Urban Retail Size Simulation Based on Monte Carlo Markov Chain and Sampling Efficiency Evalution

With a Simplified Example

Jinghong Chen

2020-4-27

1 Introduction

With the unprecedented availability of urban data, a considerable amount of effort has been put into understanding urban structures from these data in hope of inferring useful information that would help predict urban dynamics and optimize urban design. Simulating the dynamics of retail centers has been one of major and ongoing topics in these field. A statistical model developed by Lam and Girolami et.al[1] is preferred because the classical deterministic Harris and Wilson model[2] fails to take the variability and unpredictability of urban data measurement. In the model proposed in [1], the problem of simulating possible retail configurations is turned into sampling from a distribution known up to a normalizing constant. This project aims to use Monte Carlo Markov Chain(MCMC) to sample from the distribution and compare the effectiveness of different MCMC methods. Given the complexity of the original problem in [1], it is not straight forward to evaluate different MCMC methods with the original problem. Therefore, we devise a simplified example to compare and constrast the effectiveness of different MCMC methods.

2 Modelling Urban Retail System

This section lays out the urban model proposed by Lam and Girolami[1] et.al and the readers are referred to the original paper for more details. Urban configuration is modelled by residences (where capital flows out) and retail structures (where capital flows in). The flow of capital is encouraged by the 'attractiveness' of retail structures (proportional to structure size) and hindered by the 'inconvenience' of travel (proportional to travel cost). If the influx of capital exceeds maintainess cost the size of the structure will increase, otherwise decrease. Mathematically, we denote the flow between retail structure i and residence j as T_{ij}. The capital from N residences is therefore

$$O_i = \sum_{j=1}^{M} T_{ij}, i = 1, ..., N \tag{1}$$

which are known, the capital into M retail structures are

$$D_j = \sum_{i=1}^{N} T_{ij}, j = 1, ..., M \tag{2}$$

are to be determined. A suitable model for the flow is obtained by maximizing an entropy function subject to the above constraints, the resulting flows are

$$T_{ij} = O_i \frac{W_j^{\alpha} \exp\left(-\beta c_i j\right)}{\sum_{k=1}^{M} W_k^{\alpha} \exp\left(-\beta c_i k\right)} \tag{3}$$

Where α is the attractiveness and β the inconvenience. The cost matrix c_{ij} is the travel cost from residence i to retail structure j. Instead of modelling the dynamics of W with ordinary differential equation, we can model it with a stochastic differential equation. After performing an exchange of variable $X = ln(W)$. It can be shown that at steady state a solution is

$$\rho_\infty(x) = \frac{1}{Z} exp(-\gamma V(x)), Z = \int_{\mathbb{R}} exp(-\gamma V(x))dx \tag{4}$$

And the potential function $V(x)$ is given by

$$\epsilon^{-1} V(x) = -\alpha^{-1} \sum_{i=1}^{N} O_i ln \sum_{j=1}^{M} exp(\alpha x_j - \beta c_{ij}) + \kappa \sum_{j=1}^{M} exp(x_j) + \delta \sum_{j=1}^{M} x_j \tag{5}$$

where κ is the unit maintainess cost for retail structure and δ the term to prevent zones from collapsing and represent government investment. For detail derivation the readers are referred to the original paper[1]. Here we only provide a very brief overview of the model.

3 Sampling with Monte Carlo Markov Chain

From equations (4) and (5) we know the retail configuration X follows a Boltzmann-like distribution. We now would like to take samples from the distribution in order to simulate possible retail configuration. In this project we consider four MCMC approaches and compare their performances. They are Metropolis Method, Parallel Tempering (PT), Hamiltonian Monte Carlo (HMC) and Combined Monte Carlo with PT and HMC. In this section we give a very brief introduction to these sampling algorithms.

3.1 Metropolis Method

The basic idea of the Metropolis algorithm is to simulate a Markov chain in the state space of x so that the limiting/stationary/equilibrium distribution of this chain is the target distribution π. See appendix A for a brief introduction for Markov Chain. In Markov chain Monte Carlo simutlation (MCMC), one knows the equilibrium distribution and is interested in prescribing an efficient transition rule so as to reach this equilibrium.
The Metropolis Algorithm is given as follow:

The Metropolis Algorithm

- Propose a random "unbiased pertubation" of the current state x_t so as to generate a new configuration $x^{'}$. Calculate the change $\Delta h = (x^{'}) - h(x_t)$
- Generate a random number $U \sim Uniform[0,1]$. Let $x_{t+1} = x^{'}$ if $U \leq \frac{\pi(x^{'})}{\pi(x_t)} = exp(-\Delta h)$

