talk10 练习与作业

目录

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-talk10 作业.pdf`，并提交到老师指定的平台/钉群。

0.2 Talk10 内容回顾

- data summarisation functions (vector data)
  - median, mean, sd, quantile, summary
- 图形化的 data summarisation (two-D data/ tibble/ table)
  - dot plot

  - smooth
  - linear regression
  - correlation & variance explained
  - groupping & bar/ box/ plots

- statistics

  - parametric tests

    * t-test
    * one way ANNOVA
    * two way ANNOVA
    * linear regression
    * model / prediction / coefficients

  - non-parametric comparison

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"
```

引入 R 包

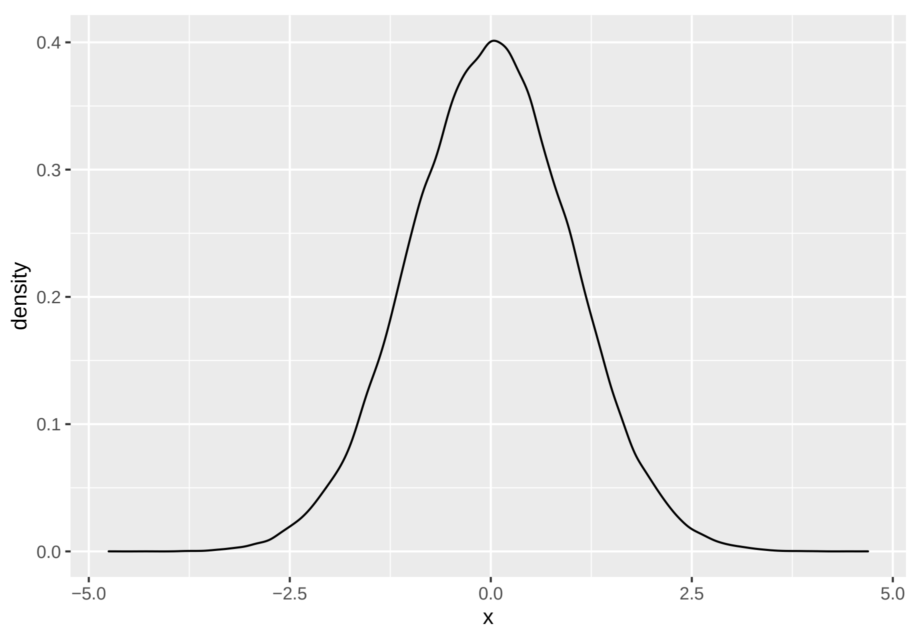
```
library(tidyverse)
library(ggsignif)
```

0.4 练习与作业 1：数据查看

- **正态分布**

1. 随机生成一个数字（numberic）组成的 vector，长度为 10 万，其值符合正态分布；
2. 用 ggplot2 的 density plot 画出其分布情况；
3. 检查 mean +- 1 * sd，mean +- 2 * sd 和 mean +- 3 * sd 范围内的取值占总值数量的百分比。

```
## 代码写这里，并运行；
# 1. 随机生成一个数字（numberic）组成的 vector，长度为 10 万，其值符合正态分布；
x <- rnorm(100000)
# 2. 用 ggplot2 的 density plot 画出其分布情况；
ggplot(data.frame(x), aes(x)) +
    geom_density()
```



```r
# 3. 检查 mean +- 1 * sd, mean +- 2 * sd 和 mean +- 3 * sd 范围内的取值占总值数量的百分比。
count <- length(x)
mean <- mean(x)
sd <- sd(x)
length(x[x > mean - sd & x < mean + sd]) / count
```

