Sampling based inference

- Resources.
  - Bishop, chapter 11
  - Koller and Friedman, chapter 12
  - Andrieu et al. (linked to on lecture page).

- Koller and Friedman uses “particles” terminology instead of “samples”.

Motivation for sampling methods

- Real problems are typically complex and high dimensional.

- Example, images as evidence for stuff in the world

Sampling based inference

- We have studied two themes in inference.
  - Marginalization / expectation / summing out or integration
  - Optimization

- Two flavors of activities
  - Fitting (inference using a model)
  - Learning (inference to find a model)

- These activities are basically the same in the generative modeling approach.

Motivation for sampling methods

- Real problems are typically complex and high dimensional.

- Suppose that we could generate samples from a distribution that is proportional to one we are interested in.

  Typical case we are often interested in is \( p(\theta|D) \)

  \[
p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)}
\]

  Consider \( \hat{p}(z) = p(\theta)p(D|\theta) \)
Motivation for sampling methods

• Generally, θ lives in a very high dimensional space.
• Generally, regions of high \( \hat{p}(z) \) is very little of that space.
• IE, the probability mass is very localized.
• Watching samples from \( \hat{p}(z) \) should provide a good maximum (one of our inference problems)

Motivation for sampling methods (II)

• Now consider computing the expectation of a function \( f(z) \) over \( p(z) \).
• Recall that this looks like \( E_{p(z)}[f] = \int f(z)p(z)dz \)
• How can we approximate or estimate E?

Motivation for sampling methods (II)

• Now consider computing the expectation of a function \( f(z) \) over \( p(z) \).
• Recall that this looks like \( E_{p(z)}[f] = \int f(z)p(z)dz \)
• A bad plan for computing E:

  Discretize the space where \( z \) lives into L blocks
  Then compute \( E_{p(z)}[f] = \frac{1}{L} \sum_{l=1}^{L} p(z) f(z) \)
Challenges for sampling

In real problems sampling \( p(z) \) is very difficult.

We typically do not know the normalization constant, \( Z \).
(So we need to use \( \tilde{p}(z) \)).

Even if we can draw samples, it is hard to know if (when) they are good, and if we have enough of them.

Evaluating \( \tilde{p}(z) \) is generally much easier (although, it can also be quite involved).

Sampling framework

We assume that sampling from \( \tilde{p}(z) \) is hard, but that evaluating \( \tilde{p}(z) \) is relatively easy.

We also assume that the dimension of \( z \) is high, and that \( \tilde{p}(z) \) may not have closed from (but we can evaluate it).

We will develop the material in the context of computing expectations, but sampling also supports picking a good answer, such as a MAP estimate of parameters.

Basic Sampling (so far)

- Uniform sampling (everything builds on this)
- Sampling from a multinomial
- Sampling for selected other distributions (e.g., Gaussian)
  - At least, Matlab knows how to do it.
- Sampling univariate distributions using the inverse of the cumulative distribution (recall from HW 2).
Basic Sampling (so far)

- Sampling directed graphical models using ancestral sampling.

Rejection Sampling

Assume that we have an easy to sample function, and a constant, \( k \), where we know that \( p(z) \leq k \cdot q(z) \).

1) Sample \( q(z) \)

2) Keep samples in proportion to \( \frac{p(z)}{k \cdot q(z)} \) and reject the rest.

Rejection Sampling

- Rejection sampling is hopeless in high dimensions, but is useful for sampling low dimensional “building block” functions.
- E.G., the Box-Muller method for generating samples from a Gaussian uses rejection sampling.

A second example where a gamma distribution is approximated by a Cauchy proposal distribution.
Rejection Sampling

- For complex functions, a good \( q() \) and \( k \) may not be available.
- One attempt to adaptively find a good \( q() \) (see Bishop 11.1.3)

