sta211

Anthony Kalaydjian - Mathieu Occhipinti

2023-05-03

Estimation d'une taille de population à partir de données de capture-marquage-recapture

Vraissemblance du modèle

La vraissemblance du modèle $\mathcal{M}$ s'écrit comme suit :

$$\begin{aligned}
[C_1 = c_1, C_{20} = c_{20}, C_{21} = c_{21}|\pi, N] &= [C_1 = c_1|\pi, N]\,[C_{20} = c_{20}, |\pi, N, C_1 = c_1]\,[C_{21} = c_{21}|\pi, N, C_1 = c_1, C_{20} = c_{20},] \\
&= [C_1 = c_1|\pi, N]\,[C_{20} = c_{20}, |\pi, N, C_1 = c_1]\,[C_{21} = c_{21}|\pi, N, C_1 = c_1] \\
&= C_{c_1}^{N}\pi^{c_1}(1-\pi)^{N-c_1}C_{c_20}^{N-c_1}\pi^{c_{20}}(1-\pi)^{N-c_1-c_{20}} \\
&= C_{c_1}^{N}\pi^{c_1}(1-\pi)^{N-c_1}C_{c_{20}}^{N-c_1}\pi^{c_{20}}(1-\pi)^{N-c_1-c_{20}}C_{c_{21}}^{c_1}\pi^{c_{21}}(1-\pi)^{C_1-c_{21}}
\end{aligned}$$

On en déduit donc la log-vraissemblance en passant au log :

$$\mathrm{l}(N,\pi) = \ln\left(C_{c_1}^{N}C_{c_{20}}^{N-c_1}C_{c_{21}}^{c_1}\right) + (c_1 + c_{20} + c_{21})\ln(\pi) + (2N - 2c_1 - c_{20} + c_1 - c_{21})\ln(1-\pi)$$

Simulation du tirage de C_1

La fonction de répartition de la loi discrète de $C_1 \sim \mathcal{B}(N,\pi)$ est la suivante :

$\forall x \in [0,1], \qquad F(x) = \sum_{k=0}^{N}\mathbb{P}(C_1 = k)1_{\{k\leq x\}}$

On remarque que $\forall u \in [0,1], \exists p \in [0,N] \quad / \quad \sum_{k=0}^{p-1}\mathbb{P}(C_1 = k) \leq u \leq \sum_{k=0}^{p}\mathbb{P}(C_1 = k)$ [1]

Ainsi, $\forall x \in [k, k+1], \quad F(x) = \sum_{k=0}^{p}\mathbb{P}(C_1 = k) \geq u$

L'inverse généralisée de la loi discrète s'écrit donc : $F^{-1}(u) = p$

Finalement, on a :

$$\boxed{\forall u \in [0,1], \quad F^{-1}(u) = \inf_{p=1,\ldots,N}\left\{p \;\middle|\; \sum_{k=0}^{p}\mathbb{P}(C_1 = k) \geq u\right\}}$$

```r
my.qbinom <- function(u, N, pi){
  p <- sapply(c(0:N), FUN=function(n) choose(N, n)*pi^n*(1-pi)^{N-n})
  cdf <- cumsum(p)
  return(findInterval(u, cdf))
}
```

[1] Avec la convention $\sum_{k=0}^{-1}\mathbb{P}(C_1 = k) = 0$

```r
my.rbinom <- function(N, pi, n.iter=1){
  U <- runif(n=n.iter, min=0, max=1)
  res <- sapply(U, FUN=function(u) my.qbinom(u,N, pi))
  return(res)
}
```

```r
n.iter <- 10000
N <- 125
pi <- 0.15

generated.C1 <- my.rbinom(N, pi, n.iter)
```

```r
resultats <- data.frame(n=1:n.iter, valeurs=factor(generated.C1, levels = 0:N))
```

```r
#frequence theorique
freq_theo =dbinom(0:N, N, pi)

#calcul de la frequence empirique
freq_emp <- c()
for (k in 0:N){
 freq_emp <- c(freq_emp, mean(generated.C1==k))
}
freq_binom <- tibble( x=0:N, freq_emp=freq_emp, freq_theo=freq_theo)

#Représentation graphique
ggplot(freq_binom) + #Tableau représenter
aes(x = x) + #Abscisse commune
geom_col(mapping = aes(y = freq_emp), #Ordonne des frquences empiriques
width = 0.2, fill = "lightblue") +
geom_point(aes(y = freq_theo), #On ajoute le point des frquences thoriques
shape = 3, col = "red", size = 3) +
xlim(0, 40) +
labs(y = "Frequence", x = "Nombre de succes")
```

