# 第四章 R中参数的估计和检验

本章的主题为如何使用R来估计参数并检验，这是数据分析中较基础的一部分内容，参数的估计和检验也是方差分析、回归分析的基础。通过阅读本章内容，读者将对参数的估计和检验有一个系统的了解，并掌握如何使用R进行参数估计与检验。

## 4.1 使用R进行点估计和区间估计

点估计和区间估计是参数估计的两种主要方法。点估计为参数估计出一个确定的值，区间估计则为参数估计出一个区间。本小节比较了这两种估计方法的优劣，并用R实现了参数的点估计和区间估计。

### 4.1.1 简单的点估计法和区间估计

在统计分析中，总体的数据往往是不能全部获得的，很多时候都只能获得一部分样本数据。比如质检中心关心一批新灯泡的寿命均值，但质检中心不能把这批灯泡全部拿来测寿命，只能从中抽取一部分灯泡出来，这部分灯泡就称为样本。质检中心通常用样本数据来估计总体的寿命均值，而这种估计过程就被称为参数估计。

参数估计主要分为点估计和区间估计，点估计又分为矩估计、极大似然估计等。矩估计是计算最方便、应用最广泛的一种估计方法，它用样本数据的一阶矩来估计总体的一阶矩，用样本的二阶矩来估计总体的二阶矩，再通过构造方程组来估计总体参数。简单来说，矩估计认为样本的均值就近似为总体的均值，样本的方差就近似为总体的方差。

> BJsales.sam <- sample(BJsales,50)

> mean(BJsales);mean(BJsales.sam)

[1] 229.978

[1] 228.912

BJsales数据集中有150份销售数据，我们不妨将它看做总体，从中抽取50个数据作为样本。sample()函数能够从总体中随机抽取样本，上述代码中sample()函数的第一个参数指定从BJsales中抽取数据，第二个参数指定抽取50份数据，sample()函数还提供了replace参数，当它为真时，总体中的数据允许被重复抽取。

查看BJsales和BJsales.sam的均值，这两个均值的差不超过1.1，此时用样本的均值来估计总体的均值是较为合理的，矩估计法在这里发挥了应有的作用。

> 149/150\*var(BJsales);var(BJsales.sam)

[1] 458.3010

[1] 438.7488

当用样本方差估计总体方差时，矩估计法的误差稍微大了一些，上述代码查看了分别样本和总体的方差。方差的计算公式为，但var()函数将所有被应用对象都看做样本，因此使用var()函数计算方差时，分母是n-1，而非n。上述代码中“149/150\*var(BJsales)”给出了总体的真实方差。矩估计法认为样本方差与总体方差之间相差一个系数，因此“var(BJsales.sam)”计算得到的结果恰好就是根据样本数据估计的总体方差。

观察R返回的结果，总体方差的估计值和实际值的差值约为20，虽然我们知道差值越小，点估计的效果就越好，但显然，很难说清估计值和实际值的差值小到什么地步时才能说点估计的效果很好。因此，区间估计的引入是有必要的。

样本参数与总体参数之间必然存在一定的联系，但它们又不大可能完全相等。区间估计认为总体参数总是落在样本参数的附近，以样本参数为中心，必然能画出一个区间来覆盖住总体参数。由于样本参数服从的分布是已知的，因此根据置信度和样本参数找出一个置信区间就是可行的。

以均值为例，当总体方差已知时，样本均值服从以总体均值为均值、以总体方差为方差的正态分布，由于样本均值已知，因此可计算出一个置信区间，其中为样本均值，为总体的标准差，为样本数据个数，为标准正态分布的上分位点。当总体方差未知时，样本均值则服从自由度为n-1的t分布，这时置信区间就变为了，其中S为样本的标准差，则为自由度为n-1的t分布的上分位点。

> mean.est <- function(x,sd=-1,alpha){

+ n <- length(x);m <- mean(x)

+ if (sd>=0) {

+ tmp <- sd/sqrt(n)\*qnorm(1-alpha/2)