Note that the pertubation rule is restricted to be *symmetric*, i.e., $T(x, x^{'}) = T(x^{'}, x)$ where T is a transition function

3.2 Parallel Tempering

Parallel Tempering(PT) is a powerful technique to sample target distribution which has different modes potentially separated by high energy barriers. The idea is to construct a family of distributions, typically by varying a parameter called temperature, and run Markov Chains for each of them in parallel. It is expected that distribution associated with higher temperature has 'flatter terrain' hence easier for the sampler to "move around" and that the distributions with the lower temperatures are more concetrated around the modes. Exchange of configuration is performed between these Markov Chains in hope of all significant regions in the distribution with the lowest temperature (which is often the distribution of interests) are reached. A more rigorous description of Parallel Tempering is presented here.

We augment the space to the product space $X_1 \times X_2 \times ... \times X_I$, were the X_i are identical copies of X. Suppose $(x_1, x_2, \ldots, x_I) \in X_1 X_2 \ldots X_I$. For the familiy of distributions $\Pi = \pi_i(x), i \in I$, we define a joint probability distribution on the product space as $\pi_p t(x_1, \ldots, x_I) = \Pi_{i \in I} \pi_i(x_i)$ and run parallel MCMC chains on all of the X_i. An "index swapping" operation is conducted in place of the temperature transition in ST.

Parallel Tempering Algorithm

- Let the current state be $(x_1, x_2, ..., x_I)$, we draw $u\ Uniform[0, 1]$
- If $u \leq \alpha_0$, we construct *parallel step*; That is, we update every $x_i^{(t)}$ to x_i^{t+1} via their respective MCMC scheme
- If $u > \alpha_0$, we conduct the *swapping step*; That is, we randomly choose a neighbouring pair, say i and $i+1$, and propose "swapping" $x_i^{(t)}$ to x_{i+1}^t. Accept with the probability $\min\left(1, \frac{\pi_i(x_{i+1}^t)\pi_{i+1}(x_i^t)}{pi_i(x_i^t)\pi_{i+1}(x_{i+1}^t)}\right)$

3.3 Hybrid Monte Carlo

The Hybrid Monte Carlo originates from Molecular Dynamic (MD) simulation, where each molecule obeys Newtonian Mechanics and it is of interests to simulate how the system of molecules evolves under these restriction. In the context of Monte Carlo Method, Newton Laws (in the form of Hamiltonian dynamics) can be used to propose new configuration. The proposal step looks very similar to MD with leap-frog algorithm only that one needs to come up with momentum proposal and masses. A brief introduction to the leap-frog algorithm is given in the appendix.

To implement Hybrid Monte Carlo, we need an Guide Hamiltonian $H^{'}(x, p) = U^{'}(x) + k(p)$ where p is an anxiliary variable with the same dimensionality as x, $k(p) = \sum_{i=1}^{d} \frac{p_i^2}{m_i}$ and the m_i are positive quantities representing the masses.

The function $U^{'}(x)$ is allowed to be different from the target one, $U(x)$

Hybrid Monte Carlo Suppose at time t we are at position x of the configuration space

- Generate a new momentum vector p from the Gaussian distribution (i.e., from $\phi(p) \propto \exp(-k(p))$)
- Run the leap-frog algorithm (or any other deterministic time-reversible and volume-preserving algorithm), starting from (x, p), for L steps with step size ϵ, to obtain a new state in the phase space, $(x^{'}, p^{'})$

It is noteworthy that the performance of the algorithm is very sensitive to the choice of L and ϵ as they affect the numerical error of the leap-frog integrator.

3.4 Combined Monte Carlo with PT and HMC

This is the scheme used in the original paper[1] to simulate retail configuration with high noise. PT and HMC can be easily combined due to the fact that HMC is only Metropolis Method with novel state proposal and that PT is just the idea of running multiple Markov Chains in parallel. The algorithm is listed below:

Combined Monte Carlo with PT and HMC

- Let the current state be $(x_1, x_2, ..., x_I)$, we draw u $Uniform[0, 1]$ and generate a new momentum vector p_i for each chain from the Gaussian
- If $u \leq \alpha_0$, we construct *parallel step*; That is, we update every $x_i^{(t)}$ to x_i^{t+1} via their respective MCMC scheme, the new state proposal is that of the HMC (i.e., by running leap-frog integrator L times with step size ϵ
- Accept the proposed state $(x^{'}, p^{'})$ with probability $\min\Big(1, \exp(-H(x^{'}, p^{'}) + H(x, p))\Big)$ and $let x_{t+1} = x$ with the remaining probability

4 A Simplified Model for MCMC Verification

Owing to the complexity of the problem, it is desirable to first verify and get some intuition of our MCMC algorithms before applying it to the actual problem. For this purpose, we contrieve a simplified model which we know what the equilibrium state should be, which we refer to later as *the mini city*.

