Méthodes spécifiques pour Simuler des lois de probabilités continues

Prof. Mohamed El Merouani

2019/2020

Plan

- Loi normale
  - Inversion
  - Somme de 12 uniformes
  - Méthode de Box-Muller
  - Variante de Marsaglia
- Utilisation des transformations
  - Loi normale
  - Loi Log-normale
- Simulation de la loi exponentielle
  - Méthode d'inversion
- Loi de Cauchy par inversion
- Loi χ^2_n (khi-deux)
- Loi t de Student
- Loi F de Snedecor

- Il existe plusieurs approximations à la fonction de répartition de la loi normale et à son inverse.
- Une d'elles conduit à la formule d'inversion approximée

$$X = \frac{U^{0,135} - (1 - U)^{0,135}}{0,1975}$$

- Elle est plus rapide que les autres méthodes que nous décrivons.
- Dans quelques applications, elle fournit une approximation suffisamment bonne.

- Ce procédé se base sur le théorème de la limite centrale et il peut être considéré comme un exemple de la méthode de transformation.
- Par le théorème de la limite centrale, si les variables $U_i, i = 1, \cdots, n$ sont i.i.d. (**i**ndépendants, **i**dentiquement **d**istribuées) selon une loi uniforme $\mathcal{U}(0,1)$, par lequel $E(U_i) = \frac{1}{2}$, $Var(U_i) = \frac{1}{12}$, la variable

$$X = \frac{\left(\sum_{i=1}^{n} U_i - \frac{n}{2}\right)}{\sqrt{\frac{n}{12}}}$$

suit approximativement une loi normale, pour n suffisamment grande.

- Une bonne approximation se produit pour $n = 12$, par lequel $X = (\sum_{i=1}^{12} U_i) - 6$ et le procédé devient

Générer $U_1, \cdots, U_{12} \sim \mathcal{U}(0,1)$
Faire $X = (\sum_{i=1}^{12} U_i) - 6$
Sortir X

- Remarquons que X satisfait $E(X) = 0$ et $Var(X) = 1$. L'approximation est assez bonne, mais aussi la courbe de densité que celle de la fonction de répartition sont différentes, avec des différences qui oseillent entre 2 et 5 mm.

- La méthode exacte pour générer la loi normale la plus connue est celle de Box-Muller (1958), qui génère un couple de variables (X, Y) normales contrées réduites et indépendantes. La fonction de densité de (X, Y) est :

$$f(x, y) = \frac{1}{2\pi} \exp\left(-\frac{(x^2 + y^2)}{2}\right).$$

- Soient (R, θ) les coordonnées polaires de (X, Y), c'est-à-dire $R^2 = X^2 + Y^2$; $\tan\theta = \frac{Y}{X}$ qui ont la fonction de densité

$$g(r, \theta) = \frac{1}{2\pi}\left(r \exp\left(-\frac{r^2}{2}\right)\right) = g_1(\theta) g_2(r)$$

- sur $(0,\infty)\times(0,2\pi)$, avec $g_1(\theta)=\frac{1}{2\pi}\mathbb{I}_{(0,2\pi)}(\theta)$, c'est-à-dire une densité uniforme $\mathcal{U}(0,2\pi)$ et $g_2(r)=r\exp\left(-\frac{r^2}{2}\right)\mathbb{I}_{(0,\infty)}(r)$, c'est-à-dire une densité telle que R^2 est une exponentielle $\mathcal{E}(\frac{1}{2})\equiv\chi_2^2$. R et θ sont indépendantes.
- R se génère facilement par inversion, alors :

$$F(r)=\int_0^r \exp\left(-\frac{s^2}{2}\right)ds = 1-\exp\left(-\frac{r^2}{2}\right)$$

- Ainsi, si $U_1\sim\mathcal{U}(0,1)$, alors $R=\sqrt{-2\ln(1-U_1)}$ qui suit la même loi que $R=\sqrt{-2\ln U_1}$.

- Alors, on a l'algorithme suivant :

Générer $U_1, U_2 \sim \mathcal{U}(0,1)$
Faire $R = \sqrt{-2\ln U_1}$, $\theta = 2\pi U_2$
Faire $X = R\cos\theta = \sqrt{-2\ln U_1}\cos 2\pi U_2$
Faire $Y = R\sin\theta = \sqrt{-2\ln U_1}\sin 2\pi U_2$
Sortir X, Y

- Les équations pour obtenir X, Y s'appellent les transformations de Box-Muller.

- Marsaglia introduit sa variante polaire de la méthode de Box-Muller, qui utilise la méthode de rejet pour éviter les opérations trigonométriques de sinus et cosinus, peu efficientes. L'algoritme est :

Jusqu'à que $W \leq 1$
Générer $U_1, U_2 \sim \mathcal{U}(0,1)$
Faire $V_1 = 2U_1 - 1$, $V_2 = 2U_2 - 1$, $W = V_1^2 + V_2^2$
Faire $C = \frac{-2\ln W}{W}$
Sortir $X = CV_1$, $Y = CV_2$

- Remarquons que si $U \sim \mathcal{U}(0,1)$, $2U - 1 \sim \mathcal{U}(-1,1)$.
- Donc, (V_1, V_2) suit une loi uniforme sur $[-1,1]^2$.

