10-708: Probabilistic Graphical Models **Spring 2017**

17 : Optimization and Monte Carlo Methods

Lecturer: Avinava Dubey *Scribes: Neil Spencer, YJ Choe*

1 Recap

1.1 Monte Carlo

Monte Carlo methods such as rejection sampling and importance sampling allow us to compute the expectation of functions of random variables. They can also be used to simply obtain samples from the underlying distribution. In graphical models, they can be used to perform inference, even when we cannot compute the marginal distribution or the partition function directly.

However, there are several limitations of Monte Carlo methods. One important issue is that the performance of Monte Carlo methods relies heavily on having good proposal distributions, which are difficult to find when the true distribution is complex and/or high-dimensional. For example, in rejection sampling and importance sampling, the proposal distribution is fixed throughout the sampling procedure. This means that if the proposal distribution does not capture the true distribution sufficiently well, the algorithm will propose a lot of bad samples and the acceptance rate will be low.

1.2 Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) gets around this issue by having a proposal distribution conditioned on the current sample. In Metropolis-Hastings, the acceptance probability of a proposal is the ratio of importance weights and favors more "important" samples.

However, Metropolis-Hastings also has a problem of its own. Consider the two-dimensional toy example in Figure 1, where $P(x)$ denotes the true distribution and $Q(x' \mid x)$ denotes the symmetric proposal distribution. Due to the symmetry of Q, the acceptance rate simply becomes $A(x', x) = \min\left\{1, \frac{P(x')}{P(x)}\right\}$.

(a) $P(x)$ (b) $Q(x' \mid x)$

Figure 1: When Metropolis-Hastings can suffer.

Although the proposal distribution does not assume any correlation between the two dimensions, the (un-

known) true distribution has a high correlation, as shown in Figure 1. Then, when the current sample is at x_1, where the contour of P is flat (i.e. gradient is small), the proposal distribution $Q(x' \mid x_1)$ has a relatively small variance, so it will explore the sample space more slowly – that is, it will display a random walk behavior. Conversely, when the current sample is at x_2, where the contour of P is steep (i.e. gradient is large), the same proposal distribution $Q(x' \mid x_2)$ will now have a relatively large variance, so that many proposed samples will be rejected.

The two contrasting cases demonstrate that simply adjusting the variance of Q is not enough, because the same variance can be small in certain regions and large in others. When the variance is too small, the next sample is too close to the previous one. When the variance is too large, the next sample can more easily reach a low-density region in P, so it is more easily rejected. Either case leads to a lower effective sample size and slower convergence to the invariant distribution.

How do we get around this issue? One way is to make use of the gradient of P, as suggested above; this leads to Hamiltonian Monte Carlo (Section 2). Another way is to approximate P directly, for example using variational inference, and using the approximation as our proposal distribution (Section 3).

2 Hamiltonian Monte Carlo

Hamiltonian Monte Carlo, or Hybrid Monte Carlo, is a specialized Markov Chain Monte Carlo procedure which unites traditional Markov Chain Monte Carlo with molecular dynamics. It was originally proposed by [1], but [2] is widely credited with introducing it to the statistics and machine learning communities in the context of performing inference on Bayesian neural networks.

For a more thorough explanation of HMC and its variants than is provided here, see the classic survey paper [3], or the more recent[4].

2.1 Hamiltonian Dynamics

Before introducing HMC, it is necessary to provide some background on the molecular dynamics on which it is based: Hamiltonian dynamics. Note that it is not essential that one grasps the motivating physics to understand the components of the HMC algorithm. However, the basic physical concepts are useful in that they provide intuition.

The basics of Hamiltonian dynamics are as follows. Consider the physical state of an object. Let q and p denote the object's position and momentum, respectively. Note that each of these variables has the same dimension.

The Hamiltonian of the object is defined as

$$H(q, p) = U(q) + K(p)$$

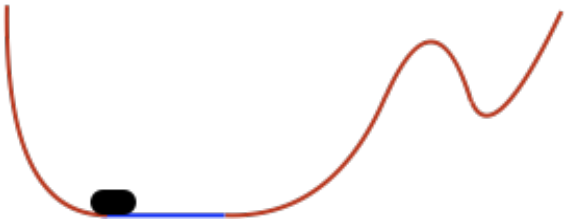
where U is the potential energy and K is the kinetic energy. Hamiltonian dynamics describe the nature by which the momentum and position change through time.

