Applied Data Science II - Homework 1

Phileas Dazeley Gaist

10/01/2021

ISLR 2.4: Questions 1, 2, 8, 10

Libraries

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(PerformanceAnalytics)
```

```
## Loading required package: xts

## Loading required package: zoo
```

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

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

##
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':
##
##     first, last

##
## Attaching package: 'PerformanceAnalytics'

## The following object is masked from 'package:graphics':
##
##     legend
```

1. ISLR 2.4 - 1

a) Better: The estimated f fit will be more accurate to the true f, especially given the large sample size.

b) Worse: The estimated f will be less accurate, and the risk of over fitting is much increased due to the small n.

c) Better: Nonlinearity in data is better suited to flexible models, and which will produce closer fits to the data.

d) Worse: High variance in error terms indicates the use of an overly flexible method which finds patterns where there are none in the true f, it is a sign of over fitting.

ISLR 2.4 - 2

a)

Problem type: regression (response (CEO salary) is quantitative). Interest: inference (we are interested in understanding how factors affect the CEO salary) n = 500 (firms), p = 3 (profit, number of employees, industry)

b)

Problem type: classification (response (success or failure) is qualitative) Interest: prediction (we are interested in predicting how the factors affect success or failure) n = 20 (similar products that were previously launched), p = 14 (price charged for the product, marketing budget, competition price, and ten other variables.)

c)

Problem type: regression (response (% change in the dollar) is quantitative) Interest: prediction (we are interested in predicting % change in the dollar) n = 52 (weeks), p = 3 (% change in the US market, % change in the British market, and % change in the German market)

ISLR 2.4 - 8

a)

```
college <- read_csv("Homework 1 data/College.csv")  # load the data
```

b)

```
college  # view the data
```

```
## # A tibble: 777 x 19
##    ...1  Private  Apps Accept Enroll Top10perc Top25perc F.Undergrad P.Undergrad
##    <chr> <chr>   <dbl>  <dbl>  <dbl>     <dbl>     <dbl>       <dbl>       <dbl>
##  1 Abil~ Yes      1660   1232    721        23        52        2885         537
##  2 Adel~ Yes      2186   1924    512        16        29        2683        1227
##  3 Adri~ Yes      1428   1097    336        22        50        1036          99
##  4 Agne~ Yes       417    349    137        60        89         510          63
##  5 Alas~ Yes       193    146     55        16        44         249         869
##  6 Albe~ Yes       587    479    158        38        62         678          41
##  7 Albe~ Yes       353    340    103        17        45         416         230
##  8 Albi~ Yes      1899   1720    489        37        68        1594          32
##  9 Albr~ Yes      1038    839    227        30        63         973         306
## 10 Alde~ Yes       582    498    172        21        44         799          78
## # ... with 767 more rows, and 10 more variables: Outstate <dbl>,
## #   Room.Board <dbl>, Books <dbl>, Personal <dbl>, PhD <dbl>, Terminal <dbl>,
## #   S.F.Ratio <dbl>, perc.alumni <dbl>, Expend <dbl>, Grad.Rate <dbl>
```

```
rownames(college) <- college[, 1, drop = TRUE]  # assign college names to row names
# NOTE: drop = TRUE must be specified for the code to run, this is not
# disclosed in the exercise instructions

# Alternatively, this works too: rownames(college) <- college$...1

college = college[, -1]  # remove original college names column

college  # view it again
```

```
## # A tibble: 777 x 18
##    Private  Apps Accept Enroll Top10perc Top25perc F.Undergrad P.Undergrad
##    <chr>   <dbl>  <dbl>  <dbl>     <dbl>     <dbl>       <dbl>       <dbl>
```

```
##  1 Yes      1660   1232    721        23        52        2885         537
##  2 Yes      2186   1924    512        16        29        2683        1227
##  3 Yes      1428   1097    336        22        50        1036          99
##  4 Yes       417    349    137        60        89         510          63
##  5 Yes       193    146     55        16        44         249         869
##  6 Yes       587    479    158        38        62         678          41
##  7 Yes       353    340    103        17        45         416         230
##  8 Yes      1899   1720    489        37        68        1594          32
##  9 Yes      1038    839    227        30        63         973         306
## 10 Yes       582    498    172        21        44         799          78
## # ... with 767 more rows, and 10 more variables: Outstate <dbl>,
## #   Room.Board <dbl>, Books <dbl>, Personal <dbl>, PhD <dbl>, Terminal <dbl>,
## #   S.F.Ratio <dbl>, perc.alumni <dbl>, Expend <dbl>, Grad.Rate <dbl>
```

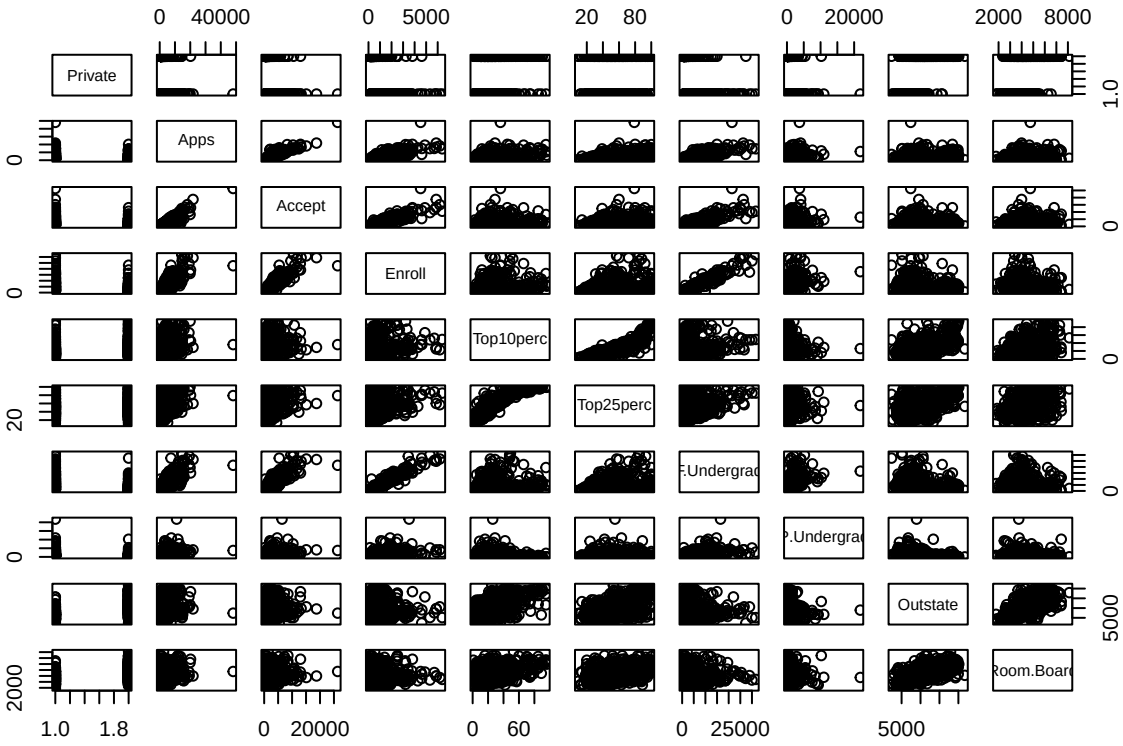
c)

```
# i)

summary(college)  # produce a numerical summary of the variables in the data set
```

