
Rapports de vraisemblance

J.R. Lobry, D. Chessel & A.B. Dufour

Rapports de vraisemblance et Khi2. Tests du rapport de vraisemblance. Une expérience entomologique.

Table des matières

1	Introduction	1
2	La loi du rapport à la vraisemblance maximale	2
3	Un paramètre et k groupes	9
4	Le G^2 d'une table de contingence	17
5	Un problème entomologique	19
	Références	34

1 Introduction

On extrait des données (Table 1) de l'exemple du logiciel `whichrun` [2], ce sont les fréquences alléliques du locus microsatellite OTS-2 dans 5 populations du Saumon quinnat à montaison hivernale, *Oncorhynchus tshawytscha*. Sachant que les hypothèses du modèle de Hardy-Weinberg sont satisfaites, estimer au maximum de vraisemblance à laquelle des 5 populations appartient un individu hétérozygote typé 066-070.

La probabilité que l'individu porte un couple d'allèles ab est $2P(a)P(b)$, donc les vraisemblances (probabilité du résultat sous l'hypothèse que l'individu appartienne à une population donnée) est :

```
a <- c(0.038, 0.002, 0.053, 0.066, 0.800)
b <- c(0.263, 0.297, 0.447, 0.368, 0.119)
round(2*a*b, dig=3)
```

```
[1] 0.020 0.001 0.047 0.049 0.190
```

Au maximum de vraisemblance, on estime qu'il appartient à la POP5.

Exercice : donner les vraisemblances pour qu'un individu homozygote typé 066 appartienne aux 5 populations :

allèle	POP1	POP2	POP3	POP4	POP5
064	0.002	0.000	0.000	0.000	0.000
066	0.038	0.002	0.053	0.066	0.800
070	0.263	0.297	0.447	0.368	0.119
072	0.010	0.004	0.000	0.004	0.000
074	0.030	0.004	0.004	0.041	0.006
078	0.004	0.002	0.000	0.000	0.000
080	0.062	0.029	0.080	0.037	0.003
082	0.016	0.025	0.000	0.000	0.000
084	0.080	0.083	0.053	0.099	0.010
086	0.315	0.359	0.164	0.252	0.045
088	0.006	0.008	0.000	0.000	0.000
090	0.002	0.000	0.000	0.000	0.000
094	0.006	0.000	0.000	0.004	0.000
096	0.082	0.060	0.004	0.045	0.010
098	0.000	0.000	0.000	0.008	0.000
100	0.020	0.025	0.013	0.037	0.000
102	0.056	0.098	0.173	0.029	0.003
104	0.002	0.000	0.009	0.008	0.000
106	0.008	0.004	0.000	0.000	0.003

TABLE 1 – Fréquences alléliques dans 5 populations.

```
[1] 0.001 0.000 0.003 0.004 0.640
```

Donner les vraisemblances pour qu'un individu homozygote typé 070 appartienne aux 5 populations :

```
[1] 0.069 0.088 0.200 0.135 0.014
```

2 La loi du rapport à la vraisemblance maximale

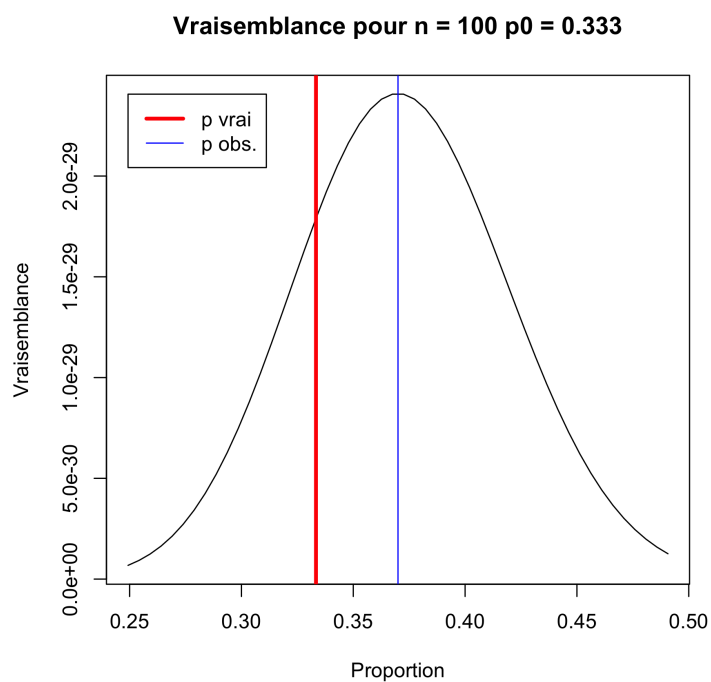
Soit une population dont une proportion p des individus ont une caractéristique donnée. La fonction de vraisemblance donne la probabilité de l'observation pour une valeur hypothétique de p . Considérons l'hypothèse nulle $p = p_0$ et une alternative p arbitraire et *supposons que l'hypothèse nulle est vraie*. Un échantillon donne une fonction de vraisemblance qui est maximum pour la fréquence observée :

```
f <- function(necha = 100, p0 = 1/3) {
  echa <- rbinom(n = necha, size = 1, p = p0)
  echaoui <- sum(echa)
  echanon <- necha - echaoui
  pest <- sum(echa)/necha
  vraisemblance <- function(p , noui, nnon) {
    (p^noui)*((1-p)^nnon)
  }
  delta <- 2.5*sqrt(pest*(1-pest)/necha)
  xmin <- pest - delta
  xmax <- pest + delta
  x <- seq(from = xmin, to = xmax,length = 50)
  y <- vraisemblance(x,echaoui,echanon)
  plot(x, y, type = "l",
```

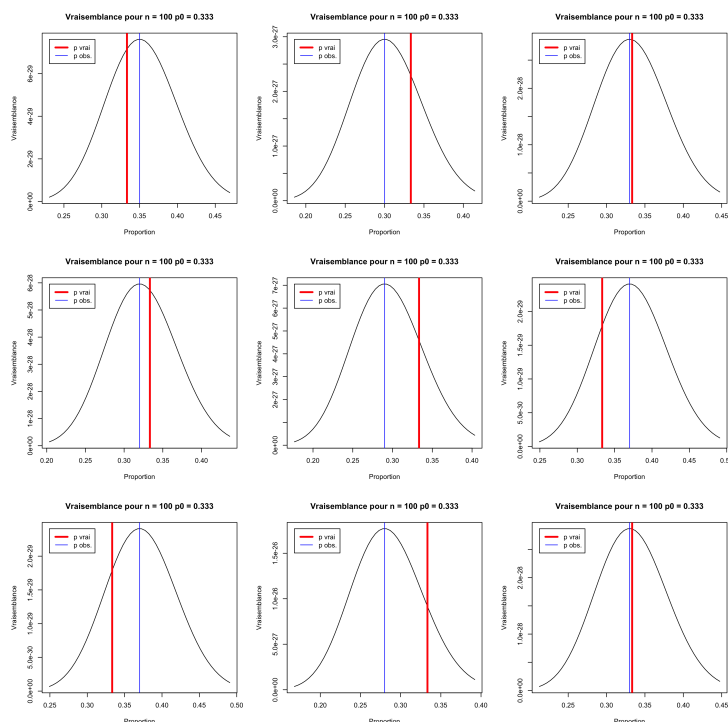
```

main = paste("Vraisemblance pour n =", necha, "p0 =", round(p0,3)),
xlab = "Proportion", ylab = "Vraisemblance")
abline(v = p0, lwd=3, col = "red")
abline(v = pest, col = "blue")
legend(xmin, max(y), c("p vrai","p obs."), lwd = c(3,1), col = c("red","blue"))
}
f()

```



Refaire l'expérience plusieurs fois :



On constate que la vraisemblance de la vraie valeur est toujours plus petite que la vraisemblance de la valeur estimée. Notons \mathbf{x} l'échantillon, θ_0 la vraie valeur du paramètre, $L(\mathbf{x}, \theta)$ la valeur de la vraisemblance d'une valeur du paramètre pour cet échantillon et R le rapport :

$$R = \frac{L(\mathbf{x}, \theta_0)}{\text{Sup}_{\theta} L(\mathbf{x}, \theta)}$$

Il est inférieur à 1. Pour une valeur donnée de la vraie valeur du paramètre, θ_0 , le rapport R change d'un échantillon à l'autre. Il a une distribution d'échantillonnage. La quantité $-2 \log(R)$ est toujours positive. On va apprécier sa distribution d'échantillonnage par simulation.

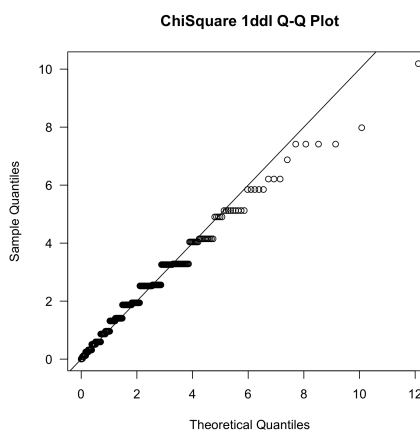
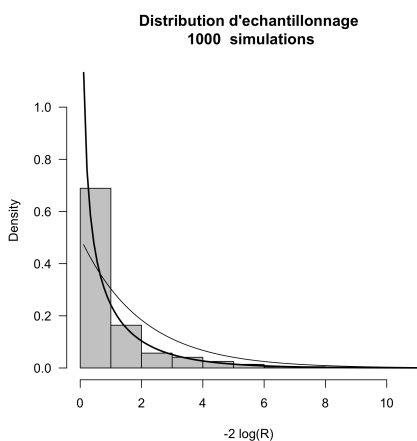
```
g <- fonction(nsimul = 1000, necha = 100, p0 = 1/3) {
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  simul <- fonction(x) {
    echa <- rbinom(necha,1,p0)
    noui <- sum(echa)
    nnon <- necha - noui
    pest <- sum(echa)/necha
    # la vraisemblance de la vraie valeur
    vraivrai <- (p0^noui)*((1-p0)^nnon)
    # la vraisemblance maximum
    vraimax <- (pest^noui)*((1-pest)^nnon)
    -2*log(vraivrai/vraimax)
  }
  z <- sapply(1:nsimul,simul)
  z.hist <- hist(z, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = 1)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, ylim = c(0,maxi), col = grey(0.8),
  xlab = "-2 log(R)", las = 1,
```

```

main = paste("Distribution d'echantillonnage\n", nsimul, " simulations")
lines(xseq, z.chi, lwd = 2)
lines(xseq, dchisq(xseq, df = 2), lwd = 1)

qqplot( qchisq(ppoints(n=nsimul), df = 1), sort(z),
        main = "ChiSquare 1ddl Q-Q Plot",
        xlab = "Theoretical Quantiles",
        ylab = "Sample Quantiles", las = 1)
abline(0,1)
}
g()

