1220086621 朱金秋 作业8

14.8

对于统计量

$$\hat{\theta} = \frac{\sum_1^n (A_i - \bar{A})(B_i - \bar{B})}{[\sum_1^n (A_i - \bar{A})^2 \sum_1^n (B_i - \bar{B})^2]^{1/2}}$$

其中 spatial 数据中 A 为$\boldsymbol{a} = (a_1, a_2, \dots, a_{26})$，B 为$\boldsymbol{b} = (b_1, b_2, \dots, b_{26})$

那么$\hat{\theta}$ *为* $26 * 2$ *的矩阵*

对$\hat{\theta}$的统计量进行 bootstrap，抽 20 次，其结果如下

```
> library(bootstrap)
> xdata <- spatial
> nrow(spatial)
[1] 26
> theta <- function(x,xdata){ cor(xdata[x,1],xdata[x,2]) }
> results <- bootstrap(1:n,20,theta,xdata)
> results
$thetastar
 [1] 0.7799398 0.8698284 0.8165488 0.7787068 0.8794446 0.8082096
 [7] 0.7734456 0.7657123 0.6721984 0.7914961 0.7607234 0.8528100
[13] 0.7564240 0.7640691 0.8507722 0.8436442 0.7557171 0.8340472
[19] 0.8498645 0.8368937
```

14.14 解：

根据题目要求我们分别再 95%置信水平中计算 $\theta = E(D_1 - D_2)$BCa 与 Percentile

置信区间。

根据下面程序计算结果

```
   Percentile             BCa
(-15.87, -15.25 )   (-15.86, -15.21 )
```

```
> library(boot)
> library(bootstrap)
> tooth_D <- tooth[,c(2,3)]
> mean(tooth_D$D1)
[1] -5.647385
> mean(tooth_D$D2)
[1] 9.946769
> re <- mean(tooth_D$D1)-mean(tooth_D$D2)
> -15.59415
[1] -15.59415
> re <- mean(tooth_D$D1)-mean(tooth_D$D2)
> #定义统计量theta
> diff.means <- function(d, f){
+   x1m <- mean(tooth_D$D1[f])
+   x2m <- mean(tooth_D$D2[f])
+   ss1 <- var(tooth_D$D1[f])
+   ss2 <- var(tooth_D$D2[f])
+   c(x1m - x2m, (ss1 + ss2)/(sum(f) - 2))
+ }
> tooth.boot <- boot(tooth_D, diff.means, R = 999)
> boot.ci(tooth.boot, type = c("perc", "bca"))
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 999 bootstrap replicates

CALL :
boot.ci(boot.out = tooth.boot, type = c("perc", "bca"))

Intervals :
Level      Percentile            BCa
95%   (-15.87, -15.25 )   (-15.86, -15.21 )
Calculations and Intervals on Original Scale
```

14.15

统计量 $\theta_1 = logE(A/B)$ ， $\theta_2 = Elog(A/B)$

分别在 95%之心水平下计算二者 BCa 置信区间

θ_1:

```
Intervals :
Level       BCa
95%   (-0.0538,   0.1638 )
```

θ_2

```
Intervals :
Level       BCa
95%   (-0.0643,   0.1769 )
```

二者的置信区间不一样，且θ_1的置信区间更小一些，这是因为log（sum）比 sum(log)运算幅度小一些

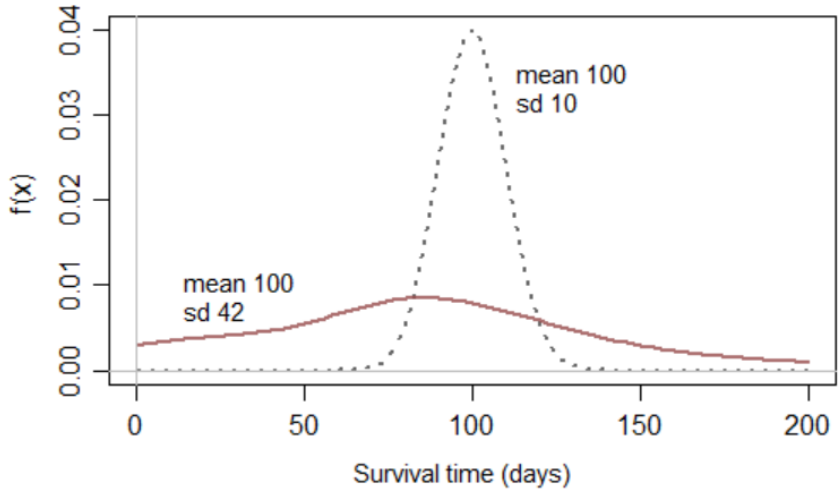
```
> library(boot)
> library(bootstrap)
> log(mean(spatial[,1]/spatial[,2]))
[1] 0.01536205
> mean(log(spatial[,1]/spatial[,2]))
[1] -Inf
> ###BCa方法
> spatial_new <- spatial[!rownames(spatial) %in% c("V14") , ]#去0
> n_2 <- nrow(spatial_new)
> #theta_1
> theta_1 <- function(d, f){
+   x1m <- log(mean(spatial_new$A[f]/spatial_new$B[f]))
+   v1 <- var(spatial_new$A[f]/spatial_new$B[f])
+   c(x1m,v1)
+ }
> spatial1.boot <- boot(spatial_new,theta_1,R = 999)
> boot.ci(spatial1.boot, type = c("bca"))
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 999 bootstrap replicates

CALL :
boot.ci(boot.out = spatial1.boot, type = c("bca"))

Intervals :
Level       BCa
95%   (-0.0538,  0.1638 )
Calculations and Intervals on Original Scale
```

```
> #theta_2
> theta_2 <- function(d, f){
+   x2m <- mean(log((spatial_new$A[f]/spatial_new$B[f])))
+   v2 <- var(spatial_new$A[f]/spatial_new$B[f])
+   c(x2m,v2)
+ }
> spatial2.boot <- boot(spatial_new,theta_1,R = 999)
> boot.ci(spatial2.boot, type = c("bca")) #计算bca
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 999 bootstrap replicates

CALL :
boot.ci(boot.out = spatial2.boot, type = c("bca"))

Intervals :
Level       BCa
95%   (-0.0643,  0.1769 )
Calculations and Intervals on Original Scale
```