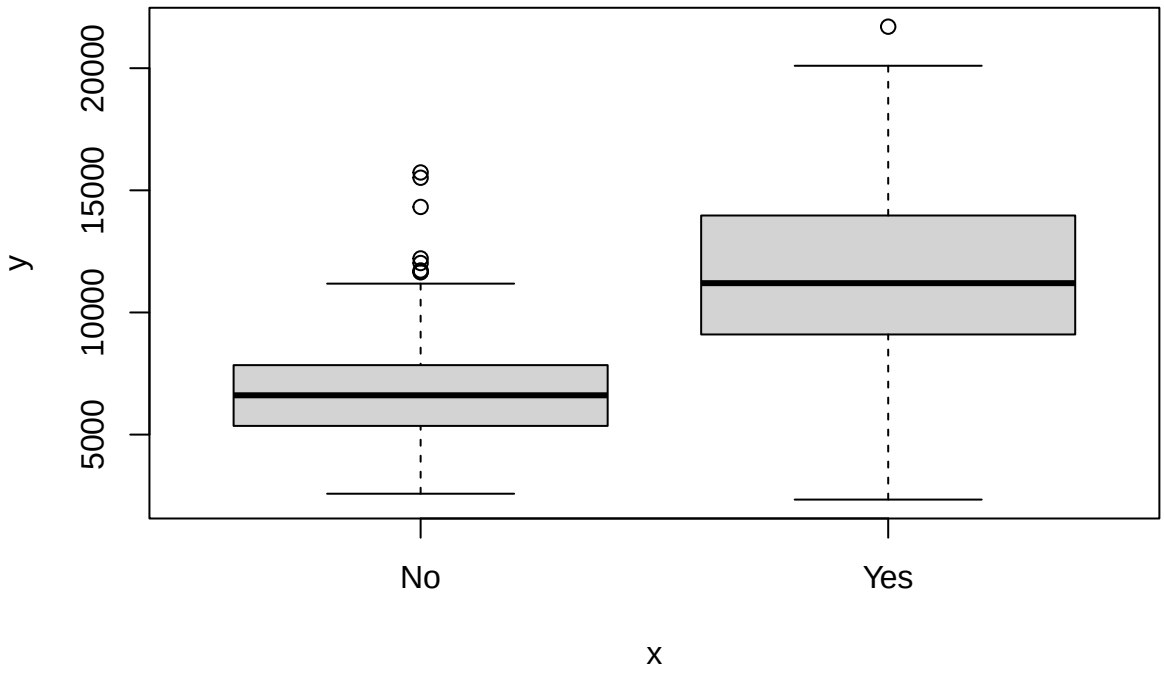
```
##    Private                Apps           Accept          Enroll
##  Length:777         Min.   :   81   Min.   :   72   Min.   :  35
##  Class :character   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242
##  Mode  :character   Median : 1558   Median : 1110   Median : 434
##                     Mean   : 3002   Mean   : 2019   Mean   : 780
##                     3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902
##                     Max.   :48094   Max.   :26330   Max.   :6392
##    Top10perc       Top25perc      F.Undergrad     P.Undergrad
##  Min.   : 1.00   Min.   :  9.0   Min.   :  139   Min.   :    1.0
##  1st Qu.:15.00   1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0
##  Median :23.00   Median : 54.0   Median : 1707   Median :  353.0
##  Mean   :27.56   Mean   : 55.8   Mean   : 3700   Mean   :  855.3
##  3rd Qu.:35.00   3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0
##  Max.   :96.00   Max.   :100.0   Max.   :31643   Max.   :21836.0
##     Outstate       Room.Board       Books           Personal
##  Min.   : 2340   Min.   :1780   Min.   :  96.0   Min.   : 250
##  1st Qu.: 7320   1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850
##  Median : 9990   Median :4200   Median : 500.0   Median :1200
##  Mean   :10441   Mean   :4358   Mean   : 549.4   Mean   :1341
##  3rd Qu.:12925   3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700
##  Max.   :21700   Max.   :8124   Max.   :2340.0   Max.   :6800
##       PhD            Terminal       S.F.Ratio      perc.alumni
##  Min.   :  8.00   Min.   : 24.0   Min.   : 2.50   Min.   : 0.00
##  1st Qu.: 62.00   1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00
##  Median : 75.00   Median : 82.0   Median :13.60   Median :21.00
##  Mean   : 72.66   Mean   : 79.7   Mean   :14.09   Mean   :22.74
##  3rd Qu.: 85.00   3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00
```

```
##  Max.   :103.00   Max.   :100.0   Max.   :39.80   Max.   :64.00
##      Expend        Grad.Rate
##  Min.   : 3186   Min.   : 10.00
##  1st Qu.: 6751   1st Qu.: 53.00
##  Median : 8377   Median : 65.00
##  Mean   : 9660   Mean   : 65.46
##  3rd Qu.:10830   3rd Qu.: 78.00
##  Max.   :56233   Max.   :118.00
```

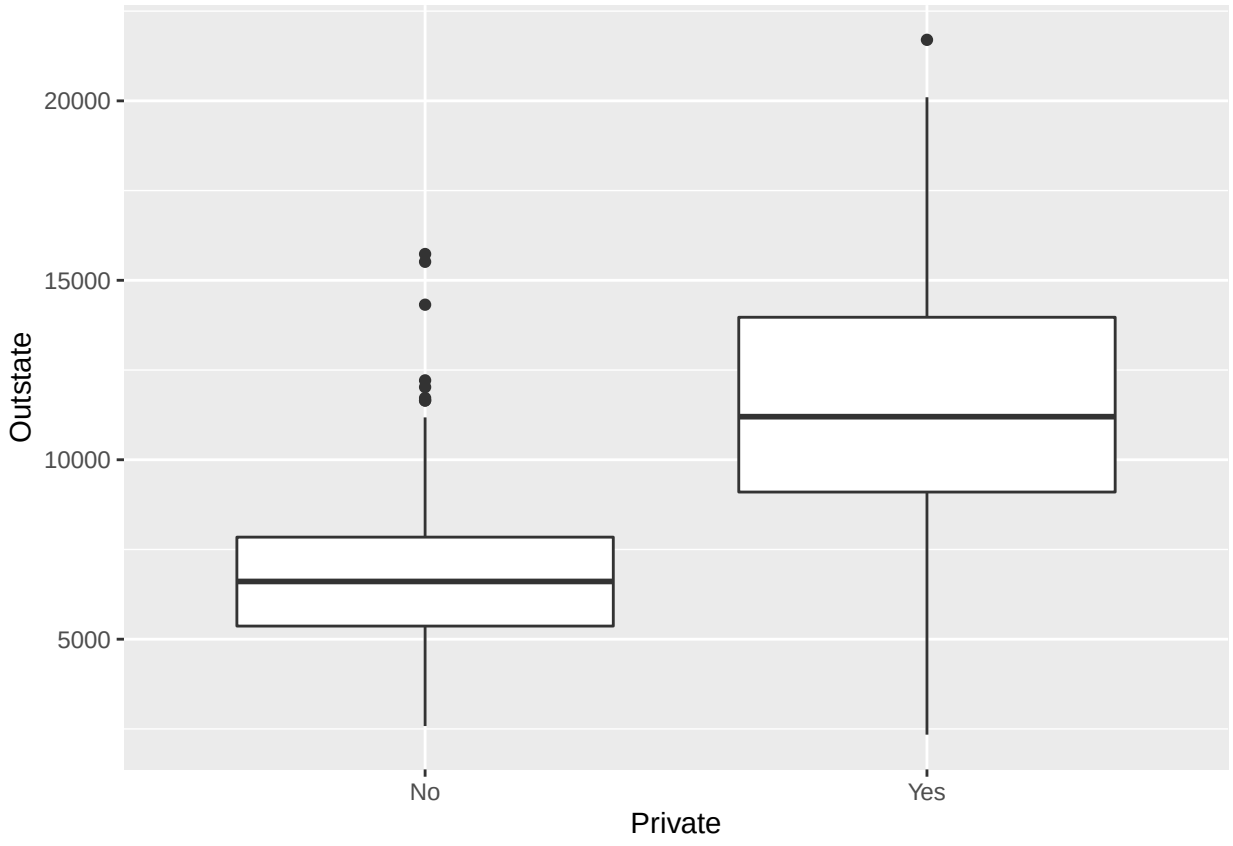
```
# ii)

# The code suggested by the exercise does not run:
# https://community.rstudio.com/t/error-in-pairs-function/81983

college$Private <- as.factor(college$Private)  # The 'Private' column
# needs to be converted to a factor before the pair function will accept it.

pairs(college[, 1:10])
```




```
# iii)

plot(college$Private, college$Outstate)
```



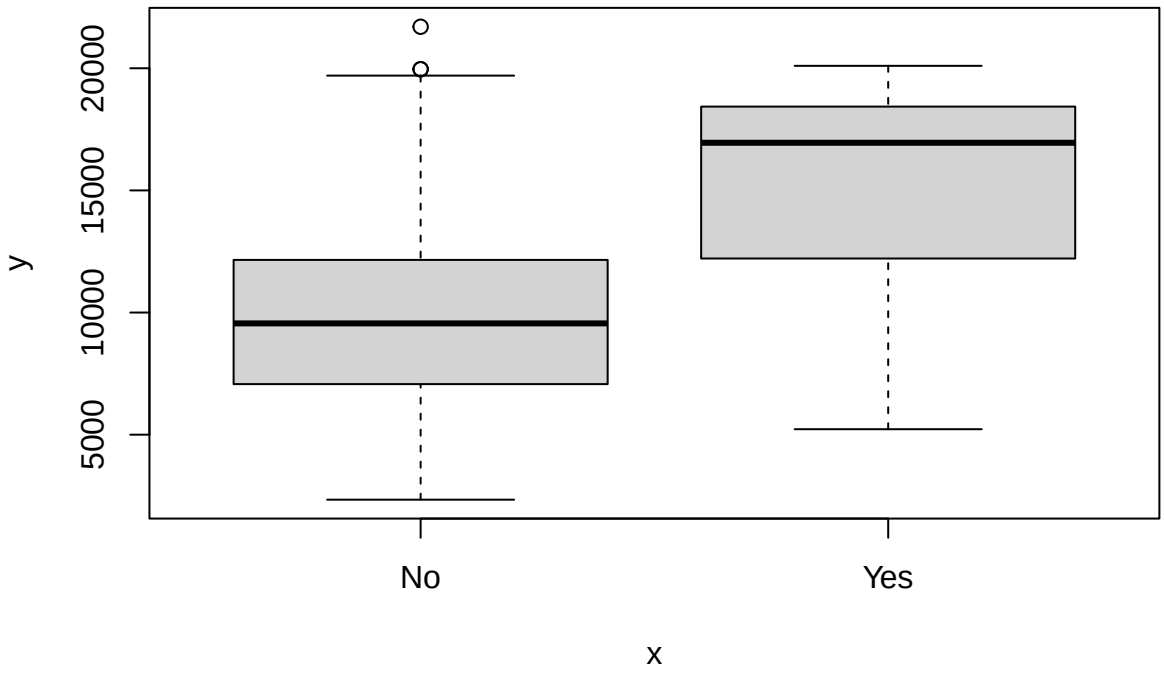

```
# ggplot version (bonus)
college %>%
    ggplot(aes(x = Private, y = Outstate)) + geom_boxplot()
```



