Scalable Metropolis-Hastings for Exact Bayesian Inference with Large Datasets

Rob Cornish   Paul Vanetti
Alexandre Bouchard-Côté   George Deligiannidis   Arnaud Doucet

July 19, 2020

Problem

Bayesian inference via MCMC is **expensive** for large datasets

Problem

Consider a posterior over **parameters** θ given n **data points** y_i:

$$\pi(\theta) = p(\theta|y_{1:n}) \propto p(\theta) \prod_{i=1}^{n} p(y_i|\theta).$$

Problem

Consider a posterior over **parameters** θ given n **data points** y_i:

$$\pi(\theta) = p(\theta|y_{1:n}) \propto p(\theta) \prod_{i=1}^{n} p(y_i|\theta).$$

Metropolis–Hastings

Given a proposal q and current state θ:

1. Propose $\theta' \sim q(\theta, \cdot)$
2. Accept θ' with probability

$$\alpha_{\mathrm{MH}}(\theta, \theta') := 1 \wedge \frac{q(\theta', \theta)\pi(\theta')}{q(\theta, \theta')\pi(\theta)} = 1 \wedge \frac{q(\theta', \theta)p(\theta')}{q(\theta, \theta')p(\theta)} \prod_{i=1}^{n} \frac{p(y_i|\theta')}{p(y_i|\theta)}$$

Problem

Consider a posterior over **parameters** θ given n **data points** y_i:

$$\pi(\theta) = p(\theta|y_{1:n}) \propto p(\theta) \prod_{i=1}^{n} p(y_i|\theta).$$

Metropolis–Hastings

Given a proposal q and current state θ:

1. Propose $\theta' \sim q(\theta, \cdot)$
2. Accept θ' with probability

$$\alpha_{\mathrm{MH}}(\theta, \theta') := 1 \wedge \frac{q(\theta', \theta)\pi(\theta')}{q(\theta, \theta')\pi(\theta)} = 1 \wedge \frac{q(\theta', \theta)p(\theta')}{q(\theta, \theta')p(\theta)} \prod_{i=1}^{n} \frac{p(y_i|\theta')}{p(y_i|\theta)}$$

$\Rightarrow O(n)$ computation per step to compute $\alpha_{\mathrm{MH}}(\theta, \theta')$

Our approach

- Want a method with cost $o(n)$ per step – subsampling

Our approach

- Want a method with cost $o(n)$ per step – subsampling
- Want our method not to reduce accuracy – exactness

Our approach

- Several existing exact subsampling methods:
  - Firefly [Maclaurin and Adams, 2014]
  - Delayed acceptance [Banterle et al., 2015]
  - Piecewise-deterministic MCMC [Bouchard-Côté et al., 2018, Bierkens et al., 2018]

Our approach

- Several existing exact subsampling methods:
  - Firefly [Maclaurin and Adams, 2014]
  - Delayed acceptance [Banterle et al., 2015]
  - Piecewise-deterministic MCMC [Bouchard-Côté et al., 2018, Bierkens et al., 2018]
- Our method: an exact subsampling scheme based on a **proxy target** that requires on average $O(1)$ or $O(1/\sqrt{n})$ likelihood evaluations per step

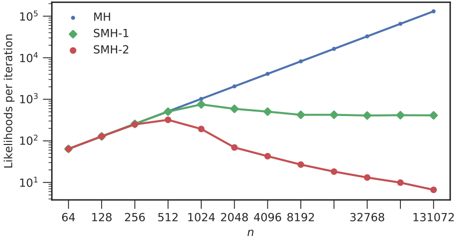

Figure 1: Average number of likelihood evaluations per iteration required by SMH for a 10-dimensional logistic regression posterior as the number of data points n increases.

Three key ingredients

1. A **factorised** MH acceptance probability
2. Procedures for fast simulation of Bernoulli random variables
3. Control performance using an approximate target ("control variates")

Ingredient 1 - Factorised Metropolis–Hastings

- Suppose we can factor the target like

$$\pi(\theta) \propto \prod_{i=1}^{n} \pi_i(\theta)$$

Ingredient 1 - Factorised Metropolis–Hastings

- Suppose we can factor the target like

$$\pi(\theta) \propto \prod_{i=1}^{n} \pi_i(\theta)$$

- Obvious choice (with a flat prior) is $\pi_i(\theta) = p(y_i|\theta)$

Ingredient 1 - Factorised Metropolis–Hastings

- Suppose we can factor the target like

$$\pi(\theta) \propto \prod_{i=1}^{n} \pi_i(\theta)$$

- Obvious choice (with a flat prior) is $\pi_i(\theta) = p(y_i|\theta)$
- Can show that (for a symmetric proposal)

$$\alpha_{\mathrm{FMH}}(\theta, \theta') := \prod_{i=1}^{n} \alpha_{\mathrm{FMH}\,i}(\theta, \theta') := \prod_{i=1}^{n} 1 \wedge \frac{\pi_i(\theta')}{\pi_i(\theta)}$$

