TP_1

EL_Hadrami_N'DOYE_Ismael_Ramde

15/01/2021

Exercice 1

```
# importation du fichier ScriptEtud.R
source("scriptsEtud.R")
```

```
n <- 500
P <- 1 / 10
theta <- 0.8
n.sim <- 1000
set.seed(2001)
piaw <- questDelicateMeth1(n,P,theta,n.sim)
```

a. Calcul de la moyenne et l'ecart type estimés

```
probaoui1 <- P * (2 * theta - 1) - theta + 1
# X est une loi Binomiale
X <- rbinom(n.sim,n,probaoui1)
a <- (theta - 1) / ((2 * theta) - 1)
b <- 1 / (n * ((2 * theta) - 1))
moyenneEst1 <- a + (b * n * probaoui1)
varX <- n * probaoui1 * (1 - probaoui1)
ecarttype1 <- sqrt(b^2 * varX)
```

b. Les valeurs theoriques correspondantes

```
mean(piaw)
```

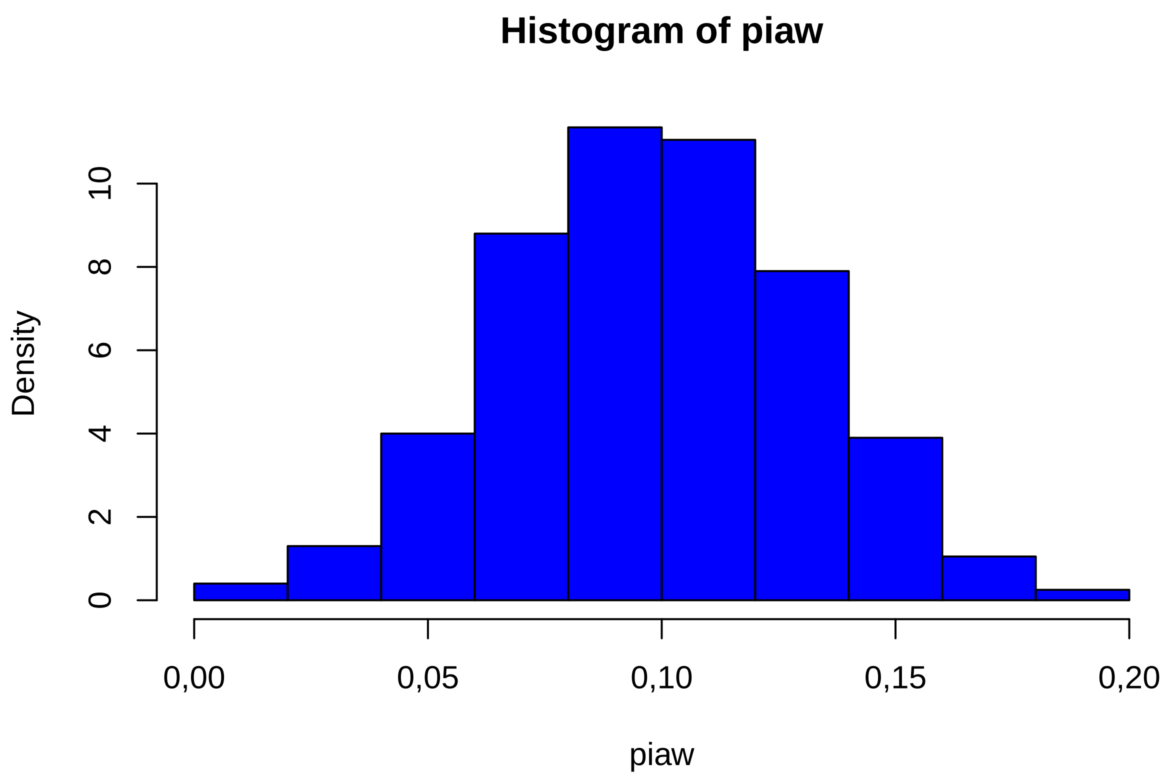
```
## [1] 0,1007333
```

```
sd(piaw)
```

```
## [1] 0,03225096
```

c.

```
hist(piaw,probability = TRUE,col = "blue")
```



2. Methode 2

```
alpha <- 0.217
pia <- questDelicateMeth2(n, P, theta, alpha, n.sim)
```

a. Calcul de la moyenne et l'ecart type estimés

```
probaoui2 <- (theta * P) + ((1 - theta) * alpha)
a <- alpha * (theta - 1) / theta
b <- 1 / (n * theta)
moyenneEst2 <- a + (b * n * probaoui2)
varX <- n * probaoui2 * (1 - probaoui2)
ecarttype2 <- sqrt((1 / (n * theta)^2) * varX)
```

b. Les valeurs theoriques correspondantes

```
mean(pia)
```

```
## [1] 0,100005
```

