TP 2 : Échantillonnage aléatoire simple et plans d'expérience

Pour ce deuxième TP, nous allons reprendre les données sur le sondage « qui joue aux jeux vidéos ? ». Ce sera pour nous l'occasion d'illustrer numériquement le théorème central limite, les intervalles de confiances et d'estimer des proportions.

Pour commencer il faut bien entendu importer nos données sous `R`. Elles sont présente dans le fichier `video.data` et vous trouverez le description de ces données en suivant ce lien.

La variable `like` admet plusieurs caractères, pour simplifier les choses on va définir une variable binaire `joue` selon le fait que l'étudiant aime ou non jouer. Pour cela les codes '1', '4' et '5' seront codés comme des '0' et le reste comme des '1'. Faites attention au code '99' qui représente une valeur manquante ! Construisez cette nouvelle variable — n'oubliez pas de la coder comme `facteur`.

Un estimateur de la proportion d'étudiants jouant aux jeux vidéos est

$$\hat{\pi} = \frac{1}{n}\sum_{i=1}^{n} X_i,$$

où X_i est notre variable binaire que nous avons précédemment créée. A l'aide de `R`, estimez la proportion d'étudiants jouant au jeux vidéos. Puisque la taille de la population est $N = 314$, à combien estimeriez vous le nombre d'étudiants jouant au jeux vidéos sur toute la promotion ?

1 Théorème central limite

Le TCL nous dit que lorsque n est assez grand, $\hat{\pi}$ suit approximativement une loi normale. Nous allons le voir numériquement sur notre étude. Idéalement nous aimerions avoir un grand nombre d'échantillons comme celui que nous avons ici et pour chacun de ces échantillons lui associer une estimation $\hat{\pi}$. Ainsi si nous avions K échantillons nous aurions $\hat{\pi}_1, \ldots, \hat{\pi}_K$ estimations et ces estimations devraient suivre une loi Normale.

Avant de commencer à travailler sur nos données nous allons travailler sur des données fictives pour illustrer le TCL. Les fonctions `rexp`, `rt` et `rcauchy` permettent de simuler des réalisations selon une loi exponentielle, de Student et de Cauchy. Faites tourner le code suivant et commenter.

```
par(mfcol = c(2, 3))
taille.echantillon <- c(10, 50, 500)

for (i in 1:3){
  donnees <- matrix(rexp(taille.echantillon[i] * 1000), 1000)
  xbar <- rowMeans(donnees)
  hist(xbar, freq = FALSE, main = paste("n =", taille.echantillon[i]))
  qqnorm(xbar)
  qqline(xbar)
}
```

Recommencez le code précédent mais en simulant selon une loi de Student à 5 degrés de liberté. Quels constats dressez vous ?

Revenons maintenant à nos données. Idéalement nous aimerions faire pareil que pour les données simulées. Malheureusement, nous n'avons qu'un seul échantillon mais à l'aide de l'ordinateur nous allons créer des échantillons « fictifs ». Cette technique est connue sous le nom de

bootstrap. Nous savons qu'il y avait $N = 314$ étudiants dans la promotion et que nous avions un échantillon de taille $n = 91$. La première étape consiste donc à reproduire la population tout entière.

```
pi.hat <- mean(joue == 1)
nb.joue <- round(314 * pi.hat)
pop.sim <- c(rep(1, nb.joue), rep(0, 314 - nb.joue))
joue.sim <- sample(pop.sim, 91)
pi.hat.sim <- mean(joue.sim == 1)
```

Justifiez le code précédant et expliquez ce que représente les objets `pop.sim`, `joue.sim` et `pi.hat.sim`.

Nous avons donc obtenu une nouvelle estimation. Mais ce n'est pas suffisant, il nous en faudrait un grand nombre. Ceci se fait à l'aide du code suivant

```
K <- 500
pi.hat.sim <- rep(NA, K)

for (i in 1:K){
  joue.sim <- sample(pop.sim, 91)
  pi.hat.sim[i] <- mean(joue.sim == 1)
}
```

Comment vérifieriez vous graphiquement que notre estimateur suit approximativement une loi normale ? Quels effets indésirables voit-on apparaître ?

A l'aide de la formule du cours, estimer l'erreur standard de $\hat{\pi}$ et comparez la avec l'écart type de `pi.hat.sim`. Conclusion ?

Interprétez la sortie graphique suivante.

```
se.cours <- sqrt(var(joue) / 91 * (314 - 91) / 314)
hist(pi.hat.sim, freq = FALSE)
lines(seq(0.5, 1, by = 0.005), dnorm(seq(0.5, 1, by = 0.005), pi.hat, se.cours),
      col = 2)
```

2 Intervalle de confiance

Le but de cette partie est de comprendre ce qui signifie le niveau de confiance d'un intervalle de confiance.

A l'aide du cours construisez un intervalle de confiance à 95% pour π. Quelle est la proportion de nos estimations bootstrap appartenant à cet intervalle de confiance ?

Ici nous avons choisi arbitrairement le niveau 95% mais nous aurions pu prendre n'importe quel niveau. Essayer de comprendre et d'interpréter le code suivant.

```
alphas <- seq(0.05, 0.95, by = 0.05)
prop <- rep(NA, length(alphas))
for (i in 1:length(alphas)){
  prob <- 1 - (1 - alphas[i]) / 2
  int.conf.sup <- pi.hat + qnorm(prob) * se.cours
  int.conf.inf <- pi.hat - qnorm(prob) * se.cours

  prop[i] <- mean((pi.hat.sim >= int.conf.inf) & (pi.hat.sim <= int.conf.sup))
}
```

```
plot(alphas, prop, xlim = c(0, 1), ylim = c(0, 1),
     xlab = "Niveau de confiance", ylab = "Proportion")
abline(0, 1)
```

3 Étude approfondie

Nous allons maintenant nous intéresser à la différence entre hommes et femmes. A l'aide de la fonction `table`, lire l'aide de cette fonction, essayer de reproduire le tableau croisé du cours du transparent 89.

Comment procéderiez vous afin de tester si le fait d'aimer jouer dépend du fait d'être un homme ou une femme ? Quel est le nom **exact** de ce test ?

Utilisez la fonction `chisq.test` de manière adéquate pour effectuer le test. Essayer de faire ce test avec l'option `correct = FALSE`. Quelles différences voyez vous ?

Nous avons vu que le test « dont je ne prononcerai pas le nom », mais que vous avez tous reconnu je l'espère, était identique à un z-test sur deux échantillons. La fonction `R` implémentant ce test est `prop.test`. Faites donc un z-test sur deux échantillons et comparez vos résultats avec le test précédent.

A l'aide de l'option `alternative` faites un test pour tester si la proportion de femmes aimant les jeux vidéos est inférieures à celle des hommes.