

TP 1

EL Hadrami N'DOYE et Ismaïl RAMDÉ

15/01/2021

```
library(tinytex)
```

Exercice 1

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

Estimation d'une proportion dans le contexte d'une question délicate

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

Question 1 . Simulation 1000 fois de la methode 1

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

On sait que la moyenne estimée correspond à l'espérance tant-disque l'écart-type estimé correspond à la racine carré de la variance. Ici X suit une loi Binomiale.

Donc $E(X) = a + b * (n * P)$ et $var = b^2 * (n * P(1 - P))$

```
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))
```

```
# Moyenne estimée
moyenneEst1 <- a + (b * n * probaoui1)
moyenneEst1
```

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

```
# Ecart type estimé
varX1 <- n * probaoui1 * (1 - probaoui1)
ecarttype1 <- sqrt(b^2 * varX1)
ecarttype1
```

```
## [1] 0,03269387
```

b. Les valeurs theoriques correspondantes

```
# moyenne
mean(piaw)
```

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

```
# écart-type
sd(piaw)
```

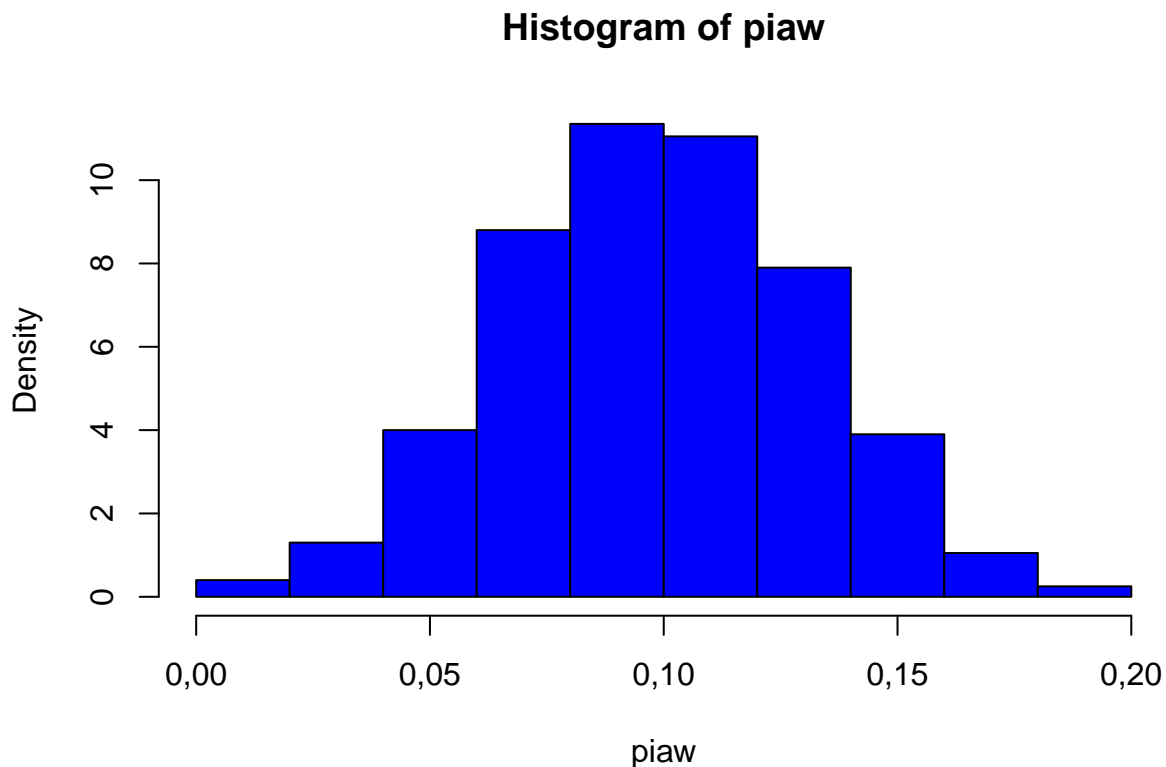
```
## [1] 0,03225096
```

c. Comment se compare la variance de cette méthode avec celle, ou l'on poserait directement la question dans le contexte où tous les répondants répondent honnêtement?

