L09-数据可视化

郑盼盼

2024-11-27

目录

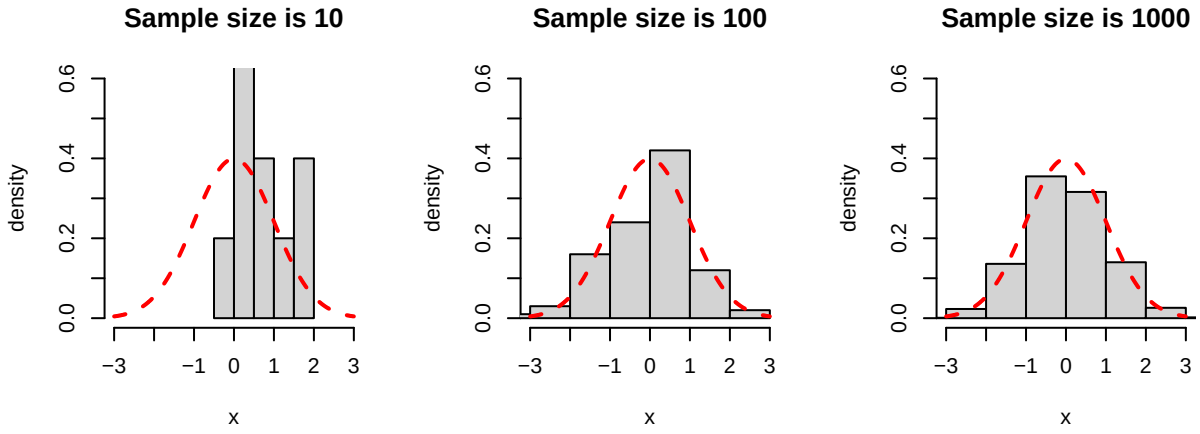
9.1 直方图的绘制

- hist(x, breaks, freq, col, main, ylab, xlab, …)
  - x 数据向量，用于生成直方图的数据
  - breaks 直方图的区间数或区间分隔点，
  - freq 是否以频数绘制 T 或以密度绘制 F
  - col 颜色 (color)
  - main 图的标题
  - xlab, ylab: x 轴和 y 轴的标签
  - xlim, `ylim`: x 轴和 y 轴的区间