This movement is governed by the following system of differential equations called Hamilton's equations:

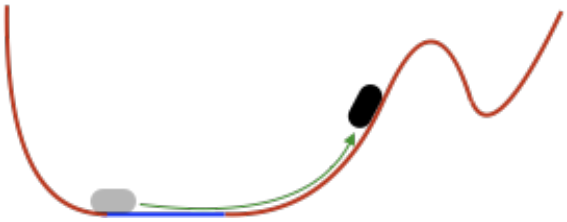
$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i} \tag{1}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i} \tag{2}$$

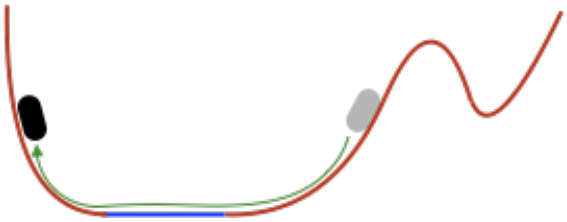
Illustration of the intuition Consider a frictionless puck lying on a surface whose height. Here, consider $U(q)$ as proportional to the height of the surface at position q. Also, let p denote the momentum of the puck. Let $K(p) = p^2/2$. Figure 2 illustrates the movement of the puck according to Hamiltonian dynamics.



(a) The initial placement of the puck



(b) Simulation of the Hamiltonian movement of the puck. The puck continues moving in the direction of the green arrow until $H(q, p) = U(q)$ and momentum is 0.



(c) After reaching maximum $U(q)$, the puck now reverses direction and continues until the same height is reached in the other direction.

Figure 2: An illustration of the intuition of Hamiltonian Dynamics. The puck is shown in black. The surface is shown in red and blue, with the blue showcasing an area where the surface is flat. The green represents the movement of the puck.

For a concrete example for which Hamilton's equations can be solved analytically, consider $U(q) = q^2/2$. Here, the solutions have the form $q(t) = r\cos(a + t)$ and $p(t) = -r\sin(a + t)$ for some a.

There are several important properties Hamiltonian dynamics which end up being useful in the context of Hamiltonian Monte Carlo. First, it is *reversible*, meaning that if the momentum of the body is reversed, it will retrace its previous movement. Also, the Hamiltonian is preserved: as the kinetic/potential energy

increases the potential/kinetic energy decreases accordingly. That is,

$$\frac{dH}{dt} = \sum_{i=1}^{d} \left[\frac{dq_i}{dt}\frac{\partial H}{\partial q_i} + \frac{dp_i}{dt}\frac{\partial H}{\partial p_i} \right] = \sum_{i=1}^{d} \left[\frac{\partial H}{\partial p_i}\frac{\partial H}{\partial q_i} + \frac{\partial H}{\partial q_i}\frac{\partial H}{\partial p_i} \right] = 0.$$

where d is the dimension of the space. This can be thought of as "conservation of energy".

Finally, volume is preserved under the mapping by Hamiltonian dynamics (Louisville's Theorem).

2.2 Numerically Simulating Hamiltonian Dynamics

In general, it is not possible to analytically solve Hamilton's equations as we did for the simple case above. Instead, it is common to discretize the simulation of the differential equations with some step size ε.

We briefly discuss two options here: Euler's method (performs poorly) and the leapfrog method (performs better).

Suppose that the momentum has the following expression (as is typical)

$$K(p) = \sum_{i=1}^{d} \frac{p_i^2}{2m_i}$$

Euler's method involves the following updates:

$$\begin{aligned} p_i(t+\varepsilon) &= p_i(t) + \varepsilon \frac{dp_i}{dt}(t) \\ &= p_i(t) - \varepsilon \frac{\partial U}{\partial q_i}(q(t)) \\ q_i(t+\varepsilon) &= q_i(t) + \varepsilon \frac{dq_i}{dt}(t) \\ &= q_i(t) + \varepsilon \frac{p_i(t)}{m_i} \end{aligned}$$

Unfortunately, Euler's method performs poorly. The result often diverges, meaning that the approximation error grows causing the Hamiltonian to no longer be preserved. Instead, the leapfrog method is used in practice.

The leapfrog method deals with this issue by only making a $\varepsilon/2$ step in p first, using that to update q, and then coming back to p for the remaining update. It consists of the following updates:

$$\begin{aligned} p_i(t+\varepsilon/2) &= p_i(t) - (\varepsilon/2)\frac{\partial U}{\partial q_i}(q(t)) \\ q_i(t+\varepsilon) &= q_i(t) + \varepsilon \frac{p_i(t+\varepsilon/2)}{m_i} \\ p_i(t+\varepsilon) &= p_i(t+\varepsilon/2) - (\varepsilon/2)\frac{\partial U}{\partial q_i}(q(t+\varepsilon)) \end{aligned}$$

The leapfrog approach diverges far less quickly than Euler's method. We now have the necessary tools to describe how to formulate a MCMC strategy using Hamiltonian dynamics.

