Département
Licence

MOSE2014 Probabilités et statistiques
Mathématiques TP machine 2
Ph. Thieullen

TP machine II
(Modélisation probabiliste)

Les TP font partie de la note de contrôle continu. Chaque étudiant (ou groupe d'étudiant) rend un unique fichier script par courrier électronique aux enseignants correspondants.

1 Echantillon de loi discrète donnée

Cette partie n'est pas à rendre.

On considère une variable aléatoire X prenant r valeurs $(\xi_1, \dots, \xi_r)$ de probabilités $p = (p_1, \dots, p_r)$. On cherche à construire une fonction sous le même modèle que `rnorm`, qu'on utilisera par la suite sous la forme

```
echantillon <- rdiscrete(n,p,xi)
```

et donnant en sortie un échantillon de taille `n`, de loi `p` et de modalités `xi`.

– Toute les fonctions suivent une syntaxe similaire à ce qui suit

```
rdiscrete <- function(n,p,xi)
{ # début de la fonction
# n = entier
# p = c(p_1,...,p_r) où p_i = nombre
# xi = c(xi_1,...,xi_r) où xi_i = caractère
...
...
return(echantillon) # résultat de la fonction
} # fin de la fonction
```

Il reste bien sûr à compléter la partie ... c'est-à-dire à évaluer `echantillon` en fonction des paramètres `n`, `p` et `xi`. L'idée est de découper l'intervalle $[0, 1]$ selon les proportions p_i, puis de lancer au hasard n points uniformément sur cet intervalle. S'ils tombent dans la i-ième tranche, c'est la modalité ξ_i qui apparaît. La solution est donnée pour l'instant (ne pas écrire les commentaires)

```
# on crée un vecteur générique de taille n et de type xi
echantillon <- vector(mode=mode(xi), length=n)
# on génére un échantillon uniform e_unif[k] de taille n
e_unif <- runif(n)
# on calcule le nombre de modalités r
r <- length(p)
# on crée les points de subdivision de l'intervalle [0,1]
pp <- c(0,cumsum(p))
# on fait tourner une boucle
for(i in 1:r) {
# on crée un vecteur logique de taille n qui contient TRUE
# à l'indice k si e_unif[k] tombe dans la i-ième tranche
ii <- (pp[i] <= e_unif) & (e_unif < pp[i+1])
# on index un vecteur par un vecteur logique : aux indices
# TRUE echantillon prend la modalité xi[i]
echantillon[ii] <- xi[i]
# fin de la boucle
}
```

– Le groupe sanguin se répartie chez les Basques selon des proportions

$$p_O = 56\%,\ p_A = 40\%,\ p_B = 3\%,\ p_{AB} = 1\%$$

légèrement différentes de celles de la polulation française en général de

$$p_O = 43\%,\ p_A = 45\%,\ p_B = 9\%,\ p_{AB} = 3\%$$

– Créer d'abord deux échantillons de taille $n = 1000$ pour chaque groupe sanguin. On donne encore la solution

```
xi <- c("O", "A", "B", "AB")
p_basque <- c(0.56, 0.40, 0.03, 0.01)
p_nation <- c(0.43, 0.45, 0.09, 0.03)
n <- 1000
e_basque <- rdiscrete(n,p_basque,xi)
e_nation <- rdiscrete(n,p_nation,xi)
```

– On factorise chaque échantillon selon les modalité `"O"`, `"A"`, `"B"`, `"AB"` (non rangées dans l'ordre lexicographique) puis on construit une table de contingence (à 1 variable ici)

```
f_basque <- factor(e_basque, levels=xi)
f_nation <- factor(e_nation, levels=xi)
t_basque <- table(f_basque)
t_nation <- table(f_nation)
```

– Afficher des informations pour mieux comprendre

```
print(t_basque)
print(t_nation)
```

– On réunit les deux lignes en un seul tableau et on affiche le diagramm en bâton

```
groupe_sanguin <- rbind(t_basque/n, t_nation/n)
barplot(groupe_sanguin, beside=TRUE, ylim=c(0,0.6),
    legend.text=c("groupe sanguin basque",
    "groupe sanguin national"))
```

On doit trouver la figure 1

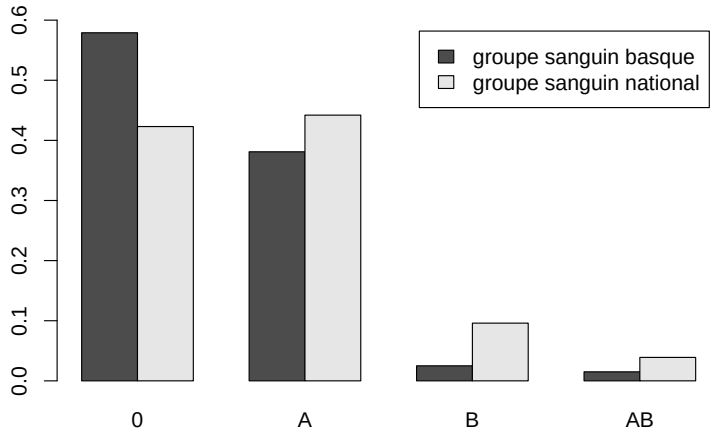

FIGURE 1 – Diagramme en bâton des deux groupes sanguins

2 Le théorème central limite

Partie à rendre. N'oubliez pas de commencer votre script par `# TP2.R` *et les noms de chaque étudiant :* `# noms, prenoms` *formant le binôme ou le trinôme.*

