

Chapitre 4

Travaux pratiques

L'objectif principal de cette série de travaux pratiques est d'explorer l'ensemble des techniques de simulation, de modélisation markovienne, et d'analyse probabiliste étudiées dans ce cours sur une série d'exemples concrets.

Nous recommandons au lecteur d'adopter une double réflexion. Tout d'abord, il conviendra de s'assurer, du moins intuitivement, de la stabilité du processus aléatoire étudié, et des fondements probabilistes du résultat recherché. Il est aussi recommandé de mener une réflexion sur les qualités numériques et physiques des algorithmes. On s'interrogera notamment sur leurs performances, et leurs limitations pratiques.

4.1 Fonctions itérées stochastiques

Cette première série de travaux pratiques se focalise sur les fonctions itérées stochastiques, et la simulation d'images fractales. Ces algorithmes stochastiques sont fondés sur l'exploration d'une surface plane par une séquence de contractions affines. Le seul aléa réside simplement dans le choix à chaque étape de l'une de ces transformations.

Ces modèles sont sans nul doute les chaînes de Markov les plus simples à simuler numériquement. Ils sont simplement définis par donnée d'une famille de transformations affines du plan $(S_i)_{i \in I}$ indexée par un ensemble fini I , muni d'une mesure de probabilité p . Pour les décrire, on introduit une suite $(\epsilon_n)_{n \geq 1}$ de variables aléatoires indépendantes et de même loi p sur I . Ces algorithmes stochastiques sont représentés par la donnée d'une chaîne de Markov

$X = (X_n)_{n \geq 0}$ définie récursivement par les équations suivantes :

$$X_0 \in \mathbb{R}^2 \quad X_n = S_{\epsilon_n}(X_{n-1}) = S_{\epsilon_n} S_{\epsilon_{n-1}}(X_{n-2}) = \dots = S_{\epsilon_n} \dots S_{\epsilon_1}(X_0)$$

4.1.1 Modèle fractal de feuille

Dans ce premier modèle, on se donne une suite de variables aléatoires indépendantes de Bernoulli $(\epsilon_n)_{n \geq 1}$ avec

$$\mathbb{P}(\epsilon_n = 1) = 1 - \mathbb{P}(\epsilon_n = 0) = 0.2993$$

Pour chaque $i = 0, 1$, on choisit des fonctions affines $S_i(x) = A_i \cdot x + b_i$ avec les matrices

$$A_0 = \begin{pmatrix} +0.4000 & -0.3733 \\ +0.0600 & +0.6000 \end{pmatrix} \quad b_0 = \begin{pmatrix} +0.3533 \\ +0.0000 \end{pmatrix}$$

et

$$A_1 = \begin{pmatrix} -0.8000 & -0.1867 \\ +0.1371 & +0.8000 \end{pmatrix} \quad b_1 = \begin{pmatrix} +1.1000 \\ +0.1000 \end{pmatrix}$$

4.1.2 Modèle fractal d'arbre

Dans ce second modèle, on se donne une suite de variables aléatoires indépendantes de Bernoulli uniformes sur $I = \{1, 2, 3\}$. Pour chaque $i = 1, 2, 3$, on choisit des fonctions affines $S_i(x) = A_i \cdot x + b_i$ avec les matrices

$$A_1 = \begin{pmatrix} 0 & 0 \\ 0 & c \end{pmatrix} \quad b_1 = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} r \cos(\varphi) & -r \sin(\varphi) \\ r \sin(\varphi) & r \cos(\varphi) \end{pmatrix} \quad b_2 = \begin{pmatrix} \frac{1}{2} - \frac{r}{2} \cos(\varphi) \\ c - \frac{r}{2} \sin(\varphi) \end{pmatrix}$$

et

$$A_3 = \begin{pmatrix} q \cos(\psi) & -r \sin(\psi) \\ q \sin(\psi) & r \cos(\psi) \end{pmatrix} \quad b_3 = \begin{pmatrix} \frac{1}{2} - \frac{q}{2} \cos(\psi) \\ \frac{3c}{5} - \frac{q}{2} \sin(\psi) \end{pmatrix}$$

Les paramètres précédents sont définis par

$$\begin{aligned} c &= 0.255, & r &= 0.75, & q &= 0.625 \\ \varphi &= -\frac{\pi}{8}, & \psi &= \frac{\pi}{5}, & |X_0| &\leq 1. \end{aligned}$$

4.1.3 Triangle de Sierpinski

On se donne une suite de variables aléatoires indépendantes de Bernoulli uniformes sur $I = \{1, 2, 3\}$. Pour chaque $i = 1, 2, 3$, on choisit des fonctions affines $S_i(x) = A_i \cdot x + b_i$ avec la matrice

$$A_1 = A_2 = A_3 = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix}$$

et le vecteurs

$$b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad b_2 = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix} \quad \text{et} \quad b_3 = \begin{pmatrix} 1/4 \\ \sqrt{3}/4 \end{pmatrix} \simeq \begin{pmatrix} 0,250 \\ 0,433 \end{pmatrix}$$

