抽样调查R语言实现

—基于R软件包sampling和survey

刘玉坤

华东师范大学　统计学院

目录

目录

概述

★ 我们将介绍几种常见的抽样方法在R语言中的实现，包括

- 简单随机抽样
- 分层抽样
- 系统抽样
- PPS抽样
- 整群抽样
- 多阶抽样

★ 每种方法包括两方面的内容：抽样和参数估计。

- 进行抽样主要使用sampling 包，需要安装。
- 基于抽样结果进行参数估计，主要使用survey包，也需要安装。

sampling包中的主要函数及用途

抽样函数	用途
srswor	简单随机抽样
strata	分层抽样，各层可以使用“srswor”，“srswr”，“poisson”，“systematic”等等抽样方法
UPsystematic	系统抽样
UPmultinomial	PPS抽样
UPbrewer	Brewer抽样
cluster	整群抽样，可以使用“srswor”，“srswr”，“poisson”，“systematic”等等，默认是“srswor”
mstage	多阶抽样，可以使用“srswor”，“srswr”，“poisson”，“systematic”等等，默认是“srswor”
inclusionprobabilities	定义总体单元入样概率
getdata	从总体数据中调用样本全部信息

survey包中的主要函数及用途

估计函数	用　途
svydesign	定义抽样设计及抽样结果
svymean	均值估计及标准差估计
svytotal	总和估计及标准差估计
svyratio	比率估计及标准差估计
svyglm	回归估计及标准差估计
predict	对目标变量进行估计

准备工作

```
install.packages(c("sampling", "survey", "RSQLite",
       "quantreg", "hexbin", "mitools", grid))
```

抽样的实例数据是agpop.csv文件.

美国政府每五年做一次有关农业的普查，收集50个州所有农场的数据。所给的数据文件agpop.csv包含了3078个美国的县（或者县级市等）的农场的数据，包含了1982年、1987年和1992年每个县所拥有的农场个数(farms)，耕地面积(acres)，耕地面积小于9英亩的小农场数量(smallf)，耕地面积大于1000英亩的大农场数目(largef)等数据。

抽样的实例数据是agpop.csv文件.

- ★ 其中包括18个变量，其中cnum, snum, rnum分别是与county, state, region相对应的数字名义变量，表示对应的编号。
- ★ 一共有4个区域(region), 50个州(state)以及3041个县。
- ★ 我们用到的变量有县(county/cnum)、州(state/snum)、区域(region/rnum)、1992年每个县的耕地面积(acres92)、1987年每个县的耕地面积(acres87)、1992年每个县拥有的农场个数(farms92)。
- ★ 所有抽样方法的目标变量均为1992年的耕地面积(acres92)

数据集的预处理

★ 原始数据中若变量acres92<= 0或者acres87<= 0或者acres82<= 0，则表示缺失。

★ 为方便处理，导入数据后要把缺失数据删除。

```
data = read.csv("d:/agpop.csv", header = T,
      sep = ',')
attach(data)
index=acres92>0&acres87>0&acres82>0
data=data[index, ]
```

目录

简单随机抽样的实现

抽样要求：用简单随机抽样(不放回)抽取容量为300的样本。

```
N=nrow(data)
n=300
s=srswor(n,N)
data.srswor=getdata(data,s)
```

★ srswor(n,N) 返回一个长度为N的向量，仅取1或0，其中1的个数为n;

★ getdata(data，s) 表示根据数据s从总体data中确定入样的数据，即样本。

参数估计: 简单估计量

为了进行参数估计，需要把变量信息pw=N/n和fpc=N这两个加到数据集中

```
pw=rep(N/n, n)
fpc=rep(N, n)
agsrswor=as.data.frame(cbind(data.srswor, pw, fpc))
```

目标变量(acres92)均值和总和的简单估计及其标准差的估计

```
dsrswor = svydesign(id=~1, weights=~pw,
     data=agsrswor,  fpc=~fpc)
summary(dsrswor)
svymean(~acres92,dsrswor,deff=TRUE)
svytotal(~acres92,dsrswor,deff=TRUE)
```