is also a **valid acceptance probability** for an MH-style algorithm

Ingredient 1 - Factorised Metropolis–Hastings

- Suppose we can factor the target like

$$\pi(\theta) \propto \prod_{i=1}^{n} \pi_i(\theta)$$

- Obvious choice (with a flat prior) is $\pi_i(\theta) = p(y_i|\theta)$
- Can show that (for a symmetric proposal)

$$\alpha_{\mathrm{FMH}}(\theta, \theta') := \prod_{i=1}^{n} \alpha_{\mathrm{FMH}\,i}(\theta, \theta') := \prod_{i=1}^{n} 1 \wedge \frac{\pi_i(\theta')}{\pi_i(\theta)}$$

  is also a **valid acceptance probability** for an MH-style algorithm
- Compare the MH acceptance probability as

$$\alpha_{\mathrm{MH}}(\theta, \theta') = 1 \wedge \prod_{i=1}^{n} \frac{\pi_i(\theta')}{\pi_i(\theta)}$$

Ingredient 1 - Factorised Metropolis–Hastings

Explicitly, (assuming symmetric q) FMH algorithm is:

Factorised Metropolis-Hastings (FMH)

1. Propose $\theta' \sim q(\theta, \cdot)$
2. Accept θ' with probability

$$\alpha_{\mathrm{FMH}}(\theta, \theta') := \prod_{i=1}^{n} \alpha_{\mathrm{FMH}\,i}(\theta, \theta') := \prod_{i=1}^{n} 1 \wedge \frac{\pi_i(\theta')}{\pi_i(\theta)}$$

Ingredient 1 - Factorised Metropolis–Hastings

Explicitly, (assuming symmetric q) FMH algorithm is:

Factorised Metropolis-Hastings (FMH)

1. Propose $\theta' \sim q(\theta, \cdot)$
2. Accept θ' with probability

$$\alpha_{\mathrm{FMH}}(\theta, \theta') := \prod_{i=1}^{n} \alpha_{\mathrm{FMH}\,i}(\theta, \theta') := \prod_{i=1}^{n} 1 \wedge \frac{\pi_i(\theta')}{\pi_i(\theta)}$$

- Can implement acceptance step by sampling **independent** $B_i \sim \mathrm{Bernoulli}(\alpha_{\mathrm{FMH}\,i}(\theta, \theta'))$ and accepting if every $B_i = 1$

Ingredient 1 - Factorised Metropolis–Hastings

Explicitly, (assuming symmetric q) FMH algorithm is:

Factorised Metropolis-Hastings (FMH)

1. Propose $\theta' \sim q(\theta, \cdot)$
2. Accept θ' with probability

$$\alpha_{\mathrm{FMH}}(\theta, \theta') := \prod_{i=1}^{n} \alpha_{\mathrm{FMH}_i}(\theta, \theta') := \prod_{i=1}^{n} 1 \wedge \frac{\pi_i(\theta')}{\pi_i(\theta)}$$

- Can implement acceptance step by sampling **independent** $B_i \sim \mathrm{Bernoulli}(\alpha_{\mathrm{FMH}_i}(\theta, \theta'))$ and accepting if every $B_i = 1$
- Can stop as soon as some $B_i = 0$: **delayed acceptance**

Ingredient 1 - Factorised Metropolis–Hastings

Explicitly, (assuming symmetric q) FMH algorithm is:

Factorised Metropolis-Hastings (FMH)

1. Propose $\theta' \sim q(\theta, \cdot)$
2. Accept θ' with probability

$$\alpha_{\mathrm{FMH}}(\theta, \theta') := \prod_{i=1}^{n} \alpha_{\mathrm{FMH}\,i}(\theta, \theta') := \prod_{i=1}^{n} 1 \wedge \frac{\pi_i(\theta')}{\pi_i(\theta)}$$

- Can implement acceptance step by sampling **independent** $B_i \sim \mathrm{Bernoulli}(\alpha_{\mathrm{FMH}\,i}(\theta, \theta'))$ and accepting if every $B_i = 1$
- Can stop as soon as some $B_i = 0$: **delayed acceptance**
- However, still must compute all n terms in order to accept

Three key ingredients

1. A factorised MH acceptance probability
2. Procedures for **fast simulation** of Bernoulli random variables
3. Control performance using an approximate target ("control variates")

Ingredient 2 - Fast Bernoulli simulation

- How can we avoid simulating these n Bernoullis?