Importance Sampling (unnormalized)

\[
p(z) = \frac{\tilde{p}(z)}{Z_p} \quad \text{and} \quad q(z) = \frac{\tilde{q}(z)}{Z_q}
\]

\[
E_{pr1}(f) = \frac{1}{L} \sum_{i=1}^{L} \frac{p(z^{(i)})}{q(z^{(i)})} f(z^{(i)}) \quad \text{(samples from } q(z^{(i)}) \text{, equivalently, } \tilde{q}(z^{(i)}))
\]

\[
= \frac{Z_q}{Z_p} \frac{1}{L} \sum_{i=1}^{L} \frac{p(z^{(i)})}{\tilde{q}(z^{(i)})} f(z^{(i)})
\]

\[
= \frac{Z_q}{Z_p} \frac{1}{L} \sum_{i=1}^{L} \tilde{\eta} f(z^{(i)}) \quad \text{(introducing } \tilde{\eta} = \frac{p(z^{(i)})}{\tilde{q}(z^{(i)})})
\]

\[
Z_p = \int \tilde{p}(z) dz
\]

\[
\frac{Z_q}{Z_p} = \frac{1}{L} \sum_{i=1}^{L} \tilde{\eta} q(z^{(i)}) \quad \text{(samples coming from } \tilde{q}(z^{(i)}))
\]

\[
E_{pr1}(f) \equiv \frac{1}{L} \sum_{i=1}^{L} \tilde{\eta}
\]

Importance Sampling (unnormalized)

\[
E_{pr1}(f) \equiv \frac{1}{L} \sum_{i=1}^{L} \tilde{\eta} f(z^{(i)}) \quad \text{(samples coming from } \tilde{q}(z^{(i)}))
\]

and \[
\frac{Z_q}{Z_p} = \frac{1}{L} \sum_{i=1}^{L} \tilde{\eta} q(z^{(i)}) \quad \text{(samples coming from } \tilde{q}(z^{(i)}))
\]

so \[
E_{pr1}(f) \equiv \frac{1}{L} \sum_{i=1}^{L} \tilde{\eta} f(z^{(i)})
\]

(samples coming from } \tilde{q}(z^{(i)}))
Importance sampling for graphical models

We know how to sample from directed graphical models where no variables are observed or conditioned on.

Suppose we want to use sampling to compute $p(Y = y)$.

$$p(Y = y) \equiv \frac{1}{L} \sum I(y^{(i)}, y) \quad \text{(samples from } p(y))$$

where $I(y^{(i)}, y) = \begin{cases} 1 & \text{if } y^{(i)} = y \\ 0 & \text{otherwise} \end{cases}$

Importance sampling for graphical models

EG, we might want to sample: $p(Y | E = e)$

or, we might want to evaluate: $p(y = Y | E = e)$

A fool-proof plan is to sample $p(y, e)$, and reject $e \neq E$

(Potentially very expensive!)

Importance sampling for graphical models

We know how to sample from directed graphical models where no variables are observed or conditioned on.

What about the case of a particular value of a subset of the variables.

EG, we might want to sample: $p(Y | E = e)$

or, we might want to evaluate: $p(y = Y | E = e)$

Importance sampling for graphical models

A natural idea is to use ancestral sampling on the graph, where we set $E=e$.

Kollar and Friedman develop this as sampling from the "mutilated" Bayesian network.
Mutilating graphical models

Set grade to $g^2$ and intelligence to $i^1$, and remove links.

Importance sampling for graphical models

A natural idea is to use ancestral sampling on the graph, where we set $E = e$.

However, when $E = e$, this can influence the correct sampling of $Y$, and we have ignored this!

Instead, we use samples from the mutilated network for the proposal distribution in importance sampling.

Importance sampling for graphical models

\[
\frac{p(y|e)}{q(y|e)} = \frac{P_{BN}(y|e)}{P_{MNB}(y|e)} = \frac{P_{BN}(y,e)}{P_{MNB}(y,e)}
\]

\[
p(y|e) \equiv \frac{1}{L} \sum_{L} P_{BN}(y,e) I(Y = y) \quad \text{(samples from } P_{MNB}(Y,e) \text{)}
\]

Markov chain Monte Carlo methods

- The approximations of expectation so far have assumed that the samples are independent draws.
- This sounds good, but in high dimensions, we do not know how to get good independent samples from the distribution.
- MCMC methods drop this requirement.
- Basic intuition
  - If you have finally found a region of high probability, stick around for a bit, enjoy yourself, grab some more samples.
Markov chain Monte Carlo methods

- Samples are conditioned on the previous one (this is the Markov chain).

- MCMC is generally a good hammer for complex, high dimensional, problems.