The mini city consists of two retail structures and three residences symmetrical in location and size. This means that one of the mode is where the two retail structures are of the same size and we verify our MCMC by observing this convergence around this equilibrium.

We will use the mini city as verification model later.

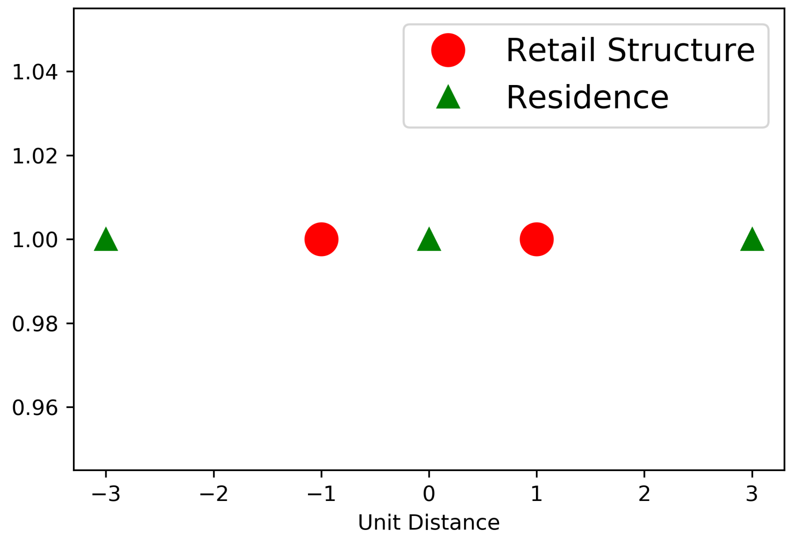


Figure 1: Configuration of the mini city

5 Results

5.1 Verifying Algorithms on the Mini City

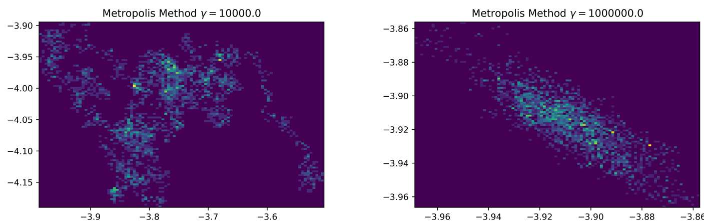
5.1.1 Deterministic Model

To obtain the reference equilibrium, we first run the deterministic model given by the ordinary differential equation $dW_j = W_j(D_j - \kappa W_j)dt$. Given our initial conditions, we obtain the equilibrium $W_{det} = [0.02, 0.02]$ where W is chosen to be less than 1 because in the real setup the sizes are normalized for numerical benefits.

5.1.2 Metropolis Method

When using MCMC scheme, we first perform a change of variable $X = \ln W$ and X is going to satisfy equations (4) and (5). The convergence to equilibrium state ($X_det = [-3.91, -3.91]$)found by the deterministic model is found to be heavily influenced by the parameter γ is the model. For a Markov Chain of length 10000, a high values of $\gamma = 10^4$ is able to reach convergence whereas when $\gamma = 10^6$ convergence is not found in the final histogram.

The result can be explained intuitively as γ is a scaling factor of our potential. The higher it is, the steeper our potential terrain is and thus the more distringuishable the global minimum. γ is actually the inverse temperature parameter in a typical Boltzmann distribution of the form $\pi \propto \exp(-1/Th(x))$.



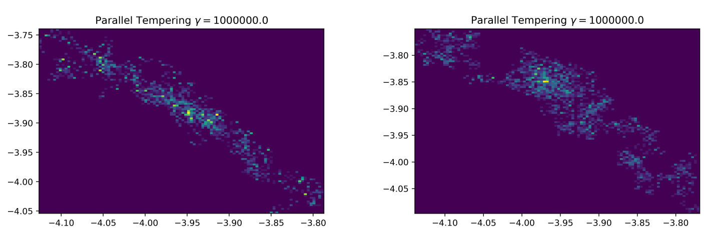
(a) Histogram with $\gamma = 10^4$ (b) Histogram with $\gamma = 10^6$

Figure 2: Comparison of Metropolis Method with different γ values

Therefore, the higher γ is, the lower the temperature and the more concentrated the potential. This is why this is used to control the level of noise in the original paper[1] where high γ value represents low noise.

