Data 621 - Homework#3

Joseph E. Garcia

March 22, 2019

```
library("pastecs")

## Warning: package 'pastecs' was built under R version 3.5.3

library("funModeling")

## Warning: package 'funModeling' was built under R version 3.5.3

## Loading required package: Hmisc

## Warning: package 'Hmisc' was built under R version 3.5.3

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

## Warning: package 'Formula' was built under R version 3.5.2

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.5.3

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':
##
##     format.pval, units

## funModeling v.1.7 :)
## Examples and tutorials at livebook.datascienceheroes.com

library("ggplot2")
library("corrgram")

## Warning: package 'corrgram' was built under R version 3.5.3

##
## Attaching package: 'corrgram'

## The following object is masked from 'package:lattice':
##
##     panel.fill
```

```r
library("leaps")
```

```
## Warning: package 'leaps' was built under R version 3.5.3
```

```r
trgData<-read.csv("https://raw.githubusercontent.com/jgarcia71/Data-621-Homeworks-Spring-2019/master/Homework%233/crime-training-data_modified.csv")
str(trgData)
```

```
## 'data.frame':    466 obs. of  13 variables:
##  $ zn     : num  0 0 0 30 0 0 0 0 0 80 ...
##  $ indus  : num  19.58 19.58 18.1 4.93 2.46 ...
##  $ chas   : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ nox    : num  0.605 0.871 0.74 0.428 0.488 0.52 0.693 0.693 0.515 0.392
...
##  $ rm     : num  7.93 5.4 6.49 6.39 7.16 ...
##  $ age    : num  96.2 100 100 7.8 92.2 71.3 100 100 38.1 19.1 ...
##  $ dis    : num  2.05 1.32 1.98 7.04 2.7 ...
##  $ rad    : int  5 5 24 6 3 5 24 24 5 1 ...
##  $ tax    : int  403 403 666 300 193 384 666 666 224 315 ...
##  $ ptratio: num  14.7 14.7 20.2 16.6 17.8 20.9 20.2 20.2 20.2 16.4 ...
##  $ lstat  : num  3.7 26.82 18.85 5.19 4.82 ...
##  $ medv   : num  50 13.4 15.4 23.7 37.9 26.5 5 7 22.2 20.9 ...
##  $ target : int  1 1 1 0 0 0 1 1 0 0 ...
```

```r
names(trgData)
```

```
##  [1] "zn"      "indus"   "chas"    "nox"     "rm"      "age"     "dis"
##  [8] "rad"     "tax"     "ptratio" "lstat"   "medv"    "target"
```

```r
head(trgData)
```

```
##   zn indus chas   nox    rm   age    dis rad tax ptratio lstat medv target
## 1  0 19.58    0 0.605 7.929  96.2 2.0459   5 403    14.7  3.70 50.0      1
## 2  0 19.58    1 0.871 5.403 100.0 1.3216   5 403    14.7 26.82 13.4      1
## 3  0 18.10    0 0.740 6.485 100.0 1.9784  24 666    20.2 18.85 15.4      1
## 4 30  4.93    0 0.428 6.393   7.8 7.0355   6 300    16.6  5.19 23.7      0
## 5  0  2.46    0 0.488 7.155  92.2 2.7006   3 193    17.8  4.82 37.9      0
## 6  0  8.56    0 0.520 6.781  71.3 2.8561   5 384    20.9  7.67 26.5      0
```

```r
summary(trgData)
```

```
##        zn             indus             chas              nox
##  Min.   :  0.00   Min.   : 0.460   Min.   :0.00000   Min.   :0.3890
##  1st Qu.:  0.00   1st Qu.: 5.145   1st Qu.:0.00000   1st Qu.:0.4480
##  Median :  0.00   Median : 9.690   Median :0.00000   Median :0.5380
##  Mean   : 11.58   Mean   :11.105   Mean   :0.07082   Mean   :0.5543
##  3rd Qu.: 16.25   3rd Qu.:18.100   3rd Qu.:0.00000   3rd Qu.:0.6240
##  Max.   :100.00   Max.   :27.740   Max.   :1.00000   Max.   :0.8710
##        rm              age              dis              rad
##  Min.   :3.863   Min.   :  2.90   Min.   : 1.130   Min.   : 1.00
##  1st Qu.:5.887   1st Qu.: 43.88   1st Qu.: 2.101   1st Qu.: 4.00
##  Median :6.210   Median : 77.15   Median : 3.191   Median : 5.00
```

```
##  Mean   :6.291   Mean   : 68.37   Mean   : 3.796   Mean   : 9.53
##  3rd Qu.:6.630   3rd Qu.: 94.10   3rd Qu.: 5.215   3rd Qu.:24.00
##  Max.   :8.780   Max.   :100.00   Max.   :12.127   Max.   :24.00
##       tax           ptratio         lstat             medv
##  Min.   :187.0   Min.   :12.6   Min.   : 1.730   Min.   : 5.00
##  1st Qu.:281.0   1st Qu.:16.9   1st Qu.: 7.043   1st Qu.:17.02
##  Median :334.5   Median :18.9   Median :11.350   Median :21.20
##  Mean   :409.5   Mean   :18.4   Mean   :12.631   Mean   :22.59
##  3rd Qu.:666.0   3rd Qu.:20.2   3rd Qu.:16.930   3rd Qu.:25.00
##  Max.   :711.0   Max.   :22.0   Max.   :37.970   Max.   :50.00
##      target
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.4914
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

```
stat.desc(trgData)
```

```
##                      zn        indus         chas          nox
## nbr.val       466.000000  466.0000000 466.00000000 4.660000e+02
## nbr.null      339.000000    0.0000000 433.00000000 0.000000e+00
## nbr.na          0.000000    0.0000000   0.00000000 0.000000e+00
## min             0.000000    0.4600000   0.00000000 3.890000e-01
## max           100.000000   27.7400000   1.00000000 8.710000e-01
## range         100.000000   27.2800000   1.00000000 4.820000e-01
## sum          5395.000000 5174.9400000  33.00000000 2.583087e+02
## median          0.000000    9.6900000   0.00000000 5.380000e-01
## mean           11.577253   11.1050215   0.07081545 5.543105e-01
## SE.mean         1.082347    0.3171281   0.01189566 5.404479e-03
## CI.mean.0.95    2.126896    0.6231817   0.02337591 1.062023e-02
## var           545.906922   46.8657296   0.06594213 1.361111e-02
## std.dev        23.364651    6.8458549   0.25679200 1.166667e-01
## coef.var        2.018152    0.6164648   3.62621425 2.104717e-01
##                      rm          age          dis          rad
## nbr.val      4.660000e+02 4.660000e+02 4.660000e+02  466.0000000
## nbr.null     0.000000e+00 0.000000e+00 0.000000e+00    0.0000000
## nbr.na       0.000000e+00 0.000000e+00 0.000000e+00    0.0000000
## min          3.863000e+00 2.900000e+00 1.129600e+00    1.0000000
## max          8.780000e+00 1.000000e+02 1.212650e+01   24.0000000
## range        4.917000e+00 9.710000e+01 1.099690e+01   23.0000000
## sum          2.931454e+03 3.185930e+04 1.768793e+03 4441.0000000
## median       6.210000e+00 7.715000e+01 3.190950e+00    5.0000000
## mean         6.290674e+00 6.836760e+01 3.795693e+00    9.5300429
## SE.mean      3.265161e-02 1.311963e+00 9.760255e-02    0.4023678
## CI.mean.0.95 6.416298e-02 2.578110e+00 1.917967e-01    0.7906844
## var          4.968153e-01 8.021005e+02 4.439236e+00   75.4453320
## std.dev      7.048513e-01 2.832138e+01 2.106950e+00    8.6859272
## coef.var     1.120470e-01 4.142515e-01 5.550896e-01    0.9114258
```

```
##                      tax      ptratio        lstat         medv
## nbr.val      4.660000e+02  466.0000000  466.0000000 4.660000e+02
## nbr.null     0.000000e+00    0.0000000    0.0000000 0.000000e+00
## nbr.na       0.000000e+00    0.0000000    0.0000000 0.000000e+00
## min          1.870000e+02   12.6000000    1.7300000 5.000000e+00
## max          7.110000e+02   22.0000000   37.9700000 5.000000e+01
## range        5.240000e+02    9.4000000   36.2400000 4.500000e+01
## sum          1.908280e+05 8573.7000000 5886.2600000 1.052660e+04
## median       3.345000e+02   18.9000000   11.3500000 2.120000e+01
## mean         4.095021e+02   18.3984979   12.6314592 2.258927e+01
## SE.mean      7.777821e+00    0.1017669    0.3289887 4.280200e-01
## CI.mean.0.95 1.528403e+01    0.1999799    0.6464888 8.410929e-01
## var          2.819044e+04    4.8261268   50.4368512 8.537171e+01
## std.dev      1.679001e+02    2.1968447    7.1018907 9.239681e+00
## coef.var     4.100103e-01    0.1194035    0.5622383 4.090297e-01
##                    target
## nbr.val      466.00000000
## nbr.null     237.00000000
## nbr.na         0.00000000
## min            0.00000000
## max            1.00000000
## range          1.00000000
## sum          229.00000000
## median         0.00000000
## mean           0.49141631
## SE.mean        0.02318353
## CI.mean.0.95   0.04555746
## var            0.25046380
## std.dev        0.50046358
## coef.var       1.01841061
```

```
trgData[!complete.cases(trgData),]
```

```
##  [1] zn      indus   chas    nox     rm      age     dis     rad
##  [9] tax     ptratio lstat   medv    target
## <0 rows> (or 0-length row.names)
```

```
print(paste0("Rows of Training Data: ", nrow(trgData)))
```

```
## [1] "Rows of Training Data: 466"
```

```
print(paste0("Columns of Training Data: ", ncol(trgData)))
```

```
## [1] "Columns of Training Data: 13"
```

```
cor(trgData)
```

```
##                  zn       indus        chas         nox          rm
## zn       1.00000000 -0.53826643 -0.04016203 -0.51704518  0.31981410
## indus   -0.53826643  1.00000000  0.06118317  0.75963008 -0.39271181
## chas    -0.04016203  0.06118317  1.00000000  0.09745577  0.09050979
## nox     -0.51704518  0.75963008  0.09745577  1.00000000 -0.29548972
```

```
## rm       0.31981410 -0.39271181  0.09050979 -0.29548972  1.00000000
## age     -0.57258054  0.63958182  0.07888366  0.73512782 -0.23281251
## dis      0.66012434 -0.70361886 -0.09657711 -0.76888404  0.19901584
## rad     -0.31548119  0.60062839 -0.01590037  0.59582984 -0.20844570
## tax     -0.31928408  0.73222922 -0.04676476  0.65387804 -0.29693430
## ptratio -0.39103573  0.39468980 -0.12866058  0.17626871 -0.36034706
## lstat   -0.43299252  0.60711023 -0.05142322  0.59624264 -0.63202445
## medv     0.37671713 -0.49617432  0.16156528 -0.43012267  0.70533679
## target  -0.43168176  0.60485074  0.08004187  0.72610622 -0.15255334
##                 age         dis         rad         tax    ptratio
## zn      -0.57258054  0.66012434 -0.31548119 -0.31928408 -0.3910357
## indus    0.63958182 -0.70361886  0.60062839  0.73222922  0.3946898
## chas     0.07888366 -0.09657711 -0.01590037 -0.04676476 -0.1286606
## nox      0.73512782 -0.76888404  0.59582984  0.65387804  0.1762687
## rm      -0.23281251  0.19901584 -0.20844570 -0.29693430 -0.3603471
## age      1.00000000 -0.75089759  0.46031430  0.51212452  0.2554479
## dis     -0.75089759  1.00000000 -0.49499193 -0.53425464 -0.2333394
## rad      0.46031430 -0.49499193  1.00000000  0.90646323  0.4714516
## tax      0.51212452 -0.53425464  0.90646323  1.00000000  0.4744223
## ptratio  0.25544785 -0.23333940  0.47145160  0.47442229  1.0000000
## lstat    0.60562001 -0.50752800  0.50310125  0.56418864  0.3773560
## medv    -0.37815605  0.25669476 -0.39766826 -0.49003287 -0.5159153
## target   0.63010625 -0.61867312  0.62810492  0.61111331  0.2508489
##               lstat       medv      target
## zn      -0.43299252  0.3767171 -0.43168176
## indus    0.60711023 -0.4961743  0.60485074
## chas    -0.05142322  0.1615653  0.08004187
## nox      0.59624264 -0.4301227  0.72610622
## rm      -0.63202445  0.7053368 -0.15255334
## age      0.60562001 -0.3781560  0.63010625
## dis     -0.50752800  0.2566948 -0.61867312
## rad      0.50310125 -0.3976683  0.62810492
## tax      0.56418864 -0.4900329  0.61111331
## ptratio  0.37735605 -0.5159153  0.25084892
## lstat    1.00000000 -0.7358008  0.46912702
## medv    -0.73580078  1.0000000 -0.27055071
## target   0.46912702 -0.2705507  1.00000000
```

```
evalData<-read.csv("https://raw.githubusercontent.com/jgarcia71/Data-621-Homeworks-Spring-2019/master/Homework%233/crime-evaluation-data_modified.csv")
str(evalData)
```

```
## 'data.frame':    40 obs. of  12 variables:
##  $ zn     : int  0 0 0 0 0 25 25 0 0 0 ...
##  $ indus  : num  7.07 8.14 8.14 8.14 5.96 5.13 5.13 4.49 4.49 2.89 ...
##  $ chas   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ nox    : num  0.469 0.538 0.538 0.538 0.499 0.453 0.453 0.449 0.449 0.4
45 ...
##  $ rm     : num  7.18 6.1 6.5 5.95 5.85 ...
##  $ age    : num  61.1 84.5 94.4 82 41.5 66.2 93.4 56.1 56.8 69.6 ...
```

```
##  $ dis    : num  4.97 4.46 4.45 3.99 3.93 ...
##  $ rad    : int  2 4 4 4 5 8 8 3 3 2 ...
##  $ tax    : int  242 307 307 307 279 284 284 247 247 276 ...
##  $ ptratio: num  17.8 21 21 21 19.2 19.7 19.7 18.5 18.5 18 ...
##  $ lstat  : num  4.03 10.26 12.8 27.71 8.77 ...
##  $ medv   : num  34.7 18.2 18.4 13.2 21 18.7 16 26.6 22.2 21.4 ...
```

```
names(evalData)
```

```
##  [1] "zn"      "indus"   "chas"    "nox"     "rm"      "age"     "dis"
##  [8] "rad"     "tax"     "ptratio" "lstat"   "medv"
```

