Chapter 16

Data Frames

A distinctive characteristic of vectors, matrices, and arrays is that all of their elements are of the same data type, that is, these data structures are atomic. Many statistical experiments generate data sets with data values of differing types, for example, `numeric` for a patient's cholesterol level, `character` for a patient's gender, and `logical` to record the presence or absence of a particular disease for a patient. A data structure that can store data of differing modes is needed to accommodate data sets of this type. A *data frame*, which is the new data structure introduced in this chapter, carries the best features of a matrix and a list. The format of a data frame is similar to that of a matrix, but the columns of a data frame can have different modes.

A data frame is an ideal data structure for storing and manipulating statistical data. The rows in a data frame tend to be "observational units," such as people, cars, etc. The columns in a data frame tend to be values collected on the observational units such as eye color, miles per gallon, etc. An entry in a data frame is known as a "cell."

The three topics considered in this chapter are (a) creating a data frame, (b) functions that can be applied to a data frame, and (c) extracting elements of a data frame.

16.1 Creating a data frame

Data frames can be created with the `data.frame` function. The syntax for the `data.frame` function is

```
data.frame(name1 = col1, name2 = col2, ..., row.names = NULL, ...)
```

where the column names `name1`, `name2`, etc. are optional parts of the arguments. The `data.frame` function is illustrated below for creating a data frame that consists of five rows and four columns. The first column is a set of five capital letters; the second column contains the numbers 5, 6, 7, 8, 9; the third column contains the complex numbers $2+6i$, $3+5i$, $4+4i$, $5+3i$, $6+2i$; the fourth column contains the logical values `TRUE`, `FALSE`, `TRUE`, `FALSE`, `TRUE`.

```
> char = LETTERS[18:22]                      # vector of 5 characters
> numb = 5:9                                 # vector of 5 numeric values
> comp = complex(5, 2:6, 6:2)                # vector of 5 complex numbers
> bool = c(TRUE, FALSE, TRUE, FALSE, TRUE)   # vector of 5 logical values
> d = data.frame(char, numb, comp, bool)     # d is a data frame
> d                                          # display d
```

```
  char numb comp  bool
1    R    5 2+6i  TRUE
2    S    6 3+5i FALSE
3    T    7 4+4i  TRUE
4    U    8 5+3i FALSE
5    V    9 6+2i  TRUE
```

The data frame d that is displayed has column names, which are inherited as the names of the objects that formed the data frame. These column headings are known as the “header.” Since the row.names argument in the call to data.frame was defaulted, there are no row names, and R simply uses consecutive integers to identify the rows when displaying d.

16.2 Functions that operate on data frames

R has a number of functions that can be applied to data frames. Some of these functions can also be applied to matrices. The nrow and ncol functions return the number of rows and the number of columns of a data frame, respectively.

```
> nrow(d)                              # number of rows
[1] 5
> ncol(d)                              # number of columns
[1] 4
```

The data frame d has five rows and four columns. The head function displays the first few rows of a data frame—which can be helpful in getting a compact view of the structure of a large data frame.

```
> head(d)                              # display header
  char numb comp  bool
1    R    5 2+6i  TRUE
2    S    6 3+5i FALSE
3    T    7 4+4i  TRUE
4    U    8 5+3i FALSE
5    V    9 6+2i  TRUE
```

Since this data frame is so small, the entire data frame gets displayed. Calling head(d, 3), for example, displays the first three rows of d. The head function can also be applied to vectors, matrices, and arrays. There is also a tail function that displays the last few rows of a data frame. The str function displays the structure of a data frame (or any other arbitrary R object).

```
> str(d)                               # display structure of d
'data.frame':   5 obs. of  4 variables:
 $ char: Factor w/ 5 levels "R","S","T","U",..: 1 2 3 4 5
 $ numb: int  5 6 7 8 9
 $ comp: cplx  2+6i 3+5i 4+4i ...
 $ bool: logi  TRUE FALSE TRUE FALSE TRUE
```

The response shows that d is a data frame consisting of five observations of four variables, which are named char, numb, comp, and bool. The first few values of the elements in each of the columns are displayed. The summary function gives a summary of the contents of a data frame (or any other arbitrary object).