- Main downside is that it is not “plug-and-play”
  - Doing well requires taking advantage to the structure of your problem
  - MCMC tends to be expensive (but take heart---there may not be any other solution, and at least your problem is being solved).

Metropolis Example

We want samples $z^{(1)}, z^{(2)}, \ldots$

Again, write $p(z) = \tilde{p}(z)/Z$

Assume that $q\left(z^{(prev)} \mid z^{(prev)} \right)$ can be sampled easily

Also assume that $q()$ is symmetric, i.e., $q\left(z_{a} \mid z_{b}\right) = q\left(z_{b} \mid z_{a}\right)$

For example, $q\left(z^{(prev)} \mid z^{(prev)} \right) \sim \mathcal{N}\left(z; z^{(prev)}, \sigma^2\right)$

---

Metropolis Example

While not_bored
{
    Sample $q\left(z^{(prev)} \mid z^{(prev)} \right)$
    
    Accept with probability $A\left(z, z^{(prev)} \right) = \min\left(1, \frac{\tilde{p}(z)}{\tilde{p}(z^{(prev)})}\right)$
    
    If accept, emit $z$, otherwise, emit $z^{(prev)}$.

}
Metropolis Example

Green follows accepted proposals
Red are rejected moves.

Stationary Markov chains

- Recall that our goal is to have our Markov chain emit samples from our target distribution.
- This implies that the distribution being sampled at time $t+1$ is the same as that of time $t$ (stationary).
- If our stationary (target) distribution is $p()$, then if imagine an ensemble of chains, they are in each state with (long-run) probability $p()$.
  - On average, a switch from $s_1$ to $s_2$ happens as often as going from $s_2$ to $s$, otherwise, the percentage of states would not be stable
- If our stationary (target) distribution is $p()$, what do the transition probabilities look like?

Markov chain view

Denote an initial probability distribution by $p(z^{(1)})$

Define transition probabilities by:

$$T(z^{(prev)}, z) = p(z|z^{(prev)})$$

(a probability distribution)

$T = T_m( )$ can change over time, but for now, assume that it it is always the same (homogeneous chain)

A given chain evolves from a sample of $p(z^{(1)})$, and is an instance from an ensemble of chains.

Detailed balance

- Detailed balance is defined by:

$$p(z)T(z, z') = p(z')T(z', z)$$

(We assume that $T(\cdot)>0$)

- Detailed balance is a sufficient condition for a stationary distribution.

- Detailed balance is also referred to as reversibility.
Detailed balance implies stationary

\[ p(z) = \sum_{z'} p(z') T(z', z) \quad \text{(marginalization)} \]

If we have detailed balance, then

\[ p(z) T(z, z') = p^{(\text{prev})}(z') T(z', z) \]

So,

\[ p(z) = \sum_{z'} p^{(\text{prev})}(z') T(z', z) = \sum_{z'} p^{(\text{prev})}(z) T(z, z') = p^{(\text{prev})}(z') \]

Hence, detailed balance implies the distribution is stationary.

Markov chain Monte Carlo methods

- Samples are conditioned on the previous one (this is the Markov chain).
- We have given up the very natural preference for independent samples.
- Basic intuition why this might be a good idea
  - If you have finally found a region of high probability, stick around for a bit, enjoy yourself, grab some more samples.
- MCMC is generally a good hammer for complex, high dimensional, problems.

Recall terminology and chain evolution

Denote an initial probability distribution by \( p(z^{(1)}) \)

Define transition probabilities by:

\[ T(z^{(\text{prev})}, z) = p(z|z^{(\text{prev})}) \]

\( T = T_m(\ ) \) can change over time, but for now, assume that it is always the same (homogeneous chain)

Chains (think ensemble) evolve according to:

\[ p(z) = \sum_{z'} p(z') T(z', z) \]

Stationary Markov chains

- Our goal is to have our Markov chain emit samples from our target distribution.
- This implies that the distribution being sampled at time \( t+1 \) is the same as that of time \( t \) (stationary).
- If our stationary (target) distribution is \( p() \), then if imagine an ensemble of chains, they are in each state with (long-run) probability \( p() \).
  - On average, a switch from s1 to s2 happens as often as going from s2 to s, otherwise, the percentage of states would not be stable.
Detailed balance

- Detailed balance is defined by:

  \[ p(z)T(z,z') = p(z')T(z',z) \]

  (We assume that \( T(*) > 0 \))

- Detailed balance is a sufficient condition for a stationary distribution.

