R在医学统计中的应用简介

录制视频清单:

R软件的安装使用8.1

1. 安装、运行R软件

   安装网址 操作录屏

2. R软件命令窗口及菜单

   菜单介绍：文件、程序包、HELP菜单

3. 命令行窗口交互及输出

```
x<-c(12,58,24,78,66,25,38,45)  #产生一列数据，放入x变量
print("x=");x
print("mean=");mean(x)
print("2+2=");2+2          #计算2+2
a<-2;b=4          #把2存储到变量a中（也可用= 或2->a代替）；把4存入变量b中
c<-a+b            #把a+b结果存入变量c中
print("c=");c   #显示变量c中存储的数据
```

   R用作计算器

```
2+3-1
2*3/4      #表达式
2^3       # 2的3次方
sqrt(10)   # 10的平方根
10^(1/3)   # 10开三次方
choose(5,2)   # 5选2组合
factorial(5)   # 5!阶乘
(choose(6,2)*7*3 + 6*choose(7,2)*3 + 6*7*choose(3,2))/choose(16,4)
abs(-10)    # 求绝对值
exp(1)      # 指数函数
log(3)      # 自然对数
pi
sin(pi/2)     #三角函数之正弦
asin(1)/pi    # 反正弦
log(100)/log(10)
```

4. 知识拓展—R图形界面程序

   Rstudio 和Rcmar图形界面的安装使用方法

   下载地址：https://www.rstudio.com

R常用命令及统计函数简介8.2

常用统计函数

max(x) 返回向量x中最大的元素

min(x) 返回向量x中最小的元素

which.max(x) 返回向量x中最大元素的下标

which.min(x) 返回向量x中最小元素的下标

mean(x) 计算样本(向量)x的均值

median(x) 计算样本(向量)x的中位数

mad(x) 计算中位绝对离差

var(x) 计算样本(向量)x的方差

sd(x) 计算向量x的标准差

range(x) 返回长度为2的向量: c(min(x) max(x))

IQR(x) 计算样本的四分位数极差

Quantile(x) 计算样本常用的分位数3

summary(x) 计算常用的描述性统计量(最小、最大、平均值、中位数和四分位数)

length(x) 返回向量x的长度

sum(x) 给出向量x的总和

prod(x) 给出向量x的乘积

rev(x) 取向量x的逆序

sort(x) 将向量x按升序排序, 选项decreasing=TRUE表示降序

order(x) 返回x的秩(升序), 选项decreasing=TRUE得到降序的秩

rank(x) 返回x的秩

var(x,y) 计算样本(向量)x与y的协方差

cov(x, y) 计算样本(向量)x与y的协方差

cor(x, y) 计算样本(向量)x与y的相关系数

在控制台输入:

```
.packages(all.available=TRUE)  #列出所有所有已安装的包
library(MASS)      #调用NASS包
getwd()              #获取当前工作目录
#setwd("d:/data")        #设置当前工作目录为 d:/data
wd<-getwd()
if (!is.null(wd)) setwd(wd)
ls()                 #显示内存中的对象
rm()                 #删除内存中对象，。   如:  > rm(r)
rm(list=ls())        #这个命令删除全部对象
help(lm)             #也可以这样使用
#?lm                 #显示关于lm的帮助
```

```
gl(5,3,15)            #5个水平，每个水平重复3次，共计15个元素
rep(1:3,each=5)       #重复1~3,5次，对比rep(1:3,5)
expand.grid(h=c(60,80), w=c(100, 300), sex=c("Male","Female"))
sleep                 #睡眠数据--内置数据
```

熟悉R的工作方式

以表格的形式编辑数据，edit()、fix()函数的使用

```
x=c(1:20)                 #变量x
y=c(20:1)                 #变量y
z=data.frame(x,y)         #由x、y组成数据框z
z                       #显示z中存放的数据
z<-edit(z)                #以表格形式编辑z中的数据并存储到z变量
```

均数、标准差的计算

```
x <- c(1,2,3,4,5,6,7,8,9,10)
sum(x)       #计算x的合计
mean(x)      #计算x的均值
sd(x)         #计算x的标准差
var(x)        #计算x的方差
sort(x ,decreasing=TRUE)          #对数据进行降序排序
#rnorm(n,mean,sd)     #产生n个均数为mean，标准差为sd的正太分布数据
```