Ingredient 2 - Fast Bernoulli simulation

- How can we avoid simulating these n Bernoullis?
- Assuming we have bounds

$$\overline{\lambda}_i(\theta, \theta') \geq -\log \alpha_{\mathrm{FMH}\,i}(\theta, \theta') =: \lambda_i(\theta, \theta')$$

we can use the following:

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^n \overline{\lambda}_i(\theta, \theta'))$
2. $X_1, \ldots, X_C \overset{\mathrm{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta, \theta') / \sum_{i=1}^n \overline{\lambda}_i(\theta, \theta')]_{1 \leq i \leq n}\right)$
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta, \theta') / \overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \leq j \leq C$

Ingredient 2 - Fast Bernoulli simulation

- How can we avoid simulating these n Bernoullis?
- Assuming we have bounds

$$\overline{\lambda}_i(\theta, \theta') \geq -\log \alpha_{\mathrm{FMH}\,i}(\theta, \theta') =: \lambda_i(\theta, \theta')$$

we can use the following:

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^n \overline{\lambda}_i(\theta, \theta'))$
2. $X_1, \ldots, X_C \overset{\mathrm{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta, \theta')/\sum_{i=1}^n \overline{\lambda}_i(\theta, \theta')]_{1 \leq i \leq n}\right)$
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta, \theta')/\overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \leq j \leq C$

$\Rightarrow \mathbb{P}(B_1 = \cdots = B_C = 0) = \alpha_{\mathrm{FMH}}(\theta, \theta')$, so can use this procedure to perform the FMH accept/reject step

Ingredient 2 - Fast Bernoulli simulation

- How can we avoid simulating these n Bernoullis?
- Assuming we have bounds

$$\overline{\lambda}_i(\theta, \theta') \geq -\log \alpha_{\mathrm{FMH}\,i}(\theta, \theta') =: \lambda_i(\theta, \theta')$$

we can use the following:

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^n \overline{\lambda}_i(\theta, \theta'))$
2. $X_1, \ldots, X_C \overset{\mathrm{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta, \theta') / \sum_{i=1}^n \overline{\lambda}_i(\theta, \theta')]_{1 \leq i \leq n}\right)$
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta, \theta') / \overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \leq j \leq C$

$\Rightarrow \mathbb{P}(B_1 = \cdots = B_C = 0) = \alpha_{\mathrm{FMH}}(\theta, \theta')$, so can use this procedure to perform the FMH accept/reject step

- Intuition: sample a discrete Poisson point process on $\{1, \ldots, n\}$ with intensity $i \mapsto \lambda_i(\theta, \theta')$ by **thinning** one with intensity $i \mapsto \overline{\lambda}_i(\theta, \theta')$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \text{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$
2. $X_1, \ldots, X_C \overset{\text{iid}}{\sim} \text{Categorical}\left([\overline{\lambda}_i(\theta, \theta') / \sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')]_{1 \le i \le n}\right)$
3. $B_j \sim \text{Bernoulli}(\lambda_{X_j}(\theta, \theta') / \overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \le j \le C$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \text{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$
2. $X_1, \ldots, X_C \overset{\text{iid}}{\sim} \text{Categorical}\left([\overline{\lambda}_i(\theta, \theta') / \sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')]_{1 \le i \le n}\right)$
3. $B_j \sim \text{Bernoulli}(\lambda_{X_j}(\theta, \theta') / \overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \le j \le C$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta')\psi_i \ge -\log \alpha_{\text{FMH}\,i}(\theta, \theta') = \lambda_i(\theta, \theta'). \qquad (*)$$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \text{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$
2. $X_1, \ldots, X_C \overset{\text{iid}}{\sim} \text{Categorical}\left([\overline{\lambda}_i(\theta, \theta')/\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')]_{1 \le i \le n}\right)$
3. $B_j \sim \text{Bernoulli}(\lambda_{X_j}(\theta, \theta')/\overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \le j \le C$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta')\psi_i \ge -\log \alpha_{\text{FMH}i}(\theta, \theta') = \lambda_i(\theta, \theta'). \qquad (*)$$

Then:

$$\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta') \sum_{i=1}^{n} \psi_i$$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')) \Rightarrow O(1)$ (after precomputing $\sum_{i=1}^n \psi_i$)
2. $X_1, \ldots, X_C \overset{\mathrm{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta,\theta') / \sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')]_{1 \le i \le n}\right)$
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta,\theta')/\overline{\lambda}_{X_j}(\theta,\theta'))$ for $1 \le j \le C$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta,\theta') = \varphi(\theta,\theta')\psi_i \ge -\log \alpha_{\mathrm{FMH}\,i}(\theta,\theta') = \lambda_i(\theta,\theta'). \qquad (*)$$

Then:

$$\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta') = \varphi(\theta,\theta') \sum_{i=1}^n \psi_i$$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$ $\Rightarrow$ $O(1)$ (after precomputing $\sum_{i=1}^{n} \psi_i$)
2. $X_1, \ldots, X_C \overset{\text{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta, \theta') / \sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')]_{1 \le i \le n}\right)$
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta, \theta') / \overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \le j \le C$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta')\psi_i \ge -\log \alpha_{\mathrm{FMH}\,i}(\theta, \theta') = \lambda_i(\theta, \theta'). \qquad (*)$$