9.1.1 基础绘制

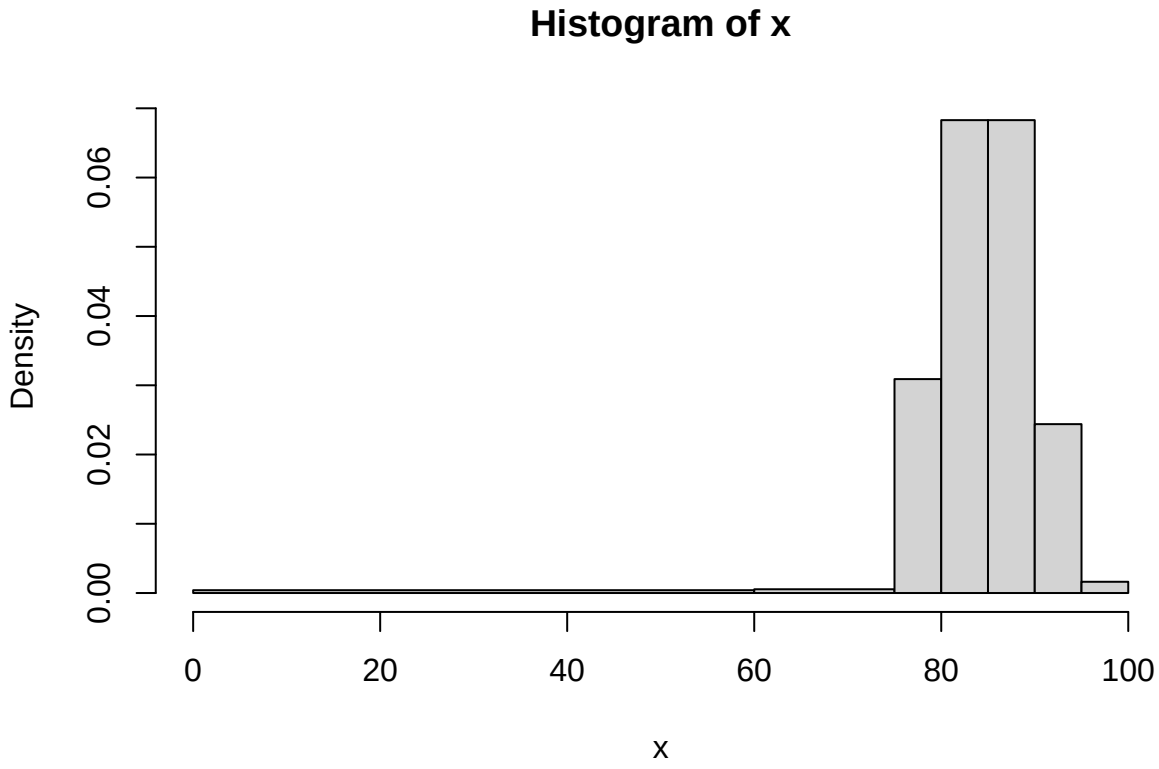
等距分组直方图 默认的绘制方法就是等距的直方图

```r
# 设置多张图片的布局
par(mfrow=c(1,3), pty="s")
xx <- seq(-3,3,by=0.001) # 定义一个向量用于之后密度曲线的绘制
# for 循环迭代不同的样本量
for (i in c(10,100,1000)) {
  x <- rnorm(i,0,1) # 生成样本量为 i 的服从标准正态分布的随机数
  # 绘制直方图
  hist(x, freq=F,
       xlim=c(-3,3), ylim=c(0,0.6),
       ylab="density", main=paste("Sample size is",i))
  # 使用 lines 函数往已有的图片中添加标准正态分布的概率密度曲线
  lines(xx, dnorm(xx,0,1), lty=2, col="red", lwd=2)
}
```



不等距直方图 我们可以通过 breaks 参数的设置，绘制不等距的直方图

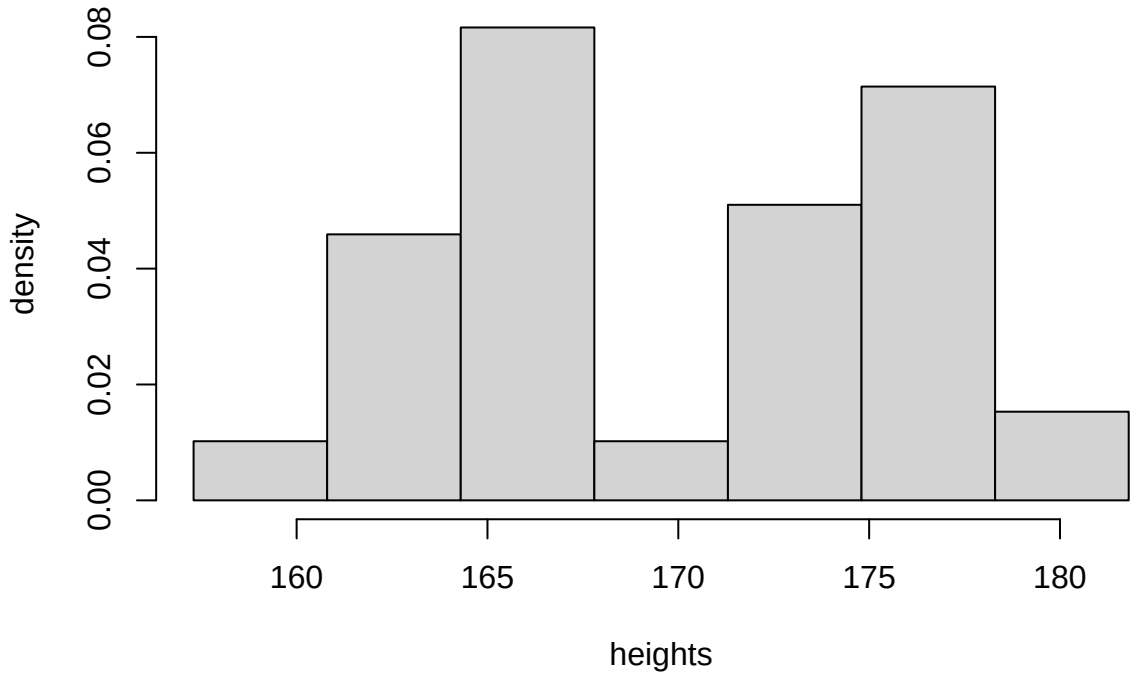
```r
x <- rnorm(120,mean=85, sd=5)
x <- c(x, runif(3,0,60))
x[x>100] <- 100
x[x<0] <- 0
hist(x, right=F, # 右闭左开区间分组
     freq=F,
     breaks = c(0,60,75,80,85,90,95,100) # 通过 breaks 来设置分隔点
     )
```



