ASSIGNMENT 1: CANCER MORTALITY

2024-11-28

```
## null device
##           1
```

##Data preparation

First, the data was imported and sampled as asked to.

```
library(readr)
train_data <- read.csv("train.csv", stringsAsFactors = FALSE)
test_data <- read.csv("test.csv", stringsAsFactors = FALSE)
```

```
##   avganncount avgdeathsperyear target_deathrate incidencerate medincome
## 1          57               26            144.4         350.1     49955
## 2         428              152            176.0         505.4     52313
## 3         146               71            183.6         404.0     40189
## 4        4025             1380            177.8         510.9     60397
## 5         113               36            121.4         413.3     54721
## 6         740              269            172.7         499.3     51395
##   popest2015 povertypercent studypercap          binnedinc medianage
## 1      10321           12.5      0.0000 (48021.6, 51046.4]      48.3
## 2      61023           15.6    180.2599 (51046.4, 54545.6]      45.4
## 3      20848           17.8      0.0000 (37413.8, 40362.7]      51.7
## 4     843954           13.1    427.7484 (54545.6, 61494.5]      35.8
## 5      16252           12.7      0.0000 (54545.6, 61494.5]      54.4
## 6     121846           15.7    837.1223 (51046.4, 54545.6]      41.0
##   medianagemale medianagefemale                  geography percentmarried
## 1          47.8            48.9  Lincoln County, Washington           57.8
## 2          43.5            48.0    Mason County, Washington           50.4
## 3          50.8            52.5  Pacific County, Washington           52.7
## 4          34.7            37.0   Pierce County, Washington           50.0
## 5          54.0            54.6 San Juan County, Washington           56.8
## 6          40.0            42.2   Skagit County, Washington           53.6
##   pctnohs18_24 pcths18_24 pctsomecol18_24 pctbachdeg18_24 pcths25_over
## 1         14.9       43.0            40.0             2.0         33.4
## 2         29.9       35.1              NA             4.5         30.4
## 3         27.3       33.9            36.5             2.2         31.6
## 4         15.6       36.3              NA             7.1         28.8
## 5         17.7       32.4              NA             5.2         17.2
## 6         25.5       33.8            37.6             3.1         26.7
##   pctbachdeg25_over pctemployed16_over pctunemployed16_over pctprivatecoverage
## 1              15.0               48.2                  4.8               61.6
## 2              11.9               44.1                 12.9               60.0
## 3              11.3               40.9                  8.9               55.8
## 4              16.2               56.6                  9.2               69.9
## 5              26.2               54.6                  5.9               67.2
## 6              15.9                 NA                  8.2               64.4
##   pctprivatecoveragealone pctempprivcoverage pctpubliccoverage
## 1                    43.9               35.1              44.0
## 2                    38.8               32.6              43.2
## 3                    33.1               25.9              50.9
## 4                      NA               44.4              31.4
## 5                      NA               27.9              41.6
## 6                      NA               38.0              38.1
##   pctpubliccoveragealone pctwhite  pctblack  pctasian pctotherrace
```

```
## 1                   22.7 94.10402 0.2701920 0.6658304    0.4921355
## 2                   20.2 84.88263 1.6532052 1.5380566    3.3146354
## 3                   24.1 89.40664 0.3051586 1.8890773    2.2862679
## 4                   16.5 74.72967 6.7108542 6.0414720    2.6991844
## 5                   18.3 92.57333 0.6517924 1.4289296    2.2374029
## 6                   20.2 85.59027 0.8060800 1.8878359    6.2265906
##   pctmarriedhouseholds birthrate
## 1             54.02746  6.796657
## 2             51.22036  4.964476
## 3             48.96703  5.889179
## 4             50.06357  5.533430
## 5             50.03892  4.586130
## 6             52.93733  5.818153
```

We need some information about the data to start making decisions about how deal with the problem.

```
train_data$binnedinc <- factor(train_data$binnedinc)
train_data$geography <- factor(train_data$geography)
```

In the summary, it can be notice that in the variable "pctsomecol18_24" shows a great percentage of missing values. According with the information of the project, the variable corresponds with "Percent of county residents ages 18-24 highest education attained: some college" so it should be decided what to do with the variable.

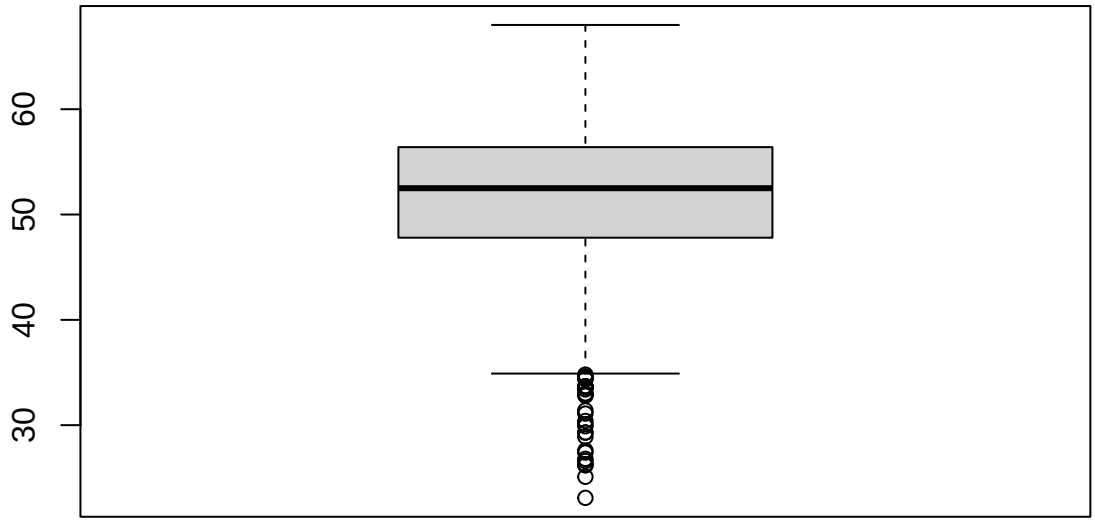
```
str(train_data)
```

```
## 'data.frame':    1831 obs. of  33 variables:
##  $ avganncount            : num  57 428 146 4025 113 ...
##  $ avgdeathsperyear       : int  26 152 71 1380 36 269 26 1118 106 120 ...
##  $ target_deathrate       : num  144 176 184 178 121 ...
##  $ incidencerate          : num  350 505 404 511 413 ...
##  $ medincome              : int  49955 52313 40189 60397 54721 51395 52673 71890 43823 49819 ...
##  $ popest2015             : int  10321 61023 20848 843954 16252 121846 11339 772501 43791 60338 ...
##  $ povertypercent         : num  12.5 15.6 17.8 13.1 12.7 15.7 12.6 9.9 19.3 15.7 ...
##  $ studypercap            : num  0 180 0 428 0 ...
##  $ binnedinc              : Factor w/ 10 levels "(34218.1, 37413.8]",..: 6 7 2 8 8 7 7 9 4 6 ...
##  $ medianage              : num  48.3 45.4 51.7 35.8 54.4 41 45.2 37.6 46.2 36.9 ...
##  $ medianagemale          : num  47.8 43.5 50.8 34.7 54 40 44.9 36.6 45.6 35.5 ...
##  $ medianagefemale        : num  48.9 48 52.5 37 54.6 42.2 45.5 38.7 46.8 39.1 ...
##  $ geography              : Factor w/ 1831 levels "Acadia Parish, Louisiana",..: 977 1065 1277 1324 1
##  $ percentmarried         : num  57.8 50.4 52.7 50 56.8 53.6 54.4 52.1 57.2 46.5 ...
##  $ pctnohs18_24           : num  14.9 29.9 27.3 15.6 17.7 25.5 20 15.4 25.1 13.7 ...
##  $ pcths18_24             : num  43 35.1 33.9 36.3 32.4 33.8 43.8 33.3 35.3 29.2 ...
##  $ pctsomecol18_24        : num  40 NA 36.5 NA NA 37.6 NA NA NA NA ...
##  $ pctbachdeg18_24        : num  2 4.5 2.2 7.1 5.2 3.1 2.4 8.3 3.9 5.6 ...
##  $ pcths25_over           : num  33.4 30.4 31.6 28.8 17.2 26.7 29.2 24.3 34.4 22.6 ...
##  $ pctbachdeg25_over      : num  15 11.9 11.3 16.2 26.2 15.9 14.2 20.9 10.7 16.4 ...
##  $ pctemployed16_over     : num  48.2 44.1 40.9 56.6 54.6 NA 51.5 62.1 46.3 54.5 ...
##  $ pctunemployed16_over   : num  4.8 12.9 8.9 9.2 5.9 8.2 8.3 7.5 9.4 6.4 ...
##  $ pctprivatecoverage     : num  61.6 60 55.8 69.9 67.2 64.4 64.4 73.3 58.3 65.7 ...
##  $ pctprivatecoveragealone: num  43.9 38.8 33.1 NA NA NA 49.7 61.6 39.3 47.6 ...
##  $ pctempprivcoverage     : num  35.1 32.6 25.9 44.4 27.9 38 42.6 54.3 33.9 40 ...
##  $ pctpubliccoverage      : num  44 43.2 50.9 31.4 41.6 38.1 36.1 25.9 45.8 38 ...
```

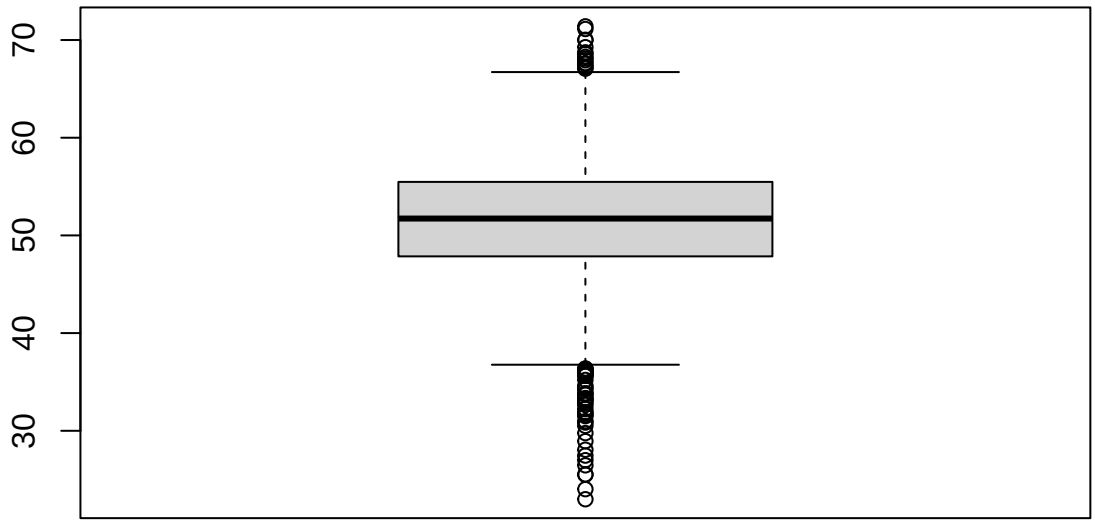
```
##  $ pctpubliccoveragealone : num  22.7 20.2 24.1 16.5 18.3 20.2 20.5 14.1 24.1 18.8 ...
##  $ pctwhite               : num  94.1 84.9 89.4 74.7 92.6 ...
##  $ pctblack               : num  0.27 1.653 0.305 6.711 0.652 ...
##  $ pctasian               : num  0.666 1.538 1.889 6.041 1.429 ...
##  $ pctotherrace           : num  0.492 3.315 2.286 2.699 2.237 ...
##  $ pctmarriedhouseholds   : num  54 51.2 49 50.1 50 ...
##  $ birthrate              : num  6.8 4.96 5.89 5.53 4.59 ...
```

This checks the type of each variable

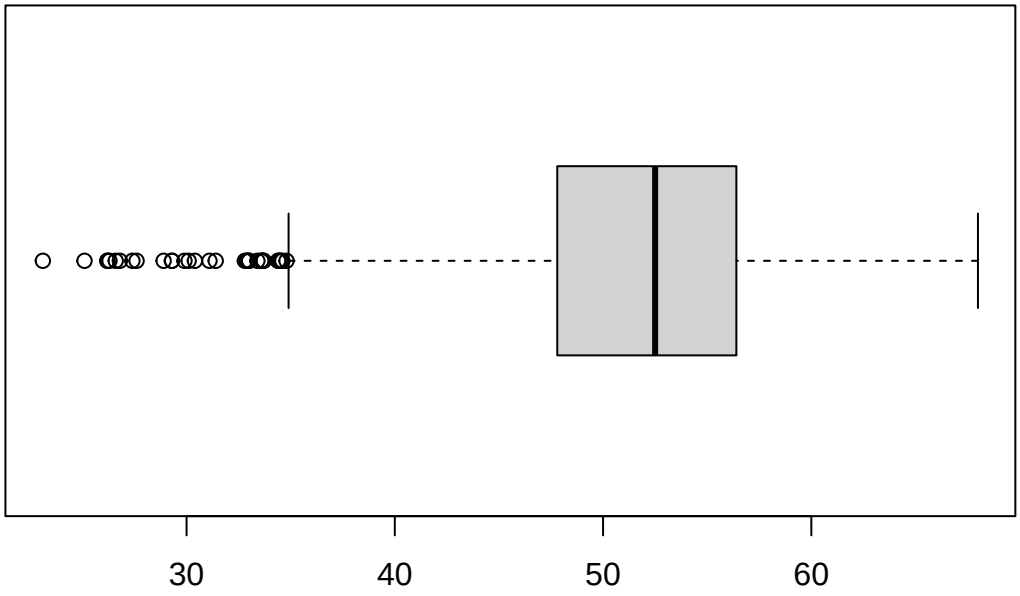
```
# Some plots are done:
boxplot(train_data$percentmarried)
```



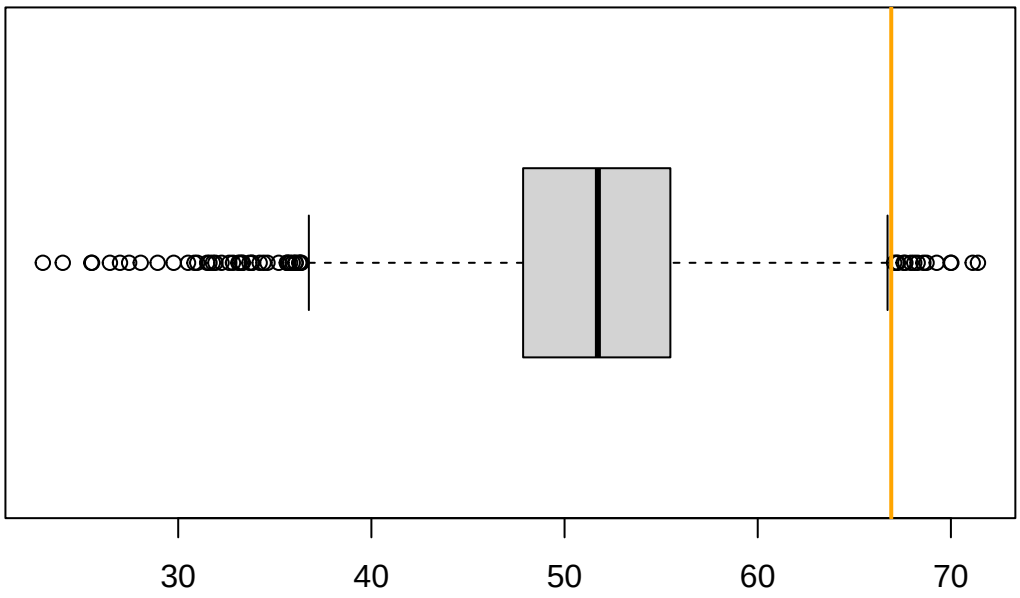

```
boxplot(train_data$pctmarriedhouseholds)
```



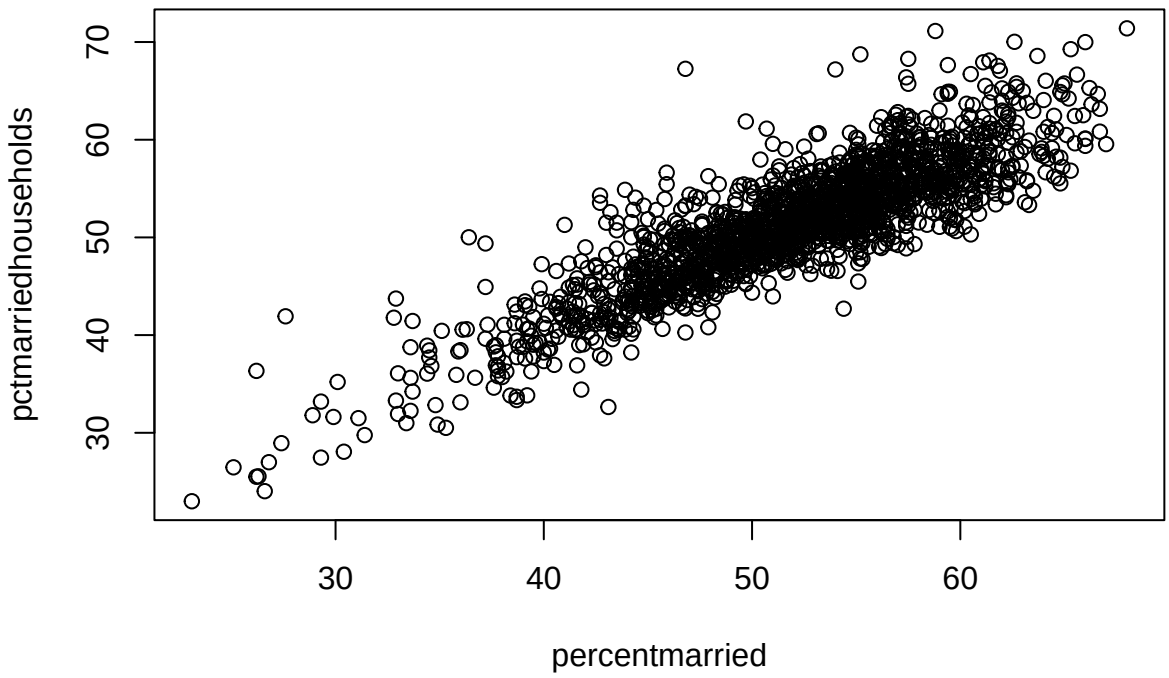

```
varout <- summary(train_data$pctmarriedhouseholds)

# Interquartile range calculation:
iqr <- varout[5] - varout[2]

umout <- varout[5] + 1.5*iqr # Upper extreme for mild outliers
usout <- varout[5] + 3*iqr # Upper extreme for extreme outliersm (iqr = inter quartile range)
boxplot(train_data$percentmarried, horizontal = TRUE)
```




```r
boxplot(train_data$pctmarriedhouseholds, horizontal = TRUE)
abline(v = umout, col = "orange", lwd = 2)
abline(v = usout, col = "red", lwd = 2)
```




```
plot(train_data[,c(14,32)])
```



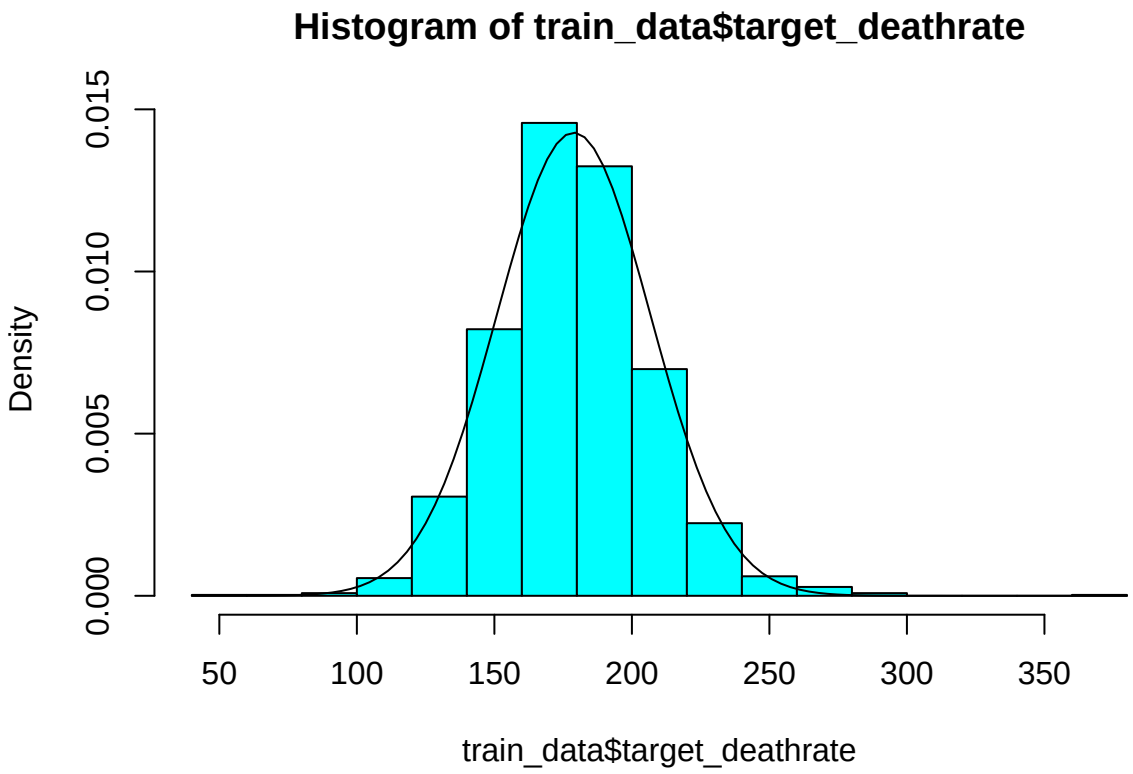

```
cor(train_data[,1:8])
```

```
##                 avganncount avgdeathsperyear target_deathrate incidencerate
## avganncount       1.00000000       0.94921579     -0.138864542    0.05905901
## avgdeathsperyear  0.94921579       1.00000000     -0.091676499    0.05037004
## target_deathrate -0.13886454      -0.09167650      1.000000000    0.44954847
## incidencerate     0.05905901       0.05037004      0.449548469    1.00000000
## medincome         0.25017407       0.20363073     -0.443682800   -0.01934883
## popest2015        0.93637042       0.97888010     -0.119022317    0.01607054
## povertypercent   -0.12236006      -0.05964856      0.450084648    0.03778847
## studypercap       0.08164650       0.06136019      0.003772418    0.08559483
##                   medincome  popest2015 povertypercent  studypercap
## avganncount       0.25017407  0.93637042    -0.12236006  0.081646500
## avgdeathsperyear  0.20363073  0.97888010    -0.05964856  0.061360187
## target_deathrate -0.44368280 -0.11902232     0.45008465  0.003772418
## incidencerate    -0.01934883  0.01607054     0.03778847  0.085594827
## medincome         1.00000000  0.21580024    -0.79181996  0.023526796
## popest2015        0.21580024  1.00000000    -0.06041443  0.055456730
## povertypercent   -0.79181996 -0.06041443     1.00000000 -0.025581370
## studypercap       0.02352680  0.05545673    -0.02558137  1.000000000
```

Determine if the response variable (deathrate) has an acceptably normal distribution.

It is not normally distributed

```
hist(train_data$target_deathrate, breaks = 15, freq = FALSE, col="cyan")
curve(dnorm(x, mean(train_data$target_deathrate), sd(train_data$target_deathrate)), add = TRUE)
```



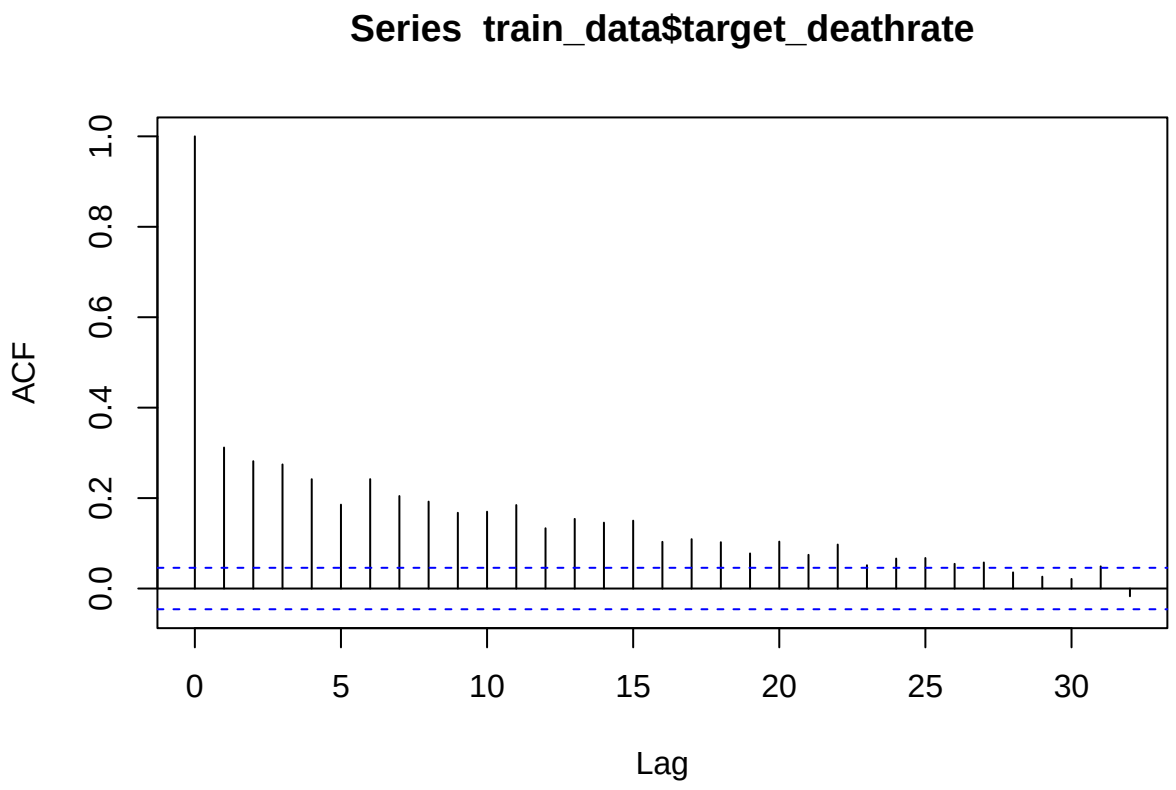

```
shapiro.test(train_data$target_deathrate)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  train_data$target_deathrate
## W = 0.98647, p-value = 4.149e-12
```

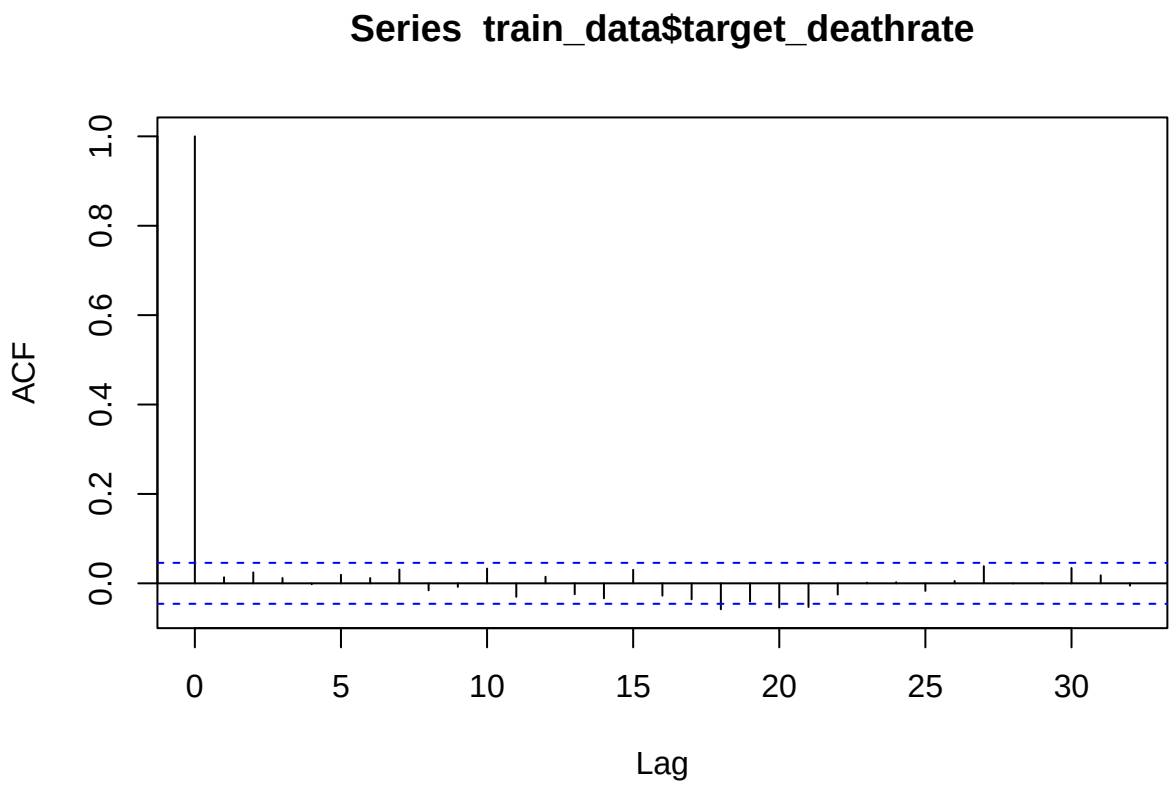
Address tests to discard serial correlation.

using acf we can see there is autocorrelation, so we randomize the dataframe to get rid of it.

```
acf(train_data$target_deathrate)
```

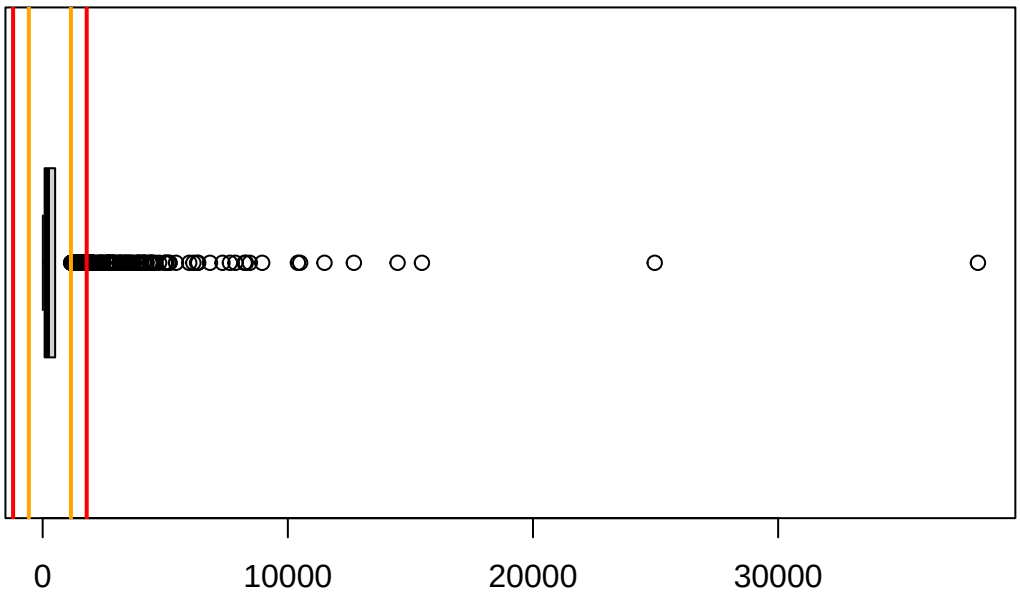



```
ll <- sample(1:nrow(train_data),nrow(train_data))
train_data <- train_data[ll,]
acf(train_data$target_deathrate)
```



Detect univariant and multivariant outliers and retain all of them in exploratory analysis.

```
sum <- summary(train_data$avganncount)
iqr <- sum[5] - sum[2]
lmout <- sum[2] - 1.5*iqr
umout <- sum[5] + 1.5*iqr
lsout <- sum[2] - 3*iqr
usout <- sum[5] + 3*iqr
boxplot(train_data$avganncount, horizontal = TRUE)
abline(v = umout, col = "orange", lwd = 2)
abline(v = usout, col = "red", lwd = 2)
abline(v = lmout, col = "orange", lwd = 2)
abline(v = lsout, col = "red", lwd = 2)
```

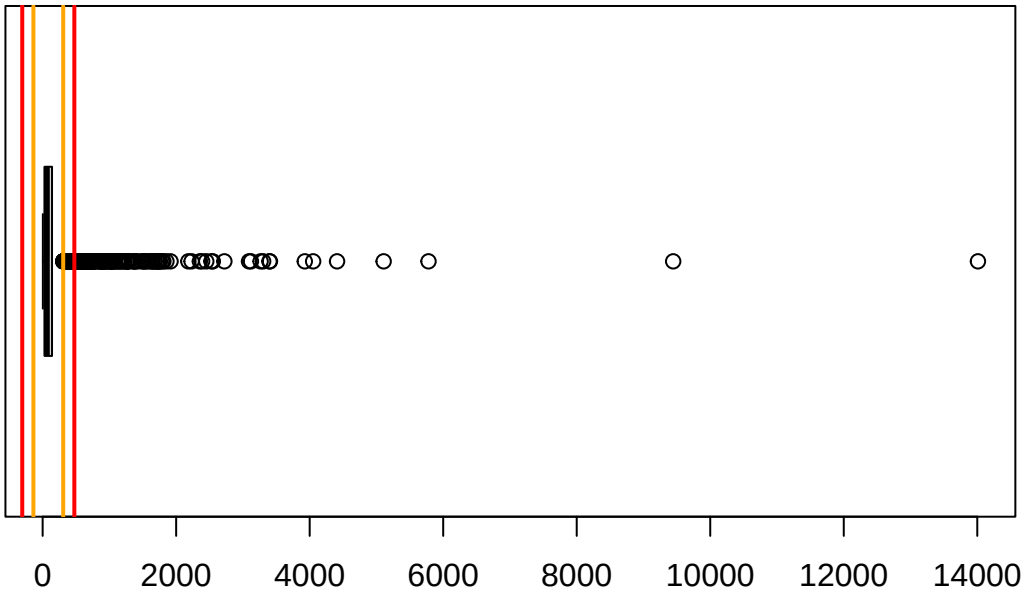



```
sevout <- which(train_data$avganncount > usout); sevout
```

```
##   [1]    3   13   14   26   38   43   48   53   75   80   84   89  116  135  136
##  [16]  143  145  148  153  155  157  169  183  192  193  225  226  227  249  250
##  [31]  252  253  283  287  289  316  318  329  359  371  373  375  380  383  400
##  [46]  403  406  416  425  432  447  478  481  490  491  532  542  548  553  554
##  [61]  563  580  614  616  621  626  634  641  644  649  651  657  659  667  672
##  [76]  685  689  698  703  714  724  727  760  782  787  789  794  801  808  813
##  [91]  816  817  820  835  850  852  869  880  896  901  904  907  924  926  930
## [106]  932  937  940  959  961  971  990  994 1000 1006 1010 1017 1021 1023 1033
## [121] 1035 1036 1038 1039 1041 1042 1044 1055 1058 1060 1081 1085 1088 1094 1095
## [136] 1098 1138 1139 1141 1148 1151 1155 1159 1162 1168 1189 1190 1198 1208 1211
## [151] 1217 1223 1225 1230 1236 1239 1241 1251 1263 1277 1290 1292 1305 1321 1357
## [166] 1361 1366 1370 1374 1378 1384 1394 1414 1422 1429 1430 1451 1463 1468 1473
## [181] 1490 1491 1492 1500 1501 1506 1526 1539 1553 1554 1565 1566 1567 1574 1590
## [196] 1593 1605 1621 1627 1639 1662 1690 1705 1707 1716 1731 1733 1741 1749 1751
## [211] 1756 1759 1768 1771 1775 1784 1801 1810 1811 1814 1816 1820 1821 1823
```

```
sum <- summary(train_data$avgdeathsperyear)
iqr <- sum[5] - sum[2]
lmout <- sum[2] - 1.5*iqr
umout <- sum[5] + 1.5*iqr
lsout <- sum[2] - 3*iqr
usout <- sum[5] + 3*iqr
boxplot(train_data$avgdeathsperyear, horizontal = TRUE)
```

```
abline(v = umout, col = "orange", lwd = 2)
abline(v = usout, col = "red", lwd = 2)
abline(v = lmout, col = "orange", lwd = 2)
abline(v = lsout, col = "red", lwd = 2)
```

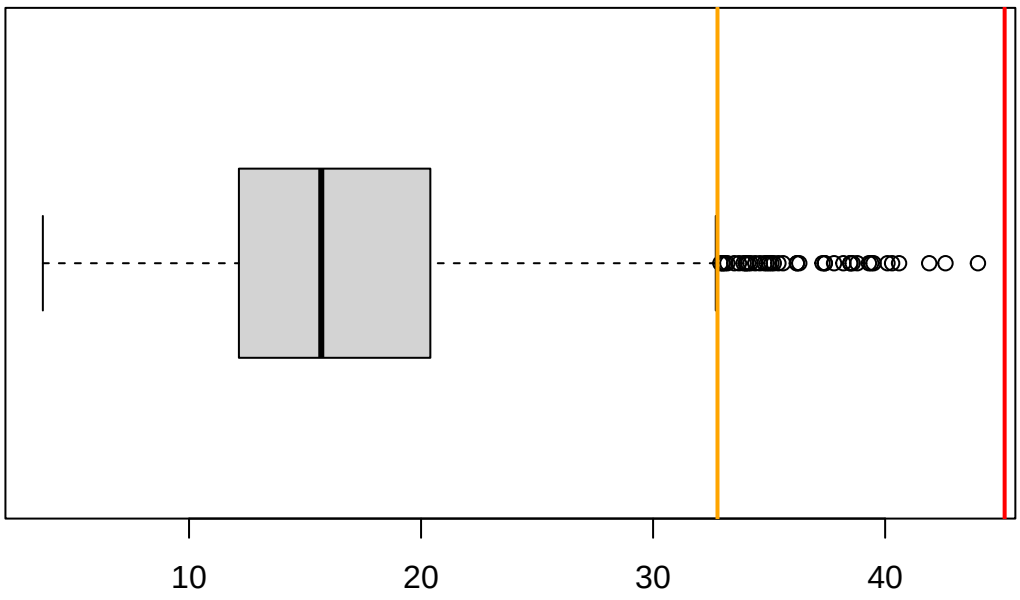



```
sevout <- which(train_data$avgdeathsperyear > usout); sevout
```

```
##   [1]    6   13   14   26   43   71   84   89  153  155  157  169  185  193  217
##  [16]  286  287  289  316  318  329  359  366  373  375  378  380  400  425  432
##  [31]  481  494  507  515  542  552  554  563  580  602  614  616  621  630  634
##  [46]  657  676  685  698  714  724  754  760  782  794  808  813  816  820  831
##  [61]  835  850  851  880  901  926  930  959  971 1000 1010 1017 1021 1035 1036
##  [76] 1041 1055 1060 1068 1081 1102 1138 1139 1148 1159 1168 1187 1198 1214 1217
##  [91] 1224 1225 1239 1263 1292 1305 1329 1357 1366 1370 1374 1378 1384 1401 1402
## [106] 1422 1425 1430 1500 1501 1526 1539 1566 1574 1593 1641 1678 1698 1705 1707
## [121] 1731 1733 1741 1749 1751 1756 1757 1775 1777 1781 1784 1802 1810 1813 1814
## [136] 1818 1821 1823
```

```
sum <- summary(train_data$povertypercent)
iqr <- sum[5] - sum[2]
lmout <- sum[2] - 1.5*iqr
umout <- sum[5] + 1.5*iqr
lsout <- sum[2] - 3*iqr
usout <- sum[5] + 3*iqr
```

