第 5 章

向　量

向量是相同类型元素的序列。除了第一个元素之外,每个元素都有唯一前驱;除了最后一个元素外,每个元素都有唯一后继。因为元素之间有前驱后继关系,所以每个元素都具有唯一索引。在线性代数中,含 N 个元素的向量称为 N 维向量。在几何中,从原点穿过 N 维中给定点的定向射线称为 N 维向量。向量的长度为其含有的元素个数,可使用 R 函数 length 查询。标量是长度为 1 的向量。向量中元素的数据类型可使用 R 函数 typeof 查询。向量如果由数值类型的元素组成,则称为数值向量;如果由字符类型的元素组成,则称为字符向量,字符类型的元素就是一个字符串;由逻辑值 TRUE 或者 FALSE 组成的向量称为逻辑向量。

5.1 创建向量

创建向量的方法有枚举法、描述法和数列法等。

5.1.1 枚举法

枚举法是把向量中的元素作为函数 c 的参数逐一列出来创建向量。函数调用 c()的含义是“把参数组合(combine)成向量”。下面是使用枚举法创建向量的例子。

创建由两个实数 0.5、0.6 构成的向量:

```
>c(0.5, 0.6)
[1] 0.5 0.6
```

创建由逻辑值“真、假、真”组成的向量:

```
>c(TRUE, FALSE,TRUE)
[1]  TRUE FALSE  TRUE
```

创建由字符 a、b、c 组成的向量:

```
>c("a", "b", "c")
[1] "a""b""c"
```

如果以 NULL 为枚举元素创建向量,则 NULL 并不是向量的元素;而 NA、NaN、Inf 都可以作为向量的元素。例如:

```
>x <- c(2, NULL, 3); x
[1]  2  3
>x<- c(2,NA,3); x
[1]  2  NA  3
```

```
>x<- c(2,NaN,3); x
[1]  2    NaN   3
>x<- c(2,Inf,3); x
[1]  2    Inf   3
```

也就是说,c(2,NULL,3)等同于 c(2,3)。向量 c(2,NULL,3)的长度为 2:

```
> length(c(2, NULL, 3))
[1] 2
```

5.1.2 描述法

描述法是通过元素类型和长度两个参数来创建向量。该方法适合创建长向量。为向量预先分配空间然后修改向量中的元素要比动态增、删、改向量中的元素效率高些。

创建元素为逻辑类型、长度为 5 的向量:

```
> vector("logical", 5)
[1] FALSE FALSE FALSE FALSE FALSE
```

创建元素为整数类型、长度为 5 的向量:

```
> vector("integer", 5)
[1] 0 0 0 0 0
```

创建元素为实数类型、长度为 5 的向量:

```
> vector("double", 5)
[1] 0 0 0 0 0
```

创建元素为字符类型、长度为 5 的向量:

```
> vector("character", 5)
[1] "" "" "" "" ""
```

使用函数调用 logical()、numeric()、character()也能创建特定类型的向量。例如:

```
> logical(5)
[1] FALSE FALSE FALSE FALSE FALSE
> numeric(5)
[1] 0 0 0 0 0
> character(5)
[1] "" "" "" "" ""
> aNumber <- 12
> length(aNumber)
[1] 1
```

没有元素的向量称为空向量。空向量虽然长度为 0,但具有类型:

```
> vector()
logical(0)
> integer()
integer(0)
> character()
character(0)
```

函数调用 typeof()查询向量中元素的类型。表 5-1 描述了 typeof()可能的返回值以及它们的含义。typeof()的返回值可能是 logical、integer、double、complex、character 或 raw 等。

表 5-1　typeof()可能的返回值以及它们的含义

返回值	含义	返回值	含义
NULL	无	symbol	符号
logical	逻辑	closure	闭包(函数)
integer	整数	externalptr	外部指针
double	实数	weakref	弱引用
complex	复数	pairlist	成对列表
character	字符	environment	环境
list	列表	promise	承诺
raw	含字节值的向量	language	语言
…	任意数量的参数	special	不可针对参数求值的内置函数
any	任何类型	builtin	可针对参数求值的内置函数
expression	表达式		

