TP13/14 : Illustration du Théorème Central Limite

Pré-requis : je vous invite à consulter les chapitres de cours correspondants sur ma page (support informatique). Pour ce TP, on pourra en particulier se reporter à la section « Tracés d'histogrammes » du CH 8.

▶ Dans votre dossier `Info_2a`, créez le dossier `TP_13`.

I. Théorème Central Limite

Le but de ce TP est d'illustrer le **Théorème Central Limite**[1] (TCL). Ce théorème énonce la **convergence en loi** de suites de variables aléatoires. Commençons par rappeler la définition de convergence en loi.

Définition

Soit $(X_n)_{n\in\mathbb{N}^*}$ une suite de v.a.r. définies sur $(\Omega, \mathscr{A}, \mathbb{P})$.

Soit X une v.a.r. définie sur $(\Omega, \mathscr{A}, \mathbb{P})$.

Soient F_{X_n} et F_X les fonctions de répartition de associées à ces v.a.r.

- On dit que la suite $(X_n)_{n\in\mathbb{N}^*}$ **converge en loi** vers X si :

$$\boxed{\lim_{n\to+\infty} F_{X_n}(x) = F_X(x)}$$

en tout point $x \in \mathbb{R}$ où F_X est continue.

- On note alors : $X_n \xrightarrow{\mathcal{L}} X$.

Théorème 1. *Théorème Central Limite*

Soit $(X_n)_{n\in\mathbb{N}^*}$ une suite de v.a.r. indépendantes, de même loi, d'espérance m et d'écart-type σ.

Notons $S_n = \sum\limits_{k=1}^{n} X_k$ et $\overline{X_n} = \dfrac{S_n}{n}$.

Notons enfin ${S_n}^* = \dfrac{S_n - nm}{\sigma\sqrt{n}} = \dfrac{\overline{X_n} - m}{\frac{\sigma}{\sqrt{n}}}$ la v.a.r. centrée réduite associée à S_n.

- *Alors $({S_n}^*)$ converge en loi vers une v.a.r. de loi normale $\mathcal{N}(0,1)$.*
- *En particulier, pour tout $a \in \overline{\mathbb{R}}$ et tout $b \in \overline{\mathbb{R}}$ tels que $a \leqslant b$, on a :*

$$\boxed{\lim_{n\to+\infty} \mathbb{P}(a \leqslant {S_n}^* \leqslant b) = \int_a^b \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}\, dt}$$

☞ Ainsi, pout n suffisamment grand, la loi de ${S_n}^*$ approche la loi $\mathcal{N}(0,1)$.
Nous allons illustrer cette propriété pour des suites de v.a.r. (X_n) dont tous les éléments suivent une loi uniforme / de Bernoulli / de Poisson.

[1] On parle aussi du Théorème de la Limite Centrée (TLC).

Dans la suite, on considère les suites de v.a.r. (X_n) et (S_n) comme définies dans l'énoncé.

- Pour $n \in \mathbb{N}^*$, déterminer l'espérance et la variance de S_n.

- Déterminer l'espérance et la variance de ${S_n}^*$.

- Déterminer $\overline{X_n}^*$ la v.a.r. centrée réduite associée à X_n.

II. La loi normale en Scilab

- Coder dans un onglet **SciNotes** la densité d'une loi normale centrée réduite.
  On nommera cette fonction `densiteNormaleCR`.

- Rappeler l'expression de la fonction de répartition Φ de la loi normale centrée réduite.

- Quelle méthode (non aléatoire) peut-on proposer pour calculer une version approchée de cette fonction de répartition ?

En **Scilab**, ce calcul approché est implémenté par la fonction `cdfnor` (**c**umulative **d**istribution **f**unction **nor**mal distribution). L'appel général est le suivant :

```
[P,Q] = cdfnor("PQ", X, Mean, Std)
```

Détaillons les différents éléments de cet appel.

× `X` : borne supérieure d'intégration,

× `Mean` : moyenne choisie (la loi n'est pas forcément centrée),

× `Std` : écart-type choisi (la loi n'est pas forcément réduite),

× `P` : résultat de l'appel. Autrement dit, c'est le calcul approché de $\displaystyle\int_{-\infty}^{\mathtt{X}} \varphi(t)\ dt$,

× `Q` : second résultat de l'appel. Cette variable contient le calcul approché de $\displaystyle\int_{\mathtt{X}}^{+\infty} \varphi(t)\ dt$.

▶ Donner une relation entre `P` et `Q`.

▶ Coder dans un nouvel onglet **SciNotes** la fonction de répartition de loi normale centrée réduite. On nommera cette fonction `repartitionNormaleCR`.

Remarque

- L'option `"PQ"` sert à signifier que l'on cherche à déterminer les paramètres `P` et `Q`. Pour ce faire, tous les autres paramètres doivent être fournis lors de l'appel.
- En fait, `cdfnor` permet de réaliser le calcul de n'importe lequel des paramètres de la distribution normale si tous les autres paramètres sont fournis à la fonction. Plus précisément, on peut écrire :

  × `X = cdfnor("X", Mean, Std, P, Q)`,

  × `Mean = cdfnor("Mean", Std, P, Q, X)`,

  × `Std = cdfnor("Std", P, Q, X, Mean)`.

III. Simulation de loi normale à l'aide de lois usuelles

Le cadre du TCL permet de simuler une v.a.r. suivant la loi normale centrée réduite à l'aide de lois usuelles. Dans la suite, on compare la densité de probabilité théorique avec celle obtenue par simulation. Pour ce faire, on procède comme suit :

× on produit `N` observations de la simulation de ${S_n}^*$,

× on trace l'histogramme des fréquences associés.