```
## [1] 0.68305
```

```r
length(x[x > mean - 2 * sd & x < mean + 2 * sd]) / count
```

```
## [1] 0.95342
```

```r
length(x[x > mean - 3 * sd & x < mean + 3 * sd]) / count
```

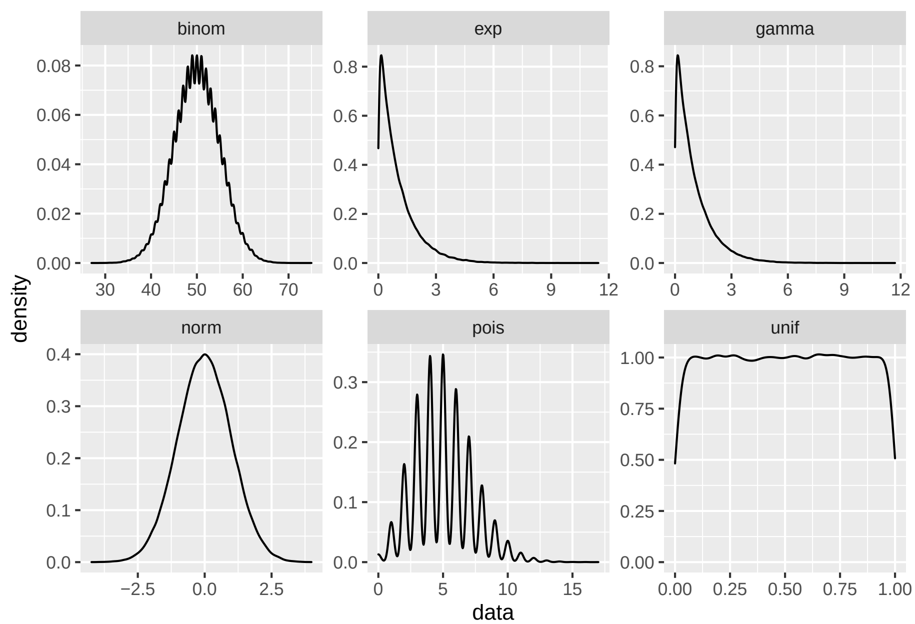
```
## [1] 0.99728
```

- **用函数生成符合以下分布的数值，并做图：**

另外，在英文名后给出对应的中文名：

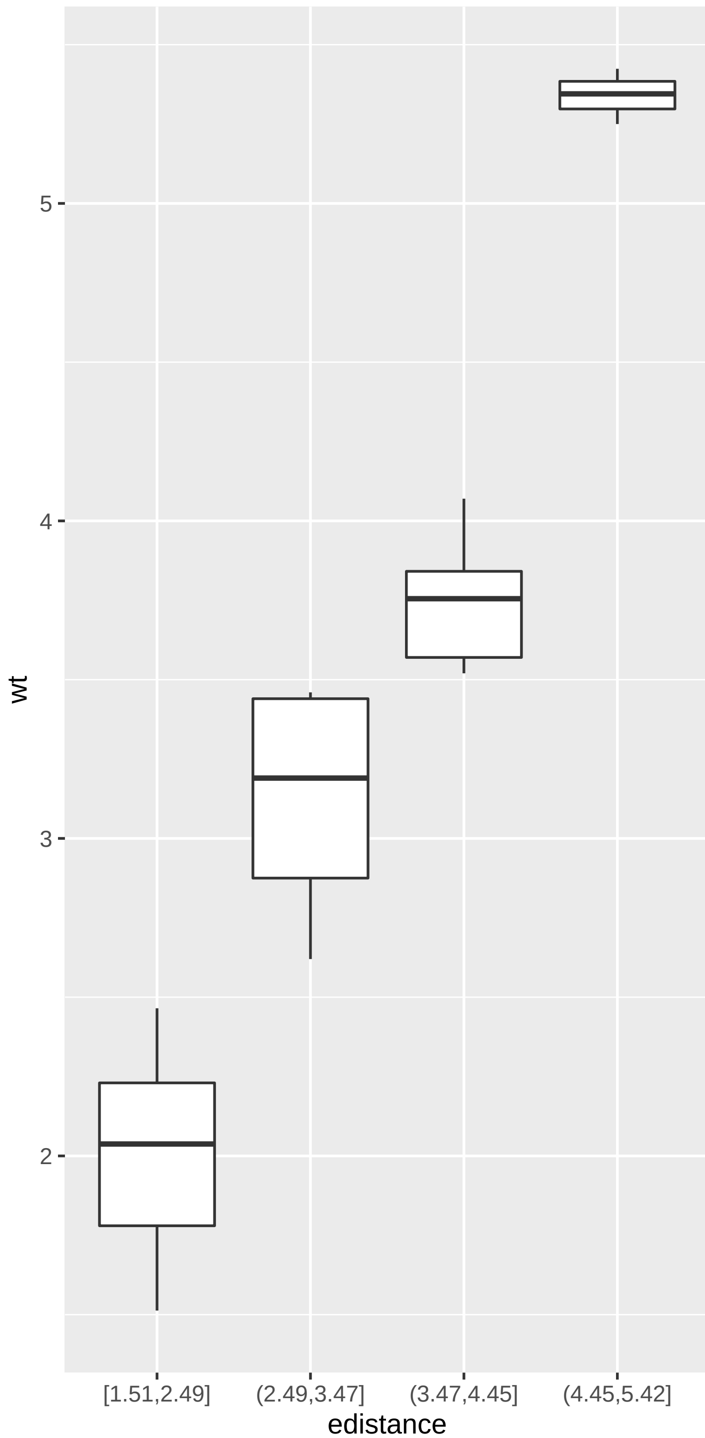
```
-   Uniform Distribution（均匀分布）

