

第四周——列表与因子

题目目的

- (一) 掌握列表的创建与操作。
- (二) 掌握因子变量的定义。
- (三) 掌握与因子相关的 R 函数。

题目

题目一：创建脚本文件 test0401.R，完成下面操作。

使用 `list()` 函数创建一个包含以下元素的列表 `my.list1`：

- 一个向量 (vector)：包含 5 个整数，从 1 到 5；
- 一个标量 (scalar)：赋值为 6.78
- 一个矩阵 (matrix)：3 行 4 列，填充数据由 `rpois(12, 20)` 产生；
- 一个数据框 (data.frame)：含有 2 列，列名分别为 Name 和 Age，Name 的数据为“Alice”，“Bob”，“Charlie”，“David”，“Eve”；Age 的数据为 23, 27, 22, 31, 29

用 `[]` 运算符切片列表，获取 `my.list1` 中的第 1、第 3、第 4 个子对象，并将其赋值给一个新的变量 `vector_slice`。

用 `$` 和 `[[]]` 运算符分别引用 `my.list1` 中的列表的标量和矩阵部分。

提取 `my.list1` 中的 `data.frame` 子对象中 Name 为 Bob 的行。

题目二：创建脚本文件 test0402.R，完成下面操作。

用 `list()` 函数创建如实验一所示的列表，并赋值给 `my.list2`。

给列表 `my.list2` 添加一个子对象 `char`，值为”New Element”。

删除列表 `my.list2` 中的子对象 `matrix`。

修改列表 `my.list` 中 `vector` 向量的最后一个元素，将其改为 6。

给列表 `my.list2` 中 `vector` 向量的第一个元素重命名，改为”First”。

使用 `lapply()` 函数计算列表 `my.list2` 中的所有子对象的长度。

使用 `sapply()` 函数计算列表 `my.list2` 中的所有子对象的长度。

题目三：因子变量的定义与操作。创建脚本文件 test0403.R，完成下面操作。

用 `data.frame` 函数定义一个数据框 `df`，数据如下所示，并用参数 `stringsAsFactors` 将 `diabetes` 和 `status` 这两个变量设为因子变量，然后用 `str` 查看 `df` 的结构。

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

用 `factor` 函数将数据框 `df` 中 `status` 列转换为有序因子变量，用 `str` 函数观察 `status` 的结构。

用 `factor` 函数创建一个有序因子变量 `ord_factor`，变量的值为”low”，”medium”，”high”，”medium”，”low”，”high”；因子的序为”low”、”medium”、”high”。

题目四：split、lapply 和 tapply 函数的使用。创建脚本文件 test0404.R，完成下面操作。下面操作使用鸢尾花（iris）数据集，该数据集包含了 150 个鸢尾花的四个属性（Sepal.Length, Sepal.Width, Petal.Length, Petal.Width）和一个分类变量 Species。

用 split 函数以 Species 为分组标准对数据集进行分组。

计算每组数据的数据量。

计算每组的花萼（sepal）和花瓣（petal）的平均长度。

题目五：tapply 函数、by 函数和 aggregate 函数的应用。创建脚本文件 test0405.R，完成下面操作。

用 by 函数以 Species 为分组标准对 iris 数据集分组计算花萼和花瓣的平均长度。

用 tapply 函数对 iris 数据集按 Species 进行分组汇总，计算花萼和花瓣宽度的中位数。

用 aggregate 函数对 iris 数据集进行分组汇总，计算花萼和花瓣的平均长度与平均宽度。

题目六：cut 函数和 table 函数的应用。打开脚本文件 test0406.R，完成下面操作。

用 cut 函数将 age 变量分为三个年龄段（“Young”，“Middle”，“Old”），对应的年龄区间为 (18, 35], (35, 55], (55, 70]，并赋值给变量 age.group。

用 table 函数统计每个 age.group 年龄段在数据集 data 出现的频数。

统计每个年龄段不同收入水平的频数。

```
set.seed(123)
ages <- sample(18:65, 100, replace = TRUE)
```

```
income_levels <- sample(c("Low", "Medium", "High"), 100, replace = TRUE)
data <- data.frame(age = ages, income.level = income_levels)
```

答案及解析

题目一：

```
my.list1 <- list(
  vector = 1:5,
  scalar = 6.78,
  matrix = matrix(rpois(12, 20), nrow = 3, ncol = 4),
  data_frame = data.frame(
    Name = c("Alice", "Bob", "Charlie", "David", "Eve"),
    Age = c(23, 27, 22, 31, 29)
  )
)

print(my.list1)
```

