Bayesian Estimation of Deer Population Dynamics Using Hamiltonian Monte Carlo Algorithm

Tomoro Kimura, Kazuyuki Nakamura
1Graduate School of Advanced Mathematical Sciences, Meiji University
4-21-1 Nakano, Nakano-ku, Tokyo, Japan
2JST, PRESTO
4-1-8 Honcho, Kawaguchi, Saitama, Japan

E-mail: kmr0207tomoro@outlook.com, knaka@meiji.ac.jp

Abstract

For the estimation of deer population dynamics, Bayesian method using Gibbs sampler which is one of the Markov Chain Monte Carlo (MCMC) method has been adopted in the previous studies although the method takes much time for calculation. In this study, we introduce Hamiltonian Monte Carlo (HMC) method as a new approach which is more efficient than Gibbs sampling. To verify our proposal method contributes to deer population managements, we modified the model of a previous study to apply Hamiltonian dynamics and conducted numerical experiments by comparing the results which are estimated in two methods with using simulation based data. As a result, we confirmed HMC method is more time-efficient than Gibbs sampler because the number of samples abandoned for after procedures in HMC method was much less than in Gibbs sampler. Therefore, we conclude that the estimation using HMC is valuable for the estimation of deer population dynamics.

1 Introduction

For the past several decades, overabundant deer have caused problems across Japan [1]. Due to negative affection of overabundance, natural scenery such as natural vegetation and ecosystem changed seriously. Hence, controlling deer population is very important in wildlife management. It is necessary to estimate deer population dynamics and the factors that influence their fluctuation for efficient control.

The state-space model is suitable for modeling the population dynamics and provides reasonable estimation [2, 3, 4]. Deer population of each year fluctuates with uncertainty deriving from various natural factors and the data used for estimation include observation errors. The state-space model represents these characteristics of the dynamics. With this framework, deer population dynamics can be described temporally and spatially.

However, it is not easy to estimate parameters because of complexity of the model. In the past studies, Markov Chain Monte Carlo (MCMC) method has been adopted for the estimation [2, 3, 4]. MCMC is a method to estimate parameters existing in background of observed data by generating random samples from probability distributions of parameters with using Markov chain. There are several ways for implementing MCMC method. Especially, Gibbs sampler has been used in the previous researches [2, 3, 4] because of the practical utility that it can be handled by several probabilistic programming languages; WinBUGS [5] and JAGS [6].

Although Gibbs sampler enables us to estimate the population dynamics, it often takes much time to generate enough number of samples [7]. This method generates Markov chains by iteration of sampling from conditional distributions fixed all variables without a variable. The large numbers of Markov chain steps are consequently required for dynamical transitions of samples on the state space because proposed newer samples correlate with the samples of previous steps. This static behavior lowers time-efficiency in the estimation. We consider Gibbs sampler is not optimum method to estimate deer population dynamics with the model including complex distributions and the highly dimensional parameters. Therefore, introducing more efficient method for estimation can contribute to ecological managements.

In this study, we adopt a new method instead of Gibbs sampling, which is called Hamiltonian Monte Carlo (HMC) algorithm. HMC algorithm is a hybrid method of Hamiltonian dynamics and Metropolis algorithm [7]. The samples from posterior distributions can be obtained efficiently. We can handle this method by the probabilistic programming languages, Stan [8]. Stan is multi-platform and has been developed actively for this several years though WinBUGS basically operates with Windows OS only and stopped being developed.

Purpose of the study is to verify the applicability of HMC method as the way of the population estimation.
We rebuild some parts of the model in previous study to apply HMC method. For evaluating efficiency of the methods, we generate simulation data and estimate parameters which explain those data with using HMC and Gibbs sampler methods respectively. We illustrate research area for our simulation by some square shaped meshes as shown graphically in Figure 1 though there are some arguments about how research area divide.

Fig. 1: Research area divided into 10 × 10 meshes.

2 Method

2.1 Model

We constructed the model in this study by modifying the model of Iijima et al. (2013) [3] because HMC method postulates that all parameters are continuous to calculate Hamiltonian numerically. Although summing out is one of the basic solutions to deal with the model including discrete variables [9], in this model, it causes combinational explosion due to wide state space and large parameter size. We considered rewriting the model without discrete parameters is the best solution.

Our model follows the framework of the state-space model, inheriting the previous models. The state-space model is composed of a state process and an observation process. The state process explains fluctuation of latent deer population, and observation process describes the relationships between observed data and latent population dynamics regarded as true state.