4.1.4 Le Dragon de Heighways

On se donne une suite de variables aléatoires indépendantes de Bernoulli uniformes sur $I = \{1, 2\}$. Pour chaque $i = 1, 2$, on choisit des fonctions affines $S_i(x) = A_i \cdot x + b_i$ avec les matrices

$$A_1 = \begin{pmatrix} 1/2 & -1/2 \\ 1/2 & 1/2 \end{pmatrix} \quad \text{et} \quad A_2 = \begin{pmatrix} -1/2 & -1/2 \\ 1/2 & -1/2 \end{pmatrix}$$

et les vecteurs $b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $b_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$.

4.2 Estimation d'une médiane

Soit $U : \mathbb{R} \rightarrow \mathbb{R}$ la fonction de répartition d'une variable aléatoire réelle Y

$$U(x) = \mathbb{P}(Y \leq x) = \mathbb{E}(1_{Y \leq x}).$$

L'objectif de cette étude est d'estimer numériquement la médiane $x^* \in \mathbb{R}$ de cette distribution. On rappelle que la médiane est le point $x^* \in \mathbb{R}$ séparant en deux parties les masses de probabilités d'une variable réelle

$$\mathbb{P}(Y \leq x^*) = \frac{1}{2} = \mathbb{P}(Y > x^*).$$

- Donner des conditions suffisantes sur la fonction de répartition pour que la condition (2.4) soit satisfaite (pour $a = 1/2$ et $x_a = x^*$).

– Soit $\mathcal{U} : \mathbb{R} \times \mathbb{R} \longrightarrow \mathbb{R}$ la fonction définie par

$$\mathcal{U}(x, y) = 1_{]-\infty, x]}(y).$$

Montrer que

$$U(x) = \mathbb{E}(\mathcal{U}(x, Y)).$$

– Construire un algorithme de Robbins-Monro permettant d'approcher cette médiane. On pourra concrétiser cette étude avec des exemples concrets de mesures de probabilités usuelles, et rechercher d'autres niveaux

$$\mathbb{P}(Y \leq x_a) = a$$

pour des valeurs du paramètre a , plus ou moins grandes.

4.3 Voyageur de commerce

Ce problème classique d'optimisation combinatoire peut se définir de la façon suivante. On se donne un ensemble fini

$$\mathcal{V} = \{v_1, \dots, v_N\}$$

de $N \geq 1$ villes ainsi que la distance mutuelle entre chaque ville. Ce qui revient à se donner une distance $d : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}_+$. Le voyageur de commerce partant de la ville v_1 doit trouver une permutation

$$\sigma = \begin{pmatrix} 1 & \dots & N \\ \sigma(1) & \dots & \sigma(N) \end{pmatrix}$$

telle que $\sigma(1) = 1$, de sorte à minimiser la distance

$$d(v_{\sigma(1)}, v_{\sigma(2)}) + \dots + d(v_{\sigma(N-1)}, v_{\sigma(N)}) + d(v_{\sigma(N)}, v_{\sigma(1)})$$

qu'il aura parcourue en effectuant le circuit

$$v_{\sigma(1)} = v_1 \longrightarrow v_{\sigma(2)} \longrightarrow \dots \longrightarrow v_{\sigma(N)} \longrightarrow v_{\sigma(1)} = v_1.$$

On notera par la suite \mathcal{G}_N l'ensemble des permutations de $\{1, \dots, N\}$. Pour chaque $\sigma \in \mathcal{G}_N$, on utilisera la convention $\sigma(N+1) = \sigma(1)$ et on notera

$$U : \sigma \in \mathcal{G}_N \mapsto U(\sigma) = \sum_{p=1}^N d(v_{\sigma(p)}, v_{\sigma(p+1)}) \in \mathbb{R}_+.$$

Pour chaque $p \geq 1$, on désigne par $\mathcal{G}_N(p)$ et $\mathcal{G}_N^*(p)$ les sous-ensembles de \mathcal{G}_N définis par

$$\begin{aligned}\mathcal{G}_N(p) &= \{\sigma \in \mathcal{G}_N ; \sigma(p) = 1\} \\ \mathcal{G}_N^*(p) &= \{\sigma \in \mathcal{G}_N(p) ; U(\sigma) = \min_{\tau \in \mathcal{G}_N(p)} U(\tau)\}.\end{aligned}$$

On prendra comme espace d'état $E = \mathcal{G}_N$. Il existe sur cet espace diverses façons de spécifier des systèmes de voisinage.