- Detailed balance is also referred to as reversibility.

Detailed balance (cont)

- Detailed balance (for \( p() \)) means that if our chain was generating samples from \( p() \), it would continue to do so.
  - We will address how it gets there shortly

- Does the Metropolis algorithm have detailed balance?

Detailed balance implies stationary

\[
p(z) = \sum_{z'} p(z') T(z',z) \quad \text{(marginalization)}
\]

If we have detailed balance, then

\[
p(z)T(z,z') = p(z')T(z',z)
\]

So,

\[
p(z) = \sum_{z'} p(z')T(z',z) = \sum_{z'} p(z') T(z,z') = p(z')
\]

Hence, detailed balance implies the distribution is stationary.

Metropolis Example

While not_bored

\{ 
  Sample \( q(z | z^{(prev)}) \)
  
  Accept with probability \( A(z,z^{(prev)}) = \min \left( 1, \frac{\hat{p}(z)}{\hat{p}(z^{(prev)})} \right) \)
  
  If accept, emit \( z \), otherwise, emit \( z^{(prev)} \).
\}

Same as \( \frac{p(z)}{p(z^{(prev)})} \)
Ergodic chains

- Different starting probabilities will give different chains
- We want our chains to converge (in the limit) to the same stationary state, regardless of starting distribution.
- Such chains are called ergodic, and the common stationary state is called the equilibrium state.
- Ergodic chains have a unique equilibrium.

When do our chains converge?

- Important theorem tells us that (for finite state spaces*) our chains converge to equilibrium under two relatively weak conditions.
- (1) Irreducible
  - We can get from any state to any other state
- (2) Aperiodic
  - The chain does not get trapped in cycles
- These are true for detailed balance which is sufficient, but not necessary for convergence.

*Infinite or uncountable state spaces introduces additional complexities.

Recall that in Metropolis, \[ A(z,z') = \min \left( 1, \frac{p(z)}{p(z')} \right) \]
\[ p(z')q(z|z')A(z,z') = q(z|z')\min(p(z'),p(z)) = q(z'|z)\min(p(z'),p(z)) \] (q() is symmetric)
\[ = p(z)q(z'|z)\min\left( \frac{p(z')}{p(z)},1 \right) = p(z)q(z'|z)\min\left( 1, \frac{p(z')}{p(z)} \right) = p(z)q(z'|z)A(z',z) \]
Intuition behind ergodic chains

Let $p^{(i)}(z)$ be the distribution at some time (e.g., initial distribution)

Let $p^{*}(z)$ be the stationary distribution

Let $p^{(i)}(z) = p^{*}(z) - q^{(i)}(z)$

Note that the elements of $p^{(i+1)}(z)$ and $p^{*}(z)$ sum to one, and thus the elements of $q(z)$ sum to zero.

Matrix-vector representation

Chains (think ensemble) evolve according to:

$p(z) = \sum_{z'} p(z') T(z',z)$

Matrix vector representation:

$\mathbf{p} = \mathbf{T} \mathbf{p}'$

And, after $n$ iterations after a starting point:

$p^{(n)} = \mathbf{T}^n \mathbf{p}^{(0)}$
Matrix representation

A single transition is given by

\[ p = T p' \]

Note what happens for stationary state:

\[ p' = T p' \]

So, \( p' \) is an eigenvector with eigenvalue one.

And, intuitively, if things converge, \( p' = T^n p^{(0)} \)

Aside on stochastic Matrices

• A right (row) stochastic matrix has non-negative entries, and its rows sum to one.

• A left (column) stochastic matrix has non-negative entries, and its columns sum to one.

• A doubly stochastic matrix has both properties.

Aside on stochastic Matrices

• \( T \) is a left (column) stochastic matrix.
  – If you are right handed, take the transpose

• The column vector, \( p \), also has non-negative elements, that sum to one (sometimes this is called a stochastic vector).

