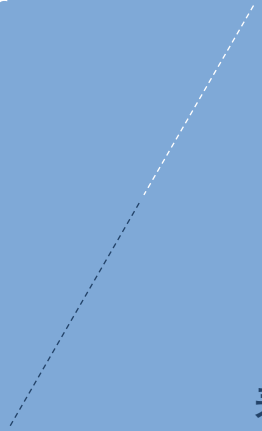
R语言在CGED-Q JSL中的运用

陈俊
香港科技大学社会科学部
2025年5月(V2.1.1)

Four



表格输出

01

简单制表

4.1.1 `table()`函数

- 简单制表函数`table()`，格式为：

```
`table(x, exclude = , useNA = ,)`
```

- `x`为变量名，可以输入两个变量，用逗号（英文状态）并列。
- `exclude`为排除某一个类型的值，比如空白、缺失值、塗黑等特定字符；如果等于NULL，就是不排除任何值；如果是NA（缺失值），就是排除NA的值；如果等于特定字符，需要加上引号再输入特定字符
- `useNA`为选择是否统计NA的频数，如果等于`no`则为不统计；如果选择`ifany`或者`always`则为统计

4.1.2 代码示例

- 用“籍贯省”变量作为行变量，前节创建的“qiren”作为列变量进行简单制表：

```
table(JSL1900_1912_clean$籍贯省, JSL1900_1912_clean$qiren, exclude= NULL, useNA= "always")
```

- 使用者可尝试复现以上代码，并利用其它变量进行简单制表

02
整理变量

4.2.1 `factor()`函数

- 大型的历史数据库是以字符串类型存储的，经常会有两种字符组合表达的意思相同，但却被归类为两种不同的类型

- 比如地名中，“京师”和“顺天”可归类为一个类型；“甘肃”和“甘肃”两者是繁体和简体的区别，应当归类为同一种类型

- 诸如此类情况，在历史数据库中比较多，我们需要利用`factor()`函数对变量进行简单地整理，`factor()`函数在之前的转换变量环节也有涉及，它的格式为：

```
`factor(x, levels = c(…) , labels = c(…)) `
```

- `x`为需要整理的变量。`levels`括号中为该变量中需要整理的字符，可以同时整理多个字符。用英文状态下引号引用，用逗号分隔。`labels`括号中为整理后的标签。根据levels整理出的变量，需要与levels中的变量一一对应，用英文状态下引号引用，逗号分隔

4.2.2 整理“籍贯省”变量

- 我们以缙绅录数据库中的“籍贯省”变量为例
- “籍贯省”变量中由于记载版本的差异，出现很多异体字、繁体字混用的情况，我们将“籍贯省”统一整理为简体字
- 同时，“籍贯省”变量中，许多类型对应的记录数较少，我们需要化零为整，方便后续研究示例代码如下

```
# 查看原始变量的状况
table(JSL1900_1912_clean$籍贯省)

# 生成一个新的变量，利用`factor()`函数对不同类型进行一一对应地整理
JSL1900_1912_clean$jiguansheng_sort <- factor(JSL1900_1912_clean$籍贯省, levels = c("","？ ？ ","？ 西","奥國","比國","丹國","德國","俄國","法國","韓國","和國","美國","腦威國","葡國","日本國","瑞典國","義國","英國","安徽","奉天","福","福建","甘肅","广西","廣","廣？ ","廣東","廣東駐防","廣西","貴州","漢軍","河南","黑龍江","湖北","湖南","吉林","江","江南","江蘇","江西","滿洲","蒙古","南","山东","山東","山西","陝西","顺天","順天","順天府","四川","天津","西","新疆","雲","雲南","浙江","直隸"), labels = c("","不详","不详","外国","外国","外国","外国","外国","外国","外国","外国","外国","外国","外国","外国","外国","外国","安徽","奉天","福建","福建","甘肅","广西","不详","不详","广东","广东","广西","贵州","不详","河南","黑龙江","湖北","湖南","吉林","不详","江南","江苏","江西","不详","不详","不详","山东","山东","山西","陕西","顺天", "顺天","顺天","四川","天津","不详","新疆","云南","云南","浙江","直隶"))

# 查看变量整理后的状况
table(JSL1900_1912_clean$jiguansheng_sort, JSL1900_1912_clean$qiren, exclude= NULL, useNA= "always")
```

代码解析

- 这是一种比较基础的方法，需要手动识别，一一对应，适用于类型比较少的变量
- 比如“籍贯省”，整理出来差不多30余种类型；相反，“出身”这种变量，`table()`函数运行之后区域三会出现数百种类型，就不太适合`factor()`函数整理
- 我们后续会介绍变量整理的简便方法（数据库整理），也会用到之前建议大家下载的数据库整理的便捷代码。
- 变量整理完成后，区域三的表格效果将会变得更加简洁

4.2.3 整理“贡生”变量

- 我们利用同样方法整理“贡生”，“贡生”出现在“出身一”变量中，也就是说，我们仅仅对“出身一”中的一种种类进行整理

- 《清史稿 选举五》记载：贡生出身具体有恩贡、岁贡、优贡、拔贡、副贡以及廪贡、增贡、附贡等，贡生以恩、拔、副、岁、优为正途，其余皆为异途

- 根据《清史稿》，我们编写代码区分正途和异途贡生

```
# 查看原始变量“出身一”中各类贡生的状况
table(JSL1900_1912_clean$出身一, JSL1900_1912_clean$qiren)

# 生成一个新的变量，利用`factor()`函数对不同类型进行一一对应地整理。
JSL1900_1912_clean$gongsheng <- factor(JSL1900_1912_clean$出身一, levels = c("挨貢","拔貢","拔貢生","撥貢","恩？","恩貢","附贡","附貢生","副贡","副貢","副貢生","廩贡","廩貢","廩貢生","崴貢","歲貢","歲貢生","優貢","優貢生","增贡","增貢","增貢生"), labels = c("正途贡生","正途贡生","正途贡生","正途贡生","正途贡生","正途贡生","异途贡生","异途贡生","正途贡生","正途贡生","正途贡生","异途贡生","异途贡生","异途贡生","正途贡生","正途贡生","正途贡生","正途贡生","正途贡生","异途贡生","异途贡生","异途贡生"))

# 查看变量整理后的状况
table(JSL1900_1912_clean$gongsheng)
```

