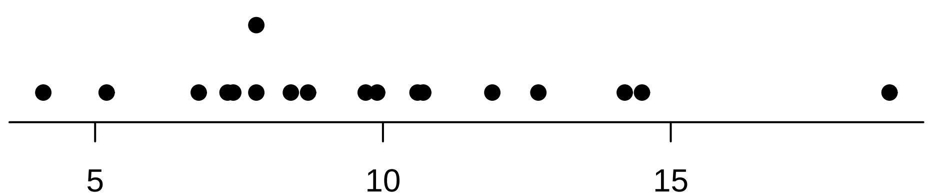
1. Problems 2.4 and 2.5 (page 30). Related R function: `hist`.

Solution: The data for these problems are measurements of the activity of monoamine oxydase (MAO) measured in nmoles benzylaldehyde product per 10^8 pletelets from 18 schizophrenia patients.

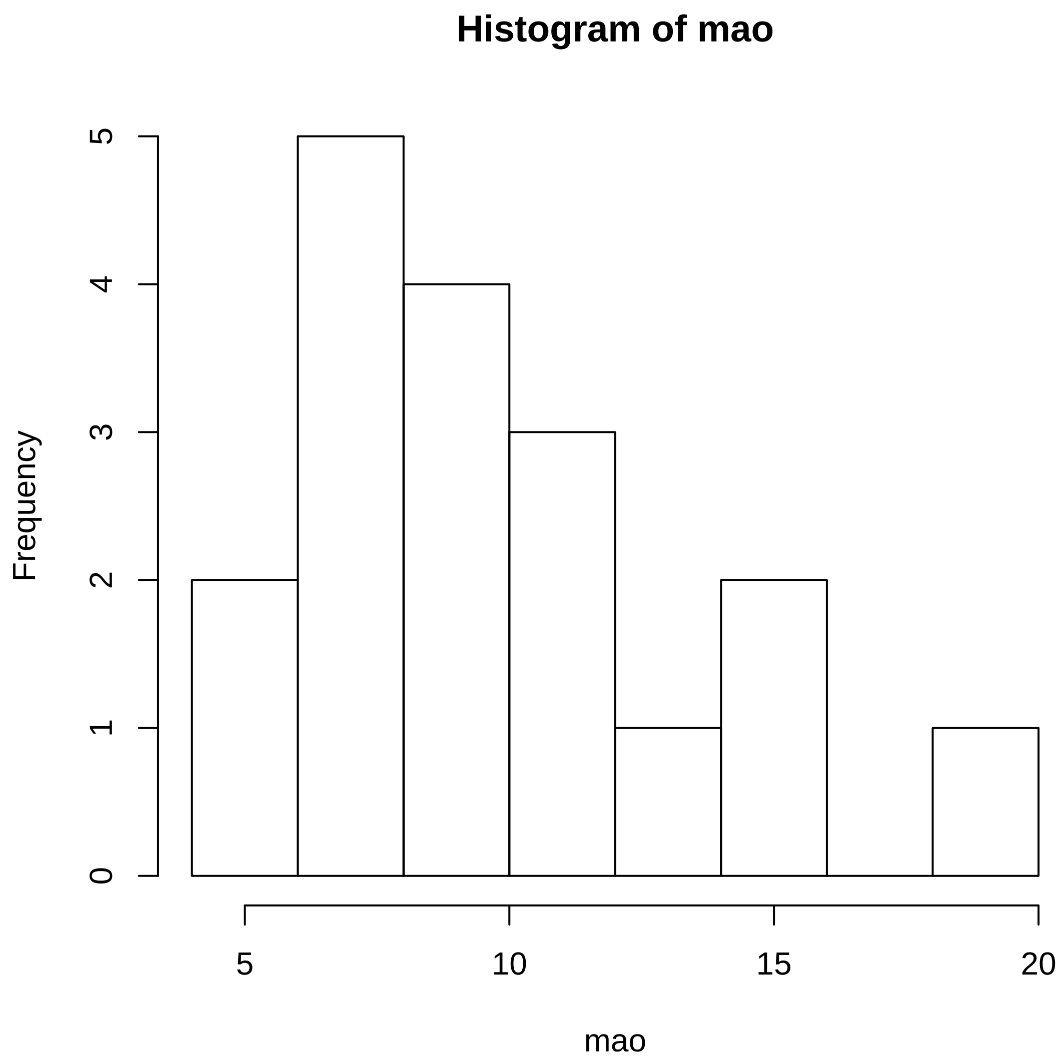
Here is a dotplot of the data.

```
R> source("dotplot.r")
R> mao <- c(6.8, 8.4, 8.7, 11.9, 14.2, 18.8, 9.9, 4.1, 9.7, 12.7,
+      5.2, 7.8, 7.8, 7.4, 7.3, 10.6, 14.5, 10.7)
R> dotplot(mao)
```