```
head(evalData)
```

```
##   zn indus chas   nox    rm  age    dis rad tax ptratio lstat medv
## 1  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8  4.03 34.7
## 2  0  8.14    0 0.538 6.096 84.5 4.4619   4 307    21.0 10.26 18.2
## 3  0  8.14    0 0.538 6.495 94.4 4.4547   4 307    21.0 12.80 18.4
## 4  0  8.14    0 0.538 5.950 82.0 3.9900   4 307    21.0 27.71 13.2
## 5  0  5.96    0 0.499 5.850 41.5 3.9342   5 279    19.2  8.77 21.0
## 6 25  5.13    0 0.453 5.741 66.2 7.2254   8 284    19.7 13.15 18.7
```

```
summary(evalData)
```

```
##        zn              indus             chas           nox
##  Min.   : 0.000   Min.   : 1.760   Min.   :0.00   Min.   :0.3850
##  1st Qu.: 0.000   1st Qu.: 5.692   1st Qu.:0.00   1st Qu.:0.4713
##  Median : 0.000   Median : 8.915   Median :0.00   Median :0.5380
##  Mean   : 8.875   Mean   :11.507   Mean   :0.05   Mean   :0.5592
##  3rd Qu.: 0.000   3rd Qu.:18.100   3rd Qu.:0.00   3rd Qu.:0.6258
##  Max.   :90.000   Max.   :25.650   Max.   :1.00   Max.   :0.7400
##        rm             age              dis              rad
##  Min.   :3.561   Min.   :  6.80   Min.   :1.202   Min.   : 1.000
##  1st Qu.:5.874   1st Qu.: 56.62   1st Qu.:2.041   1st Qu.: 4.000
##  Median :6.143   Median : 83.25   Median :3.373   Median : 5.000
##  Mean   :6.214   Mean   : 70.99   Mean   :3.787   Mean   : 9.775
##  3rd Qu.:6.532   3rd Qu.: 93.10   3rd Qu.:4.527   3rd Qu.:24.000
##  Max.   :8.247   Max.   :100.00   Max.   :9.089   Max.   :24.000
##       tax           ptratio          lstat            medv
##  Min.   :188.0   Min.   :14.70   Min.   : 2.960   Min.   : 8.40
##  1st Qu.:276.8   1st Qu.:18.40   1st Qu.: 6.435   1st Qu.:16.98
##  Median :307.0   Median :19.60   Median :11.685   Median :20.55
##  Mean   :393.5   Mean   :19.12   Mean   :12.905   Mean   :21.88
##  3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:17.363   3rd Qu.:25.00
##  Max.   :666.0   Max.   :21.20   Max.   :34.020   Max.   :50.00
```

```
cor(evalData)
```

```
##                  zn       indus         chas          nox          rm
## zn       1.00000000 -0.48057259 -0.089779946 -0.510818344  0.20519793
## indus   -0.48057259  1.00000000  0.092806250  0.818299097 -0.37711090
## chas    -0.08977995  0.09280625  1.000000000  0.001782619  0.09343143
```

```
## nox     -0.51081834  0.81829910  0.001782619  1.000000000 -0.38958806
## rm       0.20519793 -0.37711090  0.093431432 -0.389588062  1.00000000
## age     -0.52600877  0.71140151  0.210595065  0.680367121 -0.33163913
## dis      0.72008117 -0.75963647 -0.136879071 -0.776897267  0.28269128
## rad     -0.27042091  0.53424758  0.108318624  0.808963479 -0.22537840
## tax     -0.27434132  0.60482449  0.102194823  0.854357512 -0.24839167
## ptratio -0.39878767  0.22951739  0.039913790  0.409490882 -0.26797458
## lstat   -0.18898716  0.56840155 -0.084467265  0.532946414 -0.40914604
## medv     0.14719168 -0.33268395  0.370228083 -0.387602615  0.56339442
##                age        dis        rad        tax     ptratio
## zn      -0.5260088  0.7200812 -0.2704209 -0.2743413 -0.39878767
## indus    0.7114015 -0.7596365  0.5342476  0.6048245  0.22951739
## chas     0.2105951 -0.1368791  0.1083186  0.1021948  0.03991379
## nox      0.6803671 -0.7768973  0.8089635  0.8543575  0.40949088
## rm      -0.3316391  0.2826913 -0.2253784 -0.2483917 -0.26797458
## age      1.0000000 -0.7144471  0.4056180  0.4542307  0.35216431
## dis     -0.7144471  1.0000000 -0.4903915 -0.5395048 -0.24324824
## rad      0.4056180 -0.4903915  1.0000000  0.9571809  0.40704345
## tax      0.4542307 -0.5395048  0.9571809  1.0000000  0.35236368
## ptratio  0.3521643 -0.2432482  0.4070434  0.3523637  1.00000000
## lstat    0.5693032 -0.3827549  0.3359428  0.3385416  0.35952375
## medv    -0.3560393  0.1662015 -0.1892231 -0.2255919 -0.38990222
##              lstat       medv
## zn      -0.18898716  0.1471917
## indus    0.56840155 -0.3326840
## chas    -0.08446727  0.3702281
## nox      0.53294641 -0.3876026
## rm      -0.40914604  0.5633944
## age      0.56930318 -0.3560393
## dis     -0.38275492  0.1662015
## rad      0.33594277 -0.1892231
## tax      0.33854156 -0.2255919
## ptratio  0.35952375 -0.3899022
## lstat    1.00000000 -0.7648272
## medv    -0.76482715  1.0000000
```

```
stat.desc(trgData)
```

```
##                        zn        indus         chas          nox
## nbr.val       466.000000  466.0000000 466.00000000 4.660000e+02
## nbr.null      339.000000    0.0000000 433.00000000 0.000000e+00
## nbr.na          0.000000    0.0000000   0.00000000 0.000000e+00
## min             0.000000    0.4600000   0.00000000 3.890000e-01
## max           100.000000   27.7400000   1.00000000 8.710000e-01
## range         100.000000   27.2800000   1.00000000 4.820000e-01
## sum          5395.000000 5174.9400000  33.00000000 2.583087e+02
## median          0.000000    9.6900000   0.00000000 5.380000e-01
## mean           11.577253   11.1050215   0.07081545 5.543105e-01
## SE.mean         1.082347    0.3171281   0.01189566 5.404479e-03
## CI.mean.0.95    2.126896    0.6231817   0.02337591 1.062023e-02
```

```
## var           545.906922   46.8657296   0.06594213 1.361111e-02
## std.dev        23.364651    6.8458549   0.25679200 1.166667e-01
## coef.var        2.018152    0.6164648   3.62621425 2.104717e-01
##                      rm          age          dis          rad
## nbr.val      4.660000e+02 4.660000e+02 4.660000e+02  466.0000000
## nbr.null     0.000000e+00 0.000000e+00 0.000000e+00    0.0000000
## nbr.na       0.000000e+00 0.000000e+00 0.000000e+00    0.0000000
## min          3.863000e+00 2.900000e+00 1.129600e+00    1.0000000
## max          8.780000e+00 1.000000e+02 1.212650e+01   24.0000000
## range        4.917000e+00 9.710000e+01 1.099690e+01   23.0000000
## sum          2.931454e+03 3.185930e+04 1.768793e+03 4441.0000000
## median       6.210000e+00 7.715000e+01 3.190950e+00    5.0000000
## mean         6.290674e+00 6.836760e+01 3.795693e+00    9.5300429
## SE.mean      3.265161e-02 1.311963e+00 9.760255e-02    0.4023678
## CI.mean.0.95 6.416298e-02 2.578110e+00 1.917967e-01    0.7906844
## var          4.968153e-01 8.021005e+02 4.439236e+00   75.4453320
## std.dev      7.048513e-01 2.832138e+01 2.106950e+00    8.6859272
## coef.var     1.120470e-01 4.142515e-01 5.550896e-01    0.9114258
##                     tax      ptratio        lstat         medv
## nbr.val      4.660000e+02  466.0000000  466.0000000 4.660000e+02
## nbr.null     0.000000e+00    0.0000000    0.0000000 0.000000e+00
## nbr.na       0.000000e+00    0.0000000    0.0000000 0.000000e+00
## min          1.870000e+02   12.6000000    1.7300000 5.000000e+00
## max          7.110000e+02   22.0000000   37.9700000 5.000000e+01
## range        5.240000e+02    9.4000000   36.2400000 4.500000e+01
## sum          1.908280e+05 8573.7000000 5886.2600000 1.052660e+04
## median       3.345000e+02   18.9000000   11.3500000 2.120000e+01
## mean         4.095021e+02   18.3984979   12.6314592 2.258927e+01
## SE.mean      7.777821e+00    0.1017669    0.3289887 4.280200e-01
## CI.mean.0.95 1.528403e+01    0.1999799    0.6464888 8.410929e-01
## var          2.819044e+04    4.8261268   50.4368512 8.537171e+01
## std.dev      1.679001e+02    2.1968447    7.1018907 9.239681e+00
## coef.var     4.100103e-01    0.1194035    0.5622383 4.090297e-01
##                    target
## nbr.val      466.00000000
## nbr.null     237.00000000
## nbr.na         0.00000000
## min            0.00000000
## max            1.00000000
## range          1.00000000
## sum          229.00000000
## median         0.00000000
## mean           0.49141631
## SE.mean        0.02318353
## CI.mean.0.95   0.04555746
## var            0.25046380
## std.dev        0.50046358
## coef.var       1.01841061

print(paste("Rows of Evaluation Data: ", nrow(evalData)))
```

```
## [1] "Rows of Evaluation Data:  40"
```

```
print(paste("Columns of Evaluation Data: ", ncol(evalData)))
```

```
## [1] "Columns of Evaluation Data:  12"
```

```
df_status(trgData)
```

```
##    variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1        zn     339   72.75    0    0     0     0 numeric     26
## 2     indus       0    0.00    0    0     0     0 numeric     73
## 3      chas     433   92.92    0    0     0     0 integer      2
## 4       nox       0    0.00    0    0     0     0 numeric     79
## 5        rm       0    0.00    0    0     0     0 numeric    419
## 6       age       0    0.00    0    0     0     0 numeric    333
## 7       dis       0    0.00    0    0     0     0 numeric    380
## 8       rad       0    0.00    0    0     0     0 integer      9
## 9       tax       0    0.00    0    0     0     0 integer     63
## 10  ptratio       0    0.00    0    0     0     0 numeric     46
## 11    lstat       0    0.00    0    0     0     0 numeric    424
## 12     medv       0    0.00    0    0     0     0 numeric    218
## 13   target     237   50.86    0    0     0     0 integer      2
```

```
df_status(evalData)
```

```
##    variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1        zn      33    82.5    0    0     0     0 integer      6
## 2     indus       0     0.0    0    0     0     0 numeric     22
## 3      chas      38    95.0    0    0     0     0 integer      2
## 4       nox       0     0.0    0    0     0     0 numeric     28
## 5        rm       0     0.0    0    0     0     0 numeric     40
## 6       age       0     0.0    0    0     0     0 numeric     39
## 7       dis       0     0.0    0    0     0     0 numeric     40
## 8       rad       0     0.0    0    0     0     0 integer      9
## 9       tax       0     0.0    0    0     0     0 integer     21
## 10  ptratio       0     0.0    0    0     0     0 numeric     17
## 11    lstat       0     0.0    0    0     0     0 numeric     40
## 12     medv       0     0.0    0    0     0     0 numeric     37
```

```
apply(trgData, 2, function(x) any(is.na(x)))
```

```
##      zn   indus    chas     nox      rm     age     dis     rad     tax
##   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE
## ptratio   lstat    medv  target
##   FALSE   FALSE   FALSE   FALSE
```

```
apply(evalData, 2, function(x) any(is.na(x)))
```

```
##      zn   indus    chas     nox      rm     age     dis     rad     tax
##   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE   FALSE
## ptratio   lstat    medv
##   FALSE   FALSE   FALSE
```

```
bucket.zn<-trgData[,'zn']
summary(bucket.zn)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    0.00    0.00    0.00   11.58   16.25  100.00
```

```
bucket.zn
```

```
##   [1]   0.0   0.0   0.0  30.0   0.0   0.0   0.0   0.0   0.0  80.0  22.0
##  [12]   0.0   0.0  22.0   0.0   0.0 100.0  20.0   0.0   0.0   0.0   0.0
##  [23]   0.0  18.0   0.0  60.0   0.0   0.0  25.0  25.0   0.0   0.0   0.0
##  [34]   0.0   0.0  80.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0  80.0
##  [45]  12.5   0.0   0.0   0.0   0.0   0.0  55.0  12.5   0.0   0.0   0.0
##  [56]   0.0  20.0  20.0   0.0   0.0   0.0   0.0   0.0  45.0  35.0   0.0
##  [67]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
##  [78]   0.0   0.0   0.0   0.0   0.0   0.0  80.0  20.0  30.0   0.0   0.0
##  [89]   0.0  21.0  25.0  70.0   0.0   0.0  45.0   0.0  40.0   0.0   0.0
## [100]   0.0   0.0   0.0   0.0  75.0   0.0   0.0   0.0  55.0   0.0   0.0
## [111]   0.0  25.0   0.0  52.5   0.0  82.5   0.0   0.0   0.0   0.0   0.0
## [122]   0.0   0.0   0.0  20.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
## [133]  25.0   0.0  90.0   0.0  12.5   0.0   0.0  30.0   0.0   0.0   0.0
## [144]   0.0   0.0   0.0   0.0   0.0   0.0  12.5   0.0  34.0   0.0   0.0
## [155]   0.0  40.0   0.0  82.5   0.0   0.0   0.0   0.0  20.0  80.0   0.0
## [166]  22.0  52.5   0.0   0.0   0.0  75.0   0.0   0.0  28.0  20.0   0.0
## [177]   0.0   0.0   0.0  20.0   0.0  30.0   0.0  90.0   0.0   0.0  90.0
## [188]   0.0   0.0   0.0  55.0   0.0  20.0   0.0   0.0   0.0   0.0   0.0
## [199]   0.0  20.0  80.0   0.0   0.0   0.0  25.0   0.0  70.0  20.0  22.0
## [210]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0  20.0   0.0  12.5
## [221]  20.0   0.0  35.0  20.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
## [232]   0.0   0.0   0.0   0.0  95.0  80.0  34.0   0.0   0.0   0.0   0.0
## [243]   0.0   0.0  40.0   0.0   0.0   0.0   0.0   0.0  20.0   0.0   0.0
## [254]  95.0  20.0   0.0   0.0   0.0  12.5   0.0  40.0  22.0   0.0   0.0
## [265]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
## [276]   0.0   0.0   0.0   0.0  22.0  60.0  12.5   0.0   0.0   0.0   0.0
## [287]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
## [298]   0.0  40.0   0.0  40.0   0.0   0.0   0.0   0.0  33.0   0.0   0.0
## [309]  30.0  20.0   0.0   0.0   0.0   0.0   0.0   0.0  40.0   0.0   0.0
## [320]   0.0   0.0  20.0   0.0  30.0   0.0   0.0  75.0  80.0   0.0  22.0
## [331]   0.0  12.5   0.0  35.0   0.0   0.0  80.0  20.0   0.0   0.0   0.0
## [342]   0.0  80.0   0.0  21.0   0.0   0.0   0.0  21.0  20.0  95.0  12.5
## [353]   0.0   0.0   0.0  33.0   0.0  20.0   0.0  28.0   0.0  85.0   0.0
## [364]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0  21.0   0.0  22.0
## [375]   0.0   0.0   0.0  52.5   0.0  34.0   0.0   0.0   0.0   0.0   0.0
## [386]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0  90.0
## [397]   0.0   0.0   0.0   0.0   0.0  45.0   0.0   0.0   0.0   0.0  45.0
## [408]  85.0   0.0   0.0   0.0   0.0  22.0   0.0   0.0  70.0  28.0   0.0
## [419]   0.0   0.0  60.0   0.0   0.0   0.0   0.0  25.0   0.0   0.0   0.0
## [430]   0.0  17.5  80.0   0.0   0.0  45.0   0.0  45.0   0.0  95.0   0.0
## [441]   0.0   0.0   0.0  80.0   0.0  80.0   0.0  60.0   0.0   0.0   0.0
```