```



L'ajustement est bon. *Surtout il ne dépend pas de la loi utilisée.* Essayer avec une loi de Poisson (justifier les transformations) :

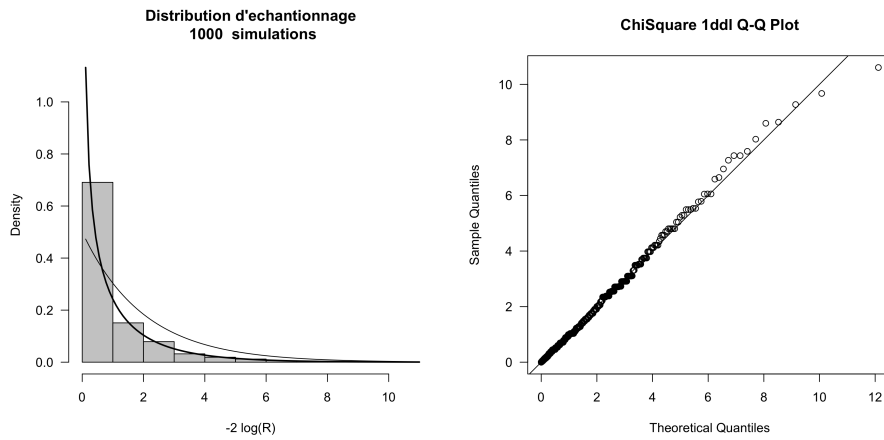
```

g.pois <- function(nsimul = 1000, necha = 100, lambda = 3.2) {
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  simul <- function(x) {
    echa <- rpois(necha, lambda)
    lest <- mean(echa)
    logvraivrai <- sum(dpois(echa, lambda, log = TRUE))
    logvraimax <- sum(dpois(echa, lest, log = TRUE))
    2*(logvraimax-logvraivrai)
  }
  z <- sapply(1:nsimul,simul)

  z.hist <- hist(z, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = 1)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, ylim = c(0,maxi), col = grey(0.8),
       xlab = "-2 log(R)", las = 1,
       main = paste("Distribution d'echantionnage\n", nsimul, " simulations"))
  lines(xseq, z.chi, lwd = 2)
  lines(xseq, dchisq(xseq, df = 2), lwd = 1)

  qqplot( qchisq(ppoints(n=nsimul), df = 1), sort(z),
          main = "ChiSquare 1ddl Q-Q Plot",
          xlab = "Theoretical Quantiles",
          ylab = "Sample Quantiles", las = 1)
  abline(0,1)
}
g.pois()

```

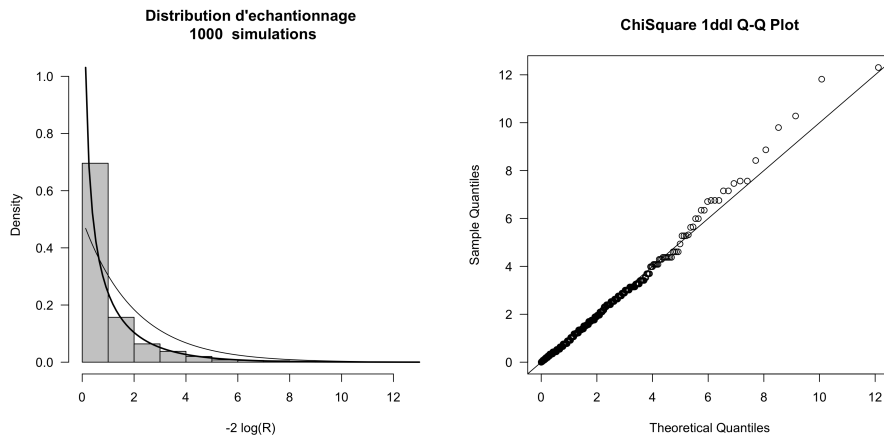


Essayer avec une loi binomiale :

```
g.binom <- function(nsimul = 1000, necha = 100, size = 10, prob = 0.75) {
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  simul <- function(x) {
    echa <- rbinom(necha, size, prob)
    pest <- mean(echa)/size
    logvraivrai <- sum(dbinom(echa, size, prob, log = TRUE))
    logvraimax <- sum(dbinom(echa, size, pest, log = TRUE))
    2*(logvraimax-logvraivrai)
  }
  z <- sapply(1:nsimul,simul)

  z.hist <- hist(z, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = 1)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, ylim = c(0,maxi), col = grey(0.8),
  xlab = "-2 log(R)", las = 1,
  main = paste("Distribution d'échantonnage\n", nsimul, " simulations"))
  lines(xseq, z.chi, lwd = 2)
  lines(xseq, dchisq(xseq, df = 2), lwd = 1)

  qqplot( qchisq(ppoints(n=nsimul), df = 1), sort(z),
  main = "ChiSquare 1ddl Q-Q Plot",
  xlab = "Theoretical Quantiles",
  ylab = "Sample Quantiles", las = 1)
  abline(0,1)
}
g.binom()
```



Essayer avec une loi exponentielle :

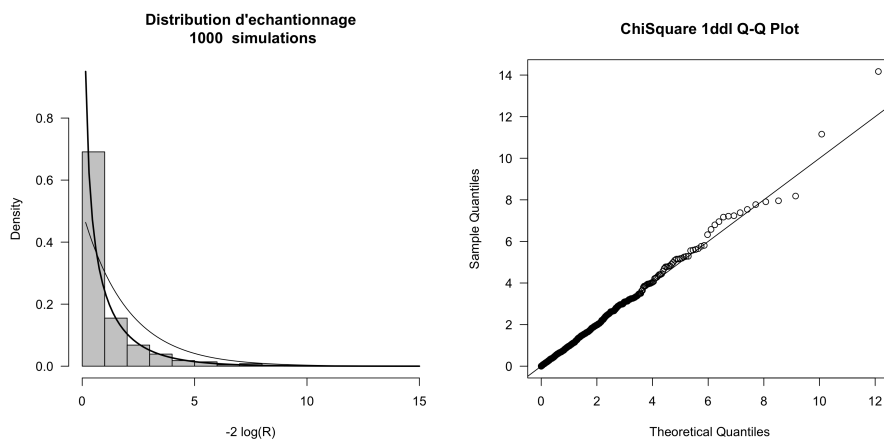
```

g.exp <- function(nsimul = 1000, necha = 100, alpha = 3.68) {
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  simul <- function(x) {
    echa <- rexp(necha, alpha)
    alphaest <- 1/mean(echa)
    logvraivrai <- sum(dexp(echa, alpha, log = TRUE))
    logvraimax <- sum(dexp(echa, alphaest, log = TRUE))
    2*(logvraimax-logvraivrai)
  }
  z <- sapply(1:nsimul,simul)

  z.hist <- hist(z, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = 1)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, ylim = c(0,maxi), col = grey(0.8),
  xlab = "-2 log(R)", las = 1,
  main = paste("Distribution d'échantonnage\n", nsimul, " simulations"))
  lines(xseq, z.chi, lwd = 2)
  lines(xseq, dchisq(xseq, df = 2), lwd = 1)

  qqplot( qchisq(ppoints(n=nsimul), df = 1), sort(z),
  main = "ChiSquare 1ddl Q-Q Plot",
  xlab = "Theoretical Quantiles",
  ylab = "Sample Quantiles", las = 1)
  abline(0,1)
}
g.exp()

```



Dans tous les cas, $-2 \log(R)$ suit *asymptotiquement* un χ^2 à un degré de liberté (démonstration dans [9]).

Ce qui est essentiel est que dans le rapport il y ait :

- ★ au numérateur la vraisemblance de la vraie valeur
- ★ au dénominateur la vraisemblance de l'estimation au maximum de vraisemblance

Évidemment, en statistique, on ne connaît jamais la vraie valeur, mais si on fait l'hypothèse que cette vraie valeur est θ_0 (donnée *a priori*) alors le rapport à la vraisemblance maximale permet de faire un test de compatibilité avec l'échantillon.

Exemple ([9] p. 330) : Sur un échantillon de 200 individus d'une commune, 45 % sont favorable à l'implantation d'un centre commercial. Ceci contredit-il l'hypothèse qu'un habitant sur deux y est favorable ?

Habituellement :

```
prop.test(90,200)
  1-sample proportions test with continuity correction
data: 90 out of 200, null probability 0.5
X-squared = 1.805, df = 1, p-value = 0.1791
alternative hypothesis: true p is not equal to 0.5
95 percent confidence interval:
 0.3802225 0.5217297
sample estimates:
 p
0.45

binom.test(90,200)
  Exact binomial test
data: 90 and 200
number of successes = 90, number of trials = 200, p-value = 0.179
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.3797536 0.5217507
sample estimates:
probability of success
0.45
```


Le test du rapport à la vraisemblance maximale donne simplement :

```
logvmax <- 90*log(0.45)+110*log(0.55)
logvhyp <- 200*log(0.5)
(w <- 2*(logvmax-logvhyp))
[1] 2.003347
1-pchisq(w,1)
[1] 0.1569523
```

Cette approche a surtout des conséquences dans la comparaison des hypothèses à plusieurs paramètres.