9.1.2 频率直方图的应用

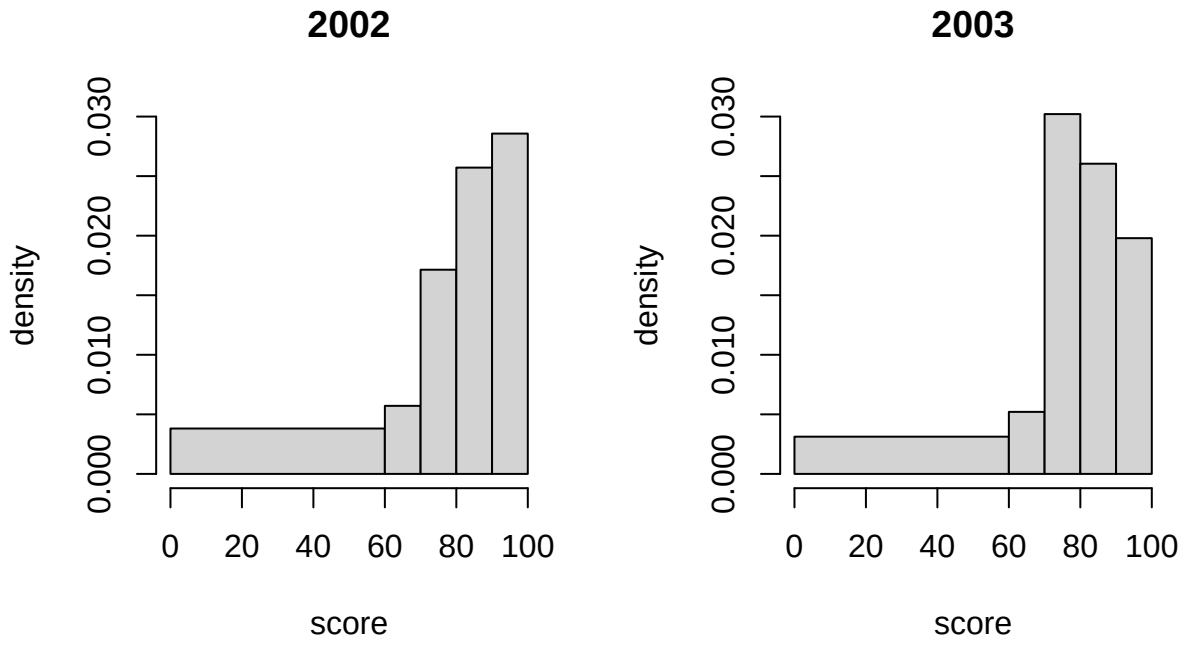
- P136

```
prob01 <- c(171.8,175.6,177.3,171.1,172.1,177.3,175.0,176.6,
     179.1,176.4,172.6,177.3,174.5,172.6,181.3,175.2,
     172.2,176.9,180.0,176.8,177.4,175.3,174.5,177.6,
     174.4,175.2,173.4,172.9,164.2,161.4,165.7,165.3,
     164.9,158.3,163.5,165.7,166.1,165.5,164.9,160.2,
     166.0,164.6,166.5,166.8,164.7, 166.0,164.0,164.0,
     163.9,162.1,165.7,168.7,163.3,163.5,167.2,165.1);
g <- seq(157.3, 181.8, 3.5);
hist(prob01, breaks=g, freq=F,
     main="", xlab="heights", ylab="density")
```