```
## [452]   0.0  33.0   0.0   0.0  25.0   0.0  12.5   0.0   0.0   0.0   0.0
## [463]   0.0   0.0   0.0   0.0
```

```
bucket.indus<-trgData[,'indus']
summary(bucket.indus)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.460   5.145   9.690  11.105  18.100  27.740
```

```
bucket.indus
```

```
##   [1] 19.58 19.58 18.10  4.93  2.46  8.56 18.10 18.10  5.19  3.64  5.86
##  [12] 12.83 18.10  5.86  2.46  2.18  1.32  3.97 18.10 18.10  3.24  6.20
##  [23]  2.89  2.31  9.90  2.93  5.19 18.10  4.86  5.13  6.20  8.56  2.89
##  [34] 18.10  5.19  4.95  2.46 18.10  4.39 19.58  3.24 18.10  4.05  1.91
##  [45]  7.87  6.91 18.10  9.90 18.10  8.14  2.25  7.87  5.96  1.89 21.89
##  [56] 10.59  3.33  3.33  7.07 18.10 19.58  9.90 18.10  3.44  6.06  8.14
##  [67] 27.74 18.10  9.69  6.20 18.10  8.56 19.58 13.89 18.10 10.81 19.58
##  [78] 13.89  2.18  7.38 10.01 18.10 18.10  4.95  3.97  4.93 19.58 18.10
##  [89] 10.01  5.64  4.86  2.24 18.10  7.38  3.44 18.10  1.25 18.10 18.10
## [100]  5.19  8.14 10.59 19.58  4.00 18.10 18.10 11.93  3.78 21.89  3.24
## [111]  6.20  5.13  5.19  5.32  6.91  2.03 18.10  9.90 21.89  6.20  8.14
## [122]  6.91  6.20 18.10  3.97 10.01 18.10  5.19  8.56 13.92 18.10 18.10
## [133]  4.86 18.10  1.21  2.89  7.87  8.14 21.89  4.93  6.91  5.96 19.58
## [144] 21.89  9.69  6.20 21.89  6.20 18.10  6.07 18.10  6.09 10.59 27.74
## [155] 19.58  6.41 18.10  2.03  8.56 19.58 18.10  6.20  3.33  3.37  4.05
## [166]  5.86  5.32 18.10 18.10 18.10  2.95 19.58  8.14 15.04  3.97 18.10
## [177] 18.10 12.83 18.10  3.97  3.41  4.93 18.10  1.22  2.46  8.56  3.75
## [188] 18.10 18.10 18.10  3.78  7.38  6.96 18.10 19.58  6.91 11.93 18.10
## [199] 18.10  3.97  3.37 19.58 18.10 21.89  5.13  8.14  2.24  6.96  5.86
## [210] 13.89 18.10  9.69 21.89 25.65 18.10  5.19 18.10  3.97 18.10  6.07
## [221]  3.97  9.90  1.52  3.97  6.20 13.92 27.74  6.20  8.14 18.10 18.10
## [232] 18.10  7.38 18.10 13.89  1.47  1.91  6.09  4.05 21.89 18.10  9.90
## [243] 19.58 18.10  6.41 25.65 18.10 19.58  8.56 10.81  3.33  2.46 18.10
## [254]  1.47  3.97 18.10 25.65  2.89  6.07 19.58  6.41  5.86  9.69 10.01
## [265] 19.58  3.41 19.58 19.58 18.10 11.93 18.10 18.10 21.89  4.05 25.65
## [276] 18.10 18.10  8.14 18.10  5.86  1.69  7.87 21.89 18.10 10.81 18.10
## [287]  9.90 10.01 18.10 18.10 19.58  8.14  6.20  4.49 27.74 18.10  7.38
## [298] 18.10  6.41  6.91  6.41 10.59 10.59  9.69  9.90  2.18 21.89 18.10
## [309]  4.93  6.96 10.59 18.10  4.05  4.49 21.89 10.01  1.25  2.46  7.38
## [320] 18.10  6.20  3.97 18.10  4.93  9.90 13.92  2.95  1.52  8.14  5.86
## [331] 18.10  7.87  8.14  6.06 18.10 18.10  3.64  6.96 12.83  9.90 19.58
## [342]  2.46  0.46 18.10  5.64  8.14 18.10  2.18  5.64  6.96  2.68  7.87
## [353] 25.65 10.59  8.56  2.18 10.01  3.97 18.10 15.04  6.91  4.15  8.56
## [364]  3.41 18.10 10.59  9.69 18.10 21.89 10.01 18.10  5.64 18.10  5.86
## [375] 18.10 18.10 18.10  5.32 18.10  6.09 10.59  6.20 12.83 18.10 18.10
## [386] 18.10 18.10 27.74  4.39  6.20 10.81  4.05 18.10  6.91 18.10  2.02
## [397] 13.92 18.10 18.10 19.58 18.10  3.44 18.10 12.83 18.10  4.05  3.44
## [408]  0.74 18.10  8.56  9.90  6.91  5.86 11.93  8.56  2.24 15.04 10.01
## [419] 19.58  2.46  1.69 19.58 19.58  8.14  8.14  5.13 19.58 19.58 18.10
## [430] 18.10  1.38  1.52  9.69  5.96  3.44  8.14  3.44 18.10  2.68  8.14
```

```
## [441] 13.92 18.10 18.10  1.52  3.41  4.95 18.10  2.93 18.10 18.10 18.10
## [452]  8.14  2.18 18.10  8.14  4.86 10.59  7.87 18.10  6.20 18.10 18.10
## [463] 18.10 18.10 12.83 18.10
```

```
bucket.zn.indus<-cbind(trgData$zn,trgData$indus)
summary(bucket.zn.indus)
```

```
##        V1               V2
##  Min.   :  0.00   Min.   : 0.460
##  1st Qu.:  0.00   1st Qu.: 5.145
##  Median :  0.00   Median : 9.690
##  Mean   : 11.58   Mean   :11.105
##  3rd Qu.: 16.25   3rd Qu.:18.100
##  Max.   :100.00   Max.   :27.740
```

```
bucket.zn.indus
```

```
##         [,1]  [,2]
##   [1,]   0.0 19.58
##   [2,]   0.0 19.58
##   [3,]   0.0 18.10
##   [4,]  30.0  4.93
##   [5,]   0.0  2.46
##   [6,]   0.0  8.56
##   [7,]   0.0 18.10
##   [8,]   0.0 18.10
##   [9,]   0.0  5.19
##  [10,]  80.0  3.64
##  [11,]  22.0  5.86
##  [12,]   0.0 12.83
##  [13,]   0.0 18.10
##  [14,]  22.0  5.86
##  [15,]   0.0  2.46
##  [16,]   0.0  2.18
##  [17,] 100.0  1.32
##  [18,]  20.0  3.97
##  [19,]   0.0 18.10
##  [20,]   0.0 18.10
##  [21,]   0.0  3.24
##  [22,]   0.0  6.20
##  [23,]   0.0  2.89
##  [24,]  18.0  2.31
##  [25,]   0.0  9.90
##  [26,]  60.0  2.93
##  [27,]   0.0  5.19
##  [28,]   0.0 18.10
##  [29,]  25.0  4.86
##  [30,]  25.0  5.13
##  [31,]   0.0  6.20
##  [32,]   0.0  8.56
##  [33,]   0.0  2.89
```

```
##  [34,]   0.0 18.10
##  [35,]   0.0  5.19
##  [36,]  80.0  4.95
##  [37,]   0.0  2.46
##  [38,]   0.0 18.10
##  [39,]   0.0  4.39
##  [40,]   0.0 19.58
##  [41,]   0.0  3.24
##  [42,]   0.0 18.10
##  [43,]   0.0  4.05
##  [44,]  80.0  1.91
##  [45,]  12.5  7.87
##  [46,]   0.0  6.91
##  [47,]   0.0 18.10
##  [48,]   0.0  9.90
##  [49,]   0.0 18.10
##  [50,]   0.0  8.14
##  [51,]  55.0  2.25
##  [52,]  12.5  7.87
##  [53,]   0.0  5.96
##  [54,]   0.0  1.89
##  [55,]   0.0 21.89
##  [56,]   0.0 10.59
##  [57,]  20.0  3.33
##  [58,]  20.0  3.33
##  [59,]   0.0  7.07
##  [60,]   0.0 18.10
##  [61,]   0.0 19.58
##  [62,]   0.0  9.90
##  [63,]   0.0 18.10
##  [64,]  45.0  3.44
##  [65,]  35.0  6.06
##  [66,]   0.0  8.14
##  [67,]   0.0 27.74
##  [68,]   0.0 18.10
##  [69,]   0.0  9.69
##  [70,]   0.0  6.20
##  [71,]   0.0 18.10
##  [72,]   0.0  8.56
##  [73,]   0.0 19.58
##  [74,]   0.0 13.89
##  [75,]   0.0 18.10
##  [76,]   0.0 10.81
##  [77,]   0.0 19.58
##  [78,]   0.0 13.89
##  [79,]   0.0  2.18
##  [80,]   0.0  7.38
##  [81,]   0.0 10.01
##  [82,]   0.0 18.10
##  [83,]   0.0 18.10
```

```
##  [84,]  80.0  4.95
##  [85,]  20.0  3.97
##  [86,]  30.0  4.93
##  [87,]   0.0 19.58
##  [88,]   0.0 18.10
##  [89,]   0.0 10.01
##  [90,]  21.0  5.64
##  [91,]  25.0  4.86
##  [92,]  70.0  2.24
##  [93,]   0.0 18.10
##  [94,]   0.0  7.38
##  [95,]  45.0  3.44
##  [96,]   0.0 18.10
##  [97,]  40.0  1.25
##  [98,]   0.0 18.10
##  [99,]   0.0 18.10
## [100,]   0.0  5.19
## [101,]   0.0  8.14
## [102,]   0.0 10.59
## [103,]   0.0 19.58
## [104,]  75.0  4.00
## [105,]   0.0 18.10
## [106,]   0.0 18.10
## [107,]   0.0 11.93
## [108,]  55.0  3.78
## [109,]   0.0 21.89
## [110,]   0.0  3.24
## [111,]   0.0  6.20
## [112,]  25.0  5.13
## [113,]   0.0  5.19
## [114,]  52.5  5.32
## [115,]   0.0  6.91
## [116,]  82.5  2.03
## [117,]   0.0 18.10
## [118,]   0.0  9.90
## [119,]   0.0 21.89
## [120,]   0.0  6.20
## [121,]   0.0  8.14
## [122,]   0.0  6.91
## [123,]   0.0  6.20
## [124,]   0.0 18.10
## [125,]  20.0  3.97
## [126,]   0.0 10.01
## [127,]   0.0 18.10
## [128,]   0.0  5.19
## [129,]   0.0  8.56
## [130,]   0.0 13.92
## [131,]   0.0 18.10
## [132,]   0.0 18.10
## [133,]  25.0  4.86
```

```
## [134,]   0.0 18.10
## [135,]  90.0  1.21
## [136,]   0.0  2.89
## [137,]  12.5  7.87
## [138,]   0.0  8.14
## [139,]   0.0 21.89
## [140,]  30.0  4.93
## [141,]   0.0  6.91
## [142,]   0.0  5.96
## [143,]   0.0 19.58
## [144,]   0.0 21.89
## [145,]   0.0  9.69
## [146,]   0.0  6.20
## [147,]   0.0 21.89
## [148,]   0.0  6.20
## [149,]   0.0 18.10
## [150,]  12.5  6.07
## [151,]   0.0 18.10
## [152,]  34.0  6.09
## [153,]   0.0 10.59
## [154,]   0.0 27.74
## [155,]   0.0 19.58
## [156,]  40.0  6.41
## [157,]   0.0 18.10
## [158,]  82.5  2.03
## [159,]   0.0  8.56
## [160,]   0.0 19.58
## [161,]   0.0 18.10
## [162,]   0.0  6.20
## [163,]  20.0  3.33
## [164,]  80.0  3.37
## [165,]   0.0  4.05
## [166,]  22.0  5.86
## [167,]  52.5  5.32
## [168,]   0.0 18.10
## [169,]   0.0 18.10
## [170,]   0.0 18.10
## [171,]  75.0  2.95
## [172,]   0.0 19.58
## [173,]   0.0  8.14
## [174,]  28.0 15.04
## [175,]  20.0  3.97
## [176,]   0.0 18.10
## [177,]   0.0 18.10
## [178,]   0.0 12.83
## [179,]   0.0 18.10
## [180,]  20.0  3.97
## [181,]   0.0  3.41
## [182,]  30.0  4.93
## [183,]   0.0 18.10
```

```
## [184,]  90.0  1.22
## [185,]   0.0  2.46
## [186,]   0.0  8.56
## [187,]  90.0  3.75
## [188,]   0.0 18.10
## [189,]   0.0 18.10
## [190,]   0.0 18.10
## [191,]  55.0  3.78
## [192,]   0.0  7.38
## [193,]  20.0  6.96
## [194,]   0.0 18.10
## [195,]   0.0 19.58
## [196,]   0.0  6.91
## [197,]   0.0 11.93
## [198,]   0.0 18.10
## [199,]   0.0 18.10
## [200,]  20.0  3.97
## [201,]  80.0  3.37
## [202,]   0.0 19.58
## [203,]   0.0 18.10
## [204,]   0.0 21.89
## [205,]  25.0  5.13
## [206,]   0.0  8.14
## [207,]  70.0  2.24
## [208,]  20.0  6.96
## [209,]  22.0  5.86
## [210,]   0.0 13.89
## [211,]   0.0 18.10
## [212,]   0.0  9.69
## [213,]   0.0 21.89
## [214,]   0.0 25.65
## [215,]   0.0 18.10
## [216,]   0.0  5.19
## [217,]   0.0 18.10
## [218,]  20.0  3.97
## [219,]   0.0 18.10
## [220,]  12.5  6.07
## [221,]  20.0  3.97
## [222,]   0.0  9.90
## [223,]  35.0  1.52
## [224,]  20.0  3.97
## [225,]   0.0  6.20
## [226,]   0.0 13.92
## [227,]   0.0 27.74
## [228,]   0.0  6.20
## [229,]   0.0  8.14
## [230,]   0.0 18.10
## [231,]   0.0 18.10
## [232,]   0.0 18.10
## [233,]   0.0  7.38
```

