Chapter 2 Exercises

Aidan O'Keeffe

2023-03-06

Conceptual

Exercise 1

Problem For each of parts (a) through (d), indicate whether we would generally expect the performance of a flexible statistical learning method to be better or worse than an inflexible model. Justify your answer.

a) The sample size n is extremely large, and the number of predictors p is small.
b) The number of predictors p is extremely large, and the number of observations n is small.
c) The relationship between the predictors and response is highly nonlinear.
d) The variance of the error terms, i.e. $\sigma^2 = \text{Var}(\epsilon)$, is extremely high.

Solution

a) A flexible model would likely perform much better than an inflexible model for two reasons. First, the relationship between the predictors and the response becomes more and more clear as the sample size grows, and since such relationships are almost always nonlinear, a flexible method could capitalize on the abundance of data to find such relationships. Second, because the number of predictors is small, we are in a low dimensional setting, meaning that we do not suffer from the curse of dimensionality, hence we could even expect extremely flexible nonparametric models to show good performance.
b) In this case, we would expect better performance from a less flexible model. Given a small n, as p increases, overfitting becomes inevitable, which would lead to a high test error rate. A less flexible model could identify a broad trend from what data *is* available while not fitting the sample too closely.
c) If the relationship between the predictors and the response is highly nonlinear, then a flexible method will perform better because it will provide a smaller model bias than a less flexible method.
d) The problem with high-variance errors is that they obscure the true response-predictor relationship. In this case, a flexible method will be fit to the errors, almost certainly leading to poor test performance. As with scenario (b), we'd be better off using a less flexible model and identifying a broad trend in the data, thereby not fitting the noise too closely.

Exercise 2

Problem

Explain whether each scenario is a classification or regression problem, and indicate whether we are more interested in inference or prediction. Finally, provide n and p.

a) We collect a set of data on the top 500 firms in the US. For each firm, we record profit, number of employees, industry, and the CEO salary. We are interested in understanding which factors affect CEO salary.

b) We are considering launching a new product and wish to know whether it will be a *success* or *failure*. We collect data on 20 similar products that were previously launched. For each product, we have recorded whether it was a success or failure, price charged for the product, marketing budget, competition price, and ten other variables.
c) We are interested in predicting the % change in the USD/Euro exchange rate in relation to the weekly changes in the world stock markets. Hence, we collect weekly data for all of 2012. For each week, we record the % change in the USD/Euro, the % change in the US market, the % change in the British market, and the % change in the German market.

Solution

a) This is a regression problem because CEO salary is a continuous quantity, and the interest is in inference, as indicated by the phrase "we are interested in understanding". $n = 500$, $p = 3$.
b) This is a classification problem because the response is a discrete quantity, *success* or *failure*. The interest here is in prediction because we want to know what the outcome will be for this particular "new product" we're launching. $n = 20$, $p = 13$.
c) This is a regression problem because % change is a continuous quantity. The problem tells us outright that it's one of prediction. $n = 52$ (the number of weeks in a year), and $p = 3$.

Exercise 3

Problem

We now revisit the bias-variance decomposition.

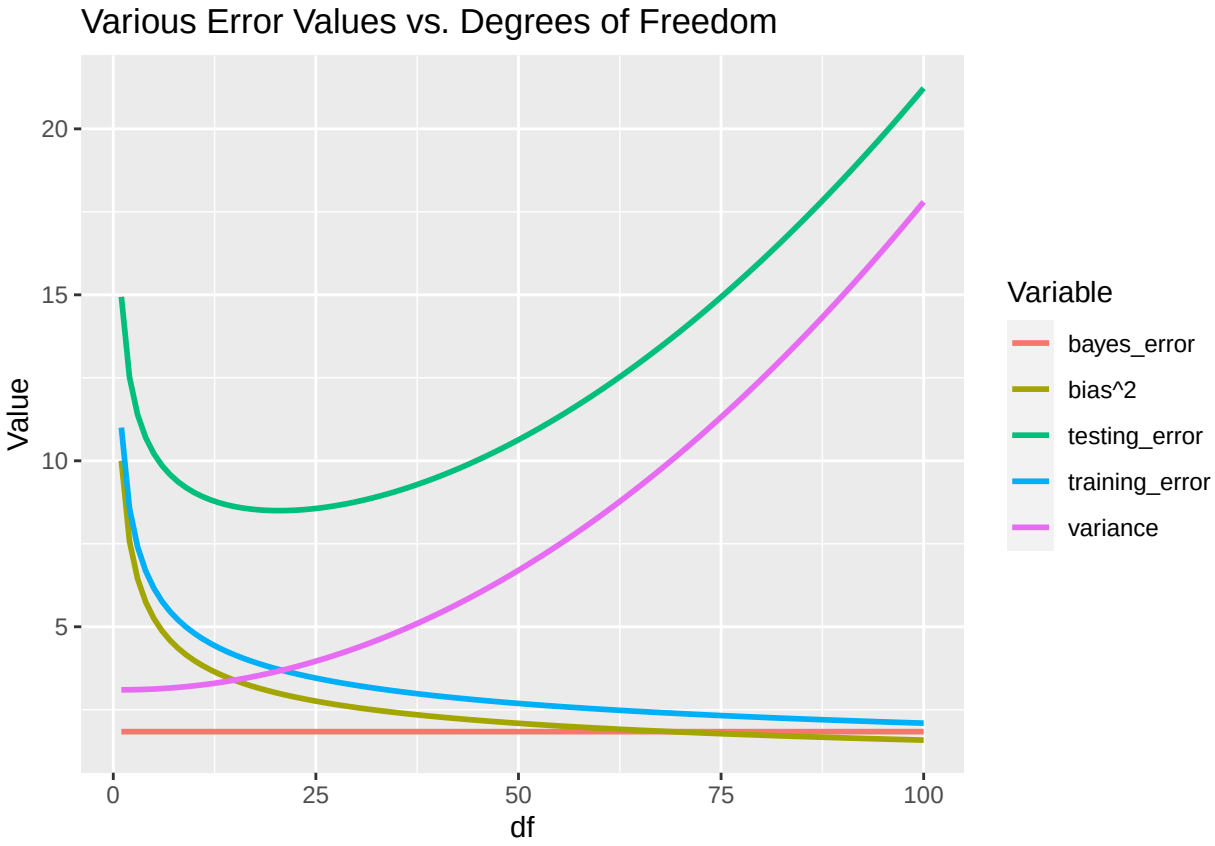
a) Provide a sketch of typical (squared) bias, variance, training error, test error, and Bayes (or irreducible) error curves, on a single plot, as we go from less flexible statistical learning methods towards more flexible approaches. The x-axis should represent the amount of flexibility in the method, and the y-axis should represent the values for each curve. There should be five curves. Make sure to label each one.
b) Explain why each of the five curves has the shape displayed in part (a).

Solution

a) The code block below graphs the five curves in question. Note that the specific parameter values and functional forms don't matter, they were chosen simply because they produce the correct general shape.

```
library(tidyverse)

# Generate the curves to be plotted
df = 1:100
bias2 = 10/(df^0.4)
variance = 0.006*((df-1)/2)^2 + 3.1
train = 11/(df^0.36)
bayes = rep(1.84, 100)
test = bias2 + variance + bayes

data = data.frame( rep(1:100, 5),
                   rep(c("bias^2", "variance", "training_error", "testing_error", "bayes_error"),
                       each=100)  )
data = data.frame(data, c(bias2,variance,train,test,bayes  ))
```

```r
names(data) = c("df", "Variable", "Value")

# Plot them
plt <- ggplot(data=data, aes(x=df, y=Value, group=Variable))+
  geom_line(aes(color=Variable), linewidth=1)+
  labs(x="df", y="Value")+
  ggtitle("Various Error Values vs. Degrees of Freedom")

plt
```



b) The Bayes error is a constant because it is defined as the variance of the error terms, that is, $\text{Var}(\epsilon)$, a quantity which does not vary with the degrees of freedom. The squared bias is represented as a decreasing function that converges to 0, which is the case because as a model becomes more flexible, its form is less constrained, and so the error due to model bias approaches 0. The testing error (in an ideal setting) is represented as the sum of the squared bias, the variance, and the Bayes error, which is justified by the bias-variance decomposition. The training error is represented as another decreasing function that converges to 0, which is the case because as model flexibility increases, it becomes possible to overfit, even to the extreme case of interpolating the data points. Finally, the variance is an increasing function of flexibility. The text doesn't give a detailed argument as to why this is the case, but nonetheless says it is so.

Exercise 4

Problem

You will now think of some real life applications for statistical learning.

a) Describe three real-life applications in which *classification* might be useful. Describe the response, as well as the predictors. Is the goal of each application inference or prediction? Explain your answer.
b) Describe three real-life applications in which *regression* might be useful. Describe the response, as well as the predictors. Is the goal of each application inference or prediction? Explain your answer.
c) Describe three real-life applications in which *cluster analysis* might be useful.

Solution

a) i. We may want to understand (infer) if a student will pass or fail a class on the basis of their sex, age, GPA, and major.
   ii. We may want to predict if a political candidate will win or lose an election on the basis of their sex, age, political party, state, fundraising, and if they are the incumbent or not.
   iii. We may want to predict if a car will sell or not on the basis of make, model, year, and mileage.

b) i. We may want to relate (infer) the mean delivery time of packets on a computer network on the basis of the number of nodes, traffic over time, and an assortment of measures of network structure.
   ii. We may want to predict the change in sales on the basis of radio, TV, and newspaper advertising budgets and location data.
   iii. We may want to predict body fat as a function of neck, waist, and hip measurements, weight, height, and biological sex.

c) i. We want to identify online shoppers who are likely to buy a certain product by finding those who have bought many items with the same tags/product descriptors as the one in question.
   ii. We want to advertise a fuel rewards program by identifying drivers who drive many miles, use a diesel or gas engine, and what kind of vehicle they drive.
   iii. We want to identify residents of a neighborhood who are using many resources, such as water, electricity, and gas.

Exercise 5

Problem

What are the advantages and disadvantages of a very flexible (versus less flexible) approach for regression or classification? Under what circumstances might a more flexible approach be preferred to a less flexible approach? When might a less flexible approach be preferred?

Solution

Focusing on regression, flexible methods are capable of producing more accurate predictions than less flexible methods. They perform well when we have a large sample size, when the variance of the error terms is small, and when the number of predictors is much smaller than the sample size. If any of these assumptions are violated, then a less flexible method will likely exhibit better performance.

Exercise 6

Problem

Describe the differences between a parametric and non-parametric statistical learning approach. What are the advantages of a parametric approach to regression of classification (as opposed to a non-parametric approach)? What are its disadvantages?

Solution

Parametric models assume a functional form of the relationship between the response and the predictors, whereas non-parametric functions don't. Parametric approaches are easier to interpret than non-parametric approaches, they are less computationally intensive to fit, and they perform better in high dimensional settings, although they may give less accurate predictions than non-parametric methods due to having higher model bias.

Exercise 7

Problem

The table below provides a training data set containing six observations, three predictors, and one qualitative response variable.

```
data = data.frame(c(0,2,0,0,-1,1),c(3,0,1,1,0,1),c(0,0,3,2,1,1),
                  c("Red","Red","Red","Green","Green","Red"))
row.names(data) = 1:6
names(data) = c("X1", "X2", "X3", "Y")
data
```

```
##   X1 X2 X3     Y
## 1  0  3  0   Red
## 2  2  0  0   Red
## 3  0  1  3   Red
## 4  0  1  2 Green
## 5 -1  0  1 Green
## 6  1  1  1   Red
```

Suppose we wish to use this data set to make a prediction for Y when $X_1 = X_2 = X_3 = 0$ using *K-nearest neighbors*.

a) Compute the Euclidean distance between each observation and the test point, $X_1 = X_2 = X_3 = 0$.
b) What is our prediction with K=1? Why?
c) What is our prediction with K=3? Why?
d) If the Bayes decision boundary in this problem is highly nonlinear, then would we expect the *best* value for K to be large or small? Why?

Solution

a) In the code block below, the Euclidean distance of each observation from the test point is calculated, and then the rows are put in ascending order according to the distance.

```
distances <- data.frame(sqrt(data$X1^2 + data$X2^2 + data$X3^2), data$Y) %>%
  set_names(c("dist", "Y")) %>%
  arrange(dist)
distances
```

```
##       dist     Y
## 1 1.414214 Green
## 2 1.732051   Red
## 3 2.000000   Red
## 4 2.236068 Green
## 5 3.000000   Red
## 6 3.162278   Red
```

b) When $K = 1$, our prediction is green because a plurality of the K points closest to the test point are green.
c) When $K = 3$, our prediction is red because a plurality of the K points closest to the test point are red.
d) If the Bayes decision boundary is highly nonlinear, then the optimal value of K should be small. When K is small, the KNN classification boundary is able to take on a more squiggly shape, on account of only being determined locally. When K is large, the decision boundary is less flexible because it is smoothed out by a larger set of neighbors.