5.1.3　数列法

数列法是通过设置首项、末项和公差等参数，函数调用 seq()生成具有数列特征的向量。下面是使用数列法创建向量的例子。

把首项为 1、末项为 10，公差为 2 的数列创建为向量：

```
> seq(1,10,by = 2)
[1] 1 3 5 7 9
```

把首项为 1、末项为 11，长度为 3 的数列创建为向量：

```
> seq(1, 11, length = 3)
[1]  1  6  11
```

这种情况下，R 首先计算公差为 5，然后得出数列。首项和末项的参数名分别是 from 和 to，默认公差为 1：

```
>seq(from = 1, to = 10, by = 2)
```

如果把公差为 0 的等差数列创建为向量，则使用 rep()函数调用，读作 repeat。例如，把 2 3 重复 5 次：

```
>rep(2:3, times = 5)
[1] 2 3 2 3 2 3 2 3 2 3
```

把 2 3 各自重复 5 次：

```
> rep(2:3, each = 5)
[1] 2 2 2 2 2 3 3 3 3 3
```

把整数序列 2 3 3 4 4 4 5 5 5 5 创建为向量。经观察，发现该序列把 2 重复 1 次、把 3 重复 2 次、把 4 重复 3 次、把 5 重复 4 次，所以使用参数 c(2,3,4,5)和 c(1,2,3,4)来调用 rep()：

```
> rep(c(2,3,4,5), c(1,2,3,4))
[1] 2 3 3 4 4 4 5 5 5 5
```

R 中内置了两个向量，即 LETTERS 和 letters，分别是 26 个大写字母和小写字母：

```
> LETTERS
[1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O" "P" "Q" "R"
[19] "S" "T" "U" "V" "W" "X" "Y" "Z"
```

冒号运算符生成以操作数作为首项和末项，公差为 1 的等差数列：

```
> c <- 1:10
> c
[1]  1  2  3  4  5  6  7  8  9  10
```

1：10 的结果与 seq(1,10)相同。

5.1.4 向量元素的命名

创建向量时可通过在实际参数前的“＝”命名向量中的元素。例如：

```
> c(FIRST=11, SECOND= 12, THIRD = 13)
  FIRST SECOND  THIRD
     11     12     13
```

创建向量后则通过 names()为向量中各个元素命名，例如：

```
> x <- c(11, 12, 13) ; x
[1] 11 12 13
> names(x)
NULL
> names(x) <- c("FIRST","SECOND","THIRD")
> x
FIRST SECOND  THIRD
    11     12     13
```

如果向量元素的名字中含有空格、小数点，或者以数字打头，则要使用双引号引起来。例如：

```
>c(FIRST = 2, "D.D" = 3)
FIRST   D.D
    2     3
```

5.1.5 类型判断与类型转换

判断向量类型的函数和类型转换函数如表 5-2 所示。

表 5-2 判断函数和转换函数

判断函数	转换函数	判断函数	转换函数
is.numeric()	as.numeric()	is.data.frame()	as.data.frame()
is.character()	as.character()	is.factor()	as.factor()
is.vector()	as.vector()	is.logical()	as.logical()
is.matrix()	as.matrix()		

例如：

```
> a <- c(1,2,3);a
[1] 1 2 3
> is.numeric(a)
[1] TRUE
> is.vector(a)
[1] TRUE
```

将 a 转换为字符类型：

```
> a <- as.character(a);a
[1] "1""2""3"
> is.numeric(a)
[1] FALSE
> is.vector(a)
[1] TRUE
> is.character(a)
[1] TRUE
```

将逻辑类型数据转换为 0 与 1：

```
> b <- c(TRUE,FALSE) ;b
[1] TRUE FALSE
> b <- as.numeric(b); b
[1] 1 0
```

5.2 访问向量

访问就是读或写，写操作包括增加、修改和删除。可通过元素的索引(index)或者名字以及成员访问运算符[]访问向量中的元素。索引号是一个正整数，表示访问向量中该索引位置上的元素；在索引号前面使用减号-表示访问除了该索引位置以外的元素。R 中向量元素的索引从 1 开始，这与通常的统计或数学软件一致；而 C 语言等计算机高级语言的数组索引则从 0 开始。例如，假设有向量 alpha：

```
>alpha<- c(42,7,64,9)
```

