L8: Dependent Monte Carlo Sampling, Overview

1 Overview of MCMC

1. **Dependent samples:** In contrast to direct sampling using the inverse probability integral transform, rejection sampling, and importance sampling, which all yield independent samples from a target distribution $p(\theta)$, or for Bayesian inference, $p(\theta|\boldsymbol{y})$, Markov Chain Monte Carlo (MCMC) methods yield dependent samples from the target distribution.

   In particular the *dependent* samples are generated from a Markov chain which has a *limiting stationary distribution* that equals the target distribution $p(\theta)$: that the chain values have *converged* to be a sample from $p(\theta)$.

2. **Iterative conditional generation:** Given an initial value for θ, denoted θ^0, the next value θ^1 is generated from a conditional probability distribution given θ^0, say $g(\theta^1|\theta^0)$. This is subsequently repeated in an iterative manner: at the i^{th} iteration, θ^i is generated from $g(\theta^i|\theta^{i-1})$. Let N denote the total number of sample values generated.

3. **Burn-in:** In most cases the initial values, say θ^1, θ^2, ..., θ^B, are *not* distributed according to the target distribution, $p(\theta)$. These initial sample values are then discarded and are not used for making inferences about $p(\theta)$. This initial sample generation period is called the *Burn-in period*.

4. **Sample for inference:** The additional $N - B$ sample values that are generated, θ^{B+1}, θ^{B+2}, ..., θ^N, are the ones that used for inference. For example, suppose that θ is a scalar, the expected value of a function of θ, $h(\theta)$, is estimated by:

$$\hat{E}[h(\theta)] = \frac{1}{N-B}\sum_{i=B+1}^{N} h(\theta^i) \tag{1}$$

   Other sorts of inferences described previously can be carried out using this sample. For example:

$$\widehat{\Pr}(a \le \theta \le b) = \frac{1}{N-B}\sum_{i=B+1}^{N} I(a \le \theta^i \le b)$$

   and

$$\widehat{Var}(\theta) = \frac{1}{N-B}\sum_{i=B+1}^{N}\left(\theta^i - \hat{\theta}\right)^2$$

   or

$$\widehat{Var}(\theta) = \left(\frac{1}{N-B}\sum_{i=B+1}^{N}(\theta^i)^2\right) - \left(\frac{1}{N-B}\sum_{i=B+1}^{N}\theta^i\right)^2$$

   Comments.

   - The 2nd estimate for $Var(\theta)$ is based on $V(X) = E[X^2]$ - $(E[X])^2$.
   - For $\widehat{Var}(\theta)$, the divisor $N - B - 1$ would be technically better, but with large $N - B$ the effect is negligible.
   - Alternative notation is to let N=number of iterations used for inference, thus the total number of iterations is $B + N$.

Comments

- The art of MCMC is choosing or constructing the Markov chain that will yield (eventually) samples from the target distribution, $p(\theta)$. In other words, Markov chains have the right limiting stationary distribution.
- The Metropolis-Hastings algorithm and the Gibbs Sampling are two popular methods that achieve this.
- Determining the length of the burn-in period is not an exact science. The commonly used methods for determining B are indicative that the chain has not converged, rather than it has converged.
- Multiple chains: one common sense approach to determining burn-in is to run multiple chains, e.g., 3, with different initial values, θ_j^0, for chains j=1,2,3. Then examine the point in the iterations at which the simulated values have similar probability distributions; e.g., after discarding B=1000 iterations, do the histograms for the remain $N - B$ values look similar?
- Burn-in time is very much a function of the Markov chain that is used—some converge faster than others.
- The MCMC estimate, e.g., eq'n (1), is again an estimate of an integral. The properties of the estimate, e.g., consistency, are not based on the standard Strong Law of Large Numbers (SLLN) or the Central Limit Theorem (CLT) because the sample values are dependent, not independent. Instead the properties depend upon various *ergodic* theorems that are Markov chain versions of the SLLN and CLT.
- Related to the previous point, the Monte Carlo error in the MCMC estimates cannot be calculated as simply as for the independent samples. Instead, various time series type approaches are used to estimate Monte Carlo error.
- Some argue that burn-in is unnecessary and a waste of samples. See Geyer, 1992, Statistical Science; see also "Burn-in is unnecessary" available at `http://users.stat.umn.edu/~geyer/mcmc/burn.html`

2 Brief Introduction to Markov Chains

Here we give a brief introduction to discrete time and discrete state-space Markov chains[1].