3 Un paramètre et k groupes

Supposons que $\mathbf{x} = (x_1, \dots, x_n)$ est un échantillon aléatoire simple d'une variable poissonnienne de paramètre λ . Supposons que $\mathbf{y} = (y_1, \dots, y_p)$ est un échantillon aléatoire simple d'une variable poissonnienne de paramètre μ . Supposons enfin que les deux échantillons sont indépendants. L'hypothèse nulle H_0 est $\lambda = \mu$ et l'alternative H_1 est $\lambda \neq \mu$.

Si l'hypothèse nulle est vraie, l'estimation au maximum de vraisemblance du paramètre donne :

$$\hat{\tau} = \frac{(x_1 + \dots + x_n) + (y_1 + \dots + y_p)}{n + p} \quad (1)$$

La vraisemblance de l'hypothèse nulle est alors :

$$L(H_0) = \prod_{i=1}^n e^{-\hat{\tau}} \frac{\hat{\tau}^{x_i}}{x_i!} \prod_{j=1}^p e^{-\hat{\tau}} \frac{\hat{\tau}^{y_j}}{y_j!} \quad (2)$$

Si l'hypothèse alternative est vraie, la vraisemblance maximum est obtenue par deux estimations séparées :

$$\hat{\lambda} = \frac{x_1 + \dots + x_n}{n} \quad \hat{\mu} = \frac{y_1 + \dots + y_p}{p} \quad (3)$$

et vaut :

$$L(H_1) = \prod_{i=1}^n e^{-\hat{\lambda}} \frac{\hat{\lambda}^{x_i}}{x_i!} \prod_{j=1}^p e^{-\hat{\mu}} \frac{\hat{\mu}^{y_j}}{y_j!} \quad (4)$$

Faire une simulation de cette situation avec l'hypothèse nulle vraie :

```
h <- fonction(necha1 = 12, necha2 = 11, lambda = 1.25)
{
  x <- rpois(necha1, lambda)
  y <- rpois(necha2, lambda)
  v1 <- prod(dpois(c(x,y),mean(c(x,y))))
  v2 <- prod(dpois(x,mean(x)))*prod(dpois(y,mean(y)))
  print(paste("v1 =",v1))
  print(paste("v2 =",v2))
  print(iffelse(v2 > v1, "v2 > v1", "v2 <= v1"))
}
h()
```

```
[1] "v1 = 9.17090605446721e-14"
[1] "v2 = 1.00163897089307e-13"
[1] "v2 > v1"
```

Recommencer autant de fois que vous voulez :

```
[1] "v1 = 1.83160007932752e-15"
[1] "v2 = 1.30003128436965e-14"
[1] "v2 > v1"
[1] "v1 = 1.13760555715002e-14"
[1] "v2 = 1.1637312559176e-14"
[1] "v2 > v1"
```

La seconde est toujours plus grande que la première. En effet, pour le premier échantillon $\mathbf{x} = (x_1, \dots, x_n)$, l'estimation $\hat{\lambda}$ (3) maximise la vraisemblance, et est donc supérieure à toute autre valeur de la vraisemblance pour une autre valeur du paramètre, par exemple la valeur de $\hat{\tau}$ (1) :

$$0 \leq \prod_{i=1}^n e^{-\hat{\tau}} \frac{\hat{\tau}^{x_i}}{x_i!} \leq \prod_{i=1}^n e^{-\hat{\lambda}} \frac{\hat{\lambda}^{x_i}}{x_i!} \quad (5)$$

De même, pour le deuxième échantillon $\mathbf{y} = (y_1, \dots, y_n)$, l'estimation $\hat{\mu}$ maximise la vraisemblance, et on a :

$$0 \leq \prod_{j=1}^p e^{-\hat{\tau}} \frac{\hat{\tau}^{y_j}}{y_j!} \leq \prod_{j=1}^p e^{-\hat{\mu}} \frac{\hat{\mu}^{y_j}}{y_j!} \quad (6)$$

Des inégalités (5) et (6) on déduit que :

$$0 \leq \prod_{i=1}^n e^{-\hat{\tau}} \frac{\hat{\tau}^{x_i}}{x_i!} \prod_{j=1}^p e^{-\hat{\tau}} \frac{\hat{\tau}^{y_j}}{y_j!} \leq \prod_{i=1}^n e^{-\hat{\lambda}} \frac{\hat{\lambda}^{x_i}}{x_i!} \prod_{j=1}^p e^{-\hat{\mu}} \frac{\hat{\mu}^{y_j}}{y_j!} \quad (7)$$

Et donc d'après (2) et (4) que :

$$L(H_0) \leq L(H_1) \quad (8)$$

On aura donc toujours :

$$-2 \log \left(\frac{L(H_0)}{L(H_1)} \right) \geq 0 \quad (9)$$

Estimer la distribution d'échantillonnage de cette quantité quand l'hypothèse nulle est vraie (justifier le contenu) :

```
hsim.2 <- fonction(nsimul = 1000, necha1 = 12, necha2 = 11, lambda = 1.25)
{
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  f1 <- fonction(x) {
    x <- rpois(necha1, lambda)
    y <- rpois(necha2, lambda)
    v1 <- sum(dpois(c(x, y), mean( c(x, y)), log = TRUE))
  }
}
```

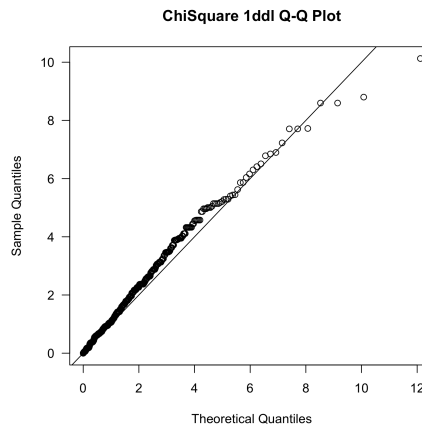
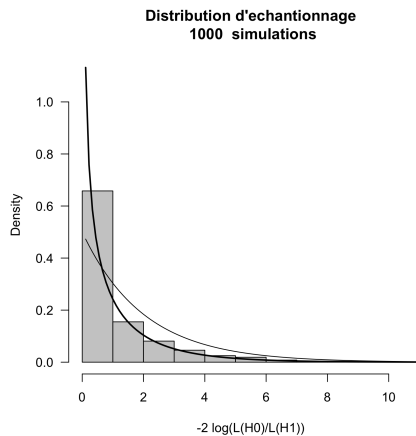
```

v2 <- sum(dpois(x, mean(x), log = TRUE)) + sum(dpois(y, mean(y), log = TRUE))
2*(v2-v1)
}
z <- sapply(1:nsimul, f1)

z.hist <- hist(z, plot = FALSE)
xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
z.chi <- dchisq(xseq, df = 1)
maxi <- max(z.hist$density, z.chi)
hist(z, proba = TRUE, ylim = c(0,maxi), col = grey(0.8),
xlab = "-2 log(L(H0)/L(H1))", las = 1,
main = paste("Distribution d'echantionnage\n", nsimul, " simulations"))
lines(xseq, z.chi, lwd = 2)
lines(xseq, dchisq(xseq, df = 2), lwd = 1)

qqplot( qchisq(ppoints(n=nsimul), df = 1), sort(z),
main = "ChiSquare 1ddl Q-Q Plot",
xlab = "Theoretical Quantiles",
ylab = "Sample Quantiles", las = 1)
abline(0,1)
}
hsim.2()

```



Étendre cette simulation à 4 sous-échantillons :

```

hsim.4 <- function(nsimul = 1000, n1 = 80, n2 = 150, n3 = 30, n4 = 60, lambda = 3.25)
{
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  f1 <- function(x) {
    x1 <- rpois(n1, lambda)
    x2 <- rpois(n2, lambda)
    x3 <- rpois(n3, lambda)
    x4 <- rpois(n4, lambda)
    w <- c(x1, x2, x3, x4)
    v1 <- sum(dpois(w, mean(w), log = TRUE))
    v2 <- sum(dpois(x1, mean(x1), log = TRUE)) + sum(dpois(x2, mean(x2), log = TRUE)) +
      sum(dpois(x3, mean(x3), log = TRUE)) + sum(dpois(x4, mean(x4), log = TRUE))
    2*(v2-v1)
  }
  z <- sapply(1:nsimul, f1)

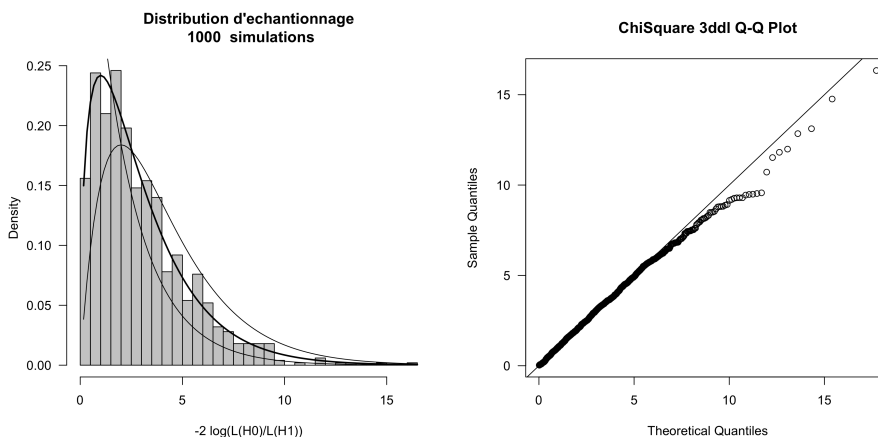
  z.hist <- hist(z, nclass = 50, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = 3)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, nclass = 50, ylim = c(0,maxi), col = grey(0.8),
xlab = "-2 log(L(H0)/L(H1))", las = 1,
main = paste("Distribution d'echantionnage\n", nsimul, " simulations"))
lines(xseq, z.chi, lwd = 2)
lines(xseq, dchisq(xseq, df = 2), lwd = 1)
}