-   Normal Distribution（正态分布）

-   Binomial Distribution（二项分布）

-   Poisson Distribution（泊松分布）

-   Exponential Distribution（指数分布）

-   Gamma Distribution（伽马分布）
```

```r
## 代码写这里，并运行；
distributions = bind_rows(
  tibble(dtr = "unif", data = runif(100000)),
  tibble(dtr = "norm", data = rnorm(100000)),
  tibble(dtr = "binom", data = rbinom(100000, 100, 0.5)),
  tibble(dtr = "pois", data = rpois(100000, 5)),
  tibble(dtr = "exp", data = rexp(100000)),
  tibble(dtr = "gamma", data = rgamma(100000, 1))
)
ggplot(distributions, aes(x = data)) +
  geom_density() +
  facet_wrap(~dtr, ncol = 3, scales = "free")
```



- 分组的问题
  - 什么是 equal-sized bin 和 equal-distance bin？以 mtcars 为例，将 wt 列按两种方法分组，并显示结果。

```
## 代码写这里，并运行；
mtcars_bin <- mtcars %>%
    mutate(
        esize = ntile( wt, 4 ), ## equal-size bining
        edistance = cut(
            wt, ## equal-distance
            breaks = seq(
                from = min(wt),
                to = max(wt),
                by = (max(wt) - min(wt)) / 4 ),
            include.lowest = T ))
# equal-distance 是等距离分组。
ggplot( mtcars_bin, aes( edistance, wt ) ) +
    geom_boxplot()
```



wt
5
4
3
2
[1.51,2.49]
(2.49,3.47]
(3.47,4.45]
(4.45,5.42]
edistance

```
# equal-size bining 保证每个分组的样本数量相同
table(mtcars_bin$esize)
```

```
##
## 1 2 3 4
## 8 8 8 8
```

- **boxplot 中 outlier 值的鉴定**
  - 以 swiss$Infant.Mortality 为例，找到它的 outlier 并打印出来；

```
## 代码写这里，并运行；
# 鉴定 outlier
swiss %>%
    mutate(
        outlier = ifelse(
            Infant.Mortality < quantile(Infant.Mortality, 0.25) - 1.5 * IQR(Infant.Mort
            Infant.Mortality > quantile(Infant.Mortality, 0.75) + 1.5 * IQR(Infant.Mort
            "outlier", "normal" )) %>%
    filter(outlier == "outlier") %>%
    select(Infant.Mortality)
```

```
##            Infant.Mortality
## La Vallee              10.8
```

- **以男女生步数数据为例，进行以下计算：**

首先用以下代码装入 Data:

```r
source("../data/talk10/input_data1.R") ## 装入 Data data.frame ...
head(Data)
```

```
##   Student    Sex Teacher Steps Rating
## 1       a female  Catbus  8000      7
## 2       b female  Catbus  9000     10
## 3       c female  Catbus 10000      9
## 4       d female  Catbus  7000      5
## 5       e female  Catbus  6000      4
## 6       f female  Catbus  8000      8
```

-    分别用`t.test`和`wilcox.test`比较男女生步数是否有显著差异；打印出`p.value`