Dans le cas où tous les répondants répondent honnêtement, la variance serait inférieure à celle de la méthode précédente.

d. Tracé d'un histogramme des valeurs observées de $\hat{\pi}_{AW}$

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



e. Commentaires

Après les différents calculs, on remarque la moyenne estimée et la l'écart-type estimé sont respectivement égaux à la moyenne théorique et l'écart-type théorique. Aussi on trouve que les estimateurs sont sans biais. Pour finir à travers l'histogramme, on voit que la distribution suit une loi normale.

2. Reprise du travail avec la méthode 2 en utilisant $\alpha = 0.217$

```
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)
```

```
# Moyenne estimée
moyenneEst2 <- a + (b * n * probaoui2)
moyenneEst2
```

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

```
# Ecart type estimé
varX2 <- n * probaoui2 * (1 - probaoui2)
ecarttype2 <- sqrt((1 / (n * theta)^2) * varX2)
ecarttype2
```

```
## [1] 0,01838583
```

b. Les valeurs theoriques correspsecarreondantes

```
# moyenne
mean(pia)
```

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

```
# écart-type
sd(pia)
```

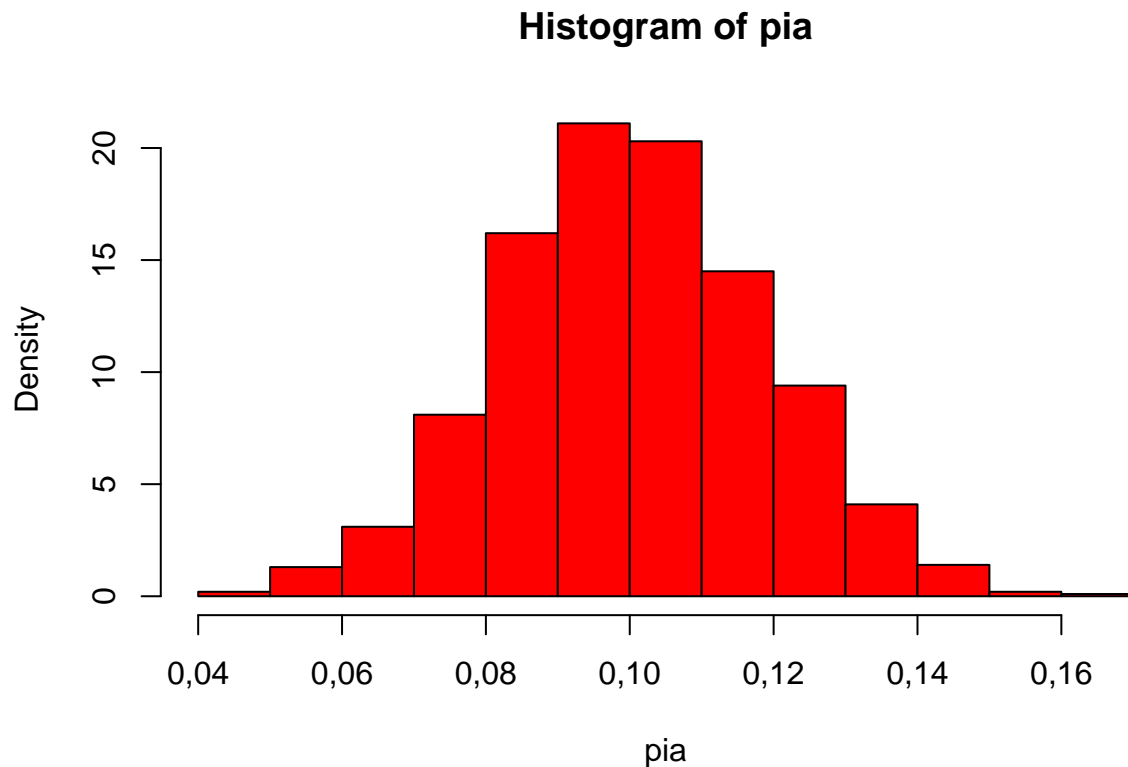
```
## [1] 0,01842001
```

c. Comment se compare la variance de cette méthode avec celle où, l'on poserait directement la question dans le contexte où tous les répondants répondent honnêtement\$?