```
## [234,]   0.0 18.10
## [235,]   0.0 13.89
## [236,]  95.0  1.47
## [237,]  80.0  1.91
## [238,]  34.0  6.09
## [239,]   0.0  4.05
## [240,]   0.0 21.89
## [241,]   0.0 18.10
## [242,]   0.0  9.90
## [243,]   0.0 19.58
## [244,]   0.0 18.10
## [245,]  40.0  6.41
## [246,]   0.0 25.65
## [247,]   0.0 18.10
## [248,]   0.0 19.58
## [249,]   0.0  8.56
## [250,]   0.0 10.81
## [251,]  20.0  3.33
## [252,]   0.0  2.46
## [253,]   0.0 18.10
## [254,]  95.0  1.47
## [255,]  20.0  3.97
## [256,]   0.0 18.10
## [257,]   0.0 25.65
## [258,]   0.0  2.89
## [259,]  12.5  6.07
## [260,]   0.0 19.58
## [261,]  40.0  6.41
## [262,]  22.0  5.86
## [263,]   0.0  9.69
## [264,]   0.0 10.01
## [265,]   0.0 19.58
## [266,]   0.0  3.41
## [267,]   0.0 19.58
## [268,]   0.0 19.58
## [269,]   0.0 18.10
## [270,]   0.0 11.93
## [271,]   0.0 18.10
## [272,]   0.0 18.10
## [273,]   0.0 21.89
## [274,]   0.0  4.05
## [275,]   0.0 25.65
## [276,]   0.0 18.10
## [277,]   0.0 18.10
## [278,]   0.0  8.14
## [279,]   0.0 18.10
## [280,]  22.0  5.86
## [281,]  60.0  1.69
## [282,]  12.5  7.87
## [283,]   0.0 21.89
```

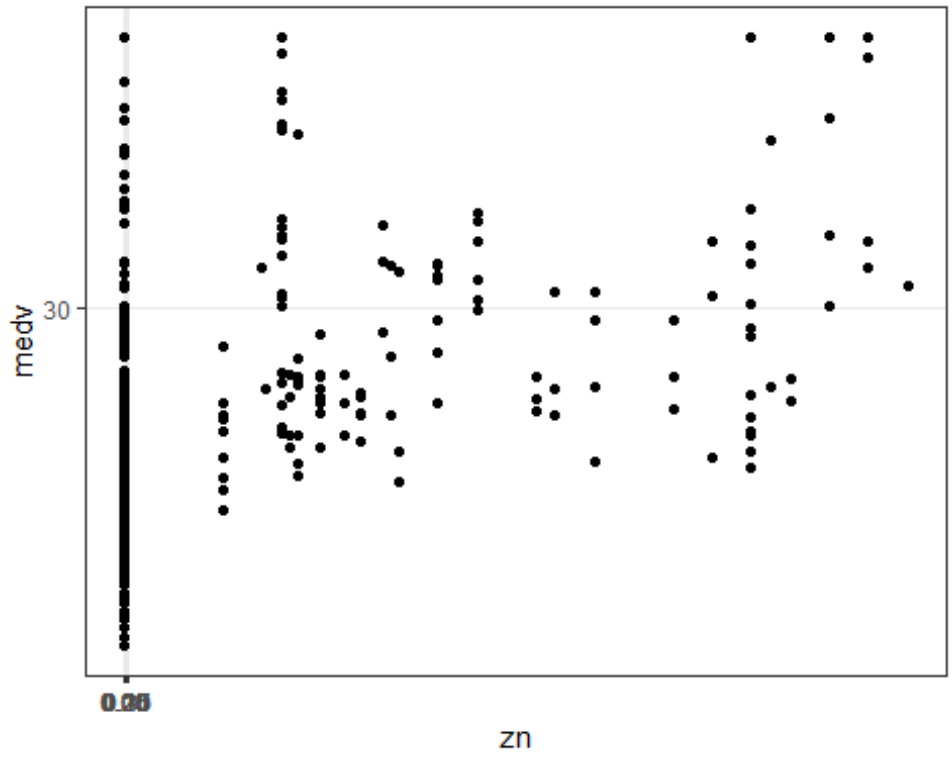
```
## [284,]   0.0 18.10
## [285,]   0.0 10.81
## [286,]   0.0 18.10
## [287,]   0.0  9.90
## [288,]   0.0 10.01
## [289,]   0.0 18.10
## [290,]   0.0 18.10
## [291,]   0.0 19.58
## [292,]   0.0  8.14
## [293,]   0.0  6.20
## [294,]   0.0  4.49
## [295,]   0.0 27.74
## [296,]   0.0 18.10
## [297,]   0.0  7.38
## [298,]   0.0 18.10
## [299,]  40.0  6.41
## [300,]   0.0  6.91
## [301,]  40.0  6.41
## [302,]   0.0 10.59
## [303,]   0.0 10.59
## [304,]   0.0  9.69
## [305,]   0.0  9.90
## [306,]  33.0  2.18
## [307,]   0.0 21.89
## [308,]   0.0 18.10
## [309,]  30.0  4.93
## [310,]  20.0  6.96
## [311,]   0.0 10.59
## [312,]   0.0 18.10
## [313,]   0.0  4.05
## [314,]   0.0  4.49
## [315,]   0.0 21.89
## [316,]   0.0 10.01
## [317,]  40.0  1.25
## [318,]   0.0  2.46
## [319,]   0.0  7.38
## [320,]   0.0 18.10
## [321,]   0.0  6.20
## [322,]  20.0  3.97
## [323,]   0.0 18.10
## [324,]  30.0  4.93
## [325,]   0.0  9.90
## [326,]   0.0 13.92
## [327,]  75.0  2.95
## [328,]  80.0  1.52
## [329,]   0.0  8.14
## [330,]  22.0  5.86
## [331,]   0.0 18.10
## [332,]  12.5  7.87
## [333,]   0.0  8.14
```

```
## [334,]  35.0  6.06
## [335,]   0.0 18.10
## [336,]   0.0 18.10
## [337,]  80.0  3.64
## [338,]  20.0  6.96
## [339,]   0.0 12.83
## [340,]   0.0  9.90
## [341,]   0.0 19.58
## [342,]   0.0  2.46
## [343,]  80.0  0.46
## [344,]   0.0 18.10
## [345,]  21.0  5.64
## [346,]   0.0  8.14
## [347,]   0.0 18.10
## [348,]   0.0  2.18
## [349,]  21.0  5.64
## [350,]  20.0  6.96
## [351,]  95.0  2.68
## [352,]  12.5  7.87
## [353,]   0.0 25.65
## [354,]   0.0 10.59
## [355,]   0.0  8.56
## [356,]  33.0  2.18
## [357,]   0.0 10.01
## [358,]  20.0  3.97
## [359,]   0.0 18.10
## [360,]  28.0 15.04
## [361,]   0.0  6.91
## [362,]  85.0  4.15
## [363,]   0.0  8.56
## [364,]   0.0  3.41
## [365,]   0.0 18.10
## [366,]   0.0 10.59
## [367,]   0.0  9.69
## [368,]   0.0 18.10
## [369,]   0.0 21.89
## [370,]   0.0 10.01
## [371,]   0.0 18.10
## [372,]  21.0  5.64
## [373,]   0.0 18.10
## [374,]  22.0  5.86
## [375,]   0.0 18.10
## [376,]   0.0 18.10
## [377,]   0.0 18.10
## [378,]  52.5  5.32
## [379,]   0.0 18.10
## [380,]  34.0  6.09
## [381,]   0.0 10.59
## [382,]   0.0  6.20
## [383,]   0.0 12.83
```

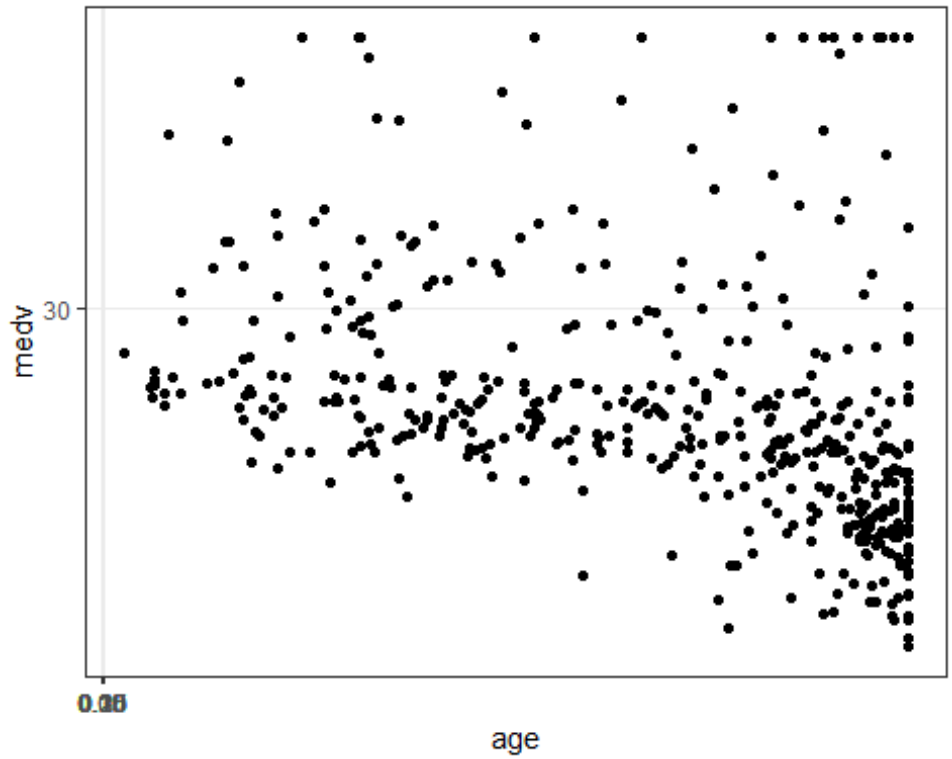
```
## [384,]   0.0 18.10
## [385,]   0.0 18.10
## [386,]   0.0 18.10
## [387,]   0.0 18.10
## [388,]   0.0 27.74
## [389,]   0.0  4.39
## [390,]   0.0  6.20
## [391,]   0.0 10.81
## [392,]   0.0  4.05
## [393,]   0.0 18.10
## [394,]   0.0  6.91
## [395,]   0.0 18.10
## [396,]  90.0  2.02
## [397,]   0.0 13.92
## [398,]   0.0 18.10
## [399,]   0.0 18.10
## [400,]   0.0 19.58
## [401,]   0.0 18.10
## [402,]  45.0  3.44
## [403,]   0.0 18.10
## [404,]   0.0 12.83
## [405,]   0.0 18.10
## [406,]   0.0  4.05
## [407,]  45.0  3.44
## [408,]  85.0  0.74
## [409,]   0.0 18.10
## [410,]   0.0  8.56
## [411,]   0.0  9.90
## [412,]   0.0  6.91
## [413,]  22.0  5.86
## [414,]   0.0 11.93
## [415,]   0.0  8.56
## [416,]  70.0  2.24
## [417,]  28.0 15.04
## [418,]   0.0 10.01
## [419,]   0.0 19.58
## [420,]   0.0  2.46
## [421,]  60.0  1.69
## [422,]   0.0 19.58
## [423,]   0.0 19.58
## [424,]   0.0  8.14
## [425,]   0.0  8.14
## [426,]  25.0  5.13
## [427,]   0.0 19.58
## [428,]   0.0 19.58
## [429,]   0.0 18.10
## [430,]   0.0 18.10
## [431,]  17.5  1.38
## [432,]  80.0  1.52
## [433,]   0.0  9.69
```

```
## [434,]   0.0  5.96
## [435,]  45.0  3.44
## [436,]   0.0  8.14
## [437,]  45.0  3.44
## [438,]   0.0 18.10
## [439,]  95.0  2.68
## [440,]   0.0  8.14
## [441,]   0.0 13.92
## [442,]   0.0 18.10
## [443,]   0.0 18.10
## [444,]  80.0  1.52
## [445,]   0.0  3.41
## [446,]  80.0  4.95
## [447,]   0.0 18.10
## [448,]  60.0  2.93
## [449,]   0.0 18.10
## [450,]   0.0 18.10
## [451,]   0.0 18.10
## [452,]   0.0  8.14
## [453,]  33.0  2.18
## [454,]   0.0 18.10
## [455,]   0.0  8.14
## [456,]  25.0  4.86
## [457,]   0.0 10.59
## [458,]  12.5  7.87
## [459,]   0.0 18.10
## [460,]   0.0  6.20
## [461,]   0.0 18.10
## [462,]   0.0 18.10
## [463,]   0.0 18.10
## [464,]   0.0 18.10
## [465,]   0.0 12.83
## [466,]   0.0 18.10
```

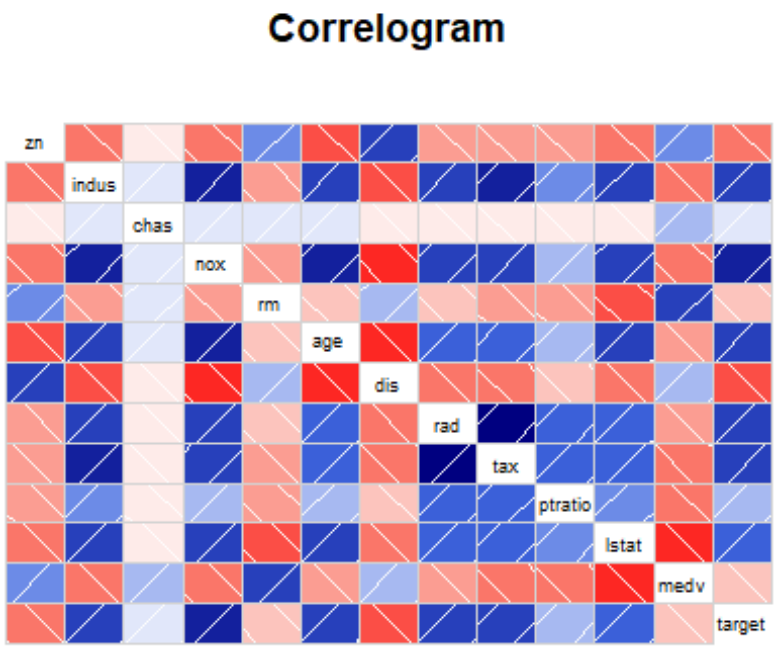
```
ggplot(trgData, aes(zn, medv)) + geom_point() + scale_x_continuous("zn", brea
ks = seq(0,0.35,0.05))+ scale_y_continuous("medv", breaks = seq(0,270,by = 30
))+ theme_bw()
```



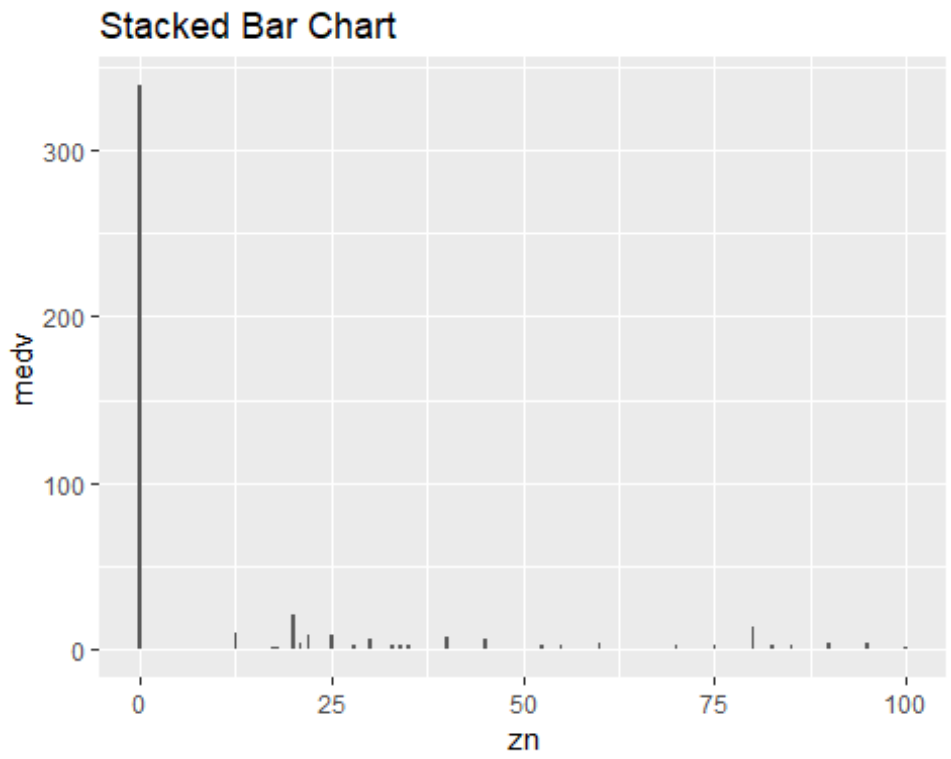