5.1.3 Parallel Tempering

We setup five temparature parameters and run the Parallel Tempering Algorithm on the mini city. Because the idea of parallel tempering is that the chains with higher temperature will hopefully help the lower-temperature chain explore the terrain, it is reasonable to set $\gamma = 10^6$ so that the target chain will converge. Since the mini city has one equilibrium only owing to its simplicity, it is not the best model to demonstrate the benefits of parallel tempering which is to escape local minima and explore the distribution more thoroughly. A more suitable model is the Ising Model used to simulate magnetization. A brief introduction and results with regard to the Ising Model is in the appendix.



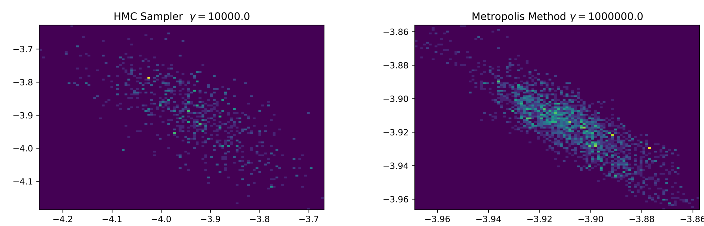
(a) PT Histogram with temperature = 1, length 20000 (b) PT Histogram with temperature = 16, length 20000

Figure 3: PT Histograms at different temperature

As expected, the histogram with higher temperature is sparser.

5.1.4 Hybrid Monte Carlo

We set the masses to 1 and draw the momentum vector from a Gaussian distribution with mean 0. The variance, leap frog steps L and step size ϵ are tuned until convergence is observed with acceptance rate at around 0.7. Noteably, the HMC scheme is able to converge to local minimum at lower value of γ (high noise). When $\gamma = 10^4$, the Metropolis method fails to find the minimum with length of 10000. In contrast, the HMC scheme converges nicely with only 1000 in length under the same γ. This can be explained by the more cleverly chosen state proposal which includes a term that "pulls" the proposal state to minima. This is a nice property for noisy data. However, it is only observed that when $\gamma = 10^6$ representing low noise, the HMC degrades to a single point. The "gravity" of the global maximum becomes too strong for the sampler to escape. This might be problematic with multi-mode distribution so parameters of the HMC should be tuned depending on the question at head.



(a) HMC Histogram with $\gamma = 10^4$, length 1000

(b) Metropolis Histogram with $\gamma = 10^6$, length 10000

Figure 4: Comparison of HMC and Metropolis with high noise

5.1.5 Combined PT and HMC

5.2 Results for London Retail Centre

In this section we present the results based on London retail data as discussed in the original paper[1].

5.2.1 Deterministic Model

As a reference, we run the simple deterministic model for London retail and residence configuration. As it turns out, running the model with $\alpha = 2, \beta = 0.5$ leads to a reduction of 23% in total size and the resulting configurations are more even in sizes than the initial configuration. This suggests that it is not surprising if the MCMC scheme gives us samples with smaller total size.

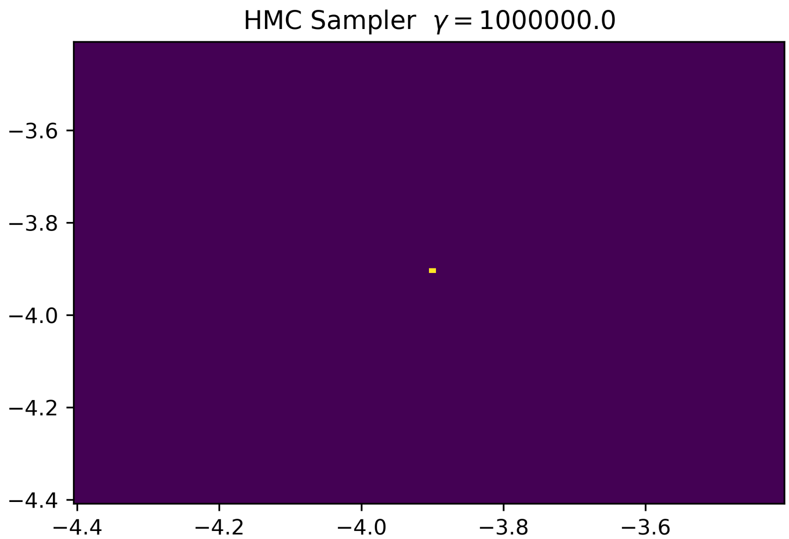


Figure 5: HMC Histogram with $\gamma = 10^6$

5.2.2 Metropolis Monte Carlo

5.2.3 Parallel Tempering

5.2.4 Hybrid Monte Carlo

5.2.5 Combined PT and HMC

6 Discussion

7 Conclusion

References

[1] Ellam L, Girolami M, Pavliotis GA, and Wilson A. Stochastic modelling of urban structure. *Proc. R. Soc.A*, 474, 2017.