

Chapter 2 Lab: Introduction to R

Divya Hegde

16th March,24

Instructions and time-saving hints. To begin this lab, download the archive and unpack it. Inside the folder, you will find a Rmarkdown file and csv files for lab data. Please keep the Rmarkdown file in the same directory as the data files.

The purpose of the first part of this lab is to quickly review the basics of R. To review the basics, you will reproduce the commands blocks in JW Section 2.3 in this Rmarkdown file. **Please do not retype all the R commands as shown in Section 2.3.** Instead, go to link to text for Lab 2, cut and paste the text into your Rmarkdown file. Then break up the R commands into the R chunks shown in the text. (R chunks are braced by triple backticks and a leading {r}, as before.) Do not put all commands into the same block! The idea is to imitate the code chunks in Section 2.3 with the additional plots and output omitted from the text. Below, I have given you the first two blocks for reference.

After breaking the text into the correct chunks, you may knit the document. You will notice two problems when knitting. The first is the appearance of a data editor window (3 times), due to the `fix()` commands in the blocks. You may simply quit these windows each time, or comment out the `fix()` statements. The second problem is an error message, due to the block at the top of p. 50 in JW. You may comment out this `plot()` statement to knit again.

Congratulations! When the document is successfully knitted, you should read Section 2.3 along with the Rmarkdown output. You should see the commands along with the output plots. Of course, the blocks with help commands, `fix()` commands, and intentional errors will not be reproduced. Together, JW2.3 and your Rmarkdown file will help you to review some basics of R. Note that the plotting is done with core R commands and not with the `ggplot2` package as in Homework 0.

Next, you will apply these core R basics to the `College.csv` dataset, as described in problem 8, JW p.54. Please complete the Rmarkdown document corresponding to problem statement. Submit your Rmd file along with an unzipped PDF of the result before the deadline.

Basic Commands

```
x <- c(1,3,2,5)
x
```

```
## [1] 1 3 2 5
```

```
x <- c(1,3,2,5)
x
```

```
## [1] 1 3 2 5
```

```

x = c(1,6,2)
x

## [1] 1 6 2

y = c(1,4,3)
length(x)

## [1] 3

length(y)

## [1] 3

x+y

## [1] 2 10 5

ls()

## [1] "x" "y"

rm(x,y)
ls()

## character(0)

rm(list=ls())
?matrix
x=matrix(data=c(1,2,3,4), nrow=2, ncol=2)
x

##      [,1] [,2]
## [1,]     1     3
## [2,]     2     4

x=matrix(c(1,2,3,4),2,2)
matrix(c(1,2,3,4),2,2,byrow=TRUE)

##      [,1] [,2]
## [1,]     1     2
## [2,]     3     4

sqrt(x)

##      [,1]      [,2]
## [1,] 1.000000 1.732051
## [2,] 1.414214 2.000000

```

```
x^2
```

```
##      [,1] [,2]
## [1,]     1    9
## [2,]     4   16
```

```
x=rnorm(50)
y=x+rnorm(50,mean=50,sd=.1)
cor(x,y)
```

```
## [1] 0.995754
```

```
set.seed(1303)
rnorm(50)
```

```
## [1] -1.1439763145  1.3421293656  2.1853904757  0.5363925179  0.0631929665
## [6]  0.5022344825 -0.0004167247  0.5658198405 -0.5725226890 -1.1102250073
## [11] -0.0486871234 -0.6956562176  0.8289174803  0.2066528551 -0.2356745091
## [16] -0.5563104914 -0.3647543571  0.8623550343 -0.6307715354  0.3136021252
## [21] -0.9314953177  0.8238676185  0.5233707021  0.7069214120  0.4202043256
## [26] -0.2690521547 -1.5103172999 -0.6902124766 -0.1434719524 -1.0135274099
## [31]  1.5732737361  0.0127465055  0.8726470499  0.4220661905 -0.0188157917
## [36]  2.6157489689 -0.6931401748 -0.2663217810 -0.7206364412  1.3677342065
## [41]  0.2640073322  0.6321868074 -1.3306509858  0.0268888182  1.0406363208
## [46]  1.3120237985 -0.0300020767 -0.2500257125  0.0234144857  1.6598706557
```

```
set.seed(3)
y=rnorm(100)
mean(y)
```

```
## [1] 0.01103557
```

```
var(y)
```

```
## [1] 0.7328675
```

```
sqrt(var(y))
```

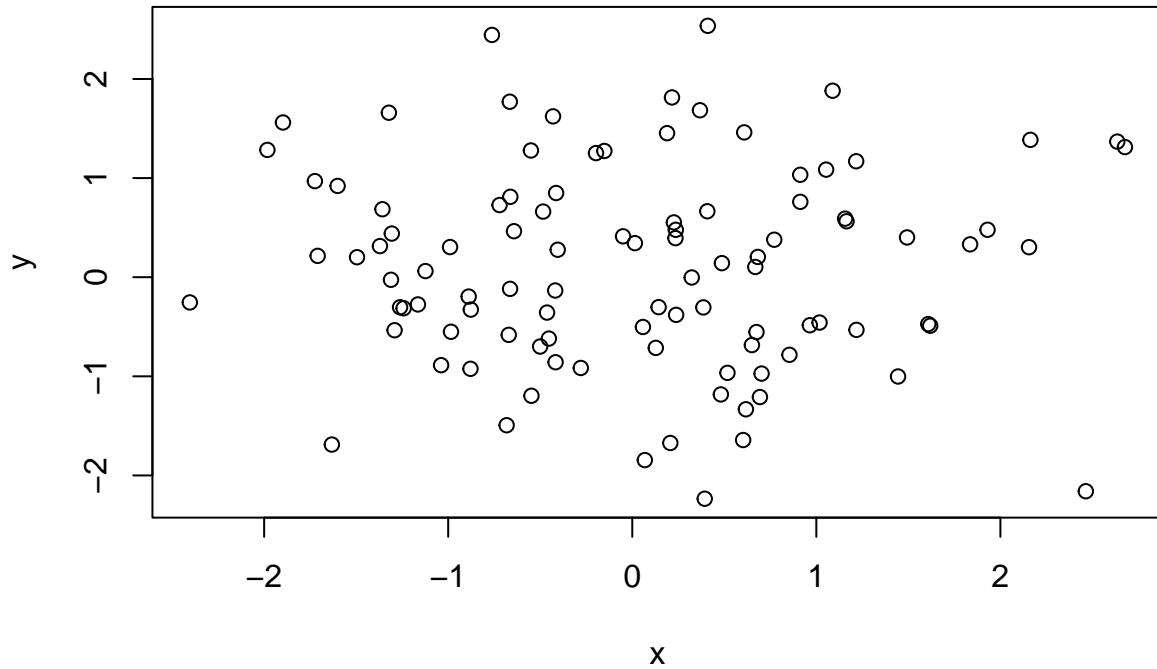
```
## [1] 0.8560768
```

```
sd(y)
```

```
## [1] 0.8560768
```

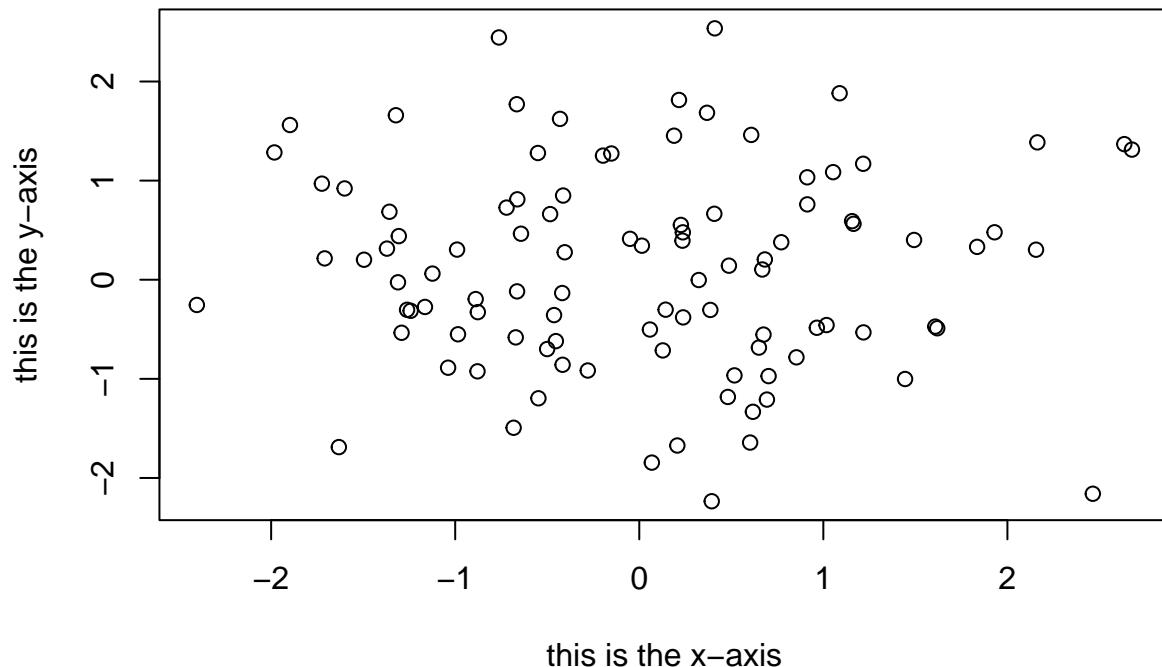
Graphics

```
x=rnorm(100)
y=rnorm(100)
plot(x,y)
```



```
plot(x,y,xlab="this is the x-axis",ylab="this is the y-axis",main="Plot of
X vs Y")
```

Plot of X vs Y



```
pdf("Figure.pdf")
plot(x,y,col="green")
dev.off()
```

```
## pdf
## 2
```

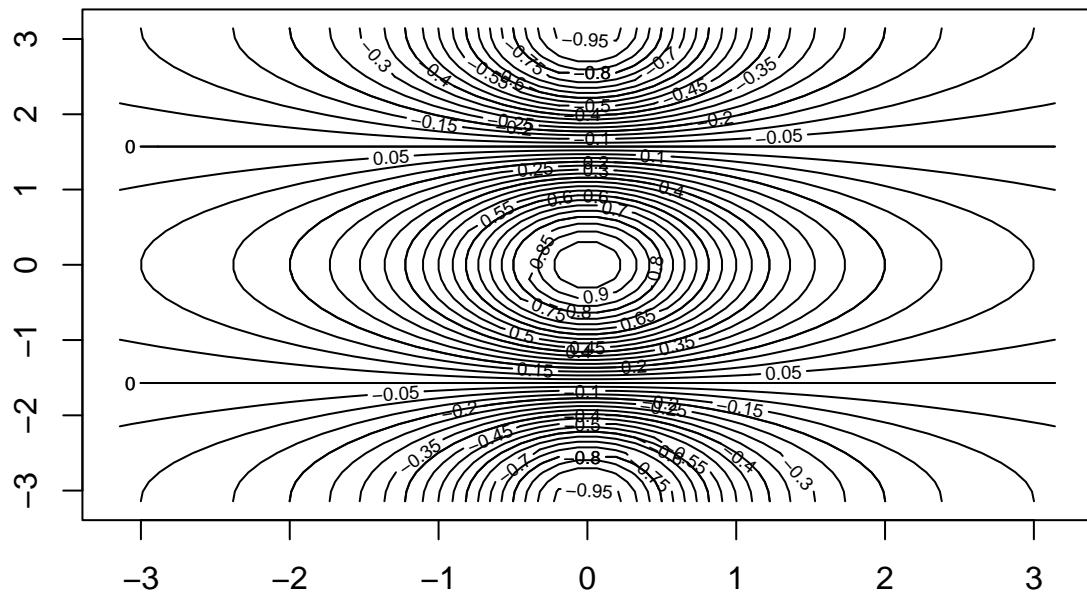
```
x=seq(1,10)
x
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

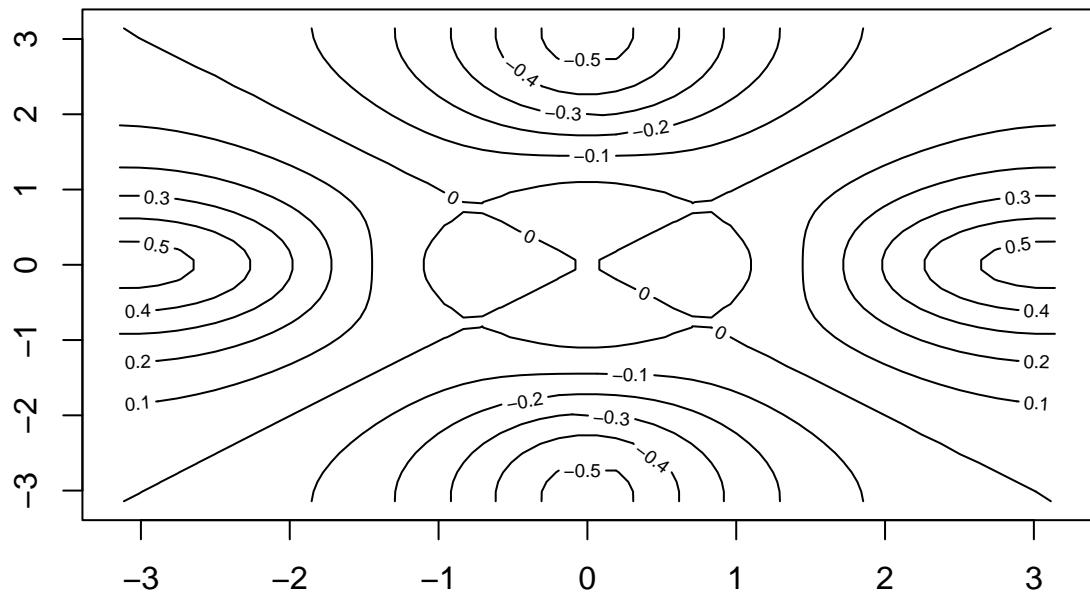
```
x=1:10
x
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

