Metdologia Bootstrap

By: S. Civit, MNPR, 17-18.

April, 2018

Contents

Resampling

- **Resampling/remostreig** terme que empreem i que porta implícit una varietat de mètodes estadístics basats en les dades disponibles (mostres) en lloc d'un conjunt d'assumpcions estàndard sobre població/ns subjacent/s.
- Aquest mètodes inclouen **permutation tests, bootstrap and jackknife**

Metodologia Bootstrap

Introducció

L'enfoc bootstrap és més general que els anteriors. Ens proporciona una metodologia general per intentar determinar la distribució mostral d'un estadístic d'interés.

Un cop coneguda aquesta distribució (o alguns aspectes rellevants d'ella, com les probabilitats o els quantils a les cues), es poden intentar realitzar els processos habituals en inferència estadística, com ara obtenir un interval de confiança o resoldre un contrast d'hipòtesis.

Pensem novament en l'exemple que és fil conductor de tot aquest tema. L'objectiu és comparar, respecte de la variable EEAUC, el grup de dones D amb el grup P. El paràmetre principal d'interés és la diferència entre les mitjanes poblacionals (o qualsevol altra mesura de localització adequada) d'ambdos grups. Diguem-ne 'delta' d'aquest paràmetre. Suposem que, de moment, hem decidit basar la nostra inferència en l'estadístic t

per dues mostres: t = ((‘mitjana mostral D’ - ‘mitjana mostral P’) - delta) / (s * sqrt(1/n1 + 1/n2)) on ‘s’ representa l’estimació global (pooled) de la desviació estàndard.

La idea central del mètode bootstrap és substituir a G = G(F) la distribució de les dades, F, desconeguda, per una estimació obtinguda a partir de les pròpies dades. Encara que la idea és d’aplicabilitat molt general, la il · lustrarem pel cas del nostre exemple concret: a G(t, n1, n2, FD, FP) podríem substituir FD i FP per les corresponents distribucions empíriques o mostrals, FDn1 i FPn2, que en són una bona estimació no paramètrica.

Teòricament, per diversos teoremes de convergència bastant generals, atès que FDn1 i FPn2 convergeixen cap a FD i FP, G(t, n1, n2, FDn1, FPn2) és una bastant bona aproximació de G. Aquesta idea és senzilla, el seu problema principal és que no hi ha manera analítica de deduir la fórmula de G(t, n1, n2, FDn1, FPn2). Als anys 80 del segle XX, Bradley Efron va proposar que, de la mateixa manera que la distribució de t la podem aproximar molt bé per simulació si especifiquem completament els models teòrics FD i FP, G(t, n1, n2, FDn1, FPn2) es pot aproximar simulant un nombre gran, B, de mostres obtingudes a partir de FDn1 i FPn2 (ara completament conegudes), sobre cada una d’aquestes mostres calculant l’estadístic (t) i aproximant G(t, n1, n2, FDn1, FPn2) (que al seu torn aproxima G(t, n1, n2, FD, FP)) a partir d’aquesta gran mostra de B valors de t.

Recordant la Situació Experimental

En un context biomèdic i a partir de N = 22 dones, aleatòriament assignades n = 11 a rebre la droga D i n = 11 un placebus P. Totes 22 prenien un anticonceptiu amb dos components: etinil estradiol (EE) i noretindrona (NET). Es volia estudiar si la presència de la droga D influia en en els nivells de EE i NET (i, per tant, en l’efectivitat i/o seguretat de l’anticonceptiu). El nivell dels components de l’anticonceptiu es mesurava mitjançant la variable “àrea sota la corba” (AUC) que designarem EEAUC i NETAUC respectivament per EE i NET (i també amb la concentració màxima Cmax, que no considerem aquí)

Aproximación descriptiva

A tot el que segueix ens centrarem en la variable EEAUC.

```
dat<-read.table("C:/Users/Usuario/Desktop/anticonceptive.txt", header=TRUE)
dat
```

```
##    Suj tratam  EEAUC   NETAUC
## 1    1      P 2623.0 197525.0
## 3    2      D 3756.2 176487.5
## 5    3      P 2227.7 151957.5
## 7    4      D 2986.9 116305.0
## 9    5      D 3858.2 125869.0
## 11   6      P 4493.7 156367.5
## 13   7      P 2985.3 184942.5
## 15   8      D 3328.1 118340.0
## 17   9      D 3005.8 176590.0
## 19  10      P 1919.7 123012.5
## 21  11      P 2496.1 122987.5
## 23  12      D 5624.5 207327.5
## 25  13      P 3016.0 173565.0
## 27  14      D 3354.9 197912.5
## 29  15      P 2985.2 156367.5
## 31  16      D 3282.8 182250.0
## 33  17      P 3385.5 201557.5
```

