Introduction to R Programming
Lecture 2

Minghao Wu
*minghaowu*_2015@163.*com*

April 12, 2015

1 Understanding the Dataset

1.1 Vector

Vectors are **one-dimensional** arrays that can hold numeric data, character data, or logical data. The combine function **c()** is used to form the vector.

```
> a = c(1, 2, 5, 3, 6, -2, 4)
> b = c("one", "two", "three")
> c = c(TRUE, TRUE, TRUE, FALSE, TRUE, FALSE)
```

1.2 Matrix

Matrix is a **two-dimensional** array where each element has the same mode (numeric, character, or logical). Matrices are created with the **matrix()** function.

```
> x = matrix(1:20, nrow=5, ncol=4, byrow=TRUE)
> x

     [,1] [,2] [,3] [,4]
[1,]    1    2    3    4
[2,]    5    6    7    8
[3,]    9   10   11   12
[4,]   13   14   15   16
[5,]   17   18   19   20

> y = matrix(1:20, nrow=5, ncol=4, byrow=FALSE)
> x

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

```
[3,]    9   10   11   12
[4,]   13   14   15   16
[5,]   17   18   19   20
```

```
> x[2,]
```

```
[1] 5 6 7 8
```

```
> x[,2]
```

```
[1]  2  6 10 14 18
```

```
> x[1,4]
```

```
[1] 4
```

```
> x[2,c(2,4)]
```

```
[1] 6 8
```

```
> x[3:5, 2]
```

```
[1] 10 14 18
```

```
> rnames=c("apple","banana","orange","melon","corn")
> cnames=c("cat","dog","bird","pig")
> x = matrix(1:20, nrow=5, ncol=4, byrow=TRUE)
> rownames(x)=rnames
> colnames(x)=cnames
> x
```

```
       cat dog bird pig
apple    1   2    3   4
banana   5   6    7   8
orange   9  10   11  12
melon   13  14   15  16
corn    17  18   19  20
```

1.3 Array

Arrays are similar to matrices but can have **more than two dimensions**. They are created with an **array()** function.

```
> dim1 = c("A1", "A2")
> dim2 = c("B1", "B2", "B3")
> dim3 = c("C1", "C2", "C3", "C4")
> dim4 = c("D1", "D2", "D3")
> z = array(1:72, c(2, 3, 4, 3), dimnames=list(dim1, dim2, dim3, dim4))
> z
```

```
, , C1, D1

   B1 B2 B3
A1  1  3  5
A2  2  4  6

, , C2, D1

   B1 B2 B3
A1  7  9 11
A2  8 10 12

, , C3, D1

   B1 B2 B3
A1 13 15 17
A2 14 16 18

, , C4, D1

   B1 B2 B3
A1 19 21 23
A2 20 22 24

, , C1, D2

   B1 B2 B3
A1 25 27 29
A2 26 28 30

, , C2, D2

   B1 B2 B3
A1 31 33 35
A2 32 34 36

, , C3, D2

   B1 B2 B3
A1 37 39 41
A2 38 40 42

, , C4, D2

   B1 B2 B3
A1 43 45 47
```

```
A2 44 46 48

, , C1, D3

   B1 B2 B3
A1 49 51 53
A2 50 52 54

, , C2, D3

   B1 B2 B3
A1 55 57 59
A2 56 58 60

, , C3, D3

   B1 B2 B3
A1 61 63 65
A2 62 64 66

, , C4, D3

   B1 B2 B3
A1 67 69 71
A2 68 70 72

> z[1,2,3,]

D1 D2 D3
15 39 63
```

1.4 Data Frame

A data frame is more general than a matrix in that different columns can contain different modes of data (numeric, character, etc.). It is similar to the datasets you would typically see in SAS, SPSS, and Stata. Data frames are the most common data structure you will deal with in R.