- P139-P140

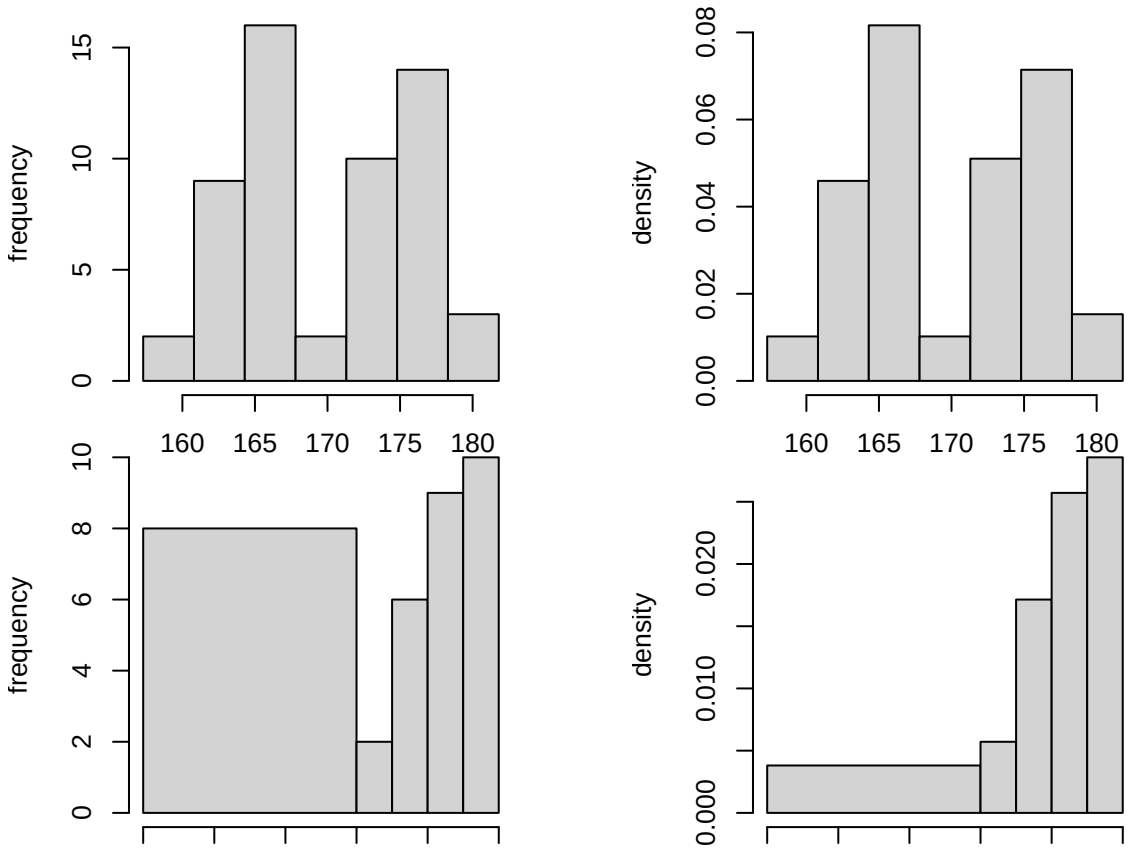
```
par(mfrow=c(1,2), pty="s")
prob02 <- c(58,91,80,99,99,71,91,47,39,50,48,98,68,72,75,60,
            93,98,59,54,84,90,87,82,80,74,72,90,80,88,80,84,
            90,48,76)
hist(prob02, right=F, freq=F,
     breaks= c(0,60,70,80,90,100),ylim=c(0,0.03),
     main="2002", ylab="density", xlab="score")
prob03 <- c(35,38,41,41,42,43,44,47,47,47,49,50,51,52,55,57,
            58,59,71,61,62,64,65,67,86,96,96,70,70,70,70,71,
            71,72,73,73,74,74,74,74,75,75,75,75,75,76,76,76,
            76,78,78,78,78,79,79,80,80,81,81,81,83,84,85,85,
            85,85,86,86,86,86,86,86,86,86,87,87,88,88,89,90,
            90,90,90,90,90,91,92,91,92,93,93,95,95,96,96,96)
hist(prob03, right=F, freq=F,
     breaks=c(0,60,70,80,90,100), ylim=c(0,0.03),
     main="2003", ylab="density", xlab="score")
```