2.3 MCMC using Hamiltonian dynamics

Hamiltonian Monte Carlo uses Hamiltonian dynamics to make proposals as part of an MCMC method. To do so, auxiliary "momentum" variables are introduced to create an auxiliary probability distribution as follows. Note that this auxiliary distribution admits the target distribution as a marginal.

Suppose that $P(q) \propto \pi(q)L(q|D)$ denotes our target density (the posterior of q given prior π and data D), with q denoting our variables of interest. Define the auxiliary distribution

$$\begin{aligned} P(q,p) &= P(q)\exp(-K(p)) \\ &= \frac{1}{Z}\exp\left(-U(q)/T\right)\exp(-K(p)) \end{aligned}$$

with

$$\begin{aligned} U(q) &= -\log\left[\pi(q)L(q|D)\right] \\ K(p) &= \sum_{i=1}^{d}\frac{p_i^2}{2m_i} \end{aligned}$$

Note that the auxiliary momentum variables are assumed to be independent gaussians.

2.4 Sampling Algorithm

We want to sample from $P(q,p)$. Note that because p is independent of q and has a tractable form (Gaussian) it is simple to perform a Gibbs step on p to update it.

To update q, we use Metropolis-Hastings with a Hamiltonian proposal. We first propose new (q^*, p^*) using Hamiltonian dynamics (enabled by discretization e.g. the leapfrog method). The MH ratio is just the ratio between the probability density of the new and old points, because the proposal is symmetric (at least in our case). That is:

$$A(q^*,p^*) = \frac{P(q^*,p^*)Q(q,p \mid q^*,p^*)}{P(q,p)Q(q^*,p^* \mid q,p)} = \frac{P(q^*,p^*)}{P(q,p)} = \exp\left(-H(q^*,p^*) + H(q,p)\right)$$

Note that this step jointly samples both p and q.

The full sampling algorithm written in R code provided by [3] is duplicated on the following page. Note that the leapfrog proposal consists of L leapfrog steps, and the momentum is negated at the end. This negation is to make the proposal reversible. The result is the cancellation of the Q terms as shown above. Since the Gaussian is symmetric, it does not affect the results of the algorithm and does not need to be performed in practice.

```
HMC = function (U, grad_U, epsilon, L, current_q)
{
q = current_q
p = rnorm(length(q),0,1) # independent standard normal variates
current_p = p
# Make a half step for momentum at the beginning
p = p - epsilon * grad_U(q) / 2
# Alternate full steps for position and momentum
for (i in 1:L)
{
# Make a full step for the position
q = q + epsilon * p
# Make a full step for the momentum, except at end of trajectory
if (i!=L) p = p - epsilon * grad_U(q)
}
# Make a half step for momentum at the end.
p = p - epsilon * grad_U(q) / 2
# Negate momentum at end of trajectory to make the proposal symmetric
p = -p
# Evaluate potential and kinetic energies at start and end of trajectory
current_U = U(current_q)
current_K = sum(current_p^2) / 2
proposed_U = U(q)
proposed_K = sum(p^2) / 2
# Accept or reject the state at end of trajectory, returning either
# the position at the end of the trajectory or the initial position
if (runif(1) < exp(current_U-proposed_U+current_K-proposed_K))
{
return (q) # accept
}
else
{
return (current_q) # reject
}
}
```

2.5 HMC is MCMC

The resultant HMC algorithm:

1. satisfies the detailed balance condition
2. can make far-off proposals with high acceptance rate
3. leaves the target distribution invariant

These properties all follow from the properties of Hamiltonian dynamics. Specifically, (1) follows from reversibility, (2) follows from preservation of the Hamiltonian, and (3) follors from preservation of volume, respectively.

2.6 Limitations

HMC has some limitations. First, it only applies from continuous variables with differentiable densities. If some of the variables are discrete or have non-differentiable densities, then we can use HMC moves on only a subset of the variables.

In addition, tuning the step size ε and the number of steps L can be difficult. For more sophisticated techniques that avoid this issue, see the discussion of the NUTS sampler in[4].

2.7 More on Discretization

Each step of the leapfrog method involves computing the gradient, which can be costly. For this reason, it can be useful to avoid this step. Here, we briefly describe some approaches.

