On Bayesian Computation

Michael I. Jordan

with Elaine Angelino, Maxim Rabinovich, Martin Wainwright and Yun Yang
Previous Work: Information Constraints on Inference

- Minimize the minimax risk under **constraints**
  - privacy constraint
  - communication constraint
  - memory constraint
- Yields **tradeoffs** linking statistical quantities (amount of data, complexity of model, dimension of parameter space, etc) to “externalities” (privacy level, bandwidth, storage, etc)
Ongoing Work: Computational Constraints on Inference

- Tradeoffs via convex relaxations
- Tradeoffs via concurrency control
- Bounds via optimization oracles
- Bounds via communication complexity
- Tradeoffs via subsampling

- All of this work has been frequentist; how about Bayesian inference?
Today’s talk: Bayesian Inference and Computation

- Integration rather than optimization
- Part I: Mixing times for MCMC
- Part II: Distributed MCMC
Part I: Mixing Times for MCMC
MCMC and Sparse Regression

- Statistical problem: predict/explain a response variable $Y$ based on a very large number of features $X_1, \ldots, X_d$.

- Formally:
  \[ Y = X_1 \beta_1 + X_2 \beta_2 + \cdots + X_d \beta_d + \text{noise}, \quad \text{or} \]
  \[ Y_{n \times 1} = X_{n \times d} \beta_{d \times 1} + \text{noise}, \quad \text{(matrix form)} \]

  number of features $d \gg$ sample size $n$.

- Sparsity condition: number of non-zero $\beta_j$’s $\ll n$.

- Goal: variable selection—identify influential features

- Can this be done efficiently (in a computational sense)?
Statistical Methodology

- Frequentist approach: penalized optimization

  \[
  f(\beta) = \frac{1}{2n} \| Y - X\beta \|_2^2 + P_\lambda(\beta)
  \]

  - Objective function
  - Goodness of fit
  - Penalty term

- Bayesian approach: Monte Carlo sampling

  \[
  P(\beta | Y) \propto P(Y | \beta) \times P(\beta)
  \]

  - Posterior probability
  - Likelihood
  - Prior probability
Computation and Bayesian inference

Posterior probability $P(\beta \mid Y)$

$$=
\underbrace{C(Y)} \times \underbrace{P(Y \mid \beta)} \times \underbrace{P(\beta)}$$

- Normalizing constant
- Likelihood
- Prior probability

- The most widely used tool for fitting Bayesian models is the sampling technique based on Markov chain Monte Carlo (MCMC)

- The theoretical analysis of the computational efficiency of MCMC algorithms lags that of optimization-based methods
Computational Complexity

- Central object of interest: the mixing time $t_{\text{mix}}$ of the Markov chain

- Bound $t_{\text{mix}}$ as a function of problem parameters $(n, d)$
  - rapidly mixing—$t_{\text{mix}}$ grows at most polynomially
  - slowly mixing—$t_{\text{mix}}$ grows exponentially

- It has long been believed by many statisticians that MCMC for Bayesian variable selection is necessarily slowly mixing; efforts have been underway to find lower bounds but such bounds have not yet been found
Model

\[ Y = X_\gamma \beta_\gamma + w, \quad w \sim \mathcal{N}(0, \phi^{-1} I_n) \]

Precision prior:
\[ \pi(\phi) \propto \frac{1}{\phi} \]

Regression prior:
\[ (\beta_\gamma \mid \gamma) \sim \mathcal{N}(0, g \phi^{-1}(X_\gamma^T X_\gamma)^{-1}) \]

Sparsity prior:
\[ \pi(\gamma) \propto \left(\frac{1}{p}\right)^{\kappa |\gamma|} \mathbb{I}[|\gamma| \leq s_0]. \]
Assumption A (Conditions on $\beta^*$)

The true regression vector has components $\beta^* = (\beta_S^*, \beta_{Sc}^*)$ that satisfy the bounds