Then:

$$\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta') \sum_{i=1}^{n} \psi_i \quad \text{and} \quad \frac{\overline{\lambda}_i(\theta, \theta')}{\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')} = \frac{\psi_i}{\sum_{i=1}^{n} \psi_i}.$$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')) \Rightarrow O(1)$ (after precomputing $\sum_{i=1}^n \psi_i$)
2. $X_1,\ldots,X_C \overset{\mathrm{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta,\theta')/\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')]_{1\le i\le n}\right) \Rightarrow O(C)$ (via Walker's alias method [Walker, 1977], after $\Theta(n)$ setup cost)
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta,\theta')/\overline{\lambda}_{X_j}(\theta,\theta'))$ for $1 \le j \le C$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta,\theta') = \varphi(\theta,\theta')\psi_i \ge -\log \alpha_{\mathrm{FMH}\,i}(\theta,\theta') = \lambda_i(\theta,\theta'). \qquad (*)$$

Then:

$$\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta') = \varphi(\theta,\theta')\sum_{i=1}^n \psi_i \qquad \text{and} \qquad \frac{\overline{\lambda}_i(\theta,\theta')}{\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')} = \frac{\psi_i}{\sum_{i=1}^n \psi_i}.$$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \text{Poisson}(\sum_{i=1}^n \overline{\lambda}_i(\theta, \theta')) \Rightarrow O(1)$ (after precomputing $\sum_{i=1}^n \psi_i$)
2. $X_1, \ldots, X_C \overset{\text{iid}}{\sim} \text{Categorical}\left([\overline{\lambda}_i(\theta, \theta') / \sum_{i=1}^n \overline{\lambda}_i(\theta, \theta')]_{1 \leq i \leq n}\right) \Rightarrow O(C)$ (via Walker's alias method [Walker, 1977], after $\Theta(n)$ setup cost)
3. $B_j \sim \text{Bernoulli}(\lambda_{X_j}(\theta, \theta') / \overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \leq j \leq C \Rightarrow O(C)$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta')\psi_i \geq -\log \alpha_{\text{FMH}\,i}(\theta, \theta') = \lambda_i(\theta, \theta'). \qquad (*)$$

Then:

$$\sum_{i=1}^n \overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta') \sum_{i=1}^n \psi_i \qquad \text{and} \qquad \frac{\overline{\lambda}_i(\theta, \theta')}{\sum_{i=1}^n \overline{\lambda}_i(\theta, \theta')} = \frac{\psi_i}{\sum_{i=1}^n \psi_i}.$$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')) \Rightarrow O(1)$ (after precomputing $\sum_{i=1}^{n} \psi_i$)
2. $X_1, \ldots, X_C \overset{\text{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta, \theta') / \sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')]_{1 \leq i \leq n}\right) \Rightarrow O(C)$ (via Walker's alias method [Walker, 1977], after $\Theta(n)$ setup cost)
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta, \theta') / \overline{\lambda}_{X_j}(\theta, \theta'))$ for $1 \leq j \leq C \Rightarrow O(C)$

$\Rightarrow$ Overall cost of $O(C)$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta')\psi_i \geq -\log \alpha_{\mathrm{FMH}\,i}(\theta, \theta') = \lambda_i(\theta, \theta'). \qquad (*)$$

Then:

$$\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta') = \varphi(\theta, \theta') \sum_{i=1}^{n} \psi_i \qquad \text{and} \qquad \frac{\overline{\lambda}_i(\theta, \theta')}{\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')} = \frac{\psi_i}{\sum_{i=1}^{n} \psi_i}.$$

Ingredient 2 - Fast Bernoulli simulation

Poisson subsampling

1. $C \sim \mathrm{Poisson}(\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')) \Rightarrow O(1)$ (after precomputing $\sum_{i=1}^n \psi_i$)
2. $X_1,\ldots,X_C \overset{\mathrm{iid}}{\sim} \mathrm{Categorical}\left([\overline{\lambda}_i(\theta,\theta')/\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')]_{1\le i\le n}\right) \Rightarrow O(C)$ (via Walker's alias method [Walker, 1977], after $\Theta(n)$ setup cost)
3. $B_j \sim \mathrm{Bernoulli}(\lambda_{X_j}(\theta,\theta')/\overline{\lambda}_{X_j}(\theta,\theta'))$ for $1 \le j \le C \Rightarrow O(C)$

$\Rightarrow$ Overall cost of $O(C)$

When is this **efficient**? Suppose our bounds have the form:

$$\overline{\lambda}_i(\theta,\theta') = \varphi(\theta,\theta')\psi_i \ge -\log \alpha_{\mathrm{FMH}\,i}(\theta,\theta') = \lambda_i(\theta,\theta'). \qquad (*)$$

Then:

$$\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta') = \varphi(\theta,\theta')\sum_{i=1}^n \psi_i \qquad \text{and} \qquad \frac{\overline{\lambda}_i(\theta,\theta')}{\sum_{i=1}^n \overline{\lambda}_i(\theta,\theta')} = \frac{\psi_i}{\sum_{i=1}^n \psi_i}.$$

(*) holds for instance if $\log \pi_i$ is Lipschitz (but will see better case later).

