Bayesian Learning

Lecture 9 - Hamiltonian Monte Carlo and Variational Inference

Mattias Villani

Department of Statistics
Stockholm University

mattiasvillani.com @matvil @matvil mattiasvillani

Lecture overview

- **Hamiltonian Monte Carlo**
- **Variational Inference**

Hamiltonian Monte Carlo

- When $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^\top$ is **high-dimensional**, $p(\boldsymbol{\theta}|\mathbf{y})$ usually located in some subregion of $\mathbb{R}^p$ with complicated geometry.
- MH: hard to find good proposal distribution $q\left(\cdot|\boldsymbol{\theta}^{(i-1)}\right)$.
- MH: use very small step sizes otherwise too many rejections.
- **Hamiltonian Monte Carlo** (**HMC**):
  - distant proposals **and**
  - high acceptance probabilities.
- Add **momentum** parameters $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_p)^\top$. New target:

$$p(\boldsymbol{\theta}, \boldsymbol{\phi}|\mathbf{y}) = p(\boldsymbol{\theta}|\mathbf{y})\, p(\boldsymbol{\phi})$$

Hamiltonian Monte Carlo

- Physics: **Hamiltonian** system $H(\boldsymbol{\theta}, \boldsymbol{\phi}) = U(\boldsymbol{\theta}) + K(\boldsymbol{\phi})$, where U is the **potential energy** and K is the **kinetic energy**.
- **Hamiltonian Dynamics**

$$\frac{d\theta_i}{dt} = \frac{\partial H}{\partial \phi_i} = \frac{\partial K}{\partial \phi_i},$$
$$\frac{d\phi_i}{dt} = -\frac{\partial H}{\partial \theta_i} = -\frac{\partial U}{\partial \theta_i}$$

- Hockey puck sliding over a friction-less surface: illustration.
- **Posterior sampling**: $U(\boldsymbol{\theta}) = -\log\left[p(\boldsymbol{\theta})\, p(\mathbf{y}|\boldsymbol{\theta})\right]$.
- Momentum: $\boldsymbol{\phi} \sim N(\mathbf{0}, \mathbf{M})$ where $\mathbf{M}$ is the mass matrix and

$$K(\boldsymbol{\phi}) = -\log\left[p(\boldsymbol{\phi})\right] = \frac{1}{2}\boldsymbol{\phi}^\top \mathbf{M}^{-1} \boldsymbol{\phi} + \text{const}$$

- If we could propose $\boldsymbol{\theta}$ in continuous time (spoiler: we can't), the acceptance probability would be one.

Hamiltonian Monte Carlo

- **Hamiltonian Dynamics**

$$\frac{d\theta_i}{dt} = \left[\mathbf{M}^{-1}\boldsymbol{\phi}\right]_i,$$
$$\frac{d\phi_i}{dt} = \frac{\partial \log p\left(\boldsymbol{\theta}|\mathbf{y}\right)}{\partial\theta_i}$$

approximated using L steps with the **leapfrog algorithm**

$$\phi_i\left(t+\frac{\varepsilon}{2}\right) = \phi_i\left(t\right) + \frac{\varepsilon}{2}\frac{\partial \log p\left(\boldsymbol{\theta}|\mathbf{y}\right)}{\partial\theta_i}|_{\theta(t)}$$
$$\theta_i\left(t+\varepsilon\right) = \theta_i\left(t\right) + \varepsilon\mathbf{M}^{-1}\phi_i\left(t+\frac{\varepsilon}{2}\right),$$
$$\phi_i\left(t+\varepsilon\right) = \phi_i\left(t+\frac{\varepsilon}{2}\right) + \frac{\varepsilon}{2}\frac{\partial \log p\left(\boldsymbol{\theta}|\mathbf{y}\right)}{\partial\theta_i}|_{\theta(t+\varepsilon)},$$

where ε is the **step size**.

- **Discretization** $\Rightarrow$ acceptance probability drops with ε.

