# ISLR | Chapter 2 Exercises

*Marshall McQuillen*

*11/13/2017*

## Conceptual

### 1

- **A**. Given the sample size n is extremely large and the number of predictors p is small, I would expect a flexible learning method to outperform an inflexible method. Since n is large, there wouldn't be a need to make any assumptions about f(x), which is the first step in an inflexible (parametric) learning method; the very size of n would give structure to f(x) and allowing the flexible method (non-parametric) to learn from the data without restrictions would produce a better model of the data, with little risk of overfitting due to large n.
- **B**. Given the numbers of predictors p is extremely high and the number of observations n is small, I would expect an inflexible method to outperform a flexible one. A flexible method runs the risk of modeling the noise in the data, leading to overfitting and a high test error rate. If some assumptions are made about the structure of f(x) (i.e. using a parametric/inflexible method), the risk of overfitting is lowered.
- **C**. Given the relationship between the predictors and the response is highly non-linear, I would expect a flexible method to outperform an inflexible method since the true f(x) is non-linear, and flexible learning methods don't make any assumptions about the f(x).
- **D**. Given that the variance of the error terms is extremely high, I would expect an inflexible method to outperform a flexible one. Without making any assumptions about f(x) (inflexible/parametric), a flexible method has a relatively high probability of modeling the variance of the error terms as a part of f(x), whereas an inflexible method would reduce that probability, at least somewhat.

### 2

- **A**. Since the question is interested in understanding which factors affect CEO salary, which is a quantitative metric, this would be a regression scenario and the word "understanding" lends towards an interest in inference, as opposed to prediction. n = 500 (firms) and p = 3 (profit, number of employees and industry).
- **B**. Determining whether a new product will be a success or failure is a classification problem, and since the response is binary, logistic regression would be a good starting (Although logistic regression has the word regression in it, due to the output of f(x) being quantitative, the output is a probability of an observation being classified as a success, therefore the method is ultimately used for classification problems). n = 20 (similar products) and p = 13 (price charged for product, marketing budget, competition price and 10 other variables).
- **C**. Interest in prediction, and since metric of interest is quantitative, this would be a regression problem. n = 52 (weeks in 2012) and p = 3 (% change in the US market, % change in the British market and the % change in the German market.)

### 3

- **A**.

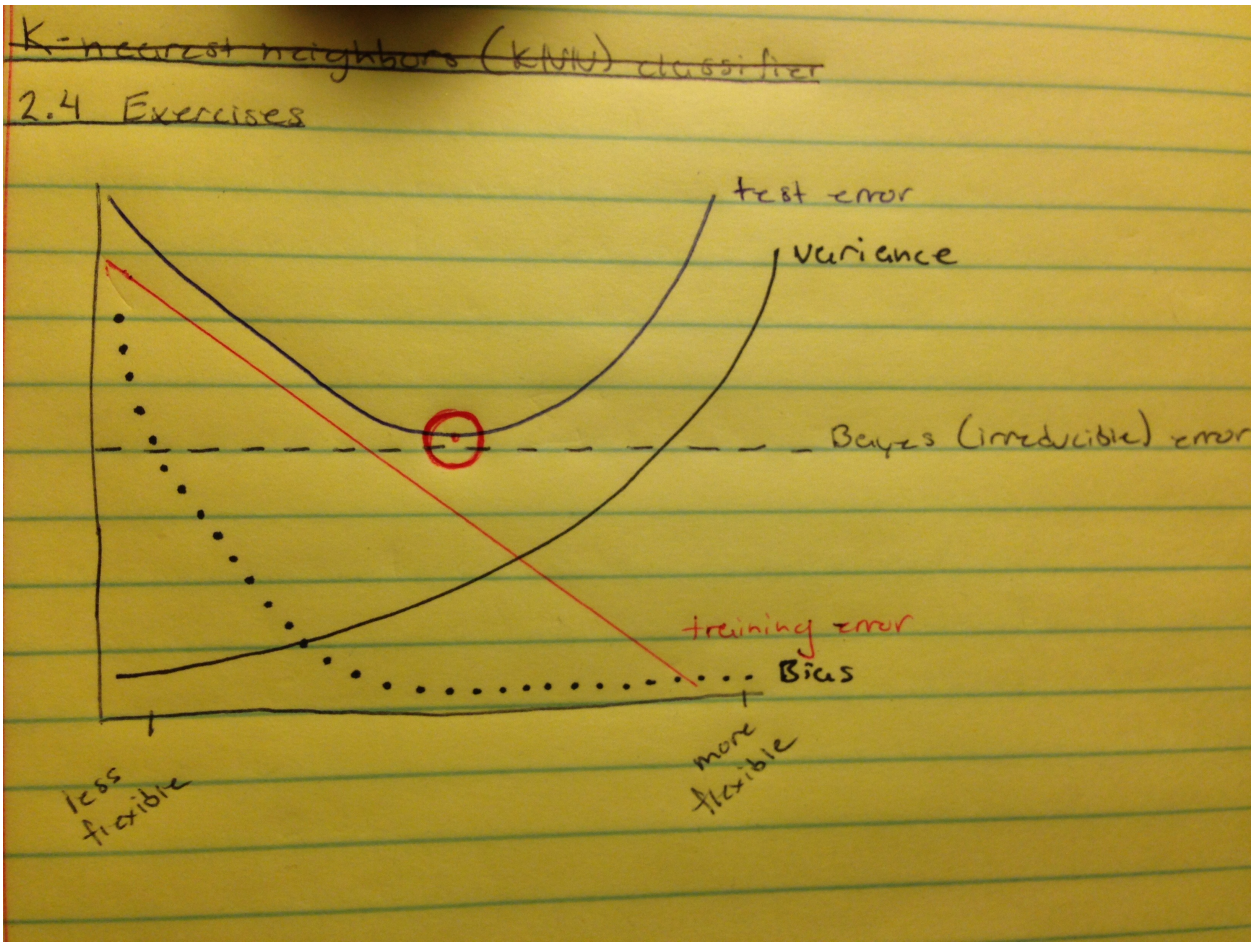

K-nearest neighbors (KNN) classifier
2.4 Exercises
test error
variance
Bayes (irreducible) error
training error
Bias
less flexible
more flexible

Figure 1:

- **B**.
- *The Bayes error curve* (dashed black line) will always be a flat line because it represents the irreducible error; even the best model cannot account for the natural variance of the error terms, hence the horizontal line.
- *The training error curve* (solid red line) has downward sloping curve because, as more flexible learning methods are used, the method can always "explain away" all of the variance in the training data.
- *The test error curve* (solid blue line) has a U-shape to it because, as more flexible methods are used, the test error is reduced *up until the point (red circle)* where the method begins to model the noise in the training data, which will lead to higher variance and therefore a higher test error rate. Note that the perfect method and model **cannot** go below the Bayes error curve.
- *The (squared) bias curve* (dotted black line) has a downward sloping curve. Bias is defined as the error that occurs by attempting to model a real life relationship mathematically. So, the simpler (less flexible) the method, the more bias will be introduced (e.g. the probability that any real life relationship is *truly* linear is low). The more flexible the method, the closer the estimate of f(x) will be to the true f(x). (given a large enough sample size)
- *The variance curve* (solid black line) has is upward sloping because as the flexiblility of the learning method increases, the more the data affect the estimate of f(x), therefore changing the data would change the estimate of f(x) drastically. On the other hand, using the least flexible method possible (horizontal line) as your way of estimating f(x), would have zero variance.

## 4

- **A**. Classification would be useful when …
- *Situation 1* … you would want to determine which treatment method a new cancer patient would respond the best too. The response would be whether the patient went into remission and the predictors would include variables such as age, prevalence of cancer in family history, diet, genetic predispositions, etc (certainly not an easy task). The goal would be prediction, since you theoretically already know the patient has cancer and you would just like to predict the treatment for which they would respond the best.
- *Situation 2* … you would want to determine whether a potential investment property would be a success or failure (Defining what a "successful" investment property is would be the important part). Given that we define a "successful" investment property as one that makes $10,000/year, the response would be whether other investment properties met that criteria, coded as 1 for success or 0 for failure, and the predictors could be location, rent per month, HOA fees, proximity to parks/malls, etc. Since the ultimate goal would be to decide whether you wanted to invest in the property or not, this would be a prediction problem.
- *Situation 3* … you want to see how different demographic variables affect someone's voting tendencies. Variables could include age, state of current residence, birth state, voting tendencies of parents, income, married, etc. The responce would be Democrate or Republican. Since the end goal is to see *how* each of the variables affect someone's voting tendencies, this would be an inference problem.
- **B**. Regression would be useful when …
- *Situation 1* … you would like to understand how variables affect a customers ecommerce order amount. The response would be the total amount their "cart" was worth at the time of checkout and predictors could be advertising budget, date, the items in their cart and time spent at each webpage. The goal would be inference so as to *understand* how each variable affects the response.
- *Situation 2* … you would like to predict the next price of a certain stock. Predictors would include the last price, current assets, current liabilities, NCAVPS, prevalence of company in news (to name a few)

and the response would be the stocks price. Since the goal is to determine the next price, this would be a prediction problem.

- *Situation 3* ... you want to see how various health habits affect a person resting heart rate. Variables could include diet, time spent exercising per week, type of exercise, age, activity level at job and the reponse would be their resting heart rate. Sinc predicting someone's resting heart rate isn't of that much importantance, this would be an inference problem; you are looking to understand *how* each variable affects the response.
- **C**. Cluster analysis would be useful when ...
- *Situation 1* ... a company would like to see what similarities their customers have with one another. Clustering could be performed on location, income level, education, etc.
- *Situation 2* ... clinical researchers would like to see what similarities patients with Alzheimer's disease have.
- *Situation 3* ... the authors of "An Introduction to Statistical Learning" want to see the various education levels of the people that bought their book.

## 5

The advantages of a flexible learning method over a less flexible learning method are that flexible learning methods are typically going to be less biased than inflexible method (holding all else equal). However, with this comes the disadvantage that more flexible method are also prone to more variance. In addition, flexible methods are usually much less interpretable than inflexible methods, so if inference is the goal, an inflexible method might be a better option. If prediction is the goal and it doesn't particularly matter if you understand how each variable affects the outcome, a more flexible method will most likely be more accurate.

## 6

The main difference between parametric and non-parametric methods is that parametric methods make some assumption about the general shape of f(x). The advantages of this are that, give that the structure is defined, the challenge of estimating f(x) is reduced to one of estimating the parameters/coefficients of each regressor. On the other hand, non-parametric models make no assumption about the structure of f(x). For both regression and classification implementations of non-parametric methods, a large sample size is needed to reduce the chance of overfitting, so when n is small, a parametric approach is probably the better option.