\$vector

[1] 1 2 3 4 5

\$scalar

[1] 6.78

\$matrix

	[,1]	[,2]	[,3]	[,4]
[1,]	10	13	9	19
[2,]	30	15	21	19
[3,]	28	30	15	12

```
$data_frame
  Name Age
1  Alice 23
2   Bob 27
3 Charlie 22
4  David 31
5   Eve 29
```

```
vector_slice <- my.list1[c(1, 3, 4)]
print(vector_slice)
```

```
$vector
[1] 1 2 3 4 5
```

```
$matrix
      [,1] [,2] [,3] [,4]
[1,]   10   13    9   19
[2,]   30   15   21   19
[3,]   28   30   15   12
```

```
$data_frame
  Name Age
1  Alice 23
2   Bob 27
3 Charlie 22
4  David 31
5   Eve 29
```

```
my.list1$scalar
```

```
[1] 6.78
```

```
my.list1[[3]]
```

```
      [,1] [,2] [,3] [,4]  
[1,]   10   13    9   19  
[2,]   30   15   21   19  
[3,]   28   30   15   12
```

```
bob_row <- my.list1$data_frame[my.list1$data_frame$Name == "Bob", ]
```

```
print(bob_row)
```

```
  Name Age  
2  Bob  27
```

题目二：

```
my.list2 <- list(  
  vector = 1:5,  
  scalar = 6.78,  
  matrix = matrix(rpois(12, 20), nrow = 3, ncol = 4),  
  data_frame = data.frame(Name = c("Alice", "Bob", "Charlie", "David", "Eve"), Age = c(  
)  
)  
my.list2$char = "New Element"  
my.list2
```

```
$vector  
[1] 1 2 3 4 5
```

```
$scalar  
[1] 6.78
```

```
$matrix
      [,1] [,2] [,3] [,4]
[1,]   24  15  17  14
[2,]   18  27  21  19
[3,]   24  22  28  29
```

```
$data_frame
      Name Age
1  Alice  23
2   Bob  27
3 Charlie 22
4  David  31
5   Eve  29
```

```
$char
[1] "New Element"
```

```
my.list2$matrix <- NULL
my.list2
```

```
$vector
[1] 1 2 3 4 5
```

```
$scalar
[1] 6.78
```

```
$data_frame
      Name Age
1  Alice  23
2   Bob  27
3 Charlie 22
4  David  31
5   Eve  29
```

```
$char  
[1] "New Element"
```

```
my.list2$vector[length(my.list2$vector)] = 6  
my.list2
```

```
$vector  
[1] 1 2 3 4 6
```

```
$scalar  
[1] 6.78
```

```
$data_frame  
      Name Age  
1  Alice  23  
2    Bob  27  
3 Charlie 22  
4  David  31  
5    Eve  29
```

```
$char  
[1] "New Element"
```

```
names(my.list2$vector)[1] <- "Frist"  
lapply(my.list2,length)
```

```
$vector  
[1] 5
```

```
$scalar  
[1] 1
```

```
$data_frame  
[1] 2
```



```
$char
[1] 1
```

```
sapply(my.list2,length)
```

vector	scalar	data_frame	char
5	1	2	1

题目三：

```
df <- data.frame(
  patientID = 1:4,
  age = c(25,34,28,52),
  diabetes = c('Type1','Tpye2','Type1','Type1'),
  status = c('Poor','Improved','Excellent','Poor'),
  stringsAsFactors = TRUE
)

str(df)
```

```
'data.frame':  4 obs. of  4 variables:
 $ patientID: int  1 2 3 4
 $ age      : num  25 34 28 52
 $ diabetes : Factor w/ 2 levels "Tpye2","Type1": 2 1 2 2
 $ status   : Factor w/ 3 levels "Excellent","Improved",...: 3 2 1 3
```

```
df_status <- factor(df$status,levels = c('Poor','Improved','Excellent'),ordered = T)
str(df_status)
```

```
Ord.factor w/ 3 levels "Poor"<"Improved"<...: 1 2 3 1
```

```
values <- c("low", "medium", "high", "medium", "low", "high")
ord_factor <- factor(values, levels = c("low", "medium", "high"), ordered = T)
str(ord_factor)
```

```
Ord.factor w/ 3 levels "low"<"medium"<...: 1 2 3 2 1 3
```