```

```

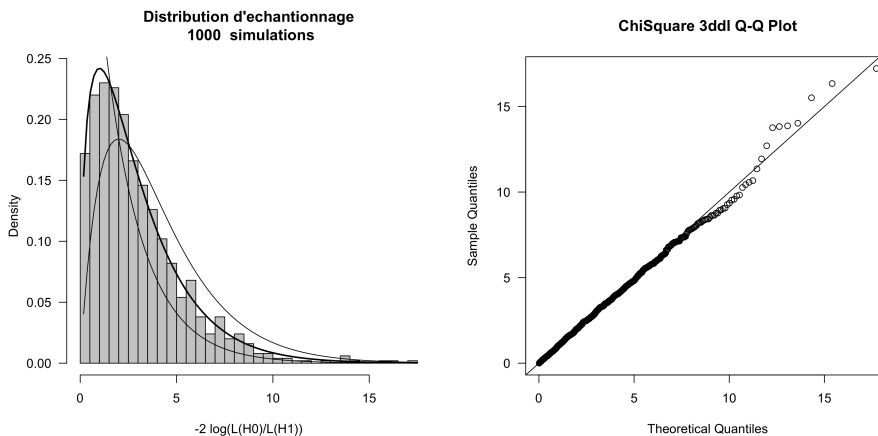
lines(xseq, dchisq(xseq, df = 4), lwd = 1)
qqplot( qchisq(ppoints(n=nsimul), df = 3), sort(z),
        main = "ChiSquare 3ddl Q-Q Plot",
        xlab = "Theoretical Quantiles",
        ylab = "Sample Quantiles", las = 1)
abline(0,1)
}
hsim.4()

```



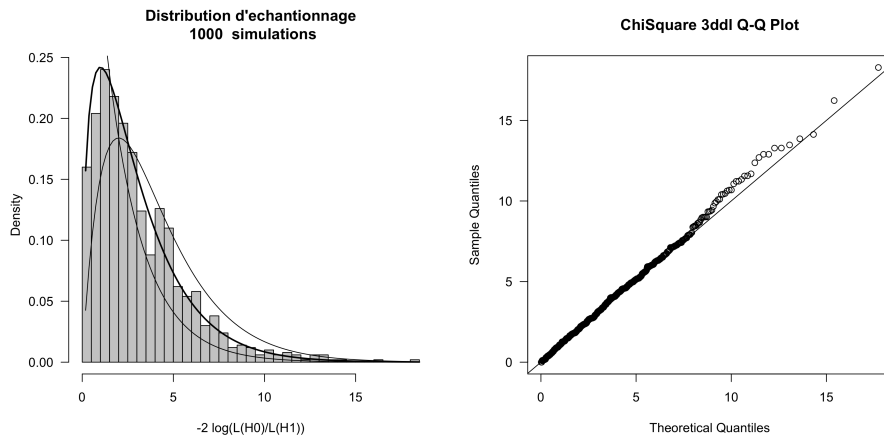
Essayer des échantillons de petites tailles :

```
hsim.4(nsimul = 1000, n1 = 8, n2 = 12, n3 = 5, n4 = 4, lambda = 3.25)
```



Essayer des échantillons de *très* petites tailles :

```
hsim.4(nsimul = 1000, n1 = 2, n2 = 2, n3 = 2, n4 = 2, lambda = 3.25)
```



Généraliser au cas de k échantillons :

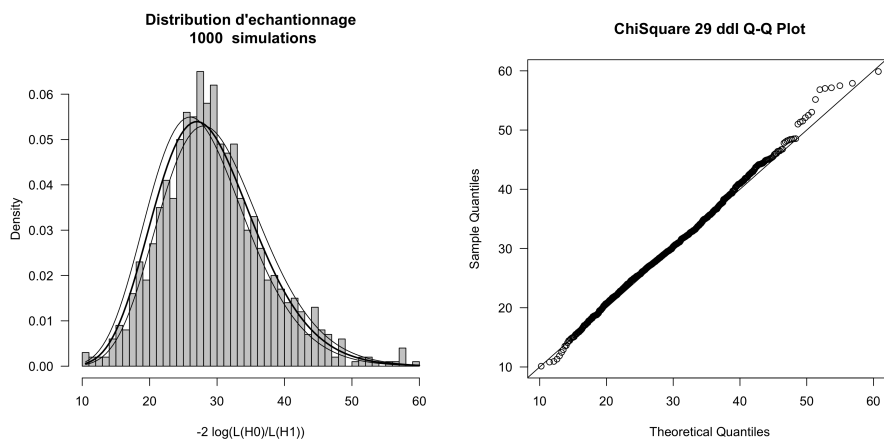
```

hsim.k <- function(nsimul = 1000, k = 30, n = rep(30, k), lambda = 3.25)
{
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  f1 <- function(x) {
    x <- lapply(n, rpois, lambda = lambda)
    w <- unlist(x)
    v1 <- sum(dpois(w, mean(w), log = TRUE))
    v2 <- sum(unlist(lapply(x, function(x) sum(dpois(x, mean(x), log = TRUE))))))
    2*(v2-v1)
  }
  z <- sapply(1:nsimul, f1)

  z.hist <- hist(z, nclass = 50, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = k - 1)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, nclass = 50, ylim = c(0,maxi), col = grey(0.8),
  xlab = "-2 log(L(H0)/L(H1))", las = 1,
  main = paste("Distribution d'échantonnage\n", nsimul, " simulations"))
  lines(xseq, z.chi, lwd = 2)
  lines(xseq, dchisq(xseq, df = k - 1 -1), lwd = 1)
  lines(xseq, dchisq(xseq, df = k - 1 +1), lwd = 1)

  qqplot( qchisq(ppoints(n=nsimul), df = k - 1), sort(z),
  main = paste("ChiSquare", k - 1, "ddl Q-Q Plot"),
  xlab = "Theoretical Quantiles",
  ylab = "Sample Quantiles", las = 1)
  abline(0,1)
}
hsim.k()

```



Essayer k échantillons d'une même variable de Bernoulli :

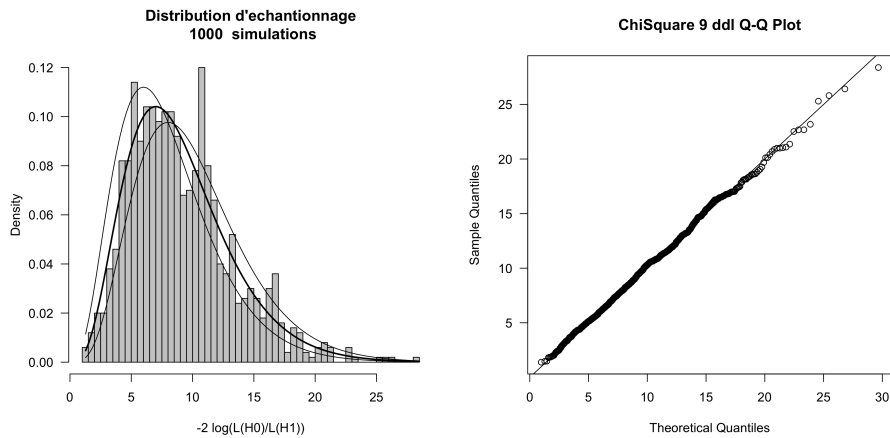
```

hber.k <- function(nsimul = 1000, k = 30, n = rep(30, k), prob = 0.25)
{
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  f1 <- function(x) {
    x <- lapply(n, rbinom, size = 1, prob = prob)
    w <- unlist(x)
    v1 <- sum(dbinom(w, 1, mean(w), log = TRUE))
    v2 <- sum(unlist(lapply(x, function(x) sum(dbinom(x, 1, mean(x), log = TRUE))))))
    2*(v2-v1)
  }
  z <- sapply(1:nsimul, f1)

  z.hist <- hist(z, nclass = 50, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = k - 1)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, nclass = 50, ylim = c(0,maxi), col = grey(0.8),
  xlab = "-2 log(L(H0)/L(H1))", las = 1,
  main = paste("Distribution d'échantonnage\n", nsimul, " simulations"))
  lines(xseq, z.chi, lwd = 2)
  lines(xseq, dchisq(xseq, df = k - 1 -1), lwd = 1)
  lines(xseq, dchisq(xseq, df = k - 1 +1), lwd = 1)

  qqplot( qchisq(ppoints(n=nsimul), df = k - 1), sort(z),
  main = paste("ChiSquare", k - 1, "ddl Q-Q Plot"),
  xlab = "Theoretical Quantiles",
  ylab = "Sample Quantiles", las = 1)
  abline(0,1)
}
hber.k(k=10)

```



Essayer avec k échantillons d'une loi exponentielle :

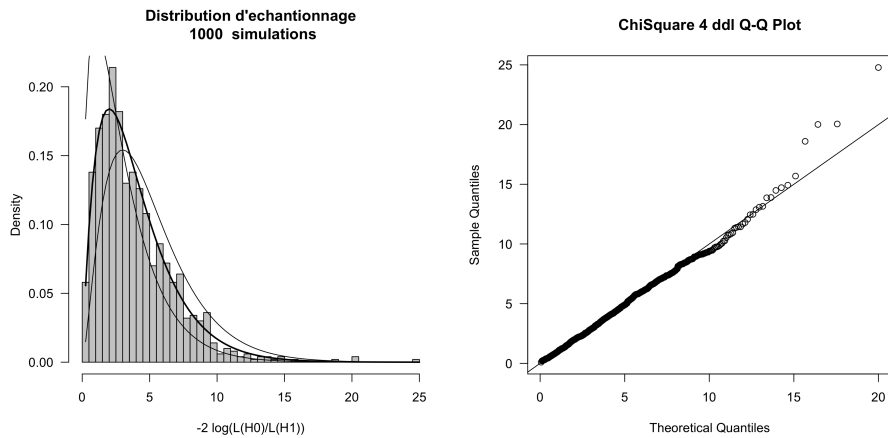
```

hexp.k <- function(nsimul = 1000, k = 30, n = rep(30, k), rate = 3.66)
{
  par(no.readonly = TRUE) -> opar
  on.exit(par(opar))
  par(mfrow = c(1,2))
  f1 <- function(x) {
    x <- lapply(n, rexp, rate = rate)
    w <- unlist(x)
    v1 <- sum(dexp(w, 1/mean(w), log = TRUE))
    v2 <- sum(unlist(lapply(x, function(x) sum(dexp(x, 1/mean(x), log = TRUE))))))
    2*(v2-v1)
  }
  z <- sapply(1:nsimul, f1)

  z.hist <- hist(z, nclass = 50, plot = FALSE)
  xseq <- seq(from = min(z.hist$breaks), to = max(z.hist$breaks), length = 100)[-1]
  z.chi <- dchisq(xseq, df = k - 1)
  maxi <- max(z.hist$density, z.chi)
  hist(z, proba = TRUE, nclass = 50, ylim = c(0,maxi), col = grey(0.8),
  xlab = "-2 log(L(H0)/L(H1))", las = 1,
  main = paste("Distribution d'échantonnage\n", nsimul, " simulations"))
  lines(xseq, z.chi, lwd = 2)
  lines(xseq, dchisq(xseq, df = k - 1 -1), lwd = 1)
  lines(xseq, dchisq(xseq, df = k - 1 +1), lwd = 1)

  qqplot( qchisq(ppoints(n=nsimul), df = k - 1), sort(z),
  main = paste("ChiSquare", k - 1, "ddl Q-Q Plot"),
  xlab = "Theoretical Quantiles",
  ylab = "Sample Quantiles", las = 1)
  abline(0,1)
}
hexp.k(k = 5)

