MCMC (and MC) Wrapup

Scott Sanner
OSU CS536
MCMC Wrapup

• Practical Implementation Details
  – Burn-in
  – Thinning
  – Blocking
  – Collapsing
  – Augmenting

• Know your samplers
  – (Adaptive) Rejection: useful low-dim. method, see book/web
  – Gibbs: great but some failure modes
  – Metropolis-Hastings: useful sampler
  – Hamiltonian Monte Carlo: state-of-the-art for continuous only

• Detailed balance
  – What is it?
  – Proofs for Metropolis-Hastings and HMC
Practical Implementation Details I

• **Burn-in**
  – Throw away first n samples of Markov chain
  – Why?
    • Because it takes a little while to fall into the stationary distribution
  – What n? Good question!
    • Textbooks address this (e.g. using multiple chains)
    • But in practice often n=1000 or n=10000 samples
      – Often implementers don’t bother and assume the non-stationary part will be averaged out if the chain is long enough
Practical Implementation Details II

• Expectation assumes samples are i.i.d.
• Consecutive samples of Markov Chain are correlated
  – Hence leading to biased mean, variance, etc.
  – However, every 10th sample is not as correlated
  – Every 1,000,000th sample much less correlated (i.i.d.?)

• Idea: thinning
  – Retain every kth sample of Markov Chain

• What k? Good question!
  – Not too large to avoid inefficiency (k=1,000,000 inefficient)
  – Not too small to lead to heavy bias (k=1: most bias)
  – Again, in practice, many use k=1
    • Biased, but most efficient use of samples
Practical Implementation Details III

- **Blocked sampling**
  - Can jointly sample \( p(x,y|z=.,w=.) \)
    - as opposed to \( p(x|y=.,z=.,w=.) \)
    - still converges
  - Question: then why not sample \( p(x,y,z,w) \)?

- **Collapsed sampling**
  - Given \( p(x,y,z) \), can marginalize \( y \) then sample \( p(x,z) \)
  - A great way to work with determinism (more later)
  - Can recover expectation \( E[y] \)... how?

- **Augmented sampling (anti-collapsing!)**
  - Can augment a graphical model with new variables so long as the marginal distribution over the original variables is unchanged
  - Sometimes makes sampling easier (see Afshar, Sanner, et al AAAI-15)
Aside: Recovery of Expectations after Collapsing

• Assume we have \( P(y,x) = P(y|x) \ P(x) \) and we trivially collapse out \( y \)
  – Then we sample \( N \ x_i \)’s from marginal distribution \( P(x) \)
• To be concrete, assume \( P(y|x) = N(y; 2x, \sigma^2) \) so \( E_{P(y|x)}[y|x] = 2x \)
  – In general, just need to know mean of our conditional distribution
  – And most often we’ll know this analytically

• Then we can recover \( E[y] \) using samples from \( P(x) \) :
  \[
  E_{P(y)}[y] = \int_y y \ P(y) \ dy = \int_y y \ \int_x P(y|x)P(x) \ dx \ dy
  = \int_x P(x) \ \int_y y \ P(y|x) \ dy \ dx \quad \text{(distributive/reverse + Fubini–Tonelli theorem)}
  = \int_x P(x) \ E_{P(y|x)}[y|x] \ dx
  = E_{P(x)}[ E_{P(y|x)}[y|x] ]
  = E_{P(x)}[ 2x ]
  = \frac{1}{N} \sum_{x_i} 2x_i \quad \text{QED or... Yay!}
  
  – If \( y \) was a parameter in our graphical model then we can recover the
    mean parameter estimate (e.g., all we need from LDA)
Sampling: Visual Overview

• Rejection sampling (MC but not MCMC)
  - Rejects many samples… slow but has its uses. We’ll come back to this later.

• Metropolis Hastings (MCMC)
  - Requires proposal tuning… slow to converge if not.