Potential problems

Two problems now to overcome:

Potential problems

Two problems now to overcome:

1. Since $C \sim \mathrm{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$, potentially $C > n$
   $\Rightarrow$ **Must ensure** $C = o(n)$ if we are to achieve anything

Potential problems

Two problems now to overcome:

1. Since $C \sim \mathrm{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$, potentially $C > n$
   $\Rightarrow$ **Must ensure** $C = o(n)$ if we are to achieve anything
2. Since each $\alpha_{\mathrm{FMH}\,i}(\theta, \theta') \leq 1$, can have $\alpha_{\mathrm{FMH}}(\theta, \theta') \to 0$ as $n \to \infty$
   $\Rightarrow$ **Must ensure** $\alpha_{\mathrm{FMH}}(\theta, \theta')$ is well behaved

Potential problems

Two problems now to overcome:

1. Since $C \sim \mathrm{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$, potentially $C > n$
   $\Rightarrow$ **Must ensure** $C = o(n)$ if we are to achieve anything
2. Since each $\alpha_{\mathrm{FMH}\,i}(\theta, \theta') \leq 1$, can have $\alpha_{\mathrm{FMH}}(\theta, \theta') \to 0$ as $n \to \infty$
   $\Rightarrow$ **Must ensure** $\alpha_{\mathrm{FMH}}(\theta, \theta')$ is well behaved

These problems are are **related** since

$$\mathbb{E}[C|\theta, \theta'] = \sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta') \quad \text{and} \quad \alpha_{\mathrm{FMH}}(\theta, \theta') \geq \exp(-\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')).$$

Potential problems

Two problems now to overcome:

1. Since $C \sim \text{Poisson}(\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta'))$, potentially $C > n$
   $\Rightarrow$ **Must ensure** $C = o(n)$ if we are to achieve anything
2. Since each $\alpha_{\text{FMH}i}(\theta, \theta') \leq 1$, can have $\alpha_{\text{FMH}}(\theta, \theta') \to 0$ as $n \to \infty$
   $\Rightarrow$ **Must ensure** $\alpha_{\text{FMH}}(\theta, \theta')$ is well behaved

These problems are are **related** since

$$\mathbb{E}[C|\theta, \theta'] = \sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta') \quad \text{and} \quad \alpha_{\text{FMH}}(\theta, \theta') \geq \exp(-\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')).$$

Key question is how to choose bounds for which $\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')$ is small.

Three key ingredients

1. A factorised MH acceptance probability
2. Procedures for fast simulation of Bernoulli random variables
3. Control performance using an **approximate target** ("control variates")

Ingredient 3 - control variates

- Write the target as

$$\pi(\theta) = \prod_{i=1}^{n} \pi_i(\theta) = \prod_{i=1}^{n} \exp(-U_i(\theta))$$

for **potentials** $U_i = -\log \pi_i(\theta)$

Ingredient 3 - control variates

- Write the target as

$$\pi(\theta) = \prod_{i=1}^{n} \pi_i(\theta) = \prod_{i=1}^{n} \exp(-U_i(\theta))$$

for **potentials** $U_i = -\log \pi_i(\theta)$

- Approximate

$$\widehat{U}_{k,i}(\theta) \approx U_i(\theta)$$

where $\widehat{U}_{k,i}$ is a k**-th order Taylor expansion** of U_i around some fixed $\widehat{\theta}$ (not depending on i)

Ingredient 3 - control variates

- Also let

$$\widehat{U}_k(\theta) := \sum_{i=1}^{n} \widehat{U}_{k,i}(\theta)$$

Ingredient 3 - control variates

- Also let

$$\widehat{U}_k(\theta) := \sum_{i=1}^{n} \widehat{U}_{k,i}(\theta) \approx U(\theta) := \sum_{i=1}^{n} U_i(\theta) = -\log \pi(\theta)$$

which is itself a Taylor expansion of $U(\theta)$ around $\widehat{\theta}$

Ingredient 3 - control variates

- Also let

$$\widehat{U}_k(\theta) := \sum_{i=1}^{n} \widehat{U}_{k,i}(\theta) \approx U(\theta) := \sum_{i=1}^{n} U_i(\theta) = -\log \pi(\theta)$$

  which is itself a Taylor expansion of $U(\theta)$ around $\widehat{\theta}$

- Explicitly

$$\begin{aligned}\widehat{U}_1(\theta) &= U(\widehat{\theta}) + \nabla U(\widehat{\theta})^\top (\theta - \widehat{\theta}) \\ \widehat{U}_2(\theta) &= U(\widehat{\theta}) + \nabla U(\widehat{\theta})^\top (\theta - \widehat{\theta}) + \frac{1}{2}(\theta - \widehat{\theta})^\top \nabla^2 U(\widehat{\theta})(\theta - \widehat{\theta})\end{aligned}$$

