Description for category variables and plot

Yiqun Wu

2022/9/15

1- Set working directory

```
setwd("C:/Users/wyq/Desktop/practice") # 设置目标工作文件夹
#setwd("C:\\Users\\wyq\\Desktop\\practice") # 设置目标工作文件夹
getwd() # 查看现在的工作文件夹
rm(list=ls()) # Cleaning the working environment清空
```

2- Data importing & Saving

```
rawdata <- read.csv("./模拟数据/rawdata.csv",header=TRUE,stringsAsFactors=F,na.strings=c("","NA"))
# 导入CSV数据，字符变量不变，空白数据设置为缺失NA
cleandata <- read.csv("./模拟数据/cleandata.csv",header=TRUE,stringsAsFactors=F,na.strings=c("","NA")) # 导入CSV数据，字符变量不变，空白数据设置为缺失NA
ls() # list objects in the working environment
```

```
## [1] "cleandata" "rawdata"
```

```
save(rawdata,file="./模拟数据/practice.Rdata") # 保存mydata R格式数据库，可同时保存多个对象
save(rawdata,cleandata,file="./模拟数据/practice.Rdata") # 保存mydata R格式数据库，可同时保存多个对象

rm(list=ls()) #清空现有内存
load("./模拟数据/practice.Rdata") # reload saved R dataset导入原来保存的R数据库
ls() # list objects in the working environment
```

```
## [1] "cleandata" "rawdata"
```

3- Dataset description

3.1- Check the first or last lines of a dataset

```
head(rawdata, n=10) # 显示数据集前十行
head(rawdata, n= -10) # 除外最后十行
tail(rawdata) # 最后六行
tail(rawdata, n=10) # 最后十行
tail(rawdata, n= -10) # 除外最前十行
rawdata[1:10, ] # 数据集中1:10行
```

3.2- show the dimensions of a dataset

```
dim(rawdata) # 展示数据库维度
```

```
## [1] 3184   10
```

3.3- show the variables names of a dataset

```
names(rawdata) # 显示数据库中所有变量名
```

```
##  [1] "id"     "sex"    "age"    "edu"    "smk"    "dnk"    "height" "weight"
##  [9] "sbp"    "dbp"
```

3.4- show the summary information for each variable in the dataset

```
summary(rawdata) # 展示数据库基本情况
```

```
##       id                 sex                 age            edu
##  Length:3184        Length:3184        Min.   :15.00   Length:3184
##  Class :character   Class :character   1st Qu.:32.00   Class :character
##  Mode  :character   Mode  :character   Median :48.00   Mode  :character
##                                        Mean   :48.04
##                                        3rd Qu.:65.00
##                                        Max.   :81.00
##
##      smk                dnk                height          weight
##  Length:3184        Length:3184        Min.   :155.0   Min.   : 43.00
##  Class :character   Class :character   1st Qu.:162.0   1st Qu.: 56.00
##  Mode  :character   Mode  :character   Median :167.0   Median : 66.00
##                                        Mean   :167.6   Mean   : 68.44
##                                        3rd Qu.:172.0   3rd Qu.: 77.00
##                                        Max.   :281.0   Max.   :391.00
##                                        NA's   :6       NA's   :41
##       sbp              dbp
##  Min.   : 73.0   Min.   : 50.00
##  1st Qu.: 97.0   1st Qu.: 63.00
##  Median :111.0   Median : 73.00
##  Mean   :114.9   Mean   : 75.06
##  3rd Qu.:132.0   3rd Qu.: 88.00
##  Max.   :335.0   Max.   :347.00
##  NA's   :12      NA's   :36
```

3.5- display the structure of a dataset

```
str(rawdata) # 展示数据库结构
```

```
## 'data.frame':    3184 obs. of  10 variables:
##  $ id    : chr  "f04049" "f11417" "f00193" "f03966" ...
##  $ sex   : chr  "男" "女" "女" "男" ...
##  $ age   : int  80 81 81 81 80 81 81 81 81 81 ...
##  $ edu   : chr  "未上学" "未上学" "中学" "大学及以上" ...
##  $ smk   : chr  NA NA "过去吸烟" NA ...
##  $ dnk   : chr  "现在饮酒" "现在饮酒" "从不饮酒" "从不饮酒" ...
##  $ height: int  171 160 155 178 165 181 160 162 168 170 ...
##  $ weight: int  64 68 62 65 74 67 68 43 96 65 ...
##  $ sbp   : int  145 150 132 146 135 133 160 88 120 140 ...
##  $ dbp   : int  89 90 80 102 90 65 95 89 77 95 ...
```

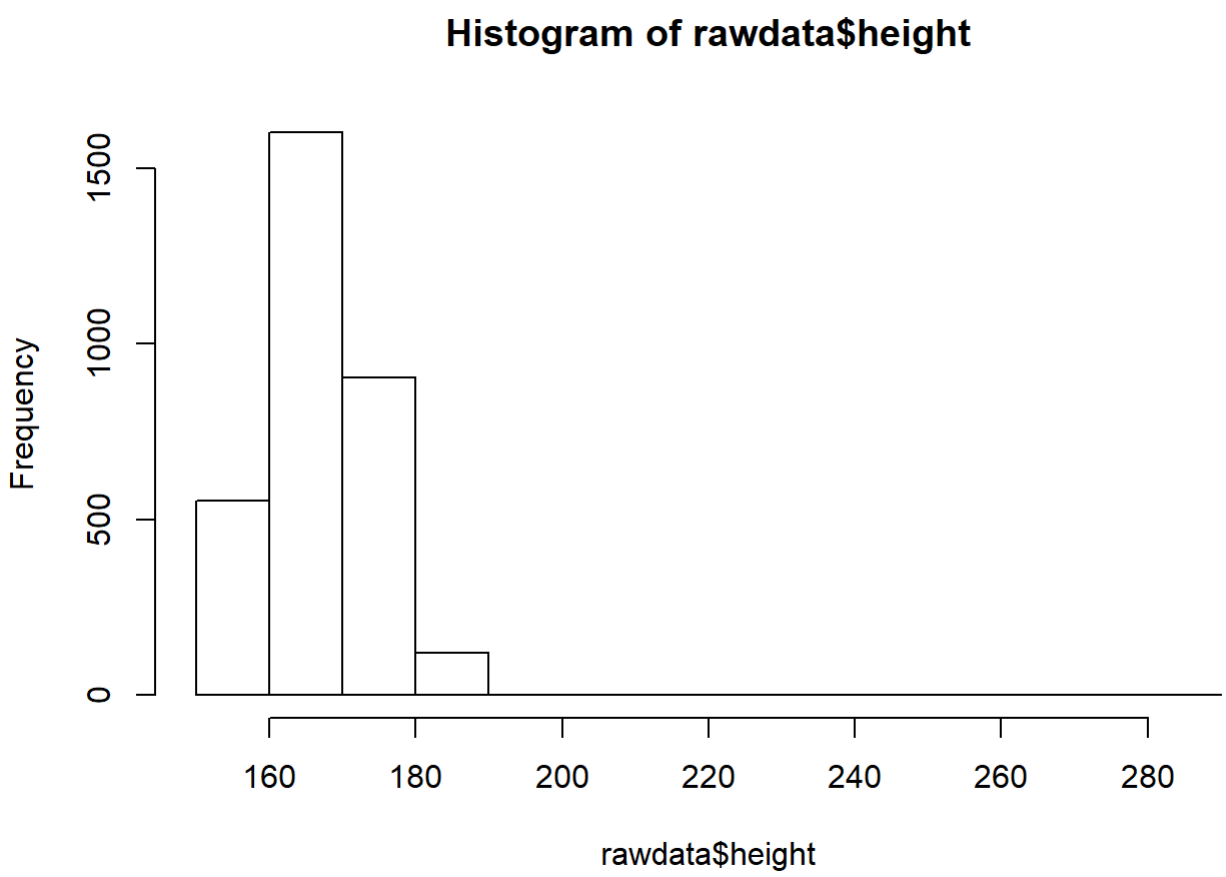
4- Data cleaning: identify outliers

4.1- 利用直方图查看分布，核查异常值

```
#查看身高的异常值
summary(rawdata$height)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   155.0   162.0   167.0   167.6   172.0   281.0       6
```

```
hist(rawdata$height)
```