Applied

Exercise 8

This exercise relates to the **College** data set, which can be found in the file **College.csv**. It contains a number of variables for 777 different universities and colleges in the US. The variables are

- **Private**: Public/private indicator
- **Apps**: Number of applications received
- **Accept**: Number of applicants accepted
- **Enroll**: Number of new students enrolled
- **Top10perc**: New students from top 10% of high school class
- **Top25perc**: New students from top 25% of high school class
- **F.Undergrad**: Number of full-time undergraduates
- **P.Undergrad**: Number of part-time undergraduates
- **Outstate**: Out-of-state tuition
- **Room.Board**: Room and board costs
- **Books**: Estimated book costs
- **Personal**: Estimated personal spending
- **PhD**: Percent of faculty with Ph.D.'s
- **Terminal**: Percent of faculty with terminal degree
- **S.F.Ratio**: Student/faculty ratio
- **perc.alumni**: Percent of alumni who donate
- **Expend**: Institution expenditure per student
- **Grad.Rate**: Graduation rate

Before reading the data into **R**, it can be viewed in Excel or a text editor.

a) Use the **read.csv()** function to read the data into **R**. Call the loaded data **college**. Make sure that you have the directory set to the correct location for the data.

```
college = read.csv("College.csv", header=TRUE)
```

b) Look at the data using the **fix()** function. You should notice that the first column is just the name of each university. We don't really want R to treat this as data. However, it may be handy to have these names for later. Try the following commands:

```
rownames(college) = college[,1]
#fix(college)
```

You should see that there is now a **row.names** column with the name of each university recorded. This means that **R** has given each row a name corresponding to the appropriate university. **R** will not try to perform calculations on the row names. However, we still need to eliminate the first column in the data where row names are stored. Try

```
college = college[,-1]
#fix(college)
```

Now you should see that the first data column is **Private**. Note that another column labeled **row.names** now appears before the **Private** column. However, this is not a data column but rather the name that **R** is giving to each row.

c) i. Use the **summary()** function to produce a numerical summary of the variables in the data set.

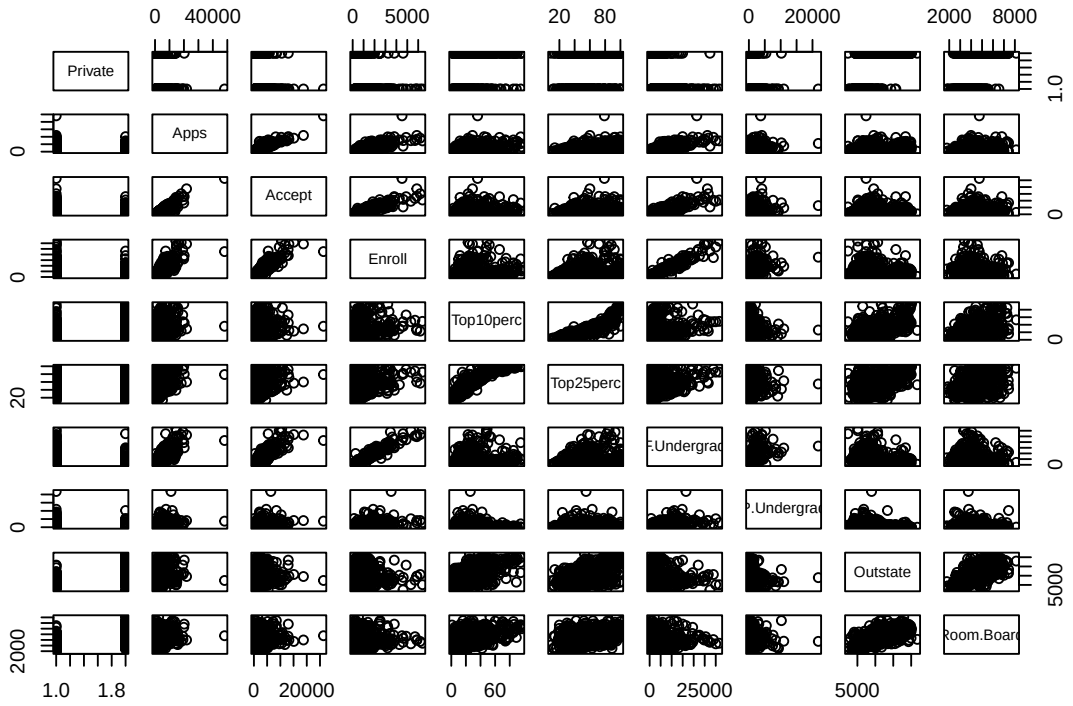
```
summary(college)
```

```
##    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
```

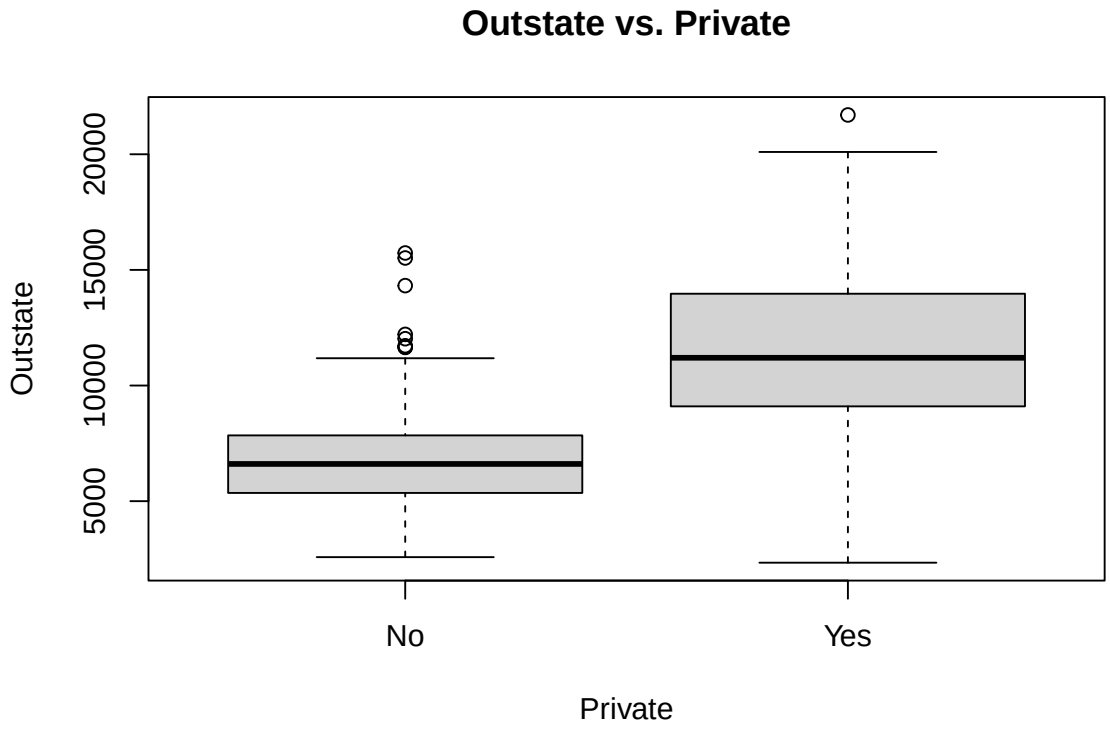
ii. Use the **pairs()** function to produce a scatterplot matrix of the first ten columns or variables of the data. Recall that you can reference the first ten columns of a matrix A using A[,1:10].

```
college$Private = as.factor(college$Private)
pairs(college[,1:10])
```



iii. Use the **plot()** function to produce side-by-side boxplots of **Outstate** versus **Private**.

```
plot(x=college$Private, y=college$Outstate, xlab="Private", ylab="Outstate", main="Outstate vs. Pri
```



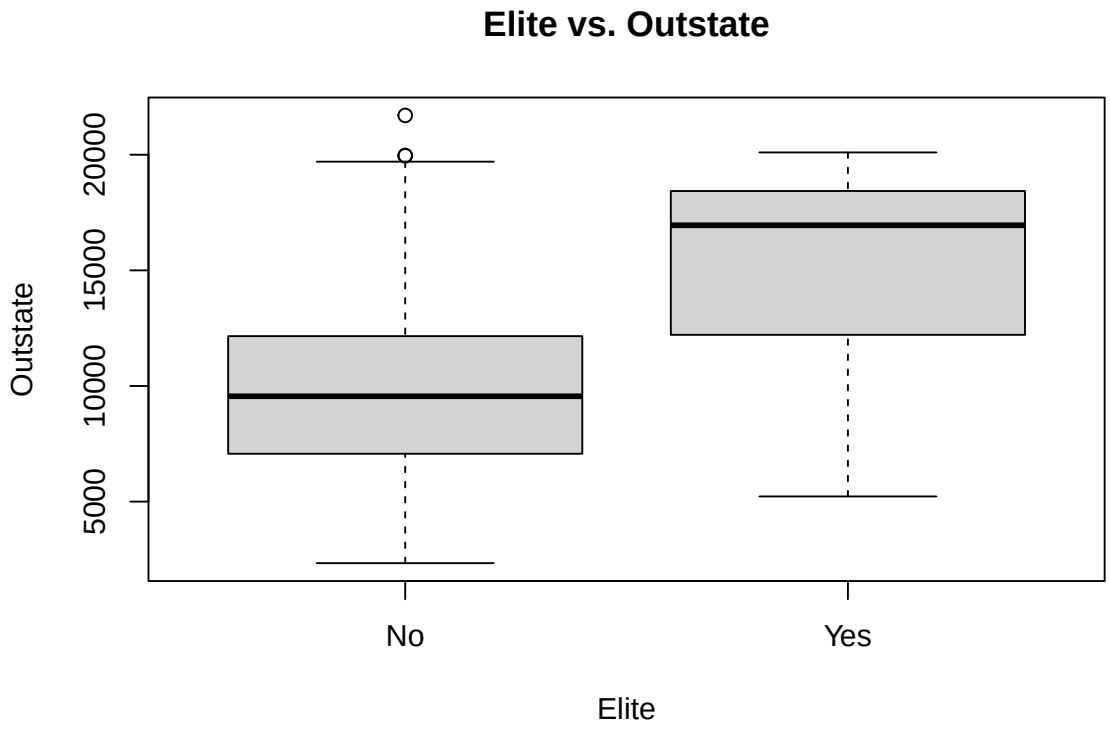
Private colleges tend to have more out-of-state students than public colleges.

iv. Create a new qualitative variable called **Elite**, by binning the **Top10perc** variable. We are going to divide universities into two groups based on whether or not the proportion of students coming from the top 10% of their high school classes exceeds 50%.

```
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
```

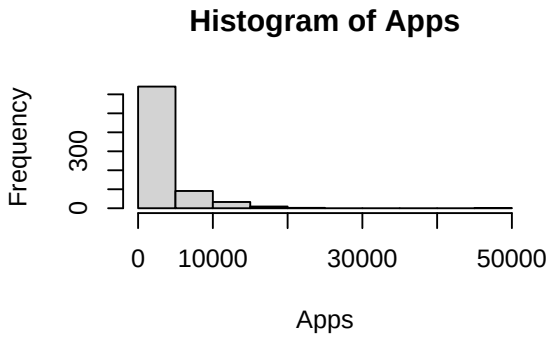
```
plot( college$Elite, college$Outstate, xlab="Elite", ylab="Outstate", main="Elite vs. Outstate")
```

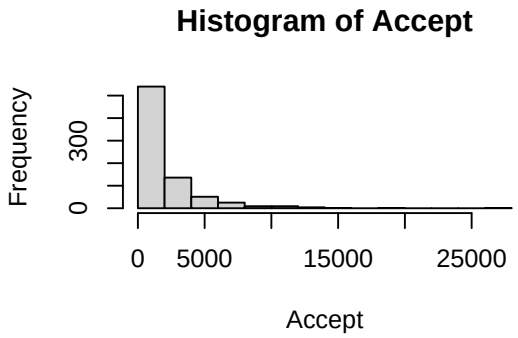


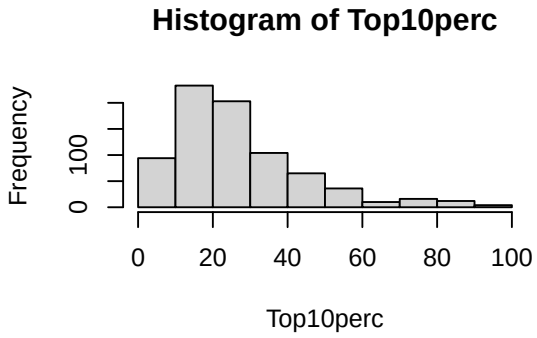
Elite colleges tend to have more out-of-state students than non-elite colleges.

v. Use the **hist()** function to produce some histograms with differing numbers of bins for a few of the quantitative variables. You may find the command **par(mfrow=c(2,2))** useful: It will divide the print window into four regions so that four plots can be made simultaneously. Modifying the arguments to this function will divide the screen in other ways.

```
par(mfrow=c(2,2))
hist(college$Apps, xlab="Apps", main="Histogram of Apps")
hist(college$Accept, xlab="Accept", main="Histogram of Accept")
hist(college$Top10perc, xlab="Top10perc", main="Histogram of Top10perc")
hist(college$S.F.Ratio, xlab="S.F.Ratio", main="Histogram of S.F.Ratio")
```