Ingredient 3 - control variates

- Also let

$$\widehat{U}_k(\theta) := \sum_{i=1}^{n} \widehat{U}_{k,i}(\theta) \approx U(\theta) := \sum_{i=1}^{n} U_i(\theta) = -\log \pi(\theta)$$

which is itself a Taylor expansion of $U(\theta)$ around $\widehat{\theta}$

- Explicitly

$$\begin{aligned}\widehat{U}_1(\theta) &= U(\widehat{\theta}) + \nabla U(\widehat{\theta})^\top(\theta - \widehat{\theta}) \\ \widehat{U}_2(\theta) &= U(\widehat{\theta}) + \nabla U(\widehat{\theta})^\top(\theta - \widehat{\theta}) + \frac{1}{2}(\theta - \widehat{\theta})^\top \nabla^2 U(\widehat{\theta})(\theta - \widehat{\theta})\end{aligned}$$

- In particular, $\exp(-\widehat{U}_2(\theta)) \approx \pi(\theta)$ is a **Gaussian approximation** to the target at $\widehat{\theta}$

Ingredient 3 - control variates

Define the Scalable Metropolis-Hastings (SMH) acceptance probability

$$\alpha_{\text{SMH-}k}(\theta, \theta') := \left(1 \wedge \frac{\exp(-\widehat{U}_k(\theta'))}{\exp(-\widehat{U}_k(\theta))}\right) \prod_{i=1}^{n} 1 \wedge \frac{\exp(\widehat{U}_{k,i}(\theta') - U_i(\theta'))}{\exp(\widehat{U}_{k,i}(\theta) - U_i(\theta))}.$$

Ingredient 3 - control variates

Define the Scalable Metropolis-Hastings (SMH) acceptance probability

$$\alpha_{\text{SMH-}k}(\theta, \theta') := \left(1 \wedge \frac{\exp(-\widehat{U}_k(\theta'))}{\exp(-\widehat{U}_k(\theta))}\right) \prod_{i=1}^{n} 1 \wedge \frac{\exp(\widehat{U}_{k,i}(\theta') - U_i(\theta'))}{\exp(\widehat{U}_{k,i}(\theta) - U_i(\theta))}.$$

- Corresponds to FMH using the factorisations

$$\pi = \underbrace{\exp(-\widehat{U}_k)}_{\pi_{n+1}} \prod_{i=1}^{n} \underbrace{\exp(\widehat{U}_{k,i} - U_i)}_{\pi_i}$$

Ingredient 3 - control variates

Define the Scalable Metropolis-Hastings (SMH) acceptance probability

$$\alpha_{\text{SMH-}k}(\theta,\theta') := \left(1 \wedge \frac{\exp(-\widehat{U}_k(\theta'))}{\exp(-\widehat{U}_k(\theta))}\right) \prod_{i=1}^{n} 1 \wedge \frac{\exp(\widehat{U}_{k,i}(\theta') - U_i(\theta'))}{\exp(\widehat{U}_{k,i}(\theta) - U_i(\theta))}.$$

- Corresponds to FMH using the factorisations

$$\pi = \underbrace{\exp(-\widehat{U}_k)}_{\pi_{n+1}} \prod_{i=1}^{n} \underbrace{\exp(\widehat{U}_{k,i} - U_i)}_{\pi_i}$$

- First factor can be simulated directly in $O(1)$ time

Ingredient 3 - control variates

Define the Scalable Metropolis-Hastings (SMH) acceptance probability

$$\alpha_{\mathrm{SMH}\text{-}k}(\theta,\theta') := \left(1 \wedge \frac{\exp(-\widehat{U}_k(\theta'))}{\exp(-\widehat{U}_k(\theta))}\right) \prod_{i=1}^{n} 1 \wedge \frac{\exp(\widehat{U}_{k,i}(\theta') - U_i(\theta'))}{\exp(\widehat{U}_{k,i}(\theta) - U_i(\theta))}.$$

- Corresponds to FMH using the factorisations

$$\pi = \underbrace{\exp(-\widehat{U}_k)}_{\pi_{n+1}} \prod_{i=1}^{n} \underbrace{\exp(\widehat{U}_{k,i} - U_i)}_{\pi_i}$$

- First factor can be simulated directly in $O(1)$ time
- Remaining factors can be simulated with Poisson subsampling

Ingredient 3 - control variates

- Recall we need upper bounds

$$-\log \alpha_{\mathrm{FMH}\,i}(\theta, \theta') \leq \varphi(\theta, \theta')\psi_i =: \overline{\lambda}_i(\theta, \theta')$$

Ingredient 3 - control variates

- Recall we need upper bounds

$$-\log \alpha_{\mathrm{FMH}i}(\theta, \theta') \leq \varphi(\theta, \theta')\psi_i =: \overline{\lambda}_i(\theta, \theta')$$

- Possible to show that, if we can find constants

$$\overline{U}_{k+1,i} \geq \sup_{\substack{\theta \in \Theta \\ |\beta| = k+1}} |\partial^\beta U_i(\theta)| \qquad (*)$$

then we can use

$$\overline{\lambda}_i(\theta, \theta') := \underbrace{(\|\theta - \widehat{\theta}\|_1^{k+1} + \|\theta' - \widehat{\theta}\|_1^{k+1})}_{\varphi(\theta, \theta')} \underbrace{\frac{\overline{U}_{k+1,i}}{(k+1)!}}_{\psi_i}$$