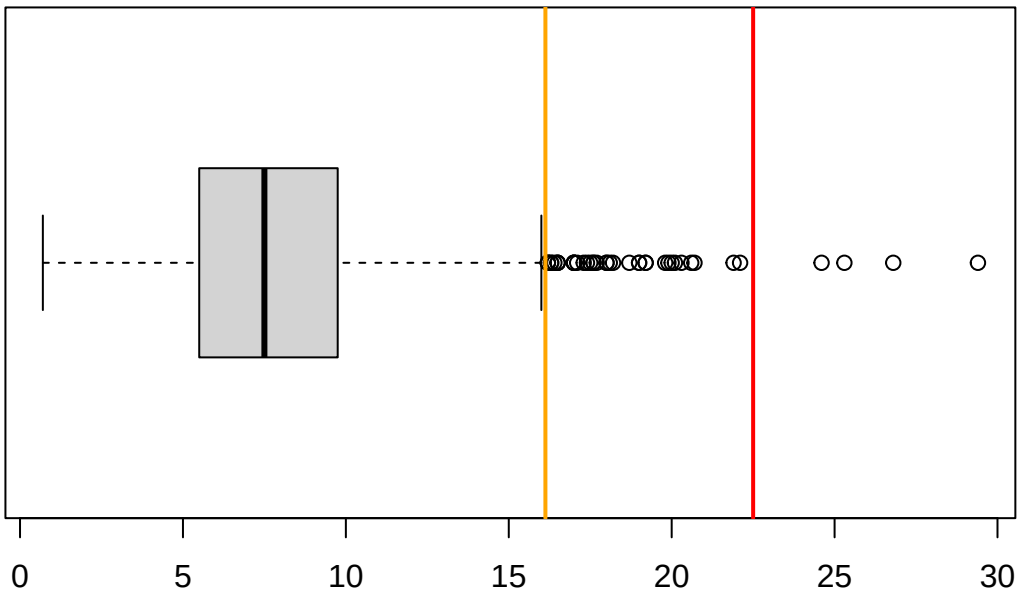
```r
boxplot(train_data$povertypercent, horizontal = TRUE)
abline(v = umout, col = "orange", lwd = 2)
abline(v = usout, col = "red", lwd = 2)
abline(v = lmout, col = "orange", lwd = 2)
abline(v = lsout, col = "red", lwd = 2)
```




```r
sevout <- which(train_data$povertypercent > usout); sevout
```

```
## integer(0)
```

```r
sum <- summary(train_data$pctunemployed16_over)
iqr <- sum[5] - sum[2]
lmout <- sum[2] - 1.5*iqr
umout <- sum[5] + 1.5*iqr
lsout <- sum[2] - 3*iqr
usout <- sum[5] + 3*iqr
boxplot(train_data$pctunemployed16_over, horizontal = TRUE)
abline(v = umout, col = "orange", lwd = 2)
abline(v = usout, col = "red", lwd = 2)
abline(v = lmout, col = "orange", lwd = 2)
abline(v = lsout, col = "red", lwd = 2)
```



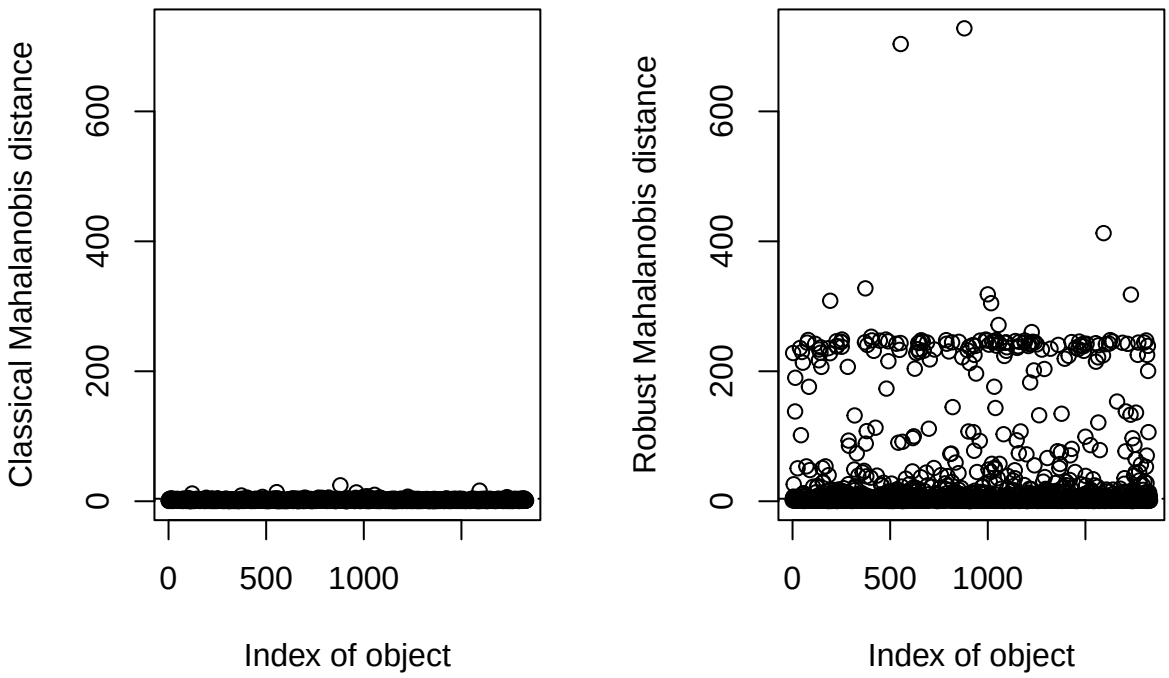

```
sevout <- which(train_data$pctunemployed16_over > usout); sevout
```

```
## [1]    8   30  402 1397
```

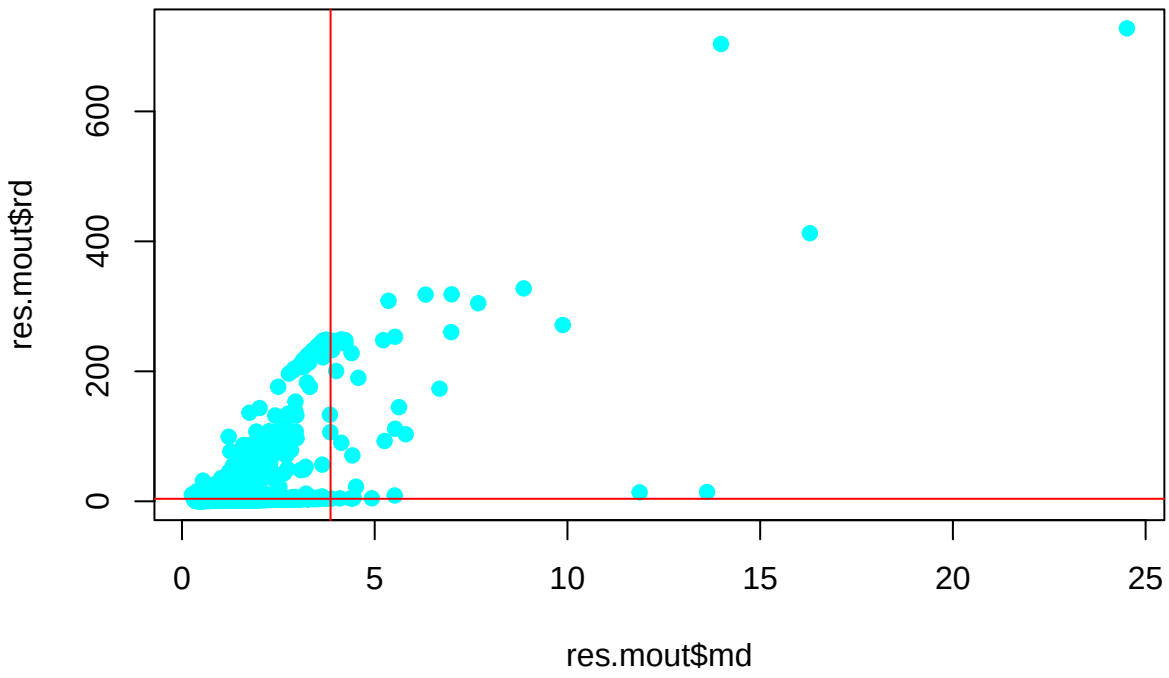
```
library(chemometrics)
```

```
## Loading required package: rpart
```

```
res.mout <- Moutlier(train_data[,1:4], quantile = 0.995)
```




```
plot(res.mout$md, res.mout$rd, col= "cyan", pch = 19)
abline(h=res.mout$cutoff, col = "red")
abline(v=res.mout$cutoff, col = "red")
text(res.mout$md, res.mout$rd, label = row.names(df), cex = 0.5)
```




```
mult_outliers <- which((res.mout$md > res.mout$cutoff) & (res.mout$rd > res.mout$cutoff))
print(mult_outliers)
```

```
##  657 1476 1168  168  237  634   73  793   60 1120 1559   70  784  218  471   66
##   14   75  101  121  193  218  250  253  294  373  393  403  447  481  542  548
##   67  734  615 1516  815  890  654 1633   76  821  864  600   18  971  899 1829
##  554  559  634  644  689  698  706  768  787  817  820  880  916  959  962  990
## 1485  608  805 1826 1648 1046   74  795  883   77 1513  892 1420 1247 1496  205
## 1000 1017 1023 1042 1055 1081 1094 1223 1225 1230 1241 1422 1593 1733 1811 1821
```

Errors and missing values (if any) detection. Apply an imputation technique for both train and test datasets, if needed.

We remove pctsomecol18_24, since it has over 70% of NAs. Using impute PCA for the other two variables (pctemployed16_over and pctprivatecoveragealone) with missing values, we see that the summary of each variable doesn't change meaningfully

```
library(missMDA)
res.misMDA <- imputePCA(train_data[,c(-17, -9, -13)])

summary(train_data$pctprivatecoveragealone)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   16.80   41.50   49.00   48.65   55.50   78.90     356
```

```r
summary(res.misMDA$completeObs[,"pctprivatecoveragealone"])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.80   41.72   48.90   48.67   55.45   78.90
```

```r
summary(train_data$pctemployed16_over)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   23.90   48.60   54.50   54.21   60.30   80.10      82
```

```r
summary(res.misMDA$completeObs[,"pctemployed16_over"])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.90   48.70   54.60   54.25   60.20   80.10
```

```r
train_data[,c(-17, -9, -13)] <- res.misMDA$completeObs
train_data <- train_data[,-17]
```

6. Errors and missing values (if any) detection. Apply an imputation technique for both train and test datasets, if needed.

We applied multiple imputation using the mice package with the Predictive Mean Matching (PMM) method to handle missing data in the training and testing datasets. Variables unrelated to the analysis ("pctnohs18_24", "pcths18_24", "pctsomecol18_24", "pctbachdeg18_24", "pcths25_over", "pctbachdeg25_over") were excluded, and "geography" was omitted as it has a unique value for each county (factor with one level), making it irrelevant for the model's analysis. The imputed datasets will support all subsequent analysis and modeling steps.

```r
library(mice)
```

```
##
## Attaching package: 'mice'

## The following object is masked from 'package:stats':
##
##     filter

## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```r
df_process<-train_data[,-c(13,15:20)]
colSums(is.na(df_process))
```

```
##             avganncount        avgdeathsperyear        target_deathrate
##                       0                       0                       0
##           incidencerate              medincome             popest2015
##                       0                       0                       0
##          povertypercent             studypercap              binnedinc
```

```
##                       0                       0                       0
##              medianage          medianagemale        medianagefemale
##                       0                       0                       0
##         percentmarried   pctunemployed16_over     pctprivatecoverage
##                       0                       0                       0
## pctprivatecoveragealone     pctempprivcoverage      pctpubliccoverage
##                       0                       0                       0
##  pctpubliccoveragealone               pctwhite               pctblack
##                       0                       0                       0
##                pctasian           pctotherrace   pctmarriedhouseholds
##                       0                       0                       0
##              birthrate
##                       0
```

```
imputed_train <- mice(df_process, method = 'pmm', m = 5)
```

```
##
##  iter imp variable
##   1   1
##   1   2
##   1   3
##   1   4
##   1   5
##   2   1
##   2   2
##   2   3
##   2   4
##   2   5
##   3   1
##   3   2
##   3   3
##   3   4
##   3   5
##   4   1
##   4   2
##   4   3
##   4   4
##   4   5
##   5   1
##   5   2
##   5   3
##   5   4
##   5   5
```

```
train_imputed <- complete(imputed_train, 1)

colSums(is.na(train_imputed))
```

```
##             avganncount        avgdeathsperyear        target_deathrate
##                       0                       0                       0
##           incidencerate              medincome             popest2015
##                       0                       0                       0
##          povertypercent            studypercap              binnedinc
```

```
##                       0                       0                       0
##              medianage          medianagemale        medianagefemale
##                       0                       0                       0
##         percentmarried   pctunemployed16_over     pctprivatecoverage
##                       0                       0                       0
## pctprivatecoveragealone     pctempprivcoverage      pctpubliccoverage
##                       0                       0                       0
##  pctpubliccoveragealone               pctwhite               pctblack
##                       0                       0                       0
##                pctasian           pctotherrace   pctmarriedhouseholds
##                       0                       0                       0
##               birthrate
##                       0
```

```
#test data
colSums(is.na(test_data))
```

```
##             avganncount        avgdeathsperyear        target_deathrate
##                       0                       0                       0
##           incidencerate               medincome              popest2015
##                       0                       0                       0
##          povertypercent             studypercap              binnedinc
##                       0                       0                       0
##              medianage          medianagemale        medianagefemale
##                       0                       0                       0
##              geography         percentmarried           pctnohs18_24
##                       0                       0                       0
##             pcths18_24        pctsomecol18_24        pctbachdeg18_24
##                       0                     909                       0
##           pcths25_over      pctbachdeg25_over     pctemployed16_over
##                       0                       0                      70
##    pctunemployed16_over     pctprivatecoverage pctprivatecoveragealone
##                       0                       0                     253
##      pctempprivcoverage      pctpubliccoverage  pctpubliccoveragealone
##                       0                       0                       0
##                pctwhite               pctblack                pctasian
##                       0                       0                       0
##           pctotherrace   pctmarriedhouseholds               birthrate
##                       0                       0                       0
```

```
imputed_test <- mice(test_data[,-c(13,15:20)], method = 'pmm', m = 5)
```

```
##
##  iter imp variable
##   1   1  pctemployed16_over  pctprivatecoveragealone
##   1   2  pctemployed16_over  pctprivatecoveragealone
##   1   3  pctemployed16_over  pctprivatecoveragealone
##   1   4  pctemployed16_over  pctprivatecoveragealone
##   1   5  pctemployed16_over  pctprivatecoveragealone
##   2   1  pctemployed16_over  pctprivatecoveragealone
##   2   2  pctemployed16_over  pctprivatecoveragealone
##   2   3  pctemployed16_over  pctprivatecoveragealone
##   2   4  pctemployed16_over  pctprivatecoveragealone
```

```
##   2   5  pctemployed16_over  pctprivatecoveragealone
##   3   1  pctemployed16_over  pctprivatecoveragealone
##   3   2  pctemployed16_over  pctprivatecoveragealone
##   3   3  pctemployed16_over  pctprivatecoveragealone
##   3   4  pctemployed16_over  pctprivatecoveragealone
##   3   5  pctemployed16_over  pctprivatecoveragealone
##   4   1  pctemployed16_over  pctprivatecoveragealone
##   4   2  pctemployed16_over  pctprivatecoveragealone
##   4   3  pctemployed16_over  pctprivatecoveragealone
##   4   4  pctemployed16_over  pctprivatecoveragealone
##   4   5  pctemployed16_over  pctprivatecoveragealone
##   5   1  pctemployed16_over  pctprivatecoveragealone
##   5   2  pctemployed16_over  pctprivatecoveragealone
##   5   3  pctemployed16_over  pctprivatecoveragealone
##   5   4  pctemployed16_over  pctprivatecoveragealone
##   5   5  pctemployed16_over  pctprivatecoveragealone

## Warning: Number of logged events: 1
```

```
test_imputed <- complete(imputed_test, 1)
colSums(is.na(test_imputed))
```

```
##             avganncount        avgdeathsperyear          target_deathrate
##                       0                       0                         0
##           incidencerate              medincome              popest2015
##                       0                       0                         0
##          povertypercent            studypercap               binnedinc
##                       0                       0                         0
##              medianage          medianagemale        medianagefemale
##                       0                       0                         0
##          percentmarried     pctemployed16_over    pctunemployed16_over
##                       0                       0                         0
##      pctprivatecoverage pctprivatecoveragealone      pctempprivcoverage
##                       0                       0                         0
##       pctpubliccoverage  pctpubliccoveragealone               pctwhite
##                       0                       0                         0
##               pctblack               pctasian           pctotherrace
##                       0                       0                         0
##    pctmarriedhouseholds              birthrate
##                       0                       0
```

7. Preliminary exploratory analysis to describe observed relations has to be undertaken.

When evaluating the possible correlations between predictors and the response variable, adequate correlations are found for the variables pctpubliccoveragealone, incidencerate, povertypercent,pcths25_over, percentmarried ,pctmarriedhouseholds, pctprivatecoveragealone,pctemployed16_over, medincome . The same can be observed in the correlation graph.

```
library(corrplot)
```

```
## corrplot 0.94 loaded
```

```r
require(FactoMineR)
```

```
## Loading required package: FactoMineR
```

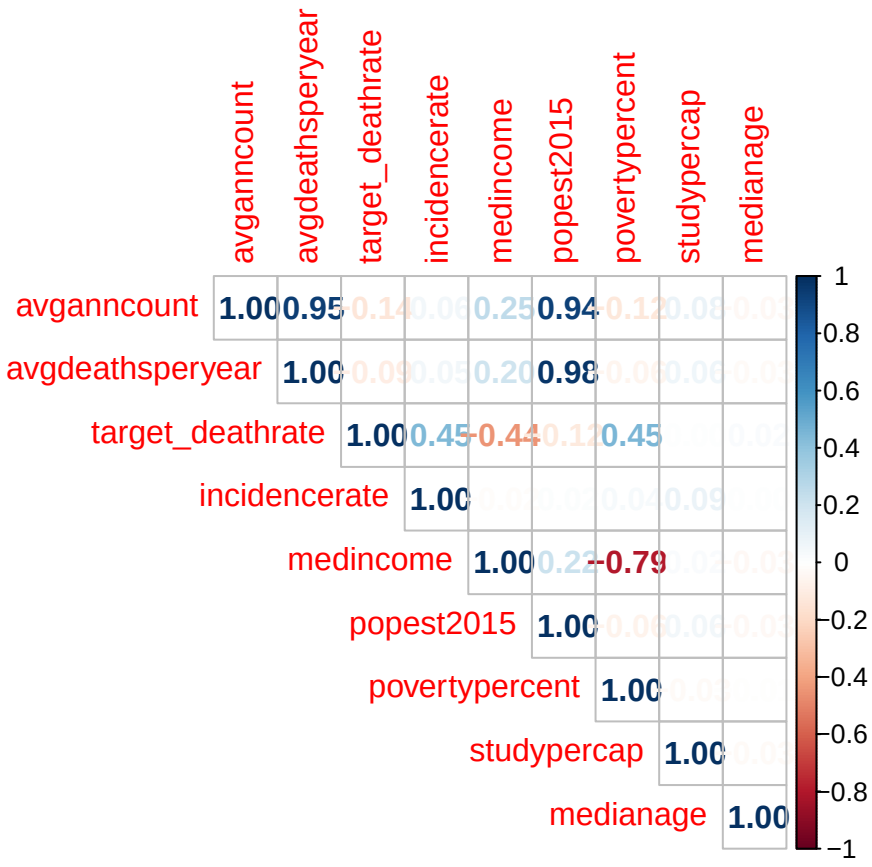
```r
res.con = condes(train_imputed,3)
res.con$quanti
```

```
##                          correlation      p.value
## pctpubliccoveragealone   0.46629578 1.687857e-99
## povertypercent           0.45008465 5.144264e-92
## incidencerate            0.44954847 8.953861e-92
## pctpubliccoverage        0.42436751 6.056303e-81
## pctunemployed16_over     0.39780681 1.777035e-70
## pctblack                 0.26875452 1.156196e-31
## birthrate               -0.06756043 3.824884e-03
## avgdeathsperyear        -0.09167650 8.549496e-05
## popest2015              -0.11902232 3.261480e-07
## avganncount             -0.13886454 2.417857e-09
## pctwhite                -0.18152553 4.983753e-15
## pctotherrace            -0.18153593 4.965708e-15
## pctasian                -0.19436412 4.806194e-17
## percentmarried          -0.27253733 1.511951e-32
## pctempprivcoverage      -0.28877690 1.677032e-36
## pctmarriedhouseholds    -0.29440643 6.182702e-38
## pctprivatecoverage      -0.40354149 1.173008e-72
## pctprivatecoveragealone -0.41693035 6.443710e-78
## medincome               -0.44368280 3.606607e-89
```

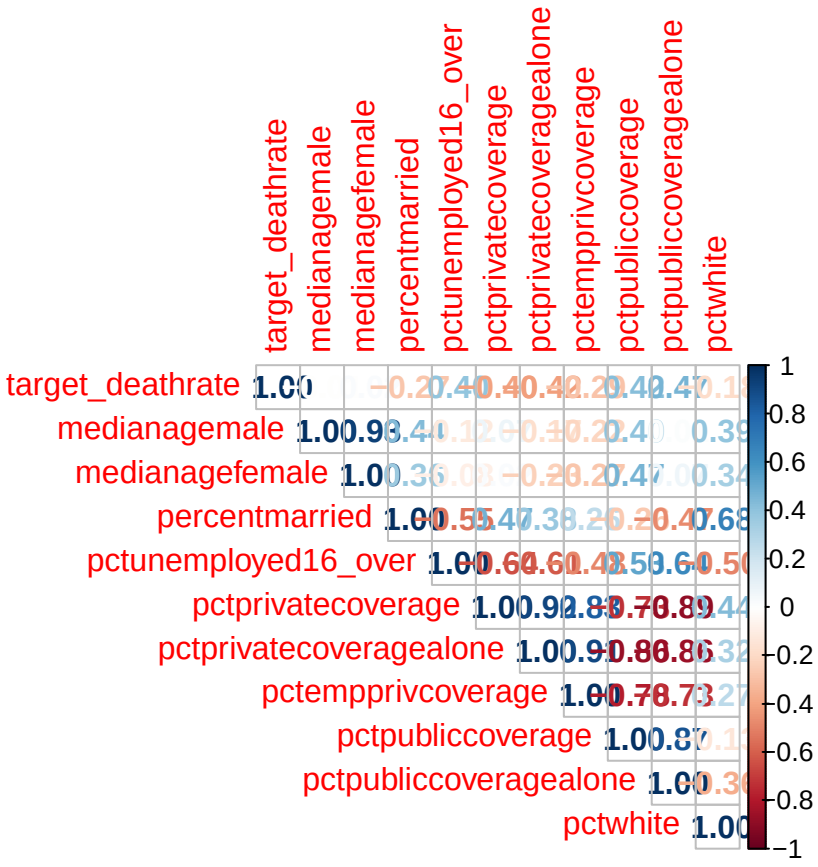
```r
res.con$quali
```

```
##                  R2      p.value
## binnedinc 0.2217324 7.669702e-93
```

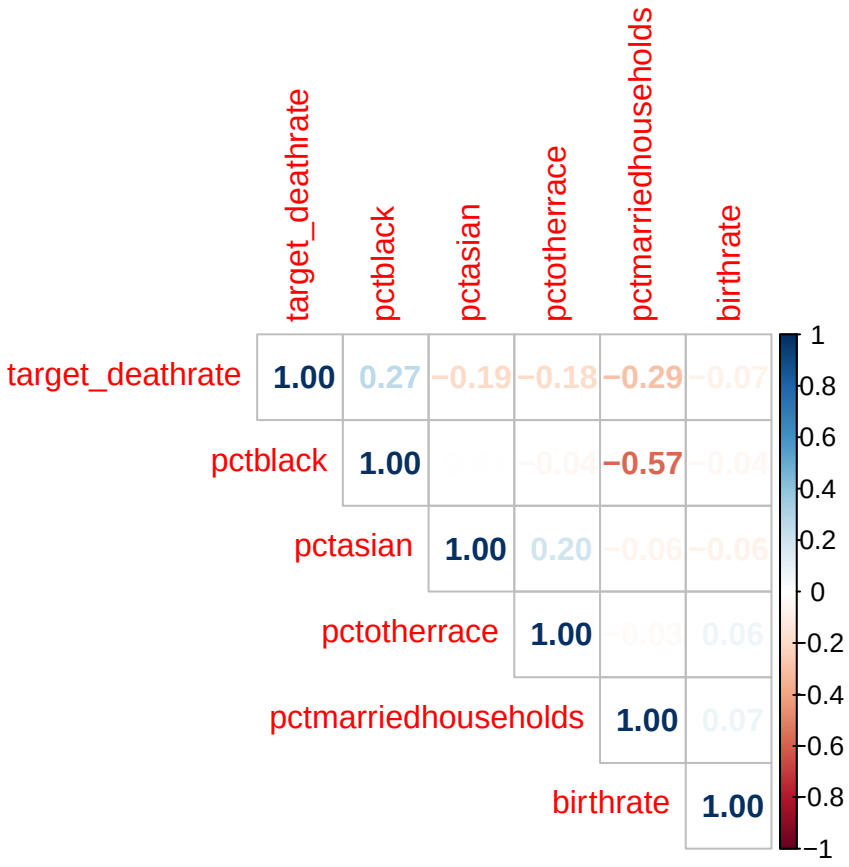
```r
# Create a df with the selected variables
df_selected1 <- train_imputed[ ,c(1:8,10)]
df_selected2 <- train_imputed [,c(3,11:20)]
df_selected3 <- train_imputed [,c(3,21:25)]
# Calculate the correlation matrix
cor_matrix1 <- cor(df_selected1, use = "complete.obs")
cor_matrix2 <- cor(df_selected2, use = "complete.obs")
cor_matrix3 <- cor(df_selected3, use = "complete.obs")
# Plot the correlation matrix
corrplot(cor_matrix1, method = 'number', type = "upper")
```




```
corrplot(cor_matrix2, method = 'number', type = "upper")
```



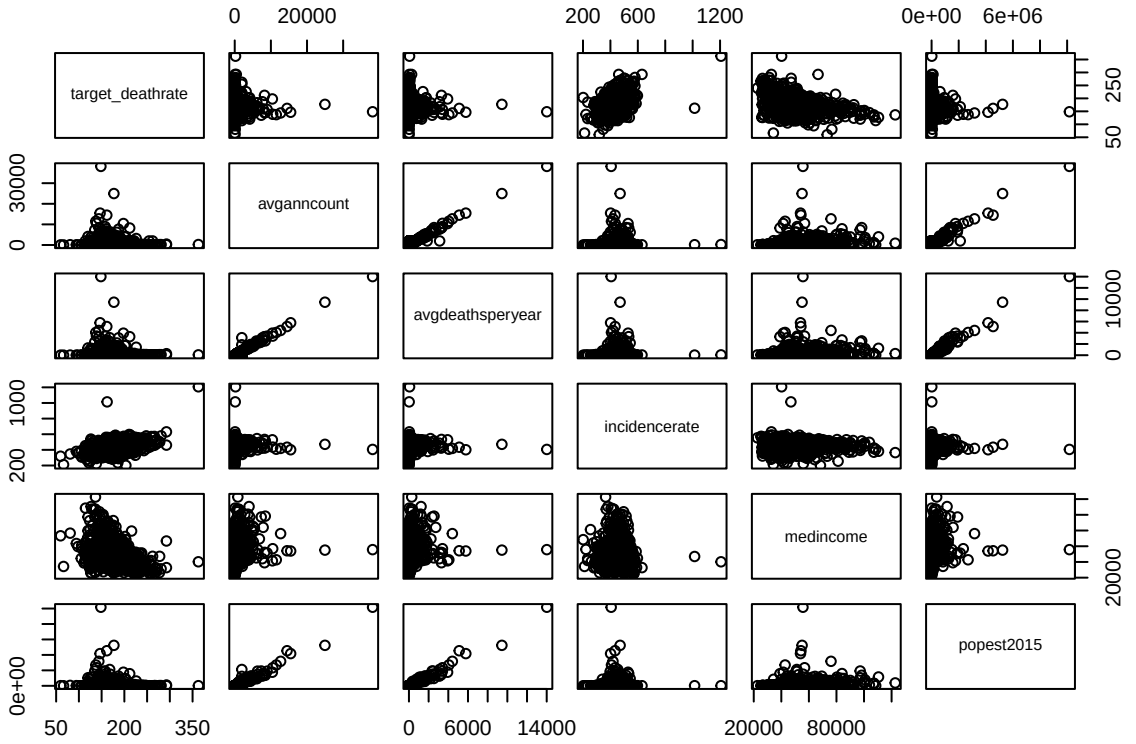

```
corrplot(cor_matrix3, method = 'number', type = "upper")
```



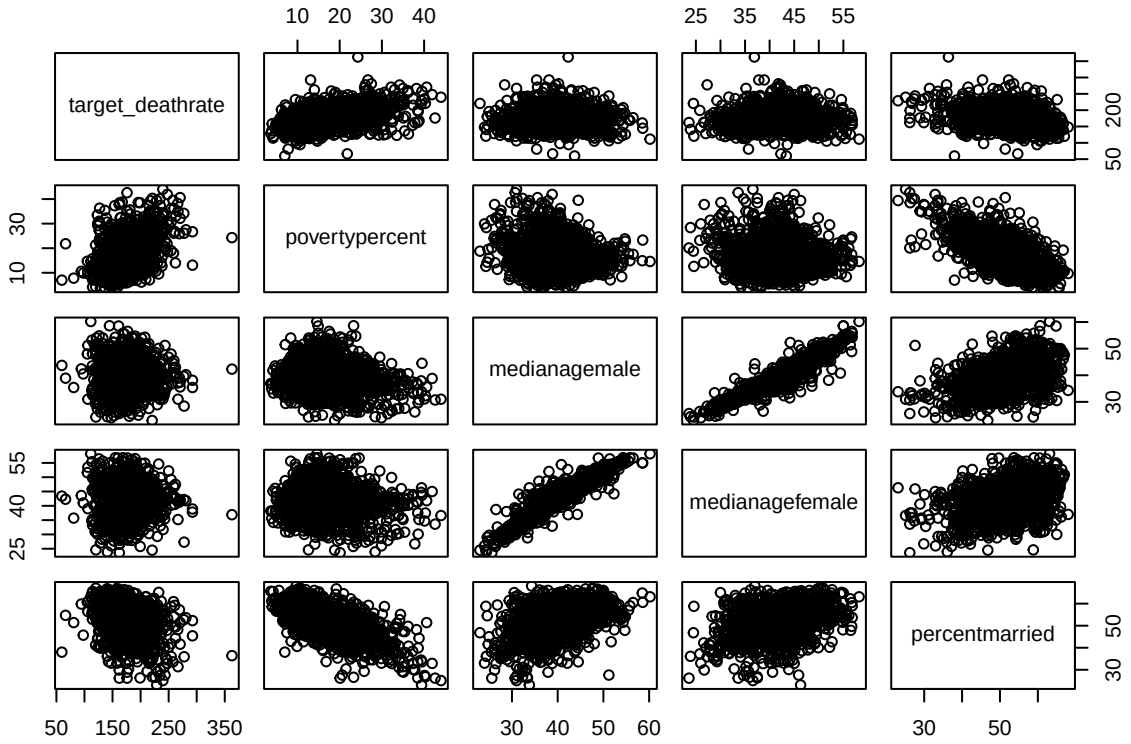
8. If you can improve linear relations or limit the effect of influential data, you must consider suitable transformations for variables.

When observing the pairwise scatter plot, no non-linear patterns are observed between predictors and the response variable, that is, it is not necessary to apply logarithmic or power transformations.

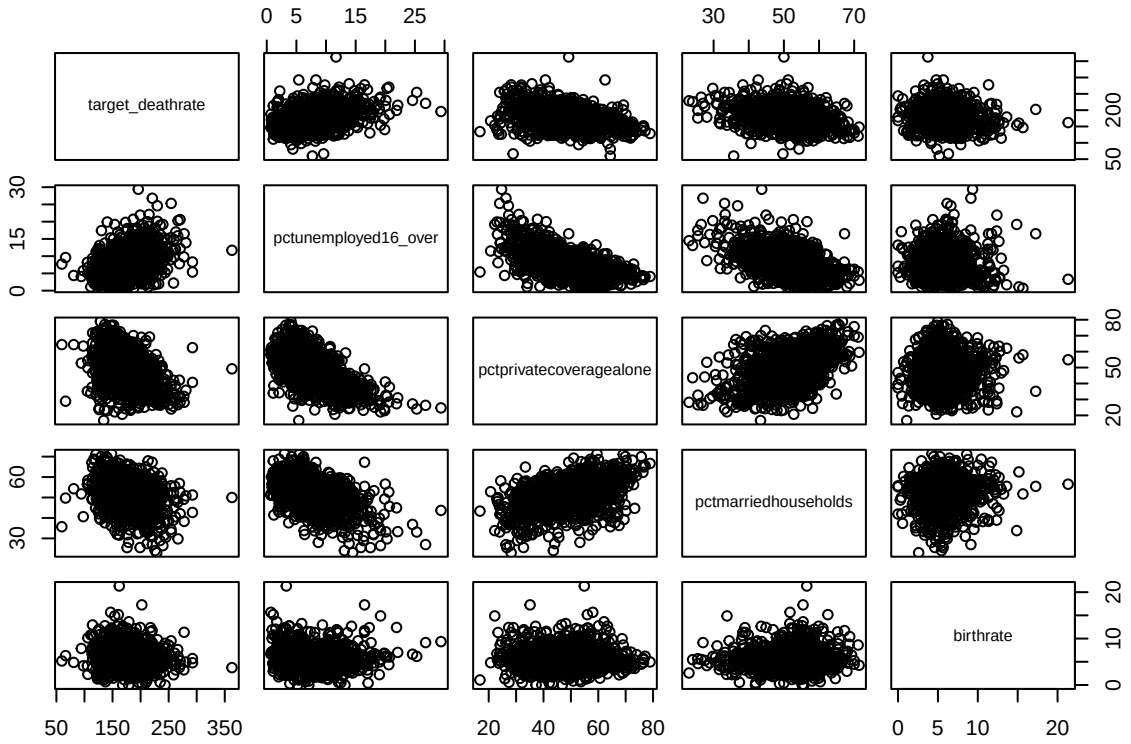
```
pairs(train_imputed[,c("target_deathrate","avganncount",
"avgdeathsperyear","incidencerate" ,"medincome", "popest2015")])
```




```
pairs(train_imputed[,c("target_deathrate","povertypercent" ,
    "medianagemale","medianagefemale", "percentmarried")])
```




```
pairs(train_imputed[,c("target_deathrate", "pctunemployed16_over","pctprivatecoveragealone", "pctmarrie
```



9. Apart from the retained factor variables, you can consider other categorical variables that can be defined from categorized numeric variables. Do not forget to implement new variable definitions in the test sample.