```r
## 代码写这里，并运行；
# wilcox.test p.value
with(Data, wilcox.test(Steps ~ Sex))$p.value
```

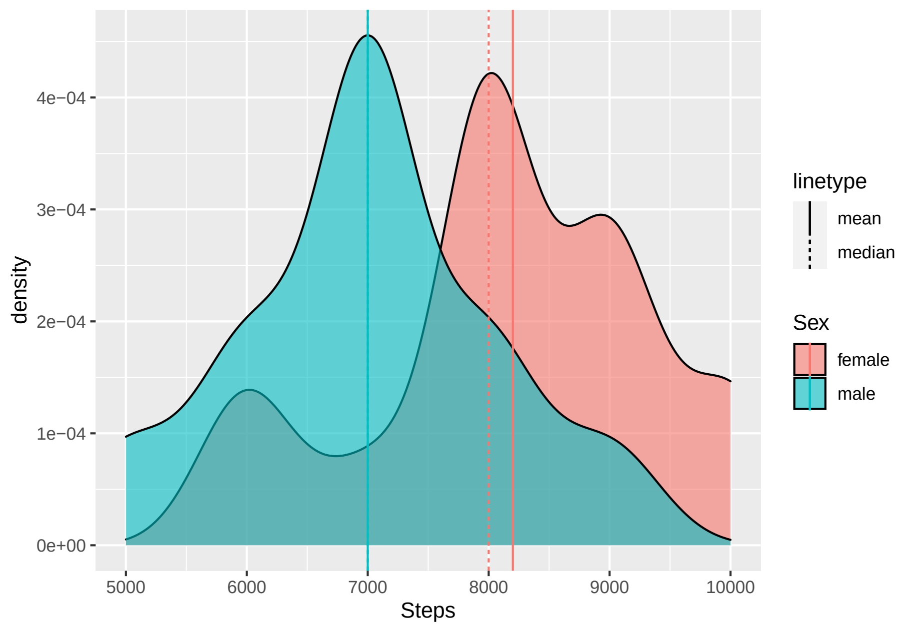
```
## [1] 0.01773304
```

```r
# t.test p.value
t.test(Data$Steps ~ Data$Sex)$p.value
```

```
## [1] 0.01461209
```

```r
# 计算两组步数的均值和中位数
annotation <- Data %>%
    group_by(Sex) %>%
    summarise(
        mean = mean(Steps),
        median = median(Steps))

# Data 的步数分布
ggplot(Data, aes(Steps, fill = Sex)) +
```

```
  geom_density(position="dodge", alpha = 0.6) +
  # 在指定位置标注线
  geom_vline(
    data = annotation,
    aes(xintercept = mean, linetype = "mean", color = Sex)) +
  geom_vline(
    data = annotation,
    aes(xintercept = median, linetype = "median", color = Sex))
```



- 两种检测方法的`p.value`哪个更显著？为什么？

答：t.test 检验更显著，t 检验是参数方法，而 Wilcoxon 秩和检验是非参数方法。当资料满足正态性的假设，参数方法比非参数方法检验效能更高。计算使用的男女步数的数据比较符合正态分布。

- **以下是学生参加辅导班前后的成绩情况，请计算同学们的成绩是否有普遍提高？**

注：先用以下代码装入数据：

```
source("../data/talk10/input_data2.R")
head(scores)
```

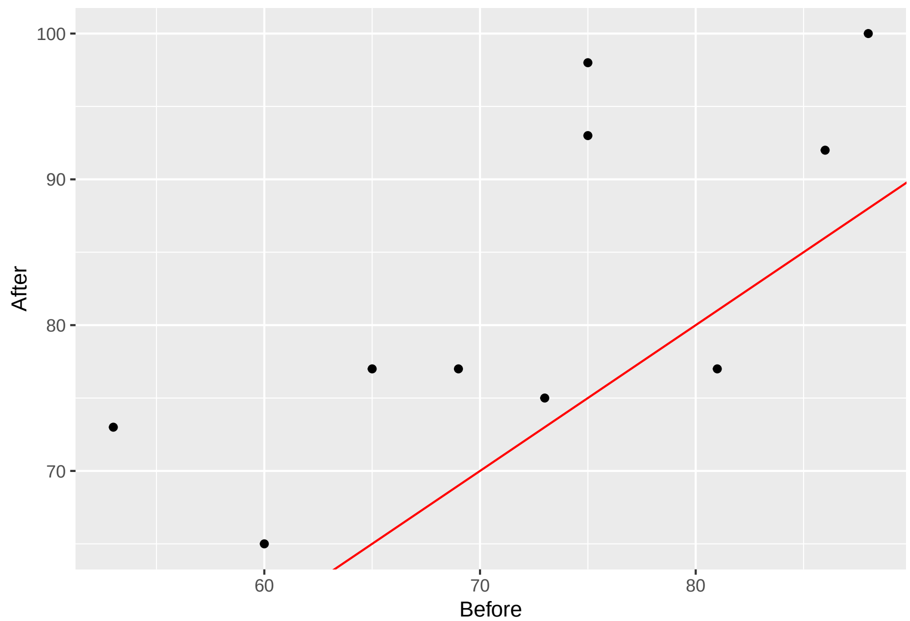
```
##     Time Student Score
## 1 Before       a    65
## 2 Before       b    75
## 3 Before       c    86
## 4 Before       d    69
## 5 Before       e    60
## 6 Before       f    81
```

注：计算时请使用 `paired = T` 参数；

