实验一：R 基础和多元数据图示法

林泽钦 3160104013 统计 1601

目录

1 实验目的和要求 **1**

1.1 实验目的 1

1.2 实验内容 1

1.3 实验环境 2

2 实验过程与结果 **2**

2.1 正态、卡方随机数单变量分析 2

2.2 中心极限定理的模拟 4

2.3 多元数据分析 8

1 实验目的和要求

1.1 实验目的

通过本试验项目，使学生理解并掌握 R 软件包有关数据文件创建和整理的基本操作；熟悉 R 语言的基本，如构建向量、矩阵，数据表单等；单变量数据分析，单变量描述统计量来进行分析；轮廓图、雷达图、调和曲线图和散布图矩阵。

- 建立 R 数据集
- 单变量数据包括集中趋势的特征值：均值、众数、中位数等。；离散趋势的特征值：标准差、方差，(0.05, 0.95, 0.025, 0.975,0.1, 0.9) 分位数。
- 利用随机数分析样本容量与模拟次数对中心极限定理的影响，通过直方图和密度曲线图来刻画逼近正态分布的效果。
- 多元数据分析图示，轮廓图、雷达图、调和曲线图和散布图矩阵。

综合分析数据，写出分析报告。

1.2 实验内容

- 分别产生 100 个 N(0,1) 分布和卡方、自由度为 5 分布的随机数，进行单变量的分析。
- 附表中的数据用于多元数据分析。

1.3 实验环境

- R-3.5.1
- RStudio

2 实验过程与结果

2.1 正态、卡方随机数单变量分析

单变量数据包括集中趋势的特征值以及离散趋势的特征值。方便起见，将计算数据的数字特征编写为函数。此外，R 没有自带众数的计算函数，所以需要自己编写一个 getMode 函数用于计算众数。

```
# calculate the mode of data
getMode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}

# mean, median and mode of data
centerSummary <- function(v) {
    vMean <- mean(v)
    vMedian <- median(v)
    vMode <- getMode(v)
    vSummary <- list(mean=vMean, median=vMedian, mode=vMode)
    vSummary
}
```

```r
# standard deviation, variance and quantile of data
discreteSummary <- function(v){
    vSd <- sd(v)
    vVar <- vSd^2
    q <- c(0.05, 0.95, 0.025, 0.975, 0.1, 0.9)
    vQuantile <- quantile(v, q)
    vSummary <- list(sd=vSd, var=vVar, quantile=vQuantile)
    vSummary
}
```

之后只要生成正态、卡方随机数并调用上述函数进行单变量分析即可。下面是正态随机数的结果：

```r
# normal random numbers
normalData <- rnorm(100, mean = 0, sd = 1)
centerSummary(normalData)
```

```
## $mean
## [1] -0.1644051
##
## $median
## [1] -0.2708262
##
## $mode
## [1] 1.129685
```

```r
discreteSummary(normalData)
```

```
## $sd
## [1] 1.02904
##
## $var
## [1] 1.058924
##
```

```
## $quantile
##         5%       95%      2.5%     97.5%       10%       90%
## -1.742131  1.499418 -2.209280  1.754974 -1.358877  1.199819
```

下面是卡方随机数的结果：

```
# chi-square random numbers
chiData <- rchisq(100, 5, ncp = 0)
centerSummary(chiData)
```

```
## $mean
## [1] 4.908967
##
## $median
## [1] 4.067282
##
## $mode
## [1] 3.446424
```

```
discreteSummary(chiData)
```

```
## $sd
## [1] 3.236862
##
## $var
## [1] 10.47728
##
## $quantile
##         5%       95%      2.5%     97.5%       10%       90%
##  1.141941 10.527272  1.024692 13.070670  1.703694  9.494391
```

