Statistiques TP 3
Théorème central limite. Méthode de MonteCarlo

Rappel : toujours faire $with(Statistics); with(plots);$... et sauver souvent !

1 Théorème central limite

On rappelle que le théorème central limite affirme que si (X_n) est une suite de variables aléatoires indépendantes de même loi et d'espérance μ et de variance σ^2 finies, $\sqrt{n}\sigma(\bar{x}-\mu)$ (où $\bar{x}$ désigne $\frac{\sum_{i=1}^{n} X_i}{n}$) converge en loi vers une loi normale centrée réduite. Cette quantité reste donc aléatoire, mais on sait comment elle est distribuée asymptotiquement.

1. On va commencer par observer ce phénomène si les X_i sont des variables de Bernoulli, et on va donc considérer une loi binomiale (pourquoi ?)
   On peut par exemple mettre en évidence le théorème central limite sur le graphes des lois : soit Z une variable aléatoire qui suit une loi binomiale $B(n,p)$ ($Z := RandomVariable(Binomial(n,p))$).
   (a) Rappeler l'espérance de Z et sa variance. Pour $n = 500$ et $p = 0.3$, obtenir $P(Z = 40)$ à l'aide de la commande *ProbabilityFunction*. Obtenir le graphe des $(k, P(Z = k))$ à l'aide de *DensityPlot*. Comparer avec la densité d'une variable aléatoire normale de même espérance et variance.
   (b) Vers quoi converge en loi $\frac{Z-np}{\sqrt{np(1-p)}}$? Faire une procédure qui prend les paramètres de la loi binomiale n et p en entrée, et trace le graphe formé des points $(M_k)_{0\leq k\leq n}$ de coordonnées $(\frac{k-np}{\sqrt{np(1-p)}}, \sqrt{np(1-p)}P(Z = k))$, ainsi que la densité de la loi normale centrée réduite.
   Rappel : une procédure se présente sous la forme :
   $Bidule := proc(parametresenentree)$
   $instructions;$
   $end proc;$
   Pour aller à la ligne sans effectuer tout de suite l'instruction avec maple, faire "shift+ entrée". Tester la procédure avec diverses valeurs de n et p. (prendre $n \leq 1000$).
   (c) On peut aussi voir le théorème central limite sur des échantillons. Faire une procédure qui prend n, p et N_{obs} en entrée, génère un N_{obs} échantillon $(Z_1, \dots Z_{Nobs})$ de la loi binomiale $B(n,p)$, obtient un vecteur comportant les $\frac{Z_i-np}{\sqrt{np(1-p)}}$ (en utilisant par exemple $\mathbf{map(x->f(x),v)}$ qui applique la fonction f à tous les éléments d'une liste ou d'un vecteur), trace un histogramme des $\frac{Z_i-np}{\sqrt{np(1-p)}}$ ainsi que la densité de la loi normale centrée réduite. Lancer la procédure avec $N_{obs} = 1000$, $n = 100$ et $p = 0.1$ puis 0.3, 0.5, 0.99. Recommencer avec $n = 300$.
2. On reprend le lancer de dé de la derniere fois ($X := RandomVariable(DiscreteUniform(1,6))$), et on va utiliser la fonction de répartition, plus fiable que l'histogramme. Construire une procédure qui prend en entrée deux entiers n et N_{obs}, génère N_{obs} n-échantillons de X, calcule pour chacun de ces échantillons $\frac{\sqrt{n}}{\sigma}(\bar{x}-m)$, stocke ces valeurs dans une liste, et enfin trace sur un même graphique la fonction de répartition empirique de ces grandeurs ainsi que le graphe de la fonction de répartition empirique de la loi normale centrée réduite.
   (Rappels : on peut tracer la fonction de répartition empirique en traçant la suite de points de coordonnées $(Quantile(i/100), i/100)$.
   Pour tracer une suite de points de coordonnées A_i, B_i on peut utiliser
   $\mathbf{plot([seq([A_i, B_i], i = 1..100)]);}$
3. Tester la procédure avec $Nobs = 1000$ et $n = 30, 50, 100, 300, 1000$.

2 Méthode de MonteCarlo

Soit $(U_1, \ldots, U_d)$ un vecteur de d variables aléatoires indépendantes toutes de loi uniforme sur $[0,1]$, et f une fonction de carré intégrable sur $[0,1]^d$. On considère la variable aléatoire $X = f(U_1, \ldots, U_d)$. D'après la loi forte des grands nombres, si X_n est une suite de variables aléatoires indépendantes toutes de même loi que X,
$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} X_i = E(f(U_1, \ldots, U_d)) = \int_{[0,1]^d} f(u_1, \ldots, u_d) du_1 \ldots du_d$.
La méthode de Monte Carlo consiste à utiliser ceci pour calculer une approximation de
$\int_{[0,1]^d} f(u_1, \ldots, u_d) du_1 \ldots du_d$. Le théorème central limite permet de plus d'avoir une idée de l'ordre de grandeur de l'erreur. La convergence est en $\frac{\sigma}{\sqrt{n}}$, où σ est l'écart-type de X, et cela ne dépend pas de d. L'intérêt par rapport à d'autres méthodes déterministes, qui approchent l'intégrale par exemple par des sommes de Riemann, se fait donc surtout sentir pour des intégrales multiples. En effet l'erreur avec ce genre de méthodes est en général proportionnel au Pas de la subdivision; mais le nombre d'itérations de ces algorithmes est proportionnel à $(Pas)^{-d}$.

On va se servir de la méthode de Monte-Carlo pour calculer π.

1. On va d'abord utiliser $\pi = \int_0^1 4\sqrt{1-x^2}dx$. Générer un 500 échantillon $[U_1, \ldots, U_{500}]$ de loi uniforme sur $[0,1]$. Obtenir l'échantillon $[X_1, \ldots, X_{500}]$ où $X_i = 4\sqrt{1-U_i^2}$ (par exemple en utilisant *map*). Donner une estimation de $\pi = E[X]$ et de l'écart-type de X.

2. On va déterminer un intervalle de confiance pour π en utilisant $\pi = 4\int_{[0,1]^2} 1_{x^2+y^2\leq 1} dxdy$. Si (U_1, U_2) suit une loi uniforme sur $[0,1]^2$, quelle est la loi de $X = 41_{U_1^2+U_2^2\leq 1}$? Donner son espérance et son écart-type en fonction de π.

   Utiliser votre 500 échantillon précédent $[U_1, \ldots, U_{500}]$ pour obtenir une estimation de π et de l'écart-type de X_i.

3. Estimer π en utilisant $\pi = 6\int_{[0,1]^3} 1_{x^2+y^2+z^2\leq 1} dxdydz$.

3 Challenge

J'ai mis sur ma page web (http://www.math.u-bordeaux1.fr/˜ chabanol), dans le répertoire Enseignement puis Statistiques, 6 échantillons obtenus en simulant des lois classiques. Récupérez-les et faites-les lire par maple à l'aide de la commande *importdata("toto").* Le nom du fichier doit être entre guillemets. Sous linux, l'adresse sera donnée sour la forme, par exemple,
importdata("/usr/local/Stat/TP/toto").
Sous windows, l'adresse peut être donnée de deux manières :
importdata("C : \\Stat\\TP\\toto") ou
importdata("C:/Stat/TP /toto").
Serez-vous capable de deviner quelle loi a été utilisée pour générer chaque échantillon ?