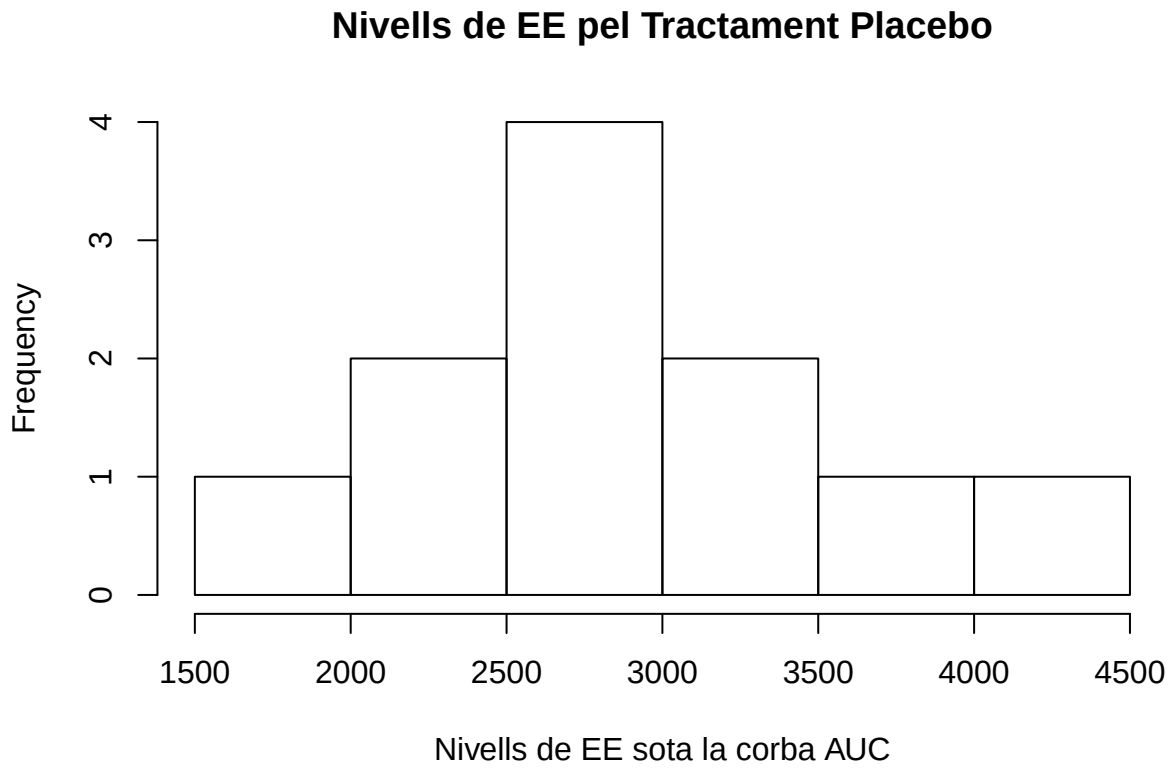
```
## 35  18      D 5018.0 278532.5
## 37  19      P 2622.0 155790.0
## 39  20      D 2472.0  86467.5
## 41  21      D 3819.6 142359.8
## 43  22      P 3863.5 232197.5
```

```
class(dat$EEAUC)
```