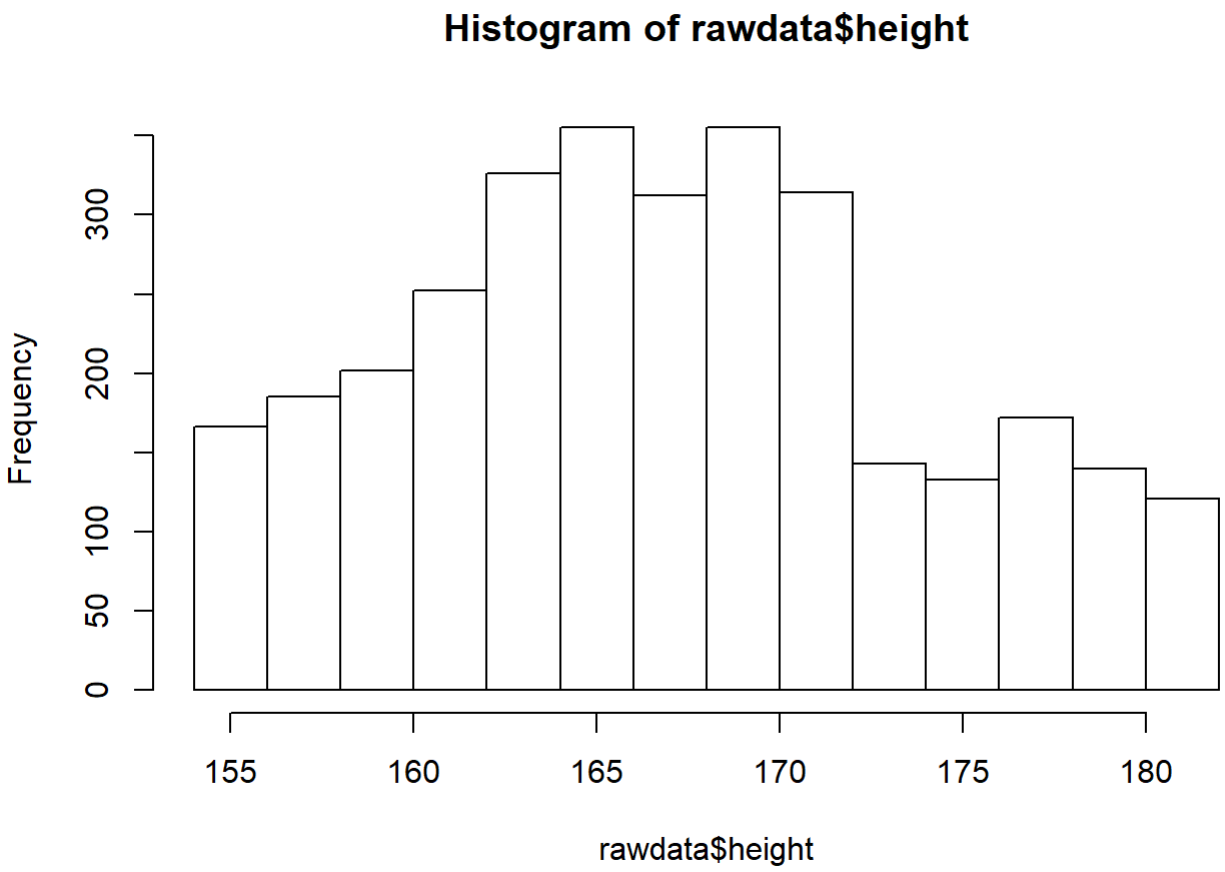

```
#去除身高超过220cm的异常值
table(rawdata$height>220)
```

```
##
## FALSE  TRUE
##  3176     2
```

```
rawdata$height[rawdata$height>220] <- NA
summary(rawdata$height)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   155.0   162.0   167.0   167.5   172.0   182.0       8
```

```
hist(rawdata$height)
```



4.2- 利用频数表查看分布，核查异常值

```
# 查看体重的异常值
summary(rawdata$weight)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   43.00   56.00   66.00   68.44   77.00  391.00      41
```

```
table(cut(rawdata$weight,seq(40,400,10),right=F))
```

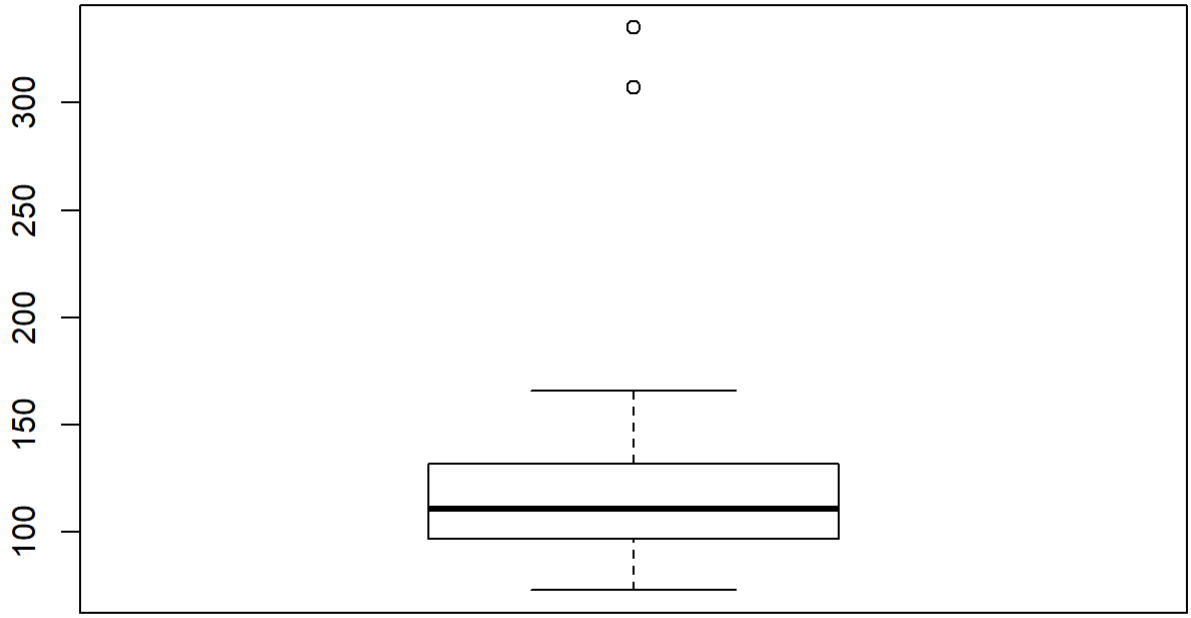
```
##
##   [40,50)   [50,60)   [60,70)   [70,80)   [80,90)  [90,100) [100,110) [110,120)
##       353       676       822       587       283       313       106         0
## [120,130) [130,140) [140,150) [150,160) [160,170) [170,180) [180,190) [190,200)
##         0         0         0         0         0         0         0         0
## [200,210) [210,220) [220,230) [230,240) [240,250) [250,260) [260,270) [270,280)
##         0         0         0         0         0         0         0         0
## [280,290) [290,300) [300,310) [310,320) [320,330) [330,340) [340,350) [350,360)
##         0         0         0         0         0         0         0         0
## [360,370) [370,380) [380,390) [390,400)
##         2         0         0         1
```

```
# 去除体重超过360kg的异常值
rawdata$weight[rawdata$weight>360] <- NA
table(cut(rawdata$weight,seq(40,400,10),right=F))
```

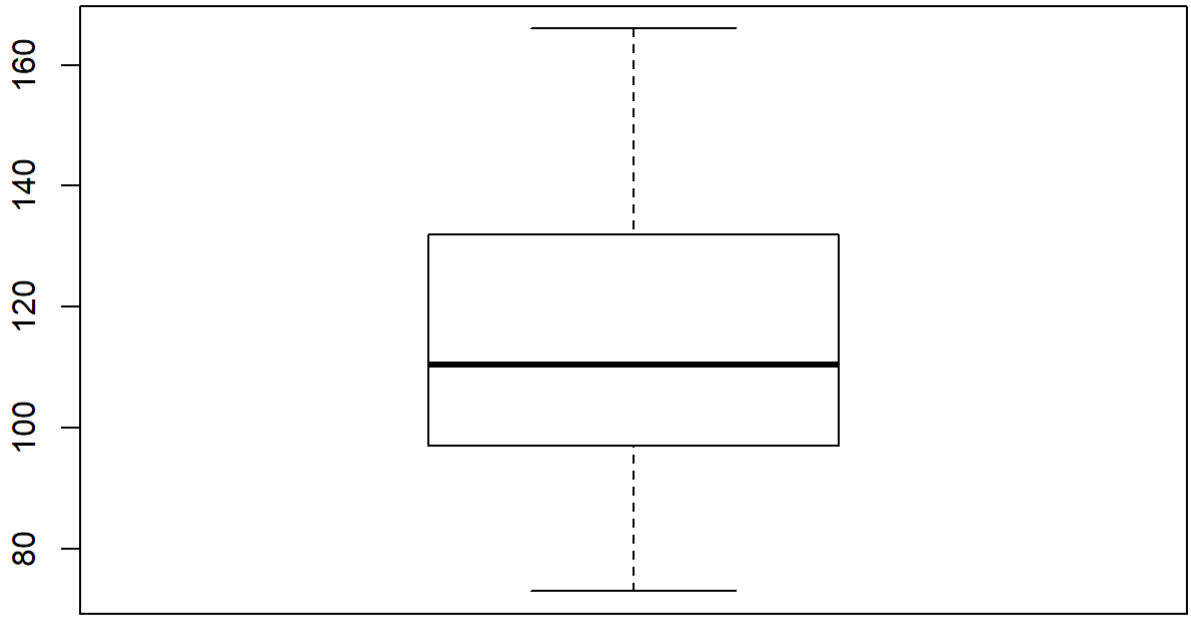
```
##
##   [40,50)   [50,60)   [60,70)   [70,80)   [80,90)  [90,100) [100,110) [110,120)
##       353       676       822       587       283       313       106         0
## [120,130) [130,140) [140,150) [150,160) [160,170) [170,180) [180,190) [190,200)
##         0         0         0         0         0         0         0         0
## [200,210) [210,220) [220,230) [230,240) [240,250) [250,260) [260,270) [270,280)
##         0         0         0         0         0         0         0         0
## [280,290) [290,300) [300,310) [310,320) [320,330) [330,340) [340,350) [350,360)
##         0         0         0         0         0         0         0         0
## [360,370) [370,380) [380,390) [390,400)
##         0         0         0         0
```

4.3- 利用箱式图查看分布，核查异常值

```
# 查看SBP的异常值
boxplot(rawdata$sbp)
```




```
# 去除SBP超过300mmHg的异常值
rawdata$sbp[rawdata$sbp>300] <- NA
boxplot(rawdata$sbp)
```



