2000 Proceedings of the Autumn Conference, Korean Statistical Society

A Comparison study of Hybrid Monte Carlo Algorithm

황진수[1], 전성해[2], 이찬범[3]

요 약

베이지안 신경망 모형(Bayesian Neural Networks Models)에서 주어진 입력값(input)은 블랙 박스(Black-Box)와 같은 신경망 구조의 각 층(layer)을 거쳐서 출력값(output)으로 계산된다. 새로운 입력 데이터에 대한 예측값은 사후분포(posterior distribution)의 기대값(mean)에 의해 계산된다. 주어진 사전분포(prior distribution)와 학습데이터에 의한 가능도함수(likelihood functions)를 통해 계산되어진 사후분포는 매우 복잡한 구조를 갖게 됨으로서 기대값의 적분계산에 대한 어려움이 발생한다. 이때 확률적 추정에 의한 근사 방법인 몬테칼로 적분을 이용한다. 이러한 방법으로서 Hybrid Monte Carlo 알고리즘은 우수한 결과를 제공하여준다(Neal 1996). 본 논문에서는 Hybrid Monte Carlo 알고리즘과 기존에 많이 사용되고 있는 Gibbs sampling, Metropolis algorithm, 그리고 Slice Sampling등의 몬테칼로 방법들을 비교한다.

주요용어 : bayesian neural networks, gibbs sampling, metropolis algorithm, slice sampling, hybrid monte carlo

1. 서론

일반적인 신경망(neural networks)구조에서 네트워크 모수들(network parameters)의 갱신과 새로운 입력값에 대한 예측값의 추정에 확률분포를 적용할 수 있다. 베이지안 추론은 무한 네트워크(infinite networks)를 포함한 신경망의 학습에 있어서 효율적인 결과를 제공해 준다. 베이지안 추론이 적용되지 않는 일반 신경망과는 달리 데이터들이 갖고 있는 규칙들을 확률적으로 결정해 주고, 예측에 대한 불확실성을 양적으로 나타내어 준다. 신경망 모형에 대한 베이지안 추론의 목적은 학습 데이터의 입력값(input value)과 목적값(target value)으로부터 다음 식과 같은 새로운 입력값에 대한 출력값의 예측분포(Predictive Distribution)를 구하는 것이다.

$$P(y^{(n+1)}|x^{(n+1)},(x^{(1)},y^{(1)}),\ldots,(x^{(n)},y^{(n)}))$$
$$= \int P(y^{(n+1)}|x^{(n+1)},\theta)P(\theta|(x^{(1)},y^{(1)}),\ldots,(x^{(n)},y^{(n)}))d\theta$$

여기서 θ 는 가중치와 bias로 구성된 네트워크의 모수들이다. 위의 예측분포를 이용하여 새로운 입력값 $x^{(n+1)}$에 대한 예측값은 다음 식과 같이 예측분포의 기대값으로 구한다.

$$\hat{y}_k^{(n+1)} = \int f_k(x^{(n+1)},\theta)P(\theta|(x^{(1)},y^{(1)}),\ldots,(x^{(n)},y^{(n)}))d\theta$$

하지만 사후분포의 구조는 매우 복잡하여 원시함수에 의한 정적분을 불가능하게 한다. 모수의

1) (402-751) 인천광역시 남구 용현동 253 인하대학교 통계학과 부교수
2) (402-751) 인천광역시 남구 용현동 253 인하대학교 통계학과 박사과정
3) (402-751) 인천광역시 남구 용현동 253 인하대학교 통계학과 석사과정

차원이 다차원(multi-dimension)일 경우는 수치 해석적인 정적분도 매우 어렵다. 때문에 확률적 추정에 의한 적분값의 계산인 몬테칼로 적분이 필요하다. 이러한 몬테칼로의 기법들 중에서 대표적인 방법들인 Gibbs Sampling, Metropolis Algorithm과 Hybrid Monte Carlo(HMC) 알고리즘을 비교하고, 아울러 최근에 제안된 Slice Sampling 방법과 HMC와의 비교도 수행한다.