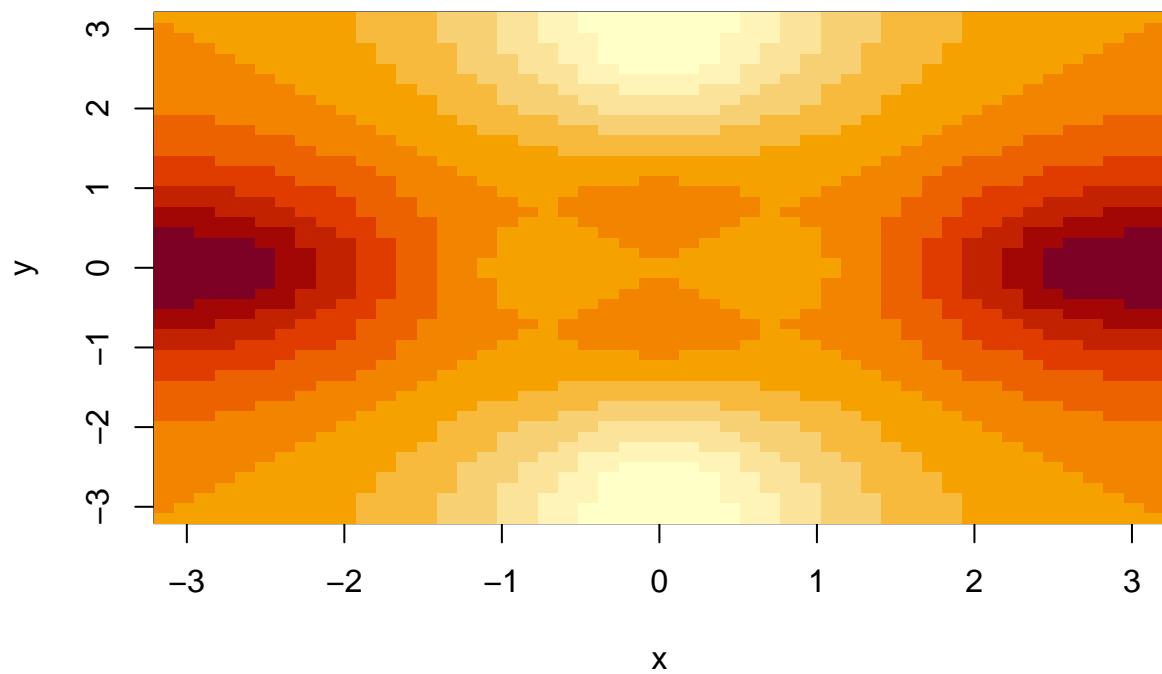
```
x=seq(-pi,pi,length=50)
y=x
f=outer(x,y,function(x,y)cos(y)/(1+x^2))
contour(x,y,f)
contour(x,y,f,nlevels=45,add=T)
```



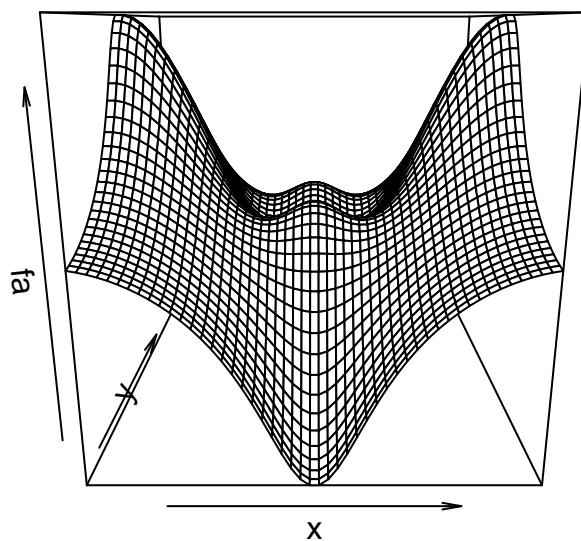
```
fa=(f-t(f))/2  
contour(x,y,fa,nlevels=15)
```



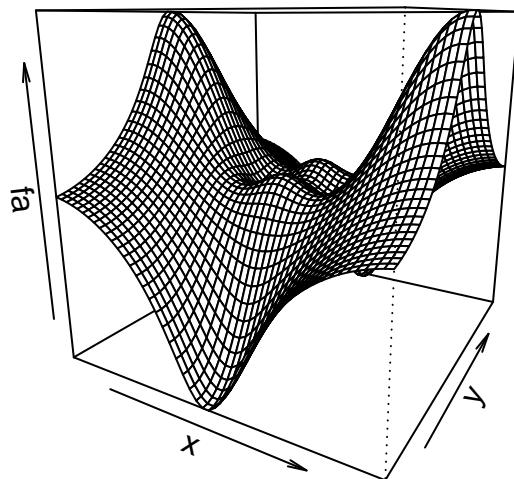
```
image(x,y,fa)
```



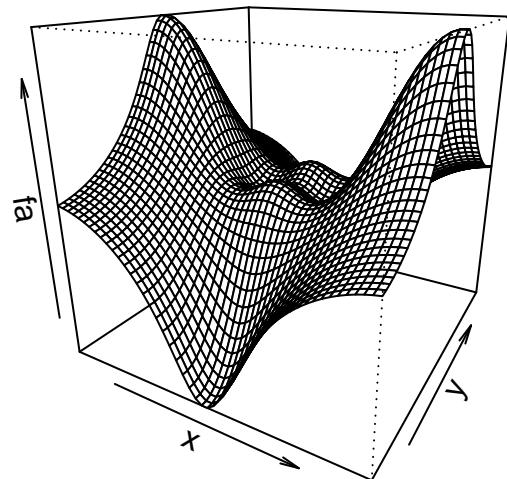
```
persp(x,y,fa)
```



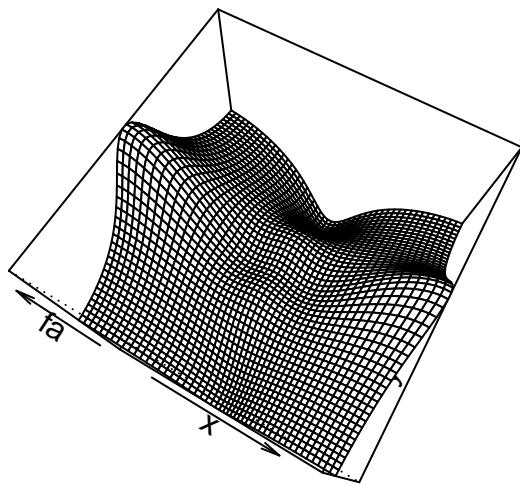
```
persp(x,y,fa,theta=30)
```



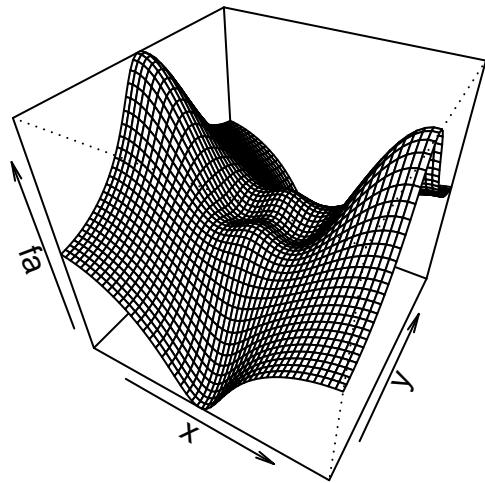
```
persp(x,y,fa,theta=30,phi=20)
```



```
persp(x,y,fa,theta=30,phi=70)
```



```
persp(x,y,fa,theta=30,phi=40)
```



Indexing Data

```
A=matrix(1:16,4,4)
A
```

```
##      [,1] [,2] [,3] [,4]
## [1,]     1    5    9   13
## [2,]     2    6   10   14
## [3,]     3    7   11   15
## [4,]     4    8   12   16
```

```
A[2,3]
```

```
## [1] 10
```

```
A[c(1,3),c(2,4)]
```

```
##      [,1] [,2]
## [1,]     5   13
## [2,]     7   15
```

```
A[1:3,2:4]
```

```
##      [,1] [,2] [,3]
## [1,]     5    9   13
## [2,]     6   10   14
## [3,]     7   11   15
```

```
A[1:2,]
```

```
##      [,1] [,2] [,3] [,4]
## [1,]     1    5    9   13
## [2,]     2    6   10   14
```

```
A[,1:2]
```

```
##      [,1] [,2]
## [1,]     1    5
## [2,]     2    6
## [3,]     3    7
## [4,]     4    8
```

```
A[1,]
```

```
## [1] 1 5 9 13
```

```
A[-c(1,3),]
```

```
##      [,1] [,2] [,3] [,4]
## [1,]     2    6   10   14
## [2,]     4    8   12   16
```

```
A[-c(1,3),-c(1,3,4)]
```

```
## [1] 6 8
```

```
dim(A)
```

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

Loading Data

```
Auto=read.table("Auto.data")
fix(Auto)
Auto=read.table("Auto.data",header=T,na.strings="?")
fix(Auto)
Auto=read.csv("Auto.csv",header=T,na.strings="?")
fix(Auto)
dim(Auto)
```

```

## [1] 397   9

Auto[1:4,]

##   mpg cylinders displacement horsepower weight acceleration year origin
## 1 18         8          307        130    3504       12.0     70      1
## 2 15         8          350        165    3693       11.5     70      1
## 3 18         8          318        150    3436       11.0     70      1
## 4 16         8          304        150    3433       12.0     70      1
##                               name
## 1 chevrolet chevelle malibu
## 2          buick skylark 320
## 3      plymouth satellite
## 4          amc rebel sst

Auto=na.omit(Auto)
dim(Auto)

## [1] 392   9

names(Auto)

## [1] "mpg"           "cylinders"      "displacement"   "horsepower"    "weight"
## [6] "acceleration"  "year"          "origin"         "name"

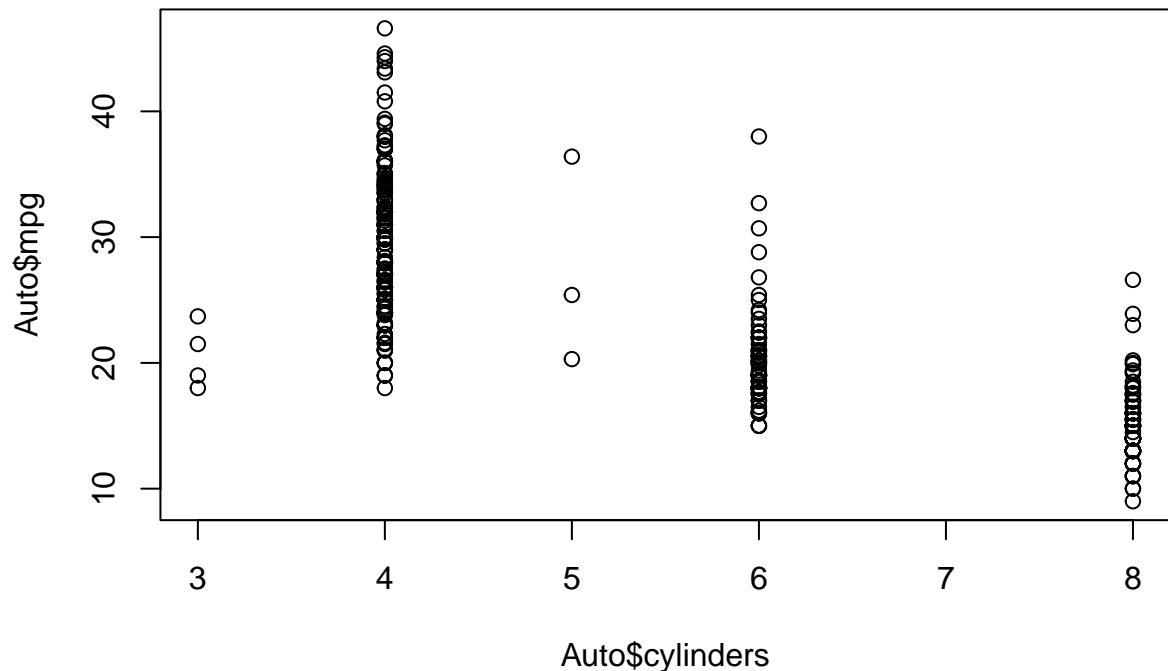
```

