chapter 2

Creating a dataset in R

Instructor: Li, Han

Contents

- objects: vector, matrix, data frame, factor, list...
- input data from external files: text, Excel...

Object

- name
- types of objects: vector, matrix, array, data frame, factor, list
- creation
  - assign a value
  - create a blank object, for example, list()

Different classes of objects

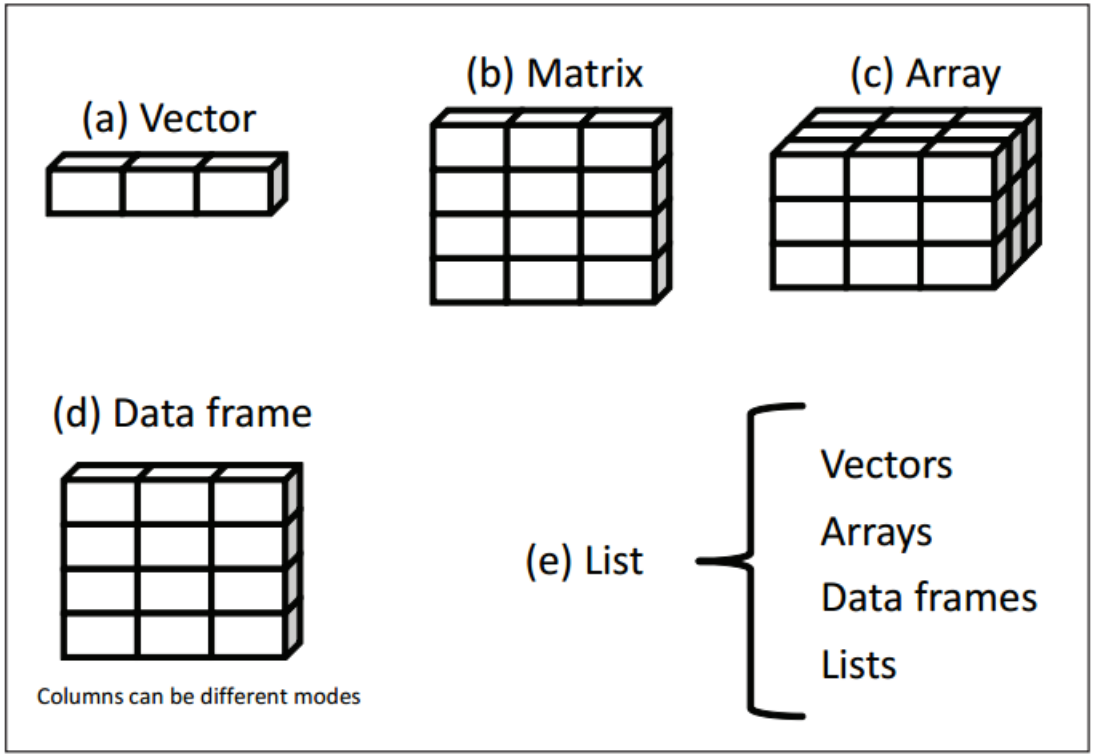


Figure 2.1 **R data structures**

Assignment

- “<-” used to indicate assignment
  - x <- 2
  - x <- c(1,2,3,4,5,6,7)
  - x <- 1:7
  - x <- seq(from=1,to=10,by=2)
  - x <- seq(1,10,2)
- Note that we could also use "=" to assign the value.

Naming Convention

- must start with a letter (A-Z or a-z)
- can contain letters, digits (0-9), and/or periods “.”
- case-sensitive
  - mydata is different from MyData
- do not use underscore “_”

Vector

- a one-dimensional array that can hold numerical, character or logical data
- x <- c(...)

- a <- c(1, 2, 5, 3, 6, -2, 4)
- b <- c("one", "two", "three")
- d <- c(TRUE, TRUE, TRUE, FALSE, TRUE, FALSE)

- How to refer to the element of a vector
  - x[i]: i-th element
  - x[vec]: the elements whose positions are specified in vec

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> a <- c(1, 2, 5, 3, 6, -2, 4)
> b <- c("one", "two", "three")
> d <- c(TRUE, TRUE, TRUE, FALSE, TRUE, FALSE)
> a[1]
[1] 1
> a[c(1,3,5)]
[1] 1 5 6
> a[-1]
[1]  2  5  3  6 -2  4
> a[-c(1,2)]
[1]  5  3  6 -2  4
> b[2]
[1] "two"
> d[1:5]
[1]  TRUE  TRUE  TRUE FALSE  TRUE
>
```

Matrix

- a two-dimensional array that each element has the same mode (numeric, character, or logical)