查询向量 alpha 中的第 2 个元素：

```
>alpha[2]
7
```

查询向量 alpha 中的第 1 和第 4 个元素：

```
>alpha[c(1,4)]
[1] 42 9
```

查询向量 alpha 中的第 2～4 个元素：

```
>alpha[2:4]
[1]  7 64  9
```

或者：

```
>alpha[c(2,3,4)]
[1]  7 64  9
```

查询向量 alpha 中除了第 2 个元素之外的其他元素：

```
>alpha[-2]
[1] 42 64 9
```

如果定义了元素的名字，可以通过名字访问向量而不使用索引：

```
>alpha <- c(FIRST = 2, "D.D" = 3)
FIRST   D.D
    2     3
>alpha["FIRST"]
FIRST
    2
```

向量上的关系运算的结果是一个逻辑向量。例如，判断向量中每个元素是否大于 10：

```
>alpha > 10
[1] TRUE  FALSE  TRUE  FALSE
```

而通过逻辑向量能够查询向量中对应逻辑为真的元素：

```
>alpha[c(TRUE, FALSE, TRUE, FALSE)]
[1] 42 64
```

这两个步骤可以合为一步，直接在[]中使用关系表达式：

```
>alpha[alpha>10]
[1] 42 64
```

通过组合已有向量和新元素可实现向量元素的添加。例如，把 55 添加到向量 alpha 中：

```
>alpha <- c(1,2,3)
>alpha <- c(alpha,55)
>alpha
[1] 1 2 3 55
```

组合两个向量 alpha 和 beta：

```
>alpha <- c(1, 2, 3)
>beta <- c(4, 5, 6)
>c(alpha,beta)
[1] 1 2 3 4 5 6
```

函数 append 用来在指定位置添加新的元素。例如，在向量 alpha 中的第 3 个元素后面插入 10：

```
>alpha <- c(1,2,3,4,5,6)
> append(alpha,10,after=3)
[1]  1  2  3  10  4  5  6
```

通过重新绑定可实现对向量元素的修改。例如，把向量 alpha 中索引位置 2 上的元素改为 9：

```
>alpha <- c(3,4,2); alpha
[1] 3 4 2
>alpha[2] <- 9
>alpha
[1] 3 9 2
```

把向量 alpha 中索引位置 7 上的元素绑定为 9：

```
>alpha  <- c(3,4,2); alpha
[1] 3 4 2
>alpha[7] <- 9; alpha
[1] 5 4 2 NA NA NA 9
```

修改前的向量长度为 3，而要求修改索引位置 7，那么索引位置 4、5、6 为 NA。

函数 replace 也可以修改指定索引位置上的元素。例如，把向量 alpha 中索引位置 7 和 8 上的元素都改为 0：

```
>alpha
[1]  1  7  3  4  5  6  NA  NA  7
> replace(alpha, list = c(7,8), 0)
[1]  1  7  3  4  5  6  0   0   7
>alpha
[1]  1  7  3  4  5  6  NA  NA  7
```

如果想删除某元素，那么使用负的索引查询该索引以外的元素，然后与一个新的名字绑定即可。例如，从向量 alpha 中删除元素 3：

```
>alpha <- c(3,4,2); alpha
[1] 3 4 2
>beta <- alpha[-1]; beta
[1] 4 2
```

以向量为函数的参数，一般会对每个元素都应用函数。例如，计算向量的绝对值：

```
>abs(c(1, -4, 6,-25))
[1] 1 4 6 25
```

5.3　算术运算

向量的算术运算符有加、减、乘、除、幂、取余数和取商等，如表 5-3 所示。

表 5-3 向量的算术运算符

运 算 符	描 述	运 算 符	描 述
+	加	^	幂
−	减	%%	整除,取余数
*	乘	%/%	整除,取商
/	除		

例如,两个向量 alpha 和 beta 相加:

```
>alpha<- c(2,3,2,3,2,3)
>beta <- c(2,3)
>gamma<- alpha + beta
>gamma
[1] 4 6 4 6 4 6
```