Additional Graphical and Numerical Summaries

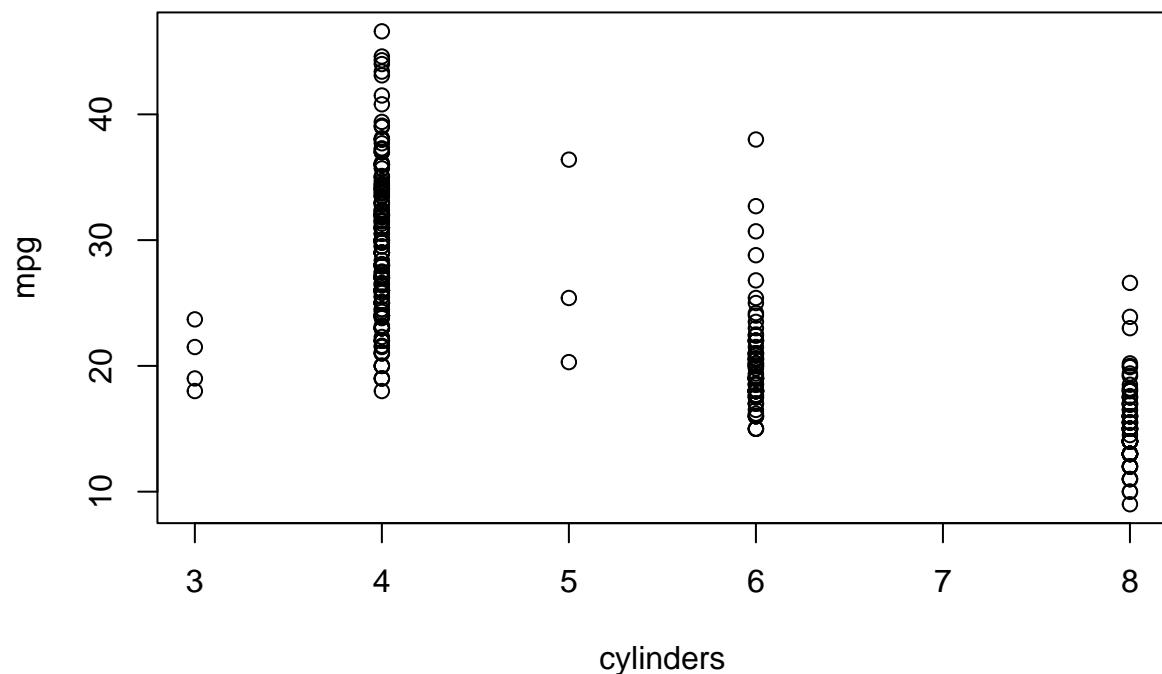
```

#plot(cylinders, mpg)
plot(Auto$cylinders, Auto$mpg)

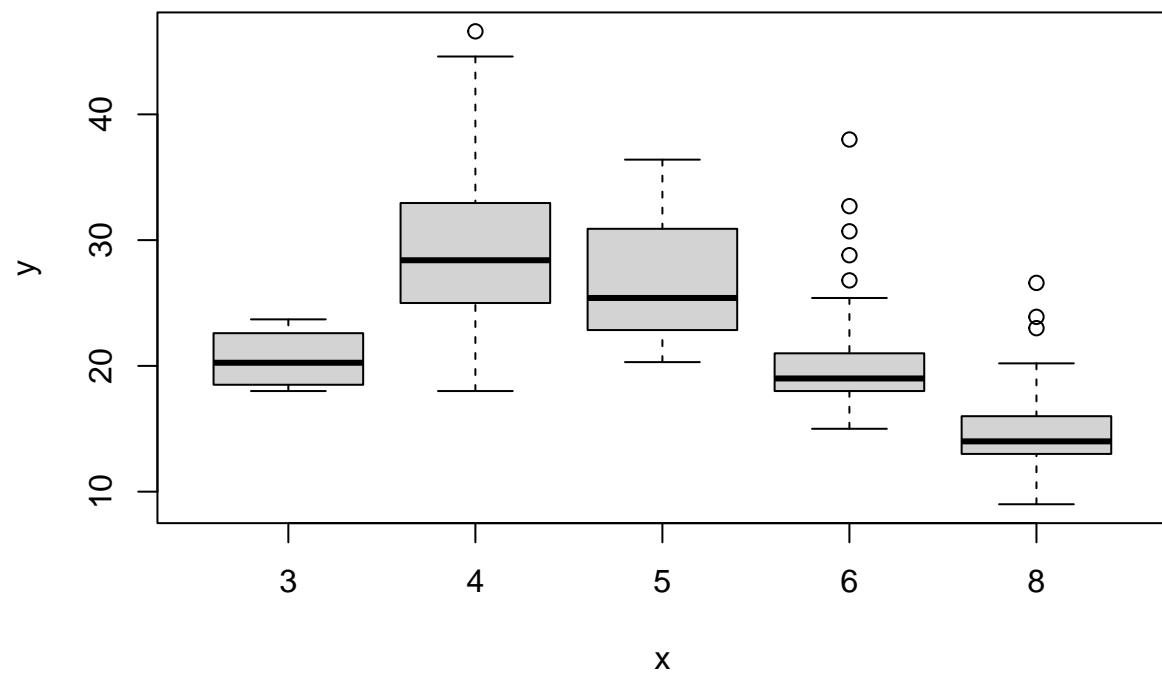
```



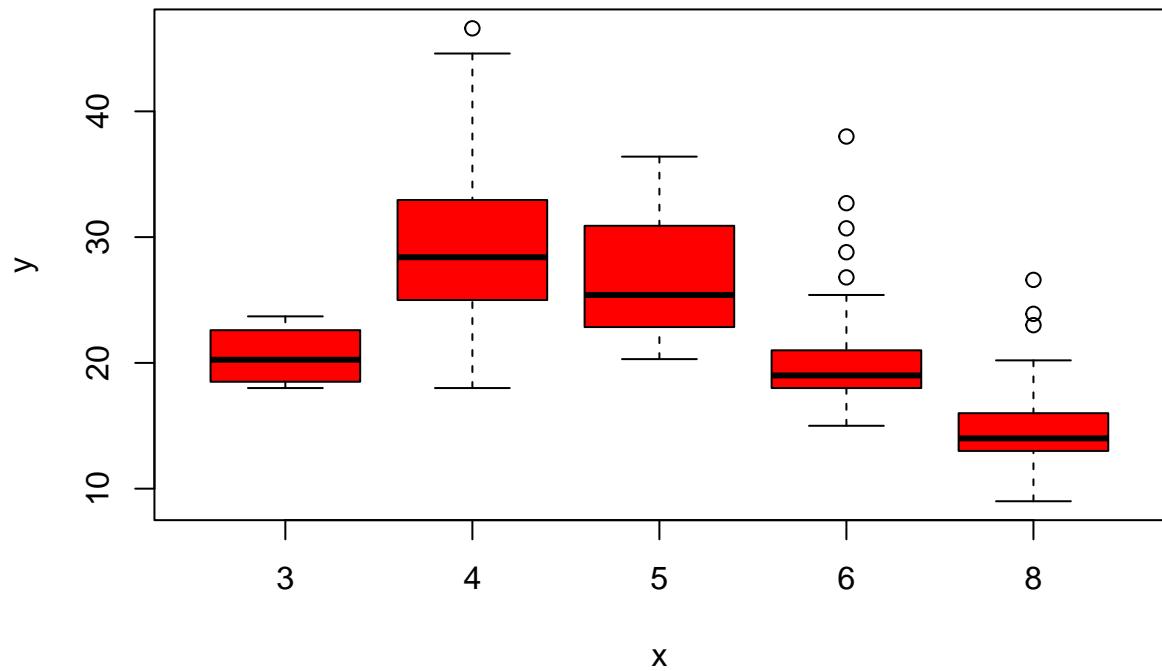
```
attach(Auto)
plot(cylinders, mpg)
```



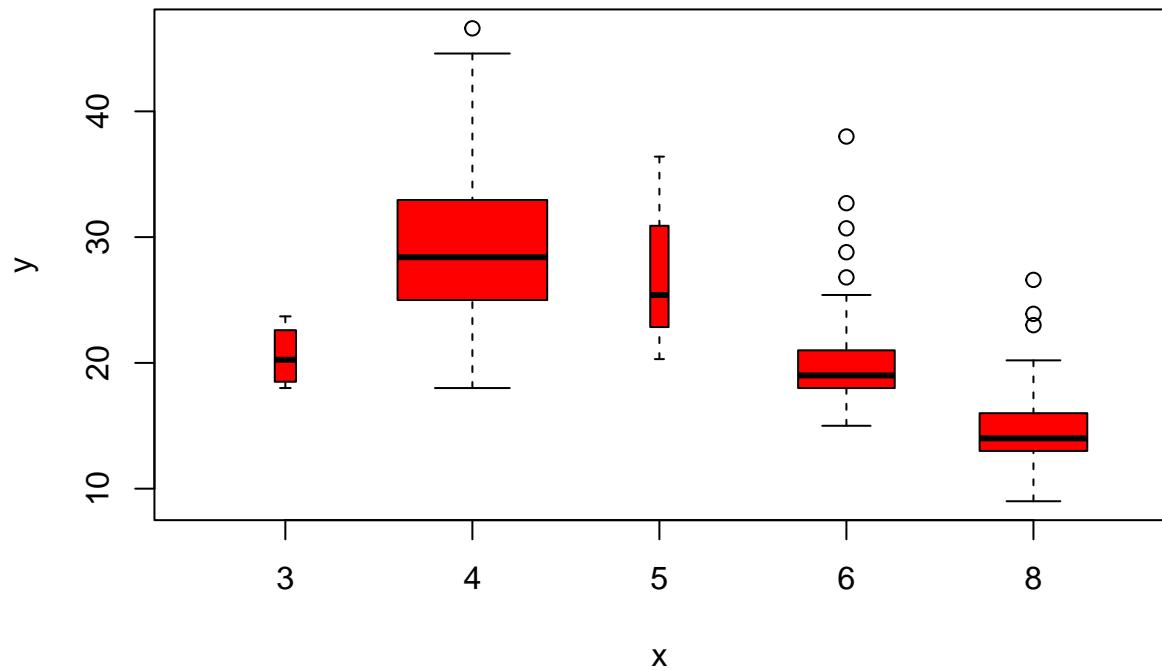
```
cylinders=as.factor(cylinders)
plot(cylinders, mpg)
```