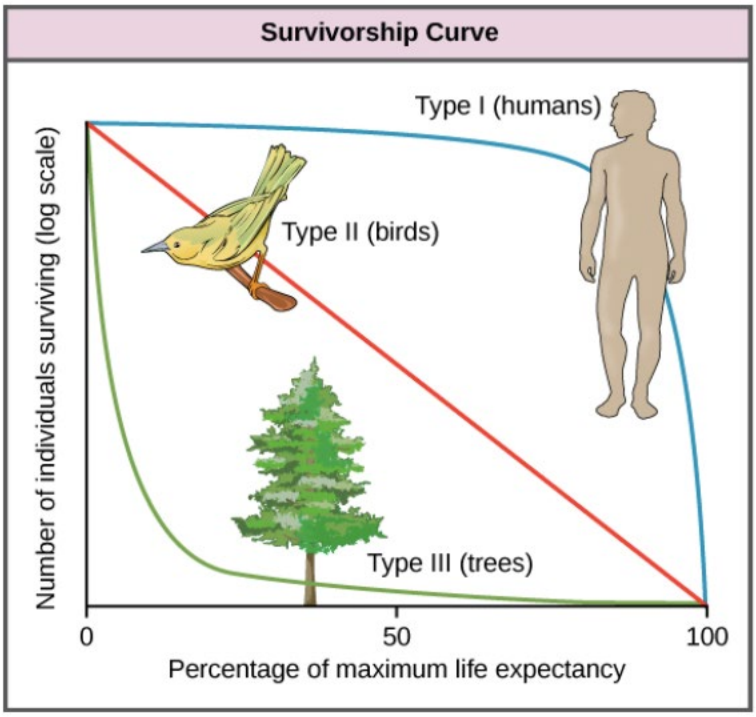
第十六章

16.1

因为生存时间算法 logarithm of survival times 满足正态分布的特性，对于单个种群，生存时间长的，或者很短的种群中分布很少，而大部分集中分布在生存时间均值附近。因此有正太分布的特性。



而各自的单个种群的生存时间来说，并没有那种特性



exponential distribution is used for modeling survival is due to the life strategies involved in organisms living in nature.

16.4

（a）计算 ASL

```r
> N <- 100
> Z <- matrix(NA,nrow = N,ncol = 7)
> for (i in 1:N) {
+   Z[i,] <- rnorm(7, mean = 129, sd = 66.8)
+ }
> #bootstrap
> B <- 1000
> theta <- matrix(NA,nrow = N,ncol = B)
> mean_row <- c()
> for (t in 1:N) {
+   mean_row[t] <- mean(Z[t,])
+ }
> for (j in 1:N) {
+   for (i in 1:B) {
+     Index_1 <- sample(c(1:7),7,replace=T);
+     z_boot <- c()
+     z_boot <- Z[j,]
+     data_1 <- z_boot[Index_1]
+     mean_z <- mean(data_1)
+     sd_z <- sd(data_1)
+     theta[j,i] <- (mean_z-mean_row[j])*sqrt(7)/sd_z
+   }
+ }
> t_obs <- c()
> for (m in 1:N) {
+   t_obs[m] <- (mean(Z[m,])-129)*sqrt(7)/sd(Z[m,])
+ }
> #求ASL
```

```r
> (Avg_ASL <- mean(ASL_boot))
[1] 0.52614
```

由于$\widehat{ASL} > 0.1$不拒绝原假设，$\mu_Z = 129$成立。

（b）计算 ASL

```
> Z <- matrix(NA,nrow = N,ncol = 7)
> for (i in 1:N) {
+     Z[i,] <- rnorm(7, mean = 170, sd = 66.8)
+ }
> #bootstrap
> B <- 1000
> theta <- matrix(NA,nrow = N,ncol = B)
> mean_row <- c()
> for (t in 1:N) {
+     mean_row[t] <- mean(Z[t,])
+ }
> z_boot <- c()
> for (j in 1:N) {
+     for (i in 1:B) {
+         Index_1 <- sample(c(1:7),7,replace=T)
+         z_boot <- Z[j,]
+         data_1 <- z_boot[Index_1]
+         mean_z <- mean(data_1)
+         sd_z <- sd(data_1)
+         theta[j,i] <- (mean_z-129)*sqrt(7)/sd_z
+     }
+ }
> ASL_boot <- c()
> for (h in 1:N) {
+     count <- 0
+     for (k in 1:B) {
+         if(abs(theta[h,k]) >= abs(t_obs[h]))
+         {count <- count + 1}
+         else
+         {count <- count}
+     }
+     ASL_boot[h] <- count/B
+ }
> ASL_boot
```

```
> (Avg_ASL <- mean(ASL_boot))
```

[1]0.0850854

由于$\widehat{ASL} < 0.1$拒绝原假设，接受备择假设，$\mu_Z = 129$不成立。