

R 语言编程：基于 tidyverse

第 18 讲 描述性统计

张敬信

2022 年 12 月 6 日

哈尔滨商业大学

R 语言就是因统计分析而生的编程语言，可以很方便地完成各种统计计算、统计模拟、统计建模等。

统计学是关于数据的科学，是一套有关数据收集整理（获取及预处理数据）、描述统计（汇总、图表描述）、分析推断（选择适当的统计方法研究数据，并从数据中提取有用信息进而得出结论）的方法。

描述性统计，主要是通过计算汇总统计量、绘制统计图来描述数据。

一. 若干概念

1. 随机变量

当一件事情的结果无法预料时，就叫随机现象。表示随机现象一组结果的变量就是**随机变量**。

比如说，调查了 100 个人的身高，这 100 个身高的数据是随机变量身高的数据。并不是说这些身高值是不固定可变的，而是这 100 个身高值是一次调查的结果，再调查 100 个人就是另一组不同的 100 个身高值。

2. 概率分布

随机变量既然是这样随机的，还有必要研究它吗？有必要！因为把多个随机结果放在一起的时候，能发现一定的规律性。比如 100 人的身高可能对称地分布在 175cm 附近，离得越远人数越少，即表现出一种正态分布规律性。

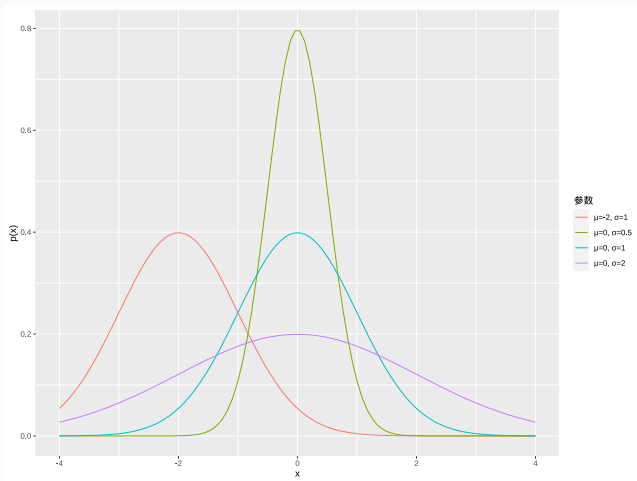
随机现象五花八门，但每一种随机现象表现出来的规律性是固定的，用数学语言表达出来就是**概率分布**。所以，不同概率分布就是不同随机现象规律性的数学描述。

同一种概率分布，也不是都相同，这是由不同参数值决定和区分的。

统计学最常用到四大概率分布：正态分布、t 分布、卡方分布、F 分布。

比如正态分布 $N(\mu, \sigma^2)$, μ 和 σ 就是参数, 它们只要取不同值, 就是不同的分布形状:

```
library(tidyverse)
tibble(
  x = seq(-4,4,length.out = 100),
  `=0, =0.5` = dnorm(x, 0, 0.5),
  `=0, =1` = dnorm(x, 0, 1),
  `=0, =2` = dnorm(x, 0, 2),
  `=-2, =1` = dnorm(x, -2, 1)
) %>%
  pivot_longer(-x, names_to = " 参数",
               values_to = "p(x)") %>%
  ggplot(aes(x, `p(x)`, color = 参数)) +
  geom_line()
```



