
T.P. de Méthodes de Monté-Carlo

1. Télécharger la version du logiciel R qui convient votre système d'exploitation à partir de la page : <https://www.r-project.org/>
2. Lancer l'installation en suivant les instructions.
3. Une fois l'installation terminée, ouvrir le logiciel R.
4. Télécharger et installer quelques packages.

Pour installer un package, on tape sur la console :

```
>install.packages("dplyr") #ce package est utilisé dans la manipulation des données.
```

Une fois un package installé, il faudra le charger en mémoire à chaque fois une nouvelle session est ouverte :

```
>library(dplyr) #sans guillemets!
```

Une autre manière pour installer un package c'est d'aller dans le menu packages, sélectionner installer le(s) package(s). Dans la fenêtre qui s'ouvre, choisir dans la liste :

```
Rcmdr #pour rajouter des menus cliquables.
```

Exemples d'autres packages importants et utilisés en Statistique :

```
lattice      : pour obtenir des graphiques avancés.  
Hmisc       : présentation avancée des résumés statistiques.  
FactoMineR  : analyses statistiques.
```

5. Ouvrir ou créer un nouveau **script** et (à la fin) le sauvegarder.

Au lieu d'utiliser la console de R, il vaut mieux de travailler sur un script, où les commandes seront stockées et on pourra facilement l'ouvrir, éditer et exécuter entièrement ou en partie si on veut.

Exercice 1 :

Que font ces lignes de commande? Indiquer ce que R vous retourne :

```
>qnorm(0,975)
```

```
>dnorm(0)
```

```
>pnorm(1,96)
```

```
>rnorm(20)
```

```
>rnorm(10,mean=5,sd=0,5)
```

```
>x=seq(-3,3,0,1); pdf=dnorm(x); plot(x,pdf,type="l");
```

```
>runif(3)
```

```
>rt(5,10)
```

Exercice 2 :

1. Générer 100 nombres aléatoires qui suivent une loi uniforme $(0, 1)$ (prendre 25 pour germe ou graine).
2. Représenter-les graphiquement et de plusieurs façons.
3. Générer, par la méthode d'inversion, sous R, les lois : Uniforme, Exponentielle et Weibull.
4. Représenter des densités de la loi de Cauchy pour différents paramètres par différentes couleur sur le même repère.

Exercice 3 :

1. Utiliser `sample()` pour générer $n = 10$ valeurs aléatoires de la loi de probabilité suivante :

x_i	1	2	3	4	5	6	7
p_i	0,1	0,4	0,3	0,1	0,05	0,03	0,02

2. Examiner que $\bar{X} \sim \mathcal{N}(\mu, \frac{\sigma^2}{n})$ approximativement, lorsque n est grand pour la loi de la question précédente ($n = 10, n = 20, n = 30$).

Exercice 4 :

Construire un vecteur `simul1` composé de 1000 éléments dont les valeurs sont des réalisations d'une variable $X \sim \mathcal{N}(15, 3)$.

Faire l'histogramme des fréquences de `simul1` et supposer la densité à X . Faire le `boxplot` de `simul1`.

Exercice 5 :

1. On veut générer $n = 1000$ variables exponentielles de paramètre 1. Utiliser la méthode d'inversion. Superposer la densité théorique et un estimateur de la densité.
2. Même question avec les fonctions dédiées `rexp` et `dexp` sous R.
3. Interpréter les commandes suivantes :

```
>t0<-proc.time()
>E<-runif(10^6)
>proc.time()-t0
```

Comparer les durées respectives des méthodes des questions précédentes pour n très grand.

Références :

1. Vincent Goulet : "Introduction à la programmation en R" qu'on peut télécharger à partir du site : https://cran.r-project.org/doc/contrib/Goulet_introduction_programmation_R.pdf
2. W. John Braun & Duncan J. Murdoch : "A first course in statistical programming with R", Cambridge University Press, July 2016.
3. La feuille de pompe sur <https://cran.r-project.org/doc/contrib/Short-refcard.pdf>