

EXEMPLE D'UTILISATION DE LA SIMULATION POUR UNE APPROCHE ALTERNATIVE À LA RÉOLUTION, AU LYCÉE, DU PROBLÈME DES CHAÎNES DE LONGUEUR 6

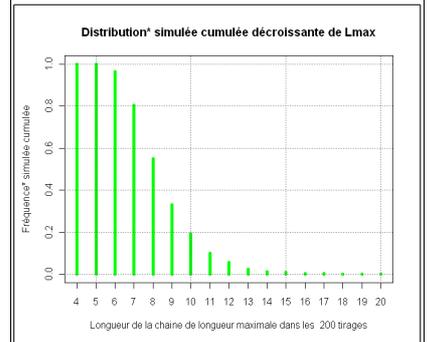
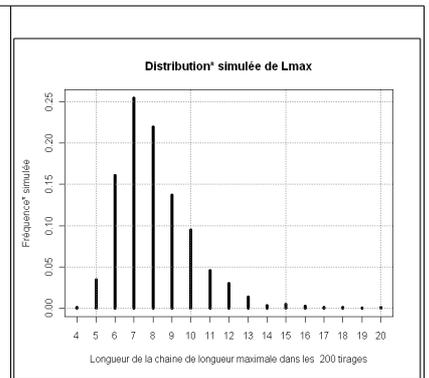
- On lance 200 fois une pièce équilibrée, quelle est la probabilité d'obtenir (au moins) une suite d'au moins 6 piles ou 6 faces consécutifs (on dit une “chaîne” de longueur 6 ou plus).
- C'est un autre exemple de problème dont la résolution n'est mathématiquement pas accessible aux élèves du lycée. La simulation permet d'obtenir une estimation de cette probabilité. L'algorithme est simplifié par l'utilisation de la fonction interne R `rle(...)` qui fournit un tableau des effectifs des longueurs de toutes les chaînes rencontrées.
- Je vais en profiter pour établir la distribution simulée de la variable **LMax** longueur de la chaîne de longueur maximale.
- Pour comprendre comment marche la fonction `rle(...)` de R qui détecte et comptabilise les “chaînes” voici un exemple détaillé en lignes de commandes :

```
# QueFait rle(...) ?
> L = 6 ; piece = c("P", "F") ; tirages = 20
> serieLMax <- NULL ; i <- 1
> #
> (experience <- sample(piece, tirages, replace = T))
[1] "F" "F" "F" "P" "F" "F" "P" "P" "F" "P" "P" "P"
[13] "F" "P" "P" "P" "F" "P" "F" "F"
> (chaines <- rle(experience))
Run Length Encoding
 lengths: int [1:13] 2 1 1 1 3 1 2 1 2 2 ...
 values : chr [1:13] "P" "F" "P" "F" "P" "F" ...
> chaines$length
[1] 2 1 1 1 3 1 2 1 2 2 1 2 1
> serieLMax[i] <- max(chaines$length)
> i
[1] 2
> serieLMax
[1] 3 3
> (FreqLongSupL <- sum(serieLMax >= L) / i)
[1] 0
> #
> i <- i + 1
●●● Après 48 simulations de 20 jets, on obtient :
> i
[1] 48
> serieLMax
[1] 3 3 3 6 3 5 5 4 3 5 4 5 3 4 3 6 4 6 3 5 5 6 3 4
[25] 8 5 4 4 5 3 5 6 3 5 8 3 7 3 5 7 2 4 5 4 4 5 5 5
> (FreqLongSupL <- sum(serieLMax >= L) / i)
[1] 0.1875
```