3. 概率论与数理统计

概率论就是研究随机现象规律性，即各种概率分布及性质的理论。数理统计所研究的数据是带有随机性的，所以需要借助概率论中的概率分布理论加以描述和做出统计推断。所以说：

概率论是数理统计的理论基础，数理统计是概率论的一种应用

4. 区分数据类型

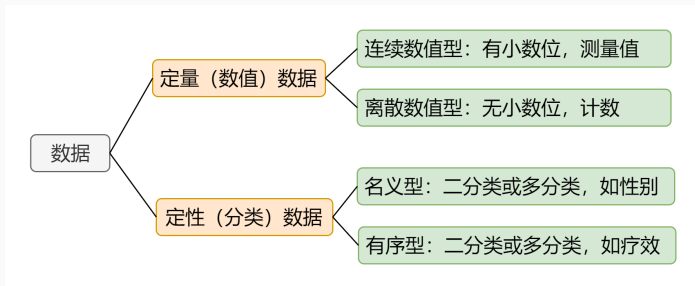


图 1: 常见的数据类型

区分数据类型非常有必要，因为不同数据类型适用的统计分析方法是不同的！

5. 总体和样本

- **总体 (population)**: 是包含所研究的全部个体 (数据) 的集合。
- **样本 (sample)**: 从总体中抽取的一部分个体的集合, 样本包含个体的数目称为样本量。

抽样的目的是根据样本数据提供的信息推断总体的特征, 或者说, 用样本统计量推断总体参数。

比如, 要研究哈尔滨市成年男性的身高, 则所有哈尔滨市成年男性的身高数据就是总体, 但实际上不可能把所有这些身高都测量一遍, 只能是随机抽取一部分, 比如 100 人, 测得身高数据, 这就是样本, 样本量是 100。

抽样调查结果的可靠性不在于样本数量大不大（当然也不能太少），更主要的是科学抽样，使样本足够代表总体。

身高数据大致服从正态分布，所有哈尔滨市成年男性身高的均值 μ 和标准差 σ ，就是总体参数。用样本的 100 人的平均身高作为 μ 的估计，就是用样本统计量推断总体参数。

6. 参数与统计量

- **参数 (parameter)**: 用来描述总体特征的概括性值, 是研究者想要了解的总体的某种特征值, 如总体均值 (μ)、总体方差 (σ^2)、总体比例 (π) 等。
- **统计量 (statistic)**: 是用来描述样本特征的概括性数字度量, 是根据样本数据计算出来的量, 由于抽样是随机的, 因此统计量是样本的函数。与上面总体参数对应的统计量是样本均值 (\bar{x})、样本方差 (s^2)、样本比例 (p) 等。

由于总体数据通常是不知道的, 故参数是未知常数。所以才进行抽样, 根据样本计算出相应统计量值去估计总体参数值。

二. (样本) 统计量

1. 数据位置的统计量

(1) 均值 (Mean)

均值，度量数据分布的中心位置：

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

(2) 中位数 (Median)

中位数，是位于最中间的那个数据，比中位数大和小的数据各占观测值的一半。先将数据从小到大排序为： $x_{(1)}, \dots, x_{(n)}$ ，然后计算：

$$x_{0.5} = \begin{cases} x_{(\frac{n+1}{2})}, & n \text{ 为奇数} \\ \frac{1}{2} \left(x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)} \right), & n \text{ 为偶数} \end{cases}$$

中位数的优点是具有稳健性，即不受个别极端数据的影响。一般来说，正态分布的数据用均值描述，偏态分布的数据最好是用中位数描述。比如，人均工资有被平均了的感觉，中位数工资才是更合适的中间收入。

(3) 分位数 (Quantile)

中位数是 0.5 分位数, 位于 0.5 位置的数。

0.25 分位数, 称为下四分位数 (Q1), 是位于 0.25 那个位置的数, 即比它小的数占比是 0.25, 比它大的数占比是 0.75.

0.75 分位数, 称为上四分位数 (Q3).

