Stochastic Simulation

Autumn Semester 2024

Prof. Fabio Nobile *Assistant: Matteo Raviola*

Lab 12 – 5 December 2024

Markov Chain Monte Carlo

Exercise 1

Consider the Random Walk Metropolis–Hastings (RWMH) algorithm with proposal density $q(x, y) = g_\sigma(y - x)$ and target density $f\colon \mathbb{R} \to \mathbb{R}^+$. Let g_σ denote the density of the $\mathcal{N}(0, \sigma^2)$ distribution and suppose that

$$f(y) = \frac{1}{Z} \exp\left[-\left(\frac{1}{4}y^4 - \frac{1}{2}y^2 + \frac{1}{4}\right)\right],$$

where Z is such that f is a PDF on $\mathcal{X} = \mathbb{R}$. Suppose we wish to estimate $\mu = \mathbb{E}_f(\phi)$ for a suitable function $\phi\colon \mathbb{R} \to \mathbb{R}$. Let $\hat{\mu}_n^{\mathrm{MH}}$ be the estimator for μ based on the Markov chain of length n generated by the RWMH algorithm. Derive asymptotic confidence intervals for μ at probability level α using the CLT for Metropolis–Hastings Markov Chains. Within your simulations, estimate[1] this confidence interval and stop the Markov chain once the half-length of the interval is smaller than a given tolerance $\tau > 0$. Implement the following heuristics to estimate the required *time average variance constant* and compare their performance for different functions ϕ; namely $\phi(x) = x^p, p \in \mathbb{N}$. The time average variance constant is the asymptotic variance introduced in Theorem 8.10 of the lecture notes.

1. *Initial positive sequence estimator:*

$$\tilde{\sigma}^2 \approx \hat{\sigma}^2_{\mathrm{pos},n} := -\hat{c}_n(0) + 2\sum_{k=1}^{K}\big(\hat{c}_n(2k) + \hat{c}_n(2k+1)\big),$$

   where K is the largest integer such that $\hat{c}_n(2k) + \hat{c}_n(2k+1) > 0$ for all $k = 1, \dots, K$. Here,

$$\hat{c}_n(j) := \frac{1}{n}\sum_{i=1}^{n-j}\Big(\phi(X_i) - \hat{\mu}_n^{\mathrm{MH}}\Big)\Big(\phi(X_{i+j}) - \hat{\mu}_n^{\mathrm{MH}}\Big)$$

   is an appropriate covariance estimator in this context.

2. *Initial monotone sequence estimator:*

$$\tilde{\sigma}^2 \approx \hat{\sigma}^2_{\mathrm{mon},n} := -\hat{c}_n(0) + 2\sum_{k=1}^{K}\min_{1\le j\le k}\big\{\hat{c}_n(2j) + \hat{c}_n(2j+1)\big\},$$

   where K and $\hat{c}_n$ are as for the initial positive sequence estimator above.

[1] The estimation has to be carried out on-the-fly, that is while the Markov chain evolves.

3. *Batch means estimator:* Suppose the Markov chain is $X_1, \ldots, X_n$ at iteration n. Divide these n values into $N_b \in \mathbb{N}$ batches, each of length $N_\ell = n/N_b$. A typical decomposition is $N_\ell = n^{1-a}$ and $N_b = n^a$ for $a \in [0,1]$, for example $a = 0.5$, modulo integer rounding. Let

$$\hat{\mu}_i = \frac{1}{N_\ell} \sum_{j=(i-1)N_\ell+1}^{iN_\ell} \phi(X_j) \ , \quad i = 1, \ldots, N_b \ ,$$

be the sample mean of the i-th batch. For N_ℓ sufficiently large, one can consider the batch means $\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_{N_b}$ to be approximately mutually independent. Consequently, one can estimate the time average variance constant σ^2 by the sample variance estimator for independent realizations:

$$\tilde{\sigma}^2 \approx \hat{\sigma}^2_{\mathrm{BM},n} := \frac{n}{N_b} \frac{1}{N_b - 1} \sum_{i=1}^{N_b} \left(\hat{\mu}_i - \hat{\mu}_n^{\mathrm{MH}}\right)^2 .$$

In addition, instead of using all of the points in the Markov chain, one can as well discard a burn-in time B and replace $\sum_{k=1}$ with $\sum_{k=B}$ in the above estimators. Experiment with the effects of burn-in time on the above asymptotic variance estimators.

Exercise 2

Consider the following probability density function in $\mathbb{R}^2$

$$f(\boldsymbol{x}) = \frac{1}{Z}\left[\exp\left\{-(1-x_1)^2 - (x_2 - x_1^2)^2\right\} + \exp\left\{-(x_1+1)^2 - (x_2 + 3 + x_1^2)^2\right\}\right], \quad \boldsymbol{x} = (x_1, x_2)^T \in \mathbb{R}^2,$$

where Z is the (unknown) normalization constant. To generate samples from f, we consider Metropolis-Hastings (MH) type MCMC algorithms. Let us denote with $p(\cdot; \boldsymbol{\mu}, \Sigma)$ [2] the pdf of a $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ multivariate Gaussian random variable and by

- K_1 the Markov kernel associated to an *Independent Sampler* MH algorithm with proposal density $q_1(\boldsymbol{x}, \mathbf{y}) = \frac{1}{2} p(\mathbf{y}; (1,1)^T, 0.5\, I_{2\times 2}) + \frac{1}{2} p(\mathbf{y}; (-1,-3)^T, 0.5\, I_{2\times 2})$
- K_2 the Markov kernel associated to a *Random Walk* MH algorithm with proposal density $q_2(\boldsymbol{x}, \mathbf{y}) = p(\mathbf{y}; \boldsymbol{x}, \sigma^2\, I_{2\times 2})$, where $\sigma = 0.15$.

1. Write the explicit expression of the densities corresponding to the Markov kernels K_1 and K_2.
2. Consider now the kernel $K(\omega) = \omega K_1 + (1-\omega) K_2$, with $\omega \in [0,1]$. Show that $K(\omega)$ is a reversible Markov kernel that has f as invariant distribution.
3. Implement a MCMC algorithm that uses the Markov kernel $K(\omega)$ to estimate $\mathbb{E}_f[\|\mathbf{X}\|^2]$, with $\mathbf{X} = (X_1, X_2)^T \sim f(\cdot)$. Include the usual MCMC diagnostic plots in your experiment and describe how you would chose the sample size (length of the chain) to guarantee an error on the computation of $\mathbb{E}_f[\|\mathbf{X}\|^2]$ smaller than $\varepsilon = 1$ with confidence at least .95. Estimate also the expected acceptance rate $\chi(\omega)$ of the implemented algorithm. Try at least two different values for ω.

[2] $p(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{\det(2\pi\Sigma)}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x}-\boldsymbol{\mu})}$

4. Write a closed-form formula for the expected acceptance rate $\chi(\omega)$ from the expression of the kernel $K(\omega)$. How could one use the results in the previous point to find the value of ω that leads to a target acceptance rate, say $\chi(\omega) = 0.4$?

Hint: You can use the following `python` commands to plot your results:

```
import statsmodels.graphics.tsaplots as sm
import seaborn as sbn
.
.
.
sbn.kdeplot(x)   # To plot emperical density of samples x
sm.plot_acf(QoI)  # To plot autocorrelation plot of a quantity of interest QoI
```