4.4- 利用茎叶图查看分布，核查异常值

```
# 查看DBP的异常值
stem(rawdata$dbp)
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##    4 | 00000000000000000000000000000000000000000000000011111111111111111+466
##    6 | 00000000000000000000000000000000000000000000000000000000000000000+1302
##    8 | 00000000000000000000000000000000000000000000000011111111111111111+940
##   10 | 00000000000000000000000000000000000000000000000011111111111111111+119
##   12 |
##   14 |
##   16 |
##   18 |
##   20 |
##   22 |
##   24 |
##   26 |
##   28 |
##   30 |
##   32 |
##   34 | 7
```

```
# 去除DBP超过300mmHg的异常值
rawdata$dbp[rawdata$dbp>300] <- NA
stem(rawdata$dbp)
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##    5 | 00000000000000000000000000000000000000000000000011111111111111111+156
##    5 | 55555555555555555555555555555555555555555555555555555566666666666+230
##    6 | 00000000000000000000000000000000000000000000000000000000000000000+337
##    6 | 55555555555555555555555555555555555555555555555555555555555555555+365
##    7 | 00000000000000000000000000000000000000000000000000000011111111+190
##    7 | 55555555555555555555555555555555555555555555566666666666666666666666+170
##    8 | 00000000000000000000000000000000000000000000011111111111111111111+181
##    8 | 55555555555555555555555555555555555555555555566666666666666666666666+173
##    9 | 00000000000000000000000000000000000000000000000000000111111111+173
##    9 | 55555555555555555555555555555555566666666666666666666666666666666+173
##   10 | 00000000000000000000000000000000000000000000000011111111111111111+92
##   10 | 5555555555555555555555555
```

```
# 随机选择50个个体的DBP值，绘制茎叶图
stem(sample(rawdata$dbp,50)) # 茎叶图更适合小样本
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##    5 | 45689
##    6 | 11113455777
##    7 | 1237
##    8 | 01233688999
##    9 | 000123344447899
##   10 | 334
```

5- Variable editing

R中一些常用的变量类型

类型	举例
character	“骨关节炎”,“骨关节病”,“指骨关节炎”
numeric	10,20,30,40
factor	男[1]，女[2]
Date	“2010-01-01”
logical	TRUE, FALSE

可以利用class()函数来查看变量类型

5.1- create new variables

```
# calculating BMI
library(dplyr) # 加载dplyr包
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
cleandata <- dplyr::mutate(cleandata,bmi=weight*10000/(height^2)) # 生成新变量BMI（注，生成新变量有多种方法）
cleandata[1:5,c("id","height","weight","bmi")] # 查看新生成的bmi
```

```
##       id height weight      bmi
## 1 f00193    155     62 25.80645
## 2 f26354    165     74 27.18090
## 3 f21524    181     67 20.45115
## 4 f10495    160     68 26.56250
## 5 f16368    168     96 34.01361
```

```
cleandata$bmi2 <- (cleandata$weight*10000)/(cleandata$height^2) # 直接利用基本函数生成bmi2
table(cleandata$bmi2==cleandata$bmi) # 比较bmi和bmi2两个变量是否相同
```

```
##
## TRUE
## 2406
```

5.2- change continuous variables to categorical variables

BMI to BMI groups

```
summary(cleandata$bmi) # 描述bmi
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.53   20.20   23.78   24.24   27.68   38.87
```

```
cleandata <- dplyr::mutate(cleandata,bmigrp=cut(bmi,c(14,18,24,28,40),right=F)) # 生成BMI分组变量bmigrp
cleandata[1:5,c("id","height","weight","bmi","bmigrp")] # 查看新生成的bmigrp分组变量
```

```
##       id height weight      bmi  bmigrp
## 1 f00193    155     62 25.80645 [24,28)
## 2 f26354    165     74 27.18090 [24,28)
## 3 f21524    181     67 20.45115 [18,24)
## 4 f10495    160     68 26.56250 [24,28)
## 5 f16368    168     96 34.01361 [28,40)
```

```
table(cleandata$bmigrp) # 生成bmi分组频数表
```

```
##
## [14,18) [18,24) [24,28) [28,40)
##     255     979     605     567
```

Age to age groups

```
summary(cleandata$age) # 描述age
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   15.00   31.00   48.00   48.15   65.00   81.00
```

```
cleandata <- dplyr::mutate(cleandata,
                           agegrp=cut(age,c(15,30,45,60,75,90),include.lowest = T,right=F)) # 生成年龄分组变量agegrp
summary(cleandata$agegrp) # 生成年龄分组频数表
```

```
## [15,30) [30,45) [45,60) [60,75) [75,90]
##     516     555     555     544     236
```

BP levels to hypertension groups

```
cleandata <- dplyr::mutate(cleandata,
                           hypt=ifelse(sbp>=140|dbp>=90,"yes","no")
                           )# 生成高血压分组变量hypt
table(cleandata$hypt) # 生成高血压分组频数表
```

```
##
##   no  yes
## 1626  780
```

5.2- categorical variables: factor levels

```
str(cleandata) # 查看数据结构，原来sex,edu,smk,dnk都是cha字符型变量
table(cleandata$edu) #频数表的结果按照类别字母排序
cleandata <- dplyr::mutate(cleandata,
                           sex=factor(sex,levels=c("女","男")), # 类别排序：女、男，女为参照
                           edu=factor(edu,levels=c("未上学","小学","中学","大学及以上")), # 类别排序："未上学","小学","中学","大学及以上", "未上学"为参照
                           smk=factor(smk,levels=c("从不吸烟","过去吸烟","现在吸烟")), # 类别排序："从不吸烟","过去吸烟","现在吸烟", "从不吸烟"为参照
                           dnk=factor(dnk,levels=c("从不饮酒","过去饮酒","现在饮酒")), # 类别排序："从不饮酒","过去饮酒","现在饮酒", "从不饮酒"为参照
                           hypt=factor(hypt,levels=c("no","yes")), # 类别排序："no","yes", "no"为参照
                          )
str(cleandata) # 查看数据结构，以上字符变量均变为factor
```

```
table(cleandata$edu) #频数表的结果按照定义类别排序展示
```

```
##
## 大学及以上      未上学        未知        小学        中学
##        760         482          25         353         786
```

6- Descriptions

常用的一些函数

函数	描述
range()	最小值，最大值
min(),max()	最小值，最大值
mean(),median()	中位数
sd()	标准差data5$start.tm[1:10]
table()	频数表
cor()	相关
summary()	数据基础统计，连续变量及分类变量
na.rm=TRUE or useNA = “ifany”	去掉缺失值 保留缺失值

6.1- tables for categorical variables*

```
summary(cleandata$hypt,na.rm=TRUE) # 频数分布
```

```
##    Length     Class      Mode
##      2406 character character
```

```
table(cleandata$hypt,useNA = "ifany") # 频数分布
```

```
##
##   no  yes
## 1626  780
```

```
prop.table(table(cleandata$hypt,useNA = "ifany")) # 频数比例
```

```
##
##        no       yes
## 0.6758105 0.3241895
```

```
table(cleandata$sex,cleandata$hypt,useNA="ifany") # 两个变量交叉表格
```

```
##
##       no yes
##   男 757 378
##   女 869 402
```

```
prop.table(table(cleandata$sex,cleandata$hypt,useNA="ifany"),1) # 两个变量，横向比例
```

```
##
##             no       yes
##   男 0.6669604 0.3330396
##   女 0.6837136 0.3162864
```

```
prop.table(table(cleandata$sex,cleandata$hypt,useNA="ifany"),2) # 两个变量，纵向比例
```

```
##
##             no       yes
##   男 0.4655597 0.4846154
##   女 0.5344403 0.5153846
```

```
table(cleandata$sex,cleandata$hypt,cleandata$agegrp) # 可以尝试更高维的交叉表格
```

```
## , ,  = [15,30)
##
##
##       no yes
##   男 169  47
##   女 234  66
##
## , ,  = [30,45)
##
##
##       no yes
##   男 181  83
##   女 207  84
##
## , ,  = [45,60)
##
##
##       no yes
##   男 175 101
##   女 177 102
##
## , ,  = [60,75)
##
##
##       no yes
##   男 164  95
##   女 179 106
##
## , ,  = [75,90]
##
##
##       no yes
##   男  68  52
##   女  72  44
```

6.2- descriptions for continous variables

```
mean(cleandata$bmi,na.rm=TRUE) # 均数
```

```
## [1] 24.24364
```

```
sd(cleandata$bmi,na.rm=TRUE) # 标准差
```

```
## [1] 5.122615
```

```
median(cleandata$bmi,na.rm=TRUE) # 中位数
```

```
## [1] 23.78121
```

```
range(cleandata$bmi,na.rm=TRUE) # 最小值，最大值
```

```
## [1] 14.53488 38.86603
```

```
min(cleandata$bmi,na.rm=TRUE) # 最小值
```

```
## [1] 14.53488
```

```
max(cleandata$bmi,na.rm=TRUE) # 最大值
```

```
## [1] 38.86603
```

```
summary(cleandata$bmi,na.rm=TRUE) # 最小值，最大值，上下四分位数，中位数，均数，缺失值
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.53   20.20   23.78   24.24   27.68   38.87
```

```
dplyr::summarize(group_by(cleandata,sex), # 按性别分类
                 N=n(),
                 na=sum(is.na(bmi)),
                 mean=mean(bmi,na.rm=TRUE),
                 sd=sd(bmi,na.rm=TRUE),
                 min=min(bmi,na.rm=TRUE),
                 max=max(bmi,na.rm=TRUE)
                 )
```

```
## # A tibble: 2 x 7
##   sex       N    na  mean    sd   min   max
##   <chr> <int> <int> <dbl> <dbl> <dbl> <dbl>
## 1 男     1135     0  26.6  5.34  16.0  38.9
## 2 女     1271     0  22.1  3.85  14.5  30.8
```