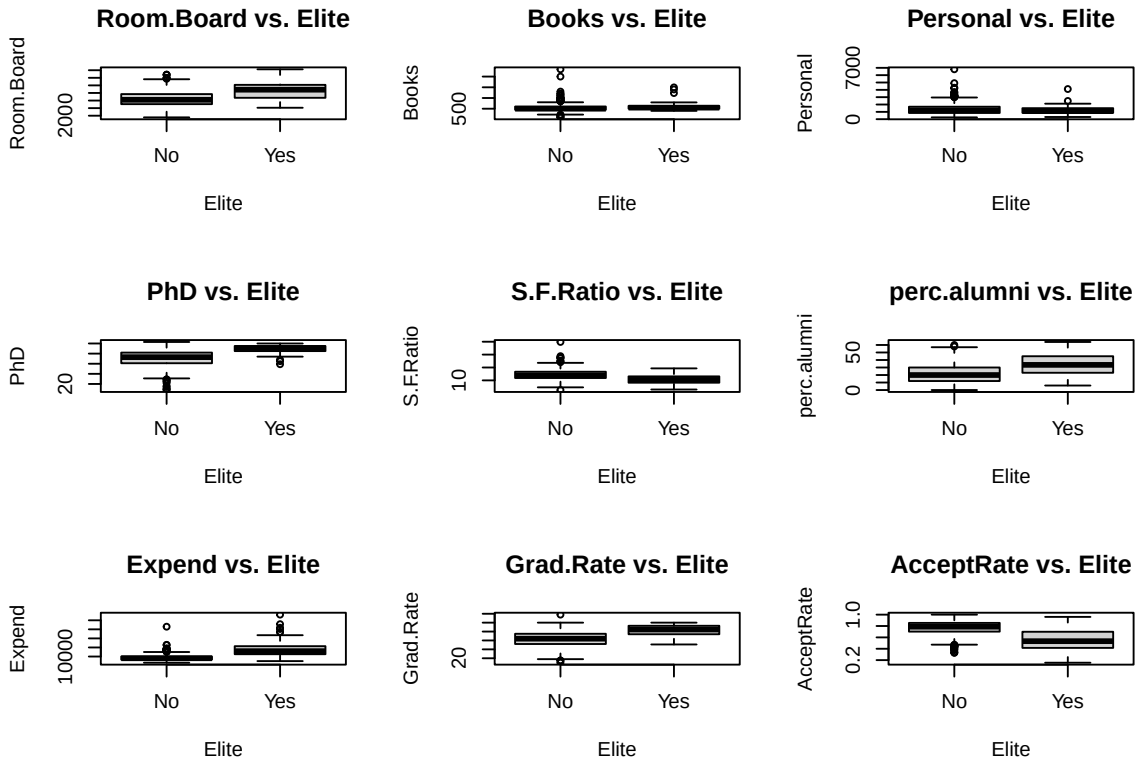




Many colleges recieve only a few applications (<5000), whereas a few colleges recieve many applications. Many colleges accept only a few students (<2500), whereas a few colleges accept many students. The distributions for **Top10perc** and **S.F.Ratio** are approximately normal, although the former has a heavy upper tail, which may indicate that there are some elite colleges who have much higher numbers of incoming students who graduated in the top 10% of their high school class.

vi Continue exploring your data, and provide a brief summary of what you discover.

```
# Okay, let's calculate acceptance rate
college$AcceptRate = college$Accept/college$Apps
par(mfrow=c(3,3))
# And now, I want to look at different variables as a function of elite
for (i in c(10,11,12,13,15,16,17,18,20)){
  plot(x=college$Elite, y=college[,i], xlab="Elite", ylab=names(college)[i], main=paste(names(colle
}
```



Elite is positively correlated with **Outstate**, **Room.Board**, **PhD**, **perc.alumni**, **Expend**, **Grad.Rate**, and is negatively correlated with **S.F.Ratio** and **AcceptRate**. All of these associations make sense, as generally, you would expect that elite schools are more expensive and attract the best students and faculty. The acceptance rates are lower, which may help to explain why the student-to-faculty ratios are lower for elite schools.

Below, the percentage of private and publuc schools that are elite is reported, respectively.

```
# How many private schools are elite? Public schools?
sum( college$Private == "Yes" & college$Elite == "Yes")/length(college$Private)*100
```

```
## [1] 8.365508
```

```
sum( college$Private == "No" & college$Elite == "Yes")/length(college$Private)*100
```

```
## [1] 1.673102
```

Finally, in the histogram of **top10perc** produced in part (c.v), note that the upper tail of the distribution was rather heavy, with a high peak around 20% and a low peak around 80%. It may be possible to model this distribution as a mixture of two normal distributions of **top10perc**, one for non-elite schools and one for elite schools. As preliminary evidence, consider that the means of **top10percent** over these two groups are very different.

```
# Mean of top10perc given elite
mean(college$Top10perc[college$Elite == "Yes"])
```

```
## [1] 67.61538
```

```r
# Mean of top10perc given not elite
mean(college$Top10perc[college$Elite == "No"])
```

```
## [1] 23.0887
```

Exercise 9

This exercise involves the **Auto** data set studied in the lab. Make sure that the missing values have been removed from the data.

```r
Auto = read.csv("Auto.csv", header=TRUE, na.strings="?")
Auto = na.omit(Auto)
View(Auto)
```

a) Which of the predictors are quantitative, and which are qualitative?

name and **origin** are qualitative; the rest are quantitative.

b) What is the range of each quantitative predictor? You can answer this using the **range()** function.
c) What is the mean and standard deviation of each quantitative predictor?

The range, mean, and standard deviation of each predictor are reported in the following output.

```r
df = data.frame(c(range(Auto[,1]), mean(Auto[,1]), sd(Auto[,1])))
for (i in 2:7){
  df = data.frame(df, c(range(Auto[,i]), mean(Auto[,i]), sd(Auto[,i])))
}
names(df) = names(Auto)[-c(8,9)]
rownames(df) = c("min", "max", "mean", "sd")
df
```

```
##            mpg cylinders displacement horsepower    weight acceleration
## min   9.000000  3.000000       68.000   46.00000 1613.0000     8.000000
## max  46.600000  8.000000      455.000  230.00000 5140.0000    24.800000
## mean 23.445918  5.471939      194.412  104.46939 2977.5842    15.541327
## sd    7.805007  1.705783      104.644   38.49116  849.4026     2.758864
##           year
## min  70.000000
## max  82.000000
## mean 75.979592
## sd    3.683737
```

d) Now remove the 10th through 85th observations. What is the range, mean, and standard deviation of each predictor in the subset of the data that remains.

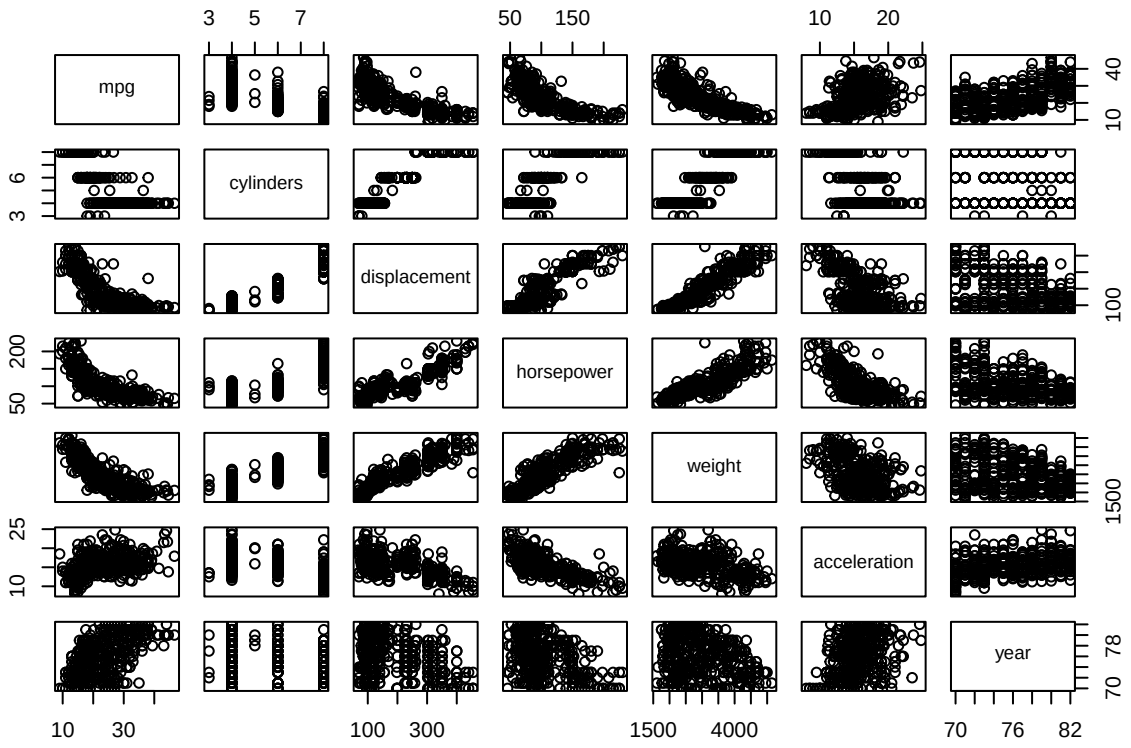
The range, mean, and standard deviation of each predictor over this subset is reported in the following output.

```
Auto2 = Auto[-c(10:85),]
df2 = data.frame(c(range(Auto2[,1]), mean(Auto2[,1]), sd(Auto2[,1])))
for (i in 2:7){
  df2 = data.frame(df2, c(range(Auto2[,i]), mean(Auto2[,i]), sd(Auto2[,i])))
}
names(df2) = names(Auto2)[-c(8,9)]
rownames(df2) = c("min", "max", "mean", "sd")
df2
```

```
##            mpg cylinders displacement horsepower    weight acceleration
## min  11.000000  3.000000     68.00000   46.00000 1649.0000     8.500000
## max  46.600000  8.000000    455.00000  230.00000 4997.0000    24.800000
## mean 24.404430  5.373418    187.24051  100.72152 2935.9715    15.726899
## sd    7.867283  1.654179     99.67837   35.70885  811.3002     2.693721
##           year
## min  70.000000
## max  82.000000
## mean 77.145570
## sd    3.106217
```

e) Using the full data set, investigate the predictors graphically, using scatterplots or other tools of your choice. Create some plots highlighting the relationships among the predictors. Comment on your findings.

```
pairs(Auto[,-c(8,9)])
```



1. Cars become more efficient over time, based on the tendency for **mpg** to increase with **year**.
2. **mpg** decreases with **cylinders**, whereas **horsepower** increases with **cylinders**, so it seems that there is a trade off between fuel efficiency and power.

f) Suppose that we wish to predict gas mileage (**mpg**) on the basis of the other variables. Do your plots suggest that any of the other variables might be useful in predicting **mpg**? Justify your answer.

Yes. _cylinders___, **year**, **displacement**, **horsepower**, and **weight** all appear to be correlated with **mpg**, so these variables should be considered if attempting to predict **mpg** from this data.

Exercise 10

This exercise involved the **Boston** housing data set. a) To begin, load in the **Boston** data set. The **Boston** data set is part of the **MASS** library in **R**.

```
library(MASS)
```

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

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

Now the data set is contained in the object **Boston**.