4.2.4 整理“鼎甲”变量

- 整理“鼎甲”，“鼎甲”是清代科举名次前三者的统称。实例代码如下：

```
JSL1900_1912_clean$dingjia <- factor(JSL1900_1912_clean$出身一, levels = c("狀元","榜眼","探花
"),labels = c("鼎甲","鼎甲","鼎甲"))
```

4.2.5 利用整理好的变量制表

- 确定“贡生”和“鼎甲”后，我们可以利用制表函数考察鼎甲群体的民族结构：

```
# 展示“鼎甲”群体中的满汉比例
table(JSL1900_1912_clean$dingjia, JSL1900_1912_clean$qiren, exclude= NULL, useNA= "always")
```

- 区域三输出结果：

```
          旗    民  <NA>
鼎甲     32   230     0
<NA> 154841 484329     0
```

4.2.6 利用整理好的变量制表

- 利用制表函数考察贡生群体的民族结构:

```
# 展示出身为正途贡生和异途贡生官员的满汉比例
table(JSL1900_1912_clean$gongsheng, JSL1900_1912_clean$qiren, exclude= NULL, useNA= "always")
```

```
             旗    民  <NA>
正途贡生    956 35444     0
异途贡生    326 33602     0
<NA>     153591 415513     0
```

4.2.7 利用整理好的变量制表

- 利用制表函数考察贡生群体的地理来源：

```
# 展示出身为正途贡生和异途贡生官员的地理来源
table(JSL1900_1912_clean$jiguansheng_sort, JSL1900_1912_clean$gongsheng, exclude= NULL, useNA= "always")
```

```
      正途贡生 异途贡生  <NA>
 安徽      999     497  24821
 奉天      283      34   3729
 福建      464     246  11653
 甘肃      208      49   3059
 广西      327     166   6352
 广东      466     299  15155
 贵州      526     252   7236
 河南      716     446  14184
# 节选
```

- 变量整理是制表一个基础的环节，使用者还可以仿照代码展示不同群体之间的联系，比如出身“贡生”官员群体取得“鼎甲”功名的具体数量；“鼎甲”功名获得者的地理分布等等

03

利用`flextable`包导出表格

4.3.1 利用`flextable`包导出二联表

- 在撰写论文的过程中，我们常常需要列出比较专业、简洁的表
- 前面介绍的`table()`函数会在区域三返回观测值，但需要复制粘贴，并重新制作表格样式，比较麻烦
- 使用`flextable`包可以导出专业的表格，以下介绍了如何利用缙绅录数据库和`flextable`包导出二联表格

4.3.2 制表前的数据清理

- 制图之前，需要用`label()`函数给前面创造的几个变量贴标签，确保表格导出时更美观

- 注意使用`label()`函数前请安装或加载`dplyr`包。同时将`zhixian`变量的判定值定义为“知县”和“非知县”，以准备数据。

```
安装和读取相应的包
install.packages("table1")
library(table1)

# 贴标签
label(JSL1900_1912_clean$jiguansheng_sort) <-"籍贯省"
label(JSL1900_1912_clean$qiren) <- "旗人"
label(JSL1900_1912_clean$gongsheng) <- "贡生"
label(JSL1900_1912_clean$zhixian) <- "知县"
JSL1900_1912_clean$zhixian[JSL1900_1912_clean$zhixian == 1] <- "知县"
JSL1900_1912_clean$zhixian[JSL1900_1912_clean$zhixian == 0] <- "非知县"
```

4.3.3 将`table()`函数制作的表格转换为矩阵数据框

- 此外，还需要将所需数据转换为数据框，`as.data.frame()`函数可以将`table()`函数制作的频数统计表格转换为数据框，列和行均会成为改造后数据框的变量

- 如果直接使用`as.data.frame()`函数，可能会得到一个不符合预期的结果，因此使用`as.data.frame.matrix()`函数，`as.data.frame.matrix()`函数能够将`table()`的频数统计表格直接转换为矩阵数据框

```
library(tibble)
Two_way_table_ft <- as.data.frame.matrix(
  table(JSL1900_1912_clean$jiguansheng_sort,JSL1900_1912_clean$qiren)) %>%
  rownames_to_column(var = "省籍")
# `rownames_to_column()`函数可以插入行名作为数据框的第一列，
# 确保`fletable`包生成表格时显示行名
```

4.3.4 利用`flextable`包将生成的数据框转换为专业的数据表格

```
ft_1 <- flextable(Two_way_table_ft)
ft_1 <- flextable(Two_way_table_ft) %>%
 add_header_row( values = "缙绅录数据库中官员的籍贯和满汉比例",
  # 将表头设置为标题，确保表格导出时标题可以被保留
  colwidths = ncol_keys(ft_1), top = TRUE) %>%  # 添加到表格顶部
 bold(i = 1, part = "header") %>%
 # 第一行表头加粗
 fontsize(i = 1, size = 14, part = "header") %>%
 # 表头增大字号
 bold(part = "header")  %>%
 # 表头加粗
 border(i = 1, part = "header", border.top = fp_border(width = 0, color = "transparent")) %>%
 # 移除第一行表头的上边框
 add_footer_lines("数据来源：CGED-Q JSL Public Release 1900-1912") %>%
 # 注明数据来源
 autofit() %>%  # 自动调整列宽
 align(align = "center", part = "all") %>%  # 全部居中
 font(fontname = "Songti SC", part = "all")  # 如果是MAC OS系统，使用中文宋体

ft_1 # 显示图片
```