```
## 代码写这里，并运行；
scores_wide <- scores %>%
    spread(Time, Score)
head(scores_wide, n = 3)
```

```
##   Student After Before
## 1       a    77     65
## 2       b    98     75
## 3       c    92     86
```

```
ggplot(scores_wide, aes(Before, After)) +
    geom_point() +
    geom_abline(intercept = 0, slope = 1, color = "red")
```




```r
t.test(scores_wide$Before, scores_wide$After, paired = T)$p.value
```

```
## [1] 0.004163495
```

```r
# 参加辅导班后成绩显著提高了
```

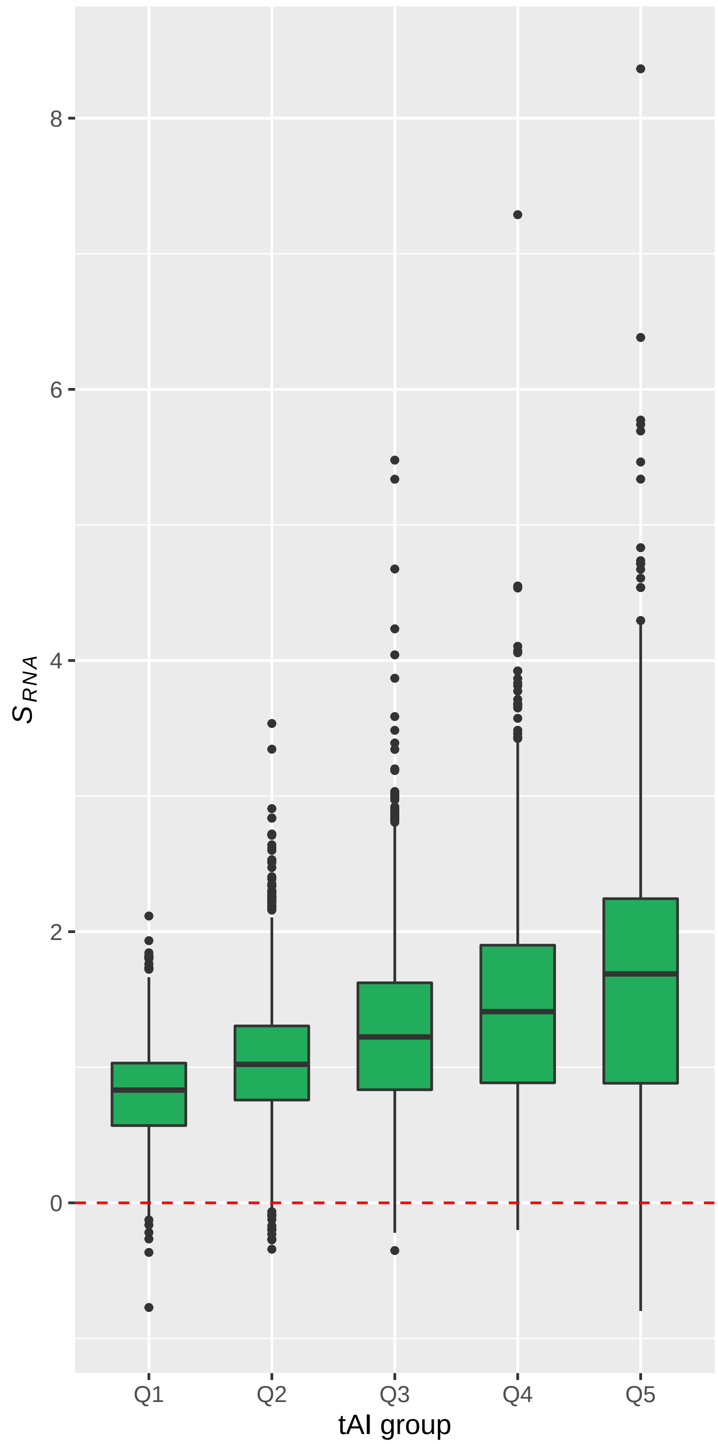
0.5 练习与作业 2：作图

- **利用 talk10 中的 data.fig3a 作图**
  - 首先用以下命令装入数据：