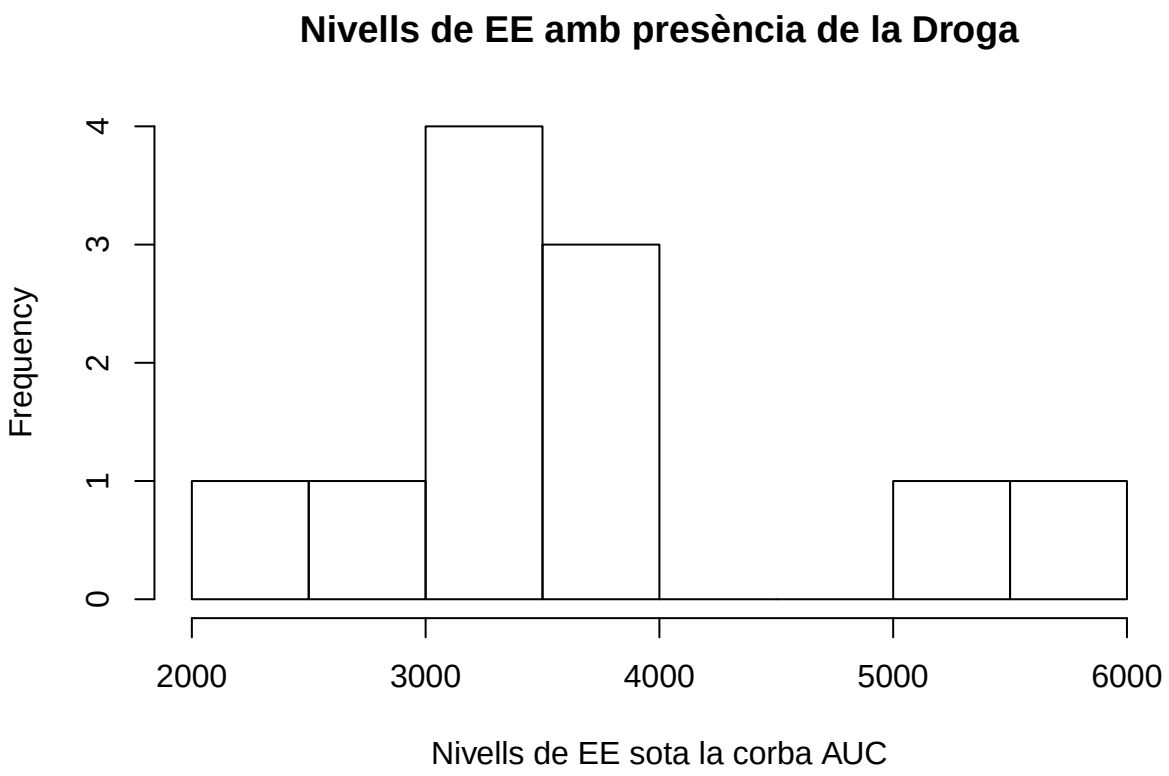
```
## [1] "numeric"
```

Indicis de normalitat (o no) de les dades:

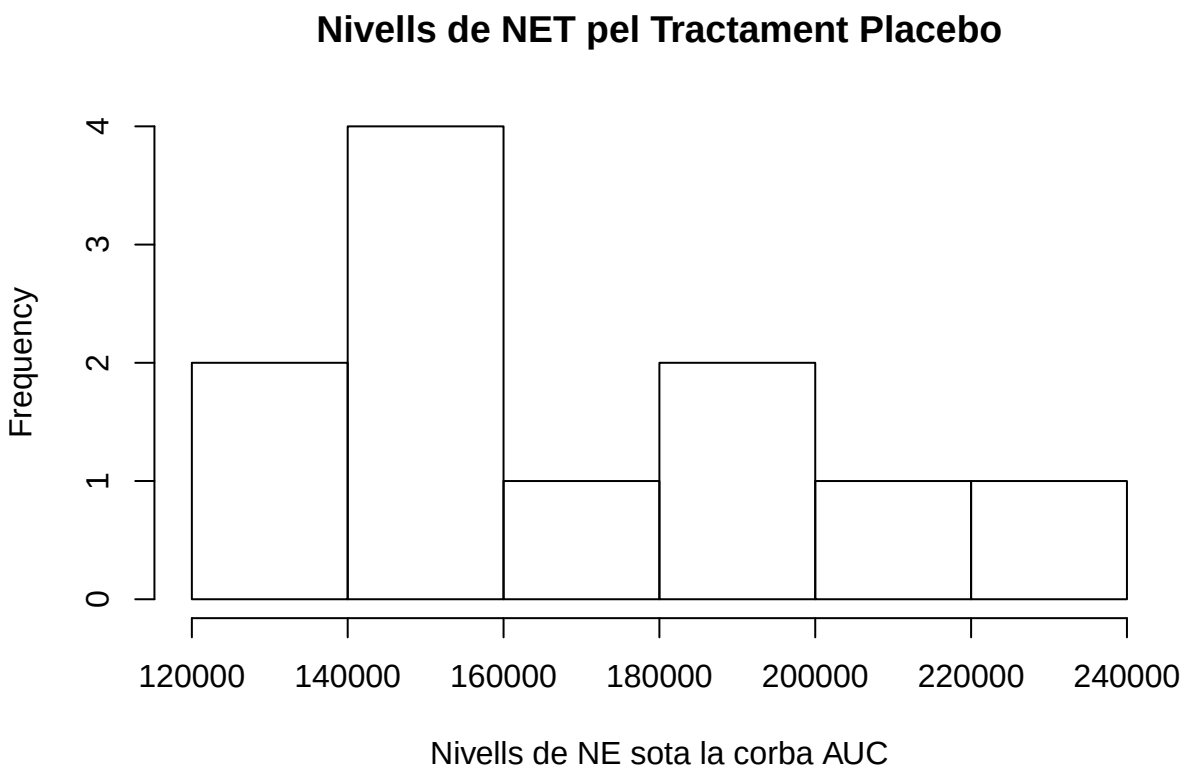
```
hist(dat[dat[,"tratam"]=="P","EEAUC"], main="Nivells de EE pel Tractament Placebo",  xlab= "Nivells de
```