The Hamiltonian Monte Carlo algorithm

- Initialize $\theta^{(0)}$ and iterate for $i = 1, 2, \ldots$

  1. Sample the starting **momentum** $\phi_s \sim N(0, \mathbf{M})$
  2. Simulate new values for (θ_p, ϕ_p) by iterating the **leapfrog algorithm** L times, starting in $\left(\theta^{(i-1)}, \phi_s\right)$.
  3. Compute the **acceptance probability**

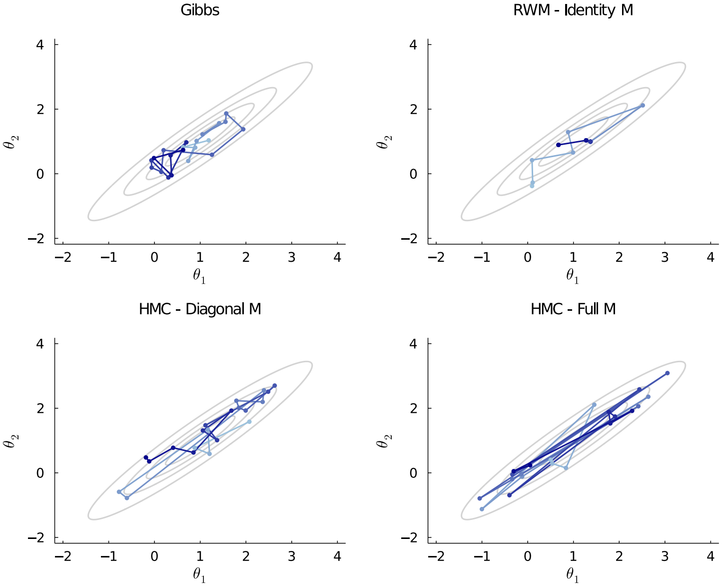
  $$\alpha = \min\left(1, \frac{p(\mathbf{y}|\theta_p)p(\theta_p)}{p(\mathbf{y}|\theta^{(i-1)})p(\theta^{(i-1)})} \frac{p(\phi_p)}{p(\phi_s)}\right)$$

  4. With probability α set $\theta^{(i)} = \theta_p$ and $\theta^{(i)} = \theta^{(i-1)}$ otherwise.

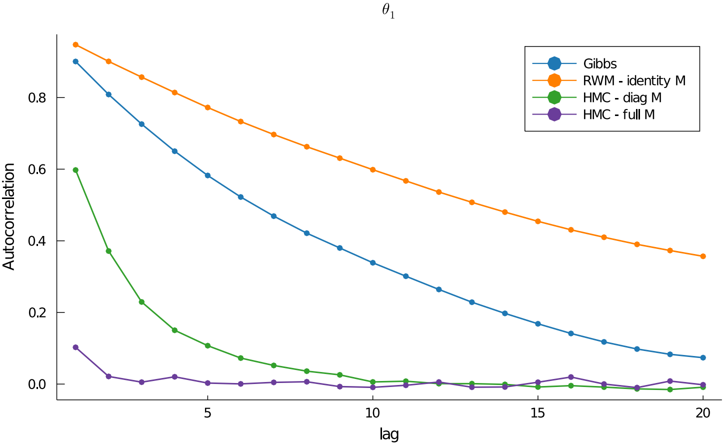
Tuning Hamiltonian Monte Carlo

- HMC is very efficient, but **needs careful tuning** to work.
- **Tuning parameters**:
  - **stepsize** ε,
  - **number of leapfrog** iterations L and
  - **mass matrix** M. (hello $J_{\mathbf{x}}^{-1}(\hat{\theta})$, our old friend)
- **No U-turn** sampler:
  - **Warm-up** to determine ε and L to get good acceptance rate.
  - Avoids U-turns in the Hamiltonian proposals.
- Drawbacks of HMC:
  - Need to **evaluate gradient of log posterior** many times during Hamiltonian iterations. Costly! (Subsampling HMC).
  - Difficulty with **multimodality** (true for most algorithms).
  - Standard HMC cannot handle **discrete parameters**. Mixture example. Some recent progress.

Comparing algorithms for bivariate normal



Comparing algorithms for bivariate normal



Variational Inference

- Approximate posterior $p(\boldsymbol{\theta}|\boldsymbol{y})$ with (simpler) distribution $q(\theta)$.
- Before: **Normal approximation** from optimization:

$$q(\boldsymbol{\theta}) = N\left[\tilde{\boldsymbol{\theta}}, J_{\mathbf{x}}^{-1}(\tilde{\boldsymbol{\theta}})\right]$$