2. Slice Sampling

균일분포를 주로 사용하는 Slice Sampling 알고리즘은 Gibbs Sampling과 Metropolis 방법보다도 간편하게 원하는 분포로부터 Sampling을 해 준다.(Neal, 1997)

2.1 표본 관측값의 Slice

표본추출을 위한 초기값으로부터 계속 갱신되어 가는 표본들은 다음 표본이 추출되는데 사용되어지는 구간인 slice를 갖게 된다. 이 슬라이스는 현재의 표본이 포함되며 크기는 보조변수를 이용하여 구한다.

2.2 Slice Sampling 알고리즘

Slice Sampling 알고리즘은 아래의 각 단계를 반복 수행한다.

step0) $f(x) \propto f^*(x)$로부터 표본을 뽑는다고 가정하자.

step1) $f^*(x_0)$를 계산한다. 여기서 x_0는 초기값이다.

step2) $U(0, f^*(x_0))$ 에서 수직 좌표인 보조변수 y 을 뽑는다.

step3) 뽑힌 y 값을 기준으로 수평으로 x_0를 포함하는 슬라이스 (x_l, x_r)를 만든다. 이 때 슬라이스는 $y < f^*(x)$인 부분이 된다.

step4) $U(x_l, x_r)$ 에서 새로운 x'를 뽑는다.

step5) $f^*(x')$를 계산한다.

step6) $f^*(x') > y$ 이면 뽑힌 x'을 받아들이고 그렇지 않으면 슬라이스 (x_l, x_r)를 수정한 후 step4)로 간다.

3. Hybrid Monte Carlo(HMC)

Hybrid Monte Carlo 알고리즘은 Gibbs Sampling기법과 Metropolis 알고리즘을 결합한 구조를 갖는다.(Duane, Kennedy, Pendleton, Roweth, 1987). HMC 알고리즘의 결과는 Monte Carlo 방법에 의한 기대치 추정에 사용될 표본이 된다. 이러한 HMC 알고리즘은 Hamitonian dynamic에 의한 Metropolis 갱신알고리즘에 의해 수행된다.

3.1 Hamiltonian 함수

네트워크 모수들의 집합(the set of network parameters)을 위치변수(position variable) q라고 하면 정준분포(canonical distribution)하에서 위치변수 q에 대한 확률밀도는 다음과 같이 정의된다.

$$f(q) \propto \exp(-E(q))$$

, 여기서 $E(q)$는 "potential energy" function $E(q) = -\log f(q) - \log Z$

그리고 q에 1대1로 대응되어지면서 dynamic method에 사용되는 운동량변수(momentum variable)를 p라 하면 q와 p에 대한 정준분포(canonical distribution)는 다음과 같이 정의된다.

$$f(q, p) \propto \exp(-H(q, p))$$

, 여기서 $H(q, p)$는 "Hamiltonian" function $H(q, p) = E(q) + K(p)$

, $K(p)$는 "kinetic energy" function $K(p) = \sum_{i=1}^{n} \frac{p_i^2}{2m_i}$

위 식에서 네트워크 모수들의 집합인 q와 운동량 변수인 p는 서로 독립이다. 또한 m_i는 각각의 p_i의 질량(mass)을 나타낸다.

3.2 Hamiltonian Dynamics

각각의 q와 p는 다음의 식에 의해 시간 (t)을 통하여 변화한다.

$$\frac{dq_i}{dt} = +\frac{\partial H}{\partial p_i} \qquad \frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

위 식에서 H는 Hamiltonian 함수이다.

3.3 Hybrid Monte Carlo(HMC) 알고리즘

HMC 알고리즘은 leapfrog step과 dynamical transition을 번갈아 가면서 반복 수행한다.

3.3.1 Leapfrog step

step1) $\hat{p}_i(t+\frac{\varepsilon}{2}) = \hat{p}_i(t) - \frac{\varepsilon}{2}\frac{\partial E}{\partial q_i}(\hat{q}(t))$, 여기서 ε는 stepsize이다.

step2) $\hat{q}_i(t+\varepsilon) = \hat{q}_i(t) + \varepsilon\frac{\hat{p}_i(t+\frac{\varepsilon}{2})}{m_i}$

step3) $\hat{p}_i(t+\varepsilon) = \hat{p}_i(t+\frac{\varepsilon}{2}) - \frac{\varepsilon}{2}\frac{\partial E}{\partial q_i}(\hat{q}(t+\varepsilon))$

step4) step1)부터 step3)을 L번 반복한다. , 여기서 L은 leapfrog step의 수이다.

