第二次作业

李馨雨

2021/9/30

向量

现有 10 个人的期末考试成绩为

100, 65, 80, 79, 88, 95, 93, 35, 56, 68

1. 创建向量 x 来存储上述数据；

```
x <- c(100, 65, 80, 79, 88, 95, 93, 35, 56, 68)
```

2. 将 x 从小到大排序，并分别找出最大值，最小值、中位数和第三大的元素；

```
sort(x,decreasing=FALSE)
```

```
##  [1]  35  56  65  68  79  80  88  93  95 100
```

```
x[which.max(x)]   # 最大值
```

```
## [1] 100
```

```
x[which.min(x)]    # 最小值
```

```
## [1] 35
```

```
median(x)          # 中位数
```

```
## [1] 79.5
```

```
y <- sort(x,decreasing=TRUE)
y[3]     # 第三大的
```

```
## [1] 93
```

3. 计算平均值、标准差和方差；

```
mean(x)  # 平均值
```

```
## [1] 75.9
```

```
sd(x)    # 标准差
```

```
## [1] 20.16846
```

```
var(x)   # 方差
```

```
## [1] 406.7667
```

4. 计算及格的人数；

```
count <- 0
for(i in 1:10){
  if(x[i]>=60){
    count=count+1
  }
}
print(count)
```

```
## [1] 8
```

5. 给 x 命名，名字为学号，分别为 “PB1”, “PB2”, … “PB10”（提示，可以先创建以名字为元素的字符型向量 y，然后进行赋值）；

```
for(i in 1:10){
  y[i] <- paste("PB",i,sep="")
}
names(x)<-y
x
```

```
##  PB1  PB2  PB3  PB4  PB5  PB6  PB7  PB8  PB9 PB10
##  100   65   80   79   88   95   93   35   56   68
```

6. 提取奇数位置的元素并显示；

```
for(i in 1:10){
  if(i%%2==1){
    print(x[i])
  }
}
```

```
## PB1
## 100
## PB3
##  80
## PB5
##  88
## PB7
##  93
## PB9
##  56
```

7. 返回偶数（不包含缺失值）。

```
x[!is.na(x)&x%%2==0]
```

```
##  PB1  PB3  PB5  PB9 PB10
##  100   80   88   56   68
```

矩阵和数组

1. 创建一个 10×10 的矩阵 `m`，其对角线元素分别为 1, 2, ..., 10；

```
m <- diag(c(1,2,3,4,5,6,7,8,9,10))
print(m)
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]    1    0    0    0    0    0    0    0    0     0
##  [2,]    0    2    0    0    0    0    0    0    0     0
##  [3,]    0    0    3    0    0    0    0    0    0     0
##  [4,]    0    0    0    4    0    0    0    0    0     0
##  [5,]    0    0    0    0    5    0    0    0    0     0
##  [6,]    0    0    0    0    0    6    0    0    0     0
##  [7,]    0    0    0    0    0    0    7    0    0     0
##  [8,]    0    0    0    0    0    0    0    8    0     0
##  [9,]    0    0    0    0    0    0    0    0    9     0
## [10,]    0    0    0    0    0    0    0    0    0    10
```

2. 计算 `m` 的逆，并求解关于 y 的线性方程组 $my = x$；

```
solve(m)
```

```
##       [,1] [,2]      [,3] [,4] [,5]      [,6]      [,7]  [,8]      [,9] [,10]
##  [1,]    1  0.0 0.0000000 0.00  0.0 0.0000000 0.0000000 0.000 0.0000000   0.0
##  [2,]    0  0.5 0.0000000 0.00  0.0 0.0000000 0.0000000 0.000 0.0000000   0.0
##  [3,]    0  0.0 0.3333333 0.00  0.0 0.0000000 0.0000000 0.000 0.0000000   0.0
##  [4,]    0  0.0 0.0000000 0.25  0.0 0.0000000 0.0000000 0.000 0.0000000   0.0
```

```
##  [5,]    0  0.0 0.0000000 0.00  0.2 0.0000000 0.0000000 0.000 0.0000000   0.0
##  [6,]    0  0.0 0.0000000 0.00  0.0 0.1666667 0.0000000 0.000 0.0000000   0.0
##  [7,]    0  0.0 0.0000000 0.00  0.0 0.0000000 0.1428571 0.000 0.0000000   0.0
##  [8,]    0  0.0 0.0000000 0.00  0.0 0.0000000 0.0000000 0.125 0.0000000   0.0
##  [9,]    0  0.0 0.0000000 0.00  0.0 0.0000000 0.0000000 0.000 0.1111111   0.0
## [10,]    0  0.0 0.0000000 0.00  0.0 0.0000000 0.0000000 0.000 0.0000000   0.1
```

```
y <- solve(m,x)
print(y)
```

```
##  [1] 100.000000  32.500000  26.666667  19.750000  17.600000  15.833333
##  [7]  13.285714   4.375000   6.222222   6.800000
```

3. 将上述矩阵的第 2 行除对角线外的所有元素赋值为 100；

```
m[2,]<-100
m[2,2]<-2
print(m)
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]    1    0    0    0    0    0    0    0    0     0
##  [2,]  100    2  100  100  100  100  100  100  100   100
##  [3,]    0    0    3    0    0    0    0    0    0     0
##  [4,]    0    0    0    4    0    0    0    0    0     0
##  [5,]    0    0    0    0    5    0    0    0    0     0
##  [6,]    0    0    0    0    0    6    0    0    0     0
##  [7,]    0    0    0    0    0    0    7    0    0     0
##  [8,]    0    0    0    0    0    0    0    8    0     0
##  [9,]    0    0    0    0    0    0    0    0    9     0
## [10,]    0    0    0    0    0    0    0    0    0    10
```