9.1.3 分组数据条形图

更改 freq 为 freq=T 即可，对于**等距分组**的条形图，其依然能反映概率密度曲线的形态；而对于**不等距分组**的条形图，其无法反映概率密度曲线的形态；

```
par(mfrow=c(2,2),pty='s', pin=c(2,2))
hist(prob01,g, freq=T,
     ylab="frequency", main="")
hist(prob01,g, freq=F,
     ylab="density",main="")
hist(prob02, right=F, freq=T,
      breaks= c(0,60,70,80,90,100),
      main="", ylab="frequency", xlab="score") |> suppressWarnings()
hist(prob02, right=F, freq=F,
    breaks= c(0,60,70,80,90,100),
    main="", ylab="density", xlab="score")
```



9.2 离散型分布的可视化

9.2.1 条形图和饼状图

```
x<-c(" 女"," 男"," 女"," 女"," 女"," 女"," 女"," 男",
     " 女"," 女"," 女"," 男"," 男"," 女"," 男"," 女",
     " 女"," 女"," 女"," 女"," 男"," 男"," 女"," 男",
     " 女"," 男"," 男"," 男"," 男")
y <- as.factor(x) # 将向量转变成因子
u <- table(y)     # 计算变量 y 的分类汇总结果
u
```

```
## y
## 女 男
## 17 12
```

```r
names(u) <- c("male", "female")
u
```

```
##   male female
##     17     12
```

```r
addmargins(u) # 添加合计项
```

```
##   male female    Sum
##     17     12     29
```

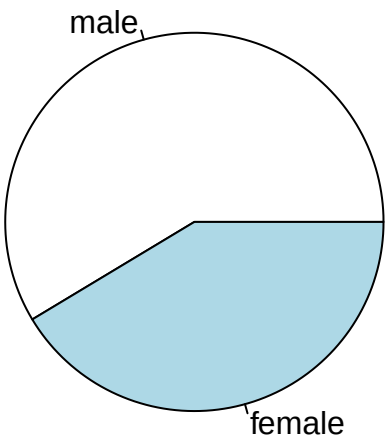
```r
v <- prop.table(u) # 统计各类别的频率
v
```

```
##      male    female
## 0.5862069 0.4137931
```

```r
addmargins(v)
```

```
##      male    female       Sum
## 0.5862069 0.4137931 1.0000000
```

```r
pie(v)
```




```r
barplot(u)
```