## 7

- **A**.

```
# create data frame
obs <- c(1,2,3,4,5,6)
x1 <- c(0,2,0,0,-1,1)
x2 <- c(3,0,1,1,0,1)
x3 <- c(0,0,3,2,1,1)
y <- c('red', 'red', 'red', 'green' ,'green', 'red')
df <- data.frame(Obs = obs, v1 = x1, v2 = x2, v3 = x3, response = y)

suppressPackageStartupMessages(library(dplyr))

x <- (df$v1 - 0)^2
y1 <- (df$v2 - 0)^2
```

```
z <- (df$v3 - 0)^2
df1 <- as.data.frame(cbind(x,y1,z))
df2 <- mutate(df1, total = rowSums(df1))
df3 <- mutate(df2, Distance = round(sqrt(total), digits = 2))
df3 <- cbind(obs, df3, y)
arrange(df3, Distance)
```

```
##   obs x y1 z total Distance     y
## 1   5 1  0 1     2     1.41 green
## 2   6 1  1 1     3     1.73   red
## 3   2 4  0 0     4     2.00   red
## 4   4 0  1 4     5     2.24 green
## 5   1 0  9 0     9     3.00   red
## 6   3 0  1 9    10     3.16   red
```

- **B**. When K = 1, the prediction for Y would be green because the "nearest neighbor" to the test point (at a distance of 1.41) is observation 5, which is green.
- **C**. When K = 3, the 3 "nearest neighbors" to the test point are green, red and red. Therefore, P("Green" | X = 0) = 0.33 and P("Red" | X = 0) = 0.66 and the test point is assigned the classification with the highest probability, red.
- **D**. If the Bayes Decision Boundary is highly non-linear, then you could expect the best (most accurate) value for K to be small. The reason for this is as K increases, the Bayes Decision Boundary becomes less and less flexible (i.e. more and more linear) because, as the classifier takes more data points into consideration, each data point has less leverage on the decision boundary.

# Applied

## 8

- **A & B**. (As opposed to reading the data in with read.csv(), I'm going to install the ISLR package and reference the data manually; the row names are already assigned)

```
# install packages
library(ISLR)
head(College)
```

```
##                              Private Apps Accept Enroll Top10perc
## Abilene Christian University     Yes 1660   1232    721        23
## Adelphi University               Yes 2186   1924    512        16
## Adrian College                   Yes 1428   1097    336        22
## Agnes Scott College              Yes  417    349    137        60
## Alaska Pacific University        Yes  193    146     55        16
## Albertson College                Yes  587    479    158        38
##                              Top25perc F.Undergrad P.Undergrad Outstate
## Abilene Christian University        52        2885         537     7440
## Adelphi University                  29        2683        1227    12280
## Adrian College                      50        1036          99    11250
## Agnes Scott College                 89         510          63    12960
## Alaska Pacific University           44         249         869     7560
## Albertson College                   62         678          41    13500
##                              Room.Board Books Personal PhD Terminal
## Abilene Christian University       3300   450     2200  70       78
```

```
## Adelphi University                  6450   750     1500  29       30
## Adrian College                      3750   400     1165  53       66
## Agnes Scott College                 5450   450      875  92       97
## Alaska Pacific University           4120   800     1500  76       72
## Albertson College                   3335   500      675  67       73
##                                S.F.Ratio perc.alumni Expend Grad.Rate
## Abilene Christian University       18.1          12   7041        60
## Adelphi University                 12.2          16  10527        56
## Adrian College                     12.9          30   8735        54
## Agnes Scott College                 7.7          37  19016        59
## Alaska Pacific University          11.9           2  10922        15
## Albertson College                   9.4          11   9727        55
```

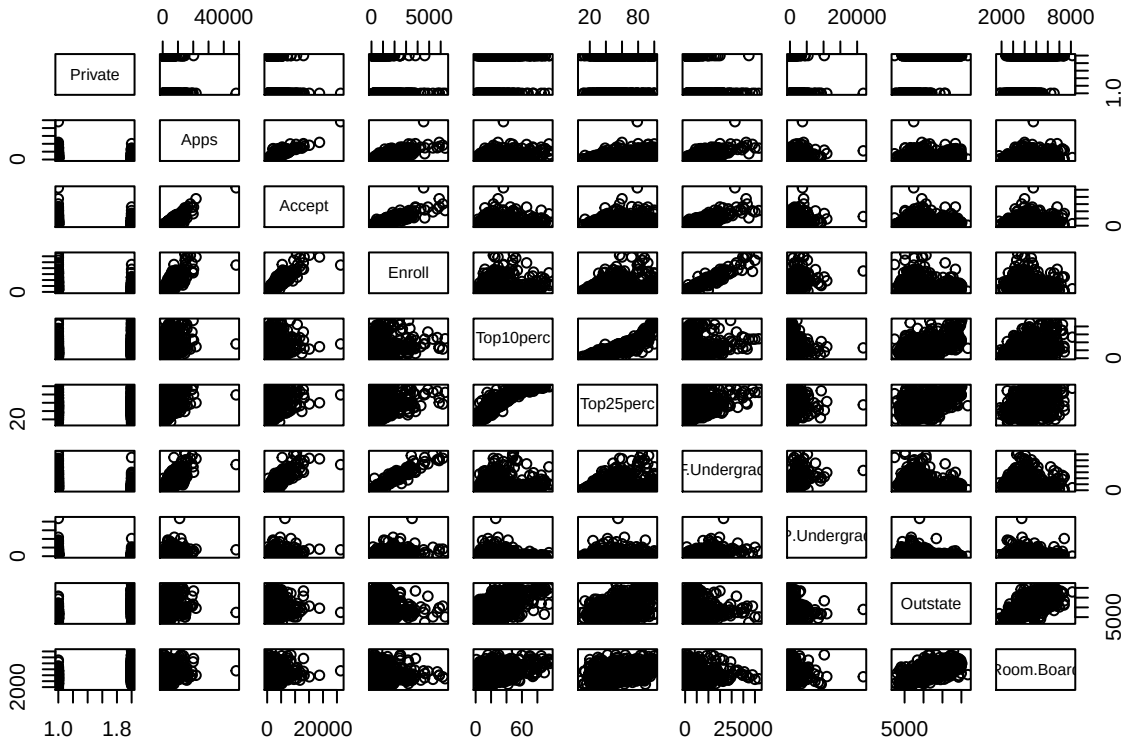
- **C**.
- *C-1*.

```
summary(College)
```

```
##  Private        Apps           Accept          Enroll       Top10perc
##  No :212   Min.   :   81   Min.   :   72   Min.   :  35   Min.   : 1.00
##  Yes:565   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242   1st Qu.:15.00
##            Median : 1558   Median : 1110   Median : 434   Median :23.00
##            Mean   : 3002   Mean   : 2019   Mean   : 780   Mean   :27.56
##            3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902   3rd Qu.:35.00
##            Max.   :48094   Max.   :26330   Max.   :6392   Max.   :96.00
##    Top25perc      F.Undergrad     P.Undergrad         Outstate
##  Min.   :  9.0   Min.   :  139   Min.   :    1.0   Min.   : 2340
##  1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0   1st Qu.: 7320
##  Median : 54.0   Median : 1707   Median :  353.0   Median : 9990
##  Mean   : 55.8   Mean   : 3700   Mean   :  855.3   Mean   :10441
##  3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0   3rd Qu.:12925
##  Max.   :100.0   Max.   :31643   Max.   :21836.0   Max.   :21700
##    Room.Board       Books           Personal         PhD
##  Min.   :1780   Min.   :  96.0   Min.   : 250   Min.   :  8.00
##  1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850   1st Qu.: 62.00
##  Median :4200   Median : 500.0   Median :1200   Median : 75.00
##  Mean   :4358   Mean   : 549.4   Mean   :1341   Mean   : 72.66
##  3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700   3rd Qu.: 85.00
##  Max.   :8124   Max.   :2340.0   Max.   :6800   Max.   :103.00
##     Terminal       S.F.Ratio      perc.alumni        Expend
##  Min.   : 24.0   Min.   : 2.50   Min.   : 0.00   Min.   : 3186
##  1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00   1st Qu.: 6751
##  Median : 82.0   Median :13.60   Median :21.00   Median : 8377
##  Mean   : 79.7   Mean   :14.09   Mean   :22.74   Mean   : 9660
##  3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00   3rd Qu.:10830
##  Max.   :100.0   Max.   :39.80   Max.   :64.00   Max.   :56233
##    Grad.Rate
##  Min.   : 10.00
##  1st Qu.: 53.00
##  Median : 65.00
##  Mean   : 65.46
##  3rd Qu.: 78.00
##  Max.   :118.00
```

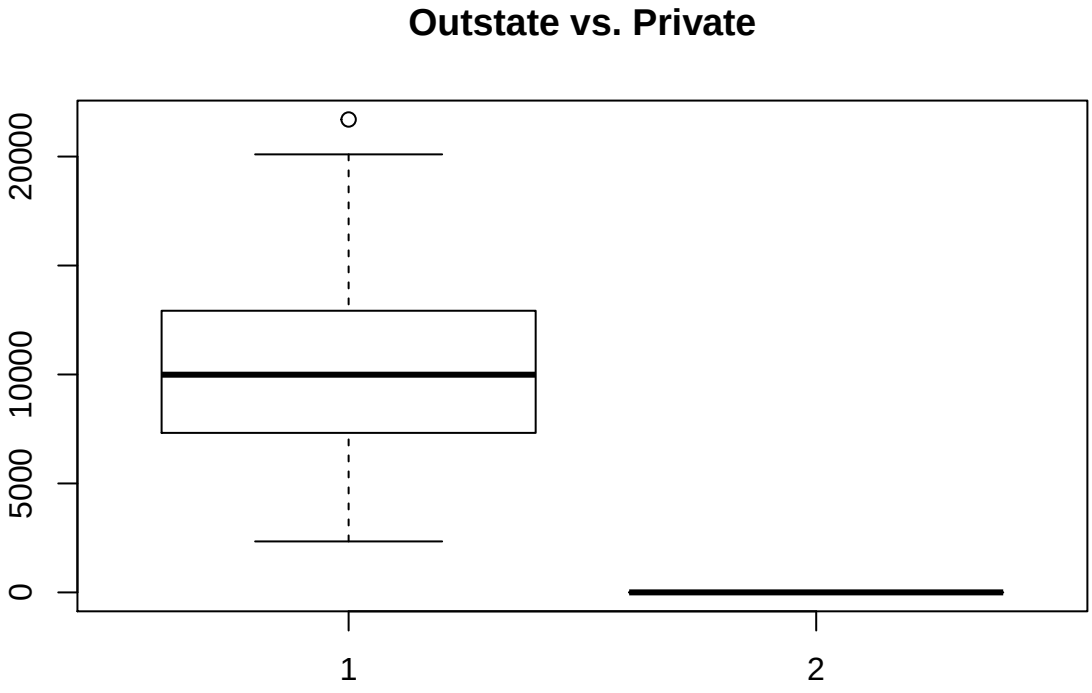
- *C-2*.