```r
data.fig3a <- read_csv( file = "../data/talk10/nc2015_data_for_fig3a.csv" )
```

```
-   利用两列数据：`tai` `zAA1.at` 做`talk10`中的`boxplot`（详见：`fig3a`的制作）;

-   用`ggsignif`为相邻的两组做统计分析（如用 `wilcox.test` 函数），并画出`p.value`;
```

```
## 代码写这里，并运行；
(fig3a <- ggplot( data.fig3a, aes( factor(tai), zAA1.at ) ) +
    geom_boxplot( fill = "#22AD5C", linetype = 1 ,outlier.size = 1, width = 0.6) +
    xlab( "tAI group" ) +
    ylab( expression( paste( italic(S[RNA]) ) ) ) +
    scale_x_discrete(breaks= 1:5 , labels= paste("Q", 1:5, sep = "") ) +
    geom_hline( yintercept = 0, colour = "red", linetype = 2))
```

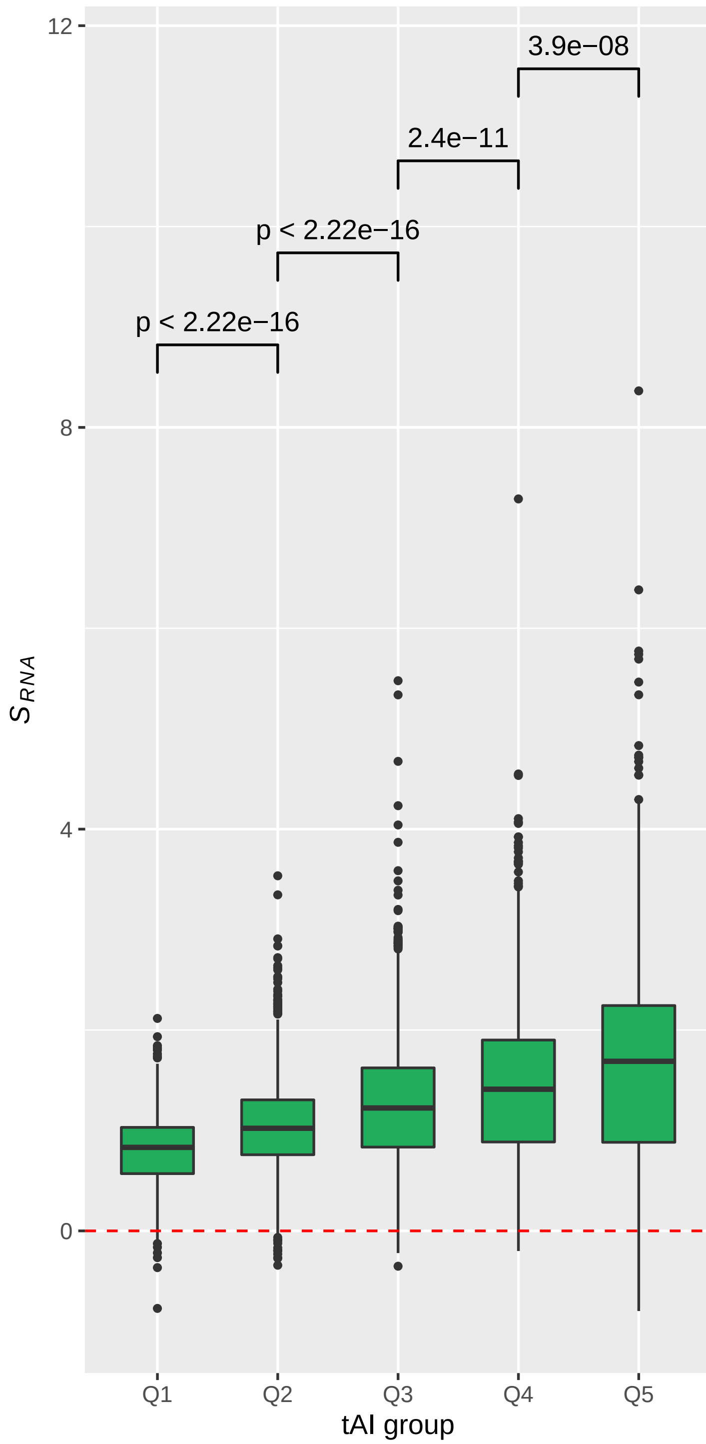


8
6
4
2
0
S_{RNA}
Q1
Q2
Q3
Q4
Q5
tAI group

问：这组数据可以用 `t.test` 吗？为什么？

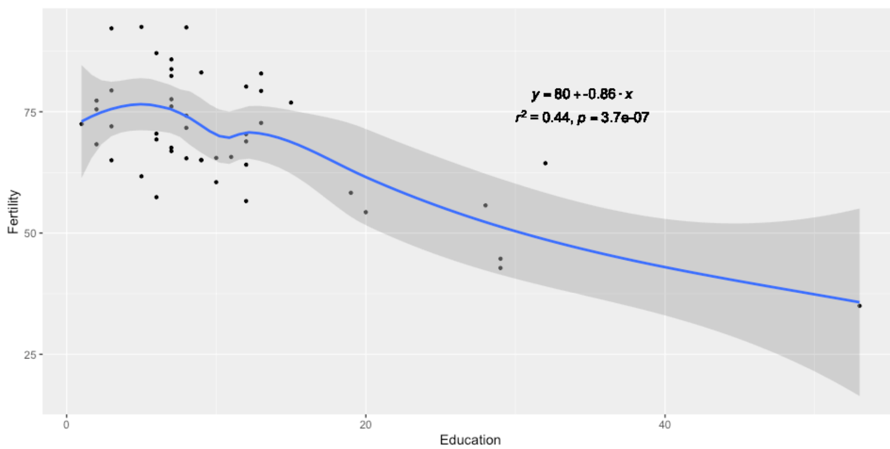
答：这组数据不能用 `t.test`，因为 `t.test` 适用于正态分布的数据，而这组数据不是正态分布的。