3.3.2 Dynamical transition

step1) 초기상태 $(q,p)=(\hat{q}(0),\hat{p}(0))$로부터 ε의 stepsize로 leapfrog step을 L번 반복해서 $(\hat{q}(\varepsilon L),\hat{p}(\varepsilon L))$을 얻는다.

step2) 운동량변수들(momentum variables)을 negate해서 $(q^*,p^*)=(\hat{q}(\varepsilon L),-\hat{p}(\varepsilon L))$를 구한다.

step3) 이렇게 구해진 candidate state에 대해 Metropolis 알고리즘에서처럼 $\min(1,\exp(-(H(q^*,p^*)-H(q,p))))$의 확률로 받아들이고 그렇지 않으면 새로운 state는 이전의 state와 같게 한다.

4. 실험 및 결과

본 논문에서는 각각의 Monte Carlo 방법들을 비교하기 위하여 이변량 정규분포에서의 Monte Carlo Sampling을 수행하였다.

즉 이변량 정규분포

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} | X \sim N\left(\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$$

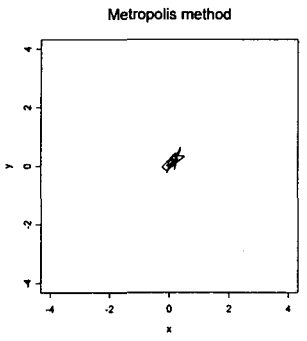
$$\rho=0.99 \quad (x_1,x_2)=(0,0)$$

에서 각각의 Monte Carlo 방법으로 Sampling을 다음과 같은 방법으로 실행하였다.

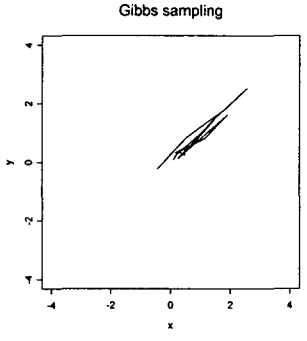
Metropolis Sampling은 stepsize를 0.15로 20번에 한번씩 sampling을 하였으며 Gibbs Sampling은 10번에 한번씩 sampling을 하였고, Slice Sampling은 stepsize를 0.15로 10번에 한번씩 sampling을 하였다. 마지막으로 Hybrid Monte Carlo는 stepsize를 0.15로 20번에 한번씩 sampling을 하였다.

아래의 simulation은 Neal의 Software for Flexible Bayesian Modeling and Markov Chain Sampling으로 실행하였다.

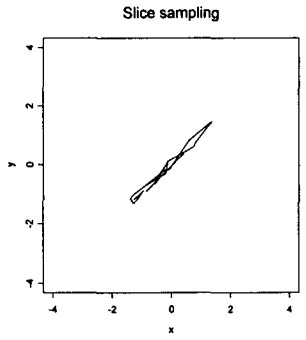
이와 같은 절차로 각 방법들의 sampling 유용성을 비교하기 위해서 원하는 전체 구간으로부터 표본이 추출되어졌는가에 대한 그림들을 그려서 비교하였다. 이변량 정규분포에서의 Metropolis, Gibbs, Slice, Hybrid Monte Carlo Sampling을 수행한 결과가 다음의 (그림6-1)부터 (그림6-4)까지에 보여진다.