Let $\{\theta^t\}$, $t = 0, 1, \ldots$ be a sequence of random variables where each θ^t can take one of a finite (or countably infinite) number of values. These possible values are called *states*.

The notation $\theta^t = j$ means that the "process" at time t is in state j.

The *state space*, denoted S, is the set of all possible states.

For example, if θ^t was a Poisson random variable, then possible states are 0, 1, 2, 3, ... (a countably infinite number) and S = the set of non-negative integers.

[1]The explanation is based on Section 1.7 Markov chains from Computational Statistics (2013, Givens and Hoeting)

2.1 Definition of a Markov chain

- The joint distribution for $\theta^0, \theta^1, \ldots, \theta^T$ can be written as the product of conditional distributions:

$$\begin{aligned}\Pr(\theta^0 = x_0, \theta^1 = x_1, \ldots, \theta^T = x_T) = \Pr(\theta^T = x_T | \theta^0 = x_0, \theta^1 = x_1, \ldots, \theta^{T-1} = x_{T-1}) \\ \times \Pr(\theta^{T-1} = x_{T-1} | \theta^0 = x_0, \theta^1 = x_1, \ldots, \theta^{T-2} = x_{T-2})) \times \ldots \\ \times \Pr(\theta^0 = x_0)\end{aligned}$$

- If θ^t given θ^{t-1} is (conditionally) independent of other values, the conditional distribution for θ^t simplifies:

$$\Pr(\theta^t = x_t | \theta^{t-1} = x_{t-1}, \ldots, \theta^0 = x_0) = \Pr(\theta^t = x_t | \theta^{t-1} = x_{t-1})$$

  This is called the Markov property. And the joint distribution simplifies considerably:

$$\Pr(\theta^0 = x_0, \theta^1 = x_1, \ldots, \theta^T = x_T) = \prod_{t=1}^{T} \Pr(\theta^t = x_t | \theta^{t-1} = x_{t-1}) \times \Pr(\theta^0 = x_0)$$

- A sequence of random variables, θ^0, θ^1, θ^2, ..., with the Markov property is called a *Markov chain*.

2.2 Terminology and Notation of a Markov chain

- Shorthand notation for $\Pr(\theta^t = j | \theta^{t-1} = i)$ is p_{ij}^t.
- p_{ij}^t is called the *one-step transition probability*.
- If the p_{ij}^t are the same for times t, i.e., p_{ij}^t=p_{ij}, then the chain is called *time homogeneous*.
- If the p_{ij}^t vary over time, the chain is called *time inhomogeneous*.
- Assume there are K states and denote the values by 1, 2, ..., K. The "movement" probabilities of a Markov chain can be neatly represented by a K by K *transition probability matrix, tpm*, P_t=

	t			
$t-1$	1	2	...	K
1	p_{11}^t	p_{12}^t	...	p_{1K}^t
2	p_{21}^t	p_{22}^t	...	p_{2K}^t
$\vdots$	$\vdots$	$\vdots$	...	$\vdots$
K	p_{K1}^t	p_{K2}^t	...	p_{KK}^t

  Note that the probabilities in each row sum to 1. If the state at time t is i then it will go be in some state at time t.

  In the case of a time homogeneous chain, let P denote the transition probability matrix.

2.3 Example

This example is taken from Givens and Hoeting. Precipitation "states", wet (more than 0.01 inches of precipitation) and dry, for San Francisco were recorded for 1814 pairs of consecutive days during the months November through March, starting from November 1990 and ending March 2002. The table below summarizes the transitions in states from day $t-1$ to day t:

	t	
$t-1$	Wet	Dry
Wet	418	256
Dry	256	884

So the state space S has 2 states: wet and dry. Let θ^t be the state on day t. Assuming time-homogeneity, the estimated tpm for θ^t is:

$$\widehat{P} = \begin{bmatrix} 0.620 & 0.380 \\ 0.224 & 0.775 \end{bmatrix}$$

For example, $\Pr(\theta^t = \text{Wet} | \theta^{t-1} = \text{Wet}) = 0.620$.

