10-707: Advanced Deep Learning **Spring 2020**

Lecture 12: February 26

Lecturer: Andrej Risteski *Scribes: Shantanu Gupta, Vishwak Srinivasan, Yufeng Shen*

Note: *LaTeX template courtesy of UC Berkeley EECS dept.*

This lecture's notes illustrate some uses of various LaTeX macros. Take a look at this and imitate.

12.1 Common Random Walk over Discrete Domains

Our goal is that we have a distribution given up to constant of proportionality and we want to produce samples from it. And the idea behind the random walk is to explore the domain of distribution via random, simple local moves. If we make enough of random moves, the random process will converge to produced samples coming from the distribution given up to the constant of proportionality.

12.1.1 Several Definitions

Definition 12.1 *A set of random variables $(X_1, X_2, \cdots, X_T)$ is **Markov** if $\forall t : P(X_t|X_{<t}) = P(X_t|X_{t-1})$ It is **homogeneous** if $P(X_t|X_{t-1})$ does not depend on t.*

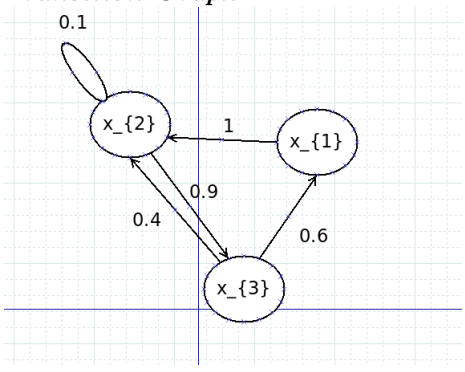
Definition 12.2 *A **homogeneous** Markov process on a discrete domain $\mathcal{X}$ by a **transition matrix** $T \in \mathbb{R}_+^{|\mathcal{X}|\times|\mathcal{X}|} : T_{ij} = P(X_{t+1} = j|X_t = i), \forall i, \sum_j T_{ij} = 1$. We also call such process **A Markov Chain/Markov Random Walk**.*

Example 12.3 *Markov chain with three states($s = 3$)*

Transition Matrix

$$T = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0.1 & 0.9 \\ 0.6 & 0.4 & 0 \end{pmatrix}$$

Transition Graph



Definition 12.4 *Stationary distribution: a distribution $\pi = (\pi_1, \cdots, \pi_{|\mathcal{X}|})$ is stationary for a Markov walk if $\pi_T = \pi$*

Remark 12.5 *1. Stationary distribution need not to be unique;*

2. Many Markov Chains have unique stationary distributions.(After taking many steps, starting with any distributions,we get to the same distribution)

12.1.2 Unique Stationary Distribution

If we wish to sample from some π, design a Markov chain which has π as stationary distribution. And there are some conditions for having a unique stationary distribution.

1. Transition graph are connected(**Irreducibility**): there is a path that transitions from any state to any other.
2. No cycles in graph(**Aperiodic**): random walk does not get trapped in cycles

Theorem 12.6 *For any irreducible + aperiodic Markov chain there is a unique π such that*

$$\forall p_0, \lim_{t\to\infty} p_0 T^t = \pi \tag{12.1}$$

Theorem 12.7 *Useful sufficient condition for π to be a stationary distribution called detailed balance as*

$$\pi_i T_{ij} = \pi_j T_{ji}, \forall (i,j) \tag{12.2}$$

Proof:

$$(\pi T)_i = \sum_j \pi_j T_{ji} = \sum_j \pi_i T_{ij} \tag{12.3}$$

$$= \pi_i \sum_j T_{ij} \tag{12.4}$$

$$= \pi_i \tag{12.5}$$

■

12.1.3 Common ways to set up random walk

12.1.3.1 Metropolis-Hastings

Suppose we are trying to sample from π defined over a domain of size m(large), up to a constant of proportionality:

$$\pi_i = \frac{b(i)}{Z}, Z = \sum_{i=1}^{m} b(i)$$

Suppose we have the transition kernel $q(i,j)$ to produce samples, and consider the random walk where we have $\alpha i, j$ and follow the rules as
$P(X_n = j | X_{n-1} = i) =$

1. from state i go to state j with prob $q(i,j)$;
2. with prob $1 - \alpha(i,j)$ go back to state i and with prob $\alpha(i,j)$ stay in state j

Thus we have

$$P(X_{n+1} = j|X_n = i) = q(i,j)\alpha(i,j), \forall j \neq i \tag{12.6}$$
$$P(X_{n+1} = i|X_n = i) = q(i,i) + \sum_{k \neq i} q(i,k)(1 - \alpha(i,k)) \tag{12.7}$$

Theorem 12.8

$$\pi_i P_{ij} = \pi_j P_{ji}, \forall j \neq i \iff \pi_i q(i,j)\alpha(i,j) = \pi_j q(j,i)\alpha(j,i), \forall j \neq i \tag{12.8}$$

