Laboratório Nacional de Computação Científica

UNIDADE DE PESQUISA DO MCTI

Algoritmo de Metropois-Hastings e diagnóstico de Cadeias

Modelos Compartimentais em Epidemiologia e Inferência Bayesiana

Gustavo Libotte e Regina Almeida

28 de janeiro de 2022

O algoritmo de Metropolis-Hastings

LNCC

- Lembre-se de que uma cadeia de Markov é uma sequência $\left\{x^{(1)}, x^{(2)}, \ldots\right\}$ tal que o mecanismo que gera $x^{(s+1)}$ pode depender do valor de $x^{(s)}$ mas não de $\left\{x^{(s-1)}, x^{(s-2)}, \ldots x^{(1)}\right\}$.
- Uma maneira mais "poética" de colocar isso é que para uma cadeia de Markov *o futuro depende do presente e não do passado*.
- O amostrador de Gibbs e o algoritmo de Metropolis são formas de gerar cadeias de Markov que se aproximam de uma distribuição de probabilidade alvo $p_0(x)$ para uma variável aleatória potencialmente com valor x.
- Na análise Bayesiana, x é tipicamente um parâmetro ou vetor de parâmetros e $p_0(x)$ é uma distribuição *a posteriori*, mas o amostrador de Gibbs e o algoritmo de Metropolis são usados de forma mais ampla.
- Agora vamos ver que esses dois algoritmos são de fato casos especiais de um algoritmo mais geral, chamado **algoritmo de Metropolis-Hastings**.
- Em seguida, descreveremos por que as cadeias de Markov geradas pelo algoritmo de Metropolis-Hastings são capazes de aproximar uma distribuição de probabilidade alvo.

O algoritmo de Metropolis-Hastings

- Primeiro, consideraremos um exemplo simples em que nossa distribuição de probabilidade alvo é $p_0(u, v)$, uma distribuição bivariada para duas variáveis aleatórias U e V.
- No problema normal de uma amostra, por exemplo, teríamos $U = \theta, V = \sigma^2$ e $p_0(u, v) = p\left(\theta, \sigma^2 \mid \boldsymbol{y}\right)$.
- Lembre-se de que o **amostrador de Gibbs** procede por amostragem iterativa de valores de U e V de suas distribuições condicionais.
- Dado $x^{(s)} = \left(u^{(s)}, v^{(s)}\right)$, um novo valor de $x^{(s+1)}$ é gerado do seguinte modo:
  - atualize U : amostre $u^{(s+1)} \sim p_0\left(u \mid v^{(s)}\right)$;
  - atualize V : amostre $v^{(s+1)} \sim p_0\left(v \mid u^{(s+1)}\right)$.
- Alternativamente, poderíamos ter primeiro amostrado $v^{(s+1)} \sim p_0\left(v \mid u^{(s)}\right)$ e então $u^{(s+1)} \sim p_0\left(u \mid v^{(s+1)}\right)$.

O algoritmo de Metropolis-Hastings

- Em contraste, o algoritmo de Metropolis propõe mudanças em $X = (U, V)$ e então aceita ou rejeita essas mudanças com base em p_0.
- Uma maneira alternativa de implementar o algoritmo de Metropolis é propor e depois aceitar ou rejeitar alterações em um elemento por vez:

1. atualize U:
   - amostre $u^* \sim J_u\left(u \mid u^{(s)}\right)$;
   - compute $r = p_0\left(u^*, v^{(s)}\right) / p_0\left(u^{(s)}, v^{(s)}\right)$;
   - defina $u^{(s+1)}$ para u^* ou $u^{(s)}$ com probabilidade $\min(1, r)$ e $\max(0, 1 - r)$.
2. atualize V:
   - amostre $v^* \sim J_v\left(v \mid v^{(s)}\right)$;
   - compute $r = p_0\left(u^{(s+1)}, v^*\right) / p_0\left(u^{(s+1)}, v^{(s)}\right)$;
   - defina $v^{(s+1)}$ para v^* ou $v^{(s)}$ com probabilidade $\min(1, r)$ e $\max(0, 1 - r)$.