How many rows are in this data set? How many columns? What do the rows and columns represent?

```
# ?Boston
```

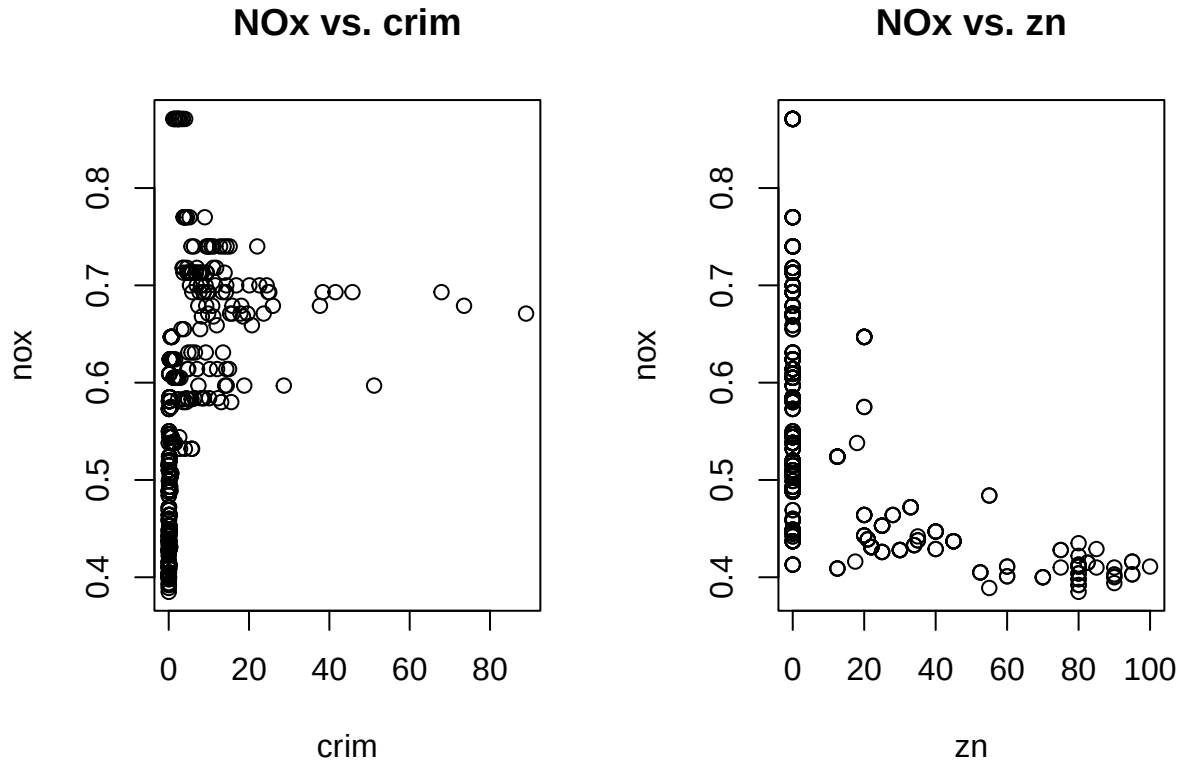
Referring to the help file, this data set has 506 rows and 14 columns. Each row represents a suburb of Boston, and the columns are:

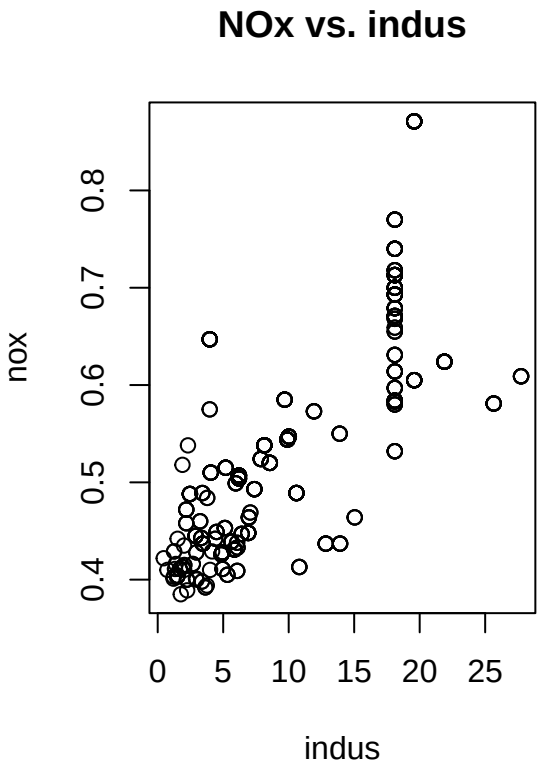
- **crim**: per capita crime rate by town
- **zn**: proportion of residential land zoned for lots over 25,000 sq.ft.
- **indus**: proportion of non-retail business acres per town
- **chas**: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
- **nox**: nitrogen oxides concentration (parts per 10 million).
- **rm**: average number of rooms per dwelling.
- **age**: proportion of owner-occupied units built prior to 1940.
- **dis**: weighted mean of distances to five Boston employment centers.
- **rad**: index of accessibility to radial highways.
- **tax**:full-value property-tax rate per $10,000.
- **ptratio**: pupil-teacher ratio by town
- **black**: $1000(Bk - 0.63)^2$ where Bk is the proportion of black people by town
- **lstat**: lower status of the population (percent)
- **medv**: median value of owner-occupied homes in $1000s

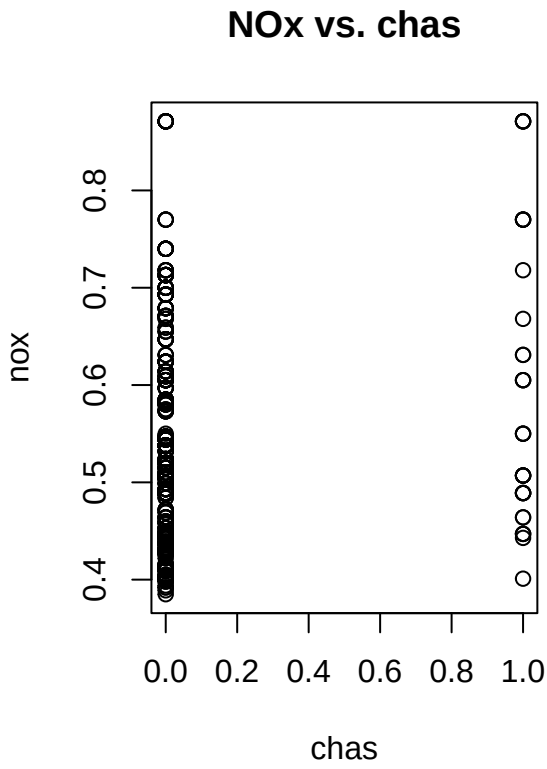
b) Make some pairwise scatterplots of the predictors (columns) in this data set. Describe your findings.

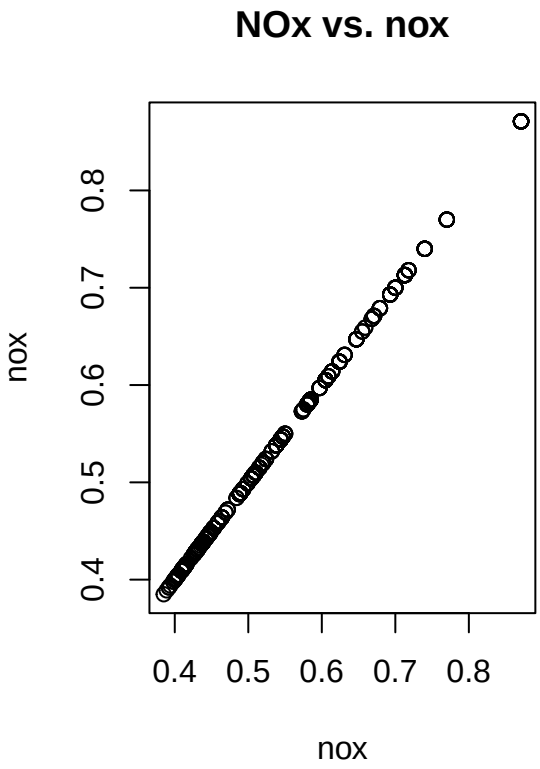
It would not be feasible to look at all of the scatterplots, as there are 91 of them in total. Instead, in this part, we'll look at scatterplots for **nox** versus all of the other predictors, as we may be interested in predicting the concentration of potentially harmful gases.

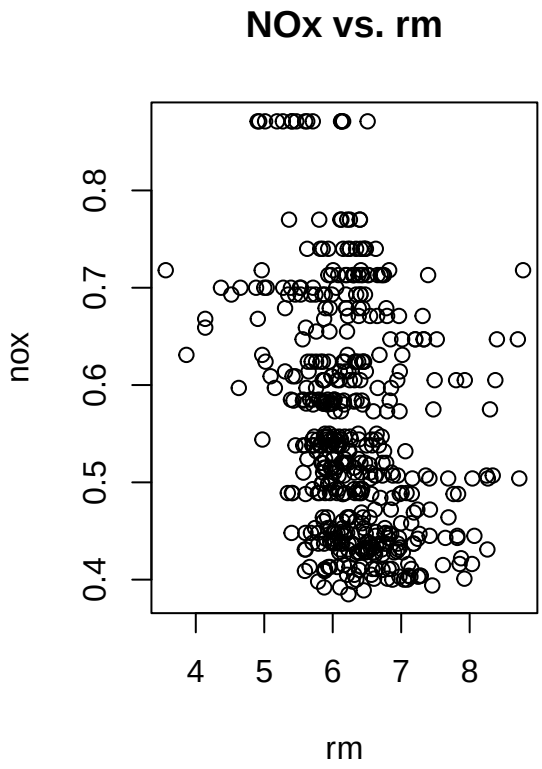
```
par(mfrow=c(1,2))
for (i in 1:14){
  plot(Boston[,i], Boston$nox, xlab=names(Boston)[i], ylab="nox", main=paste("NOx vs.", names(Boston)[i]
}
```