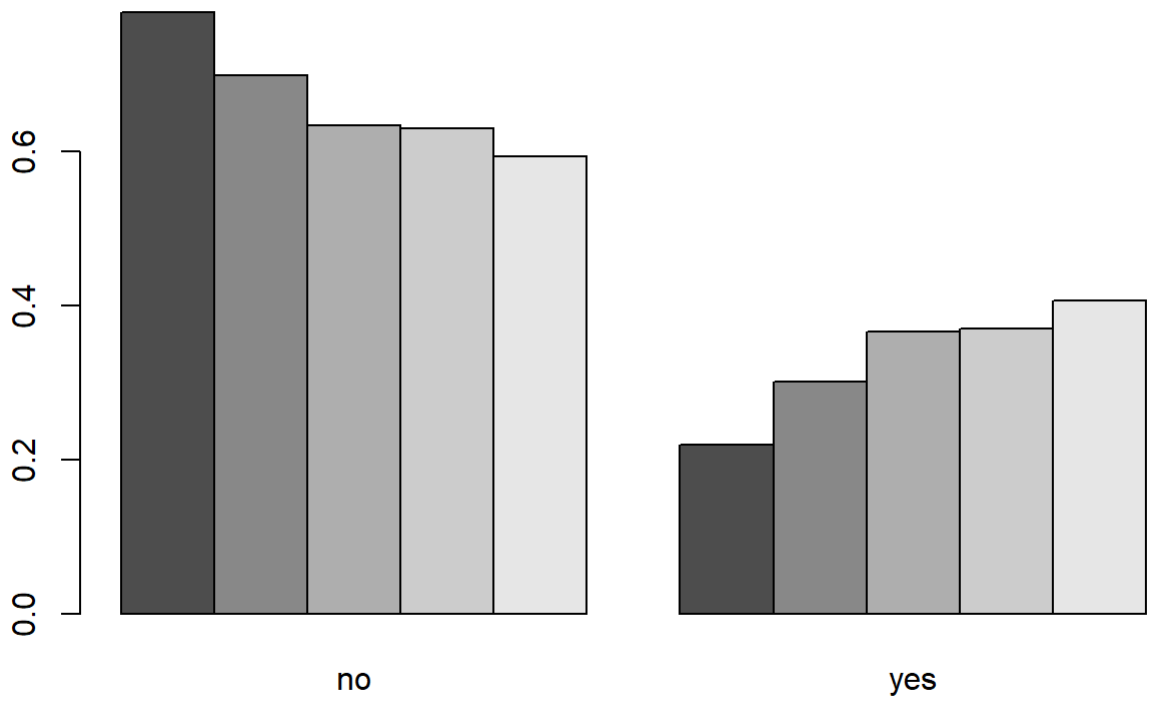
6.3- figures

6.3.1- barplot() 绘制各年龄组高血压患病率条图

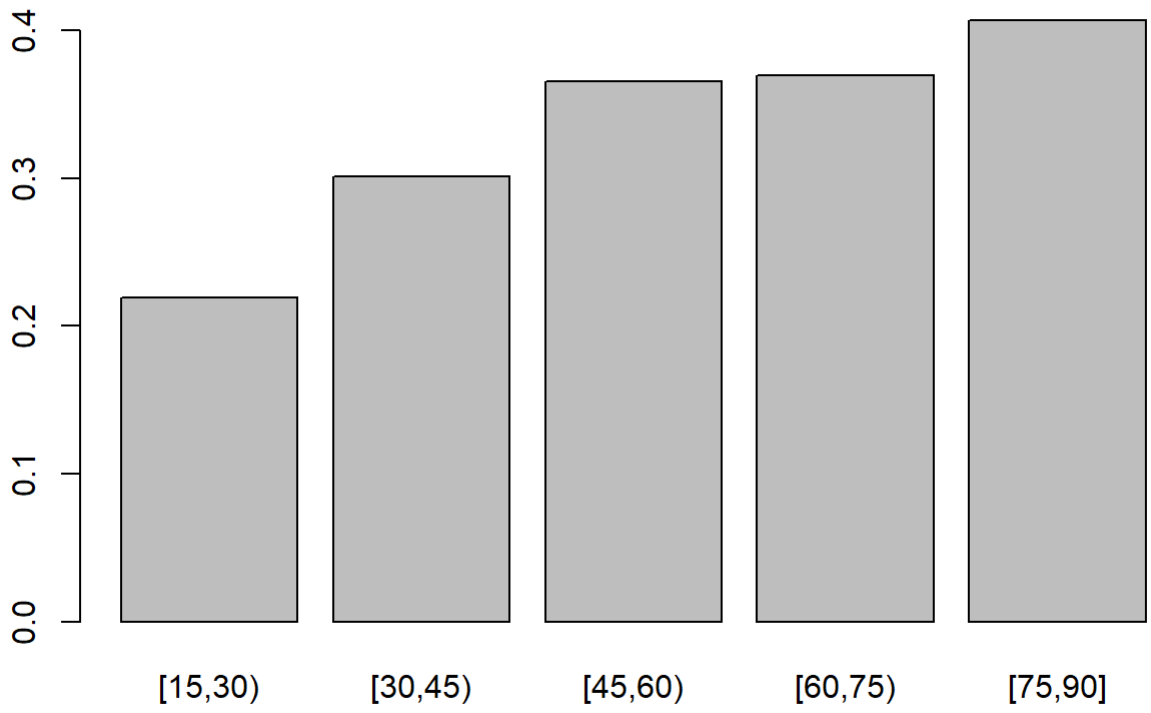
```
# 不同年龄组高血压患病率
prop.table(table(cleandata$agegrp,cleandata$hypt),1)
```

```
##
##                   no       yes
##   [15,30) 0.7810078 0.2189922
##   [30,45) 0.6990991 0.3009009
##   [45,60) 0.6342342 0.3657658
##   [60,75) 0.6305147 0.3694853
##   [75,90] 0.5932203 0.4067797
```

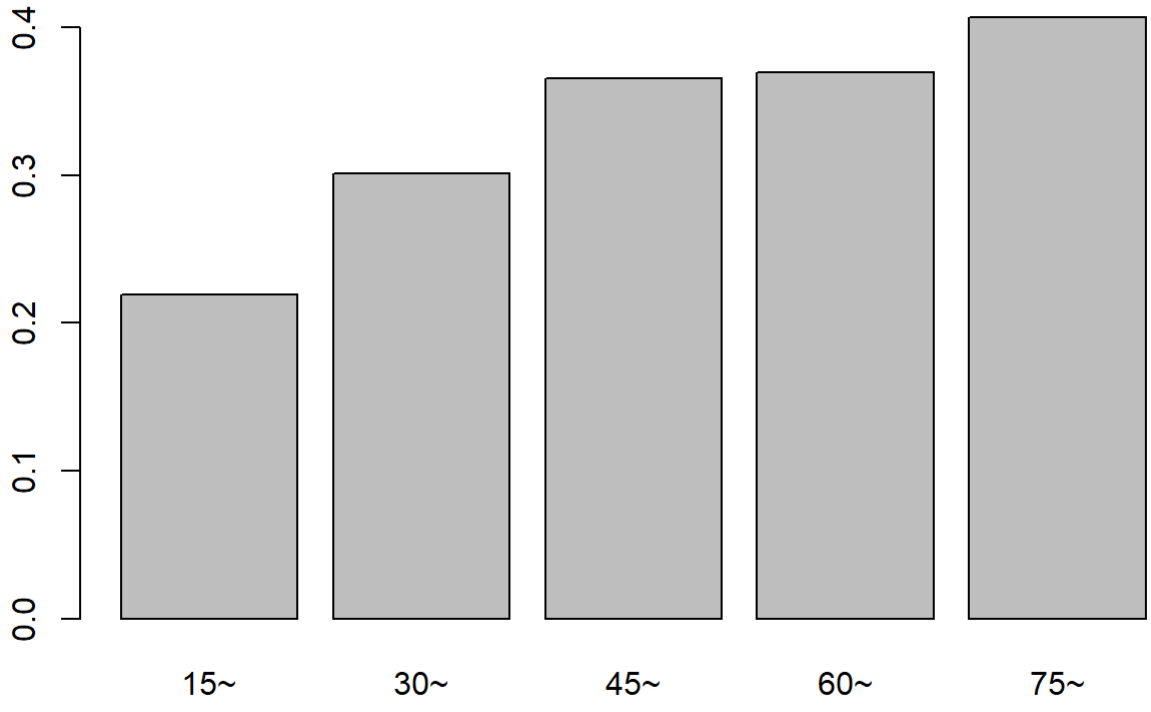
```
df <- prop.table(table(cleandata$agegrp,cleandata$hypt),1)
barplot(df,beside = TRUE)
```



