

1. Introduction

Advantages of R

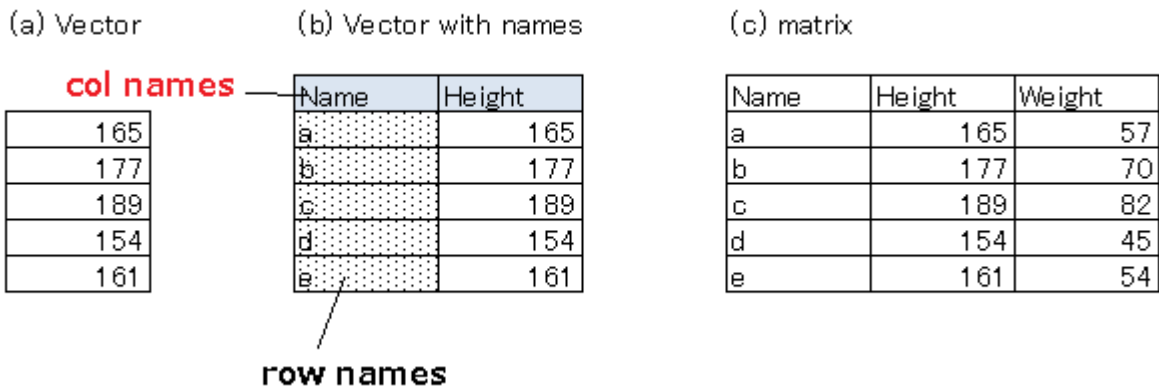
Free! Various packages! Easy to carry out some complex statistics! Easy to make beautiful graphs! Any OS!

Disadvantages

No GUI (difficult to remember)

2. How to load your data into R

(A) The format of the data



Vector=a variable that contains more than one data. Matrix=a data set that consist of more than one columns.

(B) Making your own data

Data can be made with Excel or a text editor.

In case of matrix (as in (b) and (c)), the data should be separated by Tab or Comma.

Name *TAB* Height

a *TAB* 165

b *TAB* 177

....

By default, Excel separates data by TAB. So you can simply copy and paste the Excel data to a text editor, save it as a text file (e.g. height.txt).

TRY Make your own data with a text editor

OR you can directly type in the console as:

Vector

```
> x=c(1,2,3,4)
```

```
> x
```

```
[1] 1 2 3 4
```

```
> y=(1:10)
```

```
> y
```

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

Matrix

```
> z=matrix(c(1:10),2,5) #Format: matrix(DATA, Row Number, Column Number)
```

```
> z
```

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

```
[1,] 1 3 5 7 9
```

```
[2,] 2 4 6 8 10
```

TRY Make a matrix that consist of 50 rows \times 2 columns. You can use c(1:100).

(2) To read a text file

```
> height=read.table("height.txt", header=TRUE) #Format: read.table(FILE NAME, header=TRUE)
```

```
# If the data has Col names (see above), "header" should be TRUE
```

```
# Note that TRUE should be all capital
```

```
> height
```

```
 Name Height
```

```
1 a 165
```

```
2 b 177
```

```
3 c 189
```

```
4 d 154
```

```
5 e 161
```

✂If the file is in the workspace, the file can be loaded only by its name, but if it is in other place, the full address (or folder name) should be added.

☆The alternative way to load a file

```
> height=read.table(file.choose(), header=TRUE)
```

→This will pop up a file selection window.

(3) To paste the data into R

```
X = read.delim("clipboard", header= T )
```

TRY Try to set up data from the clipboard.

3. Analyzing a vector

There are some functions for analyzing vectors:

Sum

Length

Sd

Mean

Max

Min

Range

Summary

TRY Make the following vectors, and calculate each mean *without* mean function.

```
> x
```

```
[1] 10 9 1 2 9
```

```
> y
```

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

What did you find?