- **Mean field Variational Inference (VI)**:

$$q(\boldsymbol{\theta}) = \prod_{i=1}^{p} q_i(\theta_i)$$

- **Parametric VI**: Parametric family $q_\lambda(\boldsymbol{\theta})$ with parameters λ. Example: $q(\boldsymbol{\theta}) = N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. $\boldsymbol{\lambda} = (\boldsymbol{\mu}, \mathrm{Chol}(\Sigma))$.
- Find $q(\boldsymbol{\theta})$ that **minimizes the Kullback-Leibler divergence** between the true posterior p and the approximation q:

$$KL(q,p) = \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\boldsymbol{y})} d\boldsymbol{\theta} = E_q\left[\log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\boldsymbol{y})}\right].$$

Mean field approximation

- **Mean field VI** is based on factorized approximation:

$$q(\boldsymbol{\theta}) = \prod_{j=1}^{p} q_j(\theta_j)$$

- **No specific functional forms** are assumed for the $q_j(\theta_j)$.

- **Optimal densities** can be shown to satisfy:

$$q_j(\theta_j) \propto \exp\left(E_{-\theta_j} \ln p(\mathbf{y}, \boldsymbol{\theta})\right)$$

  where $E_{-\theta_j}(\cdot)$ is the expectation with respect to $\prod_{k \neq j} q_k(\theta_k)$.

- **Structured mean field approximation**. Group subset of parameters in tractable blocks. Similar to Gibbs sampling.

Mean field approximation - algorithm

- Initialize: $q_2^*(\theta_2), ..., q_M^*(\theta_p)$
- Repeat until convergence:
  - $q_1^*(\theta_1) \leftarrow \frac{\exp\left[E_{-\theta_1} \ln p(\mathbf{y},\theta)\right]}{\int \exp\left[E_{-\theta_1} \ln p(\mathbf{y},\theta)\right] d\theta_1}$
  - $\vdots$
  - $q_p^*(\theta_p) \leftarrow \frac{\exp\left[E_{-\theta_p} \ln p(\mathbf{y},\theta)\right]}{\int \exp\left[E_{-\theta_p} \ln p(\mathbf{y},\theta)\right] d\theta_p}$
- Note: no assumptions about parametric form of the $q_i(\theta)$.
- Optimal $q_i(\theta)$ often **turn out** to be parametric (normal etc).
- Just update hyperparameters in the optimal densities.

Mean field approximation - Normal model

- **Model**: $X_i|\theta, \sigma^2 \overset{\text{iid}}{\sim} N(\theta, \sigma^2)$.
- **Prior**: $\theta \sim N(\mu_0, \tau_0^2)$ **independent** of $\sigma^2 \sim \text{Inv}-\chi^2(\nu_0, \sigma_0^2)$.
- **Mean-field approximation**: $q(\theta, \sigma^2) = q_\theta(\theta) \cdot q_{\sigma^2}(\sigma^2)$.
- Optimal densities

$$
\begin{aligned}
q_\theta^*(\theta) &\propto \exp\left[E_{q(\sigma^2)} \ln p(\theta, \sigma^2, \mathbf{x})\right] \\
q_{\sigma^2}^*(\sigma^2) &\propto \exp\left[E_{q(\theta)} \ln p(\theta, \sigma^2, \mathbf{x})\right]
\end{aligned}
$$

Normal model - VB algorithm

- **Variational density for σ^2**

$$\sigma^2 \sim Inv - \chi^2\left(\tilde{\nu}_n, \tilde{\sigma}_n^2\right)$$

where $\tilde{\nu}_n = \nu_0 + n$ and $\tilde{\sigma}_n^2 = \frac{\nu_0\sigma_0^2 + \sum_{i=1}^{n}(x_i - \tilde{\mu}_n)^2 + n\cdot\tilde{\tau}_n^2}{\nu_0 + n}$