更一般地, p **分位数**, 是位于 p 位置的数, 即比它小的数占比是 p , 比它大的数占比是 $1 - p$. 或者说 np 的数比它小, $n(1 - p)$ 的数比它大。

(4) 众数 (Mode)

众数，是观测值中出现次数最多的数，对应分布的最高峰。众数常用于分类数据，即出现频数最高的值。

R 实现：

- `mean(x)`: 计算数值向量 `x` 的均值
- `median(x)`: 计算数值向量 `x` 的中位数
- `quantile(x, p)`: 计算数值向量 `x` 的 `p` 分位数
- `rstatix::get_mode(x)`: 计算向量 `x` 的众数

2. 数据分散程度的统计量

(1) 极差 (Range)

极差，就是数据中的最大值和最小值之差。

(2) 四分位距 (Interquartile Range)

四分位距，是上下四分位数之差，即

$$IQR = Q3 - Q1$$

(3) 样本方差 (Variance)

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n |x_i - \bar{x}|^2$$

注意，分母除的是 $n - 1$ ，这是为了保证用样本方差估计总体方差时，得到的是无偏估计。

这个 $n - 1$ 也是自由度，在统计学中，几乎所有方法、所有统计量都会涉及自由度。**自由度**，是计算样本统计量时能够自由取值的数值的个数。

总体方差公式（除以 n ）时，是 n 个样本自由地从总体里抽取。但是样本方差公式时多了一个约束条件，它们的和除以 n 必须等于样本均值 \bar{x} ，所以自由度 n 减去 1 个约束条件对自由度的损失，等于 $n - 1$ 。

不同统计方法的自由度都不一样，但基本原则是每估计 1 个参数，就需要消耗 1 个自由度。

以回归分析为例，若有 m 个自变量，则需要估计 $m + 1$ 个参数（包含截距项），所以模型的 F 检验用到的自由度是 $n - (m + 1)$ 。这意味着只剩下 $n - (m + 1)$ 个可以自由取值的数值用来估计模型误差。

(4) 样本标准差 (Standard Deviation)

样本方差的平方根即为标准差 s 。标准差的量纲与原数据一致。

(5) 变异系数 (Coefficient of Variation)

变异系数，是将标准差占均值的百分比，可用于比较不同量纲数据的分散性：

$$c_v = \frac{s}{\bar{x}} \quad (\%)$$

R 实现：

- `max(x)-min(x)`：计算数值向量 x 的极差
- `IQR(x)`：计算数值向量 x 的四分位距
- `var(x)`：计算数值向量 x 的样本方差
- `sd(x)`：计算数值向量 x 的样本标准差
- `100*sd(x)/mean(x)`：计算数值向量 x 的变异系数

3. 数据分布形状的统计量

(1) 偏度 (Skewness)

偏度，刻画数据是否对称的指标：

$$SK = \frac{n}{(n-1)(n-2)} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s} \right)^3$$

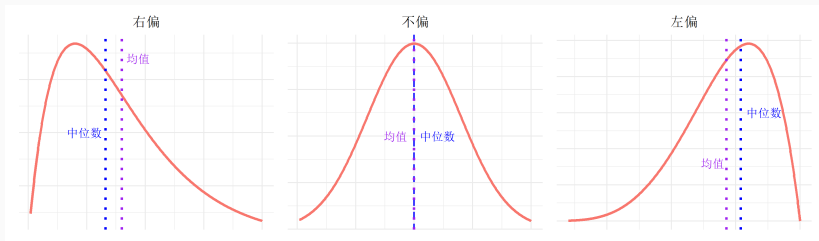


图 2：数据的三种偏态

关于均值对称的数据不偏，其偏度为 0；右拖尾的数据是右偏，其偏度为正；左拖尾的数据是左偏，其偏度为负。

(2) 峰度 (Kurtosis)

峰度，刻画数据是否尖峰的指标：

$$K = \frac{n(n+1)}{(n-1)(n-2)(n-3)} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s} \right)^4 - \frac{3(n-1)^2}{(n-2)(n-3)}$$