2.2 中心极限定理的模拟

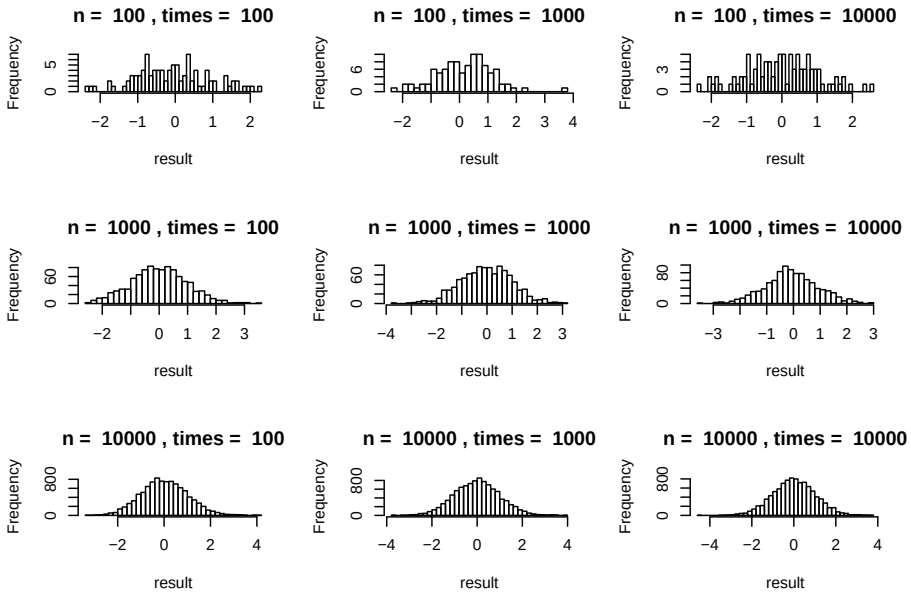
实验中使用的是卡方随机变量来模拟中心极限定理。假设有随机向量 $\xi = (\xi_1, ..., \xi_n)'$, 其每个分量独立且都服从 $\chi^2(5)$ 分布。对该随机向量进行 t

次抽样（模拟），得到样本 $\xi^{(1)},...,\xi^{(t)}$。根据 Lindeberg-Levy 中心极限定理，对每个 $i=1,...,n$，都有 $(\sum_{j=1}^{t}\xi^{(j)}-5t)/\sqrt{10t}$ 依分布收敛于 $N(0,1)$。下面的变量用于记录模拟的变量个数以及模拟次数。

```
n <- c(100, 1000, 10000)
t <- c(100, 1000, 10000)
```

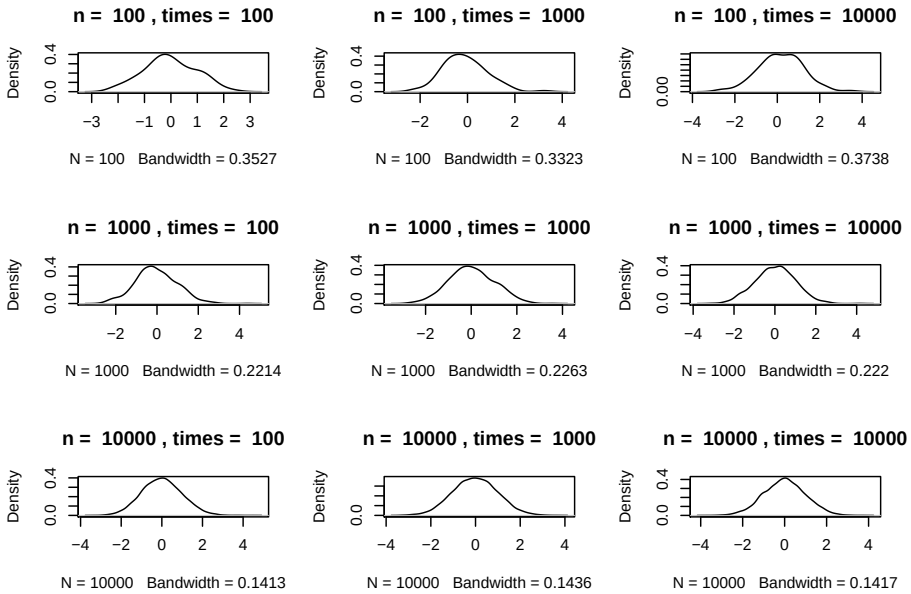
模拟并绘制直方图：

```
# split the screen into 3*3 pieces
par(mfrow=c(3,3))

for(i in 1:3) {
    for(j in 1:3) {
        sum <- rep(0, time = n[i])
        # generate chi-square random numbers
        for(k in 1:t[j])
            sum <- sum + rchisq(n[i], 5, ncp = 0)
        # normalize
        result <- (sum - 5 * t[j]) / sqrt(10 * t[j])
        hist(result, breaks = 40,
             main = paste("n = ", n[i], ", times = ", t[j]))
    }
}
```



