9.1 Parameterised model

Consider now a family of models, determined by a parameter $\theta \in \mathsf{T} = \mathbb{R}^{d_\theta}$:

$$p_u^{(\theta)}(x_{1:T}) = M_1^{(\theta)}(x_1) G_1^{(\theta)}(x_1) \prod_{t=2}^{T} M_t^{(\theta)}(x_t \mid x_{1:t-1}) G_t^{(\theta)}(x_{1:t}).$$

and suppose that $\mathrm{pr}(\theta) \geq 0$ is a function such that

$$p_u(\theta, x_{1:T}) = \mathrm{pr}(\theta) p_u^{(\theta)}(x_{1:T})$$

determines an unnormalised probability distribution $p(\theta, x_{1:T}) \propto p_u(\theta, x_{1:T})$ on $\mathsf{T} \times \mathbb{S}^T$.

Remark 9.1. In particular, if $p_u^{(\theta)}(x_{1:T})$ correspond to a parameterised SSM as in (21), that is,

$$p_u^{(\theta)}(x_{1:T}) = m_1^{(\theta)}(x_1) g_1^{(\theta)}(x_1) \prod_{t=2}^{T} m_t^{(\theta)}(x_{t-1}, x_t) g_t^{(\theta)}(x_t, y_t),$$

(cf. Remark 8.8), and pr is the prior of the parameters θ, then $p_u(\theta, x_{1:T})$ corresponds to the conditional distribution $(\theta, X_{1:T}) \mid Y_{1:T} = y_{1:T}$. This is what we care about if we are interested in (full) Bayesian time-series analysis using SSMs...

9.2 Particle marginal Metropolis-Hastings algorithm

Suppose that $n \in \mathbb{N}$ is fixed, and that $q(\theta, \theta')$ is a Metropolis-Hastings proposal on T.

Algorithm 9.2 (Particle marginal Metropolis-Hastings)**.** Let $\Theta_0 \in \mathsf{T}$, and let $(V_0^{(1:n)}, \mathbf{X}_0^{(1:n)})$ be the output of PF Algorithm 8.15 with $(n,\ M_{1:T}^{(\Theta_0)},\ G_{1:T}^{(\Theta_0)})$. For $k = 1, 2, \ldots, N$, iterate:

(i) Sample $\hat{\Theta}_k \sim q(\Theta_{k-1}, \cdot)$.

(ii) Run PF Algorithm 8.15 with $(n,\ M_{1:T}^{(\hat{\Theta}_k)},\ G_{1:T}^{(\hat{\Theta}_k)})$, and call its output $(\hat{V}_k^{(1:n)}, \hat{\mathbf{X}}_k^{(1:n)})$.

(iii) With probability

$$\min\left\{1, \frac{\mathrm{pr}(\hat{\Theta}_k) q(\hat{\Theta}_k, \Theta_{k-1}) \sum_{i=1}^n \hat{V}_k^{(i)}}{\mathrm{pr}(\Theta_{k-1}) q(\Theta_{k-1}, \hat{\Theta}_k) \sum_{j=1}^n V_{k-1}^{(j)}}\right\},$$

accept and set $(\Theta_k, V_k^{(1:n)}, \mathbf{X}_k^{(1:n)}) \leftarrow (\hat{\Theta}_k, \hat{V}_k^{(1:n)}, \hat{\mathbf{X}}_k^{(1:n)})$; otherwise reject and set $(\Theta_k, V_k^{(1:n)}, \mathbf{X}_k^{(1:n)}) \leftarrow (\Theta_{k-1}, V_{k-1}^{(1:n)}, \mathbf{X}_{k-1}^{(1:n)})$.

Report the following approximation of $\mathbb{E}_p[f(\Theta, X_{1:T})]$:

$$I_{\mathrm{PMMH}}^{(N,n)}(f) := \frac{1}{N} \sum_{k=1}^{N} \frac{\sum_{i=1}^n V_k^{(i)} f(\Theta_k, \mathbf{X}_k^{(i)})}{\sum_{j=1}^n V_k^{(j)}}.$$

Theorem 9.3. *Suppose that $q(\theta, \theta') > 0$ for all $\theta, \theta \in \mathsf{T}$. Then, for any fixed $n \in \mathbb{N}$,*

$$I^{(N,n)}_{\mathrm{PMMH}}(f) \xrightarrow{N\to\infty} \mathbb{E}_p[f(\Theta, X_{1:T})] \qquad a.s.,$$

whenever the expectation is finite.