4. 提取 m 的第 5 行第 9 列的元素并输出到命令行，然后将其重新赋值为 NA；

```
m_59 <- m[5,9]
print(m_59)
```

```
## [1] 0
```

```
m[5,9] <- NA
print(m[5,9])
```

```
## [1] NA
```

5. 给 m 的列命名为 “c1”, “c2”,…“c10”;

```
colnames(m)<- c("c1", "c2","c3","c4","c5","c6","c7","c8","c9","c10")
```

6. 给 m 增加一行，元素为上面定义的向量 x;

```
m <- rbind(m,x)
print(m)
```

```
##     c1 c2  c3  c4  c5  c6  c7  c8  c9 c10
##      1  0   0   0   0   0   0   0   0   0
##    100  2 100 100 100 100 100 100 100 100
##      0  0   3   0   0   0   0   0   0   0
##      0  0   0   4   0   0   0   0   0   0
##      0  0   0   0   5   0   0   0  NA   0
##      0  0   0   0   0   6   0   0   0   0
##      0  0   0   0   0   0   7   0   0   0
##      0  0   0   0   0   0   0   8   0   0
##      0  0   0   0   0   0   0   0   9   0
##      0  0   0   0   0   0   0   0   0  10
## x  100 65  80  79  88  95  93  35  56  68
```

7. 计算 m 的行数、列数，核对是否正确;

```
nrow(m)
```

```
## [1] 11
```

```
ncol(m)
```

```
## [1] 10
```

```
# 核对是否正确
all.equal(nrow(m),11)
```

```
## [1] TRUE
```

```
all.equal(ncol(m),10)
```

```
## [1] TRUE
```

8. 按行计算平均值，输出的结果不能包含 NA；

```
# 计算平均值时没有略掉 NA 值，相当于第五行平均值为 NA
rm <- rowMeans(m)
rm[!is.na(rm)]
```

```
##                                                 x
##  0.1 90.2  0.3  0.4  0.6  0.7  0.8  0.9  1.0 75.9
```

```
# 计算平均值时略掉 NA 值
rowMeans(m,na.rm=TRUE)
```

```
##
##  0.1000000 90.2000000  0.3000000  0.4000000  0.5555556  0.6000000  0.7000000
##                                           x
##  0.8000000  0.9000000  1.0000000 75.9000000
```

9. 构造一个 $10 \times 10 \times 3$ 的数组 arr，其中 arr[,,1] 为 m，arr[,,2] 为 2m，arr[,,3] 为 3m。

```
m<-diag(c(1,2,3,4,5,6,7,8,9,10))
m2<-m*2
m3<-m*3
m1<-cbind(m,m2,m3)
arr <- array(m1,dim=c(10,10,3))
print(arr)
```

```
## , , 1
##
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]    1    0    0    0    0    0    0    0    0     0
##  [2,]    0    2    0    0    0    0    0    0    0     0
##  [3,]    0    0    3    0    0    0    0    0    0     0
##  [4,]    0    0    0    4    0    0    0    0    0     0
##  [5,]    0    0    0    0    5    0    0    0    0     0
##  [6,]    0    0    0    0    0    6    0    0    0     0
##  [7,]    0    0    0    0    0    0    7    0    0     0
##  [8,]    0    0    0    0    0    0    0    8    0     0
##  [9,]    0    0    0    0    0    0    0    0    9     0
## [10,]    0    0    0    0    0    0    0    0    0    10
##
## , , 2
##
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]    2    0    0    0    0    0    0    0    0     0
##  [2,]    0    4    0    0    0    0    0    0    0     0
##  [3,]    0    0    6    0    0    0    0    0    0     0
##  [4,]    0    0    0    8    0    0    0    0    0     0
##  [5,]    0    0    0    0   10    0    0    0    0     0
##  [6,]    0    0    0    0    0   12    0    0    0     0
##  [7,]    0    0    0    0    0    0   14    0    0     0
```

```
##  [8,]    0    0    0    0    0    0    0   16    0     0
##  [9,]    0    0    0    0    0    0    0    0   18     0
## [10,]    0    0    0    0    0    0    0    0    0    20
##
## , , 3
##
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]    3    0    0    0    0    0    0    0    0     0
##  [2,]    0    6    0    0    0    0    0    0    0     0
##  [3,]    0    0    9    0    0    0    0    0    0     0
##  [4,]    0    0    0   12    0    0    0    0    0     0
##  [5,]    0    0    0    0   15    0    0    0    0     0
##  [6,]    0    0    0    0    0   18    0    0    0     0
##  [7,]    0    0    0    0    0    0   21    0    0     0
##  [8,]    0    0    0    0    0    0    0   24    0     0
##  [9,]    0    0    0    0    0    0    0    0   27     0
## [10,]    0    0    0    0    0    0    0    0    0    30
```

列表

构造一个列表 score，存放学生李明的成绩，如下：

```
score <- list(name=" 李明", id = "PB1", age=19, scores=c(85, 76, 90))
```

1. 提取年龄，并今年为记录时间计算其出生年份；

```
score <- list(name=" 李明", id = "PB1", age=19, scores=c(85, 76, 90))
age <- score[["age"]]
print(age)
```

```
## [1] 19
```

```
birthyear<- 2021-age
print(birthyear)
```

```
## [1] 2002
```