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




- *C-3.*

```
with(College, boxplot(Outstate, Private))
title("Outstate vs. Private")
```

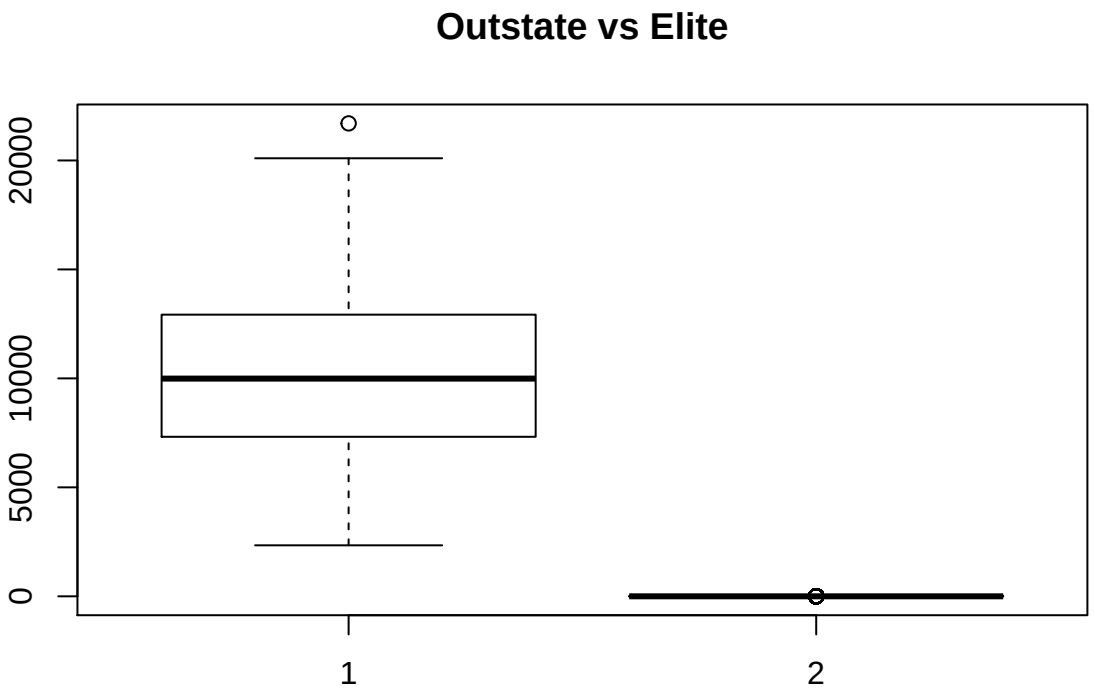



- *C-4.*

```r
# repeat the string "No" 777 times (number of rows in College data set)
Elite <- rep("No", nrow(College))
# for every observation in College, if the value in the Top10perc is greater than 50, assign "Yes" to t
Elite[College$Top10perc > 50] <- "Yes"
Elite <- as.factor(Elite)
College <-  data.frame(College, Elite)
summary(College$Elite)
```

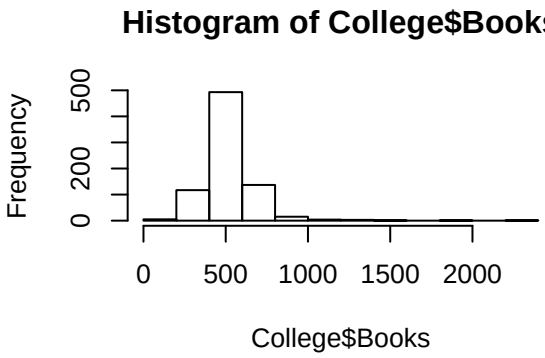
```
##  No Yes
## 699  78
```

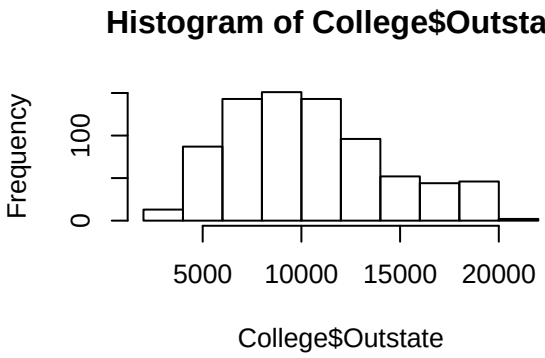
```r
with(College, boxplot(Outstate, Elite))
title("Outstate vs Elite")
```

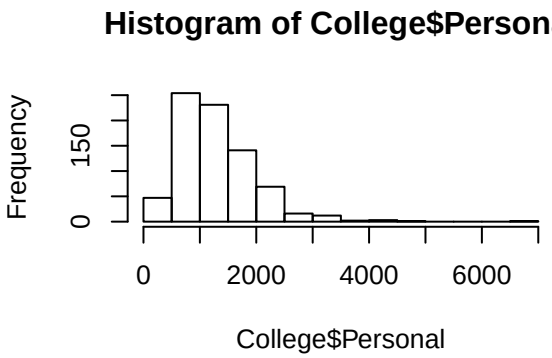


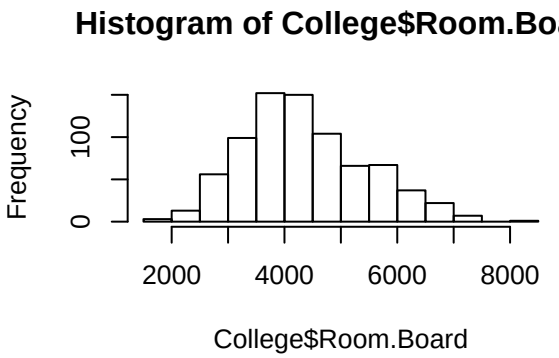

- *C-5.*

```r
par(mfrow = c(2,2))
hist(College$Books)
hist(College$Outstate)
hist(College$Personal)
hist(College$Room.Board)
```