• Gibbs Sampling (MCMC)
  - Requires inference in graphical model.

\[ p(x \mid y, \ldots) \]
\[ p(y \mid x, \ldots) \]
Failure Modes of Gibbs Sampling

- Non-communicating regions: rarely swap modes!

- Determinism (0 probabilities)
  - $p(y,x)$
  - $p(y|x)$ is deterministic
  - Never change $(y,x)$
  - Solution: collapse out determinism

![Diagram showing non-communicating regions and determinism](image)
Metropolis-Hastings

• Define a proposal $q(x' | x^{t-1})$
• At joint state $x^{t-1}$ propose next state $x'$
• Accept ($x^t = x'$) with probability:

$$a(x'|x^{(t-1)}) = \min \left[ 1, \frac{p(x') \cdot q(x^{(t-1)} | x')}{p(x^{(t-1)}) \cdot q(x' | x^{(t-1)})} \right]$$

otherwise remain ($x^t = x^{t-1}$)

• If proposal is symmetric: $q(x' | x) = q(x | x')$

$$a(x'|x^{(t-1)}) = \min \left[ 1, \frac{p(x')}{p(x^{(t-1)})} \right]$$
Hamiltonian Monte Carlo (HMC)

• Simulates an object moving in potential field
  – potential field given by the distribution

• Jumps change all dimensions at once (like MH, unlike Gibbs)
  – So not prone to failure modes of single coordinate updates of Gibbs sampling
  – Different from MH which can also update all dimensions on a sample, HMC does not require user-defined proposal, but rather computes updates using derivatives
    • If specify probability via probabilistic program, just use AutoDiff
    • See STAN tool: http://mc-stan.org/

• HMC takes large jumps thanks to momentum term
  – Samples nearly i.i.d.
  – Leads to faster mixing rate in high dimensions

• But not panacea:
  – No discrete variables, must collapse
  – Does not work well with piecewise distributions

• See excellent HMC overview from Radford Neal:
Detailed Balance

• Sufficient conditions for the unique stationary distribution to exist and be $p(x)$:
  – Markov chain must be ergodic (all states irreducible)
    • Note: if defining own proposal (MH), must lead to ergodicity
    • Symmetric proposal (Gaussian) guarantees this
  – Markov chain is reversible, i.e., satisfies detailed balance:
    $$p(x')T(x' \rightarrow x'') = p(x'')T(x'' \rightarrow x')$$
  – Sufficient but not necessary
Metropolis-Hastings satisfies Detailed Balance

- Forward and backward transition probabilities for MH:

\[ T(x^{(t-1)} \rightarrow x') = q(x' \mid x^{(t-1)}) \cdot a(x' \mid x^{(t-1)}) \]
\[ T(x' \rightarrow x^{(t-1)}) = q(x^{(t-1)} \mid x') \cdot a(x^{(t-1)} \mid x') \]

where \( q \) is any proposal and
\[ a(x' \mid x^{(t-1)}) = \min \left[ 1, \frac{p(x') \cdot q(x^{(t-1)} \mid x')}{p(x^{(t-1)}) \cdot q(x' \mid x^{(t-1)})} \right] \]

- Then detailed balance requires:

\[ p(x') \cdot q(x^{(t-1)} \mid x') \cdot a(x^{(t-1)} \mid x') = p(x^{(t-1)}) \cdot q(x' \mid x^{(t-1)}) \cdot a(x' \mid x^{(t-1)}) \]

**Proof for symmetric \( q \):** For simplicity, assume proposal is symmetric then \( q() \)'s cancel and only have to look at \( a() \)'s and \( p() \)'s. Same general proof approach if \( q \) asymmetric, but have to compare \( pq \)'s instead of just \( p \)'s as done below.