Simulation d'une réalisation possible de capture-marquage-recapture

```r
capture.sim <- function(N, pi){
  C1 <- my.rbinom(N=N, pi=pi)
  C20 <- my.rbinom(N=N-C1, pi=pi)
  C21 <- my.rbinom(N=C1, pi=pi)
  return(tibble(C1=C1, C20=C20, C21=C21))
}

capture.sim(N, pi)
```

```
## # A tibble: 1 x 3
##      C1   C20   C21
##   <int> <int> <int>
## 1    21    15     1
```

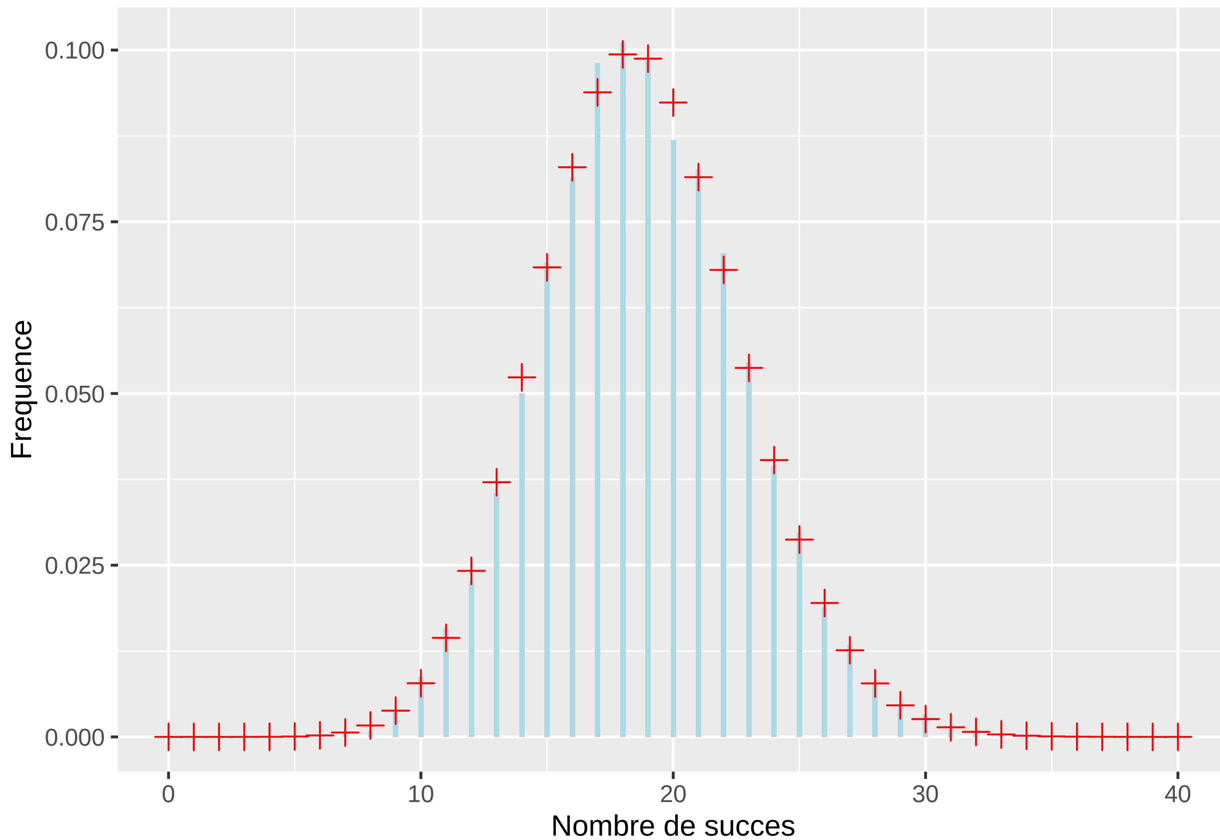

Figure 1: Comparaison des fréquences

Supposons N connu

Supposons tout d'abord que $N = 950$ (connu) et estimons l'efficacité π.

Estimateur de maximum de vraissemblance $\hat{\pi}_{MLE}$ de π