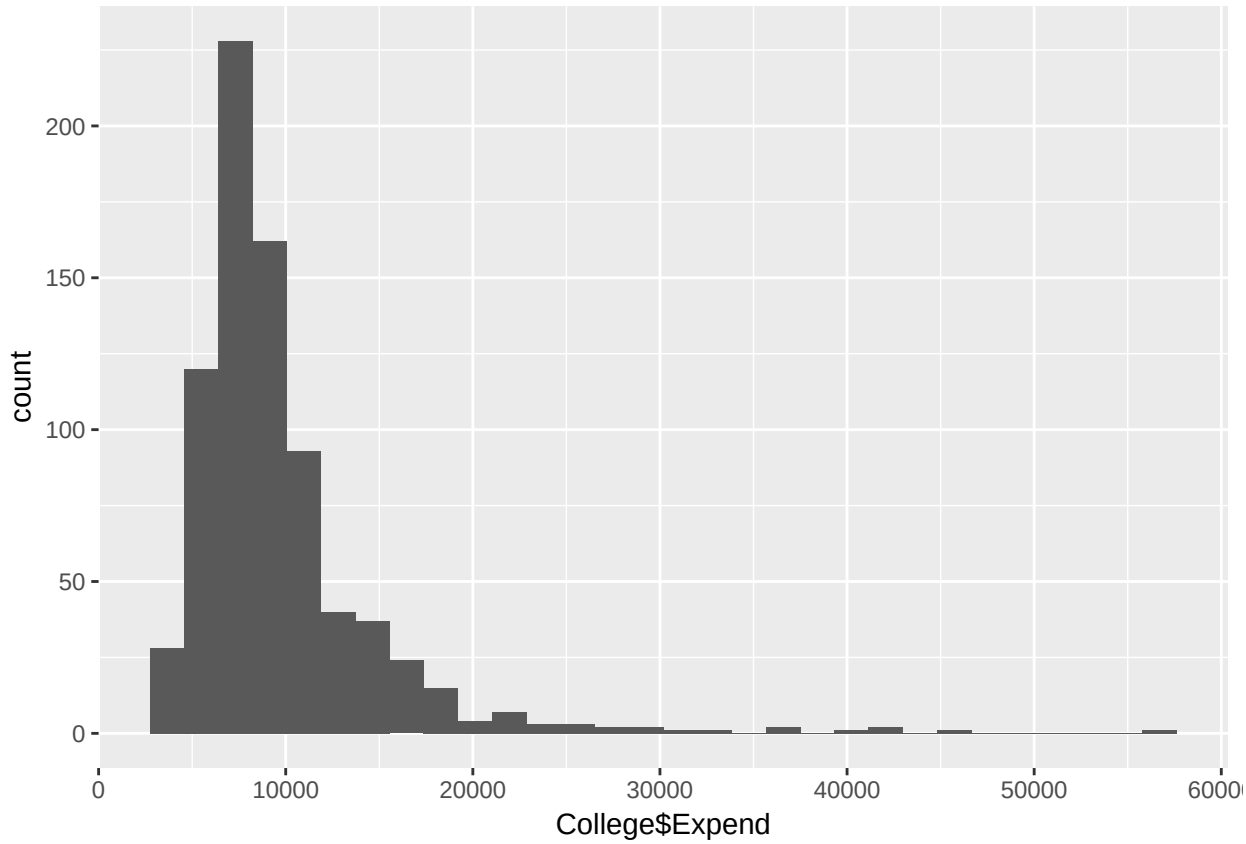



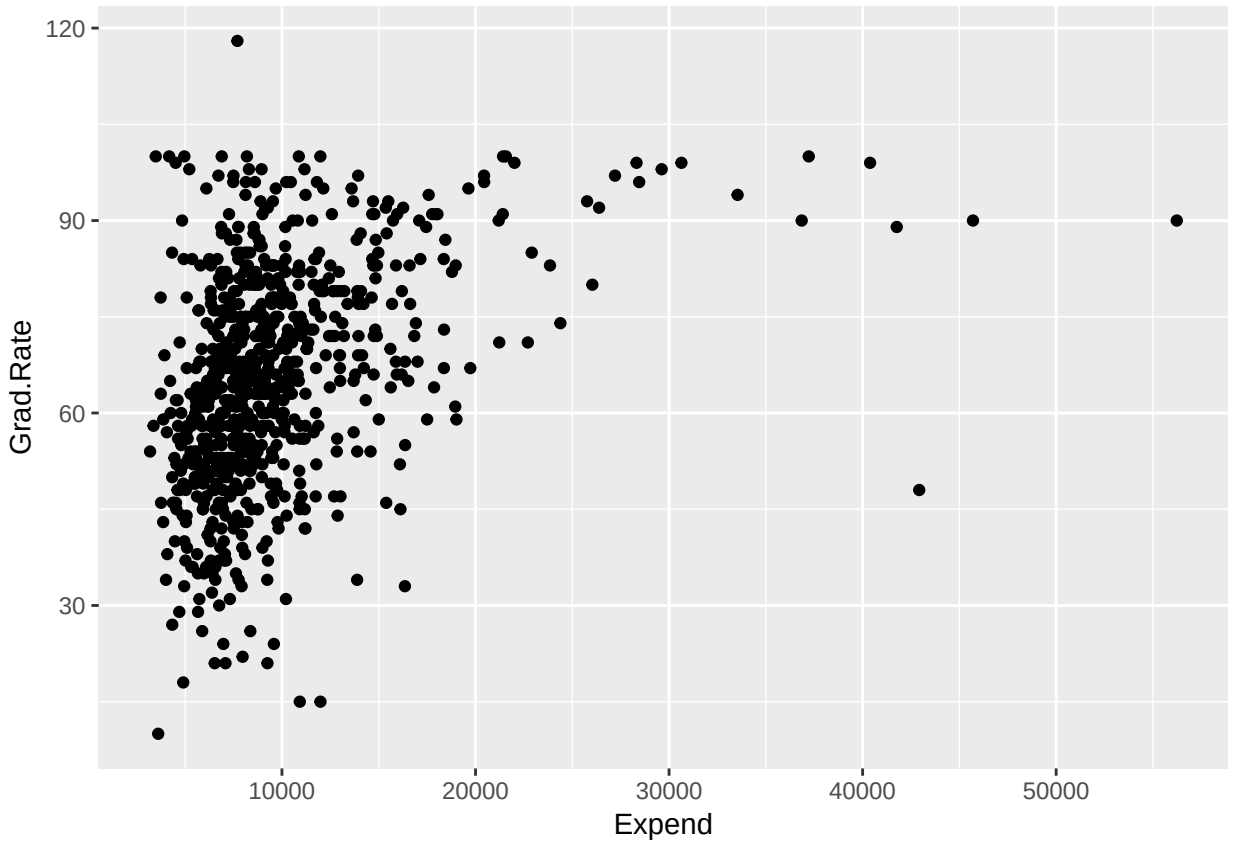

- *C-6.*

```
library(ggplot2)
qplot(College$Expend)
```

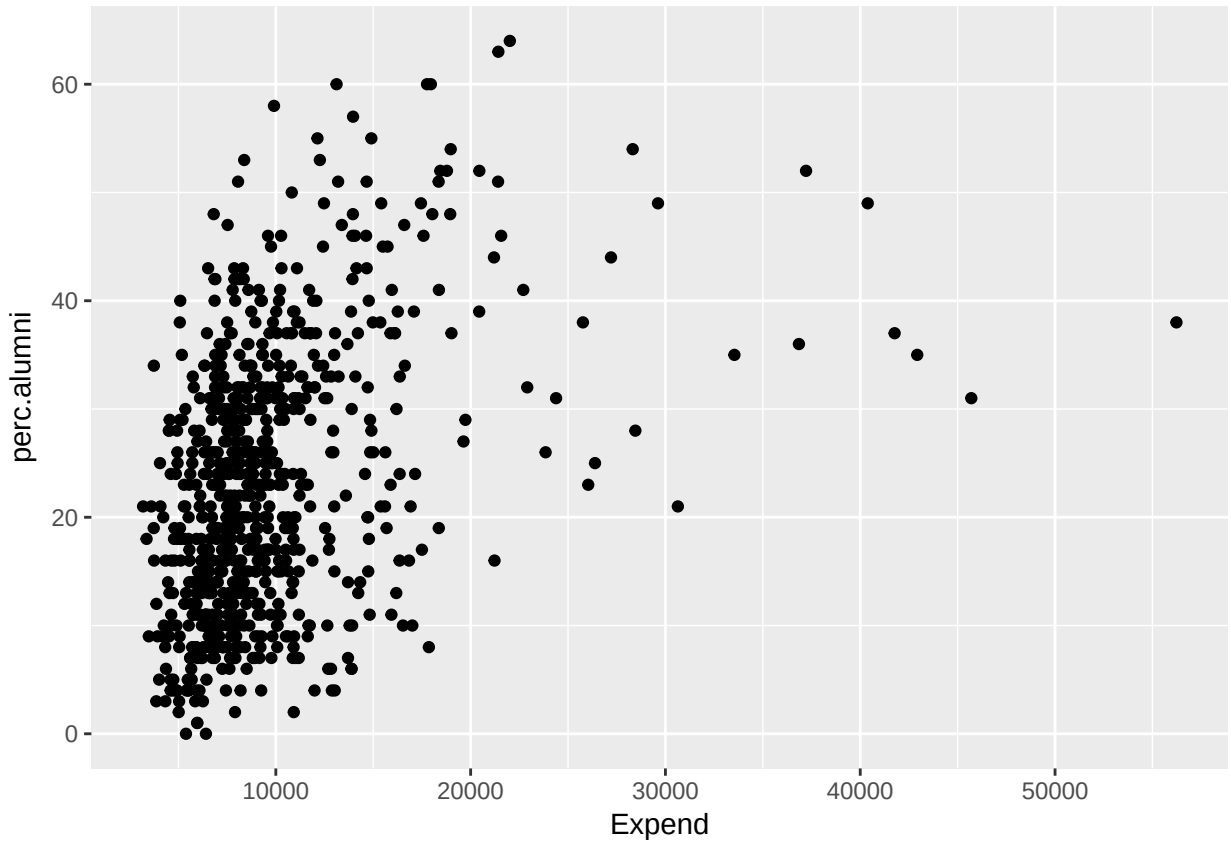
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
qplot(y = Grad.Rate, x = Expend, data = College)
```




```
qplot(y = perc.alumni, x = Expend, data = College)
```




A couple conclusions can be drawn from the above graphs:

1. The average amount spent on each student by universities is roughly $9,500
2. Although there are few universities in the data set that do spend more than $25,000 per student, those that do have graduation rates at or above 90% (excluding one outlier)
3. There seems to be a positive relationship betwwen the percent of alumni that donate to the university and the instructional expenditure per student. (Not a strong relationship however)

## 9

```
sum(is.na(Auto))
```

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

```
str(Auto)
```

```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : num  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : num  3504 3693 3436 3433 3449 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : num  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ name        : Factor w/ 304 levels "amc ambassador brougham",..: 49 36 231 14 161 141 54 223 241 1
```

- **A**.

- *Quantitative* mpg, cylinders, displacement, horsepower, weight, acceleration, year
- *Qualitative* origin, name
- **B**.

```
lapply(Auto[1:7], range)
```

```
## $mpg
## [1]  9.0 46.6
##
## $cylinders
## [1] 3 8
##
## $displacement
## [1]  68 455
##
## $horsepower
## [1]  46 230
##
## $weight
## [1] 1613 5140
##
## $acceleration
## [1]  8.0 24.8
##
## $year
## [1] 70 82
```

- **C**.

```
lapply(Auto[c(1,2,3,4,5,6,7)], mean)
```

```
## $mpg
## [1] 23.44592
##
## $cylinders
## [1] 5.471939
##
## $displacement
## [1] 194.412
##
## $horsepower
## [1] 104.4694
##
## $weight
## [1] 2977.584
##
## $acceleration
## [1] 15.54133
##
## $year
## [1] 75.97959
```

```
lapply(Auto[c(1,2,3,4,5,6,7)], sd)
```

```
## $mpg
## [1] 7.805007
```

```
##
## $cylinders
## [1] 1.705783
##
## $displacement
## [1] 104.644
##
## $horsepower
## [1] 38.49116
##
## $weight
## [1] 849.4026
##
## $acceleration
## [1] 2.758864
##
## $year
## [1] 3.683737
```

- **D**.

```
df <- Auto[-c(10:85),]
lapply(df[1:7], range)
```

```
## $mpg
## [1] 11.0 46.6
##
## $cylinders
## [1] 3 8
##
## $displacement
## [1]  68 455
##
## $horsepower
## [1]  46 230
##
## $weight
## [1] 1649 4997
##
## $acceleration
## [1]  8.5 24.8
##
## $year
## [1] 70 82
```

```
lapply(df[c(1,2,3,4,5,6,7)], mean)
```

```
## $mpg
## [1] 24.40443
##
## $cylinders
## [1] 5.373418
##
## $displacement
## [1] 187.2405
##
```

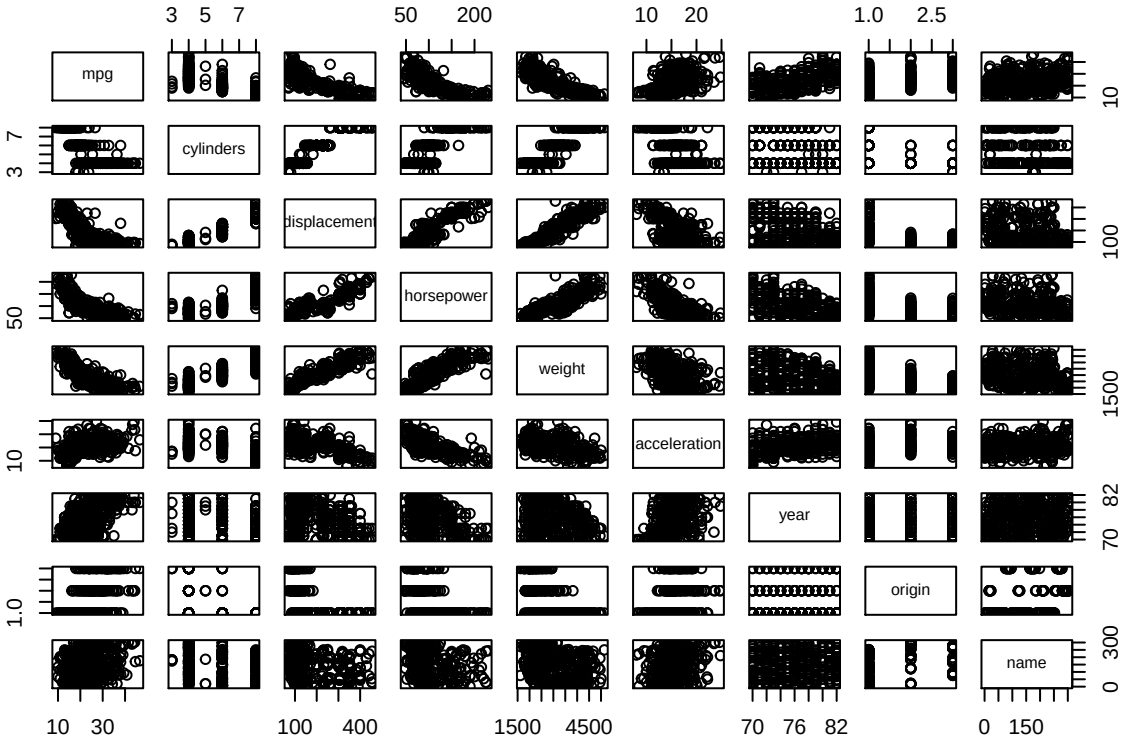
```
## $horsepower
## [1] 100.7215
##
## $weight
## [1] 2935.972
##
## $acceleration
## [1] 15.7269
##
## $year
## [1] 77.14557
```