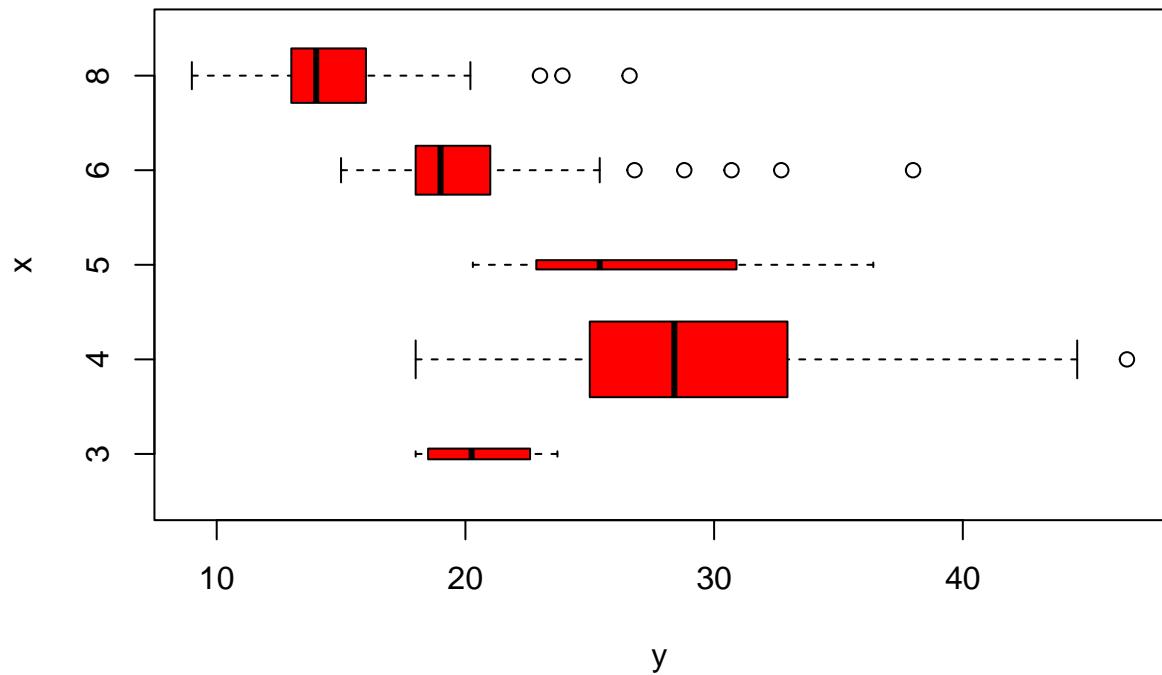
```
plot(cylinders, mpg, col="red")
```



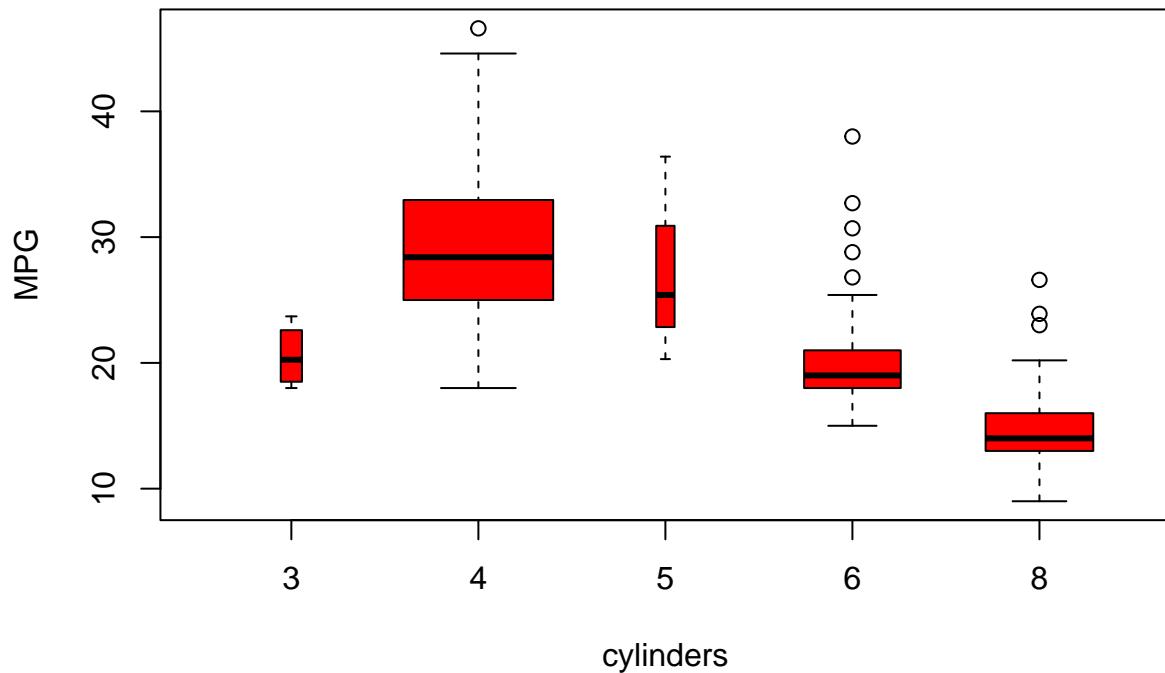
```
plot(cylinders, mpg, col="red", varwidth=T)
```



```
plot(cylinders, mpg, col="red", varwidth=T, horizontal=T)
```

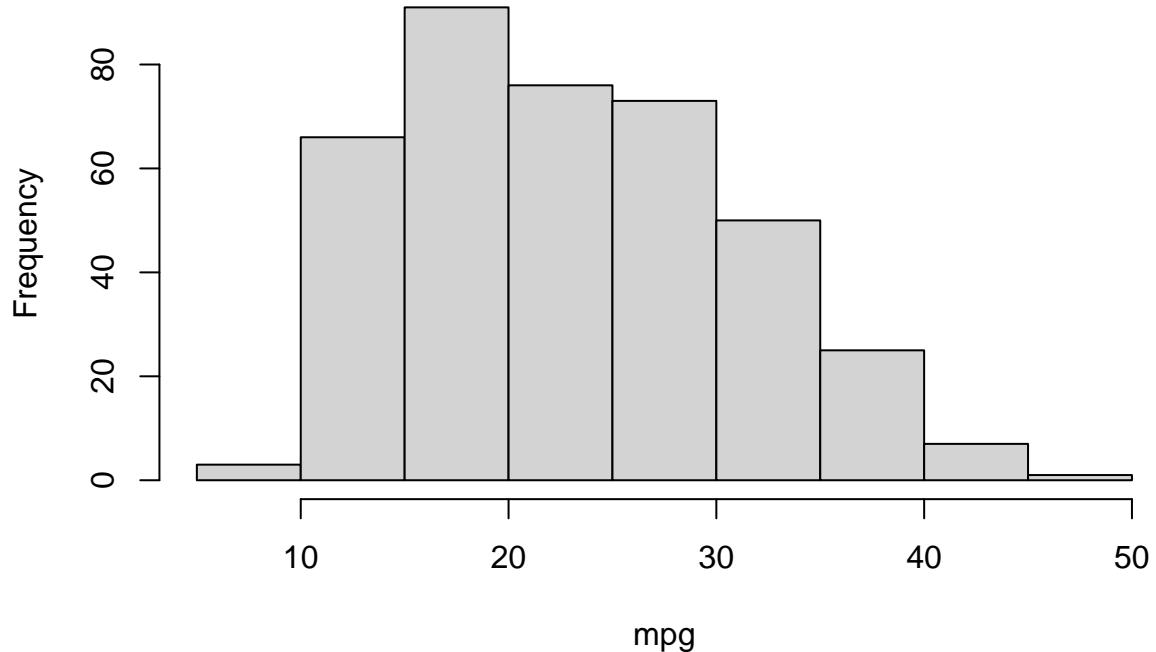


```
plot(cylinders, mpg, col="red", varwidth=T, xlab="cylinders", ylab="MPG")
```