The numerical variables are categorized below to assess whether their influence is better. When evaluating their r2 they do not exceed 20%, so it is better to work with them in numerical format.

```
train_imputed$f.avganncount <- ifelse(train_imputed$avganncount <= 80, 1,
     ifelse(train_imputed$avganncount > 80 & train_imputed$avganncount <= 175, 2,
  ifelse(train_imputed$avganncount > 175 & train_imputed$avganncount <= 509, 3,
       ifelse(train_imputed$avganncount > 509, 4,0))))

train_imputed$f.avganncount <- factor(train_imputed$f.avganncount,
 labels=c("LowAvganncount","LowMidAvganncount","HighMidAvganncount","HighAvganncount"),
     order = T, levels=c(1,2,3,4))

train_imputed$f.avgdeathsperyear <- ifelse(train_imputed$avgdeathsperyear <= 29, 1,
 ifelse(train_imputed$avgdeathsperyear > 29 & train_imputed$avgdeathsperyear <= 62, 2,
 ifelse(train_imputed$avgdeathsperyear > 62 & train_imputed$avgdeathsperyear <= 140.5, 3,
          ifelse(train_imputed$avgdeathsperyear > 104.5, 4,0))))

train_imputed$f.avgdeathsperyear <- factor(train_imputed$f.avgdeathsperyear,
        order = T,        levels=c(1,2,3,4))

train_imputed$f.incidencerate <- ifelse(train_imputed$incidencerate <= 421.4000, 1,
```

```
    ifelse(train_imputed$incidencerate > 421.4000 & train_imputed$incidencerate <= 453.5494, 2,
   ifelse(train_imputed$incidencerate > 453.5494 & train_imputed$incidencerate <= 481.3000, 3,
       ifelse(train_imputed$incidencerate > 481.3000, 4,0))))

train_imputed$f.incidencerate <- factor(train_imputed$f.incidencerate,                                  labe
            order = T,     levels=c(1,2,3,4))
table(train_imputed$f.incidencerate)
```

```
##
##     LowIncidencerate  LowMidIncidencerate HighMidIncidencerate
##                  460                  409                  504
##    HighIncidencerate
##                  458
```

```
train_imputed$f.medincome <- ifelse(train_imputed$medincome <= 39031, 1,
   ifelse(train_imputed$medincome > 39031 & train_imputed$medincome <= 45454, 2,
       ifelse(train_imputed$medincome > 45454 & train_imputed$medincome <= 52612, 3,
                  ifelse(train_imputed$medincome > 52612, 4,0))))

train_imputed$f.medincome <- factor(train_imputed$f.medincome,
    labels=c("LowMedincome","LowMidMedincome","HighMidMedincome","HighMedincome"),
      order = T,            levels=c(1,2,3,4))

train_imputed$f.povertypercent <- ifelse(train_imputed$povertypercent <= 12.15, 1,
      ifelse(train_imputed$povertypercent > 12.15 & train_imputed$povertypercent <= 15.70, 2,
          ifelse(train_imputed$povertypercent > 15.70 & train_imputed$povertypercent <= 20.40, 3,
     ifelse(train_imputed$povertypercent > 20.40, 4,0))))

train_imputed$f.povertypercent <- factor(train_imputed$f.povertypercent,
labels=c("LowPovertypercent","LowMidPovertypercent","HighMidPovertypercent","HighPovertypercent"),
order = T,      levels=c(1,2,3,4))

train_imputed$f.percentmarried <- ifelse(train_imputed$percentmarried <= 47.8, 1,
  ifelse(train_imputed$percentmarried > 47.8 & train_imputed$percentmarried <= 52.5, 2,
  ifelse(train_imputed$percentmarried > 52.5 & train_imputed$percentmarried <= 56.4, 3,
ifelse(train_imputed$percentmarried > 56.4, 4,0))))

train_imputed$f.percentmarried <- factor(train_imputed$f.percentmarried,
 labels=c("Lowpercentmarried","LowMidpercentmarried","HighMidpercentmarried","Highpercentmarried"),  orc

train_imputed$f.pctunemployed16_over <- ifelse(train_imputed$pctunemployed16_over <= 48.6, 1,
    ifelse(train_imputed$pctunemployed16_over > 48.6 & train_imputed$pctunemployed16_over <= 54.21, 2,
     ifelse(train_imputed$pctunemployed16_over > 54.21 & train_imputed$pctunemployed16_over <= 60.3, 3,
     ifelse(train_imputed$pctunemployed16_over > 60.3, 4,0))))

train_imputed$f.pctunemployed16_over <- factor(train_imputed$f.pctunemployed16_over,
    labels=c("Lowpctunemployed16_over","LowMidpctunemployed16_over",
          "HighMidpctunemployed16_over","Highpctunemployed16_over"),
          order = T,    levels=c(1,2,3,4))

train_imputed$f.pctpubliccoveragealonet <- ifelse(train_imputed$pctpubliccoveragealone <= 14.9, 1,
ifelse(train_imputed$pctpubliccoveragealone > 14.9 & train_imputed$pctpubliccoveragealone <= 18.7, 2,
 ifelse(train_imputed$pctpubliccoveragealone > 18.7 & train_imputed$pctpubliccoveragealone <= 23, 3,
```

```
 ifelse(train_imputed$pctpubliccoveragealone > 23, 4,0))))

train_imputed$f.pctpubliccoveragealone <- factor(train_imputed$f.pctpubliccoveragealone,
labels=c("Lowpctpubliccoveragealone","LowMidpctpubliccoveragealone","HighMidpctpubliccoveragealone","Hig

train_imputed$f.pctblack<- ifelse(train_imputed$pctblack <= 0.648, 1,
ifelse(train_imputed$pctblack > 0.648 & train_imputed$pctblack <= 2.323, 2,
 ifelse(train_imputed$pctblack > 2.323 & train_imputed$pctblack <= 10.867, 3,
  ifelse(train_imputed$pctblack > 10.867, 4,0))))

train_imputed$f.pctblack <- factor(train_imputed$f.pctblack,
  labels=c("Lowpctblack","LowMidpctblack","HighMidpctblack","Highpctblack"),
                        order = T, levels=c(1,2,3,4))

res.con2 = catdes(train_imputed[,c(26:35)],num.var=3, proba = 0.05)
res.con2$category
```

```
## $LowIncidencerate
##                                                  Cla/Mod  Mod/Cla   Global
## f.avganncount=LowAvganncount                    47.82609 47.82609 25.12288
## f.avgdeathsperyear=LowAvgdeathsperyear          41.34199 41.52174 25.23211
## f.pctblack=Lowpctblack                          34.49782 34.34783 25.01365
## f.percentmarried=Highpercentmarried             32.16630 31.95652 24.95904
## f.medincome=LowMedincome                        29.25764 29.13043 25.01365
## f.pctpubliccoveragealone=Lowpctpubliccoveragealone 28.72570 28.91304 25.28673
## f.avgdeathsperyear=HighMidAvgdeathsperyear      20.39474 20.21739 24.90442
## f.avganncount=HighMidAvganncount                16.70330 16.52174 24.84981
## f.percentmarried=LowMidpercentmarried           16.55773 16.52174 25.06827
## f.pctblack=Highpctblack                         16.37555 16.30435 25.01365
## f.avgdeathsperyear=HighAvgdeathsperyear         12.66376 12.60870 25.01365
## f.avganncount=HighAvganncount                   10.04367 10.00000 25.01365
##                                                      p.value    v.test
## f.avganncount=LowAvganncount                    1.272451e-35 12.457534
## f.avgdeathsperyear=LowAvgdeathsperyear          2.542440e-19  8.986925
## f.pctblack=Lowpctblack                          1.789085e-07  5.220015
## f.percentmarried=Highpercentmarried             8.314522e-05  3.935149
## f.medincome=LowMedincome                        1.977233e-02  2.330640
## f.pctpubliccoveragealone=Lowpctpubliccoveragealone 4.046667e-02  2.048953
## f.avgdeathsperyear=HighMidAvgdeathsperyear      6.579251e-03 -2.717423
## f.avganncount=HighMidAvganncount                9.133833e-07 -4.909437
## f.percentmarried=LowMidpercentmarried           4.999453e-07 -5.026334
## f.pctblack=Highpctblack                         2.892914e-07 -5.130293
## f.avgdeathsperyear=HighAvgdeathsperyear         9.387064e-14 -7.449252
## f.avganncount=HighAvganncount                   4.921593e-20 -9.165723
##
## $LowMidIncidencerate
##                                                  Cla/Mod  Mod/Cla   Global
## f.pctpubliccoveragealone=Lowpctpubliccoveragealone 26.13391 29.58435 25.28673
## f.avganncount=HighMidAvganncount                25.71429 28.60636 24.84981
## f.pctblack=Lowpctblack                          18.77729 21.02689 25.01365
## f.avganncount=HighAvganncount                   17.46725 19.55990 25.01365
##                                                      p.value    v.test
## f.pctpubliccoveragealone=Lowpctpubliccoveragealone 0.024921869  2.242612
```

```
## f.avganncount=HighMidAvganncount                    0.048315620  1.974582
## f.pctblack=Lowpctblack                              0.033105730 -2.130799
## f.avganncount=HighAvganncount                       0.003347383 -2.933894
##
## $HighMidIncidencerate
##                                                      Cla/Mod  Mod/Cla
## f.avganncount=HighAvganncount                       49.12664 44.64286
## f.pctpubliccoveragealone=LowMidpctpubliccoveragealone 33.47732 30.75397
## f.medincome=HighMidMedincome                        33.04158 29.96032
## f.povertypercent=LowPovertypercent                  31.87773 28.96825
## f.avgdeathsperyear=HighAvgdeathsperyear             31.22271 28.37302
## f.avgdeathsperyear=LowAvgdeathsperyear              23.16017 21.23016
## f.avganncount=LowMidAvganncount                     22.27074 20.23810
## f.povertypercent=HighPovertypercent                 20.92511 18.84921
## f.pctpubliccoveragealone=Highpctpubliccoveragealone 20.44444 18.25397
## f.medincome=LowMedincome                            20.30568 18.45238
## f.avganncount=LowAvganncount                        12.82609 11.70635
##                                                       Global      p.value
## f.avganncount=HighAvganncount                       25.01365 7.219219e-31
## f.pctpubliccoveragealone=LowMidpctpubliccoveragealone 25.28673 1.064475e-03
## f.medincome=HighMidMedincome                        24.95904 2.617274e-03
## f.povertypercent=LowPovertypercent                  25.01365 1.711137e-02
## f.avgdeathsperyear=HighAvgdeathsperyear             25.01365 4.247655e-02
## f.avgdeathsperyear=LowAvgdeathsperyear              25.23211 1.429698e-02
## f.avganncount=LowMidAvganncount                     25.01365 3.272958e-03
## f.povertypercent=HighPovertypercent                 24.79519 2.241306e-04
## f.pctpubliccoveragealone=Highpctpubliccoveragealone 24.57673 8.052038e-05
## f.medincome=LowMedincome                            25.01365 4.721946e-05
## f.avganncount=LowAvganncount                        25.12288 8.988895e-18
##                                                        v.test
## f.avganncount=HighAvganncount                       11.551917
## f.pctpubliccoveragealone=LowMidpctpubliccoveragealone  3.272907
## f.medincome=HighMidMedincome                         3.009443
## f.povertypercent=LowPovertypercent                   2.384306
## f.avgdeathsperyear=HighAvgdeathsperyear              2.028821
## f.avgdeathsperyear=LowAvgdeathsperyear              -2.449714
## f.avganncount=LowMidAvganncount                     -2.940866
## f.povertypercent=HighPovertypercent                 -3.690140
## f.pctpubliccoveragealone=Highpctpubliccoveragealone -3.942846
## f.medincome=LowMedincome                            -4.068980
## f.avganncount=LowAvganncount                        -8.586205
##
## $HighIncidencerate
##                                                      Cla/Mod  Mod/Cla   Global
## f.avganncount=HighMidAvganncount                    31.64835 31.44105 24.84981
## f.avgdeathsperyear=HighAvgdeathsperyear             31.00437 31.00437 25.01365
## f.pctpubliccoveragealone=Highpctpubliccoveragealone 29.55556 29.03930 24.57673
## f.pctblack=Highpctblack                             29.47598 29.47598 25.01365
## f.percentmarried=LowMidpercentmarried               29.19390 29.25764 25.06827
## f.percentmarried=Lowpercentmarried                  28.91304 29.03930 25.12288
## f.povertypercent=LowPovertypercent                  21.17904 21.17904 25.01365
## f.pctblack=Lowpctblack                              19.86900 19.86900 25.01365
## f.pctpubliccoveragealone=Lowpctpubliccoveragealone  19.00648 19.21397 25.28673
## f.percentmarried=Highpercentmarried                 18.59956 18.55895 24.95904
```

```
## f.avganncount=LowAvganncount                                18.26087 18.34061 25.12288
## f.avgdeathsperyear=LowAvgdeathsperyear                      16.23377 16.37555 25.23211
##                                                                  p.value    v.test
## f.avganncount=HighMidAvganncount                            2.120643e-04  3.704194
## f.avgdeathsperyear=HighAvgdeathsperyear                     7.661190e-04  3.364754
## f.pctpubliccoveragealone=Highpctpubliccoveragealone         1.138759e-02  2.530574
## f.pctblack=Highpctblack                                     1.185700e-02  2.516372
## f.percentmarried=LowMidpercentmarried                       1.814380e-02  2.362670
## f.percentmarried=Lowpercentmarried                          2.717509e-02  2.208994
## f.povertypercent=LowPovertypercent                          2.741500e-02 -2.205557
## f.pctblack=Lowpctblack                                      2.928909e-03 -2.975102
## f.pctpubliccoveragealone=Lowpctpubliccoveragealone          4.455796e-04 -3.511503
## f.percentmarried=Highpercentmarried                         1.963628e-04 -3.723650
## f.avganncount=LowAvganncount                                8.049243e-05 -3.942929
## f.avgdeathsperyear=LowAvgdeathsperyear                      2.122331e-07 -5.188291
```

The variable Predominant Race by County is created to evaluate whether the presence of any ethnic origin is related to the mortality rate. When generating this categorical variable, it is observed that its categories are unbalanced since the majority of the counties are predominantly identified as white race. In addition, the explained variability of the mortality rate is 3%, which does not make it suitable for entering the model.

```
raza_train<-train_imputed [,21:24]
# predominant level position
pos<-apply(raza_train,1,which.max)
pos<-as.numeric(pos)
head(pos)
```

```
## [1] 4 4 4 4 4 4
```

```
raza_princ<-names(raza_train)
raza_princ
```

```
## [1] "pctblack"             "pctasian"             "pctotherrace"
## [4] "pctmarriedhouseholds"
```

```
raza_princ<-raza_princ[pos]
train_imputed$raza_princ<-raza_princ
table(raza_princ)
```

```
## raza_princ
##             pctblack pctmarriedhouseholds         pctotherrace
##                   86                 1744                    1
```

```
res.con = condes(train_imputed[,c(1:25)],3)
res.con$quali
```

```
##                   R2      p.value
## binnedinc 0.2217324 7.669702e-93
```

10. You must take into account possible interactions between categorical and numerical variables.

Among the variables that could interact with the average per capita income classified by decile (categorical variable) are incidencerate (cancer incidence rate), studypercap (clinical trials per capita related to cancer) and povertypercent (percentage of population in poverty). Of these 3, only the povertypercent variable (percentage of population in poverty) is significant when looking for associated variables. A high value of eta2 could indicate multicollinearity problems with the average per capita income classified by decile. Now, it is reasonable to assume that povertypercent (percentage of population in poverty) can interact with incidencerate (cancer incidence rate). This is because at low levels of per capita income there should be high values of the cancer mortality rate and for high levels of per capita income there should be low values of the cancer mortality rate. When evaluating the slope graph, the rate at the different income levels presents positive slopes, which does not show drastic changes in the effect of the mortality rate. Similarly, when comparing the adjusted coefficient of determination between a model without interaction with a model with interaction, there is no significant increase (Adjusted R2 remains at 22.6%). In addition, the ANOVA results show that the interaction terms in Model 2 do not significantly improve the model's fit ($p = 0.2524$). The small reduction in RSS does not justify the added complexity. The AIC comparison also confirms that Model 1 is more optimal, as the interaction terms increase complexity without improving the fit meaningfully.

```
res.con2 = catdes(train_imputed,9)
res.con2$quanti.var
```

```
##                                 Eta2       P-value
## medincome                 0.89302352  0.000000e+00
## povertypercent            0.75189709  0.000000e+00
## pctprivatecoverage        0.61771315  0.000000e+00
## pctprivatecoveragealone   0.69219588  0.000000e+00
## pctempprivcoverage        0.57353419  0.000000e+00
## pctpubliccoverage         0.61420964  0.000000e+00
## pctpubliccoveragealone    0.59976086  0.000000e+00
## f.pctpubliccoveragealonet 0.56096961  0.000000e+00
## pctunemployed16_over      0.30967842 9.415557e-140
## pctmarriedhouseholds      0.22640464  3.429754e-95
## target_deathrate          0.22173238  7.669702e-93
## pctblack                  0.18577066  3.014792e-75
## percentmarried            0.17376436  1.467348e-69
## pctasian                  0.16270131  2.121730e-64
## pctwhite                  0.13579587  3.635234e-52
## avganncount               0.05998555  3.838843e-20
## popest2015                0.04404371  5.954300e-14
## medianagefemale           0.04143084  5.803154e-13
## avgdeathsperyear          0.04093714  8.903341e-13
## medianagemale             0.02444169  8.664131e-07
```

```
mod1<-lm(target_deathrate~povertypercent+binnedinc,train_imputed)
summary(mod1)
```

```
##
## Call:
## lm(formula = target_deathrate ~ povertypercent + binnedinc, data = train_imputed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -125.330  -13.420    1.228   14.282  171.221
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  175.1584     4.5335  38.636  < 2e-16 ***
## povertypercent                 0.7556     0.1800   4.198 2.83e-05 ***
## binnedinc(37413.8, 40362.7]   -1.9398     2.7306  -0.710  0.47755
## binnedinc(40362.7, 42724.4]   -6.2703     2.7623  -2.270  0.02333 *
## binnedinc(42724.4, 45201]     -8.8891     2.8130  -3.160  0.00160 **
## binnedinc(45201, 48021.6]     -9.4297     2.9293  -3.219  0.00131 **
## binnedinc(48021.6, 51046.4] -14.3567     3.0795  -4.662 3.36e-06 ***
## binnedinc(51046.4, 54545.6] -15.4224     3.1916  -4.832 1.46e-06 ***
## binnedinc(54545.6, 61494.5] -17.8215     3.3281  -5.355 9.65e-08 ***
## binnedinc(61494.5, 125635]  -23.3352     3.6312  -6.426 1.67e-10 ***
## binnedinc[22640, 34218.1]     8.2874     2.7429   3.021  0.00255 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.59 on 1820 degrees of freedom
## Multiple R-squared:  0.2292, Adjusted R-squared:  0.225
## F-statistic: 54.12 on 10 and 1820 DF,  p-value: < 2.2e-16
```

```
summary(mod1)$adj.r.squared
```

```
## [1] 0.2249598
```

```
mod2<-lm(target_deathrate~povertypercent*binnedinc,train_imputed)
summary(mod2)
```

```
##
## Call:
## lm(formula = target_deathrate ~ povertypercent * binnedinc, data = train_imputed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -125.742  -13.715    0.976   14.216  175.507
##
## Coefficients:
##                                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                                    183.5041    11.9620  15.341  < 2e-16
## povertypercent                                   0.3916     0.5152   0.760  0.44726
## binnedinc(37413.8, 40362.7]                     10.3977    16.8749   0.616  0.53786
## binnedinc(40362.7, 42724.4]                     -5.3549    16.1638  -0.331  0.74046
## binnedinc(42724.4, 45201]                       -4.1815    15.5503  -0.269  0.78804
## binnedinc(45201, 48021.6]                      -32.7164    16.2113  -2.018  0.04373
## binnedinc(48021.6, 51046.4]                    -25.2602    15.8941  -1.589  0.11217
## binnedinc(51046.4, 54545.6]                    -22.1676    15.4989  -1.430  0.15281
## binnedinc(54545.6, 61494.5]                    -29.7803    15.1710  -1.963  0.04980
## binnedinc(61494.5, 125635]                     -38.9299    13.4969  -2.884  0.00397
## binnedinc[22640, 34218.1]                      -11.9488    15.3012  -0.781  0.43496
## povertypercent:binnedinc(37413.8, 40362.7]      -0.6636     0.7784  -0.852  0.39408
## povertypercent:binnedinc(40362.7, 42724.4]      -0.1332     0.7719  -0.173  0.86303
## povertypercent:binnedinc(42724.4, 45201]        -0.4204     0.7814  -0.538  0.59067
```

```
## povertypercent:binnedinc(45201, 48021.6]     1.3356     0.8707   1.534  0.12523
## povertypercent:binnedinc(48021.6, 51046.4]   0.5531     0.9205   0.601  0.54804
## povertypercent:binnedinc(51046.4, 54545.6]   0.2402     0.9083   0.264  0.79144
## povertypercent:binnedinc(54545.6, 61494.5]   0.6835     0.9600   0.712  0.47659
## povertypercent:binnedinc(61494.5, 125635]    1.1825     0.8499   1.391  0.16429
## povertypercent:binnedinc[22640, 34218.1]     0.7859     0.6133   1.281  0.20022
##
## (Intercept)                                 ***
## povertypercent
## binnedinc(37413.8, 40362.7]
## binnedinc(40362.7, 42724.4]
## binnedinc(42724.4, 45201]
## binnedinc(45201, 48021.6]                   *
## binnedinc(48021.6, 51046.4]
## binnedinc(51046.4, 54545.6]
## binnedinc(54545.6, 61494.5]                 *
## binnedinc(61494.5, 125635]                  **
## binnedinc[22640, 34218.1]
## povertypercent:binnedinc(37413.8, 40362.7]
## povertypercent:binnedinc(40362.7, 42724.4]
## povertypercent:binnedinc(42724.4, 45201]
## povertypercent:binnedinc(45201, 48021.6]
## povertypercent:binnedinc(48021.6, 51046.4]
## povertypercent:binnedinc(51046.4, 54545.6]
## povertypercent:binnedinc(54545.6, 61494.5]
## povertypercent:binnedinc(61494.5, 125635]
## povertypercent:binnedinc[22640, 34218.1]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.57 on 1811 degrees of freedom
## Multiple R-squared:  0.234,  Adjusted R-squared:  0.226
## F-statistic: 29.12 on 19 and 1811 DF,  p-value: < 2.2e-16
```

```
summary(mod2)$adj.r.squared
```

```
## [1] 0.2259654
```

```
anova(mod1, mod2)
```

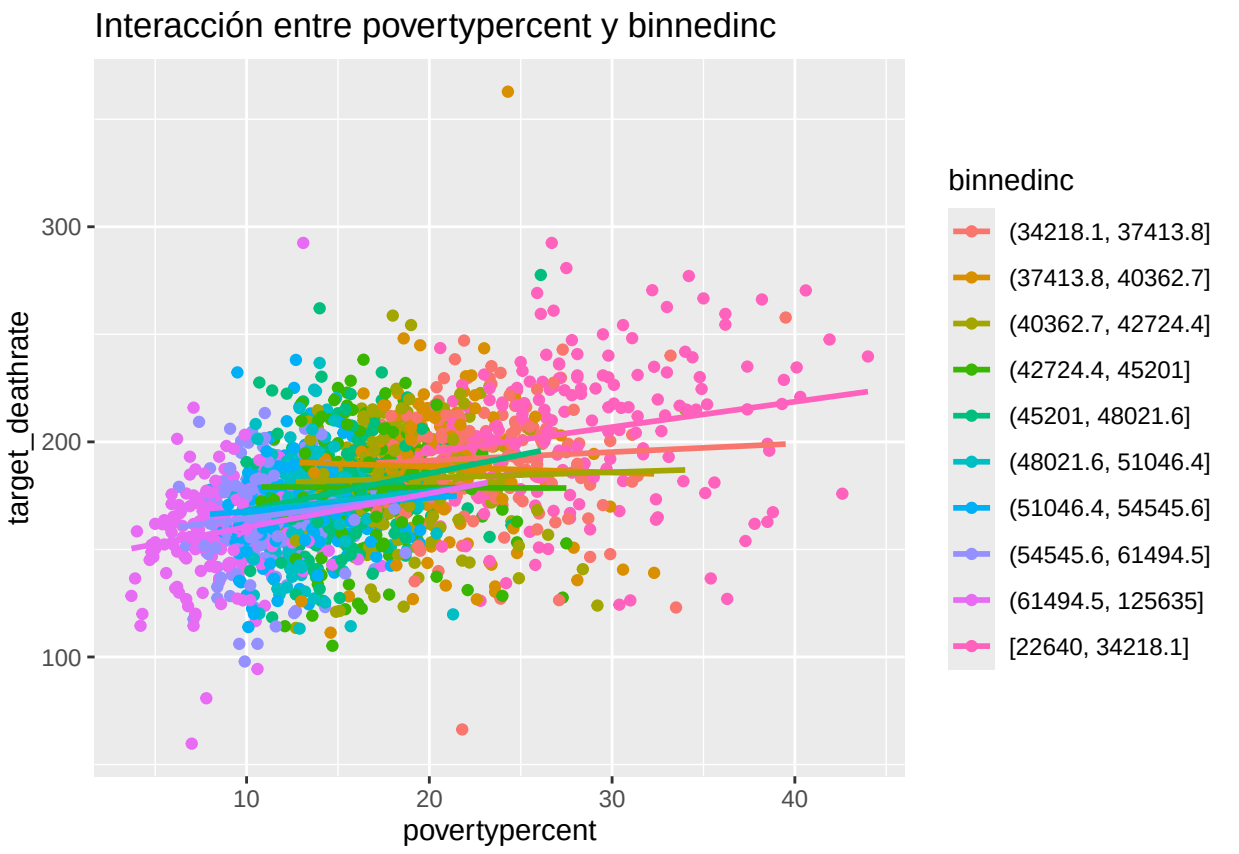
```
## Analysis of Variance Table
##
## Model 1: target_deathrate ~ povertypercent + binnedinc
## Model 2: target_deathrate ~ povertypercent * binnedinc
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1   1820 1100400
## 2   1811 1093537  9    6862.2 1.2627 0.2524
```

```
AIC(mod1, mod2)
```

```
##      df      AIC
## mod1 12 16935.93
## mod2 21 16942.47
```

```r
library(ggplot2)

# Example with incidencerate
ggplot(train_imputed, aes(x = povertypercent, y = target_deathrate, color = binnedinc)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Interacción entre povertypercent y binnedinc")
```

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



11. When building the model, you should study the presence of multicollinearity and try to reduce their impact on the model for easier interpretation.

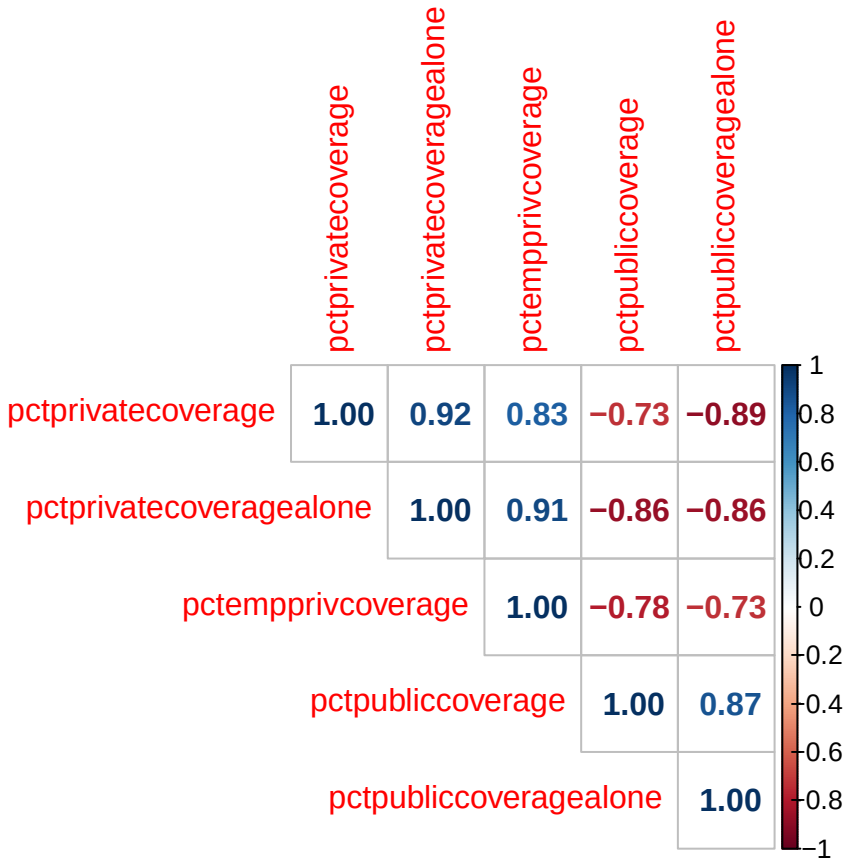
To select the variables to be entered into the model, a determination coefficient of 25% was considered. It is observed that the variables related to health care meet this requirement. However, it is expected that these variables are highly correlated since patients who are not affiliated with public insurance will be covered by private insurance. A similar situation occurs between the percentage of employed and unemployed. The correlation graphs reflect this situation so that to deal with it, the variable that is most correlated with the mortality rate is chosen, namely, pctpubliccoveragealone, pctemployed16_over. The other variables that are chosen are povertypercent, incidencerate, pctblack, pctmarriedhouseholds, pctemployed16_over, medincome and binnedinc. When estimating a regression model with these variables, an adjusted determination coefficient of approximately 44% is obtained. However, when evaluating the VIF indicator, the numerical variables are found to have VIF values less than 5, which shows that there are no multicollinearity problems

between these variables. However, when evaluating the categorical variable of the income decile, the maximum tolerable value is 5^(1/(2*9))=1.0935, which is less than the modified generalized VIF indicator for said variable. This is because this variable is associated with the average per capita income. This variable is excluded from the model and it is observed that the numerical variables maintain a VIF less than 5.

```
relation<-data.frame(res.con$quanti)
relation[(abs(relation$correlation)>0.25),]
```

```
##                          correlation      p.value
## pctpubliccoveragealone     0.4662958 1.687857e-99
## povertypercent             0.4500846 5.144264e-92
## incidencerate              0.4495485 8.953861e-92
## pctpubliccoverage          0.4243675 6.056303e-81
## pctunemployed16_over       0.3978068 1.777035e-70
## pctblack                   0.2687545 1.156196e-31
## percentmarried            -0.2725373 1.511951e-32
## pctempprivcoverage        -0.2887769 1.677032e-36
## pctmarriedhouseholds      -0.2944064 6.182702e-38
## pctprivatecoverage        -0.4035415 1.173008e-72
## pctprivatecoveragealone   -0.4169303 6.443710e-78
## medincome                 -0.4436828 3.606607e-89
```

```
salud<-train_imputed[,c("pctprivatecoverage","pctprivatecoveragealone",
      "pctempprivcoverage","pctpubliccoverage","pctpubliccoveragealone")]

cor_matrix_salud <- cor(salud, use = "complete.obs")
corrplot(cor_matrix_salud, method = 'number', type = "upper")
```




```
mod_1<-lm(target_deathrate~pctpubliccoveragealone +
          povertypercent+incidencerate+pctblack+
          pctmarriedhouseholds+
          pctunemployed16_over +medincome+binnedinc
          ,train_imputed)
summary(mod_1)
```

```
##
## Call:
## lm(formula = target_deathrate ~ pctpubliccoveragealone + povertypercent +
##     incidencerate + pctblack + pctmarriedhouseholds + pctunemployed16_over +
##     medincome + binnedinc, data = train_imputed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -126.310  -12.251   -0.295   11.757  125.581
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 7.953e+01  1.075e+01   7.396 2.14e-13 ***
## pctpubliccoveragealone      6.566e-01  1.522e-01   4.313 1.70e-05 ***
## povertypercent             -1.891e-01  2.028e-01  -0.933 0.351134
## incidencerate               2.149e-01  9.016e-03  23.839  < 2e-16 ***
## pctblack                    1.075e-01  4.636e-02   2.319 0.020497 *
## pctmarriedhouseholds        2.790e-01  1.066e-01   2.617 0.008944 **
## pctunemployed16_over        6.459e-01  2.053e-01   3.146 0.001685 **
```

```
## medincome                    -4.529e-04  1.267e-04  -3.574 0.000361 ***
## binnedinc(37413.8, 40362.7] -9.518e-01  2.350e+00  -0.405 0.685485
## binnedinc(40362.7, 42724.4] -4.462e+00  2.429e+00  -1.837 0.066365 .
## binnedinc(42724.4, 45201]   -6.550e+00  2.537e+00  -2.582 0.009913 **
## binnedinc(45201, 48021.6]   -7.147e+00  2.733e+00  -2.615 0.009004 **
## binnedinc(48021.6, 51046.4] -1.093e+01  2.988e+00  -3.657 0.000263 ***
## binnedinc(51046.4, 54545.6] -1.092e+01  3.256e+00  -3.355 0.000811 ***
## binnedinc(54545.6, 61494.5] -1.048e+01  3.661e+00  -2.862 0.004262 **
## binnedinc(61494.5, 125635]  -9.666e+00  5.232e+00  -1.847 0.064861 .
## binnedinc[22640, 34218.1]    5.899e+00  2.375e+00   2.484 0.013088 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.93 on 1814 degrees of freedom
## Multiple R-squared:  0.4432, Adjusted R-squared:  0.4383
## F-statistic: 90.26 on 16 and 1814 DF,  p-value: < 2.2e-16
```

```
library(car)
```

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

```
vif(mod_1)
```

```
##                             GVIF Df GVIF^(1/(2*Df))
## pctpubliccoveragealone  3.550590  1        1.884301
## povertypercent          7.058200  1        2.656727
## incidencerate           1.058615  1        1.028890
## pctblack                1.874794  1        1.369231
## pctmarriedhouseholds    1.975590  1        1.405557
## pctunemployed16_over    2.104390  1        1.450652
## medincome              10.217575  1        3.196494
## binnedinc              16.423616  9        1.168224
```

```
mod<-lm(target_deathrate~pctpubliccoveragealone +
          povertypercent+incidencerate+pctblack+
          pctmarriedhouseholds+
          pctunemployed16_over +medincome
        ,train_imputed)
summary(mod)
```

```
##
## Call:
## lm(formula = target_deathrate ~ pctpubliccoveragealone + povertypercent +
##     incidencerate + pctblack + pctmarriedhouseholds + pctunemployed16_over +
##     medincome, data = train_imputed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -125.455  -12.408   -0.122   11.792  125.398
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)            62.5890366  9.5416536   6.560 7.01e-11 ***
## pctpubliccoveragealone  0.7696759  0.1516964   5.074 4.30e-07 ***
## povertypercent          0.3252345  0.1815426   1.792 0.073378 .
## incidencerate           0.2123477  0.0090555  23.450  < 2e-16 ***
## pctblack                0.1251996  0.0461166   2.715 0.006693 **
## pctmarriedhouseholds    0.3885504  0.1052530   3.692 0.000229 ***
## pctunemployed16_over    0.6715878  0.2057707   3.264 0.001120 **
## medincome              -0.0005436  0.0000704  -7.722 1.87e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.07 on 1823 degrees of freedom
## Multiple R-squared:  0.4329, Adjusted R-squared:  0.4307
## F-statistic: 198.8 on 7 and 1823 DF,  p-value: < 2.2e-16
```

```
vif(mod)
```

```
## pctpubliccoveragealone         povertypercent          incidencerate
##               3.478505               5.582288               1.053638
##               pctblack   pctmarriedhouseholds   pctunemployed16_over
##               1.830704               1.899607               2.085334
##              medincome
##               3.110631
```

12. You should build the model using a technique for selecting variables (removing no significant predictors and/or stepwise selection of the best models).