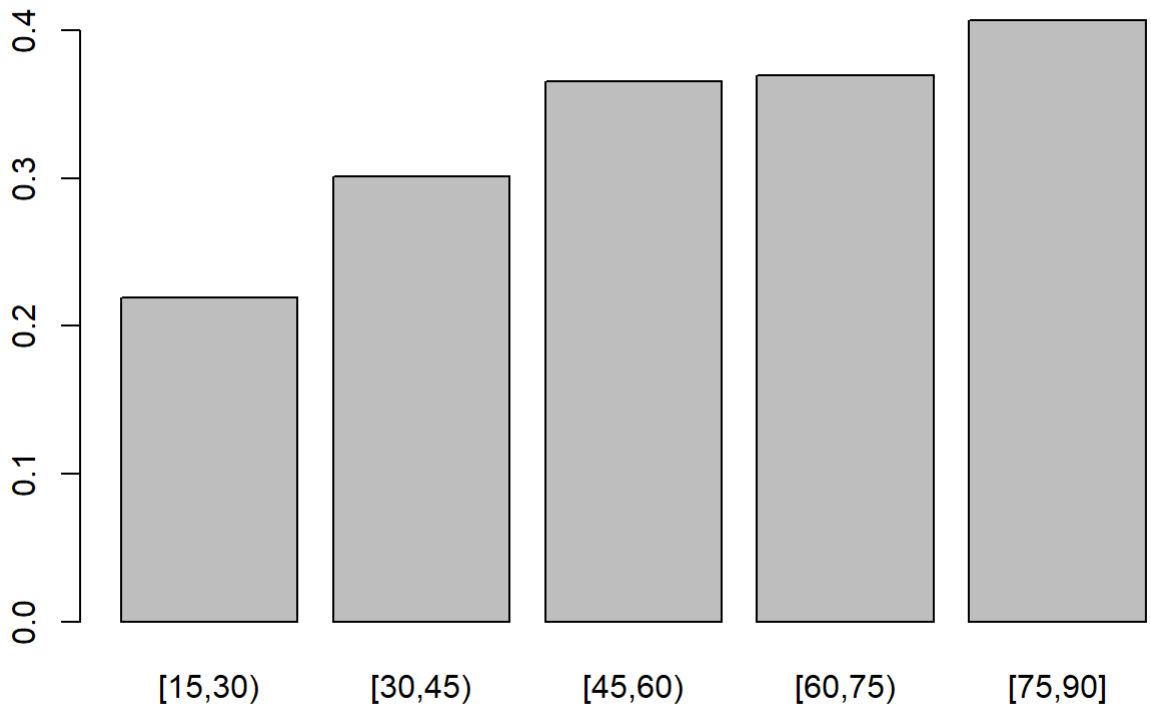

```
hypt <- df[,2]
barplot(hypt)
```



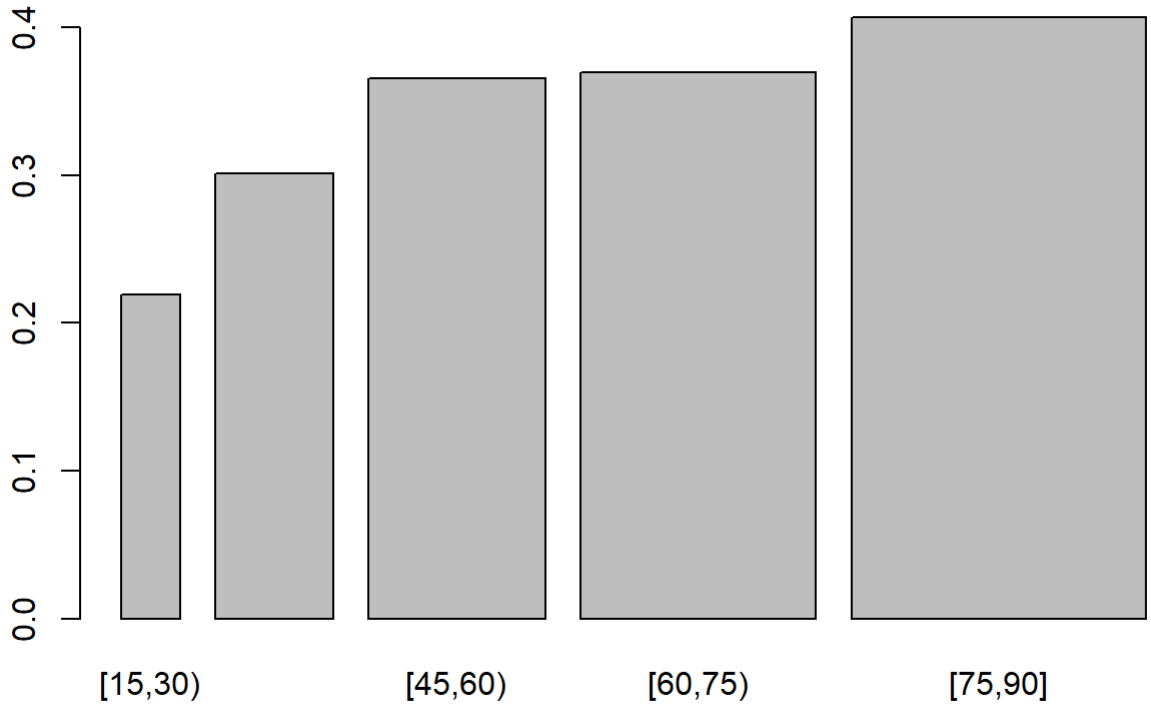

```
barplot(hypt,names.arg = c("15~","30~","45~","60~","75~"))
```