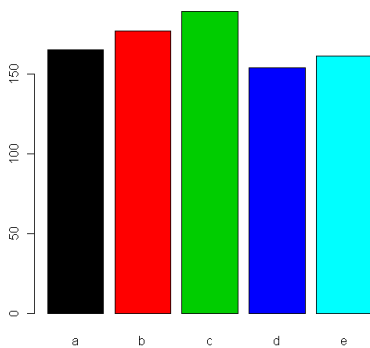
4. Graphics

4.1. Graphics with a vector

<Bar charts>

```
> barplot(height$Height, names.arg=height$Name, col=1:5)
```

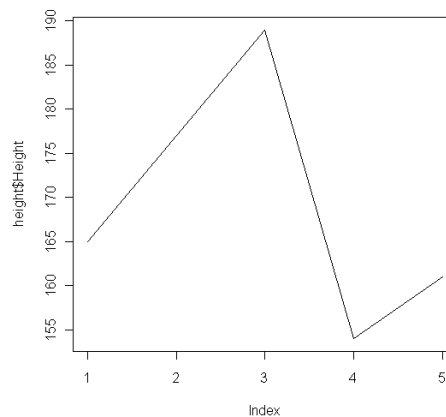
#names.arg will add the names below the bars



TRY Make a similar graph as above.

<Line charts>

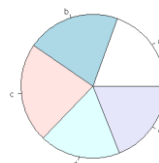
```
> plot(height$Height,type="l")
```



TRY Try type="b".

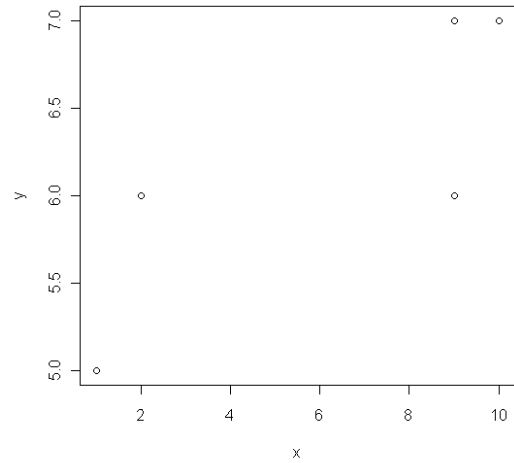
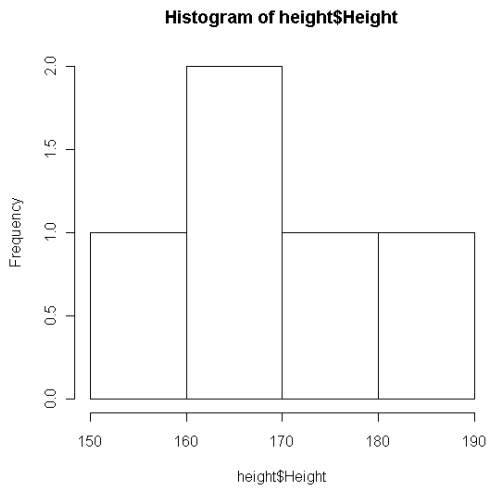
<Pie charts>

```
> pie(height$Height, labels=height$Name)
```



<Histogram>

```
> hist(height$Height)
```



4.2. Adding things onto charts

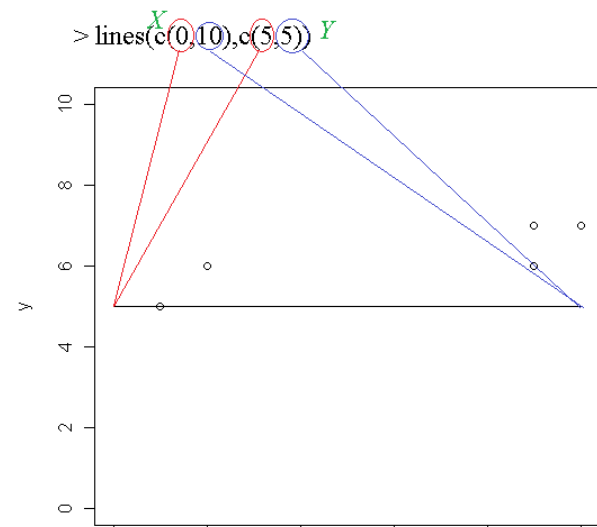
There are some functions that add things like legend and lines.