Dans le cas où tous les repondants repondent honêtement, la variance serait inférieur à celle de la méthode precedente.

d. Tracage d'un histogramme des valeurs observées de $\hat{\pi}_{AW}$

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



3. Évaluons les différentes EQM

```
# méthode (b)
EQM1 <- varX2 + (moyenneEst2 - theta)
EQM1

## [1] 53,38622

Calculons de  $\hat{\pi}_{AW}$ 
prop <- c(0, 0.1, 0.2, 0.3)
prob <- 1 - prop
prob

## [1] 1,0 0,9 0,8 0,7

set.seed(2001)
piaw1 <- rep(0,4)
piaw1

## [1] 0 0 0 0

for (i in 1:4) {
  piaw1[i] <- questDelicateMeth1(n,prob[i],theta,n.sim)
}
piaw1

## [1] 0,9733333 0,8833333 0,7766667 0,6966667

# calcul de la moyenne estimée et de la variance estimée

# Moyenne estimée
moyenEst <- mean(piaw1)
```

```
moyenEst
```

```
## [1] 0,8325
```

```
# Ecart type estimé
```

```
var1 <- var(piaw1)
```

```
var1
```

```
## [1] 0,01466204
```

```
EQM2 <- var1 + (moyenEst - theta)
```

```
EQM2
```

```
## [1] 0,04716204
```

La méthode (a) a une EQM plus petite que celle de la méthode (b).

Question 4 . Échantillonnage classique

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

```
source("Script2.R")
```

```
ybarru <- sum(pop) / N
```

```
sigmacarre <- (1 / N) * sum((pop - ybarru)^2)
```

```
S <- sqrt((sigmacarre * N) / (N - 1))
```

b. Generation d'une trame de données contenant toutes les echantillons possibles

```
head(ech.all)
```

```
##      i 1  i 2  i 3 pop [ i 1 ] pop [ i 2 ] pop [ i 3 ]
## 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
```

c. Création de la nouvelle trame de données

```
options(OutDe=",")
```

```
pop <- c (3 , 6 , 24 , 27 , 30 , 36 , 51 , 57)
```

```
N <- length ( pop )
```

```
n <- 3
```

```
(n.ech <- choose (N , n))
```

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

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

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

```
##      ybar      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
```

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

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

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

```
ybarru
```

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

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

```
## [1] 8,714213
```

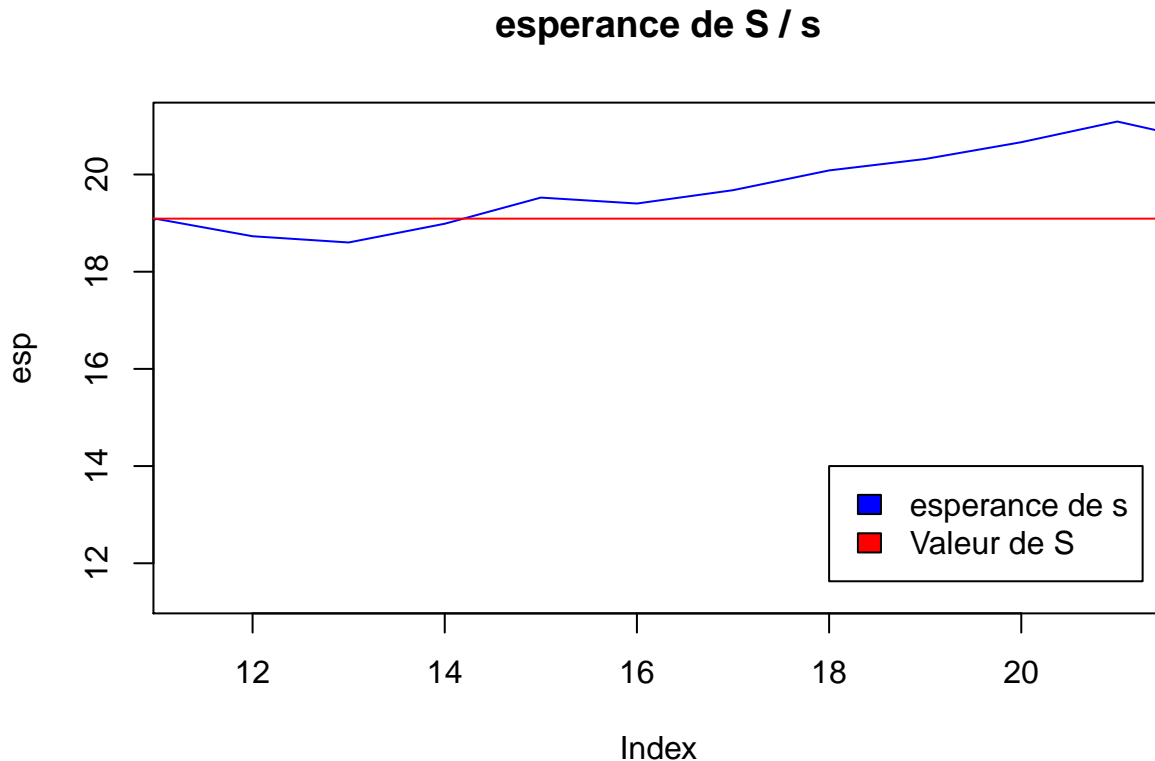
```
sygmaybarr <- sqrt((1 - (n / N)) * (S^2 / n))
sygmaybarr
```

```
## [1] 8,714213
```