Here is a histogram.

```
R> hist(mao)
```



The histogram shows a bimodal distribution with peaks near 7.5 and 14.5, but the sample size is so small that the second peak may be an artifact. The distribution is skewed slightly to the right. The dotplot and the histogram show that the largest observation (18.8) is a bit of an outlier.

Notes on grading: R does not have a dotplot function. R does have have a function called `dotchart`, but this should not be used as it produces a graph that is different than what we call a dotplot in the textbook. I have written a new dotplot function which produced the output in this solution.

Here are some things for which you should be checking, especially with hand drawn graphs.

(a) The graph should not be made with `dotchart`.
(b) The scale on the numeric axis should consistent without breaks and with the same scale throughout.
(c) The bars on histograms should cover the entire range of the class without gaps (except where there is missing data).
(d) The class intervals should all be the same size.
(e) Students may have chosen different classes for the histogram, and that is fine.

2. Problem 2.6 (page 30). Related R function: `stem`.

Solution: The data is the number of dendritic branch-segments emanating from nerve cells of newborn guinea pigs.

```
R> branches <- c(23, 30, 54, 28, 31, 29, 34, 35, 30, 27, 21, 43,
+      51, 35, 51, 49, 35, 24, 26, 29, 21, 29, 37, 27, 28, 33, 33,
+      23, 37, 27, 40, 48, 41, 20, 30, 57)
R> stem(branches)

  The decimal point is 1 digit(s) to the right of the |

  2 | 011334
  2 | 677788999
  3 | 0001334
  3 | 55577
  4 | 013
  4 | 89
  5 | 114
  5 | 7
```

Notes on grading: Students may do this by hand or with R. Here are some common errors you might see.

(a) Using more than one digit per leaf.
(b) Not including a key.
(c) Not lining up digits in columns.
(d) Not including empty rows.

3. Problem 2.13 (page 36).

   Solution: There are many possible solutions. The third number after sorting must be 15 and the numbers must sum to 100.

4. Problems 2.16 and 2.17 (page 37). Related R functions: `mean` and `median`.

   Solution: The data for these problems are the average daily weight gain (pounds per day) of none steers over a 140-day test period.

```
R> gain <- c(3.89, 3.51, 3.97, 3.31, 3.21, 3.36, 3.67, 3.24, 3.27)
R> mean(gain)

[1] 3.492222

R> median(gain)

[1] 3.36

R> gain <- c(gain, 2.46)
R> mean(gain)

[1] 3.389

R> median(gain)

[1] 3.335
```

5. Problem 2.22 (page 38).

   Solution: Each bar has width 3. At a quick glance, it looks like the median is about 40. It is skewed left, so the mean would be a bit smaller, say about 36.

   Here is a more accurate solution. The cell frequencies are: 1,2,2,3,4,6,8,8,9,12,11,15,8,9,1 which sum to 99. The fiftieth observation is in the bar of height 12, so the median is between 37 and 40. The mean is slightly smaller; 36 is a good guess.