– Permutation de deux indices :

$$\sigma \sim \tau \iff \exists i \leq j : \sigma = \begin{pmatrix} 1 & \dots & i & \dots & j & \dots & N \\ \tau(1) & \dots & \tau(j) & \dots & \tau(i) & \dots & \tau(N) \end{pmatrix}$$

– Inversion d'une suite d'indices :

$$\sigma \sim \tau \iff \exists i \leq j : \sigma = \begin{pmatrix} 1 & \dots & i+0 & \dots & i+p & \dots & i+(j-i) = j & \dots & N \\ \tau(1) & \dots & \tau(j-0) & \dots & \tau(j-p) & \dots & \tau(j-(j-i)) = \tau(j) & \dots & \tau(N) \end{pmatrix}$$

Dans les deux cas, montrer que pour chaque σ et $\tau \in \mathcal{G}_N$ il existe une suite $(\sigma_0, \dots, \sigma_N)$ d'éléments de \mathcal{G}_N telle que

$$\sigma_0 = \sigma \sim \sigma_1 \sim \dots \sim \sigma_N = \tau.$$

Pour réaliser l'algorithme de recuit, on utilisera un noyau d'exploration $K(\sigma, \tau)$ défini par l'un de ces systèmes de voisinage, c'est à dire

$$K(\sigma, \tau) = \frac{1}{|\mathcal{V}(\sigma)|} 1_{\mathcal{V}(\sigma)}(\tau), \quad \mathcal{V}(\sigma) = \{\tau \in \mathcal{G}_N ; \tau \sim \sigma\}.$$

– Vérifier que

$$\sigma \sim \tau \iff K(\sigma, \tau) > 0.$$

– En déduire que

$$\forall \sigma, \tau \in \mathcal{G}_N, \quad K^N(\sigma, \tau) > 0.$$

4.4 Filtrage de signaux linéaires et gaussiens

L'estimation de signaux numériques partiellement observés, et ayant une représentation d'état linéaire et gaussienne, peut s'effectuer de manière exacte

par le filtre de Kalman-Bucy. Cet estimateur dynamique optimal peut aussi s'interpréter comme une technique de moindres carrés récursives. Dans le cas scalaire, ces modèles linéaires, et gaussiens, s'expriment sous la forme suivante :

$$\begin{cases} X_n = a X_{n-1} + W_n, & n \geq 1 \\ Y_n = c X_n + V_n, & n \geq 0 \end{cases} \quad (4.1)$$

avec X_0, V_n, W_{n+1} , $n \geq 0$, une suite de variables aléatoires indépendantes de même loi $\mathcal{N}(0, 1)$ et, a, c deux nombres réels. La chaîne de Markov X_n représente l'évolution du signal, et la séquence Y_n ses observations bruitées.

En pratique, le filtre de Kalman-Bucy utilise les observations fournies par un capteur de mesure. On dit que ces valeurs sont **des données réelles** par opposition à des séquences d'observations que l'on pourrait construire soi-même en simulant la chaîne de Markov (X, Y) sur l'ordinateur. Dans ce dernier cas on parlera de **données simulées**.

Pour construire expérimentalement le filtre de Kalman-Bucy, on réalise tout d'abord une simulation du système (4.1). Avant de calculer le filtre de Kalman-Bucy on stockera dans un fichier les valeurs X_n et Y_n obtenues en simulant le système (4.1). Les observations simulées Y_n sont considérées comme les observations partielles et bruitées de la réalisation "inconnue" du signal X_n . En utilisant le fichier des données simulées Y_n , on peut calculer l'évolution du filtre de Kalman-Bucy, et comparer ces valeurs avec les valeurs "réelles" du signal. On pourra effectuer cette comparaison pour les valeurs $a = 1/2$ et $c = 1$.

4.5 Comparaisons filtre de Kalman et filtre particulière

Considérons le problème de filtrage scalaire déterminé par les équations linéaires suivantes :

$$\begin{cases} X_n = a X_{n-1} + b \epsilon_n W_n, & n \geq 1 \\ Y_n = c X_n + V_n, & n \geq 0 \end{cases} \quad (4.2)$$

avec $a, b, c \in \mathbb{R}$, X_0 une variable gaussienne de moyenne m_0 et d'écart type σ_0 ; ϵ_n une suite de variables aléatoires indépendantes de Bernoulli de paramètre $p \in]0, 1[$, c'est à dire

$$\forall n \geq 1, \quad \mathbb{P}(\epsilon_n \in dx) = p \delta_1(dx) + (1 - p) \delta_0(dx).$$

une suite de variables indépendantes W_n de même loi gaussienne centrée et d'écart type $\sigma_w > 0$, et enfin une suite de variables indépendantes V_n de même loi gaussienne centrée et d'écart type $\sigma_v > 0$. On conviendra que les variables X_0, V_n, W_n et ϵ_n sont supposées indépendantes.