**Full $\beta^*$ condition:** $\left\| \frac{1}{\sqrt{n}} X \beta^* \right\|_2^2 \leq g \sigma^2 \frac{\log p}{n}$

**Off-support $S^c$ condition:** $\left\| \frac{1}{\sqrt{n}} X_{S^c} \beta_{S^c}^* \right\|_2^2 \leq \tilde{L} \sigma^2 \frac{\log p}{n}$,

for some $\tilde{L} \geq 0$. 
Assumption B (Conditions on the design matrix)

The design matrix has been normalized so that \( \|X_j\|_2^2 = n \) for all \( j = 1, \ldots, p \); moreover, letting \( Z \sim N(0, I_n) \), there exist constants \( \nu > 0 \) and \( L < \infty \) such that \( L \nu \geq 4 \) and

Lower restricted eigenvalue (RE(\( s \))):

\[
\min_{|\gamma| \leq s} \lambda_{\min}\left( \frac{1}{n} X^{\top} \gamma X^\gamma \right) \geq \nu,
\]

Sparse projection condition (SI(\( s \))):

\[
\mathbb{E}_Z \left[ \max_{|\gamma| \leq s} \max_{k \in [p] \setminus \gamma} \frac{1}{\sqrt{n}} |\langle (I - \Phi^\gamma) X_k, Z \rangle| \right] \leq \frac{1}{2} \sqrt{L \nu \log p},
\]

where \( \Phi^\gamma \) denotes projection onto the span of \( \{X_j, j \in \gamma\} \).

This is a mild assumption, needed in the information-theoretic analysis of variable selection.
Assumption C (Choices of prior hyperparameters)

The noise hyperparameter $g$ and sparsity penalty hyperparameter $\kappa > 0$ are chosen such that

$$g \leq p^{2\alpha}$$

for some $\alpha > 0$, and

$$\kappa + \alpha \geq C_1(L + \tilde{L}) + 2$$

for some universal constant $C_1 > 0$. 

Assumption D (Sparsity control)

For a constant $C_0 > 8$, one of the two following conditions holds:

**Version D($s^*$):** We set $s_0 := p$ in the sparsity prior and the true sparsity $s^*$ is bounded as

$$\max\{1, s^*\} \leq \frac{1}{8C_0K} \left\{ \frac{n}{\log p} - 16\tilde{L}\sigma_0^2 \right\}$$

for some constant $K \geq 4 + \alpha + c\tilde{L}$, where $c$ is a universal constant.

**Version D($s_0$):** The sparsity parameter $s_0$ satisfies the sandwich relation

$$\max\{1, (2\nu^{-2}\omega(X) + 1)s^*\} \leq s_0 \leq \frac{1}{8C_0K} \left\{ \frac{n}{\log p} - 16\tilde{L}\sigma_0^2 \right\},$$

where $\omega(X) := \max_\gamma \| (X_\gamma X_\gamma)\inv X_\gamma^T X_{\gamma^*\setminus\gamma} \|_{op}^2$. 
Our Results

Theorem (Posterior concentration)

Given Assumption A, Assumption B with \( s = Ks^* \), Assumption C, and Assumption D\((s^*)\), if \( C_\beta \) satisfies

\[
C_\beta^2 \geq c_0 \nu^{-2} (L + \tilde{L} + \alpha + \kappa) \sigma_0^2 \frac{\log p}{n},
\]

we have \( \pi_n(\gamma^* | Y) \geq 1 - c_1 p^{-1} \nu \)

Compare to \( \ell_1 \)-based approaches, which require an irrepresentable condition:

\[
\max_{k \in [d], S: |S| = s^*} \| X_k^T X_S (X_S^T X_S)^{-1} \|_1 < 1
\]
Our Results

Theorem (Rapid mixing)

