R软件在CGED-Q JSL中的运用

（二）字符串处理

作者：陈俊（硕士研究生，华中师范大学）；康文林（教授，香港科技大学；华中师范大学）

01

创建新变量

PART ONE

02

逻辑表达式

PART TWO

03

变量转换
(数值变量和字符变量的互换)

PART THREE

04

字符串表达式

PART FOUR

01

创建新变量

运算符

+	加
-	减
*	乘
/	除
^ 或者**	求幂
x%%y	求余
x%/%y	整数除法

<	小于
<=	小于/等于
>	大于
>=	大于/等于
==	严格等于
！=	不等于
！x	不是x
x|y	x或y
isTRUE(x）	测试x是否为True

图片来源：https://blog.csdn.net/kidpea_lau/article/details/82786145

1.1删除空白名的记录

引用符号，表示引用该数据集的某一个变量

所引用的变量的名称

严格等于

判定条件，如果是字符，要加上引号，如果是数值则不用

表等于

所用数据集的名称

新变量的名称

kongbaiming <- which(JSL1900_1912$名=="空白")

（找出名字是“空白”的记录，which()函数能够返回指定值在逻辑向量中的位置，简单来说，就是利用which函数指定条件，R就能够返回满足条件的记录所在的行数。）

这是一个数据集的名字

JSL1900_1912_delete_kongbaiming <- JSL1900_1912[-kongbaiming,]

（创建一个新数据集，条件是原数据集删除掉“空白名”的记录，之后的分析都是基于新数据集来进行的。）

1.2转换变量的类型（字符转为数值）

as.numeric()函数：能将字符型变量转变为数值型变量。

格式为：data$x <- as.numeric(data$x)

实例代码：创建一个新的变量，在原变量的基础上转换变量属性

```
JSL1900_1912_delete_kongbaiming$阳历年份numeric <-
as.numeric(JSL1900_1912_delete_kongbaiming$阳历年份)
```

查看数据类型： typeof()函数能够返回变量的属性

```
typeof(JSL1900_1912_delete_kongbaiming$阳历年份numeric)
```

1.3创建新变量

data$x <- (data$y+((data$z/4)-0.25))

可以理解为这个格式，此代码只是看着长，其实是数据集名取得长，逻辑很简单。

在JSL中创建一个名为“年份季节”的新变量：

JSL1900_1912_delete_kongbaiming$年份季节 <-
(JSL1900_1912_delete_kongbaiming$阳历年份numeric+
((JSL1900_1912_delete_kongbaiming$季节号numeric/4)-
0.25))

table()函数：能够返回变量类型的个数。

table(JSL1900_1912_delete_kongbaiming$年份季节)