O algoritmo de Metropolis-Hastings

- Este algoritmo de Metropolis gera propostas de J_u e J_v e as aceita com alguma probabilidade $\min(1, r)$.
- Da mesma forma, cada passo do amostrador de Gibbs pode ser visto como gerando uma proposta a partir de uma distribuição condicional completa e então aceitando-a com probabilidade 1.
- O algoritmo de Metropolis-Hastings **generaliza** ambas as abordagens ao permitir distribuições de propostas arbitrárias.
- Um algoritmo de Metropolis-Hastings para aproximar $p_0(u, v)$ é executado da seguinte forma:

1. atualize U:
   - amostre $u^* \sim J_u\left(u \mid u^{(s)}, v^{(s)}\right)$;
   - calcular a taxa de aceitação:

$$r = \frac{p_0\left(u^*, v^{(s)}\right)}{p_0\left(u^{(s)}, v^{(s)}\right)} \times \frac{J_u\left(u^{(s)} \mid u^*, v^{(s)}\right)}{J_u\left(u^* \mid u^{(s)}, v^{(s)}\right)}$$

   - defina $u^{(s+1)}$ para u^* ou $u^{(s)}$ com probabilidade $\min(1, r)$ e $\max(0, 1 - r)$.

O algoritmo de Metropolis-Hastings

2. atualize V:
   - amostre $v^* \sim J_v\left(v \mid u^{(s+1)}, v^{(s)}\right)$;
   - calcular a taxa de aceitação:

$$r = \frac{p_0\left(u^{(s+1)}, v^*\right)}{p_0\left(u^{(s+1)}, v^{(s)}\right)} \times \frac{J_v\left(v^{(s)} \mid u^{(s+1)}, v^*\right)}{J_v\left(v^* \mid u^{(s+1)}, v^{(s)}\right)}$$

   - defina $v^{(s+1)}$ para v^* ou $v^{(s)}$ com probabilidade $\min(1, r)$ e $\max(0, 1-r)$.

- Neste algoritmo, as distribuições de proposta J_u e J_v **não precisam ser simétricas**.
- Na verdade, o único requisito é que elas não dependam de valores U ou V em nossa sequência anterior aos valores mais atuais. Esse requisito garante que a sequência seja uma **cadeia de Markov**.
- O algoritmo de Metropolis-Hastings **se parece muito** com o algoritmo Metropolis, exceto que a razão de aceitação contém um fator extra, a razão entre a probabilidade de gerar o valor atual da proposta e a probabilidade de gerar a proposta a partir do atual.
- Isso pode ser visto como um "fator de correção": Se um valor u^* é muito mais provável de ser proposto do que o valor atual $u^{(s)}$, então devemos reduzir a probabilidade de aceitar u^* de acordo, caso contrário o valor u^* será **superrepresentado** em nossa sequência.

O algoritmo de Metropolis-Hastings

- Uma **forma mais geral** do algoritmo de Metropolis-Hastings é a seguinte: dado um valor atual $x^{(s)}$ de X,
  1. Gere x^* de $J_s\left(x^* \mid x^{(s)}\right)$;
  2. Calcular a taxa de aceitação
  $$r = \frac{p_0\left(x^*\right)}{p_0\left(x^{(s)}\right)} \times \frac{J_s\left(x^{(s)} \mid x^*\right)}{J_s\left(x^* \mid x^{(s)}\right)}$$
  3. Amostre $u \sim \text{Uniforme}(0,1)$. Se $u < r$ defina $x^{(s+1)} = x^*$, ou $x^{(s+1)} = x^{(s)}$ caso contrário.
- Observe que a distribuição de proposta também pode depender do número de iteração s.
- A principal restrição que colocamos em $J_s\left(x^* \mid x^{(s)}\right)$ é que ela não depende de valores na sequência anterior a $x^{(s)}$.
- Essa restrição garante que o algoritmo gere uma **cadeia de Markov**.
- Também queremos escolher J_s para que a cadeia de Markov possa convergir para a distribuição alvo p_0.

O algoritmo de Metropolis-Hastings

- Por exemplo, queremos ter certeza de que todo valor de x tal que $p_0(x) > 0$ será eventualmente proposto (e assim aceito em uma fração do tempo), independentemente de onde começamos a cadeia de Markov.
- Um exemplo em que este não é o caso é onde os valores de X com probabilidade diferente de zero são os números inteiros, e $J_s\left(x^* \mid x^{(s)}\right)$ propõe $x^{(s)} \pm 2$ com igual probabilidade.
- Neste caso, o algoritmo de Metropolis-Hastings produz uma cadeia de Markov, mas a cadeia só irá gerar números pares se $x^{(1)}$ for par, e apenas número ímpar se $x^{(1)}$ for ímpar.
- Este tipo de cadeia de Markov é chamado **redutível**, pois o conjunto de possíveis valores X pode ser dividido em conjuntos não sobrepostos (inteiros pares e ímpares neste exemplo), entre os quais o algoritmo não pode se mover.
- Em contraste, queremos que nossa cadeia de Markov seja **irredutível**, ou seja, capaz de ir de qualquer valor de X para qualquer outro, eventualmente.