+ }

+ else {

+ tmp <- sd(x)/sqrt(n)\*qt(1-alpha/2,n-1)

+ }

+ data.frame(mean=m,sub=m-tmp,sup=m+tmp)

+ }

> mean.est(BJsales.sam,sd=sd(BJsales),alpha=0.05)

mean sub sup

1 228.912 222.9582 234.8658

> mean.est(BJsales.sam,alpha=0.05)

mean sub sup1 228.912 224.4521 236.9479

上述代码根据置信区间的计算公式编写了函数mean.est()，并将mean.est()函数应用到了样本数据中。mean.est()函数一共有三个形参，第一个参数x准备传入需要被计算的样本数据；第二个参数sd准备传入总体数据的标准差，sd已经设置为-1，当总体数据标准差未知时，就不必向函数中传入数据了；第三个参数alpha准备传入的是置信度，alpha越大，给出的置信区间就越大，结果就越不可信；alpha越小，给出的置信区间就越小，结果就越可信。

在函数的代码块中ifelse语句实现了分流，令样本数据按照总体方差已知或未知分别应用不同的计算公式，其中qnorm()函数和qt()函数能分别查询正态分布表和t分布表。函数块的最后一行语句输出了一个由均值、区间左端点、区间右端点构成的数据框。

在设置完均值的置信区间计算函数后，上述代码的最后两行分别调用了该函数，其中第一次调用时给出了总体数据的标准差，第二次调用时并未给出，这两次调用设置的置信度都是0.95。观察R返回的结果并和BJsales的均值作比较，显然，这两个置信区间都覆盖了总体均值，当总体方差已知时计算得出的置信区间要小一些，估计效果更好一些。

估计总体方差的置信区间与估计总体均值类似。当总体均值已知时，总体方差的置信区间计算公式为，其中表示用总体均值计算得到的样本方差，和则分别表示自由度为n的卡方分布的上和分位数。当总体均值未知时，总体方差的置信区间计算公式为，其中指的是样本方差。

> var.est <- function(x,m=Inf,alpha){

+ n <- length(x)

+ if (m<Inf) {

+ var <- sum((x-m)^2)/n

+ df <- n

+ }

+ else {

+ var <- var(x)

+ df <- n-1

+ }

+ sub <- df\*var/qchisq(alpha/2,df)

+ sup <- df\*var/qchisq(1-alpha/2,df)

+ data.frame(var,sub,sup)

+ }

> var.est(BJsales.sam,m=mean(BJsales),alpha=0.05)

var sub sup

1 442.5843 683.9004 309.8453

> var.est(BJsales.sam,alpha=0.05)

var sub sup

1 449.8278 698.5144 313.8822

与估计总体均值类似，上述代码首先创建了一个用于估计总体方差的函数var.est()，这个函数有三个形参，其中第一个参数准备传入样本数据；第二个参数准备传入总体均值，在这里我们已经规定了m为Inf，即无穷大，当总体均值未知时，就不需要再向函数中传入数据了；第三个参数准备传入置信度。函数块同样使用ifelse语句实现了分支，其中qchisq()函数用于查询卡方分布表。

在设置好var.est()函数后，上述代码块计算量总体均值已知和总体均值未知两种情况下的置信区间。与BJsales的实际方差相比，这两个置信区间都覆盖了真实值，在总体均值已知时，置信区间更小一些，估计效果更好一些，这与计算总体均值时一致，说明有关总体的信息知道的越多，对总体参数的估计就越准确。

### 4.1.2 估计单侧置信区间

与点估计相比，区间估计不但能估计双侧置信区间，也能估计单侧置信区间。有时候我们只关心总体参数落在样本参数某一侧的可能性大小，这时单侧置信区间就发挥作用了。与双侧置信区间相比，单侧置信区间的可能性较多，计算公式也较多一些。