9.2.2 堆积条形图与并列条形图

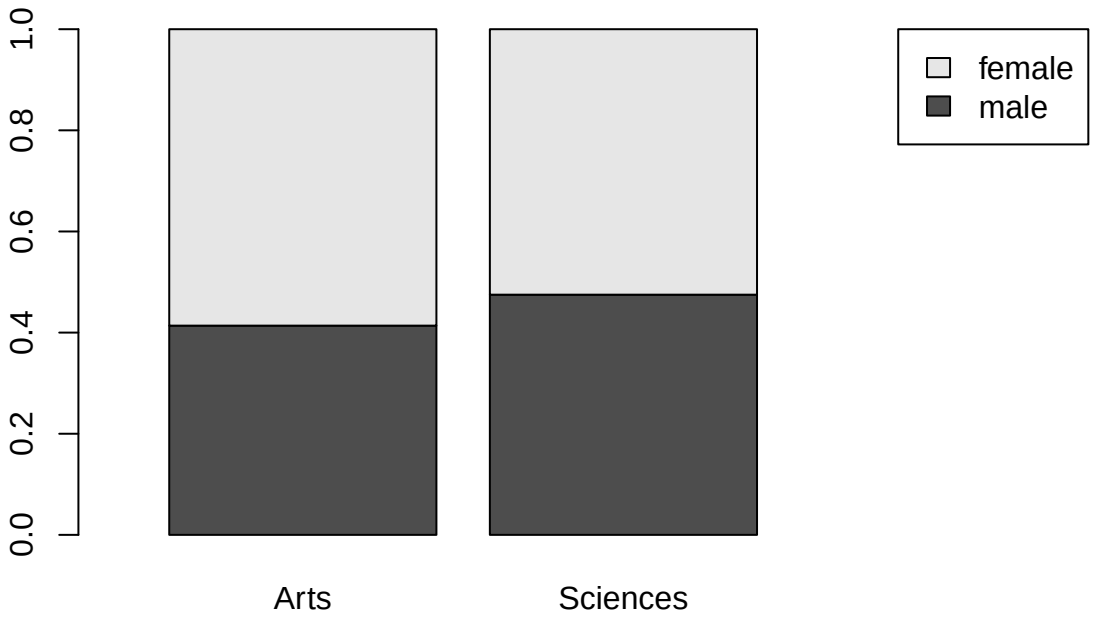
```
x <- matrix(c(12,19,17,21), nrow=2, ncol=2)
colnames(x) <- c("male", "female")
rownames(x) <- c("Arts", "Sciences")
y <- as.table(x)
prop.table(y, margin=2)
```

```
##               male    female
## Arts     0.3870968 0.4473684
## Sciences 0.6129032 0.5526316
```

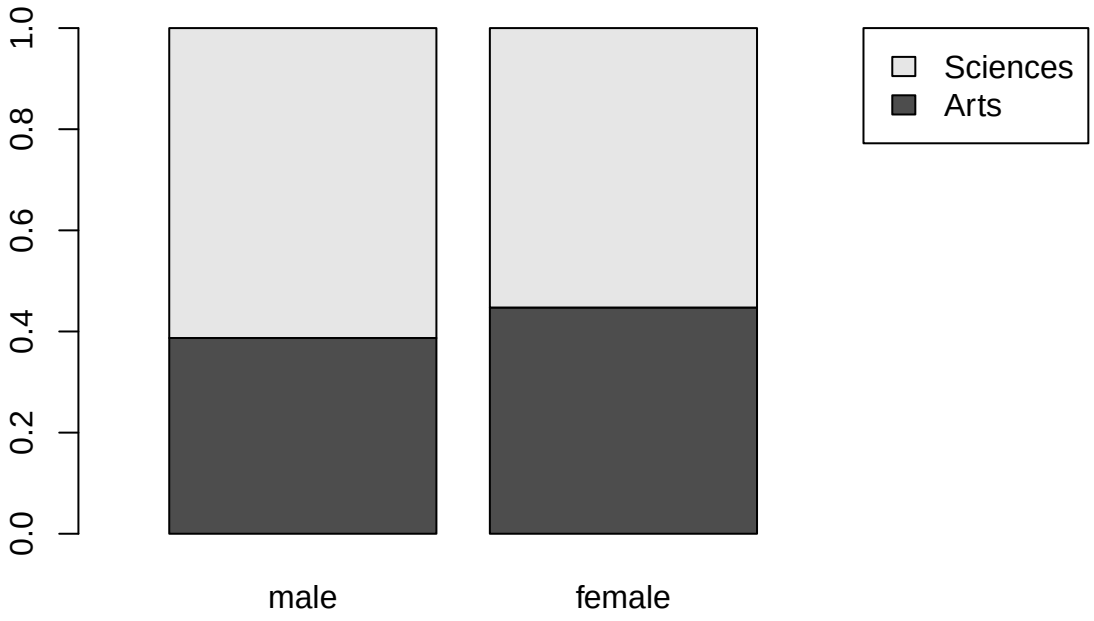
```
prop.table(t(y), margin=2)
```

```
##             Arts  Sciences
## male   0.4137931 0.4750000
## female 0.5862069 0.5250000
```