State Process

Deer population dynamics after the second year can be expressed as follows:

\[ x_{t,m} \sim N(\mu_{t-1,m}, \sigma^2_x) \]  \hspace{1cm} (1)

\[ \mu_{t-1,m} = x_{t-1,m} + \log(1 - fhc_{t-1,m}) + r_{log,m} \]  \hspace{1cm} (2)

where \( x_{t,m} \) is the logarithmic scale deer abundance in year \( t \) (\( t = 1, 2, \cdots \)) in mesh \( m \) (\( m = 1, 2, \cdots \)), \( fhc_{t-1,m} \) is the probability that a deer is hunted and culled in year \( t-1 \) in mesh \( m \), and \( r_{log,m} \) is the logarithmic scale mean population growth rate in mesh \( m \) during the research period each year. A non-informative uniform prior distribution is assigned for the prior distribution of \( \sigma_x \).

The logarithmic scale deer abundance in the first year \((x_{1,m})\) were determined as follows:

\[ x_{1,m} \sim N\left(\frac{a_m \cdot x_{1,\text{init}}}{a_m +}, \frac{\sigma_{x,\text{init}}^2}{a_m +} \right) \]  \hspace{1cm} (3)

The prior distribution of \( x_{1,m} \) is formed by a Gaussian conditional autoregressive (CAR) model. The Gaussian CAR model describes the spatial relationship such as spatially closed meshes that can have similar characters [10]. It consists of \( a_m \) and \( a_m + \). The row vector \( a_m \) stores the integer values 1 or 0 and indicates the neighbors of mesh \( m \). If a mesh is sharing the same boundary with mesh \( m \), it is a neighbor mesh. \( a_m + \) is the number of the neighbors in mesh \( m \). In fine, the mean of Gaussian CAR model is determined by the mean of the neighbors of each mesh. For the prior distribution of \( \sigma_{x,\text{init}} \), a non-informative uniform prior distribution is assigned.

The logarithmic scale mean population growth rate \((r_{log,m})\) was modeled as follows:

\[ r_{log,m} \sim N(\text{mean}, \sigma_r^2) \]  \hspace{1cm} (4)

We simplified this process from original model due to the difference of the purpose of study. We described the prior distribution of \( r_{log,m} \) by using \( r_{\text{mean}} \) and \( \sigma_r \) and assigned a non-informative Gaussian and a uniform prior distribution for the prior distribution of \( r_{\text{mean}} \) and \( \sigma_r \), respectively.

Observation Process

Pellet group density data were modeled as follows:

\[ PG_{t,m} \sim \text{Poisson}(\lambda_{PG,t,m}) \]  \hspace{1cm} (5)

\[ \lambda_{PG,t,m} = \exp(b_{PG} + x_{t,m} + \varepsilon_{PG,t,m}) \text{Route}_{t,m} \]  \hspace{1cm} (6)

\[ \varepsilon_{PG,t,m} \sim N(0, \sigma_{PG}^2) \]  \hspace{1cm} (7)

Where \( PG_{t,m} \) is the pellet group density data in year \( t \) in mesh \( m \), \( b_{PG} \) is the rate conversion parameter of \( x_{t,m} \) to \( PG_{t,m} \), \( \varepsilon_{PG,t,m} \) is a Gaussian error term of pellet group density in year \( t \) in mesh \( m \), \( \text{Route}_{t,m} \) is the length of the census route in year \( t \) in mesh \( m \). In our simulation study, we set the same value for \( \text{Route}_{t,m} = 1 \). For the prior distribution of \( \sigma_{PG} \), a non-informative uniform prior distribution is assigned.

SPUE data were modeled as follows:

\[ SD_{t,m} \sim \text{Poisson}(\lambda_{SD,t,m}) \]  \hspace{1cm} (8)

\[ \lambda_{SD,t,m} = \exp(b_{SD} + x_{t,m} + \varepsilon_{SD,t,m}) \text{Effort}_{t,m} \]  \hspace{1cm} (9)

\[ \varepsilon_{SD,t,m} \sim N(0, \sigma_{SD}^2) \]  \hspace{1cm} (10)

Where \( SD_{t,m} \) is the number of deer seen in year \( t \) in mesh \( m \), \( b_{SD} \) is the rate conversion parameter of \( x_{t,m} \) to \( SD_{t,m} \), \( \varepsilon_{SD,t,m} \) is a Gaussian error term of SPUE in year \( t \) in mesh \( m \), \( \text{Effort}_{t,m} \) is the product of the numbers of hunters and hunting days during 1 year for each mesh. In our simulation study, we assumed annual...
hunting days is a period of 120 days and the expected number of hunters for each day and mesh is 5. Thus, we set the same value for Effort\textsubscript{t,m} in all years and meshes as \[\text{Effort}\textsubscript{t,m} = 600\]. For the prior distribution of \(\sigma_{SD}\), a non-informative uniform prior distribution is assigned.