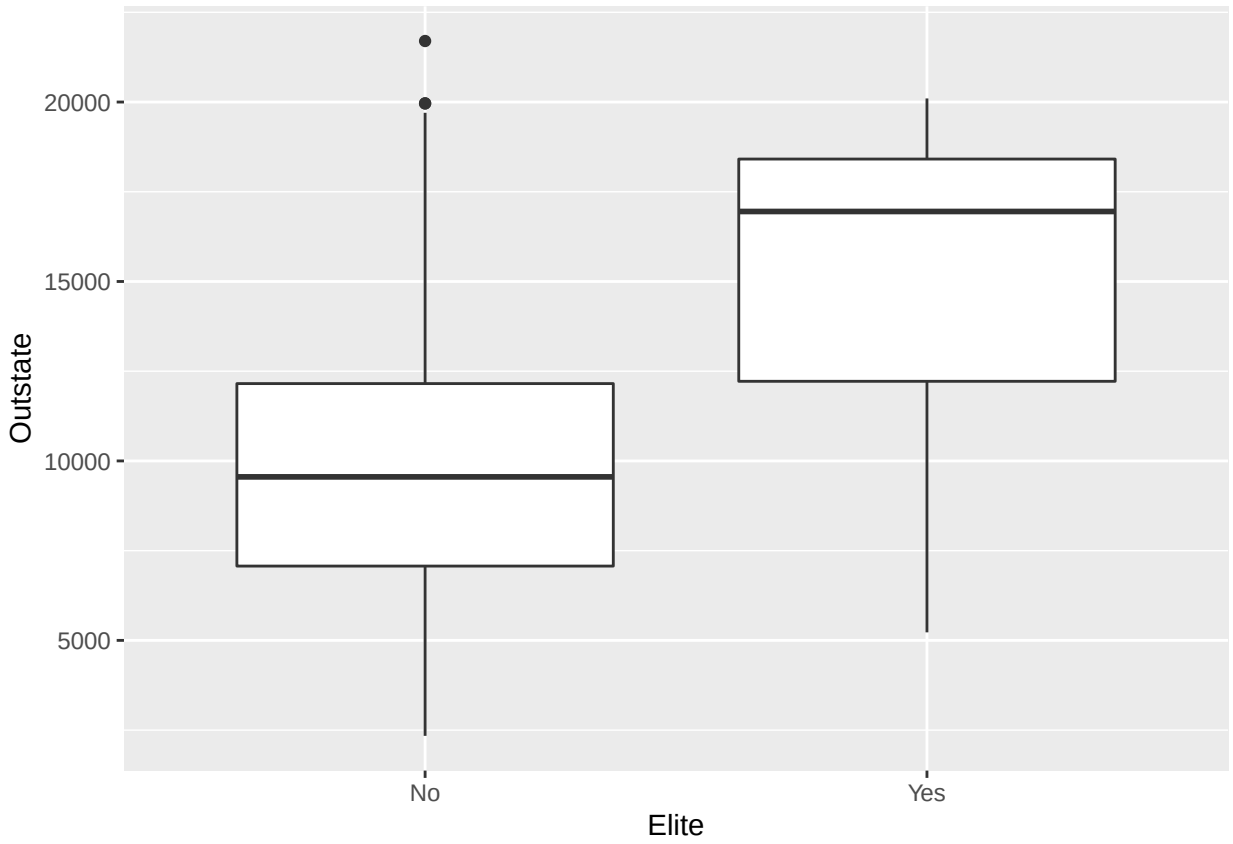
```r
# iv)

Elite <- rep("No", nrow(college))
Elite[college$Top10perc > 50] <- "Yes"
Elite <- as.factor(Elite)
college <- data.frame(college, Elite)

summary(college$Elite)
```

```
##  No Yes
## 699  78
```

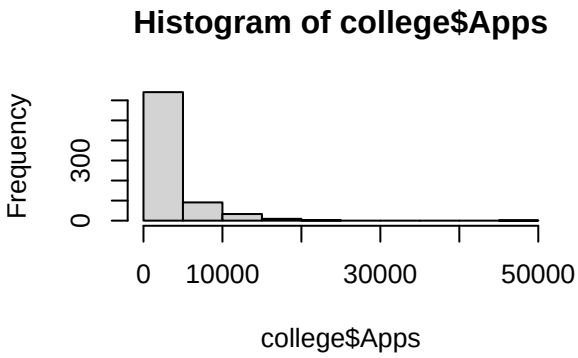
```r
plot(college$Elite, college$Outstate)
```

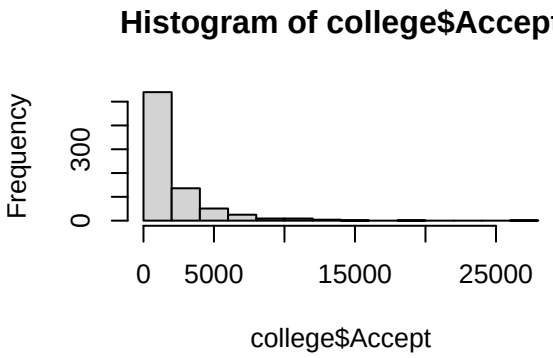


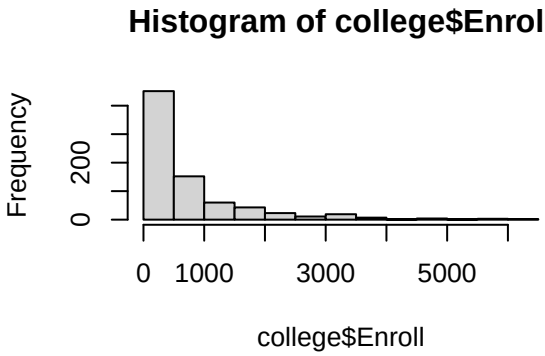

```r
# ggplot version (bonus)
college %>%
    ggplot(aes(x = Elite, y = Outstate)) + geom_boxplot()
```

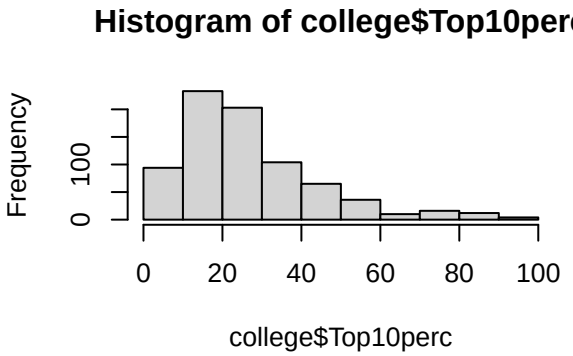



```
# v)

par(mfrow = c(2, 2))
hist(college$Apps)
hist(college$Accept)
hist(college$Enroll)
hist(college$Top10perc)
```