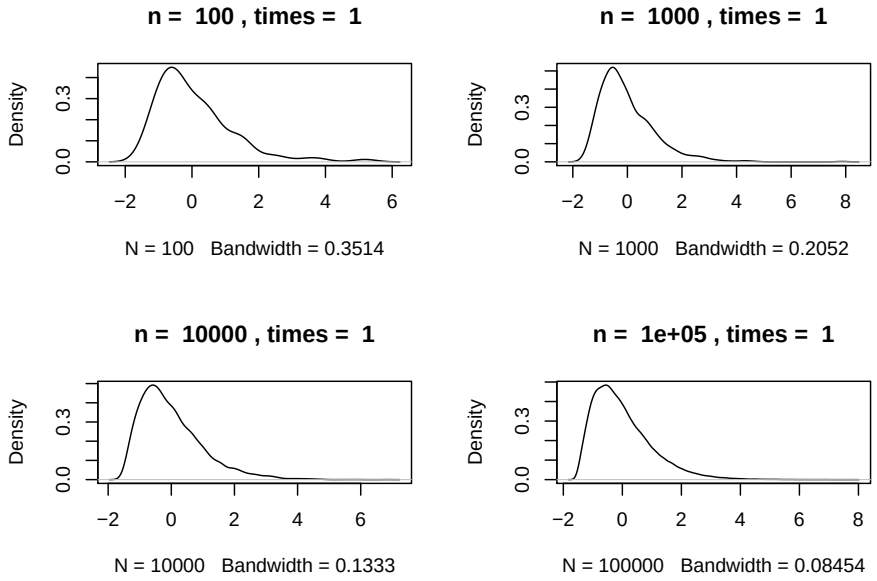
绘制密度曲线图：

```
# split the screen into 3*3 pieces
par(mfrow=c(3,3))

for(i in 1:3) {
    for(j in 1:3) {
        sum <- rep(0, time = n[i])
        # generate chi-square random numbers
        for(k in 1:t[j])
            sum <- sum + rchisq(n[i], 5, ncp = 0)
        # normalize
        result <- (sum - 5 * t[j]) / sqrt(10 * t[j])
        plot(density(result),
             main = paste("n = ", n[i], ", times = ", t[j]))
    }
}
```



从以上结果可以看出，模拟次数与变量数目的增加都会使得得到的分布越来越接近于标准正态分布。理论上来说，模拟次数的增大会使得每一个变量都越来越接近正态分布，而变量数目的增多则是使根据样本得到的经验分布曲线越来越接近于变量的原始分布。可以预见，如果模拟次数特别少，变量数目的增多并不会使得分布趋近于正态分布。比如在极端情况下，只模拟一次，那么 $n \to \infty$ 时，得到的应该是 $(\chi^2(5) - 5)/\sqrt{10}$ 的分布。

```
# split the screen into 3*3 pieces
par(mfrow=c(2, 2))

for(i in 1:4) {
    # generate chi-square random numbers
    sum <- rchisq(10^(1+i), 5, ncp = 0)
    # normalize
    result <- (sum - 5) / sqrt(10)
    plot(density(result),
         main = paste("n = ", 10^(1+i), ", times = ", 1))
}
```



而如果模拟次数足够但变量数目不足，就会导致得到的经验分布不能很好地刻画变量的真实分布。这相当于对标准正态总体取样时样本过少，那么得到的经验分布就有较大的不稳定性。上述模拟中 $n = 100, times = 10000$ 的情形就表现出了这个情况。

2.3 多元数据分析

首先是数据框的导入，这里使用了 csv 文件来导入。为了避免中文编码带来的不必要的问题，csv 文件中的中文名称被修改为英文。另外由于变量名称过长，程序中使用了缩写。

