

# TD de simulation d'équations différentielles stochastiques

Pierre Gloaguen

27/03/2020

## Packages nécessaires

Il suffira de charger `tidyverse`:

```
library(tidyverse)
```

Si jamais vous ne l'avez pas ou cela ne fonctionne pas (ce qui semble arriver sur certains Mac), chargez simplement les 3 packages suivant:

```
library(dplyr)
library(purrr)
library(ggplot2)
library(tibble)
```

## 1 Un modèle de croissance logistique

On s'intéresse à un modèle de dynamique de population plus réaliste que le modèle de croissance malthusienne vu en cours.

On note  $X(t)$  la taille de la population au temps  $t$ . On suppose que le processus  $\{X(t)\}_{t \geq 0}$  satisfait l'équation différentielle stochastique suivante:

$$dX(t) = rX(t) \left(1 - \frac{X(t)}{K}\right) dt + \sigma X(t) dB(t),$$

où  $r, K, \sigma$  sont trois paramètres **strictement positifs**.

1. Identifiez la fonction de dérive et la fonction de diffusion de cette EDS.
2. Codez ces deux fonctions dans le logiciel R. Ces deux fonctions prendront comme argument une taille de population  $x$  et leurs paramètres (donc  $r$  et  $K$  pour la dérive, et  $\sigma$  pour la diffusion).
3. En reprenant le squelette de fonction de simulation d'une EDS vu en cours, écrire une fonction de simulation de cette EDS, qui prendra en plus comme argument les paramètres  $r, K$  et  $\sigma$ .
4. Pour le vecteur de temps suivant:

```
my_times <- seq(from = 0, to = 10, by = 0.01) # "by" donne le pas de temps h
```

simulez le processus pour  $r = 0.1, 0.5$ ,  $K = 50, 500$ , et  $\sigma = 0.1, 3$ . Dans chaque cas, vous ferez varier le point de départ entre  $0.1K$  et  $2K$ . Vous ferez plusieurs simulations afin d'évaluer la variabilité du processus.

5. Quelle interprétation faites vous des 3 paramètres?
6. **Validité du schéma d'Euler:** Dans l'exercice précédent, pour  $r = 0.5$ ,  $K = 500$  et  $\sigma = 0.1$ , faites 500 simulations et conservez les points finaux (c'est à dire la valeur du processus au temps 10). Tracez la distribution de ces points à l'aide d'un histogramme (ou de la fonction `density`, ou `geom_density`). Refaites cet exercice pour un `by = 0.001`, `by = 1` et `by = 2` dans le vecteur `my_times`. Comparez les

différents histogrammes. Que remarquez vous? D'après vous, quel est le bon pas de temps pour simuler ce processus?

## 2 EDS en deux dimensions, un modèle proie prédateur stochastique

On s'intéresse à la taille de deux populations en interaction (une population de proie et une population de prédateurs). Au temps  $t$ , la taille de la population est donnée par le vecteur  $Z = \begin{pmatrix} X(t) \\ Y(t) \end{pmatrix}$ . On suppose que le processus  $\left\{ \begin{pmatrix} X(t) \\ Y(t) \end{pmatrix} \right\}_{t \geq 0}$  satisfait l'équation différentielle stochastique en deux dimensions

$$\begin{aligned} dX(t) &= X(t) (a_{10} - a_{11}X(t) - a_{12}Y(t)) dt + \sigma_1 X(t) dB_1(t) \\ dY(t) &= Y(t) (-a_{20} + a_{21}X(t) - a_{22}Y(t)) dt + \sigma_2 Y(t) dB_2(t) \end{aligned}$$

où  $\{a_{ij}, \sigma_i\}_{1 \leq i \leq 2, 0 \leq j \leq 2}$  sont paramètres **positifs**, et  $\{B_1(t)\}_{t \geq 0}$  et  $\{B_2(t)\}_{t \geq 0}$  sont deux mouvements Browniens **indépendants**.

1. Identifiez les différentes fonctions de dérive et de diffusion et codez les dans R. Elles dépendront d'un vecteur  $z$  de longueur 2 et des différents paramètres inconnus.
2. Interpréter les différents paramètres.
3. Ecrivez la fonction de simulation du modèle.
4. Pour les jeux de paramètres suivants (correspondant à 3 scénarios), simulez le processus pour  $z(0) = (10, 5)$  et des temps allant de 0 à 5 (pas de temps de 0.01). Que constatez vous? Essayez d'interpréter les résultats

```
parameters_list <- tibble(a10 = c(12, 4, 0.2), # 1 par scenario
                          a11 = c(0.05, 1, 0.05),
                          a12 = c(1, 1, 1),
                          a20 = c(2, 2, 2),
                          a21 = c(0.2, 0.2, 0.2),
                          a22 = c(0.1, 0.1, 0.1),
                          sigma1 = c(0.5, 0.5, 2),
                          sigma2 = c(0.2, 0.5, 0.2))
```