```
lapply(df[c(1,2,3,4,5,6,7)], sd)
```

```
## $mpg
## [1] 7.867283
##
## $cylinders
## [1] 1.654179
##
## $displacement
## [1] 99.67837
##
## $horsepower
## [1] 35.70885
##
## $weight
## [1] 811.3002
##
## $acceleration
## [1] 2.693721
##
## $year
## [1] 3.106217
```

- **E**.

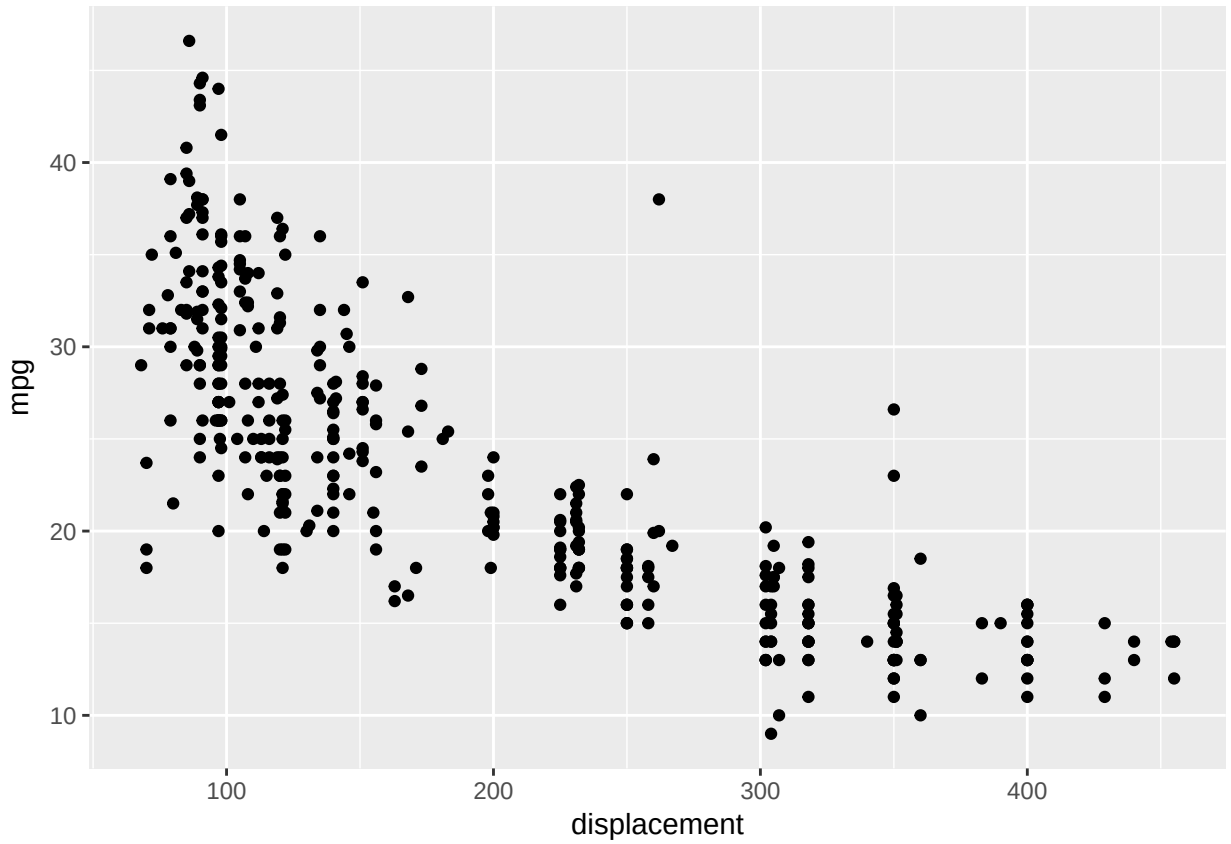
```
pairs(Auto)
```



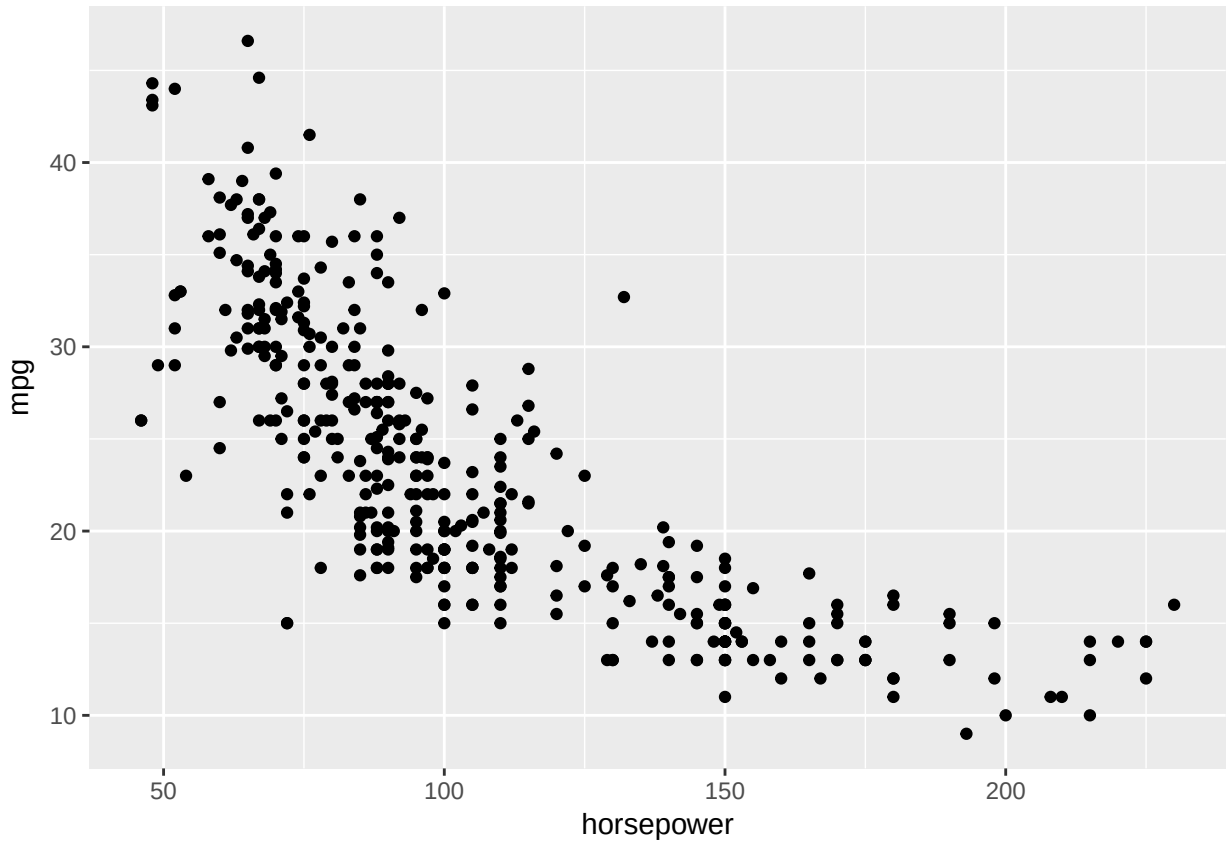

Although using the pairs function doesn't creat the most visually appealing, it does allow quick views of how the relationship between each variable and all the other. Most notably, there seem so be a negative linear relationship between mpg and displacement, horsepower and weight. Conversely, as one would expect those thee variable seem to have positive linear relationships with one another.

- **F**. As stated in my previous answer, it seems as though displacement, horsepower and weight all have negative linear relationships with mpg. We can take a closer look by zooming in on how those specific variable relate to mpg. Creating a rough linear model of mpg as predicted by the other three variables (displacement, horsepower and weight) returns a model that explains 70% of the variance in mpg.

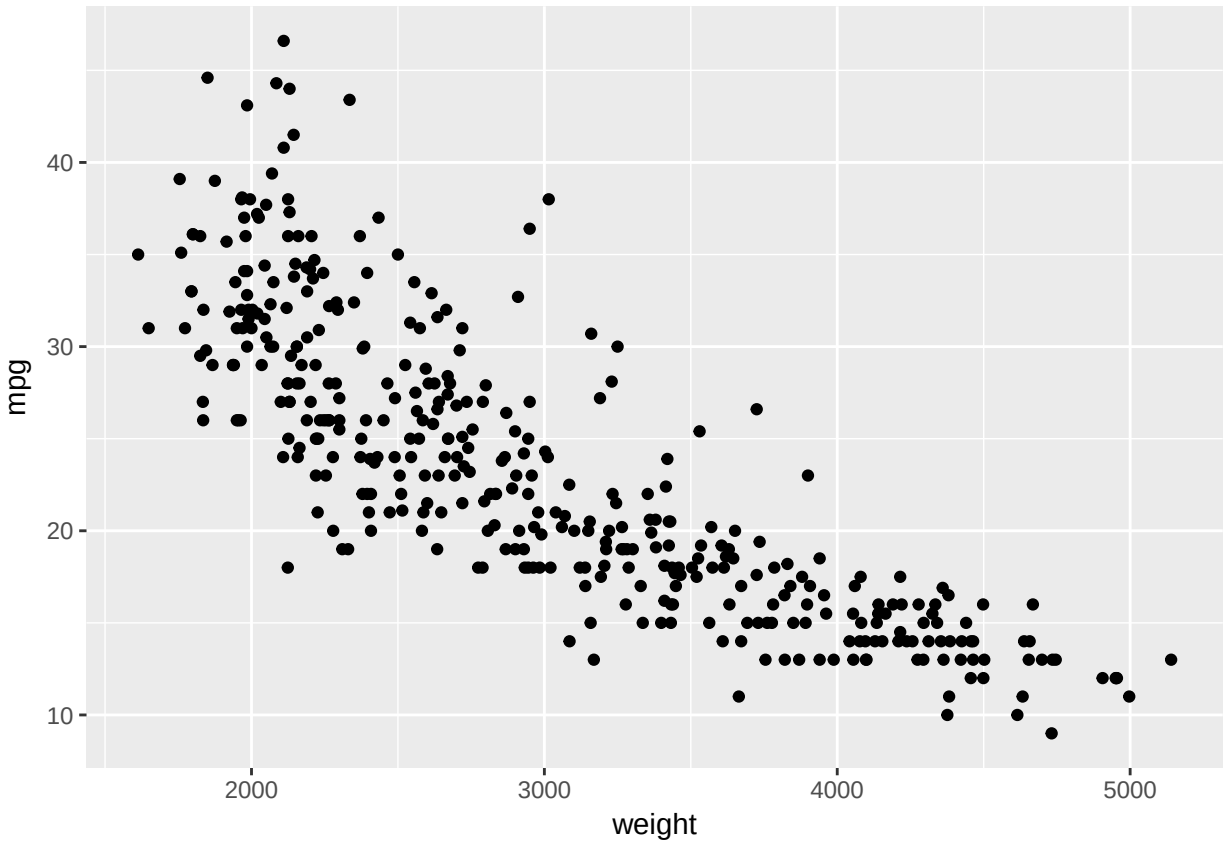
```
qplot(y = mpg, x = displacement, data = Auto)
```




```
qplot(y = mpg, x = horsepower, data = Auto)
```




```
qplot(y = mpg, x = weight, data = Auto)
```




```
fit <- lm(mpg ~ displacement + horsepower + weight, data = Auto)
summary(fit)
```

```
##
## Call:
## lm(formula = mpg ~ displacement + horsepower + weight, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3347  -2.8028  -0.3402   2.2037  16.2409
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  44.8559357  1.1959200  37.507  < 2e-16 ***
## displacement -0.0057688  0.0065819  -0.876  0.38132
## horsepower   -0.0416741  0.0128139  -3.252  0.00125 **
## weight       -0.0053516  0.0007124  -7.513 4.04e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.241 on 388 degrees of freedom
## Multiple R-squared:  0.707,  Adjusted R-squared:  0.7047
## F-statistic:   312 on 3 and 388 DF,  p-value: < 2.2e-16
```

## 10

- **A**.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

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