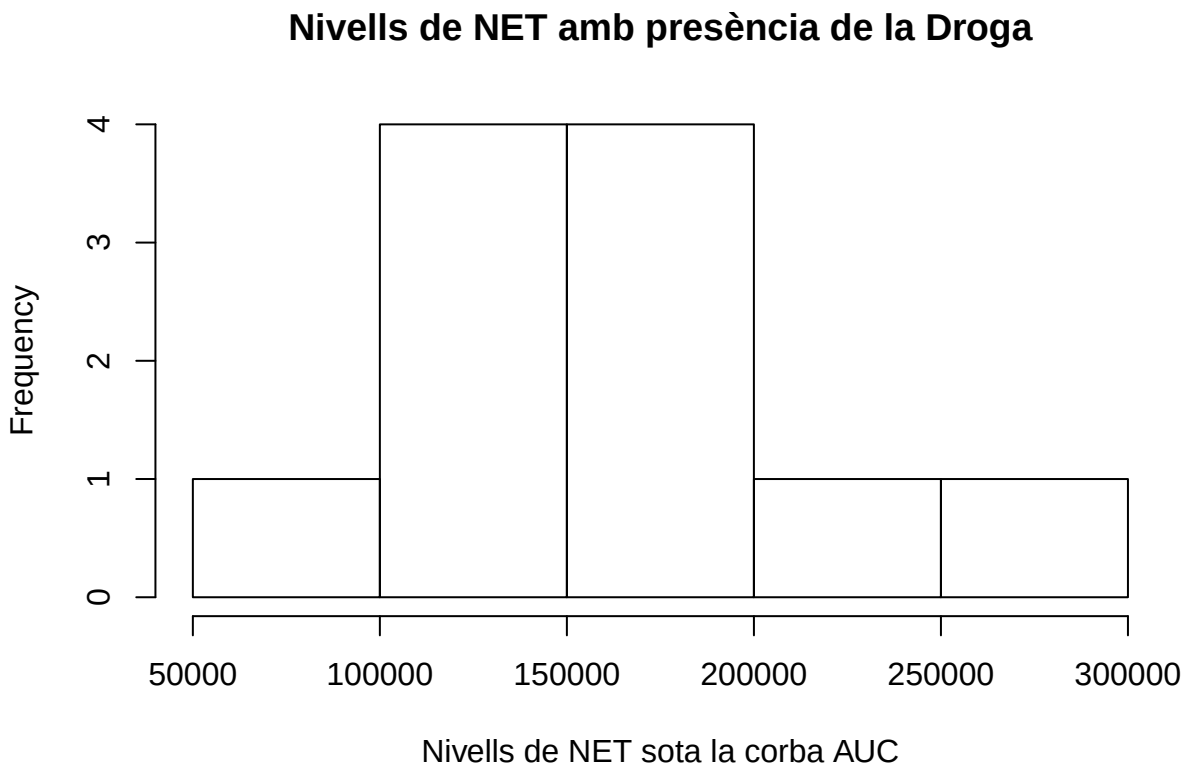

```
hist(dat[dat[,"tratam"]=="D","EEAUC"],main="Nivells de EE amb presència de la Droga",  xlab= "Nivells d
```




```
hist(dat[dat[,"tratam"]=="P","NETAUC"],main="Nivells de NET pel Tractament Placebo",  xlab= "Nivells de
```