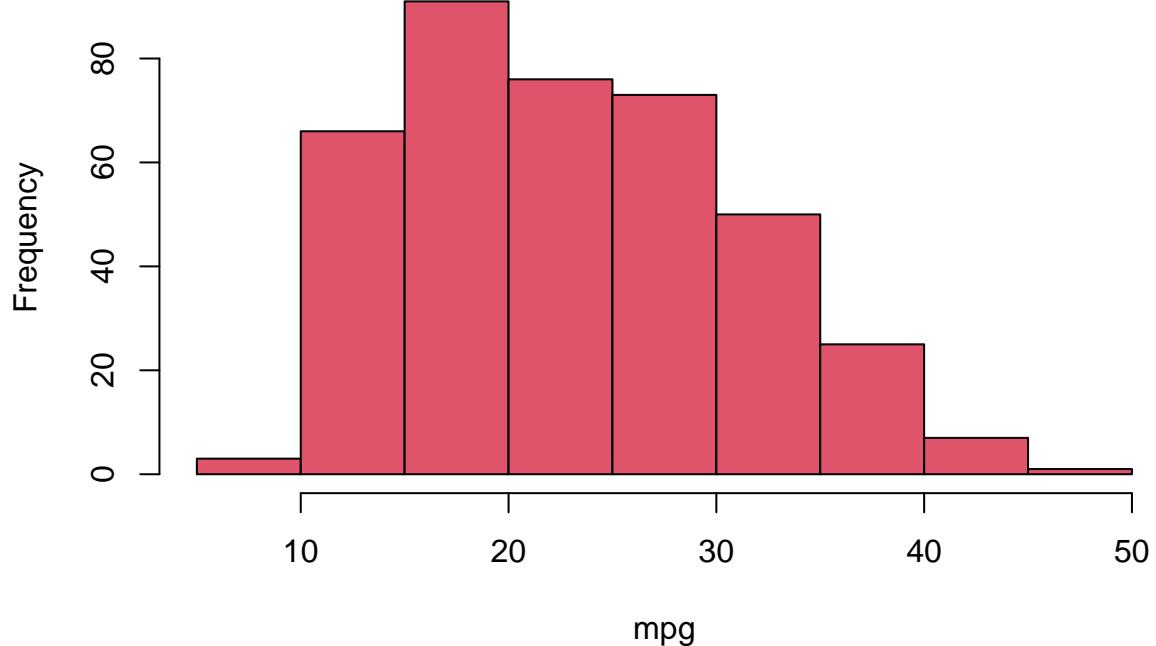
```
hist(mpg)
```

Histogram of mpg



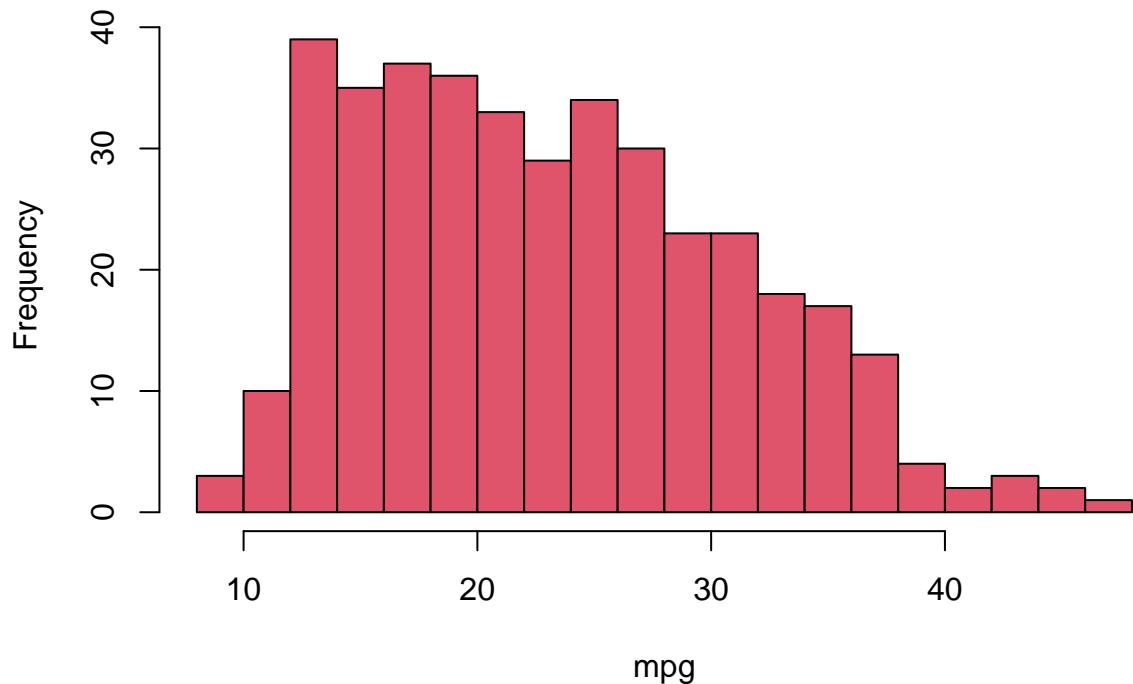
```
hist(mpg,col=2)
```

Histogram of mpg

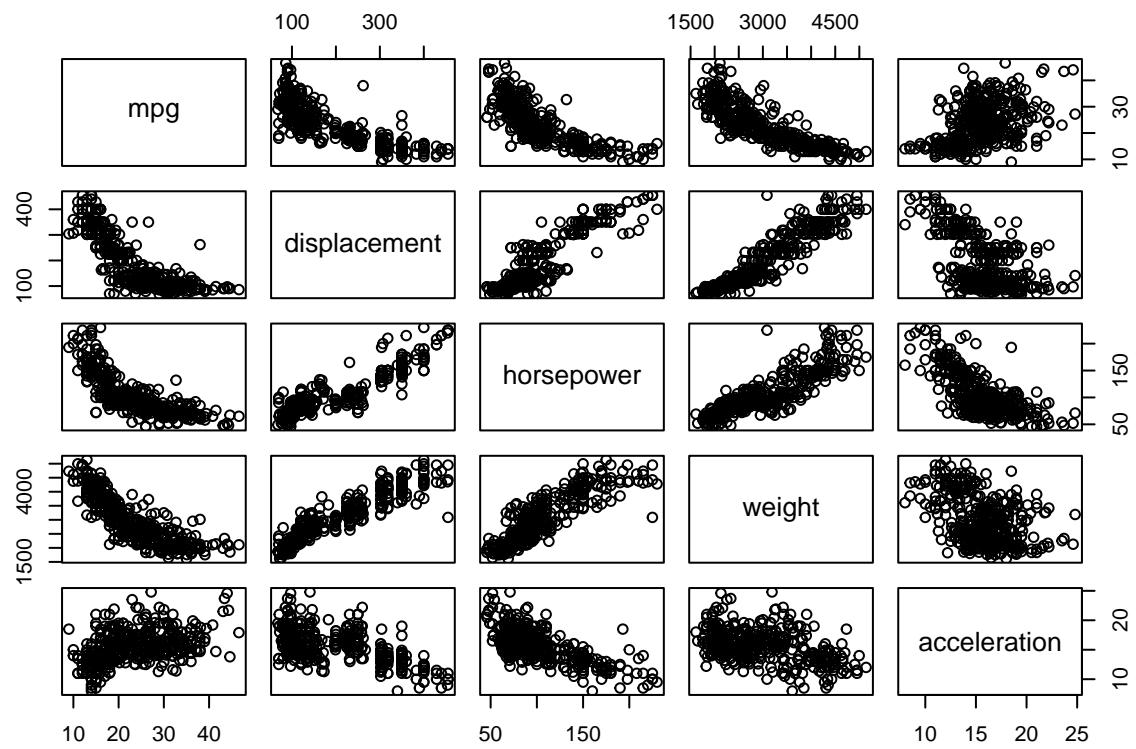


```
hist(mpg,col=2,breaks=15)
```

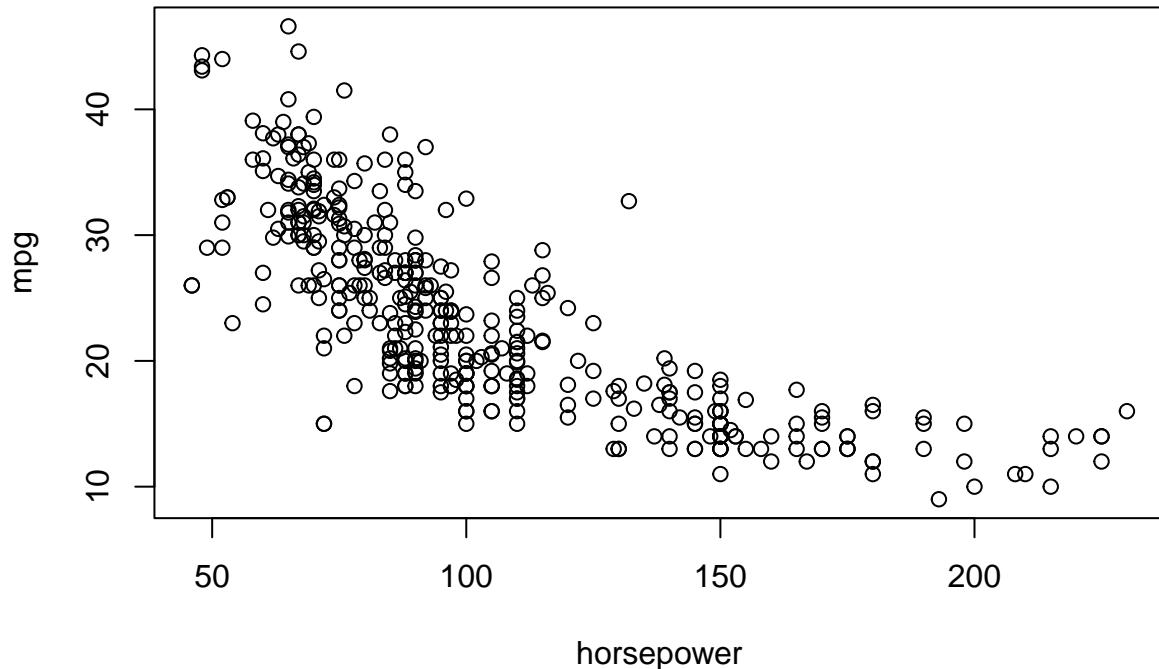
Histogram of mpg



```
#pairs(Auto)
pairs(~ mpg + displacement + horsepower + weight + acceleration, Auto)
```



```
plot(horsepower,mpg)
identify(horsepower,mpg,name)
```



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

```
summary(Auto)
```

```
##      mpg      cylinders      displacement      horsepower      weight
##  Min.   : 9.00  Min.   :3.000  Min.   :68.0  Min.   :46.0  Min.   :1613
##  1st Qu.:17.00  1st Qu.:4.000  1st Qu.:105.0 1st Qu.:75.0  1st Qu.:2225
##  Median :22.75  Median :4.000  Median :151.0  Median :93.5  Median :2804
##  Mean   :23.45  Mean   :5.472  Mean   :194.4  Mean   :104.5  Mean   :2978
##  3rd Qu.:29.00  3rd Qu.:8.000  3rd Qu.:275.8  3rd Qu.:126.0 3rd Qu.:3615
##  Max.   :46.60  Max.   :8.000  Max.   :455.0  Max.   :230.0  Max.   :5140
##      acceleration      year      origin      name
##  Min.   : 8.00  Min.   :70.00  Min.   :1.000  Length:392
##  1st Qu.:13.78  1st Qu.:73.00  1st Qu.:1.000  Class :character
##  Median :15.50  Median :76.00  Median :1.000  Mode   :character
##  Mean   :15.54  Mean   :75.98  Mean   :1.577
##  3rd Qu.:17.02  3rd Qu.:79.00  3rd Qu.:2.000
##  Max.   :24.80  Max.   :82.00  Max.   :3.000
```

```
summary(mpg)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##  9.00    17.00  22.75  23.45  29.00  46.60
```

```
#Applied JW p.54 (8.)##
```

Here, we are using the `College` data set, found in `College.csv`. Before reading the data into R, it can be viewed in Excel or a text editor.

1. 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=T,na.strings="?")  
dim(college)
```

```
## [1] 777 19
```

2. 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:

```
#fix(college)  
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 the 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.

3. Please complete these parts.

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

```

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]`.

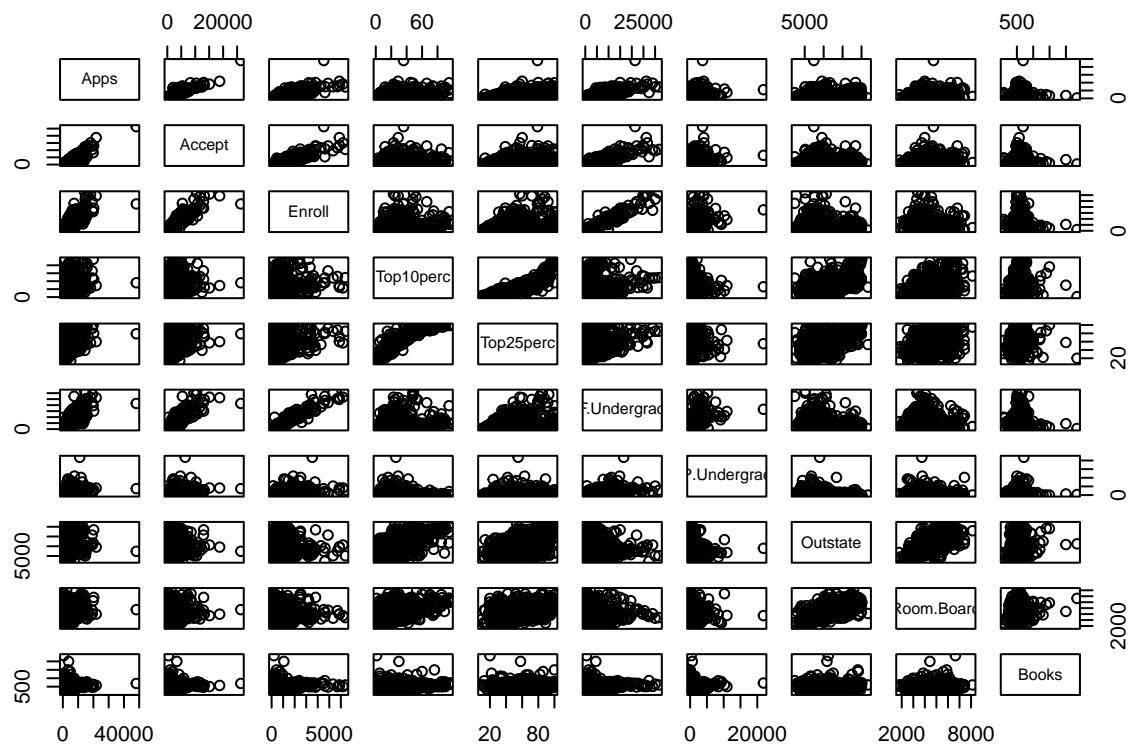
```

#data1 = college[,1:10]
#pairs(data1[,sapply(data1,is.numeric)])
```

```

data = college[,sapply(college,is.numeric)] #gets the first 10 numeric columns
pairs(data[,1:10])

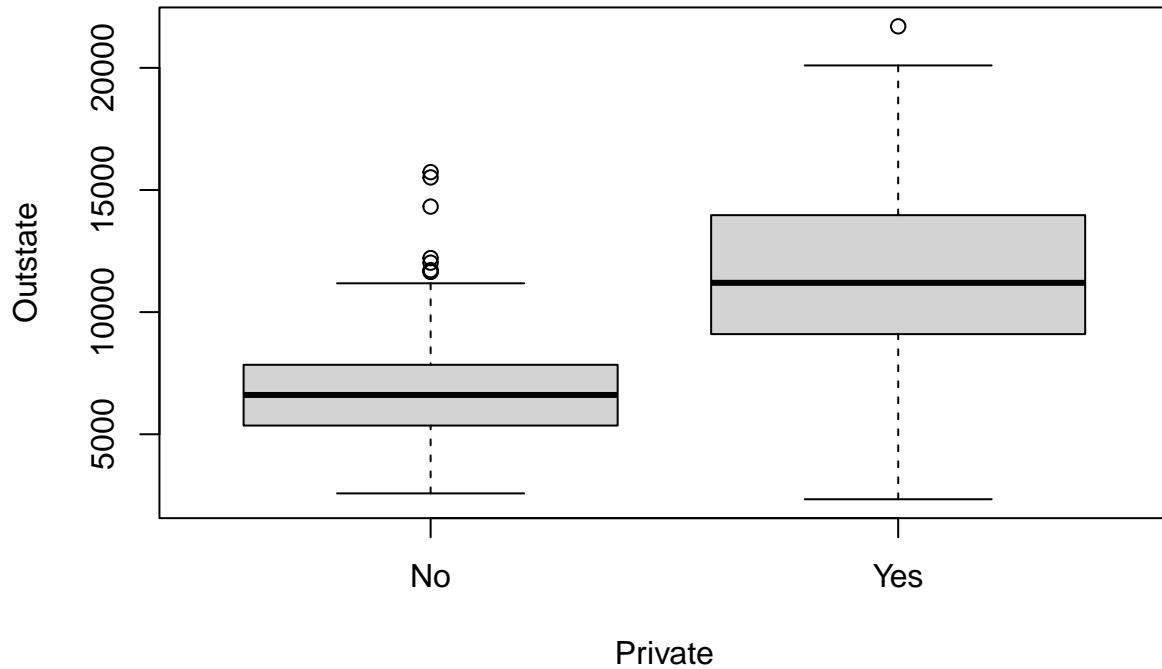
```



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