O algoritmo de Metropolis-Hastings

- Além disso, queremos que J_s seja tal que a cadeia de Markov seja **aperiódica** e **recorrente**.
- Um valor x é periódico com período $k > 1$ em uma cadeia de Markov se ele só puder ser visitado a cada k iterações. Se x é periódico, então para cada S há um número infinito de iterações $s > S$ para as quais $\mathrm{P}\left(x^{(s)} = x\right) = 0$.
- Como queremos que a distribuição de $x^{(s)}$ convirja para p_0, devemos ter certeza de que se $p_0(x) > 0$, então x **não é periódico** em nossa cadeia de Markov.
- Uma cadeia de Markov sem estados periódicos é chamada de **aperiódica**.
- Finalmente, se $x^{(s)} = x$ para alguma iteração s, isso deve significar que $p_0(x) > 0$. Portanto, queremos que nossa cadeia de Markov seja capaz de retornar a x de tempos em tempos enquanto executamos nossa cadeia (caso contrário, a fração relativa de x na cadeia irá para zero, mesmo que $p_0(x) > 0$).
- Um valor x é dito **recorrente** se, quando continuamos a executar a cadeia de Markov a partir de x, temos a garantia de eventualmente retornar a x.
- Claramente, queremos que todos os valores possíveis de X sejam recorrentes em nossa cadeia de Markov.

O algoritmo de Metropolis-Hastings

- Uma cadeia de Markov **irredutível, aperiódica e recorrente** é um objeto muito bem comportado.
- Um teorema da teoria da probabilidade diz que a distribuição empírica de amostras geradas a partir de tal cadeia de Markov irá convergir:

Teorema Ergódico

Se $\left\{x^{(1)}, x^{(2)}, \ldots\right\}$ é uma cadeia de Markov irredutível, aperiódica e recorrente, então existe uma distribuição de probabilidade única π tal que $s \to \infty$,

- $\mathrm{P}\left(x^{(s)} \in A\right) \to \pi(A)$ para qualquer conjunto A.
- $\frac{1}{s}\sum g\left(x^{(s)}\right) \to \int g(x)\pi(x)\mathrm{d}x$.

- A distribuição π é chamada de **distribuição estacionária** da cadeia de Markov e tem a seguinte propriedade:
  - Se $x^{(s)} \sim \pi$, e $x^{(s+1)}$ for gerado da cadeia de Markov começando em $x^{(s)}$, então $\mathrm{P}\left(x^{(s+1)} \in A\right) = \pi(A)$.

Introdução ao diagnóstico MCMC

- O propósito da aproximação de Monte Carlo ou da aproximação de Monte Carlo via cadeias de Markov é obter uma sequência de valores de parâmetros $\left\{\phi^{(1)}, \ldots, \phi^{(S)}\right\}$ tal que

$$\frac{1}{S}\sum_{s=1}^{S} g\left(\phi^{(s)}\right) \approx \int g(\phi)p(\phi)d\phi$$

  para quaisquer funções g de interesse.

- Em outras palavras, queremos a média empírica de $\left\{g\left(\phi^{(1)}\right), \ldots, g\left(\phi^{(S)}\right)\right\}$ para aproximar o valor esperado de $g(\phi)$ sob uma distribuição de probabilidade alvo $p(\phi)$ (a distribuição *a posteriori*).
- Para que esta seja uma boa aproximação para uma ampla gama de funções g, precisamos que distribuição empírica da sequência simulada $\left\{\phi^{(1)}, \ldots, \phi^{(S)}\right\}$ se "pareça" com a distribuição $p(\phi)$.
- Aproximações de Monte Carlo ou de Monte Carlo via cadeias de Markov são duas maneiras de gerar tal sequência.