峰度是以标准正态分布为基准，标准正态分布的峰度为 0；尖峰薄尾的分布峰度为正；平峰厚尾的分布峰度为负。

`datawizard` 包提供了 `skewness()` 和 `kurtosis()` 函数分别计算偏度和峰度。

- 很多包提供了同时对多个变量进行（分组）描述汇总所有常见统计量的函数，其中 tidy 风格的是 `rstatix::get_summary_stats()` 和 `dlookr::describe()`.

```
library(rstatix)
```

```
iris %>%
```

```
  group_by(Species) %>%
```

```
  get_summary_stats(type = "full")
```

```
#> # A tibble: 12 x 14
```

```
#>   Species variable      n   min   max median    q1    q3
```

```
#>   <fct>    <fct>    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl>
```

```
#> 1 setosa  Sepal.Le~   50   4.3   5.8     5     4.8   5.2
```

```
#> 2 setosa  Sepal.Wi~   50   2.3   4.4     3.4   3.2   3.6
```

```
#> 3 setosa  Petal.Le~   50    1    1.9     1.5   1.4   1.5
```

```
#> # ... with 9 more rows, and 2 more variables: se <dbl>,
```

三. 统计图

描述统计是从不同方面对数据做了概要，想要进一步了解和探索数据，离不开绘制统计图。不同类型的数据，适用不同类型的统计图。

1. 分类数据的统计图

(1) 条形图 (Bar)

条形图是最常用的类别比较图，是用竖直（或水平）的条形展示分类变量的分布（频数），条形的高度代表频数。

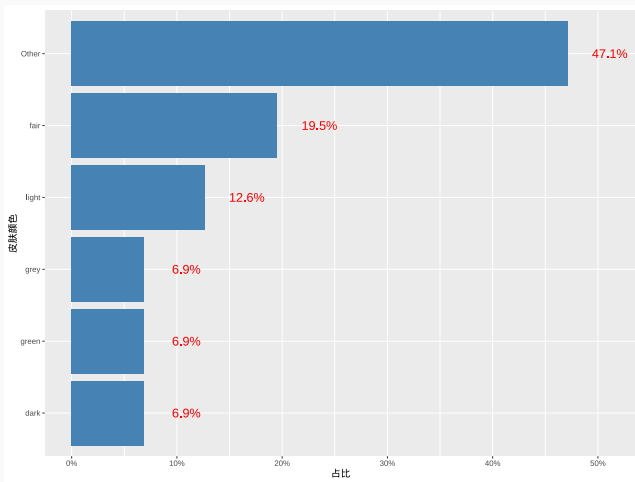
- `geom_bar()`: 对原始数据绘制条形图
- `geom_col()`: 对汇总频数/频率的数据用绘制条形图

以 `starwars` 数据集 `skin_color` 绘制条形图为例：

- 用 `fct_lump()` 将频数 ≤ 5 的类别做了合并
- 分组汇总，计算各组频数和频率
- 绘制条形图，将分类变量 `skin_color` 按频率做了因子重排序，实现了对” 条形” 排序
- 在条形旁边增加文字注释，标记该条形所占百分比
- 翻转坐标轴，变成水平条形图

```
df = starwars %>%  
  mutate(skin_color = fct_lump(skin_color, n = 5)) %>%  
  count(skin_color, sort = TRUE) %>%  
  mutate(p = n / sum(n))  
df  
#> # A tibble: 6 x 3  
#>   skin_color      n      p  
#>   <fct>      <int> <dbl>  
#> 1 Other          41 0.471  
#> 2 fair           17 0.195  
#> 3 light          11 0.126  
#> # ... with 3 more rows
```

```
ggplot(df, aes(fct_reorder(skin_color, p), p)) +  
  geom_col(fill = "steelblue") +  
  # 同 geom_bar(stat = "identity")  
  scale_y_continuous(labels = scales::percent) +  
  labs(x = " 皮肤颜色", y = " 占比") +  
  geom_text(aes(y = p + 0.04,  
                label = str_c(round(p*100,1), "%")),  
            size = 5, color = "red") +  
  coord_flip()
```



(2) 饼图 (Pie)