缙绅录数据库中官员的籍贯和满汉比例

省籍	旗	民
	153,810	122,350
不详	2	23
外国	689	4
安徽	52	26,265
奉天	0	4,046
福建	11	12,352
甘肃	10	3,306
广西	5	6,840
广东	33	15,887
贵州	1	8,013
河南	32	15,314
黑龙江	0	22
湖北	29	16,948
湖南	5	25,417
吉林	0	302
江南	0	104
江苏	13	39,647
江西	9	18,299
山东	3	19,818
山西	3	7,728
陕西	17	7,564
顺天	59	31,877
四川	45	21,938
天津	0	5
新疆	0	4
云南	16	6,541
浙江	21	51,726
直隶	8	22,219

数据来源：CGED-Q JSL Public Release 1900-1912

4.3.5 导出表格

```
# 导出为html格式
save_as_html(ft_1, path = "/Users/jc/Documents/R export/table/缙绅录数据库中官员的籍贯和满汉比例.html")
# 设置为你自己的路径

# 导出为图片
save_as_image(ft_1, path = "/Users/jc/Documents/R export/image/缙绅录数据库中官员的籍贯和满汉比例.png", zoom = 4)
# 设置为你自己的路径

# 导出为docx文件
save_as_docx(ft_1, path = "/Users/jc/Documents/R export/office/缙绅录数据库中官员的籍贯和满汉比例.docx")
# 设置为你自己的路径

# 导出为pptx文件
save_as_pptx(ft_1, path = "/Users/jc/Documents/R export/office/缙绅录数据库中官员的籍贯和满汉比例.pptx")
# 设置为你自己的路径
```

4.3.6 利用`flextable`包导出三联表

- 以上是二联表格的制作和导出过程，但在缙绅录数据库的分析当中，我们有可能对多个变量进行操作

- 多变量表格的表现形式之一便是三联嵌套表，三联嵌套表通常以一个变量作为行变量，两个变量作为列变量，且列变量一嵌套于列变量二之上

- 而这其中，嵌套关系的处理最为关键。`flextable`包主要的功能是对表格进行优化，并不带有统计功能，所以我们在利用`flextable`包制作三联嵌套表时，需要对列变量的嵌套关系进行计算。在制作二联表时，我们运用了`jiguansheng_sort`和`qiren`两个变量，现在我们再引入一个`zhixian`变量制作三联表

- 首先需要准备数据，将`jiguansheng_sort`作为行名，`qiren`和`zhixian`作为列变量，同时使`zhixian`嵌套于`qiren`之上，由于`qiren`和`zhixian`都是0，1值，所以嵌套格式下一共有4种组合，总体制表逻辑是先计算四种组合对应的统计数，再用`flextable`包制成三联表，示例如下：

4.3.7 转换数据框并计算嵌套组合

```
Three_way_table_ft <- JSL1900_1912_clean %>%
 group_by(jiguansheng_sort) %>% # 按`jiguansheng_sort`对数据进行分组
 summarise(
  n_非知县_民 = sum(qiren == "民" & zhixian == "非知县"),
  n_非知县_旗 = sum(qiren == "旗" & zhixian == "非知县"),
  n_知县_民 = sum(qiren == "民" & zhixian == "知县"),
  n_知县_旗 = sum(qiren == "旗" & zhixian == "知县"),
  total = n()  # 计算四种组合对应的频数，并添加一列计算各行的频数之和
 ) %>%
 mutate( # 现在将百分比引入表格，使单元格的格式变为数量（百分比），
      # 百分比的计算公式为单元格统计数/总样本量
  `非知县_民` = sprintf("%d (%.1f%%)",  n_非知县_民, n_非知县_民 / 639432 * 100),
  `非知县_旗` = sprintf("%d (%.1f%%)",  n_非知县_旗, n_非知县_旗 / 639432 * 100),
  `知县_民` = sprintf("%d (%.1f%%)", n_知县_民, n_知县_民 / 639432 * 100),
  `知县_旗` = sprintf("%d (%.1f%%)", n_知县_旗, n_知县_旗 / 639432 * 100)
 ) %>%
 bind_rows( # 添加一个总计行，展示各列的频数之和
  summarise(.,
          jiguansheng_sort = "总计",
          `非知县_民` = as.character(sum(n_非知县_民)),
          `非知县_旗` = as.character(sum(n_非知县_旗)),
          `知县_民` = as.character(sum(n_知县_民)),
          `知县_旗` = as.character(sum(n_知县_旗)),
          total = sum(total))) %>%
 select( jiguansheng_sort, `非知县_民`, `非知县_旗`, `知县_民`, `知县_旗`, total ) # 精简表格
```