On cherche à montrer graphiquement pourquoi la distribution d'une loi binomiale centrée réduite

$$Z_n = \frac{Y_n - np}{\sqrt{np(1-p)}}, \quad Y_n \overset{loi}{\sim} \mathcal{B}(n,p), \quad \mathbb{E}[Y_n] = np, \ \mathrm{Var}(Y_n) = np(1-p)$$

converge en loi vers la loi normale $\mathcal{N}(0,1)$. On constate d'abord que Z_n prend les valeurs disctrètes

$$z_{n,k} = \frac{k - np}{\sqrt{np(1-p)}}, \quad \text{avec probabilité} \quad \mathbb{P}(Z_n = z_{n,k}) = \binom{n}{k} p^k (1-p)^{n-k}.$$

On constate ensuite que le pas entre deux $z_{n,k}$ successifs est égal à

$$\Delta_n = 1/\sqrt{np(1-p)}.$$

On définit ainsi une densité $f_n(z)$ constante entre deux $z_{n,k}$ successifs par la formule

$$f_n(z)\Delta_n = \mathbb{P}(Z_n = z_{n,k}), \quad \forall\ z \in]z_{n,k-1}, z_{n,k}].$$

– On demande de représenter sur un même graphique les trois distributions $n = 10$, $n = 100$ et $n = 1\ 000$ pour $p = 0.5$ (figure 2) et pour $p = 0.05$ (figure 3). On superposera aussi à chaque fois la distribution de la loi normale centrée réduite.

On remarque que la convergence de Z_n vers $\mathcal{N}(0, 1)$ est beaucoup plus lente lorqsque p est proche de 0 ou de 1 : pour un échantillon de taille $n = 100$ et pour $p = 0.05$, l'erreur commise entre les deux distributions est très importante.

– Indication : Faire simultanément les cas $p = 0.5$ et $p = 0.05$. On découpera l'espace graphiqe en 6 cases avec

```
par(mfrow=c(2,3))
```

Pour tracer l'histogramme de Z_n on tracera deux fois les graphes de $f_{n,k} = \mathbb{P}(Z_n = z_{n,k})/\Delta_n$ en fonction de $z_{n,k}$ selon le modèle suivant

```
# créer l'échantillon z_nk et la distribution f_nk
plot(z_nk, f_nk, type="h", xlim=c(-3,3), ylim=c(0,0.4) )
par(new=TRUE)
plot(z_nk, f_nk, type="s", xlim=c(-3,3), ylim=c(0,0.4) )
```

On superposera aussi la courbe de la densité de la loi normale (aller voir l'aide en ligne de `dnorm`)

```
# créer un échantillon z_norm de [-3,3] de pas 0.01
# calculer la densité f_norm de la loi normale en z_norm
par(new=TRUE)
plot(z_norm, f_norm, type="l", xlim=c(-3,3), ylim=c(0,0.4))
```

Ou pourra compléter les graphiques par de la couleur `col=...`, un titre `main=...`, des noms d'axe `xlab=...`, `ylab=...`

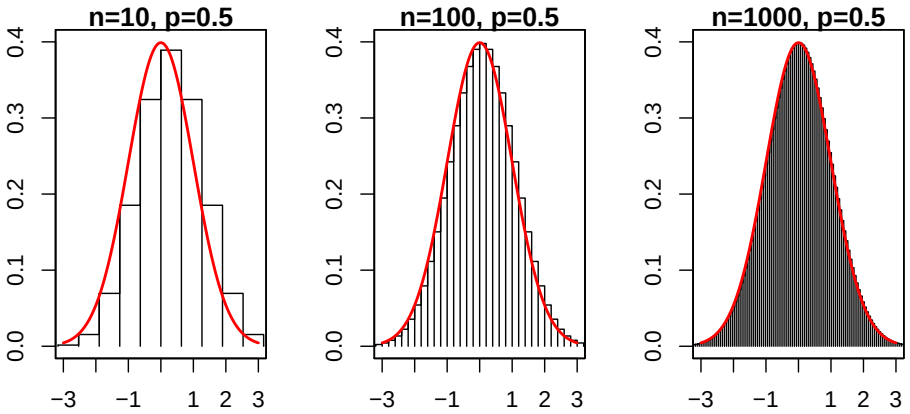

FIGURE 2 – Distribution de la loi binomiale symétrique

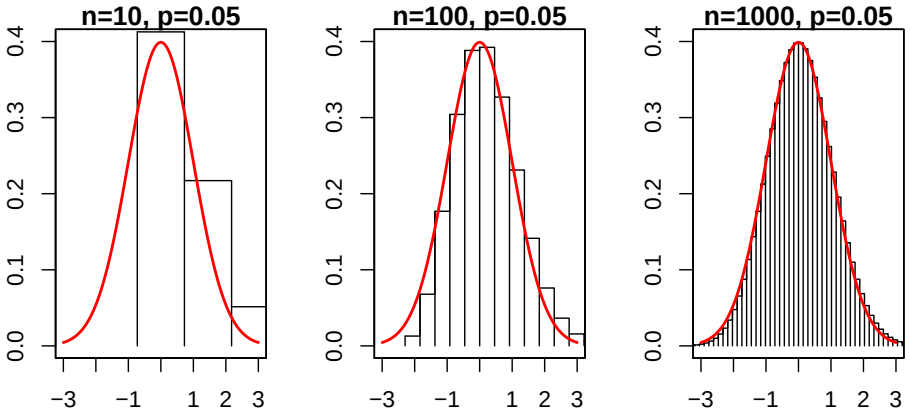

FIGURE 3 – Distribution de la loi binomiale excentrée