```
ggplot(trgData, aes(age, medv)) + geom_point() + scale_x_continuous("age", br
eaks = seq(0,0.35,0.05))+ scale_y_continuous("medv", breaks = seq(0,270,by =
30))+ theme_bw()
```



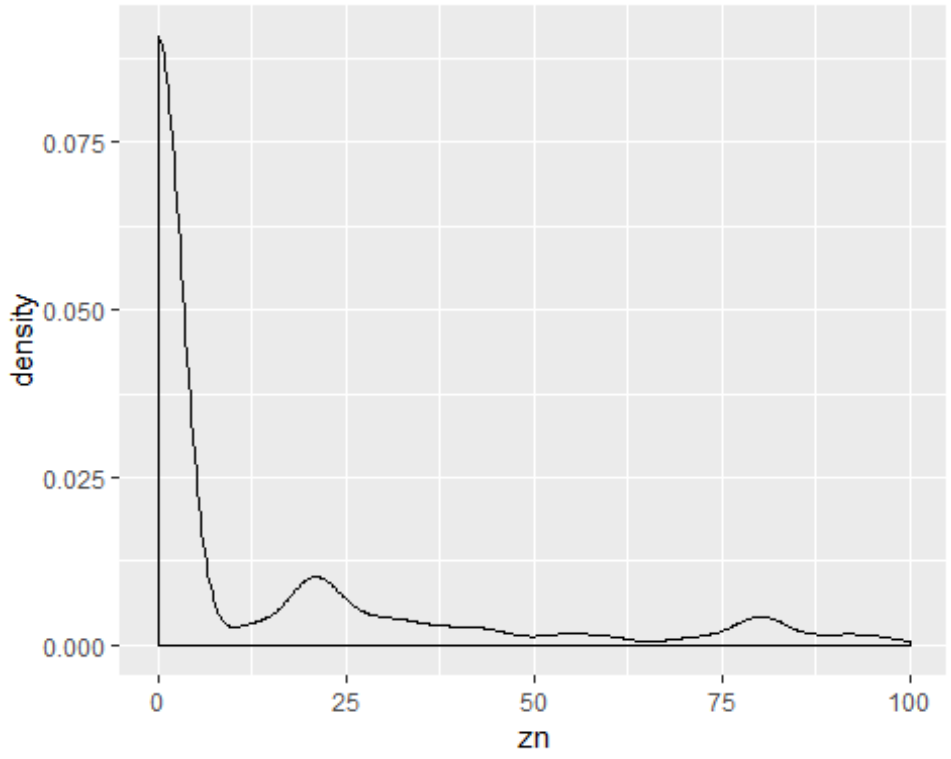
```
corrgram(trgData, order=NULL, panel=panel.shade, text.panel=panel.txt, main="
Correlogram")
```



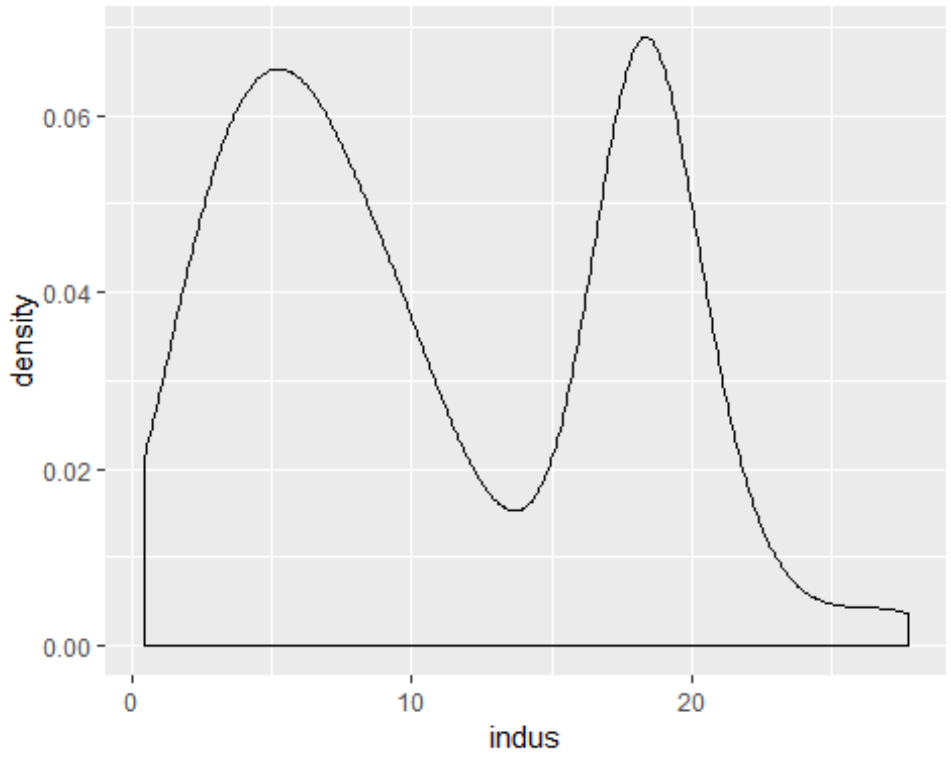

```
ggplot(trgData, aes(zn, fill =medv)) + geom_bar()+labs(title = "Stacked Bar C
hart", x = "zn", y = "medv")
```



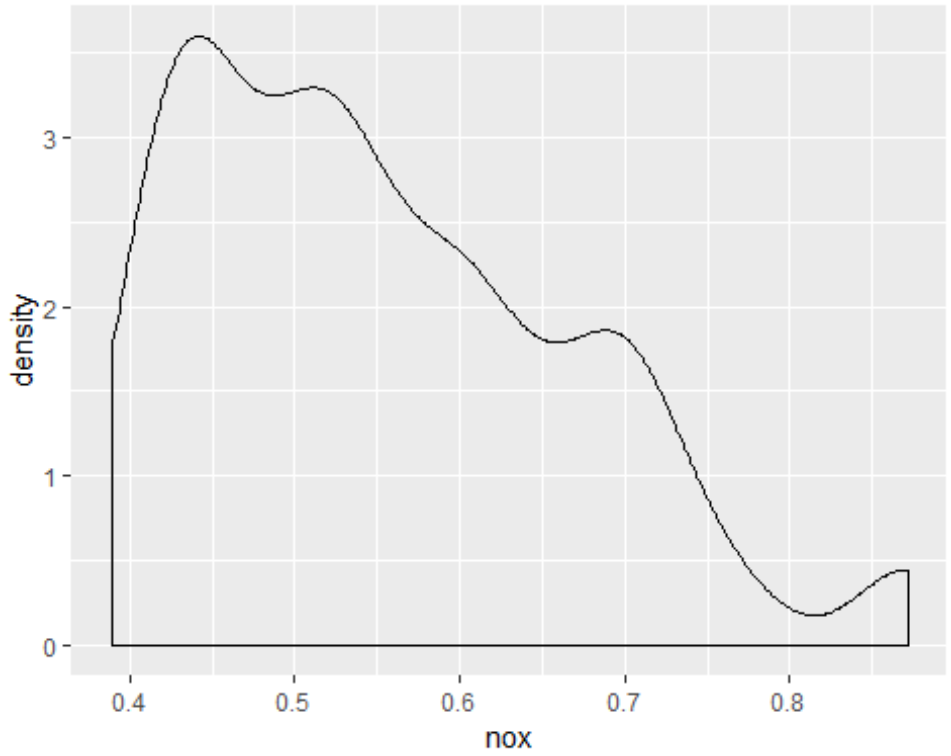

```
ggplot(trgData, aes(x = zn)) + geom_density() #right-skewed normal
```



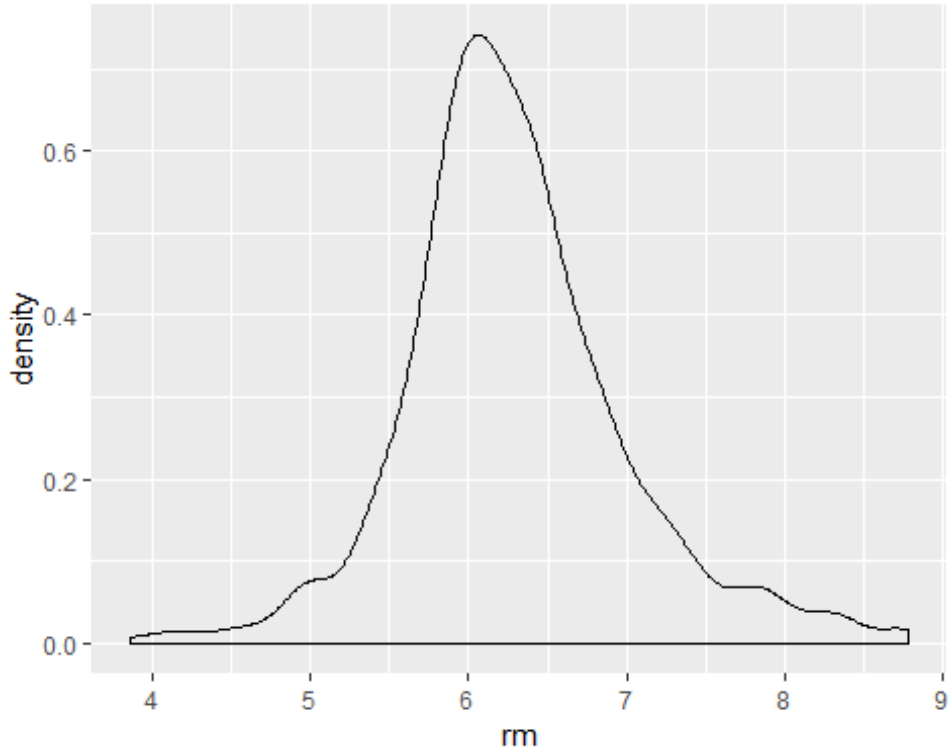

```
ggplot(trgData, aes(x = indus)) + geom_density() #NOT normal
```



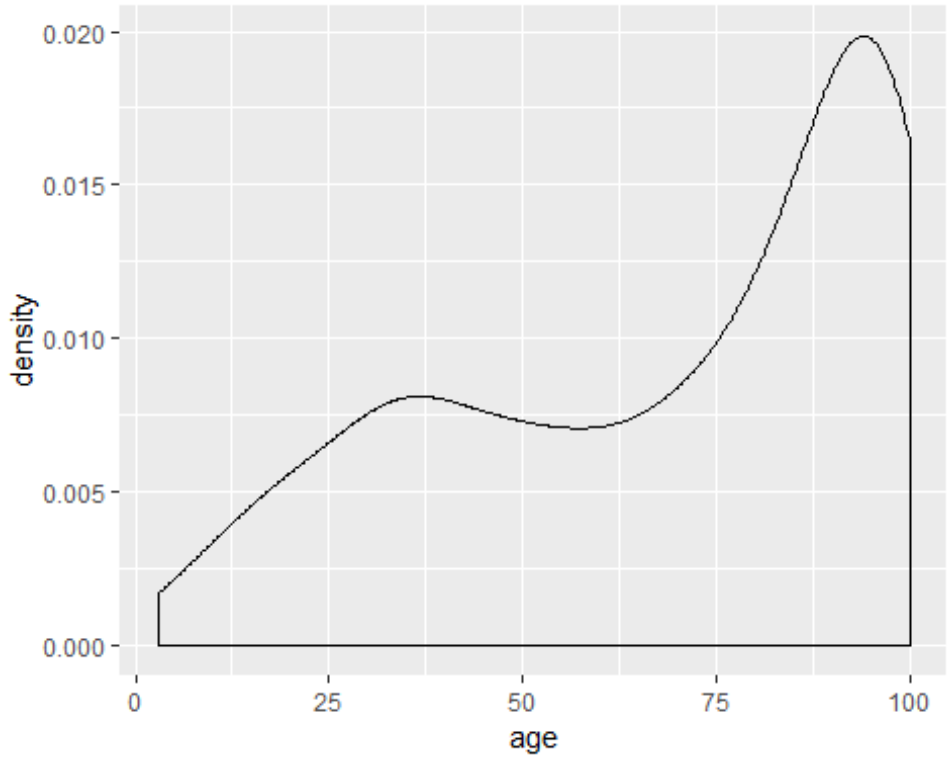