Introdução ao diagnóstico MCMC

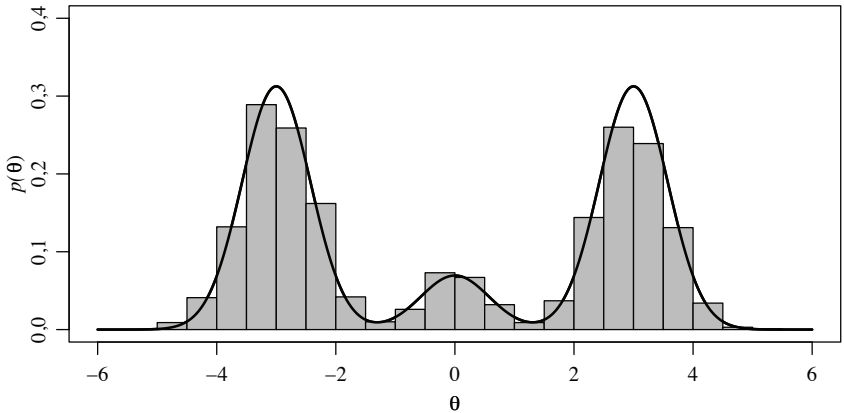
- Amostras de MC **independentes** criam automaticamente uma sequência que representa $p(\phi)$: a probabilidade de que $\phi^{(s)} \in A$ para qualquer conjunto A seja $\int_A p(\phi)d\phi$.
- Isso **é verdade** para cada $s \in \{1, \ldots, S\}$ e condicionalmente ou incondicionalmente nos outros valores na sequência.
- Isso **não é verdade** para amostras MCMC, caso em que tudo o que temos certeza é que

$$\lim_{s\to\infty} \mathrm{P}\left(\phi^{(s)} \in A\right) = \int_A p(\phi)d\phi.$$

- Vamos explorar as diferenças entre MC e MCMC com um exemplo simples.
- Nossa distribuição alvo será a distribuição de probabilidade conjunta de duas variáveis: uma variável discreta $\lambda \in \{1, 2, 3\}$ e uma variável contínua $\theta \in \mathbb{R}$.
- A densidade alvo para este exemplo será definida como $\{\mathrm{P}(\lambda = 1), \mathrm{P}(\lambda = 2), \mathrm{P}(\lambda = 3)\} = (0{,}45,\ 0{,}10,\ 0{,}45)$ e $p(\theta \mid \lambda) = \mathcal{N}(\theta, \mu_\lambda, \sigma_\lambda)$, onde
  - $(\mu_1, \mu_2, \mu_3) = (-3, 0, 3)$
  - $(\sigma_1^2, \sigma_2^2, \sigma_3^2) = (1/3, 1/3, 1/3)$

Introdução ao diagnóstico MCMC

- Esta é uma mistura de três densidades normais, onde podemos pensar em λ como sendo uma variável de associação de grupo e $(\mu_\lambda, \sigma^2_\lambda)$ como a média e variância da população para o grupo λ.



Introdução ao diagnóstico MCMC

LNCC

- É muito fácil obter amostras **independentes** de Monte Carlo a partir da distribuição conjunta de $\phi = (\lambda, \theta)$.
- Primeiro, um valor de λ é amostrado de sua distribuição marginal, então o valor é inserido em $p(\theta \mid \lambda)$, do qual um valor de θ é amostrado.
- O par amostrado (λ, θ) representa uma amostra da distribuição conjunta de $p(\lambda, \theta) = p(\lambda)p(\theta \mid \lambda)$.
- A distribuição empírica das amostras θ fornece uma aproximação da distribuição marginal $p(\theta) = \sum p(\theta \mid \lambda)p(\lambda)$.
- Um histograma de 1.000 valores de θ gerados pela aproximação de Monte Carlo dessa maneira é mostrado na figura anterior.
- A distribuição empírica das amostras de Monte Carlo parece aproximar bem $p(\theta)$.