利用apply()函数计算多列或多行数据的均数、标准差等描述统计指标。

apply(X, MARGIN, FUN, ...) 函数的参数：

X---向量、矩阵
MARGIN 1表示矩阵行，2表示矩阵列，也可以是c(1,2)表示行和列。
FUN—函数名

以swiss内部数据为例，计算描述统计指标：

```
head(swiss)                  #  显示头部数据
apply(swiss,2,mean)          #计算swiss数据框中每一列的均数
apply(swiss,2,sd)            #计算swiss数据框中每一列的标准差
apply(swiss,2,min)           #计算swiss数据框中每一列的最小值
apply(swiss,2,max)           #计算swiss数据框中每一列的最大值
apply(swiss,2,sum)           #计算swiss数据框中每一列的合计值
apply(swiss,2,summary)       #计算Swiss数据框中每一列的描述统计
```

交互命令：

启动R

R version 4.0.2 (2020-06-22) -- "Taking Off Again"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R是自由软件，不带任何担保。
在某些条件下你可以将其自由散布。
用'license()'或'licence()'来看散布的详细条件。

R是个合作计划，有许多人为之做出了贡献.
用'contributors()'来看合作者的详细情况
用'citation()'会告诉你如何在出版物中正确地引用R或R程序包。

用'demo()'来看一些示范程序，用'help()'来阅读在线帮助文件，或
用'help.start()'通过HTML浏览器来看帮助文件。
用'q()'退出R.

>

help.start(),网页式帮助信息

输入 练习代码：

生成两个伪正态随机数向量 x 和 y，进行回归分析和作图。

```
x <- rnorm(50)
y <- rnorm(x)
plot(x, y)   #画二维散点图。一个图形窗口会自动出现。
ls()         #查看当前工作空间里面的 R 对象。
rm(x, y)     #去掉不再需要的对象。(清空)。
x <- 1:20     #等价于 x = (1, 2, ..., 20)。
w <- 1 + sqrt(x)/2  #标准差的`权重'向量。
dummy <- data.frame(x=x, y= x + rnorm(x)*w)
dummy               #创建一个由x 和 y构成的双列数据框，查看它们。
fm <- lm(y ~ x, data=dummy)
summary(fm)          #拟合 y 对 x 的简单线性回归，查看 分析结果。
fm1 <- lm(y ~ x, data=dummy, weight=1/w^2)
summary(fm1)           #现在我们已经知道标准差，做一个加权回归。
attach(dummy)          #让数据框中的列项可以像一般的变量那样使用。
lrf <- lowess(x, y)    #做一个非参局部回归。
plot(x, y)             #标准散点图。
lines(x, lrf$y)        #增加局部回归曲线。
abline(0, 1, lty=3)    #真正的回归曲线：(截距 0，斜率 1)。
abline(coef(fm))       #无权重回归曲线。
abline(coef(fm1), col = "red")      #加权回归曲线。
abline(h=0,col=3)                  #加一条水平线，线颜色为3（绿色）
detach()               #将数据框从搜索路径中去除。
plot(fitted(fm), resid(fm),
xlab="Fitted values",
ylab="Residuals",
main="Residuals vs Fitted")         #回归数据估计值与残差数据作图。
rm(fm, fm1, lrf, x, dummy)          #再次清空对象，任务结束。
```

读取文件中的数据

```
#read.table()的使用：
library(MASS)                  # 载入包 "MASS"，包中含有数据集"hills"
write.table(hills,file="test.txt")     #把hills数据集中的数据存入文件data.txt
test.data<- read.table("test.txt",header=T)  #header=T表示将数据的第一行作为标题
head(test.data) #显示test.data中的头部数据
data <- read.table(file=file.choose(),header=T)
tail(data)       #显示读取数据的尾部数据
#可以弹出对话框，选择要读入的文件。

#read.csv()读取tuition.csv文件中的数据：
#data<-read.csv(file=file.choose(),header=T)
#data
#summary(data)        #显示data数据概要统计。
```

