

TD n° 6 : Estimation d'autres paramètres ; ST

Exercice 1. Sur les 15287 employés d'une entreprise, on souhaite connaître la proportion p_U d'entre eux qui pratique au moins une activité sportive. On décide de former 4 strates en fonction de l'âge des employés. On considère alors :

- la strate U_1 : ensemble des employés âgés de 18 à 25 ans,
- la strate U_2 : ensemble des employés âgés de 26 à 39 ans,
- la strate U_3 : ensemble des employés âgés de 40 à 55 ans,
- la strate U_4 : ensemble des employés âgés de 56 à 65 ans.

On dispose des informations suivantes :

U_h	U_1	U_2	U_3	U_4
N_h	2536	5210	5871	1670
n_h	150	200	200	20
p_{ω_h}	0.81	0.55	0.45	0.25

1. Donner une estimation ponctuelle de p_U .
2. Donner une estimation ponctuelle de l'écart-type de l'estimateur de p_U .
3. Déterminer un intervalle de confiance pour p_U au niveau 90%.

Exercice 2. On veut estimer le taux de réussite à la session d'examens de juin dans une université qui comprend 1250 inscrits en première année, 800 en deuxième, 500 en troisième et 300 en quatrième. On veut estimer le taux de réussite à partir des résultats de 600 étudiants.

1. On prélève un échantillon de 600 étudiants suivant un plan de sondage aléatoire de type PESR. On trouve un taux de réussite de 67%. Donner un intervalle de confiance du taux de réussite global au niveau 95%.
2. Est-ce que l'estimation aurait été meilleure avec un plan de sondage aléatoire de type ST avec pour strates les années d'étude ?
3. Combien d'étudiants aurait-il fallu prendre par année pour faire un plan de sondage aléatoire de type STP ?
4. Avec un échantillon de 600 étudiants prélevé suivant un plan de sondage aléatoire de type STP, on obtient :

$$p_{\omega_1} = 0.78, \quad p_{\omega_2} = 0.70, \quad p_{\omega_3} = 0.64, \quad p_{\omega_4} = 0.53.$$

Donner une estimation ponctuelle de taux de réussite global.

Exercice 3. Une population U est partagée en 3 strates U_1 , U_2 et U_3 . On prélève un échantillon de 100 individus suivant un plan de sondage aléatoire de type ST et on mesure un caractère binaire Y sur chacun d'entre eux. On s'intéresse donc à l'estimation de la proportion-population p_U . On dispose des informations suivantes :

Strate U_h	U_1	U_2	U_3
Effectif N_h	175	375	450
Proportion p_{U_h}	0.21	0.53	0.62

1. On considère un plan de sondage aléatoire de type STP.
 - (a) Déterminer les effectifs des échantillons pour chacune des strates.
 - (b) Calculer l'erreur quadratique moyenne de l'estimateur de p_U .
2. On considère maintenant un plan de sondage aléatoire de type STO.
 - (a) Déterminer les effectifs des échantillons pour chacune des strates.
 - (b) Calculer l'erreur quadratique moyenne de l'estimateur de p_U .
3. Comparer les résultats des 2 plans de sondage considérés.

Exercice 4. On étudie le caractère $Y = \text{"fumeur"}$ sur 22 garçons.

1. Décrire brièvement l'enjeu des commandes R suivantes :

```

U_1 = c("Bob", "Nico", "Ali", "Fabien", "Malik", "John", "Jean",
"Chris", "Karl")
Y_1 = c(0, 1, 0, 1, 1, 1, 0, 0, 1)
U_2 = c("Jean", "Bill", "Omar", "Raul", "Mia")
Y_2 = c(1, 1, 0, 0, 1)
U_3 = c("Paul", "Chael", "Nathan", "Sam", "Tom", "Tim", "Leo", "Kevin")
Y_3 = c(1, 0, 1, 1, 1, 0, 1, 1)
dat = cbind.data.frame(c(U_1, U_2, U_3), c(rep(1, length(U_1)),
rep(2, length(U_2)), rep(3, length(U_3))))
names(dat) = c("noms", "souspop")
U = c(U_1, U_2, U_3)
Y = c(Y_1, Y_2, Y_3)
p_w = NULL
var_p_w = NULL
u = NULL
library(sampling)

for (i in (1:30)) {
s = strata(dat, "souspop", size = c(3, 2, 3), method = "srswor")
s
u = rbind(u,U[s[,2]])
N_h = c(length(U_1), length(U_2), length(U_3))
N = sum(N_h)
n_h = c(3, 2, 3)
y = Y[s[,2]]
p_w_h = unlist(tapply(y, s$souspop, mean))
p_w[i] = sum(N_h * p_w_h) / N
var_p_w[i] = (1 / N^2) * sum(N_h^2 * (1-n_h / N_h) * (p_w_h *
(1 - p_w_h) / (n_h - 1) ))
}

u