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

```
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("esperance de s","Valeur de S"),fill = c("blue","red"),
      col=c("blue,red"))
```



D'après 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 ybar dont leurs valeurs sont superieur a $\bar{y}_u + 5$ puis on divise par le nombre d'element du vecteur ybar pour avoir la probabilité.

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

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

- plus de cinq unités

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

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

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

- plus de 25%

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

```
length(which(ybar > 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

$$\bar{y} - 2\hat{\sigma}_{\bar{y}} \leq \bar{y}_u \leq \bar{y} + 2\hat{\sigma}_{\bar{y}}$$

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

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

```
## [1] 0,8928571
```

Donc le niveau de confiance de cette interval est de 89.28%

i. Le niveau de confiance de l'intervalle de confiance donné par la formule

$$\bar{y} - 3\hat{\sigma}_{\bar{y}} \leq \bar{y}_u \leq \bar{y} + 3\hat{\sigma}_{\bar{y}}$$

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

```
## [1] 0,9642857
```

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

j. Le niveau de confiance de l'intervalle de confiance donné par la formule

$$\bar{y} - 2\sigma_{\bar{y}} \leq \bar{y}_u \leq \bar{y} + 2\sigma_{\bar{y}}$$

```
born.inf <- ybar - (2 * sygmaybarr)
born.sup <- ybar + (2 * sygmaybarr)
length(which(ybarru >= born.inf & ybarru <= born.sup)) / size
```

```
## [1] 0,9642857
```

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

5.a: Dans quelle mesure nos simulations confirment-elles que \bar{y} , est un estimateur sans biais de \bar{y}_u

```
# Realisation d'un script creant un echantillon de taille 5 1000 fois
y <- c (36 , 36.5 , 37 , 37.5 , 38 , 38.5 , 39 , 39.5 ,
        40 , 40.5,41 ,41.5 , 42 , 42.5 , 43 ,
        43.5 , 44 , 44.5 , 45 , 45.5 , 46 ,46.5)
e <- c (0.02 , 0.03 , 0.04 , 0.06 , 0.10 , 0.12 , 0.13 , 0.10 , 0.07 , 0.05 ,
        0.05 , 0.04 , 0.04,
        0.03 , 0.03 , 0.02 , 0.02 , 0.01 , 0.01 , 0.01 , 0.01 , 0.01)
# Taille de l'echantillon
NE <- 1000
ne <- 5
echt <- matrix(rep(NA,NE),nrow = NE,ncol = ne)
set.seed(2001)
for (i in 1:NE){
  echt[i,] <- sample (y ,ne , replace = TRUE , prob = e )
}
```



```
ybarr <- apply(echt, 1, mean)
mean(ybarr)
```

```
## [1] 39,768
```

```
ybarru <- mean(y)
ybarru
```

```
## [1] 41,25
```

Après avoir simulé 1000 fois des échantillons de taille 5 on constate que \bar{y} est un estimateur sans biais de \bar{y}_u à partir de la mesure 40.

5.b Calcule de la variance des \bar{y} Théoriquement, que devrait être cette variance?, Énonçons le théorème qui justifie notre réponse