• Fun facts that we should do on the board
  – The product of a stochastic matrix and vector is a stochastic vector.
  – The product of two stochastic matrices is a stochastic matrix.

Aside on (stochastic) Matrix powers

Consider the eigenvalue decomposition of \( T \), \( T = E \Lambda E^{-1} \)

So, \( T^N = E \Lambda^N E^{-1} \)

Since \( T^N \) cannot grow without bound, the eigenvalues are inside \([-1, 1]\).

In fact, for our situation, the second biggest absolute value of the eigenvalues is less than one (not so easy to prove).
Aside on (stochastic) Matrix powers

We have $T^N = E \Lambda^N E^{-1}$

$$
\Lambda = \begin{pmatrix}
1 & 1 & \cdots \\
\lambda_2 & \lambda_2 & \cdots \\
\vdots & \vdots & \vdots \\
\lambda_K & \lambda_K & \cdots \\
\end{pmatrix}
$$

and

$$
\Lambda^{-1} = \begin{pmatrix}
1 & 0 & \cdots \\
0 & 1 & \cdots \\
\vdots & \vdots & \vdots \\
0 & 0 & \cdots \\
\end{pmatrix}
$$

$$
\Lambda^{-1} E^{-1} = \begin{pmatrix}
e^T_i \\
0 \\
\vdots \\
0
\end{pmatrix}
$$

and $E \Lambda^{-1} E^{-1} p \parallel e_i \parallel p^*$

So, $\Lambda^{-1} E^{-1} = \begin{pmatrix}
e^T_i \\
0 \\
\vdots \\
0
\end{pmatrix}$

In summary, $p^* \parallel e_i$ and $p^*$ stochastic means that $E \Lambda^{-1} E^{-1} p = p^*$

This is true, no matter what the initial point $p$ is.

So, glossing over details, we have convergence to equilibrium.
According to the previous, if $T$ is a stochastic matrix, then:

$$p^* \equiv T^N p$$

(No matter what $p$! They all will give the same answer).

Also, $p^* \parallel e^{(1)}$

---

**Main points about P-F**

- The maximal eigenvalue is strictly maximal (item 1).
- The corresponding eigenvector is “simple” (item 2).
- It has all positive (or negative) components (item 3).
- There is no other eigenvector that can be made non-negative.
- The maximal eigenvalue of a stochastic matrix has absolute value 1 (item 8 applied to stochastic matrix).

**Summary**

$p^* = TP^*$ is an eigenvector with eigenvalue one.

We have written it as $p^* \parallel e^{(1)}$ because $e^{(1)}$ is the eigenvector normalized to norm 1 (standard form).

Intuitively (perhaps), $T$ will reduce any component of $p$ orthogonal to $p^*$, and $T^N$ will kill off such components as $N \to \infty$. 

---

**Justification relies on Perron Frobenius theorem**

Let $A^r = (a_{ij})$ be an $n \times n$ positive matrix: $a_{ij} > 0$ for $1 \leq i,j \leq n$. Then the following statements hold.

1. There is a positive real number $r$, called the Perron root or the Perron–Frobenius eigenvalue, such that $r$ is an eigenvalue of $A$ and any other eigenvalue $\lambda$ (possibly, complex) is strictly smaller than $r$ in absolute value. If $r < 0$, then the spectral radius $\rho(A)$ is equal to $r$.
2. The Perron–Frobenius eigenvalue is simple: it is a simple root of the characteristic polynomial of $A$. Consequently, the eigenspace associated to it is one-dimensional. (The same is true for the left eigenspace, i.e., the eigenspace for $A^T$).
3. There exists an eigenvector $v = (v_1, \ldots, v_n)$ of $A$ with eigenvalue $r$ such that all components of $v$ are positive: $A v = r v$, $v > 0$ for $1 \leq i \leq n$. (Respectively, there exists a positive left eigenvector $w = (w_1, \ldots, w_n)$ with $w^T A = r w^T$, $w > 0$).
4. There are no other positive (nonzero) eigenvectors except $v$ (respectively, left eigenvectors except $w$), i.e., all other eigenvectors must have at least one negative or non-real component.
5. $\lim_{n \to \infty} A^r = r^nv$, where the left and right eigenvectors for $A$ are normalized so that $w^T r = 1$. Moreover, the matrix $r^n v w^T$ is the projection onto the eigenspace corresponding to $r$. This projection is called the Perron projection.
6. Collatz–Wielandt formula: for all non-negative non-zero vectors $x$, let $g(x)$ be the minimum value of $(Ax)^T x$ taken over all such $x$, such that $x^T r = 0$. Then $g$ is a real valued function whose maximum is the Perron–Frobenius eigenvalue.
7. A “Min-max” Collatz–Wielandt formula takes a form similar to the one above: for all strictly positive vectors $x$, let $g(x)$ be the maximum value of $(Ax)^T x$ taken over $x$. Then $g$ is a real valued function whose minimum is the Perron–Frobenius eigenvalue.
8. The Perron–Frobenius eigenvalue satisfies the inequalities