参数估计：比估计

目标变量(acres92)均值和总和的比率估计及其标准差的估计

```
acres.ratio<-svyratio(~acres92,~acres87,dsrswor)
popm<-data.frame(acres87=mean(data$acres87))
predict(acres.ratio,popm$acres87)

acres.ratio<-svyratio(~acres92,~acres87,dsrswor)
popt<-data.frame(acres87=sum(data$acres87))
predict(acres.ratio,popt$acres87)
```

参数估计：回归估计

目标变量(acres92)均值和总和的回归估计及其标准差的估计

```
acres.reg<-svyglm(acres92~acres87,design=dsrswor)
popm<-data.frame(acres87=mean(data$acres87))
predict(acres.reg,newdata=popm)

acres.reg<-svyglm(acres92~acres87,design=dsrswor)
popt<-data.frame(acres87=sum(data$acres87))
predict(acres.reg,newdata=popt, tatal=N)
```

目录

分层随机抽样的实现

抽样要求：以region为分层变量, 每层用SRS抽取75个样本单元.

★ 定义分层抽样涉及到的变量：总体单元数N, 第h层单元数N_h，层权W_h，层数L, 各层样本量n_h.

```
N=nrow(data)
Nh=table(data$region)
Wh=Nh/N
L=length(unique(data$region))
nh=rep(75,L)
```

★ 调用分层函数strata

```
st=strata(data[order(data$region),],"region",
    nh,"srswor")
```

★ 调用getdata(data, s) 抽取数据

```
data.strata=getdata(data,st)
```

参数估计准备工作

★ 定义样本权重, 即入样概率的倒数

```
pw = 1/st$prob
```

★ 定义fpc变量

```
fpc=as.numeric(table(data$region)[data.strata$region])
```

★ 将权重和fpc变量加入到数据集中

```
agstrat=as.data.frame(cbind(data.strata,pw,fpc))
```

简单估计

★ 调用svydesign, 并查看抽样结果

```
dstrat<-svydesign(id=~1,strata=~region,
      weights=~pw, data=agstrat,fpc=~fpc)
summary(dstrat)
```

★ 目标变量(acres92)均值和总和的简单估计及其标准差的估计

```
svymean(~acres92, dstrat, deff=TRUE)
svytotal(~acres92, dstrat, deff=TRUE)
```

分别比率估计

★ 目标变量(acres92)均值的分别比率估计及其标准差的估计

```
sep.ratio=svyratio(~acres92, ~acres87, dstrat,
     separate=TRUE)
popm = data.frame(acres87=tapply(data$acres87,
     INDEX=data$region, FUN=mean))
predict(sep.ratio, popm$acres87*Wh)
```

★ 目标变量(acres92) 总和的分别比率估计及其标准差的估计

```
sep.ratio=svyratio(~acres92, ~acres87, dstrat,
     separate=TRUE)
popt = data.frame(acres87=tapply(data$acres87,
     INDEX=data$region, FUN=sum))
predict(sep.ratio, popt$acres87*Wh)
```

联合比率估计

★ 目标变量(acres92)均值的联合比率估计及其标准差的估计

```
com.ratio=svyratio(~acres92, ~acres87, dstrat)
popm = data.frame(acres87= mean(data$acres87))
predict(com.ratio, popm$acres87)
```

★ 目标变量(acres92) 总和的联合比率估计及其标准差的估计

```
com.ratio=svyratio(~acres92, ~acres87, dstrat)
popm = data.frame(acres87= sum(data$acres87))
predict(com.ratio, popm$acres87)
```

回归估计

★ 目标变量(acres92)均值的回归估计及其标准差的估计

```
com.reg=svyglm(acres92~acres87, dstrat)
popm = data.frame(acres87= mean(data$acres87))
predict(com.reg, newdata=popm)
```

★ 目标变量(acres92) 总和的回归估计及其标准差的估计

```
com.reg=svyglm(acres92~acres87, dstrat)
popt = data.frame(acres87= sum(data$acres87))
predict(com.reg, newdata=popt, total=N)
```

目录

系统抽样的实现

抽样要求：采用等距抽样法, 抽取容量为300的样本.