2. 最后一个元素展示的是语数英三科的成绩，提取出来记为 y, 并计算其平均成绩；

```
y<-score[[4]]
print(y)
```

```
## [1] 85 76 90
```

```
mean(y)
```

```
## [1] 83.66667
```

3. 增加一个元素记录是否是优秀学生，其中优秀学生的定义为平均成绩大于 85 分。

```
if(mean(score$scores)>85){
  score$是否是优秀学生<-" 是"
}else
{
  score$是否是优秀学生<-" 否"
}
score
```

```
## $name
## [1] "李明"
##
## $id
## [1] "PB1"
```

```
##
## $age
## [1] 19
##
## $scores
## [1] 85 76 90
##
## $是否是优秀学生
## [1] "否"
```

数据框和处理数据

covid19.csv 文档里面记录了某地区新冠病人的住院信息。

1. 把 covid19.csv 读入到数据框 pat1 中，输出其列名和前三行的内容；

```
setwd("D:/homework_R/HW2")
pat1 <- read.csv("covid19.csv", header = TRUE)
colnames(pat1)
```

```
## [1] "序号"     "分型"     "性别"     "年龄"     "入院时间" "出院时间" "疗程"
```

```
head(pat1,n=3)
```

```
##   序号 分型 性别 年龄 入院时间 出院时间 疗程
## 1    1 重型   男   49 20200121 20200211   22
## 2    2 重型       47 20200121 20200214   25
## 3    3 重型   男   51 20200124 20200207   15
```

2. 找出性别中有缺失值的行并将其去掉，去掉后的数据框仍然记为 pat1；

```
# 找到性别中有缺失值的行
pat1 <- read.csv("covid19.csv", header = TRUE, na.strings = "")
pat1[is.na(pat1$性别),]
```

```
##   序号 分型 性别 年龄 入院时间 出院时间 疗程
## 2    2 重型 <NA>   47 20200121 20200214   25
```

```
# 去掉该行并将其记为 pat1
pat1<-pat1[which(is.na(pat1$性别)==FALSE),]
pat1
```

```
##    序号   分型 性别 年龄 入院时间 出院时间 疗程
## 1     1   重型   男   49 20200121 20200211   22
## 3     3   重型   男   51 20200124 20200207   15
## 4     4 普通型   男   24 20200122 20200203   13
## 5     5 普通型   男   47 20200122 20200208   18
## 6     6 普通型   女   40 20200125 20200208   15
## 7     7 普通型   女   50 20200125 20200203   10
## 8     8 普通型   男   55 20200126 20200210   16
## 9     9 普通型   女   24 20200123 20200210   19
## 10   10 普通型   女   50 20200128 20200210   14
## 11   11 普通型   男   51 20200128 20200217   21
## 12   12   重型   男   29 20200129 20200211   14
## 13   13 普通型   男   49 20200129 20200226   29
## 14   14 普通型   男   53 20200129 20200213   16
## 15   15 普通型   男   22 20200129 20200213   16
## 16   16 普通型   男   22 20200129 20200225   28
## 17   17 普通型   男   21 20200129 20200224   27
## 18   18 普通型   男   39 20200129 20200215   18
## 19   19 普通型   男   22 20200129 20200303   NA
## 20   20 普通型   女   22 20200130 20200225   27
## 21   21 普通型   女   16 20200130 20200212   14
## 22   22 普通型   女   37 20200130 20200214   16
## 23   23 普通型   男   37 20200130 20200215   17
## 24   24 普通型   女   47 20200131 20200220   21
## 25   25 普通型   男   34 20200131 20200213   14
## 26   26 普通型   女   32 20200131 20200220   21
## 27   27 普通型   女   27 20200201 20200219   19
```

```
## 28   28 普通型   男   28 20200131 20200225   26
## 29   29 普通型   女   27 20200131 20200305   NA
## 30   30   重型   男   68 20200201 20200223   24
## 31   31 普通型   女   66 20200201 20200220   20
## 32   32 普通型   男   67 20200201 20200223   23
## 33   33 普通型   男   35 20200201 20200220   20
## 34   34 普通型   女   43 20200202 20200222   21
## 35   35 普通型   女   56 20200203 20200225   23
## 36   36 普通型   男   21 20200203 20200218   16
## 37   37 普通型   女   53 20200203 20200305   NA
## 38   38 普通型   女   24 20200204 20200216   13
## 39   39 普通型   男   43 20200204 20200219   16
## 40   40 普通型   女   51 20200206 20200213    8
## 41   41 普通型   女   49 20200206 20200216   11
## 42   42 普通型   男   35 20200128 20200301   NA
## 43   43 普通型   男   23 20200128 20200216   19
## 44   44 普通型   女   47 20200128 20200210   14
## 45   45 普通型   女   45 20200207 20200301   NA
## 46   46 普通型   女   42 20200207 20200302   NA
## 47   47 普通型   女   69 20200207 20200301   NA
## 48   48 普通型   女   48 20200207 20200220   14
## 49   49 普通型   女   50 20200205 20200226   22
## 50   50 普通型   女   35 20200205 20200227   NA
## 51   51 普通型   女   55 20220205 20200222   18
## 52   54   重型   男   88 20200210 20200224   15
## 53   55   重型   男   63 20200211 20200307   NA
## 54   56   重型   男   36 20200207 20200218   12
## 55   57   重型   男   57 20200201 20200229   NA
## 56   58   重型   男   62 20200206 20200223   18
## 57   59   重型   男   53 20200209 20200218   11
## 58   60   重型   男   42 20200209 20200223   11
## 59   61 普通型   男   52 20200205 20200217   13
## 60   63   重型   女   56 20200209 20200226   18
```

```
## 61   65   重型   男   61 20200208 20200217   11
## 62   68 普通型   男   75 20200131 20200211   13
## 63   71   重型   男   91 20200206 20200224   19
## 64   73 普通型   男   44 20200208 20200224   17
## 65   74   重型   女   44 20200205 20200222   18
## 66   75 普通型   男   30 20200122 20200129    8
## 67   76 普通型   男   40 20200128 20200206   10
## 68   77 普通型   女   52 20200129 20200220   23
## 69   78 普通型   女   29 20200129 20200206    9
## 70   79   重型   男   69 20200206 20200220   15
## 71   80   重型   女   56 20200131 20200223   24
## 72   81   重型   男   68 20200213 20200224   12
## 73   82 普通型   男    5 20200131 20200213   NA
## 74   83   重型   男   72 20200221 20200225    5
## 75   84   重型   男   29 20200208 20200220   13
```