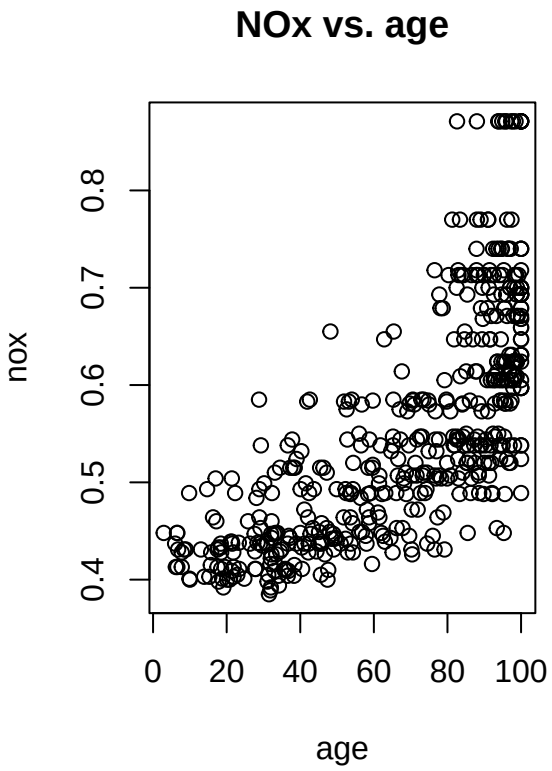


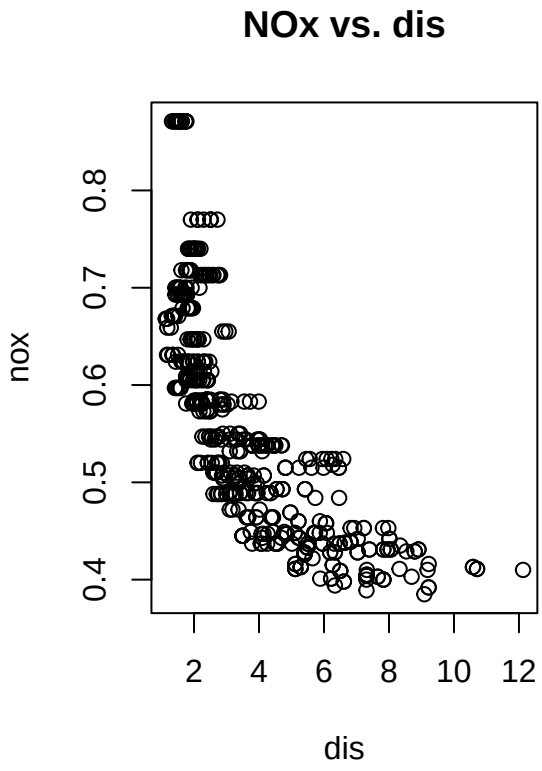

NOx vs. indus
nox
0.4
0.5
0.6
0.7
0.8
0
5
10
15
20
25
indus

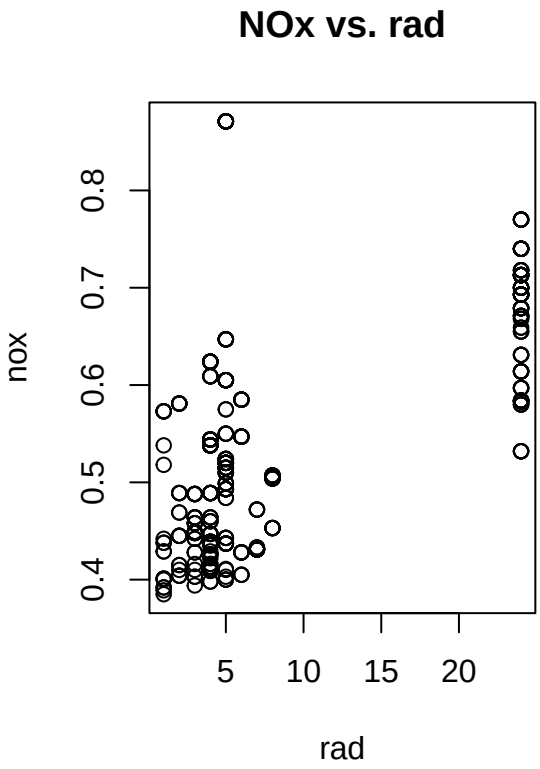

NOx vs. chas
nox
0.4
0.5
0.6
0.7
0.8
0.0
0.2
0.4
0.6
0.8
1.0
chas

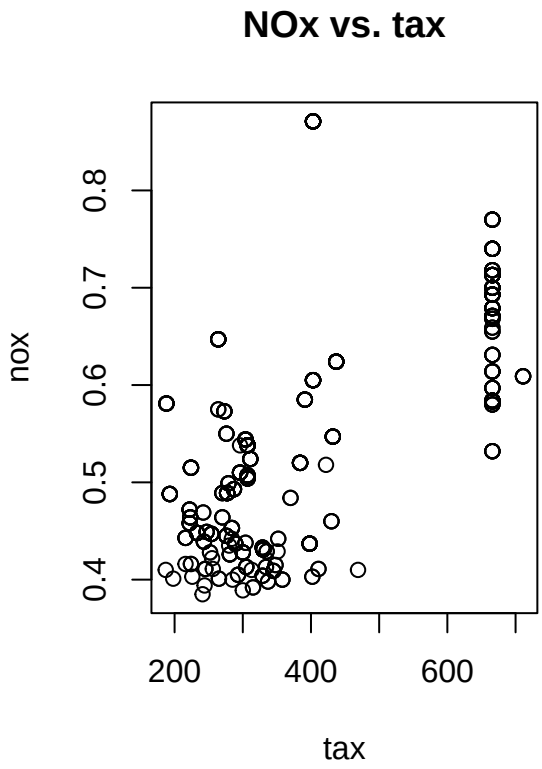

NOx vs. nox
nox
0.4
0.5
0.6
0.7
0.8
nox

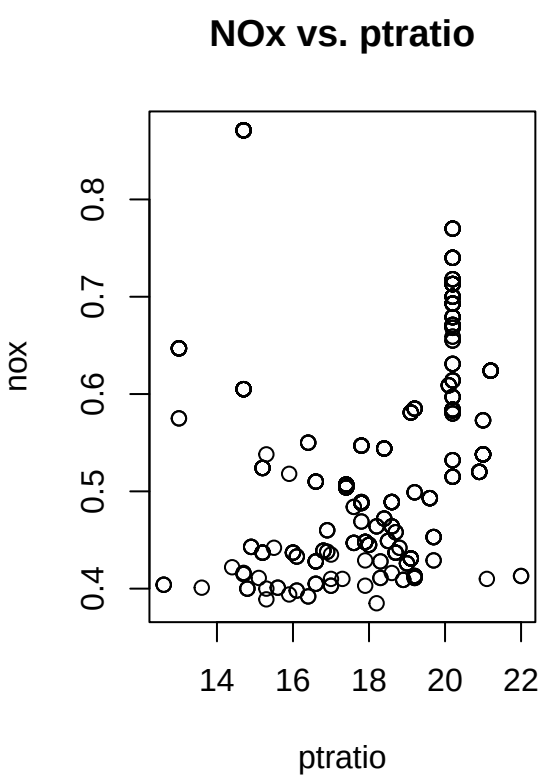

NOx vs. rm
nox
0.4
0.5
0.6
0.7
0.8
4
5
6
7
8
rm

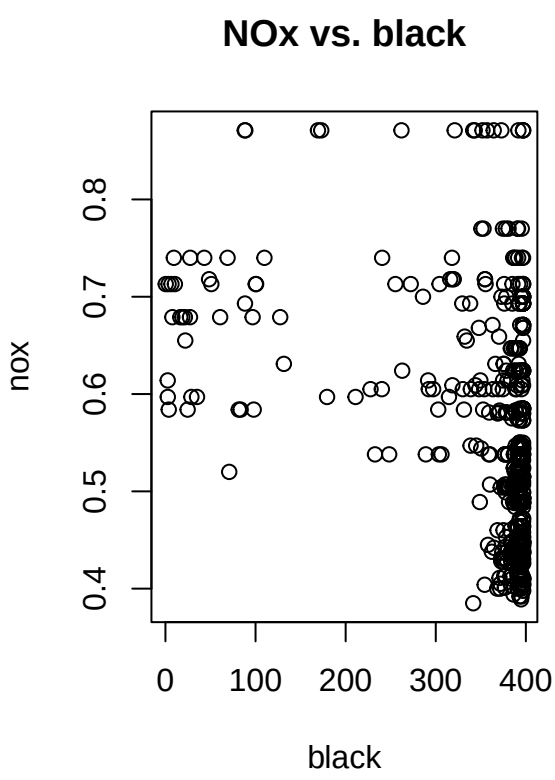

NOx vs. age
nox
0.4
0.5
0.6
0.7
0.8
0
20
40
60
80
100
age

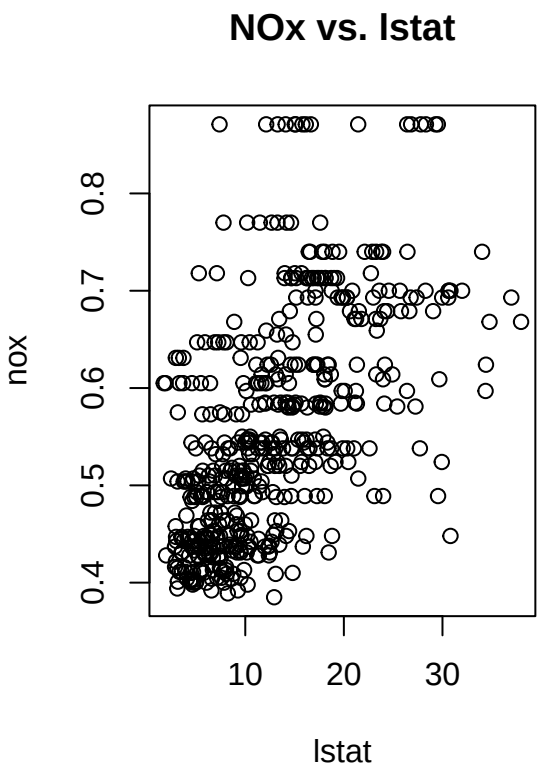

NOx vs. dis
nox
0.4
0.5
0.6
0.7
0.8
2
4
6
8
10
12
dis

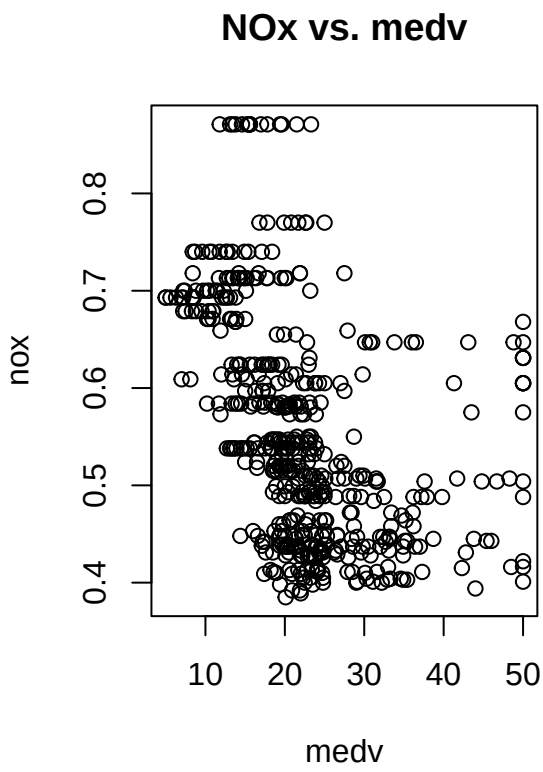

NOx vs. rad
nox
0.4
0.5
0.6
0.7
0.8
5
10
15
20
rad


NOx vs. tax
nox
0.4
0.5
0.6
0.7
0.8
200
400
600
tax


NOx vs. ptratio
nox
0.4
0.5
0.6
0.7
0.8
14
16
18
20
22
ptratio


NOx vs. black
nox
0.4
0.5
0.6
0.7
0.8
0
100
200
300
400
black



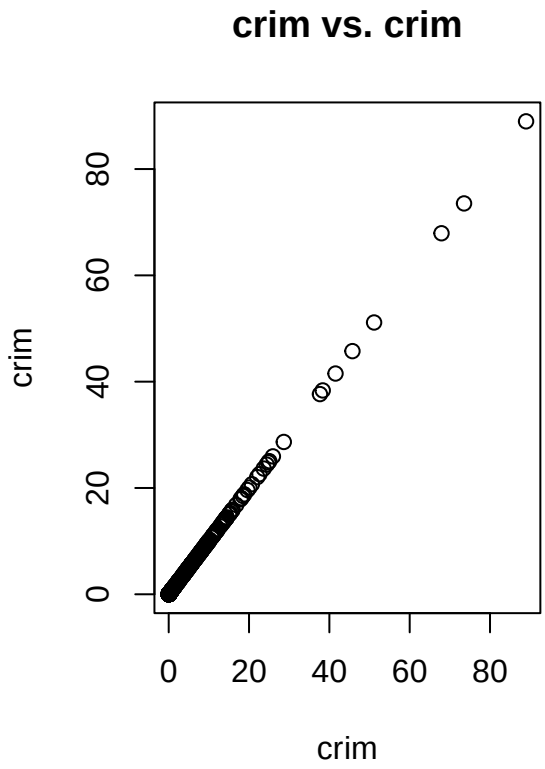


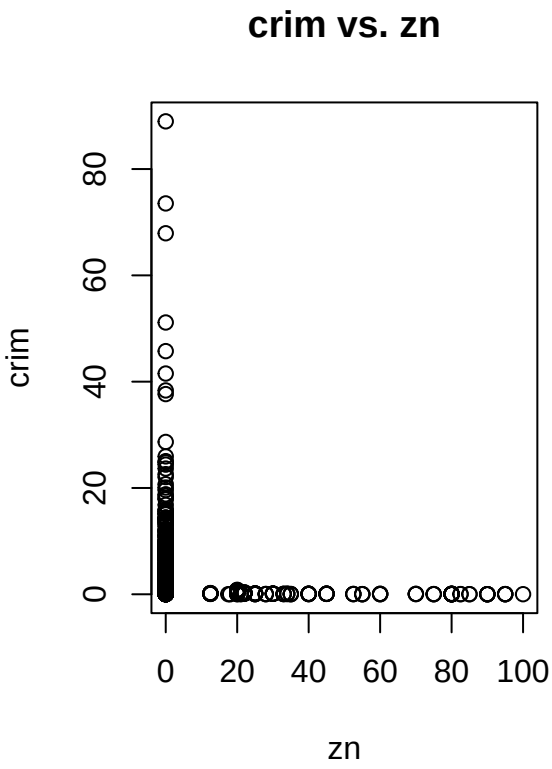
From these scatterplots, we can observe that **nox** has the strongest positive correlations with **indus**, **age**, **rad**, **tax**, **ptratio**, and **lstat**, and the strongest negative correlations with **medv**, **black**, **dis**, and **zn**.

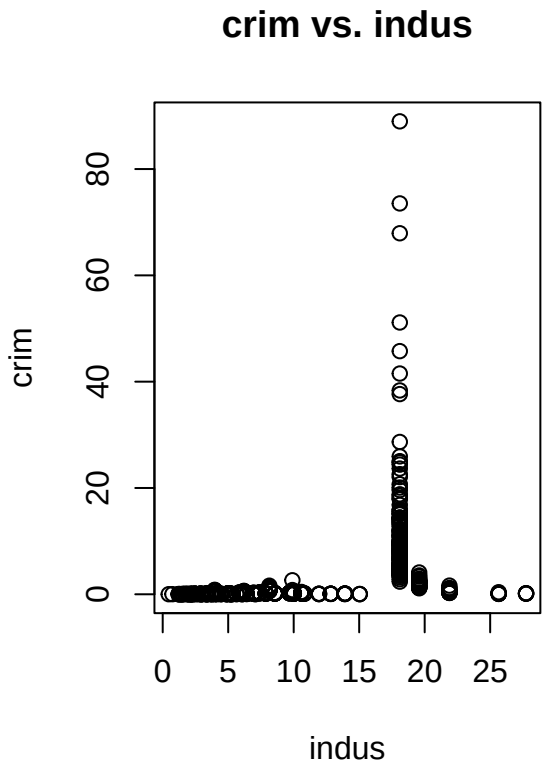
c) Are there any predictors associated with per capita crime rate? If so, explain the relationship.

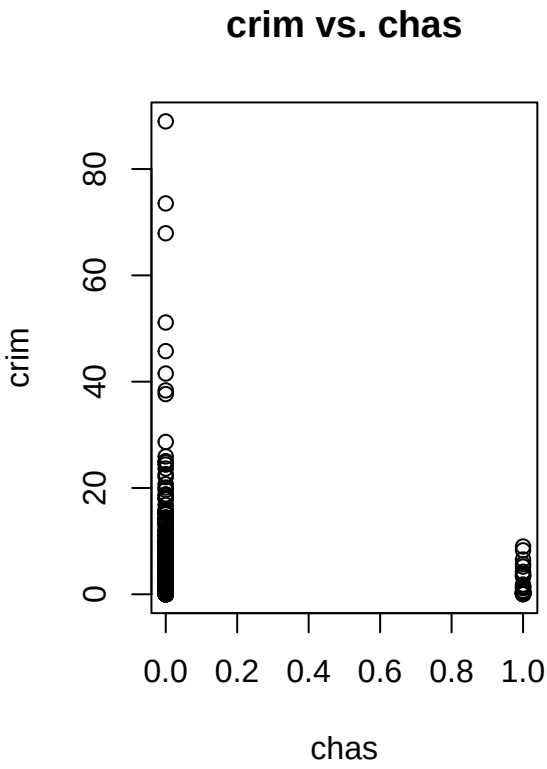
Crime is another potentially harmful thing we would wish to predict, so we should look at the scatterplots of crime vs. the other predictors.