Ingredient 3 - control variates

- Recall we need upper bounds

$$-\log \alpha_{\mathrm{FMH}\,i}(\theta, \theta') \le \varphi(\theta, \theta')\psi_i =: \overline{\lambda}_i(\theta, \theta')$$

- Possible to show that, if we can find constants

$$\overline{U}_{k+1,i} \ge \sup_{\substack{\theta \in \Theta \\ |\beta| = k+1}} |\partial^\beta U_i(\theta)| \tag{*}$$

then we can use

$$\overline{\lambda}_i(\theta, \theta') := \underbrace{\left(\|\theta - \widehat{\theta}\|_1^{k+1} + \|\theta' - \widehat{\theta}\|_1^{k+1}\right)}_{\varphi(\theta, \theta')} \underbrace{\frac{\overline{U}_{k+1,i}}{(k+1)!}}_{\psi_i}$$

- (*) constitutes the **only quantity** that must be specified by hand to use our method on a given model

Ingredient 3 - control variates

Heuristically, suppose

Ingredient 3 - control variates

Heuristically, suppose

- $\theta \sim \pi$ (chain is at stationarity)

Ingredient 3 - control variates

Heuristically, suppose

- $\theta \sim \pi$ (chain is at stationarity)
- $\|\theta - \theta_{\text{MAP}}\| = O(1/\sqrt{n})$ ($1/\sqrt{n}$ concentration - key assumption)

Ingredient 3 - control variates

Heuristically, suppose

- $\theta \sim \pi$ (chain is at stationarity)
- $\|\theta - \theta_{\mathrm{MAP}}\| = O(1/\sqrt{n})$ ($1/\sqrt{n}$ concentration - key assumption)
- $\|\theta' - \theta\| = O(1/\sqrt{n})$ (proposal is scaled like $1/\sqrt{n}$)

Ingredient 3 - control variates

Heuristically, suppose

- $\theta \sim \pi$ (chain is at stationarity)
- $\|\theta - \theta_{\mathrm{MAP}}\| = O(1/\sqrt{n})$ ($1/\sqrt{n}$ concentration - key assumption)
- $\|\theta' - \theta\| = O(1/\sqrt{n})$ (proposal is scaled like $1/\sqrt{n}$)
- $\|\widehat{\theta} - \theta_{\mathrm{MAP}}\| = O(1/\sqrt{n})$ ($\widehat{\theta}$ is not too far from mode)

Ingredient 3 - control variates

Heuristically, suppose

- $\theta \sim \pi$ (chain is at stationarity)
- $\|\theta - \theta_{\mathrm{MAP}}\| = O(1/\sqrt{n})$ ($1/\sqrt{n}$ concentration - key assumption)
- $\|\theta' - \theta\| = O(1/\sqrt{n})$ (proposal is scaled like $1/\sqrt{n}$)
- $\|\widehat{\theta} - \theta_{\mathrm{MAP}}\| = O(1/\sqrt{n})$ ($\widehat{\theta}$ is not too far from mode)

then by the triangle inequality

$$\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta') = \underbrace{(\|\theta - \widehat{\theta}\|_1^{k+1} + \|\theta' - \widehat{\theta}\|_1^{k+1})}_{O(n^{-(k+1)/2})} \underbrace{\sum_{i=1}^{n} \frac{\overline{U}_{k+1,i}}{(k+1)!}}_{O(n)} = O(n^{(1-k)/2})$$

Ingredient 3 - control variates

Heuristically, suppose

- $\theta \sim \pi$ (chain is at stationarity)
- $\|\theta - \theta_{\mathrm{MAP}}\| = O(1/\sqrt{n})$ ($1/\sqrt{n}$ concentration - key assumption)
- $\|\theta' - \theta\| = O(1/\sqrt{n})$ (proposal is scaled like $1/\sqrt{n}$)
- $\|\widehat{\theta} - \theta_{\mathrm{MAP}}\| = O(1/\sqrt{n})$ ($\widehat{\theta}$ is not too far from mode)

then by the triangle inequality

$$\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta') = \underbrace{\left(\|\theta - \widehat{\theta}\|_1^{k+1} + \|\theta' - \widehat{\theta}\|_1^{k+1}\right)}_{O(n^{-(k+1)/2})} \underbrace{\sum_{i=1}^{n} \frac{\overline{U}_{k+1,i}}{(k+1)!}}_{O(n)} = O(n^{(1-k)/2})$$

In particular, $\sum_{i=1}^{n} \overline{\lambda}_i(\theta, \theta')$ is $O(1)$ if $k = 1$ and $O(1/\sqrt{n})$ if $k = 2$

Summary

This directly yields an **average cost** per step

$$\mathbb{E}[C|\theta,\theta'] = \sum_{i=1}^{n} \overline{\lambda}_i(\theta,\theta') = \begin{cases} O(1), & k=1 \\ O(1/\sqrt{n}) & k=2. \end{cases}$$

