

talk04 练习与作业

目录

0.1 练习和作业说明	1
0.2 Talk04 内容回顾	1
0.3 练习与作业：用户验证	1
0.4 练习与作业 1：R session 管理	2
0.5 练习与作业 2：Factor 基础	3
0.6 练习与作业 3：用 mouse genes 数据做图	8

0.1 练习和作业说明

将相关代码填写入以 “{r}” 标志的代码框中，运行并看到正确的结果；

完成后，用工具栏里的”Knit” 按键生成 PDF 文档；

将 PDF 文档改为：姓名-学号-talk04 作业.pdf，并提交到老师指定的平台/钉群。

0.2 Talk04 内容回顾

待写 ...

0.3 练习与作业：用户验证

请运行以下命令，验证你的用户名。

如你当前用户名不能体现你的真实姓名，请改为拼音后再运行本作业！

```
Sys.info()[["user"]]
```

```
## [1] "mingyuwang"
```

```
Sys.getenv("HOME")
```

```
## [1] "C:/Users/rhong/Documents"
```

0.4 练习与作业 1: R session 管理

0.4.1 完成以下操作

- 定义一些变量（比如 x, y, z 并赋值；内容随意）
- 从外部文件装入一些数据（可自行创建一个 4 行 5 列的数据，内容随意）
- 保存 workspace 到.RData
- 列出当前工作空间内的所有变量
- 删除当前工作空间内所有变量
- 从.RData 文件恢复保存的数据
- 再次列出当前工作空间内的所有变量，以确认变量已恢复
- 随机删除两个变量
- 再次列出当前工作空间内的所有变量

```
## 代码写这里，并运行；  
x <- 1  
y <- 2  
z <- 3  
data <- read.table("data/Table1.txt", header = TRUE)  
# save.image(file = ".RData")
```

```
rm(list = ls())  
ls()
```

```
## character(0)
```

```
load(file = ".RData")  
ls()
```

```
## [1] "data" "x"      "y"      "z"
```

```
rm(list = c("x", "y"))  
ls()
```

```
## [1] "data" "z"
```

0.5 练习与作业 2: Factor 基础

0.5.1 factors 增加

- 创建一个变量:

```
x <- c("single", "married", "married", "single");
```

- 为 x 增加两个 levels, single, married;
- 以下操作能成功吗?

```
x[3] <- "widowed";
```

- 如果不, 请提供解决方案;

```
## 代码写这里，并运行；  
x <- c("single", "married", "married", "single")  
x <- factor(x, levels = c("single", "married"))  
try(x[3] <- "widowed")
```

```
## Warning in `[<-.factor`(`*tmp*`, 3, value = "widowed"): 因子层次有错，产生了NA
```

```
# 解决方案  
x <- factor(x, levels = c("single", "married", "widowed"))  
try(x[3] <- "widowed")  
x
```

```
## [1] single married widowed single  
## Levels: single married widowed
```

0.5.2 factors 改变

- 创建一个变量：

```
v = c("a", "b", "a", "c", "b")
```

- 将其转化为 factor，查看变量内容
- 将其第一个 levels 的值改为任意字符，再次查看变量内容

```
## 代码写这里，并运行；  
v <- c("a", "b", "a", "c", "b")  
v <- factor(v)  
v
```

```
## [1] a b a c b  
## Levels: a b c
```

```
levels(v)[1] <- "d"
v
```

```
## [1] d b d c b
## Levels: d b c
```

- 比较改变前后的 `v` 的内容，改变 `levels` 的操作使 `v` 发生了什么变化？

答：改变 `levels` 的操作使 `v` 的内容发生了变化，将原来的“a”改为了“d”。

0.5.3 factors 合并

- 创建两个由随机大写字母组成的 `factors`
- 合并两个变量，使其 `factors` 得以在合并后保留

```
## 代码写这里，并运行；
x <- factor(sample(LETTERS, 10, replace = TRUE))
y <- factor(sample(LETTERS, 10, replace = TRUE))
x
```

```
## [1] B C I Z F Y W A U E
## Levels: A B C E F I U W Y Z
```

```
y
```

```
## [1] T X Z Z U I R C L X
## Levels: C I L R T U X Z
```

```
c(x, y)
```

```
## [1] B C I Z F Y W A U E T X Z Z U I R C L X
## Levels: A B C E F I U W Y Z L R T X
```