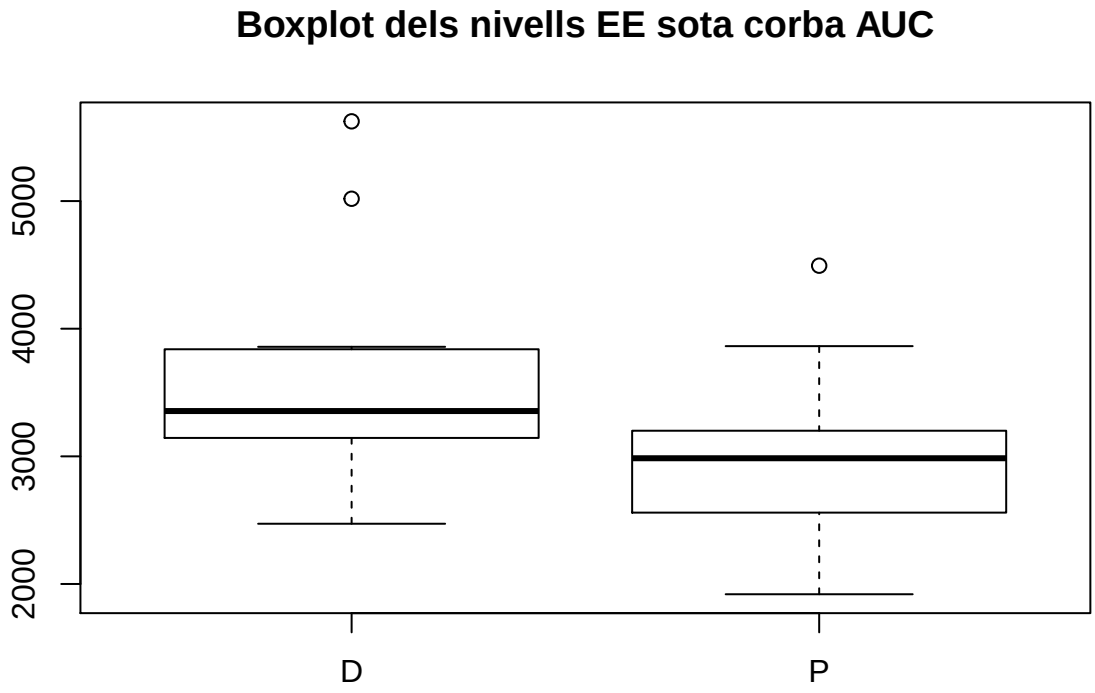



```
hist(dat[dat[,"tratam"]=="D","NETAUC"],main="Nivells de NET amb presència de la Droga",  xlab= "Nivells
```

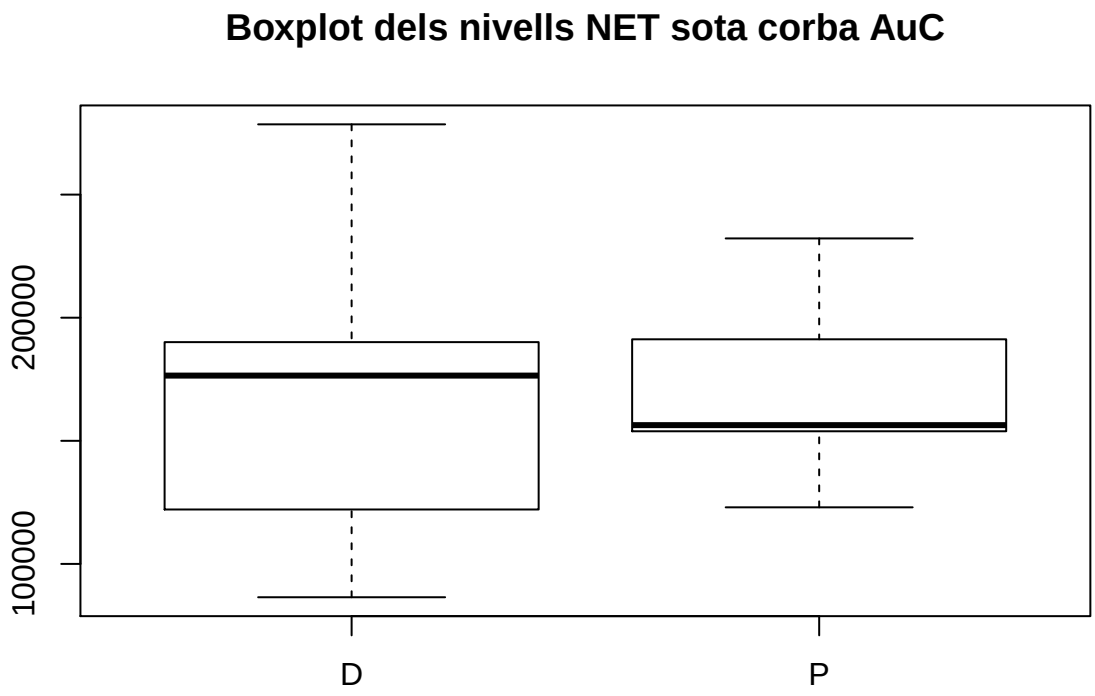


Indicis d'igualtat de la dispersió (o no) de les dades P vs D

```
boxplot(EEAUC ~ tratam, main="Boxplot dels nivells EE sota corba AUC",data = dat)
```




```
boxplot(NETAUC ~ tratam, main="Boxplot dels nivells NET sota corba AuC",data = dat)
```