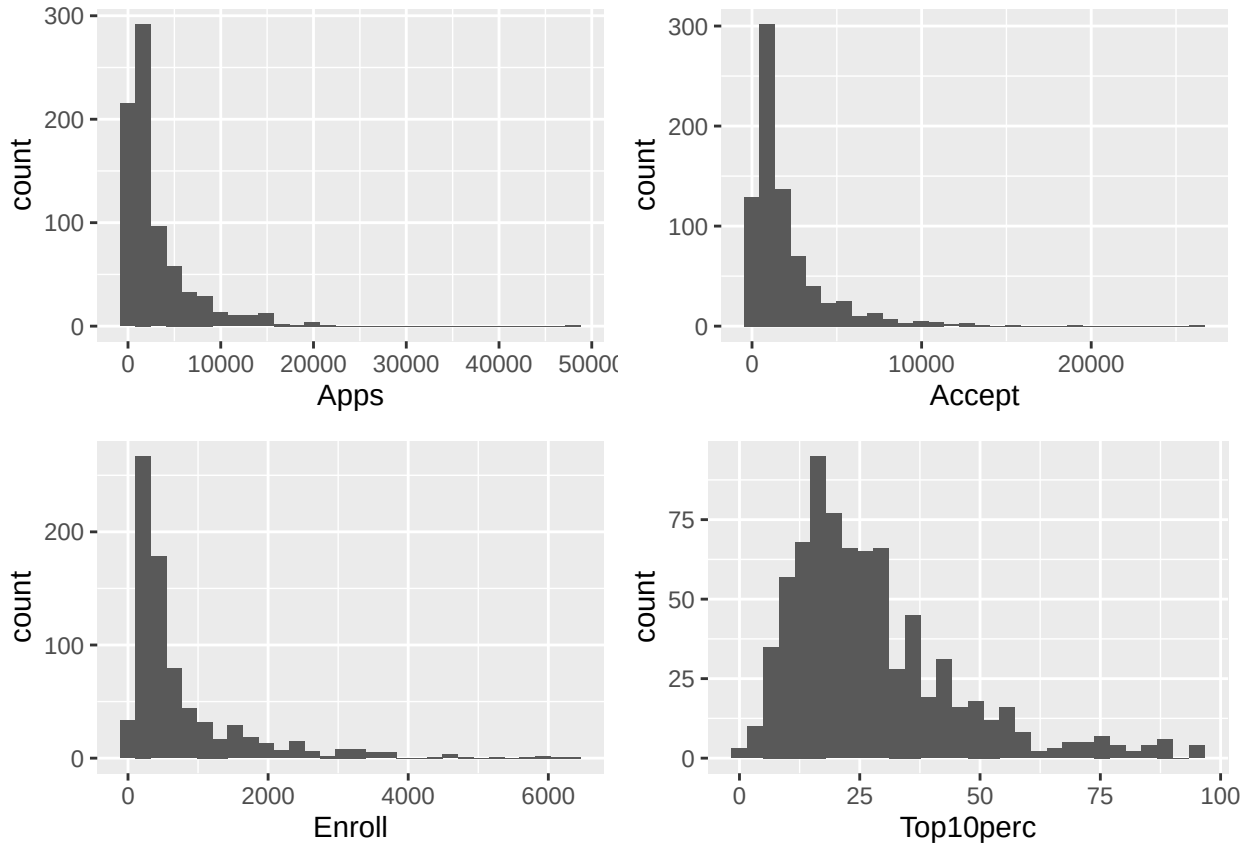




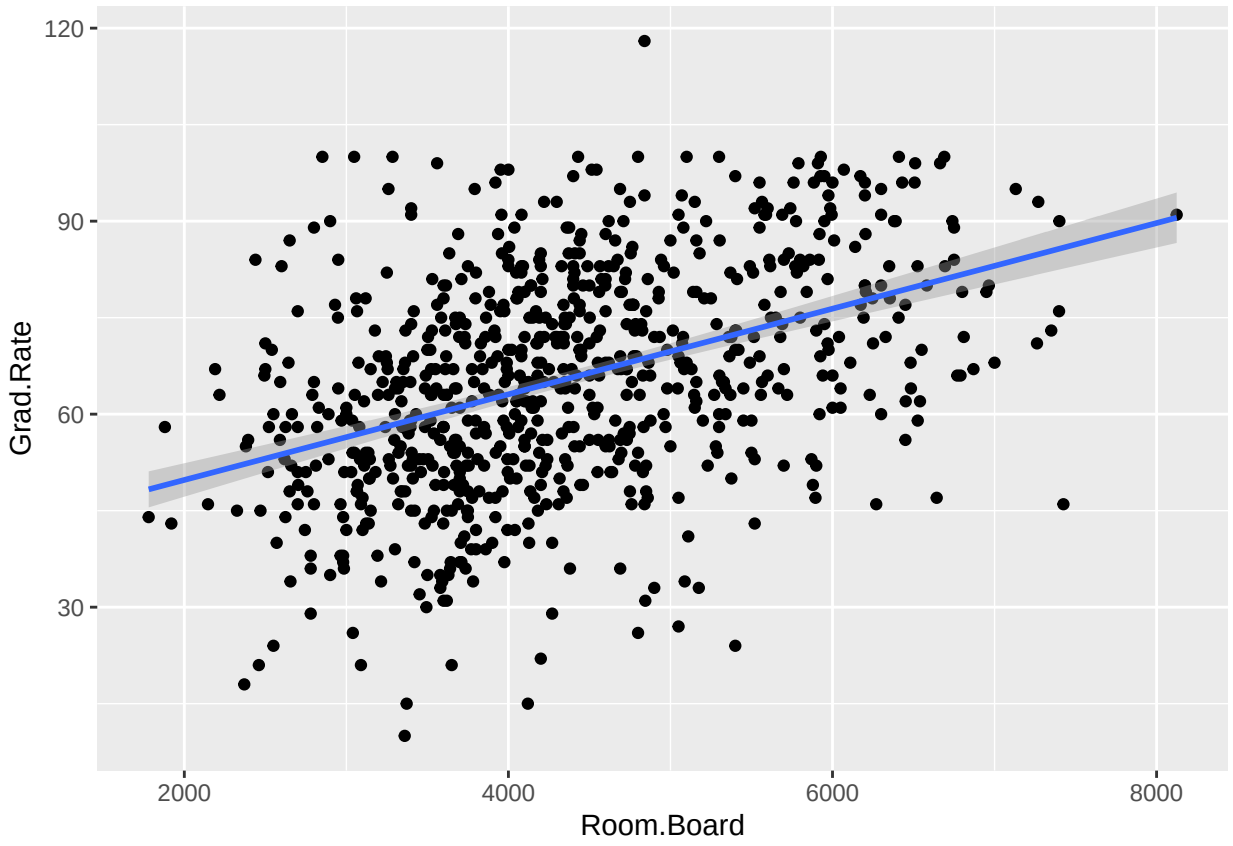




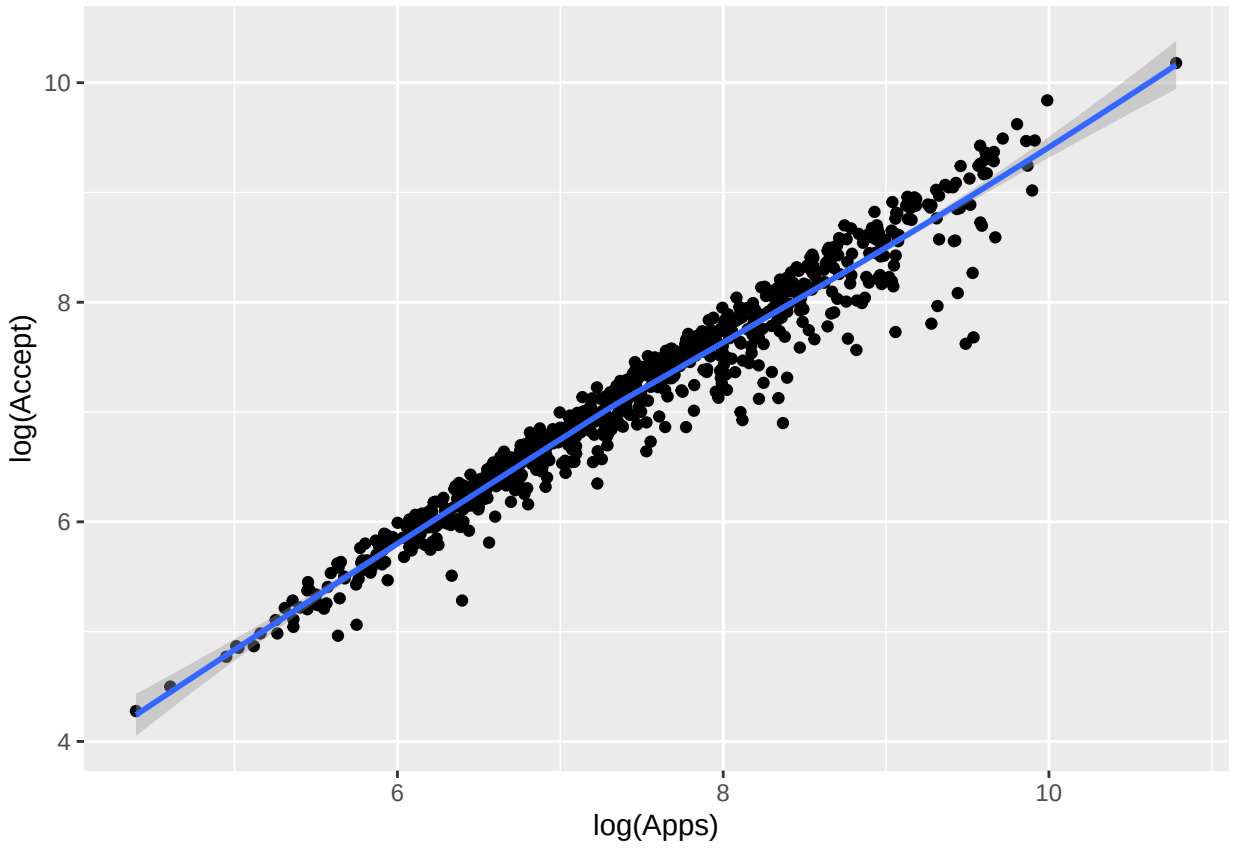

```
# ggplot version (bonus)
apps_plot <- college %>%
    ggplot() + geom_histogram(aes(x = Apps))
accept_plot <- college %>%
    ggplot() + geom_histogram(aes(x = Accept))
enroll_plot <- college %>%
    ggplot() + geom_histogram(aes(x = Enroll))
top10perc_plot <- college %>%
    ggplot() + geom_histogram(aes(x = Top10perc))

grid.arrange(apps_plot, accept_plot, enroll_plot, top10perc_plot, ncol = 2, nrow = 2)
```