```r
ggplot(trgData, aes(x = nox)) + geom_density() #Poisson
```



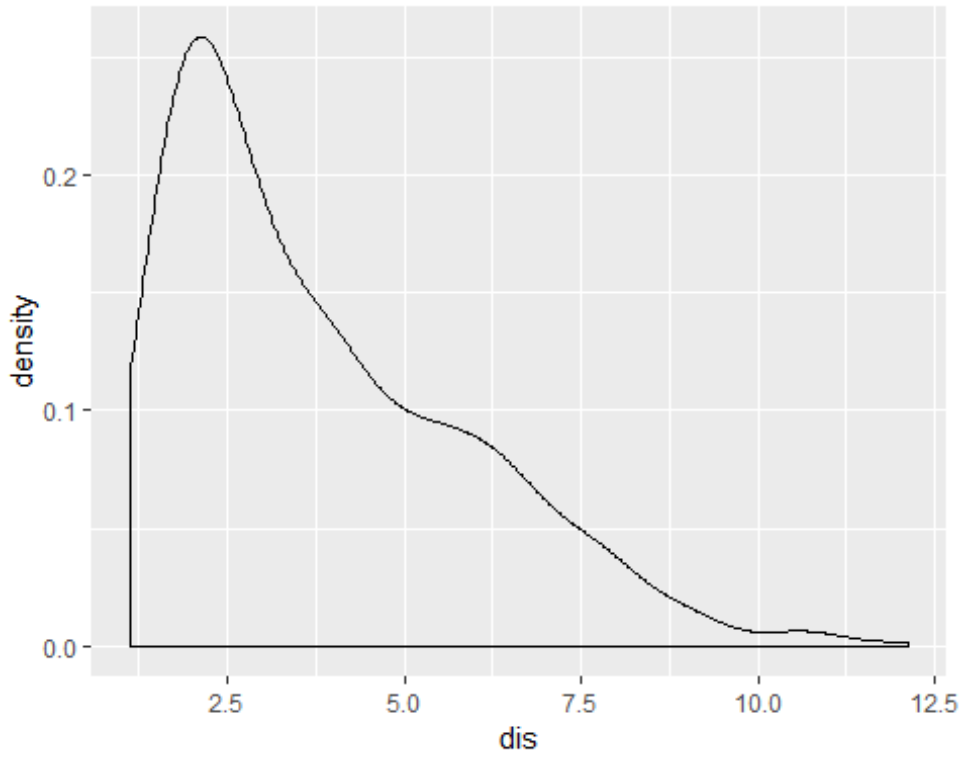

```r
ggplot(trgData, aes(x = rm)) + geom_density() #normal
```



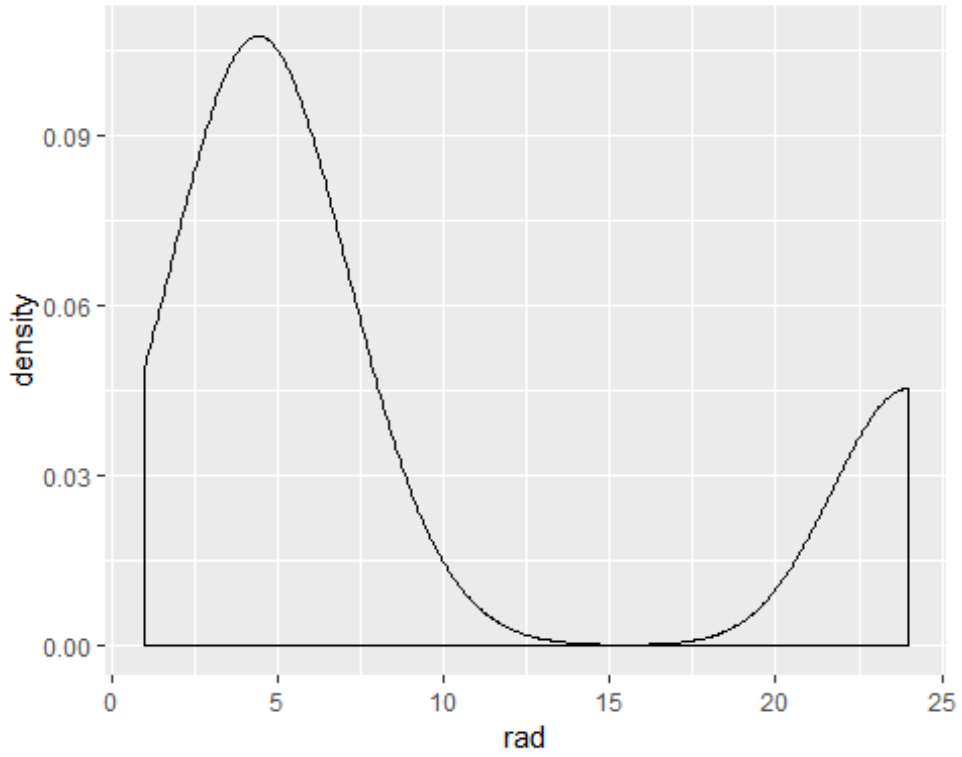

```r
ggplot(trgData, aes(x = age)) + geom_density() #some other distribution, seem
s wrong
```



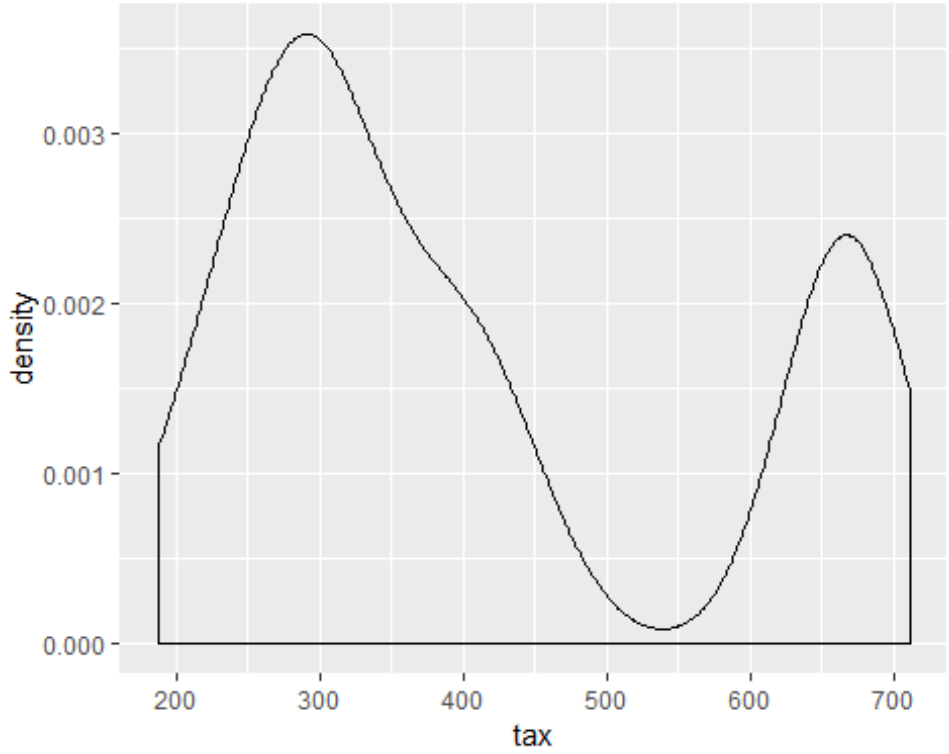
```r
ggplot(trgData, aes(x = dis)) + geom_density() #Poisson
```



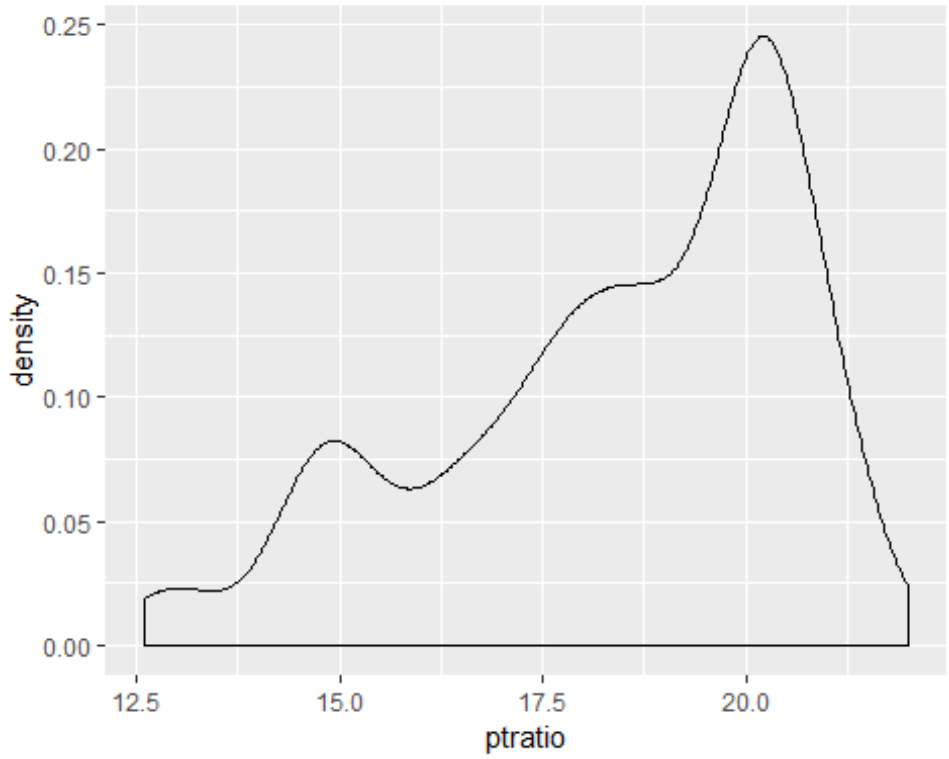

```r
ggplot(trgData, aes(x = rad)) + geom_density() #Nor normal - two humps
```



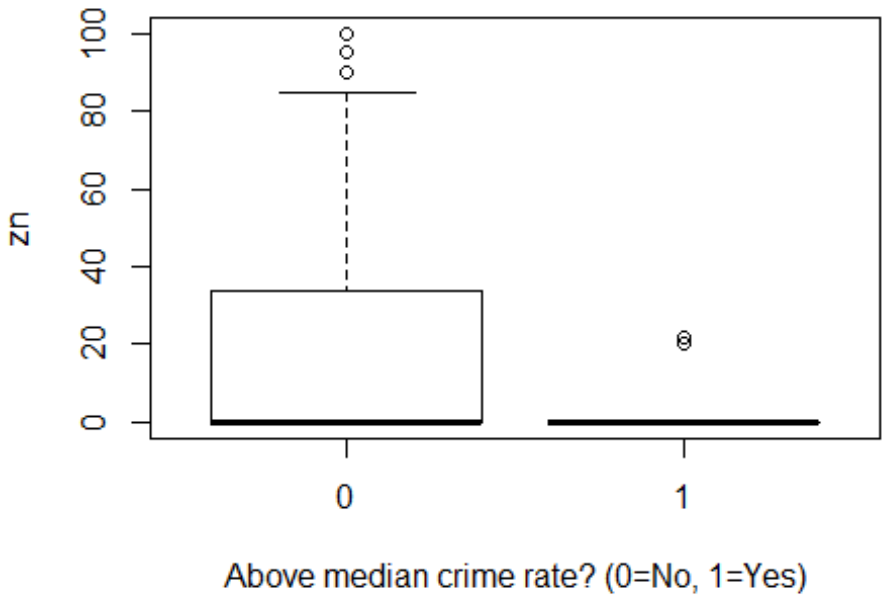
```r
ggplot(trgData, aes(x = tax)) + geom_density() #Not normal (two humps)
```



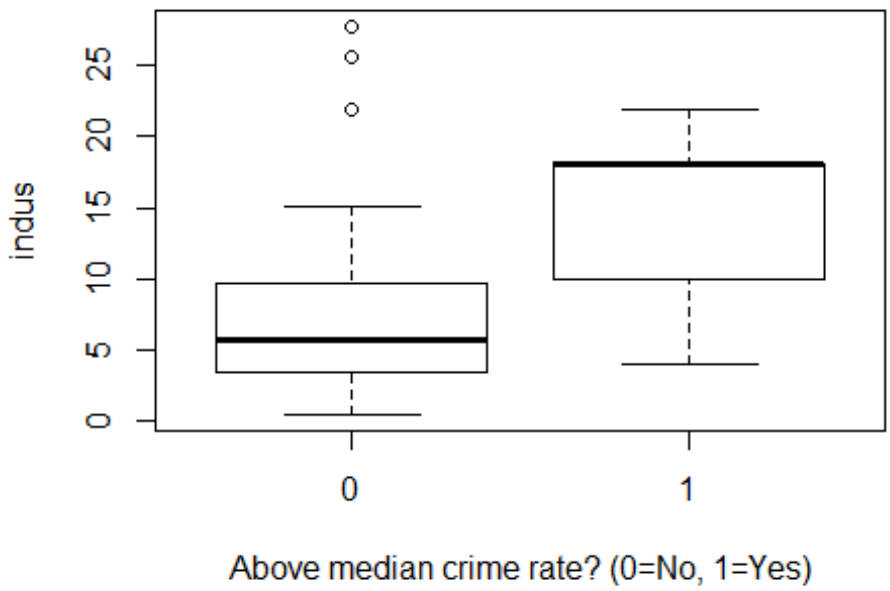