Proof. (**) Let $Q_\theta(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1})$ stand for the distribution of *all* random variables $X^{(i)}_t$ and $A^{(i)}_t$ generated in Algorithm 8.15 with $(n, M^{(\theta)}_{1:T}, G^{(\theta)}_{1:T})$, and let $v^{(k)}(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1})$ and $\mathbf{x}^{(k)}(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1})$ stand for how the outputs $V^{(k)}$ and $\mathbf{X}^{(k)}$ are determined from $X^{(1:n)}_{1:T}$ and $A^{(1:n)}_{1:T-1}$. Define the following unnormalised distribution (sic!)

$$\pi_u(\theta, x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1}) = \mathrm{pr}(\theta) Q_\theta(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1}) \sum_{k=1}^{n} v^{(k)}(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1}),$$

then Algorithm 9.9 may be seen as a Metropolis-Hastings with target $\pi \propto \pi_u$ and proposal $\tilde{q}(\theta, x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1}; \hat{\theta}, \hat{x}^{(1:n)}_{1:T}, \hat{a}^{(1:n)}_{1:T-1}) = q(\theta, \hat{\theta}) Q_{\hat{\theta}}(\hat{x}^{(1:n)}_{1:T}, \hat{a}^{(1:n)}_{1:T-1})$.

Theorem 8.22 implies that for any $\varphi : \mathbb{S}^T \to \mathbb{R}$ such that the integral below is finite,

$$\sum_{a^{(1:n)}_{1:T-1}} \int Q_\theta(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1}) \Big(\sum_{i=1}^{n} v^{(i)}(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1}) \varphi\big(\mathbf{x}^{(i)}(x^{(1:n)}_{1:T}, a^{(1:n)}_{1:T-1})\big) \Big) \mathrm{d}x^{(1:n)}_{1:T}$$
$$= \int p^{(\theta)}_u(x_{1:T}) \varphi(x_{1:T}) \mathrm{d}x_{1:T}.$$

This implies that for any function $f : \mathsf{T} \times \mathbb{S}^T \to \mathbb{R}$,

$$\mathbb{E}_\pi \left[\frac{\sum_{i=1}^n v^{(i)}(\mathbf{X}^{(1:n)}_{1:T}, A^{(1:n)}_{1:T-1}) f(\Theta, \mathbf{x}^{(i)}(\mathbf{X}^{(1:n)}_{1:T}, A^{(1:n)}_{1:T-1}))}{\sum_{j=1}^n v^{(j)}(\mathbf{X}^{(1:n)}_{1:T}, A^{(1:n)}_{1:T-1})} \right] = \mathbb{E}_p[f(\Theta, X_{1:T})].$$

The proof is complete once we are convinced that the Markov chain $(\Theta_i, X^{(1:n)}_{1:T}, A^{(1:n)}_{1:T-1})$, and consequently $V^{(1:n)}_i, \mathbf{X}^{(1:n)}_i)_{i\geq 1}$, is Harris, which follows because it is π-irreducible Metropolis-Hastings. □

Remark 9.4 (*). If we are only interested in the variable Θ in p, the PMMH Algorithm 8.15 may be seen as an instance of a so-called *pseudo-marginal* Metropolis-Hastings algorithm [1, 14].

9.3 Conditional particle filter (*)

The PMMH is a relatively simple combination of the PF and Metropolis-Hastings. It can, however, get 'stuck' (many rejects), when $\sum_j V^{(j)}_{k-1}$ gets over-estimated (unusually high value). The paper [3] contained also another scheme, which has better scalability properties wrt. T. It is based on a modified *conditional* particle filter (CPF) algorithm.

Algorithm 9.5. CPF$(n, M^{(\theta)}_{1:T}, G^{(\theta)}_{1:T}, X^*_{1:T})$

(i) Set $X_1^{(1)} = X_1^*$ and sample $X_1^{(2:n)} \overset{\text{i.i.d.}}{\sim} M_1$. Set $\mathbf{X}_1^{(1:n)} = X_1^{(1:n)}$.

(ii) Calculate $\omega_1^{(i)} := G_1(\mathbf{X}_1^{(i)})$ and set $\bar{\omega}_1^{(i)} := \omega_1^{(i)}/\omega_1^*$ where $\omega_1^* = \sum_{j=1}^n \omega_1^{(j)}$.

For $t = 2, \ldots, T$, do:

(iii) Sample $A_{t-1}^{(2:n)}$ independently with $\mathbb{P}(A_{t-1}^{(i)} = j) = \bar{\omega}_{t-1}^{(j)}$, $j \in 1{:}n$.

(iv) Set $X_t^{(1)} = X_t^*$ and sample $X_t^{(i)} \sim M_t(\,\cdot \mid \mathbf{X}_{t-1}^{(A_{t-1}^{(i)})})$ for $i = 2{:}n$.