★ 定义入样概率，抽取样本下标，然后调用getdata提取样本单元

```
N=nrow(data)
n=300
pik=rep(n/N, N)
s=UPsystematic(pik)
data.sys = getdata(data,s)
```

参数估计

```
pw = 1/pik[s==1]
fpc=rep(N, n)
agsys=as.data.frame(cbind(data.sys, pw, fpc))

dsys=svydesign(id=~1, weights=pw,
     data=agsys, fpc=~fpc)
summary(dsys)
svymean(~acres92, dsys, deff=TRUE)
svytotal(~acres92, dsys, deff=TRUE)
```

目录

系统抽样的实现

抽样要求：以1992年每个县所拥有的农场个数(farms92)为规模变量，采用PPS抽样抽取样本容量为300的样本。

★ 调用inclusionprobabilities函数定义每个总体单元的入样概率，第一参数定义规模变量，第二个参数定义样本容量

```
N=nrow(data)
n=300
pik=inclusionprobabilities(data$farms92, n)
s=UPmultinomial(pik)
data.pps = data[s!=0, ]

## data.pps = getdata(data,s)
## 这句话等价于 data[s==1, ],   但是在PPS抽样是
## 有放回抽样,  所以s的某些元素会大于1.
```

参数估计: HH估计

★ 计算每次抽样中每个单元被抽中的概率Z_i, 等于π_i/n. 同时计算每个单元被抽中的次数Q_i.

```
Z= pik[s!=0]/n
Q= s[s!=0]
```

★ 目标变量(acres92)的总和的估计及其标准误差的估计

```
YHH = mean(data.pps$acres92/Z*Q)
vars= 1/(n*(n-1))*sum((data.pps$acres92/Z-YHH)^2*Q)
std = sqrt(vars)

YHHm=YHH/N
stdm=std/N
```

目录

系统抽样的实现

抽样要求：以1992年每个县所拥有的农场个数(farms92)为规模变量，采用Brewer抽样抽取样本容量为300的样本。

★ 调用inclusionprobabilities函数定义每个总体单元的入样概率，第一参数定义规模变量，第二个参数定义样本容量

```
n=300
pik=inclusionprobabilities(data$farms92, n)
s=UPbrewer(pik)
data.brewer= getdata(data,s)

## 等价的，  可以使用 data.brewer = data[s==1, ]
```

参数估计：HT估计

★ 定义每个单元入样的概率

```
p= pik[s== 1]
```

★ 将样本单元入样概率加入到样本单元的数据集中

```
agbrewer = as.data.frame(cbind(data.brewer, p))
```

★ 定义抽样设计及抽样结果，并查看

```
dbrewer = svydesign(id=~1, fpc=~p, data=agbrewer,
     pps="brewer")
summary(dbrewer)
```

★ 目标变量(acres92)均值的简单估计及其标准差的估计

```
svymean(~acres92, dbrewer, deff=TRUE)
svytotal(~acres92,dbrewer, deff=TRUE)
```

目录

整群抽样的实现

抽样要求：以state为分群变量，用SRS方法抽取5个群。

★ 设定样本群数为5.

```
n=5
```

★ 调用整群抽样函数cluster, 使用SRS抽群。

```
c1=cluster(data, "state", size=n, method="srswor",
    description=TRUE)
head(c1)                ### 查看 c1 的前几行数据
c=getdata(data, c1)     ### 抽出数据
```

参数估计

★ 定义每个单元入样的概率

```
N  = nlevels(data$state)        ##  总体中群的个数
fpc= rep(N, nrow(c))
pw = rep(N/n, nrow(c))
```

★ 把变量c, pw和fpc合并

```
agclus = as.data.frame(cbind(c, pw, fpc))
```

★ 定义抽样设计及抽样结果，并查看

```
dclus = svydesign(id=~state, weights=~pw,
        data=agclus, fpc=~fpc)
summary(dclus)
```

★ 目标变量(acres92)均值的简单估计及其标准差的估计

```
svymean(~acres92, dclus)## 也可以加入选项 deff=TRUE
svytotal(~acres92,dclus)
```

目录

等概率两阶抽样

抽样要求

1. 第一阶段以state为PSU（共50个），抽取若干PSU
2. 第二阶段，在每个PSU内部以county为SSU, 抽取若干SSU。两阶段均采用SRS抽样.

