Bayesian Analysis for Exponential Random Graph Models Using the Double Metropolis-Hastings Sampler

Replicação: Gabriel Lima Novais

Conteúdo

Introdução

- Markov Chain Monte Carlo (MCMC) no contexto da modelagem Bayesiana deste artigo servirá para lidar com as constantes de normalização.
- Mas precisaremos encontrar a constante de normalização ao mesmo tempo que tentamos amostrar da posteriori.
- Dessa forma, uma solução proposta reside na aplicação do algoritmo de Metropolis Hastings com a adição de uma variável auxiliar.

Introdução

Considere o esquema a seguir:

- $h(x|\theta)$ um modelo de probabilidade não normalizado.
- $x \in X$ e $\theta \in \Theta$.
- $Z(\theta) = \int_X h(x|\theta)dx$ uma função de normalização.
- $p(\theta)$ a priori da densidade de θ.

Então a posteriori $\pi(\theta|x)$ será:

$$\pi(\theta|x) \propto p(\theta)\frac{h(x|\theta)}{Z(\theta)}$$

Introdução

- O problema na construção do MCMC reside no fato de que $Z(\theta)$ não consegue ser analiticamente calculada.
- A probabilidade de aceitção calculada em cada passo do MH implica em calcular $Z(\theta)$ tanto no valor atual como no valor proposto de θ.
- Uma forma de lidar com este problema pode ser com a inclusão de uma variável auxiliar. Exemplo: DMH.

Introdução

De maneira geral como funciona essa variável auxiliar:

- Introduza uma variável auxiliar $y \sim f(y|\theta, x)$
- Suponha que $\pi(\theta|x) \propto p(\theta)h(x|\theta)/Z(\theta)$
- Então a target aumentada será:

  $\pi(\theta, y|x) \propto f(y|\theta, x)p(\theta)h(x|\theta)/Z(\theta)$
- A densidade marginal será : $\int_X \pi(\theta, y|x)dy = \pi(\theta|x)$
- Considere a proposal conjunta: $q(\theta', y'|\theta, y) = q(y'|\theta')q(\theta'|\theta)$.
- Por fim tome convenientemente $q(y'|\theta')$ como $h(y'|\theta')/Z(\theta')$

Introdução

Agora o cálculo da probabilidade de aceitação do MH será tal que:

$$\alpha = min(1, \frac{\pi(\theta, y|x)q(\theta, y|\theta', y')}{\pi(\theta', y'|x)q(\theta', y'|\theta, y)}) \implies$$

$$\alpha = min(1, \frac{f(y'|\theta', x)p(\theta')h(x|\theta')Z(\theta)q(y|\theta)q(\theta|\theta')}{f(y|\theta, x)p(\theta)h(x|\theta)Z(\theta')q(y'|\theta')q(\theta'|\theta)}) \implies$$

$$\alpha = min(1, \frac{f(y'|\theta', x)p(\theta')h(x|\theta')Z(\theta)h(y|\theta)Z(\theta')q(\theta|\theta')}{f(y|\theta, x)p(\theta)h(x|\theta)Z(\theta')h(y'|\theta')Z(\theta)q(\theta'|\theta)}) \implies$$

$$\alpha = min(1, \frac{f(y'|\theta', x)p(\theta')h(x|\theta')h(y|\theta)q(\theta|\theta')}{f(y|\theta, x)p(\theta)h(x|\theta)h(y'|\theta')q(\theta'|\theta)})$$

Introdução

- Uma variável auxiliar "bem escolhida", quando no cálculo da razão de aceitação, termina por cancelar a função de normalização e assim o MH pode ser aplicado.
- A aplicação do método de DMH será realizada em cima da modelagem de redes com uma função de distribuição.
- Objetivo: recuperar certos parâmetros desta rede e realizar uma comparação com outro método similar.

Exponential Random Graphs Models (ERGM)

- A família de grafos exponenciais aleatórios (ERGM) são muito utilizados para modelar redes sociais.
- Dizemos que a rede pertence a uma ERGM se a probabilidade de um nó se a ligar a outro é determinada por uma função de densidade exponencial, que geralmente é descrita segundo a seguinte equação:

$$f(x|\theta) = \frac{exp(\sum_i \theta_i S_i(x))}{k(\theta)}$$

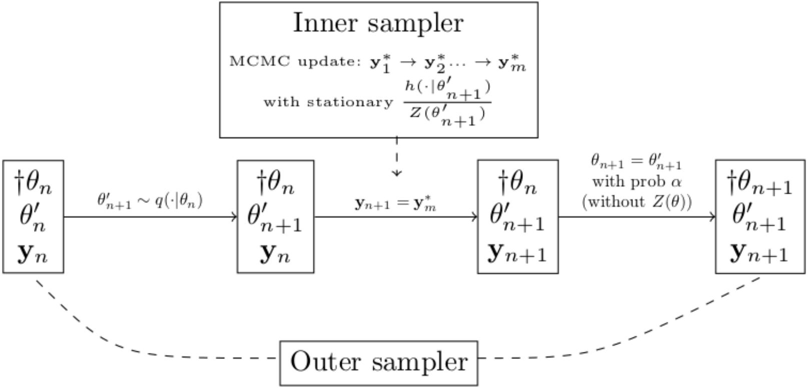
Exponential Random Graphs Models (ERGM)