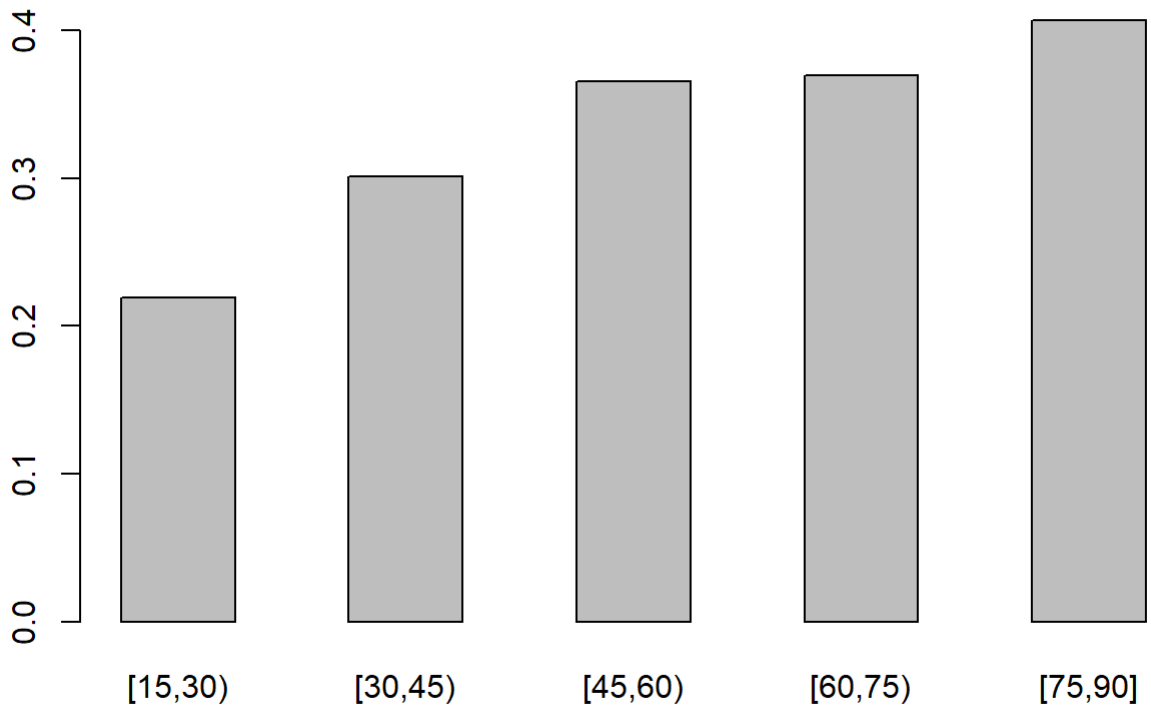

```
barplot(hypt,width=2)
```



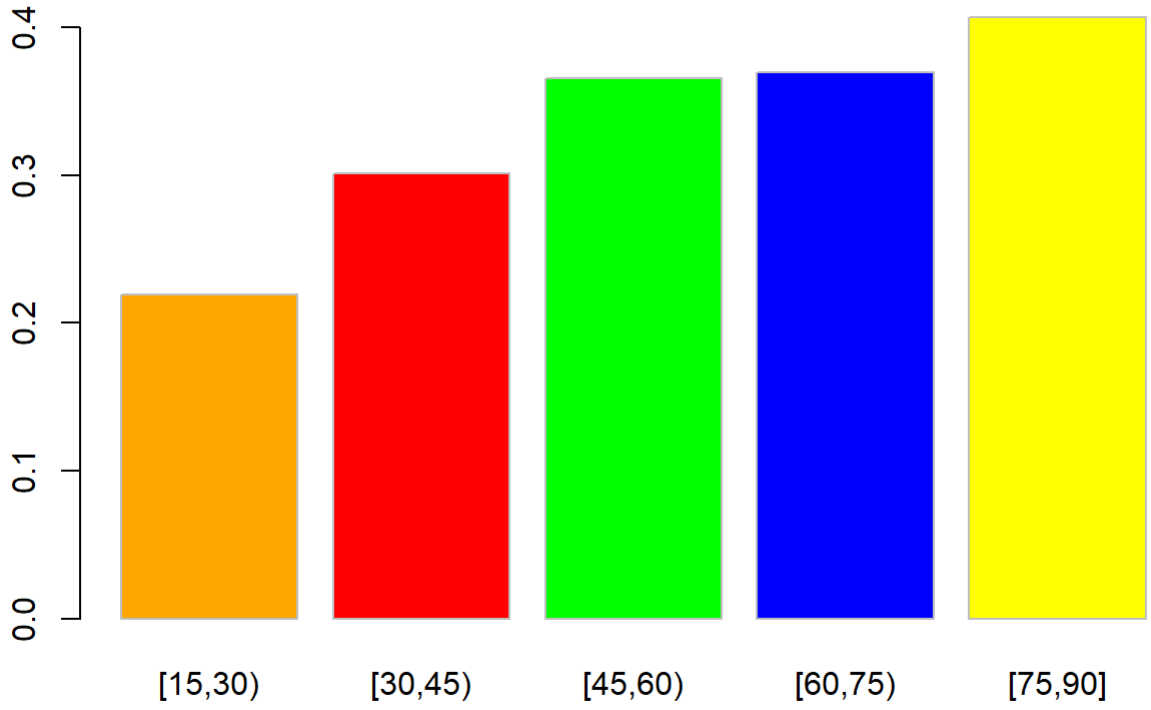

```
barplot(hypt,width=1:5)
```



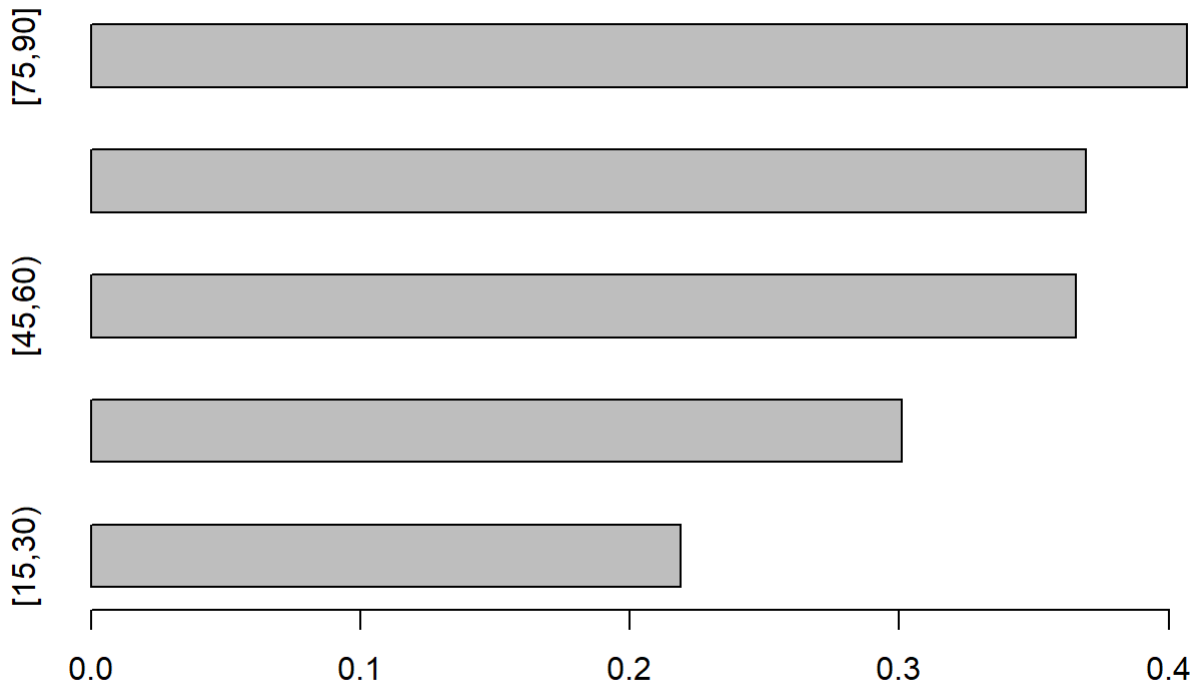

```
barplot(hypt,width=1,space=1)
```



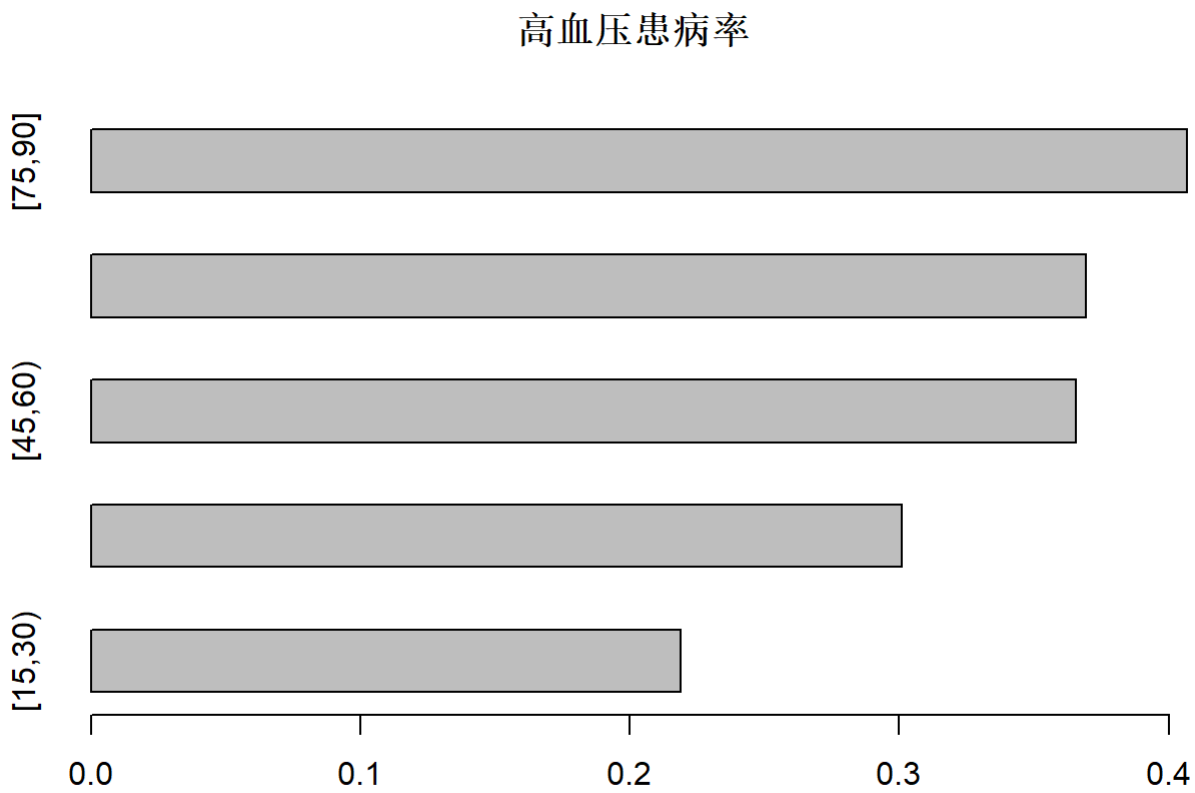

```
barplot(hypt,col = c("orange","red","green","blue","yellow"),border="grey")
```



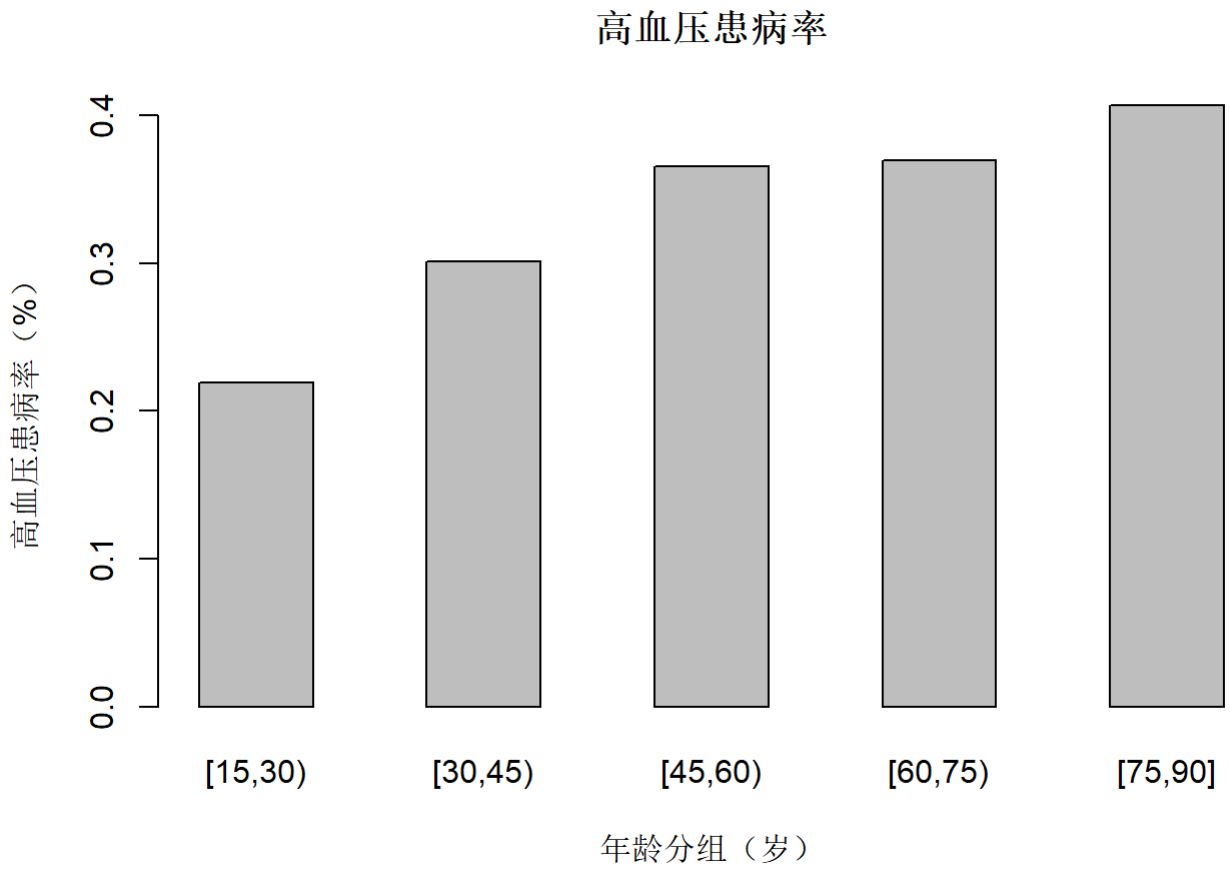

```
barplot(hypt,width=1,space=1,horiz=TRUE)
```