Mides mostrals i re-ordenant les dades:

```
n <- tapply( dat[,"EEAUC"], dat[,"tratam"], length)
n
```

```
##  D  P
## 11 11
```

```
# Ara 'n' conté les mides mostrals de cada grup, n[1] == n[2] == 11
N <- sum(n)
# 'N' serà el total d'observacions, 22
n1 <- n[1]
n2 <- n[2]
```

Vector de valors d'EEAUC, ordenats de manera que els 11 primers corresponguin a "D" i els 11 següents a "P":

```
auc <- dat[order(dat$tratam), "EEAUC"]
auc
```

```
##  [1] 3756.2 2986.9 3858.2 3328.1 3005.8 5624.5 3354.9 3282.8 5018.0 2472.0
## [11] 3819.6 2623.0 2227.7 4493.7 2985.3 1919.7 2496.1 3016.0 2985.2 3385.5
## [21] 2622.0 3863.5
```

```
auc[1:n1]          # Valors EEAUC per tots els casos "D"
```

```
##  [1] 3756.2 2986.9 3858.2 3328.1 3005.8 5624.5 3354.9 3282.8 5018.0 2472.0
```

```
## [11] 3819.6
```

```
auc[(n1+1):N]     # i per tots els casos "P"
```

```
##  [1] 2623.0 2227.7 4493.7 2985.3 1919.7 2496.1 3016.0 2985.2 3385.5 2622.0
## [11] 3863.5
```

Bootstrap en acció: Bootstrap NO PARAMÈTRIC

Mostra original:

```
auc
```

```
##  [1] 3756.2 2986.9 3858.2 3328.1 3005.8 5624.5 3354.9 3282.8 5018.0 2472.0
## [11] 3819.6 2623.0 2227.7 4493.7 2985.3 1919.7 2496.1 3016.0 2985.2 3385.5
## [21] 2622.0 3863.5
```

Obtenció d'una mostra a partir de FDn1: Remostra dins D:

Cal obtenir una nova mostra extraient amb probabilitat 1/n1 els valors de la mostra original de D. Això és equivalent a **agafar a l'atzar** i **amb reemplaçament** n1 valors de la mostra original (es parla d'una "remostra" de la mostra original):

Inicialitzem la seqüència aleatòria de R en un punt donat, per a que sigui repetible la simulació que farem:

```
set.seed(123)
# Les n1 = 11 primeres dades de 'auc' provenen d'una determinada situació experimental (administrar "D"
# Remostra dins D:
aucD.boot = sample(auc[1:n1], replace = TRUE)
aucD.boot
```

```
##  [1] 3328.1 5018.0 3005.8 2472.0 3819.6 3756.2 5624.5 2472.0 3354.9 5624.5
## [11] 3819.6
```

Si tot just abans d'aquest **sample** hem fet **set.seed(123)** veiem que la remostra s'ha format agafant a l'atzar, en primer lloc, l'element quart de la mostra original, a continuació el novè, etc. Veiem també que estem agafant elements **a l'atzar amb reemplaçament**, per exemple a les posicions quarta i vuitena de la remostra observem que s'ha repetit l'element desè de la mostra original, mentre que alguna dada de la mostra original, com la segona, no apareix a la remostra

Obtenció d'una mostra a partir de FPn2: Remostra dins P:

Les n2 = 11 darreres dades de 'auc' foren obtingudes administrant "P", també les simulem per separat:

```
# Remostra dins P:
aucP.boot = sample(auc[(n1+1):N], replace = TRUE)
aucP.boot
```

```
##  [1] 1919.7 2985.2 3016.0 2227.7 2622.0 4493.7 2623.0 2985.3 3863.5 2622.0
## [11] 2985.2
```

Remostra global:

```
c(aucD.boot, aucP.boot)
```

```
##  [1] 3328.1 5018.0 3005.8 2472.0 3819.6 3756.2 5624.5 2472.0 3354.9 5624.5
## [11] 3819.6 1919.7 2985.2 3016.0 2227.7 2622.0 4493.7 2623.0 2985.3 3863.5
## [21] 2622.0 2985.2
```