**Case 1:** \( x' = x^{(t-1)} \Rightarrow p(x') = p(x^{(t-1)}) \Rightarrow a() \)'s also match. So equality holds.

**Case 2:** \( x' \neq x^{(t-1)} \): Let \( p(x') > p(x^{(t-1)}) \). Then \( a(x' \mid x^{(t-1)}) = 1 \) and \( a(x^{(t-1)} \mid x') = p(x^{(t-1)}) / p(x') \).

Plugging these in, we see both sides match. Reverse case \( p(x^{(t-1)}) > p(x') \) by symmetry.
Gibbs is special case of MH

- For a Gibbs proposal, second arg of min in MH is 1:

\[
a(x'|x^{(t-1)}) = \min \left[ 1, \frac{p(x') \cdot q(x^{(t-1)} | x')}{p(x^{(t-1)}) \cdot q(x' | x^{(t-1)})} \right]
\]

Proof: \( p(x) = p(x_i | x_{\backslash i}) \cdot p(x_{\backslash i}) \). Can divide both \( p \) and \( q \) into transition for \( x_i \) and \( x_{\backslash i} \). Since \( x_{\backslash i} \) is same in \( x' \) and \( x^{t-1} \) these parts of \( p() \) and \( q() \) cancel. Then for 2nd arg of \( \min[\cdot] \) we get \( p(x_i' | x_{\backslash i}) \cdot q(x_i^{t-1} | x_{\backslash i}) / p(x_i^{t-1} | x_{\backslash i}) \cdot q(x_i' | x_{\backslash i}) \). Considering that the proposal \( q(x_i' | x_{\backslash i}) = p(x_i' | x_{\backslash i}) \) and likewise \( q(x_i^{t-1} | x_{\backslash i}) = p(x_i^{t-1} | x_{\backslash i}) \), \( p() \)'s cancel with \( q() \)'s yielding 1.

Also, full state update requires cycling through all variables. If we start in stationary \( p() \), easy to see that after one full update that we should stay in stationary.

- Then Gibbs is MH where proposal always accepts
  - MH satisfies detailed balance \( \Rightarrow \) Gibbs must also satisfy detailed balance
  - Note that Gibbs proposal derived from joint, hence no “tuning” needed
MCMC Wrapup

- State-of-the-art for sampling based inference
  - Many techniques
  - Each technique has its advantages and disadvantages
    - Some discussed here, many more
    - Takes a lot of experience to know when to use what
    - See MCMC Handbook
      http://www.mcmchandbook.net/HandbookTableofContents.html
Returning to non-MCMC Methods

Critical to cover two final techniques
Monte Carlo Methods

- **Forward Sampling**
  - Obvious first approach for Bayes net sampling
  - Rejects a lot when we have evidence

- **Is there a way to avoid rejection?**
  - Yes, likelihood weighting
  - Which is justified by importance sampling
  - Still does not resolve forward sampling issues
    - I.e., prior can be misleading when sampling from posterior
  - But importance sampling is a useful trick to know!
Recap: Forward sampling

Evidence: \( X_3 = 0 \)

// generate sample \( k \)
1. Sample \( x_1 \) from \( P(x_1) \)
2. Sample \( x_2 \) from \( P(x_2 | x_1) \)
3. Sample \( x_3 \) from \( P(x_3 | x_1) \)
4. If \( x_3 \neq 0 \), reject sample and start from 1, otherwise
5. Sample \( x_4 \) from \( P(x_4 | x_2, x_3) \)
Recap: Forward Sampling w/ Evidence

Input: Bayesian network
\[ X = \{X_1, \ldots, X_N\}, \text{N- #nodes} \]
\[ E - \text{evidence}, \text{T - # samples} \]

Output: T samples consistent with E

1. For \( t = 1 \) to \( T \)
2. For \( i = 1 \) to \( N \)
3. \( X_i \leftarrow \text{sample } x_i^t \text{ from } P(x_i \mid \text{pa}_i) \)
4. If \( X_i \in E \) and \( X_i \neq x_i \), reject sample:
5. \( i = 1 \) and go to step 2
Likelihood Weighting

For each each $X_i$ in topological order $o = (X_1,\ldots, X_n)$:

$$w_k = 1$$

if $X_i \notin E$

$$X_i \leftarrow \text{sample } x_i \text{ from } P(x_i \mid pa_i)$$

else

assign $X_i = e_i$

$$w_k = w_k \cdot P(e_i \mid pa_i)$$

Don’t discard evidence, weight sample by its likelihood!