tidyverse包是安装使用

请按下列步骤完成数据筛选、过滤、整理、去除缺失值、分组计算工作。

```
#install.packages("tidyverse")       #安装tidyverse 包
require("tidyverse")                          #调用tidyverse 包
#data()                              #显示R自身带的可用数据集
#view(starwars)                #显示starvars数据框数据
head(starwars)
starwars %>%                          #建立数据管道
  select(sex,height,species,mass) %>%  #筛选starvars中4个变量
  filter(species=="Human") %>%         #过滤出species=="Human"的数据
  na.omit() %>%                        #剔除缺失值
  mutate(height=height/100) %>%        #生成新变量 height
  mutate(BMI=mass/height^2) %>%        #生成新变量 BMI
  group_by(sex) %>%                              #以sex为分组变量分组
  summarise(Average_BMI=mean(BMI))     #计算各组体重指数的平均值
```

统计分析实例 8.3

通过R自带数据和统计教材的计算实例，学习R在常用统计分析中的使用方法。
1．描述性统计
对MASS package包中hills数据集的变量进行描述性统计计算（均数、标准差、最大、最小值、百分位数计算）。学会描述性统计计算的函数summary()使用。
2．统计推断
本节的学习需要拥有相应的统计知识。使用R软件进行 t检验、方差分析、简单回归、卡方检验。

使用R进行数据的描述性统计分析

- 请回顾前面学习的常用统计函数

```
library(MASS)     # 载入包 "MASS"，包中含有数据集"hills"
head(hills)       #显示hills数据集中的数据
summary(hills)    # 列出"hills"中各变量的常用描述性统计结果
cor(hills)        # "hills"的相关矩阵
```

- 在R中用prop.test来估计比例的置信区间。
  例如：在n=100个人中，x=30个人喜欢看电影的人的比例p的置信度为95%的置信区间可以如下计算：

```
prop.test(30, 100, conf.level=0.95)    #结果为【21.45%，40.11%】
```

统计推断

t检验、配对t检验，成组t检验 方差分析ANOVA

单样本资料t检验

```
x <- c ( 36,32,31,25,28,36,40,32,41,26,35,35,32,87,33,35 )
t.test ( x , mu = 0 )
```

配对资料t检验对

```
x <- c( 78.1, 72.4 , 76.2, 74.3 ,77.4 ,78.4 , 76.0 , 75.5 , 76.7 , 77.3 )
y <- c( 79.1 , 81.0 , 77.3 , 79.1 , 80.0 , 79.1 , 79.1 , 77.3, 80.2 , 82.1 )
t.test(x,y,paired=TRUE)         #配对资料t检验
t.test(x,y,paired=FALSE)      #非配对资料t检验
```

成组数据（独立样本）t检验：

矩阵方式

```
a=c(2.3,2.5,2.5,2.7,2.5,2.8,3.3,3.5,3.6,3.8,3.9,4.6)  #a观测数据
b=rep(1:2,each=6)   #b分组变量，分1、2两组，每组6个数据
x=cbind(a,b)        #a,b组成矩阵x
x                   #显示x中的数据
t.test(a[b==1],a[b==2],var.equal=TRUE)   #等方差假定
t.test(a[b==1],a[b==2])                  #不等方差假定
```

Data.frame方式配对t检验

```
a=c(2.3,2.5,2.5,2.7,2.5,2.8,3.3,3.5,3.6,3.8,3.9,4.6)  #a观测数据
b=rep(1:2,each=6)          #b分组，每组6个数据
x<-data.frame(a,b)         #建立数据框x
x                          #显示x中的数据
with(x,t.test(a[b==1],a[b==2]))                  #不等方差假定
with(x,t.test(a[b==1],a[b==2],var.equal=TRUE))   #等方差假定
```

方差分析ANOVA

1.单因素方差分析

```
mouse<-data.frame(x=c(2,4,3,2,4,7,7,2,2,5,4,5,6,8,5,10,7,
    12,12,6,6,7,11,6,6,7,9,5,5,10,6,3,10),
    a=factor(c(rep(1,11),rep(2,10),rep(3,12))))
mouse.aov<-aov(x~a,data=mouse)
summary(mouse.aov)
boxplot(x~a,data=mouse,col="red")       #均数对比箱式图
```