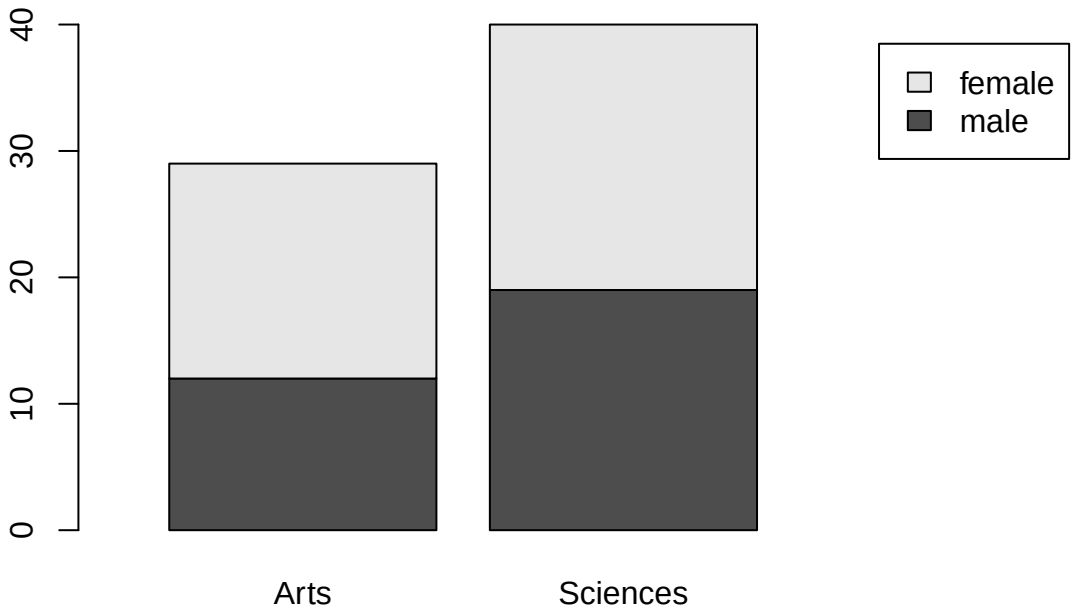
```
# 等高堆积条形图
barplot(prop.table(t(y),margin=2),
        xlim=c(0,3.5),
        legend.text=colnames(y),
        args.legend=list(x="topright"))
```



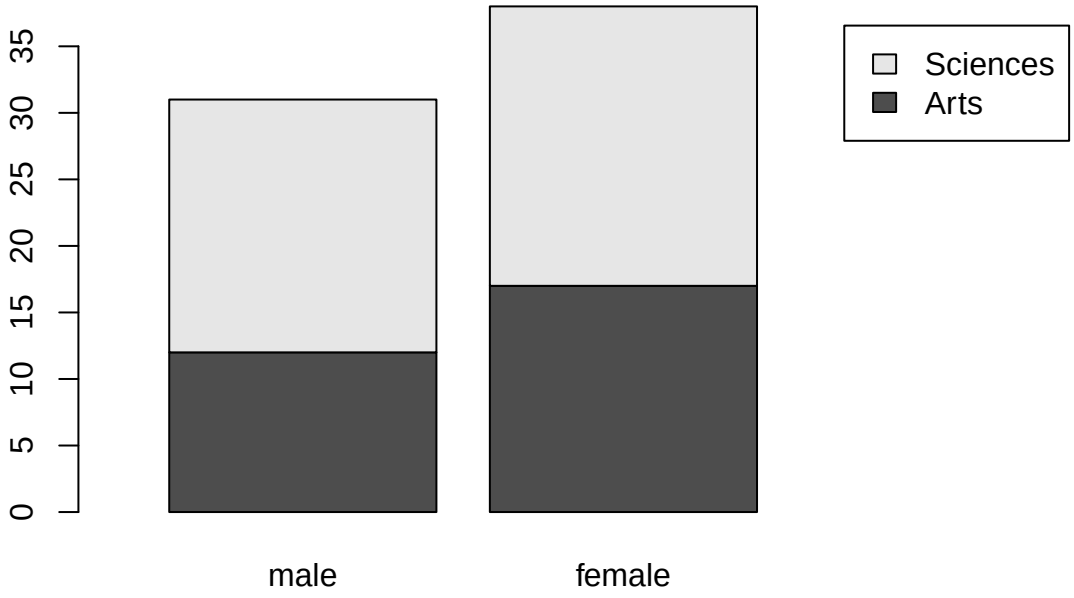

```
barplot(prop.table(y,margin=2),
        xlim=c(0,3.5),
        legend.text=rownames(y), args.legend = list(x="topright"))
```