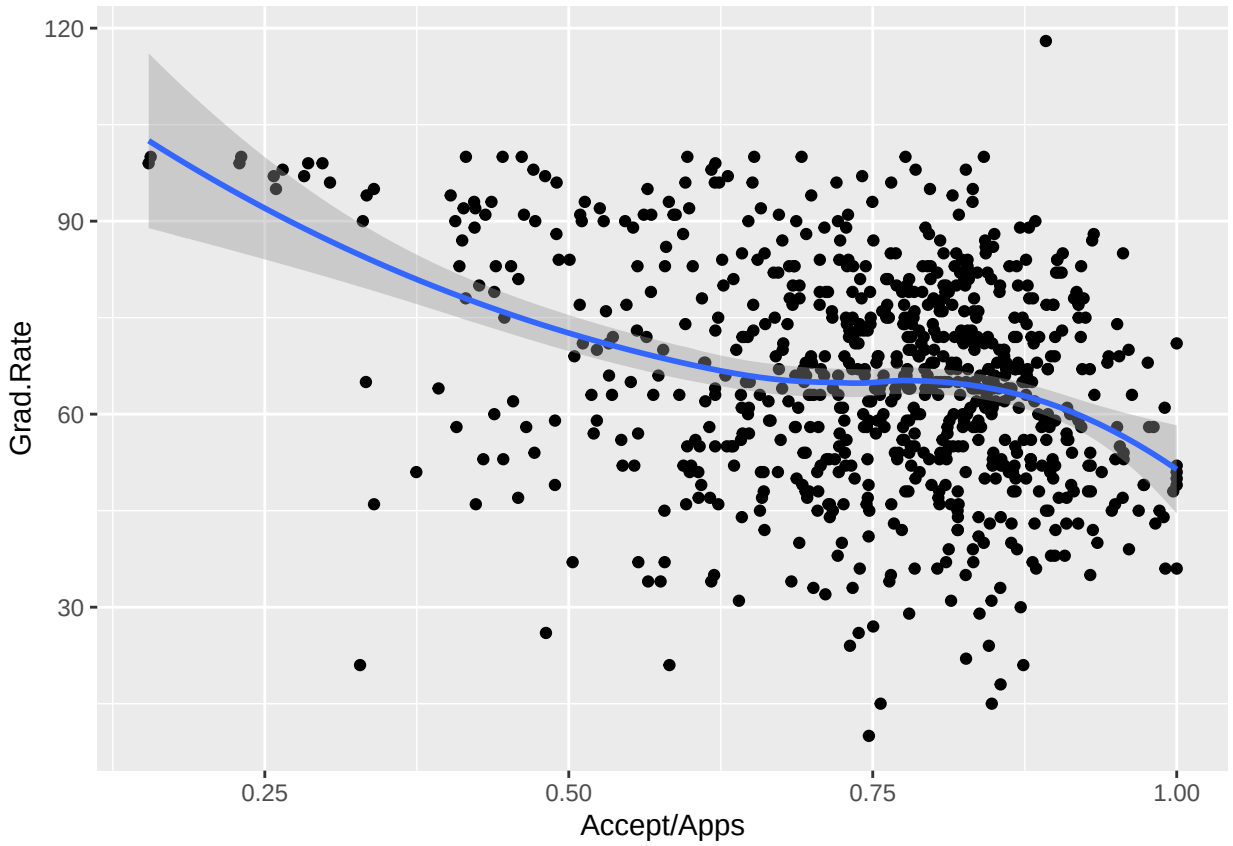

```
# vi)

# The cost of room & board and the graduation rate exhibit a possible linear
# relationship:
college %>%
    ggplot(aes(x = Room.Board, y = Grad.Rate)) + geom_point() + geom_smooth(method = "lm")
```




```
# The log of Applications and the log of Acceptances exhibit strong linear
# correlation:
college %>%
    ggplot(aes(x = log(Apps), y = log(Accept))) + geom_point() + geom_smooth()
```




```
# The acceptance rate of a college appears correlated not quite linearly with
# the graduation rate:
college %>%
    ggplot(aes(x = Accept/Apps, y = Grad.Rate)) + geom_point() + geom_smooth(method = "loess")
```




```
cor.test((college$Accept/college$Apps), college$Grad.Rate, alternative = "two.sided",
    conf.level = 0.95)
```

```
##
##  Pearson's product-moment correlation
##
## data:  (college$Accept/college$Apps) and college$Grad.Rate
## t = -8.3397, df = 775, p-value = 3.39e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3502356 -0.2211010
## sample estimates:
##        cor
## -0.2869715
```

ISLR 2.4 - 10

This exercise involves the Boston housing data set.

a)

```
library(MASS)
head(Boston, 5)
```

```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio  black lstat
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90  4.98
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90  9.14
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83  4.03
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63  2.94
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90  5.33
##   medv
## 1 24.0
## 2 21.6
## 3 34.7
## 4 33.4
## 5 36.2
```

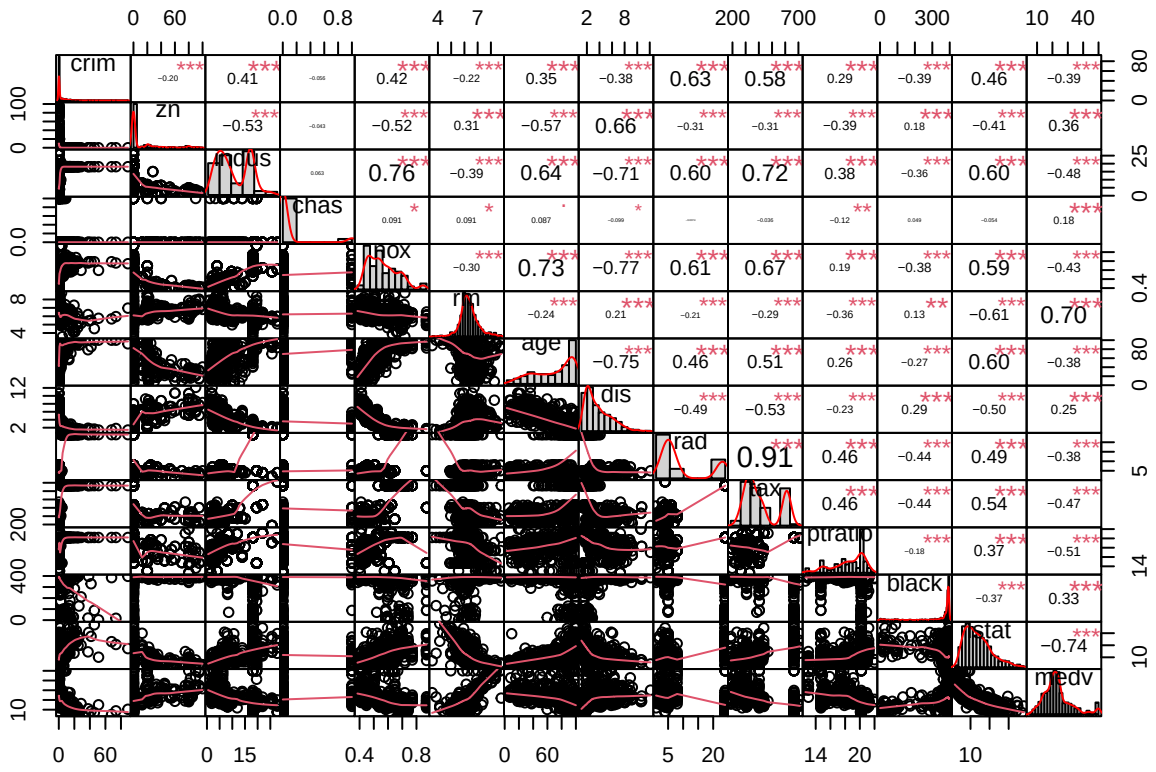
```
`?`(Boston)

dim(Boston)
```

```
## [1] 506  14
```

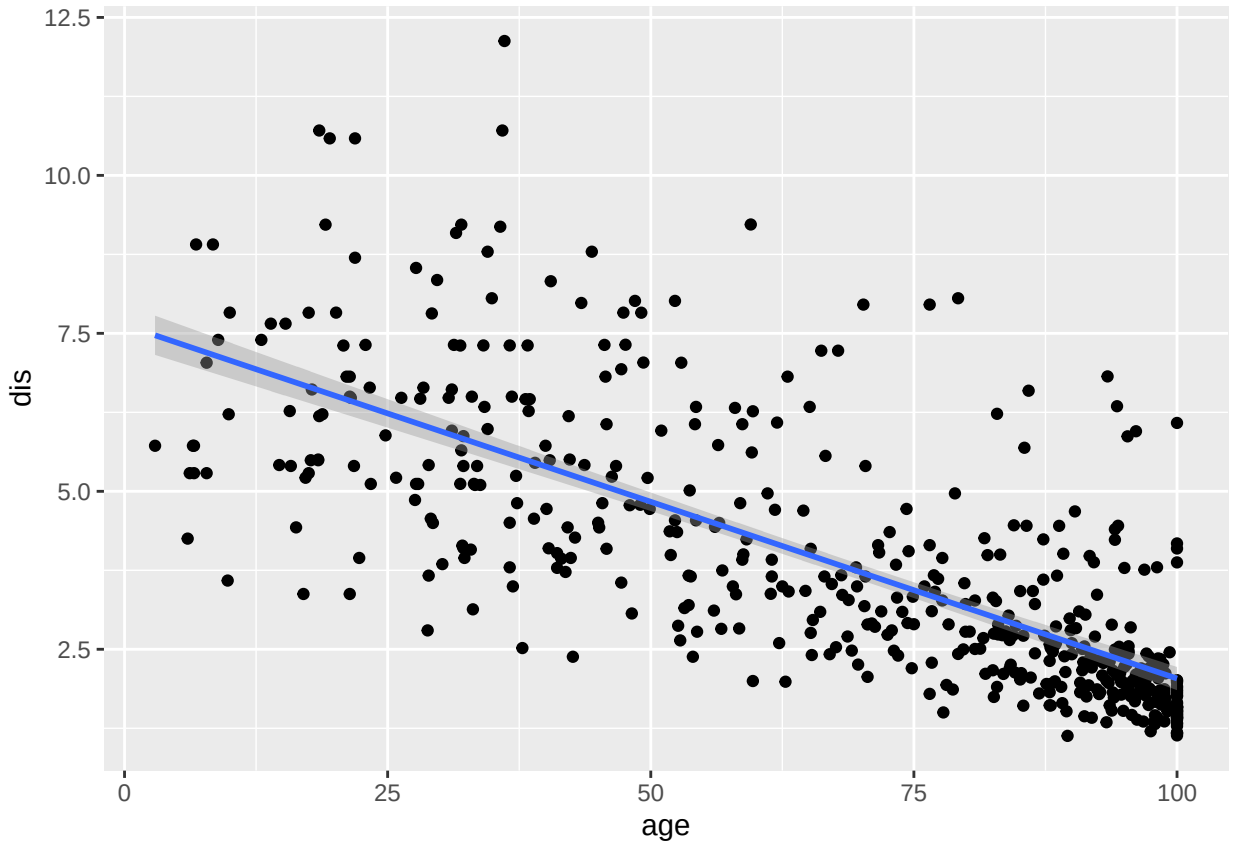
```
# The Boston data set contains 506 rows, and 14 columns. The columns represent
# predictor (independent) variables, while the rows represent response
# (dependent) variables.
```

b)

```
chart.Correlation(Boston, histogram = TRUE)
```




```
Boston %>%
    ggplot(aes(x = age, y = dis)) + geom_point() + geom_smooth(method = "lm")
```




```
cor.test(Boston$age, Boston$dis, alternative = "two.sided", conf.level = 0.95)
```

