Markov Chains and Algorithmic Applications: WEEKS 8&9

1 Sampling

1.1 Introduction

In this lecture we are interested in finding good sampling techniques to obtain samples from a probability distribution. In other words, given a probability distribution π on S, how can we pick a random $i \in S$ such that $\mathbb{P}(i) = \pi_i$?

But why would we want to do this ?

Example 1.1 (Monte Carlo Integration)**.** Suppose we want to compute $\mathbb{E}(f(X))$, with $X \sim \pi$ (i.e. $\mathbb{P}(X = i) = \pi_i$, $i \in S$). By the definition of expectation we have

$$\mathbb{E}(f(X)) = \sum_{i \in S} f(i)\pi_i \tag{1}$$

Depending on the set S, the above expression can be too expensive to compute exactly (i.e. computing it requires exponential time in $|S|$).

Instead of evaluating (1), we can compute the following approximation: take M i.i.d. samples $X_1, \ldots, X_M$ from distribution π and compute

$$\frac{1}{M} \sum_{k=1}^{M} f(X_k) \tag{2}$$

Given some conditions on $f(x)$, the law of large numbers guarantees

$$\frac{1}{M} \sum_{k=1}^{M} f(X_k) \underset{M \to \infty}{\longrightarrow} \mathbb{E}(f(X)) \text{ almost surely}$$

But how big should M be for the approximation to be good ? The variance of (2) is given by

$$\mathrm{Var}\left(\frac{1}{M} \sum_{k=1}^{M} f(X_k) \right) = \frac{1}{M} \mathrm{Var}(f(X_1)) = \mathcal{O}\left(\frac{1}{M} \right)$$

so $\frac{1}{M} \sum_{k=1}^{M} f(X_k) \approx \mathbb{E}(f(X)) \pm \frac{C}{\sqrt{M}}$. We see that a good approximation requires taking M quite large.

A "simple" way to obtain samples is as follows:

Example 1.2 ("Simple" Sampling)**.** Let X be a π-distributed random variable on $S = \mathbb{N}$. If we can generate a continuous $\mathcal{U}(0,1)$ random variable U, then we decide

$$X = \begin{cases} 0 & 0 \leq U \leq \pi_0, \\ 1 & \pi_0 < U \leq \pi_0 + \pi_1, \\ \quad \vdots & \\ i & \sum_{j=0}^{i-1} \pi_j < U \leq \sum_{j=0}^{i} \pi_j \\ \quad \vdots & \end{cases}$$

Hence $\mathbb{P}(X = i) = \pi_i$.

As simple as the above sampling scheme seems, terms of the form $\sum_{j=0}^{i} \pi_j$ (cdf of X) can be difficult to compute because we need to know each term π_j exactly: for π_j of the form $\frac{h(j)}{Z}$, the normalization constant $Z = \sum_{j \in S} h(j)$ can be non-trivial to compute depending on S, as we will see below.

For the rest of the lecture, we will detail alternative sampling methods to try to side-step the issues above.

1.2 Importance Sampling

Consider again the Monte Carlo integration problem given above: our aim here is to find a better estimate of $\mathbb{E}(f(X))$.

For this purpose, take another distribution $\psi = (\psi_i,\, i \in S)$ from which we know how to sample and let us define the coefficients $w_i = \frac{\pi_i}{\psi_i}$. Then

$$\mathbb{E}(f(X)) = \sum_{i \in S} f(i)\pi_i = \sum_{i \in S} f(i) w_i \psi_i = \mathbb{E}(f(Y)w(Y))$$

with $Y \sim \psi$. Since we know how to sample from ψ, we can approximate $\mathbb{E}(f(Y)w(Y))$ by choosing M i.i.d. samples $Y_1, \ldots, Y_M$ from ψ and computing $\frac{1}{M}\sum_{k=1}^{M} f(Y_k)w(Y_k)$. We then have

$$\mathrm{Var}\left(\frac{1}{M}\sum_{k=1}^{M} f(Y_k)w(Y_k)\right) = \frac{1}{M}\mathrm{Var}\left(f(Y_1)w(Y_1)\right)$$

As we did not assume anything in particular about the distribution ψ, we can choose it so as to *minimize* the variance of $f(Y_1)w(Y_1)$, which improves the approximation of the expectation (but note that the order in M remains the same).

Remark 1.3. Why is this method called *importance sampling* ? It turns out that the distribution ψ minimizing the above variance puts more weight than π itself on the states i with a large probability π_i, and less weight on those with a small probability π_i: only the "important" states are therefore sampled with this method.

1.3 Rejection Sampling

Consider yet again the Monte Carlo integration problem (i.e. for $X \sim \pi$, compute $\mathbb{E}(f(X))$), but assume now that we are unable to sample directly from π (essentially because of the computation cost of this operation).