饼图，是用每个扇形的圆心角大小表示每部分所量所占的比例，注意饼图很难去精确比较不同部分的大小。

Hadley 认为饼图可以通过极坐标变换得到，没有提供绘制饼图的几何对象，另外从展示分类数据角度来说，饼图也不是一个好的选择。

注：饼图模板案例见第 17 讲。

2. 连续数据的统计图

(1) 直方图 (Histogram)

连续数据常用直方图来展示变量取值的分布，利用直方图可以估计总体的概率密度。

将变量取值的范围分成若干区间。直方图是用面积而不是用高度来表示数，总面积是 100%。每个区间矩形的面积恰是落在该区间内的百分数（频率），所以

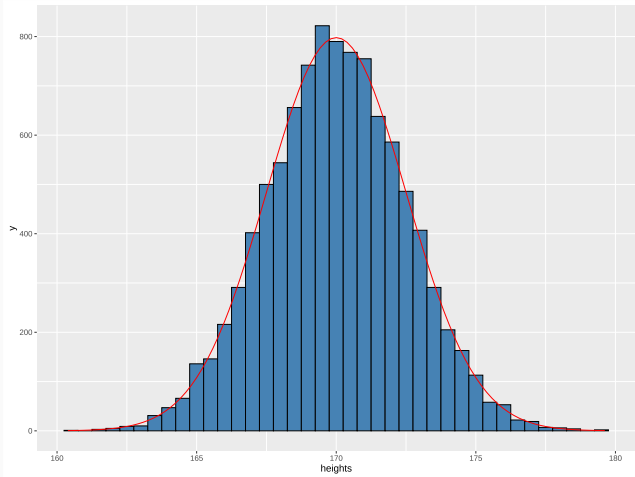
$$\text{矩形的高} = \text{频率} / \text{区间长度} = \text{密度}$$

特别地，若区间是等长的，则“矩形的高”就是频率。注意：直方图矩形之间是没有间隔的。

用 `geom_histogram()` 绘制直方图。频率直方图与概率密度曲线正好搭配，因为频率直方图的条形宽度趋于 0，就是概率密度曲线。

若想绘制频数直方图 + 概率密度曲线，就需要对密度做一个放大：条形宽度 * 样本数倍。

```
set.seed(123)
df = tibble(heights = rnorm(10000, 170, 2.5))
ggplot(df, aes(x = heights)) +
  geom_histogram(fill = "steelblue", color = "black",
                 binwidth = 0.5) +
  stat_function(
    fun = ~ dnorm(.x, mean=170, sd=2.5) * 0.5 * 10000,
    color = "red")
```



注：若想在同一张图上叠加多个直方图，以对比分类变量不同水平的概率分布，更适合用 `geom_freqpoly()` 绘制频率多边形图；函数 `geom_density()` 绘制核密度估计曲线。

(2) 箱线图 (Boxplot)

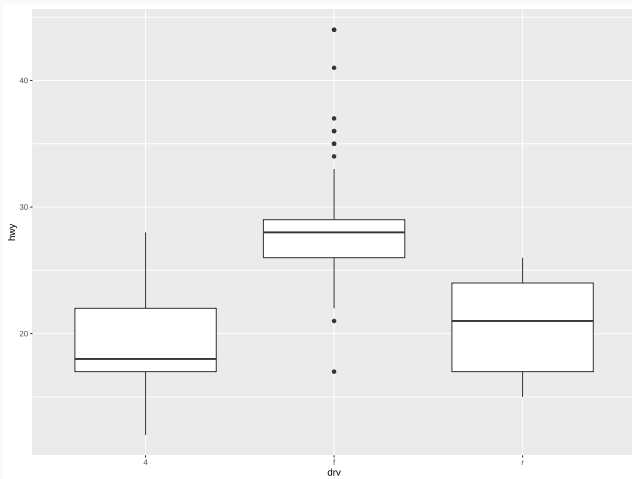
箱线图，是在一条数轴上：

- 以数据的上下四分位数 (Q1-Q3) 为界画一个矩形盒子 (中间 50% 的数据落在盒内)；
- 在数据的中位数位置画一条线段为中位线；
- 默认延长线为盒长的 1.5 倍，之外的点认为是异常值。