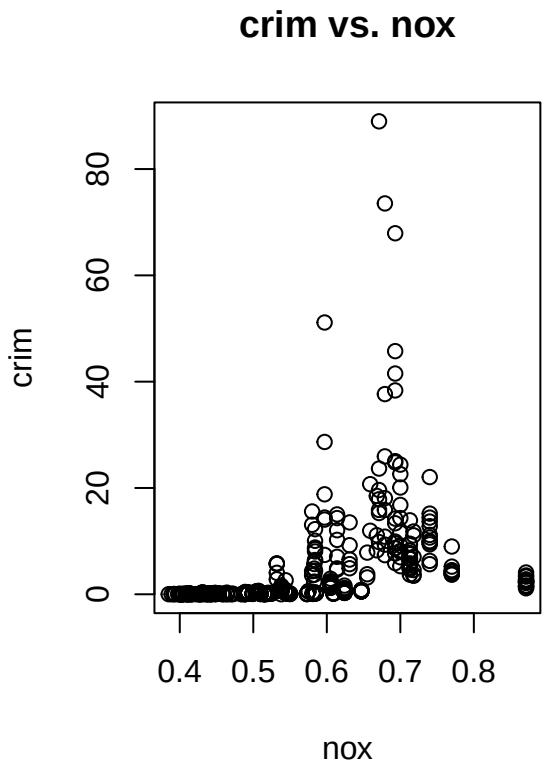
```
par(mfrow=c(1,2))
for (i in 1:14){
  plot(Boston[,i], Boston$crim, xlab=names(Boston)[i], ylab="crim", main=paste("crim vs.", names(Boston)
}
```

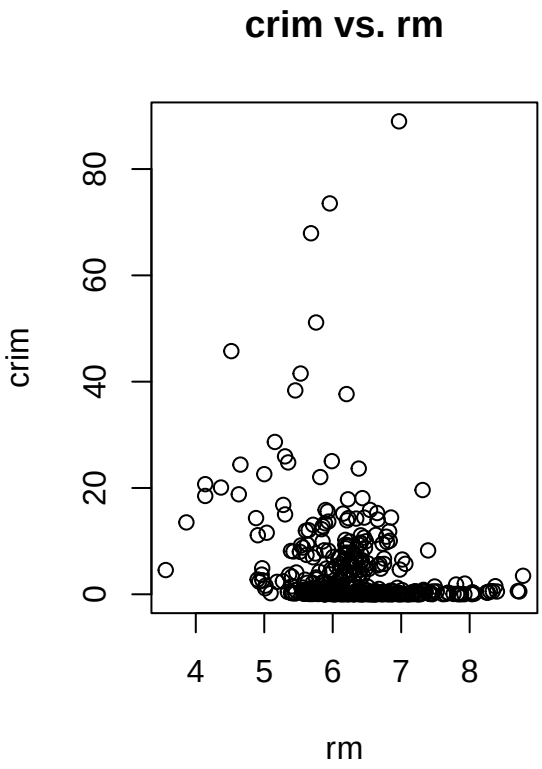

crim vs. crim
crim
0
20
40
60
80
0
20
40
60
80
crim

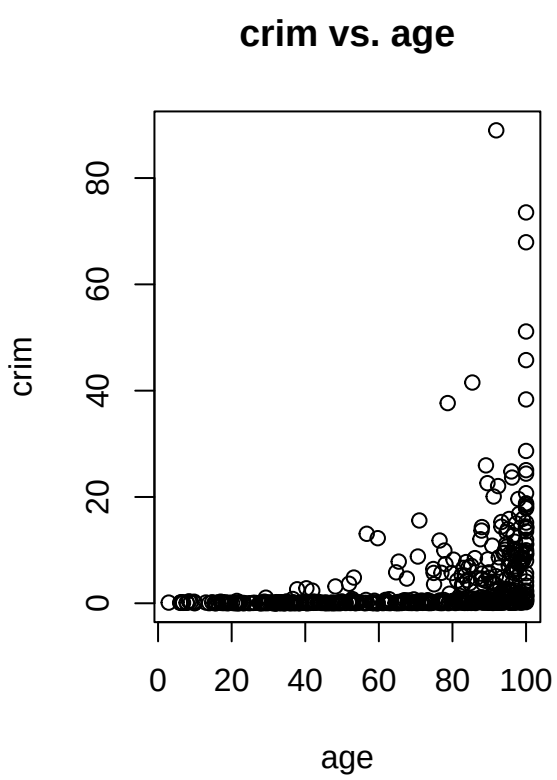

crim vs. zn
crim
0
20
40
60
80
0
20
40
60
80
100
zn

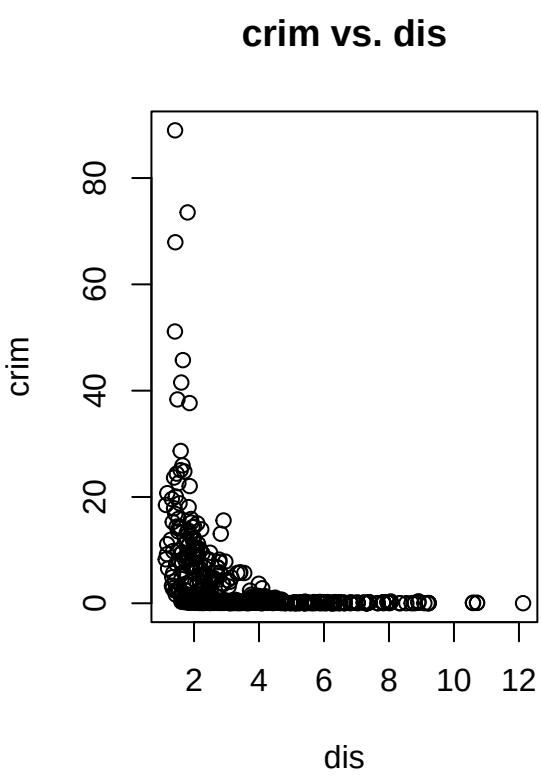

crim vs. indus
crim
0
20
40
60
80
0
5
10
15
20
25
indus

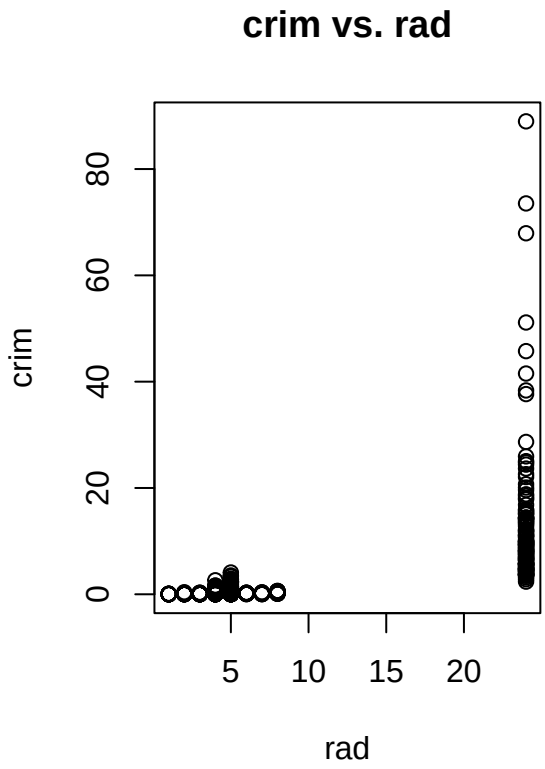

crim vs. chas
crim
0
20
40
60
80
0.0
0.2
0.4
0.6
0.8
1.0
chas

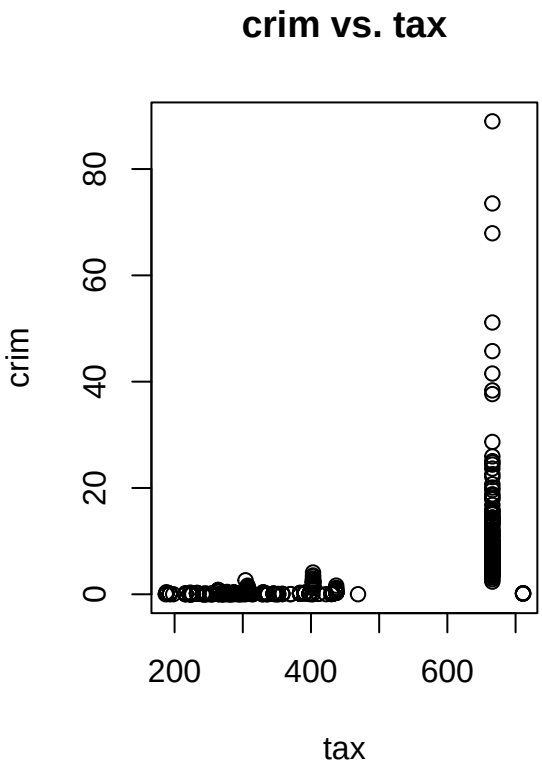

crim vs. nox
crim
0
20
40
60
80
0.4
0.5
0.6
0.7
0.8
nox

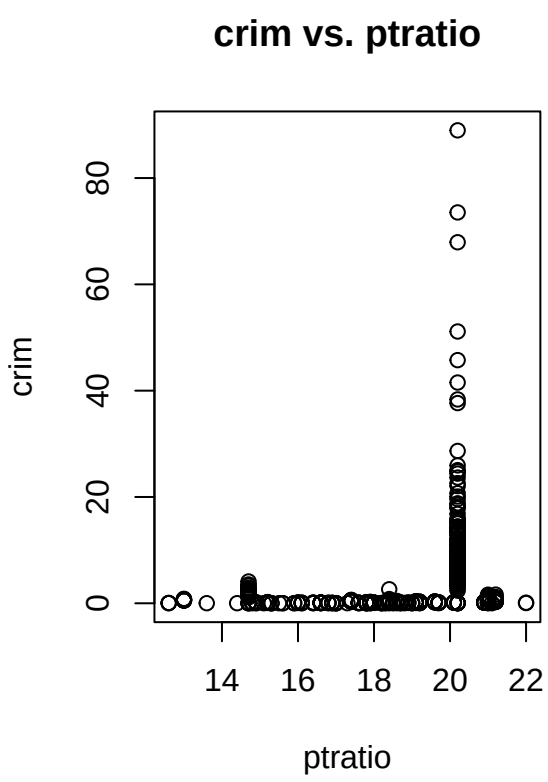

crim vs. rm
crim
0
20
40
60
80
4
5
6
7
8
rm

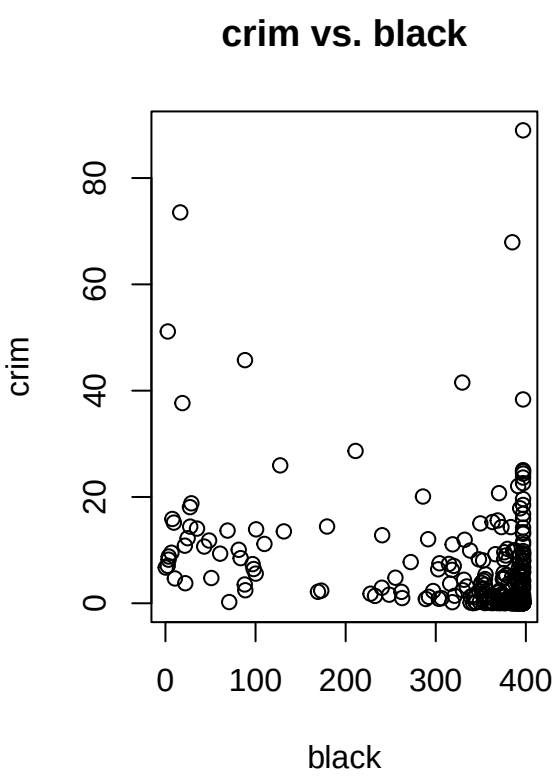

crim vs. age
crim
0
20
40
60
80
0
20
40
60
80
100
age

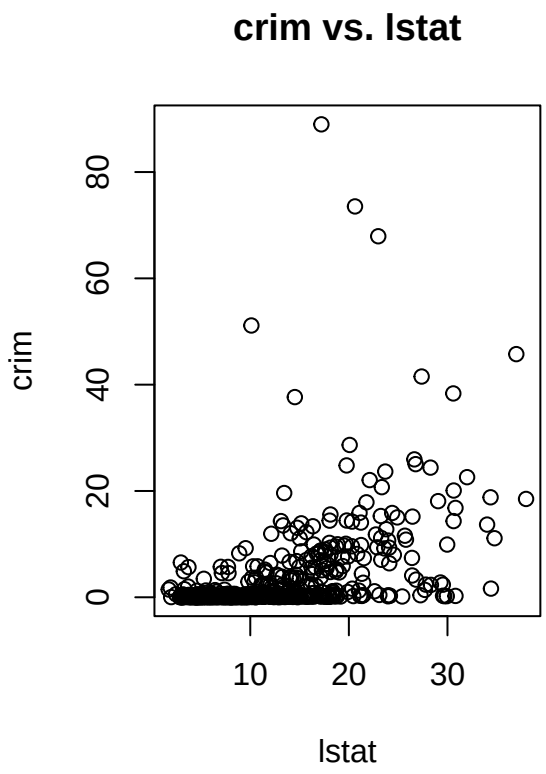

crim vs. dis
crim
0
20
40
60
80
2
4
6
8
10
12
dis

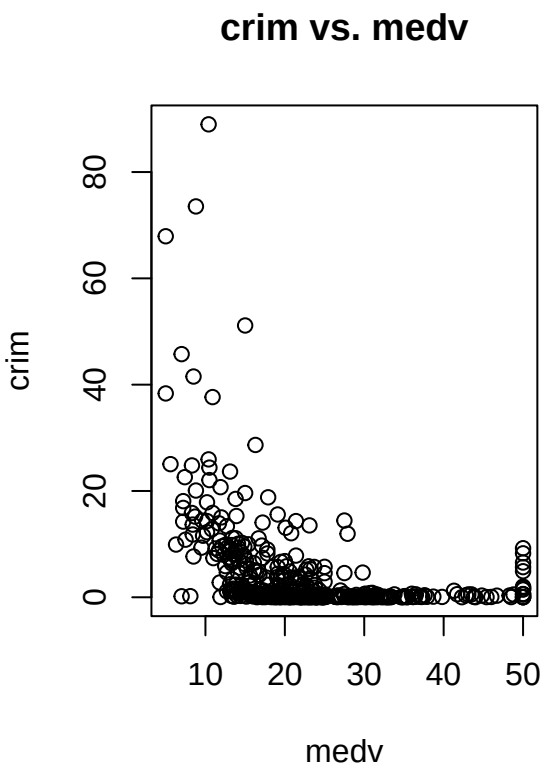

crim vs. rad
crim
0
20
40
60
80
5
10
15
20
rad


crim vs. tax
crim
0
20
40
60
80
200
400
600
tax


crim vs. ptratio
crim
0
20
40
60
80
14
16
18
20
22
ptratio


crim vs. black
crim
0
20
40
60
80
0
100
200
300
400
black





Crime is positively correlated with **nox**, **age**, **rad**, **tax**, **lstat**, and negatively correlated with **medv**.

d) Do any of the suburbs of Boston appear to have particularly high crime rates? Tax rates? Pupil-teacher ratios? Comment on the range of each predictor.