```
dim(Boston)
```

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

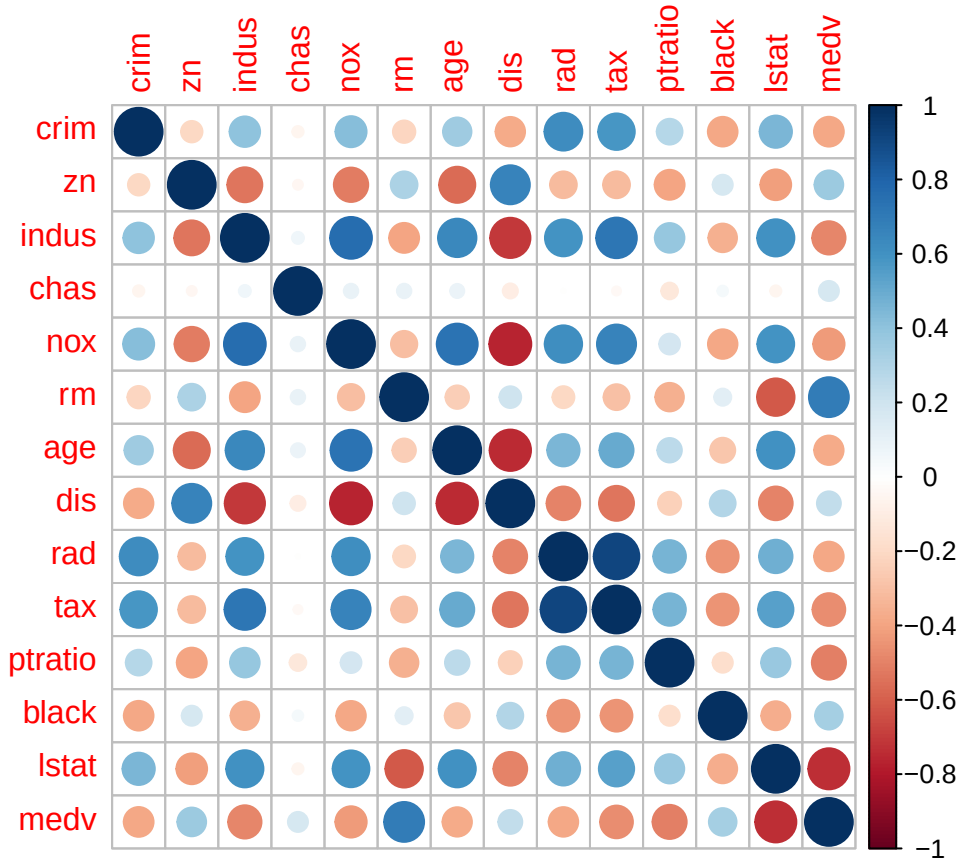
There are 506 observations (suburbs in Boston) and 14 columns, representing various attributes of the suburbs. (e.g. the variable "crim" is the per capita crime rate by town)

- **B**.

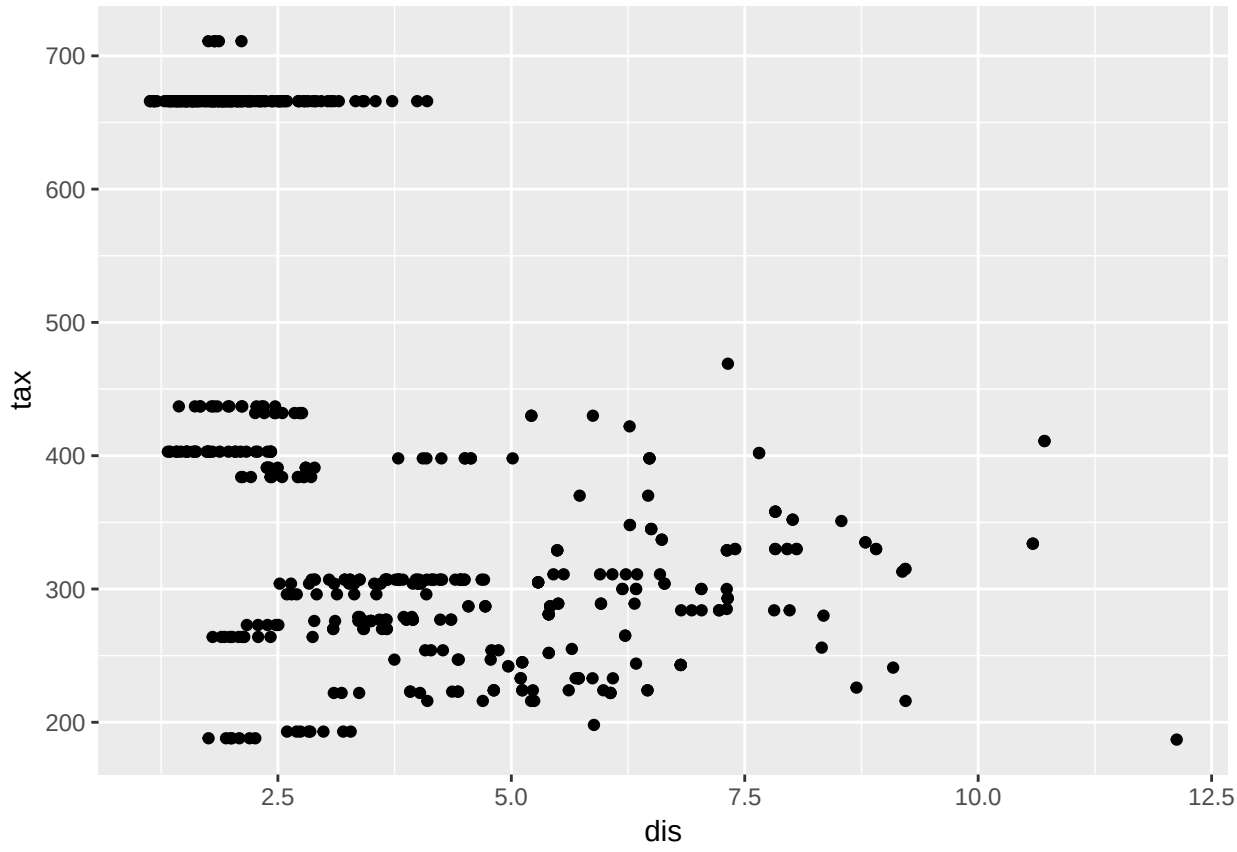
```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

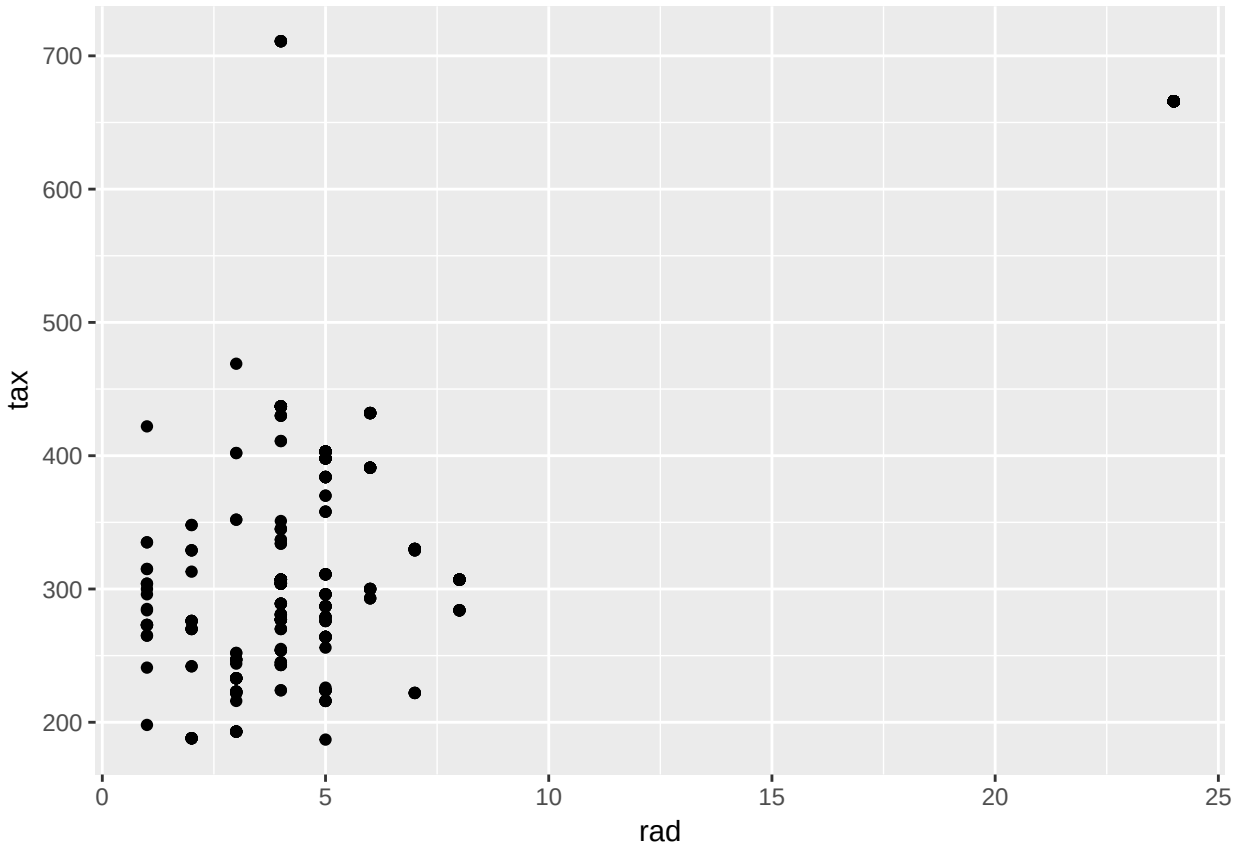
```
cors <- cor(Boston)
corrplot(cors)
```




```
qplot(dis, tax, data = Boston)
```