4.3.8 使用`flextable`包制成三联表

```
ft_2 <- flextable(Three_way_table_ft)
ft_2 <- flextable(Three_way_table_ft) %>% set_header_labels( # 设置各列的标签
   jiguansheng_sort = “省籍”, `非知县_民` = “民人”, `非知县_旗` = “旗人”, `知县_民` = “民人”, `知县_旗
` = “旗人”, total = "样本量" ) %>%
 add_header_row( # 加入一行作为嵌套上层行，并设置名称，单元格宽度
  values = c(“”, “非知县”, “知县”, “”), colwidths = c(1, 2, 2, 1) ) %>%
 add_header_row( # 在表头加入一行作为标题行，并设置标题
  values = “知县的地理来源与满汉比例”, colwidths = ncol_keys(ft_2), top = TRUE) %>%
 bold(i = 1, part = "header") %>% # 第一行表头加粗
 bold(part = "header")  %>% # 表头加粗
 fontsize(i = 1, size = 14, part = "header") %>%
 border(i = 1, part = "header", # 设置表头上框线宽度为0并设置为透明
      border.top = fp_border(width = 0, color = "transparent")) %>%
 add_footer_lines("注：1）省籍第一行的空白代表无对应省籍记载
            2）单元格百分比计算公式为单元格内统计数/总样本量
            3）本表统计的是缙绅录数据库的记载数
            4）数据来源：CGED-Q JSL Public Release 1900-1912") %>%
 autofit() %>% # 自动调整列宽
 align(align = "center", part = "all") %>% # 全部居中
 font(fontname = "Songti SC", part = "all") # 如果是MAC OS系统，使用中文宋体
ft_2 # 显示表格
```

知县的地理来源与满汉比例

	非知县		知县		
省籍	民人	旗人	民人	旗人	样本量
	122125 (19.1%)	150437 (23.5%)	225 (0.0%)	3373 (0.5%)	276,160
不详	21 (0.0%)	2 (0.0%)	2 (0.0%)	0 (0.0%)	25
外国	4 (0.0%)	689 (0.1%)	0 (0.0%)	0 (0.0%)	693
安徽	22396 (3.5%)	52 (0.0%)	3869 (0.6%)	0 (0.0%)	26,317
奉天	3478 (0.5%)	0 (0.0%)	568 (0.1%)	0 (0.0%)	4,046
福建	9767 (1.5%)	11 (0.0%)	2585 (0.4%)	0 (0.0%)	12,363
甘肃	2364 (0.4%)	7 (0.0%)	942 (0.1%)	3 (0.0%)	3,316
广西	4901 (0.8%)	5 (0.0%)	1939 (0.3%)	0 (0.0%)	6,845
广东	14444 (2.3%)	33 (0.0%)	1443 (0.2%)	0 (0.0%)	15,920
贵州	6104 (1.0%)	1 (0.0%)	1909 (0.3%)	0 (0.0%)	8,014
河南	12718 (2.0%)	31 (0.0%)	2596 (0.4%)	1 (0.0%)	15,346
黑龙江	22 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	22
湖北	13835 (2.2%)	28 (0.0%)	3113 (0.5%)	1 (0.0%)	16,977
湖南	21828 (3.4%)	5 (0.0%)	3589 (0.6%)	0 (0.0%)	25,422
吉林	254 (0.0%)	0 (0.0%)	48 (0.0%)	0 (0.0%)	302
江南	103 (0.0%)	0 (0.0%)	1 (0.0%)	0 (0.0%)	104
江苏	34688 (5.4%)	13 (0.0%)	4959 (0.8%)	0 (0.0%)	39,660
江西	15161 (2.4%)	9 (0.0%)	3138 (0.5%)	0 (0.0%)	18,308
山东	17317 (2.7%)	3 (0.0%)	2501 (0.4%)	0 (0.0%)	19,821
山西	6455 (1.0%)	1 (0.0%)	1273 (0.2%)	2 (0.0%)	7,731
陕西	6172 (1.0%)	17 (0.0%)	1392 (0.2%)	0 (0.0%)	7,581
顺天	29746 (4.7%)	57 (0.0%)	2131 (0.3%)	2 (0.0%)	31,936
四川	18691 (2.9%)	44 (0.0%)	3247 (0.5%)	1 (0.0%)	21,983
天津	5 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	5
新疆	4 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	4
云南	5008 (0.8%)	16 (0.0%)	1533 (0.2%)	0 (0.0%)	6,557
浙江	45808 (7.2%)	17 (0.0%)	5918 (0.9%)	4 (0.0%)	51,747
直隶	19734 (3.1%)	8 (0.0%)	2485 (0.4%)	0 (0.0%)	22,227
总计	433153	151486	51406	3387	639,432

注：1）省籍第一行的空白代表无对应省籍记载 2）单元格百分比计算公式为单元格内统计数/总样本量 设置表头上框线宽度为0并设置为透明 3）本表统计的是缙绅录数据库的记载数 4）数据来源：CGED-Q JSL Public Release 1900-1912

4.3.9 导出表格

```
# 导出为图片
save_as_image(
  ft_2,
  path = "/Users/jc/Documents/R export/image/知县的地理来源和满汉比例.png",
  zoom = 4)
# 设置为你自己的路径
```

注意事项

- 三联表的制作相对复杂，因为要涉及到对两个变量的嵌套组合，变量的类型越多，嵌套的组合也就越多，计算量也就越大
- 但是在缙绅录数据库这一类以字符为主的数据库中，几乎没有离散数据，更多的是逻辑判定值，因此，其总的计算量其实并不大
- 但数据统计的环节尤其重要，务必要保证数据的精确性

04

利用`table1`包导出表格

4.4 利用`table1`包导出二联表

4.4.1 `table1`函数

- 除了`flextable`包以外，`table1()`也可以制作出适应学术规范的表格，并展示在区域四，以供导出。`table1()`的格式为：

```
`table1(x, data, overall = , rowlabelhead = ,) `
```

- `x`为需要制表的变量，如果是多个变量，用`+`号连接。`|`号用来分层，即`|`号之后通常是列变量。*号用于嵌套，嵌套两个逻辑值。比较复杂，后面可结合代码理解