Suppose that Assumption A, Assumption B with \( s = s_0 \), Assumption C, and Assumption D(\( s_0 \)) all hold. Then there are universal constants \( c_1, c_2 \) such that, for any \( \epsilon \in (0, 1) \), the \( \epsilon \)-mixing time of the Metropolis-Hastings chain is upper bounded as

\[
\tau_\epsilon \leq c_1 \, ps_0^2 \left( c_2 \alpha (n + s_0) \log p + \log(1/\epsilon) + 2 \right)
\]

with probability at least \( 1 - c_3 p^{-c_4} \).
High Level Proof Idea

- A Metropolis-Hasting random walk on the $d$-dim hypercube $\{0, 1\}^d$
- Canonical path ensemble argument: for any model $\gamma \neq \gamma^*$, find a path from $\gamma$ to $\gamma^*$, along which acceptance ratios are high, where $\gamma^*$ is the true model
Part II: Distributed MCMC
Traditional MCMC

- **Serial, iterative** algorithm for generating samples
- **Slow** for two reasons:
  1. Large number of iterations required to converge
  2. Each iteration depends on the entire dataset
- Most research on MCMC has targeted (1)
- Recent threads of work target (2)
Serial MCMC

Data

Single core

Samples
Data-parallel MCMC

Data

Parallel cores

“Samples”
Aggregate samples from across partitions — but how?

Data

Parallel cores

“Samples”

Aggregate
Factorization motivates a data-parallel approach

\[ \pi(\theta | x) \propto \pi(\theta) \pi(x | \theta) = \prod_{j=1}^{J} \pi(\theta)^{1/J} \pi(x^{(j)} | \theta) \]

\( \pi(\theta | x) \) \hspace{1cm} \pi(\theta) \hspace{1cm} \pi(x | \theta) \hspace{1cm} \text{posterior} \hspace{1cm} \text{prior} \hspace{1cm} \text{likelihood} \hspace{1cm} \text{sub-posterior}
Factorization motivates a data-parallel approach

\[
\pi(\theta | x) \propto \pi(\theta) \pi(x | \theta) = \prod_{j=1}^{J} \pi(\theta)^{1/J} \pi(x^{(j)} | \theta)
\]

- Partition the data as \(x^{(1)}, \ldots, x^{(J)}\) across \(J\) cores
- The \(j\)th core samples from a distribution proportional to the \(j\)th sub-posterior (a ‘piece’ of the full posterior)
- Aggregate the sub-posterior samples to form approximate full posterior samples
Aggregation strategies for sub-posterior samples

\[
\pi(\theta | \mathbf{x}) \propto \pi(\theta) \pi(\mathbf{x} | \theta) = \prod_{j=1}^{J} \pi(\theta)^{1/J} \pi(\mathbf{x}(j) | \theta)
\]

Sub-posterior density estimation (Neiswanger et al, UAI 2014)
Weierstrass samplers (Wang & Dunson, 2013)
Weighted averaging of sub-posterior samples

- Consensus Monte Carlo (Scott et al, Bayes 250, 2013)
- Variational Consensus Monte Carlo (Rabinovich et al, NIPS 2015)
Aggregate ‘horizontally’ across partitions
Naïve aggregation = Average

\[
\text{Aggregate} (\text{, } \text{, } \text{, }) = 0.5 \times + 0.5 \times
\]
Less naïve aggregation = Weighted average

\[
\text{Aggregate}\left(\begin{array}{c}
\end{array}\right) = 0.58 \times \text{print} + 0.42 \times \text{print}
\]
Consensus Monte Carlo (Scott et al, 2013)

Weights are inverse covariance matrices
Motivated by Gaussian assumptions
Designed at Google for the MapReduce framework
Variational Consensus Monte Carlo

**Goal:** Choose the aggregation function to best approximate the target distribution

**Method:** Convex optimization via variational Bayes
Variational Consensus Monte Carlo

**Goal:** Choose the aggregation function to best approximate the target distribution

**Method:** Convex optimization via variational Bayes

\[ F = \text{aggregation function} \]
\[ q_F = \text{approximate distribution} \]

\[ \mathcal{L}(F) = \mathbb{E}_{q_F} [\log \pi(X, \theta)] + H[q_F] \]

- **objective**
- **likelihood**
- **entropy**
**Goal:** Choose the aggregation function to best approximate the target distribution