Block count density data were modeled as follows:

\[
\begin{align*}
BC\textsubscript{t,m} &\sim \text{Poisson}(\lambda_{BC\textsubscript{t,m}}) \quad (11) \\
\lambda_{BC\textsubscript{t,m}} &= \exp(x_{t,m} + \varepsilon_{BC\textsubscript{t,m}}) \text{Area}_{t,m} \quad (12) \\
\varepsilon_{BC\textsubscript{t,m}} &\sim N(0, \sigma_{BC}^2) \quad (13)
\end{align*}
\]

Where \(BC\textsubscript{t,m}\) is the number of deer seen by block count in year \(t\) in mesh \(m\), \(\varepsilon_{BC\textsubscript{t,m}}\) is a Gaussian error term of block count density in year \(t\) in mesh \(m\), \(\text{Area}_{t,m}\) is the area of the census in year \(t\) in mesh \(m\). In our simulation study, we set the same value for \(\text{Area}_{t,m}\) in all years and meshes as \(\text{Area}_{t,m} = 0.1\). For the prior distribution of \(\sigma_{BC}\), a non-informative uniform prior distribution is assigned.

We modified the process obtaining the number of deer hunted and culled by data by introducing overdispersed Poisson distribution instead of binomial distribution to eliminate discrete parameters from the model. The model was as follows:

\[
\begin{align*}
NHC\textsubscript{t,m} &\sim \text{Poisson}(\lambda_{NHC\textsubscript{t,m}}) \quad (14) \\
\lambda_{NHC\textsubscript{t,m}} &= \exp(x_{t,m} + \varepsilon_{NHC\textsubscript{t,m}}) fhc\textsubscript{t,m} \quad (15) \\
\varepsilon_{NHC\textsubscript{t,m}} &\sim N(0, \sigma_{NHC}^2) \quad (16)
\end{align*}
\]

where \(NHC\textsubscript{t,m}\) is the number of deer hunted and culled in year \(t\) in mesh \(m\), \(\varepsilon_{NHC\textsubscript{t,m}}\) is a Gaussian error term of the number of deer hunted and culled in year \(t\) in mesh \(m\). For the prior distribution of \(\sigma_{NHC}\), a non-informative uniform prior distribution is assigned. The probability that a certain deer is hunted and culled was modeled as follows:

\[
\begin{align*}
\logit(fhc\textsubscript{t,m}) &\sim N(\mu_{fhc\text{,t}}, \sigma_{fhc}^2) \quad (17) \\
\mu_{fhc\text{,t}} &\sim N(\mu_{fhc\text{,t-1}}, \sigma_{\mu_{fhc}}^2) \quad (18)
\end{align*}
\]

A non-informative Gaussian distribution is assigned for the prior distribution of \(\mu_{fhc\text{,1}}\). For the prior distributions of \(\sigma_{fhc}\) and \(\sigma_{\mu_{fhc}}\), a non-informative uniform prior distribution is assigned.

### 2.2 HMC Algorithm

MCMC method is a generic name of the methods for generating random samples from target distributions with using Markov chain which transitions on the state-space stochastically. There are various algorithms which can implement MCMC method. Those algorithms respectively differ on the points such as how to build Markov chains and how to accept proposed values as new steps of the Markov chains. HMC method which we introduced in this study applies the principles of the Hamiltonian dynamics to construct Markov chains. It is enabled by regarding the state variable and its changing rate at the infinitesimal time as the position variable and the momentum variable, respectively [7].