Il reste à préciser quelle loi doit suivre les variables X_k.
On s'intéresse dans la suite aux cas suivants :

1) $X_k \hookrightarrow \mathcal{U}([0,1[)$, *2)* $X_k \hookrightarrow \mathcal{B}(p)$, *3)* $X_k \hookrightarrow \mathcal{P}(\lambda)$.

III.1. Simulation de la loi normale via les « 12 uniformes »

Dans le TCL, la convergence peut être très rapide. La loi de la variable ${S_n}^*$ est alors une bonne approximation de la loi $\mathcal{N}(0,1)$ pour des valeurs de n assez petites.

III.1.a) Rappels sur la loi uniforme

- Si $(a,b) \in \mathbb{R}^2$ avec $a < b$, que signifie $X \hookrightarrow \mathcal{U}([a,b])$?

- Rappeler la valeur de $\mathbb{E}(X)$ et $\mathbb{V}(X)$ si $X \hookrightarrow \mathcal{U}([a,b])$.

- En déduire la valeur de $m = \mathbb{E}(X)$ et $\sigma^2 = \mathbb{V}(X)$ dans le cas où $X \hookrightarrow \mathcal{U}([0,1[)$.

III.1.b) Simulation de la densité de probabilité

Dans ce paragraphe :

- × on considère des variables X_k indépendantes telles que $X_k \hookrightarrow \mathcal{U}([0,1[)$.
- × on choisit $n = 12$. Autrement dit, $S_n = S_{12}$.
- × on simule la v.a.r. $S_{12}{}^*$. Plus précisément, on représente l'histogramme des effectifs issu de N observations de $S_{12}{}^*$ et on compare ce résultat à la densité de la loi $\mathcal{N}(0,1)$.

► Que vaut alors $S_{12}{}^*$? En donner une expression simple à l'aide de S_{12}.

► Par quel appel simule-t-on, à l'aide de la fonction `grand`, `1` échantillon de `12` v.a.r. indépendantes suivant la loi $\mathcal{U}([0,1[)$? En déduire la simulation de la v.a.r. S_{12}.

► Par quel appel simule-t-on, à l'aide de la fonction `grand`, `N = 10000` échantillons de `12` v.a.r. indépendantes suivant la loi $\mathcal{U}([0,1[)$? En déduire la simulation de `N` échantillons de S_{12}.

► Recopier et compléter le programme suivant.
On l'enregistrera sous le nom `comparaison_dnormale_12_unif.sce`.

```
// Valeur des paramètres
N = 10000
nbC = 100

// Simulations de N observations de S12
G =
S =

// Simulations de N observations de S12*
Scr =

// Tracé de la densité théorique
plot
// Tracé de la densité observée
histplot

legend(["Simu - 12 uniformes","Densité N(0,1)"], "in_upper_right")
xtitle(["Théorème Central Limite :";
" diagramme des fréquences des 12 uniformes";
" comparaison avec le tracé de la densité de N(0,1)"])
```

III.1.c) Simulation de la fonction de répartition

On se propose maintenant de comparer la fonction de répartition théorique avec sa version obtenue par simulation. Pour ce faire, on procède comme suit :

× on récupère le tableau des effectifs des N observations précédentes,

× on trace le diagramme en bâtons des effectifs **cumulés** associé.

× compare ce diagramme avec la fonction de répartition de la loi $\mathcal{N}(0,1)$.

On pourra faire appel à la fonction `dsearch` qui permet de déterminer la fréquence des valeurs contenues dans un vecteur. L'appel général est le suivant :

```
[indClasse, effectif] = dsearch(Obs, classe)
```

Détaillons les différents éléments de cet appel.

× `Obs` : une série d'observations,

× `classe` : une liste de réels rangés dans l'ordre strictement croissant définissant les classes,
Par exemple, `classe = [0, 3, 7, 9]` *permet de définir trois classes :*

   1) *la* $1^{\text{ère}}$ *classe est l'intervalle* `[0, 3]`,

   2) *la* $2^{\text{ème}}$ *classe est l'intervalle* `]3,7]`,

   3) *la* $3^{\text{ème}}$ *classe est l'intervalle* `]7,9]`.

× `indClasse` : indique, pour chaque observation, la classe à laquelle elle appartient,

× `effectif` : indique, pour chaque classe, le nombre d'observations qu'elle contient.

► Commenter le résultat de l'appel :

```
[indClasse, effectif] = dsearch([8.4, 7, 0, 1, 9, 8, %pi, 2, sqrt(2)], [0, 3, 7, 9])
```

► Par quel appel obtient-on le tableau `effCumule` des effectifs cumulés à l'aide du tableau `effectif` des effectifs ?

► Comment obtenir la fréquence des observations dans chaque classe ?

- La fonction `dsearch` est l'analogue, dans le cas continu, d'une fonction dont on s'est servi dans le cas discret. Rappeler le nom de cette fonction et décrire brièvement son utilisation.

- Recopier et compléter le programme suivant.
On l'enregistrera sous le nom `comparaison_rnormale_12_unif.sce`.

```
// Valeur des paramètres
N = 10000
nbC = 100

// Simulations de N observations de S12
G =
S =

// Simulations de N observations de S12*
Scr =

// Simulation de la fonction de répartition
classe =
[indice, effectif] = dsearch
effCumule =

// Tracé de la fonction de répartition observée
bar
// Tracé de la fonction répartition théorique
plot

legend(["Simu - 12 uniformes","Fonction répartition N(0,1)"],
"in_upper_left")
xtitle(["Théorème Central Limite :";
" diagramme des fréquences cumulées des 12 uniformes";
" comparaison avec le tracé de la fct répartition de N(0,1)"])
```