- `data`为所引用的数据集

- `overall`为总计。需要在等号后添加内容,比如`overall`、`total`等。如果令其等于F，则不显示总计的数值

- `rowlabelhead`是抬头行标题，对应第一列变量，可以等于任意字符，需要添加引号

4.4.2 利用`table1`包导出二联表

- 尝试利用该函数制作可以导出的官员地理来源和满汉比例二联表格，实例代码如下：

```
library(table1)
table1(~jiguansheng_sort|qiren, data = JSL1900_1912_clean, overall = "总计")
# 注意，前面的`~`符号不可省略，`|`用来分层，即`籍贯省`为行，`qiren`为列，两者可以颠倒顺序
```

	旗 (N=154873)	民 (N=484559)	总计 (N=639432)
籍贯省			
	153810 (99.3%)	122350 (25.2%)	276160 (43.2%)
不详	2 (0.0%)	23 (0.0%)	25 (0.0%)
外国	689 (0.4%)	4 (0.0%)	693 (0.1%)
安徽	52 (0.0%)	26265 (5.4%)	26317 (4.1%)
奉天	0 (0%)	4046 (0.8%)	4046 (0.6%)
福建	11 (0.0%)	12352 (2.5%)	12363 (1.9%)
甘肃	10 (0.0%)	3306 (0.7%)	3316 (0.5%)
广西	5 (0.0%)	6840 (1.4%)	6845 (1.1%)
广东	33 (0.0%)	15887 (3.3%)	15920 (2.5%)
贵州	1 (0.0%)	8013 (1.7%)	8014 (1.3%)
河南	32 (0.0%)	15314 (3.2%)	15346 (2.4%)
黑龙江	0 (0%)	22 (0.0%)	22 (0.0%)
湖北	29 (0.0%)	16948 (3.5%)	16977 (2.7%)
湖南	5 (0.0%)	25417 (5.2%)	25422 (4.0%)
吉林	0 (0%)	302 (0.1%)	302 (0.0%)
江南	0 (0%)	104 (0.0%)	104 (0.0%)
江苏	13 (0.0%)	39647 (8.2%)	39660 (6.2%)
江西	9 (0.0%)	18299 (3.8%)	18308 (2.9%)
山东	3 (0.0%)	19818 (4.1%)	19821 (3.1%)
山西	3 (0.0%)	7728 (1.6%)	7731 (1.2%)
陕西	17 (0.0%)	7564 (1.6%)	7581 (1.2%)
顺天	59 (0.0%)	31877 (6.6%)	31936 (5.0%)
四川	45 (0.0%)	21938 (4.5%)	21983 (3.4%)
天津	0 (0%)	5 (0.0%)	5 (0.0%)
新疆	0 (0%)	4 (0.0%)	4 (0.0%)
云南	16 (0.0%)	6541 (1.3%)	6557 (1.0%)
浙江	21 (0.0%)	51726 (10.7%)	51747 (8.1%)
直隶	8 (0.0%)	22219 (4.6%)	22227 (3.5%)

4.4.3 利用`table1`包导出二联表

- 尝试利用该函数制作可以导出的官员地理来源和满汉比例二联表格，同时在行变量上添加一个新变量，实例代码如下：

```
library(table1)
table1(~jiguansheng_sort+gongsheng|qiren, data = JSL1900_1912_clean, overall = "总计")
# 使用`+`号在行添加一个变量`gongsheng`，使`jiguansheng_sort`和`gongsheng`同时为行，`qiren`为列
```

	旗 (N=154873)	民 (N=484559)	总计 (N=639432)
籍贯省			
	153810 (99.3%)	122350 (25.2%)	276160 (43.2%)
不详	2 (0.0%)	23 (0.0%)	25 (0.0%)
外国	689 (0.4%)	4 (0.0%)	693 (0.1%)
安徽	52 (0.0%)	26265 (5.4%)	26317 (4.1%)
奉天	0 (0%)	4046 (0.8%)	4046 (0.6%)
福建	11 (0.0%)	12352 (2.5%)	12363 (1.9%)
甘肃	10 (0.0%)	3306 (0.7%)	3316 (0.5%)
广西	5 (0.0%)	6840 (1.4%)	6845 (1.1%)
广东	33 (0.0%)	15887 (3.3%)	15920 (2.5%)
贵州	1 (0.0%)	8013 (1.7%)	8014 (1.3%)
河南	32 (0.0%)	15314 (3.2%)	15346 (2.4%)
黑龙江	0 (0%)	22 (0.0%)	22 (0.0%)
湖北	29 (0.0%)	16948 (3.5%)	16977 (2.7%)
湖南	5 (0.0%)	25417 (5.2%)	25422 (4.0%)
吉林	0 (0%)	302 (0.1%)	302 (0.0%)
江南	0 (0%)	104 (0.0%)	104 (0.0%)
江苏	13 (0.0%)	39647 (8.2%)	39660 (6.2%)
江西	9 (0.0%)	18299 (3.8%)	18308 (2.9%)
山东	3 (0.0%)	19818 (4.1%)	19821 (3.1%)
山西	3 (0.0%)	7728 (1.6%)	7731 (1.2%)
陕西	17 (0.0%)	7564 (1.6%)	7581 (1.2%)
顺天	59 (0.0%)	31877 (6.6%)	31936 (5.0%)
四川	45 (0.0%)	21938 (4.5%)	21983 (3.4%)
天津	0 (0%)	5 (0.0%)	5 (0.0%)
新疆	0 (0%)	4 (0.0%)	4 (0.0%)
云南	16 (0.0%)	6541 (1.3%)	6557 (1.0%)
浙江	21 (0.0%)	51726 (10.7%)	51747 (8.1%)
直隶	8 (0.0%)	22219 (4.6%)	22227 (3.5%)
贡生			
正途贡生	956 (0.6%)	35444 (7.3%)	36400 (5.7%)
异途贡生	326 (0.2%)	33602 (6.9%)	33928 (5.3%)
Missing	153591 (99.2%)	415513 (85.8%)	569104 (89.0%)