```
## 代码写这里，并运行；
fig3a + geom_signif( comparisons = list(1:2, 2:3, 3:4, 4:5), test = wilcox.test,
    step_increase = 0.1 )
```



3.9e−08
2.4e−11
p < 2.22e−16
p < 2.22e−16
12
8
4
0
S_{RNA}
Q1
Q2
Q3
Q4
Q5
tAI group

- **用系统自带变量 mtcars 做图**
  - 用散点图表示 wt（x-轴）与 mpg（y-轴）的关系
  - 添加线性回归直线图层
  - 计算 wt）与 mpg 的相关性，并将结果以公式添加到图上。其最终效果如下图所示（注：相关代码可在 talk09 中找到）：

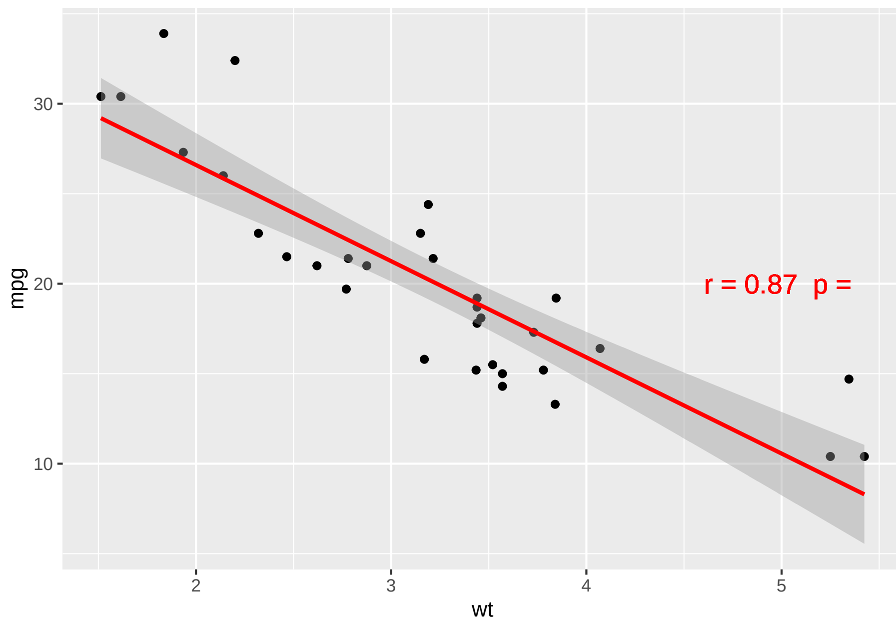