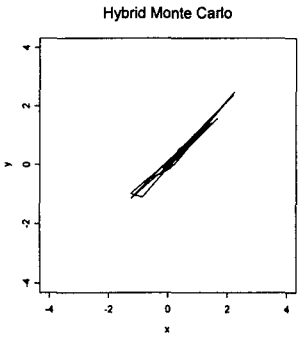
(그림6-1)



(그림6-2)



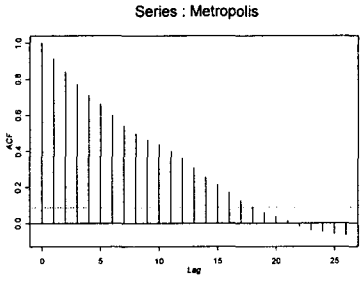
(그림6-3)



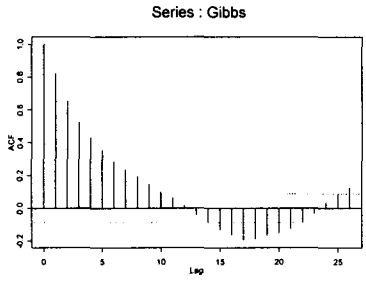
(그림6-4)

위 실험결과의 그림을 보면 (그림6-1)의 Metropolis 방법은 표본들이 추출되어야 할 전 구간에서 골고루 분포되어 있지 않고, 특정구간에서 집중적으로 모여 있음을 볼 수 있다. Gibbs sampling 방법도 전 구간 중에서 절반의 구간에서만 표본이 추출되어진 것을 볼 수 있다. 반면에 Slice sampling은 Metropolis나 Gibbs보다는 비교적 넓은 구간에서 표본이 추출되었지만 표본이 추출되어야 할 전체 구간에서는 sampling이 되지 못하는 것을 볼 수 있다. 하지만 Hybrid Monte Carlo Sampling은 표본이 전체구간에 걸쳐 골고루 추출되어진 것을 볼 수 있다.

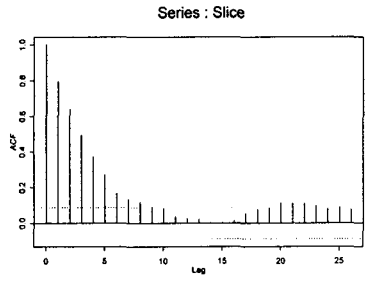
또다른 비교방법으로 (그림6-5)에서 (그림6-8)의 실험결과는 각 Monte Carlo sampling의 ACF(autocorrelation function)를 보여주고 있다. 다른 sampling방법보다 Hybrid Monte Carlo Sampling이 작은 lag값에서도 자기상관의 효과에서 벗어나는 것을 볼 수 있다.



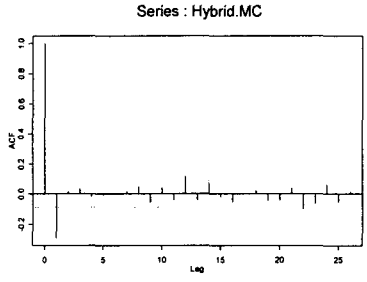
(그림6-5)



(그림6-6)



(그림6-7)



(그림6-8)

평균 rejection rate는 Metropolis sampling이 0.4152 이고, Hybrid Monte Carlo sampling은 0.16 으로서 Metropolis보다 낮은 것을 알 수 있다.

5. 결론

다른 몬테칼로 알고리즘들에 비해 HMC 알고리즘은 적은 수의 표본추출 작업으로도 서로 독립적인 표본을 얻을 수 있게 해주고, 또한 임의보행(random walk behaviour)도 감소시킨다. 특히 다차원 모수공간을 갖는 모델에 있어서 매우 우수한 표본추출 방법이 된다.
앞으로 HMC 알고리즘은 베이지안의 신경망을 포함한 무한 신경망의 점근된 예측분포의 기대값을 계산하는 데에도 사용되어질 수 있으리라 기대된다. 하지만 HMC 알고리즘의 계산시간을 줄이는 기법에 대한 연구도 필요하다고 생각된다.

참고문헌

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987) Hybrid Monte Carlo, Physics Letter B, vol. 195, pp. 216-222

Neal, R. M. (1993) Probabilistic inference using Markov Chain Monte Carlo methods, Technical Report CRG-TR-93-1, Dept. of Computer Science, Univ. of Toronto.

Neal, R. M. (1996) Bayesian Learning for Neural Networks, Number 118 in Lecture Notes in Statistics. New York: Springer.

Neal, R. M. (1997) Markov Chain Monte Carlo Methods Based on 'Slicing' the Density Function, Technical Report No. 9722, Department of Statistics, University of Toronto.