0.5.4 利用 factor 排序

以下变量包含了几个月份，请使用 `factor`，使其能按月份，而不是英文字符串排序：

```
mon <- c("Mar", "Nov", "Mar", "Aug", "Sep", "Jun", "Nov", "Nov", "Oct", "Jun", "May", "Sep", "Dec",
```

```
## 代码写这里，并运行；
```

```
mon <- c("Mar", "Nov", "Mar", "Aug", "Sep", "Jun", "Nov",
        "Nov", "Oct", "Jun", "May", "Sep", "Dec", "Jul", "Nov")
mon <- factor(mon, levels = c("Jan", "Feb", "Mar",
                              "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"))
sort(mon)
```

```
## [1] Mar Mar May Jun Jun Jul Aug Sep Sep Oct Nov Nov Nov Nov Dec
## Levels: Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
```

0.5.5 forcats 的问题

`forcats` 包中的 `fct_inorder`, `fct_infreq` 和 `fct_inseq` 函数的作用是什么？

请使用 `forcats` 包中的 `gss_cat` 数据举例说明

```
## 代码写这里，并运行；
```

```
library(forcats)
```

```
## Warning: 程辑包'forcats'是用R版本4.1.3 来建造的
```

```
head(gss_cat)
```

```
##   year      marital age race      rincome      partyid
## 1 2000 Never married  26 White $8000 to 9999 Ind,near rep
## 2 2000   Divorced  48 White $8000 to 9999 Not str republican
```

```
## 3 2000      Widowed  67 White Not applicable      Independent
## 4 2000 Never married 39 White Not applicable      Ind,near rep
## 5 2000      Divorced 25 White Not applicable      Not str democrat
## 6 2000      Married  25 White $20000 - 24999      Strong democrat
##              relig          denom tvhours
## 1      Protestant Southern baptist      12
## 2      Protestant Baptist-dk which      NA
## 3      Protestant No denomination      2
## 4 Orthodox-christian Not applicable      4
## 5              None Not applicable      1
## 6      Protestant Southern baptist      NA
```

```
# fct_inorder: 按出现顺序为 levels 排序
fct_inorder(gss_cat$marital) %>% levels()
```

```
## [1] "Never married" "Divorced"      "Widowed"      "Married"
## [5] "Separated"      "No answer"
```

```
# fct_infreq: 按出现频率为 levels 排序, 出现频率高的排在前面
fct_infreq(gss_cat$marital) %>% levels()
```

```
## [1] "Married"      "Never married" "Divorced"      "Widowed"
## [5] "Separated"      "No answer"
```

```
# fct_inseq: 根据 level 的数字大小为 levels 排序, 要求 factor levels 为数字
factor(gss_cat$age, levels = 80:60) %>% levels()
```

```
## [1] "80" "79" "78" "77" "76" "75" "74" "73" "72" "71" "70" "69" "68" "67" "66"
## [16] "65" "64" "63" "62" "61" "60"
```

```
factor(gss_cat$age, levels = 80:60) %>% fct_inseq() %>% levels()
```

```
## [1] "60" "61" "62" "63" "64" "65" "66" "67" "68" "69" "70" "71" "72" "73" "74"
## [16] "75" "76" "77" "78" "79" "80"
```