两个向量相减:

```
>gamma - beta
[1] 2 3 2 3 2 3
```

两个向量相乘:

```
>gamma <- alpha * beta; gamma
[1] 4 9 4 9 4 9
```

两个向量相除:

```
>gamma / beta
[1] 2 3 2 3 2 3
```

两个向量相除,取余数:

```
> c(3, 4, 5, 6, 7, 8) %% c(2, 3)
[1] 1 1 1 0 1 2
```

两个向量相除,取商:

```
> c(3, 4, 5, 6, 7, 8) %/% c(2, 3)
[1] 1 1 2 2 3 2
```

对向量取负:

```
> -alpha
[1] -2 -3 -2 -3 -2 -3
```

如果两个向量长度不同,那么将重复使用较短向量与较长向量进行运算,这称为向量运算中的循环法则。例如:

```
> alpha <- c(2, 3, 2, 3, 2, 3)
> beta <- c(2, 3)
>alpha+beta
[1] 4 6 4 6 4 6
```

在这个例子中,向量 alpha 的长度是 6,向量 beta 的长度是 2,那么根据循环法则,首先重复向量 beta 直到与 alpha 的长度一致,成为(2,3,2,3,2,3),然后再相加。

如果较短向量的长度不能整除较长向量的长度，则截断短向量，使得运算结果向量与较长向量的长度一致。例如：

```
>alpha <- c(2, 3, 2, 3, 2) ; beta <- c(2, 3)
> alpha + beta
[1] 4 6 4 6 4
```

在这个例子中，向量 alpha 的长度是 5，向量 beta 的长度是 2，向量 alpha 不是向量 beta 的整数倍，那么在循环向量 beta 两次后截断向量 beta，成为(2，3，2，3，2)，然后再相加。

向量的内积(·)，也称为数量积，或点乘，就是把两个向量对应元素相乘之后再相加，结果是一个标量。内积的 R 运算符是 *。

使用内积可计算两个向量的夹角余弦。一般地，设有 n 维向量：

$$\boldsymbol{\alpha}=\begin{pmatrix}a_1\\a_2\\\vdots\\a_n\end{pmatrix},\quad \boldsymbol{\beta}=\begin{pmatrix}b_1\\b_2\\\vdots\\b_n\end{pmatrix}$$

向量 $\boldsymbol{\alpha}$ 和向量 $\boldsymbol{\beta}$ 的内积 $\boldsymbol{\alpha}\cdot\boldsymbol{\beta}=a_1b_1+a_2b_2+\cdots+a_nb_n$，有

$$\boldsymbol{\alpha}\cdot\boldsymbol{\beta}=|\boldsymbol{\alpha}||\boldsymbol{\beta}|\cos\theta$$

其中，$||$表示向量的长度，θ 是两个向量的夹角。

设 $\boldsymbol{\alpha}=c(a_1,a_2)$，$\boldsymbol{\beta}=c(b_1,b_2)$，下面给出证明。

令 $\boldsymbol{\gamma}=\boldsymbol{\alpha}-\boldsymbol{\beta}$。根据三角形余弦定理：

$$\boldsymbol{\gamma}^2=\boldsymbol{\alpha}^2+\boldsymbol{\beta}^2-2|\boldsymbol{\alpha}||\boldsymbol{\beta}|\cos\theta$$

代入 $\boldsymbol{\gamma}=\boldsymbol{\alpha}-\boldsymbol{\beta}$，得 $(\boldsymbol{\alpha}-\boldsymbol{\beta})^2=\boldsymbol{\alpha}^2+\boldsymbol{\beta}^2-2|\boldsymbol{\alpha}||\boldsymbol{\beta}|\cos\theta$，展开：

$$(\boldsymbol{\alpha}-\boldsymbol{\beta})(\boldsymbol{\alpha}-\boldsymbol{\beta})=\boldsymbol{\alpha}^2+\boldsymbol{\beta}^2-2|\boldsymbol{\alpha}||\boldsymbol{\beta}|\cos\theta$$

$$\boldsymbol{\alpha}^2-2\boldsymbol{\alpha}\boldsymbol{\beta}+\boldsymbol{\beta}^2=\boldsymbol{\alpha}^2+\boldsymbol{\beta}^2-2|\boldsymbol{\alpha}||\boldsymbol{\beta}|\cos\theta$$