Let state variable \(z = \{z_i\}\) denotes the vector of all parameters. We can describe a relationship between the state variable and potential energy as follows:

\[
p(z) = \frac{1}{Z_p} \exp(-E(z)) \quad (19)
\]

where \(E(z)\) is the potential energy. It corresponds to the minus logarithmic scale value of the joint probability distribution function of the state variable \(z\) without normalizing constant. Supposing that the state variable \(z\) evolves under continuous time \(\tau\), we define the momentum variables \(r = \{r_i\}\) and kinetic energy \(K(r)\) as follows:

\[
r_i = \frac{dz_i}{d\tau}, \quad K(r) = \frac{1}{2} \|r\|^2 = \frac{1}{2} \sum_i r_i^2. \quad (21)
\]

The Hamiltonian \(H(z, r)\) can be described as the sum of the potential and kinetic energies. It conserves all energies across the entire system:

\[
H(z, r) = E(z) + K(r). \quad (22)
\]

If the position variable and the momentum variable follow the Hamiltonian equations:

\[
\frac{dz_i}{d\tau} = \frac{\partial H}{\partial r_i}, \quad \frac{dr_i}{d\tau} = \frac{\partial H}{\partial z_i}, \quad (23)
\]

Hamiltonian \(H(z, r)\) is conserved constant. That means that \(H(z, r)\) is kept constant with newly proposed position and momentum variables which are obtained by time integration of equation (23). As a result, the value of the joint distribution of \(p(z, r)\) is also kept with newly proposed \(z\) and \(r\). The acceptance probability of the proposed sample using Hamiltonian dynamics is theoretically one. HMC method utilizes these properties for generating Markov chains of parameters.

The method is enabled by following procedure:

1. Set the initial values of \((z, r)\);
2. Simulate the motion on a contour line of constant Hamiltonian by the leapfrog discretization with a stepsize \(\epsilon\) and apply its \(L\)th step value to a proposed state of the new Markov chain step as \((z^*, r^*)\);
3. Accept or reject the proposed state with probability

\[
\min(1, \exp\{H(z, r) - H(z^*, r^*)\})
\]

4. Sample the momentum variable \(r\) from standard normal distribution with fixed state variable \(z\) to break the condition that Hamiltonian is constant;
5. Iterate from process 2 to process 4 until enough samples are obtained.


3 Numerical Experiments

3.1 Simulation

For evaluating our method, we generated simulation data. We simulated the time series of latent deer abundance and several observation data for 4 years. Deer abundance area is divided into 10 × 10 meshes. Each mesh has simulated time series and observations throughout the entire state space efficiently. Generally, MCMC sample series need abandonment of some header samples as the burn-in period which depend on the initial value before the convergence. It is also required that the converged series are thinned out to remove autocorrelations between each steps of the chain. In HMC method, the number of samples which should be abandoned for these after procedures is less than the other methods. Therefore HMC method is the beneficial algorithm which realizes efficient sampling of MCMC method for high dimensional state space.

The advantage of HMC method is the point providing both of the drastic evolutions of the state variable and the high acceptance probability. Its Markov chains can converge to stationary process with the small number of iterations, and search throughout the entire state space efficiently. Generally, MCMC sample series need abandonment of some header samples as the burn-in period which depend on the initial value before the convergence. It is also required that the converged series are thinned out to remove autocorrelations between each steps of the chain. In HMC method, the number of samples which should be abandoned for these after procedures is less than the other methods. Therefore HMC method is the beneficial algorithm which realizes efficient sampling of MCMC method for high dimensional state space.

3.2 Results

In implementing estimation program with Gibbs sampler method, we were forced to set the narrow prior distributions because the wide range prior distributions containing impractical values caused the run-time error arising from arithmetic overflow. We assigned uniform distributions U(0, 3) for the standard deviations $\sigma$ and settled the matter.

Figure 2 is the heat maps which indicates simulated true deer abundance (2-1) and a results of estimation by two methods (2-2, 2-3). Each heat map expresses logarithmic scaled populations $x_{t,m}$ in the research area divided into 10 × 10 meshes. Figure 2-2 and 2-3 correspond with the mean of the posterior distribution estimated by HMC and Gibbs sampler method, respectively. The root mean squared error (RMSE) between true and estimated mean abundance was 22.97 in HMC method and 24.23 in Gibbs sampler method (Table 1).

<table>
<thead>
<tr>
<th>Method</th>
<th>$\exp(x_{t,m})$</th>
<th>$r_m$</th>
<th>$f_{hct,m}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>HMC</td>
<td>22.97</td>
<td>0.12</td>
<td>0.02</td>
</tr>
<tr>
<td>Gibbs sampler</td>
<td>24.23</td>
<td>0.18</td>
<td>0.02</td>
</tr>
</tbody>
</table>

Figure 3 is the trace plots of MCMC samples obtained by two methods. Horizontal axis and vertical axis respectively mean the number of step and sampled values of a parameter $r_1$. We ran three MCMC chains and plotted them by three colors. This figure describes that HMC method requires a few iterations to reach the stationary distribution and Gibbs sampler requires more than 200 iterations. Gibbs sampler took from 10,000 to 15,000 steps for convergence to stationary distribution.