- A função de densidade acima é função do input x, que denota a matrix de adjacência da rede. Logo, a probabilidade seria algo como o elemento da matriz ser 0 ou 1.
- O objetivo será estimar cada parâmetro θ_i associado a cada estatística suficiente $S_i(x)$ do numerador. O denominador $k(\theta)$ é a constante de normalização intratável.
- Veja que a aplicação do DMH surge como uma solução viável para estimar θ_i com $k(\theta)$ pela função de densidade.

Double Metropolis Hastings (DMH)

- Liang (2009) propôs o algoritmo como uma solução para amostrar de funções cujas constantes de normalização são intratáveis.
- O que o DMH faz é justamente aumentar o espaço estado da cadeia de Markov utilizando uma variável auxiliar no Metropolis Hastings tradicional.
- Pelo fato dessa variável auxiliar passar por outro Metropolis Hastings dentro do principal, o nome concebido foi o de "Double Metropolis Hastings".

Double Metropolis Hastings (DMH)



Double Metropolis Hastings (DMH)

The DMH Sampler

(a) Simulate a new sample θ' from $\pi(\theta)$ using the MH algorithm starting with θ_t.

(b) Generate an auxiliary variable y from $f(y|\theta')$ through m MH updates starting with x. The probability of transition from x to y is

$$P_{\theta'}^{(m)}(y|x) = K_{\theta'}(x \to x_1) \cdots K_{\theta'}(x_{m-1} \to y) \tag{4}$$

Accept y with probability $\min\{1, r(\theta_t, \theta', y|x)\}$, where

$$r(\theta_t, \theta', y|x) = \frac{f(y|\theta_t) P_{\theta'}^{(m)}(x|y)}{f(x|\theta_t) P_{\theta'}^{(m)}(y|x)} = \frac{f(y|\theta_t) f(x|\theta')}{f(x|\theta_t) f(y|\theta')} = \frac{\psi(y,\theta_t)\psi(x,\theta')}{\psi(x,\theta_t)\psi(y,\theta')}. \tag{5}$$

(c) Set $\theta_{t+1} = \theta'$ if the auxiliary variable is accepted in step (b), and set $\theta_{t+1} = \theta_t$ otherwise.

Noisy Double Metropolis Hastings (NDMH)

- Consiste em certas adaptações do DMH. Em que constam duas modificações principais:
  - (1) Ao invés de utilizarmos uma variável auxiliar apenas, acrescentam-se N variáveis de forma que o Inner-Sample seja feito para todas estas N variáveis.
  - (2) A razão de aceitação é então modificada e assim teremos que a probabilidade de aceitação do novo θ no segundo MH dependerá dessas variáveis acrescentadas.

Noisy Double Metropolis Hastings (NDMH)

A nova razão de aceitação pode ser então descrita por:

$$r(\theta_t, \theta', y|x) = \frac{\psi(x, \theta')}{\psi(x, \theta_t)} \frac{1}{N} \sum_{i=1}^{N} \frac{\psi(y_i, \theta_t)}{\psi(y_i, \theta')}$$

E então teremos que:

$$\alpha = min(1, r(\theta_t, \theta', y|x))$$

Aplicação e Resultados

- A base de dados escolhida para fazer a aplicação do DMH e NDMH é a Florentine, uma rede que é amplamente utilizada como "toy model".
- A rede possui 10 nós, formando uma matriz de adjacência de 10x10 de dimensão. Não existem nós que se ligam a si mesmos e as arestas não são direcionadas.
- Além desta base procurou-se utilizar uma outra simulada composta de 50 nós, e verificou-se pelo MCMCMLE o valor dos seus parâmetros. Neste caso, conforme veremos mais a frente, a simulação obteve um tempo de execução alto.

Aplicação e Resultados

- O procedimento para elaboração do DMH foi a seguinte:
  1. Tomar θ_0 de uma normal multivariada com média zero e variância alta.
  2. Considerar a evolução dos θ_t como um random walk cujo step size modifica a variância da normal e a média é atualizada pelo θ_{t-1}
  3. Atualizar os elementos da matriz de adjacência por uma probailidade específica, por meio de uma rodada única de Gibbs Sampler.
  4. Com a matriz atualizada calcular a razão de aceitação e verificar se o θ_t é aceito e assim finalizar o MH principal.

