Типы данных в R и принципы работы с ними

© А.Б. Шипунов*, Е.М. Балдин†

3 мая 2008 г.

1 Числовые векторы

Простейший вектор (рост сотрудников):

```
> x <- c(174, 162, 188, 192, 165, 168, 172)

[1] 174 162 188 192 165 168 172
```

Структура вектора:

```
> str(x)

 num [1:7] 174 162 188 192 165 168 172
NULL
```

Проверка: «А вектор ли это?»:

```
> is.vector(x)

[1] TRUE
```

2 Факторы

Текстовый (character) вектор (пол сотрудников):

```
> sex <- c("male", "female", "male", "male", "female", "male",
+     "male")

[1] "male"   "female" "male"   "male"   "female" "male"   "male"
```

*e-mail: dactylorhiza@gmail.com
†e-mail: E.M.Baldin@inp.nsk.su

```
> is.character(sex)

[1] TRUE

> is.vector(sex)

[1] TRUE

> str(sex)

 chr [1:7] "male" "female" "male" "male" "female" "male" ...
NULL
```

Содержимое текстового вектора:

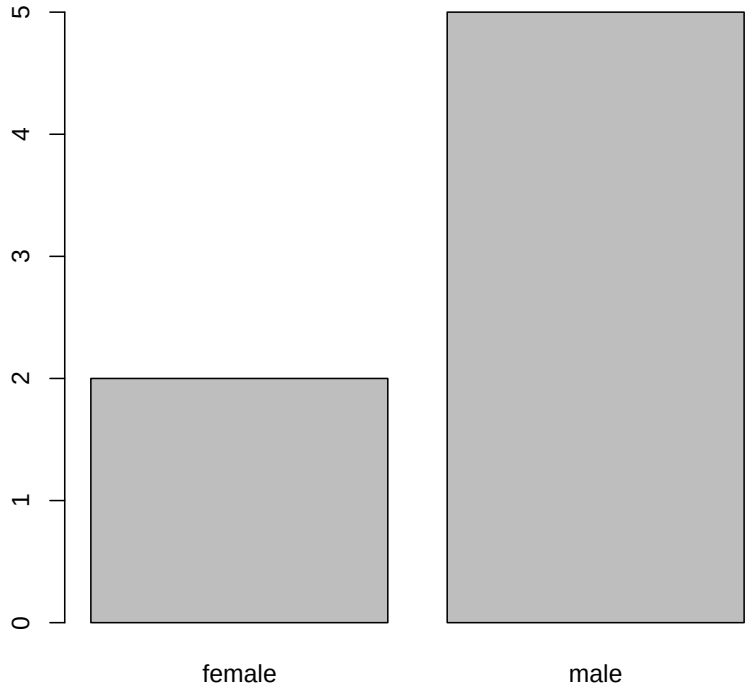
```
> sex

[1] "male"   "female" "male"   "male"   "female" "male"   "male"

> sex[1]

[1] "male"

> table(sex)

sex
female   male
     2      5
```

Задаём фактор:

```
> sex.f <- factor(sex)

[1] male   female male   male   female male   male
Levels: female male

> sex.f

[1] male   female male   male   female male   male
Levels: female male
```

График:

```
> plot(sex.f)

     [,1]
[1,]  0.7
[2,]  1.9
```



Что такое фактор:

```
> is.factor(sex.f)

[1] TRUE

> is.character(sex.f)

[1] FALSE

> str(sex.f)

 Factor w/ 2 levels "female","male": 2 1 2 2 1 2 2
NULL
```

Доступ к данным:

```
> sex.f[5:6]

[1] female male
Levels: female male
```

```
> sex.f[6:7]

[1] male male
Levels: female male

> sex.f[6:7, drop = TRUE]

[1] male male
Levels: male

> factor(as.character(sex.f[6:7]))

[1] male male
Levels: male
```