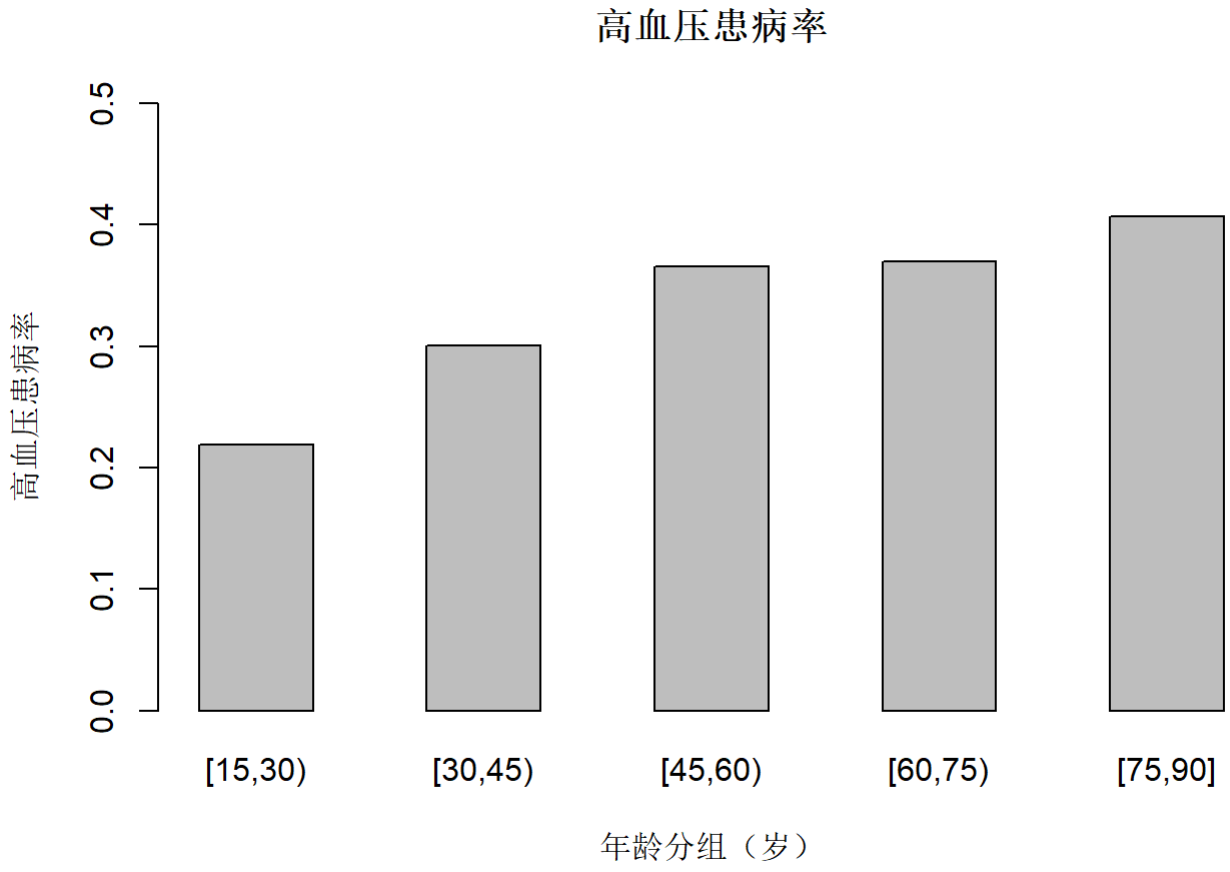

```
barplot(hypt,width=1,space=1,horiz=TRUE,main="高血压患病率")
```



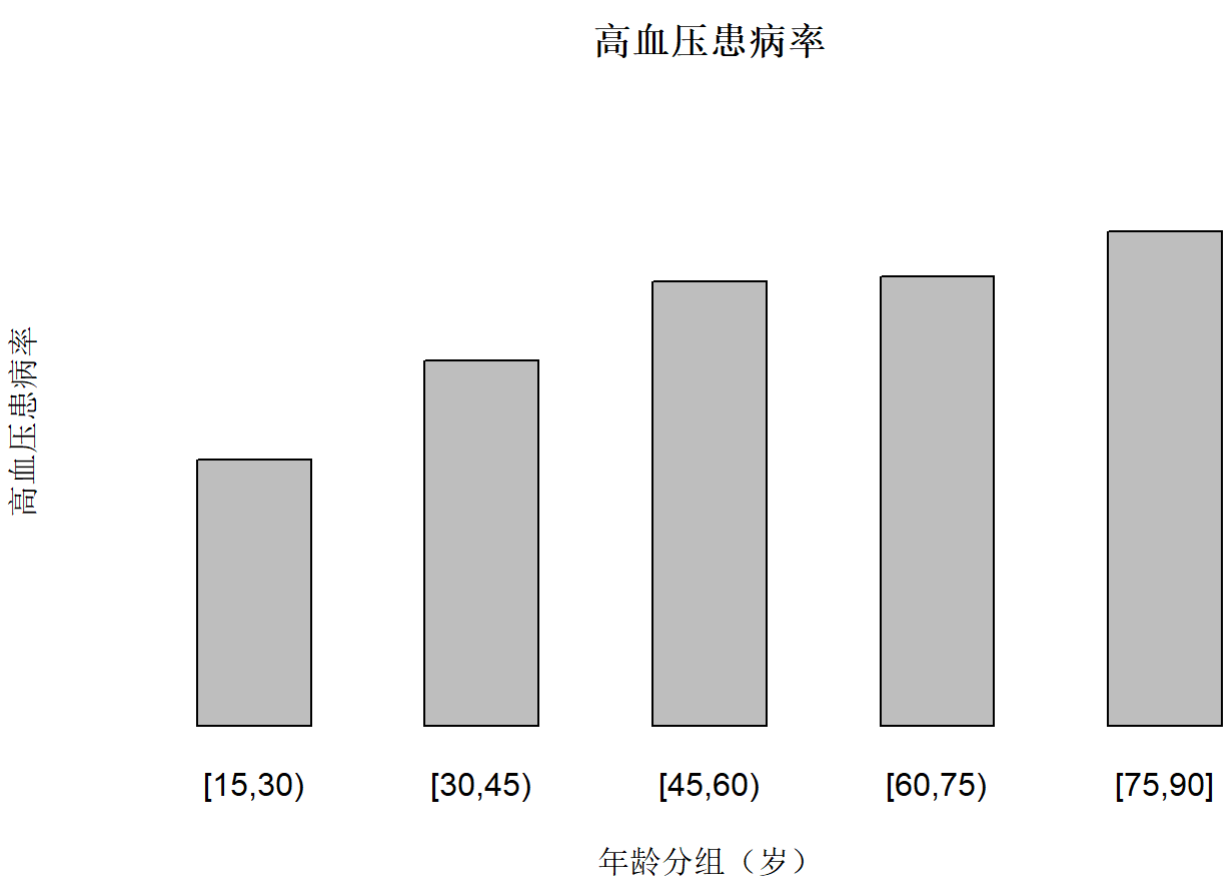

```
barplot(hypt,width=1,space=1,main="高血压患病率",xlab="年龄分组（岁）",ylab="高血压患病率（%）")
```



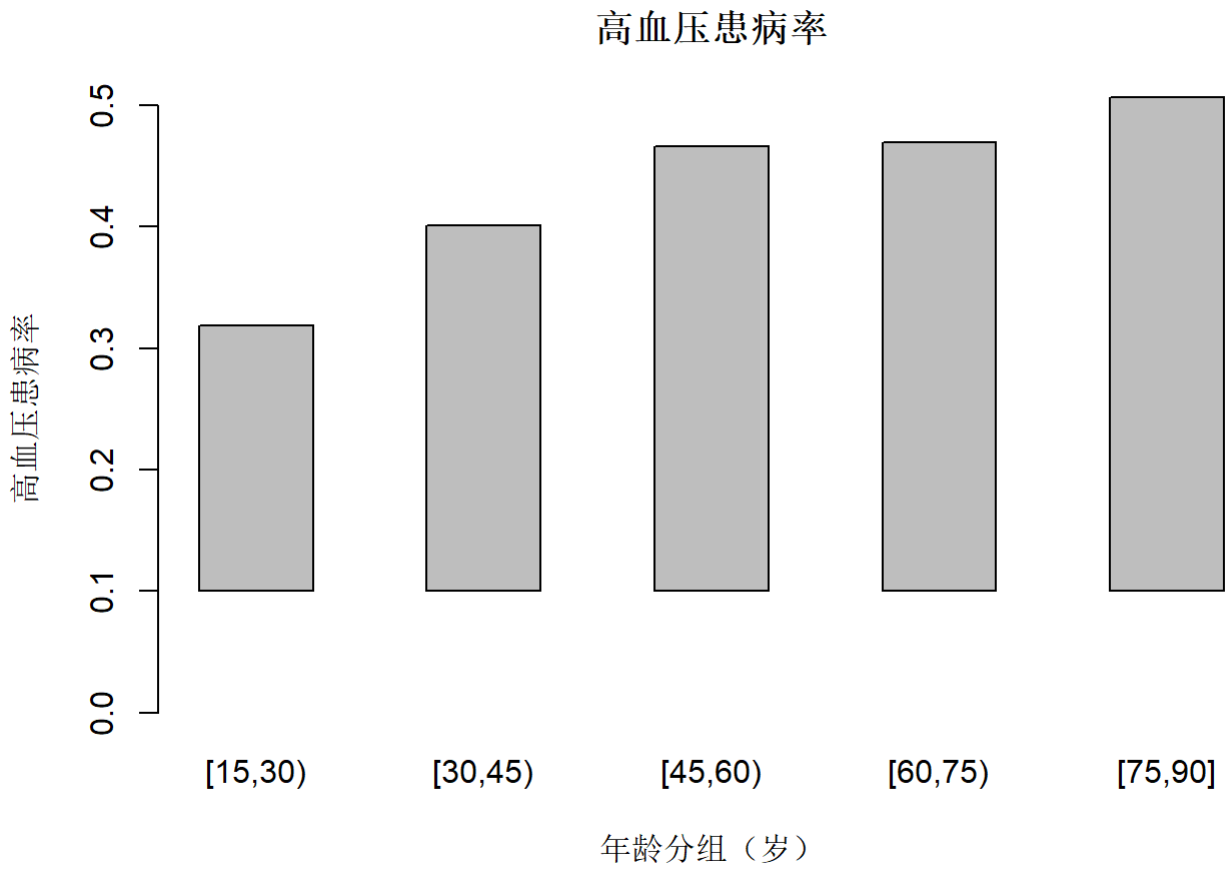

```
barplot(hypt,width=1,space=1,main="高血压患病率",xlab="年龄分组（岁）",ylab="高血压患病率",ylim=c(0,0.5))
```