```
#head(college)

boxplot(Outstate ~ Private, data = college, xlab = "Private", ylab = "Outstate")
```



```
#
```

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)
```

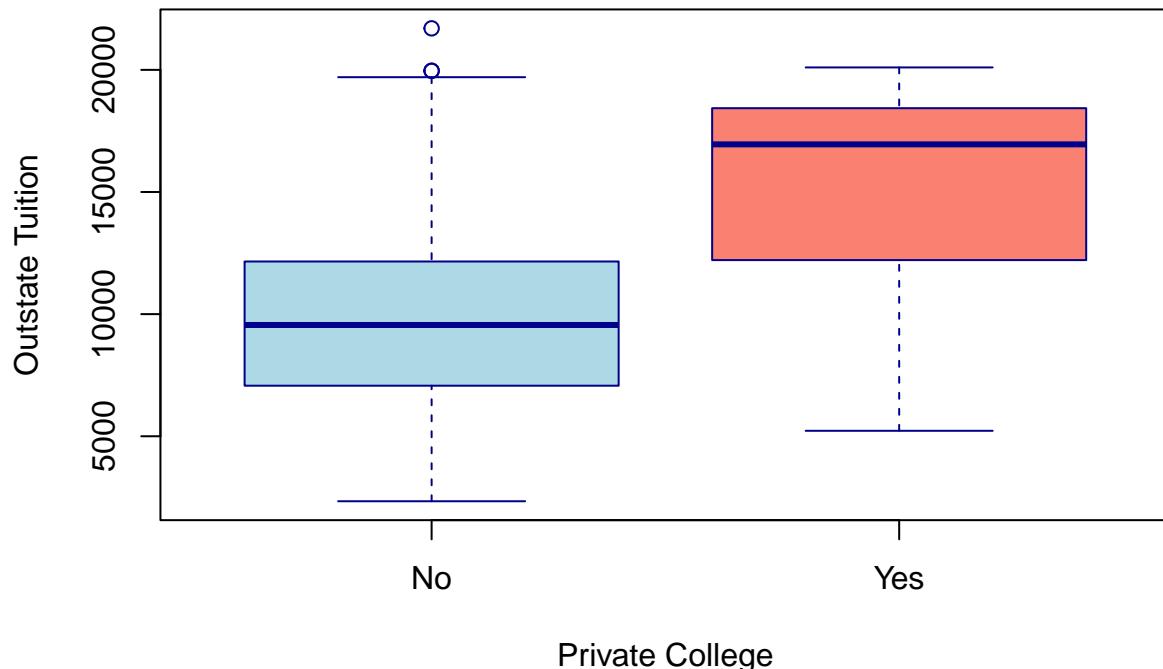
Use the `summary()` function to see how many elite universities there are. Now use the `plot()` function to produce side-by-side boxplots of `Outstate` versus `Elite`.

```
#  
summary(Elite)
```

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

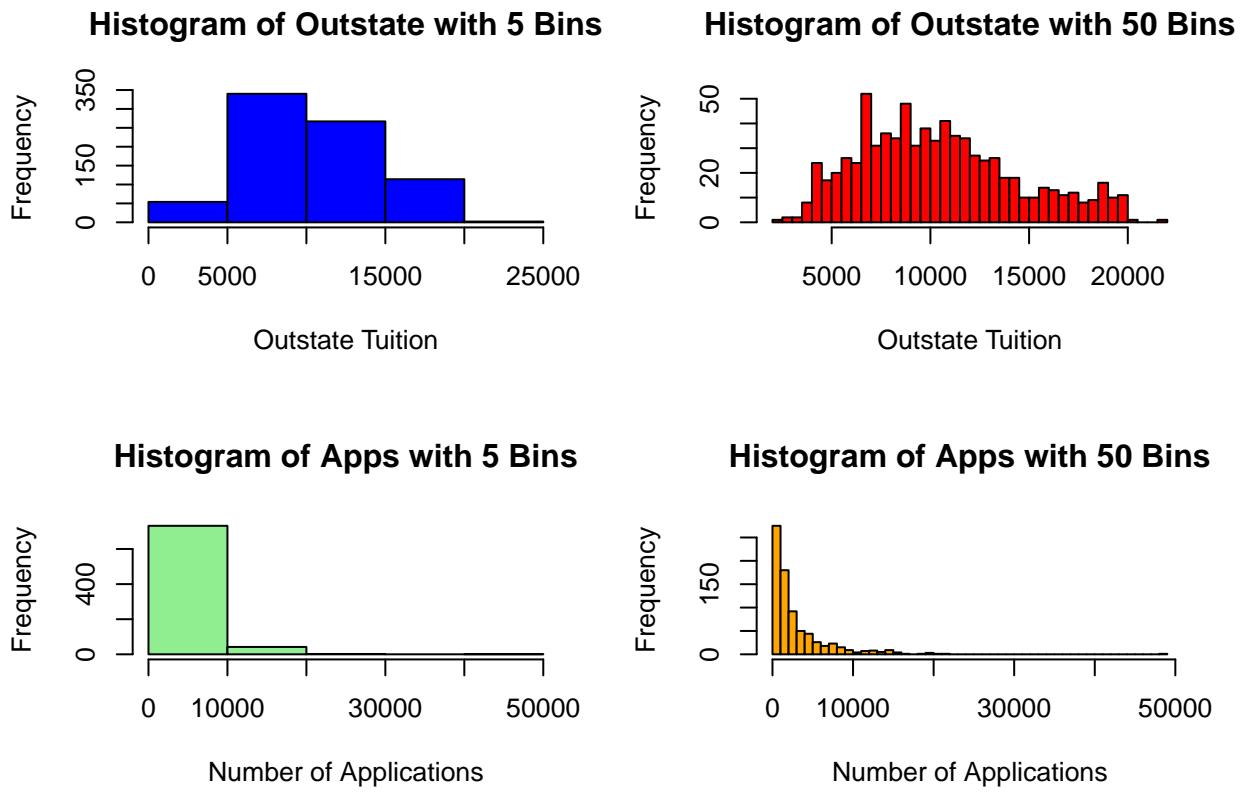
```
boxplot(Outstate ~ Elite, data = college, main = "Outstate Tuition by College Type", xlab = "Private Co."
```

Outstate Tuition by College Type



Use the `hist()` function to produce some histograms with 5 and 50 bins for `Outstate` and `Apps`. Use the command `par(mfrow=c(2,2))`: it will divide the print window into four regions so that four plots can be made simultaneously.

```
#  
par(mfrow=c(2,2))  
hist(college$Outstate, breaks=5, main="Histogram of Outstate with 5 Bins", xlab="Outstate Tuition", col="blue")  
hist(college$Outstate, breaks=50, main="Histogram of Outstate with 50 Bins", xlab="Outstate Tuition", col="blue")  
hist(college$Apps, breaks=5, main="Histogram of Apps with 5 Bins", xlab="Number of Applications", col="red")  
hist(college$Apps, breaks=50, main="Histogram of Apps with 50 Bins", xlab="Number of Applications", col="red")
```



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

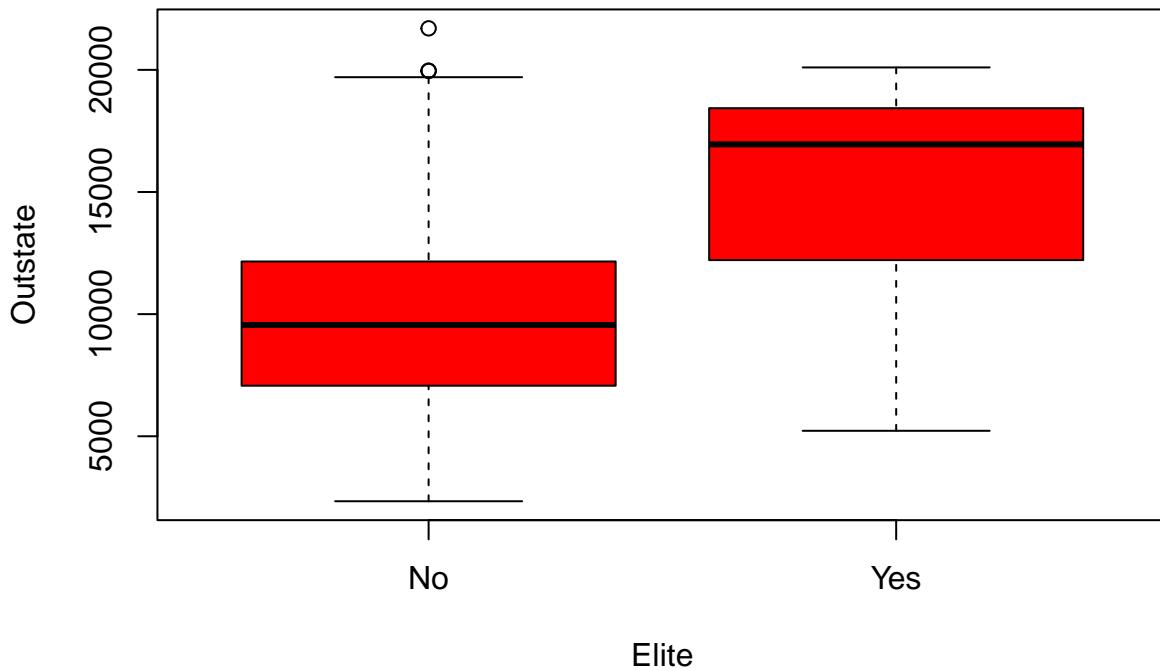
This statement is intentionally vague: as a data scientist, it will be your job to propose hypotheses about the data, and then to use the data to address your hypotheses. This is a creative, iterative, (and fun) process.

In this first lab I will propose four hypotheses for you. You must address these hypotheses with the suggested approaches. The approach will be to produce one plot for each hypothesis in a 2 by 2 set of plots. Discuss your findings for each case, including: whether the data supports or rejects the hypothesis, and to what degree the hypothesis is rejected or supported. After addressing the four hypotheses, I would like for you to propose at least one more hypothesis, and develop a methodology. Brevity matters, and less is best. Produce a single plot upon which to base your answer. Long answers will be given little credit. For this last part, you will be graded on the “interest” of the hypothesis, your approach to address it, and your discussion.

Hypothesis 1. The tuition at the best colleges, as indicated by Elite, is higher than that at other colleges.
Methodology: Use one-over-other horizontal red boxplots of `Outstate` for elite and non-elite colleges. Title and label the axes of this plot thoughtfully. Carefully use this single plot to address this hypothesis. Can you be precise about the word *higher* ?

```
color <- "red"
boxplot(Outstate ~ Elite, data = college, xlab = "Elite", ylab = "Outstate", main = "Outstate vs Elite")
```

Outstate vs Elite/Non-Elite colleges



#We can see that Elite universities have a much better tuition compared to non-elite. This hypothesis is true.