```
mymatrix <- matrix(vector, nrow=?, ncol=?,
 byrow=TRUE/FALSE, dimnames=?)
```

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> y <- matrix(1:20,4,5)
> y
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    5    9   13   17
[2,]    2    6   10   14   18
[3,]    3    7   11   15   19
[4,]    4    8   12   16   20
> y <- matrix(1:20,4,5, byrow=TRUE)
> y
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    2    3    4    5
[2,]    6    7    8    9   10
[3,]   11   12   13   14   15
[4,]   16   17   18   19   20
>
```

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> cells <- c(1,26,24,68)
> rnames <- c("R1", "R2")
> cnames <- c("C1", "C2")
> y <- matrix(cells, nrow=2, ncol=2, byrow=FALSE,
+       dimnames=list(rnames, cnames))
> y
   C1 C2
R1  1 24
R2 26 68
>
```

How to refer to the elements of a matrix

- x[i,j]: i-th, j-th element
- x[i,]: i-th row
- x[,j]: j-th column
- x[vec1, vec2]: a submatrix

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> y
   C1 C2
R1  1 24
R2 26 68
> y[1,]
C1 C2
 1 24
> y[2,]
C1 C2
26 68
> y[,2]
R1 R2
24 68
>
```

Array

- Array is similar to matrix, but has more than two dimensions

- myarray <- array(vector, dimensions=?, dimnames=?)

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> dim1 <- c("A1", "A2")
> dim2 <- c("B1", "B2", "B3")
> dim3 <- c("C1", "C2", "C3", "C4")
> z <- array(1:24, c(2,3,4), dimnames=list(dim1, d$
> z
, , C1

   B1 B2 B3
A1  1  3  5
A2  2  4  6

, , C2

   B1 B2 B3
A1  7  9 11
```

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
A2  8 10 12

, , C3

   B1 B2 B3
A1 13 15 17
A2 14 16 18

, , C4

   B1 B2 B3
A1 19 21 23
A2 20 22 24

>
```

Data frame

Data frame is more general than a matrix because its columns can have different kinds of data.

- mydata <- data.frame(col1, col2,...)

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> patientID <- c(1, 2, 3, 4)
> age <- c(25, 34, 28, 52)
> diabetes <- c("Type1", "Type2", "Type1", "Type1")
> status <- c("Poor", "Improved", "Excellent", "Poor")
> patientdata <- data.frame(patientID, age, diabetes,
+               status)
> patientdata
  patientID age diabetes    status
1         1  25    Type1      Poor
2         2  34    Type2  Improved
3         3  28    Type1 Excellent
4         4  52    Type1      Poor
>
>
```

How to refer to elements of a data frame?

- x[vec]: columns specified in vec
- x[variable.names]
- x$variable.name

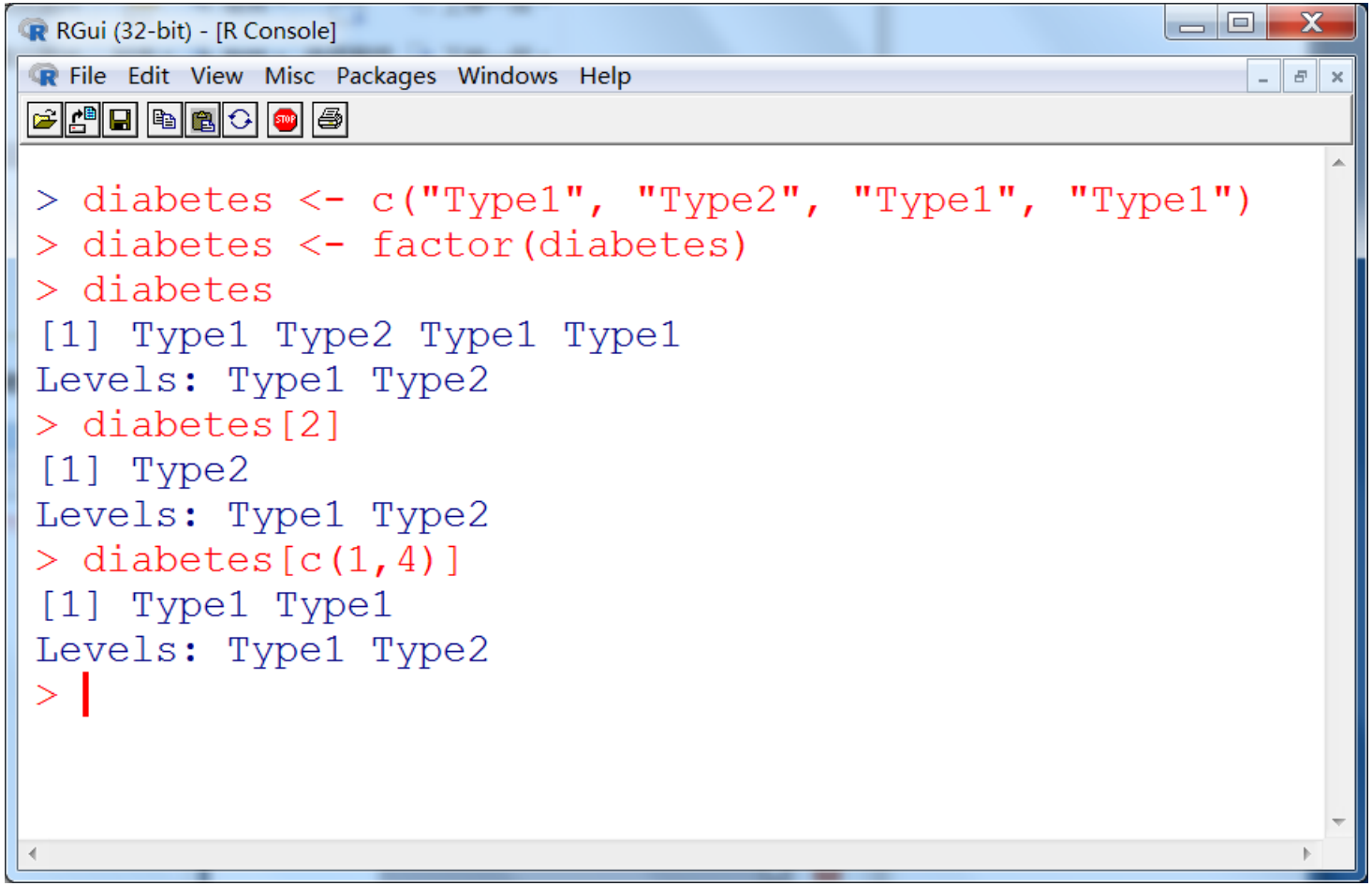
RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> patientdata[1:2]
  patientID age
1         1  25
2         2  34
3         3  28
4         4  52
> patientdata[c("diabetes", "status")]
  diabetes    status
1    Type1      Poor
2    Type2  Improved
3    Type1 Excellent
4    Type1      Poor
> patientdata$age
[1] 25 34 28 52
>
```

