TP 11. Vecteurs gaussiens

Description :
Manipulation de vecteurs gaussiens dans différentes situations.

Objectifs :

- Savoir simuler des lois normales
- Savoir simuler des vecteurs gaussiens

1 Simulation de la loi normale centrée réduite

On veut simuler des variables aléatoires indépendantes suivant la loi gaussienne centrée réduite $N(0,1)$. Il est possible de trouver une méthode de décomposition, adaptée non seulement à la densité de la loi $N(0,1)$ mais aussi aux qualités du générateur et du compilateur, qui soit plus rapide que celles qui suivent. Les méthodes que nous donnons ici sont faciles à programmer.

1.1 Algorithme polaire

Il repose sur le résultat suivant.

Théorème 1.1 *Soit (X,Y) un couple de variables aléatoires, de loi uniforme sur le disque unité*

$$D = \{(x,y)\,;\, x^2+y^2<1\}\,.$$

Écrivons ces variables aléatoires en coordonnées polaires :

$$X = R\cos(\Theta)\,;\, Y = R\sin(\Theta)\,.$$

On pose:

$$R' = \sqrt{-4\log(R)}\,.$$

Alors

$$U = R'\cos(\Theta)\; et V = R'\sin(\Theta)\,.$$

sont deux variables aléatoires indépendantes, de même loi $N(0,1)$.

La preuve de ce résultat repose sur le changement de variable en polaire qui permet d'obtenir que les variables R et Θ sont indépendantes. Il permet aussi d'obtenir que R suit une loi bêta de paramètre 2 et 1 (aussi appelée loi triangulaire sur $[0,1]$) et que Θ suit une loi uniforme sur $[0,2\pi]$. On en déduit ensuite aisément la densité de la loi du couple (U,V) par un second changement de variable en polaire.

Interprétation géométrique : on pourra faire un dessin pour représenter la position du vecteur (U,V) par rapport à celle du point initial (X,Y). Ces deux points forment le même angle avec l'axe des abcisses. Seule leur norme diffère.

On déduit de ce résultat et de son interprétration géométrique, l'algorithme polaire :

```
Repeter
X<-2*runif(1,0,1)-1
Y<-2*runif(1,0,1)-1
R2<-X*X+Y*Y
Jusqu'a (R2<1) # (X,Y) est de loi uniforme sur le disque unite (methode de rejet)
Z<- Sqrt(-2*log(R2)/R2) #facteur mutiplicatif modifiant uniquement la norme de (X,Y)
U<-Z*X
V<-Z*Y # U et V sont independants de loi N(0,1)
```

Utiliser cet algorithme pour simuler un grand nombre de réalisations indépendantes de la loi gaussienne centrée réduite. Vérifier le bon fonctionnement de votre algorithme en traçant les deux graphiques suivant. Pour commencer, sur le même graphique, l'histogramme de vos points avec la densité de la loi gaussienne centrée réduite. Ensuite, sur un autre graphique, la fonction de répartition empirique de vos points avec la fonction de répartition de la loi gaussienne centrée réduite.

1.2 Algorithme de Box-Muller

Fréquemment proposé dans les manuels, cet algorithme est basé sur la même méthodologie polaire, mais programmée différemment. Il repose sur la proposition suivante.

Proposition 1.1 *Soient U_1 et U_2 deux variables aléatoires indépendantes de loi uniforme sur $[0,1]$. On pose :*

$$\begin{aligned} X_1 &= \sqrt{-2\log U_1}\cos(2\pi U_2), \\ X_2 &= \sqrt{-2\log U_1}\sin(2\pi U_2). \end{aligned}$$

Alors X_1 et X_2 sont indépendantes et de même loi $\mathcal{N}(0,1)$.

La preuve directe de cette proposition repose sur le fait que la variable aléatoire $-2\log U_1$ suit la loi exponentielle de paramètre $1/2$ et est indépendante de la variable aléatoire $2\pi U_2$ qui, elle, suit une loi uniforme sur $[0,2\pi]$. Pour prouver ce résultat, on peut également calquer la preuve du théorème précédent en remarquant que la variable aléatoire $\sqrt{-2\log U_1}$ a la même densité que R'.