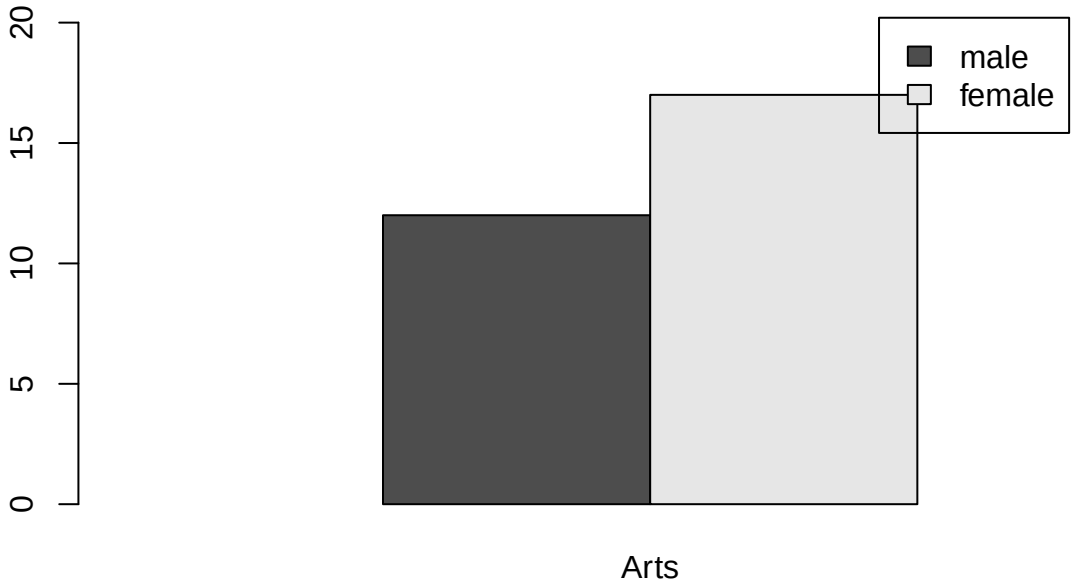

```
# 非等高堆积条形图
barplot(t(y), xlim=c(0,3.5), legend.text=colnames(y), args.legend="topright")
```




```
barplot(y, xlim=c(0,3.5), legend.text=rownames(y), args.legend="topright")
```




```
# 并列条形图
barplot(t(y), beside=T,xlim=c(0,3.5), legend.text=colnames(y), args.legend="topright")
```




```
barplot(y, beside=T,xlim=c(0,3.5), legend.text=rownames(y), args.legend="topright")
```

Questions

1. 模拟产生样本量为 20000 的标准正态分布，分别使用前 $20, 200, 2000$ 和 20000 个样本数据绘制频率直方图，分析直方图与标准正态分布密度函数之间的关系，解释为什么能用直方图认识密度函数。

2. 录入如下程序代码

```
myData <- data.frame(
  x = c(rep(" 文科",29), rep(" 理科", 40)), # 文理科生人数
  y = c(
    rep(" 男", 12),  # 文科生中男生人数
    rep(" 女", 17),  # 文科生中女生人数
    rep(" 男", 19),  # 理科生中男生人数
    rep(" 女", 21)   # 理科生中女生人数
  ))
```

解答如下问题：

1. 考察程序代码

   ```
   tmpT <- table(myData) #
   # 绘制等高条形图，男生比例对比
   tmpP <- prop.table(tmpT, margin=2) #
   barplot(tmpP,
    xlim=c(0,3.5), #
    legend.text=rownames(tmpP), #
    args.legend = list(x="topright") #
    )
   ```

   在 # 后面添加程序代码注解

2. 绘制 myData 的堆积条形图，并解释图像的含义。

3. 绘制 myData 并列条形图，并解释图像的含义。