3. 分别按性别、入院日期汇总数据并输出不同性别的入院人数、不同入院日期的入院人数；

```
# 分开统计性别和入院日期
pat1[pat1$性别==" 男",];pat1[pat1$性别==" 女",]
```

```
##    序号   分型 性别 年龄 入院时间 出院时间 疗程
## 1     1   重型   男   49 20200121 20200211   22
## 3     3   重型   男   51 20200124 20200207   15
## 4     4 普通型   男   24 20200122 20200203   13
## 5     5 普通型   男   47 20200122 20200208   18
## 8     8 普通型   男   55 20200126 20200210   16
## 11   11 普通型   男   51 20200128 20200217   21
## 12   12   重型   男   29 20200129 20200211   14
## 13   13 普通型   男   49 20200129 20200226   29
## 14   14 普通型   男   53 20200129 20200213   16
## 15   15 普通型   男   22 20200129 20200213   16
## 16   16 普通型   男   22 20200129 20200225   28
```

```
## 17   17 普通型   男   21 20200129 20200224   27
## 18   18 普通型   男   39 20200129 20200215   18
## 19   19 普通型   男   22 20200129 20200303   NA
## 23   23 普通型   男   37 20200130 20200215   17
## 25   25 普通型   男   34 20200131 20200213   14
## 28   28 普通型   男   28 20200131 20200225   26
## 30   30   重型   男   68 20200201 20200223   24
## 32   32 普通型   男   67 20200201 20200223   23
## 33   33 普通型   男   35 20200201 20200220   20
## 36   36 普通型   男   21 20200203 20200218   16
## 39   39 普通型   男   43 20200204 20200219   16
## 42   42 普通型   男   35 20200128 20200301   NA
## 43   43 普通型   男   23 20200128 20200216   19
## 52   54   重型   男   88 20200210 20200224   15
## 53   55   重型   男   63 20200211 20200307   NA
## 54   56   重型   男   36 20200207 20200218   12
## 55   57   重型   男   57 20200201 20200229   NA
## 56   58   重型   男   62 20200206 20200223   18
## 57   59   重型   男   53 20200209 20200218   11
## 58   60   重型   男   42 20200209 20200223   11
## 59   61 普通型   男   52 20200205 20200217   13
## 61   65   重型   男   61 20200208 20200217   11
## 62   68 普通型   男   75 20200131 20200211   13
## 63   71   重型   男   91 20200206 20200224   19
## 64   73 普通型   男   44 20200208 20200224   17
## 66   75 普通型   男   30 20200122 20200129    8
## 67   76 普通型   男   40 20200128 20200206   10
## 70   79   重型   男   69 20200206 20200220   15
## 72   81   重型   男   68 20200213 20200224   12
## 73   82 普通型   男    5 20200131 20200213   NA
## 74   83   重型   男   72 20200221 20200225    5
## 75   84   重型   男   29 20200208 20200220   13

##     序号   分型 性别 年龄 入院时间 出院时间 疗程
```

```
## 6     6 普通型   女   40 20200125 20200208   15
## 7     7 普通型   女   50 20200125 20200203   10
## 9     9 普通型   女   24 20200123 20200210   19
## 10   10 普通型   女   50 20200128 20200210   14
## 20   20 普通型   女   22 20200130 20200225   27
## 21   21 普通型   女   16 20200130 20200212   14
## 22   22 普通型   女   37 20200130 20200214   16
## 24   24 普通型   女   47 20200131 20200220   21
## 26   26 普通型   女   32 20200131 20200220   21
## 27   27 普通型   女   27 20200201 20200219   19
## 29   29 普通型   女   27 20200131 20200305   NA
## 31   31 普通型   女   66 20200201 20200220   20
## 34   34 普通型   女   43 20200202 20200222   21
## 35   35 普通型   女   56 20200203 20200225   23
## 37   37 普通型   女   53 20200203 20200305   NA
## 38   38 普通型   女   24 20200204 20200216   13
## 40   40 普通型   女   51 20200206 20200213    8
## 41   41 普通型   女   49 20200206 20200216   11
## 44   44 普通型   女   47 20200128 20200210   14
## 45   45 普通型   女   45 20200207 20200301   NA
## 46   46 普通型   女   42 20200207 20200302   NA
## 47   47 普通型   女   69 20200207 20200301   NA
## 48   48 普通型   女   48 20200207 20200220   14
## 49   49 普通型   女   50 20200205 20200226   22
## 50   50 普通型   女   35 20200205 20200227   NA
## 51   51 普通型   女   55 20220205 20200222   18
## 60   63   重型   女   56 20200209 20200226   18
## 65   74   重型   女   44 20200205 20200222   18
## 68   77 普通型   女   52 20200129 20200220   23
## 69   78 普通型   女   29 20200129 20200206    9
## 71   80   重型   女   56 20200131 20200223   24
```

```
pat1[order(pat1[," 入院时间"]),]
```

```
##     序号   分型 性别 年龄 入院时间 出院时间 疗程
## 1      1   重型   男   49 20200121 20200211   22
## 4      4 普通型   男   24 20200122 20200203   13
## 5      5 普通型   男   47 20200122 20200208   18
## 66    75 普通型   男   30 20200122 20200129    8
## 9      9 普通型   女   24 20200123 20200210   19
## 3      3   重型   男   51 20200124 20200207   15
## 6      6 普通型   女   40 20200125 20200208   15
## 7      7 普通型   女   50 20200125 20200203   10
## 8      8 普通型   男   55 20200126 20200210   16
## 10    10 普通型   女   50 20200128 20200210   14
## 11    11 普通型   男   51 20200128 20200217   21
## 42    42 普通型   男   35 20200128 20200301   NA
## 43    43 普通型   男   23 20200128 20200216   19
## 44    44 普通型   女   47 20200128 20200210   14
## 67    76 普通型   男   40 20200128 20200206   10
## 12    12   重型   