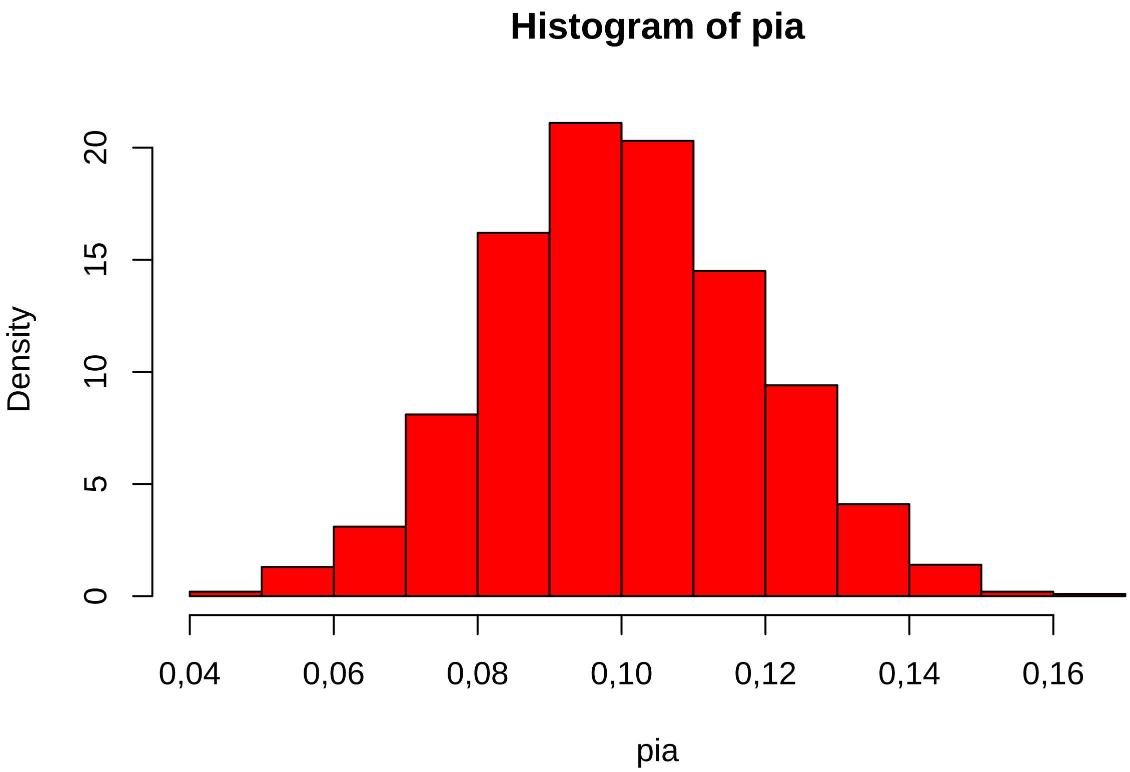
```
sd(pia)
```

```
## [1] 0,01842001
```

c.

Tracage de l'histogramme

```r
hist(pia,probability = TRUE,col = "red")
```



3.