6. Problem 2.24 (page 47). Related R functions: `quantile` and `boxplot`.

   Solution: The data are the same MAO data from Exercise 2.4. Here are the medians and quartiles (computed with two different algorithms) displayed with a modified boxplot. Notice the outlier at 18.8.

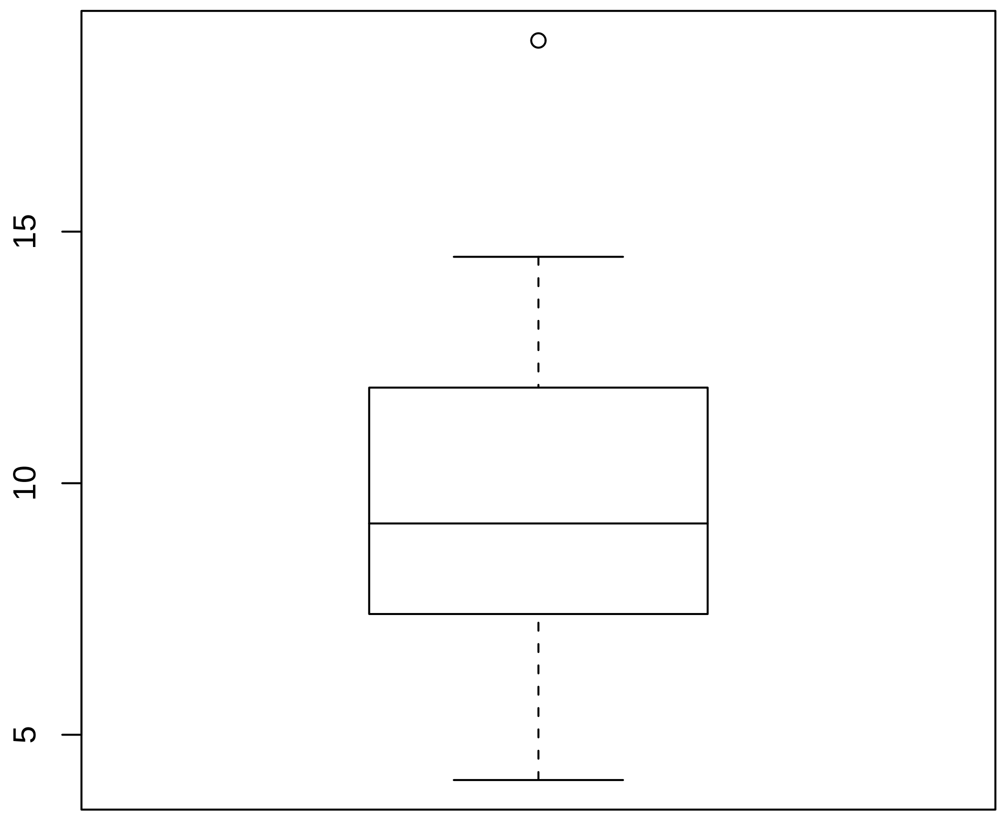
```
R> fivenum(mao)

[1]  4.1  7.4  9.2 11.9 18.8

R> quantile(mao)

  0%  25%  50%  75% 100%
 4.1  7.5  9.2 11.6 18.8

R> boxplot(mao)
```