$$\min \left\{ \sum_j n_{ij} \right\} \leq r \leq \max \left\{ \sum_j n_{ij} \right\}$$

From Wikipedia
Algebraic proof

Neal ’93 provides an algebraic proof which does not rely on spectral theory.

(A question on the final might study this further).

Summary so far

• Under reasonable (easily checked and/or arranged) conditions, our chains converge to an equilibrium state.

• Easiest way to prove (or check) that this is the case is to show detailed balance.

• To use MCMC for sampling a distribution, we simply ensure that our target distribution is the equilibrium state.

• Variations on MCMC are mostly about improving the speed of convergence for particular situations.

Summary so far

• The time it takes to get reasonably close to equilibrium (where samples come from the target distribution) is called “burn in” time.
  – I.E., how long does it take to forget the starting state.
  – There is no general way to know when this has occurred.

• The average time it takes to visit a state is called “hit time”.

• What if we really want independent samples?
  – We can take every N\text{th} sample (some theories about how long to wait exist, but it depends on the algorithm and distribution).

Metropolis-Hastings MCMC method

• Like Metropolis, but now q() is not symmetric.
Metropolis-Hastings MCMC method

While not_bored
{
    Sample $q(z|z^{\text{prev}})$
    
    Accept with probability $A(z,z^{\text{prev}}) = \min\left(1, \frac{p(z)q(z^{\text{prev}}|z)}{p(z)q(z|z^{\text{prev}})}\right)$
    
    If accept, emit $z$, otherwise, emit $z^{\text{prev}}$.
}

Does Metropolis-Hastings have detailed balance?

$$p(z')q(z'|z)A(z,z') = \min(p(z')q(z'|z), p(z)q(z'|z))$$

$$= p(z)q(z'|z)\min\left(\frac{q(z'|z')}{q(z'|z)}, \frac{p(z')}{p(z)}\right)$$

$$= p(z)q(z'|z)\min\left(1, \frac{p(z')q(z'|z)}{p(z)q(z'|z)}\right)$$

$$= p(z)q(z'|z)A(z',z)$$

Metropolis-Hastings comments

- Again it does not matter if we use unnormalized probabilities.

- It should be clear that the previous version, where $q()$ is symmetric, is a special case.

Reversible Jump MH

- Suppose the dimension of your problem is not known (e.g., you want to estimate the number of clusters).
- Sampling now includes “jumping” changes probability space
- Requires a modification to Metropolis Hastings
  - Reversible jump MCMC, Green 95, 03

- RJMCMC is only about sampling. It does not tell you the number of dimensions
  - This must come from either the prior or the likelihood.
Gibbs sampling

- Gibbs sampling is another special case of MH.

- You might notice that the transition function, \( T() \), varies (cycles) over time.
  - This is a relaxation of our assumption used to provide intuition about convergence
  - However, it still OK because the concatenation of the \( T() \) for a cycle converge

Consider a set of N variables, \( x_1, x_1, ..., x_N \), Gibbs says

Initialize \( \{ x_i^{(0)} : i = 1, ..., M \} \)

While not_bored
  {  
    For i=1 to M
      {  
        Sample \( z_i^{(t+1)} \sim p \left( z_i^{(t+1)} | z_{i-1}^{(t+1)}, ..., z_{i+1}^{(t+1)}, z_M^{(t)} \right) \)
        Always accept (emit \( z = z_1^{(t+1)}, ..., z_i^{(t+1)}, z_{i+1}^{(t+1)}, ..., z_M^{(t)} \) )
      }
  }

Examples of Gibbs

- If one can specify the conditional distributions so that they can be sampled, Gibbs is often a very good method.