2.4 Limiting behaviour of Markov chains

First some definitions of certain types of states and chains:

Recurrent state: a state to which a Markov chain returns with probability 1 is a *recurrent state*.

Nonnull state: a state for which the expected time (number of steps) until recurrence is finite is called a *nonnull state*.

Note: recurrent states in Markov chains with finite state spaces are nonnull.

Irreducible Markov chain: a chain where a state j can be reached in a finite number of steps from *any state* i is an *irreducible Markov chain*.

In other words, there is a number $m > 0$ such that $\Pr(\theta^{m+n} = j | \theta^n = i) > 0$.

Periodic Markov chain: a chain where the number of steps required to return some portions of the state space is a multiple of some integer, say d, is a *periodic Markov chain*. In other words, $\Pr(\theta^{m+n} = j | \theta^n = j) > 0$ only if m/d is an integer.

For example, the chain with the following (homogeneous) tpm, P, has period d=3:

	t		
$t-1$	a	b	c
a	0.0	1.0	0.0
b	0.0	0.0	1.0
c	1.0	0.0	0.0

A chain that is *not periodic* is an *aperiodic Markov chain*.

Ergodic Markov chain: a chain that is *irreducible*, *aperiodic*, and all its states are *nonnull* and *recurrent* is an *ergodic Markov chain*.

Marginal and Stationary distributions.

- The marginal probability distribution for the state at time t is denoted π_t. It is a vector of probabilities that sum to 1, where the ith element $\pi_t(i)$ is the *marginal probability* that $\theta^t = i$, $\pi_t(i) = \Pr(\theta^t = i)$. For example, letting superscript T mean transpose, π_t^T=(0.35, 0.15, 0.3, 0.15, 0.05), and $\pi_t(2)$ =$\Pr(\theta^t = 2)$=0.15. In other words, π_t is the marginal distribution for θ^t.
- If the tpm at time $t+1$ is P, then the marginal distribution for θ^{t+1}, π_{t+1}, is $\pi_{t+1}^T = \pi_t^T P$.
- Any discrete probability distribution π is a *stationary distribution* for P (or a Markov chain with tpm P) **if**

$$\pi^T P = \pi^T$$

(Note the dropping of the subscript t.) This means that the probability of being in state j at time t equals the probability of being in state j at time $t+1$.

Referring to the San Francisco wet and dry days P, the stationary distribution is π^T=(0.3715, 0.6284):

$$[0.3715\ 0.6284] \begin{bmatrix} 0.6202 & 0.3798 \\ 0.2245 & 0.7754 \end{bmatrix} = [0.3715\ 0.6284]$$

Three simulations from the chain with T=100,000 (see Appendix A.1 for the R code) was generated from this Markov chain. The results are shown in Table 1.

Table 1: Fraction of dry days at intermediate time points in the San Francisco Wet-Dry day Markov chain. The stationary distribution probability of dry days is 0.6284.

	Intermediate time points				
Simulation	10	100	1000	10000	100,000
1	0.400	0.680	0.669	0.636	0.629
2	0.500	0.630	0.662	0.633	0.626
3	0.800	0.620	0.639	0.630	0.626

Comments

- What this means: if a Markov chain at time t has a marginal probability distribution that is also a stationary distribution, the marginal probability distribution for all future random variables in the chain is the same marginal probability distribution. The marginal distributions for θ^t, θ^{t+1}, θ^{t+2}, ..., are identical.
- A given P can have more than one stationary distribution. For example,

	t						
$t-1$	0	1	2	3	4	5	6
0	0	1/2	0	0	0	0	1/2
1	0	1/3	1/3	1/3	0	0	0
2	0	1/3	1/3	1/3	0	0	0
3	0	1/3	1/3	1/3	0	0	0
4	0	0	0	0	1/3	1/3	1/3
5	0	0	0	0	1/3	1/3	1/3
6	0	0	0	0	1/3	1/3	1/3

Main idea of MCMC. In the context of Bayesian inference, the central idea of Markov chain Monte Carlo methods, is to generate samples from a Markov chain such that "eventually" a stationary distribution will be reached and that distribution is the posterior distribution, $p(\theta|\boldsymbol{y})$.