4.4.4 利用`table1`包导出三联表

- `table1()`包不仅可以在行上增加多个变量，也能够实现嵌套组合，进而制作三联表，示例代码如下：

```
library(table1)
table1(~jiguansheng_sort+gongsheng+dingjia|zhixian*qiren, data = JSL1900_1912_clean, overall = "总计")
```

- `jiguansheng_sort`、`gongsheng`和`dingjia`为行，`qiren`和`zhixian`将为列，且两者是嵌套关系，`zhixian`嵌套在`qiren`之上，如需导出，可以直接在区域四`Viewer`区域点击`Export`导出为图片或者`html`格式。

- 在`table1()`包中，`~`号不可省略，`|`号用来区分行变量和列变量，通常在`|`号之前，是表格中的行，`|`号之后是表格中的列。`+`号用来连接多个行变量；`*`号用来嵌套列变量。请注意符号的运用。

	知县		非知县		总计	
	旗 (N=3387)	民 (N=51406)	旗 (N=151486)	民 (N=433153)	旗 (N=154873)	民 (N=484559)
籍贯省						
	3373 (99.6%)	225 (0.4%)	150437 (99.3%)	122125 (28.2%)	153810 (99.3%)	122350 (25.2%)
不详	0 (0%)	2 (0.0%)	2 (0.0%)	21 (0.0%)	2 (0.0%)	23 (0.0%)
外国	0 (0%)	0 (0%)	689 (0.5%)	4 (0.0%)	689 (0.4%)	4 (0.0%)
安徽	0 (0%)	3869 (7.5%)	52 (0.0%)	22396 (5.2%)	52 (0.0%)	26265 (5.4%)
奉天	0 (0%)	568 (1.1%)	0 (0%)	3478 (0.8%)	0 (0%)	4046 (0.8%)
福建	0 (0%)	2585 (5.0%)	11 (0.0%)	9767 (2.3%)	11 (0.0%)	12352 (2.5%)
甘肃	3 (0.1%)	942 (1.8%)	7 (0.0%)	2364 (0.5%)	10 (0.0%)	3306 (0.7%)
广西	0 (0%)	1939 (3.8%)	5 (0.0%)	4901 (1.1%)	5 (0.0%)	6840 (1.4%)
广东	0 (0%)	1443 (2.8%)	33 (0.0%)	14444 (3.3%)	33 (0.0%)	15887 (3.3%)
贵州	0 (0%)	1909 (3.7%)	1 (0.0%)	6104 (1.4%)	1 (0.0%)	8013 (1.7%)
河南	1 (0.0%)	2596 (5.1%)	31 (0.0%)	12718 (2.9%)	32 (0.0%)	15314 (3.2%)
黑龙江	0 (0%)	0 (0%)	0 (0%)	22 (0.0%)	0 (0%)	22 (0.0%)
湖北	1 (0.0%)	3113 (6.1%)	28 (0.0%)	13835 (3.2%)	29 (0.0%)	16948 (3.5%)
湖南	0 (0%)	3589 (7.0%)	5 (0.0%)	21828 (5.0%)	5 (0.0%)	25417 (5.2%)
吉林	0 (0%)	48 (0.1%)	0 (0%)	254 (0.1%)	0 (0%)	302 (0.1%)
江南	0 (0%)	1 (0.0%)	0 (0%)	103 (0.0%)	0 (0%)	104 (0.0%)
江苏	0 (0%)	4959 (9.6%)	13 (0.0%)	34688 (8.0%)	13 (0.0%)	39647 (8.2%)
江西	0 (0%)	3138 (6.1%)	9 (0.0%)	15161 (3.5%)	9 (0.0%)	18299 (3.8%)
山东	0 (0%)	2501 (4.9%)	3 (0.0%)	17317 (4.0%)	3 (0.0%)	19818 (4.1%)
山西	2 (0.1%)	1273 (2.5%)	1 (0.0%)	6455 (1.5%)	3 (0.0%)	7728 (1.6%)
陕西	0 (0%)	1392 (2.7%)	17 (0.0%)	6172 (1.4%)	17 (0.0%)	7564 (1.6%)
顺天	2 (0.1%)	2131 (4.1%)	57 (0.0%)	29746 (6.9%)	59 (0.0%)	31877 (6.6%)
四川	1 (0.0%)	3247 (6.3%)	44 (0.0%)	18691 (4.3%)	45 (0.0%)	21938 (4.5%)
天津	0 (0%)	0 (0%)	0 (0%)	5 (0.0%)	0 (0%)	5 (0.0%)
新疆	0 (0%)	0 (0%)	0 (0%)	4 (0.0%)	0 (0%)	4 (0.0%)
云南	0 (0%)	1533 (3.0%)	16 (0.0%)	5008 (1.2%)	16 (0.0%)	6541 (1.3%)
浙江	4 (0.1%)	5918 (11.5%)	17 (0.0%)	45808 (10.6%)	21 (0.0%)	51726 (10.7%)
直隶	0 (0%)	2485 (4.8%)	8 (0.0%)	19734 (4.6%)	8 (0.0%)	22219 (4.6%)
贡生						
正途贡生	87 (2.6%)	2697 (5.2%)	869 (0.6%)	32747 (7.6%)	956 (0.6%)	35444 (7.3%)
异途贡生	25 (0.7%)	1797 (3.5%)	301 (0.2%)	31805 (7.3%)	326 (0.2%)	33602 (6.9%)
Missing	3275 (96.7%)	46912 (91.3%)	150316 (99.2%)	368601 (85.1%)	153591 (99.2%)	415513 (85.8%)
dingjia						
鼎甲	0 (0%)	0 (0%)	32 (0.0%)	230 (0.1%)	32 (0.0%)	230 (0.0%)
Missing	3387 (100%)	51406 (100%)	151454 (100.0%)	432923 (99.9%)	154841 (100.0%)	484329 (100.0%)