```
## 代码写这里，并运行；
# 计算相关性
cor( mtcars$wt, mtcars$mpg )
```

```
## [1] -0.8676594
```

```
mtcars %>%
    ggplot( aes( wt, mpg ) ) +
    geom_point() +
    geom_smooth(method = "lm", color = "red") +
    geom_text( x = 5, y = 20, label = "r = 0.87  p = ", size = 5, color = "red" )
```

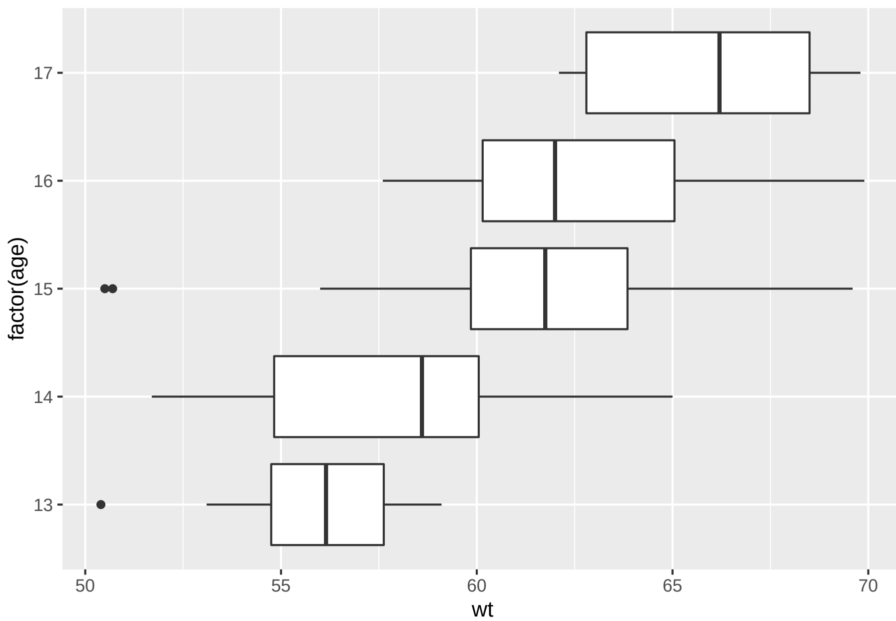
```
## `geom_smooth()` using formula 'y ~ x'
```



0.6 练习与作业 3：线性模型与预测

- 使用以下代码产生数据进行分析

```
wts2 <- bind_rows(
   tibble( class = 1, age = sample( 13:15, 20, replace = T ), wt = sample( seq(50, 60,
   tibble( class = 2, age = sample( 14:16, 20, replace = T ), wt = sample( seq(55, 65,
   tibble( class = 3, age = sample( 15:17, 20, replace = T ), wt = sample( seq(60, 70,
)

ggplot(wts2, aes( factor( age ), wt ) ) + geom_boxplot() + coord_flip()
```



- 用线性回归检查`age`, `class` 与 `wt` 的关系，构建线性回归模型；
- 以`age`, `class`为输入，用得到的模型预测`wt`；
- 计算预测的`wt`和实际`wt`的相关性；
- 用线性公式显示如何用`age`, `class`计算`wt`的值。

```
## 代码写这里，并运行；
# 用线性回归检查 age, class 与 wt 的关系，构建线性回归模型；
model <- lm( wt ~ age + class, data = wts2 )
summary( model )
```

```
##
## Call:
## lm(formula = wt ~ age + class, data = wts2)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -5.1504 -2.3743 -0.1108  2.4779  4.9374
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  52.7937     6.8644   7.691 2.24e-10 ***
## age          -0.1835     0.5346  -0.343    0.733
## class         5.0626     0.7475   6.773 7.59e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.121 on 57 degrees of freedom
## Multiple R-squared:  0.631,  Adjusted R-squared:  0.6181
## F-statistic: 48.75 on 2 and 57 DF,  p-value: 4.557e-13
```

```r
# 以 age, class 为输入，用得到的模型预测 wt;
predict(model, data.frame( age = 15, class = 2 ))
```

```
##        1
## 60.16691
```

```r
# 计算预测的 wt 和实际 wt 的相关性;
cor( predict( model ), wts2$wt )
```

```
## [1] 0.794384
```

```r
# 用线性公式显示如何用 age, class 计算 wt 的值。
model$coefficients
```

```
## (Intercept)         age       class
##  52.7937017  -0.1834719   5.0626455
```

```
paste0( "wt = ",
    model$coefficients[1], " + ",
    model$coefficients[2], " * age + ",
    model$coefficients[3], " * class" )
```

```
## [1] "wt = 52.7937017114914 + -0.183471882640584 * age + 5.06264547677261 * class"
```