Es poden generar directament remostres bootstrap de la mostra com abans, o bé generar els índexos dels elements que participaran a les remostres:

```
set.seed(123)
indx1 = sample(n1, replace = TRUE)
indx1
```

```
##  [1]  4  9  5 10 11  1  6 10  7  6 11
```

```
indx2 = sample((n1+1):N, replace = TRUE)
indx2
```

```
##  [1] 16 19 18 13 21 14 12 15 22 21 19
```

```
# remostra completa:
c(auc[indx1], auc[indx2])
```

```
##  [1] 3328.1 5018.0 3005.8 2472.0 3819.6 3756.2 5624.5 2472.0 3354.9 5624.5
## [11] 3819.6 1919.7 2985.2 3016.0 2227.7 2622.0 4493.7 2623.0 2985.3 3863.5
## [21] 2622.0 2985.2
```

```
# o bé:
auc[c(indx1, indx2)]
```

```
##  [1] 3328.1 5018.0 3005.8 2472.0 3819.6 3756.2 5624.5 2472.0 3354.9 5624.5
## [11] 3819.6 1919.7 2985.2 3016.0 2227.7 2622.0 4493.7 2623.0 2985.3 3863.5
## [21] 2622.0 2985.2
```

Càlcul de l'estadístic t sobre cada remostra:

Primer necessitem una versió més general de la funció 'tStat' (serveix tant pel test de permutacions com pel bootstrap):

```
tStat <- function(indexs1, indexs2 = -indexs1, vector.dades,
  mu = 0, var.equal = TRUE)
{
  t.test(vector.dades[indexs1], vector.dades[indexs2],
         mu = mu, var.equal = var.equal)$statistic
}
```

Delta Estimation i Prova

```
### prova:
deltaEstim <- mean(auc[1:n1]) - mean (auc[(n1+1):N])
tStat(indx1, indx2, auc, mu = deltaEstim)
```

```
##         t
## 0.4657379
```

```
B = 10000
indexsD = 1:n1
indexsP = (n1+1):N

# Generació de B remostres i càlcul de t* sobre cada una:
set.seed(123)

tBoots = replicate(B,
  tStat(
    sample(indexsD, replace = TRUE), sample(indexsP, replace = TRUE),
    auc,
    mu = deltaEstim
  )
)
```

Els primers 20 dels B valors bootstrap de t:

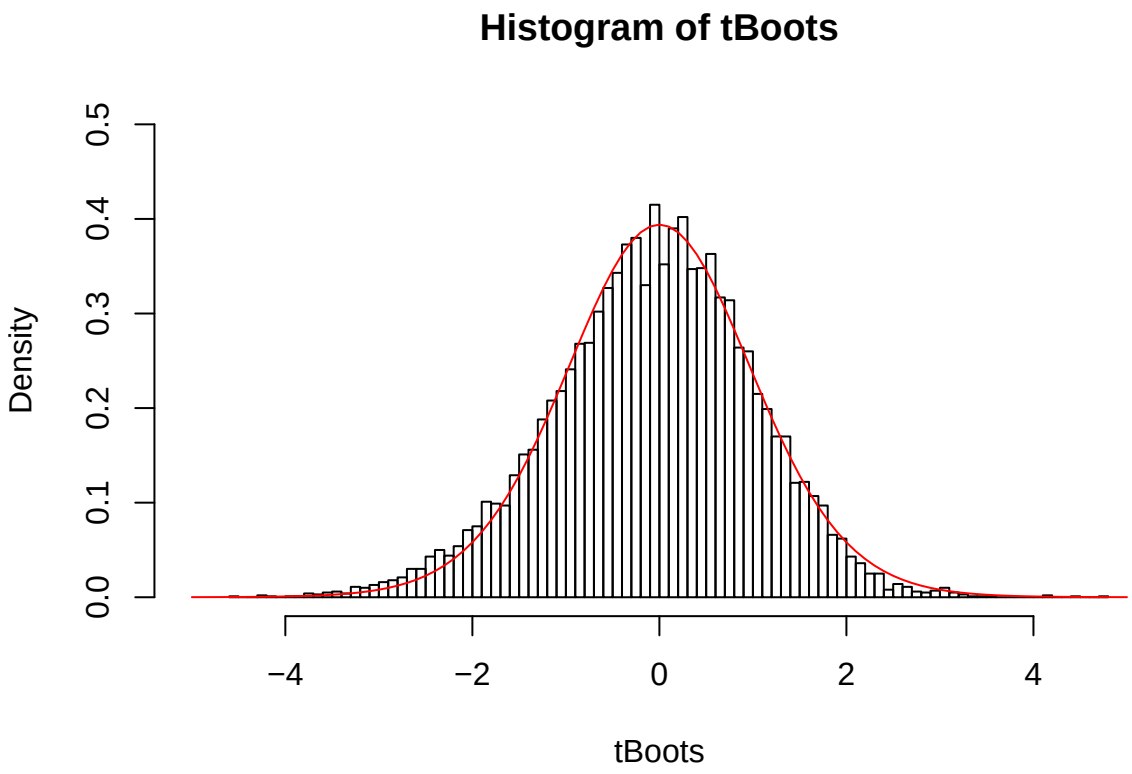
```
 tBoots[1:20]
```

```
##           t           t           t           t           t           t
##  0.46573795 -0.29197251  0.46926255  1.52903027  0.07059381 -0.59280322
##           t           t           t           t           t           t
## -0.92176755 -1.44465522 -0.30991930 -0.95446940  0.39292671 -1.01161992
##           t           t           t           t           t           t
##  0.72337175  0.71736420  0.68724851 -0.24694907 -0.43088149  0.37473930
##           t           t
## -2.79493533 -0.84117049
```