```
> 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
```

```
> swim = read.csv("http://www.macalester.edu/~kaplan/ISM/datasets/swim100m.csv")
> patientdata[1:2]

  patientID age
1         1  25
2         2  34
3         3  28
4         4  52

> patientdata[1:3]

  patientID age diabetes
1         1  25    Type1
2         2  34    Type2
3         3  28    Type1
4         4  52    Type1

> patientdata[1,1:3]

  patientID age diabetes
1         1  25    Type1

> patientdata[c(1,3),1:3]

  patientID age diabetes
1         1  25    Type1
3         3  28    Type1
```

1.5 Attach and Detach

The attach() function adds the data frame to the R search path.
The detach() function removes the data frame from the search path.

1.6 List

Lists are the most complex of the R data types. Basically, a list is an ordered collection of objects (components). A list allows you to gather a variety of (possibly unrelated) objects under one name.

```
> mylist = list(patientdata, swim, x)
> mylist
```

```
[[1]]
  patientID age diabetes    status
1         1  25    Type1      Poor
2         2  34    Type2  Improved
3         3  28    Type1 Excellent
4         4  52    Type1      Poor

[[2]]
   year  time sex
1  1905 65.80   M
2  1908 65.60   M
3  1910 62.80   M
4  1912 61.60   M
5  1918 61.40   M
6  1920 60.40   M
7  1922 58.60   M
8  1924 57.40   M
9  1934 56.80   M
10 1935 56.60   M
11 1936 56.40   M
12 1944 55.90   M
13 1947 55.80   M
14 1948 55.40   M
15 1955 54.80   M
16 1957 54.60   M
17 1961 53.60   M
18 1964 52.90   M
19 1967 52.60   M
20 1968 52.20   M
21 1970 51.90   M
22 1972 51.22   M
23 1975 50.59   M
24 1976 49.44   M
25 1981 49.36   M
26 1985 49.24   M
27 1986 48.74   M
28 1988 48.42   M
29 1994 48.21   M
30 2000 48.18   M
31 2000 47.84   M
32 1908 95.00   F
33 1910 86.60   F
34 1911 84.60   F
35 1912 78.80   F
36 1915 76.20   F
37 1920 73.60   F
```

```
38 1923 72.80   F
39 1924 72.20   F
40 1926 70.00   F
41 1929 69.40   F
42 1930 68.00   F
43 1931 66.60   F
44 1933 66.00   F
45 1934 65.40   F
46 1936 64.60   F
47 1956 62.00   F
48 1958 61.20   F
49 1960 60.20   F
50 1962 59.50   F
51 1964 58.90   F
52 1972 58.50   F
53 1973 57.54   F
54 1974 56.96   F
55 1976 55.65   F
56 1978 55.41   F
57 1980 54.79   F
58 1986 54.73   F
59 1992 54.48   F
60 1994 54.01   F
61 2000 53.77   F
62 2004 53.52   F