02

逻辑表达式

ifelse()函数

格式：

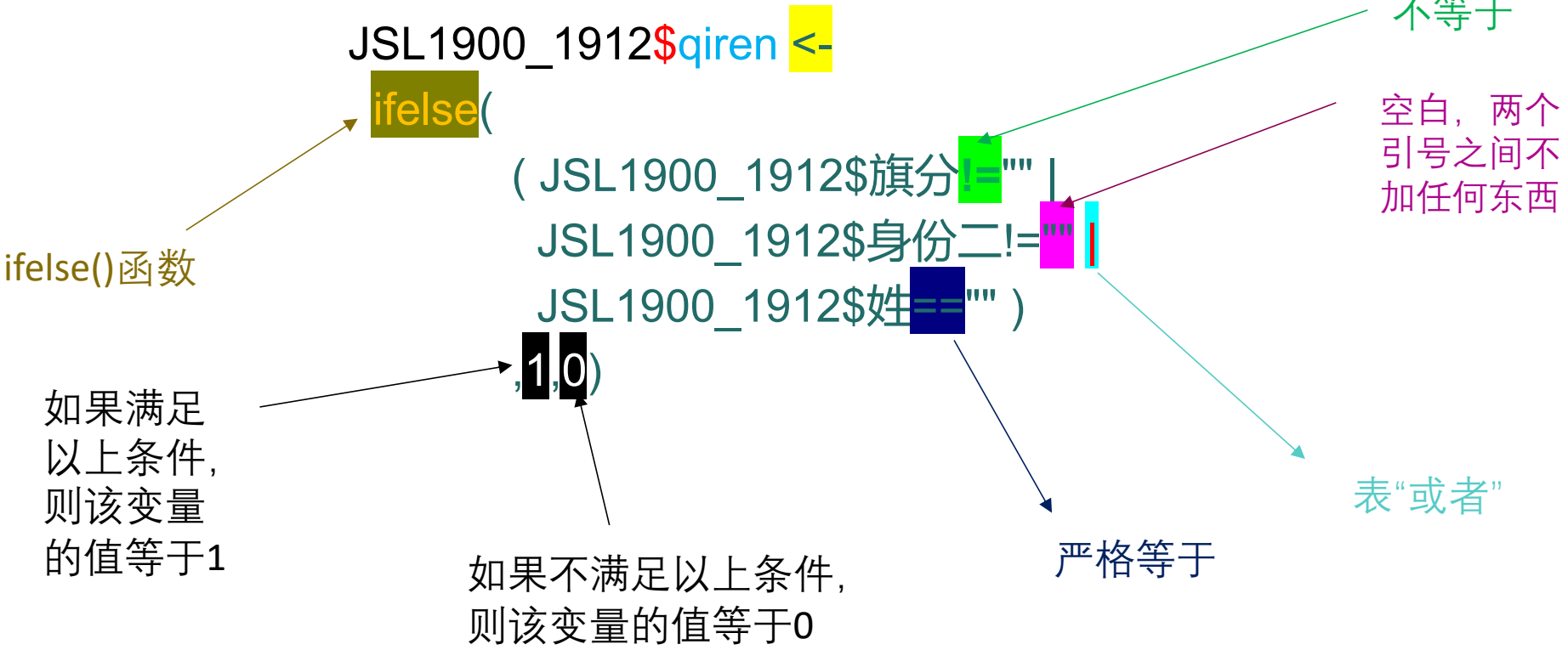
ifelse(test,yes,no)

test即为R需要执行的条件，通常是某个变量大于小于或等于某个值，简单来说就是一个判定条件。如果满足这个判定条件，R则返回yes，yes可以更改为True或者1等等。如果不满足判定条件，则返回no，同样，no可以替换为False或者0等等。

简单举例：

ifelse(年份<1900,T,F)

在数据集中用ifelse()函数生成一个带逻辑判定的新变量：



03

转换变量

3.1将数值变量转换为字符变量

如果旗人变量=1的话，就改变其值为“旗”；如果旗人变量=0的话，就改变其值为“民”：

```
JSL1900_1912_delete_kongbaiming$qiren[JSL1900_1912_delete_kongbaiming$qiren == 1] <-
 "旗“
JSL1900_1912_delete_kongbaiming$qiren[JSL1900_1912_delete_kongbaiming$qiren == 0 ] <-
 "民“
```

利用table()函数查看“旗”和“民”的数量：

```
table(JSL1900_1912_delete_kongbaiming$qiren)
```

3.2将字符变量转换为数值变量

如果“出身一”变量为“狀元”的话，就转换为1：

JSL1900_1912_delete_kongbaiming$出身一[JSL1900_1912_delete_kongbaiming$出身一 == "狀元"] <- 1

如果“出身一”变量为“榜眼”的话，就转换为2：

JSL1900_1912_delete_kongbaiming$出身一[JSL1900_1912_delete_kongbaiming$出身一 == “榜眼”] <- 2