With the variables chosen in the previous model, the best subset of predictors was selected using the bidirectional stepwise selection algorithm, finding that the best group of variables that explain the mortality rate are: pctpubliccoveragealone, incidencerate, pctblack, pctmarriedhouseholds, pctemployed16_over, medincome. With this, the variability of the mortality rate is explained at 43.39%.

```
step(mod, direction = "both")
```

```
## Start:  AIC=11169.8
## target_deathrate ~ pctpubliccoveragealone + povertypercent +
##     incidencerate + pctblack + pctmarriedhouseholds + pctunemployed16_over +
##     medincome
##
##                          Df Sum of Sq     RSS   AIC
## <none>                                  809573 11170
## - povertypercent          1      1425  810998 11171
## - pctblack                1      3273  812846 11175
## - pctunemployed16_over    1      4731  814303 11178
## - pctmarriedhouseholds    1      6052  815625 11181
## - pctpubliccoveragealone  1     11432  821005 11194
## - medincome               1     26481  836054 11227
## - incidencerate           1    244198 1053771 11650

##
## Call:
## lm(formula = target_deathrate ~ pctpubliccoveragealone + povertypercent +
```

```
##     incidencerate + pctblack + pctmarriedhouseholds + pctunemployed16_over +
##     medincome, data = train_imputed)
##
## Coefficients:
##            (Intercept)  pctpubliccoveragealone          povertypercent
##             62.5890366               0.7696759               0.3252345
##          incidencerate               pctblack    pctmarriedhouseholds
##              0.2123477               0.1251996               0.3885504
##   pctunemployed16_over               medincome
##              0.6715878              -0.0005436
```

```
mod_final<-lm(formula = target_deathrate ~ pctpubliccoveragealone + incidencerate + pctblack + pctmarri
summary(mod_final)
```

```
##
## Call:
## lm(formula = target_deathrate ~ pctpubliccoveragealone + incidencerate +
##     pctblack + pctmarriedhouseholds + pctunemployed16_over +
##     medincome, data = train_imputed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -122.933  -12.413   -0.013   11.629  125.578
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)             7.083e+01  8.366e+00   8.466  < 2e-16 ***
## pctpubliccoveragealone  8.788e-01  1.390e-01   6.321 3.26e-10 ***
## incidencerate           2.105e-01  8.999e-03  23.386  < 2e-16 ***
## pctblack                1.515e-01  4.375e-02   3.463 0.000547 ***
## pctmarriedhouseholds    3.552e-01  1.037e-01   3.427 0.000624 ***
## pctunemployed16_over    7.548e-01  2.006e-01   3.763 0.000173 ***
## medincome              -6.112e-04  5.948e-05 -10.276  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.09 on 1824 degrees of freedom
## Multiple R-squared:  0.4319, Adjusted R-squared:   0.43
## F-statistic: 231.1 on 6 and 1824 DF,  p-value: < 2.2e-16
```

```
library(car)
vif(mod_final)
```

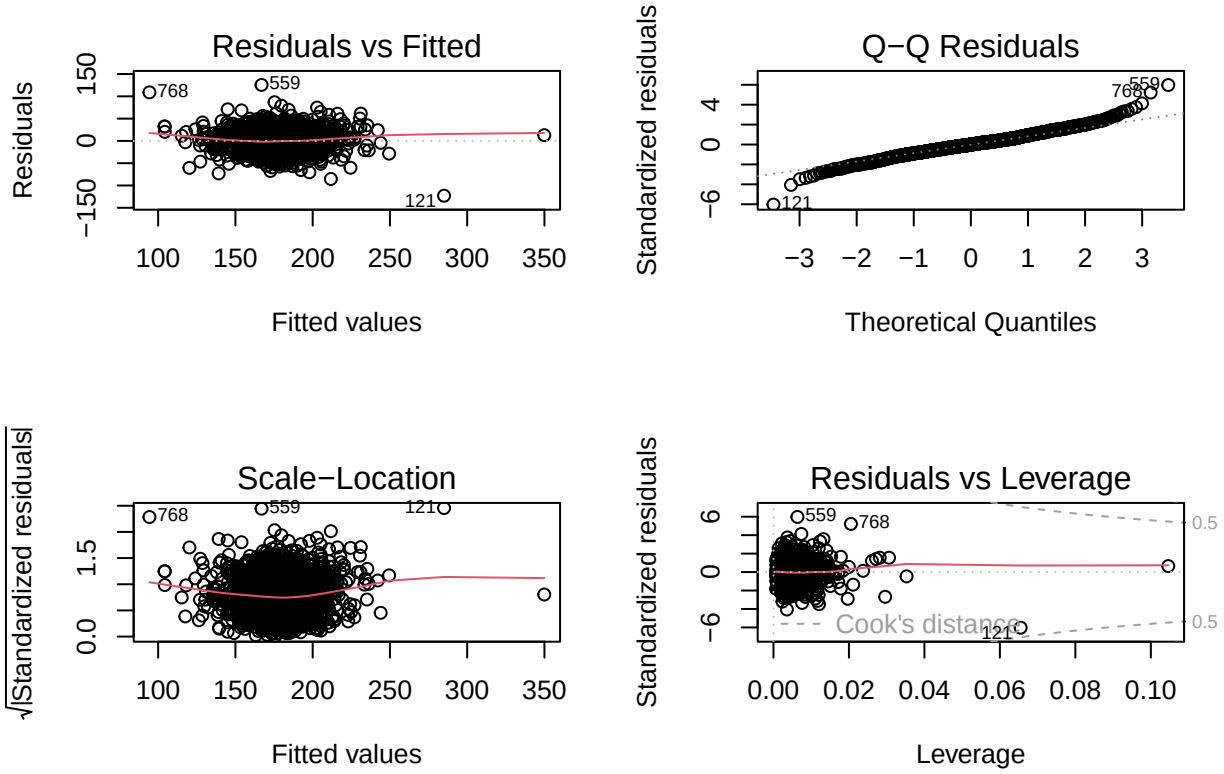
```
## pctpubliccoveragealone          incidencerate               pctblack
##               2.917999               1.039356               1.645454
##   pctmarriedhouseholds   pctunemployed16_over              medincome
##               1.840197               1.978995               2.217612
```

13. The validation of the model has to be done with graphs and / or suitable tests to verify model assumptions.

The graph of residuals vs. predicted values as well as the graph of location scale do not show the typical funnel or rhombus shape which would be a symptom of heteroscedasticity problems. On the contrary, it

is observed that the values are dispersed uniformly along the range of values of the predicted values. On the other hand, in the QQ graph, it is observed that most of the points are aligned to the diagonal line. Finally, the graph of Residuals vs. Leverage shows the possible presence of influential values in the model. In the case of atypical values (vertical axis), those residuals within the range of -3 to 3 are usually considered within normal values, while in the case of Leverage, values above 0.02 are considered. Therefore, values with high Residual and high Leverage values would be candidates for influential values. In this graph, counties 1639,654,106 are indicated as possible candidates for influential values.

```
par(mfrow=c(2,2))
plot(mod_final)
```




```
r<-rstandard(mod_final)
shapiro.test(r)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  r
## W = 0.98257, p-value = 3.833e-14
```

```
ks.test(r,"pnorm")
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
```

```
##
## data:  r
## D = 0.042769, p-value = 0.002466
## alternative hypothesis: two-sided
```

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
bptest(mod_final)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  mod_final
## BP = 10.602, df = 6, p-value = 0.1015
```

```
durbinWatsonTest(mod_final)[2]
```

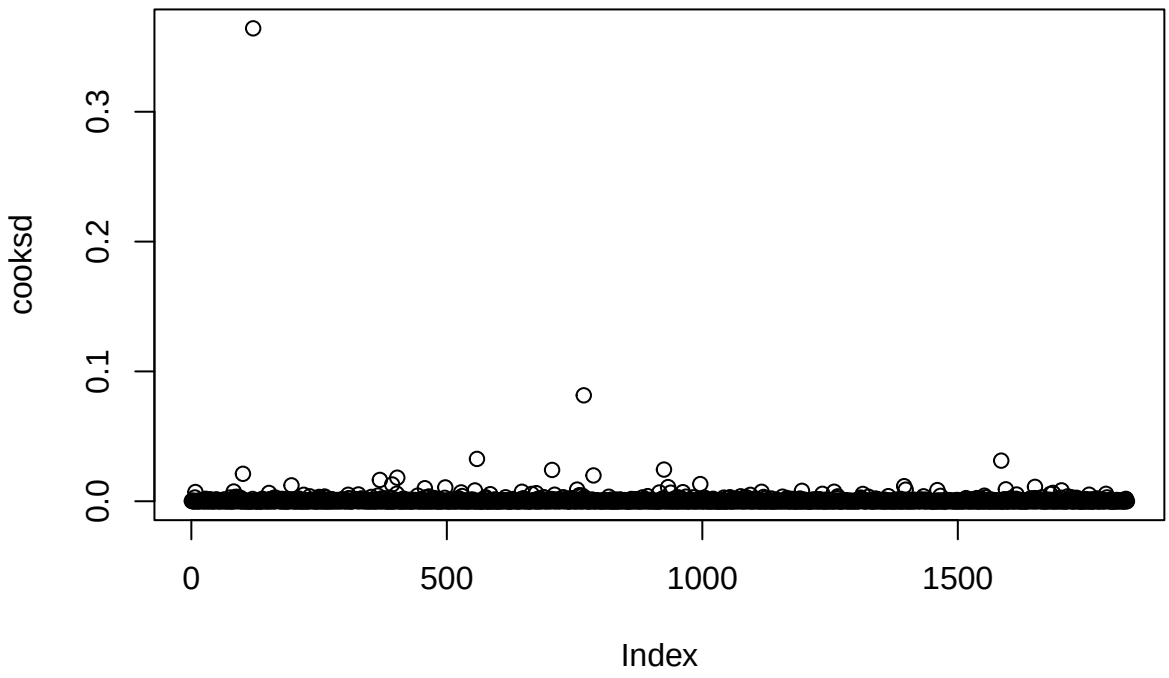
```
## $dw
## [1] 1.995591
```

When applying the Shapiro Wilks test to evaluate whether the residuals fit the Normal distribution, the test is not significant, that is, this assumption is not validated. However, when applying the Kolmogor-Smirnov test, this p-value is almost close to 1% of significance (p-value of 0.08%). On the other hand, when evaluating homoscedasticity, the Breuch Pagan test is not significant, that is, the model is homoscedastic. Regarding the autocorrelation of residuals, the reference value of Durbin-Watson is 1.5 to 2.5, which would indicate no correlation of residuals (DW=1.61).

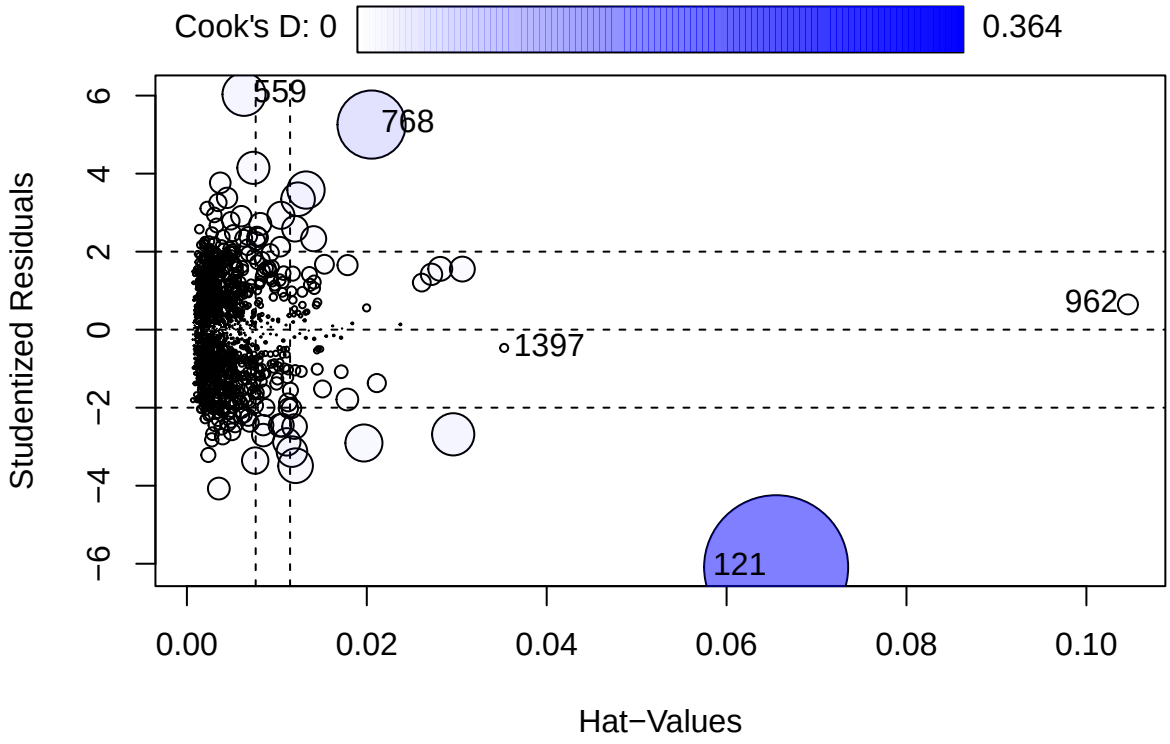
14. You must include the study of unusual and / or influential data.

The influential values analysis identifies the counties identified as 168,734,899,1615,1633 as having high values in some standardized residual, leverage or Cook distance metric. When these observations are removed from the model, it is observed that the homoscedasticity of the model is lost. Therefore, these cases should remain in the regression model.

```
cooksd <- cooks.distance(mod_final)
plot(cooksd)
```




```
influencePlot(mod_final, id=list(n=3, method="noteworthy"))
```

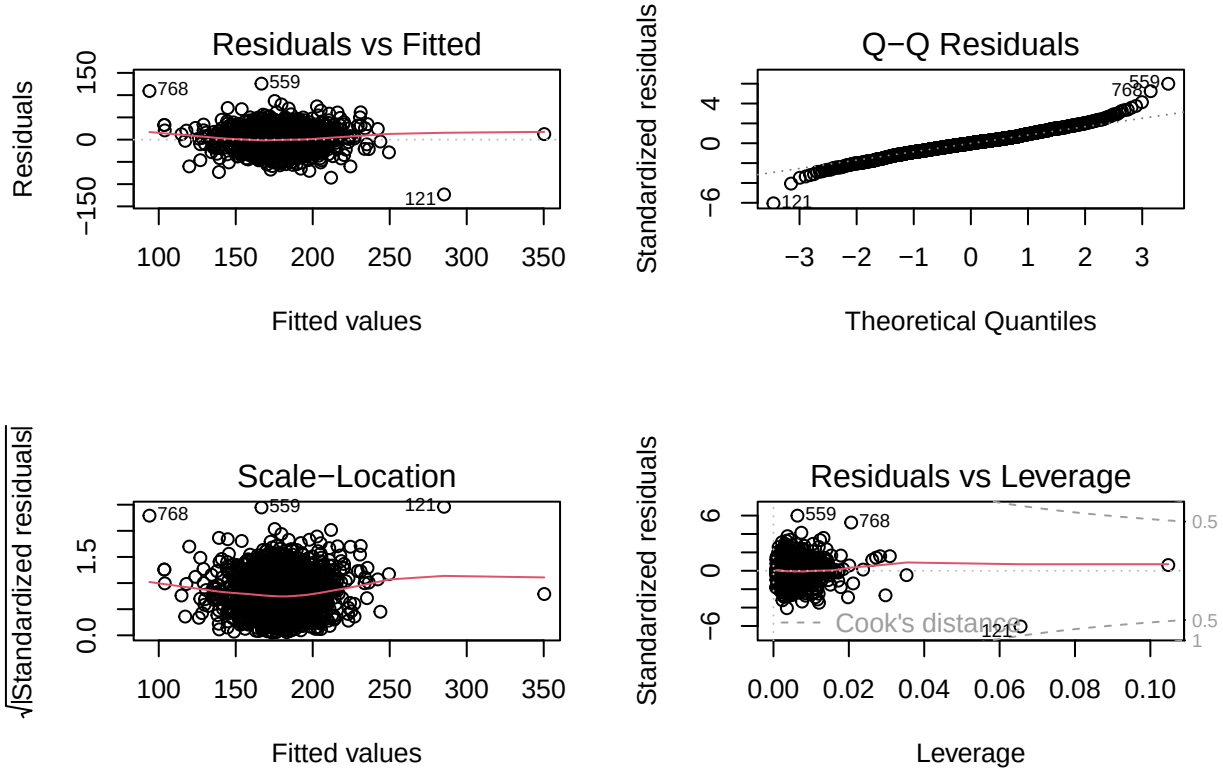



```
##          StudRes         Hat       CookD
## 121  -6.0903061 0.065507817 0.364240393
## 559   6.0321569 0.006349383 0.032583677
## 768   5.2575249 0.020528439 0.081570146
## 962   0.6467677 0.104621366 0.006984748
## 1397 -0.4675613 0.035269888 0.001142256
```

```
mod_final2<-lm(formula = target_deathrate ~ pctpubliccoveragealone + incidencerate + pctblack + pctmarri

vif(mod_final2)
```

```
## pctpubliccoveragealone          incidencerate              pctblack
##               2.916624               1.039461               1.645079
##   pctmarriedhouseholds   pctunemployed16_over             medincome
##               1.843902               1.979199               2.215495
```

```
par(mfrow=c(2,2))
plot(mod_final2)
```




```
r<-rstandard(mod_final2)
shapiro.test(r)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  r
## W = 0.98216, p-value = 2.576e-14
```

```
ks.test(r,"pnorm")
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  r
## D = 0.043619, p-value = 0.00192
## alternative hypothesis: two-sided
```

```
bptest(mod_final2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  mod_final2
## BP = 10.634, df = 6, p-value = 0.1004
```

```
durbinWatsonTest(mod_final2)[2]
```

```
## $dw
## [1] 1.999502
```

15. The resulting model should be interpreted in terms of the relationships of selected predictors and its effect on the response variable.

The interpretation of the coefficients is:

- pctpubliccoveragealone(0.8198): For each 1% increase in the percentage of county residents with government-provided health care, the mortality rate increases on average by 0.8198 per 100,000 inhabitants, holding all other variables constant.
- incidencerate (0.2128): When the average number of cancer diagnoses increases by one unit, the cancer mortality rate increases by 0.2128 per 100,000 inhabitants, holding all other variables constant.
- pctblack (0.1506): When the percentage of citizens who identify as white increases by 1%, the mortality rate increases by 0.1506 per 100,000 inhabitants, holding all other variables constant.
- pctmarriedhouseholds (0.2222): As the percentage of married households increases by 1%, the mortality rate increases by 0.2222 per 100,000 people, holding all other variables constant.
- pctemployed16_over (-0.4892): As the percentage of county residents 16 years of age or older who are employed increases by 1%, the mortality rate decreases by 0.4892 per 100,000 people, holding all other variables constant.
- medincome (-0.0005): As the county's median income increases by 1%, the mortality rate decreases by 0.0005 per 100,000 people, holding all other variables constant. These relationships are shown in the marginal effects graph. For each variable, the effect on the cancer mortality rate is quantified.

```
round(coef(mod_final),4)
```

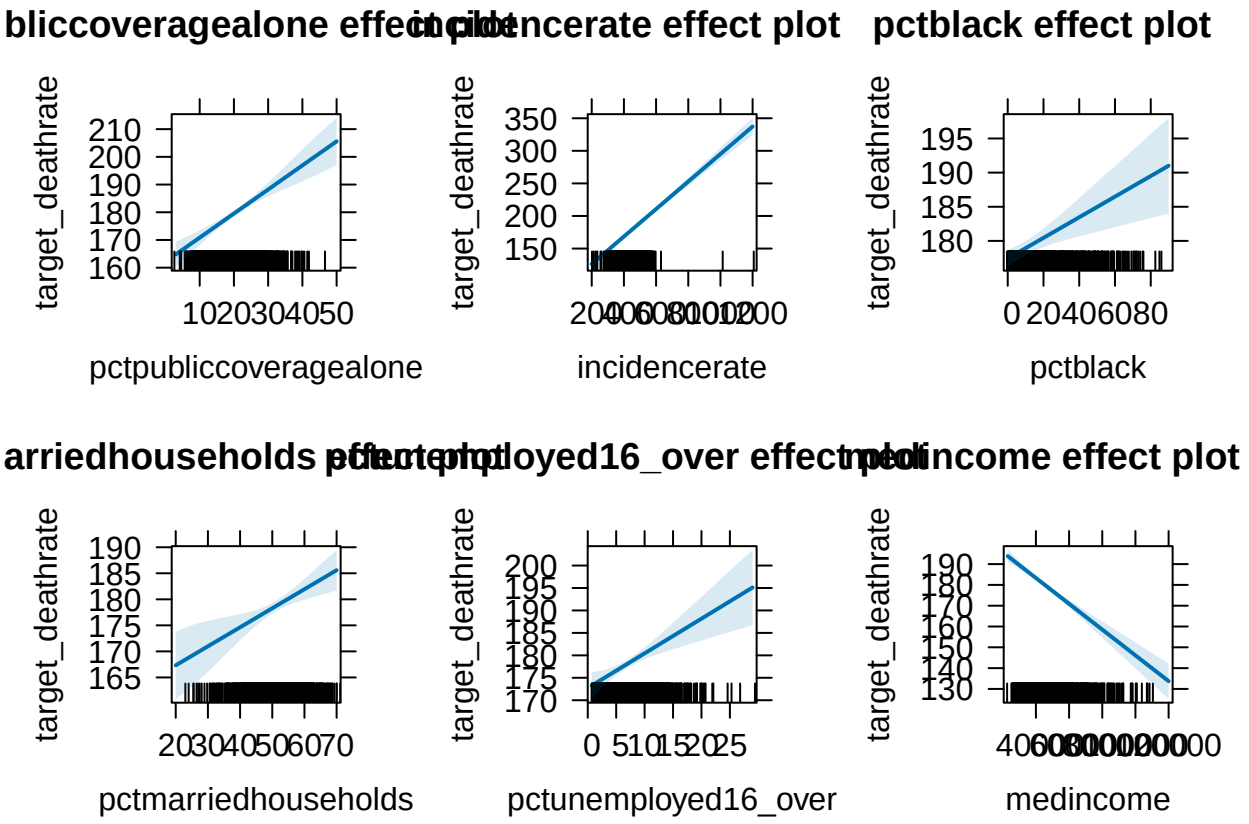
```
##            (Intercept) pctpubliccoveragealone          incidencerate
##                70.8251                 0.8788                 0.2105
##               pctblack   pctmarriedhouseholds   pctunemployed16_over
##                 0.1515                 0.3552                 0.7548
##              medincome
##                -0.0006
```

```
library(effects)
```

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

```
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```
plot(allEffects( mod_final2 ))
```



16. You have to apply your final model to the test sample and roughly assess forecasting capability.

The performance of the model was evaluated by comparing the predictions on the test data with the real values of the data set. A RMSE (Root Mean Squared Error) of 21.02 is obtained, which measures the mean squared error between the predictions and the observed values, in the same units as the target variable. The lower the RMSE, the better the model. This value indicates that, on average, the model predictions are 21.02 units away from the mortality rate. On the other hand, R2 indicates the proportion of the variability of the response variable that is explained by the model. In this case, the model explains 41.5% of the variability of the mortality rate in the test data. Finally, a MAE (Mean Absolute Error) of 15.99 is obtained, which reflects the mean absolute error between the predictions and the real values. The metrics indicate adequate performance towards the proposed data.

```
predicciones <- predict(mod_final, newdata = test_imputed)
 # Actual values of the dependent variable
 y_real <- test_imputed$target_deathrate
 # Calculation of metrics
 n <- length(y_real)
 rmse <- sqrt(sum((y_real - predicciones)^2) / n) # Root Mean Squared Error
 mae <- sum(abs(y_real - predicciones)) / n      # Mean Absolute Error
 sst <- sum((y_real - mean(y_real))^2)           # Total Sum of Squares
 ssr <- sum((y_real - predicciones)^2)           # Residual Sum of Squares
 r2 <-100*(1 - (ssr / sst))                           # R R squared
   # Return the metrics
 cbind(RMSE = rmse, MAE = mae, R2 = r2)
```

```
##          RMSE      MAE       R2
## [1,] 21.04401 16.06771 41.34925
```

17. Conclusions and contribution of each team member.

As we finish writing this report, it is our sincere thought that the entire group should be recognized in its development and transcription. However, we could appreciate the special attention of Jonàs Salat in the first treatment of the data and its imputation, and the support of Edwin Delgado and Nashly Gonzales in the subsequent analysis and model, with joint participation in the conclusions.

In conclusion, we can see that it is a linear model that is sufficient in the analysis and subsequent prediction of data offered by the proposed dataframe.

Appendix

Variables Analysis

For each variable in the dataset train, a descriptive analysis is conducted, a data quality assessment is generated, and imputation and profiling are documented.

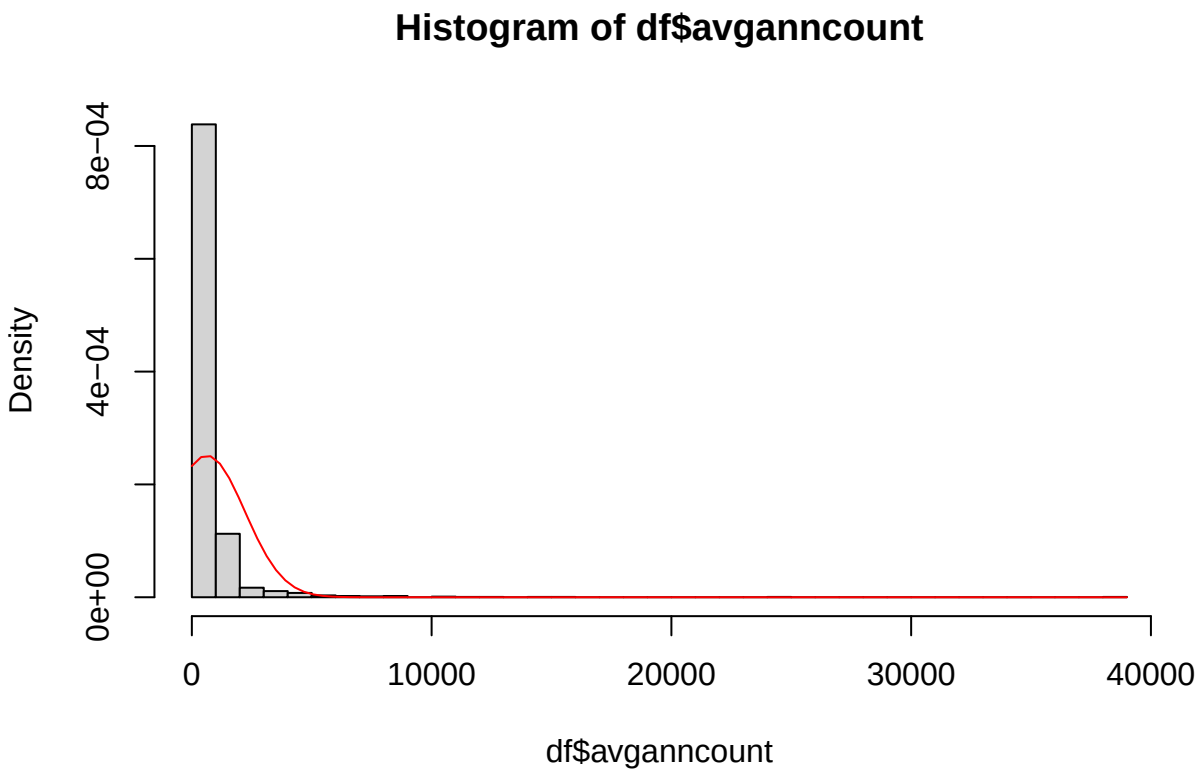
Variable 1: avganncount

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 273 outliers (out of which 252 severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
# df for the exploratory data analysis
df <- read.csv("train.csv", stringsAsFactors = FALSE)
summary(df$avganncount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     7.0    80.0   175.0   623.2   509.0 38150.0
```

```
hist(df$avganncount, breaks = 30, freq = F)
curve(dnorm(x, mean(df$avganncount), sd(df$avganncount)), add = T, col = "red")
```



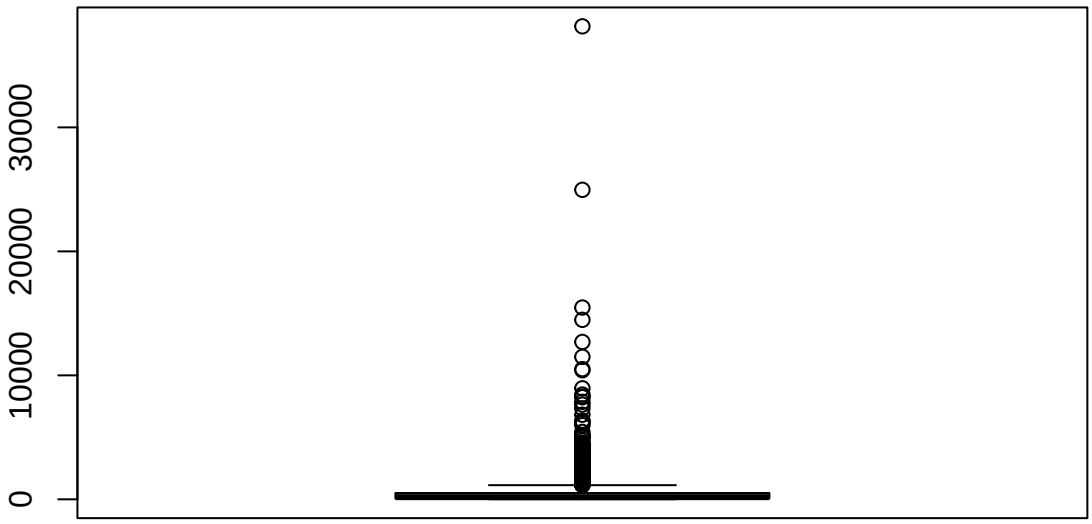

```
shapiro.test(df$avganncount)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$avganncount
## W = 0.33377, p-value < 2.2e-16
```

```
sum(is.na(df$avganncount))
```

```
## [1] 0
```

```
boxplot(df$avganncount)
```




```
bp<-boxplot(df$avganncount, id = list(n=Inf))
length(bp$out)
```

```
## [1] 273
```

```
sevout_avganncount = (quantile(df$avganncount,0.25)+(3*((quantile(df$avganncount,0.75) - quantile(df$av
length(which(df$avganncount > sevout_avganncount))
```

```
## [1] 252
```

```
df$f.avganncount <- ifelse(df$avganncount <= 80, 1,
                           ifelse(df$avganncount > 80 & df$avganncount <= 175, 2,
                           ifelse(df$avganncount > 175 & df$avganncount <= 509, 3,
                           ifelse(df$avganncount > 509, 4,0))))
df$f.avganncount <- factor(df$f.avganncount,
                           labels=c("LowAvganncount","LowMidAvganncount","HighMidAvganncount","HighAvga
                           order = T,
                           levels=c(1,2,3,4))
table(df$f.avganncount)
```

```
##
##     LowAvganncount  LowMidAvganncount HighMidAvganncount    HighAvganncount
##                460                 458                 455                 458
```

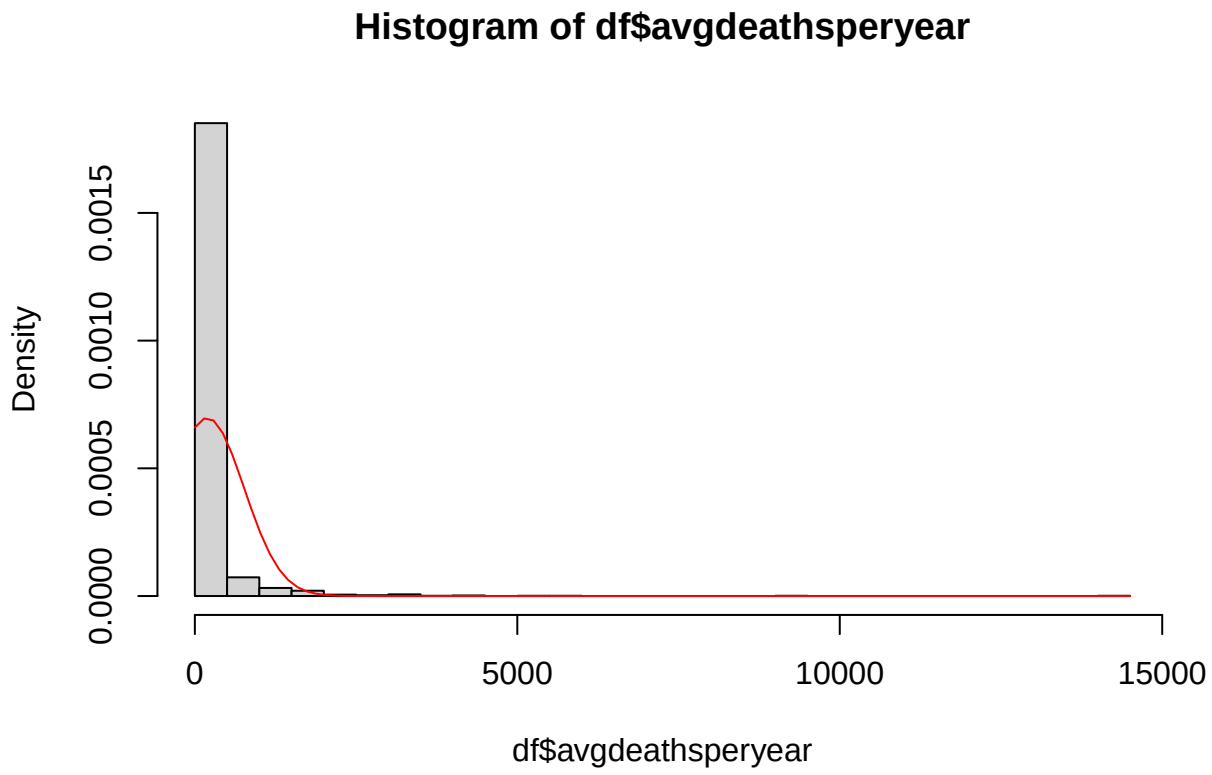
Variable 2: avgdeathsperyear

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 225 outliers (out of which 178 severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
summary(df$avgdeathsperyear)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     3.0    29.0    62.0   191.6   140.5 14010.0
```

```
hist(df$avgdeathsperyear, breaks = 30, freq = F)
curve(dnorm(x, mean(df$avgdeathsperyear), sd(df$avgdeathsperyear)), add = T, col = "red")
```



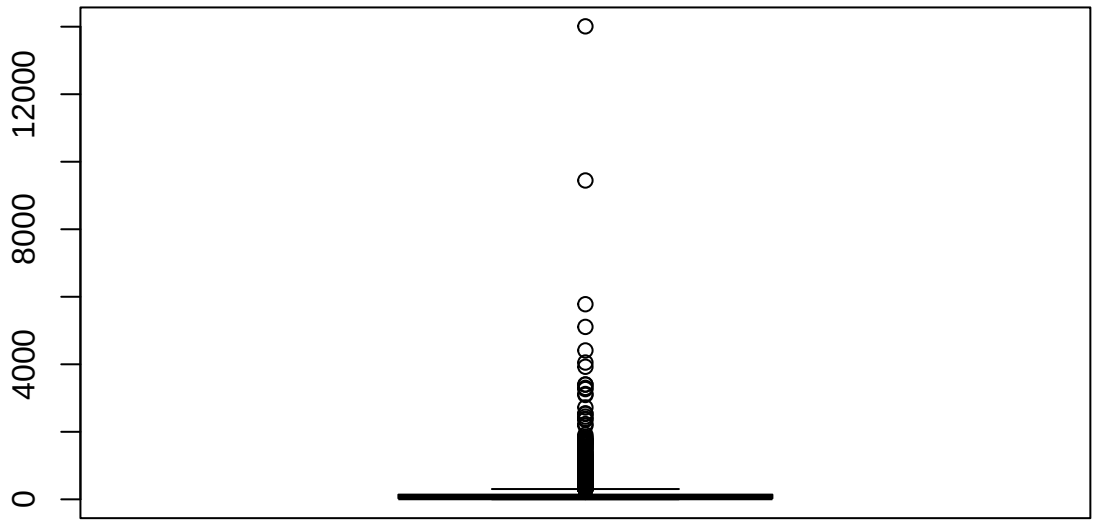

```
shapiro.test(df$avgdeathsperyear)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$avgdeathsperyear
## W = 0.26769, p-value < 2.2e-16
```

```r
sum(is.na(df$avgdeathsperyear))
```

```
## [1] 0
```

```r
boxplot(df$avgdeathsperyear)
```




```r
bp<-boxplot(df$avgdeathsperyear, id = list(n=Inf))
length(bp$out)
```

```
## [1] 225
```

```r
sevout_avgdeathsperyear = (quantile(df$avgdeathsperyear,0.25)+(3*((quantile(df$avgdeathsperyear,0.75) -
length(which(df$avgdeathsperyear > sevout_avgdeathsperyear))
```

```
## [1] 178
```

```r
df$f.avgdeathsperyear <- ifelse(df$avgdeathsperyear <= 29, 1,
                                ifelse(df$avgdeathsperyear > 29 & df$avgdeathsperyear <= 62, 2,
                                       ifelse(df$avgdeathsperyear > 62 & df$avgdeathsperyear <= 140.5, 3
                                              ifelse(df$avgdeathsperyear > 104.5, 4,0))))
df$f.avgdeathsperyear <- factor(df$f.avgdeathsperyear,
                                labels=c("LowAvgdeathsperyear","LowMidAvgdeathsperyear","HighMidAvgdeath
                                order = T,
```

```
                                levels=c(1,2,3,4))
table(df$f.avgdeathsperyear)
```

```
##
##     LowAvgdeathsperyear  LowMidAvgdeathsperyear HighMidAvgdeathsperyear
##                     462                     455                     456
##    HighAvgdeathsperyear
##                     458
```

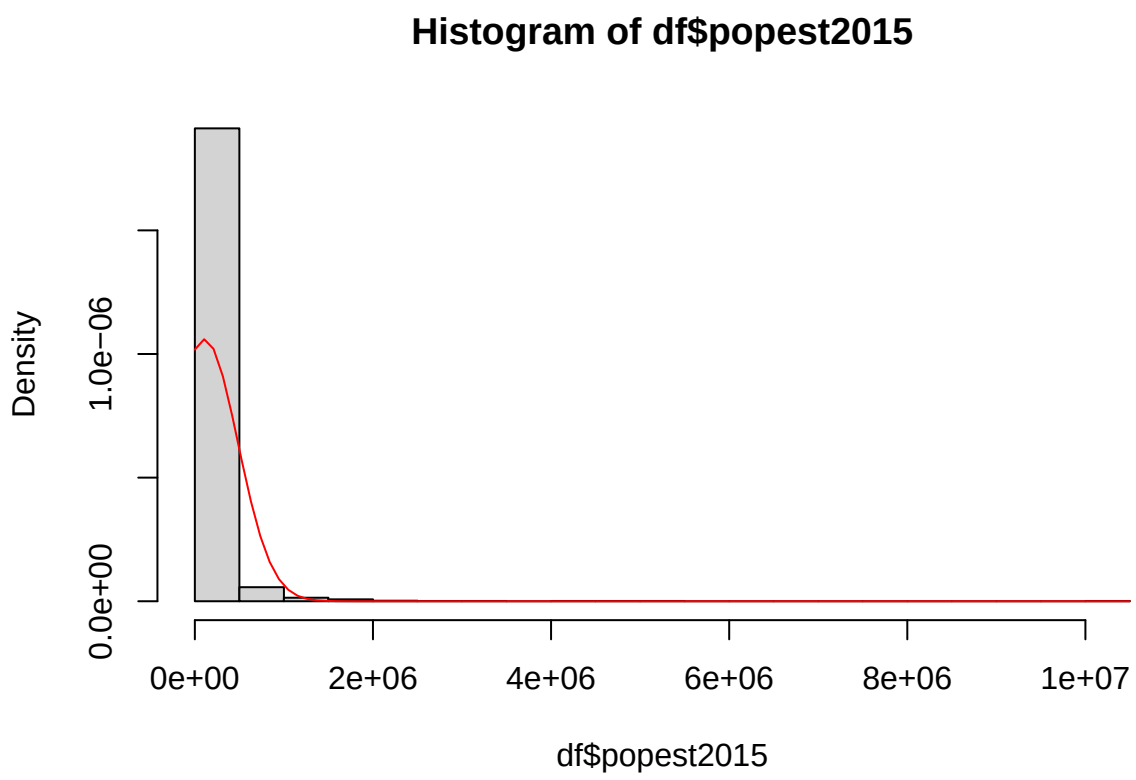
Variable 6: popest2015

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 252 outliers (out of which 210 severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
summary(df$popest2015)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##      829    12191    27158   106841    66880 10170292
```

```
hist(df$popest2015, breaks = 30, freq = F)
curve(dnorm(x, mean(df$popest2015), sd(df$popest2015)), add = T, col = "red")
```