For running estimation program, we used Stan and WinBUGS through rstan (Version 2.17.4) and R2WinBUGS (Version 2.1.21) which are packages in R (Version 3.4.4) used through Rstudio (Version 1.1.456). Our development environment was Windows 10 pro (Version 1803), Intel (R) Core (TM) i5-6200U CPU, and 64 bit operating system.
4 Discussion

Although we were forced to set narrow prior distributions in Gibbs sampler estimation, we consider the reasons of the error may not be caused by the algorithm of Gibbs sampler. Programming error often depends on the execution environment of computer and settings of software such as WinBUGS. However, HMC implemented by Stan is robust method in our environments.

In the condition that deer abundance in each mesh are about 400 at most, the RMSE value of abundance \( \exp(x_{t,m}) \) is acceptable. Similar RMSE values taken in both of the methods indicates that the accuracies do not differ with each other. We acknowledged both of the methods can be applied to the estimation of deer population dynamics.

In terms of time for calculating 1 step of Markov chains, Gibbs sampler is faster than HMC method. It derives from complexity of numerical simulation of Hamiltonian dynamics. On the other hand, HMC method is much less than Gibbs sampler in terms of the number of samples which must be abandoned for after procedures. This is because quick transition from anterior value makes burn-in period and thinning interval shorter. We noticed that HMC method provides high-quality samples though it takes time to generate 1 sample.

If calculating time increases in proportion to the number of iterations, our 1 hour calculation can generate about 1,800 elaborate samples which have been already abandoned burn-in periods and thinned out with HMC method and can generate about 70 samples with Gibbs sampler. We also noticed that HMC method is bet-
ter than Gibbs sampler in terms of time efficiency for
generation of elaborate samples.

5 Conclusion

We expressed the model without discrete parameters
because of the restriction that HMC method can not
treat the model containing discrete parameters. We
confirmed HMC method can estimate deer population
efficiently.

With running time taken sufficiently, HMC and Gibbs
sampler can be used for estimation certainly. However,
HMC method is more efficient than Gibbs sampler in
terms of time for calculation. When the model includes
highly dimensional parameters and complex construc-
tions, researchers can improve time efficiency of esti-
mation if they apply HMC method in place of Gibbs
sampler. Therefore, HMC method is valuable in the
estimation of deer abundance.

In the future work, we want to verify the benefits
of HMC method for the practical use with data ob-
served in local governmental surveys instead of simu-
importance of taking deer migration into consideration
with a large spatial scale estimation. It is necessary
to research practical utilities of HMC method for large
spatial management among local governments.

Acknowledgements

This work was supported by JST, PRESTO, JP-
MJPR1774, Japan.

References

[1] Koichi Kaji and Hayato Iijima. Sika Deer in Japan:
Science and Management of Overabundant Popu-
[2] Yamamura, Matsuda, Yokomizo, Kaji, Uno,
Tamada, Kurumada, Saitoh, Hirakawa. Harvest-
based Bayesian estimation of sika deer popula-
tions using state-space models, Popul Ecol (2008)
50:131-144.
Population Dynamics Using a Bayesian State-
Space Model With Multiple Abundance Indices,
The Journal of Wildlife Management (2013)
77(5):1038-1047.
Bayesian state-space model for evaluating the ef-
effects of localized culling on subsequent density
changes: sika deer as a case study, European Jour-
[5] Lunn, D.J., Thomas, A., Best, N., and
Spiegelhalter, D. WinBUGS - a Bayesian
modeling framework: concepts, structure,
and extensibility, Statistics and Computing,
2000, 10:325-337. URL https://www.mrc-
bsu.cam.ac.uk/software/bugs/the-bugs-project-
winbugs/.
[6] M. Plummer. JAGS: A program for analysis of
Bayesian graphical models using Gibbs sampling,
[7] Christopher M. Bishop. Pattern Recognition and
Machine Learning: Information Science and
Language Users Guide and Reference Manual, Ver-
sion 2.18.0. http://mc-stan.org
and R (Wonderful R) Kyoritsu Shuppan, pp. 203-
228, 2016.
[10] Andrew M. Latimer, Shanshan Wu, Alan E.
Gelfand, and John A. Silander, Jr. Building Statis-
tical models To Analyze Species Distributions Eco-
for convex problems Mathematical programming
120 (1), 221-259, 2009.
No-U-Turn Sampler: Adaptively Setting Path
Lengths in Hamiltonian Monte Carlo Journal of