4.4.5 `flextable`包的优势和局限性

- `flextable`包和`table1()`包均能制作和导出表格，但两者又具备各自的优势和局限性，`flextable`包的优势和局限性如下：

- 1. 优势
  - 样式定制灵活：支持单元格合并、边框控制、背景色、字体样式等精细化操作
  - 多格式输出：直接导出为 Word、PDF 等，无需手动调整格式
  - 支持复杂表格结构：支持嵌套表头、跨页表格、脚注等学术论文所需的高级功能
  - 适用范围广：可与`dplyr`等数据处理包无缝集成，适用于各类制表场景

- 2. 局限性
  - 统计功能弱：需手动计算统计量，再整合到表格中
  - 学习曲线陡峭：复杂样式需熟悉各类函数的利用
  - 无内置统计检验，依赖外部包生成统计结果

4.4.6 `table1`包的优势和局限性

- `table1`包的优势和局限性如下：

- 1. 优势
  - 自动化统计描述：一键生成分层统计表，自动计算均值、标准差、频数、百分比等值，支持连续变量和分类变量
  - 分层嵌套展示：通过公式语法实现多级分组
  - 三线表规范输出：默认生成符合学术期刊要求的三线表格式，减少手动调整

- 2. 局限性
  - 样式定制有限：依赖自定义函数render修改表格布局，自定义函数的学习和运用较为复杂
  - 输出格式单一：主要支持 HTML，Word 需复制粘贴，可能丢失格式
  - 扩展性不足：无法生成非统计类表格

注意事项

- 综上所述，两个包各自具有优势和局限性，使用者可根据运用场景进行切换
- 两者并不是相互独立的关系，可以互补使用，兼顾效率和美观

05

指定条件制表

4.5 指定单一条件制表

4.5.1 `subset()`函数

- 在实际的制表中，我们有可能不需要整个数据库的数据，只需要某一部门或某一年份的数据即可。这种指定条件的数据分析需要用到`subset()`函数，格式如下：

```
`subset(x, subset, select =c(...),)`
```

- `x`为数据集名称，`subset`为逻辑条件，即从数据集中选取数据的条件，必须是逻辑表达式，满足符号要求，可以用`&`、`|`号进行多条件筛选；`select`为指定从相关的列中选择数据。

- 现在，我们尝试提取满足单一条件的数据，示例代码如下：

4.5.2 指定单一年份并制表

```
# 清理数据，防止制表时因缺失值问题报错
JSL1900_1912_clean$贡生<-"0"
JSL1900_1912_clean$贡生[JSL1900_1912_clean$gongsheng == "正途贡生"] <- "正途贡生"
JSL1900_1912_clean$贡生[JSL1900_1912_clean$gongsheng == "异途贡生"] <- "异途贡生"
JSL1900_1912_clean$贡生[JSL1900_1912_clean$贡生 == "0"] <- "非贡生"

# 提取指定年份的数据
JSL1900_1912_only1910 <- subset(JSL1900_1912_clean,阳历年份numeric == 1910)

# 用`table1()`函数在新创建的数据集基础上制表
table1(~jiguansheng_sort|qiren*贡生, data = JSL1900_1912_only1910, overall = "total", rowlabelhead =
"")
table1(~jiguansheng_sort+贡生+dingjia|qiren*季节, data = JSL1900_1912_only1910, overall = "total",
rowlabelhead = "")
```

	旗				民			
	冬 (N=3015)	夏 (N=4006)	春 (N=3991)	秋 (N=3997)	冬 (N=11601)	夏 (N=10922)	春 (N=10827)	秋 (N=11325)
籍贯省								
	3007 (99.7%)	4002 (99.9%)	3986 (99.9%)	3993 (99.9%)	2221 (19.1%)	2267 (20.8%)	2356 (21.8%)	2244 (19.8%)
不详	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
外国	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
安徽	1 (0.0%)	1 (0.0%)	1 (0.0%)	1 (0.0%)	665 (5.7%)	631 (5.8%)	613 (5.7%)	649 (5.7%)
奉天	0 (0%)	0 (0%)	0 (0%)	0 (0%)	113 (1.0%)	103 (0.9%)	101 (0.9%)	105 (0.9%)
福建	1 (0.0%)	0 (0%)	0 (0%)	0 (0%)	359 (3.1%)	305 (2.8%)	292 (2.7%)	335 (3.0%)
甘肃	0 (0%)	0 (0%)	0 (0%)	0 (0%)	91 (0.8%)	83 (0.8%)	86 (0.8%)	85 (0.8%)
广西	0 (0%)	0 (0%)	0 (0%)	0 (0%)	185 (1.6%)	170 (1.6%)	157 (1.5%)	180 (1.6%)
广东	3 (0.1%)	0 (0%)	0 (0%)	0 (0%)	422 (3.6%)	370 (3.4%)	351 (3.2%)	393 (3.5%)
贵州	0 (0%)	0 (0%)	0 (0%)	0 (0%)	188 (1.6%)	175 (1.6%)	175 (1.6%)	186 (1.6%)
河南	0 (0%)	0 (0%)	0 (0%)	0 (0%)	378 (3.3%)	338 (3.1%)	325 (3.0%)	353 (3.1%)
黑龙江	0 (0%)	0 (0%)	0 (0%)	0 (0%)	2 (0.0%)	1 (0.0%)	1 (0.0%)	1 (0.0%)
湖北	0 (0%)	0 (0%)	1 (0.0%)	0 (0%)	480 (4.1%)	428 (3.9%)	426 (3.9%)	465 (4.1%)
湖南	0 (0%)	0 (0%)	0 (0%)	0 (0%)	653 (5.6%)	620 (5.7%)	609 (5.6%)	636 (5.6%)
吉林	0 (0%)	0 (0%)	0 (0%)	0 (0%)	17 (0.1%)	16 (0.1%)	16 (0.1%)	17 (0.2%)
江南	0 (0%)	0 (0%)	0 (0%)	0 (0%)	1 (0.0%)	2 (0.0%)	2 (0.0%)	2 (0.0%)
江苏	0 (0%)	0 (0%)	0 (0%)	0 (0%)	1103 (9.5%)	978 (9.0%)	939 (8.7%)	1075 (9.5%)
江西	0 (0%)	0 (0%)	0 (0%)	0 (0%)	449 (3.9%)	418 (3.8%)	414 (3.8%)	434 (3.8%)
山东	0 (0%)	0 (0%)	0 (0%)	0 (0%)	492 (4.2%)	463 (4.2%)	462 (4.3%)	485 (4.3%)