0.6 练习与作业 3: 用 mouse genes 数据做图

0.6.1 画图

1. 用 readr 包中的函数读取 mouse genes 文件（从本课程的 Github 页面下载 data/talk04/ ）
2. 选取常染色体的基因
3. 画以下两个基因长度 boxplot :
 - 按染色体序号排列, 比如 1, 2, 3 X, Y
 - 按基因长度中值排列, 从短 -> 长 ...

```
## 代码写这里, 并运行;
# 不显示 warning 信息和 message
options(warn = -1, message = -1)

library(readr)
library(ggplot2)
library(dplyr)

##
## 载入程辑包: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

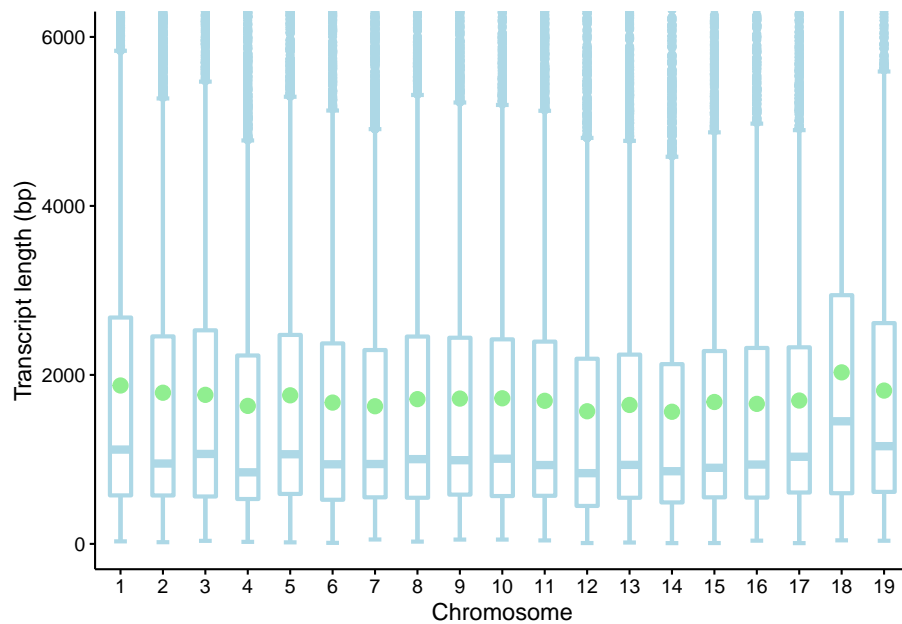


```
options(warn = 0, message = 0)
mouse_genes <- read_tsv("../data/talk04/mouse_genes_biomart_sep2018.txt",
  col_names = TRUE, show_col_types = FALSE)

colnames(mouse_genes) <- gsub(" ", "_", colnames(mouse_genes))
colnames(mouse_genes) <- gsub("/", "_", colnames(mouse_genes))
colnames(mouse_genes) <- gsub("\\(", "", colnames(mouse_genes))
colnames(mouse_genes) <- gsub("\\)", "", colnames(mouse_genes))
autosome_genes <- filter(mouse_genes, Chromosome_scaffold_name %in% 1:19)

# 按染色体序号排列
ggplot(autosome_genes, aes(x = factor(as.numeric(Chromosome_scaffold_name)),
  y = Transcript_length_including_UTRs_and_CDS)) +
  stat_boxplot(geom = "errorbar", width = 0.3, lwd = 1, color = "lightblue") +
  geom_boxplot(width = 0.5, lwd = 1, color = "lightblue") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),
    panel.grid.major.y = element_blank(),
    panel.grid.minor.y = element_blank(),
    panel.background = element_blank(),
    panel.border = element_blank(),
    axis.line = element_line(color = "black", size = 0.5),
    axis.ticks = element_line(color = "black", size = 0.5),
    axis.text.x = element_text(size = 10, color = "black"),
    axis.text.y = element_text(size = 10, color = "black"),
    axis.title.x = element_text(size = 12, color = "black"),
    axis.title.y = element_text(size = 12, color = "black"),
    legend.position = "none") +
  coord_cartesian(ylim = c(0, 6000)) +
  stat_summary(fun = mean, geom = "point", shape = 20,
    size = 5, color = "lightgreen", fill = "lightgreen") +
  xlab("Chromosome") +
```

```
ylab("Transcript length (bp)")
```



```
# 按基因长度 中值 排列, 从 短 -> 长
ggplot(autosome_genes, aes(x = reorder(Chromosome_scaffold_name,
  Transcript_length_including_UTRs_and_CDS, median),
  y = Transcript_length_including_UTRs_and_CDS)) +
  stat_boxplot(geom = "errorbar", width = 0.3, lwd = 1, color = "lightblue") +
  geom_boxplot(width = 0.5, lwd = 1, color = "lightblue") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor.y = element_blank(),
        panel.background = element_blank(),
        panel.border = element_blank(),
        axis.line = element_line(color = "black", size = 0.5),
        axis.ticks = element_line(color = "black", size = 0.5),
        axis.text.x = element_text(size = 10, color = "black"),
```

```
axis.text.y = element_text(size = 10, color = "black"),
axis.title.x = element_text(size = 12, color = "black"),
axis.title.y = element_text(size = 12, color = "black"),
legend.position = "none") +
coord_cartesian(ylim = c(0, 6000)) +
stat_summary(fun = mean, geom = "point", shape = 20,
  size = 5, color = "lightgreen", fill = "lightgreen") +
xlab("Chromosome") +
ylab("Transcript length (bp)")
```