- **Variational density for θ**

$$\theta \sim N\left(\tilde{\mu}_n, \tilde{\tau}_n^2\right)$$

where

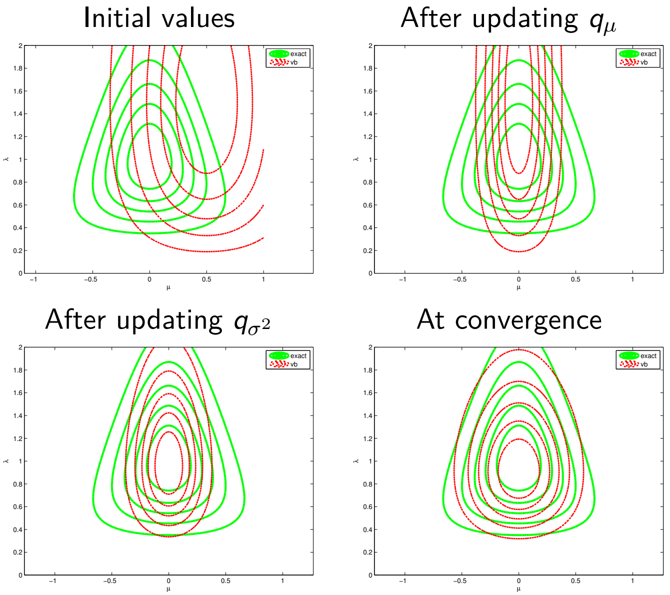
$$\tilde{\tau}_n^2 = \frac{1}{\frac{n}{\tilde{\sigma}_n^2} + \frac{1}{\tau_0^2}}$$

$$\tilde{\mu}_n = \tilde{w}\bar{x} + (1 - \tilde{w})\mu_0,$$

where

$$\tilde{w} = \frac{\frac{n}{\tilde{\sigma}_n^2}}{\frac{n}{\tilde{\sigma}_n^2} + \frac{1}{\tau_0^2}}$$

Normal example from Murphy ($\lambda = 1/\sigma^2$)



Probit regression

- **Model**:
$$\Pr(y_i = 1|\mathbf{x}_i) = \Phi(\mathbf{x}_i^T \boldsymbol{\beta})$$
- **Prior**: $\boldsymbol{\beta} \sim N(0, \Sigma_\beta)$. For example: $\Sigma_\beta = \tau^2 I$.
- **Latent variable formulation** with $u = (u_1, ..., u_n)'$
$$\mathbf{u}|\boldsymbol{\beta} \sim N(\mathbf{X}\boldsymbol{\beta}, 1)$$
and
$$y_i = \begin{cases} 0 & \text{if } u_i \leq 0 \\ 1 & \text{if } u_i > 0 \end{cases}$$
- Factorized **variational approximation**
$$q(\mathbf{u}, \boldsymbol{\beta}) = q_{\mathbf{u}}(\mathbf{u}) q_\beta(\boldsymbol{\beta})$$

VI for probit regression

- **VI posterior**

$$\boldsymbol{\beta} \sim N\left(\tilde{\mu}_\beta, \left(\mathbf{X}^\top \mathbf{X} + \Sigma_\beta^{-1}\right)^{-1}\right)$$

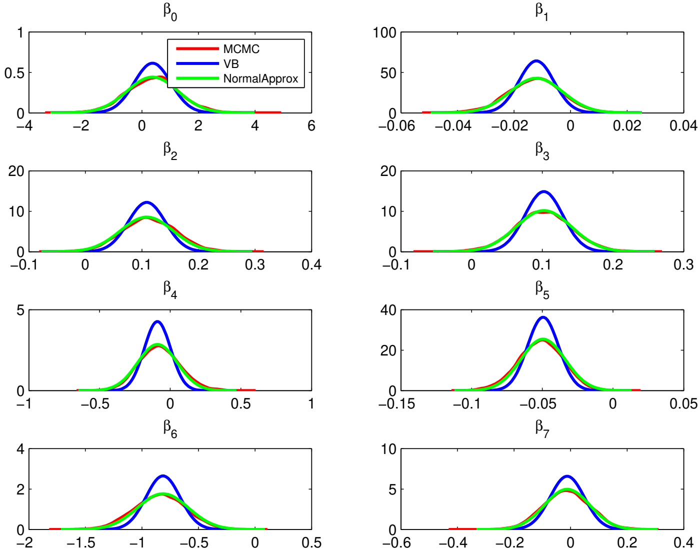
where

$$\tilde{\mu}_\beta = \left(\mathbf{X}^\top \mathbf{X} + \Sigma_\beta^{-1}\right)^{-1} \mathbf{X}^\top \tilde{\mu}_{\mathbf{u}}$$

and

$$\tilde{\mu}_{\mathbf{u}} = \mathbf{X}\tilde{\mu}_\beta + \frac{\phi\left(\mathbf{X}\tilde{\mu}_\beta\right)}{\Phi\left(\mathbf{X}\tilde{\mu}_\beta\right)^{\mathbf{y}}\left[\Phi\left(\mathbf{X}\tilde{\mu}_\beta\right) - \mathbf{1}_n\right]^{\mathbf{1}_n - \mathbf{y}}}.$$

Probit example (n=200 observations)



Probit example