The range of these particular predictors are reported below.

```
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)
```

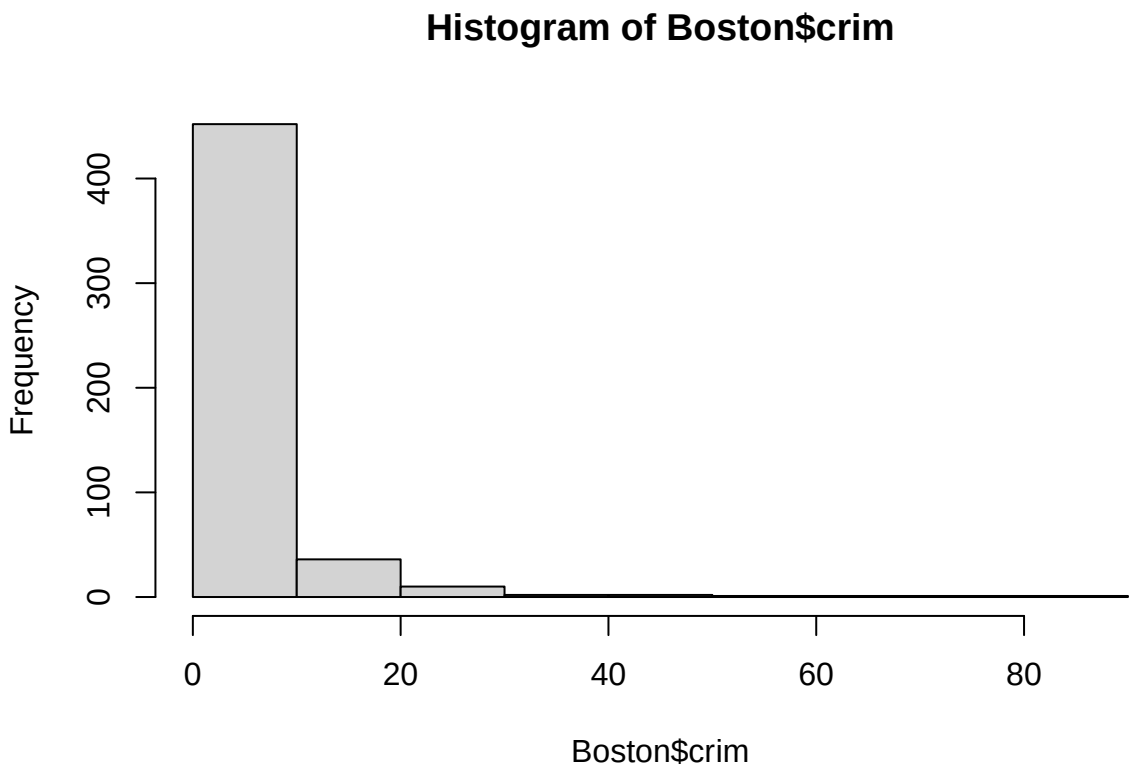
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   187.0   279.0   330.0   408.2   666.0   711.0
```

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

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

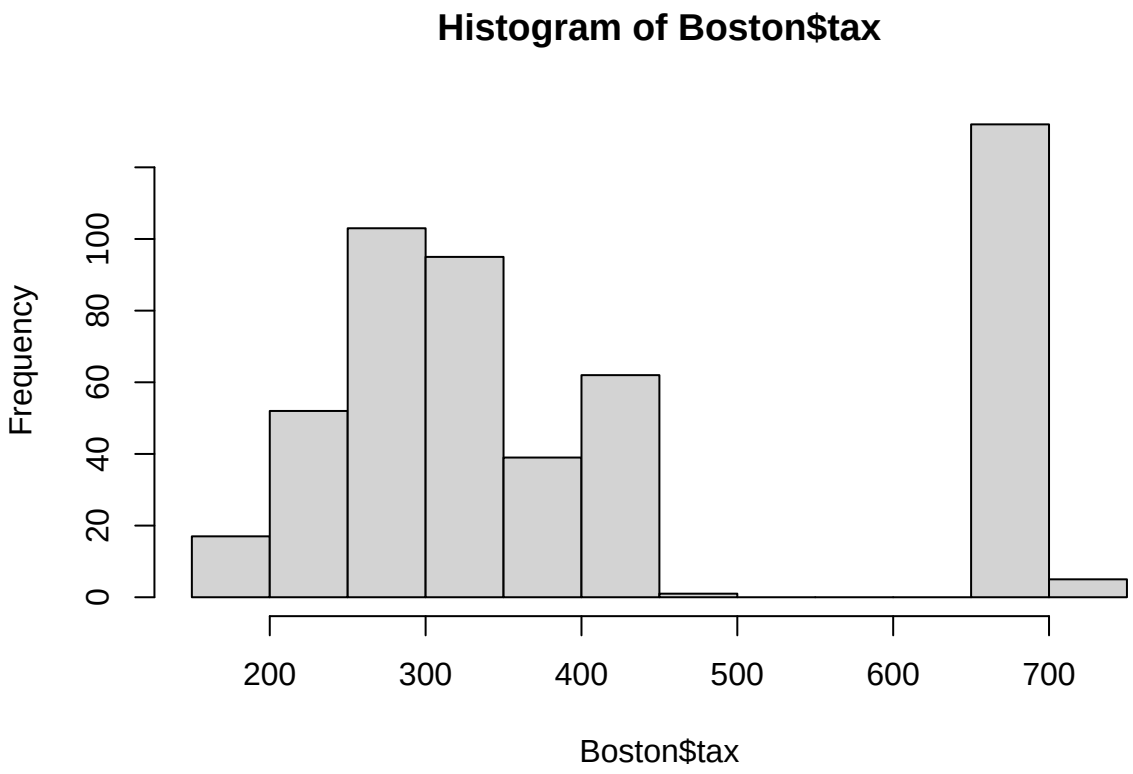
We'll also take a look at histograms of these quantities.

```
hist(Boston$crim, main="Histogram of Boston$crim")
```



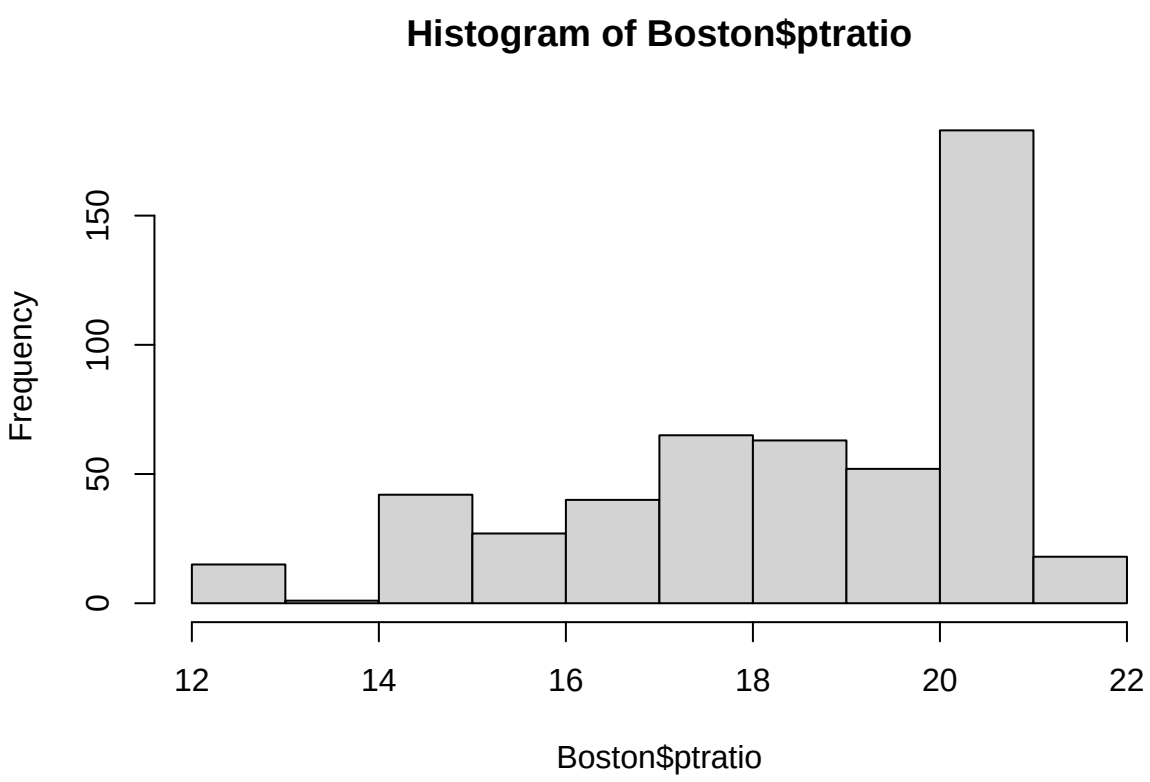
Most suburbs of Boston have a low crime rates (75% with less than 3.677 crimes per capita). But the upper tail is quite long, with one town even reaching a crime rate of 89%.

