

# TP3 stats

Romain PEREIRA

2 Avril 2018

## 1. Ajuster une loi de Bernoulli

```
N <- 10
X_N <- rbinom(n=N, size=1, prob=0.7)
X_N
```

```
## [1] 1 1 1 0 1 0 1 1 1 1
```

(a) Une estimation simple et empirique est  $p \simeq \frac{m}{N}$ , avec  $m$  le nombre de 1 dans l'échantillon.

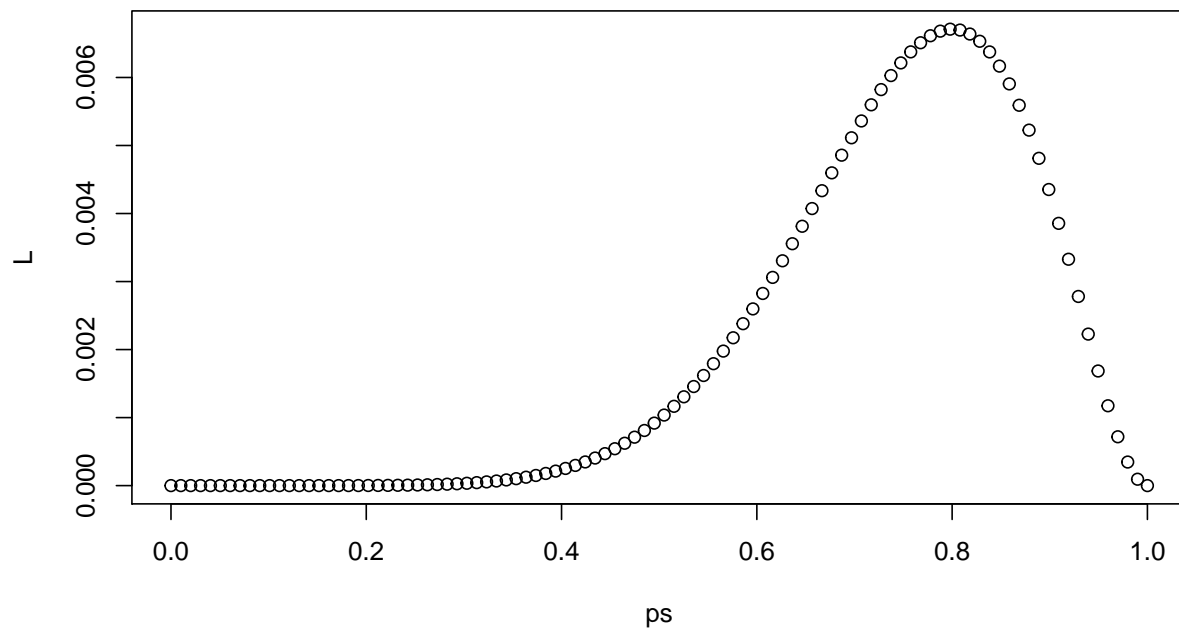
(b)

```
# fonction qui à un échantillon de Bernoulli, et une probabilité 'p',
# associe la vraisemblance d'un échantillon de Bernoulli
L_bern <- function(X_N, p) {
  s <- sum(X_N)
  n <- length(X_N)
  L_N <- (p**s) * ((1 - p)**(n - s))
  return (L_N)
}
L_bern(X_N, 0.5)
```

```
## [1] 0.0009765625
```

(c)

```
ps <- c() # vecteur discretisant [0, 1]
L <- c() # vecteur des vraisemblances
# pour chaque p dans [0, 1], avec une discretisation à n points...
n <- 100 # taille de la discretisation
p0 <- 0 # [p0, pn]
pn <- 1
step <- (pn - p0) / (n - 1) # pas de discretisation
m <- 1 # l'indice pour lequel le maximum est atteint
for (i in 1:n) {
  ps[i] <- p0 + (i - 1) * step
  L[i] <- L_bern(X_N, ps[i])
  m <- if (L[i] > L[m]) i else m
}
plot(ps, L)
```



On remarque que la vraisemblance semble être une fonction continue, admettant un maximum en  $p_m$ , et symétrique par rapport à la droite  $x = p_m$ .