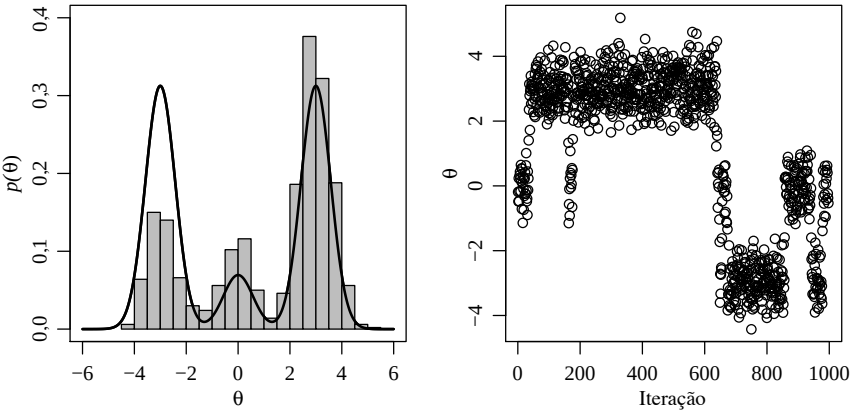
Introdução ao diagnóstico MCMC

- Também é simples construir um amostrador de Gibbs para $\phi = (\lambda, \theta)$.
- Um amostrador de Gibbs amostraria alternadamente valores de θ e λ de suas distribuições condicionais completas.
- A distribuição condicional completa de θ já foi fornecida, e usando a regra de Bayes podemos mostrar que a distribuição condicional completa de λ é dada por

$$\mathrm{P}(\lambda = d \mid \theta) = \frac{\mathrm{P}(\lambda = d) \times \mathcal{N}(\theta, \mu_d, \sigma_d)}{\sum_{d=1}^{3} \Pr(\lambda = d) \times \mathcal{N}(\theta, \mu_d, \sigma_d)}, \text{ para } d \in \{1, 2, 3\}$$

- A figura a seguir mostra um histograma de 1.000 amostras de θ gerados com o amostrador Gibbs.
- Observe que a distribuição empírica das amostras MCMC fornece uma aproximação ruim para $p(\theta)$. Valores de θ próximos a -3 estão sub-representados, enquanto valores próximos de zero e $+3$ estão super-representados.

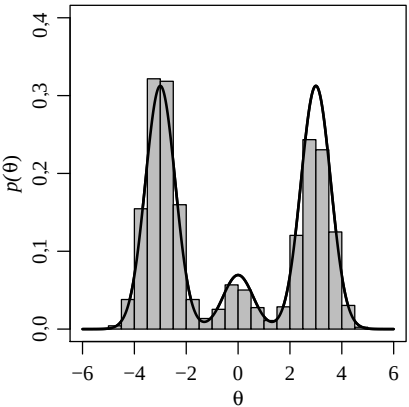
Introdução ao diagnóstico MCMC

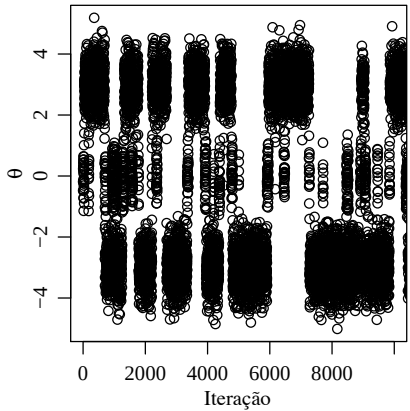


Introdução ao diagnóstico MCMC

- O que deu errado? Um gráfico dos valores θ versus o número de iteração no segundo painel da figura nos esclarece.
- Os valores θ ficam "presos" em certas regiões e raramente se movem entre as três regiões representadas pelos três valores de μ.
- O termo técnico para essa "aderência" é **autocorrelação**, ou **correlação entre valores consecutivos da cadeia**.
- Neste amostrador de Gibbs, se tivermos um valor de θ próximo de 0, por exemplo, o próximo valor de λ provavelmente será 2 .
- Se λ for 2, então o próximo valor de θ provavelmente estará próximo de 0, resultando em um alto grau de correlação positiva entre valores consecutivos de θ na cadeia.
- O amostrador de Gibbs não é garantido para fornecer uma boa aproximação para $p(\theta)$? Sim, mas *eventualmente* pode **levar muito tempo** em algumas situações.
- A figura a seguir indica que nossa aproximação melhorou muito depois de usar 10.000 iterações do amostrador Gibbs, embora ainda seja um pouco inadequada.