男   29 20200129 20200211   14
## 13    13 普通型   男   49 20200129 20200226   29
## 14    14 普通型   男   53 20200129 20200213   16
## 15    15 普通型   男   22 20200129 20200213   16
## 16    16 普通型   男   22 20200129 20200225   28
## 17    17 普通型   男   21 20200129 20200224   27
## 18    18 普通型   男   39 20200129 20200215   18
## 19    19 普通型   男   22 20200129 20200303   NA
## 68    77 普通型   女   52 20200129 20200220   23
## 69    78 普通型   女   29 20200129 20200206    9
## 20    20 普通型   女   22 20200130 20200225   27
## 21    21 普通型   女   16 20200130 20200212   14
## 22    22 普通型   女   37 20200130 20200214   16
## 23    23 普通型   男   37 20200130 20200215   17
## 24    24 普通型   女   47 20200131 20200220   21
```

```
## 25   25 普通型   男   34 20200131 20200213   14
## 26   26 普通型   女   32 20200131 20200220   21
## 28   28 普通型   男   28 20200131 20200225   26
## 29   29 普通型   女   27 20200131 20200305   NA
## 62   68 普通型   男   75 20200131 20200211   13
## 71   80   重型   女   56 20200131 20200223   24
## 73   82 普通型   男    5 20200131 20200213   NA
## 27   27 普通型   女   27 20200201 20200219   19
## 30   30   重型   男   68 20200201 20200223   24
## 31   31 普通型   女   66 20200201 20200220   20
## 32   32 普通型   男   67 20200201 20200223   23
## 33   33 普通型   男   35 20200201 20200220   20
## 55   57   重型   男   57 20200201 20200229   NA
## 34   34 普通型   女   43 20200202 20200222   21
## 35   35 普通型   女   56 20200203 20200225   23
## 36   36 普通型   男   21 20200203 20200218   16
## 37   37 普通型   女   53 20200203 20200305   NA
## 38   38 普通型   女   24 20200204 20200216   13
## 39   39 普通型   男   43 20200204 20200219   16
## 49   49 普通型   女   50 20200205 20200226   22
## 50   50 普通型   女   35 20200205 20200227   NA
## 59   61 普通型   男   52 20200205 20200217   13
## 65   74   重型   女   44 20200205 20200222   18
## 40   40 普通型   女   51 20200206 20200213    8
## 41   41 普通型   女   49 20200206 20200216   11
## 56   58   重型   男   62 20200206 20200223   18
## 63   71   重型   男   91 20200206 20200224   19
## 70   79   重型   男   69 20200206 20200220   15
## 45   45 普通型   女   45 20200207 20200301   NA
## 46   46 普通型   女   42 20200207 20200302   NA
## 47   47 普通型   女   69 20200207 20200301   NA
## 48   48 普通型   女   48 20200207 20200220   14
## 54   56   重型   男   36 20200207 20200218   12
```

```
## 61   65   重型   男   61 20200208 20200217   11
## 64   73 普通型   男   44 20200208 20200224   17
## 75   84   重型   男   29 20200208 20200220   13
## 57   59   重型   男   53 20200209 20200218   11
## 58   60   重型   男   42 20200209 20200223   11
## 60   63   重型   女   56 20200209 20200226   18
## 52   54   重型   男   88 20200210 20200224   15
## 53   55   重型   男   63 20200211 20200307   NA
## 72   81   重型   男   68 20200213 20200224   12
## 74   83   重型   男   72 20200221 20200225    5
## 51   51 普通型   女   55 20220205 20200222   18
```

```
# 对性别和入院日期交叉统计
table(pat1[," 性别"],pat1[," 入院时间"],useNA="always")
```

```
##
##        20200121 20200122 20200123 20200124 20200125 20200126 20200128 20200129
##   男          1        3        0        1        0        1        4        8
##   女          0        0        1        0        2        0        2        2
##   <NA>        0        0        0        0        0        0        0        0
##
##        20200130 20200131 20200201 20200202 20200203 20200204 20200205 20200206
##   男          1        4        4        0        1        1        1        3
##   女          3        4        2        1        2        1        3        2
##   <NA>        0        0        0        0        0        0        0        0
##
##        20200207 20200208 20200209 20200210 20200211 20200213 20200221 20220205
##   男          1        3        2        1        1        1        1        0
##   女          4        0        1        0        0        0        0        1
##   <NA>        0        0        0        0        0        0        0        0
##
##        <NA>
##   男      0
##   女      0
```

```
##   <NA>    0
```

```
# 输出不同性别入院人数
table(pat1$性别)
```

```
##
## 男 女
## 43 31
```

```
# 不同入院日期的入院人数
table(pat1$入院时间)
```

```
##
## 20200121 20200122 20200123 20200124 20200125 20200126 20200128 20200129
##        1        3        1        1        2        1        6       10
## 20200130 20200131 20200201 20200202 20200203 20200204 20200205 20200206
##        4        8        6        1        3        2        4        5
## 20200207 20200208 20200209 20200210 20200211 20200213 20200221 20220205
##        5        3        3        1        1        1        1        1
```