- La fonction suivante permet d'obtenir une série de valeurs simulées de **LMax**, dans la liste **serieLMax** et d'en faire la description sous forme de tableau des fréquences et de diagrammes en barres.

```
# DISTRIBUTIONS DE LA VARIABLE LONGUEUR DE LA CHAÎNE DE LONGUEUR MAXIMALE
# LES VALEURS DE FRÉQUENCE NULLE FIGURENT SUR LE GRAPHIQUE ET DANS LE TABLEAU
# ON NE TIEN PAS COMPTE DU FAIT QUE Lmax PEUT APPARAÎTRE PLUSIEURS FOIS
chaineDistLMax = fonction(Long = 6, tirages = 200, nbsim = 2000){
  serieLmax <- NULL ; piece = c("Pile", "Face")
  for (i in 1:nbsim) {
    expe <- sample(piece, tirages, replace = TRUE)
    chaines <- rle(expe)
    Lmax <- max(chaines$length)
    serieLmax <- c(serieLmax, Lmax)
  }
  FreqAuMoins1L <- sum(serieLmax >= Long) / nbsim
  tableLmax <- table(serieLmax)
  xmin <- min(as.numeric(names(tableLmax)))
  xmax <- max(as.numeric(names(tableLmax)))
  L <- xmin:xmax ; nbL <- length(L)
  tableLmax <- rep(0, nbL) ; names(tableLmax) <- L
  tableLmax[as.numeric(names(tableLmax)) - xmin + 1] <- tableLmax / nbsim
  freqCumDecr <- cumsum(tableLmax[nbL:1])[nbL:1]
# AFFICHAGE DES TABLEAUX DE FRÉQUENCES
  cat("Une estimation de la probabilité d'au moins 1 chaîne",
      "\nde longueur maximale", Long,
      "\nou plus vaut :\n", FreqAuMoins1L, "\n\n")
  cat("Distribution* simulée de la longueur de la chaîne (runs) de longueur
maximale\n")
  print(tableLmax)
  cat("\nDistribution* simulée cumulée décroissante de la longueur",
      "\nde la chaîne (runs) de longueur maximale\n")
  print(freqCumDecr)
# DIAG. EN BARRES DE LA DISTRIBUTION ET DE LA CUMULÉE DÉCROISSANTE, DE Lmax
  plot(L, tableLmax,
       type = "h", xaxp = c(xmin, xmax, xmax - xmin), lwd = 5,
       main = "Distribution* simulée de Lmax",
       xlab = paste("Longueur de la chaîne de longueur maximale dans les ",
tirages, "tirages"),
       ylab = "Fréquence* simulée")
  grid(col = "grey50")
  plot(L, freqCumDecr,
       type = "h", xaxp = c(xmin, xmax, xmax - xmin),
       lwd = 5, col = "green",
       main = "Distribution* simulée cumulée décroissante de Lmax",
       xlab = paste("Longueur de la chaîne de longueur maximale dans les ",
tirages, "tirages"),
       ylab = "Fréquence* simulée cumulée")
  grid(col = "grey50")
}
> chaineDistLMax()
Une estimation de la probabilité d'au moins 1 chaîne
de longueur maximale 6 ou plus vaut :
 0.965

Distribution* simulée de la longueur de la chaîne (runs) de longueur maximale
 4  5  6  7  8  9  10  11  12  13  14
0.0005 0.0345 0.1605 0.2545 0.2195 0.1365 0.0945 0.0450 0.0295 0.0135 0.0030
 15 16 17 18 19 20
0.0045 0.0020 0.0005 0.0010 0.0000 0.0005
...
```



- On remarque que dans ces 2000 simulations de 200 lancers il y a 17 simulations dans lesquelles la chaîne de longueur maximale est de taille supérieure ou égale à 15, il y en a même une de longueur 20 ! Un bon exemple contre-intuitif !
- Dans cet autre exemple, la simulation rend sa résolution accessible à des élèves de lycée. L'algorithme reste simple si l'on utilise la fonction **rle(...)**.
- L'algorithme suivant permet de faire la représentation graphique d'une simulation de 200 (= tirages) lancers dans laquelle les chaînes de longueur 6 ou plus sont colorées en rouge.

- Les lignes de commandes suivantes permettent de représenter, dans une série de 200 jets, les piles en vert et les faces en bleu et les chaînes de longueur 6 ou plus en rouge, de façon à les repérer facilement.