Proof:

$$P_{ij} = P(X_{n+1} = j|X_n = i) = q(i,j)\alpha(i,j), \forall j \neq i \tag{12.9}$$

■

Theorem 12.9

$$\textit{If } \alpha(i,j) = \min(\frac{\pi_j q(j,i)}{\pi_i q(i,j)}, 1) = \min(\frac{b_j q(j,i)}{b_i q(i,j)}, 1) \tag{12.10}$$
$$\implies (\pi_1, \cdots, \pi_m) \textit{ stationary distribution} \tag{12.11}$$

Proof:

$$\text{If } \alpha(i,j) = \frac{\pi_j q(j,i)}{\pi_i q(i,j)} \iff \alpha(i,j) = 1 \tag{12.12}$$

Thus detailed balance holds. ■

12.1.3.2 Gibbs Sampling

Consider sampling a distribution over n variables $\mathbf{x} = (x_1, \cdots, x_n)$, such that each of conditional distribution $P(x_i|\mathbf{x}_{-i})$ is easy to sample. And here are the steps for Gibbs sampling.

1. Let current state be $\mathbf{x} = (x_1, \cdots, x_n)$;
2. Pick $i \in [n]$ uniformly at random;
3. Sample $x \sim P(X_i = x|\mathbf{x}_{-i})$;
4. Update state to $\mathbf{y} = (x_1, \cdots, x_{i-1}, x, x_{i+1}, \cdots, x_n)$

This can be considered as a special Metropolis-Hastings with appropriate kernel as

$$q(\mathbf{x}, \mathbf{y}) = \frac{1}{n} P(X_i = x|X_j = x_j, \forall j \neq i) \tag{12.13}$$
$$= \frac{1}{n} \frac{P(\mathbf{y})}{P(X_j = x_j, \forall j \neq i)} \tag{12.14}$$

where $\mathbf{x} = (x_1, ..., x_n)$ and $\mathbf{y} = (x_1, ..., x_{i-1}, x, x_{i+1}, ..., x_n)$.
And in this case, $\alpha(i,j)$ will be

$$\frac{p(\mathbf{y})q(\mathbf{y},\mathbf{x})}{p(\mathbf{x})q(\mathbf{x},\mathbf{y})} = \frac{p(\mathbf{y})\frac{1}{n}\frac{P(\mathbf{x})}{P(Y_j = y_j, \forall j \neq i)}}{p(\mathbf{x})\frac{1}{n}\frac{P(\mathbf{y})}{P(X_j = x_j, \forall j \neq i)}} = \frac{p(\mathbf{y})p(\mathbf{x})}{p(\mathbf{x})p(\mathbf{y})} = 1 \tag{12.15}$$

12.2 What governs the "mixing time" of a random walk?

So far, we were concerned about designing Markov Chains that have a stationary distribution. That is, we wanted a chain such that $\forall p_0,\ \lim_{t\to\infty} p_0 T^t = \pi$.

Now, we turn our attention to a different question. In practice, we want to run the random walk for some finite number of steps t such that $\forall p_0,\ p_0 T^t \approx \pi$. Formally, this is known as the "*mixing time*".

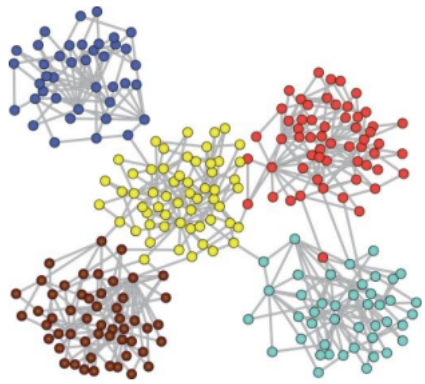
12.2.1 Conductance

One tool that we can use to analyze the mixing time of a general transition matrix T is *conductance*. Informally, this metric quantifies the fact that the transition graph does not have *bottlenecks*, that is, the graph does not contain regions that are difficult to leave.

Definition 12.10 *The conductance of a subset S is defined as*

$$\phi(S) = \frac{\sum_{i\in S, j\notin S} T_{ij}}{\sum_{i\in S} \pi_i}.$$

So the conductance is the probability of leaving a given set S normalized by the probability that we place on this set S under the stationary distribution π. In other words, it measures how easy it is to leave S given that we started in S.



As an example, the colored sets in the figure on the left have poor conductance. That is, if we started out in one of the colored regions, we would take a long time to leave them and reach a region with a different color.

- Implications of poor conductance – If $\phi(S)$ is small, then the mixing time will be large. That is, if we start in S, it will take us a long time to leave S *even if* we have the correct stationary distribution π. In this case, we say that the distribution is **multimodal**, that is, the distribution has several *poorly connected* regions of *high probability mass*. The poor connectivity implies that it is difficult to transition amongst the modes.
- Implications of good conductance – If $\phi(S)$ is large for all S, then we can claim that the mixing time is small.