The idea behind rejection sampling is the following:

1. Take a distribution ψ on S from which samples can be easily produced (e.g. take ψ uniform).
2. Take a sample X from ψ.
3. Accept X with some probability, or reject it with the complement probability.

Formally, let $\psi = (\psi_i,\, i \in S)$ be a distribution from which we can sample and define weights $\widetilde{w}_i = \dfrac{1}{c}\dfrac{\pi_i}{\psi_i}$ with $c = \max_{i \in S} \dfrac{\pi_i}{\psi_i}$ (≥ 1). The weights $\widetilde{w}_i$ play the role here of acceptance probabilities. Then

$$\mathbb{P}(X = i) = \psi_i \widetilde{w}_i = \frac{\pi_i}{c}$$

$$\mathbb{P}(X \text{ is rejected}) = 1 - \sum_{i \in S} \mathbb{P}(X = i) = 1 - \sum_{i \in S} \frac{\pi_i}{c} = 1 - \frac{1}{c}$$

We therefore have

$$\mathbb{E}(f(X)) \approx \frac{1}{M'} \sum_{k=1:X_k \text{ accepted}}^{M} f(X_k)$$

where M' is the number of accepted samples among the $X_1, \ldots, X_M$.

The disadvantage of rejection sampling is that it may end up requiring much more samples than needed due to the sample rejection process (especially when the distance between π and ψ is large, i.e. when c is large).

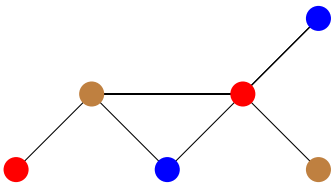
1.4 Markov Chain Monte Carlo (MCMC) Sampling

The idea behind the MCMC method to obtain samples of a distribution π on S is to construct a Markov chain on S with transition matrix P having π as its stationary distribution. The samples of π are then obtained by iterating P long enough to reach the stationary distribution π, then sampling among the states of the Markov chain. The advantage here is that a) we do not have to sample directly from π, and b) we do not even need to know everything about π, as we will see below.

For practical reasons, we want P to have certain properties:

1. π should be the unique limiting distribution of P.
2. Convergence to the stationary distribution π should be fast, so as to obtain samples within a reasonable amount of time.

Example 1.4 (Graph Coloring)**.** Let $G = (V, E)$ be a graph with vertex set V and edge set E. We want to color each vertex of the graph with one of the q colors at our disposal such that a vertex's color differs from that of all its neighbors, as seen below:



More formally, let $x = (x_v,\ v \in V)$ be a particular color configuration of the vertex set V. A *proper q-coloring* of G is any configuration x such that $\forall v, w \in V$, if $(v, w) \in E$ then $x_v \neq x_w$.

If S represents the set of all possible color configurations, then the uniform distribution π over all proper q-colorings is given by

$$\pi(x) = \frac{1}{Z}\,\mathbb{1}\{x \text{ is a proper } q\text{-coloring}\}$$

where Z is the total number of proper q-colorings in G.

Computing Z would require enumerating all possible proper q-colorings which is non-trivial depending on G. Still, we would like to sample from π without computing Z explicitly.

1.4.1 Metropolis-Hastings Algorithm

The Metropolis-Hastings algorithm is a procedure to construct a Markov chain on S having as limiting distribution π (for convenience, we assume that $\pi_i > 0$ for all $i \in S$). Here is the algorithm:

1. Select an easy-to-simulate irreducible and aperiodic Markov chain ψ on S with the constraint that $\psi_{ij} > 0$ if and only if $\psi_{ji} > 0$.[1] We call ψ the *base chain*.
2. Design acceptance probabilities $a_{ij} = \mathbb{P}(\text{transition from } i \text{ to } j \text{ is accepted})$ such that the matrix P given below has limiting distribution π.
3. Construct the matrix P as such:

$$\begin{cases} p_{ij} &= \psi_{ij}\, a_{ij}, \quad j \neq i \\ p_{ii} &= \psi_{ii} + \sum_{k \neq i} \psi_{ik}\,(1 - a_{ik}) = 1 - \sum_{k \neq i} \psi_{ik} a_{ik} \end{cases}$$

   In other words, we are adding self-loops of different weights to each state.

[1] If S is finite, then these conditions imply positive-recurrence, hence ψ is ergodic and has a unique limiting distribution, but this limiting distribution is of no interest to the algorithm.