3.Echantillonage classique

4.a : Calcul de $\bar{y}_u et$ S pour la population

```r
source("Script2.R")
ybarru <- sum(pop) / N
sigmacarre <- (1 / N) * sum((pop - ybarru)^2)
S <- sqrt((sigmacarre * N) / (N - 1))
```

4.b : Generons une trame de données contenant toutes les echantillons possibles

4.c :

```r
options(OutDe=",")
pop <- c (3 , 6 , 24 , 27 , 30 , 36 , 51 , 57)
N <- length ( pop )
n <- 3
(n.ech <- choose (N , n))
```

```
## [1] 56
```

```r
ech.all <- cbind ( choix <- t ( matrix ( combn (1: N , n ) ,n , n.ech ) ) ,( matrix (pop [ choix ] , n.
head (ech.all)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    2    3    3    6   24
## [2,]    1    2    4    3    6   27
## [3,]    1    2    5    3    6   30
## [4,]    1    2    6    3    6   36
```

```
## [5,]    1    2    7    3    6   51
## [6,]    1    2    8    3    6   57
```

```r
donneepop <- ech.all[,4:6]
ybarr <- apply(donneepop,1,mean)
s <- apply(donneepop,1,sd)
sybarr <- (1 - (n / N)) * (s^2 / n)
b.inf <- ybarr - (1.96 * sqrt(sybarr))
b.sup <- ybarr + (1.96 * sqrt(sybarr))
size <- 56
incl <- rep(0,size)
for (i in 1:size){
  if(ybarru <= b.sup[i] && ybarr >= b.inf[i]){
    incl[i] <- 1
  }
  else{
    incl[i] <- 0
  }
}
tram.donnee <- cbind(ybarr,s,b.inf,b.sup,incl)
head(tram.donnee,5)
```

```
##      ybarr        s      b.inf    b.sup incl
## [1,]    11 11,35782  0,8391437 21,16086    0
## [2,]    12 13,07670  0,3014103 23,69859    0
## [3,]    13 14,79865 -0,2390710 26,23907    0
## [4,]    15 18,24829 -1,3251646 31,32516    1
## [5,]    20 26,88866 -4,0549579 44,05496    1
```

4.d : Verifions numeriquement que $\mu_{\bar{y}} = \bar{y_{\mu}}$ et que$\sigma_{\bar{y}} = (\sqrt{1-f})\frac{S}{n}$

```r
# moyenne de chaque population a 3 echantillons
muybarr <- mean(tram.donnee[,1])
muybarr
```

```
## [1] 29,25
```

```r
ybarru
```

```
## [1] 29,25
```

```r
f <- n / N
head(sqrt(1 - f) * (s / sqrt(n)),5)
```

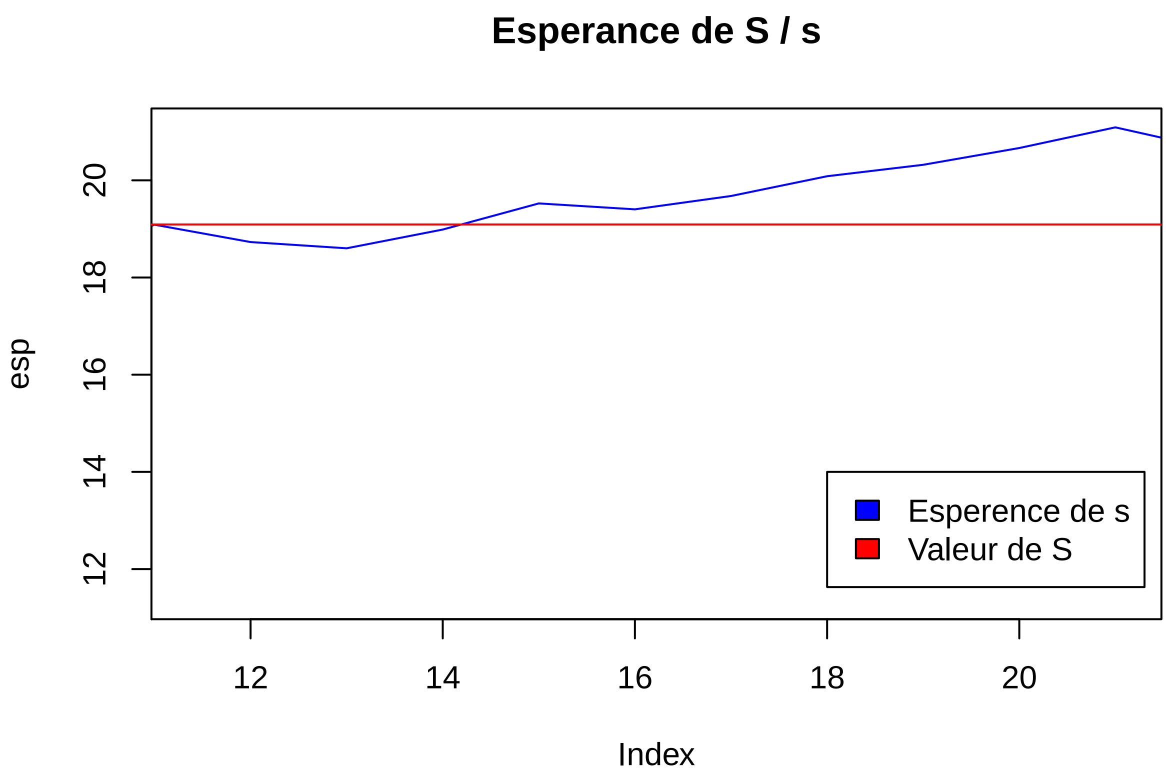
```
## [1]  5,184110  5,968668  6,754628  8,329166 12,272938
```

```r
head(sqrt(sybarr),5)
```

```
## [1]  5,184110  5,968668  6,754628  8,329166 12,272938
```