```

Cela renvoie :

```
[,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
[1,] "Bob" "Malik" "Jean" "Bill" "Omar" "Chael" "Nathan" "Tim"
[2,] "Malik" "Jean" "Chris" "Raul" "Mia" "Sam" "Tim" "Kevin"
[3,] "Bob" "Ali" "Chris" "Raul" "Mia" "Nathan" "Sam" "Tom"
[4,] "Bob" "Fabien" "Chris" "Jean" "Raul" "Nathan" "Sam" "Tom"
[5,] "John" "Jean" "Chris" "Omar" "Raul" "Paul" "Nathan" "Leo"
[6,] "Bob" "Jean" "Chris" "Omar" "Raul" "Nathan" "Tom" "Tim"
[7,] "Nico" "Ali" "John" "Bill" "Raul" "Sam" "Tom" "Tim"
[8,] "Nico" "Ali" "Karl" "Bill" "Mia" "Sam" "Tom" "Leo"
[9,] "Ali" "Fabien" "Malik" "Bill" "Omar" "Nathan" "Leo" "Kevin"
[10,] "Bob" "Nico" "Malik" "Jean" "Bill" "Chael" "Tim" "Leo"
[11,] "Nico" "Ali" "Chris" "Raul" "Mia" "Chael" "Nathan" "Tom"
[12,] "Ali" "Jean" "Karl" "Omar" "Mia" "Paul" "Chael" "Nathan"
[13,] "Nico" "Ali" "Fabien" "Jean" "Bill" "Nathan" "Sam" "Tom"
[14,] "Bob" "John" "Chris" "Jean" "Bill" "Paul" "Nathan" "Tim"
[15,] "Malik" "Jean" "Chris" "Omar" "Raul" "Paul" "Chael" "Kevin"
[16,] "Bob" "Ali" "John" "Omar" "Mia" "Paul" "Nathan" "Kevin"
[17,] "John" "Chris" "Karl" "Jean" "Raul" "Paul" "Tim" "Leo"
[18,] "Bob" "Fabien" "Jean" "Raul" "Mia" "Paul" "Sam" "Leo"
[19,] "Bob" "Malik" "Chris" "Jean" "Raul" "Paul" "Tim" "Leo"
[20,] "Nico" "Ali" "Jean" "Jean" "Omar" "Nathan" "Leo" "Kevin"
[21,] "Bob" "John" "Chris" "Jean" "Bill" "Paul" "Chael" "Leo"
[22,] "Fabien" "Jean" "Karl" "Jean" "Omar" "Nathan" "Sam" "Kevin"
[23,] "Nico" "Ali" "John" "Bill" "Mia" "Paul" "Chael" "Nathan"
[24,] "Ali" "Jean" "Karl" "Omar" "Raul" "Paul" "Nathan" "Leo"
[25,] "Malik" "John" "Chris" "Raul" "Mia" "Paul" "Chael" "Nathan"
[26,] "Nico" "John" "Jean" "Jean" "Mia" "Paul" "Chael" "Sam"
[27,] "Bob" "Ali" "John" "Bill" "Raul" "Chael" "Leo" "Kevin"
[28,] "Bob" "Jean" "Karl" "Raul" "Mia" "Paul" "Tom" "Tim"
[29,] "Ali" "John" "Jean" "Omar" "Raul" "Chael" "Nathan" "Sam"
[30,] "Ali" "Fabien" "Jean" "Bill" "Raul" "Nathan" "Sam" "Tim"
```

Puis :

```
z = qnorm(1 - (1 - 0.95) / 2)
icmin = p_w - z * sqrt(var_p_w)
icmax = p_w + z * sqrt(var_p_w)
ST = data.frame(p_w, var_p_w, icmin, icmax)

ST
```