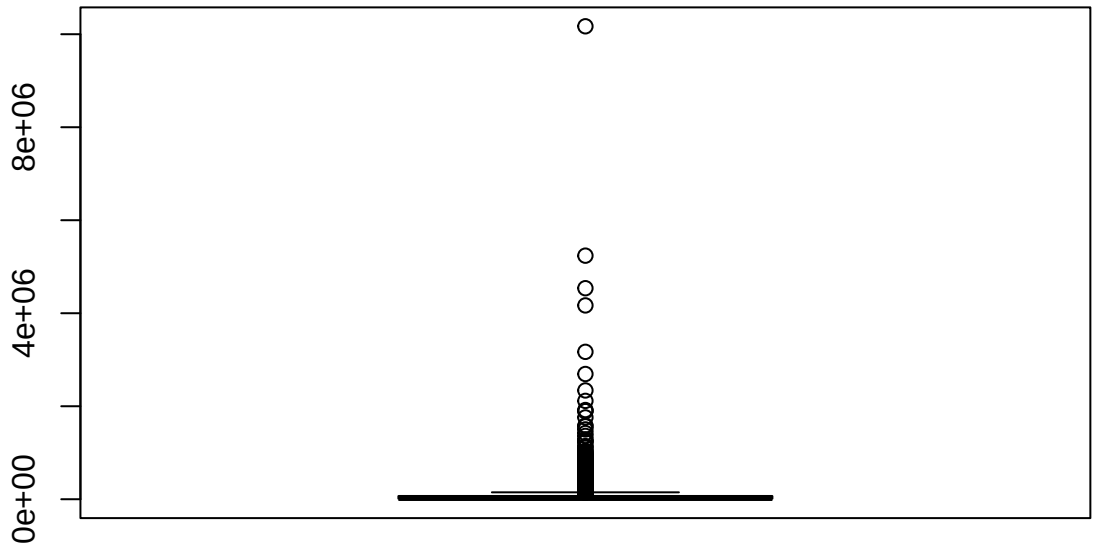
```
shapiro.test(df$popest2015)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$popest2015
## W = 0.22666, p-value < 2.2e-16
```

```
sum(is.na(df$popest2015))
```

```
## [1] 0
```

```
boxplot(df$popest2015)
```




```
bp<-boxplot(df$popest2015, id = list(n=Inf))
length(bp$out)
```

```
## [1] 252
```

```
sevout_popest2015 = (quantile(df$popest2015,0.25)+(3*((quantile(df$popest2015,0.75) - quantile(df$popes
length(which(df$popest2015 > sevout_popest2015))
```

```
## [1] 210
```

```
df$f.popest2015 <- ifelse(df$popest2015 <= 12191.0, 1,
                          ifelse(df$popest2015 > 12191.0 & df$popest2015 <= 27158.0, 2,
                                 ifelse(df$popest2015 > 27158.0 & df$popest2015 <= 66879.5, 3,
                                        ifelse(df$popest2015 > 66879.5, 4,0))))
df$f.popest2015 <- factor(df$f.popest2015,
                          labels=c("LowPopest2015","LowMidPopest2015","HighMidPopest2015","HighPopest2
                          order = T,
                          levels=c(1,2,3,4))
table(df$f.popest2015)
```

```
##
##     LowPopest2015  LowMidPopest2015 HighMidPopest2015    HighPopest2015
##               458               458               457               458
```

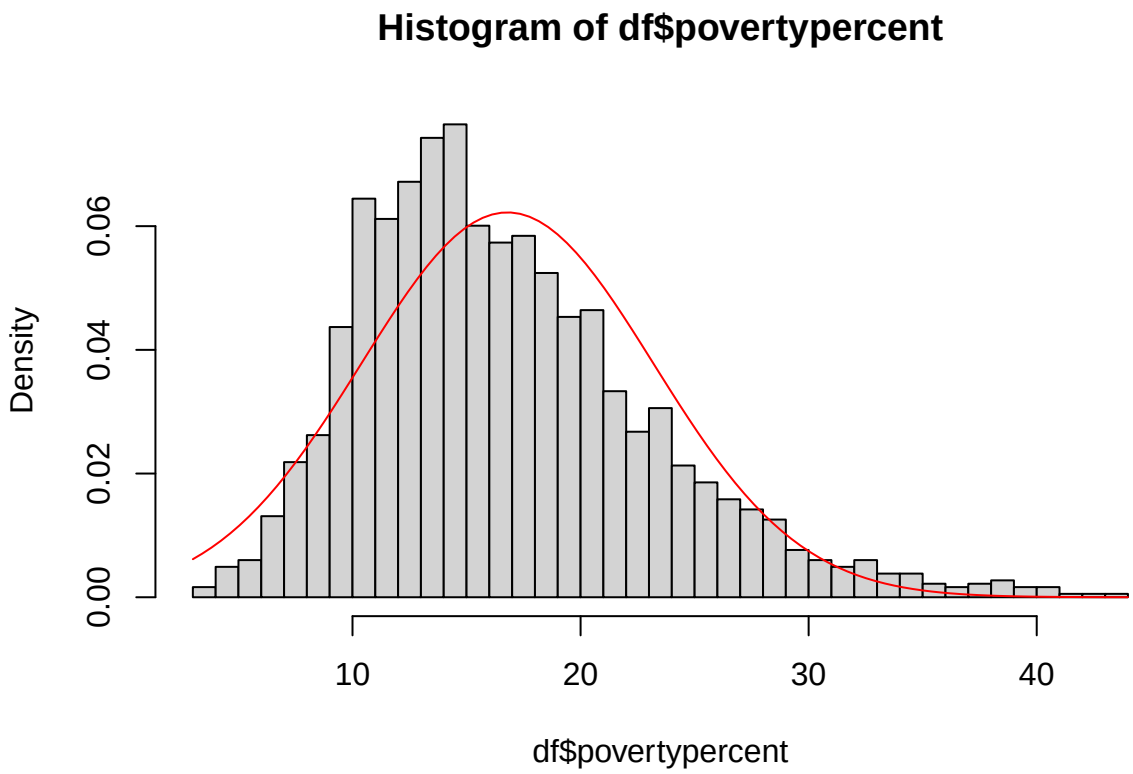
Variable 7: povertypercent

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 225 outliers (out of which 178 severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
summary(df$povertypercent)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3.70   12.15   15.70   16.79   20.40   44.00
```

```
hist(df$povertypercent, breaks = 30, freq = F)
curve(dnorm(x, mean(df$povertypercent), sd(df$povertypercent)), add = T, col = "red")
```



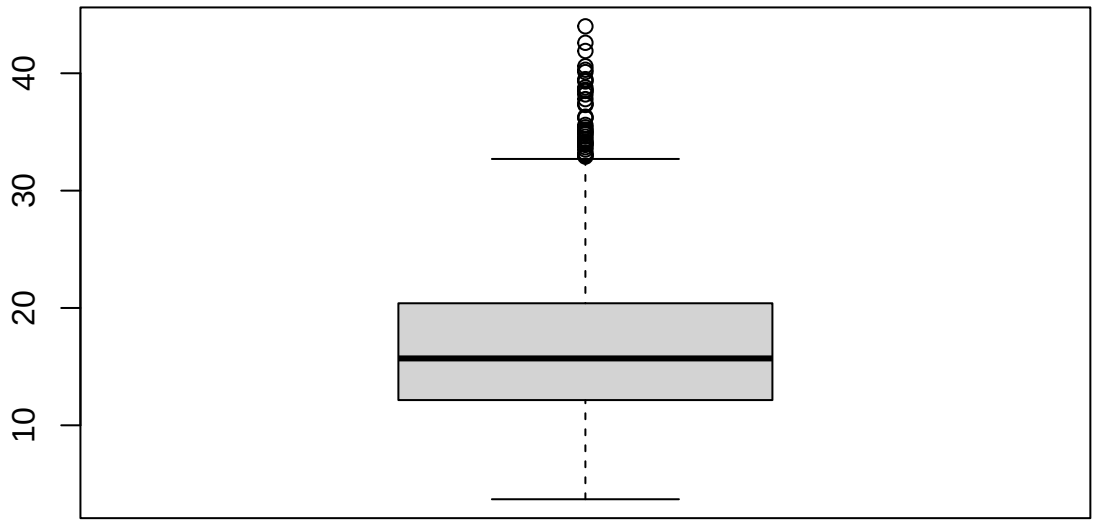

```
shapiro.test(df$povertypercent)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$povertypercent
## W = 0.95557, p-value < 2.2e-16
```

```
sum(is.na(df$povertypercent))
```

```
## [1] 0
```

```
boxplot(df$povertypercent)
```




```
bp<-boxplot(df$povertypercent, id = list(n=Inf))
length(bp$out)
```

```
## [1] 42
```

```
sevout_povertypercent = (quantile(df$povertypercent,0.25)+(3*((quantile(df$povertypercent,0.75) - quant
length(which(df$povertypercent > sevout_povertypercent))
```

```
## [1] 18
```

```
df$f.povertypercent <- ifelse(df$povertypercent <= 12.15, 1,
                              ifelse(df$povertypercent > 12.15 & df$povertypercent <= 15.70, 2,
                                     ifelse(df$povertypercent > 15.70 & df$povertypercent <= 20.40, 3,
                                            ifelse(df$povertypercent > 20.40, 4,0))))
df$f.povertypercent <- factor(df$f.povertypercent,
                              labels=c("LowPovertypercent","LowMidPovertypercent","HighMidPovertypercent",
                              order = T,
                              levels=c(1,2,3,4))
table(df$f.povertypercent)
```

```
##
##     LowPovertypercent  LowMidPovertypercent HighMidPovertypercent
##                   458                   468                   451
##    HighPovertypercent
##                   454
```

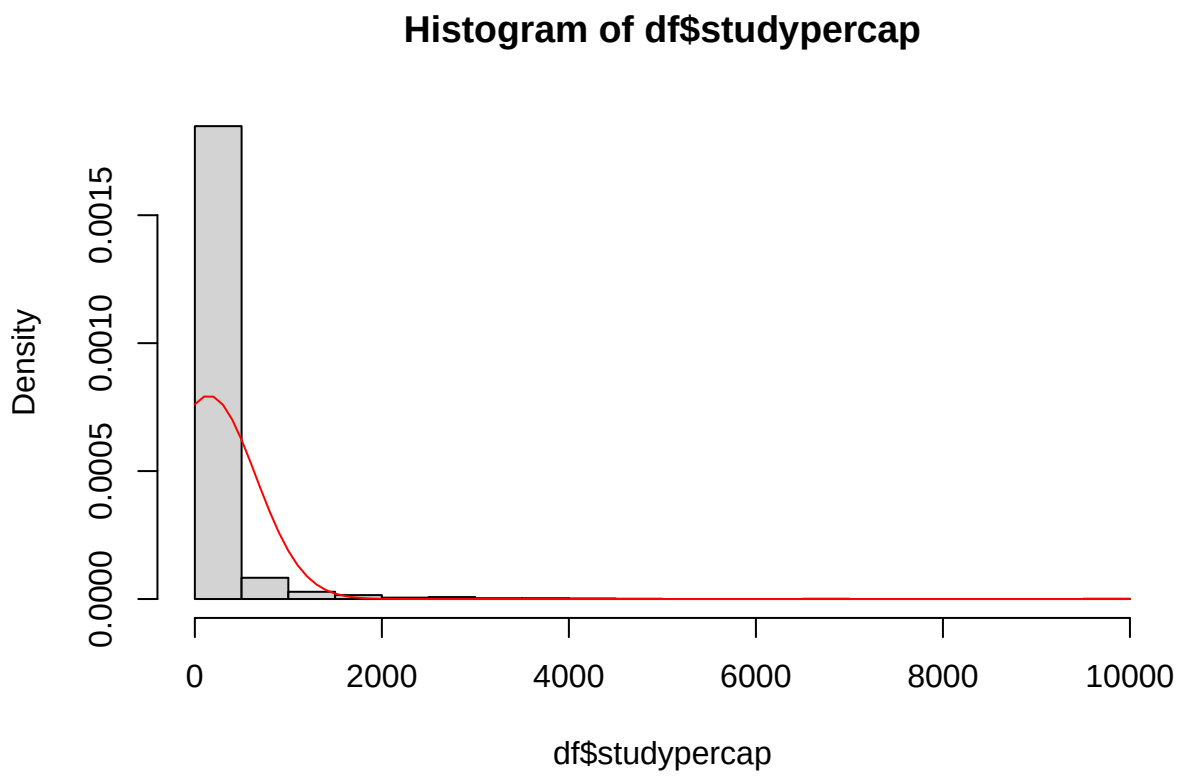
Variable 8: studypercap

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 307 outliers (out of which 281 severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
summary(df$studypercap)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0     0.0   148.2    76.0  9762.3
```

```
hist(df$studypercap, breaks = 30, freq = F)
curve(dnorm(x, mean(df$studypercap), sd(df$studypercap)), add = T, col = "red")
```



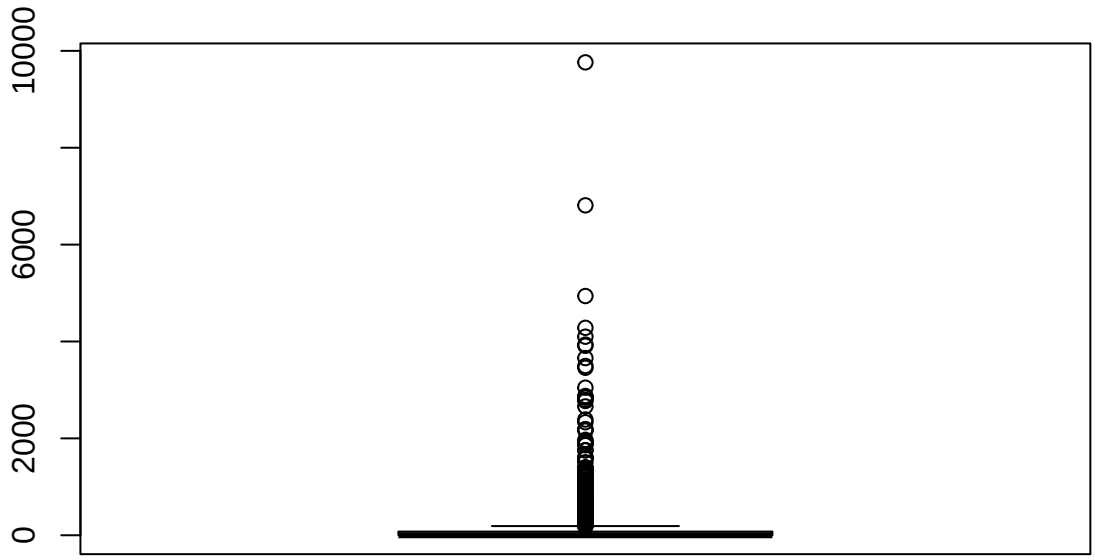

```
shapiro.test(df$studypercap)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$studypercap
## W = 0.30754, p-value < 2.2e-16
```

```r
sum(is.na(df$studypercap))
```

```
## [1] 0
```

```r
boxplot(df$studypercap)
```




```r
bp<-boxplot(df$studypercap, id = list(n=Inf))
length(bp$out)
```

```
## [1] 307
```

```r
sevout_studypercap = (quantile(df$studypercap,0.25)+(3*((quantile(df$studypercap,0.75) - quantile(df$st
length(which(df$studypercap > sevout_studypercap))
```

```
## [1] 281
```

```r
df$f.studypercap <- ifelse(df$studypercap <= 0.00000, 1,
                           ifelse(df$studypercap > 0.00000 & df$studypercap <= 0.00000, 2,
                                  ifelse(df$studypercap > 0.00000 & df$studypercap <= 76.00412, 3,
                                         ifelse(df$studypercap > 76.00412, 4,0))))
df$f.studypercap <- factor(df$f.studypercap,
                           labels=c("LowStudypercap","LowMidStudypercap","HighMidStudypercap","HighStud
                           order = T,
```

```
                            levels=c(1,2,3,4))
table(df$f.studypercap)
```

```
##
##     LowStudypercap  LowMidStudypercap HighMidStudypercap    HighStudypercap
##               1162                  0                211                458
```

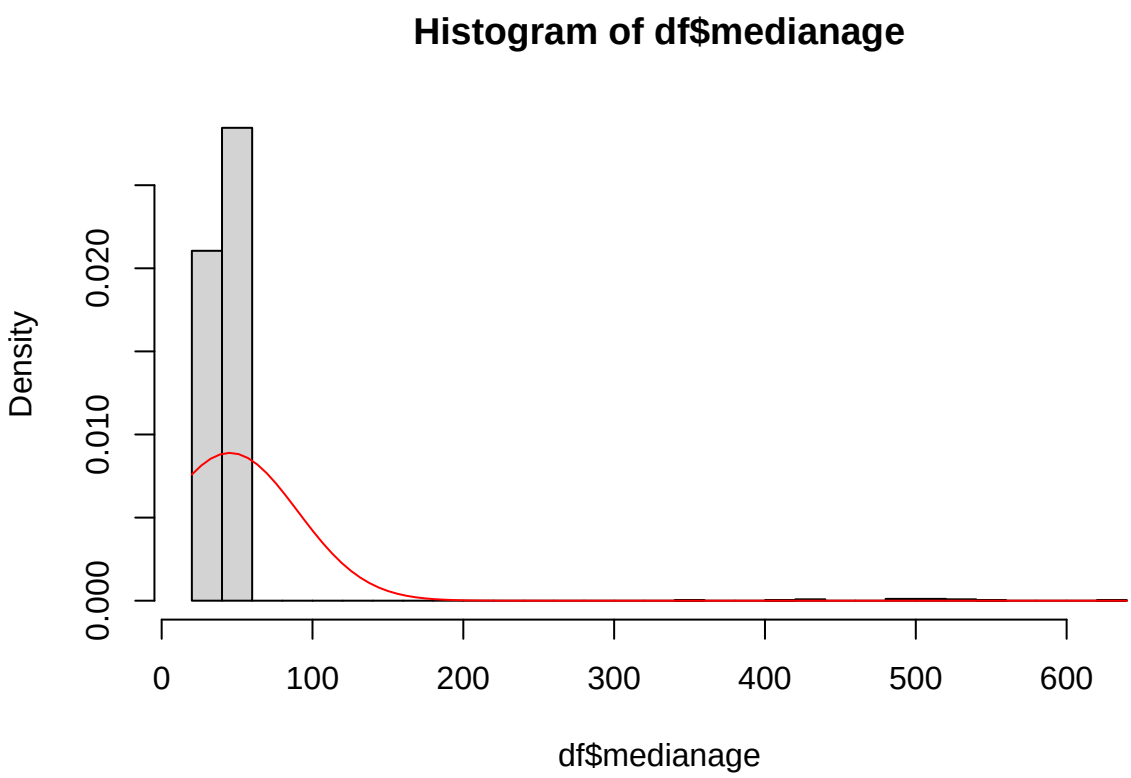
Variable 10: medianage

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 65 outliers (out of which 23 severe), all on both sides of the spectrum. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
summary(df$medianage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.30   37.90   40.90   45.25   44.00  624.00
```

```
hist(df$medianage, breaks = 30, freq = F)
curve(dnorm(x, mean(df$medianage), sd(df$medianage)), add = T, col = "red")
```



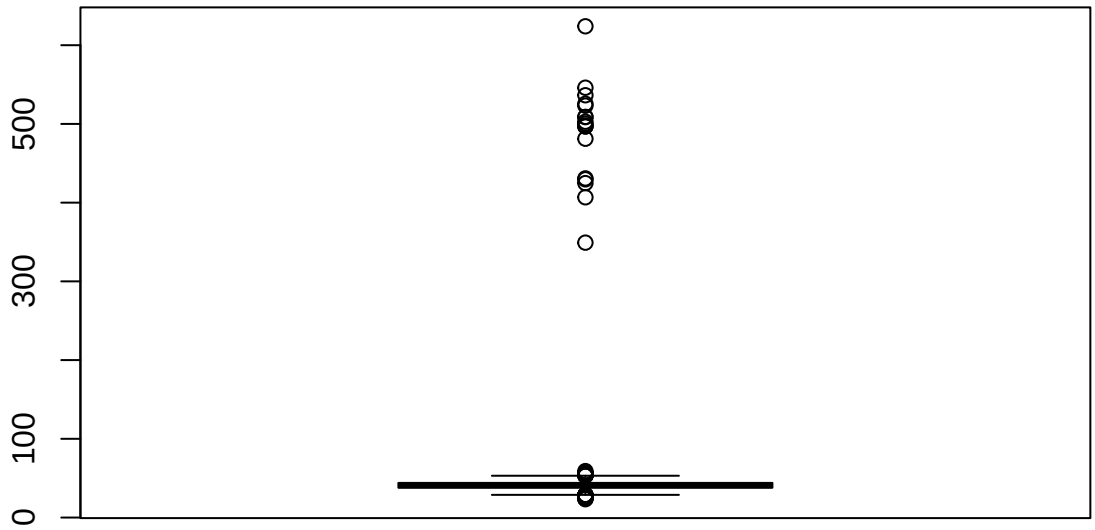
```
shapiro.test(df$medianage)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$medianage
## W = 0.14066, p-value < 2.2e-16
```

```
sum(is.na(df$medianage))
```

```
## [1] 0
```

```
boxplot(df$medianage)
```




```
bp<-boxplot(df$medianage, id = list(n=Inf))
length(bp$out)
```

```
## [1] 65
```

```
sevout_medianage = (quantile(df$medianage,0.25)+(3*((quantile(df$medianage,0.75) - quantile(df$medianag
length(which(df$medianage > sevout_medianage))
```

```
## [1] 23
```

```
df$f.medianage <- ifelse(df$medianage <= 37.9, 1,
                         ifelse(df$medianage > 37.9 & df$medianage <= 40.9, 2,
                                ifelse(df$medianage > 40.9 & df$medianage <= 44.0, 3,
                                       ifelse(df$medianage > 44.0, 4,0))))
df$f.medianage <- factor(df$f.medianage,
                         labels=c("LowMedianage","LowMidMedianage","HighMidMedianage","HighMedianage"
                         order = T,
                         levels=c(1,2,3,4))
table(df$f.medianage)
```

```
##
##     LowMedianage  LowMidMedianage HighMidMedianage    HighMedianage
##              465              453              457              456
```

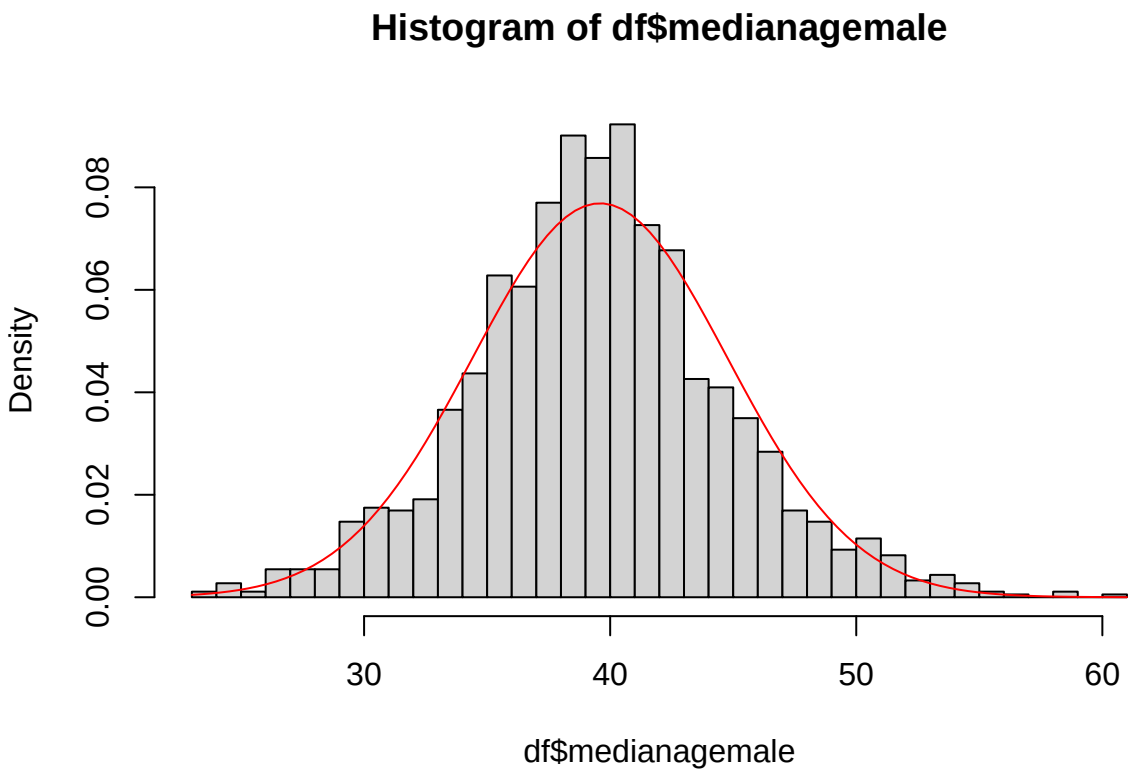
Variable 11: medianagemale

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 46 outliers (out of which 6 severe), all on both sides of the spectrum.. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
summary(df$medianagemale)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.00   36.40   39.50   39.59   42.60   60.20
```

```
hist(df$medianagemale, breaks = 30, freq = F)
curve(dnorm(x, mean(df$medianagemale), sd(df$medianagemale)), add = T, col = "red")
```



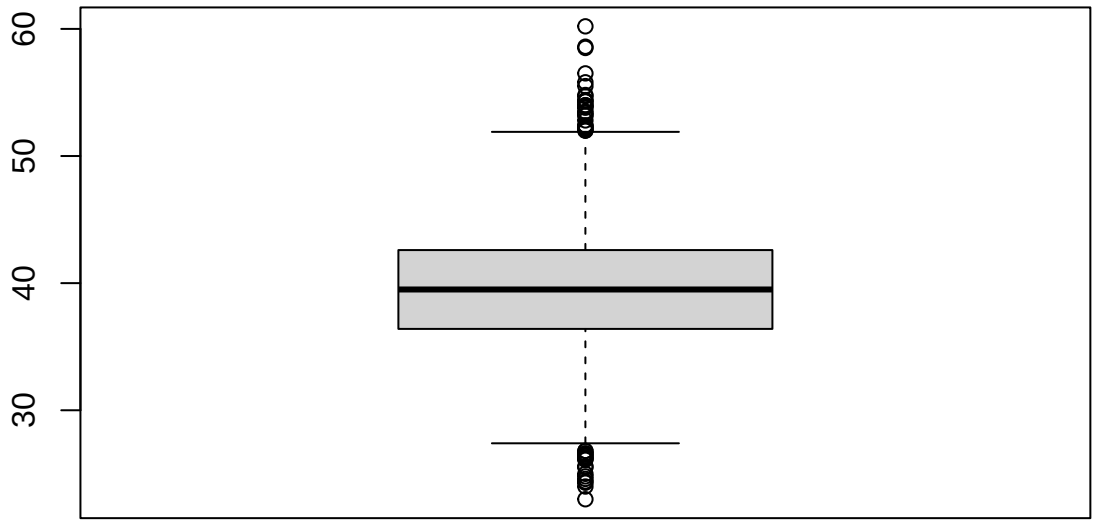

```
shapiro.test(df$medianagemale)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$medianagemale
## W = 0.99404, p-value = 9.877e-07
```

```
sum(is.na(df$medianagemale))
```

```
## [1] 0
```

```
boxplot(df$medianagemale)
```




```
bp<-boxplot(df$medianagemale, id = list(n=Inf))
length(bp$out)
```

```
## [1] 46
```

```
sevout_medianagemale = (quantile(df$medianagemale,0.25)+(3*((quantile(df$medianagemale,0.75) - quantile
length(which(df$medianagemale > sevout_medianagemale))
```

```
## [1] 6
```

```
df$f.medianagemale <- ifelse(df$medianagemale <= 36.4, 1,
                             ifelse(df$medianagemale > 36.4 & df$medianagemale <= 39.5, 2,
                                    ifelse(df$medianagemale > 39.5 & df$medianagemale <= 42.6, 3,
                                           ifelse(df$medianagemale > 42.6, 4,0))))
df$f.medianagemale <- factor(df$f.medianagemale,
                             labels=c("LowMedianagemale","LowMidMedianagemale","HighMidMedianagemale","Hig
                             order = T,
                             levels=c(1,2,3,4))
table(df$f.medianagemale)
```

```
##
##     LowMedianagemale  LowMidMedianagemale HighMidMedianagemale
##                  465                  471                  446
##    HighMedianagemale
##                  449
```

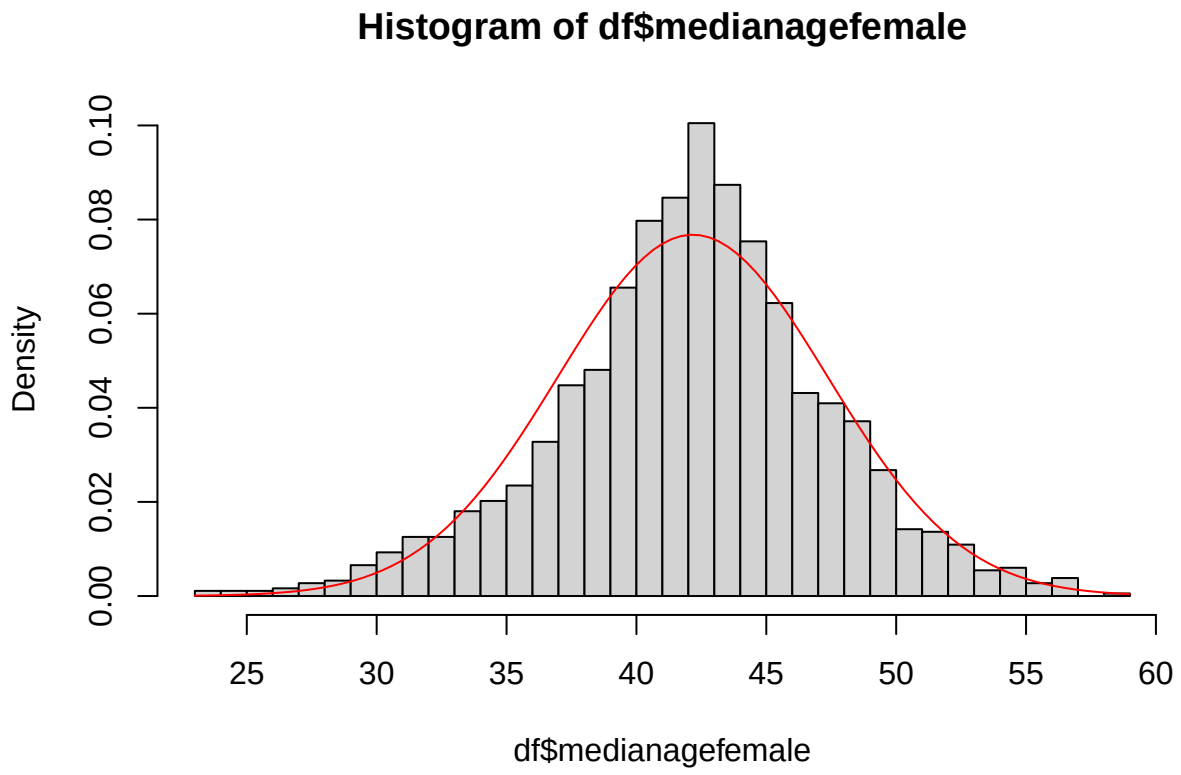
Variable 12: medianagefemale

This is a continuous ratio variable. The data does not look normally distributed, which is confirmed by the near-null p-value of the shapiro normallity test. A histogram is used to visualize the data. The variable contains no missing values thus imputation is not needed. It contains 55 outliers (out of which 1 severe), all on both sides of the spectrum. We create an additional ordinal mpg factor "f.mpg" to create a discretisation according to the quartiles.

```
summary(df$medianagefemale)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.60   39.20   42.40   42.17   45.30   58.20
```

```
hist(df$medianagefemale, breaks = 30, freq = F)
curve(dnorm(x, mean(df$medianagefemale), sd(df$medianagefemale)), add = T , col = "red")
```



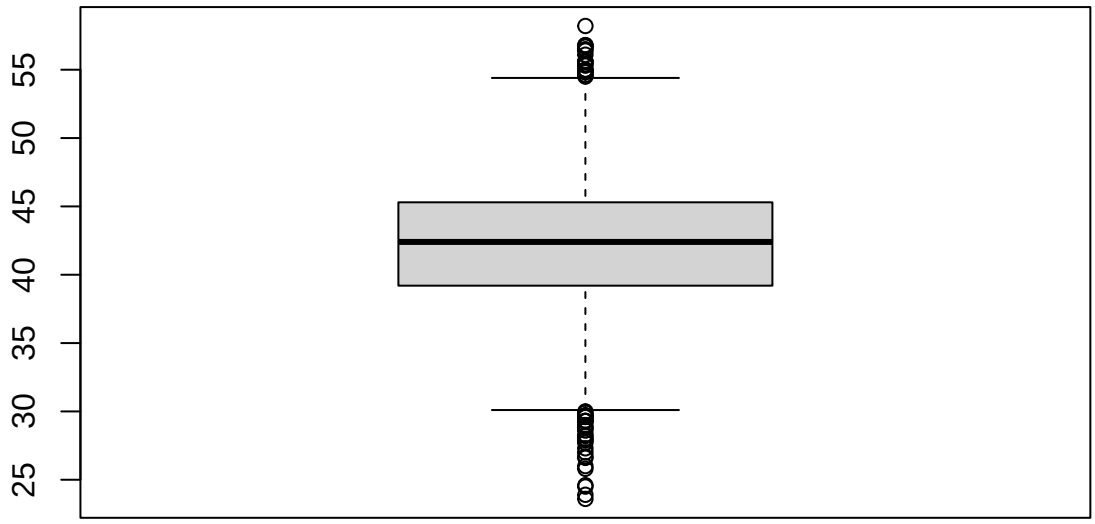

```
shapiro.test(df$medianagefemale)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$medianagefemale
## W = 0.99321, p-value = 1.817e-07
```

```r
sum(is.na(df$medianagefemale))
```

```
## [1] 0
```

```r
boxplot(df$medianagefemale)
```




```r
bp<-boxplot(df$medianagefemale, id = list(n=Inf))
length(bp$out)
```

```
## [1] 55
```

```r
sevout_medianagefemale = (quantile(df$medianagefemale,0.25)+(3*((quantile(df$medianagefemale,0.75) - qua
length(which(df$medianagefemale > sevout_medianagefemale))
```

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

```r
sevout_medianagefemale
```

```
##  25%
## 57.5
```

```
df$f.medianagefemale <- ifelse(df$medianagefemale <= 80, 1,
                           ifelse(df$medianagefemale > 80 & df$medianagefemale <= 175, 2,
                                  ifelse(df$medianagefemale > 175 & df$medianagefemale <= 509, 3,
                                         ifelse(df$medianagefemale > 509, 4,0))))
df$f.medianagefemale <- factor(df$f.medianagefemale,
                           labels=c("LowMedianagefemale","LowMidMedianagefemale","HighMidMedianagefemal
                           order = T,
                           levels=c(1,2,3,4))
table(df$f.medianagefemale)
```

```
##
##     LowMedianagefemale  LowMidMedianagefemale HighMidMedianagefemale
##                   1831                     0                      0
##    HighMedianagefemale
##                      0
```

Variable 13: geography

This is a categorical nominal variable. A bar graph is used to visualize the data. The areas with the largest number of counties are: Georgia, Arkansas, Florida, and Alabama. The variable does not contain missing values, so imputation is not necessary.

```
summary(as.factor(df$geography))
```

```
##           Acadia Parish, Louisiana                 Ada County, Idaho
##                                  1                                 1
##             Adair County, Kentucky             Adair County, Missouri
##                                  1                                 1
##                Adams County, Idaho             Adams County, Illinois
##                                  1                                 1
##                 Adams County, Iowa          Adams County, Mississippi
##                                  1                                 1
##             Adams County, Nebraska         Adams County, North Dakota
##                                  1                                 1
##                 Adams County, Ohio       Aiken County, South Carolina
##                                  1                                 1
##            Alachua County, Florida             Albany County, Wyoming
##                                  1                                 1
##         Albemarle County, Virginia            Alcona County, Michigan
##                                  1                                 1
##         Alcorn County, Mississippi Aleutians West Census Area, Alaska
##                                  1                                 1
##         Alexander County, Illinois           Alfalfa County, Oklahoma
##                                  1                                 1
##             Alger County, Michigan           Allegan County, Michigan
##                                  1                                 1
##          Allegany County, New York         Alleghany County, Virginia
##                                  1                                 1
##               Allen County, Kansas            Allen Parish, Louisiana
##                                  1                                 1
##            Alpena County, Michigan          Amador County, California
##                                  1                                 1
```

```
##           Amelia County, Virginia          Amite County, Mississippi
##                                 1                                  1
##     Anchorage Municipality, Alaska         Anderson County, Kentucky
##                                 1                                  1
##    Anderson County, South Carolina           Andrew County, Missouri
##                                 1                                  1
##         Androscoggin County, Maine      Anne Arundel County, Maryland
##                                 1                                  1
##           Anoka County, Minnesota       Anson County, North Carolina
##                                 1                                  1
##          Antelope County, Nebraska           Antrim County, Michigan
##                                 1                                  1
##            Appanoose County, Iowa           Appling County, Georgia
##                                 1                                  1
##          Arapahoe County, Colorado              Archer County, Texas
##                                 1                                  1
##         Archuleta County, Colorado           Arenac County, Michigan
##                                 1                                  1
##         Arlington County, Virginia     Armstrong County, Pennsylvania
##                                 1                                  1
##           Armstrong County, Texas           Aroostook County, Maine
##                                 1                                  1
##        Ascension Parish, Louisiana        Ashe County, North Carolina
##                                 1                                  1
##              Ashland County, Ohio            Ashtabula County, Ohio
##                                 1                                  1
##       Assumption Parish, Louisiana           Atchison County, Kansas
##                                 1                                  1
##          Atchison County, Missouri              Athens County, Ohio
##                                 1                                  1
##        Atlantic County, New Jersey            Atoka County, Oklahoma
##                                 1                                  1
##         Attala County, Mississippi             Audubon County, Iowa
##                                 1                                  1
##           Augusta County, Virginia        Aurora County, South Dakota
##                                 1                                  1
##              Austin County, Texas        Avoyelles Parish, Louisiana
##                                 1                                  1
##             Baca County, Colorado            Bacon County, Georgia
##                                 1                                  1
##             Baker County, Florida            Baker County, Georgia
##                                 1                                  1
##              Baker County, Oregon         Baltimore city, Maryland
##                                 1                                  1
##         Baltimore County, Maryland     Bamberg County, South Carolina
##                                 1                                  1
##             Banks County, Georgia            Bannock County, Idaho
##                                 1                                  1
##           Barbour County, Alabama      Barbour County, West Virginia
##                                 1                                  1
##        Barnes County, North Dakota           Barren County, Kentucky
##                                 1                                  1
##           Barron County, Wisconsin            Barrow County, Georgia
##                                 1                                  1
```

```
##             Barry County, Michigan            Barton County, Missouri
##                                  1                                  1
##              Bastrop County, Texas             Bates County, Missouri
##                                  1                                  1
##              Bath County, Virginia         Bayfield County, Wisconsin
##                                  1                                  1
##        Beadle County, South Dakota            Bear Lake County, Idaho
##                                  1                                  1
##    Beaufort County, North Carolina    Beaufort County, South Carolina
##                                  1                                  1
##            Beaver County, Oklahoma               Beaver County, Utah
##                                  1                                  1
##         Beaverhead County, Montana          Bedford County, Tennessee
##                                  1                                  1
##                 Bee County, Texas      Belknap County, New Hampshire
##                                  1                                  1
##                Bell County, Texas                           (Other)
##                                  1                               1732
```