Utiliser cet algorithme pour simuler un grand nombre de réalisations indépendantes de la loi gaussienne centrée réduite. Vérifier le bon fonctionnement de votre algorithme en traçant les deux graphiques suivant. Pour commencer, sur le même graphique, l'histogramme de vos points avec la densité de la loi gaussienne centrée réduite. Ensuite, sur un autre graphique, la fonction de répartition empirique de vos points avec la fonction de répartition de la loi gaussienne centrée réduite.

1.3 Comparaison des 2 algorithmes

Selon les compilateurs, un des deux algorithmes (polaire ou de Box-Muller) est le plus rapide.

2 Simulation de lois normales multidimensionnelles

De la simulation de la loi $N(0,1)$, on déduit celle de la loi normale d'espérance μ et de variance σ^2 quelconques par une transformation affine. Si X suit la loi $N(0,1)$, alors $Y = \mu + \sigma X$ suit la loi

$N(\mu, \sigma^2)$. La situation est analogue en dimension k quelconque. Tout d'abord, si $X_1, \cdots, X_k$ sont indépendantes et de même loi $N(0,1)$, alors le vecteur $(X_1, \cdots, X_k)$ suit la loi normale dans R^k, d'esperance nulle et de matrice de covariance identité : $N_k(0, I)$. On en déduit la simulation d'une loi normale d-dimensionnelle quelconque par la proposition suivante :

Proposition 2.1 *Soit μ un vecteur de R^k et Σ une matrice carrée de taille k symétrique positive. Soit $X = (X_1, \cdots, X_k)$ un vecteur aléatoire de loi $N_k(0, I)$ dans R^k. Soit A une matrice carrée d'ordre k telle que $AA^T = \Sigma$. Alors le vecteur $Y = AX + \mu$ est un vecteur gaussien de moyenne μ et de matrice de covariance Σ.*

La preuve est une conséquence directe de la définition d'un vecteur gaussien et de la matrice de covariance d'un vecteur aléatoire.

D'un point de vue pratique, il faut donc trouver une matrice A telle que $AA^T = \Sigma$. C'est un problème très classique. Une des réponses est implémentée dans la plupart des librairies d'algèbre linéaire. C'est la *décomposition de Cholesky*. Dans le cas où Σ est définie positive, cette méthode calcule colonne par colonne une matrice A, triangulaire inférieure telle que $AA^T = \Sigma$.

Dans le cas particulier où $k = 2$, la *décomposition de Cholesky* est explicite. En effet, soit $Y = (Y_1, Y_2)$ un vecteur gaussien de R^2. On note $\mu = (\mu_1, \mu_2)$ son esperance et

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho \\ \sigma_1\sigma_2\rho & \sigma_2^2 \end{pmatrix}$$

sa matrice de variance-covariance. Dans cette notation, σ_1^2 représente la variance de Y_1 tandis que ρ est la corrélation entre Y_1 et Y_2 : $\rho = \mathrm{Corr}(Y_1, Y_2) = \frac{\mathrm{Cov}(Y_1, Y_2)}{\sigma_1\sigma_2}$. On a donc $|\rho| \leq 1$.

Si on pose

$$A = \begin{pmatrix} \sigma_1 & 0 \\ \sigma_2\rho & \sigma_2\sqrt{1-\rho^2} \end{pmatrix}$$

alors on vérifie aisément que A est triangulaire inférieure et que $AA^T = \Sigma$.

En appliquant la proposition précédente, on constate qu'on peut simuler des réalisations indépendantes de Y en utilisant la représentation en loi suivante dans laquelle X_1 et X_2 sont deux variables aléatoires indépendantes de loi $N(0,1)$:

$$Y_1 = \mu_1 + \sigma_1 X_1 \,.$$
$$Y_2 = \mu_2 + \sigma_2 \left(\rho X_1 + \sqrt{1-\rho^2} X_2\right) \,.$$

Utilisons cette procédure pour simuler $n = 10000$ réalisations de Y suivant $N_2(0, \Sigma)$ où

$$\Sigma = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \,.$$

Cela signifie qu'on se place dans le cas où $\mu = 0$ et $\sigma_1 = \sigma_2 = 1$. On prendra soin de travailler vectoriellement (éviter autant que possible les boucles **for**) pour améliorer les temps de calcul. La procédure est écrite dans le fichier `SimulationVecteurGaussien2D.r`. On l'executera par la commande

```
> source('SimulationVecteurGaussien2D.r')
```

Dans cette fonction, on pourra faire varier la valeur de ρ. Quelle difference observer vous entre les representations graphique obtenues pour $\rho = 0.75$ et $\rho = -0.75$? Comment pouvez-vous interpreter physiquement cette observation ? (pensez a l'interprétation physique de la corrélation). Que se passe-t-il pour $\rho = 0$? Observer ce qui se passe pour $\rho = 1$ et $\rho = -1$: le fait que toutes les réalisations soient sur une droite implique que le vecteur aléatoire simulé ne possede pas de densité : la matrice Σ est alors non-inversible.