Introdução ao diagnóstico MCMC





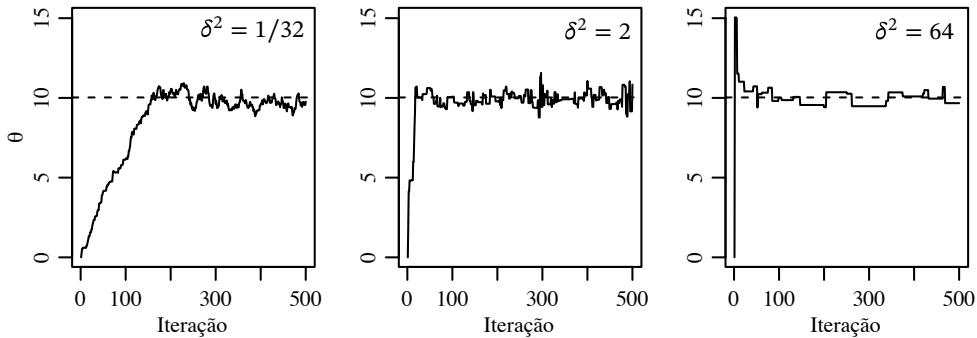
Introdução ao diagnóstico MCMC

LNCC

- Os valores θ gerados a partir de um algoritmo MCMC são estatisticamente **dependentes**.
- **Quanto maior a correlação, mais tempo levará para a cadeia de Markov atingir a estacionariedade** e mais iterações serão necessárias para obter uma boa aproximação de $p(\theta \mid y)$.
- *Grosso modo*, a quantidade de informações que obtemos sobre $\mathbb{E}[\theta \mid y]$ de amostras S positivamente correlacionadas é **menor** do que a informação que obteríamos de amostras independentes de S.
- Quanto mais correlacionada for nossa cadeia de Markov, menos informações obteremos **por iteração**.
- Na amostragem de Gibbs **não temos muito controle** sobre a correlação da cadeia de Markov, mas com o algoritmo Metropolis a correlação pode ser ajustada selecionando um **valor ótimo** de δ na distribuição proposta.
- Ao selecionar δ com cuidado, podemos **diminuir a correlação** na cadeia de Markov, levando a um aumento na taxa de convergência, um aumento no tamanho efetivo da amostra da cadeia de Markov e uma melhoria na aproximação de Monte Carlo para a posterior distribuição.

Introdução ao diagnóstico MCMC

- Para ilustrar isso, podemos executar novamente o algoritmo de Metropolis usando outros valores para δ, $\delta^2 \in \{1/32, 2, 64\}$.
- A Figura a seguir traça os primeiros 500 valores para as sequências com $\delta^2 \in \{1/32, 2, 64\}$.



Introdução ao diagnóstico MCMC

- No primeiro painel onde $\delta^2 = 1/32$, a pequena variação da proposta significa que θ^* estará muito próximo de $\theta^{(s)}$, e assim $r \approx 1$ para a maioria dos valores propostos.
- Como resultado, θ^* é aceito como o valor de $\theta^{(s+1)}$ para 87% das iterações.
- Embora essa alta taxa de aceitação mantenha a cadeia em movimento, os movimentos nunca são muito grandes e, portanto, a cadeia de Markov é altamente correlacionada.
- Uma consequência disso é que leva um grande número de iterações para a cadeia de Markov passar do valor inicial de zero para a moda da *posteriori* de 10,03.
- No outro extremo, o terceiro gráfico da figura mostra a cadeia de Markov para $\delta^2 = 64$.
- Nesse caso, a cadeia se move rapidamente para a moda da *posteriori*, mas, uma vez lá, fica "presa" por longos períodos.
- Isso ocorre porque a variância da distribuição proposta é tão grande que θ^* está frequentemente muito distante da moda da *posteriori*.
- Amostras neste algoritmo Metropolis são aceitas para apenas 5% das iterações, então $\theta^{(s+1)}$ é igual a $\theta^{(s)}$ em 95% do tempo, resultando em uma cadeia de Markov altamente correlacionada.

Introdução ao diagnóstico MCMC

- Para construir uma cadeia de Markov com **baixa correlação**, precisamos de uma variância grande o suficiente para que a cadeia de Markov possa se mover rapidamente pelo espaço de parâmetros, mas não tão grande que as amostras acabem sendo rejeitadas na maioria das vezes.
- Dentre as variâncias propostas consideradas aqui, esse balanço foi encontrado com δ^2 igual a 2, o que dá uma taxa de aceitação de 35%.
- Em geral, é uma prática comum realizar várias execuções curtas do algoritmo de Metropolis sob diferentes valores de δ até encontrar uma que forneça uma taxa de aceitação **aproximadamente entre 20 e 50%**.
- Uma vez que um valor razoável de δ é selecionado, uma cadeia de Markov **mais longa e eficiente** pode ser executada.
- Alternativamente, podem ser construídas **versões modificadas** do algoritmo de Metropolis que alteram de forma adaptativa o valor de δ no início da cadeia para encontrar automaticamente uma boa distribuição.