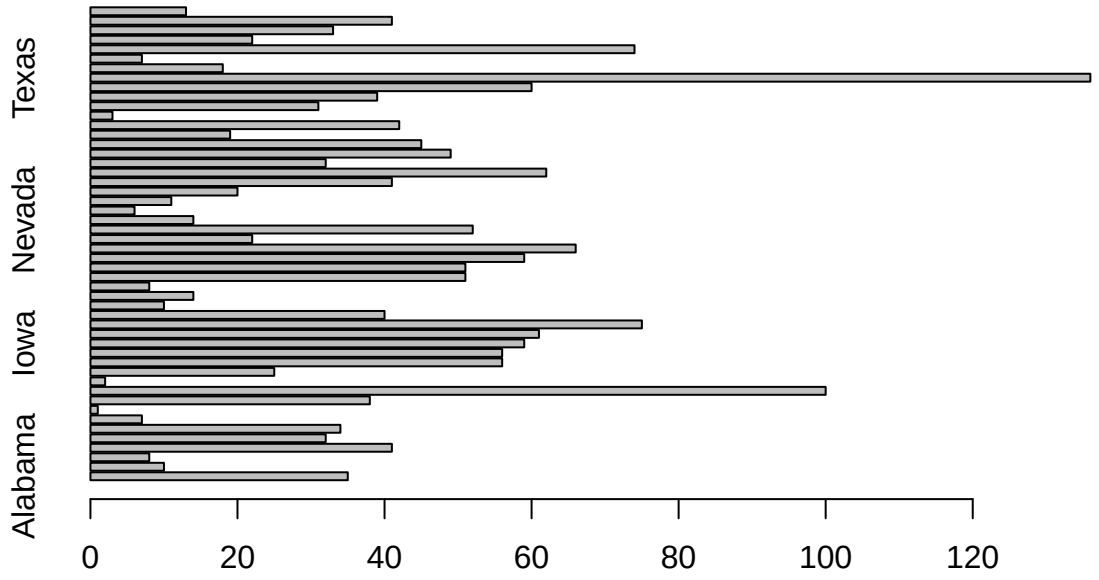
```
# Extract the state (text after the last comma) from geography
zona <- sub(".*,\\s*", "", df$geography)
table(zona)
```

```
## zona
##        Alabama         Alaska        Arizona       Arkansas     California
##             35             10              8             41             32
##       Colorado    Connecticut       Delaware        Florida        Georgia
##             34              7              1             38            100
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              2             25             56             56             59
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##             61             75             40             10             14
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              8             51             51             59             66
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##             22             52             14              6             11
##     New Mexico       New York North Carolina   North Dakota           Ohio
##             20             41             62             32             49
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##             45             19             42              3             31
##   South Dakota      Tennessee          Texas           Utah        Vermont
##             39             60            136             18              7
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##             74             22             33             41             13
```

```
zona= factor(zona)
sort(table(zona)[1:10],TRUE)
```

```
## zona
##     Georgia    Arkansas     Florida     Alabama    Colorado  California
##         100          41          38          35          34          32
##      Alaska     Arizona Connecticut    Delaware
##          10           8           7           1
```

```r
barplot(table(zona),horiz=TRUE)
```




```r
sum(is.na(zona))
```

```
## [1] 0
```

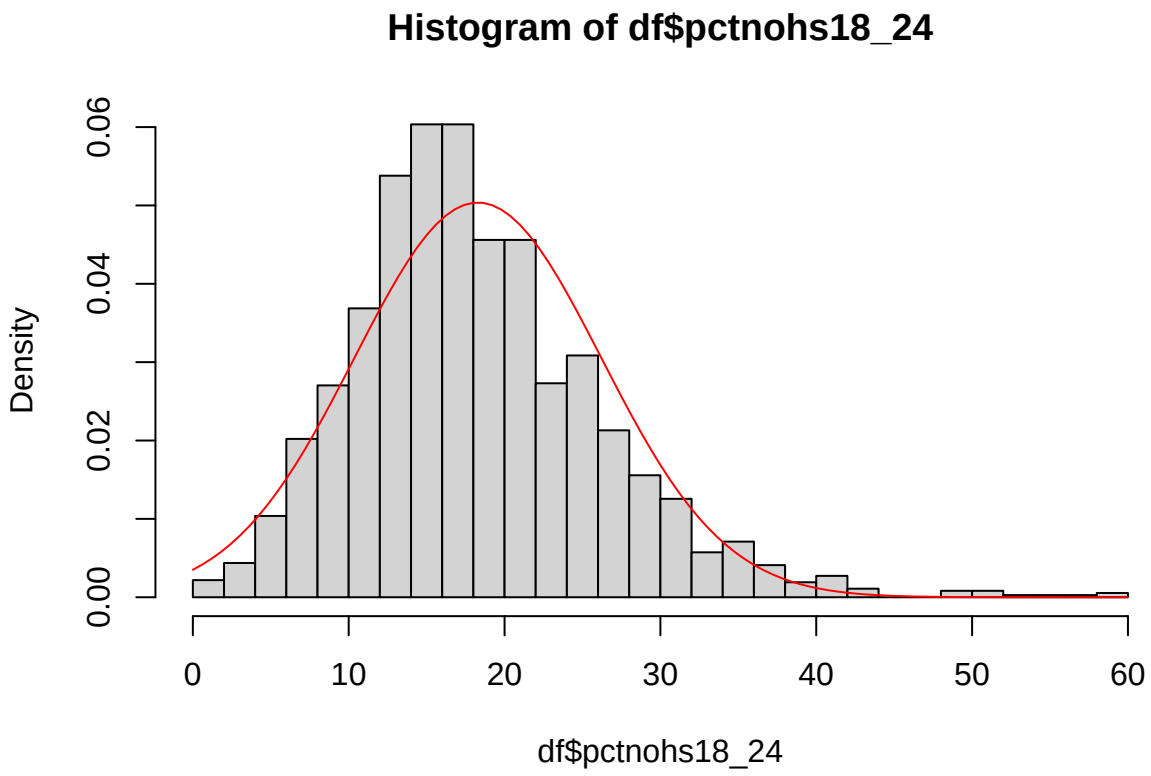
Variable 15: pctnohs18_24

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 34 outliers (of which 13 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctnohs18_24" to create a discretization according to quartiles.

```r
summary(df$pctnohs18_24)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.50   12.90   17.20   18.29   22.70   59.10
```

```r
hist(df$pctnohs18_24, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctnohs18_24), sd(df$pctnohs18_24)), add = T , col = "red")
```



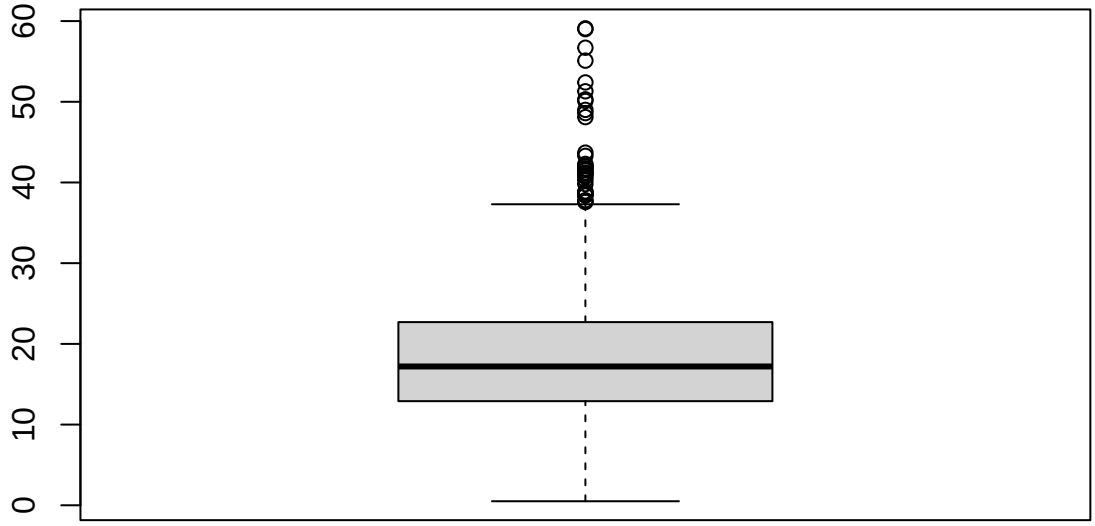

```
shapiro.test(df$pctnohs18_24)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctnohs18_24
## W = 0.96205, p-value < 2.2e-16
```

```
sum(is.na(df$pctnohs18_24))
```

```
## [1] 0
```

```
boxplot(df$pctnohs18_24)
```




```
bp<-boxplot(df$pctnohs18_24, id = list(n=Inf))
length(bp$out)
```

```
## [1] 35
```

```
sevout_pctnohs18_24 = (quantile(df$pctnohs18_24,0.25)+(3*((quantile(df$pctnohs18_24,0.75) - quantile(df$
length(which(df$pctnohs18_24 > sevout_pctnohs18_24))
```

```
## [1] 13
```

```
sevout_pctnohs18_24
```

```
##  25%
## 42.3
```

```
df$f.pctnohs18_24 <- ifelse(df$pctnohs18_24 <= 12.9, 1,
                     ifelse(df$pctnohs18_24 > 12.9 & df$pctnohs18_24 <= 17.2, 2,
                     ifelse(df$pctnohs18_24 > 17.2 & df$pctnohs18_24 <= 22.7, 3,
                        ifelse(df$pctnohs18_24 > 22.7, 4,0))))
df$f.pctnohs18_24 <- factor(df$f.pctnohs18_24 ,
       labels=c("Lowpctnohs18_24","LowMidpctnohs18_24",
                "HighMidpctnohs18_24","Highpctnohs18_24"),
               order = T, levels=c(1,2,3,4))
table(df$f.pctnohs18_24)
```

```
##
##     Lowpctnohs18_24  LowMidpctnohs18_24 HighMidpctnohs18_24    Highpctnohs18_24
##                 459                 461                 455                 456
```

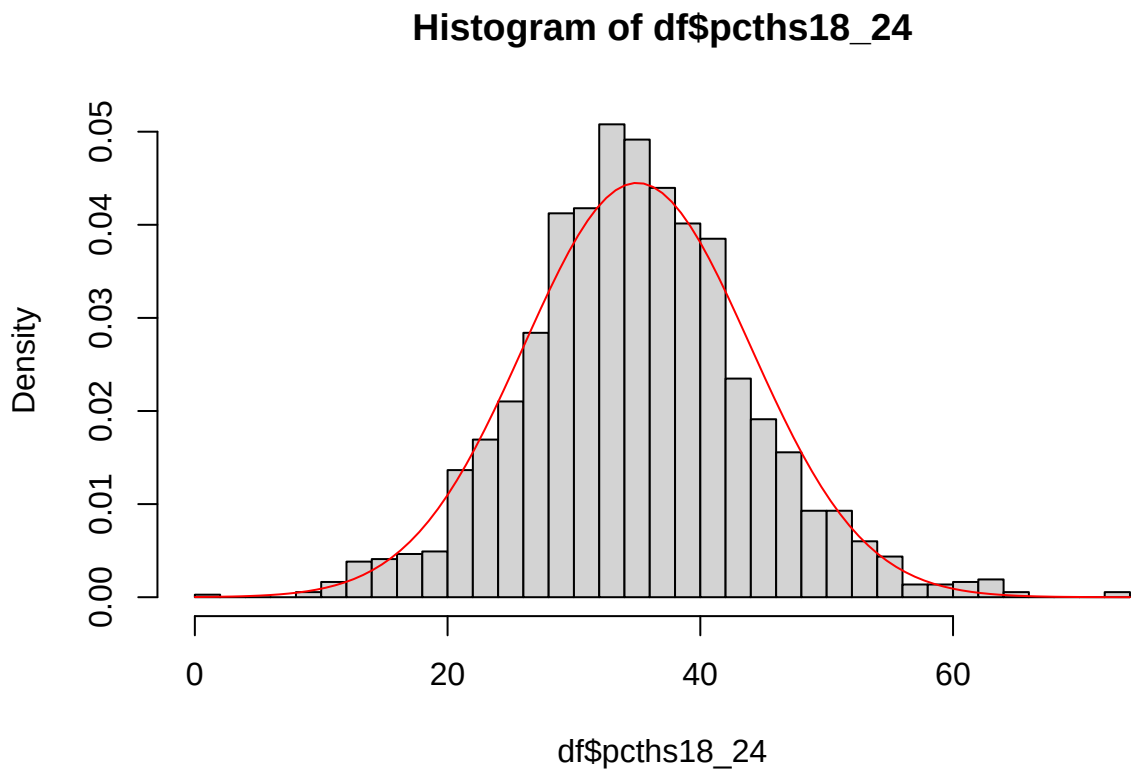
Variable 16:pcths18_24

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 36 outliers (of which 9 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pcths18_24" to create a discretization according to quartiles.

```
summary(df$pcths18_24)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0    29.2    34.7    35.0    40.5    72.5
```

```
hist(df$pcths18_24, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pcths18_24), sd(df$pcths18_24)), add = T, col = "red")
```




```
shapiro.test(df$pcths18_24)
```

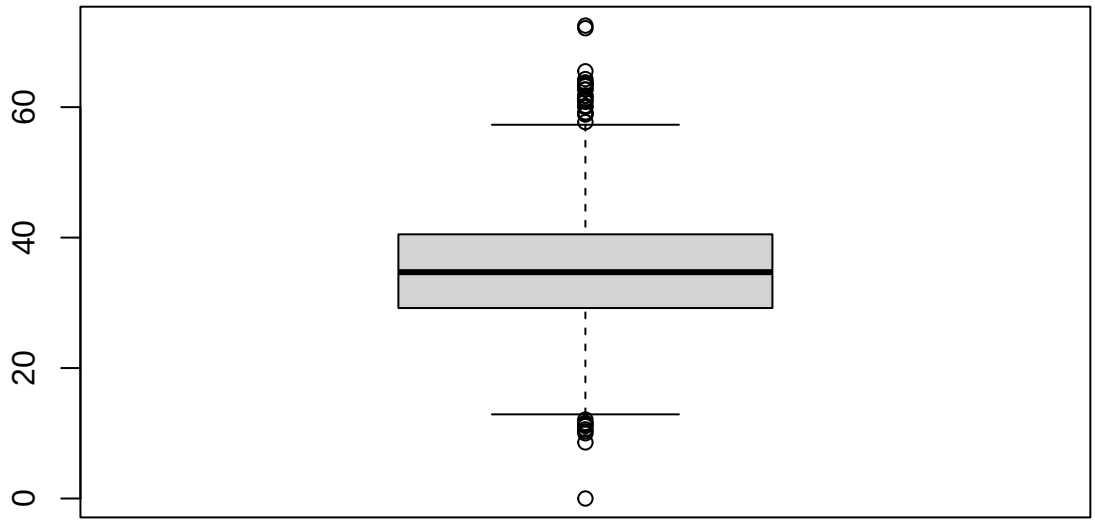
```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  df$pcths18_24
## W = 0.99323, p-value = 1.922e-07
```

```
sum(is.na(df$pcths18_24))
```

```
## [1] 0
```

```
boxplot(df$pcths18_24)
```




```
bp<-boxplot(df$pcths18_24, id = list(n=Inf))
length(bp$out)
```

```
## [1] 33
```

```
sevout_pcths18_24 = (quantile(df$pcths18_24,0.25)+(3*((quantile(df$pcths18_24,0.75) - quantile(df$pcths
length(which(df$pcths18_24 > sevout_pcths18_24))
```

```
## [1] 9
```

```
df$f.pcths18_24 <- ifelse(df$pcths18_24 <= 29.2, 1,
                   ifelse(df$pcths18_24 > 29.2 & df$pcths18_24 <= 34.7, 2,
                    ifelse(df$pcths18_24 > 34.7 & df$pcths18_24 <= 40.5, 3,
```

```
                    ifelse(df$pcths18_24 > 40.5, 4,0))))
df$f.pcths18_24<- factor(df$f.pcths18_24,
       labels=c("Lowpcths18_24","LowMidpcths18_24","HighMidpcths18_24","Highpcths18_24"),
                                order = T,  levels=c(1,2,3,4))
table(df$f.pcths18_24)
```

```
##
##     Lowpcths18_24  LowMidpcths18_24 HighMidpcths18_24    Highpcths18_24
##               461               463               456               451
```

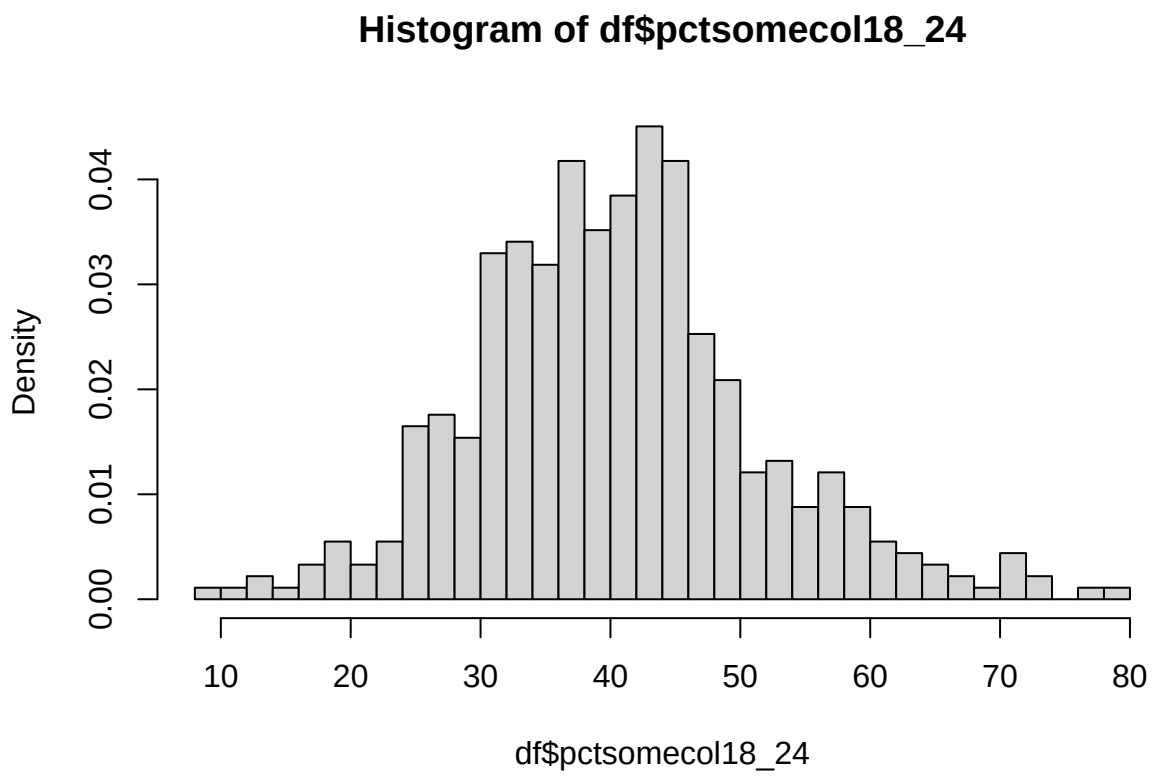
Variable 17: pctsomecol18_24

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 1376 missing values, so imputation is not necessary. It contains 15 outliers (of which 4 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctsomecol18_24" to create a discretization according to quartiles.

```
summary(df$pctsomecol18_24)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    9.60   33.25   40.10   40.48   46.10   78.30    1376
```

```
hist(df$pctsomecol18_24, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctsomecol18_24), sd(df$pctsomecol18_24)), add = T, col = "red")
```



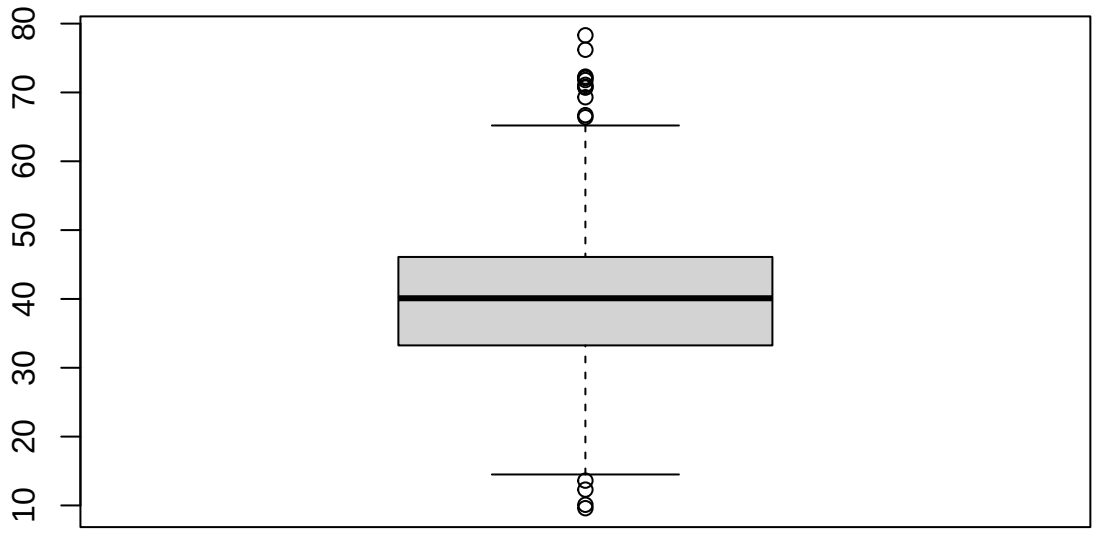
```r
shapiro.test(df$pctsomecol18_24)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctsomecol18_24
## W = 0.98569, p-value = 0.0001868
```

```r
sum(is.na(df$pctsomecol18_24))
```

```
## [1] 1376
```

```r
boxplot(df$pctsomecol18_24)
```




```r
bp<-boxplot(df$pctsomecol18_24, id = list(n=Inf))
length(bp$out)
```

```
## [1] 15
```

```r
sevout_pctsomecol18_24 = (quantile(df$pctsomecol18_24,0.25,na.rm =TRUE) +(3*((quantile(df$pctsomecol18_
length(which(df$pctsomecol18_24 > sevout_pctsomecol18_24))
```

```
## [1] 4
```

```
df$f.pctsomecol18_24 <- ifelse(df$pctsomecol18_24 <= 33.25, 1,
                        ifelse(df$pctsomecol18_24 > 33.25 & df$pctsomecol18_24 <= 40.1, 2,
                         ifelse(df$pctsomecol18_24 > 40.1 & df$pctsomecol18_24 <= 46.1, 3,
                         ifelse(df$pctsomecol18_24 > 46.1, 4,0))))
df$f.pctsomecol18_24 <- factor(df$f.pctsomecol18_24,
    labels=c("Lowpctsomecol18_24","LowMidpctsomecol18_24","HighMidpctsomecol18_24","Highpctsomecol18_24
                              order = T,levels=c(1,2,3,4))
table(df$f.pctsomecol18_24)
```

```
##
##     Lowpctsomecol18_24  LowMidpctsomecol18_24 HighMidpctsomecol18_24
##                    114                   114                   114
##    Highpctsomecol18_24
##                    113
```

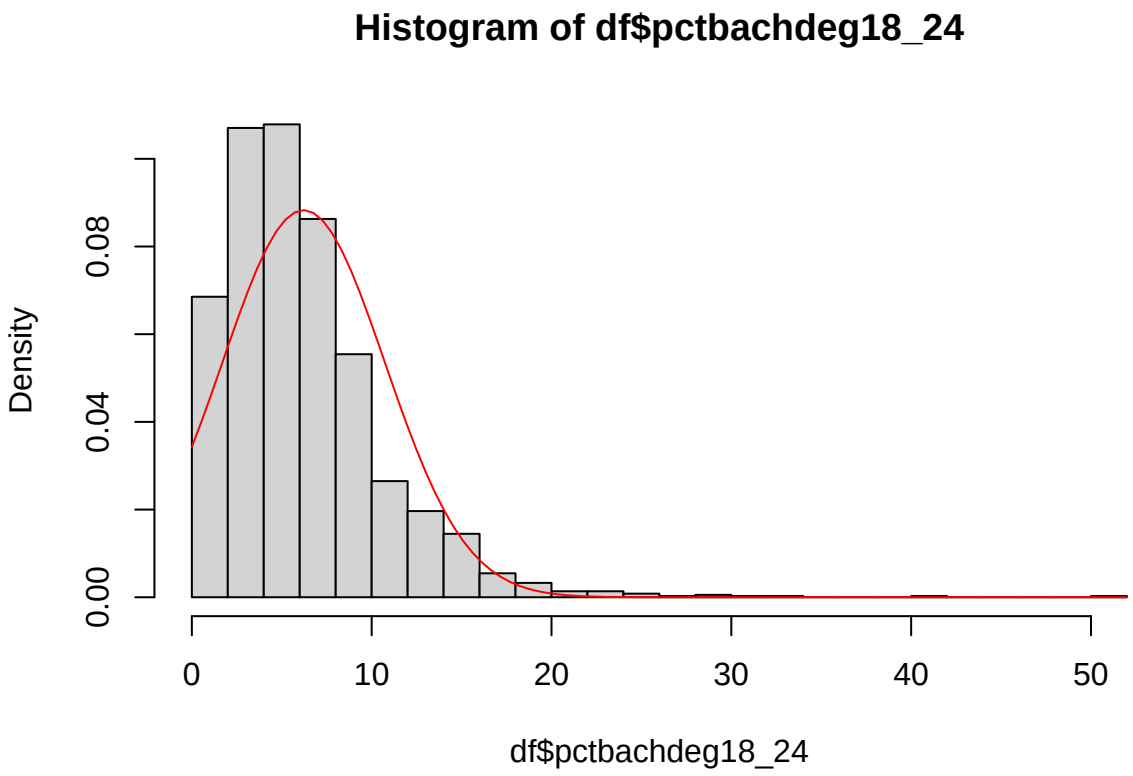
Variable 18: pctbachdeg18_24———————————

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 0 missing values, so imputation is not necessary. It contains 56 outliers (of which 31 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctbachdeg18_24" to create a discretization according to quartiles.

```
summary(df$pctbachdeg18_24)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   3.200   5.400   6.216   8.200  51.800
```

```
hist(df$pctbachdeg18_24, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctbachdeg18_24), sd(df$pctbachdeg18_24)), add = T, col = "red")
```



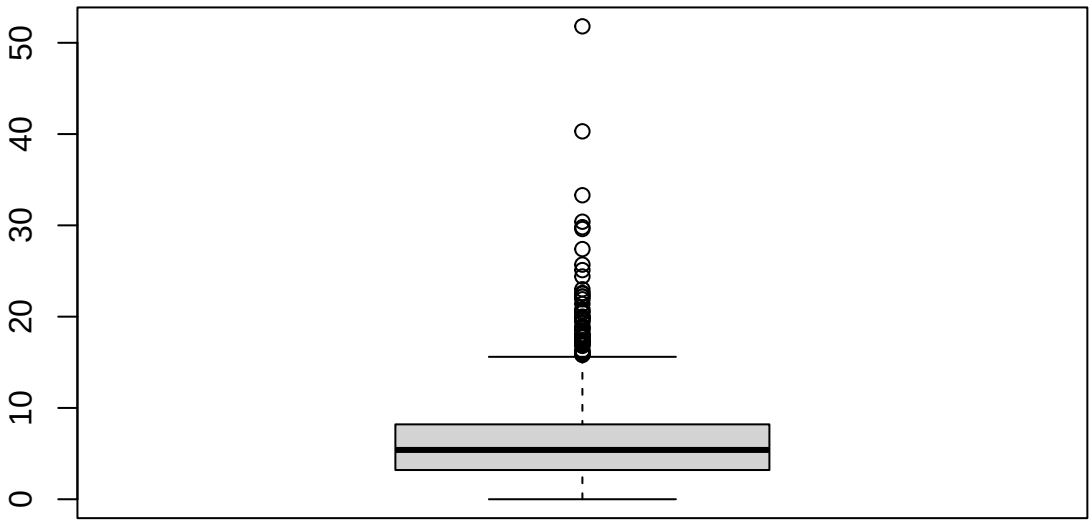

```
shapiro.test(df$pctbachdeg18_24)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctbachdeg18_24
## W = 0.87959, p-value < 2.2e-16
```

```
sum(is.na(df$pctbachdeg18_24))
```

```
## [1] 0
```

```
boxplot(df$pctbachdeg18_24)
```




```
bp<-boxplot(df$pctbachdeg18_24, id = list(n=Inf))
length(bp$out)
```

```
## [1] 56
```

```
sevout_pctbachdeg18_24 = (quantile(df$pctbachdeg18_24,0.25)+(3*((quantile(df$pctbachdeg18_24,0.75) - qua
length(which(df$pctbachdeg18_24 > sevout_pctbachdeg18_24))
```

```
## [1] 31
```

```
df$f.pctbachdeg18_24<- ifelse(df$pctbachdeg18_24 <= 3.2, 1,
                        ifelse(df$pctbachdeg18_24 > 3.2 & df$pctbachdeg18_24 <= 5.4, 2,
                        ifelse(df$pctbachdeg18_24 > 5.4 & df$pctbachdeg18_24 <= 8.2, 3,
                        ifelse(df$pctbachdeg18_24 > 8.2, 4,0))))
df$f.pctbachdeg18_24 <- factor(df$f.pctbachdeg18_24,
                               labels=c("Lowpctbachdeg18_24","LowMidpctbachdeg18_24","HighMidPopctbachdeg
                               order = T,
                               levels=c(1,2,3,4))
table(df$f.pctbachdeg18_24)
```

```
##
##       Lowpctbachdeg18_24    LowMidpctbachdeg18_24 HighMidPopctbachdeg18_24
##                      473                     445                      462
##      Highpctbachdeg18_24
##                      451
```

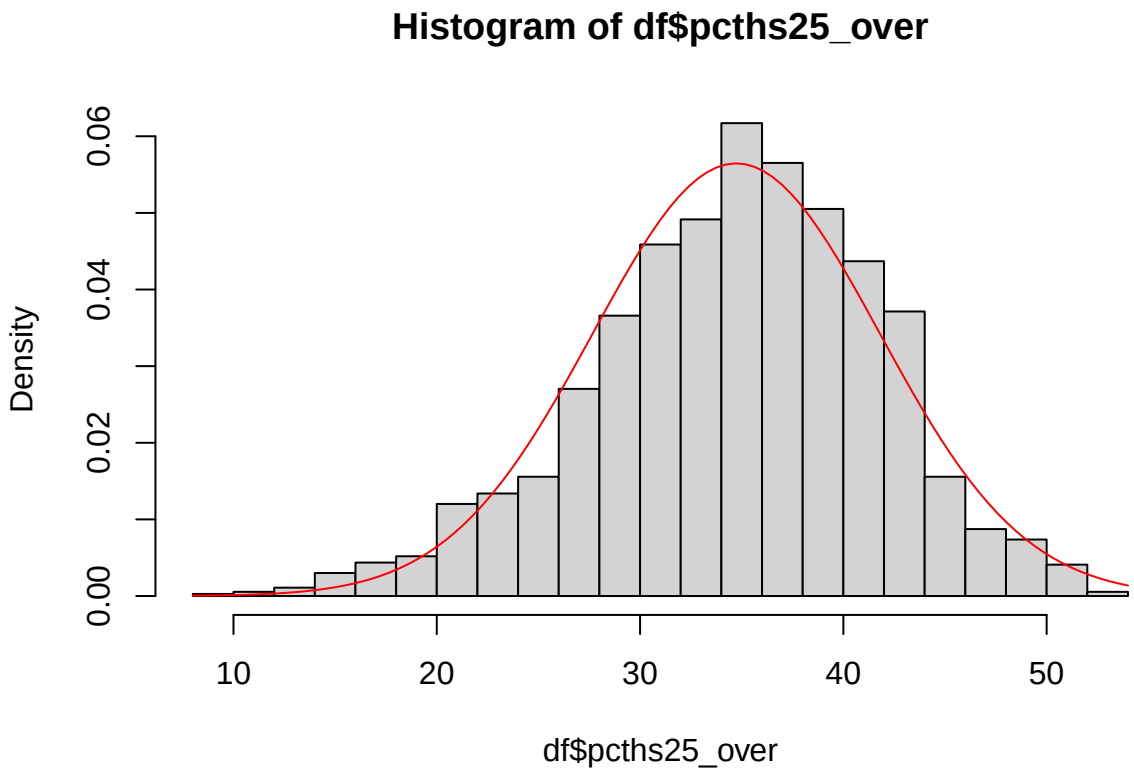
Variable 19: pcths25_over

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 0 missing values, so imputation is not necessary. It contains 18 outliers (of which 0 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pcths25_over" to create a discretization according to quartiles.

```
summary(df$pcths25_over)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    8.30   30.35   35.30   34.73   39.65   52.70
```

```
hist(df$pcths25_over, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pcths25_over), sd(df$pcths25_over)), add = T, col = "red")
```



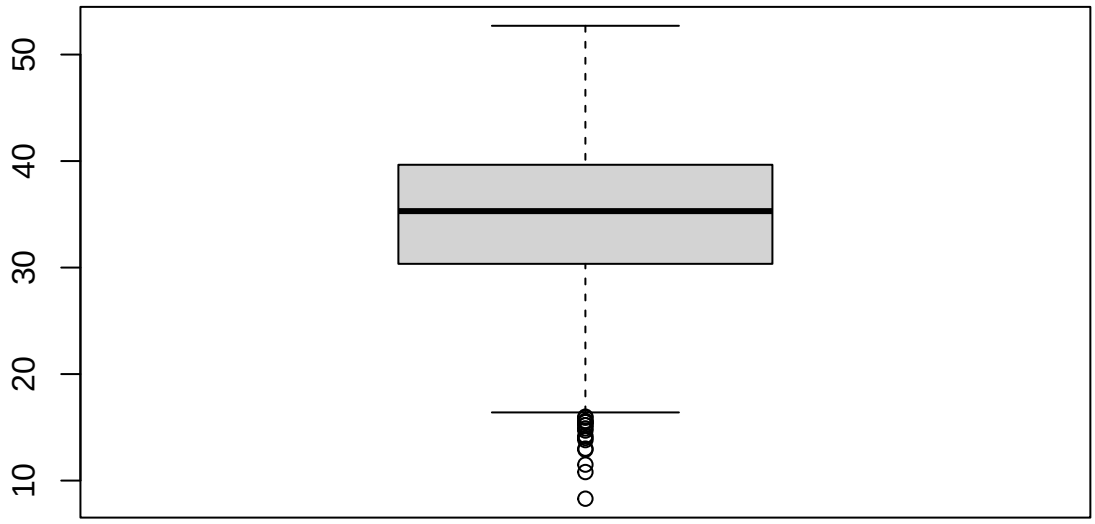

```
shapiro.test(df$pcths25_over)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pcths25_over
## W = 0.99107, p-value = 3.741e-09
```

```r
sum(is.na(df$pcths25_over))
```

```
## [1] 0
```

```r
boxplot(df$pcths25_over)
```




```r
bp<-boxplot(df$pcths25_over, id = list(n=Inf))
length(bp$out)
```

```
## [1] 18
```

```r
sevout_pcths25_over = (quantile(df$pcths25_over,0.25)+(3*((quantile(df$pcths25_over,0.75) - quantile(df$
length(which(df$pcths25_over > sevout_pcths25_over))
```

```
## [1] 0
```

```r
df$f.pcths25_over <- ifelse(df$pcths25_over <= 30.35, 1,
                    ifelse(df$pcths25_over > 30.35 & df$pcths25_over <= 35.30, 2,
                    ifelse(df$pcths25_over > 35.3 & df$pcths25_over <= 39.65, 3,
                     ifelse(df$pcths25_over > 39.65, 4,0))))
df$f.pcths25_over <- factor(df$f.pcths25_over,
    labels=c("Lowpcths25_over","LowMidpcths25_over","HighMidpcths25_over","Highpcths25_over"),
                order = T,  levels=c(1,2,3,4))
table(df$f.pcths25_over)
```

```
##
##     Lowpcths25_over  LowMidpcths25_over HighMidpcths25_over    Highpcths25_over
##                 458                 469                 446                 458
```

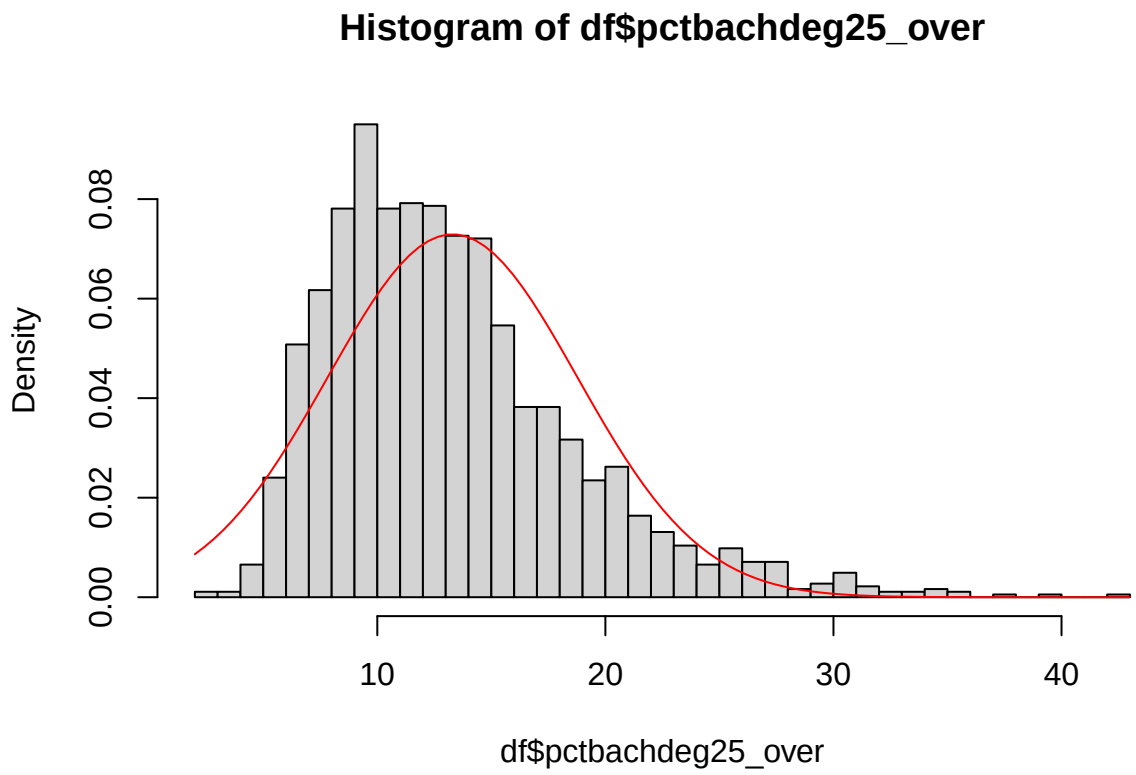
Variable 20:pctbachdeg25_over———————————

