

Devoir 2 : Debug & cie

Patrick Fournier

20 septembre 2021

Répondez aux questions en complétant le fichier `solution.R`. Veuillez respecter la structure du fichier. N'hésitez pas à consulter la documentation de R !

Exercice 1

(a)

Réécrivez les fonctions `cutVector` et `dist0` de manière à ce qu'elles ne fasse pas appel à `lapply` et `sapply`. *Indice : for*

(b)

Quelle version de `dist0` est la plus rapide ? Faites un graphique (boxplot par exemple) comparant les temps d'exécution. Enregistrez ce graphique dans un fichier pdf que vous déposerez avec le reste de vos solutions.

(c)

On a vu en classe que l'appel à `sample` domine le temps d'exécution de `dist0`. Est-ce aussi le cas pour votre fonction `dist0_loop` ?

Exercice 2

Débuguez les fonctions fournies. L'utilisation d'un débogueur est conseillée. N'oubliez pas de consulter la documentation !

(a)

Chiffre de César (Wikipedia pour plus de détails). On suppose la chaîne de caractère à encrypter alphanumérique (a-z, A-Z, 0-9) et la clé comprise entre 1 et 100 inclusivement. Notez que pour une chaîne `s` et une clé `k`, on devrait avoir que `identical(cesar(cesar(s, k), k, decode = TRUE), s)` vaut `TRUE`. *Indice : Les entrées d'un vecteur de type `raw` sont des nombres hexadécimaux.*

(b)

Génération d'une Poisson zéro-tronquée (Wikipedia...). La fonction prend en argument, en plus de la taille d'échantillon `n`, un paramètre $\lambda \in (0, \infty)$. Ne fonctionne pas pour certaines valeurs de λ , 0.5 par exemple. Corrigez cela !