Aplicação e Resultados

- Custos computacionais: além de precisar de muitas iterações (por ser uma técnica de Monte Carlos), o algoritmo também precisa varrer uma matriz cuja dimensão pode ser grande.
- Tanto para o DMH quanto para o NDMH verifica-se a ordem do algoritmo de n^2. Entretanto como a aplicação é voltada para redes, verifica-se que no máximo a complexidade se torna n^4.
- Grande parte dos custos relaciona-se não somente ao MH, mas também ao cálculo das estatísticas suficentes em especial a EP(x)

Aplicação e Resultados

- Os dados foram obtidos e gerados pelo R. A implementação do DMH foi incialmente feita no Python, mas pelo fato de não executar em tempo viável, optou-se por implementar em Julia, que nos forneceu um tempo de execução aproximadamente 5 vezes menor.
- Para a rede "Florentine" os parâmetros procurados pela estimação do modelo 2 eram de $\theta_1 = -1.66$ e $\theta_2 = 0.15$. Já para o de 50 nós simulado o resultado deveria fornecer algo próximo de $\theta_1 = -3.19$ e $\theta_2 = 0.39$.

Aplicação e Resultados

- Foram utilizados os parâmetros a seguir para ambas as redes:
  - dim $= 2$
  - $\tau = 0.25$
  - $n_{iter} = 10500/1000$
  - $var_{init} = 200$
  - step-size $= 0.2$
- obs: para o NDMH foram utilizadas 3 variáveis auxiliares.

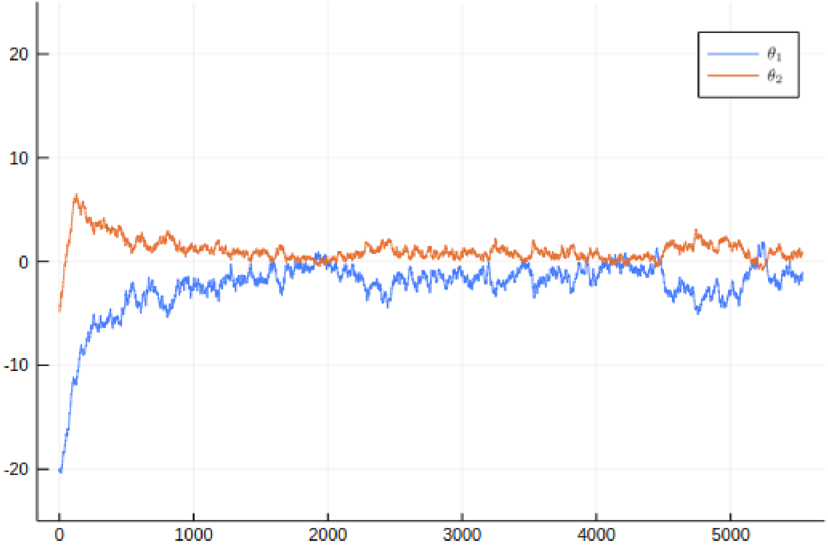
Aplicação e Resultados

Quadro comparativo entre os métodos:

Método	Rede	Tempo	Iteração	θ_1	θ_2
NDMH	10	26 min e 27 s	10500	-0.796426	0.405188
DMH	10	6 min e 51 s	10500	-1.74165	0.882828
DMH	50	2h 47 min 57 s	1000	-3.27527	0.159238

