to determine the range of outliers.

hist(data$CLAGE, main = "CLAG Variable Histogram", xlab = "CLAGE")
```



```
      #To remove the outliers we can use the "subset(DataSet name, Variable name < Bound)
DataNew  = subset(data, CLAGE < 500)
hist(DataNew$CLAGE, main = "CLAD Variable Histogram", xlab = "CLAGE")
```



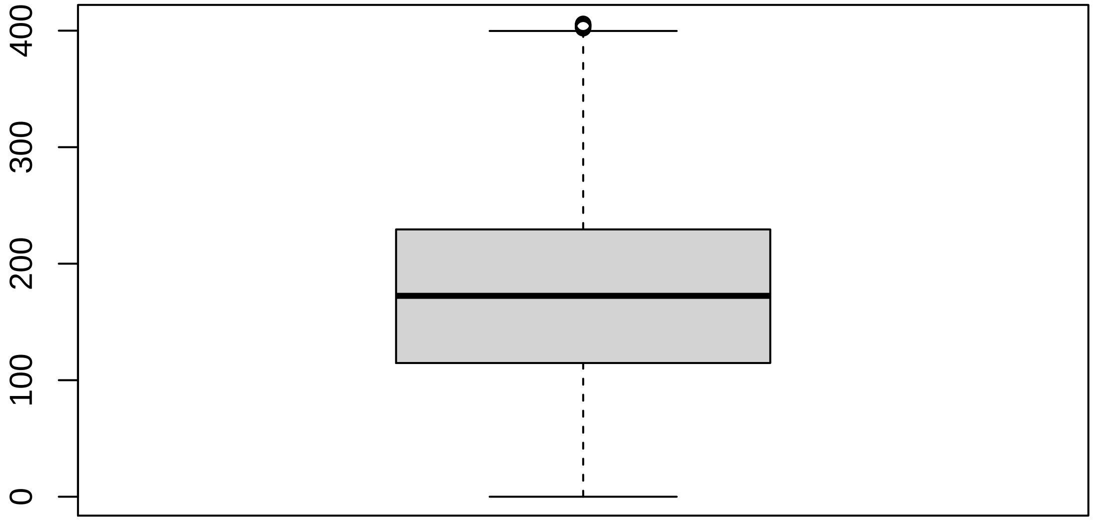
```
# If you have more than one variable with outliers you can use the following formula:
# NewData = subset(Data name, Var1 name < Bound1 & Var2 name < Bound2 & · · ·)
# METHOD 2:
#To detect the outliers, the command "boxplot.stats()$out" can be used which
#uses the Tukey’s method to identify the outliers ranged above and below the 1.5 × IQR.
boxplot(data$CLAG)
```



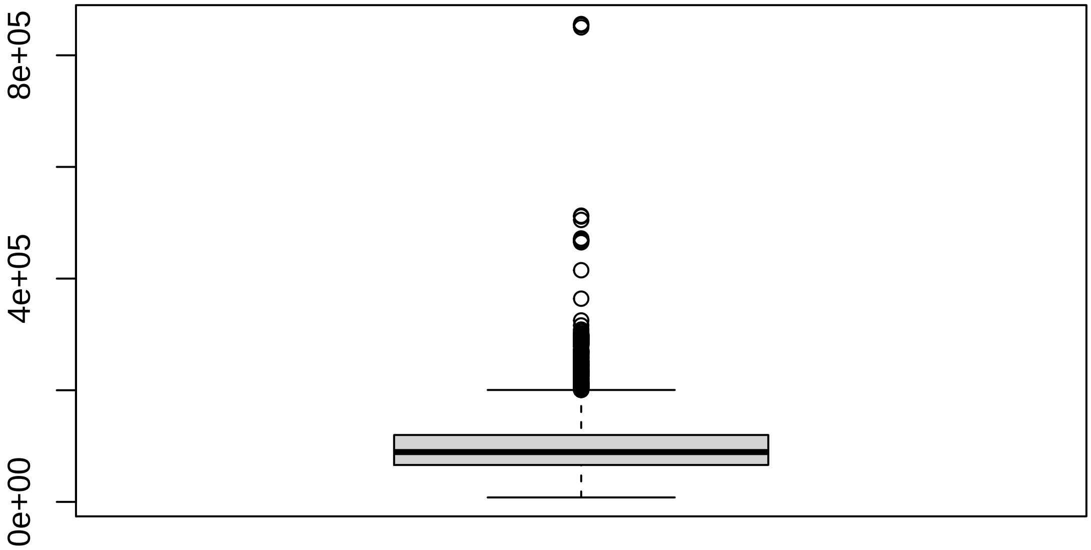
```
CLAG_OutLiers = boxplot.stats(data$CLAG)$out # We first save all the outliers in the vector
CLAG_OutLiers
```

```
##  [1]  417.6333  419.3752  475.8000  423.2096  421.5419  411.9521  419.2730
##  [8]  420.0982  411.7531  419.0333  421.3629  440.4213  427.9236  436.7518
## [15]  407.2612 1154.6333 1168.2336  630.0333  632.1032  618.7359  634.4619
## [22]  407.5856  412.0149  626.2971  623.4562  627.7024  626.7714  615.1334
## [29]  638.2754  628.1581  639.0582  622.3558  628.9819  627.6621  629.0958
## [36]  468.8667  649.7471  408.1876  648.3285  412.0205  471.8875  473.8140
## [43]  474.0271  485.9454  480.3560  476.7283  468.1781
```