Factor

Factor is an ordinal or unordinal nominal variable.



RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> status <- c("Poor", "Improved", "Excellent",
+            "Poor")
> status <- factor(status, ordered=TRUE)
> status
[1] Poor      Improved  Excellent Poor
Levels: Excellent < Improved < Poor
> status <- factor(status, ordered=TRUE,
+           levels=c("Poor", "Improved", "Excellent"))
> status
[1] Poor      Improved  Excellent Poor
Levels: Poor < Improved < Excellent
>
>
```

List

List is an ordered object that contains different kinds of sub-objects, for example, vectors, matrices, data frames and even lists.

- mylist <- list(object1, object2,...)

- How to refer to the elements of a list
  - x[[i]]:i-th compoent
  - x$variable.name

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> > g <- "my first list"
> h <- c(1,2,3)
> j <- matrix(1:6, nrow=2,ncol=3)
> k <- c("one", "two", "three")
> mylist <- list(title=g, ages=h, j, k)
> mylist
$title
[1] "my first list"

$ages
[1] 1 2 3

[[3]]
     [,1] [,2] [,3]
[1,]    1    3    5
```

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
$title
[1] "my first list"

$ages
[1] 1 2 3

[[3]]
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6

[[4]]
[1] "one"   "two"   "three"

>
```

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> mylist[[2]]
[1] 1 2 3
> mylist[["ages"]]
[1] 1 2 3
> mylist$age
[1] 1 2 3
>
```

How does R deal with different objects

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> patientID <- c(1, 2, 3, 4)
> age <- c(25, 34, 28, 52)
> diabetes <- c("Type1", "Type2", "Type1", "Type1")
> status <- c("Poor", "Improved", "Excellent", "Poor")
> patientdata <- data.frame(patientID, age, diabetes,
+                           status)
> str(patientdata)
'data.frame':   4 obs. of  4 variables:
 $ patientID: num  1 2 3 4
 $ age      : num  25 34 28 52
 $ diabetes : Factor w/ 2 levels "Type1","Type2": 1 2 1 1
 $ status   : Factor w/ 3 levels "Excellent","Improved",..: 3 2 1 3
>
```

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> summary(patientdata)
   patientID          age          diabetes       status
 Min.   :1.00   Min.   :25.00   Type1:3   Excellent:1
 1st Qu.:1.75   1st Qu.:27.25   Type2:1   Improved :1
 Median :2.50   Median :31.00             Poor     :2
 Mean   :2.50   Mean   :34.75
 3rd Qu.:3.25   3rd Qu.:38.50
 Max.   :4.00   Max.   :52.00
>
```

Attributes of an object

- attributes
  - mode(object): numeric, character, logical
  - length(object): number of elements in an object
  - dim(object): dimension of a matrix/data frame
  - class(object): type of an object
  - str(object):structure of an object
  - names(object):names of components in an object

Import data

If we already have a data file, how can we directly import it into R instead of manually creating all its objects?