3.3离散型变量转换为分类字符变量

可以看出，前两种方式都有很大的相似之处，代码的格式也基本是同一种类型。这是因为缙绅录的数值都是连续型变量，具有规律性。但当一个数据集是离散型变量，而且充满随机性的时候，该如何转换？

	score	backboard	assistant	game	score_per_game
Kobe	33643	7047	6306	1346	25.0
Jordan	32292	6672	5633	1072	30.1
James	34087	9352	9298	1258	27.1
Garnett	26071	14662	5445	1462	17.8
Duncan	26496	15091	4225	1392	19.0
Kidd	17529	8725	12091	1391	12.6
Anthony	26314	7251	3244	1114	23.6
Pippen	18940	7494	6135	1178	16.1
Curry	16419	3158	4621	699	23.5

随机抽取的NBA球员的得分、篮板、助攻、场均得分数据。
数据来源：http://www.stat-nba.com/

根据上一个表格的数据将离散型变量转化为字符型变量

总得分大于等于3万分的，定值为3：

```
NBA$score[NBA$score >= 30000] <- 3
```

总得分大于等于2万分小于3万分的，定值为2：

```
NBA$score[NBA$score >= 20000 & NBA$score < 30000] <- 2
```

总得分大于等于1万分小于2万分的，定值为1：

```
NBA$score[NBA$score >= 10000 & NBA$score < 20000] <- 1
```

用factor函数进行因子转换：

```
NBA$score <- factor( NBA$score, levels = c(1,2,3),labels = c("一万分先生","两万分先生","三万分先生") )
```

离散型变量转变为分类字符变量，其核心思想是首先转变将数值分类，再转换为因子，最后将因子转换为分类字符变量。factor()函数作为一个转换因子的函数，在其中起到了中转站的作用，其格式为factor(x, levels, labels)，x代表需要定义的变量，levels代表值，labels代表标签。

04

字符串表达式

4.1串联字符

paste()函数的格式：

paste(“x”, ”y”, sep=,collapse=)

X代表需要串联的一个变量，y代表需要串联的另一个变量，sep代表串联完成后字符内的连接符，collapse代表字符串间的连接符，用得不多，一般用令它等于null

实例代码：

```
JSL1900_1912_delete_kongbaiming$xinming <- paste(
JSL1900_1912_delete_kongbaiming$姓
,JSL1900_1912_delete_kongbaiming$名
,sep=""
,collapse=NULL
)
```

4.2提取和判断字符

------str.extract()函数：用以提取需要但不能确定位置的函数。格式为：

str.extract(x, ”y”)

x代表变量，y代表所需要的字符。

实例代码：str_extract(JSL1900_1912_delete_kongbaiming$官职一 ,"員外郎")

-----str_detect()函数：用以判断某变量是否含有特定字符。格式为：

str_detect(x, ”y”)

x代表变量，y代表所需要的字符。

实例代码：str_detect(JSL1900_1912_delete_kongbaiming$官职一 ,"員外郎“)

提取字符函数的返回值只有需要提取的值和NA，NA不便处理；而判断字符函数返回值是ture和false，是逻辑值。因此，在寻找特定字符时，常用判断函数。

熟悉str_detect()函数之后，就可以利用该函数结合ifelse()函数进行官职查找和判定。比如上图代码，先利用str_detect()函数判断官职是否为“总督”，随后，给满足“总督”条件的记录赋值1，否则赋值0。这样，便可以对总督这一群体进行单独研究。

实例代码：

```
JSL1900_1912_delete_kongbaiming$zongdu <-
ifelse(str_detect(JSL1900_1912_delete_kongbaiming$官职一,"總督") ,1,0)
```

4.2替换字符

gsub()函数：用于替换字符，其格式为：

gsub(“y”, ”z”, x)

其中，“y”代表需要替换的字符，“z”代表替换成什么，x代表变量。以上代码的含义是，将旗分变量中的“鑲藍”字符，替换为“镶蓝”字符。

实例代码：

```
gsub("鑲藍","镶蓝",JSL1900_1912_delete_kongbaiming$旗分)
```

下次学更难的