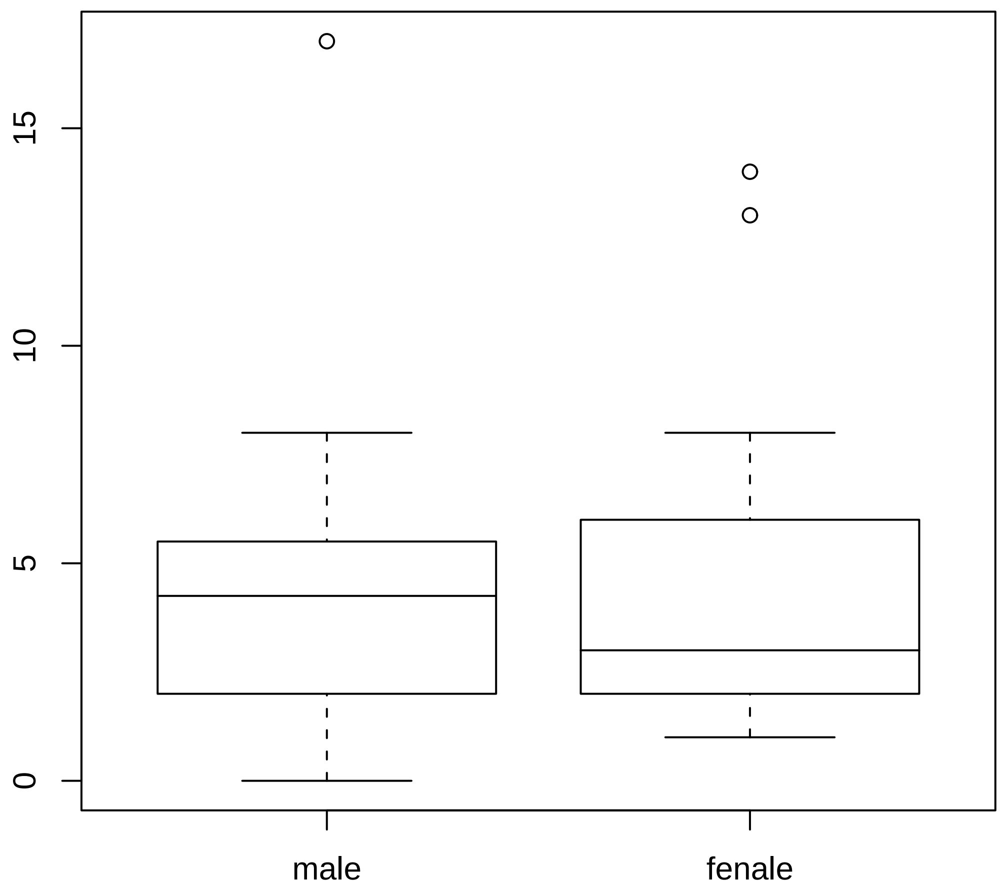
Notes on grading: Make sure that students sort the data before finding quartiles. There are different ways to define quartiles. The function `fivenum` produces a five number summary that is more consistent with how I defined quartiles in class, but I just learned of it today. The `quantile` function (which I suggested students use) actually does something different.

Students probably did not use a modified boxplot (with an outlier marked) if they constructed the boxplot by hand. This is okay, I did not lecture on that yet. As long as students are clear about the methods they use to find quartiles, different answers are acceptable.

7. Problem 2.26 (page 47). Related R functions: `boxplot`, `data.frame`, `split`, and `list`. The data is self-reported numbers of hours of exercise per week given by 25 college students, 12 men and 13 women. Here are parallel boxplots.

   Solution:

```
R> male <- c(6, 0, 2, 1, 2, 4.5, 8, 3, 17, 4.5, 4, 5)
R> female <- c(5, 13, 3, 2, 6, 14, 3, 1, 1.5, 1.5, 3, 8, 4)
R> boxplot(list(male = male, fenale = female))
```



The functions `read.table` and `split` would be useful to create parallel boxplots if the data were read in from a file. The function `data.frame` could also be used to create a "data frame" or matrix of the two variables.

For example, you could create a text file `ex2-26.txt` with the values of the two variables like this.

```
hours sex
6     male
```

```
0     male
2     male

.
.
.

5     male
5     female
13    female
3     female

.
.
.

4     female
```

Here is how to read it and create a data frame x.

```
R> x <- read.table("ex2-26.txt", header = TRUE)
```

Alternatively, you could create the data frame using the variables entered before. The function `rep` repeats a value a specified number of times.

```
R> hours <- c(male, female)
R> sex <- c(rep("male", 12), rep("female", 13))
R> x <- data.frame(hours = hours, sex = sex)
```

Finally, make the parallel boxplots. The `attach` function adds the names of the variables to the search path.

```
R> attach(x)
R> boxplot(split(hours, sex))
```