```r
ggplot(trgData, aes(x = ptratio)) + geom_density() #severely left skewed
```



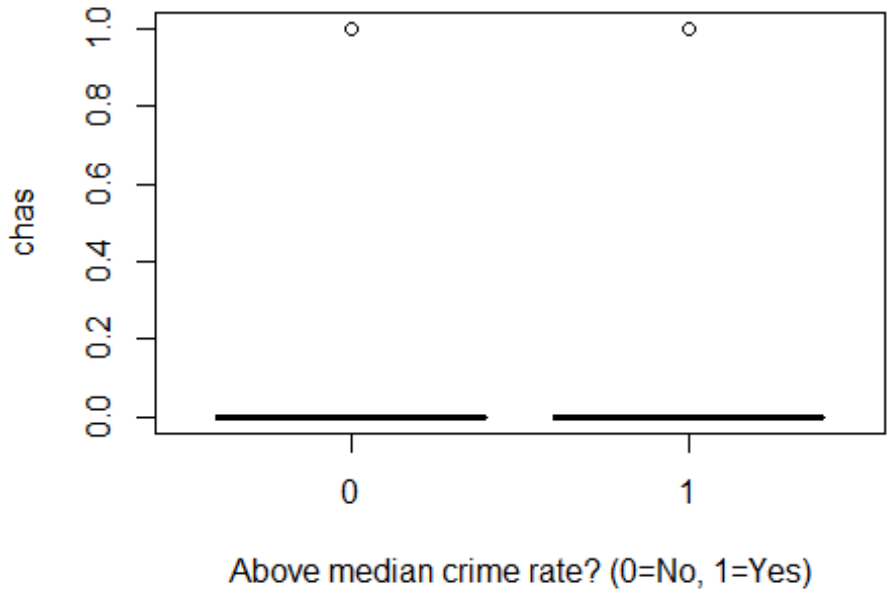
```
boxplot(zn~target, data=trgData, ylab="zn",xlab="Above median crime rate? (0=
No, 1=Yes)")
```



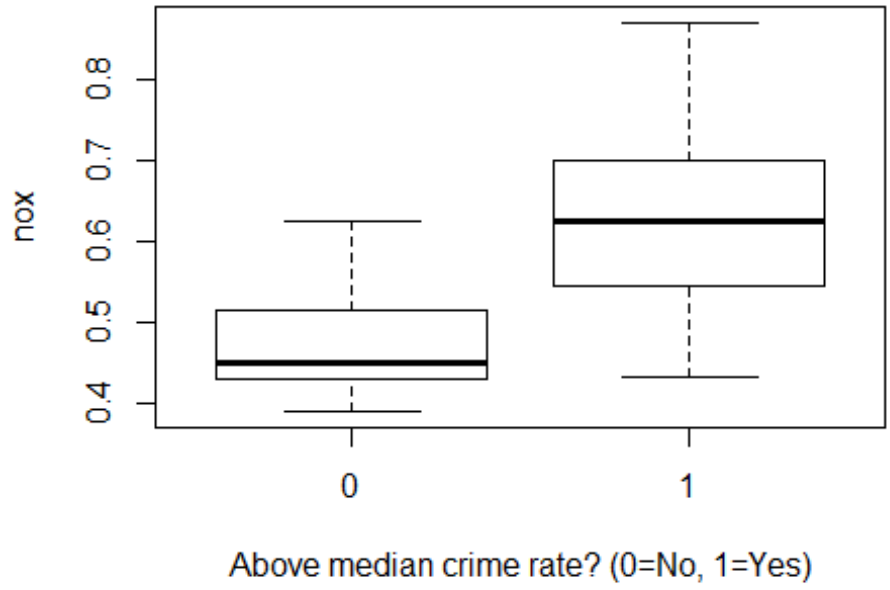

```
boxplot(indus~target, data=trgData, ylab="indus",xlab="Above median crime rat
e? (0=No, 1=Yes)")
```



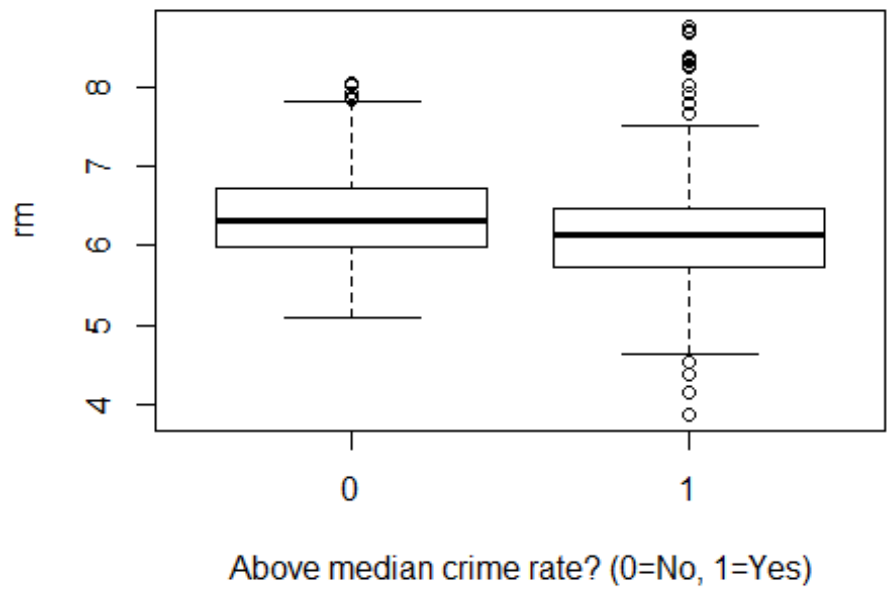

```
boxplot(chas~target, data=trgData, ylab="chas",xlab="Above median crime rate?
(0=No, 1=Yes)")
```



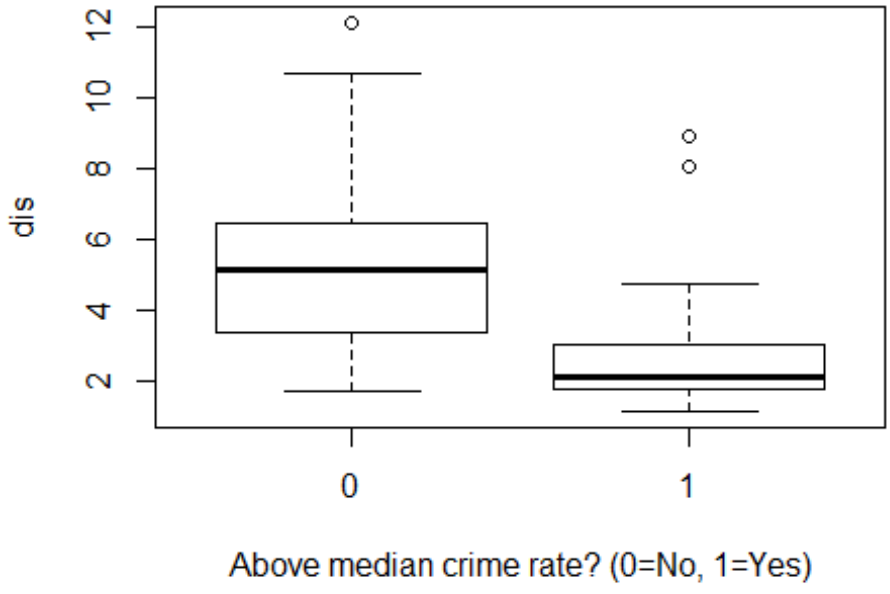
```
boxplot(nox~target, data=trgData, ylab="nox",xlab="Above median crime rate? (
0=No, 1=Yes)")
```



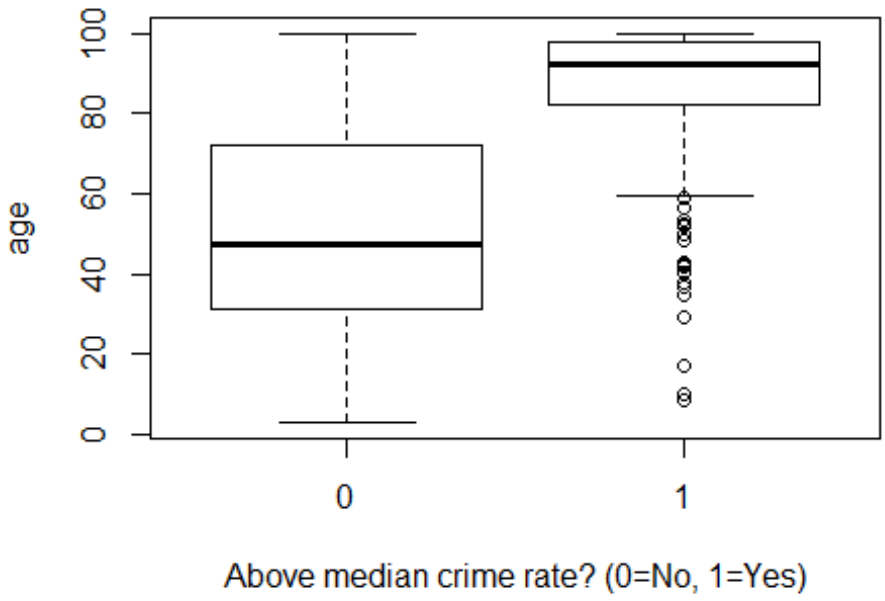

```
boxplot(rm~target, data=trgData, ylab="rm",xlab="Above median crime rate? (0=
No, 1=Yes)")
```



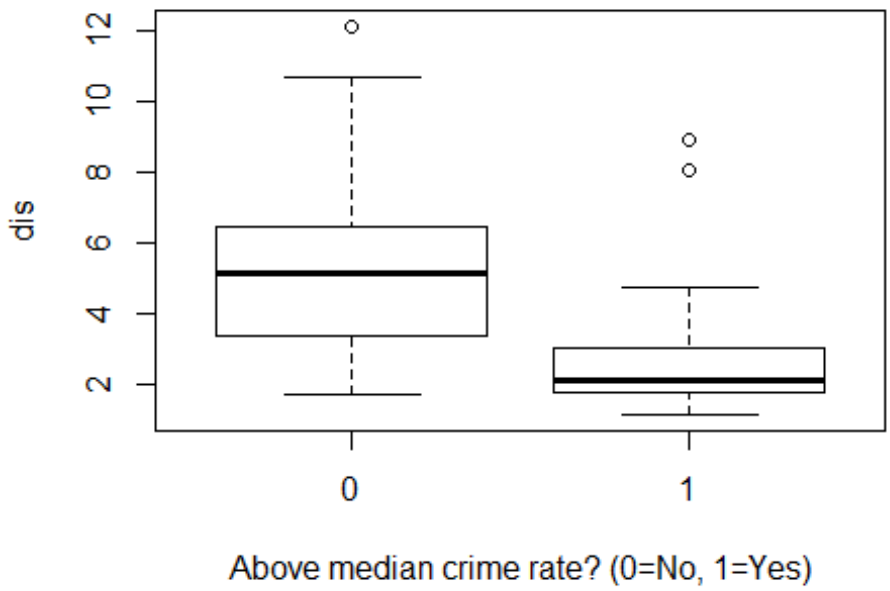

```
boxplot(dis~target, data=trgData, ylab="dis",xlab="Above median crime rate? (
0=No, 1=Yes)")
```



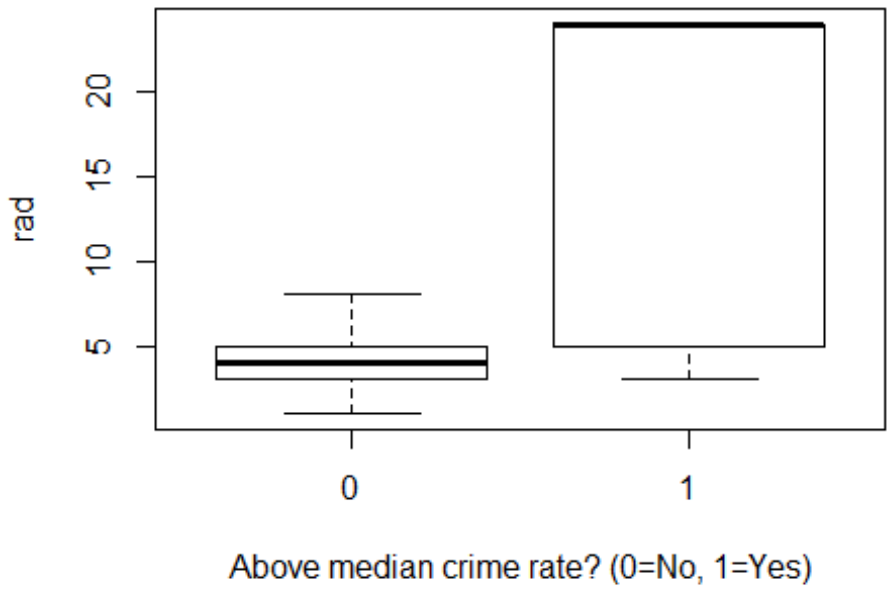
```
boxplot(age~target, data=trgData, ylab="age",xlab="Above median crime rate? (
0=No, 1=Yes)")
```



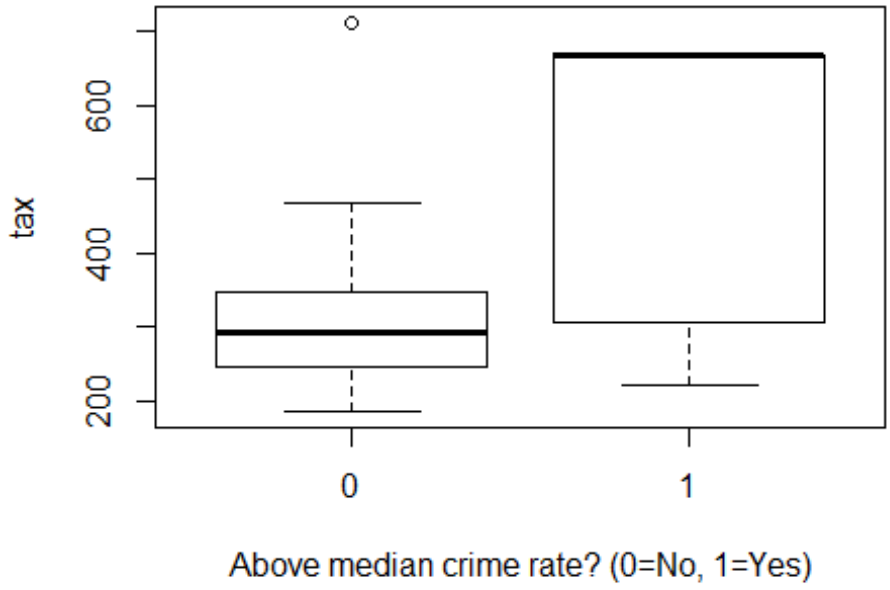

```
boxplot(dis~target, data=trgData, ylab="dis",xlab="Above median crime rate? (
0=No, 1=Yes)")
```



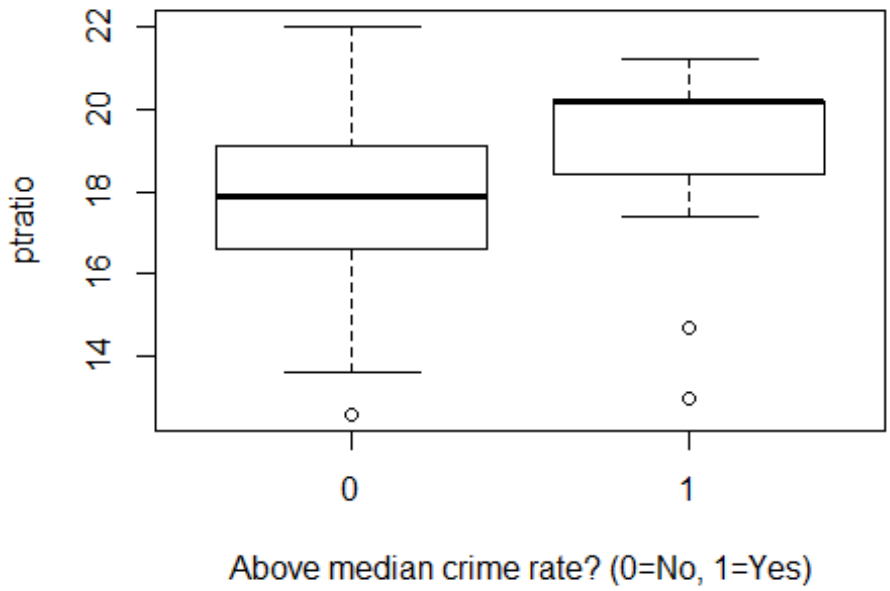