4. 分性别计算平均年龄、最大年龄和最小年龄；

```
aggregate(pat1[," 年龄"],by=pat1[" 性别"],mean)
```

```
##   性别        x
## 1   男 45.62791
## 2   女 43.29032
```

```
aggregate(pat1[," 年龄"],by=pat1[" 性别"],min)
```

```
##   性别  x
## 1   男  5
## 2   女 16
```

```
aggregate(pat1[," 年龄"],by=pat1[" 性别"],max)
```

```
##   性别  x
## 1   男 91
## 2   女 69
```

5. 把年龄分成 0—10, 11—20, ……, 70 以上各段，保存为“年龄段”变量，并将其加入到数据框 d 中；

```
for(i in 1:nrow(pat1)){
  if(pat1[i," 年龄"]>=0&pat1[i," 年龄"]<=10){
    pat1[i," 年龄段"]<- "0-10"
  }
  if(pat1[i," 年龄"]>=11&pat1[i," 年龄"]<=20){
    pat1[i," 年龄段"]<- "11-20"
  }
  if(pat1[i," 年龄"]>=21&pat1[i," 年龄"]<=30){
    pat1[i," 年龄段"]<- "21-30"
  }
  if(pat1[i," 年龄"]>=31&pat1[i," 年龄"]<=40){
    pat1[i," 年龄段"]<- "31-40"
  }
  if(pat1[i," 年龄"]>=41&pat1[i," 年龄"]<=50){
    pat1[i," 年龄段"]<- "41-50"
  }
  if(pat1[i," 年龄"]>=51&pat1[i," 年龄"]<=60){
    pat1[i," 年龄段"]<- "51-60"
  }
  if(pat1[i," 年龄"]>=61&pat1[i," 年龄"]<=70){
    pat1[i," 年龄段"]<- "61-70"
  }
  if(pat1[i," 年龄"]>=71){
    pat1[i," 年龄段"]<- "70 以上"
```

```
  }
}
head(pat1,n=10)    # 查看前十行确定写的没问题
```

```
##    序号   分型 性别 年龄 入院时间 出院时间 疗程 年龄段
## 1     1   重型   男   49 20200121 20200211   22 41-50
## 3     3   重型   男   51 20200124 20200207   15 51-60
## 4     4 普通型   男   24 20200122 20200203   13 21-30
## 5     5 普通型   男   47 20200122 20200208   18 41-50
## 6     6 普通型   女   40 20200125 20200208   15 31-40
## 7     7 普通型   女   50 20200125 20200203   10 41-50
## 8     8 普通型   男   55 20200126 20200210   16 51-60
## 9     9 普通型   女   24 20200123 20200210   19 21-30
## 10   10 普通型   女   50 20200128 20200210   14 41-50
## 11   11 普通型   男   51 20200128 20200217   21 51-60
```

6. 用年龄段和性别交叉汇总发病人数和百分比 (结果乘以 100 并保留一位小数)，保存到 “年龄性别分布.csv” 中要求将每个年龄段的男性病人数、女性发病人数存为一行。

```
t<-table(pat1[," 年龄段"],pat1[," 性别"])
t
```

```
##
##          男 女
##   0-10    1  0
##   11-20   0  1
##   21-30  11  6
##   31-40   7  4
##   41-50   6 11
##   51-60   7  7
##   61-70   7  2
##   70以上  4  0
```

```
t<-cbind(t,c(1:8),c(1:8))
# 男性发病人数占该年龄段人数百分比 简称 男性占比，女性发病人数占该年龄段人数百分比 简称 女性占
colnames(t)<-c(" 男"," 女"," 男性占比"," 女性占比")
for(i in 1:8){
  sum=t[i,1]+t[i,2]
  t[i,3]<- round(t[i,1]/sum*100,1)
  t[i,4]<- round(t[i,2]/sum*100,1)
}
t
```

```
##         男 女 男性占比 女性占比
## 0-10     1  0    100.0      0.0
## 11-20    0  1      0.0    100.0
## 21-30   11  6     64.7     35.3
## 31-40    7  4     63.6     36.4
## 41-50    6 11     35.3     64.7
## 51-60    7  7     50.0     50.0
## 61-70    7  2     77.8     22.2
## 70以上   4  0    100.0      0.0
```

```
write.csv(t,file=" 年龄性别分布.csv")
```

covid19-3.csv 文档记录病人入院前的基本信息。7. 把 covid19-3.csv 读入到数据框 pat2 中，输出其列名和前三行的内容；

```
pat2<- read.csv("covid19-3.csv",header=TRUE)
colnames(pat2)
```

```
## [1] "序号" "职业" "病史" "吸烟"
```

```
head(pat2,n=3)
```

```
##   序号 职业 病史 吸烟
```

```
## 1    1 职员 首诊   无
## 2    2 无业 首诊   无
## 3    3 职员 首诊   有
```