题目四：

```
head(iris)
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

```
iris_split <- split(iris, iris$Species)
print(iris_split)
```

```
$setosa
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa

10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3.0	1.4	0.1	setosa
14	4.3	3.0	1.1	0.1	setosa
15	5.8	4.0	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
18	5.1	3.5	1.4	0.3	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.5	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
22	5.1	3.7	1.5	0.4	setosa
23	4.6	3.6	1.0	0.2	setosa
24	5.1	3.3	1.7	0.5	setosa
25	4.8	3.4	1.9	0.2	setosa
26	5.0	3.0	1.6	0.2	setosa
27	5.0	3.4	1.6	0.4	setosa
28	5.2	3.5	1.5	0.2	setosa
29	5.2	3.4	1.4	0.2	setosa
30	4.7	3.2	1.6	0.2	setosa
31	4.8	3.1	1.6	0.2	setosa
32	5.4	3.4	1.5	0.4	setosa
33	5.2	4.1	1.5	0.1	setosa
34	5.5	4.2	1.4	0.2	setosa
35	4.9	3.1	1.5	0.2	setosa
36	5.0	3.2	1.2	0.2	setosa
37	5.5	3.5	1.3	0.2	setosa
38	4.9	3.6	1.4	0.1	setosa
39	4.4	3.0	1.3	0.2	setosa
40	5.1	3.4	1.5	0.2	setosa
41	5.0	3.5	1.3	0.3	setosa
42	4.5	2.3	1.3	0.3	setosa

43	4.4	3.2	1.3	0.2	setosa
44	5.0	3.5	1.6	0.6	setosa
45	5.1	3.8	1.9	0.4	setosa
46	4.8	3.0	1.4	0.3	setosa
47	5.1	3.8	1.6	0.2	setosa
48	4.6	3.2	1.4	0.2	setosa
49	5.3	3.7	1.5	0.2	setosa
50	5.0	3.3	1.4	0.2	setosa

\$versicolor

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
51	7.0	3.2	4.7	1.4	versicolor
52	6.4	3.2	4.5	1.5	versicolor
53	6.9	3.1	4.9	1.5	versicolor
54	5.5	2.3	4.0	1.3	versicolor
55	6.5	2.8	4.6	1.5	versicolor
56	5.7	2.8	4.5	1.3	versicolor
57	6.3	3.3	4.7	1.6	versicolor
58	4.9	2.4	3.3	1.0	versicolor
59	6.6	2.9	4.6	1.3	versicolor
60	5.2	2.7	3.9	1.4	versicolor
61	5.0	2.0	3.5	1.0	versicolor
62	5.9	3.0	4.2	1.5	versicolor
63	6.0	2.2	4.0	1.0	versicolor
64	6.1	2.9	4.7	1.4	versicolor
65	5.6	2.9	3.6	1.3	versicolor
66	6.7	3.1	4.4	1.4	versicolor
67	5.6	3.0	4.5	1.5	versicolor
68	5.8	2.7	4.1	1.0	versicolor
69	6.2	2.2	4.5	1.5	versicolor
70	5.6	2.5	3.9	1.1	versicolor
71	5.9	3.2	4.8	1.8	versicolor
72	6.1	2.8	4.0	1.3	versicolor

73	6.3	2.5	4.9	1.5 versicolor
74	6.1	2.8	4.7	1.2 versicolor
75	6.4	2.9	4.3	1.3 versicolor
76	6.6	3.0	4.4	1.4 versicolor
77	6.8	2.8	4.8	1.4 versicolor
78	6.7	3.0	5.0	1.7 versicolor
79	6.0	2.9	4.5	1.5 versicolor
80	5.7	2.6	3.5	1.0 versicolor
81	5.5	2.4	3.8	1.1 versicolor
82	5.5	2.4	3.7	1.0 versicolor
83	5.8	2.7	3.9	1.2 versicolor
84	6.0	2.7	5.1	1.6 versicolor
85	5.4	3.0	4.5	1.5 versicolor
86	6.0	3.4	4.5	1.6 versicolor
87	6.7	3.1	4.7	1.5 versicolor
88	6.3	2.3	4.4	1.3 versicolor
89	5.6	3.0	4.1	1.3 versicolor
90	5.5	2.5	4.0	1.3 versicolor
91	5.5	2.6	4.4	1.2 versicolor
92	6.1	3.0	4.6	1.4 versicolor
93	5.8	2.6	4.0	1.2 versicolor
94	5.0	2.3	3.3	1.0 versicolor
95	5.6	2.7	4.2	1.3 versicolor
96	5.7	3.0	4.2	1.2 versicolor
97	5.7	2.9	4.2	1.3 versicolor
98	6.2	2.9	4.3	1.3 versicolor
99	5.1	2.5	3.0	1.1 versicolor
100	5.7	2.8	4.1	1.3 versicolor

\$virginica

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
101	6.3	3.3	6.0	2.5	virginica
102	5.8	2.7	5.1	1.9	virginica