```



On l'aura compris, dans tous les cas, la statistique du rapport de vraisemblance :

$$-2 \log \left(\frac{L(H_0)}{L(H_1)} \right) = 2 \log(L(H_1)) - 2 \log(L(H_0))$$

suit asymptotiquement, sous l'hypothèse nulle, une loi du χ^2 dont le nombre de degré de liberté est la différence du nombre de paramètres estimés.

Ce qui est essentiel est que dans le rapport il y ait :

- ★ au numérateur la vraisemblance de l'estimation au maximum de vraisemblance d'une hypothèse H_0
- ★ au dénominateur la vraisemblance de l'estimation au maximum de vraisemblance d'une hypothèse H_1
- ★ et que le domaine de variation des paramètres sous H_0 soit contenu dans le domaine de variation des hypothèses sous H_1 .

La statistique du rapport de vraisemblance définit alors une grande classe de tests statistiques. Exemple : sur un même parcours répétés 12 fois par temps couvert et 11 fois par temps clair on a dénombré des animaux d'une espèce donnée.

```
(x <- c( 1, 1, 1, 0, 3, 1, 1, 0, 1, 4, 2, 2))
[1] 1 1 1 0 3 1 1 0 1 4 2 2
(y <- c( 2, 2, 6, 2, 3, 0, 1, 0, 3, 1, 4))
[1] 2 2 6 2 3 0 1 0 3 1 4
```

Le temps influe-t-il sur la densité d'animaux rencontrés ? On suppose que le dénombrement suit une loi de Poisson. H_0 (pas d'effet) conduit à l'estimation d'un seul paramètre au maximum de vraisemblance :

```
mean(c(x,y))
[1] 1.782609
```


La log-vraisemblance de l'hypothèse nulle est :

```
(lvH0 <- -2*sum(dpois(c(x,y), mean(c(x,y)),log = TRUE)))
[1] 78.15036
```

L'hypothèse alternative H_1 conduit à l'estimation de deux paramètres au maximum de vraisemblance :

```
mean(x)
[1] 1.416667
mean(y)
[1] 2.181818
(lvH1 <- -2*sum(dpois(x,mean(x),log=T))-2*sum(dpois(y,mean(y),log=T)))
[1] 76.26271
```

ce qui donne un Khi2 à 1 ddl :

```
lvH0 - lvH1
[1] 1.887655
```

C'est exactement le test de la régression poissonnienne :

```
(fac <- factor(c(rep("x",12),rep("y",11))))
[1] x x x x x x x x x x x x y y y y y y y y y
Levels: x y
anova(glm(c(x,y)~fac,family=poisson),test="Chi")
Analysis of Deviance Table
Model: poisson, link: log
Response: c(x, y)
Terms added sequentially (first to last)

      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
NULL    22    29.917
fac     1     1.8877    21     28.030  0.1695
```

Le glm manipule les vraisemblances par écarts au modèle saturé (chaque donnée estime un paramètre) ce qui donne les déviations résiduelles :

```
(lvsat <- -2*sum(dpois(c(x,y), c(x,y),log=T)))
[1] 48.23296
lvsat + glm(c(x,y)~fac,family=poisson)$null.deviance
[1] 78.15036
lvsat + glm(c(x,y)~fac,family=poisson)$deviance
[1] 76.26271
```

4 Le G^2 d'une table de contingence

La question est posée à 980 personnes "A quel parti êtes vous le plus fortement lié?" (exemple 2.4.4 p. 31 de [1]). Implanter une table de contingence en partant d'un vecteur :

```
(w <- c(279,73,225,165,47,191))
[1] 279 73 225 165 47 191
```

En faire une matrice :

```
(w <- matrix(data = w, nrow = 2, ncol = 3, byrow = TRUE))
      [,1] [,2] [,3]
[1,] 279  73 225
[2,] 165  47 191
```

En faire une table :

```
(w <- as.table(w))
      A  B  C
A 279 73 225
B 165 47 191
```

Ajouter les noms des modalités :

```
dimnames(w)[[1]]
[1] "A" "B"
dimnames(w)[[1]] <- c("Femmes","Hommes")
dimnames(w)[[2]]
[1] "A" "B" "C"
dimnames(w)[[2]] <- c("Democrates","Independants","Republicain")
w
      Democrates Independants Republicain
Femmes      279          73          225
Hommes      165          47          191
```

	Democrates	Independants	Republicain
Femmes	279	73	225
Hommes	165	47	191

TABLE 2 – Exemple de table de contingence, tiré de l'exemple 2.4.4 p. 31 de [1]

Tester l'indépendance :

```
chisq.test(w)
      Pearson's Chi-squared test
data:  w
X-squared = 7.0095, df = 2, p-value = 0.03005
```

$$\chi^2 = \sum_{ij} \frac{(n_{ij} - c_{ij})^2}{c_{ij}} \quad c_{ij} = \frac{n_{i\bullet} n_{\bullet j}}{n_{\bullet\bullet}} \quad (10)$$

Récupérer les effectifs théoriques sous l'hypothèse d'indépendance :

```
(west <- chisq.test(w)$expected)
      Democrates Independants Republicain
Femmes 261.4163  70.65306  244.9306
Hommes 182.5837  49.34694  171.0694
sum((w-west)^2/west)
[1] 7.009544
```

Dans les tables de contingences, si p_{ij} est la probabilité pour qu'une observation appartienne à la catégorie ij , la probabilité de l'observation est :

$$P(Y_{11} = n_{11}, \dots, Y_{IJ} = n_{IJ}) = \left(\sum_{i,j} n_{ij} \right)! \prod_{i,j} \frac{p_{ij}^{n_{ij}}}{n_{ij}!} \quad (11)$$

Si une hypothèse donne un modèle de la distribution bivariée sous la forme

$$\begin{aligned} (\pi_{ij}) \quad & 1 \leq i \leq I \\ & 1 \leq j \leq J \end{aligned}$$

alors qu'un autre modèle donne la distribution du type

$$(\mu_{ij}) \begin{matrix} 1 \leq i \leq I \\ 1 \leq j \leq J \end{matrix}$$

le jeu des logarithmes simplifie la variation de vraisemblance par :

$$-2 \log \left(L \left(\mathbf{x}, (\pi_{ij}) \begin{matrix} 1 \leq i \leq I \\ 1 \leq j \leq J \end{matrix} \right) \right) + 2 \log \left(L \left(\mathbf{x}, (\mu_{ij}) \begin{matrix} 1 \leq i \leq I \\ 1 \leq j \leq J \end{matrix} \right) \right) = 2 \sum_{ij} n_{ij} \log \frac{\mu_{ij}}{\pi_{ij}} \quad (12)$$

Le modèle complet estime $IJ - 1$ paramètres et le modèle d'indépendance des marges estime $I - 1 + J - 1$ paramètres. Le rapport de vraisemblance suit un χ^2 à $J - 1 - I + 1 - J + 1 = (I - 1)(J - 1)$ degrés de liberté et s'écrit :

$$G^2 = 2 \sum_{ij} n_{ij} \log \frac{n_{ij}}{c_{ij}} \quad c_{ij} = \frac{n_{i\bullet} n_{\bullet j}}{n_{\bullet\bullet}} \quad (13)$$

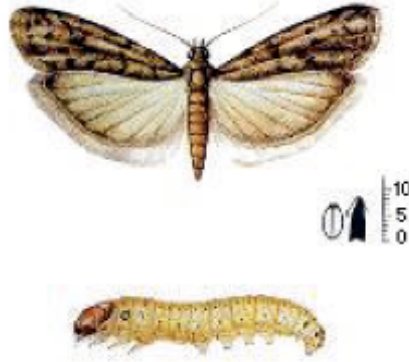
```
2*sum(w*log(w/west))
[1] 7.002594
```

Le χ^2 d'ajustement et le rapport de vraisemblance appelé G^2 sont ici presque identiques. Ils donnent le même test sur des logiques complètement différentes. Le G^2 est la porte d'entrée dans les modèles log-linéaires qui permettent d'étudier les tables de contingence à plus de deux dimensions :

```
loglin(table = w, margin = c(1,2))
2 iterations: deviation 5.684342e-14
$lrt
[1] 7.002594
$pearson
[1] 7.009544
$df
[1] 2
$margin
[1] 1 2
```

5 Un problème entomologique

Il est posé et résolu dans [4]. La Teigne de la farine, *Ephestia kuehniella* (Lepidoptera, Pyralidae) est un redoutable ravageur universellement répandu.



<http://www.casabernardo.com.br/ident17.jpg>

Ses larves sont parasitées par *Venturia canescens* (Hymenoptera : Ichneumonidae) :



<http://biomserv.univ-lyon1.fr/sitelabo/GestFich/D3/Venturia7.html>

Le parasite se développe en même temps que l'hôte.