Ici, on a  $p_m =$

```
ps[m]
```

```
## [1] 0.7979798
```

(d)

```
optimize(function(p) { L_bern(X_N, p) }, interval=c(p0, pn), maximum=TRUE)
```

```
## $maximum
## [1] 0.799982
##
## $objective
## [1] 0.006710886
```

Remarque: on trouve une valeur proche de celle obtenu avec notre discretisation en (c).

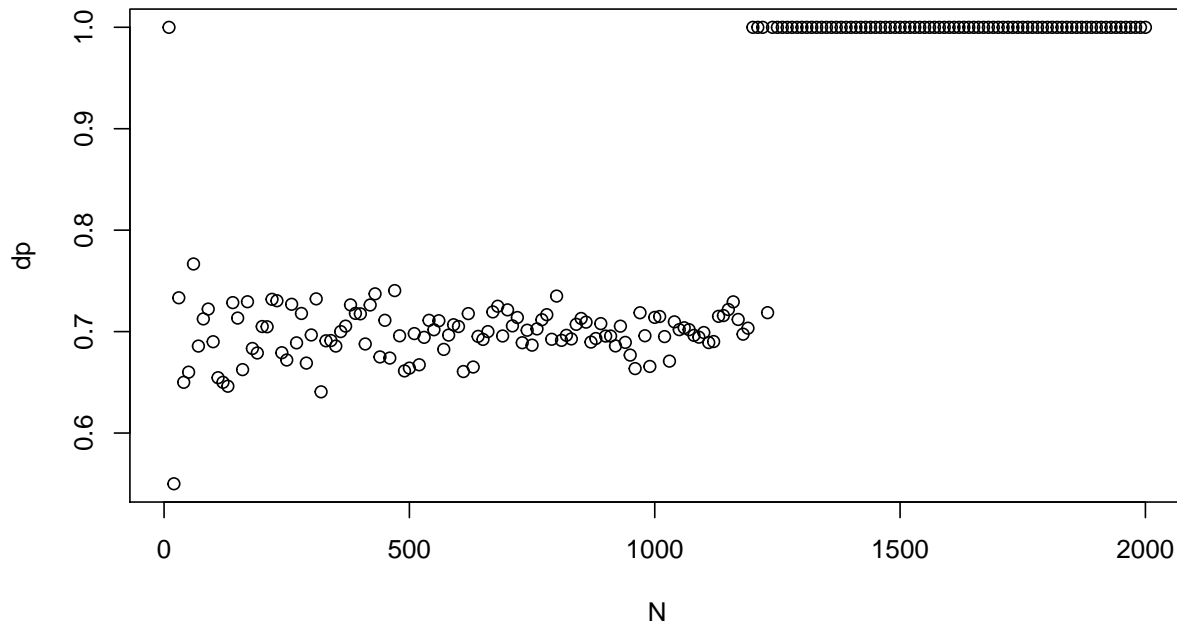
(e)

```
p    <- 0.7 # le paramètre réel de notre loi de bernouilli
N0   <- 10
Nn   <- 2000
n    <- 200
N    <- 0 * 1:n # vecteur des 'N'
dp   <- 0 * 1:n # vecteur des '| p - px |', où p = 0.7, et px la valeur d'optimize
step <- (Nn - N0) / (n - 1)
for (i in 1:n) {
  N[i] <- as.integer(N0 + (i - 1) * step)
  X_Ni <- rbinom(n=N[i], size=1, prob=p)
```

```

px    <- optimize(function(px) { L_bern(X_Ni, px) }, interval=c(0, 1), maximum=TRUE)
dp[i] <- px$maximum
}
plot(N, dp)

```



```

mean(dp)

## [1] 0.8201202

(f)
df <- read.csv(file="./distribution_inconue_2_100_realisations.csv", header=TRUE)

```

## 2. Ajuster une loi de Bernoulli