We’ll see how to use this on next slide
Likelihood Weighting

Compute Sample Likelihood:

\[ w_k = \prod_{E} P(e_i \mid pa(E_i)) \]

Compute Query:

\[ P(Y = y \mid E) = \frac{\sum_{k} w_k \delta(s^k)}{\sum_{k} w_k} = \frac{\sum_{k,Y=y} w_k}{\sum_{k} w_k} \]

where \( \delta(s^k) = \begin{cases} 1, & y^k = y \\ 0, & otherwise \end{cases} \)
Importance Sampling

• Can have any proposal $Q(x)$ that we want:

$$E_{Q(x)}\left[ \frac{P(x)}{Q(x)} f(x) \right] = \frac{1}{T} \sum_{t=1}^{T} \frac{P(x^t)}{Q(x^t)} f(x^t)$$

where $x^t \sim Q(x)$

• Prove this is same as $E_{P(x)}[f(x)]$:

$$E_{Q(x)}\left[ \frac{P(x)}{Q(x)} f(x) \right] = \int Q(x) \frac{P(x)}{Q(x)} f(x) dx = \int P(x) f(x) dx = E_{P(x)}[f(x)]$$
Connection between Importance Sampling and Likelihood Weighting

• Likelihood Weighting (LW) is importance sampling

\[
E_{Q(x)} \left[ \frac{P(x)}{Q(x)} f(x) \right] = \frac{1}{T} \sum_{t=1}^{T} \frac{P(x^t)}{Q(x^t)} f(x^t)
\]

• The LW proposal Q proposes that the evidence be sampled with probability 1
  – P factorizes into conditionals of evidence and non-evidence
    • Q and P match for the non-evidence conditional probabilities
    • Q is 1 and P is the likelihood for evidence conditional probabilities
  – Sample weight = \( \prod \) evidence conditioned on parent samples
Importance Sampling (IS)

- A few tips for computing expectations
  - $Q(x)$ can never be 0 since weight by $P(x)/Q(x)$
  - Large $P(x)/Q(x)$ undesirable
    - A few samples get all of the weight
    - Lowers effective sample size (can be computed, see link below)
  - In general:
    - $Q(x)$ should match shape of $P(x)$
    - $Q(x)$ should have heavier tails than $P(x)$ to avoid large $P(x)/Q(x)$
  - And specifically for computing $E_{P(x)}[f(x)]$:
    - Choose $Q(x) \propto |f(x)|P(x)$!
    - Can prove this leads to lowest variance sampler!
  - If normalizer constants unknown, use self-normalized IS:
    \[
    E_{P(x)}[f(x)] = \sum_{t=1}^{T} \frac{P(x^t)}{Q(x^t)} f(x^t) / \sum_{t=1}^{T} \frac{P(x^t)}{Q(x^t)}
    \]

- For more details:
  [http://www.wikicoursenote.com/wiki/A_Deeper_Look_into_Importance_Sampling](http://www.wikicoursenote.com/wiki/A_Deeper_Look_into_Importance_Sampling)
The Joy of Importance Sampling (IS)

• Counterfactual statistical reasoning
  – Given data sampled with one protocol
  – Want to determine expectation subject to different protocol (but cannot sample more data)

• Idea: actual protocol = Q, desired protocol = P
  – Compute expectations using IS!
  – Can effectively run experiments you did not run
  – Or undo known sampling biases in data collection

• Used for counterfactual A/B testing at Microsoft, etc.