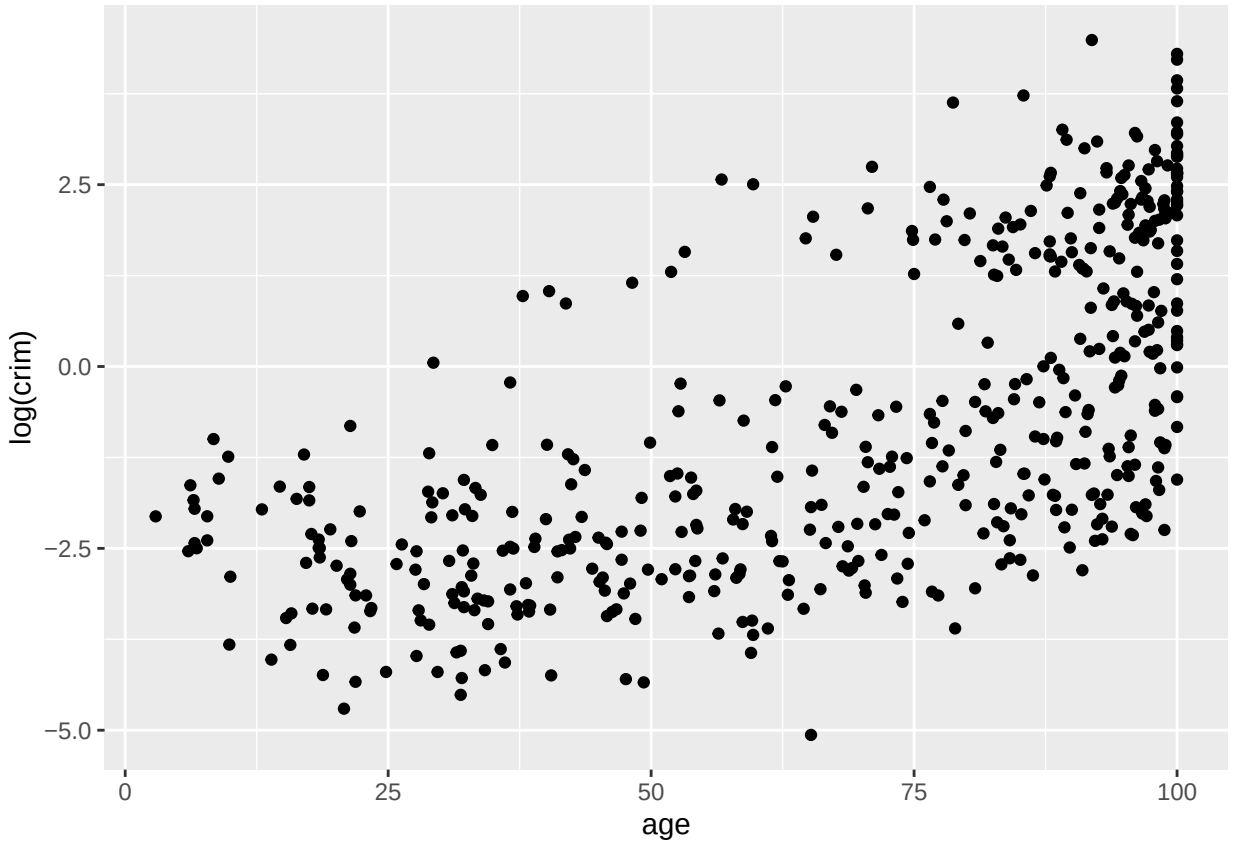
```
##
##  Pearson's product-moment correlation
##
## data:  Boston$age and Boston$dis
## t = -25.292, df = 504, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7839426 -0.7067887
## sample estimates:
##       cor
## -0.7478805
```

- There is a strong positive correlation between the proportion of owner occupied units built prior to 1914 and the nitrogen oxides concentration in parts for 10 million.
- There is a strong slightly negative correlation between the proportion of owner occupied units built prior to 1914 and the weighted mean of distances to 5 Boston employment centres.
- There is no correlation between whether a given suburb Borders the Charles River and the per capita crime rate.

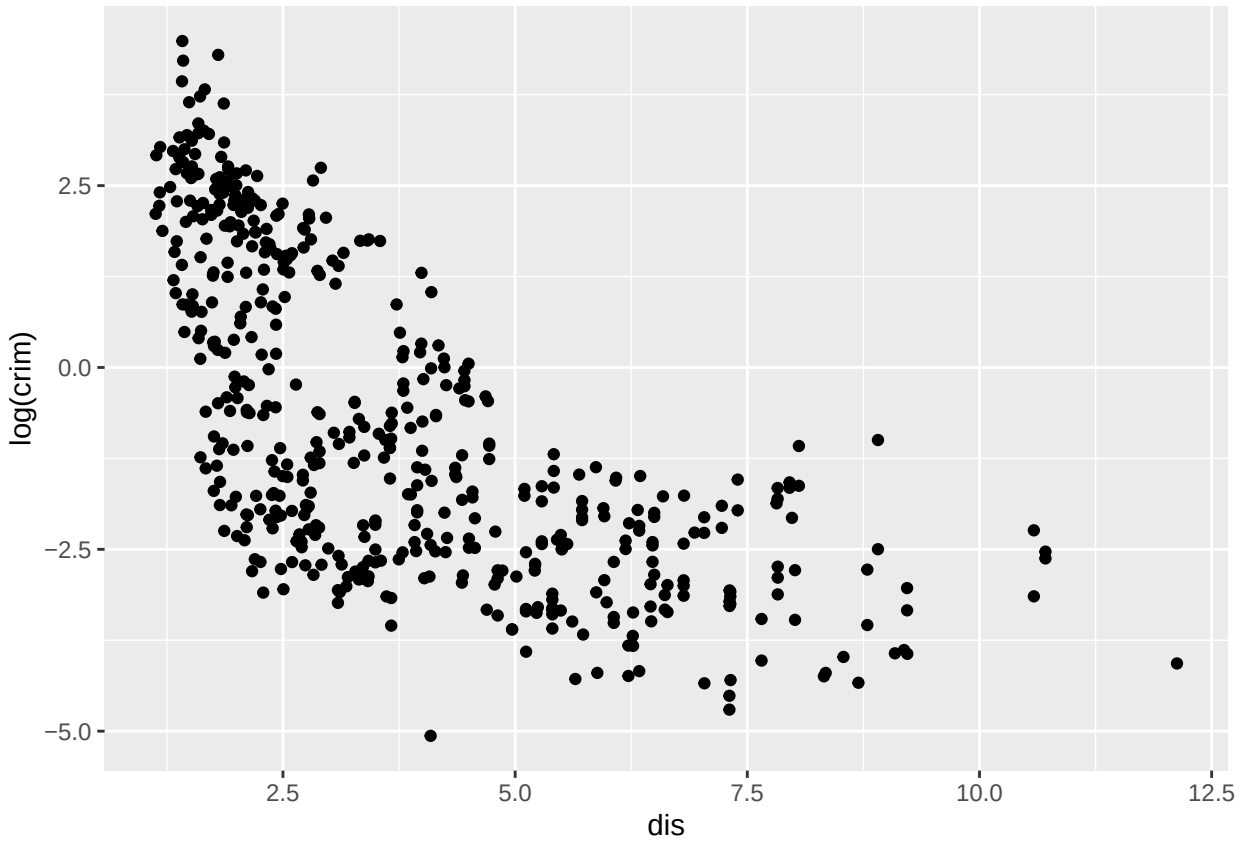
c)

Yes, for example:

```
# There is a strong positive correlation between the Proportion of owner
# occupied unit built prior to 1940 and the log per capita crime rate.
Boston %>%
    ggplot(aes(age, log(crim))) + geom_point()
```




```
# There is a strong negative correlation between the weighted mean of distances
# to 5 Boston employment centres and the log per capita crime rate.
Boston %>%
    ggplot(aes(dis, log(crim))) + geom_point()
```



d)