Histograma dels B valors bootstrap,

```
# Obrim una nova finestra gràfica:
windows(14, 14)
hist(tBoots, breaks=70, freq=F, xlim=c(-5,5), ylim=c(0,0.50))

# El comparem amb la t(n1 + n2 - 2):
rang.t <- seq(from=-5, to=+5, by=0.1)
dens.t <- dt(rang.t, df = N - 2)
lines(rang.t, dens.t, type="l", col="red", ylim=c(0,0.50))
```



Conclusions 1 Bootstrap NO PARAMÈTRIC

1. La forma de **l'aproximació bootstrap a la distribució de l'estadístic t** és molt similar a la **t de Student amb N - 2 = 20 g.d.ll**.
2. Probablement la distribució de la qual procedeix la mostra 'auc' és en gran mesura assimilable a una normal.
3. Per altres formes de distribució de les dades, **previsiblement** la distribució dels valors bootstrap seria una altra (i segurament més correcta que considerar que la distribució mostral és la t de Student)

Conclusions 2: Boostrap NO PARAMÈTRIC (valors crítics)

1. Taula de **valors crítics t** obtinguda a partir de la mostra i el mètode bootstrap:

```
taulaBoot = quantile(tBoots,
  probs = c(0.01, 0.025, 0.05, 0.1, 0.15, 0.20, 0.80, 0.85, 0.90, 0.95, 0.975, 0.99)
)
taulaBoot
```

```
##         1%       2.5%         5%        10%        15%        20%
## -2.7786863 -2.3309121 -1.9275217 -1.4584650 -1.1659644 -0.9386974
##        80%        85%        90%        95%      97.5%        99%
##  0.8220141  1.0164433  1.2672508  1.6384791  1.9364479  2.3002778
```

2. Els comparem amb els **valors crítics "teòrics"** si fos cert que la distribució mostral és la t:

```r
taula.t = qt(c(0.01, 0.025, 0.05, 0.1, 0.15, 0.20, 0.80, 0.85, 0.90, 0.95, 0.975, 0.99), df = N - 2)
names(taula.t) = names(taulaBoot)
taula.t
```

```
##         1%       2.5%         5%        10%        15%        20%
## -2.5279770 -2.0859634 -1.7247182 -1.3253407 -1.0640158 -0.8599644
##        80%        85%        90%        95%      97.5%        99%
##  0.8599644  1.0640158  1.3253407  1.7247182  2.0859634  2.5279770
```

Exercici

Les següents dades corresponen a una mostra aleatoria d'una variable aleatòria **Y que correspon als temps d'espera, en minuts**, dels clients d'un servei ???ns a ser atesos. Hi pot haver dubtes sobre quina **es la veritable distribució d'Y**, però sembla clar que no és normal.

Els valors de la mostra són: 2.04, 6.14, 10.72, 2.22, 11.90, 0.71, 3.28, 0.55, 1.21, 0.97, 0.46, 1.81, 2.04, 7.09 i 4.89.

Mitjançant **bootstrap no paramètric**,

- **Estima la distribució de l'estadístic**

$$t = \sqrt{n}(Y - \mu)/S$$

  on μ correspon a la mitjana poblacional del temps d'espera.

- Compara gràficament aquesta distribució amb la que tindríem per aquest estadístic si poguéssim suposar normalitat per la variable Y .

- Determina els quantils 0.025 i 0.975 de la distribució del estadístic a partir de la distribució bootstrap obtinguda abans i compara'ls amb els que obtindríem a partir de suposar normalitat d'Y (segurament incorrectament).