TD Monte Carlo and simulation

Nicolas Chopin

It is recommended to use Python to do these exercises (but students may use R instead if the "chargé de TD" agrees).

1 RANDU

The LCG (linear congruential generator) RANDU is defined by the following recurence (over the range of integers $1, \ldots, 2^{31} - 1$):

$$x_i = 65539 x_{i-1} \mod 2^{31}$$

and $u_i = x_i / 2^{31} \in (0, 1)$. As we have discussed during the course, RANDU is a notoriously bad generator.

1. Show that (u_i, u_{i-1}, u_{i-2}) always takes values on the union of 15 hyperplanes. (Note: $65539 = 2^{16} + 3$).
2. Implement RANDU, generate 2×10^4 points, and do a scatter plot of u_i as a function of u_{i-1}, for the subset of i such that $0.5 \leq u_{i-1} \leq 0.51$. Discuss.
3. Do a 3D plot of the triples (u_i, u_{i-1}, u_{i-2}) to illustrate the phenomenon discussed in Question 1.
4. Another defect of RANDU is the poor behaviour of the least significant bits. Illustrate this phenomenon.
5. Promise you will never use RANDU to your "chargé(e) de TD".
6. If time permits, you can try to see if the default generator in your favorite language behaves better than RANDU (according to the points above, but you can also think of other ways to test a pseudo-random generator).

2 Laplace and rejection

1. Derive and implement a random generator for the Laplace distribution:

$$p(x) = \frac{1}{2} \exp(-|x|)$$

2. Propose a rejection sampler for the $N(0, 1)$ distribution, using the Laplace distribution as the proposal. Implement it and test it.
3. Give the acceptance rate of the algorithm. Find a way to make the acceptance condition simpler to evaluate.
4. Is it possible to do the opposite? (i.e. to sample from the Laplace distribution, using a rejection sampler based on the normal distribution.)

3 Improved Box-Muller

The improved Box-Muller algorithm samples two unit Gaussian variates as follows:

- Repeat: $U_1, U_2 \sim \mathcal{U}[-1, 1]$ until $U_1^2 + U_2^2 \leq 1$.
- Return $X = U_1 \sqrt{-2(\log S)/S}$ and $Y = U_2 \sqrt{-2(\log S)/S}$, where $S = U_1^2 + U_2^2$.

1. Show that the output of the algorithm has the desired distribution.
2. How can you compare the performance of this algorithm with the standard Box-Muller algorithm? Implement the comparison.

4 Geometric distribution

Propose various algorithms to sample from a geometric distribution. Implement them and discuss which one is best.

5 Control variates, antithetic variables, QMC

Consider $X \sim N_d(0, I_d)$ (standard Gaussian distribution of dimension d). We want to compute the probability that $X \in A$ for a certain $A \subset \mathbb{R}^2$. Consider for instance:

$$A = \{(x_1, \ldots, x_d), \quad \left| \prod_{i=1}^{d} x_i \right| \leq c\}.$$

1. Propose a Monte Carlo algorithm to approximate this probability. What issue occurs when C becomes small?
2. Use the idea of control variates to improve the results of the first question.
3. Can you use antitetic variables in this setting? Discuss.
4. Same question with QMC (quasi-Monte Carlo). Perform experiments to assess the performance gain brought by QMC.

6 Importance sampling

We consider again a standard multivariate Gaussian distribution. We want to approximate this time expectations with respect to the distribution of X conditional on $X \in A$, where A is some measurable subset of $\mathbb{R}^2$. Explain why this conditional distribution has probability density:

$$f(x) = \frac{(2\pi)^{-d/2}}{P(X \in A)} \exp\{-\frac{1}{2} \sum_{i=1}^{d} x_i^2\}.$$

1. A friend of yours proposes the following method: (a) sample X_i from the base distribution (standard Gaussian of dimension d); (b) compute the empirical mean and covariance matrix of the simulated points that fell in A; (c) implement importance sampling based on a Gaussian proposal with mean and variance equal to the moments computed in (b). Is this a good idea? Discuss and/or implement this approach.
2. Try to adapt this solution to get a valid anwser.

7 Metropolis-Hastings

Consider the bivariate distribution with PDF:

$$p(x_1, x_2) = \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left\{-\frac{1}{2(1-\rho^2)}(x_1^2 + x_2^2 - 2\rho x_1 x_2)\right\}$$

with $\rho \in [-1, 1]$.

1. Which distribution do you recognize?
2. Derive the conditional distributions of each component, and implement the corresponding Gibbs sampler. Represent in the plane the contours of the density target, and the evolution of the simulated Markov chain. What happens when $\rho \to 1$?
3. Implement an RWHM algorithm (random walk Hastings-Metropolis) with a Gaussian proposal with variance σ. Start with $\sigma = \tau I_2$, and try to see what happens for different values of τ, especially when ρ is close to 1. Then try with σ proportional to the variance of the target distribution. Comment.

8 Ising

An Ising distribution is a distribution over binary vectors $x = (x_1, \ldots, x_n) \in \{0, 1\}^n$, with probability

$$p(x) \propto \exp\left\{ \alpha \sum_i x_i + \beta \sum_{i \sim j} 1\{x_i = x_j\} \right\}$$

where $i \sim j$ means components i and j are 'neighbours'. For instance, you can assume that these components are laid out on a $k \times k$ square grid, $n = k^2$, and that the neighbours of a given location are the four components which are at distance one (i.e. the one immediately on top, the one immediately below, and so on).