Il y a deux types d'expériences élémentaires. Dans le premier, une expérience élémentaire consiste à placer un nombre donné de larves d'hôtes sains (stade 3 âgées de 21 jours) dans un cylindre rempli de 10 grammes de semoule et à compter le nombre d'adultes émergents. L'effectif de départ (larves) varie de 10 à 500 et permet de modéliser la survie en fonction de la densité.

Dans le second, une expérience élémentaire consiste à placer un nombre donné de larves d'hôtes pratiquement tous parasités au préalable (stade 3 âgées de 21 jours) dans un cylindre de même type. Chaque larve hôte contient un parasite : les deux larves se développent simultanément et l'hôte meurt juste avant l'émergence du parasite. Le résultat est le nombre d'adultes émergents qui sont alors tous des parasites. L'effectif de départ (larves) varie également de 10 à 500 et permet de modéliser la survie du parasite en fonction de la densité de l'hôte.

Le travail pour préparer les tubes contenant un grand nombre de larves est énorme et l'expérience s'est déroulée sur deux années. Il y a 45 expériences élémentaires et les résultats nous sont directement communiqués par les auteurs (Table 3). Importer les données sous la forme d'un `data.frame` :

```
cbetal2002 <- read.table("http://pbil.univ-lyon1.fr/R/donnees/cbetal2002.txt", header=TRUE)
summary(cbetal2002)
```

id		n		surv		para	
Min.	: 1	Min.	: 10.0	Min.	: 2.0	non:	21
1st Qu.	:12	1st Qu.	: 50.0	1st Qu.	: 46.0	oui:	24
Median	:23	Median	:200.0	Median	:121.0		
Mean	:23	Mean	:207.1	Mean	:108.7		
3rd Qu.	:34	3rd Qu.	:350.0	3rd Qu.	:163.0		
Max.	:45	Max.	:500.0	Max.	:227.0		

	id	n	surv	para
1	1	25	22	non
2	2	50	50	non
3	3	100	83	non
4	4	150	121	non
5	5	200	134	non
6	6	250	149	non
7	7	300	130	non
8	8	350	156	non
9	9	400	121	non
10	10	450	138	non
11	11	500	92	non
12	12	25	14	oui
13	13	25	20	oui
14	14	25	24	oui
15	15	50	40	oui
16	16	50	46	oui
17	17	100	82	oui
18	18	150	94	oui
19	19	200	170	oui
20	20	250	180	oui
21	21	300	190	oui
22	22	350	147	oui
23	23	400	188	oui
24	24	450	217	oui
25	25	500	183	oui
26	26	10	9	non
27	27	25	25	non
28	28	50	43	non
29	29	100	84	non
30	30	150	89	non
31	31	200	122	non
32	32	250	163	non
33	33	300	168	non
34	34	350	129	non
35	35	400	138	non
36	36	10	2	oui
37	37	25	11	oui
38	38	50	31	oui
39	39	100	71	oui
40	40	150	113	oui
41	41	200	123	oui
42	42	250	175	oui
43	43	300	202	oui
44	44	350	176	oui
45	45	400	227	oui

TABLE 3 – Les données de [4]

Exercice : essayer de reproduire les données la figure 1 de [4] :

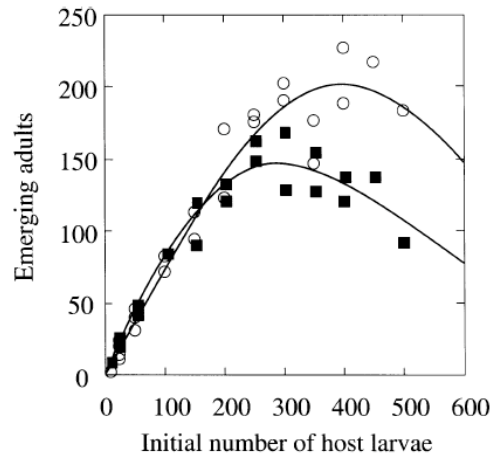
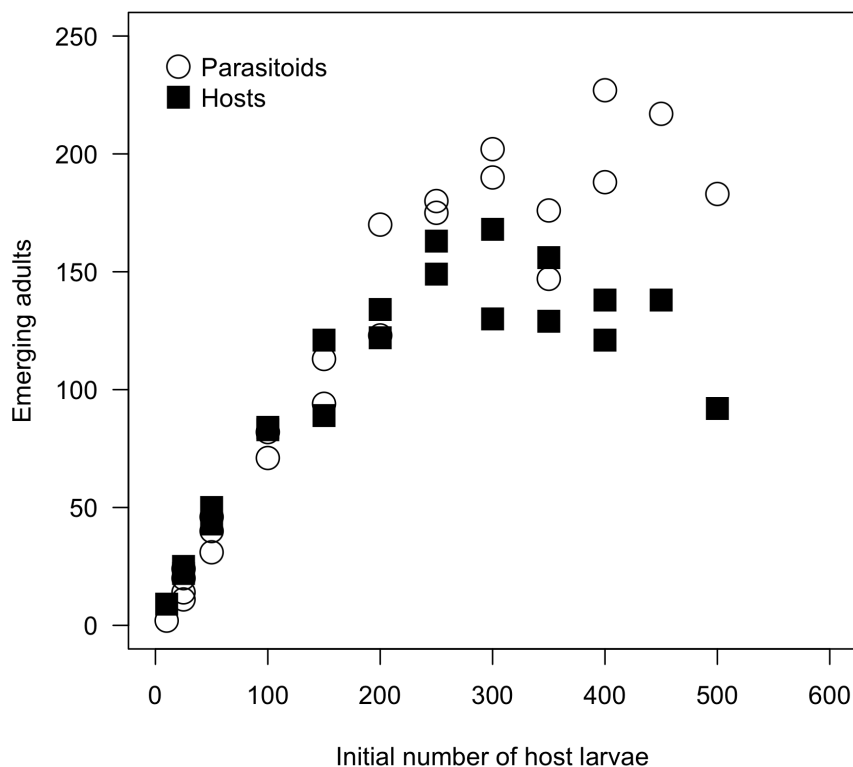


Fig. 1. The number of emerging adult parasitoids (○) and hosts (■) as a function of the initial number of host larvae (data pooled over the 2 years in which the experiments were performed). The lines result from fitting the data to the generalised Ricker model (see text).

Vous pouvez reproduire le graphique ci-après avec les fonctions `plot()`, `points()` et `legend()` :

Number of emerging adults as function of the initial number of host larvae



Le but est de comparer les modèles de survie des hôtes et des parasites qui sont manifestement différents. Le résultat d'une expérience est une variable binomiale de paramètre n_i (fixée) et p_i (à modéliser). Le modèle à utiliser est du type :

$$p(n) = \frac{\alpha}{1 + \left(\frac{n}{K}\right)^\theta} \tag{14}$$

Le nombre d'adultes émergents, A , en fonction du nombre initial d'hôtes, x , est alors donné par :

$$A(x) = \frac{\alpha x}{1 + \left(\frac{x}{K}\right)^\theta} \tag{15}$$

où α est la pente de la relation $A - x$ pour les faibles valeurs de x , K la capacité du milieu, et θ un paramètre qui contrôle l'allure générale du modèle. Pour se familiariser avec ce modèle, qui est une généralisation du modèle de Beverton et Holt [6, 5], on définit une fonction :

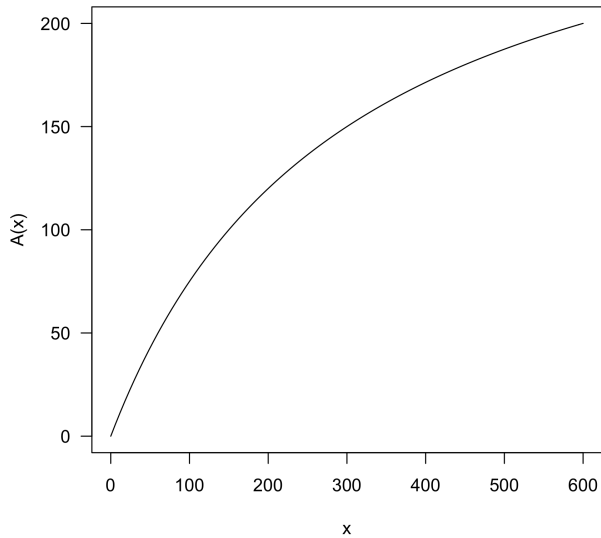
```
A <- fonction(x) {
  p <- alpha/(1+(x/K)^theta)
```



```

        return(x*p)
    }
    alpha <- 1
    K <- 300
    theta <- 1
    curve(A, xlim = c(0, 600), las = 1)

```

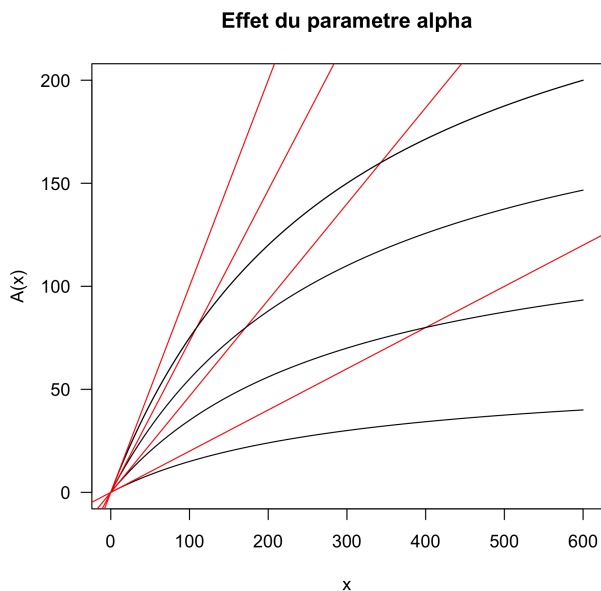


Vérifier que le paramètre α contrôle bien la pente initiale :

```

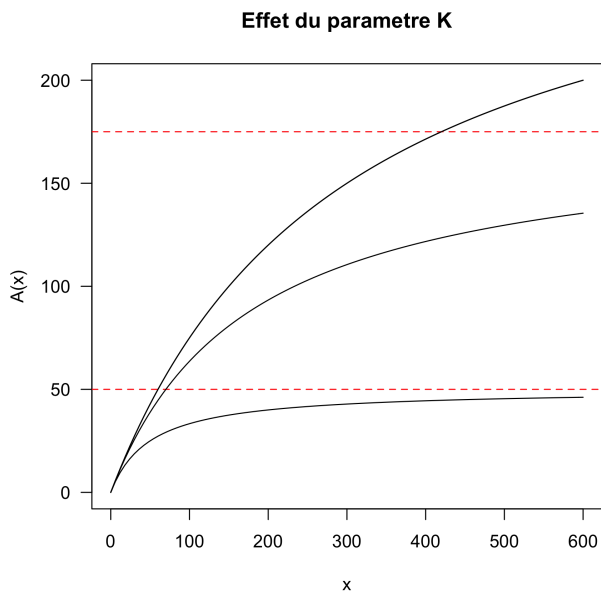
curve(A, xlim = c(0, 600), las = 1, main = "Effet du parametre alpha")
for(alpha in seq(from = 0.2, to = 1, length = 4))
{
  curve(A, add = TRUE)
  abline(c(0,alpha), col = "red")
}