12.2.2 A misconception about random walks

A common misconception about random walks is that for a random walk to mix well, it must touch every state in the domain. This is not true. If it were, this would defeat the purpose of running a Marvok Chain

as we could instead just enumerate every possible configuration in our domain. This would be equivalent of computing the partition function directly in a brute force manner.

However, it is true that with a reasonable probability, some sets of moves in the random walk should allow us to get anywhere in the domain for good mixing.

12.3 Random Walks over Continuous domain a.k.a. Langevin Dynamics

Here, we are interested in sampling from distributions of the form:

$$p(x) \propto \exp(-f(x)) \tag{12.16}$$

$f(x)$ is a continuous function from which we can extract gradients. For instance, consider $f(x)$ to be parameterized by a neural network.

A natural algorithm / random walk to sample from such a distribution is given by the following GD-like recursion:

$$x_{t+1} = x_t - \eta \nabla f(x_t) + \sqrt{2\eta}\xi_k \tag{12.17}$$

where $\xi_k \sim \mathcal{N}(0, 1)$. When $\eta \to 0$, we get what is known as a stochastic differential equation.

The equilibrium or the stationary distribution of x is:

$$p(x) \propto \exp(-f(x)) \tag{12.18}$$

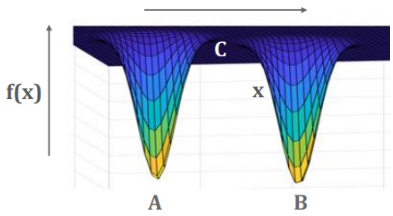
Note that this is actually the same distribution we intended to sample from in the first place. The above result regarding the stationary distribution is non-trivial to show.

12.3.1 Sampling for different kinds of f

Analogous to optimization, where we have a distinction between convex functions and non-convex functions, in Langevin Dynamics, we have a distinction between functions f that result in "easier" sampling and "harder sampling".

f is convex	f is non-convex
Unimodal	Multi-modal
Faster mixing times	Bad mixing times
Provable guarantees for efficiency (polynomial time)	Worst case: #P-hard

While there are efficient algorithms for sampling from distributions that have convex f, the unimodality makes hard to model real-life data that needn't be unimodal. However it should be easy to see why multi-modal distributions are hard to sample from:



Starting the random walk in the neighborhood of A and trying to get to the point B while passing through C takes exponential time. If we didn't have the Gaussian noise, we would be purely stuck at the basin.

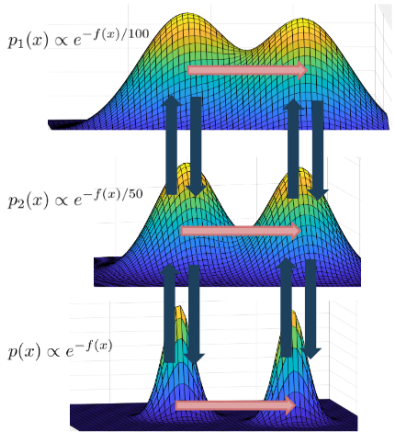
This is similar to the difficulty faced while trying to approximate a function with multiple modes with a "simple" distribution (say unimodal).

12.3.1.1 Potential Solutions for multimodal distributions / non-convex f

One way to cancel the effect of multiple modes, is to flatten the distribution. This can be achieved by scaling f appropriately - $\frac{f(x)}{50}$ is flatter that $\frac{f(x)}{10}$. Intuitively, sampling from flatter distributions is easier - the flattest is the uniform distribution.

This idea leads to a set of generic approaches that are called *simulated tempering / annealing*. The general algorithm - informally stated - is as follows:

- Run multiple chains at different temperatures
- Swap locations occasionally to assist lower temperature chains



Algorithm 1 Simulated tempering + MCMC

Require: List of temperatures L

1: Initialize $\{M_k\}_{k=1}^{|L|}$, M_k is the Markov chain at temperature L_k with stationary distribution p_k
2: Set current point x_0 and temperature k_0, $t = 0$
3: Run tempering chain below
4: **while do**
5: With probability 0.5, evolve x_t via M_{k_t} to x_{t+1}. New state: $\{x_{t+1}, k_t\}$
6: With probability 0.5, choose random k' and set next state to $\{x_t, k'\}$ with probability min $\left\{\frac{p_{k'}(x)}{p_{k_t}(x_t)}, 1\right\}$
7: **end while**

Theorem 12.11 *Let $p(x) \propto \exp(-f(x))$ be a mixture of nshifts of a d-dimensional log concave distribution. Then Langevin dynamics with simulated tempering run for* poly(n, d) *iterations samples from distribution close to p.*

Some intuition for why it works:

- Choose highest temperature that enables fast convergence i.e., flat hills.
- Partition the domain into blocks that effectively covers a mode satisfying (1) fast convergence inside each block, blocks aren't too small.