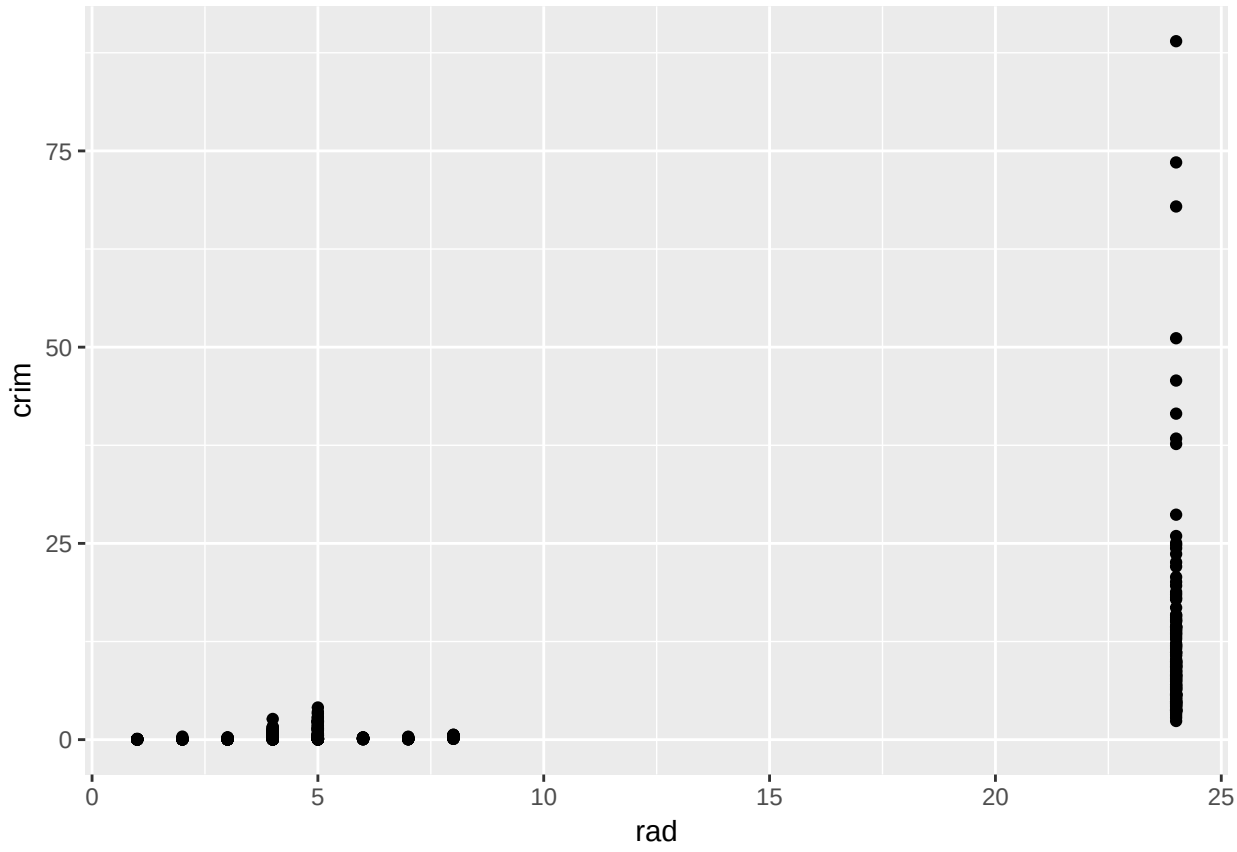

```
qplot(rad, tax, data = Boston)
```



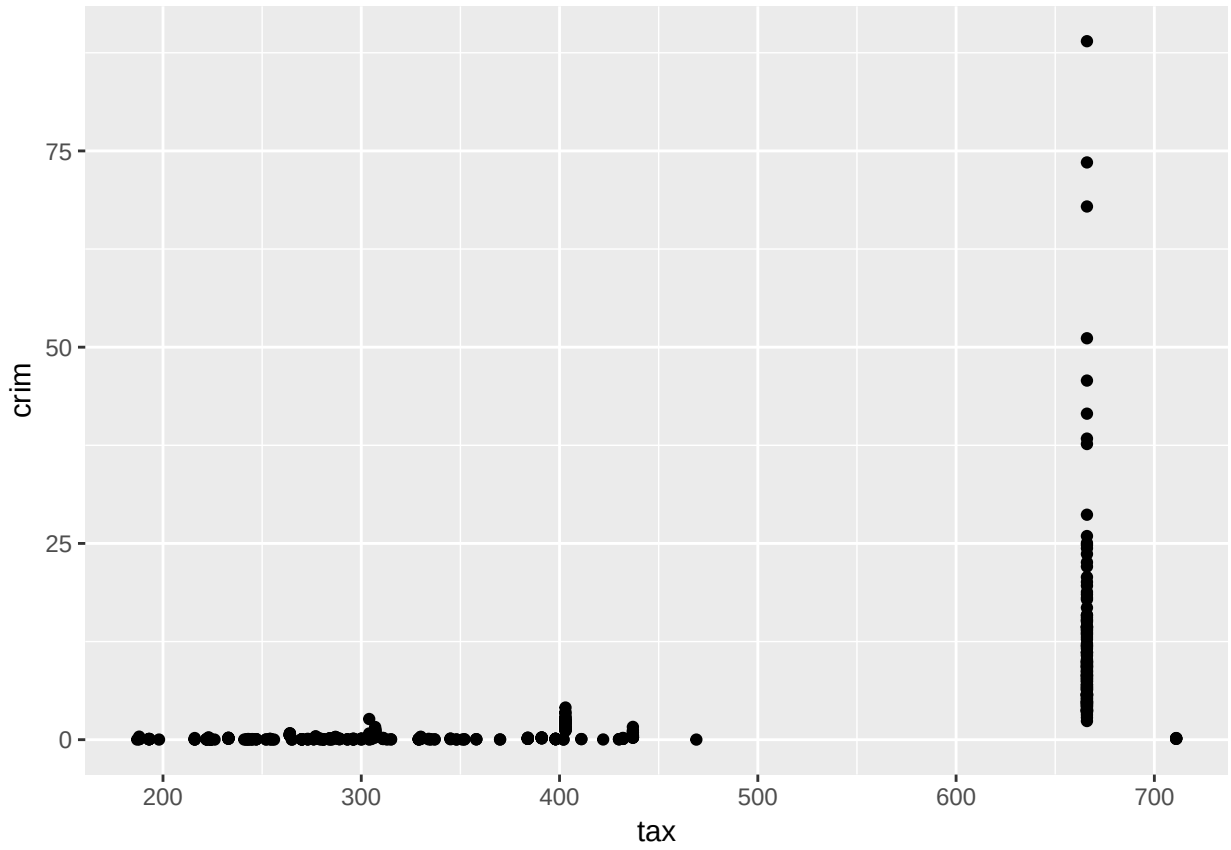

Looking at the correlation matrix , the tax and dis variables have a relatively negative correlation and tax and rad (index of accessibility to radial highways) are the most positively correlated variables in the data set. Looking closer at these two relationships, both have data points with high leverage.

- **C**. Referencing the correlation matrix, the rad and tax variables are the ones that are the most correlated with per capita crime rate (crim).

```
qplot(rad, crim, data = Boston)
```



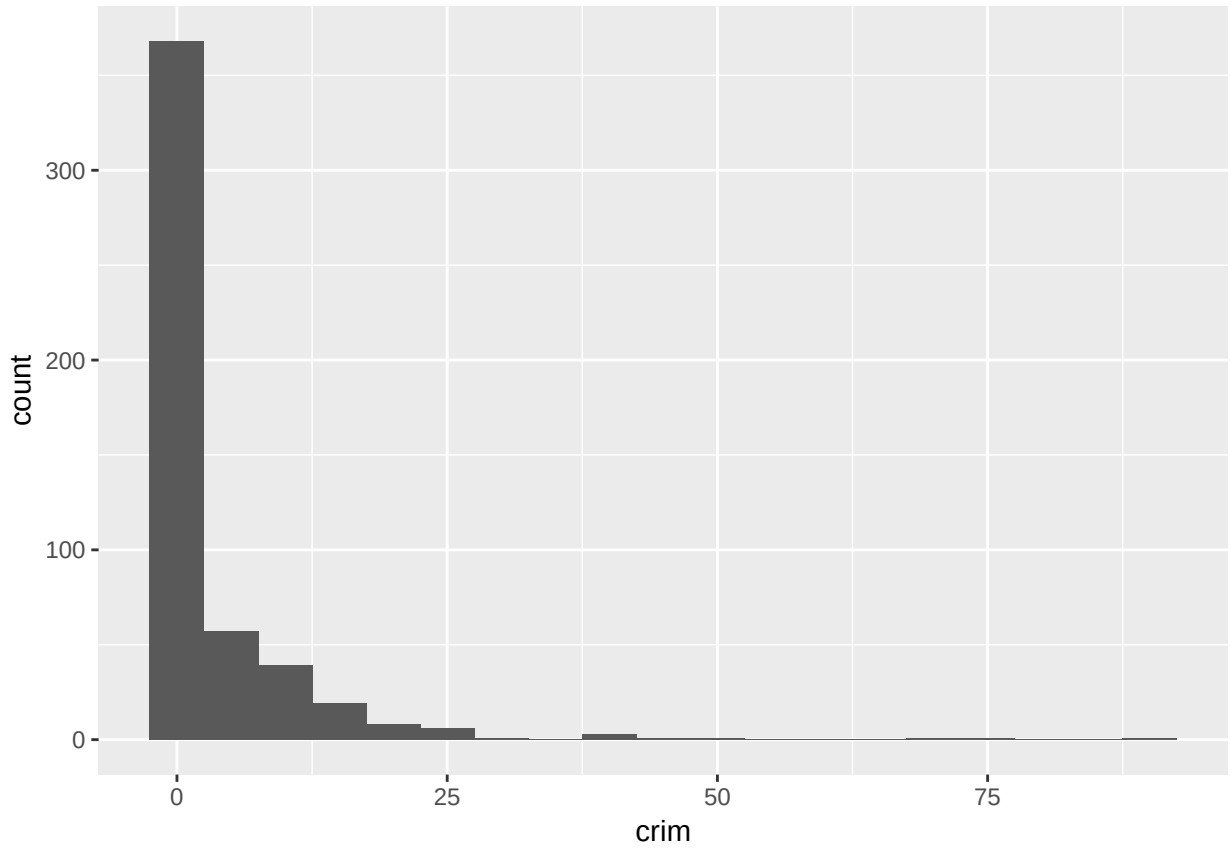

```
qplot(tax, crim, data = Boston)
```




Both of these plots have the same basic shape in that as rad/tax increase to close to their maximum values, the distribution of crim is becomes much more spread out. Given any value for tax (rad) up to roughly 450 (8), crim deviates very little. However, once the tax (rad) approach their maximum values, the values for crim have a much higher variance.

