testrunif

(gH)

15 mars 2016

## Méthode utilisée

On génère de chacun . Toutes les valeurs générées sont rangées dans une matrice de résultats, à raison d'un essai par ligne. Pour chaque essai, on comptabilise le nombre de valeurs avant et après 0.5 et on effectue un test binomial pour voir si la différence est significative au seuil . En fin de fonction, on renvoie le nombre de fois où la différence est significative.

## Résultats des tests

source("testrunif.r",encoding="latin1")   
nbs <- testRunif(10,5000)

##   
## extrait des tirages aléatoires  
## [,1] [,2] [,3]  
## essai001 0.01856805 0.2027442 0.18172115  
## essai002 0.84709111 0.4922841 0.11878043  
## essai003 0.98724561 0.2462825 0.01921215  
## essai004 0.97189860 0.7024336 0.33985950  
## essai005 0.12464824 0.9634934 0.02455024  
##   
## extrait des résultats  
## nbVal avant 0.5 nbVal Après 0.5 p-value code sig.  
## 1 3000 2000 0.0000000 \*\*\*  
## 2 2000 3000 0.0000000 \*\*\*  
## 3 3500 1500 0.0000000 \*\*\*  
## 4 1500 3500 0.0000000 \*\*\*  
## 5 3000 2000 0.0000000 \*\*\*

## Annexe : code de la fonction testrunif()

## [1] "testRunif <- function(nbEssais=10,nbValeurs=50) {"   
## [2] ""   
## [3] " ## 1. g\xe9n\xe9ration des valeurs al\xe9atoires"   
## [4] ""   
## [5] " # remplissage de la matrice des r\xe9sultats"   
## [6] ""   
## [7] " matRes <- matrix(nrow=nbEssais,ncol=nbValeurs)"   
## [8] " row.names(matRes) <- paste(\"essai\",sprintf(\"%03d\",1:nbEssais),sep=\"\")"   
## [9] ""   
## [10] " for (idl in (1:nbEssais)) {"   
## [11] " matRes[idl,] <- runif(nbEssais)"   
## [12] " } # fin pour idl"   
## [13] ""   
## [14] " cat(\"\\nextrait des tirages al\xe9atoires\\n\")"   
## [15] " print(matRes[(1:5),(1:3)])"   
## [16] ""   
## [17] " ## 2. analyse des valeurs al\xe9atoires"   
## [18] ""   
## [19] " # si la r\xe9partition est uniforme, on doit avoir"   
## [20] " # autant de valeurs avant 0.5 qu'\xe0pr\xe8s 0.5"   
## [21] ""   
## [22] " repVal <- as.data.frame(matrix(nrow=nbEssais,ncol=4))"   
## [23] " names(repVal) <- c(\"nbVal avant 0.5\",\"nbVal Apr\xe8s 0.5\",\"p-value\",\"code sig.\")"  
## [24] ""   
## [25] " for (idl in (1:nbEssais)) {"   
## [26] " repVal[idl,1] <- sum( matRes[idl,] < 0.5 ) # valeurs avant"   
## [27] " repVal[idl,2] <- sum( matRes[idl,] > 0.5 ) # valeurs apr\xe8s"   
## [28] " # la diff\xe9rence est-elle significative ?"   
## [29] " pvalue <- binom.test( as.numeric(repVal[idl,(1:2)]), p=1/2 )$p.value"   
## [30] " repVal[idl,3] <- sprintf(\"%13.7f\",pvalue)"   
## [31] " repVal[idl,4] <- as.sigcode(pvalue)"   
## [32] " } # fin pour idl"   
## [33] ""   
## [34] " # affichage (extrait)"   
## [35] ""   
## [36] " cat(\"\\nextrait des r\xe9sultats\\n\")"   
## [37] " print(repVal[(1:5),])"   
## [38] ""   
## [39] ""   
## [40] " # autre fa\xe7on de le v\xe9rifier : tester si la m\xe9diane"   
## [41] " # est \xe9gale \xe0 0.5"   
## [42] ""   
## [43] " # autre fa\xe7on de le v\xe9rifier : tester si les m\xe9dianes"   
## [44] " # sont \xe9gales"   
## [45] ""   
## [46] " # on renvoie le nombre de fois o\xf9 la diff\xe9rence est significative"   
## [47] ""   
## [48] " nbSig <- sum(repVal[,3]<0.05)"   
## [49] " return( nbSig)"   
## [50] ""   
## [51] "} # fin de fonction testRunif"