```
varybarr <- var(ybarr)
varybarr
```

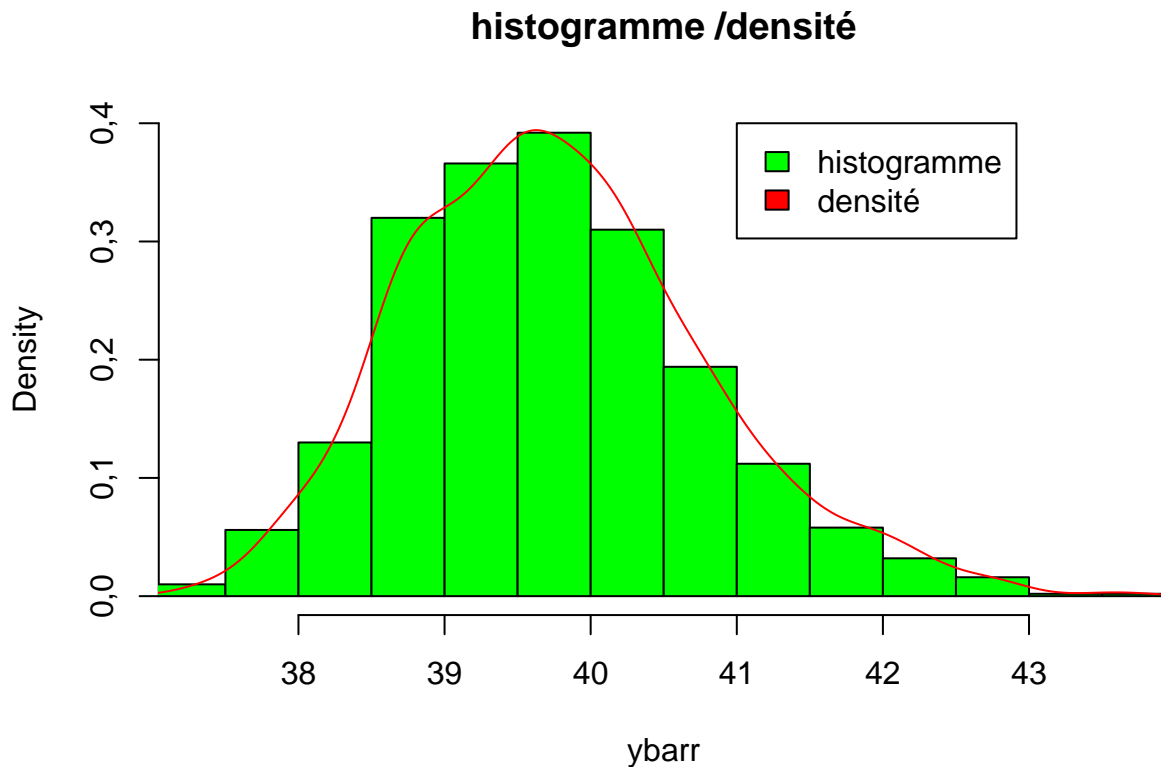
```
## [1] 1,048444
```

```
(1 / (NE - 1)) * sum((ybarr - mean(ybarr))^2)
```

```
## [1] 1,048444
```

5.c La variable \bar{y} suit-elle une loi normale:

```
hist(ybarr, probability = TRUE, xlim = c(min(ybarr), max(ybarr)), ylim = c(0, 0.4), col = "green",
     main = "histogramme / densité")
lines(density(ybarr), col = "red")
legend(x = 41, y = 0.4, c("histogramme", "densité"), fill = c("green", "red"), col = c("green", "red"))
```



Test de normalité

```
shapiro.test(ybarr)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: ybarr  
## W = 0,98492, p-value = 1,216e-08
```

Soient les deux hypothèses suivantes:

H_0 : la variable \bar{y} est une loi normale

H_a : la variable \bar{y} n'est pas une loi normale

-Décision

le p-value est inférieur a la valeur de alpha fixée ($\alpha = 0.05$), donc on rejette H_0 et on accepte H_a .
Donc \bar{y} n'est pas une loi normale.

Question d:

```
# calcul de sigma y bar  
sycarr <- (1 / length(y)) * sum((y - ybarru)^2)  
sigmaybar <- sqrt(sycarr / ne)  
length(which(ybarr > 1.96 * sd(ybarr))) / NE
```

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

e. La probabilité de recouvrement de l'intervalle de confiance