所以

$$-2\boldsymbol{\alpha}\boldsymbol{\beta}=-2|\boldsymbol{\alpha}||\boldsymbol{\beta}|\cos\theta$$

$$\boldsymbol{\alpha}\boldsymbol{\beta}=|\boldsymbol{\alpha}||\boldsymbol{\beta}|\cos\theta$$

这样，使用内积可计算两个向量的夹角余弦：

$$\cos\theta=\frac{\boldsymbol{\alpha}\cdot\boldsymbol{\beta}}{|\boldsymbol{\alpha}||\boldsymbol{\beta}|}$$

向量 $\boldsymbol{\alpha}$、$\boldsymbol{\beta}$ 的长度都是可以计算的，从而 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 间的夹角也可以从反三角函数计算得出。进而可以进一步判断两个向量是否为同一方向或正交(即垂直)等方向关系：

$\boldsymbol{\alpha}\cdot\boldsymbol{\beta}>0$：夹角为 0°～90°，同向。

$\boldsymbol{\alpha}\cdot\boldsymbol{\beta}=0$：正交，相互垂直。

$\boldsymbol{\alpha}\cdot\boldsymbol{\beta}<0$：夹角为 90°～180°，异向。

向量的叉乘(×)也称为向量积、外积，使用运算符%o%实现。例如：

```
> c(1,2) %o% c(3,4)
     [,1] [,2]
[1,]    3    4
[2,]    6    8
```

```
> c(1,2,3) %o% c(3,4,5)
     [,1] [,2] [,3]
[1,]    3    4    5
[2,]    6    8   10
[3,]    9   12   15
```

一个向量乘以一个标量,就是线性代数中的矩阵的数乘运算。

```
x <- c(1, 10)
> x %o% 2
     [,1]
[1,]    2
[2,]   20
```

一般地,对于向量 $\boldsymbol{\alpha}$ 和向量 $\boldsymbol{\beta}$:

$$\boldsymbol{\alpha}=(a,b,c)$$

$$\boldsymbol{\beta}=(x,y,z)$$

叉乘 $\boldsymbol{\alpha}\times\boldsymbol{\beta}$ 为

$$\begin{array}{cccccc} + & + & + & - & - & - \\ \boldsymbol{i} & \boldsymbol{j} & \boldsymbol{k} & \boldsymbol{i} & \boldsymbol{j} & \boldsymbol{k} \\ a & b & c & a & b & c \\ x & y & z & x & y & z \end{array}$$

即

$$+\boldsymbol{i}bz+\boldsymbol{j}cx+\boldsymbol{k}ay-\boldsymbol{i}cy-\boldsymbol{j}az-\boldsymbol{k}bx=\boldsymbol{i}(bz-cy)+\boldsymbol{j}(cx-az)+\boldsymbol{k}(ay-yx)$$

其中,$\boldsymbol{i}$、$\boldsymbol{j}$、$\boldsymbol{k}$ 是基向量:

$$\boldsymbol{i}=(1,0,0)$$

$$\boldsymbol{j}=(0,1,0)$$

$$\boldsymbol{k}=(0,0,1)$$

代入公式得

$$\boldsymbol{\alpha}\times\boldsymbol{\beta}=(bz-yc,cx-az,ay-bx)$$

5.4 逻辑运算和关系运算

逻辑向量的元素只能为 TRUE、FALSE。TRUE、FALSE 可以分别简写为 T、F,但实际上 T 和 F 仅仅是值为 TRUE、FALSE 的变量。

逻辑运算和关系运算的结果是逻辑向量。逻辑运算符和关系运算符如表 5-4 和表 5-5 所示。

表 5-4 逻辑运算符

运算符	描述
!	非
|	或
&	与