We must now choose the weights a_{ij} so that $p_{ij}(n) \underset{n\to\infty}{\longrightarrow} \pi_j$. Moreover, we were able to upper-bound the mixing time of chains satisfying detailed balance in the previous lectures, so we would like P to satisfy this condition too: $\pi_i p_{ij} = \pi_j p_{ji}$

Theorem 1.5 (Metropolis-Hastings)**.** If $a_{ij} = \min\left(1, \frac{\pi_j \psi_{ji}}{\pi_i \psi_{ij}}\right)$, then the matrix P constructed above is ergodic with stationary distribution π. Moreover, P satisfies detailed balance.

Proof. By assumption, ψ is irreducible and aperiodic, and $\forall i, j \in S, \psi_{ij} > 0$ iff $\psi_{ji} > 0$. So if $\psi_{ij} > 0$, then $a_{ij} > 0$ and $p_{ij} > 0$ also. Therefore, P is also irreducible and aperiodic. We then have

$$\pi_i p_{ij} = \pi_i \psi_{ij} a_{ij} = \pi_i \psi_{ij} \min\left(1, \frac{\pi_j \psi_{ji}}{\pi_i \psi_{ij}}\right) = \min\left(\pi_i \psi_{ij}, \pi_j \psi_{ji}\right)$$

whose expression is symmetric in i, j. It is therefore also equal to $\pi_j\, p_{ji}$: detailed balance holds and P has π as stationary distribution.

Finally, since P is irreducible and has a stationary distribution π, then by a previously seen theorem, P must be positive-recurrent and pi msut be unique. therefore P is ergodic and π is also a limiting distribution. □

Remark 1.6. If $\psi_{ij} = \psi_{ji}$, then the expression for a_{ij} simplifies to $a_{ij} = \min\left(1, \frac{\pi_j}{\pi_i}\right)$.
The intuition behind choosing a_{ij} as such is the following: if $\pi_j > \pi_i$ the transition $i \to j$ should be taken with probability 1 since the chain is heading towards the more probable state j. However if $\pi_j < \pi_i$, then the move $i \to j$ should be taken with probability $\frac{\pi_j}{\pi_i} < 1$. In other words, the chain should tend towards the states having high probability, but it should be able to return to less probable states in order not to get stuck in a state that locally maximizes π.

Remark 1.7. The advantage of the Metropolis-Hastings algorithm is that the acceptance probabilities a_{ij} depend on π only through the ratios $\frac{\pi_j}{\pi_i}$, which can be significantly easier to compute than π_i and π_j separately! In the graph coloring example given previously, $\frac{\pi_j}{\pi_i} = \frac{\mathbb{1}\{j \text{ is a proper } q\text{-coloring}\}}{\mathbb{1}\{i \text{ is a proper } q\text{-coloring}\}}$, so we can avoid computing the expensive normalization constant Z entirely.

Example 1.8 (Metropolized Independent Sampling)**.** To obtain samples of distribution π on S, we choose the base chain ψ such that $\psi_{ij} = \psi_j > 0\ \forall i, j \in S$ (i.e. the process realizations are just sequences of i.i.d. random variables).

The acceptance probabilities are $a_{ij} = \min\left(1, \frac{w_j}{w_i}\right)$ with $w_i = \frac{\pi_i}{\psi_i}$, so the transition probabilities of P are given by

$$\begin{cases} p_{ij} & = \psi_{ij} a_{ij} = \psi_j \min\left(1, \frac{w_j}{w_i}\right), \quad j \neq i \\ p_{ii} & = 1 - \sum_{k\neq i} \psi_{ik} a_{ik} = 1 - \sum_{k\neq i} \psi_k \min\left(1, \frac{w_k}{w_i}\right) \end{cases}$$

In this particular example, one can show the following (no proof given here):

Theorem 1.9 (Liu)**.** Let $\lambda_0 \geq \lambda_1 \geq \ldots \geq \lambda_{N-1}$ be the eigenvalues of P, and $\lambda_* = \max(\lambda_1, -\lambda_{N-1})$. Then

$$\lambda_* = 1 - \frac{1}{w_*}, \quad \text{where } w_* = \max_{i\in S} \frac{\pi_i}{\psi_i} > 1$$

Correspondingly, the spectral gap $\gamma = \frac{1}{w_*}$.

From the above and the previous lectures, we find that

$$\|P_i^n - \pi\|_{\text{TV}} \leq \frac{\lambda_*^n}{2\sqrt{\pi_i}} \leq \frac{1}{2\sqrt{\pi_i}} e^{-\gamma n} = \frac{1}{2\sqrt{\pi_i}} e^{-\frac{n}{w_*}}$$

Therefore, if w_* is large (i.e. if the distance between π and ψ is large), then convergence to the stationary distribution π is slow (this resembles the situation we already encountered with rejection sampling).