```



Vérifier que le paramètre K contrôle bien l'amplitude maximum de la réponse :

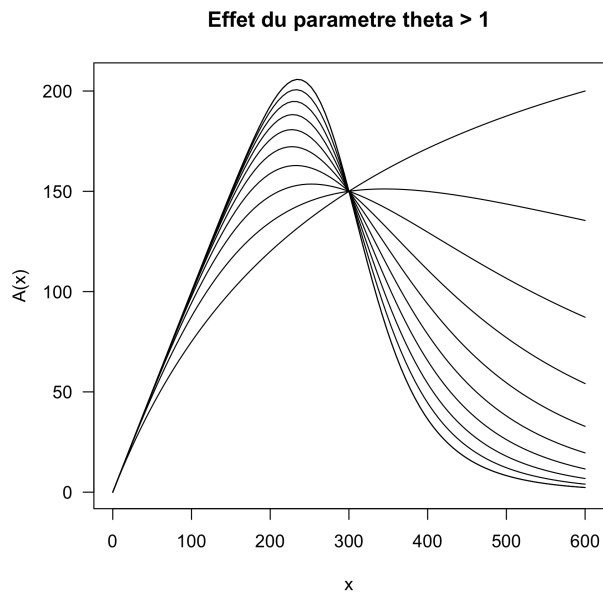
```
curve(A, xlim = c(0, 600), las = 1, main = "Effet du parametre K")
for(K in seq(from = 50, to = 300, length = 3))
{
  curve(A, add = TRUE)
  abline(h=K, col = "red", lty = 2)
}
```



Étudier l'effet du paramètre θ , quel est son sens biologique ?

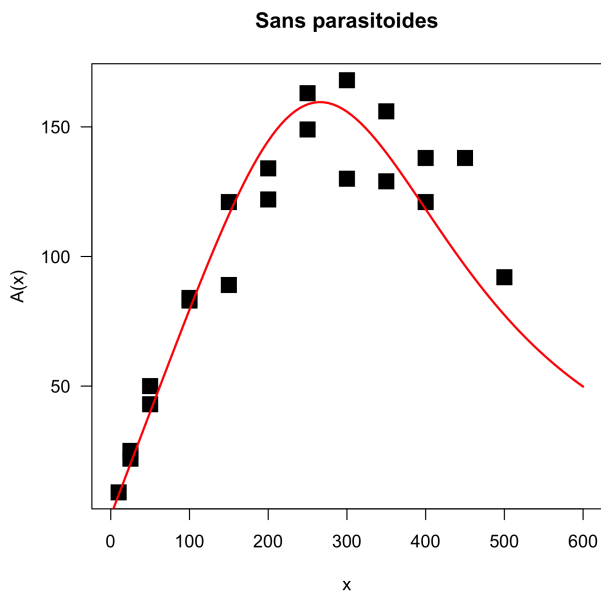
```
curve(A, xlim = c(0, 600), las = 1, main = "Effet du parametre theta > 1")
for(theta in seq(from = 1, to = 8, length = 10))
```

```
{  
  curve(A, add = TRUE)  
}
```

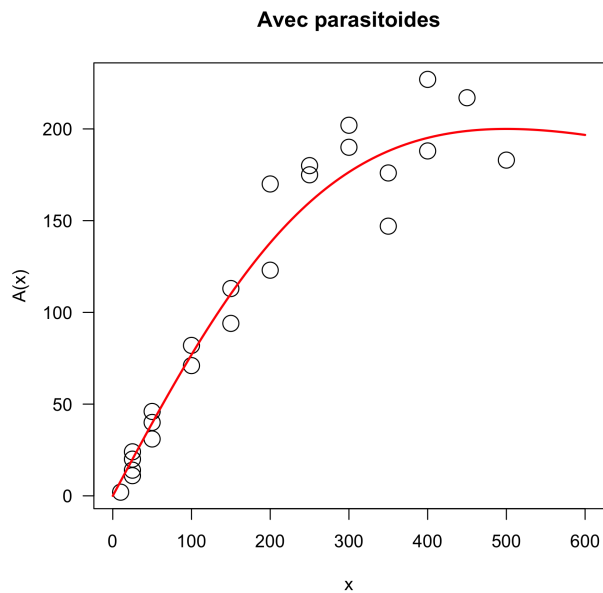


Ajuster par la méthode du pifomètre [7] le modèle pour l'expérience sans parasitoïdes.

```
host <- cbetal2002[cbetal2002$para == "non",]  
plot(host$n, host$surv, las = 1, cex = 2, xlim = c(0, 600), pch = 15,  
      xlab = "x", ylab = "A(x)", main = "Sans parasitoïdes")  
alpha <- 0.8  
K <- 350  
theta <- 4  
curve(A, add = TRUE, col = "red", lwd = 2)
```



Faire la même chose pour l'expérience avec parasitoïdes. Le but est d'avoir des valeurs de départ dans la fonction de minimisation qui, au moins, ne soient pas délirantes.



Nous pouvons maintenant utiliser ces premières estimations grossières pour maximiser la vraisemblance avec la fonction `nlm()` :

```
LL <- fonction(p, xobs, yobs) {
  alpha <- p[1]
  K <- p[2]
  theta <- p[3]
  proba <- alpha/(1+(xobs/K)^theta)
```

```

    ll <- sum(dbinom(x = yobs, size = xobs, prob = proba, log = TRUE))
    return(-2*ll)
  }
  (whost <- nlm(f = LL, p = c(0.8, 350, 4), xobs = host$n , yobs = host$surv) )
$minimum
[1] 178.3176
$estimate
[1] 0.9024473 316.0099897 2.3007631

$gradient
[1] 2.813749e-06 -1.852749e-08 1.173551e-06

$code
[1] 1

$iterations
[1] 23
  (wpara <- nlm(f = LL, p = c(0.8, 500, 2), xobs = para$n , yobs = para$surv) )
$minimum
[1] 267.3179
$estimate
[1] 0.7491453 500.8508529 3.0496566

$gradient
[1] 1.656986e-07 9.136243e-11 1.006521e-08

$code
[1] 1

$iterations
[1] 34

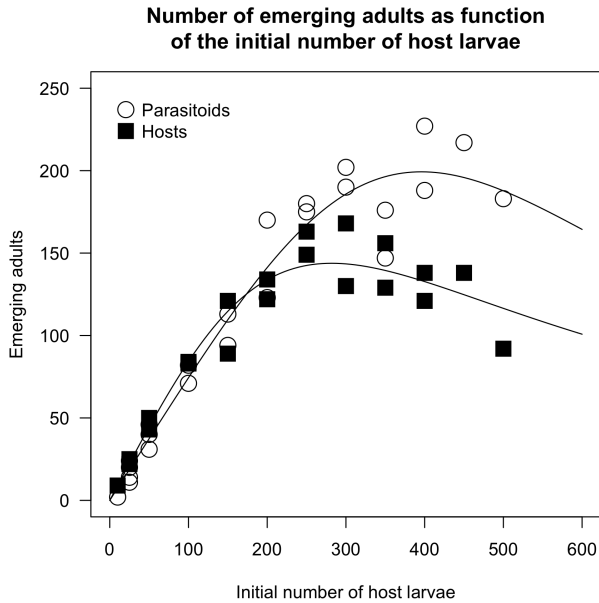
```

Représenter les résultats sous la même forme que la figure 1 de [4] :

```

plot(para$n, para$surv, las = 1, cex = 2, xlim = c(0, 600),
      ylim = c(0,250), ylab ="Emerging adults",
      xlab = "Initial number of host larvae",
      main = "Number of emerging adults as function\nof the initial number of host larvae")
points(host$n, host$surv, pch=15, cex = 2)
legend(0, 250, c("Parasitoids","Hosts"), pch = c(1,15), pt.cex = 2, bty = "n")
alpha <- wpara$estimate[1]
K <- wpara$estimate[2]
theta <- wpara$estimate[3]
curve(A, add = TRUE)
alpha <- whost$estimate[1]
K <- whost$estimate[2]
theta <- whost$estimate[3]
curve(A, add = TRUE)

```



Vérifier que les estimations des paramètres sont conformes à celles publiées par les auteurs dans la table 1 de [4] :

Parameter	Healthy hosts	Parasitised host
Generalised Ricker model		
α	0.935 ± 0.028	0.753 ± 0.017
K	392.48 ± 9.68	574.086 ± 19.549
θ	1.595 ± 0.130	2.506 ± 0.295
Generalised Beverton and Holt model		
α	0.902 ± 0.029	0.749 ± 0.016
K	316.01 ± 12.24	500.85 ± 14.37
θ	2.300 ± 0.194	3.0496 ± 0.340

Les modèles obtenus pour les deux conditions d'expérience sont visiblement différents. Mais la question est de savoir si tous leurs paramètres sont *significativement* différents. On peut commencer par le premier paramètre, α , et comparer la vraisemblance du modèle général à 6 paramètres $(\alpha_1, \alpha_2, K_1, K_2, \theta_1, \theta_2)$ avec un modèle à 5 paramètres $(\alpha, K_1, K_2, \theta_1, \theta_2)$.