One way of trying to avoid repeatedly performing this computation is to use the Langevin dynamics version of HMC, which amounts to the leapfrog approach with $L = 1$.

In practice, Langevin dynamics often allows one to skip the acceptance-checking step by ensuring that the Hamiltonian does not change much, in which case updating p can be ignored completely as well. Since p is updated through Gibbs and is not the variable we are targeting, there is no need to save the updated momentums.

Stochastic Langevin dynamics involves Langevin dynamics with a stochastic gradient (only a subset of the data is used). Here, if the an appropriate additional noise term is included in the updates, the correct invariant distribution is targeted asymptotically.

3 Combining Variational Inference with MCMC

As we described earlier, the quality of the proposal distribution directly affects the convergence and mixing properties of MCMC. An alternative to using HMC is to use variational inference to obtain a good proposal distribution for Metropolis-Hastings. This method is referred to as variational MCMC [5].

3.1 Recap of Variational Inference

Assuming that data x came from a complex model p with latent variables z and parameters θ, variational inference approximates $p(z \mid x, \theta)$ with a tractable variational distribution $q(z \mid \lambda)$ with variational parameters λ. Jensen's inequality provides the evidence lower bound (ELBO) to the logarithm of the marginal likelihood:

$$\log p(x \mid \theta) \geq \mathbb{E}_q \left[\log p(x, z \mid \theta)\right] - \mathbb{E}_q \left[\log q(z \mid \lambda)\right]$$

When even evaluating $p(x \mid \theta)$ is intractable, we can further introduce another set of variational parameters ξ to p. We can then obtain an estimate P^{est} of the true posterior $p(\theta \mid x)$ such that

$$p(\theta \mid x) \geq P^{est}(\theta \mid x, \lambda, \xi)$$

3.2 Variational MCMC

Now consider running the Metropolis-Hastings algorithm to sample from the posterior distribution $p(\theta \mid x)$. We can define the proposal distribution to be

$$Q(\theta' \mid \theta) = P^{est}(\theta' \mid x, \lambda, \xi)$$

From our previous discussion, we can expect that the algorithm will work well if P^{est} is sufficiently close to p. However, this is rarely the case in high dimensions, because of correlations introduced by higher-order moments that are not captured by simpler variational approximations such as mean-field distributions. Thus, the algorithm still has low acceptance rates in high dimensions.

To resolve this issue, we can design the proposal in blocks of parameters such that the variational approximation models higher-order moments. This poses a trade-off between the block size and the quality of our samples. Alternatively, we can use a mixture of random walk and variational proposals as our proposal distribution, while making sure that we can efficiently compute the acceptance ratio involving the mixture. In experiments, it can be shown that these modifications can improve the mixing rate of the Metropolis-hastings sampler [5].

3.3 Bridging the Gap

For large-scale problems, we can incorporate some of the recently introduced stochastic methods for variational inference [6, 7, 8, 9]. Using stochastic variational methods to estimate the proposal distribution P^{est} can lead to gradient-based Monte Carlo methods or Hamiltonian variational inference (HVI) [10], which combines stochastic variational methods with Hamiltonian Monte Carlo.

4 Conclusions

Hamiltonian Monte Carlo, variational Markov Chain Monte Carlo, and their variants all attempt to choose a good proposal distribution for a complex and high-dimensional distribution. A good proposal distribution can improve convergence rates, mixing time, and acceptance rates, but most importantly, it allows the sampler to yield uncorrelated samples within a reasonable amount of computations.

References

[1] Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics letters B*, 195(2):216–222, 1987.

[2] Radford M Neal. Bayesian training of backpropagation networks by the hybrid monte carlo method. Technical report, Citeseer, 1992.

[3] Radford M Neal et al. Mcmc using hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2:113–162, 2011.

[4] Michael Betancourt. A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*, 2017.

[5] Nando De Freitas, Pedro Højen-Sørensen, Michael I Jordan, and Stuart Russell. Variational mcmc. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 120–127. Morgan Kaufmann Publishers Inc., 2001.

[6] Tim Salimans, David A Knowles, et al. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

[7] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, number 2014, 2013.

[8] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages II–1278–II–1286. JMLR.org, 2014.

[9] J Paisley, David M Blei, and Michael I Jordan. Variational bayesian inference with stochastic search. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 1367–1374, 2012.

[10] Tim Salimans, Diederik P Kingma, and Max Welling. Markov chain monte carlo and variational inference: Bridging the gap. In *ICML*, volume 37, pages 1218–1226, 2015.