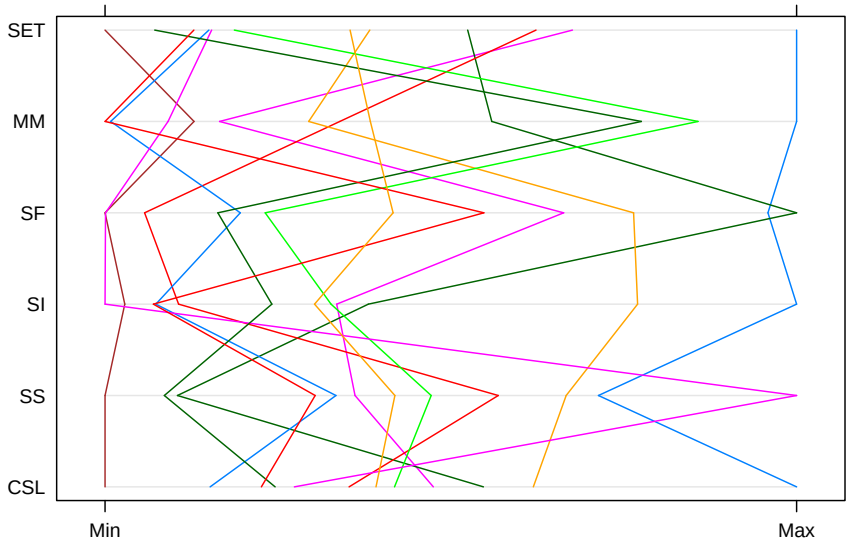
- CSL: Citizen Scientific Literacy
- SS: Science Staff
- SI: Science Infrastructure
- SF: Science Funding
- MM: Mass Media
- SET: Science Education and Training

```
tmp <- read.csv("Pro1Data.csv")
ScienceData = data.frame(
  CSL=tmp$Citizen.Scientific.Literacy,
  SS=tmp$Science.staff,
  SI=tmp$Science.infrastructure,
  SF=tmp$Science.funding,
  MM=tmp$Mass.media,
  SET=tmp$Science.education.and.training,
  row.names=tmp$Area)
```

2.3.1 轮廓图

这里调用了 lattice 包的 parallelplot 函数。

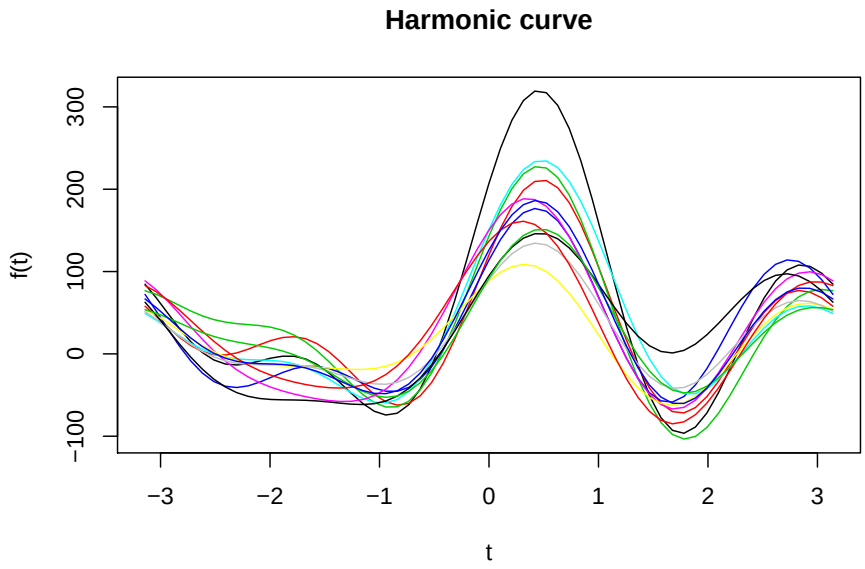
```
library(lattice)
parallelplot(ScienceData)
```



2.3.2 调和函数图

调和函数图的绘制借助了下面这个函数。此函数的代码参考了《R 软件与统计建模》一书。

```
unison <- function(x){
  if (is.data.frame(x) == TRUE)
    x <- as.matrix(x)
  t <- seq(-pi, pi, pi/30)
  m <- nrow(x); n<-ncol(x)
  f <- array(0, c(m,length(t)))
  for(i in 1:m){
    f[i,] <- x[i,1]/sqrt(2)
    for( j in 2:n){
      if (j%%2 == 0)
        f[i,] <- f[i,]+x[i,j]*sin(j/2*t)
      else
        f[i,] <- f[i,]+x[i,j]*cos(j%/%2*t)
    }
  }
  plot(c(-pi,pi), c(min(f), max(f)), type = "n",
       main = "Harmonic curve",
       xlab = "t", ylab = "f(t)")
  for(i in 1:m) lines(t, f[i,] , col = i)
}

unison(ScienceData)
```