8. 合并 pat1 和 pat2，并将合并后的数据框定义为 pat，要求合并后保留 pat1 所有的行;

```
pat <- merge(pat1,pat2,by.x=" 序号",by.y=" 序号",all.x=TRUE)
head(pat)
```

```
##   序号   分型 性别 年龄 入院时间 出院时间 疗程 年龄段     职业 病史 吸烟
## 1    1   重型   男   49 20200121 20200211   22 41-50     职员 首诊   无
## 2    3   重型   男   51 20200124 20200207   15 51-60     职员 首诊   有
## 3    4 普通型   男   24 20200122 20200203   13 21-30 地铁工作 首诊   无
## 4    5 普通型   男   47 20200122 20200208   18 41-50     销售 首诊   有
## 5    6 普通型   女   40 20200125 20200208   15 31-40     个体 首诊   无
## 6    7 普通型   女   50 20200125 20200203   10 41-50     <NA> <NA> <NA>
```

```
dim(pat1);dim(pat2);dim(pat)  # 验证合并满足要求
```

```
## [1] 74  8
```

```
## [1] 69  4
```

```
## [1] 74 11
```

9. 显示抽烟、年龄超过 40 且住院时间超过 10 天的病人分型和年龄。

```
# 对日期数据进行类型修改并添加住院时间这一列
pat[," 入院时间"] <- as.Date(as.character(pat[," 入院时间"]), format = "%Y%m%d")
pat[," 出院时间"] <- as.Date(as.character(pat[," 出院时间"]), format = "%Y%m%d")
pat[," 住院时间"] <- as.numeric(pat[," 出院时间"] - pat[," 入院时间"]) + 1
pat
```