☆ Adding lines

Lines(c(a,b), c(c,d))

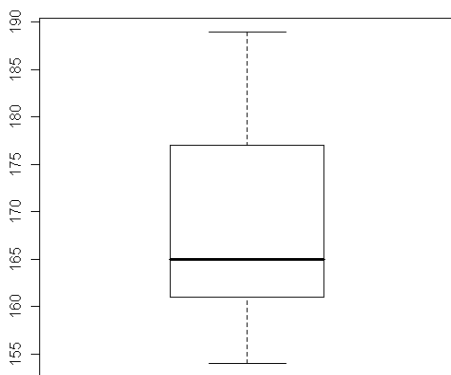
```
> plot(x,y, xlim=c(0,10), ylim=c(0,10))
```

```
> lines(c(0,10),c(5,5))
```



<Boxplot>

```
> boxplot(height$Height)
```



<Plot two vectors>

```
> x
```

```
[1] 1 2 9 9 10
```

```
> y
```

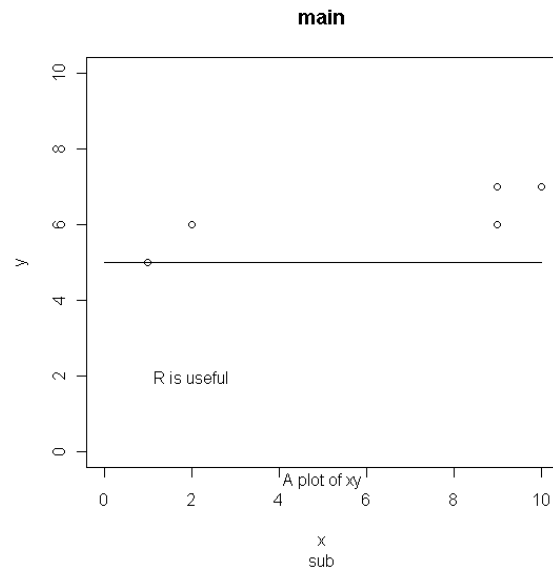
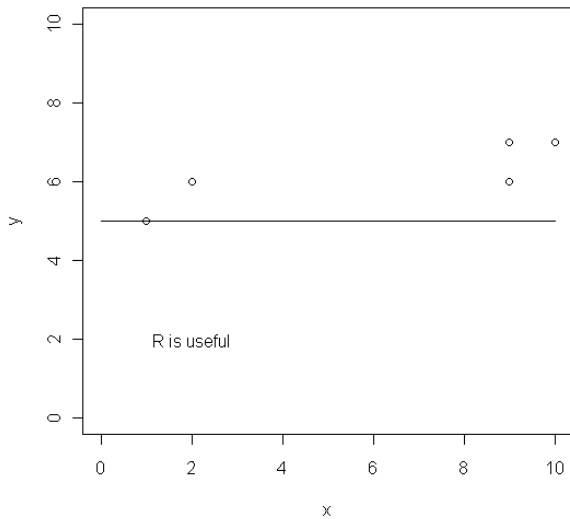
```
[1] 5 6 6 7 7
```

```
> plot(x,y)
```

☆ Adding text

(a) Inside the chart

```
> text(2,2,"R is useful")
```



(c) Legend

```
> legend(6, 4,c("sin", "cos", "tan"),col=2:4, lty = 1:3,bg='gray90')
```

(b) Outside the chart

```
> title("main","sub")
```

```
> mtext("A plot of xy", side=1)
```

#If you want to place the text at far side, you can use the parameter “line” as line=3. The number shows the number of lines between the text and the chart.

#The number in “side” indicates the position of the text