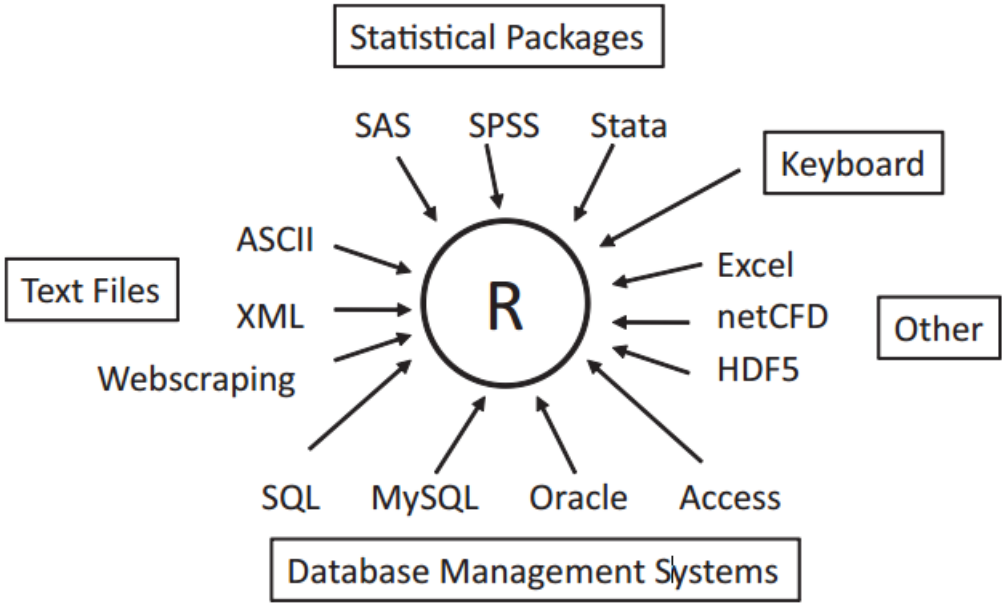


Figure 2.2 **Sources of data that can be imported into R**

Import a data from a text file

```
mydataframe <- read.table(file,
header=TRUE/FALSE, sep=?, row.names=?)
```

studentgrades.txt

```
StudentID, First, Last, Math, Science, Social Studies
011, Bob, Smith, 90, 80, 67
012, Jane, Weary, 75, , 80
010, Dan, "Thorn, III", 65, 75, 70
040, Mary, "O'Leary", 90, 95, 92
```

- grades <- read.table("studentgrades.txt", header=TRUE, sep=",", row.names="StudentID")

RGui (32-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> grades <- read.table("studentgrades.txt", header=TRUE, sep=",", row.names="StudentID")
> class(grades)
[1] "data.frame"
> dim(grades)
[1] 4 5
> grades
   First        Last Math Science Social.Studies
11   Bob       Smith   90      80             67
12  Jane       Weary   75      NA             80
10   Dan  Thorn, III   65      75             70
40  Mary     O'Leary   90      95             92
> str(grades)
'data.frame':   4 obs. of  5 variables:
 $ First         : Factor w/ 4 levels " Bob"," Dan",..: 1 3 2 4
 $ Last          : Factor w/ 4 levels " O'Leary"," Smith",..: 2 4 3 1
 $ Math          : int  90 75 65 90
 $ Science       : int  80 NA 75 95
 $ Social.Studies: int  67 80 70 92
>
```

Import data from an excel file

The “xlsx” package can be used to access spreadsheets in excel format. Be sure to download and install the package before first use. The read.xlsx() function imports a worksheet from an XLSX file into a data frame.

studentgrades2.xlsx

	A	B
1	studentID	age
2	1	17
3	2	18
4	3	17
5	4	17
6	5	17
7	6	18
8	7	18
9	8	17
10	9	17
11	10	17

Import data from an excel file

- install.packages("xlsx")
- library(xlsx)
- workbook <- “D:/studentgrades2.xlsx”
- grades2 <- read.xlsx(workbook, 1)

Practice

- Try all the examples by yourself.
- Check the attributes of different objects.
- Try to input text/excel files.