- **D**. It seem that the vast majority of suburbs in Boston have low per capita crime rates. However, their is a long tail on the histogram, indicating that certain suburbs have very high crime rates. Subsetting the data to only those suburbs with crime rates greater than 30, one commonality that jumps out is the age variable; most of the units in these suburbs are older (built before 1940).

```
# high crime rates
qplot(crim, data = Boston, binwidth = 5)
```

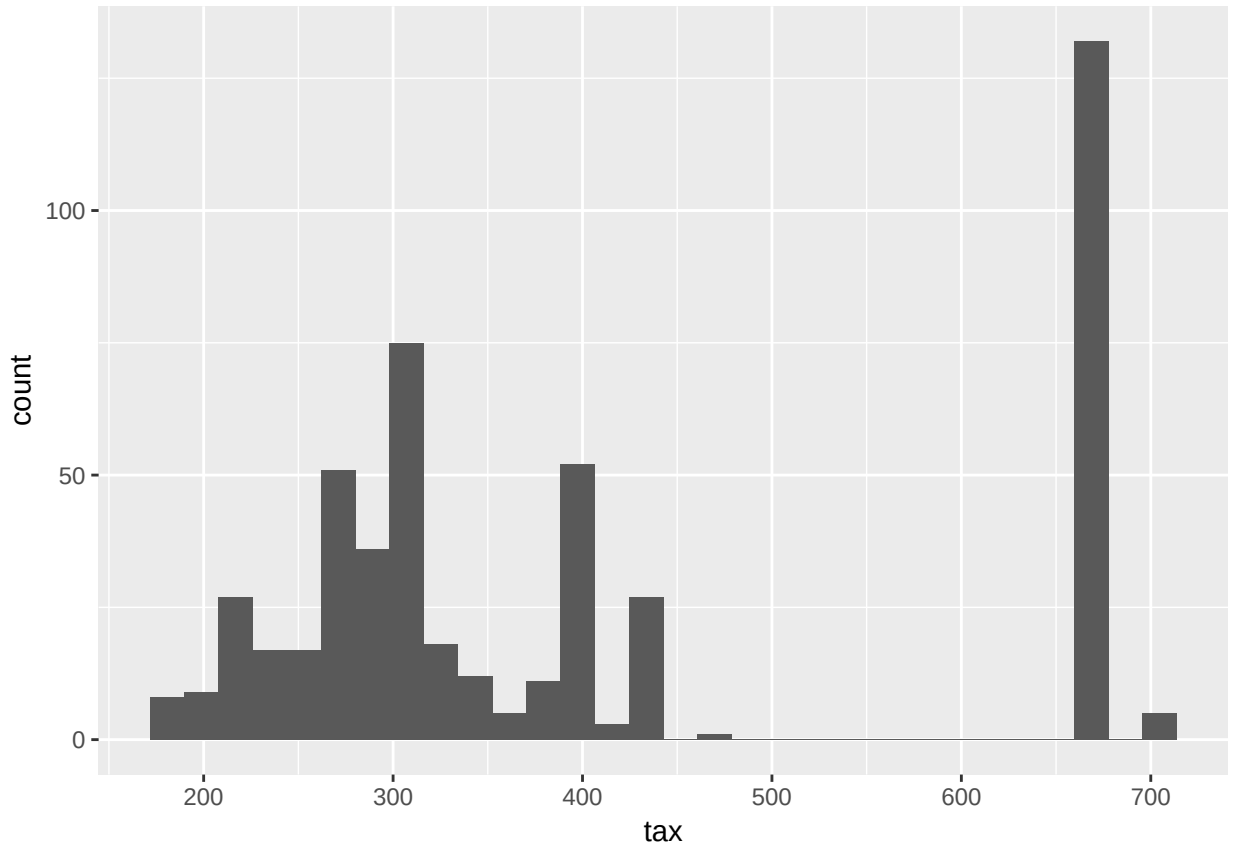



```
subset(Boston, crim > 30)
```

```
##        crim zn indus chas   nox    rm   age    dis rad tax ptratio  black
## 381 88.9762  0  18.1    0 0.671 6.968  91.9 1.4165  24 666    20.2 396.90
## 399 38.3518  0  18.1    0 0.693 5.453 100.0 1.4896  24 666    20.2 396.90
## 405 41.5292  0  18.1    0 0.693 5.531  85.4 1.6074  24 666    20.2 329.46
## 406 67.9208  0  18.1    0 0.693 5.683 100.0 1.4254  24 666    20.2 384.97
## 411 51.1358  0  18.1    0 0.597 5.757 100.0 1.4130  24 666    20.2   2.60
## 415 45.7461  0  18.1    0 0.693 4.519 100.0 1.6582  24 666    20.2  88.27
## 419 73.5341  0  18.1    0 0.679 5.957 100.0 1.8026  24 666    20.2  16.45
## 428 37.6619  0  18.1    0 0.679 6.202  78.7 1.8629  24 666    20.2  18.82
##     lstat medv
## 381 17.21 10.4
## 399 30.59  5.0
## 405 27.38  8.5
## 406 22.98  5.0
## 411 10.11 15.0
## 415 36.98  7.0
## 419 20.62  8.8
## 428 14.52 10.9
```

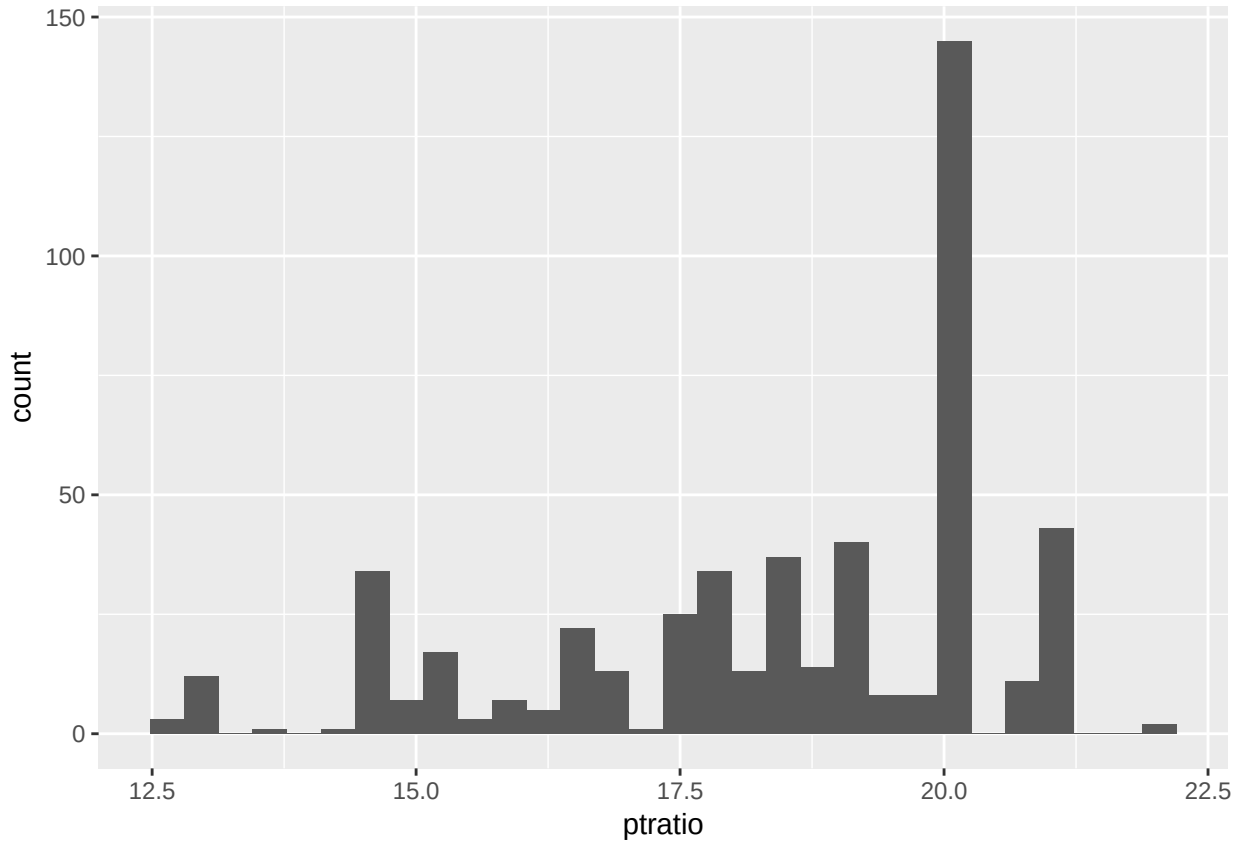
```
# high tax rates
qplot(tax, data = Boston)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# pupil-teacher ratio
qplot(ptratio, data = Boston)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




While crime and tax rates have a unique distribution, the pupil-teacher ratio distribution doesn't have any exteme tails.

- **E**. 35 suburbs bound the Charles River.

```
table(Boston$chas)
```

```
##
##   0   1
## 471  35
```

- **F**.

```
median(Boston$ptratio)
```

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

- **G**. Suburbs 399 and 406 have the lowest median value of owner occupied homes. Both of these suburbs are in the upper extremes of most of the other variables as well; all units in both suburbs were build before 1940, both have crime rates that are in the tail of the distribution, have high black populations, high levels of nitrogen oxide.

```
subset(Boston, medv == min(Boston$medv))
```

```
##        crim zn indus chas   nox    rm age    dis rad tax ptratio  black
## 399 38.3518  0  18.1    0 0.693 5.453 100 1.4896  24 666    20.2 396.90
## 406 67.9208  0  18.1    0 0.693 5.683 100 1.4254  24 666    20.2 384.97
##     lstat medv
## 399 30.59    5
## 406 22.98    5
```

- **H**. 64 suburbs average more than 7 rooms per dwelling and 13 average more than 8 per dwelling. All of the suburbs that average more than 8 rooms per dwelling have mostly older buildings, and, most notably, have the highest median value of owner-occupied homes in the data set.

```
nrow(subset(Boston, rm > 7))
```

```
## [1] 64
```

```
nrow(subset(Boston, rm > 8))
```

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

```
subset(Boston, rm > 8)
```

```
##        crim zn indus chas    nox    rm  age    dis rad tax ptratio  black
## 98  0.12083  0  2.89    0 0.4450 8.069 76.0 3.4952   2 276    18.0 396.90
## 164 1.51902  0 19.58    1 0.6050 8.375 93.9 2.1620   5 403    14.7 388.45
## 205 0.02009 95  2.68    0 0.4161 8.034 31.9 5.1180   4 224    14.7 390.55
## 225 0.31533  0  6.20    0 0.5040 8.266 78.3 2.8944   8 307    17.4 385.05
## 226 0.52693  0  6.20    0 0.5040 8.725 83.0 2.8944   8 307    17.4 382.00
## 227 0.38214  0  6.20    0 0.5040 8.040 86.5 3.2157   8 307    17.4 387.38
## 233 0.57529  0  6.20    0 0.5070 8.337 73.3 3.8384   8 307    17.4 385.91
## 234 0.33147  0  6.20    0 0.5070 8.247 70.4 3.6519   8 307    17.4 378.95
## 254 0.36894 22  5.86    0 0.4310 8.259  8.4 8.9067   7 330    19.1 396.90
## 258 0.61154 20  3.97    0 0.6470 8.704 86.9 1.8010   5 264    13.0 389.70
## 263 0.52014 20  3.97    0 0.6470 8.398 91.5 2.2885   5 264    13.0 386.86
## 268 0.57834 20  3.97    0 0.5750 8.297 67.0 2.4216   5 264    13.0 384.54
## 365 3.47428  0 18.10    1 0.7180 8.780 82.9 1.9047  24 666    20.2 354.55
##     lstat medv
## 98   4.21 38.7
## 164  3.32 50.0
## 205  2.88 50.0
## 225  4.14 44.8
## 226  4.63 50.0
## 227  3.13 37.6
## 233  2.47 41.7
## 234  3.95 48.3
## 254  3.54 42.8
## 258  5.12 50.0
## 263  5.91 48.8
## 268  7.44 50.0
## 365  5.29 21.9
```