Summary

This directly yields an **average cost** per step

$$\mathbb{E}[C|\theta,\theta'] = \sum_{i=1}^{n} \overline{\lambda}_i(\theta,\theta') = \begin{cases} O(1), & k=1 \\ O(1/\sqrt{n}) & k=2. \end{cases}$$

Likewise, acceptance probability is **stable** since

$$\alpha_{\text{SMH-}k}(\theta,\theta') := \underbrace{\left(1 \wedge \frac{\exp(-\widehat{U}_k(\theta'))}{\exp(-\widehat{U}_k(\theta))}\right)}_{\substack{\geq \exp(-O(1)) \\ \text{(can show)}}} \underbrace{\prod_{i=1}^{n} 1 \wedge \frac{\exp(\widehat{U}_{k,i}(\theta') - U_i(\theta'))}{\exp(\widehat{U}_{k,i}(\theta) - U_i(\theta))}}_{\geq \exp(-\sum_{i=1}^{n} \overline{\lambda}_i(\theta,\theta'))}.$$

Summary

This directly yields an **average cost** per step

$$\mathbb{E}[C|\theta,\theta'] = \sum_{i=1}^{n} \overline{\lambda}_i(\theta,\theta') = \begin{cases} O(1), & k=1 \\ O(1/\sqrt{n}) & k=2. \end{cases}$$

Likewise, acceptance probability is **stable** since

$$\alpha_{\text{SMH-}k}(\theta,\theta') := \underbrace{\left(1 \wedge \frac{\exp(-\widehat{U}_k(\theta'))}{\exp(-\widehat{U}_k(\theta))}\right)}_{\substack{\geq \exp(-O(1)) \\ \text{(can show)}}} \underbrace{\prod_{i=1}^{n} 1 \wedge \frac{\exp(\widehat{U}_{k,i}(\theta') - U_i(\theta'))}{\exp(\widehat{U}_{k,i}(\theta) - U_i(\theta))}}_{\geq \exp(-\sum_{i=1}^{n} \overline{\lambda}_i(\theta,\theta'))}.$$

Can do even better with a $\exp(-\widehat{U}_k)$-**reversible proposal** (first term vanishes).

Application - logistic regression

- We consider logistic regression with covariates $x_i \in \mathbb{R}^d$ and responses $y_i \in \{0, 1\}$

$$
\begin{aligned}
p(y_i|\theta, x_i) &= \mathrm{Bernoulli}(y_i|\frac{1}{1+\exp(-\theta^\top x_i)}) \\
\Rightarrow U_i(\theta) &= -\log p(y_i|\theta, x_i) = \log(1+\exp(\theta^T x_i)) - y_i\theta^\top x_i
\end{aligned}
$$

Application - logistic regression

- We consider logistic regression with covariates $x_i \in \mathbb{R}^d$ and responses $y_i \in \{0, 1\}$

$$
\begin{aligned}
p(y_i|\theta, x_i) &= \mathrm{Bernoulli}(y_i|\frac{1}{1+\exp(-\theta^\top x_i)}) \\
\Rightarrow U_i(\theta) &= -\log p(y_i|\theta, x_i) = \log(1+\exp(\theta^T x_i)) - y_i\theta^\top x_i
\end{aligned}
$$

- Admits upper bounds

$$
\overline{U}_{2,i} = \frac{1}{4}\max_{1\leq j\leq d}|x_{ij}|^2 \qquad \overline{U}_{3,i} = \frac{1}{6\sqrt{3}}\max_{1\leq j\leq d}|x_{ij}|^3
$$

Application - logistic regression

Empirical result for $d = 10$

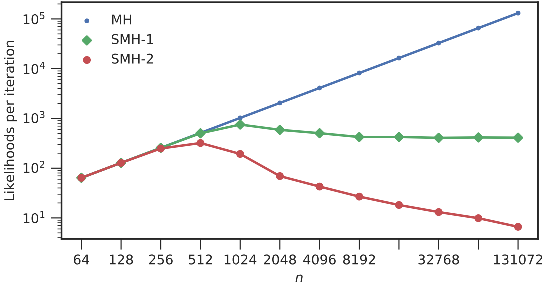

Figure 2: Average number of likelihood evaluations per iteration required by SMH for a 10-dimensional logistic regression posterior as the number of data points n increases.

Application - logistic regression

Empirical result for $d = 10$

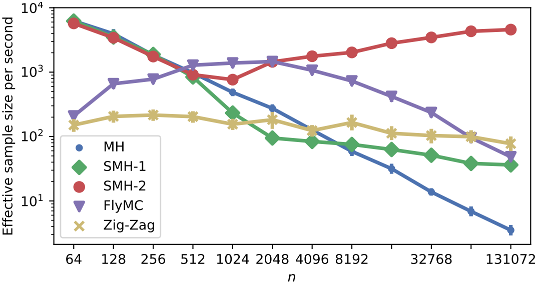

Figure 3: Effective sample size per second of computation for posterior mean of first regression coefficient (higher is better)

Thanks for listening

Find us later at poster #202.