箱线图的主要应用就是，剔除数据的异常值、判断数据的偏态和尾重、可视化组间差异。

用 `geom_boxplot()` 绘制箱线图，例如比较不同 `drv` 下, `hwy` 的组间差异：

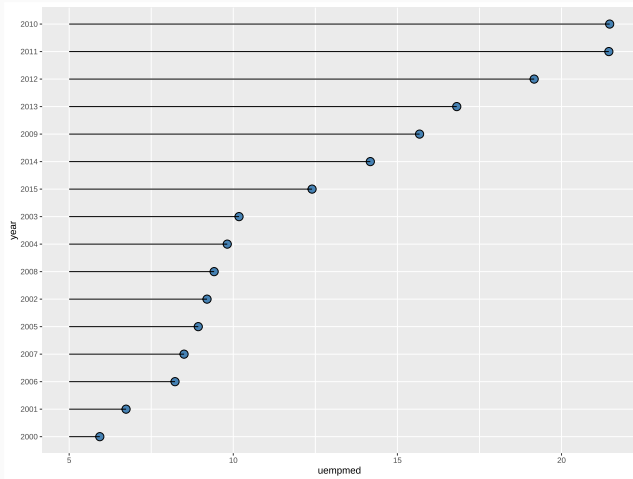
```
ggplot(mpg, aes(x = drv, y = hwy)) +  
  geom_boxplot()    # 水平翻转加图层 coord_flip()
```



(3) 克利夫兰点图 (Cleveland Dot Plot)

克利夫兰点图适合展示多类别之间的比较：x 轴是分类变量，每一类对应一个类均值或频数 (y 值)，画一个圆点；并根据 y 值大小对 x 轴类别排序；通常需要再做一次坐标翻转。

```
economics %>%  
  group_by(year = lubridate::year(date)) %>%  
  summarise(uempmed = mean(uempmed)) %>%  
  filter(year >= 2000) %>%  
  ggplot(aes(reorder(year, uempmed), uempmed)) +  
  geom_point(size = 4, shape = 21,  
             fill = "steelblue", color = "black") +  
  geom_segment(aes(xend = ..x.., yend = 5)) +  
  xlab("year") +  
  coord_flip()
```



(4) 均值线与误差棒图 (Mean Lines, Error Bars)

以 ToothGrowth 数据集为例，先自定义分组汇总函数计算分组均值和标准误：

```
my_summary = function(data, .summary_var, ...) {  
  summary_var = enquo(.summary_var)  
  data %>%  
    group_by(...) %>%  
    summarise(mean = mean(!!summary_var, na.rm = TRUE),  
              sd = sd(!!summary_var, na.rm = TRUE)) %>%  
    mutate(se = sd / sqrt(n()))  
}
```

