Stat260: Bayesian Modeling and Inference **Lecture Date: April 5, 2010**

Monte Carlo Sampling

Lecturer: Michael I. Jordan *Scribe: Sagar Jain*

1 Monte Carlo Sampling

Monte Carlo sampling is often used in two kinds of related problems.

- Sampling from a distribution $p(x)$, often a posterior distribution.
- Computing approximate integrals of the form $\int f(x)p(x)\,dx$ i.e., computing expectation of $f(x)$ using density $p(x)$.

The above problems are related because if we can sample from $p(x)$ then, we can also solve the problem of computing integrals.

Suppose $\{x^{(i)}\}$ is an i.i.d. random sample drawn from $p(x)$. Then, the *strong law of large lumbers* (SLLN) says:

$$\frac{1}{N}\sum_{i=1}^{N} f(x^{(i)}) \overset{a.s}{\rightarrow} \int f(x)p(x)\,dx$$

Moreover, the rate of convergence is proportional to $\sqrt{N}$. However, the proportionality constant increases exponentially with the dimension of the integral.

We will now describe various sampling algorithms when direct sampling from a density $p(x)$ is not possible.

2 Rejection Sampling

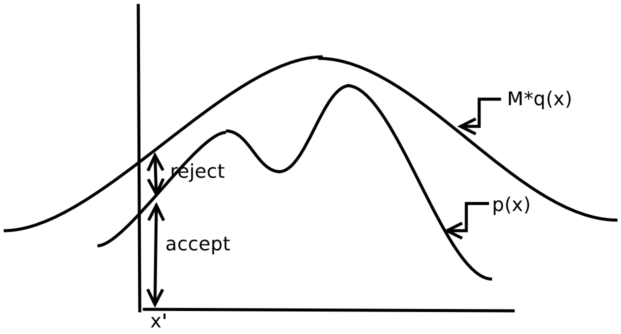

Figure 1: Rejection Sampling

Suppose we want to sample from the density $p(x)$ as shown in Figure 1. If we can sample uniformly from the 2-D region under the curve, then this process is same as sampling from $p(x)$. In rejection sampling, another density $q(x)$ is considered from which we can sample directly under the restriction that $p(x) < Mq(x)$ where $M > 1$ is an appropriate bound on $\frac{p(x)}{q(x)}$. The rejection sampling algorithm is described below.

```
$i \leftarrow 0$
while $i \neq N$ do
    $x^{(i)} \sim q(x)$
    $u \sim U(0,1)$
    if $u < \frac{p(x^{(i)}}{Mq(x^{(i)})}$ then
        accept $x^{(i)}$
        $i \leftarrow i + 1$
    else
        reject $x^{(i)}$
    end if
end while
```

Informally, all this process does is samples $x^{(i)}$ from some distribution and then it decides whether to accept it or reject. The main problem with this process is that M is generally large in high-dimensional spaces and since p(*accept*) $\propto \frac{1}{M}$, many samples will get rejected.

3 Adaptive Rejection Sampling

This method works only for log concave densities. The basic idea is to form an upper envelope (the upper bound on $p(x)$) adaptively and use this in place of $Mq(x)$ in rejection sampling.

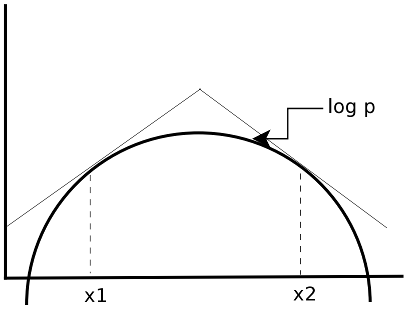

Figure 2: Adaptive Rejection Sampling

As shown in Figure 2, the log density $\log p(x)$ is considered. $x^{(i)}$ is then sampled from the upper envelope, and either accepted or rejected as in rejection sampling. If it is rejected, a tangent is drawn passing through $x = x^{(i)}$ and $y = log(p)$ and used to reduce the upper envelope to reduce the number of rejected samples. The intersection of these tangent planes enable the formation of envelope adaptively. To sample from the upper envelope, we need to transform from log space by exponentiating and using properties of the exponential distribution.

One problem with this approach is that it involves painful book-keeping to find the intersection of hyperplanes and usually this approach does not work for high dimensions.

4 Importance Sampling

Our goal is to compute $I(f) = \int f(x)p(x)\,dx$.