- Most tracts are reported to have low crime rates, so the distribution on crime rate observations skew towards lower values. However the tail of the distribution is long, indicating the existence of a smaller number of towns with higher crime rates. The mean crime rate for the data set is 3.61, and 128 of the 506 towns represented in the data have crime rates above the mean. (reaching a maximum value of 88.97).
- There's a great gap between, on the left of the distribution, a majority of towns represented in the data set, where inhabitants pay lower taxes, and towns where inhabitants pay higher taxes on the right of the distribution. And this gap indicates the presence of hidden variables.
- There is a slight skew awards higher pupil-teacher ratios by town.

```
# summaries of the range of distributions and measures of central tendency:
summary(Boston$crim)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##   0.00632   0.08204   0.25651   3.61352   3.67708 88.97620
```

```
summary(Boston$tax)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   187.0   279.0   330.0   408.2   666.0   711.0
```

```r
summary(Boston$ptratio)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   12.60   17.40   19.05   18.46   20.20   22.00
```

```r
# numbers of observations above the mean of each distribution
dim(subset(Boston, crim > 3.61))[1]
```

```
## [1] 128
```

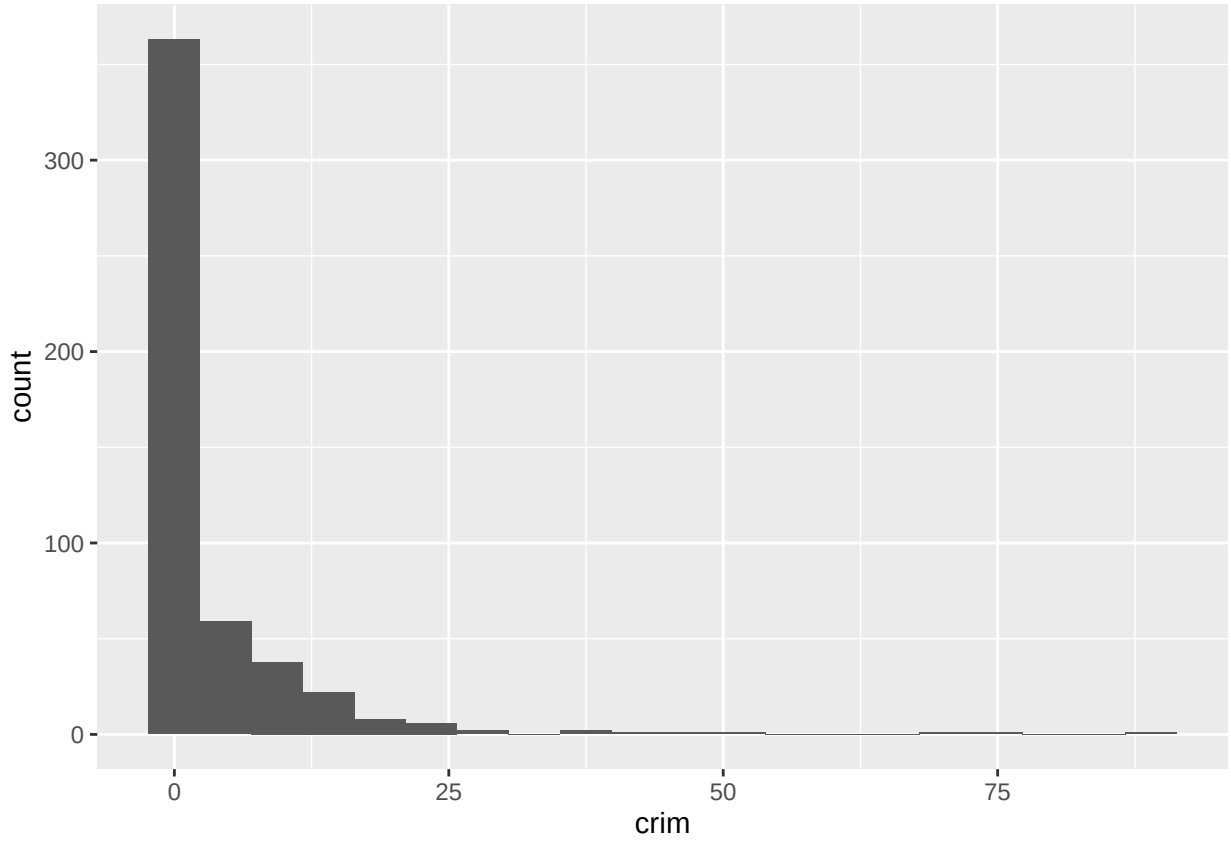
```r
dim(subset(Boston, tax > 408.2))[1]
```

```
## [1] 168
```

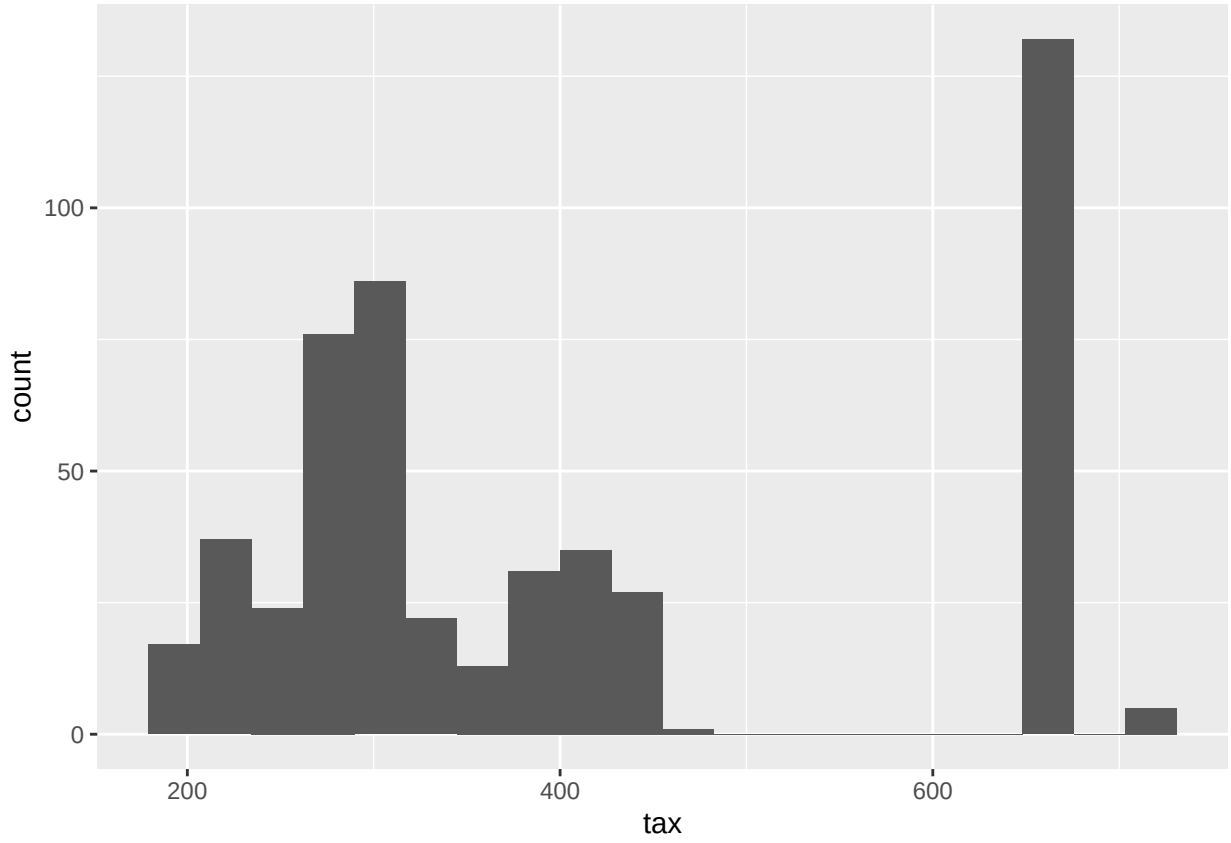
```r
dim(subset(Boston, ptratio > 18.46))[1]
```

```
## [1] 292
```