* 节选部分

4.5.3 指定单一地区并制表

```
# 提取指定地区的数据
JSL1900_1912_only顺天 <- subset(JSL1900_1912_clean,jiguansheng_sort == "顺天")

# 制表检验提取数据结果
table1(~贡生|qiren, data = JSL1900_1912_only顺天, overall = "total", rowlabelhead = "")
```

	旗 (N=59)	民 (N=31877)	total (N=31936)
贡生			
非贡生	59 (100%)	30385 (95.3%)	30444 (95.3%)
异途贡生	0 (0%)	672 (2.1%)	672 (2.1%)
正途贡生	0 (0%)	820 (2.6%)	820 (2.6%)

4.5.4 利用`subset()`函数剔除缺失值

- `subset()`函数还可以用于剔除缺失值,可剔除指定列中的缺失值，参考格式为

```
`data<-subset(data,column_name!="NA")`
```

4.5.5 指定多个条件并制表

- 以上讲述的指定单一条件，即要么年份是1910，要么官员任职地区是顺天。如果需要指定多个条件，比如需要研究某个年份某个地区的官员群体，就需要用到`&`，示例如下：

```
# 创建一个只有1910年顺天地区记录的数据集
JSL1900_1912_顺天1910 <- subset(
  JSL1900_1912_clean, jiguansheng_sort=="顺天" & 阳历年份numeric == 1910)

table1(~机构一|qiren,
       data = JSL1900_1912_顺天1910,
       overall = "total",
       rowlabelhead = "")

# 创建一个只有1910年顺天地区现任候补官员记录的数据集
JSL1900_1912_顺天1910候补 <- subset(
  JSL1900_1912_clean,
  jiguansheng_sort=="顺天" & 阳历年份numeric == 1910 & houbu == 1)

table1(~机构一|qiren, data = JSL1900_1912_顺天1910候补, overall = "total", rowlabelhead = "") # 展示表格
```

	民 (N=85)	total (N=85)
機構一		
學部衙門	4 (4.7%)	4 (4.7%)
禮部衙門	24 (28.2%)	24 (28.2%)
軍機處	1 (1.2%)	1 (1.2%)
農工商部衙门	12 (14.1%)	12 (14.1%)
郵傳部衙门	12 (14.1%)	12 (14.1%)
陸軍部衙門	32 (37.6%)	32 (37.6%)

4.5.6 “多选一”条件

- 在实际分析中，我们还会遇到条件多选一的情况，数据集设置多个条件只要满足其中一个即可的情况，这时需要用到`|`符号。

- 创建一个只有1908年之后在西南三省任职官员记录的数据集，西南三省包括云、贵、川三省。因此，在这三省任意一个省份任职的官员都可被纳入到新数据集中，代码中，我们用`|`（或者）符号进行连接。示例如下：

```
JSL1900_1912_西南after1908 <- subset(JSL1900_1912_clean, (jiguansheng_sort == "四川" | jiguansheng_sort == "贵州" | jiguansheng_sort == "云南" ) & (阳历年份numeric > 1908 ))

table1(~籍贯省|qiren*季节, data = JSL1900_1912_西南after1908, overall = "total", rowlabelhead = "") # 展示表格
```

	旗				民				total			
	1 (N=3)	2 (N=3)	3 (N=2)	4 (N=2)	1 (N=3667)	2 (N=2477)	3 (N=2516)	4 (N=2885)	1 (N=3670)	2 (N=2480)	3 (N=2518)	4 (N=2887)
人省												
雲南	3 (100%)	3 (100%)	2 (100%)	2 (100%)	647 (17.6%)	442 (17.8%)	438 (17.4%)	512 (17.7%)	650 (17.7%)	445 (17.9%)	440 (17.5%)	514 (17.8%)
四川	0 (0%)	0 (0%)	0 (0%)	0 (0%)	2231 (60.8%)	1500 (60.6%)	1533 (60.9%)	1757 (60.9%)	2231 (60.8%)	1500 (60.5%)	1533 (60.9%)	1757 (60.9%)
貴州	0 (0%)	0 (0%)	0 (0%)	0 (0%)	789 (21.5%)	535 (21.6%)	545 (21.7%)	616 (21.4%)	789 (21.5%)	535 (21.6%)	545 (21.6%)	616 (21.3%)

注意事项

- 以上是本章制表的内容，这些都是比较基础的用法
- 如果有更高需求的使用者，可以在RStudio中键入`?flextable`或者`help("table1")`查看帮助或者学习新的参数

Thanks!