Of course MCMC methods can be used to generate samples from an arbitrary distribution.

The "art" of MCMC methods is to construct chains that have a tpm which will yield a (limiting) stationary distribution equal to the target distribution.

Reversible chains and the Detailed Balance Condition. One can roughly view what are reversible chains as a means of 'creating' a Markov chain that has the 'right' P for producing a (limiting) stationary distribution equal to the target distribution.

Reversible chain: a time homogeneous Markov chain with tpm P such that

$$\pi(i)\Pr(\theta^t = j|\theta^{t-1} = i) = \pi(j)\Pr(\theta^t = i|\theta^{t-1} = j) \tag{2}$$

then π is *a* stationary distribution for the chain and such a chain is called a *reversible Markov chain*.

- Equation 2 is called the *detailed balance equation*.
- Equation 2 is sometimes written

$$\pi(i)p_{ij} = \pi(j)p_{ji}$$

- Such a chain is called reversible because the joint distribution for two consecutive realisations is the same whether the chain is run forwards or backwards.
- The San Francisco wet and dry days chain is reversible (showing more digits to reduce rounding errors):

$$\pi(Wet)\Pr(\theta^t = Dry|\theta^{t-1} = Wet) = \pi(Dry)\Pr(\theta^t = Wet|\theta^{t-1} = Dry), namely$$
$$0.3715546 * 0.3798220 = 0.6284454 * 0.2245614 = 0.1411246$$

Key Theoretical Results. If a Markov chain with tpm P is irreducible and aperiodic and it has a stationary distribution π, then

- **π is *unique*, i.e., there is only one stationary distribution.**
- **The limiting distribution is the stationary distribution**:

$$\lim_{n\to\infty} \Pr(\theta^{t+n} = j|\theta^t = i) = \pi(j) \tag{3}$$

- Another way to state eq'n (3): "if θ^1, θ^2, ..., are realisations from an irreducible and aperiodic Markov chain with stationary distribution π, then θ^n converges in distribution to the distribution π" (Givens and Hoeting, p 16).
- Given the tpm P, the components of π are solutions to the following equations:

$$\sum_{i\in S} \pi(i) = 1$$
$$\pi(j) = \sum_{i\in S} \pi(i)\Pr(\theta^t = j|\theta^{t-1} = i)$$

subject to the constraint that $\pi(j) \geq 0$, for all j.

Another theoretical consequence is the following *ergodic theorem*, which is a generalisation of the strong law of large numbers: For any function h such that $E_\pi[h(\theta)]$ exists,

$$\Pr\left(\lim_{n\to\infty}\frac{1}{n}\sum_{t=1}^{n}h(\theta^t)=E_\pi[h(\theta)]\right)=1 \tag{4}$$

(almost sure convergence.)

Thus, a sample from a Markov chain can be used to estimate integrals.

3 Metropolis-Hastings Algorithm: Part 1

The objective is to generate a sample from a target distribution, $p(\Theta)$, where Θ is usually a vector, $\Theta = (\theta_1, \theta_2, \ldots, \theta_q)$. The cases we are primarily interested in are where $p(\Theta)$ is a posterior distribution: $p(\Theta|\boldsymbol{y})$. However, we will momentarily omit explicit conditioning on data.

The Metropolis-Hastings (MH) algorithm is an iterative sampling procedure that generates a sequence of random variables, Θ^0, Θ^1, ..., Θ^N. In contrast to previous generation methods, like direct sampling, or rejection sampling, the value generated at iteration t depends on the value from iteration $t-1$, thus the resulting sample of values are not individually independent. MH has similarities with rejection sampling in that (at iteration t) there is a distribution that generates Θ^t and with a certain probability the generated value will be kept, i.e., added to the sample, otherwise it will be discarded. However, in contrast to rejection sampling, if the generated value is rejected, the previous value, Θ^{t-1}, is used as the t value, too: $\Theta^t = \Theta^{t-1}$.

In rejection sampling, we called the distribution used to generate Θ the envelope distribution, and we denoted it $g(\theta)$. In MH sampling that distribution is sometimes called the *proposal distribution*, which will denote it q and to indicate it dependence on previous values, it will sometimes be written $q(\Theta^t|\Theta^{t-1})$. We will call the generated value the *candidate value* and denote it Θ^c.