2.双因素方差分析

```
x<-c(2.21,2.32,3.15,1.86,2.56,1.98,2.37,2.88,3.05,3.42,2.91,2.64,
     3.67,3.29, 2.45 ,2.74 ,3.15 ,3.44 ,2.61, 2.86 ,4.25 ,4.56 ,
     4.33, 3.89, 3.78 ,4.62,4.71, 3.56, 3.77, 4.23)
grp=gl(3,10)                    #处理组标识
block=gl(10,1,30)               #区组标识
x                               #显示原始数据x
y=data.frame(x,grp,block)
summary(aov(x~grp+block,data=y))
```

简单回归

```
x <-c( 0.1,0.11,0.12,0.13, 0.14,0.15,0.16,0.17,0.18,0.20,0.21,0.23)
y <-c(42,43.5,45,45.5,45,47.5,49,53,50,55,55,60)
lm(y~x)
summary(lm (y ~ x ))
```

卡方检验

```
x <- c ( 60 , 3, 32 , 11 )
dim ( x ) < - c ( 2 , 2 )
chisq.test( x )
```

图形绘制 8.4

图形工具是 R语言里面一个非常重要和多用途的组成部分。 我们可以用这些图形工具显示各种各样的统计图并且创建一些全新的图。图形工具既可交互式使用，也可以批处理使用。在许多情况下，交互式使用是最有效的。 打开R 时，它会启动一个图形设备驱动 （device driver）。该驱动会自动打开特定的图形窗口,以显示交互式的图片，这些图片可以存储到指定文件中。

R绘图功能演示

```
demo()
demo(graphics)
demo(pursp)
```

绘图函数的参数介绍:

绘图函数的参数：
add= F 逻辑型参数，若为T，表示在原图的添加图形。
type="" p点，l 线，b点线，h竖直线，n空白，o将点覆盖在线上。
xlab="", ylab="" 字符型参数，用于显示特定的 x和y轴的标签。
xlim=c(0,10), ylim=c( )指定轴的刻度（数值）上下限。
main="" 字符型参数，图形标题，以大字体显示于图形顶部。
sub="" 字符型参数，子标题，以小字体显示于 x轴下部
col=" " 指定图形颜色。
col.axis=" " ， col.lab=" " ， col.main=" "，col.sub=" "指定相关元素的颜色。颜色函数colors()共657种取值。
例如：

```
x <- c(23,56,89,64,25)
pie(x,main="Pie Chart",col.main="blue")      #设置标题及颜色
```

绘图颜色处理

色彩的处理
① colors()函数
colors() 可得到该函数的 657个取值，这些值常被给bg, col等属性赋值。

```
par(bg="mistyrose")                          #指定作图的背景
plot(1:10,1:10,type="l",col="mediumblue")    #绘出彩色的线
```

② 颜色渐变函数

```
rainbow(100)              #函数指定七彩虹颜色：红橙黄绿蓝靛紫，
heat.colors(100)          #函数用来指定颜色由红变到橙再到白，
terrain.colors(100)       #函数用来指定颜色由绿色变到棕色再到白色，
topo.colors(100)          #函数用来指定颜色由蓝色变到绿色再到白色，
cm.colors(100)            #函数用来指定颜色由青色变到紫色再到白色
```

各种颜色的直方图绘制：

```
x=runif(1000,-250,250)      #后面要使用的数据(10000个均匀函数随机数)
hist(x,breaks=seq(-250,250,5),col=rainbow(100))
hist(x,breaks=seq(-250,250,5),col=heat.colors(100))
hist(x,breaks=seq(-250,250,5),col=terrain.colors(100))
hist(x,breaks=seq(-250,250,5),col=topo.colors(100))
hist(x,breaks=seq(-250,250,5),col=cm.colors(100))
```