★ 抽样方法一: 调用多阶抽样函数：mstage.

- 该函数需要预先设定好每个阶段抽取的样本量。
- 先抽取25个PSU, 在每个PSU内部抽取10个SSU.

★ 抽样方法二：不用mstage，自行分阶段抽取。第一阶段仍然抽取25个，第二阶段抽取约20%的SSU.

抽样方法一的实现

★ 数据整理：由于第二阶段要抽取至少10个SSU, 所以要提出SSU个数少于10的PSU.

```
s.size   = table(data$state)[data$state]
data.new = data[s.size>=10, ]
```

★ 多阶抽样函数mstage要求数据框中的变量已经按照第一阶段变量、第二阶段变量排好次序：

```
data.new = data.new[order(data.new$snum, data.new$cnum), ]
```

★ 然后抽样

```
n  = 25;   mi = rep(10, n)
m1 = mstage(data.new, stage=list("cluster", ""),
     varnames=list("snum", "county"), size=list(n, mi),
     method=c("srswor","srswor"), description=TRUE)
```

★ 查看抽样结果和第一阶段抽取到的PSU. mstage返回的是两个抽样框，名字分别是'1'和'2'.

```
m=getdata(data.new, m1$"2")
```

抽样方法二的实现

★ 第一阶段抽样：调用抽样函数cluster进行第一阶段抽样。其中第一个参数是总体数据集，第二个参数是PSU变量，size 表示要抽取的PSU个数，method是抽样方法。

```
m1=cluster(data.new, "state", size=n,
   method="srswor", description=TRUE)
```

★ 查看抽样结果和第一阶段抽取到的PSU. cluster 返回的是一个包括PSU变量“state”，单元标志和入样概率"Prob"的数据框。

```
m=getdata(data.new, m1)
result=unique(m$state)
cat("PSU selected in stage 1:", result, '\n')
```

抽样方法二的实现(续)

★ 第二节阶段抽样

```
sm=NULL
for(i in 1:n)
{
   mi = m[m$state==result[i], ]
   ni=round(nrow(mi)/5)+1
   si=srswor(ni, nrow(mi))
   si=mi[si!=0, ]
   sm=rbind(sm, si)
}
```

参数估计

★ 准备工作

```
fpc1= rep(50, nrow(sm))
fpc2= table(data.new$state)[sm$snum]
sm2 = as.data.frame(cbind(sm, fpc1, fpc2))
dsm = svydesign(id=~state+county, fpc=~fpc1+fpc2,
      data=sm2)
summary(dsm)
```

★ 目标变量(acres92)均值的简单估计及其标准差的估计

```
svymean(~acres92, dsm)## 也可以加入选项 deff=TRUE
svytotal(~acres92, dsm)
```

目录

PPS+SRS的两阶抽样

抽样要求

1. 第一阶段以region为PSU（共4个），从中以PPS抽样抽取2个PSU
2. 第二阶段，在每个PSU内部以county为SSU，用SRS抽样抽取150个SSU。

★ 第一阶段抽样：PPS抽样

```
M = table(data$region)
z = rep(M/sum(M), M)
n = 2
ind = cluster(data, "region", size=n,
    method="srswr", pik=z, description=TRUE)
```

第二阶段抽样

★ 查看第一阶段抽样结果

```
m = getdata(data, ind)
result=unique(m$rnum)
cat("Clusters selected in stage 1: ",  result, '\n')
```

★ 第二阶段抽样:

```
sm = NULL
for(i in 1:length(result))
{
   mi = m[m$snum==result[i], ]
   si=srswor(150, nrow(mi))
   si=mi[si!=0, ]
   sm=rbind(sm, si)
}
```

参数估计

```
#############################################
####   estimate mean
  M0=nrow(data)
  mean =  mean(ybar)
  SE.mean =  sqrt(var(ybar)/M0)
  cbind(mean, SE.mean)

####   estimate total
  total = M0*mean
  SE.total = M0*SE.mean
  cbind(total, SE.total)
```