2.3.3 雷达图

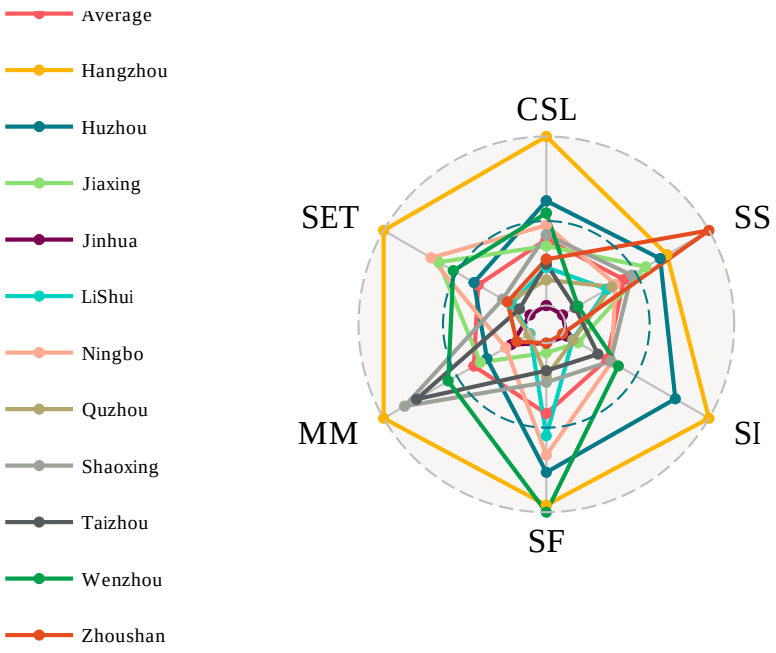
调用了 ggradar 包，参考 https://github.com/ricardo-bion/ggradar 。首先是相关包的调用：

```
library(ggradar)
suppressPackageStartupMessages(library(dplyr))
library(scales)
library(tibble)
library(extrafont)
extrafont::font_import(pattern = 'Circular', prompt=FALSE)
```

之后需要对数据的格式进行稍微修改，然后就可以绘制出雷达图：

```
ScienceData %>%
  rownames_to_column( var = "group" ) %>%
  mutate_at(vars(-group),funs(rescale)) ->
  ScienceData_Radar
```

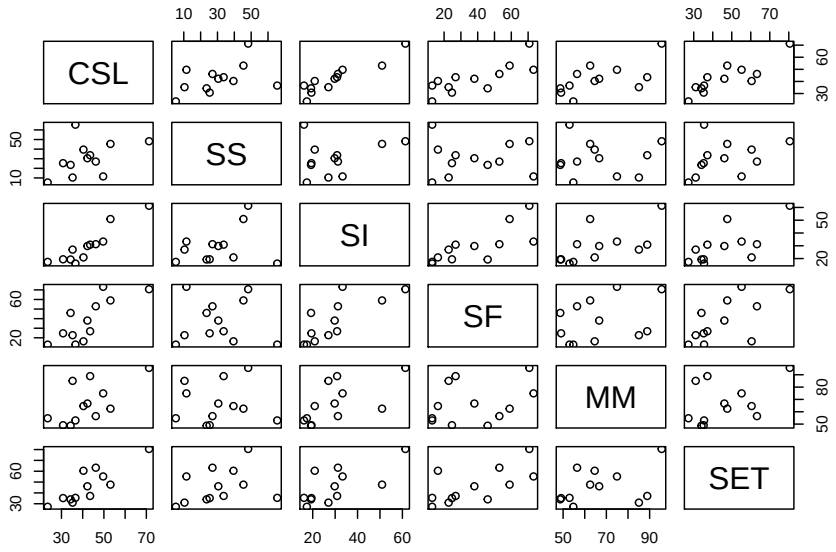
```
ggradar(ScienceData_Radar,
        group.line.width = 1, group.point.size = 2,
        grid.label.size = 4, axis.label.size = 6,
        legend.text.size = 10, label.gridline.min=FALSE,
        label.gridline.mid=FALSE, label.gridline.max=FALSE)
```



2.3.4 散点矩阵图

调用内置的 pairs 函数。

```
pairs(ScienceData)
```


CSL
SS
SI
SF
MM
SET
10 30 50
20 40 60
30 50 70
10 50
20 60
30 60
20 50
50 80
30 50 70
20 40 60
50 70 90