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 0 missing values, so imputation is not necessary. It contains 59 outliers (of which 27 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctbachdeg25_over" to create a discretization according to quartiles.

```
summary(df$pctbachdeg25_over)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     2.5     9.3    12.3    13.3    16.0    42.2
```

```
hist(df$pctbachdeg25_over, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctbachdeg25_over), sd(df$pctbachdeg25_over)), add = T, col = "red")
```




```
shapiro.test(df$pctbachdeg25_over)
```

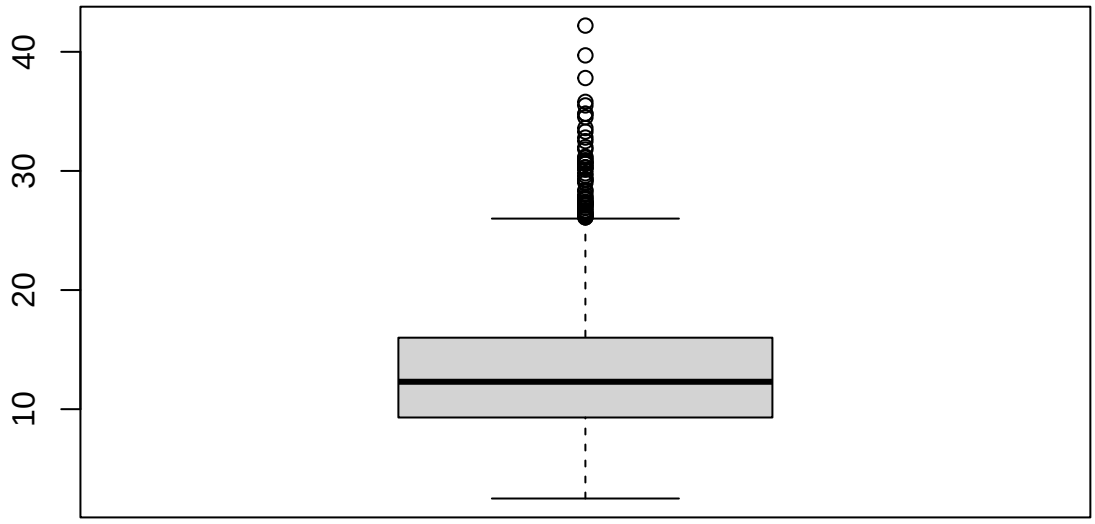
```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  df$pctbachdeg25_over
## W = 0.92998, p-value < 2.2e-16
```

```
sum(is.na(df$pctbachdeg25_over))
```

```
## [1] 0
```

```
boxplot(df$pctbachdeg25_over)
```




```
bp<-boxplot(df$pctbachdeg25_over, id = list(n=Inf))
length(bp$out)
```

```
## [1] 59
```

```
sevout_pctbachdeg25_over = (quantile(df$pctbachdeg25_over,0.25)+(3*((quantile(df$pctbachdeg25_over,0.75)
length(which(df$pctbachdeg25_over > sevout_pctbachdeg25_over))
```

```
## [1] 27
```

```
df$f.pctbachdeg25_over <- ifelse(df$pctbachdeg25_over <= 9.3, 1,
                          ifelse(df$pctbachdeg25_over > 9.3 & df$pctbachdeg25_over <= 12.3, 2,
                          ifelse(df$pctbachdeg25_over > 12.3 & df$pctbachdeg25_over <= 16, 3,
```

```
                            ifelse(df$pctbachdeg25_over > 16, 4,0))))
df$f.pctbachdeg25_over <- factor(df$f.pctbachdeg25_over,
                            labels=c("LowPovertypercent","LowMidPovertypercent","HighMidPovertypercen
                            order = T,
                            levels=c(1,2,3,4))
table(df$f.pctbachdeg25_over)
```

```
##
##     LowPovertypercent  LowMidPovertypercent HighMidPovertypercent
##                   459                   458                   463
##    HighPovertypercent
##                   451
```

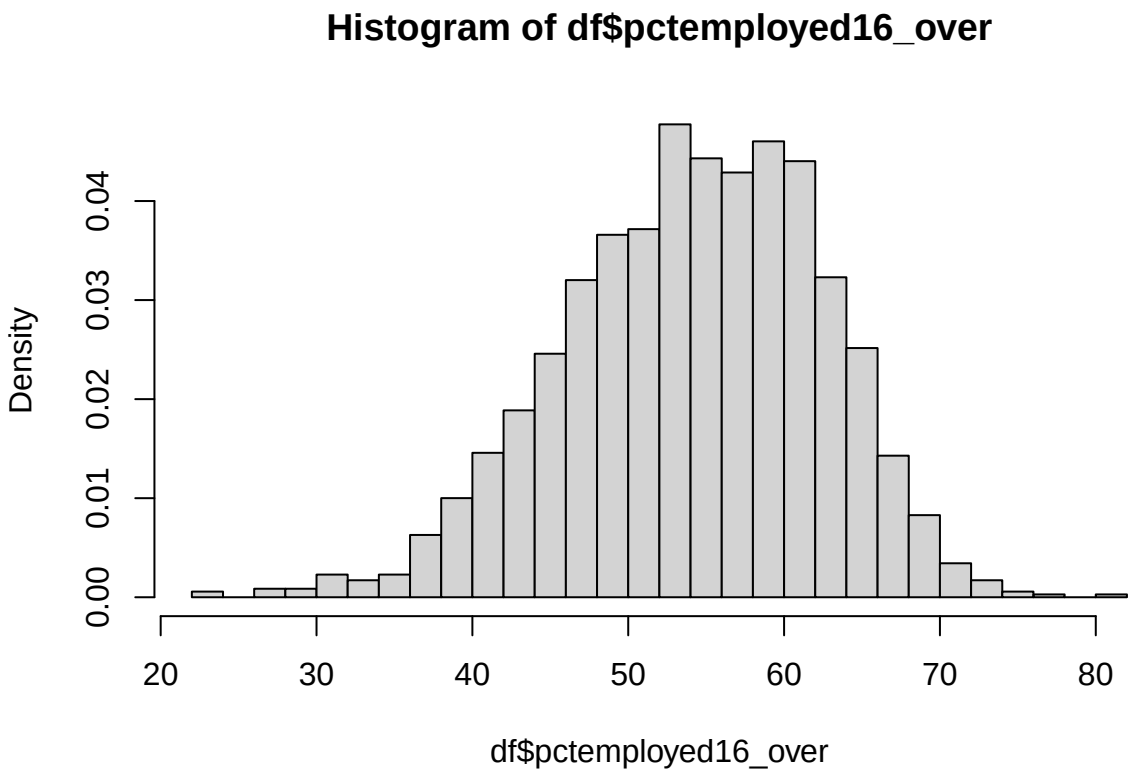
Variable 21: pctemployed16_over

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 82 missing values, so imputation is not necessary. It contains 11 outliers (of which 0 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctemployed16_over" to create a discretization according to quartiles.

```
summary(df$pctemployed16_over)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   23.90   48.60   54.50   54.21   60.30   80.10      82
```

```
hist(df$pctemployed16_over, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctemployed16_over), sd(df$pctemployed16_over)), add = T, col = "red")
```



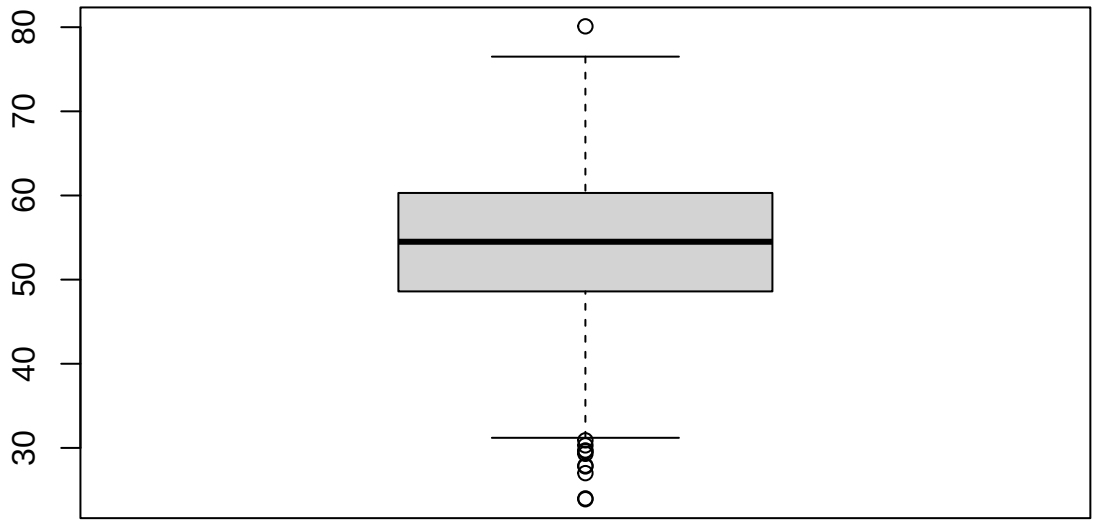

```
shapiro.test(df$pctemployed16_over)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctemployed16_over
## W = 0.99196, p-value = 3.371e-08
```

```
sum(is.na(df$pctemployed16_over))
```

```
## [1] 82
```

```
boxplot(df$pctemployed16_over)
```




```
bp<-boxplot(df$pctemployed16_over, id = list(n=Inf))
length(bp$out)
```

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

```
sevout_pctemployed16_over = (quantile(df$pctemployed16_over,0.25,na.rm=TRUE)+
              (3*((quantile(df$pctemployed16_over,0.75,na.rm=TRUE) -
                     quantile(df$pctemployed16_over, 0.25,na.rm=TRUE)))))
length(which(df$pctemployed16_over > sevout_pctemployed16_over))
```

```
## [1] 0
```

```
df$f.pctemployed16_over <- ifelse(df$pctemployed16_over <= 48.6, 1,
                        ifelse(df$pctemployed16_over > 48.6 & df$pctemployed16_over <= 54.21, 2,
                        ifelse(df$pctemployed16_over > 54.21 & df$pctemployed16_over <= 60.3, 3,
                        ifelse(df$pctemployed16_over > 60.3, 4,0))))
df$f.pctemployed16_over <- factor(df$f.pctemployed16_over,
    labels=c("Lowpctemployed16_over","LowMidpctemployed16_over",
             "HighMidpctemployed16_over","Highpctemployed16_over"),
                              order = T,    levels=c(1,2,3,4))
table(df$f.pctemployed16_over)
```

```
##
##     Lowpctemployed16_over  LowMidpctemployed16_over HighMidpctemployed16_over
```

```
##                         442                         412                         466
##    Highpctemployed16_over
##                         429
```

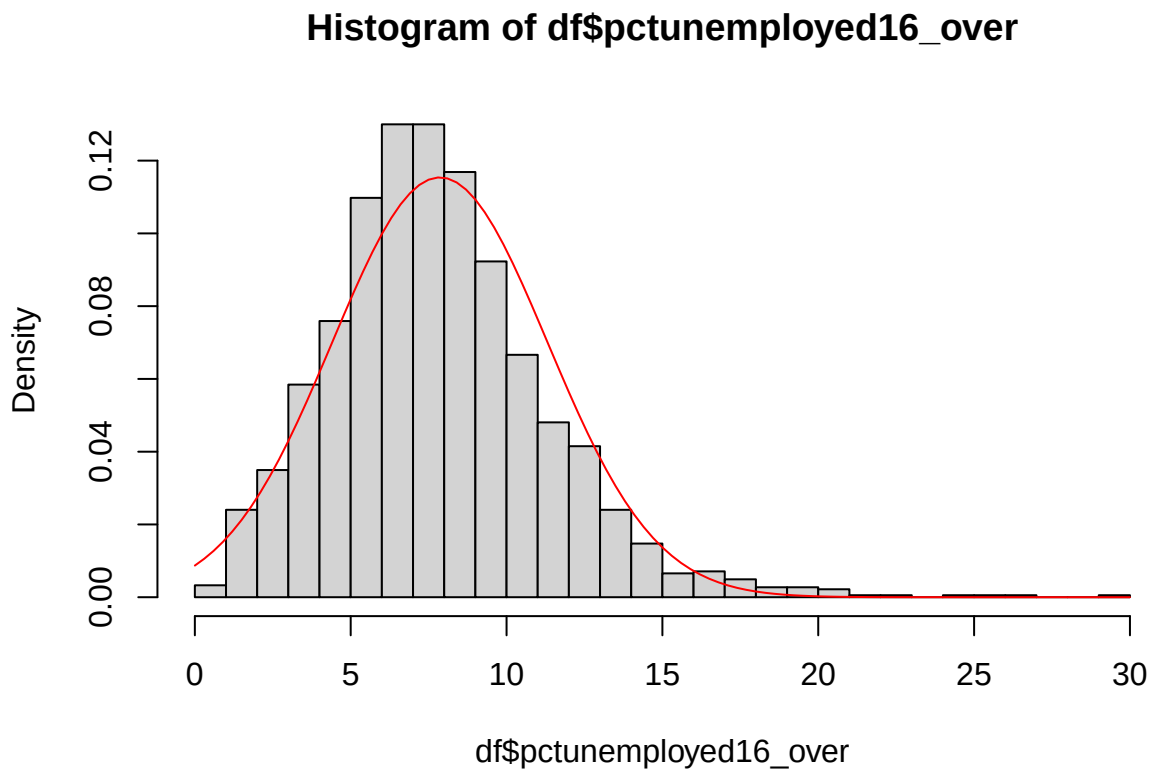
Variable 22: pctunemployed16_over

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 0 missing values, so imputation is not necessary. It contains 42 outliers (of which 18 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctunemployed16_over" to create a discretization according to quartiles.

```
summary(df$pctunemployed16_over)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.700   5.500   7.500   7.861   9.750  29.400
```

```
hist(df$pctunemployed16_over, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctunemployed16_over), sd(df$pctunemployed16_over)), add = T, col = "red")
```




```
shapiro.test(df$pctunemployed16_over)
```

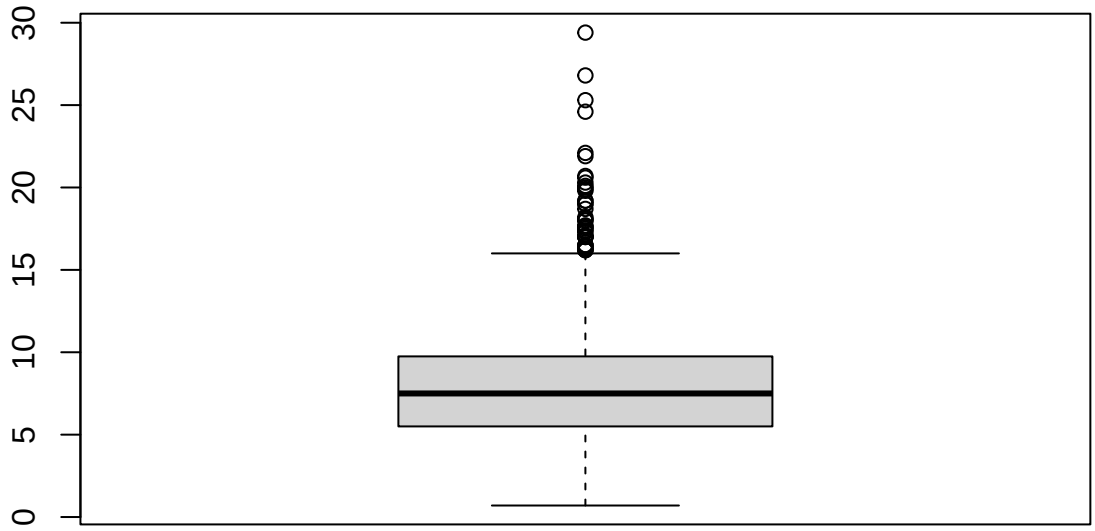
```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  df$pctunemployed16_over
## W = 0.9612, p-value < 2.2e-16
```

```
sum(is.na(df$pctunemployed16_over))
```

```
## [1] 0
```

```
boxplot(df$pctunemployed16_over)
```




```
bp<-boxplot(df$pctunemployed16_over, id = list(n=Inf))
length(bp$out)
```

```
## [1] 42
```

```
sevout_pctunemployed16_over = (quantile(df$pctunemployed16_over,0.25,na.rm =TRUE)+(3*((quantile(df$pctu
length(which(df$pctunemployed16_over > sevout_pctunemployed16_over))
```

```
## [1] 18
```

```
df$f.pctunemployed16_over<- ifelse(df$pctunemployed16_over <= 5.5, 1,
                       ifelse(df$pctunemployed16_over > 5.5 & df$pctunemployed16_over <= 7.5, 2,
                       ifelse(df$pctunemployed16_over > 7.5 & df$pctunemployed16_over <= 9.75, 3,
```

```
                      ifelse(df$pctunemployed16_over > 9.75, 4,0))))
df$f.pctunemployed16_over <- factor(df$f.pctunemployed16_over,
  labels=c("Lowpctunemployed16_over","LowMidpctunemployed16_over","HighMidpctunemployed16_over","Highpc
                               order = T, levels=c(1,2,3,4))
table(df$f.pctunemployed16_over)
```

```
##
##     Lowpctunemployed16_over  LowMidpctunemployed16_over
##                         467                         453
## HighMidpctunemployed16_over    Highpctunemployed16_over
##                         453                         458
```

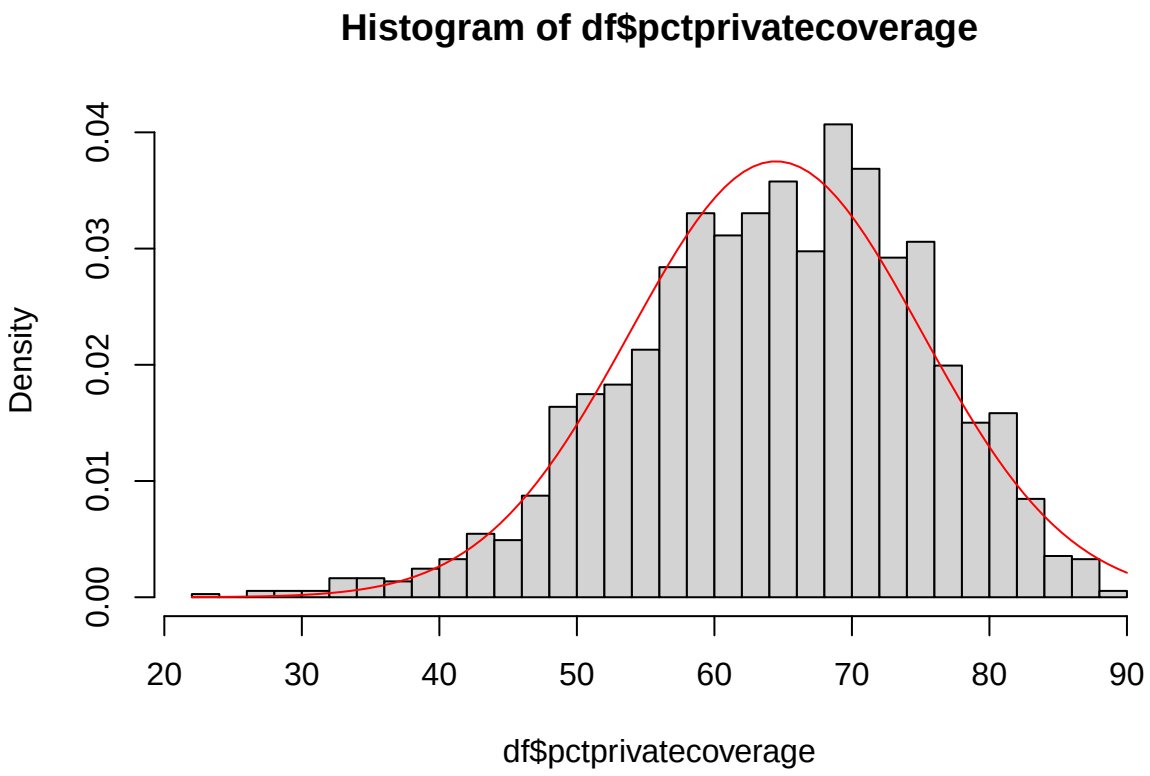
Variable 23: pctprivatecoverage

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 0 missing values, so imputation is not necessary. It contains 17 outliers (of which 0 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctprivatecoverage" to create a discretization according to quartiles.

```
summary(df$pctprivatecoverage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.40   57.50   65.20   64.47   72.10   89.60
```

```
hist(df$pctprivatecoverage, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctprivatecoverage), sd(df$pctprivatecoverage)), add = T, col = "red")
```



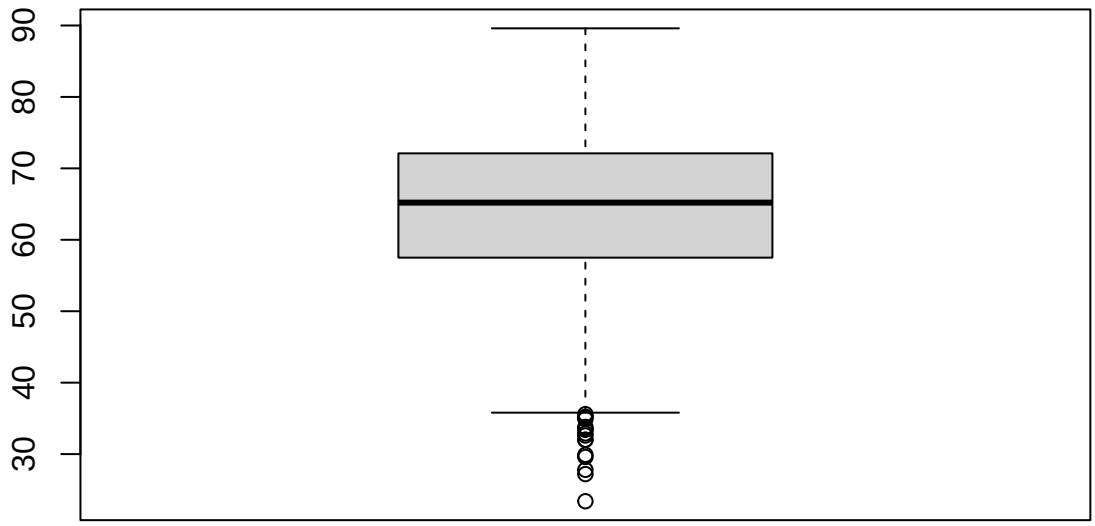

```
shapiro.test(df$pctprivatecoverage)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctprivatecoverage
## W = 0.98964, p-value = 3.725e-10
```

```
sum(is.na(df$pctprivatecoverage))
```

```
## [1] 0
```

```
boxplot(df$pctprivatecoverage)
```




```
bp<-boxplot(df$pctprivatecoverage, id = list(n=Inf))
length(bp$out)
```

```
## [1] 17
```

```
sevout_pctprivatecoverage = (quantile(df$pctprivatecoverage,0.25)+(3*((quantile(df$pctprivatecoverage,0
length(which(df$pctprivatecoverage > sevout_pctprivatecoverage))
```

```
## [1] 0
```

```
df$f.pctprivatecoverage <- ifelse(df$pctprivatecoverage <= 57.5, 1,
                                  ifelse(df$pctprivatecoverage > 57.5 & df$pctprivatecoverage <= 65.2, 2,
                                         ifelse(df$pctprivatecoverage > 65.2 & df$pctprivatecoverage <= 72.
                                                ifelse(df$pctprivatecoverage > 72.1, 4,0))))
df$f.pctprivatecoverage <- factor(df$f.pctprivatecoverage,
            labels=c("Lowpctprivatecoverage","LowMidpctprivatecoverage","HighMidpctprivatecoverage","Hig
                order = T,       levels=c(1,2,3,4))
table(df$f.pctprivatecoverage)
```

```
##
##     Lowpctprivatecoverage  LowMidpctprivatecoverage HighMidpctprivatecoverage
##                       460                       464                       451
##    Highpctprivatecoverage
##                       456
```

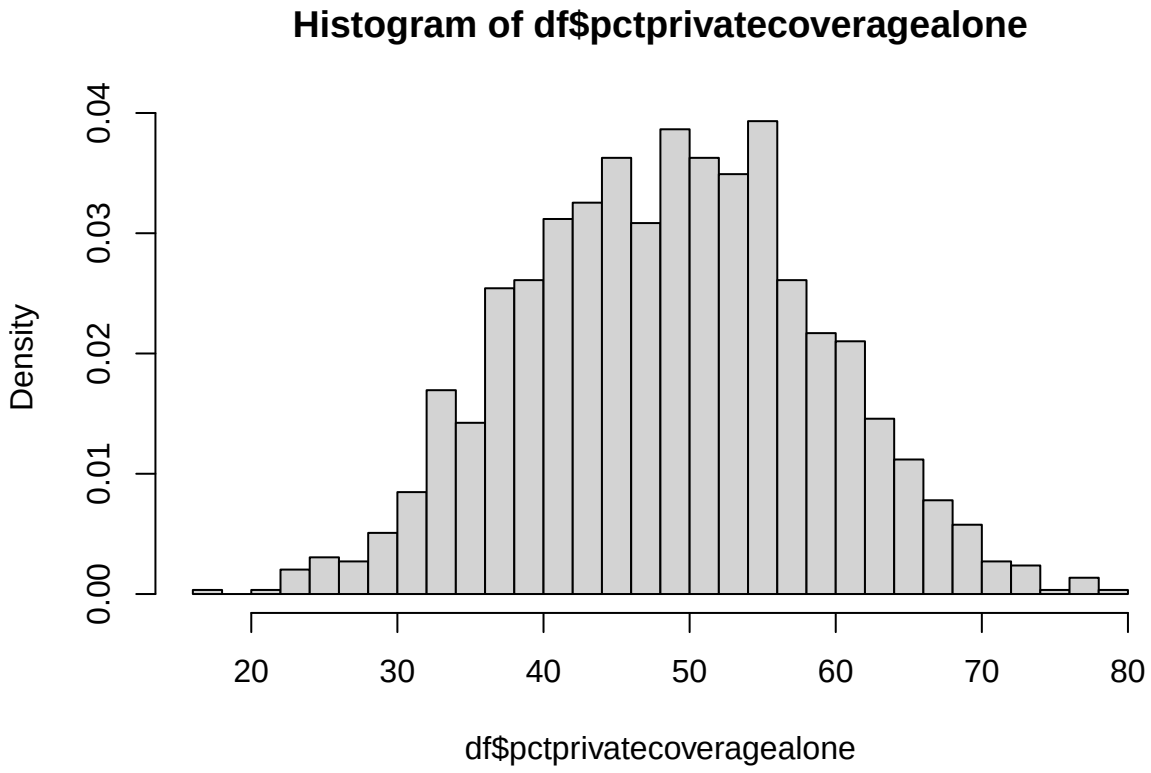
Variable 24: pctprivatecoveragealone

This is a continuous proportion variable. The data appear normally distributed, which is confirmed by the p-value greater than 0.05 from the Shapiro normality test. A histogram is used to visualize the data. The variable contains 356 missing values, so imputation is not necessary. It contains 4 outliers (of which 0 are severe), all on the high side of the spectrum. We created an additional ordinal mpg factor "f.pctprivatecoveragealone" to create a discretization according to quartiles.

```
summary(df$pctprivatecoveragealone)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   16.80   41.50   49.00   48.65   55.50   78.90     356
```

```
hist(df$pctprivatecoveragealone, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctprivatecoveragealone), sd(df$pctprivatecoveragealone)), add = T, col = "red")
```



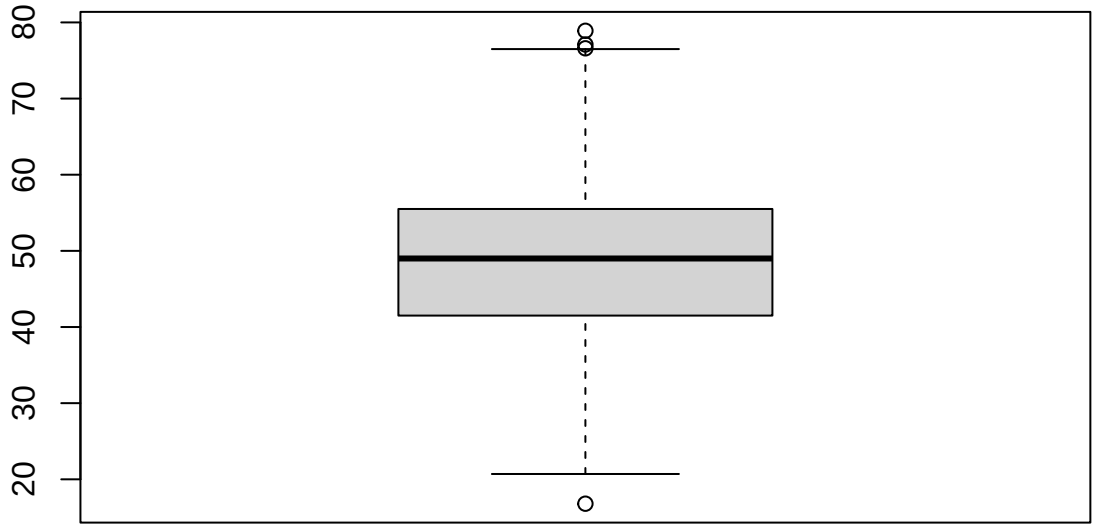

```
shapiro.test(df$pctprivatecoveragealone)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctprivatecoveragealone
## W = 0.99831, p-value = 0.1453
```

```r
sum(is.na(df$pctprivatecoveragealone))
```

```
## [1] 356
```

```r
boxplot(df$pctprivatecoveragealone)
```




```r
bp<-boxplot(df$pctprivatecoveragealone, id = list(n=Inf))
length(bp$out)
```

```
## [1] 4
```

```r
sevout_pctprivatecoveragealone = (quantile(df$pctprivatecoveragealone,0.25,na.rm =TRUE)+(3*((quantile(d
length(which(df$pctprivatecoveragealone > sevout_pctprivatecoveragealone))
```

```
## [1] 0
```

```r
df$f.pctprivatecoveragealone <- ifelse(df$pctprivatecoveragealone <= 41.5, 1,
                                ifelse(df$pctprivatecoveragealone > 41.5 & df$pctprivatecoveragealone <= 
                                       ifelse(df$pctprivatecoveragealone > 49 & df$pctprivatecoveragealon
                                              ifelse(df$pctprivatecoveragealone > 55.5, 4,0))))
df$f.pctprivatecoveragealone <- factor(df$f.pctprivatecoveragealone,
  labels=c("Lowpctprivatecoveragealone","LowMidpctprivatecoveragealone","HighMidpctprivatecoveragealone
                                order = T, levels=c(1,2,3,4))
table(df$f.pctprivatecoveragealone)
```

```
##
##     Lowpctprivatecoveragealone  LowMidpctprivatecoveragealone
##                            371                            370
## HighMidpctprivatecoveragealone    Highpctprivatecoveragealone
##                            367                            367
```

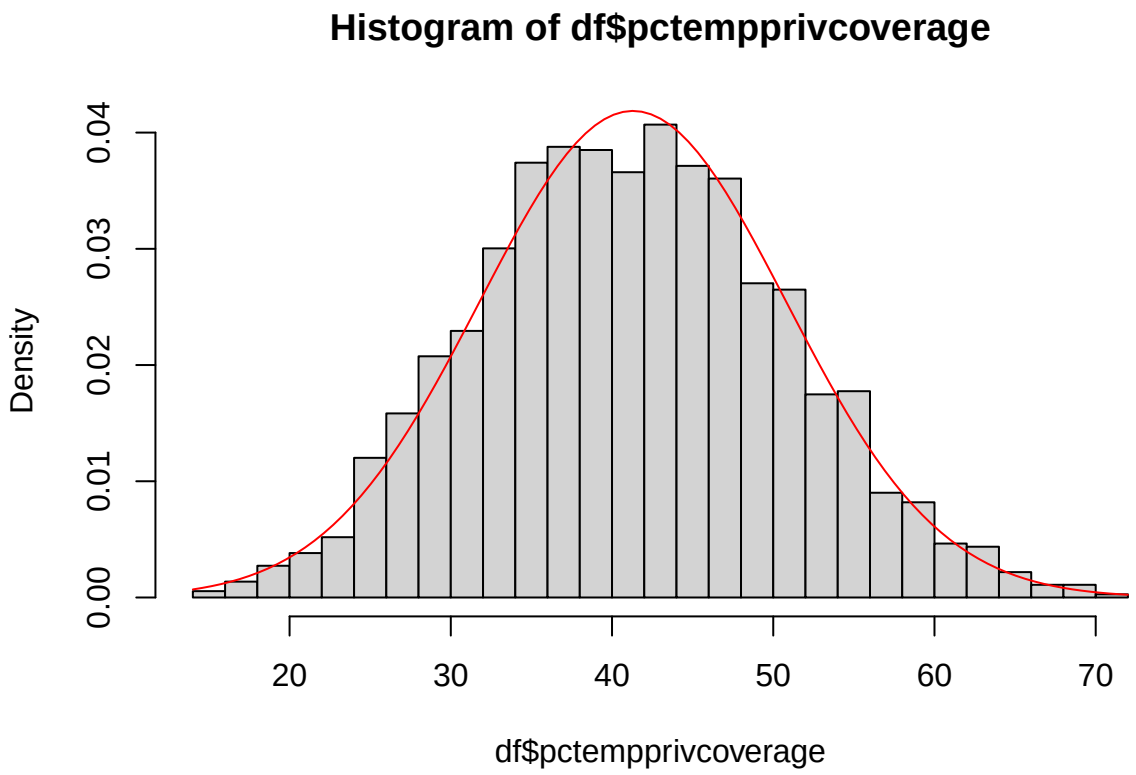
Variable 25: pctempprivcoverage

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 7 outliers (of which 0 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctempprivcoverage" to create a discretization according to quartiles.

```
summary(df$pctempprivcoverage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.30   34.60   41.10   41.29   47.70   70.20
```

```
hist(df$pctempprivcoverage, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctempprivcoverage), sd(df$pctempprivcoverage)), add = T, col = "red")
```



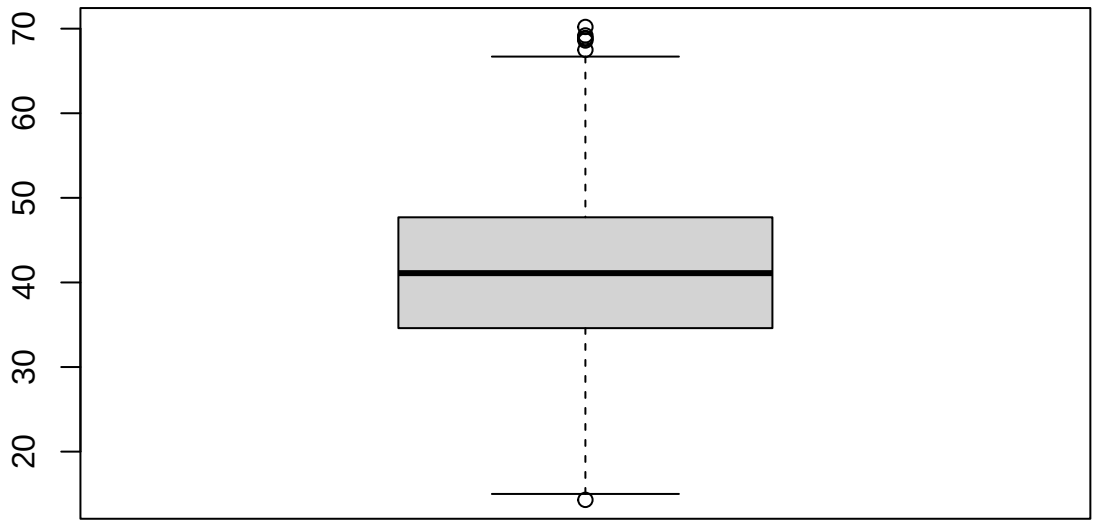

```
shapiro.test(df$pctempprivcoverage)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctempprivcoverage
## W = 0.99807, p-value = 0.02861
```

```
sum(is.na(df$pctempprivcoverage))
```

```
## [1] 0
```

```
boxplot(df$pctempprivcoverage)
```




```
bp<-boxplot(df$pctempprivcoverage, id = list(n=Inf))
length(bp$out)
```

```
## [1] 7
```

```
sevout_pctempprivcoverage = (quantile(df$pctempprivcoverage,0.25)+(3*((quantile(df$pctempprivcoverage,0
length(which(df$pctempprivcoverage > sevout_pctempprivcoverage))
```

```
## [1] 0
```

```
df$f.pctempprivcoverage <- ifelse(df$pctempprivcoverage <= 34.6, 1,
                          ifelse(df$pctempprivcoverage > 34.6 & df$pctempprivcoverage <= 41.1, 2,
                          ifelse(df$pctempprivcoverage > 41.1 & df$pctempprivcoverage <= 47.7, 3,
                          ifelse(df$pctempprivcoverage > 47.7, 4,0))))
df$f.pctempprivcoverage <- factor(df$f.pctempprivcoverage,
      labels=c("Lowpctempprivcoverage","LowMidpctempprivcoverage","HighMidpctempprivcoverage","Highpcte
                                order = T, levels=c(1,2,3,4))
table(df$f.pctempprivcoverage)
```

```
##
##     Lowpctempprivcoverage  LowMidpctempprivcoverage HighMidpctempprivcoverage
##                       465                       454                       456
##    Highpctempprivcoverage
##                       456
```