Pour le modèle général, $(\alpha_1, \alpha_2, K_1, K_2, \theta_1, \theta_2)$, on doit retrouver les estimations précédentes :

```
LL6 <- function(p)
{
  l11 <- LL(p[1:3], xobs = host$n, yobs = host$surv)
  l12 <- LL(p[4:6], xobs = para$n, yobs = para$surv)
  return(l11 + l12)
}
w6 <- nlm(f = LL6, p = c(0.8, 350, 4, 0.8, 500, 2))
w6[1:2]
$minimum
[1] 445.6358
$estimate
[1] 0.9024537 316.0069756 2.3007264 0.7492276 500.6159087 3.0517035
```

Pour le modèle avec α en commun, $(\alpha, K_1, K_2, \theta_1, \theta_2)$, on trouve :

```
LL5a <- function(p)
{
  ll1 <- LL(p[1:3], xobs = host$n, yobs = host$surv)
  ll2 <- LL(p[c(1,4,5)], xobs = para$n, yobs = para$surv)
  return(ll1 + ll2)
}
w5a <- nlm(f = LL5a, p = c(0.8, 350, 4, 500, 2))
w5a[1:2]
$minimum
[1] 466.8047
$estimate
[1] 0.7907972 358.2271178 3.0167512 489.8387372 2.5345657
```

Ce qui est significativement moins bon :

```
1 - pchisq(q = w5a$minimum - w6$minimum, df = 1)
[1] 4.205331e-06
```

Pour le modèle avec K en commun, $(\alpha_1, \alpha_2, K, \theta_1, \theta_2)$, on trouve que l'hypothèse nulle est trivialement fausse :

```
LL5k <- function(p)
{
  ll1 <- LL(p[1:3], xobs = host$n, yobs = host$surv)
  ll2 <- LL(p[c(4,2,5)], xobs = para$n, yobs = para$surv)
  return(ll1 + ll2)
}
w5k <- nlm(f = LL5k, p = c(0.8, 350, 4, 0.8, 2))
w5k[1:2]
$minimum
[1] 571.8581
$estimate
[1] 0.8864545 323.4531536 2.3933193 1.0999855 0.4808793
```

Pour le modèle avec θ en commun, $(\alpha_1, \alpha_2, K_1, K_2, \theta)$, on trouve :

```
LL5t <- function(p)
{
  ll1 <- LL(p[1:3], xobs = host$n, yobs = host$surv)
  ll2 <- LL(p[c(4,5,3)], xobs = para$n, yobs = para$surv)
  return(ll1 + ll2)
}
w5t <- nlm(f = LL5t, p = c(0.8, 350, 4, 0.8, 500))
w5t[1:2]
$minimum
[1] 449.4962
$estimate
[1] 0.8749616 327.2323758 2.5473465 0.7650645 510.9296964
1 - pchisq(q = w5t$minimum - w6$minimum, df = 1)
[1] 0.04943743
```

Les trois couples de paramètres sont donc significativement différents. On peut enfin vérifier l'hypothèse binomiale :

```
alpha <- whost$estimate[1]
K <- whost$estimate[2]
theta <- whost$estimate[3]
(obstheo <- cbind(host$surv, A(host$n)))
      [,1]      [,2]
[1,]  22 22.495536
[2,]  50 44.482776
[3,]  83 84.274343
[4,] 121 114.710676
[5,] 134 133.788615
[6,] 149 142.497027
[7,] 130 143.453740
[8,] 156 139.453169
[9,] 121 132.717513
[10,] 138 124.752838
[11,]  92 116.477961
```

```

[12,] 9 9.021275
[13,] 25 22.495536
[14,] 43 44.482776
[15,] 84 84.274343
[16,] 89 114.710676
[17,] 122 133.788615
[18,] 163 142.497027
[19,] 168 143.453740
[20,] 129 139.453169
[21,] 138 132.717513

(hostchi <- chisq.test(obstheo))
      Pearson's Chi-squared test
data:  obstheo
X-squared = 14.0888, df = 20, p-value = 0.826
1-pchisq(q = hostchi$statistic, df = nrow(obstheo) - 1 - 3)
X-squared
0.6608091

alpha <- wpara$estimate[1]
K <- wpara$estimate[2]
theta <- wpara$estimate[3]
(obstheo <- cbind(para$surv, A(para$n)))

      [,1]      [,2]
[1,] 14 18.726626
[2,] 20 18.726626
[3,] 24 18.726626
[4,] 40 37.424058
[5,] 46 37.424058
[6,] 82 74.368125
[7,] 94 109.598777
[8,] 170 141.236652
[9,] 180 167.198116
[10,] 190 185.815178
[11,] 147 196.370466
[12,] 188 199.275320
[13,] 217 195.832757
[14,] 183 187.771889
[15,] 2 7.491404
[16,] 11 18.726626
[17,] 31 37.424058
[18,] 71 74.368125
[19,] 113 109.598777
[20,] 123 141.236652
[21,] 175 167.198116
[22,] 202 185.815178
[23,] 176 196.370466
[24,] 227 199.275320

(parachi <- chisq.test(obstheo))
      Pearson's Chi-squared test
data:  obstheo
X-squared = 26.6088, df = 23, p-value = 0.273
1-pchisq(q = parachi$statistic, df = nrow(obstheo) - 1 - 3)
X-squared
0.1466332

```

Rien ne permet de contredire une variabilité binomiale des résultats. C'est peut-être, du point de vue statistique, le résultat le plus extraordinaire de cette expérience remarquablement maîtrisée.

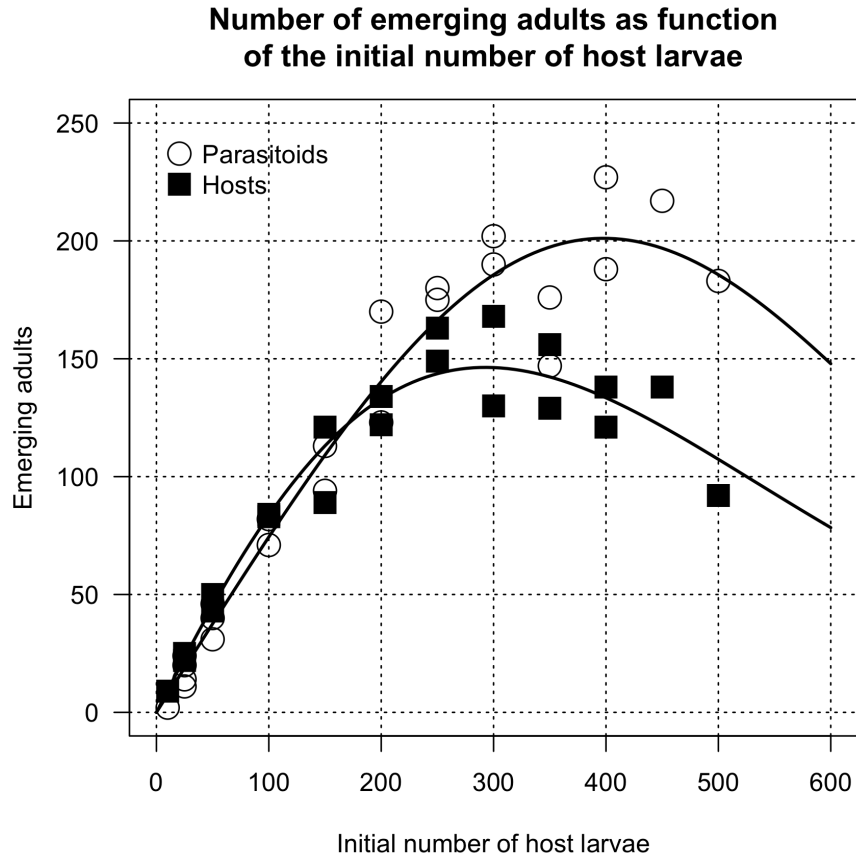
Pour reproduire exactement la figure 1 de [4] il faut utiliser le modèle de Ricker [8, 10, 3, 5] généralisé :

$$p(n) = \alpha e^{-\left(\frac{n}{K}\right)^\theta} \quad (16)$$

Soit en effectifs théoriques :

$$A(n) = n\alpha e^{-\left(\frac{n}{K}\right)^\theta} \quad (17)$$

Exercice : reproduire exactement la figure 1 de [4], on peut utiliser la fonction `grid()` pour faciliter la comparaison :



Références

- [1] A. Agresti. *An introduction to categorical data analysis*. John Wiley, New York, 1996.
- [2] M.A. Banks and W. Eichert. Computer note. whichrun (version 3.2) : a computer program for population assignment of individuals based on multilocus genotype data. *Journal of Heredity*, 91 :87–89, 2000.
- [3] C. Bernstein. Density dependence and the stability of host-parasitoid systems. *Oikos*, 47 :176–180, 1986.
- [4] Carlos Bernstein, Annie Heizmann, and Emmanuel Desouhant. Intraspecific competition between healthy and parasitised hosts in a host-parasitoid system : consequences for life-history traits. *Ecological Entomology*, 27(4) :415–423, 2002.
- [5] W.M. Getz. A hypothesis regarding the abruptness of density dependence and the growth rate of populations. *Ecology*, 77 :2014–2026, 1996.
- [6] J. Maynard Smith and M. Slatkin. The stability of predator-prey system. *Ecology*, 54 :384–391, 1973.
- [7] G. Pif. Sideral glasses to increase the accuracy of bull’s-eyes fit method in non-linear regression. *Pif Gadget*, 1 :1–48, 1969.
- [8] M.J. Pomerantz, W.R. Tomas, and M.E. Gilpin. Asymetries in population growth regulated by intraspecific competition : empirical studies and model test. *Oecologia*, 47 :311–322, 1980.
- [9] G. Saporta. *Probabilités, analyse des données et statistique*. Technip, Paris, 1990.
- [10] W.R. Thomas, M.J. Pomerantz, and M.E. Gilpin. Chaos, asymmetric growth and group selection for dynamical stability. *Ecology*, 61 :1312–1320, 1980.