4.5.1 Approximation linéaire et gaussienne

L'évolution du signal X (et celle de l'observation Y) est bien une équation linéaire, cependant les aléas $\epsilon_n W_n$ ne sont pas gaussiens. En toute rigueur, on ne peut donc pas appliquer le filtre de Kalman-Bucy.

Une première approximation consiste à "remplacer" dans le modèle de filtrage (4.2) les perturbations $\epsilon_n W_n$ par des variables gaussiennes \tilde{W}_n , de même moyenne et de même écart type que les précédentes, c'est à dire

$$\forall n \geq 1, \quad \mathbb{E}(\tilde{W}_n) = 0 \quad \text{et} \quad \sigma_{\tilde{w}}^2 = \mathbb{E}(\tilde{W}_n^2) = \mathbb{E}((\epsilon_n W_n)^2) = p \sigma_w^2.$$

Le problème de filtrage ainsi modifié s'exprime sous la forme linéaire et gaussienne désirée

$$\begin{cases} \tilde{X}_n = a \tilde{X}_{n-1} + b \tilde{W}_n, & n \geq 1 & (\tilde{X}_0 = X_0) \\ \tilde{Y}_n = c \tilde{X}_n + V_n, & n \geq 0. \end{cases}$$

En appliquant le filtre de Kalman à ce problème linéaire gaussien, on obtient une solution approchée des espérances conditionnelles associées au problème initial

$$\mathbb{E}(\tilde{X}_n | \tilde{Y}_0 = y_0, \dots, \tilde{Y}_n = y_n) \sim \mathbb{E}(X_n | Y_0 = y_0, \dots, Y_n = y_n).$$

4.5.2 Filtrage particulaire

L'approximation des distributions conditionnelles

$$\mathbb{P}(X_n \in dx | Y_0, \dots, Y_n)$$

peut aussi se faire en utilisant l'algorithme génétique décrit dans la section 3.5. Dans notre situation, cet algorithme se présente comme une chaîne de Markov à valeurs dans \mathbb{R}^N , où N désigne le paramètre d'approximation et le nombre de particules. On initialise l'algorithme particulaire en simulant un vecteur

$$\xi_0 = (\xi_0^1, \dots, \xi_0^N) \in \mathbb{R}^N$$

formé de N variables indépendantes de même loi que X_0 . Dans notre situation ces variables sont gaussiennes de moyenne m_0 et d'écart type σ_0 . Les étapes de correction et prédiction, caractérisant l'évolution des lois conditionnelles, sont représentées respectivement par les étapes de sélection et de mutation d'un algorithme génétique.

$$\xi_n \xrightarrow{\text{Sélection}} \widehat{\xi}_n \xrightarrow{\text{Mutation}} \xi_{n+1}$$

- **Sélection** : Le mécanisme de sélection est lié à l'observation courante Y_n . Il se réalise en simulant un vecteur

$$\widehat{\xi}_n = (\widehat{\xi}_n^1, \dots, \widehat{\xi}_n^N) \in \mathbb{R}^N$$

formé de N variables indépendantes (conditionnellement en ξ_n) et de même loi

$$\sum_{i=1}^N \frac{e^{-\frac{1}{2\sigma_v^2}(Y_n - c \xi_n^i)^2}}{\sum_{j=1}^N e^{-\frac{1}{2\sigma_v^2}(Y_n - c \xi_n^j)^2}} \delta_{\xi_n^i}.$$

- **Mutation** : Le mécanisme de mutation vise à prédire l'état du signal à l'étape $(n+1)$ avant de recevoir l'observation Y_{n+1} . Dans notre cas on réalise cette transition en posant

$$\xi_{n+1}^i = a \widehat{\xi}_n^i + b \epsilon_n^i W_n^i.$$

où ϵ_n^i désigne une suite de variables indépendantes de même loi que ϵ_n , et W_n^i une suite de variables indépendantes de même loi que W_n .

Le lecteur pourra simuler cet algorithme, et le comparer avec le filtre de Kalman sur la base des valeurs suivantes

$$\begin{array}{lll} m_0 = 0 & \sigma_w \in \{5, 10, 25\} & p \in \{10^{-1}, 10^{-2}, 10^{-3}\} \\ \sigma_0 \in \{1, 5, 10\} & \sigma_v \in \{1, 5, 10\} & N \in \{10^2, 10^3, 10^4\} \\ a = 1/2 & b = 1 & c = 1. \end{array}$$

4.5.3 Filtrage de signaux radar

Les méthodes d'approximation par filtre de Kalman, et particulières décrites dans la section précédente peuvent toutes deux s'étendre à des modèles markovien de signal-observation plus généraux. Le lecteur désireux d'explorer les performances, et les limitations de ces techniques, peut approfondir ces questions sur le modèle simplifié de traitement du signal radar décrit dans la section 2.3.