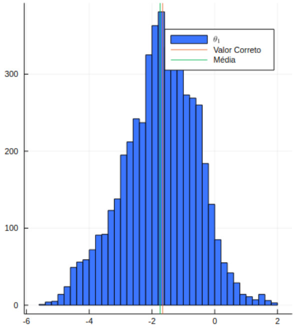
Aplicação e Resultados

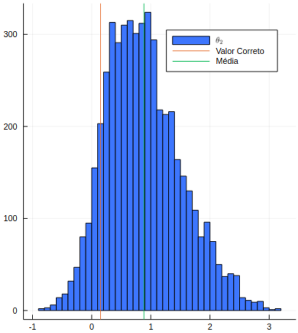
DMH - rede: 10 nós



Aplicação e Resultados

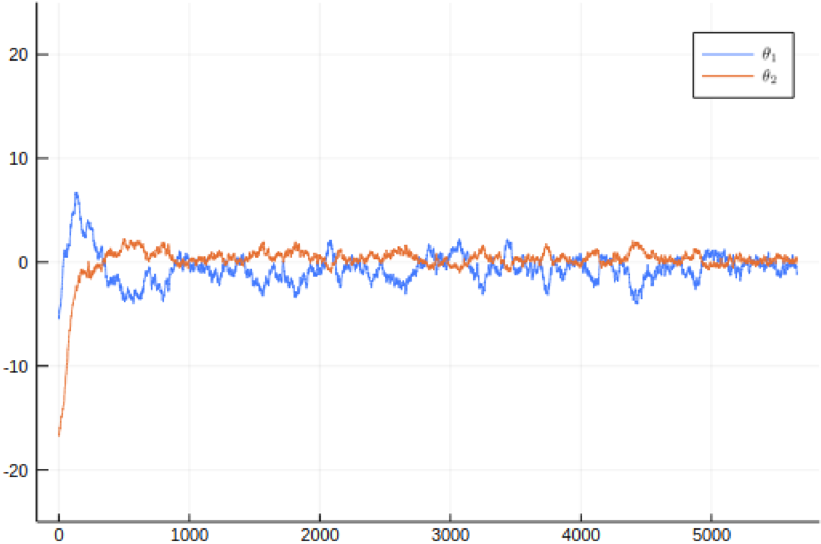
DMH - rede: 10 nós





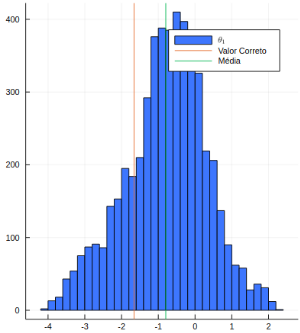
Aplicação e Resultados

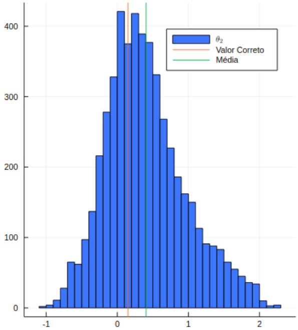
NDMH - rede: 10 nós



Aplicação e Resultados

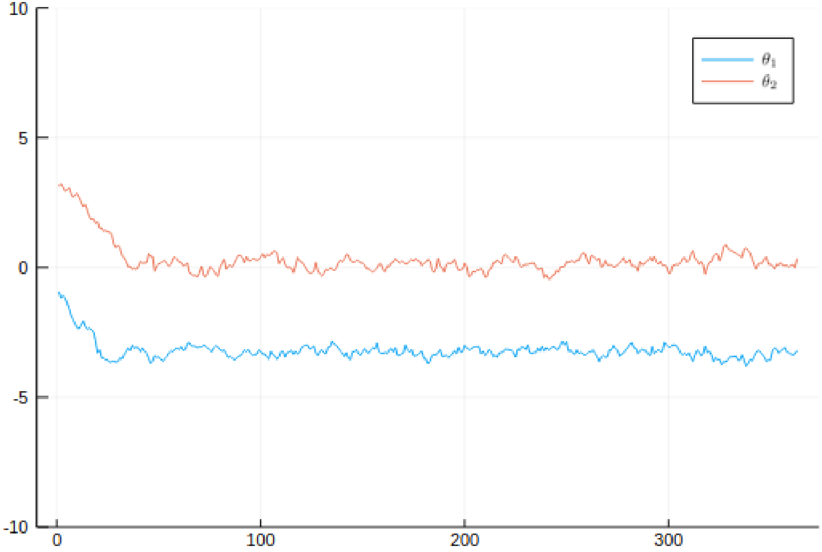
NDMH - rede: 10 nós





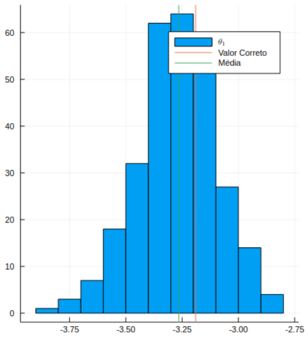
Aplicação e Resultados

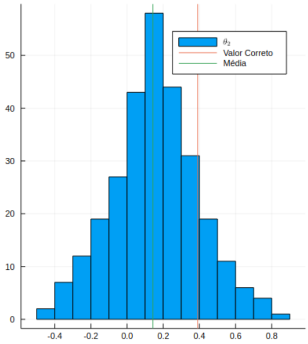
DMH - rede: 50 nós



Aplicação e Resultados

DMH - rede: 50 nós





Referências

- PARK, Jaewoo; HARAN, Murali. Bayesian inference in the presence of intractable normalizing functions.
- JIN, Ick Hoon; LIANG, Faming. Bayesian analysis for exponential random graph model using double Metropolis-Hastings sampler.
- LIANG, Faming; JIN, Ick-Hoon. A Monte Carlo Metropolis-Hastings algorithm for sampling from distributions with intractable normalizing constants.
- STIVALA, Alex; ROBINS, Garry; LOMI, Alessandro. Exponential random graph model parameter estimation for very large directed networks..