**Method:** Convex optimization via variational Bayes

\[ F = \text{aggregation function} \]
\[ q_F = \text{approximate distribution} \]

\[ \tilde{\mathcal{L}}(F) = \mathbb{E}_{q_F} [\log \pi (X, \theta)] + \tilde{H}[q_F] \]

- **Objective:** \( \mathbb{E}_{q_F} [\log \pi (X, \theta)] \)
- **Likelihood:** \( \mathbb{E}_{q_F} [\log \pi (X, \theta)] \)
- **Relaxed Entropy:** \( \tilde{H}[q_F] \)
**Goal:** Choose the aggregation function to best approximate the target distribution

**Method:** Convex optimization via variational Bayes

\[ F = \text{aggregation function} \]
\[ q_F = \text{approximate distribution} \]

\[ \tilde{L}(F) = \mathbb{E}_{q_F}[\log \pi(X, \theta)] + \tilde{H}[q_F] \]

No mean field assumption
Variational Consensus Monte Carlo

Aggregate \( (\cdot, \cdot) \) = \[ + \]

- **Optimize** over weight matrices
- **Restrict** to valid solutions when parameter vectors constrained
Variational Consensus Monte Carlo

**Theorem (Entropy relaxation)**

*Under mild structural assumptions, we can choose*

\[
\tilde{\mathcal{H}}[q_F] = c_0 + \frac{1}{K} \sum_{k=1}^{K} h_k(F),
\]

*with each \( h_k \) a concave function of \( F \) such that*

\[
\mathcal{H}[q_F] \geq \tilde{\mathcal{H}}[q_F].
\]

*We therefore have*

\[
\mathcal{L}(F) \geq \tilde{\mathcal{L}}(F).
\]
Theorem (Concavity of the variational Bayes objective)

Under mild structural assumptions, the relaxed variational Bayes objective

\[ \tilde{\mathcal{L}}(F) = \mathbb{E}_{q_F} [\log \pi (X, \theta)] + \tilde{H}[q_F] \]

is concave in \( F \).
Empirical evaluation

- Compare three aggregation strategies:
  - Uniform average
  - Gaussian-motivated weighted average (CMC)
  - Optimized weighted average (VCMC)

- For each algorithm $\mathcal{A}$, report approximation error of some expectation $\mathbb{E}_\pi[f]$, relative to serial MCMC

$$\epsilon_{\mathcal{A}}(f) = \frac{|\mathbb{E}_\mathcal{A}[f] - \mathbb{E}_{\text{MCMC}}[f]|}{|\mathbb{E}_{\text{MCMC}}[f]|}$$

- Preliminary speedup results
Example 1: High-dimensional Bayesian probit regression

$\#\text{data} = 100,000, \ d = 300$

First moment estimation error, relative to serial MCMC

(Error truncated at 2.0)
Example 2: High-dimensional covariance estimation

Normal-inverse Wishart model

\#data = 100,000, \#dim = 100 \implies 5,050 parameters

(L) First moment estimation error (R) Eigenvalue estimation error
Example 3: Mixture of 8, 8-dim Gaussians

Error relative to serial MCMC, for cluster comembership probabilities of pairs of test data points

![Error Graphs]

- Left: $\sigma = 1$
- Right: $\sigma = 2$

Legend:
- Uniform
- Gaussian
- VCMC

Number of cores: 5, 10, 25, 50, 100
VCMC reduces CMC error at the cost of speedup ($\sim 2x$)

VCMC speedup is approximately linear

Number of cores

<table>
<thead>
<tr>
<th>Cores</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.00</td>
</tr>
<tr>
<td>10</td>
<td>0.02</td>
</tr>
<tr>
<td>25</td>
<td>0.04</td>
</tr>
<tr>
<td>50</td>
<td>0.06</td>
</tr>
<tr>
<td>100</td>
<td>0.08</td>
</tr>
</tbody>
</table>

Gaussian mixture ($\sigma=1$)

Speedup relative to serial
Discussion

Contributions

- Convex optimization framework for Consensus Monte Carlo
- Structured aggregation accounting for constrained parameters
- Entropy relaxation
- Empirical evaluation