If we have a density $q(x)$ which is easy to sample from, we can sample $x^{(i)} \overset{iid}{\sim} q(x)$. Define the importance weight as:

$$w(x^{(i)}) = \frac{p(x^{(i)})}{q(x^{(i)})}$$

Consider the weighted Monte Carlo sum:

$$\begin{aligned}
\frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})w(x^{(i)}) &= \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})\frac{p(x^{(i)})}{q(x^{(i)})} \\
&\overset{a.s.}{\rightarrow} \int \left(f(x)\frac{p(x)}{q(x)} \right) q(x)\,dx \quad \text{(Law of Large Numbers)} \\
&= \int f(x)p(x)\,dx
\end{aligned}$$

In principle, we can sample from any distribution $q(x)$. In practice, we would like to choose $q(x)$ as close as possible to $|f(x)|w(x)$ to reduce the variance of our estimator (more on this below).

Remark 1. We do not need need to know the normalization constants for $p(x)$ and $q(x)$. Since w is $\frac{p}{q}$, we can compute

$$\int f(x)p(x)\,dx \quad \approx \quad \frac{\sum_{i=1}^{N} f(x^{(i)})w(x^{(i)})}{\sum_{i=1}^{N} w(x^{(i)})} \tag{1}$$

where because of the ratio, the normalizing constants will cancel.

Thus, we can now compute integrals using importance sampling. However, we do not directly get samples from $p(x)$. To get samples from $p(x)$, we must sample from the weighted sample from our importance sampler. This process is called *Sampling Importance Re-sampling* (SIR), which is described later in Section 5.

Now, coming back to the question of how to pick $q(x)$. Ideally, we would like to pick $q(x)$ such that the variance of $f(x)w(x)$ is minimum.

$$\begin{aligned}
\texttt{Var}_{q(x)} f(x)w(x) &= \mathbb{E}_{q(x)} f(x)^2 w(x)^2 - I(f)^2 \\
\mathbb{E}_{q(x)} f(x)^2 w(x)^2 &\geq \left(\mathbb{E}_{q(x)}|f(x)|w(x)\right)^2 \quad \text{(By Jensen's Inequality for concave functions)} \\
&= \left(\int |f(x)|p(x)\,dx\right)^2
\end{aligned}$$

The term $I(f)^2$ is independent of q. So, the best $q^*(x)$ which makes the variance minimum is given by:

$$q^*(x) = \frac{|f(x)|p(x)}{\int |f(x)|p(x)\,dx}$$

One problem with picking $q(x)$ is that since $p(x)$ was hard to sample from thus, $|f(x)|p(x)$ usually would also be hard to sample from.

5 Sampling Importance Re-sampling (SIR)

SIR is simply sampling

$$x^{(i)*} \sim \hat{p}_N(x) \equiv \frac{1}{N}\sum_{i=1}^{N} w(x^{(i)})\delta_{x^{(i)}}(x)$$

This generates a sample from $p(x)$ and is really just sampling with replacement from collection $\{x^{(i)}\}$ with probability proportional to normalized weights.

$$\hat{\hat{p}}_N(x) = \frac{1}{N}\sum_{i=1}^{N} \delta_{x^{(i)*}}(x)$$

Statistically, there is no advantage in working with SIR because of the introduction of variance again in re-sampling.

6 Adaptive Importance Sampling (AIS)

Now the importance function q is parameterized with λ such that $q \in Q = \{q_\lambda; \lambda \in \Lambda\}$. AIS starts with a rough guess of λ and importance sampling is run in an iterative way and λ is continually updated.

Derivative of variance term *w.r.t.* λ is given by:

$$D(\lambda) = \mathbb{E}_{q(x,\lambda)}\left(2f(x)^2 w(x,\lambda)\frac{\partial w(x,\lambda)}{\partial \lambda}\right)$$

Thus, the update rule for λ is given by:

$$\lambda_{t+1} = \lambda_t - \alpha\frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})^2 w(x^{(i)},\lambda_t)\frac{\partial w(x^{(i)},\lambda_t)}{\partial \lambda}$$

There are a couple of issues with AIS:

- λ is moving and $x^{(i)}$ is sampled from $q(x,\lambda)$.
- Can be problematic if the step size is too small or too large.

7 Metropolis-Hastings

In Metropolis-Hastings sampling, samples mostly move towards higher density regions, but sometimes also move downhill. In comparison to rejection sampling where we always throw away the rejected samples, here we sometimes keep those samples as well. Its pseudo-code is given below.

```
Init $x^{(0)}$
for $i = 0$ to $N - 1$ do
    $u \sim U(0,1)$
    $x^* \sim q(x^*|x^{(i)})$
```

```
5:      if $u < min\left\{1, \frac{p(x^*)q(x^{(i)}|x^*)}{p(x^{(i)})q(x^*|x^{(i)})}\right\}$ then
6:          $x^{(i+1)} \leftarrow x^*$
7:      else
8:          $x^{(i+1)} \leftarrow x^{(i)}$
9:      end if
10: end for
```

Remark 2. In line 5 of the algorithm, if q is symmetric then $\frac{q(x^{(i)}|x^*)}{q(x^*|x^{(i)})} = 1$. This term was later introduced to the original Metropolis algorithm by Hastings.