103	7.1	3.0	5.9	2.1 virginica
104	6.3	2.9	5.6	1.8 virginica
105	6.5	3.0	5.8	2.2 virginica
106	7.6	3.0	6.6	2.1 virginica
107	4.9	2.5	4.5	1.7 virginica
108	7.3	2.9	6.3	1.8 virginica
109	6.7	2.5	5.8	1.8 virginica
110	7.2	3.6	6.1	2.5 virginica
111	6.5	3.2	5.1	2.0 virginica
112	6.4	2.7	5.3	1.9 virginica
113	6.8	3.0	5.5	2.1 virginica
114	5.7	2.5	5.0	2.0 virginica
115	5.8	2.8	5.1	2.4 virginica
116	6.4	3.2	5.3	2.3 virginica
117	6.5	3.0	5.5	1.8 virginica
118	7.7	3.8	6.7	2.2 virginica
119	7.7	2.6	6.9	2.3 virginica
120	6.0	2.2	5.0	1.5 virginica
121	6.9	3.2	5.7	2.3 virginica
122	5.6	2.8	4.9	2.0 virginica
123	7.7	2.8	6.7	2.0 virginica
124	6.3	2.7	4.9	1.8 virginica
125	6.7	3.3	5.7	2.1 virginica
126	7.2	3.2	6.0	1.8 virginica
127	6.2	2.8	4.8	1.8 virginica
128	6.1	3.0	4.9	1.8 virginica
129	6.4	2.8	5.6	2.1 virginica
130	7.2	3.0	5.8	1.6 virginica
131	7.4	2.8	6.1	1.9 virginica
132	7.9	3.8	6.4	2.0 virginica
133	6.4	2.8	5.6	2.2 virginica
134	6.3	2.8	5.1	1.5 virginica
135	6.1	2.6	5.6	1.4 virginica

136	7.7	3.0	6.1	2.3 virginica
137	6.3	3.4	5.6	2.4 virginica
138	6.4	3.1	5.5	1.8 virginica
139	6.0	3.0	4.8	1.8 virginica
140	6.9	3.1	5.4	2.1 virginica
141	6.7	3.1	5.6	2.4 virginica
142	6.9	3.1	5.1	2.3 virginica
143	5.8	2.7	5.1	1.9 virginica
144	6.8	3.2	5.9	2.3 virginica
145	6.7	3.3	5.7	2.5 virginica
146	6.7	3.0	5.2	2.3 virginica
147	6.3	2.5	5.0	1.9 virginica
148	6.5	3.0	5.2	2.0 virginica
149	6.2	3.4	5.4	2.3 virginica
150	5.9	3.0	5.1	1.8 virginica

```
lapply(iris_split,nrow)
```

```
$setosa
```

```
[1] 50
```

```
$versicolor
```

```
[1] 50
```

```
$virginica
```

```
[1] 50
```

```
str(iris)
```

```
'data.frame': 150 obs. of 5 variables:
```

```
$ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
```

```
$ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
```

```
$ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
```

```
$ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
```

```
$ Species      : Factor w/ 3 levels "setosa","versicolor",...: 1 1 1 1 1 1 1 1 1 1 1 ...
```

```
tapply(iris$Sepal.Length,list(iris$Species),mean)
```

```
setosa versicolor virginica
5.006      5.936      6.588
```

```
tapply(iris$Petal.Length,list(iris$Species),mean)
```

```
setosa versicolor virginica
1.462      4.260      5.552
```

题目五:

```
by(iris$Petal.Length,iris$Species,mean)
```

```
iris$Species: setosa
[1] 1.462
```

```
-----
iris$Species: versicolor
[1] 4.26
```

```
-----
iris$Species: virginica
[1] 5.552
```

```
by(iris$Sepal.Length,iris$Species,mean)
```

```
iris$Species: setosa
[1] 5.006
```

```
-----
iris$Species: versicolor
[1] 5.936
```



```
iris$Species: virginica
[1] 6.588
```

```
tapply(iris$Petal.Length,list(iris$Species),median)
```

```
setosa versicolor virginica
1.50      4.35      5.55
```

```
tapply(iris$Sepal.Length,list(iris$Species),median)
```

```
setosa versicolor virginica
5.0      5.9      6.5
```

```
aggregate(iris[,-5],list(iris[[5]]),mean)
```

	Group.1	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
1	setosa	5.006	3.428	1.462	0.246
2	versicolor	5.936	2.770	4.260	1.326
3	virginica	6.588	2.974	5.552	2.026

题目六:

```
set.seed(123)
ages <- sample(18:65, 100, replace = TRUE)
income_levels <- sample(c("Low", "Medium", "High"), 100, replace = TRUE)
data <- data.frame(age = ages, income.level = income_levels)

age.group = cut(data$age,
                 c(18,35,55,70),
                 labels = c("Young", "Middle", "Old"))
table(age.group)
```

```
age.group
  Young Middle   Old
    35    48    17
```

```
table(list(age.group,data$income.level))
```

```
      .2
.1      High Low Medium
  Young    13   8    14
  Middle    15  16    17
   Old      5   5     7
```