On pourra également vérifier que la matrice A proposée ci-dessus est bien la décomposition de Cholesky de Σ par les commandes suivantes :

```
> rho=0.75
> Sigma=rbind(c(1,rho),c(rho,1))
> A=rbind(c(1,0),c(rho,sqrt(1-rho^2)))
> t(A)
> chol(Sigma)
```

Que se passe-t-il si on prend $\rho = \pm 1$?

```
> rho=1
> Sigma=rbind(c(1,rho),c(rho,1))
> chol(Sigma)
```

Pourtant, on a bien $AA^T = \Sigma$:

```
> rho=1
> Sigma=rbind(c(1,rho),c(rho,1))
> A=rbind(c(1,0),c(rho,sqrt(1-rho^2)))
> A%*%t(A)
> Sigma
```

La fonction `chol` de R ne fonctionne que pour des matrices symétrique *définie*-positive.

3 Une situation concrète issue de "la guerre des boutons" : à vous de jouer.

3.1 Il était une fois...

Dans le centre de la France peu après la deuxime guerre mondiale, deux villages respiraient la tranquilité... D'un coté, il y avait Longeverne et de l'autre Velrans. Comme tous les jours, les enfants de chaque village se rendaient à leur école respective. Or, un matin où Grand Gibus et P'tit Gibus allaient à l'cole de leur village Longeverne, les enfants de Velrans les ont maltraités. C'est ainsi que les enfants de Longeverne dirigés par Lebrac et ceux de Velrans dirigés par l'Aztec organisèrent une grande bataille où il tait convenu que les prisonniers subiraient le lourd tribu de se voir oter la totalité de leurs boutons.

3.2 Le problème de P'tit Gibus

Afin de récupérer le plus grand nombre de boutons possible, P'tit Gibus s'est confectionné un sac en toile. Malheureusement, il n'a pas eu tout à fait assez de tissu pour le finir entièrement. Enfin, ce n'est pas un gros problème car seulement les boutons de diamètre inférieur à 5mm tombent du sac. On suppose que le diamètre des boutons suit une loi normale de moyenne 8mm et d'écart-type 2mm.

3.2.1 Simulation

- Simuler $n = 10000$ valeurs indépendantes de même loi $\mathcal{N}(8, 4)$. On utilisera ici l'algorithme de Box-Muller.
- Calculer la moyenne empirique de ces n observations. Commenter.
- Calculer la variance empirique de ces n observations. Commenter.
- Représenter à l'aide d'un histogramme des fréquences la distribution empirique de ces n observations.

3.2.2 Comment aider P'tit Gibus

Lors de la dernière bataille, P'tit Gibus a récupérer 40 boutons dans son sac. Déterminer la loi de la variable aléatoire comptant le nombre de boutons perdus.

Quelle est la probabilité que P'tit Gibus perde plus de 5 boutons ? Vérifier par la simulation.

Pensez-vous que le sac de P'tit Gibus soit efficace ?

3.3 Le bilan des gains...

On note

- Y_1 le poids des boutons de bretelles détenus par la bande de Lebrac à la fin d'une bataille (en dag).(positif si a correspond un gain, négatif si ça correspond une perte)
- Y_2 le poids des boutons de pantalons détenus par la bande de Lebrac à la fin d'une bataille (en dag).(positif si a correspond un gain, négatif si ça correspond une perte)

On suppose que (Y_1, Y_2) est un couple gaussien centré et de matrice de covariance :

$$\Sigma = \begin{pmatrix} 2 & 3 \\ 3 & 5 \end{pmatrix}.$$

On sait que les deux bandes concluent à la fin des batailles des arrangements par rapport par exemple aux conditions de libération des prisonniers qui sont modélisés par la combinaison suivante:

$$\begin{aligned} X_1 &= 2Y_1 - Y_2, \\ X_2 &= Y_2 - Y_1. \end{aligned}$$

1. En utilisant les résultats de la section 2 sur les vecteurs gaussiens bi-dimensionnels, montrer que X_1 et X_2 sont indépendantes et de même loi $\mathcal{N}(0,1)$.