The enrollment rate is the fraction of accepted students who enrolled. *Hypothesis 2. The enrollment rate at the best colleges, as indicated by Elite, is higher than that at other colleges.* **Methodology:** Create a new variable called EnrollRate, using Enroll and Accept. Use the attach() function to make your code cleaner. Use one-over-other horizontal green boxplots of EnrollRate for elite and non-elite colleges. Title and label the axes of this plot thoughtfully. Carefully use this single plot to address this hypothesis. Can you be precise about the word *higher* ? Can you comment on the presence of outliers in the boxplots?

```
color <- "green"
attach(college)
```

```
## The following object is masked _by_ .GlobalEnv:
##
##      Elite
```

```
EnrollRate <- Enroll / Accept
print(EnrollRate)
```

```
## [1] 0.58522727 0.26611227 0.30628988 0.39255014 0.37671233 0.32985386
## [7] 0.30294118 0.28430233 0.27056019 0.34538153 0.33122807 0.25473684
## [13] 0.37179487 0.35648148 0.50159744 0.20128088 0.42137097 0.46585903
## [19] 0.45738636 0.50774613 0.38124054 0.40951973 0.42441860 0.36486224
## [25] 0.49700599 0.55002892 0.28882353 0.45206891 0.50097466 0.29975875
```

```

## [31] 0.42206897 0.36967419 0.30935252 0.42650104 0.48462665 0.57812500
## [37] 0.34009547 0.37874465 0.35586735 0.44251262 0.40941176 0.43564356
## [43] 0.26113360 0.49383984 0.30769231 0.62257697 0.35806452 0.34862385
## [49] 0.27467811 0.39312039 0.24509804 0.66304348 0.27083333 0.47130435
## [55] 0.48809524 0.46428571 0.33850727 0.50558659 0.40661479 0.29291920
## [61] 0.41020608 0.41947361 0.45757576 0.31077914 0.26977762 0.57664234
## [67] 0.97882736 0.44159544 0.43387816 0.85431322 0.45137388 0.38641975
## [73] 0.22606871 0.41424419 0.34364261 0.45344130 0.35264901 1.00000000
## [79] 0.43227666 0.45021251 0.60383598 0.49066468 0.50766610 0.34336525
## [85] 0.32489146 0.38888889 0.30968968 0.22899442 0.35519677 0.48790747
## [91] 0.30413455 0.22591888 0.38617021 0.33068182 0.34470990 0.15351005
## [97] 0.32619440 0.56513761 0.28846154 0.48387097 0.51262626 0.36028751
## [103] 0.35624013 0.35015850 0.50074590 0.60273973 0.32432432 0.45525292
## [109] 0.46192893 0.49606299 0.70833333 0.43076923 0.55874674 0.71592539
## [115] 0.29595828 0.22195240 0.49117647 0.28520225 0.43770211 0.44563280
## [121] 0.37061995 0.65265487 0.34571645 0.29173355 0.35698198 0.40286624
## [127] 0.38387097 0.27314815 0.36742424 0.49719101 0.59143519 0.45255474
## [133] 0.33983287 0.37500000 0.42281879 0.46610169 0.46644295 0.38971023
## [139] 0.39182228 0.29368030 0.31071655 0.34759189 0.83544304 0.39413681
## [145] 0.45129534 0.52255639 0.49074074 0.28973843 0.46153846 0.28331177
## [151] 0.36609337 0.28818661 0.30888575 0.24684685 0.50443599 0.25363489
## [157] 0.38381743 0.30118110 0.47822261 0.47280335 0.37744035 0.48571429
## [163] 0.23387451 0.29891304 0.19180780 0.77427184 0.47311047 0.34414669
## [169] 0.32826748 0.58362989 0.49457177 0.32607851 0.20316456 0.48307692
## [175] 0.40662728 0.27236581 0.38274128 0.47728938 0.77712610 0.44715447
## [181] 0.37776289 0.36797555 0.51590909 0.48899756 0.33002705 0.23114487
## [187] 0.33014354 0.60096154 0.30796841 0.37074303 0.34984520 0.29654511
## [193] 0.67914013 0.29982047 0.24889868 0.39782016 0.23341303 0.42481203
## [199] 0.30352304 0.47338936 0.33101266 0.51515152 0.35961538 0.34815156
## [205] 0.51428571 0.32776618 0.39780375 0.49486405 0.50442478 0.44462025
## [211] 0.09975397 0.57299270 0.53284672 0.35110200 0.35534591 0.35698925
## [217] 0.44382022 0.40495868 0.39921405 0.29474516 0.41269841 0.48247136
## [223] 0.50276121 0.52883383 0.45195730 0.23317113 0.60683761 0.34514436
## [229] 0.49911504 0.55808081 0.30504305 0.39018692 0.58454106 0.40159506
## [235] 0.31528662 0.35627530 0.50129199 0.31101512 0.51621622 0.30929791
## [241] 0.45776173 0.43881857 0.25462703 0.39757576 0.47670807 0.38162450
## [247] 0.37873357 0.82547170 0.75468165 0.24927536 0.74180139 0.31118881
## [253] 0.37005650 0.38002774 0.46577181 0.33470508 0.24026910 0.23020478
## [259] 0.43172691 0.31150442 0.42076871 0.38422392 0.24423077 0.29069767
## [265] 0.62944162 0.55368098 0.48207885 0.33290978 0.35147059 0.35967140
## [271] 0.35096870 0.40711462 0.41639698 0.66099559 0.44347957 0.86524823
## [277] 0.25864023 0.46349677 0.24755700 0.39394513 0.63902439 0.48559382
## [283] 0.70123457 0.38880986 0.26436448 0.41704036 0.60087719 0.34691502
## [289] 0.69521472 0.25739130 0.33605887 0.35211268 0.26527958 0.45669291
## [295] 0.36182336 0.47333333 0.33846154 0.57632399 0.33914940 0.23813489
## [301] 0.44360902 0.42476489 0.51106195 0.60463768 0.41635688 0.50545094
## [307] 0.34213305 0.35446205 0.30188679 0.25371747 0.34858681 0.41414141
## [313] 0.48920863 0.23040153 0.37619048 0.64590747 0.29880843 0.73553719
## [319] 0.38552189 0.45877378 0.29390519 0.38716578 0.41075604 0.20132953
## [325] 0.61666333 0.71128731 0.23525661 0.28287002 0.42343934 0.29334235
## [331] 0.42169540 0.36119403 0.24400000 0.44422311 0.30213904 0.31720430
## [337] 0.56464380 0.29800499 0.32543353 0.28266667 0.57896707 0.50298507
## [343] 0.35625000 0.26174708 0.36630435 0.54746318 0.45124717 0.34710744
## [349] 0.35779817 0.44672897 0.27720207 0.19819820 0.42569270 0.37601863

```

```

## [355] 0.50373832 0.65665236 0.21441441 0.45498783 0.31740614 0.33812950
## [361] 0.36964618 0.48704663 0.33354964 0.61195055 0.43921852 0.42244479
## [367] 0.40937997 0.45104895 0.67975831 0.31219512 0.40997831 0.36161776
## [373] 0.38249400 0.45588235 0.79740260 0.49099786 0.93827160 0.68853695
## [379] 0.32146490 0.40556660 0.25412015 0.58823529 0.62729196 0.40648496
## [385] 0.46188341 0.54944547 0.31727749 0.43027414 0.44840525 0.45205479
## [391] 0.40030441 0.45652174 0.55760369 0.50157729 0.33450704 0.28300259
## [397] 0.34244604 0.21042831 0.42173112 0.68865979 0.25163399 0.62303665
## [403] 0.40672451 0.80403458 0.33333333 0.39720395 0.38768719 0.34580342
## [409] 0.21329640 0.26002227 0.23880597 0.46260926 0.44960362 0.28301887
## [415] 0.34841629 0.70091424 0.48753463 0.35394363 0.29639661 0.40012168
## [421] 0.33204045 0.49585799 0.74902724 0.45129470 0.36576923 0.38523077
## [427] 0.33024691 0.45447501 0.24503072 0.28051554 0.28659476 0.28565329
## [433] 0.38358641 0.26941176 0.71071953 0.45882353 0.55737286 0.35518672
## [439] 0.58214748 0.22080838 0.40586667 0.36206897 0.57634731 0.33922261
## [445] 0.56847545 0.33352668 0.33382425 0.91417166 0.24423963 0.37172359
## [451] 0.30208333 0.22597865 0.42250000 0.34920635 0.62299854 0.27822581
## [457] 0.35655254 0.47493286 0.36298077 0.56464251 0.29079498 0.31338028
## [463] 0.39285714 0.42008487 0.37436136 0.35594606 0.37826541 0.23094340
## [469] 0.27435065 0.22771588 0.37142857 0.22472989 0.21354451 0.30115512
## [475] 0.42115768 0.36528497 0.24413408 0.40209790 0.68877551 0.24215407
## [481] 0.33188532 0.43925234 0.48567435 0.17166730 0.13242009 0.18550186
## [487] 0.26845992 0.38440111 0.33097231 0.58106473 0.51204819 0.34466019
## [493] 0.55500000 0.32894737 0.31818182 0.38741887 0.39834025 0.33485114
## [499] 0.53542234 0.45760599 0.67692308 0.33550725 0.44530783 0.37508846
## [505] 0.46722689 0.45595054 0.31250000 0.46478873 0.32139738 0.49534050
## [511] 0.30641026 0.31953940 0.34244792 0.56104651 0.31719128 0.21993671
## [517] 0.40452756 0.32077922 0.28050491 0.24811083 0.34014020 0.37078652
## [523] 0.32505176 0.43675418 0.37723785 0.34403670 0.52750000 0.21664223
## [529] 0.39549437 0.65701043 0.75294118 0.63857143 0.56764706 0.45284738
## [535] 0.33748191 0.58737420 0.60982109 0.54477612 0.54088050 0.48387097
## [541] 0.38596491 0.48258706 0.35812449 0.50000000 0.28560548 0.43884892
## [547] 0.33270677 0.27747253 0.38181818 0.28864060 0.41007194 0.40588235
## [553] 0.45695897 0.41824441 0.35303777 0.47901235 0.39853301 0.30424339
## [559] 0.44965168 0.25434783 0.20036965 0.28494972 0.31992953 0.24738126
## [565] 0.25364759 0.32133450 0.29160740 0.29440182 0.29092209 0.25054801
## [571] 0.18176780 0.23806866 0.28396680 0.30854430 0.30753968 0.36318408
## [577] 0.33636364 0.59562842 0.22333333 0.40018315 0.54395604 0.60766233
## [583] 0.50519751 0.38811302 0.50827423 0.80246533 0.55688623 0.49194847
## [589] 0.26603575 0.34604905 0.33470508 0.40830450 0.25045372 0.26585095
## [595] 0.52910053 0.49189189 0.33058306 0.23863415 0.38978495 0.33442802
## [601] 0.46093750 0.30841121 0.52241379 0.74444444 0.68629344 0.38960252
## [607] 0.22997680 0.64819196 0.38130841 0.30743581 0.43363947 0.28718520
## [613] 0.41836735 0.28471566 0.30924306 0.28985507 0.56771546 0.24551971
## [619] 0.42681427 0.41002716 0.42180155 0.29554455 0.62504802 0.48961552
## [625] 0.47319714 0.30407524 0.66193131 0.23323013 0.59633628 0.54545455
## [631] 0.46612466 0.61679790 0.37972244 0.33048958 0.30739488 0.38737004
## [637] 0.30505013 0.37812983 0.52975048 0.67276888 0.55088323 0.49334910
## [643] 0.63403062 0.45071194 0.41980719 0.81268882 0.63901345 0.58737298
## [649] 0.35333333 0.33102094 0.40245148 0.55655806 0.38955190 0.36284470
## [655] 0.375777801 0.73454869 0.44692737 0.74862989 0.42350886 0.54838710
## [661] 0.51513514 0.56246851 0.37819370 0.47094801 0.39213536 0.28215488
## [667] 0.24345364 0.25390272 0.27814570 0.22608221 0.32394881 0.31260895
## [673] 0.74285714 0.30931339 0.67321429 0.40034394 0.40119760 0.29148035

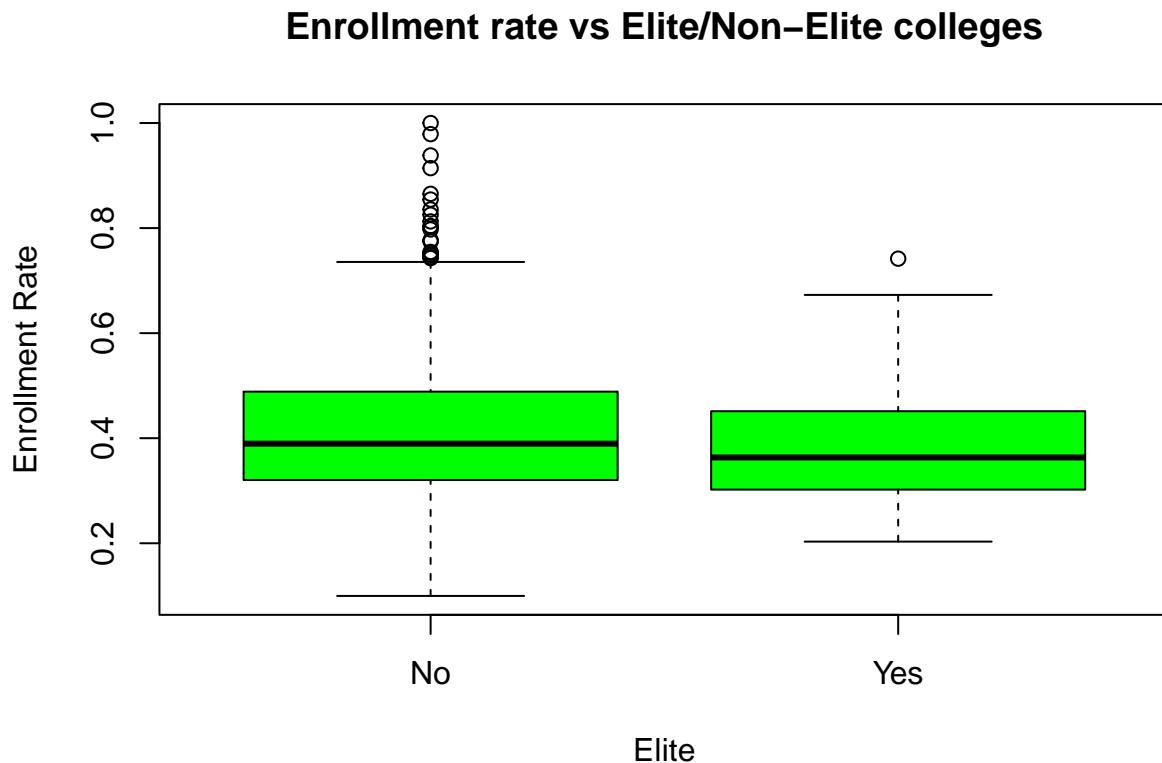
```

```

## [679] 0.48829701 0.60571964 0.51174168 0.45821804 0.66071429 0.56087118
## [685] 0.56584603 0.55672796 0.54387097 0.41193182 0.29143716 0.33747412
## [691] 0.44701349 0.53440214 0.28878162 0.49739970 0.47587189 0.37639553
## [697] 0.52390641 0.37582418 0.39580532 0.39143889 0.42361873 0.51031200
## [703] 0.70778364 0.42477876 0.30019493 0.33076923 0.61627907 0.30110017
## [709] 0.31961620 0.34789558 0.26431639 0.44809837 0.28852459 0.36496288
## [715] 0.28136646 0.31888889 0.52316602 0.18687873 0.41091493 0.41083744
## [721] 0.37750836 0.53595506 0.36012862 0.32122905 0.30363636 0.38777372
## [727] 0.22717622 0.41787977 0.23844837 0.52731245 0.33128834 0.55244755
## [733] 0.48917749 0.46357086 0.54166667 0.35958632 0.43370508 0.36087177
## [739] 0.38768014 0.45009416 0.34500000 0.41961096 0.28252033 0.38075372
## [745] 0.27804215 0.37215034 0.43474600 0.38372093 0.33273056 0.43533487
## [751] 0.29583333 0.49228612 0.59565217 0.30371449 0.28211921 0.35971944
## [757] 0.32179607 0.40172786 0.33646113 0.30328442 0.29766390 0.57586837
## [763] 0.52183908 0.42248996 0.35384615 0.37659784 0.63556424 0.42603878
## [769] 0.58593750 0.33064980 0.29197861 0.29472774 0.35841584 0.38504155
## [775] 0.36292428 0.53689360 0.37250674