- Typical examples include symmetric systems like the Markov random fields we had for images.
  - With a Markov property, the conditional probability can be quite simple.
Examples of Gibbs

(From Dellaert and Zhu tutorial)

Gibbs as MH

\[ q_i(z|z^*) = p(\hat{z}_i|z_{-i}^*) \quad \text{and} \quad q_i(z^*|z) = p(z_i^*|z_{-i}) \]

And we have \( z_{ki} = z_{ki}^* \) because only \( i \) changes.

Gibbs as MH

\[
A(z^*, z) = \frac{p(z^*)q_i(z^*|z^*)}{p(z)q_i(z^*|z)} = \frac{p(z^*)p(z_i^*|z_{-i})q_i(z^*)}{p(z)p(z_i|z_{-i})q_i(z^*)}
\]

\[
= \frac{p(z_i)q_i(z_i^*)p(z_i^*)}{p(z_i)p(z_i|z_{-i})p(z_i^*)}
\]

\[
= 1
\]
Exploring the space

- Algorithms like Metropolis-Hastings exhibit “random walk behavior” if the step size (proposal variance) is small.
- If the step size is too big, then you get rejected too often.
- Adaptive methods exist (see slice sampling in Bishop).
- Another approach is to combine samplers with different properties.

Combined samplers

1. Initialise $x^{(0)}$.
2. For $i = 0$ to $N - 1$
   - Sample $u \sim U_{[0,1]}$.
   - If $u < \nu$
     - Apply the MH algorithm with a global proposal.
   - else
     - Apply the MH algorithm with a random walk proposal.

Annealing

- Analogy with physical systems.
- Relevant for optimization (not integration).
- Powers of probability distributions emphasize the peaks.
- If we are looking for a maximum within a lot of distracting peaks, this can help.

Annealing

- Define a temperature $T$, and a cooling schedule (black magic part).
- Lower temperatures correspond to emphasized maximal peaks.
  - Hence we exponentiate by $(1/T)$. 
Annealing

1. Initialise $x^{(0)}$ and set $T_0 = 1$.
2. For $i = 0$ to $N - 1$
   - Sample $u \sim \mathcal{U}_{[0,1]}$.
   - Sample $x^* \sim q(x^*|x^{(i)})$.
   - If $u < A(x^{(i)}, x^*) = \min\left\{1, \frac{p(x^*) q(x^{(i)}|x^*)}{p(x^{(i)}) q(x^{(i)}|x^{(i)})}\right\}$
     
     $x^{(i+1)} = x^*$
   
   else
     $x^{(i+1)} = x^{(i)}$
   
   - Set $T_{i+1}$ according to a chosen cooling schedule.

(From Andrieu et al)

Continuous versus discrete variables

- Derivatives of continuous distributions can tell you about the structure of your problem.
  - Opportunities for going much faster
- Naive approach is gradient ascent with added stochastic properties
  - Take a step, then perturb the result.
- Typical approach is to link the probability distribution to a potential energy function
  - Follow the system to find low energy (high probability)
  - Stochastic sampling via random momentum
  - An effective example method is Hybrid Monte Carlo

Hybrid Monte Carlo

- A more effective example method is Hybrid Monte Carlo
- Link the probability distribution to a potential energy function
  - Alternate stochastic sampling with “dynamics”.
    - The dynamics follow the system to find low energy (high probability)
- HMC is an “auxiliary variable sampler”
  - Important trick
    - To sample $p(z)$ we sample $p(z,r)$ or $p(z,r_1,r_2,...)$
    - Ignore the auxiliary variables when we use the samples.
Hamiltonian Dynamics

\[ p(z) = \frac{1}{Z} \exp(-E(z)) \]

We equate \( z \) with position, so \( E(z) \) is the potential energy.

High probability \( \Leftrightarrow \) Low energy

\[ E(z) = -\log(Z_p) - \log(p(z)). \]

Hamiltonian Dynamics

Recall that the gradient, \( \nabla \), is the vector of partial derivatives.