[[3]]
       cat dog bird pig
apple    1   2    3   4
banana   5   6    7   8
orange   9  10   11  12
melon   13  14   15  16
corn    17  18   19  20
```

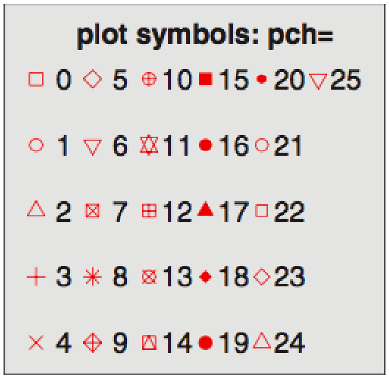
2 Graphs

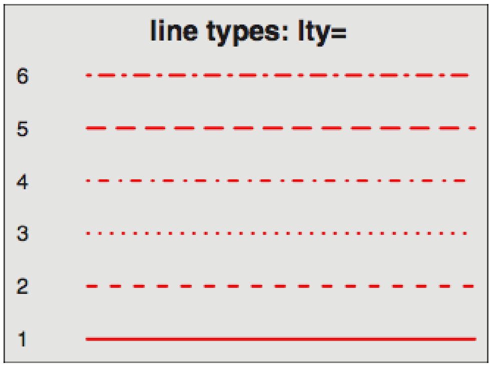
2.1 Graphical parameters

You can customize many features of a graph (fonts, colors, axes, titles) through options called graphical parameters. They are specified with an **par()** function.

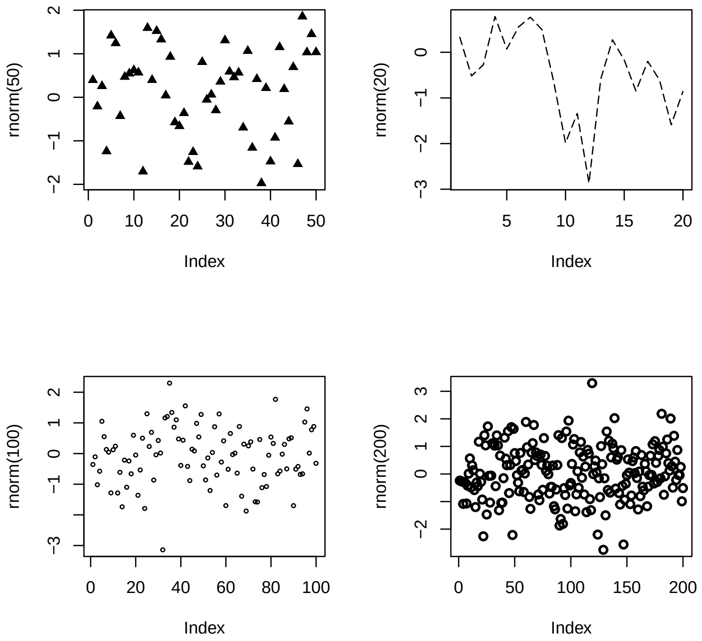
```
> par(mfrow=c(2,2))
> plot(rnorm(50),pch=17)
> plot(rnorm(20),type="l",lty=5)
> plot(rnorm(100),cex=0.5)
```

Parameter	Description
`pch`	Specifies the symbol to use when plotting points (see figure 3.4).
`cex`	Specifies the symbol size. `cex` is a number indicating the amount by which plotting symbols should be scaled relative to the default. 1=default, 1.5 is 50% larger, 0.5 is 50% smaller, and so forth.
`lty`	Specifies the line type (see figure 3.5).
`lwd`	Specifies the line width. `lwd` is expressed relative to the default (default=1). For example, `lwd=2` generates a line twice as wide as the default.





```
> plot(rnorm(200),lwd=2)
>
```



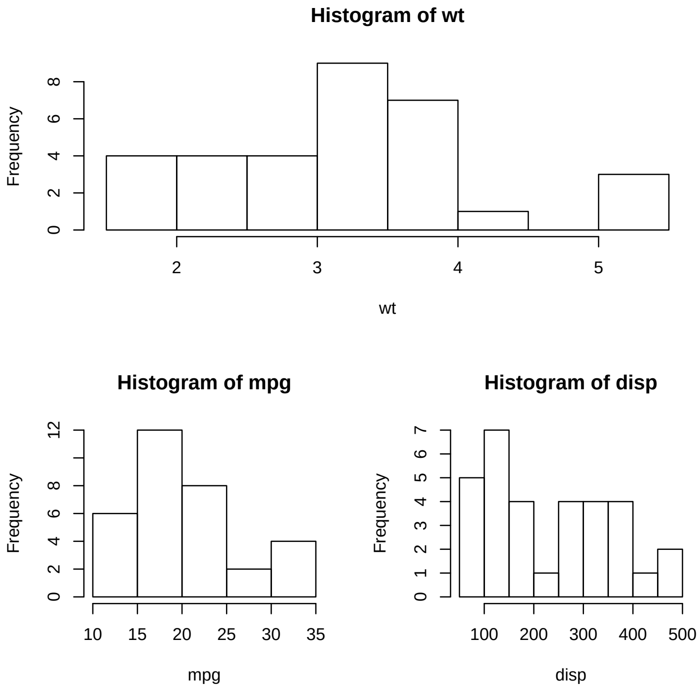
2.2 Text, Axes, and Legends

title()
axis()
legend()

2.3 Layout

The layout() function has the form layout(mat) where mat is a matrix object speci- fying the location of the multiple plots to combine.

```
> attach(mtcars)
> layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
> hist(wt)
> hist(mpg)
> hist(disp)
> detach(mtcars)
```



3 Next Topic

Operators, Control Flow & User-defined Function