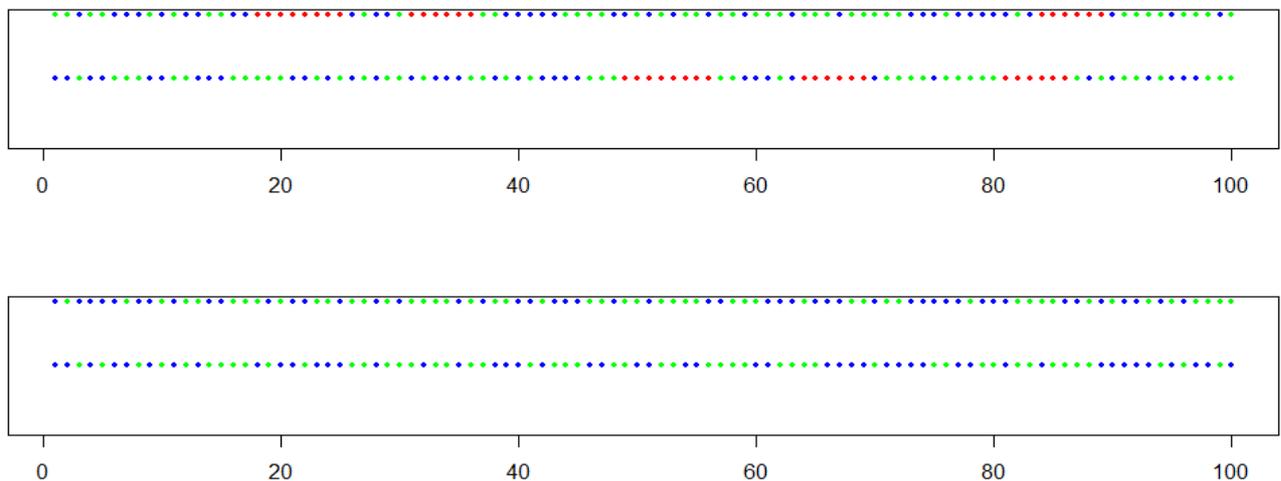
```
# REPRÉSENTATION GRAPHIQUES DES SIMULATIONS ET DES CHAÎNES DE LONGUEUR 6 AU MOINS
# SIMULATION DE 200 TIRAGES (un nombre pair pour étager la représentation graphique)
# PILE = 1, FACE = 2 PAR EXEMPLE
tirages <- 200 ; m <- tirages / 2
seriejets <- sample(c(1, 2), tirages, replace = TRUE)

# LISTE DE COULEURS EN CORRESPONDANCE AVEC LES PILES ET FACES
# PILE EN VERT, FACE EN BLEU
seriecouleurs <- c("green", "blue")[seriejets]

# LA FONCTION rle(...) FOURNIT LA LONGUEUR DE TOUTES LES CHAÎNES DE LA SÉRIE DE JETS
chaines <- rle(seriejets)

# ON IDENTIFIE DES CHAÎNES DE LONGUEUR 6 OU PLUS ET ON LES COLORIE EN ROUGE
# LE RESTE DE LA SÉRIE GARDE LES COULEURS INITIALES DÉFINIES DANS LA LISTE seriecouleurs
# LES CHAÎNES DE LONGUEUR 6 OU PLUS SONT REPÉRÉES PAR LEUR INDICE DE DÉBUT indDeb
# LEUR INDICE DE FIN indFin
L <- 6
nbchaines <- length(chaines$lengths)
seriecouL <- NULL
indices <- cumsum(chaines$lengths)
for (i in 1:(nbchaines)) {
  if (i == 1) {
    indDeb <- 1 ; indFin <- chaines$lengths[1]
  } else {
    indDeb <- indices[i - 1] + 1 ; indFin <- indices[i]
  }
  if (chaines$lengths[i] >= L) {
    seriecouL <- c(seriecouL, rep("red", chaines$lengths[i]))
  } else {
    seriecouL <- c(seriecouL, seriecouleurs[indDeb:indFin])
  }
}
# LES REPRÉSENTATIONS GRAPHIQUES
# AVEC LES ROUGES
plot(1:m, rep(2, m), ylim = c(0, 2), pch = 19, cex = .5, col = seriecouL[1:m], yaxt = "n")
  points(1:m, rep(1, m), pch = 19, cex = .5, col = seriecouL[(m + 1):tirages])

# SANS LES ROUGES
plot(1:m, rep(2, m), ylim = c(0, 2), pch = 19, cex = .5, col = seriecouleurs[1:m], yaxt = "n")
  points(1:m, rep(1, m), pch = 19, cex = .5, col = seriecouleurs[(m + 1):tirages])
```



- On peut remarque qu'on observe très fréquemment des chaînes de longueur 6 ou plus, les simulations n'en contenant pas sont donc plutôt rares. Il arrive aussi assez souvent que l'on trouve plusieurs chaînes de longueur 6 ou plus dans un simulation de 200 jets.