The steps in the MH algorithm are the following.

At iteration t, given a sample value at iteration $t-1$, Θ^{t-1}:

1. Generate a candidate value, Θ^c, from the proposal distribution, $q(\Theta^c|\Theta^{t-1})$.
2. Calculate the Metropolis Hastings ratio, MHR:
$$MHR(\Theta^{t-1}, \Theta^c) = \frac{p(\Theta^c)\ q(\Theta^{t-1}|\Theta^c)}{p(\Theta^{t-1})\ q(\Theta^c|\Theta^{t-1})} \tag{5}$$
Note that this ratio can be larger than 1.
3. Generate a Uniform(0,1) random variable u.
4. If $u \leq \min\left(1, MHR(\Theta^{t-1}, \Theta^c)\right)$, then set $\Theta^t = \Theta^c$, thus "keep" Θ^c.
Else set $\Theta^t = \Theta^{t-1}$.
5. Set $t = t+1$ and go back to Step 1

Note: this a generating samples from a Markov chain because the probability of selecting Θ^t is dependent on the value of Θ^{t-1}.

Key point from Bayesian perspective.

- The normalising constant of the posterior distribution does not need to be known due to the cancellation in MHR:
$$\begin{aligned} MHR(\Theta^{t-1}, \Theta^c) &= \frac{p(\Theta^c|\boldsymbol{y})\ q(\Theta^{t-1}|\Theta^c)}{p(\Theta^{t-1}|\boldsymbol{y})\ q(\Theta^c|\Theta^{t-1})} \\ &= \frac{\frac{\pi(\Theta^c)f(\boldsymbol{y}|\Theta^c)}{m(\boldsymbol{y})}\ q(\Theta^{t-1}|\Theta^c)}{\frac{\pi(\Theta^{t-1})f(\boldsymbol{y}|\Theta^{t-1})}{m(\boldsymbol{y})}\ q(\Theta^c|\Theta^{t-1})} \\ &= \frac{\pi(\Theta^c)f(\boldsymbol{y}|\Theta^c)\ q(\Theta^{t-1}|\Theta^c)}{\pi(\Theta^{t-1})f(\boldsymbol{y}|\Theta^{t-1})\ q(\Theta^c|\Theta^{t-1})} \end{aligned}$$

A R Code

A.1 Simulation of the San Francisco Wet and Dry days Markov chain

```
#-- Demonstrating time homogeneous Markov chain, using the San Francisco wet and dry days
wet.dry.data <- matrix(c(418,256,256,884),nrow=2,ncol=2,byrow=TRUE,
                       dimnames=list(c("Wet.t-1","Dry.t-1"),c("Wet.t","Dry.t")))

P <- wet.dry.data/apply(wet.dry.data,1,sum)
pi.station <- c(P[2,1]/(P[1,2]+P[2,1]),P[1,2]/(P[1,2]+P[2,1]))

cat("stationary dist=",pi.station,"\n")
# 0.3715546 0.6284454

cat("p*P =\n")
rbind(pi.station) %*% P
#                Wet.t     Dry.t
#  pi.station 0.3715546 0.6284454

day.type <- c("Wet","Dry")

 set.seed(573)
  T <- 100000
  num.sims <- 3
  out <- matrix(data=NA,nrow=num.sims,ncol=T)

  for(i in 1:num.sims) {
   # 0=Wet, and 1= Dry (success)
   out[i,1]   <- rbinom(n=1,size=1,prob=0.5)
   for(j in 2:T) {
     #if wet, probability dry = 0.38
     #if dry, probability dry = 0.78
     p     <- P[out[i,j-1]+1,2]
     temp  <- rbinom(n=1,size=1,prob=p)
     out[i,j] <- temp
   }
  cat("Fraction dry=",sum(out[i,]==1)/T,"\n")
  }

check.points <- c(10,100,1000,10000,10000)
frac.dry.mat <- matrix(data=NA,nrow=num.sims,ncol=length(check.points),
    dimnames=list(paste("sim",1:num.sims),check.points))

for(i in 1:num.sims) {
  for(j in 1:length(check.points)) {
    frac.dry.mat[i,j] <- sum(out[i,1:check.points[j]]==1)/check.points[j]
  }
}
```

November 8, 2020