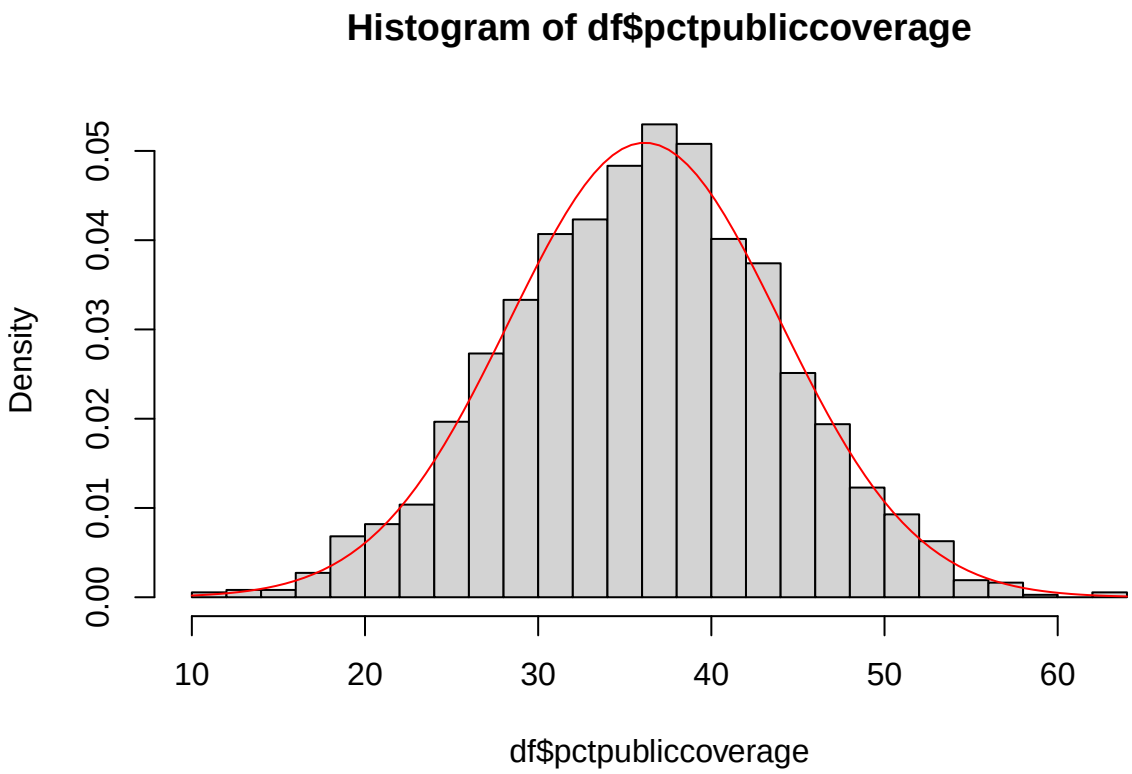
Variable 26:pctpubliccoverage———————————

This is a continuous proportion variable. The data appear normally distributed, which is confirmed by the p-value greater than 0.05 from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 13 outliers (of which 1 is severe), all on the high side of the spectrum. We created an additional ordinal mpg factor "f.pctpubliccoverage" to create a discretization according to quartiles.

```
summary(df$pctpubliccoverage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   11.20   30.90   36.30   36.15   41.40   62.70
```

```
hist(df$pctpubliccoverage, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctpubliccoverage), sd(df$pctpubliccoverage)), add = T, col = "red")
```



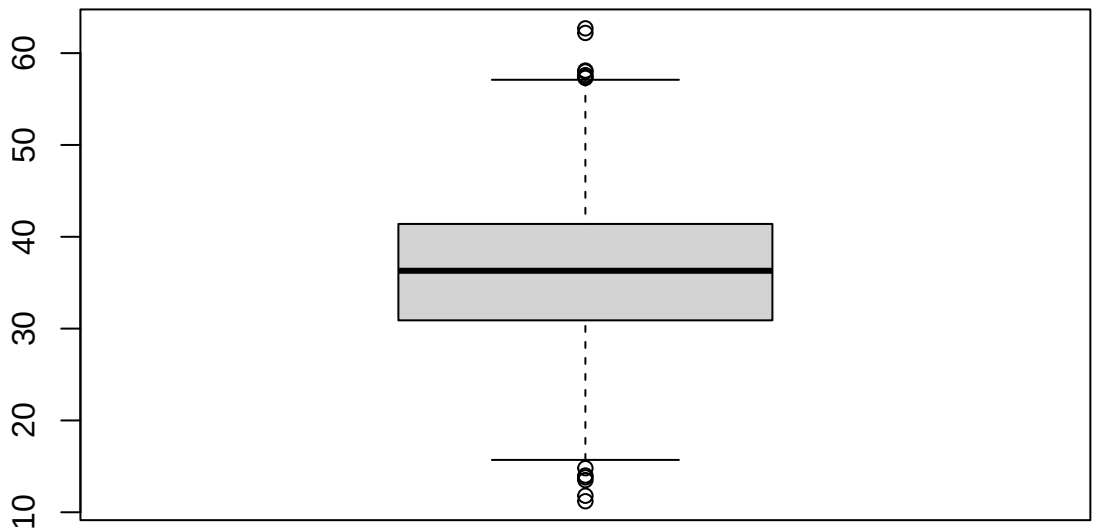

```
shapiro.test(df$pctpubliccoverage)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctpubliccoverage
## W = 0.99947, p-value = 0.9186
```

```
sum(is.na(df$pctpubliccoverage))
```

```
## [1] 0
```

```
boxplot(df$pctpubliccoverage)
```




```
bp<-boxplot(df$pctpubliccoverage, id = list(n=Inf))
length(bp$out)
```

```
## [1] 13
```

```
sevout_pctpubliccoverage = (quantile(df$pctpubliccoverage,0.25)+(3*((quantile(df$pctpubliccoverage,0.75)
length(which(df$pctpubliccoverage > sevout_pctpubliccoverage))
```

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

```
df$f.pctpubliccoverage <- ifelse(df$pctpubliccoverage <= 30.9, 1,
                          ifelse(df$pctpubliccoverage > 30.9 & df$pctpubliccoverage <= 36.3, 2,
                         ifelse(df$pctpubliccoverage > 36.3 & df$pctpubliccoverage <= 41.4, 3,
                           ifelse(df$pctpubliccoverage > 41.4, 4,0))))
df$f.pctpubliccoverage <- factor(df$f.pctpubliccoverage,
  labels=c("Lowpctpubliccoverage","LowMidpctpubliccoverage","HighMidpctpubliccoverage","Highpctpublicco
                               order = T, levels=c(1,2,3,4))
table(df$f.pctpubliccoverage)
```

```
##
##     Lowpctpubliccoverage  LowMidpctpubliccoverage HighMidpctpubliccoverage
##                      463                     459                     454
##    Highpctpubliccoverage
##                      455
```

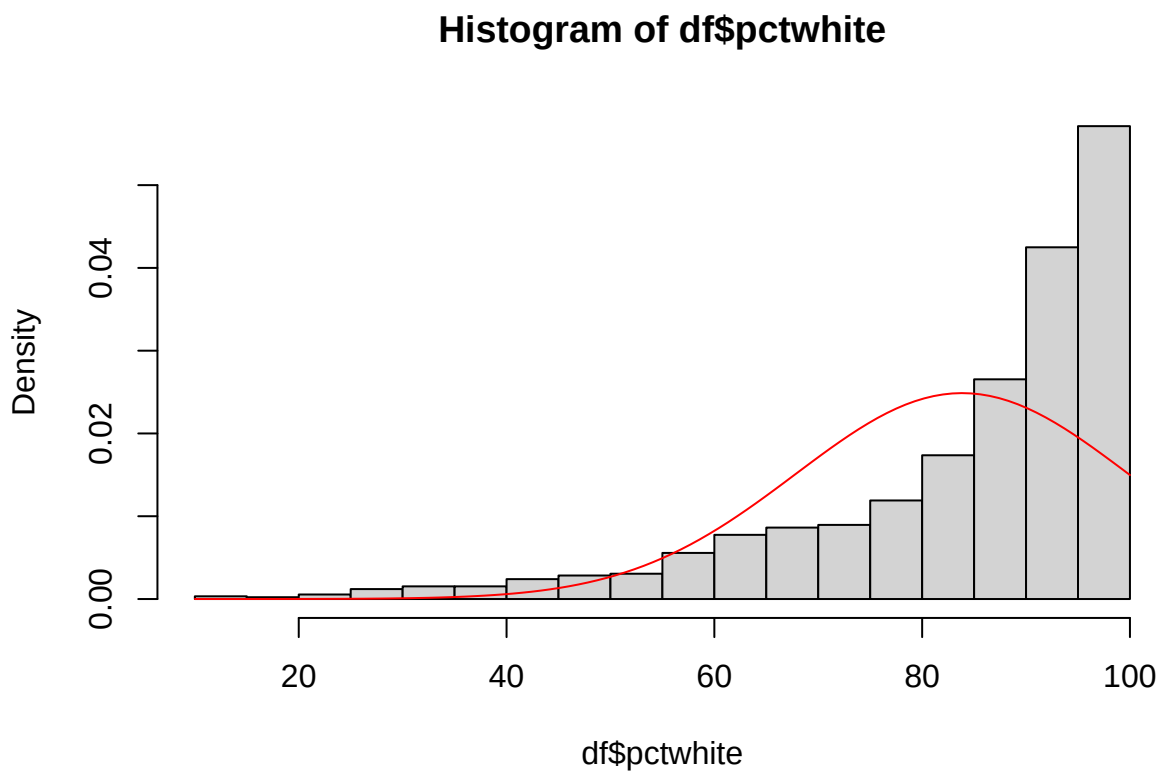
Variable 28: pctwhite

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 97 outliers (of which 0 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctwhite" to create a discretization according to quartiles.

```
summary(df$pctwhite)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   12.27   77.31   89.90   83.85   95.57   99.69
```

```
hist(df$pctwhite, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctwhite), sd(df$pctwhite)), add = T, col = "red")
```



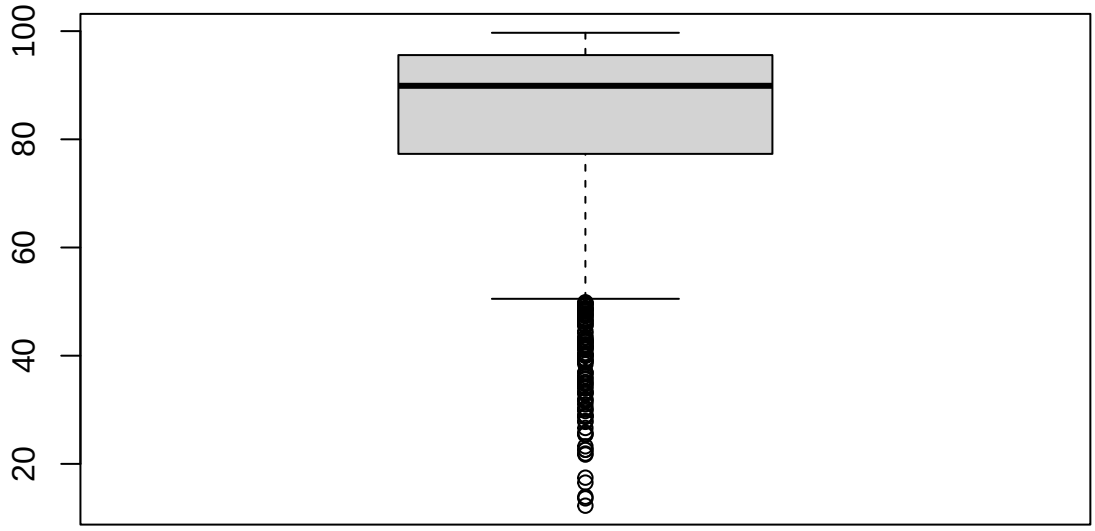

```
shapiro.test(df$pctwhite)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctwhite
## W = 0.80758, p-value < 2.2e-16
```

```r
sum(is.na(df$pctwhite))
```

```
## [1] 0
```

```r
boxplot(df$pctwhite)
```




```r
bp<-boxplot(df$pctwhite, id = list(n=Inf))
length(bp$out)
```

```
## [1] 97
```

```r
sevout_pctwhite = (quantile(df$pctwhite,0.25)+(3*((quantile(df$pctwhite,0.75) - quantile(df$pctwhite, 0
length(which(df$pctwhite > sevout_pctwhite))
```

```
## [1] 0
```

```r
df$f.pctwhite <- ifelse(df$pctwhite <= 77.31, 1,
                ifelse(df$pctwhite > 77.31 & df$pctwhite <= 89.9, 2,
                ifelse(df$pctwhite > 89.9 & df$pctwhite <= 95.57, 3,
                ifelse(df$pctwhite > 95.57, 4,0))))
df$f.pctwhite <- factor(df$f.pctwhite,
                            labels=c("Lowpctwhite","LowMidpctwhite","HighMidpctwhite","Highpctwhite")
                            order = T, levels=c(1,2,3,4))
table(df$f.pctwhite)
```

```
##
##     Lowpctwhite  LowMidpctwhite HighMidpctwhite    Highpctwhite
##             458             459             456             458
```

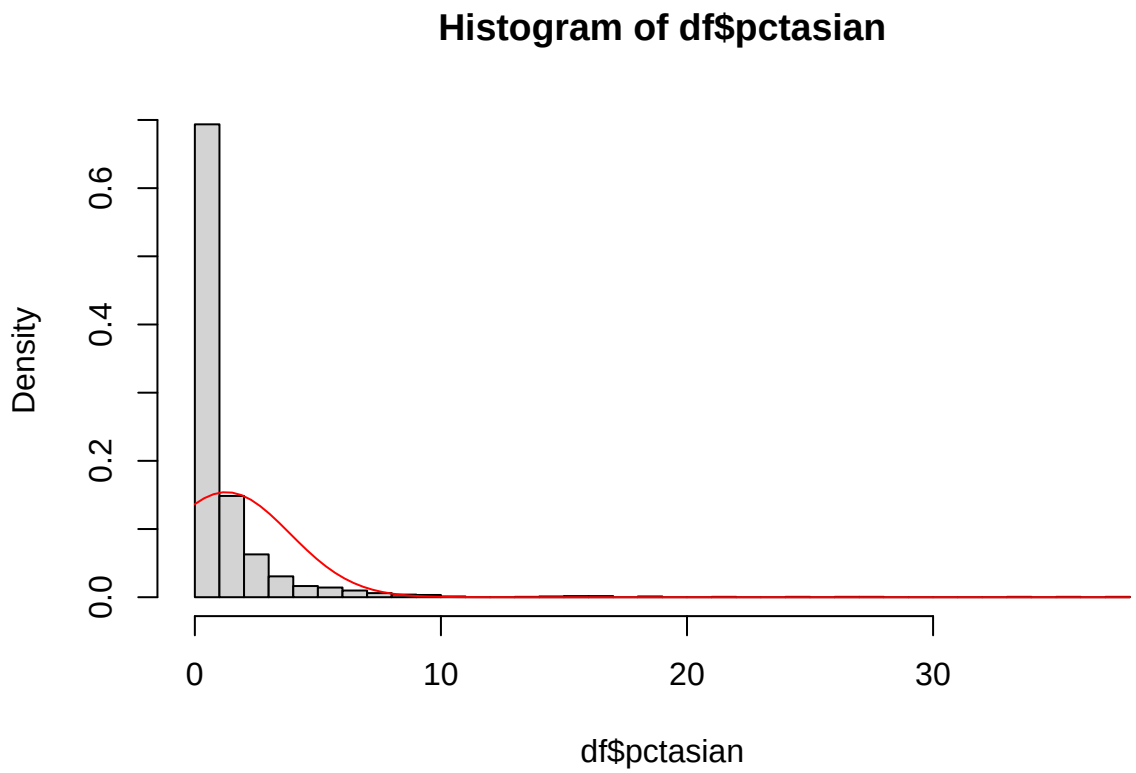
Variable 30:pctasian———————————

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 198 outliers (of which 156 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctasian" to create a discretization according to quartiles.

```
summary(df$pctasian)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.2582  0.5495  1.2743  1.2515 37.1569
```

```
hist(df$pctasian, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctasian), sd(df$pctasian)), add = T, col = "red")
```




```
shapiro.test(df$pctasian)
```

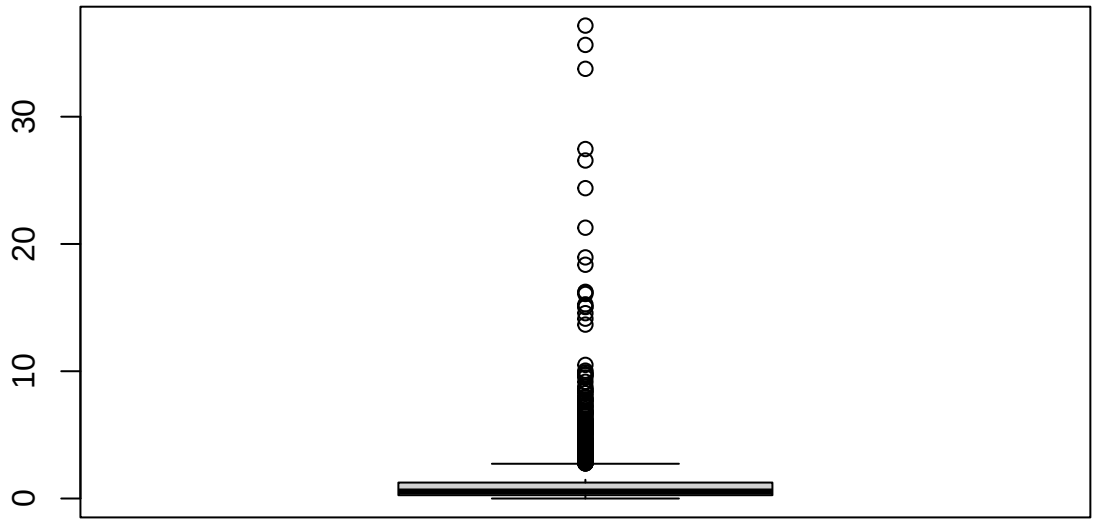
```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  df$pctasian
## W = 0.41908, p-value < 2.2e-16
```

```
sum(is.na(df$pctasian))
```

```
## [1] 0
```

```
boxplot(df$pctasian)
```




```
bp<-boxplot(df$pctasian, id = list(n=Inf))
length(bp$out)
```

```
## [1] 198
```

```
sevout_pctasian= (quantile(df$pctasian,0.25)+(3*((quantile(df$pctasian,0.75) - quantile(df$pctasian, 0.
length(which(df$pctasian > sevout_pctasian))
```

```
## [1] 156
```

```
df$f.pctasian <- ifelse(df$pctasian <= 0.2582, 1,
                        ifelse(df$pctasian > 0.2582 & df$pctasian <= 0.5495, 2,
                               ifelse(df$pctasian > 0.5495 & df$pctasian <= 1.2515, 3,
```

```
                                              ifelse(df$pctasian > 1.2515, 4,0))))
df$f.pctasian <- factor(df$f.pctasian,
labels=c("Lowpctasian","LowMidpctasian","HighMidpctasian","Highpctasian"),
                           order = T,levels=c(1,2,3,4))
table(df$f.pctasian)
```

```
##
##     Lowpctasian  LowMidpctasian HighMidpctasian    Highpctasian
##             458             457             458             458
```

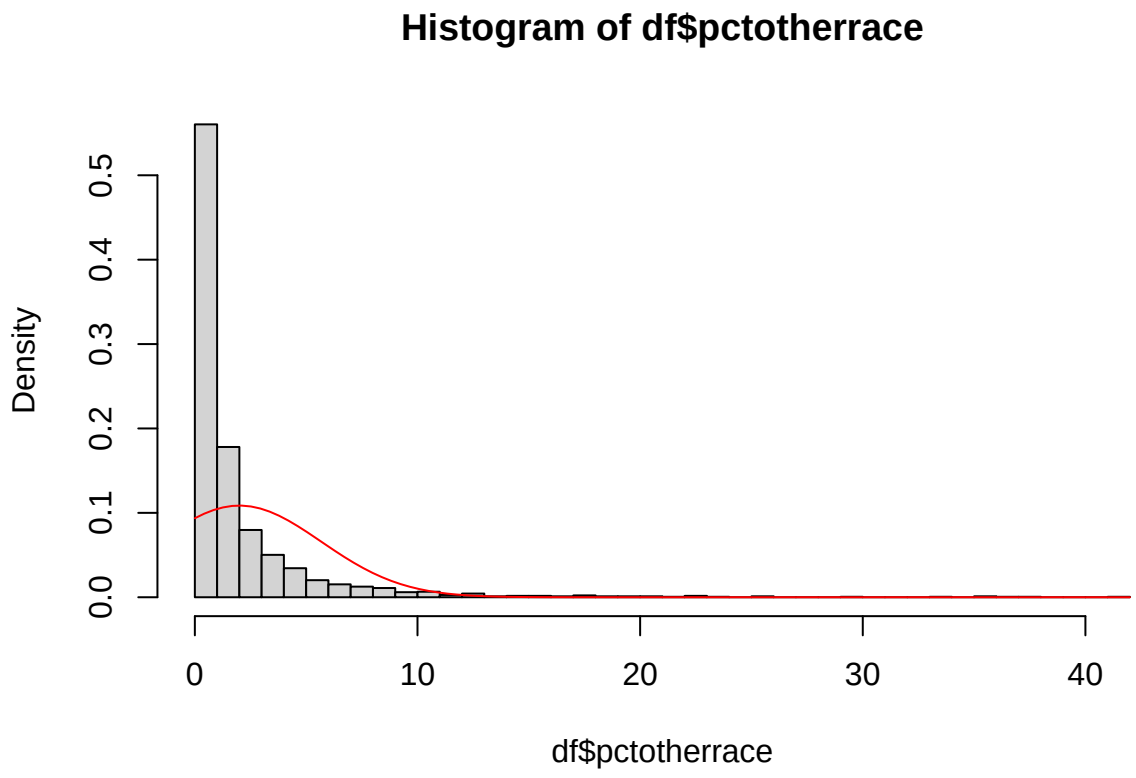
Variable 31: pctotherrace

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 181 outliers (of which 148 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctotherrace" to create a discretization according to quartiles.

```
summary(df$pctotherrace)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.2867  0.7826  2.0031  2.1066 41.9303
```

```
hist(df$pctotherrace, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctotherrace), sd(df$pctotherrace)), add = T, col = "red")
```



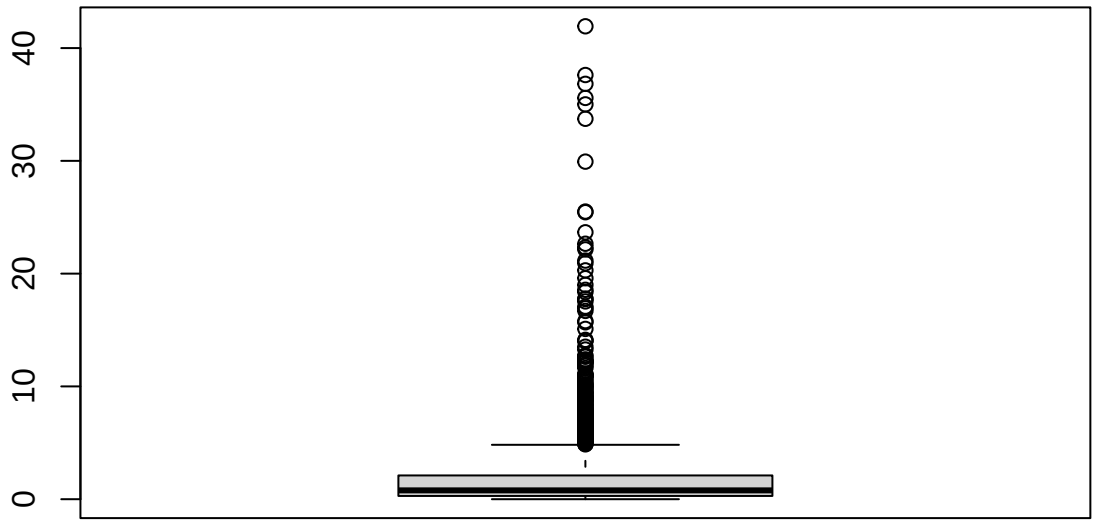
```
shapiro.test(df$pctotherrace)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctotherrace
## W = 0.50981, p-value < 2.2e-16
```

```
sum(is.na(df$pctotherrace))
```

```
## [1] 0
```

```
boxplot(df$pctotherrace)
```




```
bp<-boxplot(df$pctotherrace, id = list(n=Inf))
length(bp$out)
```

```
## [1] 181
```

```
sevout_pctotherrace = (quantile(df$pctotherrace,0.25)+(3*((quantile(df$pctotherrace,0.75) - quantile(df$
length(which(df$pctotherrace > sevout_pctotherrace))
```

```
## [1] 148
```

```
df$f.pctotherrace <- ifelse(df$pctotherrace <= 0.2867, 1,
                  ifelse(df$pctotherrace > 0.2867 & df$pctotherrace <= 0.7826, 2,
                  ifelse(df$pctotherrace > 0.7826 & df$pctotherrace <= 2.1066, 3,
                     ifelse(df$pctotherrace > 2.1066, 4,0))))
df$f.pctotherrace <- factor(df$f.pctotherrace,
    labels=c("Lowpctotherrace","LowMidpctotherrace","HighMidpctotherrace","Highpctotherrace"),
      order = T, levels=c(1,2,3,4))
table(df$f.pctotherrace)
```

```
##
##     Lowpctotherrace  LowMidpctotherrace HighMidpctotherrace    Highpctotherrace
##                 458                 458                 457                 458
```

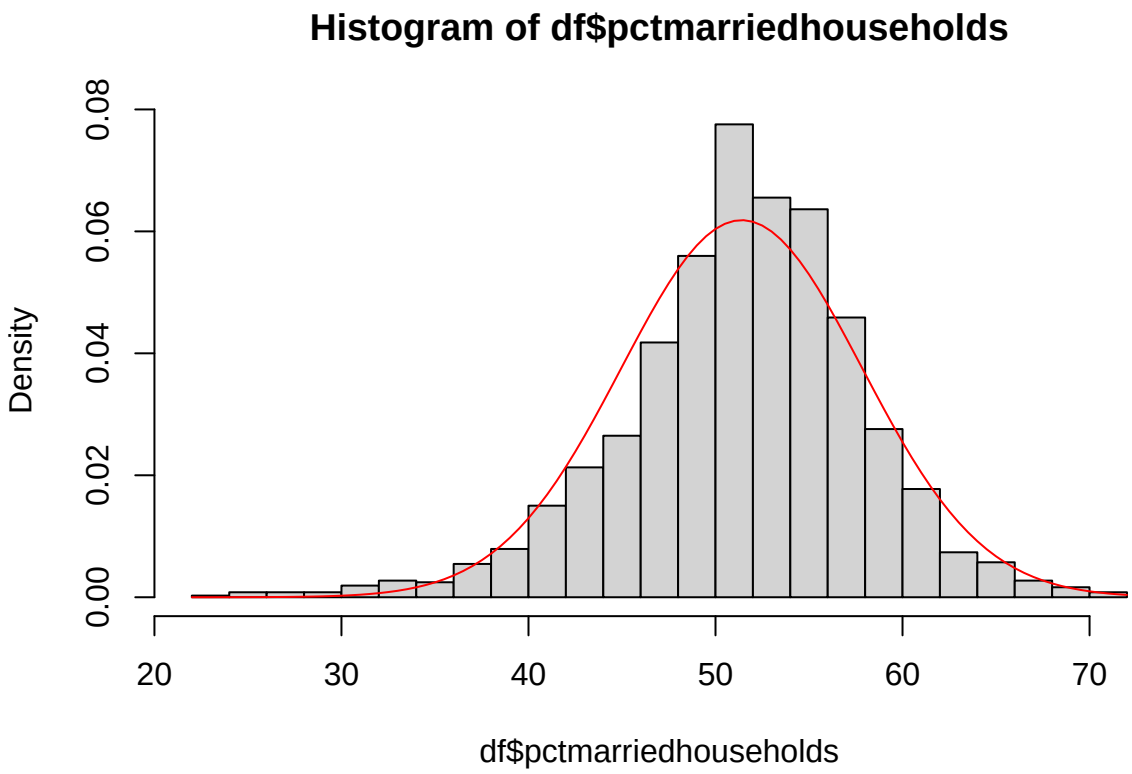
Variable 32:pctmarriedhouseholds———————————

This is a continuous proportion variable. The data do not appear normally distributed, which is confirmed by the near-zero p-value from the Shapiro normality test. A histogram is used to visualize the data. The variable contains no missing values, so imputation is not necessary. It contains 57 outliers (of which 2 are severe), all on the high side of the spectrum. We create an additional ordinal mpg factor "f.pctotherrace" to create a discretization according to quartiles.

```
summary(df$pctmarriedhouseholds)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   22.99   47.85   51.73   51.40   55.48   71.40
```

```
hist(df$pctmarriedhouseholds, breaks = 30, freq = F)
curve(dnorm(x, mean(df$pctmarriedhouseholds), sd(df$pctmarriedhouseholds)), add = T, col = "red")
```



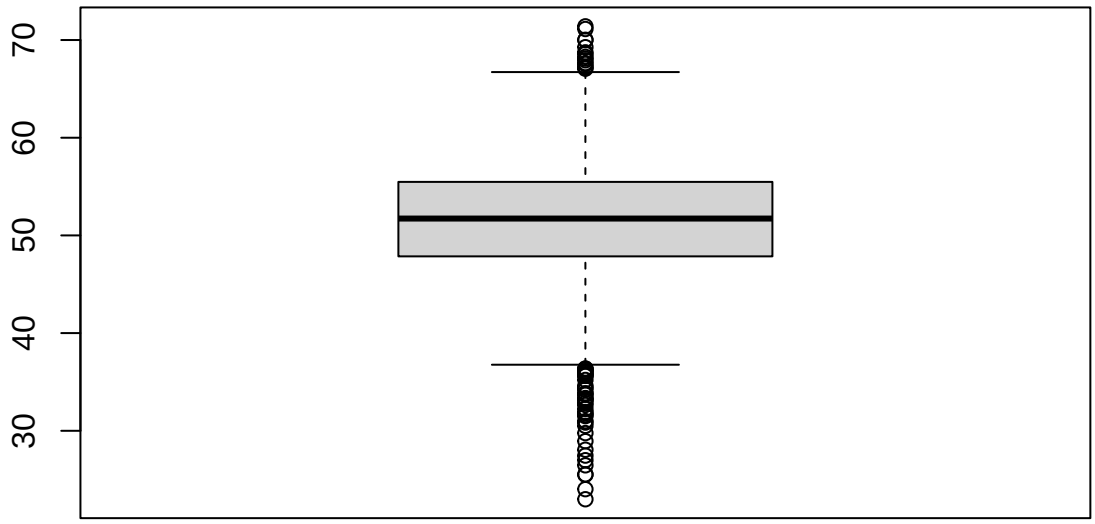

```
shapiro.test(df$pctmarriedhouseholds)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$pctmarriedhouseholds
## W = 0.9816, p-value = 1.341e-14
```

```
sum(is.na(df$pctmarriedhouseholds))
```

```
## [1] 0
```

```
boxplot(df$pctmarriedhouseholds)
```




```
bp<-boxplot(df$pctmarriedhouseholds, id = list(n=Inf))
length(bp$out)
```

```
## [1] 57
```

```
sevout_pctmarriedhouseholds = (quantile(df$pctmarriedhouseholds,0.25)+(3*((quantile(df$pctmarriedhouseh
length(which(df$pctmarriedhouseholds > sevout_pctmarriedhouseholds))
```

```
## [1] 2
```

```
df$f.pctmarriedhouseholds <- ifelse(df$pctmarriedhouseholds <= 47.85, 1,
                             ifelse(df$pctmarriedhouseholds > 47.85 & df$pctmarriedhouseholds <= 51.73
                                    ifelse(df$pctmarriedhouseholds > 51.73 & df$pctmarriedhouseholds <=
                                           ifelse(df$pctmarriedhouseholds > 55.48, 4,0))))
df$f.pctmarriedhouseholds<- factor(df$f.pctmarriedhouseholds,
 labels=c("Lowpctmarriedhouseholds","LowMidpctmarriedhouseholds","HighMidpctmarriedhouseholds","Highpctm
 order = T,     levels=c(1,2,3,4))
table(df$f.pctmarriedhouseholds)
```

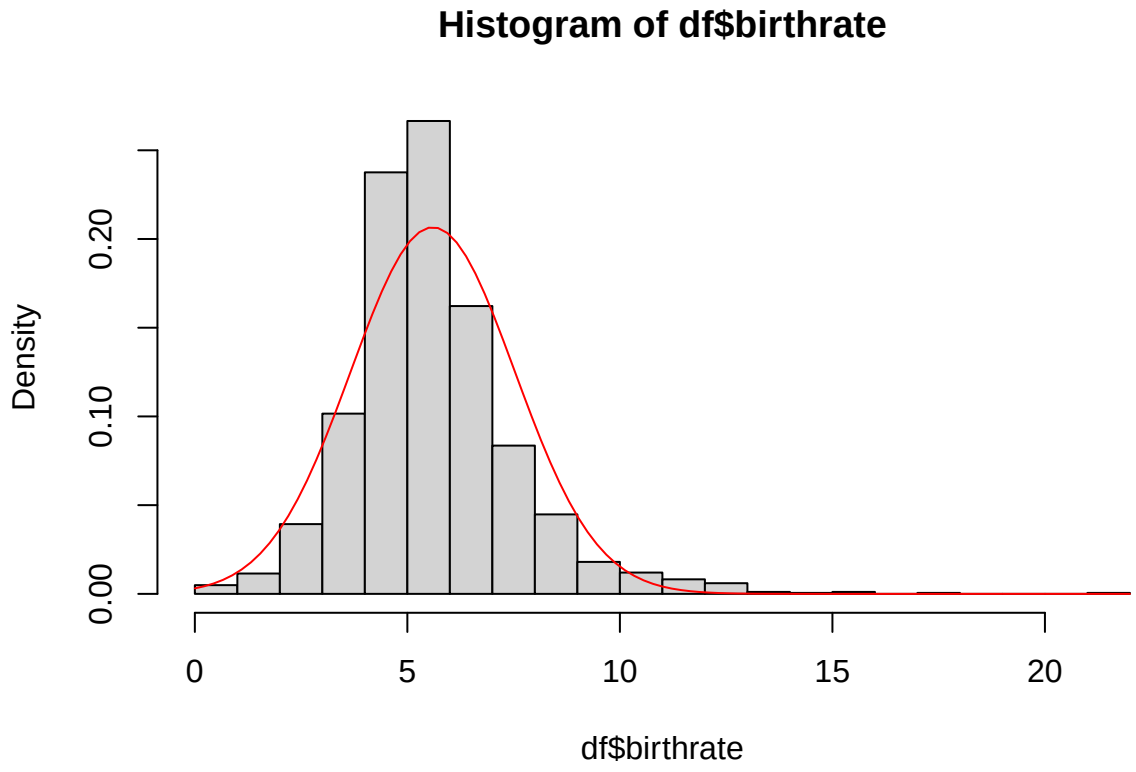
```
##
##     Lowpctmarriedhouseholds  LowMidpctmarriedhouseholds
##                         457                         460
## HighMidpctmarriedhouseholds    Highpctmarriedhouseholds
##                         456                         458
```

Variable 33: birthrate

```
summary(df$birthrate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   4.528   5.355   5.597   6.414  21.326
```

```
hist(df$birthrate, breaks = 30, freq = F)
curve(dnorm(x, mean(df$birthrate), sd(df$birthrate)), add = T, col = "red")
```



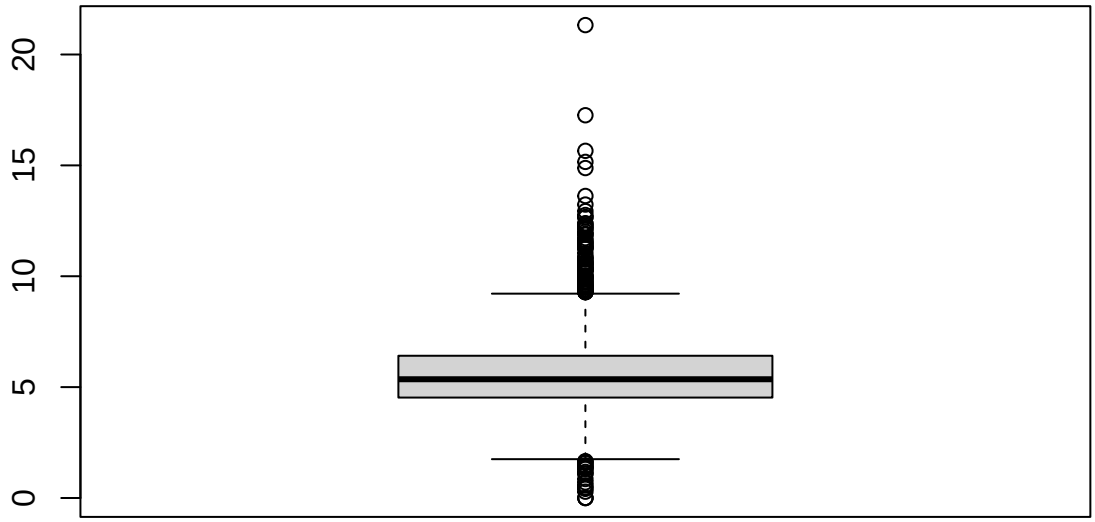

```
shapiro.test(df$birthrate)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$birthrate
## W = 0.93107, p-value < 2.2e-16
```

```
sum(is.na(df$birthrate))
```

```
## [1] 0
```

```
boxplot(df$birthrate)
```




```
bp<-boxplot(df$birthrate, id = list(n=Inf))
length(bp$out)
```

```
## [1] 104
```

```
sevout_birthrate = (quantile(df$birthrate,0.25)+(3*((quantile(df$birthrate,0.75) - quantile(df$birthrat
length(which(df$birthrate > sevout_birthrate))
```

```
## [1] 52
```

```
df$f.birthrate <- ifelse(df$birthrate <= 4.528, 1,
                         ifelse(df$birthrate > 4.528 & df$birthrate <= 5.355, 2,
                                ifelse(df$birthrate > 5.355 & df$birthrate <= 6.414, 3,
                                       ifelse(df$birthrate > 6.414, 4,0))))
df$f.birthrate <- factor(df$f.birthrate,
                         labels=c("Lowbirthrate","LowMidbirthrate","HighMidbirthrate","Highbirthra
                         order = T,
                         levels=c(1,2,3,4))
table(df$f.birthrate)
```

```
##
##     Lowbirthrate  LowMidbirthrate HighMidbirthrate    Highbirthrate
##              458              458              456              459
```