```
se <- apply(echt,1,sd)  
a <- ybarr - (1.96 * (se / sqrt(ne)))  
b <- ybarr + (1.96 * (se / sqrt(ne)))  
length(which(ybarru <= b & ybarru >=a)) / NE
```

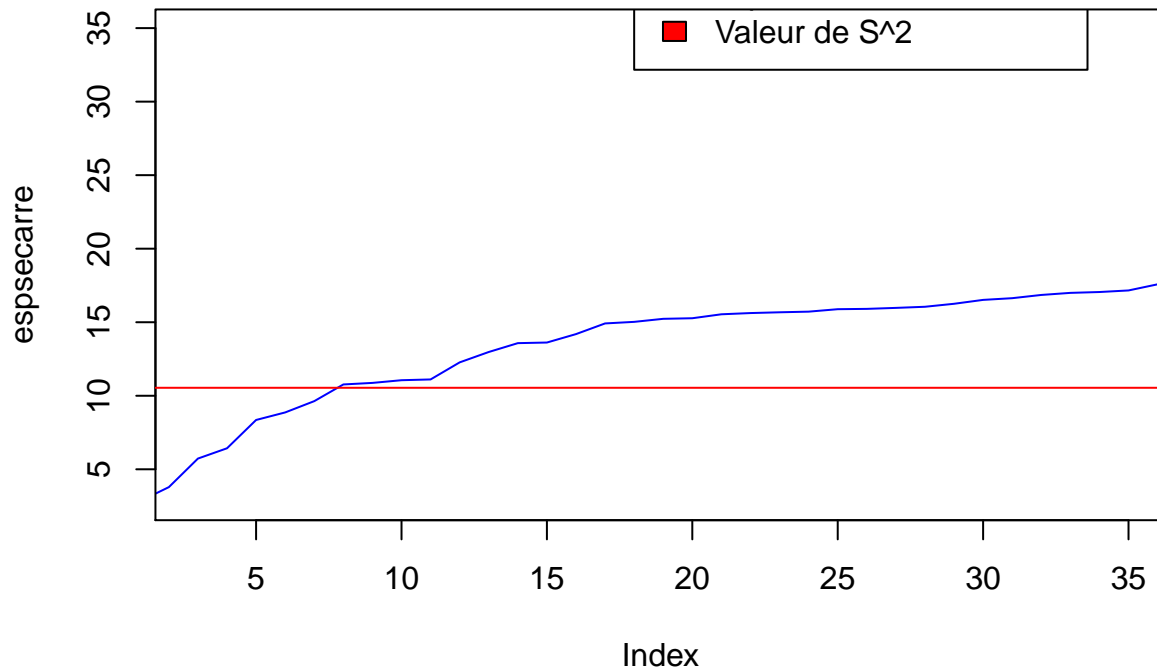
```
## [1] 0,546
```

Donc la probabilité que l'intervall de confiance qui recouvre la vraie moyenne est de 54.6%.

Question f : Est-ce que vos simulations semblent indiquer que l'estimateur s^2 est sans biais de S^2

```
secarr <- se^2  
sizey <- length(y)  
# calcul de S^2  
S.carre <- (1 / (sizey - 1)) * sum((y - mean(y))^2)  
espsecarre <- cumsum(secarr / (1:length(secarr)))  
plot(espsecarre,xlim = c(min(espsecarre),max(espsecarre)),type = "l",  
     col="blue",main = "espsecarreerance de S / s")  
abline(h = S.carre,col="red")  
legend(x = 18,y = 40,legend=c("espsecarreerance de s^2","Valeur de S^2"),  
      fill = c("blue","red"),col=c("blue,red"))
```

espsecarreance de S / s



D'après la figure on observe que l'esperance de s^2 ne converge pas vers la valeur de S^2 . Donc s^2 n'est pas un estimateur sans biais de S^2 .

Question g: La théorie statistique indique que dans certaines conditions l'intervalle de confiance calculé par la formule

$$\frac{(n-1) * s^2}{\chi_{\alpha/2;n-1}^2} < S^2 < \frac{(n-1) * s^2}{\chi_{1-\alpha/2;n-1}^2}$$

est un intervalle de confiance à 95%. Estimons le niveau réel de cet intervalle de confiance.

```
alhaveuil <- 0.05
bsup <- ((ne - 1) * secarr) / qchisq(alhaveuil/2,ne - 1)
binf<- ((ne - 1) * secarr) / qchisq(1-(alhaveuil / 2),ne - 1)
length(which(S.carre < bsup & S.carre > binf)) / NE
```

```
## [1] 0,85
```

Le niveau reel de cet intervall de confiance est de 86.6%.