```
Data<- data[-which(data$CLAG %in% CLAG_OutLiers),] #REMOVING OUTLIERS FROM DATA
boxplot(Data$CLAG)
```



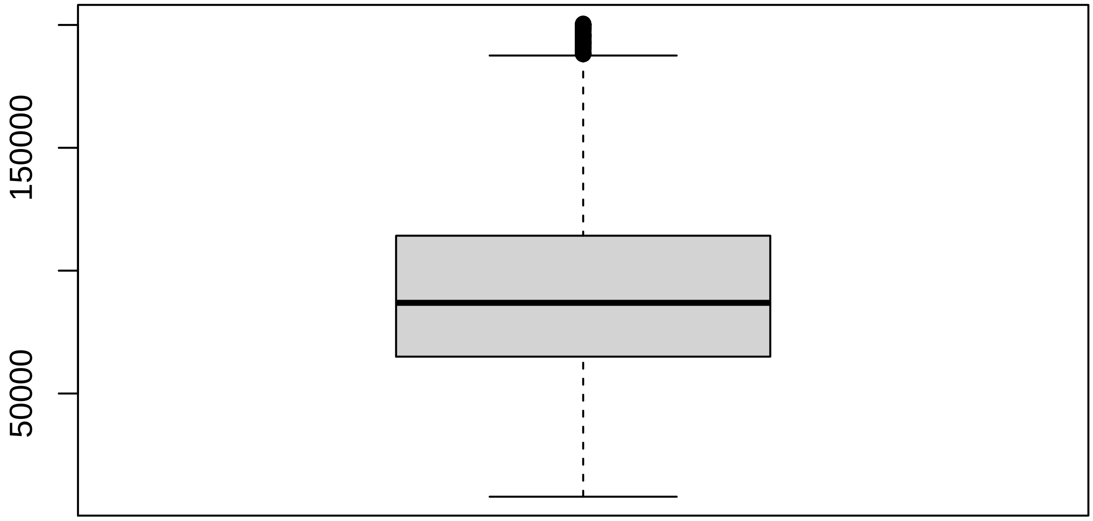
```
boxplot(data$VALUE)
```



```
VALUE_OutLiers = boxplot.stats(data$VALUE)$out # We first save all the outliers in the vector
VALUE_OutLiers
```

```
##   [1] 245300 251962 250155 245730 208910 247611 205981 203936 251771 246758
##  [11] 249071 251935 202962 251426 201689 202788 201281 210000 203815 206201
##  [21] 209931 205346 219783 219936 201713 225750 215784 268000 201820 203341
##  [31] 203720 226000 228670 235000 214523 202186 200480 215014 227295 208924
##  [41] 227171 232176 229929 234454 230920 201245 210072 227737 210595 200707
##  [51] 212995 235912 207562 202800 205950 206148 204282 206368 209364 208429
##  [61] 202989 215548 208782 210685 212530 203737 205608 217000 200902 219300
##  [71] 224270 222227 224233 277500 204000 308600 201500 201214 286955 220843
##  [81] 219297 204082 220886 230000 234004 209695 204192 204384 282972 235968
##  [91] 230443 212505 284790 200594 234269 230513 231933 281186 260000 264462
## [101] 232998 216500 280000 266793 237546 285749 233603 233800 267036 204963
## [111] 266430 225184 238729 266670 262210 261393 289931 281351 289991 211936
## [121] 202500 226000 236250 241279 207200 236200 232760 220000 211000 231000
## [131] 243809 201918 207511 214014 245988 210724 209649 243327 267506 264772
## [141] 247025 263958 211230 212089 210298 250814 242602 208421 239546 211151
## [151] 260479 207647 286283 203202 208432 245422 241754 249773 212536 207302
## [161] 260638 283978 250164 267675 208775 233480 211400 282068 286555 237302
## [171] 212953 246354 206788 211014 202542 206521 229116 207997 209726 207737
## [181] 207035 227617 239990 243593 242544 206030 232345 238745 214558 245685
## [191] 285000 211558 201000 291314 289260 291490 290762 294326 293000 282000
## [201] 240000 285921 284199 293118 291013 301984 298239 290923 298090 282839
## [211] 283022 299299 215000 293790 294372 221100 299720 297294 286938 296728
## [221] 293949 289091 298682 284049 290039 286305 224716 316000 265000 505000
## [231] 271738 268436 270992 324987 268745 512650 267238 268857 205493 272874
## [241] 202894 364000 270794 210065 209950 270751 269450 511164 207976 235000
## [251] 208296 465000 225000 225000 850000 415000 208657 467112 203712 205613
## [261] 466731 208676 467818 290000 202877 207314 471827 469694 854112 854114
## [271] 227168 469748 469771 297444 291222 466755 299171 855909 268000 235500
## [281] 230000 295000 300900 288000 250000 245000 235000 271676 244322 255435
## [291] 252724 257077 255026 251643 256589 258678 305514 257688 299772 297280
## [301] 256977 207797 207803 201928 281000 290239 288000 288525 291242 295551
## [311] 293252 293901 294367 294169 288512 292380 289430 215000 224630
```

```
Data<- data[-which(data$VALUE %in% VALUE_OutLiers),] #REMOVING OUTLIERS FROM DATA
boxplot(Data$VALUE)
```


150000
50000