对总体均值估计单侧置信区间时，总体方差已知或总体方差未知时的置信区间计算公式并不相同。当总体方差已知时，总体均值的置信区间为或，当总体方差未知时，总体均值的置信区间为或。

> BJsales.sam <- sample(BJsales,50)

> mean.est <- function(x,sd=-1,side,alpha){

+ n <- length(x);m <- mean(x)

+ if (sd>=0) {

+ if (side<0) {

+ tmp <- sd/sqrt(n)\*qnorm(1-alpha)

+ sub <- -Inf;sup <- m+tmp

+ }

+ else if (side>0) {

+ tmp <- sd/sqrt(n)\*qnorm(1-alpha)

+ sub <- m-tmp;sup <- Inf

+ }

+ else {

+ tmp <- sd/sqrt(n)\*qnorm(1-alpha/2)

+ sub <- m-tmp;sup <- m+tmp

+ }

+ }

+ else {

+ if (side<0) {

+ tmp <- sd/sqrt(n)\*qt(1-alpha,n-1)

+ sub <- -Inf;sup <- m+tmp

+ }

+ else if (side>0) {

+ tmp <- sd/sqrt(n)\*qt(1-alpha,n-1)

+ sub <- m-tmp;sup <- Inf

+ }

+ else {

+ tmp <- sd/sqrt(n)\*qt(1-alpha/2,n-1)

+ sub <- m-tmp;sup <- m+tmp

+ }

+ }

+ data.frame(mean=m,sub,sup)

+ }

上述代码创建了一个新的mean.est()函数。与之前的mean.est()函数相比，新函数增加了一个side参数，在原有函数的基础上扩展了计算单侧置信区间的功能。当side参数为负数时，mean.est()函数将计算左侧为负无穷的置信区间；当side参数为正数时，mean.est()函数将计算右侧为正无穷的置信区间；当side参数为0时，mean.est()函数将计算双侧置信区间。

mean.est()函数的函数块中首先用ifelse语句分开了总体方差已知与未知的两种情况，在每一种情况下，又嵌套了两组ifelse语句来判断side参数与0 的关系，每一个分支的最末端都由置信区间的计算公式构成。

> mean.est(BJsales.sam,side=1,alpha=0.05)

mean sub sup1 230.7 230.4629 Inf

> mean.est(BJsales.sam,side=-1,alpha=0.05)

mean sub sup1 230.7 -Inf 230.9371

> mean.est(BJsales.sam,side=0,alpha=0.05)

mean sub sup1 230.7 230.4158 230.9842

创建好新的mean.est()函数后，上述代码查看了BJsales.sam的两个单侧置信区间以及双侧置信区间。此时的双侧置信区间与之前的双侧置信区间已不再一致，这是由于我们在创建新的mean.est()函数时重新抽取了一组样本数据，因此置信区间也会发生变化。

观察两个单侧置信区间，它们中的实数端点的值与均值的差是相同的，这是由于计算单侧置信区间宽度的公式是一样的。但这个差值与双侧置信区间端点和均值的差值并不相同，这是由于与并不成倍数关系。同样的，与也不成倍数关系。

计算总体方差的单侧置信区间时同样需要区分总体均值是否已知。当总体均值已知时，总体方差的单侧置信区间为或，当总体均值未知时，总体方差的单侧置信区间为或。

> var.est <- function(x,m=Inf,side,alpha){

+ n <- length(x)

+ if (m<Inf) {

+ var <- sum((x-m)^2)/n;df <- n

+ }

+ else {

+ var <- var(x);df <- n-1

+ }

+ if (side<0) {

+ sub <- 0

+ sup <- df\*var/qchisq(alpha,df)

+ }

+ else if (side>0) {

+ sub <- df\*var/qchisq(1-alpha,df)

+ sup <- Inf

+ }

+ else {

+ sub <- df\*var/qchisq(1-alpha/2,df)

+ sup <- df\*var/qchisq(alpha/2,df)

+ }

+ data.frame(var,sub,sup)

+ }