Приведение к цифровому виду:

```
> as.numeric(sex.f)

[1] 2 1 2 2 1 2 2
```

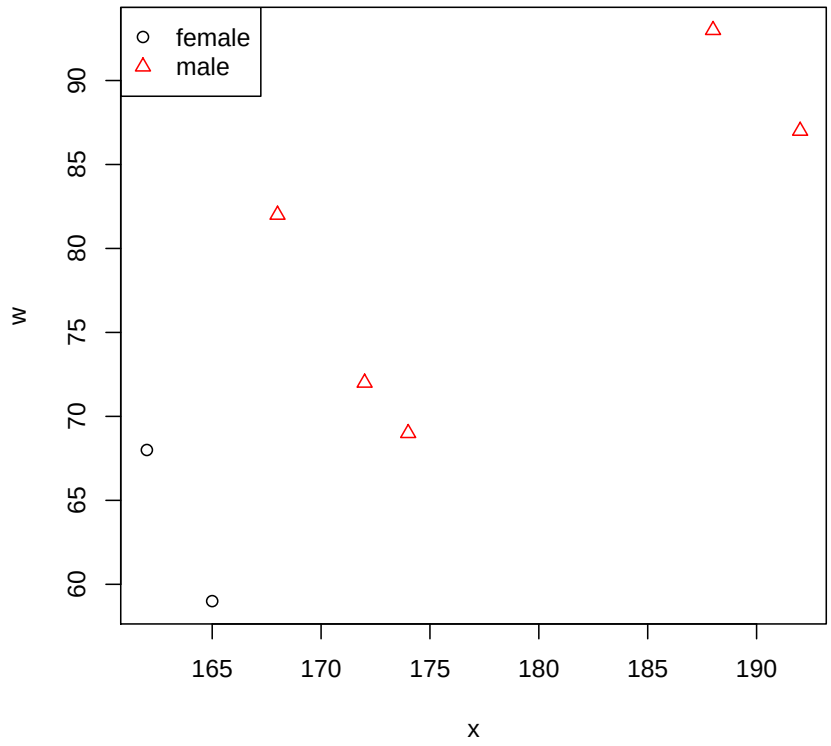
Ещё один вектор (вес сотрудников):

```
> w <- c(69, 68, 93, 87, 59, 82, 72)

[1] 69 68 93 87 59 82 72
```

Построение графика:

```
> plot(x, w, pch = as.numeric(sex.f), col = as.numeric(sex.f))

NULL

> legend("topleft", pch = 1:2, col = 1:2, legend = levels(sex.f))

$rect
$rect$w
[1] 5.993882

$rect$h
[1] 4.385668

$rect$left
[1] 160.8

$rect$top
[1] 94.36
```

```
$text
$text$x
[1] 163.3208 163.3208

$text$y
[1] 92.89811 91.43622
```



Ещё один текстовый вектор (размер маек сотрудников):

```
> m <- c("L", "S", "XL", "XXL", "S", "M", "L")

[1] "L"   "S"   "XL"  "XXL" "S"   "M"   "L"

> m.f <- factor(m)

[1] L   S   XL  XXL S   M   L
Levels: L M S XL XXL

> m.f
```

```
[1] L   S   XL  XXL S   M   L
Levels: L M S XL XXL
```

Упорядочиваем текстовые данные:

```
> m.o <- ordered(m.f, levels = c("S", "M", "L", "XL", "XXL"))

[1] L   S   XL  XXL S   M   L
Levels: S < M < L < XL < XXL

> m.o

[1] L   S   XL  XXL S   M   L
Levels: S < M < L < XL < XXL
```

3 Пропущенные данные

Вектор данных с пропущенными значениями (время на сон) :

```
> h <- c(8, 10, NA, NA, 8, NA, 8)

[1]  8 10 NA NA  8 NA  8

> h

[1]  8 10 NA NA  8 NA  8
```

Вычисление среднего если есть пропущенные данные:

```
> mean(h)

[1] NA

> mean(h, na.rm = TRUE)

[1] 8.5

> mean(na.omit(h))

[1] 8.5
```

Замена пропущенных данных на среднее по выборке:

```
> h[is.na(h)] <- mean(h, na.rm = TRUE)

[1] 8.5

> h

[1]  8.0 10.0  8.5  8.5  8.0  8.5  8.0
```

4 Матрицы

Матрицы 2×2:

```
> m <- 1:4

[1] 1 2 3 4

> m

[1] 1 2 3 4

> ma <- matrix(m, ncol = 2, byrow = TRUE)

     [,1] [,2]
[1,]    1    2
[2,]    3    4

> ma

     [,1] [,2]
[1,]    1    2
[2,]    3    4

> str(ma)

 int [1:2, 1:2] 1 3 2 4
NULL

> str(m)

 int [1:4] 1 2 3 4
NULL

> mb <- m

[1] 1 2 3 4

> mb

[1] 1 2 3 4

> attr(mb, "dim") <- c(2, 2)

[1] 2 2

> mb
```

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

Матрица $2 \times 2 \times 2$:

```
> m3 <- 1:8

[1] 1 2 3 4 5 6 7 8

> dim(m3) <- c(2, 2, 2)

[1] 2 2 2

> m3

, , 1

     [,1] [,2]
[1,]    1    3
[2,]    2    4

, , 2

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

5 Списки

Обычный список:

```
> l <- list("R", 1:3, TRUE, NA, list("r", 4))

[[1]]
[1] "R"

[[2]]
[1] 1 2 3

[[3]]
[1] TRUE

[[4]]
[1] NA
```

```
[[5]]
[[5]][[1]]
[1] "r"

[[5]][[2]]
[1] 4
```

```
> l
```

```
[[1]]
[1] "R"

[[2]]
[1] 1 2 3

[[3]]
[1] TRUE

[[4]]
[1] NA

[[5]]
[[5]][[1]]
[1] "r"

[[5]][[2]]
[1] 4
```

Способы доступа к данным:

