Atelier CIMPA Dakar 2011

Introduction à R

1 Operations de bases

Pour se familiariser avec R, cliquer sur `http://www.grappa.univ-lille3.fr/~ppreux/ensg/miashs/fouilleDeDonneesI/tp/introduction-a-R/` et effectuer le TP d'introduction à R. S'arrêter avant la partie *data frame*.

Quand vous ne savez pas ce que fait une fonction, pensez à utiliser l'aide de R en tapant par exemple *?runif* pour avoir l'aide de *runif*. Une introduction très complète à R est disponible sur `http://cran.r-project.org/`.

2 Ecrire une fonction avec R

Dans un fichier *pairimpair.R*, recopier le code suivant

```
myf<-function(n)
{
for (i in 1:n)
{
##teste si le nombre est pair ou impair
if(i%%2==0)
##Le caractère \n correspond à 1 saut de ligne
cat(i, "est un nombre pair\n")
else
cat(i, "est un nombre impair\n")
}
return (n)
}
```

Pour utiliser cette fonction, taper les lignes suivantes :

```
>source("pairimpair.R")

>myf(1)

>myf(5)

>myf(10)

>a<-myf(10)

>a
```

Vous êtes désormais parés pour écrire vos propres fonctions. Pour chaque exercice, écrivez vos fonctions et vos lignes de commande dans un fichier dédié à cet exercice.

2.1 Simulation d'une variable aléatoire par inversion

L'algorithme d'inversion permet de simuler une variable aléatoire X de fonction de répartition F à partir d'une variable aléatoire U de loi uniforme sur $[0, 1]$. Il suffit de poser

$$X = F^{-1}(U).$$

Coder une fonction *myexp* de sorte à pouvoir simuler n variables aléatoires de loi exponentielle de paramètre λ. Pour vérifier que votre fonction est correcte, comparer vos simulations avec les simulation obtenues avec la fonction *rexp*. Vous pouvez utiliser la fonction *qqplot* pour comparer les simulations obtenues avec les deux fonctions.

2.2 Processus de Poisson

Commencer par écrire une fonction dont l'argument est un temps t et qui simule les temps d'occurrence entre $[0, t]$ d'un processus de Poisson de taux $\lambda = 1$. On rappelle que les temps inter-occurence sont distribués suivant une loi exponentielle de paramètre λ. *Idée :* On pourra se servir de l'expression *while(condition)*$\{$*statement*$\}$. Simuler le processus de Poisson 1000 fois et représenter l'histogramme du nombre d'occurrence N_{10} d'événements entre $[0, 10]$. Montrer que N_{10} suit une loi bien connue en utilisant la fonction *qqplot*.

2.3 Méthode de Monte Carlo

On va chercher à estimer le nombre π avec une méthode de Monte Carlo. Pour ce faire, écrire un fonction qui simule n points dans le carré centré à l'origine dont les cotés sont de longueurs 2. *Idée :* Simuler les coordonnées des points avec la commande *runif.* Compter la proportion de points qui se trouve dans le cercle centré à l'origine et de rayon 1 et en déduire une estimation de π.

2.4 Simulation d'un mouvement Brownien

Ecrire une fonction dont l'argument est un vecteur $t = (t_1, \dots, t_n)$ et qui retourne un vecteur de même longueur qui est la réalisation d'un mouvement brownien W_t aux temps $(t_1, \dots, t_n)$. On rappelle qu'en loi

$$W_{t_i} = W_{t_{i-1}} + X,\ 1 < i \leq n,$$

ou X est une gaussienne centrée de variance $t_i - t_{i-1}$. Représenter avec la commande *plot* la trajectoire d'un mouvement brownien ainsi simulé.