4.e : s est - il un estimateur sans biais de S

```r
esp <- cumsum(tram.donnee[,2]) / (1:size)
plot(esp,xlim = c(min(esp),max(esp)),type = "l",col="blue",main = "Esperance de S / s")
abline(h = S,col="red")
legend(x = 18,y = 14,legend=c("Esperence de s","Valeur de S"),fill = c("blue","red"),col=c("blue,red"))
```



D'apres la figure on observe que l'esperance de s ne converge pas vers la valeur de S. Donc s n'est pas un estimateur sans biais de S.

f. Déterminer la probabilité de commettre dans l'estimation de la moyenne une erreur

- plus de deux unités

On compte toutes les indices du vecteurs ybarr($\mu_{\bar{y}}$) dont leurs valeurs sont superieur a $\bar{y_\mu} + 2$ puis on divise par le nombre d'element du vecteur $\mu_{\bar{y}}$ pour avoir la probabilité.

```
length(which(ybarr > ybarru + 2 )) / size
```

```
## [1] 0,3571429
```

- plus de cinq unités

On compte toutes les indices du vecteurs ybarr($\mu_{\bar{y}}$) dont leurs valeurs sont superieur a $\bar{y_\mu} + 5$ puis on divise par le nombre d'element du vecteur $\mu_{\bar{y}}$ pour avoir la probabilité.

```
length(which(ybarr > ybarru + 5 )) / size
```

```
## [1] 0,3035714
```

- plus de 25%

On compte toutes les indices du vecteurs ybarr($\mu_{\bar{y}}$) dont leurs valeurs sont superieur a $\bar{y_\mu} + 0.25$ puis on divise par le nombre d'element du vecteur $\mu_{\bar{y}}$ pour avoir la probabilité.

```
length(which(ybarr > ybarru + 0.25 )) / size
```

```
## [1] 0,4821429
```

g. Calcul de la probabilité de se tromper de plus de 20% dans l'estimation de l'écart type S de la population.

```
length(which(s > S + 0.20 )) / size
```

```
## [1] 0,3571429
```

h.Le niveau de confiance de l'intervalle de confiance donné par la formule $\bar{y} - 2\widehat{\sigma}_{\bar{y}} \leq \bar{y}_u \leq \bar{y} + 3\widehat{\sigma}_{\bar{y}}$

On compte le nombre d'element de $\bar{y} - 2\widehat{\sigma}_{\bar{y}}$ et $\bar{y} + 3\widehat{\sigma}_{\bar{y}}$ dont $\bar{y} - 2\widehat{\sigma}_{\bar{y}} \leq \bar{y}_u \leq \bar{y} + 3\widehat{\sigma}_{\bar{y}}$

```
born.inf <- ybarr - (2 * sqrt(sybarr))
born.sup <- ybarr + (3 * sqrt(sybarr))
length(which(ybarru >= born.inf & ybarru <= born.sup)) / size
```

```
## [1] 0,9285714
```

Donc le niveau de confiance de cette intervall est de 92.86%