```
boxplot(rad~target, data=trgData, ylab="rad",xlab="Above median crime rate? (
0=No, 1=Yes)")
```



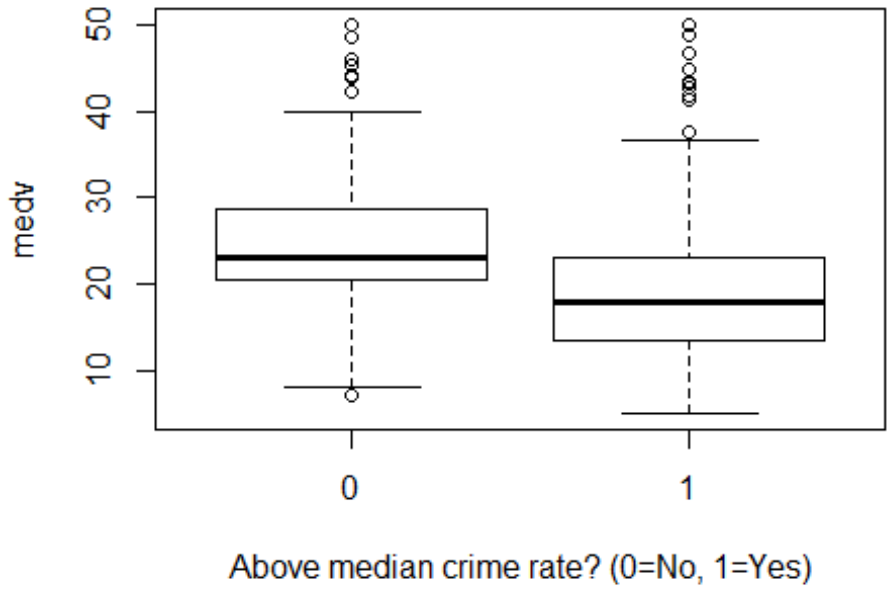
```
boxplot(tax~target, data=trgData, ylab="tax",xlab="Above median crime rate? (
0=No, 1=Yes)")
```



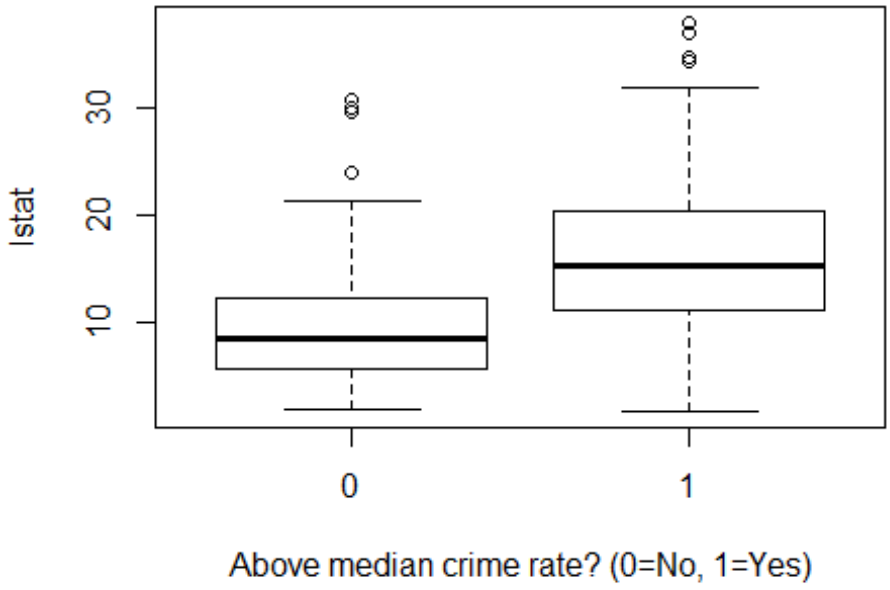

```
boxplot(ptratio~target, data=trgData, ylab="ptratio",xlab="Above median crime
rate? (0=No, 1=Yes)")
```



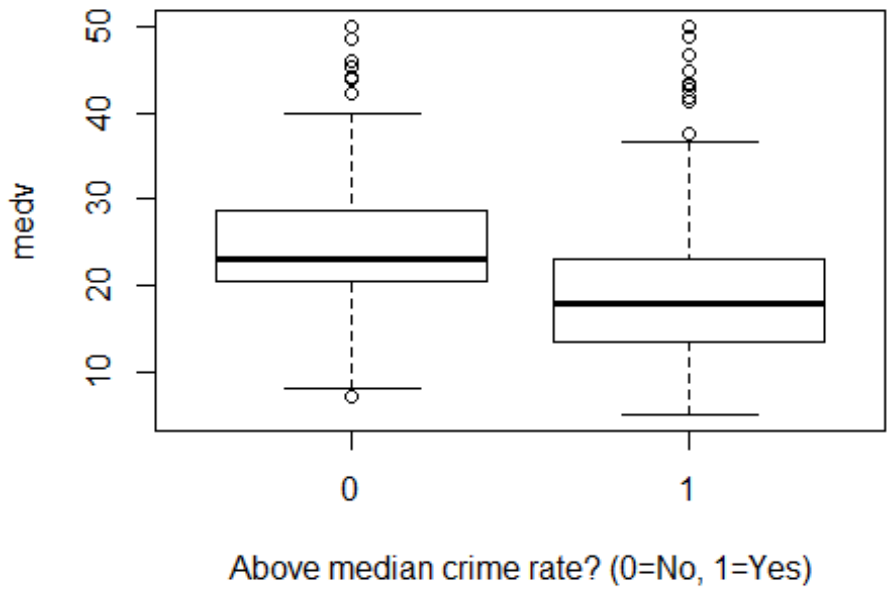

```
boxplot(medv~target, data=trgData, ylab="medv",xlab="Above median crime rate?
(0=No, 1=Yes)")
```



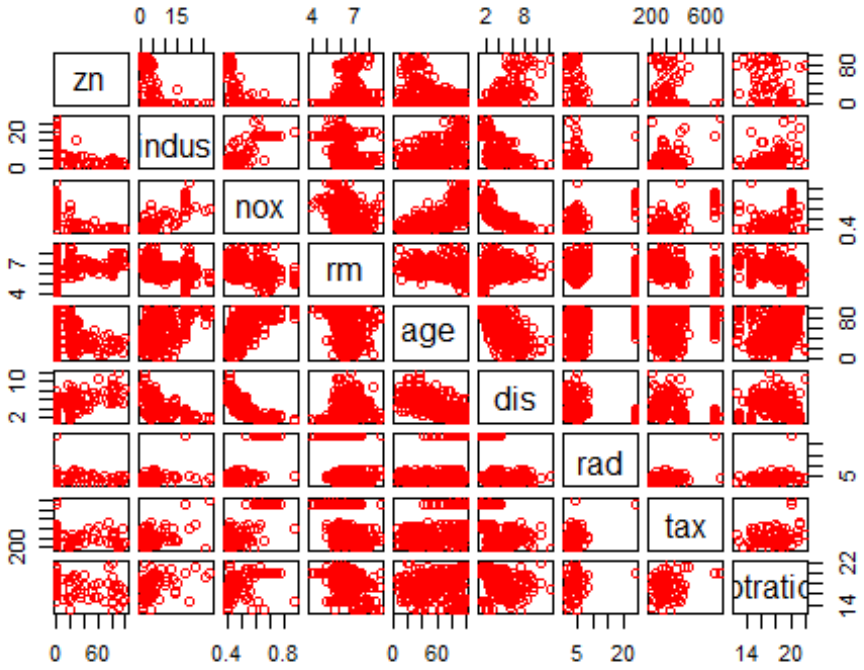
```r
boxplot(lstat~target, data=trgData, ylab="lstat",xlab="Above median crime rate? (0=No, 1=Yes)")
```



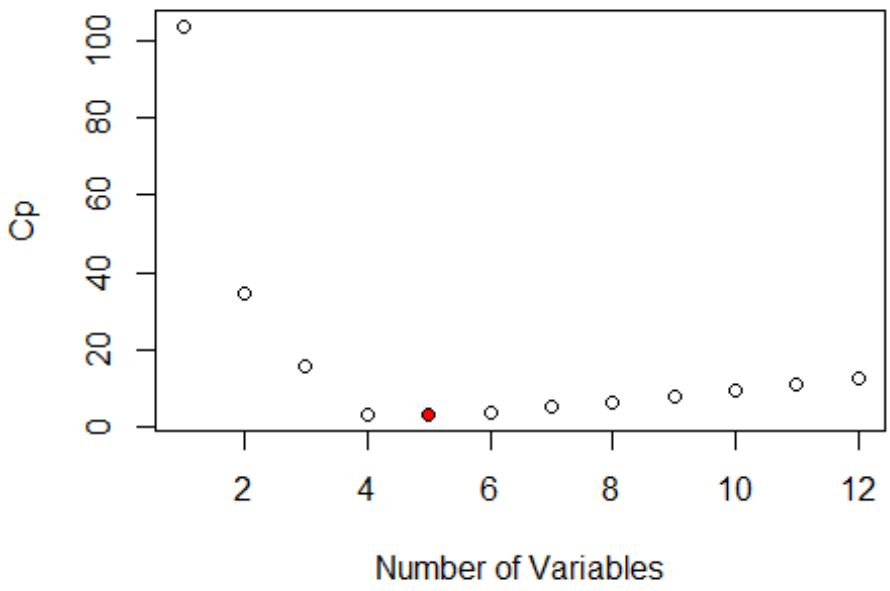

```r
boxplot(medv~target, data=trgData, ylab="medv",xlab="Above median crime rate? (0=No, 1=Yes)")
```



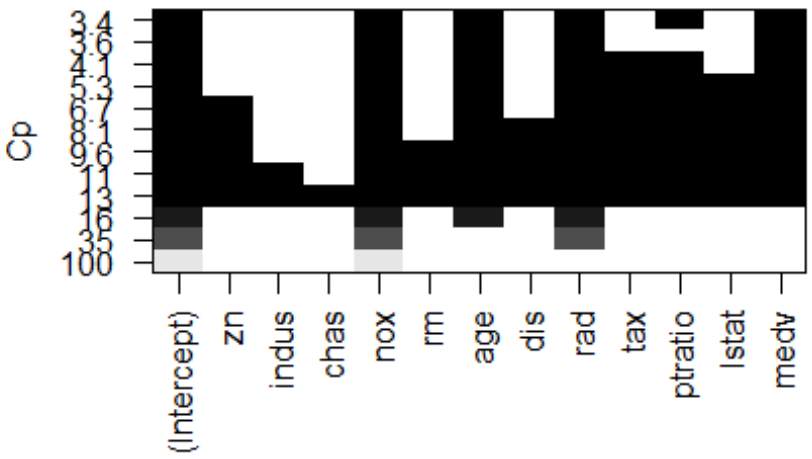

```
pairs(~zn+indus+nox+rm+age+dis+rad+tax+ptratio,data=trgData,gap=0.4,cex.label
s=1.5,col='red')
```



```
crimefit.full <-regsubsets(target~.,data=trgData,nvmax=14)
summary(crimefit.full)

## Subset selection object
## Call: regsubsets.formula(target ~ ., data = trgData, nvmax = 14)
## 12 Variables  (and intercept)
##         Forced in Forced out
## zn          FALSE      FALSE
## indus       FALSE      FALSE
## chas        FALSE      FALSE
## nox         FALSE      FALSE
## rm          FALSE      FALSE
## age         FALSE      FALSE
## dis         FALSE      FALSE
## rad         FALSE      FALSE
## tax         FALSE      FALSE
## ptratio     FALSE      FALSE
## lstat       FALSE      FALSE
## medv        FALSE      FALSE
## 1 subsets of each size up to 12
## Selection Algorithm: exhaustive
##           zn  indus chas nox rm  age dis rad tax ptratio lstat medv
## 1  ( 1 )  " " " "   " "  "*" " " " " " " " " " "     " "   " "
## 2  ( 1 )  " " " "   " "  "*" " " " " " " "*" " " " "     " "   " "
## 3  ( 1 )  " " " "   " "  "*" " " "*" " " "*" " " " "     " "   " "
## 4  ( 1 )  " " " "   " "  "*" " " "*" " " "*" " " " "     " "   "*"
## 5  ( 1 )  " " " "   " "  "*" " " "*" " " "*" " " "*"     " "   "*"
## 6  ( 1 )  " " " "   " "  "*" " " "*" " " "*" "*" "*"     " "   "*"
## 7  ( 1 )  " " " "   " "  "*" " " "*" " " "*" "*" "*"     "*"   "*"
## 8  ( 1 )  "*" " "   " "  "*" " " "*" " " "*" "*" "*"     "*"   "*"
## 9  ( 1 )  "*" " "   " "  "*" " " "*" "*" "*" "*" "*"     "*"   "*"
## 10  ( 1 ) "*" " "   " "  "*" "*" "*" "*" "*" "*" "*"     "*"   "*"
## 11  ( 1 ) "*" "*"   " "  "*" "*" "*" "*" "*" "*" "*"     "*"   "*"
## 12  ( 1 ) "*" "*"   "*"  "*" "*" "*" "*" "*" "*" "*"     "*"   "*"

crime.summary=summary(crimefit.full)
names(crime.summary)

## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"

plot(crime.summary$cp,xlab="Number of Variables", ylab="Cp")
which.min(crime.summary$cp)

## [1] 5

points(5,crime.summary$cp[5],pch=20,col="red")
```




```
plot(crimefit.full,scale="Cp")
```




```
coef(crimefit.full,5)
```

```
##  (Intercept)          nox          age          rad      ptratio
## -1.412836094  1.956694224  0.003531713  0.017106647  0.012716341
##         medv
##  0.008021190
```

```
bestsubsetModel1 <- glm(target ~ nox + age + rad + ptratio + medv, family=bin
omial, data = trgData)
summary(bestsubsetModel1)
```

```
##
## Call:
## glm(formula = target ~ nox + age + rad + ptratio + medv, family = binomial
,
##     data = trgData)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.96654  -0.29783  -0.03987   0.00769   2.80829
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -24.936540   3.683449  -6.770 1.29e-11 ***
## nox          25.334778   4.084106   6.203 5.53e-10 ***
## age           0.019403   0.009308   2.085  0.03711 *
## rad           0.512600   0.114818   4.464 8.03e-06 ***
## ptratio       0.274193   0.098737   2.777  0.00549 **
## medv          0.085445   0.027979   3.054  0.00226 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 224.71  on 460  degrees of freedom
## AIC: 236.71
##
## Number of Fisher Scoring iterations: 8
```