常用绘图函数的使用方法

```
library(MASS)        #调用MASS数据集
head(hills)                #显示hills 爬山数据
pairs(hills)     # "hills"中各变量间散点图
x<-c(23,56,58,42,66,14)   #第一个扇区23/(23+56+58+42+66+14),以此类推……
pie(x)     #输出饼图
```

plot()

```
x<-rnorm(50); y<-rnorm(50)
plot(x,y)
```

plot()箱式图

```
y<-c(1600,1610,1650,1680,1700,1700,1780,1500,1640,1400,1700,1750,1640,1550,1568
     ,1620,1640,1600,1740,1800,1510,1520,1530,1570,1640,1600)
f<-factor(c(rep(1,7),rep(2,5),rep(3,8),rep(4,6)))
plot(f,y)          #观察图形输出
```

hist() 绘制直方图函数（略）

barplot() 泊松分布直条图、boxplot() 绘制箱式图函数

```
x<-rpois(100,5)                    #模拟泊松分布数据
#boxplot(x)                    #箱式图绘制
y<-table(x)                            #生成频数表
barplot(y,col=rainbow(20))   #绘制条形图
```

正态分布曲线绘制

```
x=c(-4:4)
curve(dnorm(x), -3, 3, main="Normal Curve")      #绘制N(0,1)正态分布曲线。
abline(v=0.4, col = 2)                                          #v-垂直线在X轴的位置0.4
                                                                #col颜色- 1黑色 2红色 3蓝色……
```

知识扩展：

回归分析脚本

对一批涂料进行研究，确定搅拌速度对杂质含量的影响，数据如表7-5所示，试进行回归分析。

rate：转速数据；imperity：杂质率数据

```
rate<-c(20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42)
impurity <-c(8.4, 9.5, 11.8, 10.4, 13.3, 14.8, 13.2, 14.7, 16.4, 16.5, 18.9,
18.5)
plot(impurity~rate)
reg<-lm(impurity~rate)
abline(reg,col="red")
plot(fitted(reg),resid(reg),main="Residual and Fitted Scatter")
abline(h=0,col=3)
qqnorm(resid(reg),main = "Residual Normal Q-Q figure")
summary(reg)
lm(formula = impurity ~ rate)
anova(lm(formula = impurity ~ rate))  # 回归方差分析
```

词云制作

安装必要的R包，jiebaR，wordclound2，tm

```
# install.packages("jiebaR")         #中文分词包
# install.packages("wordcloud2")     #安装词云包wordcloud2
library(wordcloud2)              #调用词云包
#wordcloud2(demoFreqC)           #绘制词云
#wordcloud2(demoFreq, size = 1,shape='star')                  #星型英文词云
wordcloud2(demoFreqC, size = 1.55,background='black')        #汉字词云
```

中文词云制作

用jiebaR分词，运行速度比用tm,tmcn,Rwordseg分词快很多！试下这段代码：

```
library(jiebaR)
library(wordcloud2)
#data <- read.csv("d.csv",encoding="UTF-8")
data <- readLines("d.csv",encoding="UTF-8")
data <- unique(data)# 去除重复的数据
data <- gsub('[a-zA-Z0-9]',' ',data)
#data<-sample(data,50)
data <- gsub('[的是了不在上也对]',' ',data)
cutter=worker() #结巴分词实例cutter
text<-segment(data, cutter, mod = NULL)
data=freq(text)
text
#data
wordcloud2(data,shape = 'circle')
#wordcloud2(data,shape = 'star')
```

R参考文献

```
网站：https://cran.r-project.org/other-docs.html 有英文、中文PDF电子书。
```

当点击R命令交互窗口的帮助菜单时，可获得丰富的帮助文档。常见问题（FAQ）：

交互窗口“帮助”菜单中，有本地化的常见问题解答。在 CRAN 的网站上R-FAQ 会定期更新：https://cran.r-project.org/doc/FAQ/R-FAQ.html

在https://cran.r-project.org/manuals.html 网站上的R手册：包括：

- R简介 [R-intro.pdf]
- R 安装和管理 [R-admin.pdf]
- R 数据导入/导出 [R-data.pdf]
- 编写R扩展[R-exts.pdf]
- R语言（参考）定义[R-lang.pdf]

  这些文档有不同的格式（pdf、html、text）提供，文档都是英文版。可互联网搜索对应的中文版