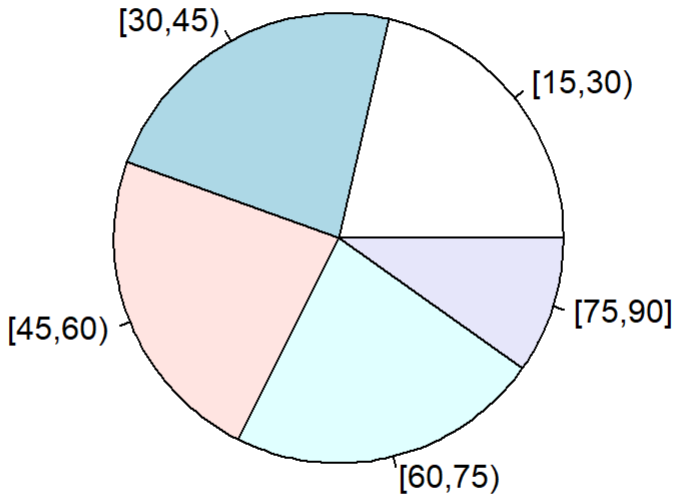
```
barplot(hypt,width=1,space=1,main="高血压患病率",xlab="年龄分组（岁）",ylab="高血压患病率",ylim=c(0,0.5),axes=F)
```



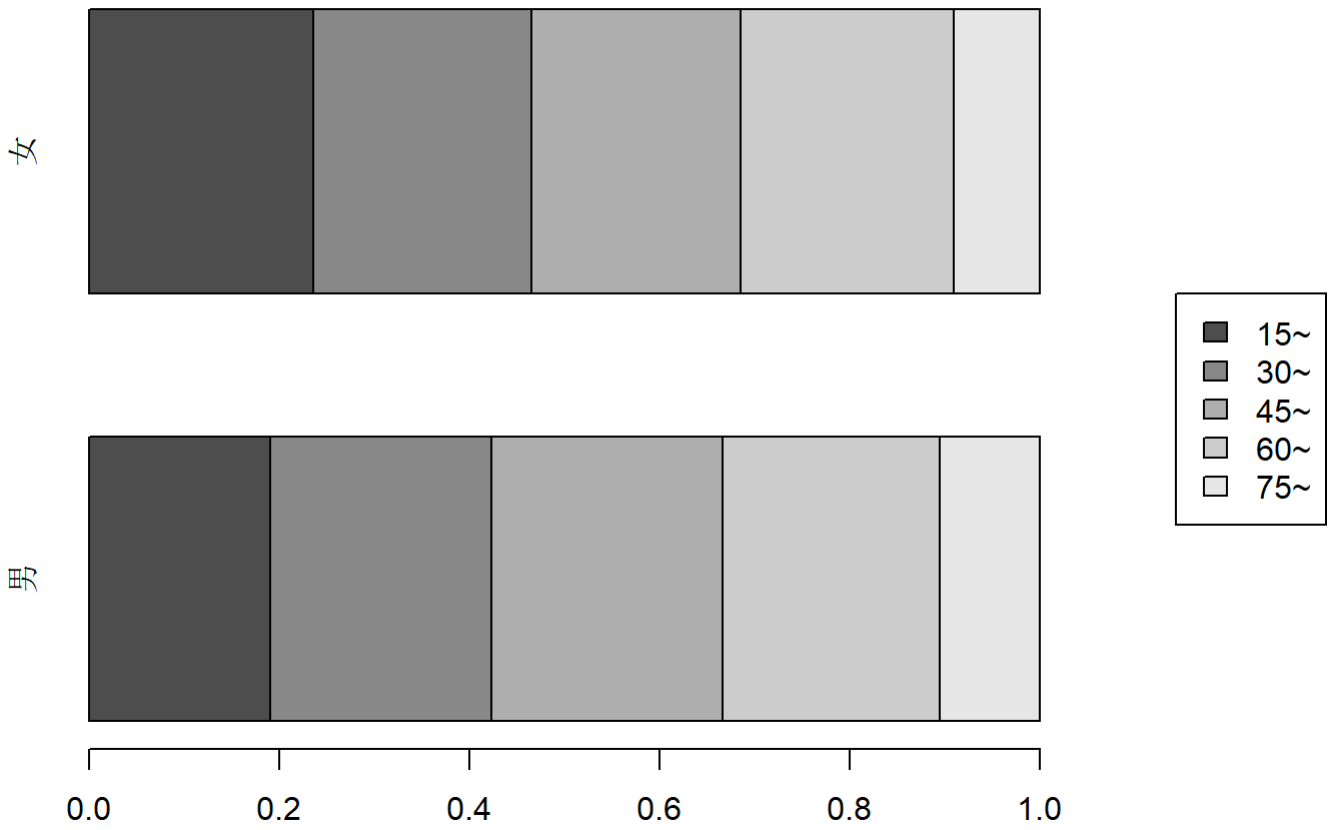

```
barplot(hypt,width=1,space=1,main="高血压患病率",xlab="年龄分组（岁）",ylab="高血压患病率",ylim=c(0.0,0.5),offset=0.1)
```




```
# 不同性别年龄分布
pie(table(cleandata$agegrp)) # 年龄分布，Pie
```

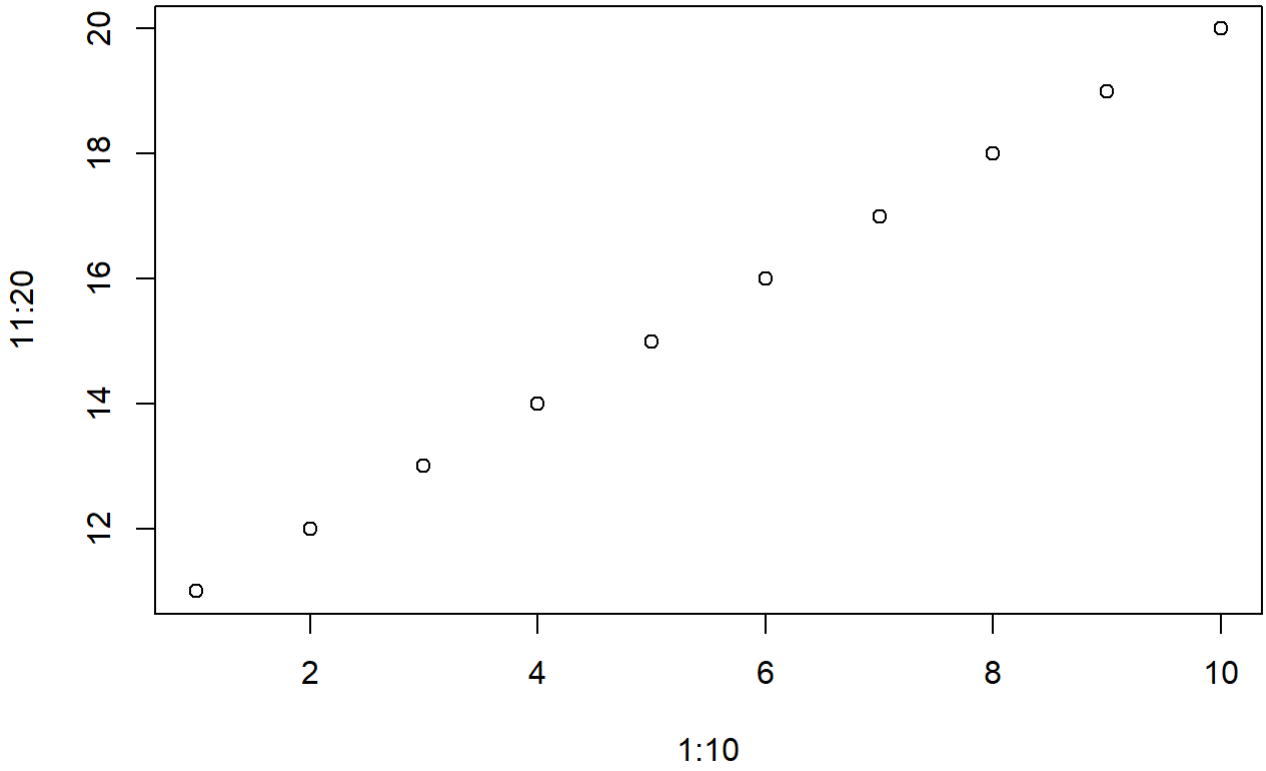


```
df <- prop.table(table(cleandata$agegrp,cleandata$sex),2)
par(mai=c(0.5,0.5,0.5,1.5))
barplot(df,width = 1,space=0.5,horiz = T,legend.text = c("15~","30~","45~","60~","75~"),args.legend =c(x=1.3,y=2))
```



6.3.2 - plot()

```
plot(1:10,11:20) # 散点图
```




```
plot(cleandata$age,cleandata$sbp) # 散点图
```