	序号	分型	性别	年龄	入院时间	出院时间	疗程	年龄段	职业	病史
1	1	重型	男	49	2020-01-21	2020-02-11	22	41-50	职员	首诊
2	3	重型	男	51	2020-01-24	2020-02-07	15	51-60	职员	首诊
3	4	普通型	男	24	2020-01-22	2020-02-03	13	21-30	地铁工作	首诊
4	5	普通型	男	47	2020-01-22	2020-02-08	18	41-50	销售	首诊
5	6	普通型	女	40	2020-01-25	2020-02-08	15	31-40	个体	首诊
6	7	普通型	女	50	2020-01-25	2020-02-03	10	41-50	<NA>	<NA>
7	8	普通型	男	55	2020-01-26	2020-02-10	16	51-60	无业	首诊
8	9	普通型	女	24	2020-01-23	2020-02-10	19	21-30	职员	首诊
9	10	普通型	女	50	2020-01-28	2020-02-10	14	41-50	个体	首诊
10	11	普通型	男	51	2020-01-28	2020-02-17	21	51-60	干部	首诊
11	12	重型	男	29	2020-01-29	2020-02-11	14	21-30	职员	首诊
12	13	普通型	男	49	2020-01-29	2020-02-26	29	41-50	职员	首诊
13	14	普通型	男	53	2020-01-29	2020-02-13	16	51-60	职员	首诊
14	15	普通型	男	22	2020-01-29	2020-02-13	16	21-30	学生	首诊
15	16	普通型	男	22	2020-01-29	2020-02-25	28	21-30	<NA>	<NA>
16	17	普通型	男	21	2020-01-29	2020-02-24	27	21-30	学生	首诊
17	18	普通型	男	39	2020-01-29	2020-02-15	18	31-40	无业	首诊
18	19	普通型	男	22	2020-01-29	2020-03-03	NA	21-30	学生	首诊
19	20	普通型	女	22	2020-01-30	2020-02-25	27	21-30	职员	首诊
20	21	普通型	女	16	2020-01-30	2020-02-12	14	11-20	学生	首诊
21	22	普通型	女	37	2020-01-30	2020-02-14	16	31-40	个体经营	首诊
22	23	普通型	男	37	2020-01-30	2020-02-15	17	31-40	个体经营	首诊
23	24	普通型	女	47	2020-01-31	2020-02-20	21	41-50	个体经营	首诊
24	25	普通型	男	34	2020-01-31	2020-02-13	14	31-40	个体	首诊
25	26	普通型	女	32	2020-01-31	2020-02-20	21	31-40	无业	首诊
26	27	普通型	女	27	2020-02-01	2020-02-19	19	21-30	教师	首诊
27	28	普通型	男	28	2020-01-31	2020-02-25	26	21-30	采购	首诊
28	29	普通型	女	27	2020-01-31	2020-03-05	NA	21-30	<NA>	<NA>
29	30	重型	男	68	2020-02-01	2020-02-23	24	61-70	农民	首诊
30	31	普通型	女	66	2020-02-01	2020-02-20	20	61-70	无业	首诊
31	32	普通型	男	67	2020-02-01	2020-02-23	23	61-70	退休	首诊
32	33	普通型	男	35	2020-02-01	2020-02-20	20	31-40	不详	首诊

```
## 33   34 普通型   女  43 2020-02-02 2020-02-22  21  41-50      职员      首诊
## 34   35 普通型   女  56 2020-02-03 2020-02-25  23  51-60      工人      首诊
## 35   36 普通型   男  21 2020-02-03 2020-02-18  16  21-30      学生      首诊
## 36   37 普通型   女  53 2020-02-03 2020-03-05  NA  51-60      无业      首诊
## 37   38 普通型   女  24 2020-02-04 2020-02-16  13  21-30      无业      首诊
## 38   39 普通型   男  43 2020-02-04 2020-02-19  16  41-50      工人      首诊
## 39   40 普通型   女  51 2020-02-06 2020-02-13   8  51-60    公务员      首诊
## 40   41 普通型   女  49 2020-02-06 2020-02-16  11  41-50      农民      首诊
## 41   42 普通型   男  35 2020-01-28 2020-03-01  NA  31-40      <NA>      <NA>
## 42   43 普通型   男  23 2020-01-28 2020-02-16  19  21-30      工人      首诊
## 43   44 普通型   女  47 2020-01-28 2020-02-10  14  41-50      工人      首诊
## 44   45 普通型   女  45 2020-02-07 2020-03-01  NA  41-50      无业      首诊
## 45   46 普通型   女  42 2020-02-07 2020-03-02  NA  41-50      个体      首诊
## 46   47 普通型   女  69 2020-02-07 2020-03-01  NA  61-70      无业      首诊
## 47   48 普通型   女  48 2020-02-07 2020-02-20  14  41-50      无业      首诊
## 48   49 普通型   女  50 2020-02-05 2020-02-26  22  41-50      医生      首诊
## 49   50 普通型   女  35 2020-02-05 2020-02-27  NA  31-40      职员      首诊
## 50   51 普通型   女  55 2022-02-05 2020-02-22  18  51-60  自由职业      首诊
## 51   54   重型   男  88 2020-02-10 2020-02-24  15 70以上      退休  外院转入
## 52   55   重型   男  63 2020-02-11 2020-03-07  NA  61-70      退休  外院转入
## 53   56   重型   男  36 2020-02-07 2020-02-18  12  31-40      职员  外院转入
## 54   57   重型   男  57 2020-02-01 2020-02-29  NA  51-60      农民      首诊
## 55   58   重型   男  62 2020-02-06 2020-02-23  18  61-70  个体经营  外院转入
## 56   59   重型   男  53 2020-02-09 2020-02-18  11  51-60      农民  外院转入
## 57   60   重型   男  42 2020-02-09 2020-02-23  11  41-50      农民  外院转入
## 58   61 普通型   男  52 2020-02-05 2020-02-17  13  51-60      农民  外院转入
## 59   63   重型   女  56 2020-02-09 2020-02-26  18  51-60      农民  外院转入
## 60   65   重型   男  61 2020-02-08 2020-02-17  11  61-70      <NA>      <NA>
## 61   68 普通型   男  75 2020-01-31 2020-02-11  13 70以上      退休      首诊
## 62   71   重型   男  91 2020-02-06 2020-02-24  19 70以上        无      首诊
## 63   73 普通型   男  44 2020-02-08 2020-02-24  17  41-50      无业      首诊
## 64   74   重型   女  44 2020-02-05 2020-02-22  18  41-50      个体  外院转入
## 65   75 普通型   男  30 2020-01-22 2020-01-29   8  21-30      金融      首诊
```

```
## 66   76 普通型   男   40 2020-01-28 2020-02-06   10  31-40   汽修工     首诊
## 67   77 普通型   女   52 2020-01-29 2020-02-20   23  51-60 服务行业     首诊
## 68   78 普通型   女   29 2020-01-29 2020-02-06    9  21-30   公务员     首诊
## 69   79   重型   男   69 2020-02-06 2020-02-20   15  61-70     退休     首诊
## 70   80   重型   女   56 2020-01-31 2020-02-23   24  51-60     农民     首诊
## 71   81   重型   男   68 2020-02-13 2020-02-24   12  61-70     医生 外院转入
## 72   82 普通型   男    5 2020-01-31 2020-02-13   NA   0-10       无     首诊
## 73   83   重型   男   72 2020-02-21 2020-02-25    5 70以上     <NA>     <NA>
## 74   84   重型   男   29 2020-02-08 2020-02-20   13  21-30     无业     首诊
##    吸烟 住院时间
## 1    无       22
## 2    有       15
## 3    无       13
## 4    有       18
## 5    无       15
## 6  <NA>       10
## 7    无       16
## 8    无       19
## 9    无       14
## 10   无       21
## 11   无       14
## 12   无       29
## 13   无       16
## 14   无       16
## 15 <NA>       28
## 16   无       27
## 17   无       18
## 18   无       35
## 19   无       27
## 20   无       14
## 21   无       16
## 22   无       17
## 23   无       21
```

```
## 24   有       14
## 25   无       21
## 26   无       19
## 27   无       26
## 28 <NA>       35
## 29   无       23
## 30   无       20
## 31   无       23
## 32   无       20
## 33   无       21
## 34   无       23
## 35   无       16
## 36   无       32
## 37   无       13
## 38   无       16
## 39   无        8
## 40   无       11
## 41 <NA>       34
## 42   无       20
## 43   无       14
## 44   无       24
## 45   无       25
## 46   无       24
## 47   无       14
## 48   无       22
## 49   无       23
## 50   无     -713
## 51   无       15
## 52   无       26
## 53   无       12
## 54   无       29
## 55   无       18
## 56   无       10
```

```
## 57   无       15
## 58   无       13
## 59   无       18
## 60 <NA>       10
## 61   无       12
## 62   无       19
## 63   无       17
## 64   无       18
## 65   无        8
## 66   无       10
## 67   无       23
## 68   无        9
## 69   无       15
## 70   无       24
## 71   无       12
## 72   无       14
## 73 <NA>        5
## 74   无       13
```

```
pat[(pat$吸烟==" 有") & (pat$年龄>40) & (pat$住院时间>10),c(" 分型"," 年龄")]
```

```
##     分型 年龄
## 2   重型   51
## 4 普通型   47
```