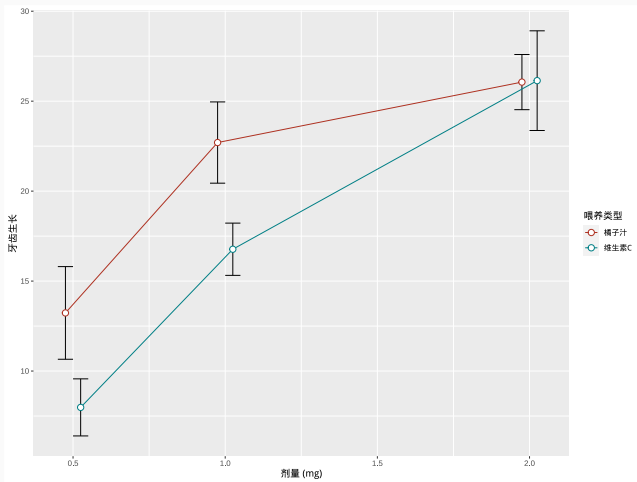
```
df = my_summary(ToothGrowth, len, supp, dose)
df
```

```
#> # A tibble: 6 x 5
#> # Groups:   supp [2]
#>   supp    dose  mean    sd    se
#>   <fct> <dbl> <dbl> <dbl> <dbl>
#> 1 OJ      0.5  13.2  4.46  2.57
#> 2 OJ      1    22.7  3.91  2.26
#> 3 OJ      2    26.1  2.66  1.53
#> # ... with 3 more rows
```

```

pd = position_dodge(0.1)
ggplot(df, aes(dose, mean, color = supp, group = supp)) +
  geom_errorbar(aes(ymin = mean - se, ymax = mean + se),
                color = "black", width = 0.1, position = pd) +
  geom_line(position = pd) +
  geom_point(position = pd, size = 3, shape = 21,
             fill = "white") +
  xlab(" 剂量 (mg)") + ylab(" 牙齿生长") +
  scale_color_hue(name = " 喂养类型", breaks = c("OJ", "VC"),
                  labels = c(" 橘子汁", " 维生素 C"), l = 40) +
  scale_y_continuous(breaks = 0:20 * 5)

```



四. 列联表

对分类变量做描述统计，通常是计算各水平值出现的频数和占比，得到**列联表**（交叉表）。用 `table()` 可以实现，但功能很弱还不够 tidy.

`janitor` 包提供了更强大的 `tabyl()` 函数，可以生成一个、两个、三个变量的列联表，再结合 `adorn_*` 函数，可以很方便地按想要的格式添加行列合计、占比等。

- 一维列联表，添加合计行：

```
library(janitor)
mpg %>%
  tabyl(drv) %>%
  adorn_totals("row") %>%
  adorn_pct_formatting()
#>      drv    n percent
#>      4  103   44.0%
#>      f  106   45.3%
#>      r   25   10.7%
#> Total 234  100.0%
```

添加合计行
设置百分比格式

- 二维列联表，添加列占比和频数

```
mpg %>%  
  tabyl(drv, cyl) %>%  
  adorn_percentages("col") %>%           # 添加列占比  
  adorn_pct_formatting(digits = 2) %>%    # 设置百分比格式  
  adorn_ns()                             # 添加频数  
  
#>   drv           4           5           6           8  
#>   4 28.40% (23)    0.00% (0) 40.51% (32) 68.57% (48)  
#>   f 71.60% (58) 100.00% (4) 54.43% (43)  1.43% (1)  
#>   r  0.00% (0)   0.00% (0)  5.06% (4) 30.00% (21)
```

注：三维列联表是针对 3 个分类变量，结果就像多维数组的“分页”。

另外，还有很多包能将描述性统计、回归模型的结果变成规范的表格样式，代表性的是 `modelsummary` 包；实验设计（表）在科研、生产中应用广泛，各种常用的实验设计，可以用 `DoE.base` 包实现。

本篇主要参阅 (张敬信, 2022), (冯国双, 2018), (贾俊平, 2018), (Chang, 2018), (Chang, 2018), 以及包文档，模板感谢 (黄湘云, 2021), (谢益辉, 2021).

参考文献

Chang, W. (2018). *R Graphics Cookbook*. O'Reilly, 2 edition.

冯国双 (2018). 白话统计. 电子工业出版社, 北京, 1 edition.

张敬信 (2022). *R 语言编程: 基于 tidyverse*. 人民邮电出版社, 北京.

谢益辉 (2021). *rmarkdown: Dynamic Documents for R*.

贾俊平 (2018). 统计学. 中国人民大学出版社, 北京, 7 edition.

黄湘云 (2021). *Github: R-Markdown-Template*.