Loi normale :

Variante de Marsaglia :

Si, nous ajoutons la restriction $V_1^2 + V_2^2 < 1$, le couple (V_1, V_2) suit une loi uniforme sur le disque unité. Si on note R et θ les coordonnées polaires de V_1 et V_2, on preuve que R^2 et θ sont variables indépendantes, avec $R^2 \sim \mathcal{U}(0,1)$ et $\theta \sim \mathcal{U}(0, 2\pi)$. Appliquons maintenant les transformations de Box avec $W = R^2 \sim \mathcal{U}(0,1)$ et $\theta \sim \mathcal{U}(0, 2\pi)$

$$\sin\theta = \frac{V_2}{\sqrt{V_1^2 + V_2^2}} = \frac{V_2}{\sqrt{W}}, \ \cos\theta = \frac{V_1}{\sqrt{V_1^2 + V_2^2}} = \frac{V_1}{\sqrt{W}}$$

Alors $X = \sqrt{-2\ln W}\cos\theta = CV_1, \ Y = \sqrt{-2\ln W}\sin\theta = CV_2$.
Puisque cet algorithme utilise la méthode de rejet pour éviter de calculer des fonctions trigonométriques, pratiquement il est plus rapide que celui de Box-Muller, mais avec une complexité additionnelle dans la programmation.

Utilisations des transformations :

Dans des occasions, il est possible d'utiliser des transformations entre variables aléatoires, de façon que si nous savons générer à partir d'une d'entre elles, on peut le faire pour les autres

- Supposons que l'on a simulé la loi normale centrée réduite $Y \sim \mathcal{N}(0,1)$ par une des méthodes vue précédement.
- Si On veut simuler la loi normale $N \sim \mathcal{N}(\mu, \sigma^2)$, il suffit de faire la transformation $X = \mu + \sigma Y$.

- Supposons que l'on a généré des variables normales Y comme précédement.
- On sait que si X est Log-normale, $\ln X$ est normale.
- Donc, il suffit de faire :

Générer Y normale
Sortir $X = \exp Y$

Simulation de la loi exponentielle :

On sait que si Y tire son échantillon à partir d'une loi exponentielle $\mathcal{E}xp(1)$, alors $X = \frac{Y}{\lambda}$ suit une loi exponentielle $\mathcal{E}xp(\lambda)$.
Par cela, supposons dans la suite que $\lambda = 1$.

Simulation de la loi exponentielle :

Méthode d'inversion :

La méthode d'inversion est simple, puisque la fonction de répartition est $F(x) = 1 - e^{-x}$,
Alors $X = -\ln(1-U)$
Comme $1-U$ suit la même loi de probabilité que U, il en résulte que $X = -\ln U$.
L'algorithme sera :

Générer $U \sim \mathcal{U}(0,1)$,
Faire $X = -\ln U$,
Sortir X

Malgré la simplicité théorique de ce procédé, il peut être lent sur ordinateur si l'opérateur logarithme népérien n'est pas implémenté dans le hardware.

Loi de Cauchy par inversion :

On considère le cas de simulation d'une variable aléatoire X qui suit une loi de Cauchy $\mathcal{C}(\alpha, \beta)$ en utilisant la méthode d'inversion.
La fonction de répartition de X est :

$$F(x) = \frac{1}{2} + \frac{1}{\pi} \arctan\left(\frac{x-\alpha}{\beta}\right)$$

Par inversion, on obtient :

$$X = F^{-1}(U) = \alpha + \beta \tan\left[\pi(U - \frac{1}{2})\right]$$

L'algorithme est :

Générer $U \sim \mathcal{U}(0,1)$,
Faire $X = \alpha + \beta \tan(\pi(U - 0.5))$
Sortir X

Loi χ^2_n (khi-deux) :

- Si les variables $Z_1, \cdots, Z_n$ sont indépendantes et suivent toutes une loi normale $\mathcal{N}(0,1)$, alors $X = \sum_{i=1}^n Z_i^2$ suit une loi χ^2_n.
- Cela suggère de générer n variables normales centrées réduites, les élever au carré et les sommer pour générer la loi χ^2_n.
- Lorsque $n > 30$, on peut utiliser l'approximation normale de la variable χ^2_n donnée par la formule $Z \simeq \sqrt{2X} - \sqrt{2n-1}$.
- Résolvons en X, on obtient $X = \frac{(Z+\sqrt{2n-1})^2}{2}$ et le procédé est immédiat.

Loi t de Student :

- Si Z est une variable normale $\mathcal{N}(0,1)$ et Y une variable indépendante de Z qui suit une loi χ^2_n, alors $X = \frac{Z}{\sqrt{\frac{Y}{n}}}(*)$ suit une loi t de Student avec n degrés de liberté.
- Pour générer la variable X, nous générons Z et Y, pour lesquelles nous connaissons déjà le procédé, et nous appliquons la formule $(*)$ précédente.
- Lorsque n est grande $(n > 30)$, nous utilisons l'approximation à la normale de la t de Student.

Loi F de Snedecor :

- Si Y_1 est une variable aléatoire qui suit une loi $\chi^2_{n_1}$ et Y_2 est une variable aléatoire qui suit une loi $\chi^2_{n_2}$ et toutes les deux sont indépendantes, alors :

$$X = \frac{Y_1/n_1}{Y_2/n_2} \quad (**)$$

  suit une loi F de Snedecor avec n_1 et n_2 degrés de liberté.

- Pour générer une loi F de Snedecor, premièrement nous générons deux variables $\chi^2_{n_1}$, $\chi^2_{n_2}$ et après nous appliquons $(**)$.