```

```
boxplot(EnrollRate ~ Elite, data = college, xlab = "Elite", ylab = "Enrollment Rate", main = "Enrollment Rate vs Elite/Non-Elite colleges")
```



```
# This hypothesis is incorrect as non-elite have much higher enrollment rate with a median of 0.4 while
#Non-elite universities has a lot of outliers compared to Elite college. There's one Elite college with
```

Hypothesis 3. The number of applications per enrolled student is higher at elite colleges than other colleges.
Methodology: Use one-over-other horizontal blue boxplots of applications per enrolled student for elite

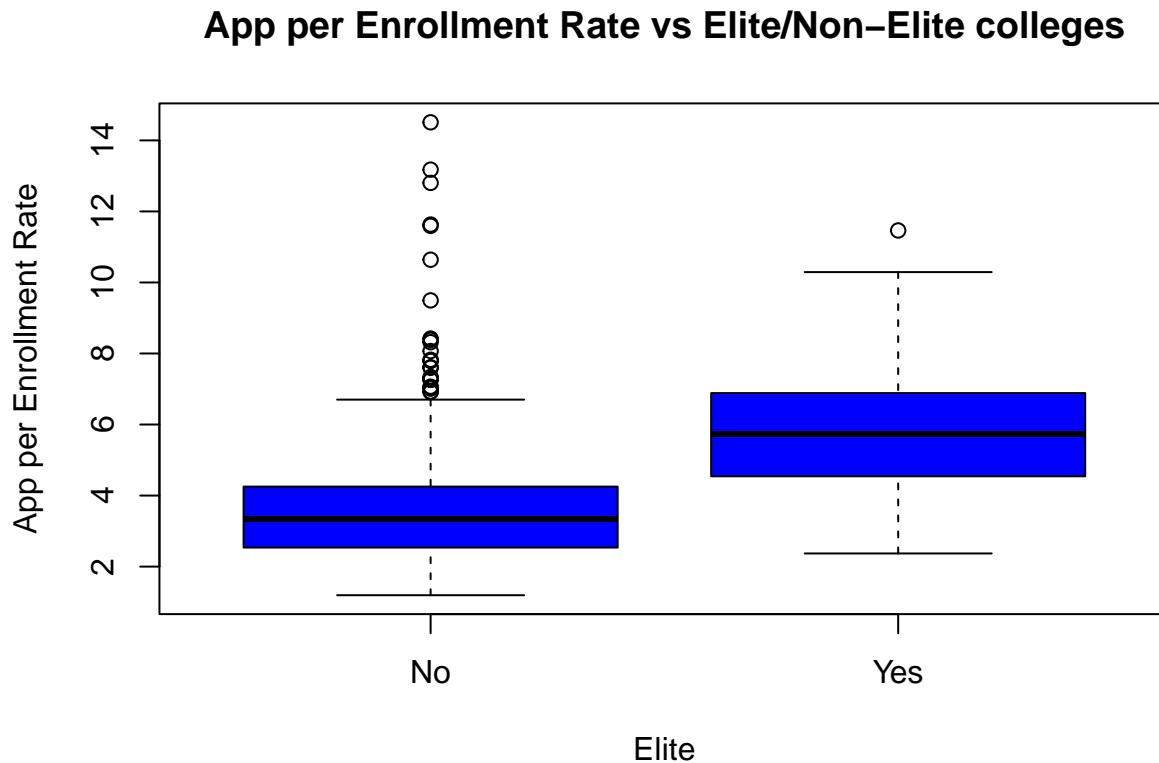
and non-elite colleges. Title and label the axes of this plot thoughtfully. Carefully use this single plot to address this hypothesis. Can you be precise about the word *higher* ? Can you comment on the presence of outliers in the boxplots?

```
color = "blue"
attach(college)

## The following object is masked _by_ .GlobalEnv:
##
##      Elite

## The following objects are masked from college (pos = 3):
##
##      Accept, Apps, Books, Elite, Enroll, Expend, F.Undergrad, Grad.Rate,
##      Outstate, P.Undergrad, perc.alumni, Personal, PhD, Private,
##      Room.Board, S.F.Ratio, Terminal, Top10perc, Top25perc

college$AppPerEnrollment = college$Apps/college$Enroll
boxplot(AppPerEnrollment ~ Elite, data = college, xlab = "Elite", ylab = "App per Enrollment Rate", main = "App per Enrollment Rate vs Elite/Non-Elite colleges", col = "blue")
```

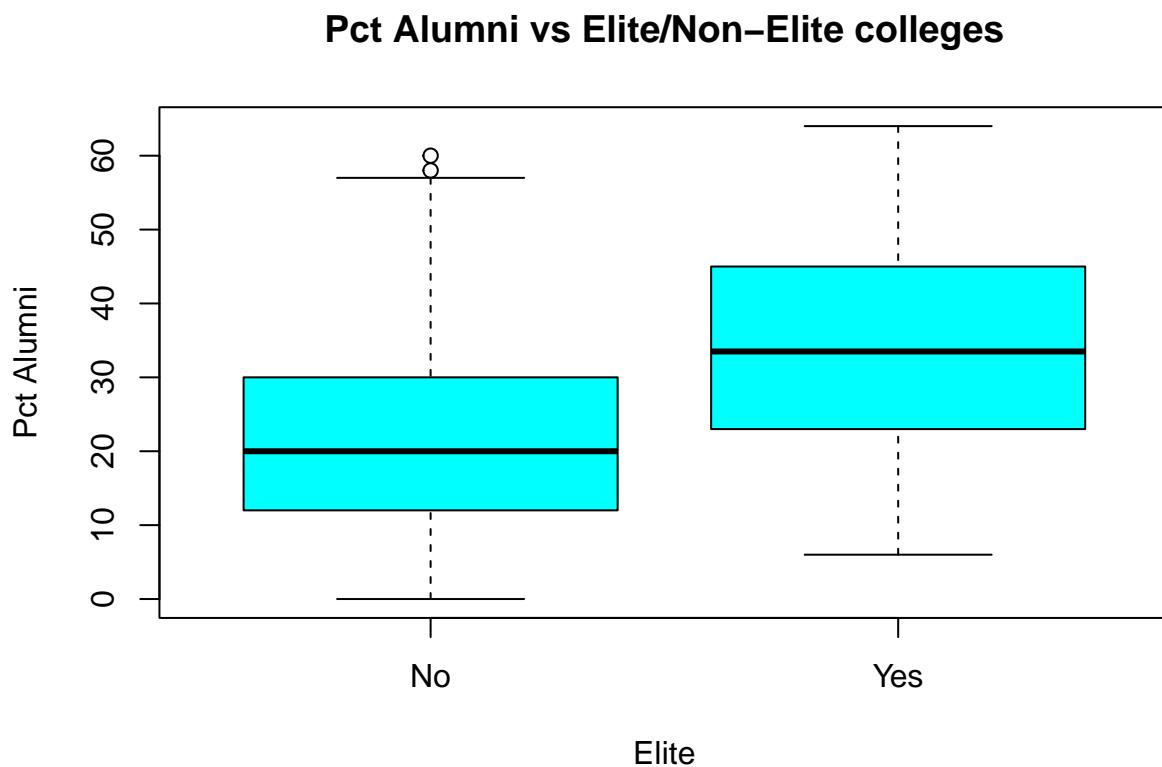


This hypothesis is correct. The app per enrollment rate is higher for elite universities with a median of ~5.5 compared to ~3.2 for non-elite universities.

Hypothesis 4. The fraction of alumni who donate is higher at elite colleges than other colleges. **Methodology:** Use one-over-other horizontal cyan boxplots of perc.alumni for elite and non-elite colleges. Title and

label the axes of this plot thoughtfully. Carefully use this single plot to address this hypothesis. Can you be precise about the word *higher* ?

```
#  
color = "cyan"  
attach(college)  
  
## The following object is masked _by_ .GlobalEnv:  
##  
##      Elite  
  
## The following objects are masked from college (pos = 3):  
##  
##      Accept, Apps, Books, Elite, Enroll, Expend, F.Undergrad, Grad.Rate,  
##      Outstate, P.Undergrad, perc.alumni, Personal, PhD, Private,  
##      Room.Board, S.F.Ratio, Terminal, Top10perc, Top25perc  
  
## The following objects are masked from college (pos = 4):  
##  
##      Accept, Apps, Books, Elite, Enroll, Expend, F.Undergrad, Grad.Rate,  
##      Outstate, P.Undergrad, perc.alumni, Personal, PhD, Private,  
##      Room.Board, S.F.Ratio, Terminal, Top10perc, Top25perc  
  
boxplot(perc.alumni ~ Elite, data = college, xlab = "Elite", ylab = "Pct Alumni", main = "Pct Alumni vs Elite")
```



This hypothesis holds true. We can see that the median of pct alumni donation in elite is 35% where a