```
> h[3]
```

```
[1] 8.5
```

```
> ma[2, 1]
```

```
[1] 3
```

```
> l[1]
```

```
[[1]]
[1] "R"
```

```
> str(l[1])
```

```
List of 1
 $ : chr "R"
NULL
```

```
> l[[1]]
```

```
[1] "R"
```

```
> str(l[[1]])
```

```
 chr "R"
NULL
```

Именование списка:

```
> names(l) <- c("first", "second", "third", "fourth", "fifth")
```

```
[1] "first"  "second" "third"  "fourth" "fifth"
```

```
> str(l$first)
```

```
 chr "R"
NULL
```

Именование векторов и матриц:

```
> names(w) <- c("Коля", "Женя", "Петя", "Саша", "Катя", "Вася",
+     "Жора")
```

```
[1] "Коля" "Женя" "Петя" "Саша" "Катя" "Вася" "Жора"
```

```
> w
```

```
Коля Женя Петя Саша Катя Вася Жора
  69   68   93   87   59   82   72
```

```
> w["Женя"]
```

```
Женя
  68
```

```
> rownames(ma) <- c("a1", "a2")
```

```
[1] "a1" "a2"
```

```
> colnames(ma) <- c("b1", "b2")
```

```
[1] "b1" "b2"
```

```
> ma
```

```
   b1 b2
a1  1  2
a2  3  4
```

6 Таблицы данных

Пример таблицы данных:

```
> d <- data.frame(weight = w, height = x, size = m.o, sex = sex.f)

     weight height size    sex
Коля     69    174    L   male
Женя     68    162    S female
Петя     93    188   XL   male
Саша     87    192  XXL   male
Катя     59    165    S female
Вася     82    168    M   male
Жора     72    172    L   male

> d

     weight height size    sex
Коля     69    174    L   male
Женя     68    162    S female
Петя     93    188   XL   male
Саша     87    192  XXL   male
Катя     59    165    S female
Вася     82    168    M   male
Жора     72    172    L   male

> str(d)

'data.frame':        7 obs. of  4 variables:
 $ weight: num  69 68 93 87 59 82 72
 $ height: num  174 162 188 192 165 168 172
 $ size  : Ord.factor w/ 5 levels "S"<"M"<"L"<"XL"<..: 3 1 4 5 1 2 3
 $ sex   : Factor w/ 2 levels "female","male": 2 1 2 2 1 2 2
NULL
```

Доступ к данным таблицы:

```
> d$weight

[1] 69 68 93 87 59 82 72

> d[[1]]

[1] 69 68 93 87 59 82 72

> d[, 1]
```

```
[1] 69 68 93 87 59 82 72
```

```
> d[, "weight"]
```

```
[1] 69 68 93 87 59 82 72
```

Выбор нужных колонок:

```
> d[, 2:4]
```

```
     height size    sex
Коля    174    L   male
Женя    162    S female
Петя    188   XL   male
Саша    192  XXL   male
Катя    165    S female
Вася    168    M   male
Жора    172    L   male
```

```
> d[, -1]
```

```
     height size    sex
Коля    174    L   male
Женя    162    S female
Петя    188   XL   male
Саша    192  XXL   male
Катя    165    S female
Вася    168    M   male
Жора    172    L   male
```

Выборка данных по условию:

```
> d$sex == "female"
```

```
[1] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE
```

```
> d[d$sex == "female", ]
```

```
     weight height size    sex
Женя     68    162    S female
Катя     59    165    S female
```

Сортировка выборки:

```
> d[order(d$sex, d$height), ]
```

```
     weight height size    sex
Женя     68    162    S female
Катя     59    165    S female
Вася     82    168    M   male
Жора     72    172    L   male
Коля     69    174    L   male
Петя     93    188   XL   male
Саша     87    192  XXL   male
```

7 Векторизованные вычисления

Простые операции над векторами:

```
> w * 1000

 Коля  Женя  Петя  Саша  Катя  Вася  Жора
69000 68000 93000 87000 59000 82000 72000
```

Циклы:

```
> for (i in seq_along(w)) {
+     w[i] <- w[i] * 1000
+ }

 Жора
72000

> w

 Коля  Женя  Петя  Саша  Катя  Вася  Жора
69000 68000 93000 87000 59000 82000 72000
```

Операции с матрицами:

```
> ma + mb

   b1 b2
a1  2  5
a2  5  8

> 1:8 + 1:2

[1]  2  4  4  6  6  8  8 10
```

Векторные операции:

```
> apply(trees, 2, mean)
```

```
   Girth   Height   Volume
13.24839 76.00000 30.17097

> sapply(d, as.numeric)

     weight height size sex
[1,]     69    174    3   2
[2,]     68    162    1   1
[3,]     93    188    4   2
[4,]     87    192    5   2
[5,]     59    165    1   1
[6,]     82    168    2   2
[7,]     72    172    3   2

> by(d[, 1:2], d$sex, mean)

d$sex: female
weight height
  63.5  163.5
------------------------------------------------------------
d$sex: male
weight height
  80.6  178.8

> lapply(d, is.factor)

$weight
[1] FALSE

$height
[1] FALSE

$size
[1] TRUE

$sex
[1] TRUE
```