2. Simuler n couples indépendants et de même loi que (Y_1, Y_2). Vérifier votre simulation.

3. Lebrac n'est pas très assidu à l'école et son père menace de l'envoyer en pension si la bande gagne moins de 0.4 dag de boutons de bretelles, et si elle perd plus de 3 dag de boutons de pantalons. Calculer à l'aide d'une simulation la probabilité que Lebrac soit envoyé en pension à la prochaine bataille. De façon facultative, on pourra calculer la valeur théorique de cette probabilité et comparer.

4. En fait, après la mise en place d'une tactique particulière, les lois des variables Y_1 et Y_2 ont été un peu modifiée. (Y_1, Y_2) est maintenant un couple gaussien de moyenne $(-0.4, 2)$ et de matrice de covariance Σ. Simuler n couples gaussiens suivant cette loi. Cette tactique est-elle avantageuse pour Lebrac ? Quelle sorte de boutons (bretelles ou pantalons) conseillerez-vous à Lebrac de prendre ou de ne pas perdre en priorité ?

4 Vers les statistiques

4.1 Intuition sur le signe de la corrélation

Simuler un échantillon x de taille $n = 10000$ la loi normale $N(1,4)$ et un échantillon y de la loi normale $N(0,1)$ par l'algorithme de Box-Muller. Calculer $z = x + 2y$. Représenter dans le plan les couples de points (x_i, z_i). La corrélation entre x et z est-elle positive ou negative ? Vérifier cela par un calcul théorique.

4.2 Loi du chi-deux et illustration du théorème de Cochran

1. Pour $m = 2$, puis $m = 4$, simuler m échantillons de taille $n = 10000$ de la loi normale $N(0,1)$. Ces échantillons seront placés dans une matrice X à n lignes et m colonnes. Calculer l'échantillon y obtenu en calculant la somme des carrés de chaque ligne de cette matrice. Représenter un histogramme des valeurs de y. Superposer sur le même graphique la densité de la loi du chi-deux à m degrés de liberté (fonction dchisq). Représenter la fonction de répartition empirique de y. Superposer sur le même graphique la fonction de répartition de la loi du chi-deux à m degrés de liberté (fonction pchisq). On pourra utiliser les commandes suivantes :

```
n<-10000                                # dimension du vecteur
m<-2
X <- matrix(rnorm(m*n),n,m)             # matrice d'echantillons en colonnes
y <- rowSums(X^2)                       # echantillon des normes carrees
hist(y,probability=T,col="blue",main ="Histogramme de y")
curve(dchisq(x,m), col="red", add=T)    # Histogramme et densite theorique de y
```

puis

```
ytri <- sort(y)                          # tri par ordre croissant
ord <- (1:n)/n                           # ordonnees
plot(ytri, ord, type="s", col="blue")
curve(pchisq(x,m), col="red", add=T)
            # fonction de repartition empirique et theorique de y
```

2. Calculer la moyenne empiriques M de chacune des lignes de X, retrancher M a chacune des colonnes de X et calculer l'échantillon z obtenu en prenant la somme des carrés de chaque ligne. Représenter un histogramme des valeurs de z. Superposer sur le même graphique la densité de la loi du chi-deux à $m-1$ degrés de liberté. Représenter la fonction de répartition empirique de z. Superposer sur le même graphique la fonction de répartition de la loi du chi-deux à $m-1$ degrés de liberté.

```
M <- rowMeans(X)                 # moyennes empiriques par lignes
z <-rowSums((X-cbind(M,M))^2)  # echantillon des variances empiriques
hist(z,probability=T,col="blue",main ="Histogramme de z")
curve(dchisq(x,(m-1)), col="red", add=T)
                                 # Histogramme et densite theorique de z
```

Justifier que le vecteur z de la deuxième ligne peut également être obtenu par la commande

```
z= y-m*M^2
```

puis on pourra tracer

```
ztri <- sort(z)                # tri par ordre croissant
ord <- (1:n)/n                 # ordonnees
plot(ztri, ord, type="s", col="blue")
curve(pchisq(x,(m-1)), col="red", add=T)
                # Fonction de repartition empirique et theorique de z
```