Recall from physics that force is the negative gradient of energy

From before \( E(z) = -\log(Z_p) - \log(p(z)) \)

So \( \nabla E(z) = \nabla (-\log(p(z))) \)

Or, in terms of log probabilities, we define

\[ \Delta(z) = \nabla \left( \log(p(z)) \right) = -\nabla E(z) \quad \text{(This is the force)} \]

Hamiltonian Dynamics

\[ p(z) = \frac{1}{Z} \exp(-E(z)) \quad \text{and} \quad \nabla E(z) = \nabla (-\log(p(z))) \]

Let \( r \) be the momentum vector for the system. Denote the kinetic energy by \( K(r) \).

\[ K(r) = \frac{1}{2} \|r\|^2 = \frac{1}{2} \sum r_i^2 \quad \text{(We assume that mass is one).} \]

Hamiltonian Dynamics

\[ H(z,r) = E(z) + K(r) \quad \text{(conserved)} \]

Our distribution with auxiliary variables is

\[ p(z,r) = \frac{1}{Z} \exp(-H(z,r)) \]
Hamiltonian Dynamics

\[ H(z, r) = E(z) + K(r) \] (conserved)

We follow \( z \) according to \( H \) with a random \( r \)

This can rapidly transport us towards (but not to) a local minimum thus avoiding random walk.

To follow \( H \), we observe that \( z \) changes proportional to \( r \), and \( r \) changes proportion to force \( -\nabla E \).

Again \( -\nabla E = \nabla (\log(p(z))) = \Delta p(z) \)

Following Dynamics

In HMC we follow the dynamics for \( L \) time steps of size \( \tau \) (tunable parameters).

In the "leap frog" method for each \( \tau \).
1. Take 1/2 step in \( r \).
2. Take a full step in \( z \).
3. Take 1/2 step in \( r \).

Following Dynamics

For \( L \) leap frog steps we have.

1. Take 1/2 step in \( r \).
2. \((L - 1)\) times take a full steps in \( z \), then \( r \).
3. Take a full step in \( z \).
4. Take 1/2 step in \( r \)

Following Dynamics

To take a full step in \( z \).

\[ z(\tau + 1) = z(\tau) + \epsilon \cdot \Delta(r(\tau)) \]

(\( \epsilon \) is the step size).
Following Dynamics

To take 1/2 step in $r$.

$$r\left(\tau + \frac{1}{2}\right) = r(\tau) + \frac{1}{2} \epsilon \cdot \Delta(z(\tau))$$

Where $\Delta(z(\tau)) = \nabla \log(p(z(\tau))) = -\nabla E(z)$ (force)

HMC dynamics step acceptance

- If our integration is perfect (i.e., in the limit as $t \rightarrow 0$) then energy is conserved.
  - Thus the value of distribution $p(z, r)$ is the same after the dynamics.

- If we assume no integration errors, we simply accept this step

- If we want to account for error accumulation, we accept the result according to:

$$\min\left(1, \frac{p(z', r')}{p(z, r)}\right) = \min\left(1, \exp\left[H(z, r) - H(z', r')\right]\right)$$

HMC stochastic step

- Typical instantiations sample the momentum variable

- Two common strategies
  - Sample the $r$ independently from a Gaussian
  - Sample $r$ from a Gaussian using Gibbs

- Note that in both of these cases the proposals are always accepted.
Putting it all together
(A typical vision lab sampler)

• Discrete variables are sampled using (reversible jump) Metropolis Hastings.

• Continuous variables are sampled using stochastic dynamics (essentially hybrid Monte Carlo).

• Discrete variables typically control topology or components
  – The number of components and their type (block, cylinder)
  – How components are connected (branches from a stem)

A typical vision lab sampler

• Randomly proposing structure is too expensive because of the high rejection rate.

• Solution (part one) is to use data driven sampling
  – Proposals are conditioned on distributions computed before we begin using the data
  – For example, the probability of a corner being present in each point in the image.

• Solution (part two) is to delay acceptance
  – Adjust continuous parameters using stochastic dynamics so that the proposed structure is a good fit to the data.

• We thus alternate between
  – (1) data driven proposals for new structure (or to switch or kill existing structure)
  – (2) exploring the continuous parameters of the structure

• Additional gains in optimization through having multiple samplers running in parallel exchange information