(v) Set $\mathbf{X}_t^{(1)} = (\mathbf{X}_{t-1}^{(1)}, X_t^*)$ and $\mathbf{X}_t^{(i)} = (\mathbf{X}_{t-1}^{(A_{t-1}^{(i)})}, X_t^{(i)})$ for $i = 2{:}n$.

(vi) Calculate $\omega_t^{(i)} := G_t(\mathbf{X}_t^{(i)})$ and set $\bar{\omega}_t^{(i)} := \omega_t^{(i)}/\omega_t^*$ where $\omega_t^* = \sum_{j=1}^n \omega_t^{(j)}$.

Draw $B \sim \text{Categorical}(\bar{\omega}_T^{(1:n)})$ and output $\mathbf{X}_T^{(B)}$.

Remark 9.6. The CPF defines a *Markov transition* in the trajectory space $\mathbb{S}^T$. It turns out that the transition is reversible with respect to $p^{(\theta)} \propto p_u^{(\theta)}$, again thanks to Theorem 8.22.

Remark 9.7. When $M_t(x_t \mid x_{1:t-1}) = M_t(x_t \mid x_{t-1})$ and $G_t(x_{1.t}) = G_t(x_{t-1:t})$, the CPF may be substantially enhanced by applying it together with so-called *backward sampling* [32] (or the equivalent ancestor sampling [15]). That is, instead of selecting one of $\mathbf{X}_T^{(1:n)}$, the output is 'reselected' among all particles $X_{1:T}^{(1:n)}$ as follows: $(X_1^{(B_1)}, \ldots, X_{T-1}^{(B_{T-1})}, X_T^{(B_T)})$, where $B_T = B$ and for $t = T-1, \ldots, 1$:

$$\mathbb{P}(B_t = i \mid B_{t+1} = j) \propto \omega_t^{(i)} M_{t+1}(X_{t+1}^{(j)} \mid X_t^{(i)}) G_{t+1}(X_t^{(i)}, X_{t+1}^{(j)}). \tag{25}$$

The backward sampling version of the CPF is also $p^{(\theta)}$-reversible [6]. (Note also that if $G_t(x_{t-1:t}) = G_t(x_t)$, then the term $G_{t+1}(\,\cdot\,)$ vanishes from (25).)

Remark 9.8. When using the CPF n has to increase in T linearly $n = O(T)$, but with the backward sampling modification, n need not be increased wrt. T [cf. 13].

9.4 Particle Gibbs (*)

Algorithm 9.9 (Particle Gibbs)**.** Let $\Theta_0 \in \mathsf{T}$ and $\mathbf{X}_0 \in \mathsf{Sp}^T$ such that $p_u(\Theta_0, \mathbf{X}_0) > 0$.

For $k = 1, 2, \ldots, N$, iterate:

(i) $\mathbf{X}_k \leftarrow \text{CPF}(n, M_{1:T}^{(\Theta_{k-1})}, G_{1:T}^{(\Theta_{k-1})}, \mathbf{X}_{k-1})$.

(ii) Sample $\hat{\Theta}_k \sim q(\Theta_{k-1}, \,\cdot\,)$, and with probability

$$\min\left\{1, \frac{p_u(\hat{\Theta}_k, \mathbf{X}_k) q(\hat{\Theta}_k, \Theta_{k-1})}{p_u(\Theta_{k-1}, \mathbf{X}_k) q(\Theta_{k-1}, \hat{\Theta}_k)}\right\},$$

accept and set $\Theta_k \leftarrow \hat{\Theta}_k$; otherwise reject and set $\Theta_k \leftarrow \Theta_{k-1}$.

Report the following approximation of $\mathbb{E}_p[f(\Theta, X_{1:T})]$:

$$I_{\text{PG}}^{(N,n)}(f) := \frac{1}{N} \sum_{k=1}^{N} f(\Theta_k, \mathbf{X}_k).$$

Theorem 9.10. *The particle Gibbs defines a Markov transition which leaves p invariant.*

Proof. Step (i) is a component-wise update of $\mathbf{X}_{k-1} \to \mathbf{X}_k$ by the CPF that leaves the conditional $\propto p_u^{(\Theta_{k-1})}$ invariant, and the step (ii) is a Metropolis-within-Gibbs step. □

Remark 9.11. Consider the PMMH output, and sample $I_k \sim \text{Categorical}(W_k^{(1:n)})$, where $W_k^{(i)} = V_k^{(i)}/(\sum_{j=1}^n V_k^{(j)})$, then we may also use

$$\hat{I}_{\text{PMMH}}^{(N,n)}(f) := \frac{1}{N}\sum_{k=1}^{N} f(\Theta_k, \mathbf{X}_k^{(I_k)}),$$

which remains consistent, but it worse in terms of variance.