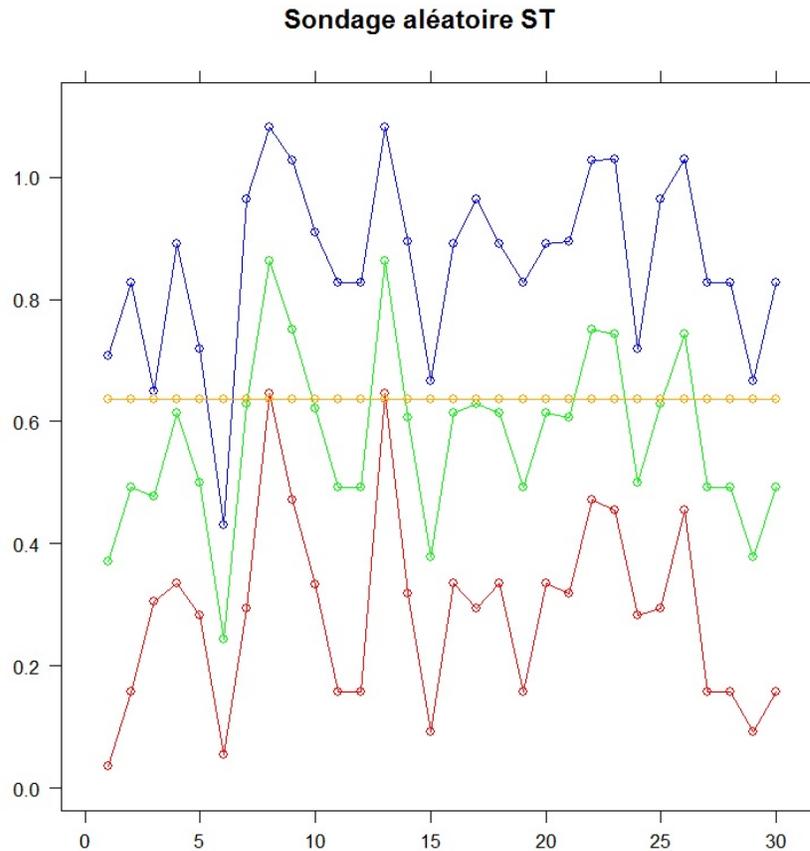
Cela renvoie :

```
      p_w      var_p_w      icmin      icmax
1 0.3712121 0.029327365 0.03556370 0.7068605
2 0.4924242 0.029327365 0.15677582 0.8280727
3 0.4772727 0.007747934 0.30475209 0.6497934
4 0.6136364 0.020144628 0.33545520 0.8918175
5 0.5000000 0.012396694 0.28177674 0.7182233
6 0.2424242 0.009182736 0.05460757 0.4302409
7 0.6287879 0.029327365 0.29313946 0.9644363
8 0.8636364 0.012396694 0.64541311 1.0818596
9 0.7500000 0.020144628 0.47181884 1.0281812
10 0.6212121 0.021579431 0.33329461 0.9091296
11 0.4924242 0.029327365 0.15677582 0.8280727
12 0.4924242 0.029327365 0.15677582 0.8280727
13 0.8636364 0.012396694 0.64541311 1.0818596
14 0.6060606 0.021579431 0.31814309 0.8939781
15 0.3787879 0.021579431 0.09087037 0.6667054
16 0.6136364 0.020144628 0.33545520 0.8918175
17 0.6287879 0.029327365 0.29313946 0.9644363
18 0.6136364 0.020144628 0.33545520 0.8918175
19 0.4924242 0.029327365 0.15677582 0.8280727
20 0.6136364 0.020144628 0.33545520 0.8918175
21 0.6060606 0.021579431 0.31814309 0.8939781
22 0.7500000 0.020144628 0.47181884 1.0281812
23 0.7424242 0.021579431 0.45450673 1.0303418
24 0.5000000 0.012396694 0.28177674 0.7182233
25 0.6287879 0.029327365 0.29313946 0.9644363
26 0.7424242 0.021579431 0.45450673 1.0303418
27 0.4924242 0.029327365 0.15677582 0.8280727
28 0.4924242 0.029327365 0.15677582 0.8280727
29 0.3787879 0.021579431 0.09087037 0.6667054
30 0.4924242 0.029327365 0.15677582 0.8280727
```

2. Quel est le niveau des intervalles de confiance considérés ? Que faut-il changer dans les commandes pour avoir un niveau de confiance de 99% ?
3. Calculer la moyenne-population. Combien y-a t-il d'intervalles de confiance-échantillon qui recouvrent la moyenne-population ?
4. On illustre graphiquement ces résultats numériques en faisant :

```
p_U = rep(mean(Y), 30)
library(lattice)
x = 1:30
xyplot(ST$icmin + ST$icmax + p_w + p_U ~ x, col = c("red", "blue",
"green", "orange"), type = "o", main = "Sondage aléatoire ST",
xlab = " ", ylab = " ")
```

Cela renvoie :



Retrouver graphiquement le résultat de la question précédente.