```
hist(Boston$tax, main="Histogram of Boston$tax")
```



The **tax** distribution is bimodal, which may indicate that most neighborhoods were affordable, with a large handful being much more expensive.

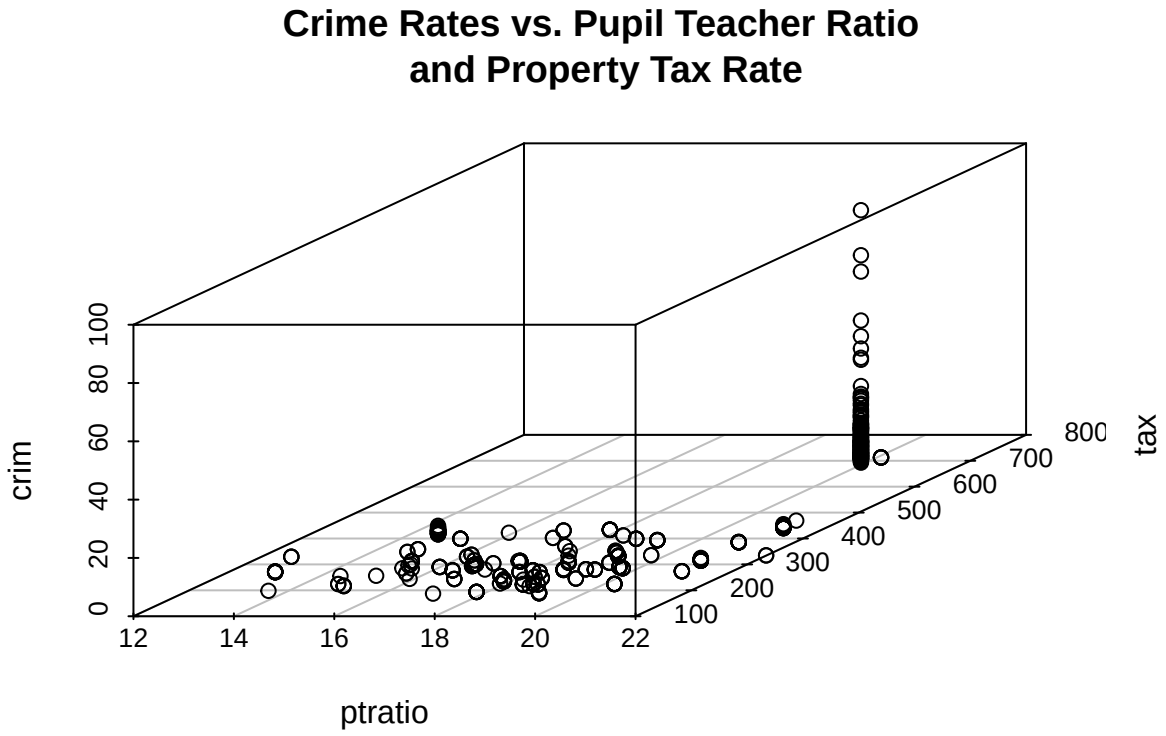
```
hist(Boston$ptratio, main="Histogram of Boston$ptratio")
```



The **ptratio** distribution is left skewed, which means that there are many suburbs of Boston where each teacher has to tend to many students. Referring back to the scatterplot of **crim** vs. **ptratio** produced in part c, note that the neighborhoods with the highest crime rates have **ptratio** values near the 3rd quartile. Interestingly, neighborhoods with **ptratio** values between the 3rd and 4th quartiles have lower crime rates than those near the 3rd quartile.

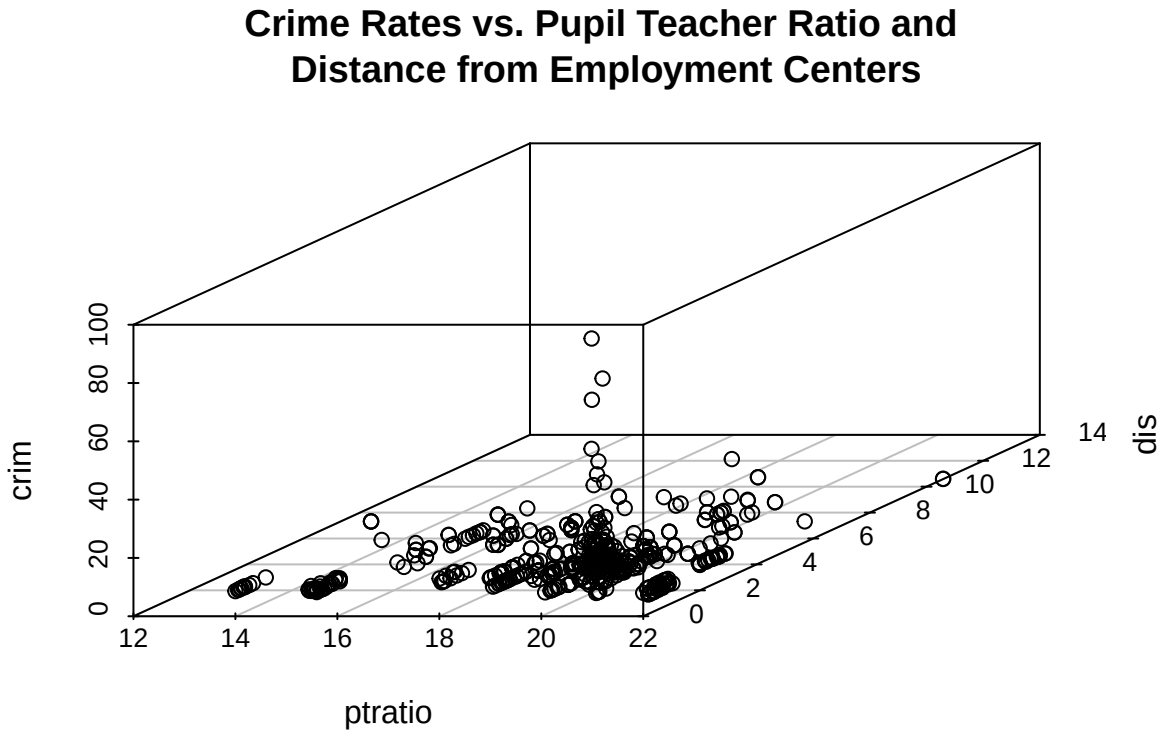
Inspecting for potential confounders leads to some interesting observations.

```
scatterplot3d(x=Boston$ptratio, y=Boston$tax, z=Boston$crim, xlab="ptratio", ylab="tax", zlab="crim", ma
```



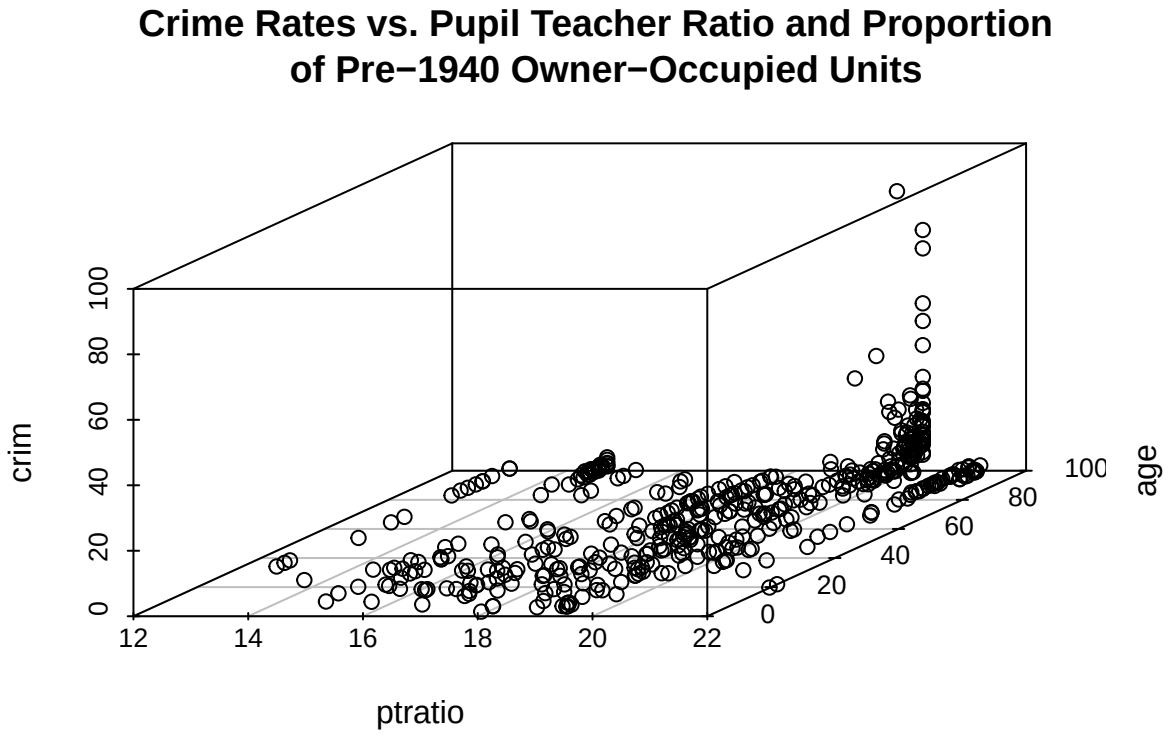
Notice that the suburbs with the highest crime rates have ptratios at the 3rd quartile and very high tax rates.

```
scatterplot3d(x=Boston$ptratio, y=Boston$dis, z=Boston$crim, xlab="ptratio", ylab="dis", zlab="crim", ma
```



All of the suburbs with the highest crime rates are very near to the city.

```
scatterplot3d(x=Boston$ptratio, y=Boston$age, z=Boston$crim, xlab="ptratio", ylab="age", zlab="crim", ma
```



All of the highest crime rate towns are very old.

From this, we observe that the towns with the highest crime rates have many old buildings and are very close to Boston-proper.

e) How many of the suburbs in this data set bound the Charles river?

```
count = sum(Boston$chas)
percent = sum(Boston$chas)/length(Boston$chas) * 100
print( paste(toString(count),"suburbs border the Charles River, accounting for ",toString(percent),"% o
```

```
## [1] "35 suburbs border the Charles River, accounting for  6.91699604743083 % of all suburbs in this
```

f) What is the median pupil-teacher ratio among the towns in this data set?

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

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

g) Which suburb of Boston has the lowest median value of owner-occupied homes? What are the values of the other predictors for that suburb, and how do those values compare to the overall ranged for those predictors? Comment on your findings.

```
min_val = min(Boston$medv)
min_val_town = Boston[ Boston$medv == 5 ,]
min_val_town
```

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

There are two suburbs with the lowest median value of owner-occupied homes, 399 and 406. We show how the value of their predictors compares with the others by giving their percentiles below.

```
percentile = ecdf(Boston$crim)
min_val_town_percentiles = data.frame(percentile(min_val_town$crim))
for (i in 2:14){
  percentile = ecdf(Boston[,i])
  min_val_town_percentiles = data.frame(min_val_town_percentiles, percentile(min_val_town[,i]))
}
names(min_val_town_percentiles) = names(Boston)
row.names(min_val_town_percentiles) = c(399, 406)
min_val_town_percentiles
```

```
##          crim        zn     indus    chas       nox        rm age        dis
## 399 0.9881423 0.7351779 0.8873518 0.93083 0.8577075 0.0770751   1 0.05731225
## 406 0.9960474 0.7351779 0.8873518 0.93083 0.8577075 0.1363636   1 0.04150198
##     rad       tax   ptratio     black     lstat        medv
## 399   1 0.9901186 0.8893281 1.0000000 0.9782609 0.003952569
## 406   1 0.9901186 0.8893281 0.3498024 0.8992095 0.003952569
```

The values of these suburbs' predictors are very similar, except for the percentiles of **rm**, **dis**, **black**, and **lstat**.

h) In this data set, how many of the suburbs average more than seven rooms per dwelling? More than eight rooms per dwelling? Comment.

```
# This is the percentage of suburbs averaging more than seven rooms per dwelling
sum(Boston$rm > 7) / length(Boston$rm > 7) * 100
```

```
## [1] 12.64822
```

```
# This is the percentage of suburbs averaging more than eight rooms per dwelling
sum(Boston$rm > 8) / length(Boston$rm > 8)
```

```
## [1] 0.0256917
```

Now, to see how these values compare to the sample as a whole, we take the mean of each predictor over this subset, and then obtain the quantile of those means in the whole sample.

```
# STEP 1: Calculate the mean of each predictor over this subset
rooms8 = Boston[Boston$rm > 8,]
rooms8
```

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

```
df = data.frame(mean(rooms8[,1]))
for (i in 2:14){
  df = data.frame(df, mean(rooms8[,i]))
}
names(df) = names(Boston)
df[2,] = rep(0,14)
append(df, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0));
```

```
## $crim
## [1] 0.7187954 0.0000000
##
## $zn
## [1] 13.61538  0.00000
##
## $indus
## [1] 7.078462 0.000000
##
## $chas
## [1] 0.1538462 0.0000000
##
```

```
## $nox
## [1] 0.5392385 0.0000000
##
## $rm
## [1] 8.348538 0.000000
##
## $age
## [1] 71.53846  0.00000
##
## $dis
## [1] 3.430192 0.000000
##
## $rad
## [1] 7.461538 0.000000
##
## $tax
## [1] 325.0769   0.0000
##
## $ptratio
## [1] 16.36154  0.00000
##
## $black
## [1] 385.2108   0.0000
##
## $lstat
## [1] 4.31 0.00
##
## $medv
## [1] 44.2  0.0
##
## [[15]]
## [1] 0
##
## [[16]]
## [1] 0
##
## [[17]]
## [1] 0
##
## [[18]]
## [1] 0
##
## [[19]]
## [1] 0
##
## [[20]]
## [1] 0
##
## [[21]]
## [1] 0
##
## [[22]]
## [1] 0
##
```

```
## [[23]]
## [1] 0
##
## [[24]]
## [1] 0
##
## [[25]]
## [1] 0
##
## [[26]]
## [1] 0
##
## [[27]]
## [1] 0
##
## [[28]]
## [1] 0
```

```
row.names(df) = c("mean", "quantile")

# STEP 2: Calculate the percentile of each mean within the distribution of the corresponding predictor
for (i in 1:14){
  quantile = ecdf(Boston[,i])
  df[2,i] = quantile(df[1,i])
}
df
```

```
##               crim         zn     indus      chas       nox        rm
## mean     0.7187954 13.6153846 7.0784615 0.1538462 0.5392385 8.3485385
## quantile 0.6264822  0.7549407 0.3992095 0.9308300 0.5375494 0.9901186
##                 age       dis       rad         tax    ptratio       black
## mean     71.5384615 3.4301923 7.4615385 325.0769231 16.3615385 385.2107692
## quantile  0.4486166 0.5395257 0.6916996   0.4743083  0.1778656   0.3557312
##               lstat       medv
## mean     4.31000000 44.2000000
## quantile 0.07509881  0.9545455
```

These towns have a very high median value. Many of them are on the Charleston River. They have a low **ptratio**, which means that there are many teachers for each student. Only a very small percentage of the population is of lower status. The crime is middling. This implies that these neighborhoods are wealthy, perhaps with many apartment buildings. It would be interesting to know if the crimes being committed in these suburbs are property crimes, as it's a common phenomenon for wealthy neighborhoods near big cities to suffer from such crime. It's surprising that the mean property tax is so low.