Analogously, it is direct to use a more 'refined' estimator in the PG, where the selection of output (sampling of B in Algorihm 9.5) is 'Rao-Blackwellised'...

Remark 9.12. Some authors refer also the CPF as 'particle Gibbs', but the terminology here follows the terminology in the original paper [3].

References

[1] C. Andrieu and G. O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *Ann. Statist.*, 37(2):697–725, 2009.

[2] C. Andrieu and J. Thoms. A tutorial on adaptive MCMC. *Statist. Comput.*, 18 (4):343–373, Dec. 2008.

[3] C. Andrieu, A. Doucet, and R. Holenstein. Particle Markov chain Monte Carlo methods. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 72(3):269–342, 2010. (with discussion).

[4] O. Cappé, E. Moulines, and T. Rydén. *Inference in Hidden Markov Models*. Springer, 2005.

[5] B. Carpenter, A. Gelman, M. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. A. Brubaker, J. Guo, P. Li, and A. Riddell. Stan: A probabilistic programming language. *J. Stat. Softw.*, 20:1–37, 2016.

[6] N. Chopin and S. S. Singh. On particle Gibbs sampling. *Bernoulli*, 21(3):1855–1883, 2015.

[7] P. Del Moral. *Feynman-Kac Formulae:: Genealogical and Intercating Particle Systems with Applications*. Springer, New York, 2004.

[8] A. Durmus, E. Moulines, and E. Saksman. On the convergence of Hamiltonian Monte Carlo. *Ann. Statist.*, to appear.

[9] J. M. Flegal and G. L. Jones. Batch means and spectral variance estimators in Markov chain Monte Carlo. *Ann. Statist.*, 38(2):1034–1070, 2010.

[10] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, 1993.

[11] H. Haario, E. Saksman, and J. Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242, 2001.

[12] A. M. Johansen, L. Evers, and N. Whiteley. Monte Carlo methods. Lecture notes, University of Bristol, 2010.

[13] A. Lee, S. S. Singh, and M. Vihola. Coupled conditional backward sampling particle filter. *Ann. Statist.*, to appear.

[14] L. Lin, K. Liu, and J. Sloan. A noisy Monte Carlo algorithm. *Phys. Rev. D*, 61(7): 074505, 2000.
[15] F. Lindsten, M. I. Jordan, and T. B. Schön. Particle Gibbs with ancestor sampling. *J. Mach. Learn. Res.*, 15(1):2145–2184, 2014.
[16] M. Matsumoto and T. Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 8(1):3–30, 1998.
[17] S. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Cambridge University Press, second edition, 2009. ISBN 978-0-521-73182-9.
[18] R. M. Neal. MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, editors, *Handbook of Markov Chain Monte Carlo*, volume 2. CRC Press New York, NY, 2011.
[19] G. Nicholls. Part A Simulation and statistical programming. Lecture notes, University of Oxford, 2015.
[20] E. Nummelin. MC's for MCMC'ists. *Int. Statist. Rev.*, 70(2):215–240, 2002.
[21] A. B. Owen. *Monte Carlo theory, methods and examples*. 2013.
[22] A. Penttinen. MATS442 Stokastinen simulointi. Lecture notes, University of Jyväskylä, 2010. (In Finnish).
[23] P. H. Peskun. Optimum Monte-Carlo sampling using Markov chains. *Biometrika*, 60(3):607–612, 1973.
[24] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, New York, 1999.
[25] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, New York, second edition, 2004.
[26] G. O. Roberts and J. S. Rosenthal. Harris recurrence of Metropolis-within-Gibbs and trans-dimensional Markov chains. *Ann. Appl. Probab.*, 16(4):2123–2139, 2006.
[27] G. O. Roberts, A. Gelman, and W. R. Gilks. Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.*, 7(1):110–120, 1997.
[28] D. J. Spiegelhalter, A. Thomas, N. G. Best, W. R. Gilks, and D. Lunn. BUGS: Bayesian inference using Gibbs sampling, 1996–2008. URL `http://www.mrc-bsu.cam.ac.uk/bugs/`.
[29] L. Tierney. Markov chains for exploring posterior distributions. *Ann. Statist.*, 22 (4):1701–1728, 1994.
[30] L. Tierney. A note on Metropolis-Hastings kernels for general state spaces. *Ann. Appl. Probab.*, 8(1):1–9, 1998.
[31] M. Vihola. Robust adaptive Metropolis algorithm with coerced acceptance rate. *Statist. Comput.*, 22(5):997–1008, 2012.
[32] N. Whiteley. Discussion on Particle Markov chain Monte Carlo methods. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 72(3):306–307, 2010.