1. Derive the full conditional distribution of a given component x_i. Derive a Gibbs sampler to sample from an Ising distribution.
2. Suggest a method to visualise the mixing of the simulated chain (i.e., how fast the chain is forgetting the past). Compare the mixing of the chain for different values of $\beta > 0$. Discuss.
3. There is a 'critical' value β_0 for β beyond which the mixing of the chain becomes very poor. Try to find this value and give some intuition on what is going on when $\beta > \beta_0$.

9 Hierarchical Bayesian modelling

Consider the Bayesian model briefly discussed in the introduction:

$$\begin{aligned} y_i | \theta_i &\sim \text{Bin}(n_i, \theta_i) \\ \theta_i &\sim \text{Beta}(\alpha, \beta) \\ \alpha, \beta &\sim \text{Gamma}(a, b) \end{aligned}$$

where a, b are fixed hyper-parameters, and the (y_i, n_i), $i = 1, \ldots, I$ are observed.

1. Derive the posterior density of all the unknown variables, and their full conditional distributions.
2. Simulate data from this model (using $\alpha = 1$, $\beta = 10$ as true values) and implement the corresponding Gibbs sampler. Explain how you can assess its performance.
3. What happens if you decide to "freeze" one variable in this Gibbs sampler (i.e. one variable is fixed to a certain value)? What is the invariant distribution in this case? Explain how you can use this remark to "debug" your algorithm.
4. If time permits, try to run the Gibbs sampler for the following dataset https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/VOQCHQ. You can extract a given year (e.g. 2000), and set y_i to be number of voters for one candidate (e.g., Gore) in county i. What is the point of using such a model in this case? (You can check the Wikipedia page on "small area estimation" to get an idea.)

10 Gaussian Mixtures

We consider a simplified Gaussian mixture model, with K Gaussian components $N(\mu_k, 1)$, each with weight (prior probability) $1/K$. The density of the observations is therefore:

$$\frac{1}{K}\sum_{k=1}^{K}\varphi(y;\mu_k,1)$$

where $\varphi(y;\mu_k,1)$ is the pdf of a $N(\mu_k,1)$ variable: $\varphi(y;\mu,\sigma^2) = (2\pi)^{-1/2}\exp\left\{-(y-\mu)^2/2\sigma^2\right\}$.

1. Simulate a dataset from this model. You may take $n = 100$, $K = 2$, and two well separated values for μ_1 and μ_2.
2. We consider a $N(0, 10^2)$ prior for each μ_i (independently). Implement a Gibbs sampler that targets the posterior distribution of (μ_1, μ_2) given the observations $y_{1:n}$. Hint: you must introduce latent variables z_i, where z_i is the Gaussian component that is associated to each individual observation y_i.
3. The likelihood of the model is permutation-invariant (e.g. when you swap μ_1 with μ_2 when $K = 2$). Do a scatter plot of the simulated pairs (μ_1, μ_2) (obtained from the Gibbs sampler of the previous question). Try different settings and discuss.
4. Generalise to unknown variances for the Gaussian components (use an inverse-gaussian prior), then unknown prior probabilities for the different components (for $K = 2$ take a Beta distribution; then for $K \geq 3$, you may consider a Dirichlet distribution.)

11 Discretization error in option pricing

1. Let (W_t) be a Wiener process. Derive the distribution of W_t conditionally on W_a and W_b for $a \leq t \leq b$.
2. Let $S(t)$ be the Black-Scholes process defined by

$$S(t) = \exp\{(\mu - \sigma^2/2)t + \sigma W_t\}$$

or equivalently:

$$dS(t) = \mu S(t)\,dt + \sigma S(t) dW_t.$$

Use Monte Carlo to approximate the price of a European option defined by:

$$V = \mathbb{E}\left[e^{-rT}\left(K - S(T)\right)^+\right]$$

for $T = 1$, $r = 0.02$, $\mu = 0.5$, $\sigma = 0.5$, $K = 2$.

3. The BS process is now treated as an unknown process which cannot be simulated exactly (at given times), and that must therefore be discretized. Use question 1 to build an iterative algorithm that gradually reduces the discretization step, until the discretization error seems small. Hint: at step k, the discretization step will be $T2^{-k-k_0}$, and the goal is to recycle the simulations of step $k-1$ by inserting 2^{k+k_0-1} new points. You may compare the confidence intervals on V obtained at iteration k and $k-1$, and decide accordingly whether you must continue, stop, or possibly increase the number of simulations to allow for a finer comparison.

12 Cross-Entropy Method

The Rosenbrock function in dimension d is defined as follows:

$$S(x) = \sum_{i=1}^{d} 100(x_{i+1} - x_i)^2 + (x_i - 1)^2$$

It admits as global minimum the point $x^\star = (1, \ldots, 1)$. It's a popular benchmark in optimization. Propose a CE (Cross-Entropy) algorithm to obtain the global minimum of any function $S : \mathbb{R}^d \to \mathbb{R}$, program it and apply it to the Rosenbrock function, for different values of d.

13 ABC

A network has n individuals; X_{ij} is a binary variable which is one when individuals i are j connected, and zero otherwise. A possible model for this network is

$$p(x|\theta) = \frac{1}{Z(\theta)} \exp\left\{\theta^T S(x)\right\}$$

where $x = (x_{ij})_{i,j=1,\ldots,n}$, and S is some summary of the network x, e.g. $S(x)$ is bi-dimensional, the first component is $\sum x_{ij}$, the second component is the number of individuals with 3 connections or more.

1. Propose and implement an algorithm to simulate from the model for a given θ. Use it to simulate a network of size $n = 20$. (Several answers may be considered.)
2. Assume θ is unknown and must be estimated. Propose an ABC algorithm to estimate θ based on one or several networks. (You may take a uniform prior for each component of θ).