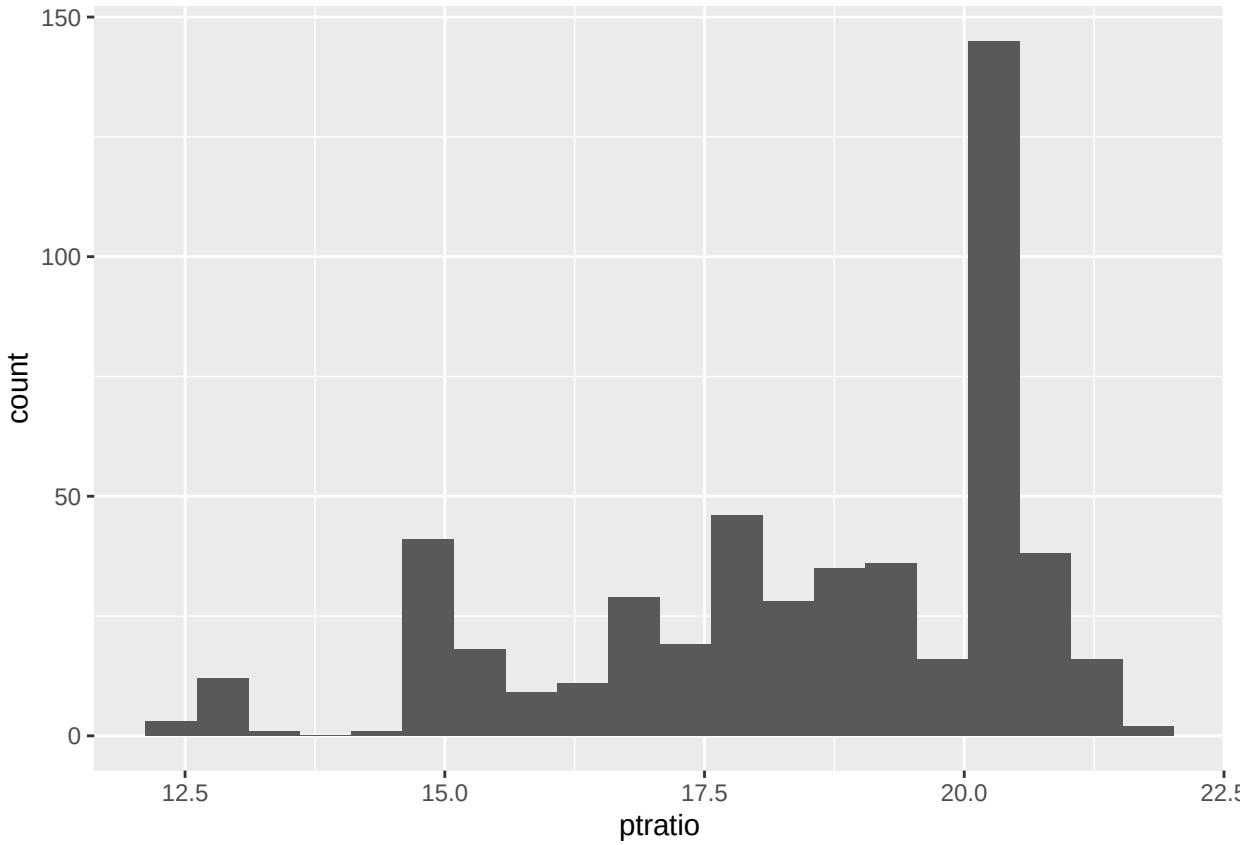
```r
# histograms
ggplot(data = Boston, aes(crim)) + geom_histogram(bins = 20)
```



```
ggplot(data = Boston, aes(tax)) + geom_histogram(bins = 20)
```




```
ggplot(data = Boston, aes(ptratio)) + geom_histogram(bins = 20)
```



e)

There are 35 tracts that border the Charles River.

```
# numbers of observations (towns) that border the Charles river:
dim(subset(Boston, chas == 1))[1]
```

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

f)

The median pupil-teacher ratio in the data set is 19.05

```
# median pupil-teacher ratio in the data set:
median(Boston$ptratio)
```

```
## [1] 19.05
```

g)

There are actually two tracts tied at this minimum value of $5k median value of owner-occupied homes.

Comparisons to the distributions of predictors for the entire dataset are reported in the second table below.

```r
# data frame of summaries of distributions (same formatting as following data
# frame for ease of reading)
summaries <- summary(Boston) %>%
    as.data.frame() %>%
    dplyr::select(-Var1) %>%
    separate(Freq, into = c("Name", "Value"), sep = ":") %>%
    pivot_wider(names_from = "Name", values_from = "Value") %>%
    mutate_all(as.numeric)
summaries
```

```
## # A tibble: 14 x 7
##     Var2 `Min.   ` `1st Qu.` `Median ` `Mean   ` `3rd Qu.` `Max.   `
##    <dbl>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>
##  1     1   0.00632    0.0820     0.257    3.61       3.68     89.0
##  2     2   0          0          0       11.4       12.5     100
##  3     3   0.46       5.19       9.69    11.1       18.1      27.7
##  4     4   0          0          0        0.0692     0         1
##  5     5   0.385      0.449      0.538    0.555      0.624     0.871
##  6     6   3.56       5.89       6.21     6.28       6.62      8.78
##  7     7   2.9       45.0       77.5     68.6       94.1     100
##  8     8   1.13       2.1        3.21     3.80       5.19     12.1
##  9     9   1          4          5        9.55      24        24
## 10    10 187        279        330      408.       666       711
## 11    11  12.6       17.4       19.0     18.5       20.2      22
## 12    12   0.32     375.       391.     357.       396.      397.
## 13    13   1.73       6.95      11.4     12.6       17.0      38.0
## 14    14   5         17.0       21.2     22.5       25        50
```

```r
# values of the predictors for the tract with the min median home value:
Boston %>%
    filter(medv == min(medv)) %>%
    t() %>%
    as.data.frame() %>%
    mutate(`V1 - sample mean` = V1 - summaries$`Mean   `, `V2 - sample mean` = V2 -
        summaries$`Mean   `)
```

```
##                V1       V2 V1 - sample mean V2 - sample mean
## crim     38.3518  67.9208         34.73828         64.30728
## zn        0.0000   0.0000        -11.36000        -11.36000
## indus    18.1000  18.1000          6.96000          6.96000
## chas      0.0000   0.0000         -0.06917         -0.06917
## nox       0.6930   0.6930          0.13830          0.13830
## rm        5.4530   5.6830         -0.83200         -0.60200
## age     100.0000 100.0000         31.43000         31.43000
## dis       1.4896   1.4254         -2.30540         -2.36960
## rad      24.0000  24.0000         14.45100         14.45100
```

```
## tax     666.0000 666.0000        257.80000        257.80000
## ptratio  20.2000  20.2000          1.74000          1.74000
## black   396.9000 384.9700         40.23000         28.30000
## lstat    30.5900  22.9800         17.94000         10.33000
## medv      5.0000   5.0000        -17.53000        -17.53000
```

h)

- 64 census tracts have on average more than 7 rooms per dwelling
- 13 census tracts have on average more than 8 rooms per dwelling

The census tracts that average more than eight rooms per dwelling have, on average, crim, indus, nox, dis, rad, tax, ptratio, and lstat values under the data set's mean, and values of other independent variables about the data set's mean.

```
Boston %>%
    filter(rm > 7) %>%
    summarise(`>7 rooms` = n())
```

```
##   >7 rooms
## 1       64
```

```
Boston %>%
    filter(rm > 8) %>%
    summarise(`>8 rooms` = n())
```

```
##   >8 rooms
## 1       13
```

```
# Mean of predictor values for tracts containing >8 rooms on average vs data
# set sample mean:
Boston %>%
    filter(rm > 8) %>%
    summarise_all(mean) %>%
    pivot_longer(everything(), names_to = "Names", values_to = "Values") %>%
    mutate(`Mean of values for rm>8 == true - sample mean` = Values - summaries$`Mean   `)
```

```
## # A tibble: 14 x 3
##    Names    Values `Mean of values for rm>8 == true - sample mean`
##    <chr>     <dbl>                                          <dbl>
##  1 crim      0.719                                         -2.89
##  2 zn       13.6                                            2.26
##  3 indus     7.08                                          -4.06
##  4 chas      0.154                                          0.0847
##  5 nox       0.539                                         -0.0155
```

```
##  6 rm        8.35                                          2.06
##  7 age      71.5                                           2.97
##  8 dis       3.43                                         -0.365
##  9 rad       7.46                                         -2.09
## 10 tax     325.                                          -83.1
## 11 ptratio  16.4                                          -2.10
## 12 black   385.                                           28.5
## 13 lstat     4.31                                         -8.34
## 14 medv     44.2                                          21.7
```

Session Info

```
sessionInfo()
```

```
## R version 4.1.2 (2021-11-01)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] MASS_7.3-54               PerformanceAnalytics_2.0.4
##  [3] xts_0.12.1                zoo_1.8-9
##  [5] gridExtra_2.3             forcats_0.5.1
##  [7] stringr_1.4.0             dplyr_1.0.7
##  [9] purrr_0.3.4               readr_2.1.0
## [11] tidyr_1.1.4               tibble_3.1.6
## [13] ggplot2_3.3.5             tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.7        lubridate_1.8.0   lattice_0.20-45   assertthat_0.2.1
##  [5] digest_0.6.28     utf8_1.2.2        R6_2.5.1          cellranger_1.1.0
##  [9] backports_1.4.0   reprex_2.0.1      evaluate_0.14     highr_0.9
## [13] httr_1.4.2        pillar_1.6.4      rlang_0.4.12      readxl_1.3.1
## [17] rstudioapi_0.13   Matrix_1.3-4      rmarkdown_2.11    splines_4.1.2
## [21] labeling_0.4.2    bit_4.0.4         munsell_0.5.0     broom_0.7.10
## [25] compiler_4.1.2    modelr_0.1.8      xfun_0.28         pkgconfig_2.0.3
```

```
## [29] mgcv_1.8-38      htmltools_0.5.2  tidyselect_1.1.1 quadprog_1.5-8
## [33] fansi_0.5.0      crayon_1.4.2     tzdb_0.2.0       dbplyr_2.1.1
## [37] withr_2.4.2      grid_4.1.2       nlme_3.1-153     jsonlite_1.7.2
## [41] gtable_0.3.0     lifecycle_1.0.1  DBI_1.1.1        magrittr_2.0.1
## [45] formatR_1.11     scales_1.1.1     cli_3.1.0        stringi_1.7.5
## [49] vroom_1.5.6      farver_2.1.0     fs_1.5.0         xml2_1.3.2
## [53] ellipsis_0.3.2   generics_0.1.1   vctrs_0.3.8      tools_4.1.2
## [57] bit64_4.0.5      glue_1.5.0       hms_1.1.1        parallel_4.1.2
## [61] fastmap_1.1.0    yaml_2.2.1       colorspace_2.0-2 rvest_1.0.2
## [65] knitr_1.36       haven_2.4.3
```