Lecture 21

Lecturer: Anshumali Shrivastava

Scribe By: Aaron Amartuvshin (aa212), Ritvik Chilakamarthy (rc157), Omkar Shelke (os30), Swarna Nallam (sn103), Abhinav Jain (aj70)

1 Recap of Markov Processes

1.1 Markov Property

The Markov property defines a stochastic process where the probability of reaching the current state depends solely on the previous state, not on the sequence of events that preceded it. Mathematically, for a Markov chain with a transition matrix P and a probability distribution π, this is represented by:

$$\pi_t = \pi_{t-1} P$$

or more generally:

$$\pi_t = \pi_0 P^t \tag{1}$$

This highlights how the state distribution evolves over time based on its initial distribution and the transition dynamics.

1.2 Stationary Distribution

A stationary distribution π for a Markov chain remains constant when further transformations are applied, implying the system has reached equilibrium. This condition is expressed as:

$$\pi = \pi P$$

Here, π acts as an eigenvector of the transition matrix P associated with an eigenvalue of 1. This stationary state is crucial for understanding the long-term behavior of Markov processes.

1.3 Exponential Convergence

The exponential convergence property of a Markov chain refers to the rapid approach of the distribution π_t to its stationary distribution as time increases. Given the evolution Eq. 1 where π_0 is the initial distribution and P is the transition matrix, this convergence is driven by the eigenstructure of P. The dominant term in this expansion is associated with the eigenvector e_1, which corresponds to the eigenvalue $\lambda_1 = 1$. All other eigenvalues λ_i satisfy $|\lambda_i| < 1$ causing their contributions to diminish exponentially as t grows. Thus, as $t \to \infty$ the distribution π_t converges to a stationary distribution proportional to e_1 ensuring that $\pi_t \approx a_1 e_1$, where a_1 is a coefficient determined by the initial state π_0. This property highlights the chain's rapid stabilization to equilibrium.

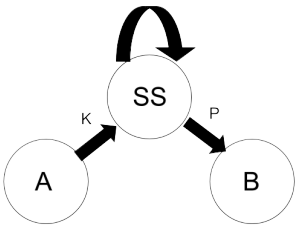

Figure 1: If a MC has a self loop and reversible, then it must be aperiodic

2 Properties of Markov Chains

2.1 Aperiodicity

A Markov chain is aperiodic if it does not get locked into cycles, allowing it to return to any state at irregular intervals rather than fixed cycles. This property is important for convergence to the stationary distribution, as a cyclic or periodic Markov chain may struggle to explore all states uniformly. A chain with aperiodicity mixes well, leading to faster convergence. If the Markov chain is aperiodic, it can avoid cyclic patterns and better achieve convergence to the stationary distribution. This can be formally stated as -

$$GCD\{t : p(x, y)^t > 0\} = 1$$

2.2 Irreducibility

Irreducibility means that every state in the Markov chain can be reached from any other state, making the chain fully connected. This property ensures that no part of the state space is isolated, and all states have a non-zero probability of being visited eventually. By ensuring irreducibility, the Markov chain can achieve a unique stationary distribution that reflects the long-term behavior across the entire state space. This can be formally stated as:

$$\exists t, s.t. P(x, y)^t > 0$$

where, $P(x, y)^t$ is the probability of reaching state y from state x in t steps.

2.3 Reversibility

Reversibility (or detailed balance) ensures that the Markov chain has a symmetric transition structure with respect to the stationary distribution. This property is defined by the detailed balance condition:

$$\pi(x)P(x, y) = \pi(y)P(y, x)$$

for all states x and y. Reversibility is a sufficient condition for the stationary distribution π, making the Markov chain easier to analyze and often accelerating convergence to equilibrium.

3 Markov Chain Example - Shuffling Algorithm

Algorithm Steps

1. **Initialize the Sequence**:

- Start with an ordered sequence $A = (1, 2, \ldots, n)$ (or any initial arrangement of n elements).

2. **Repeat for Some Number of Steps**:
   - For a fixed number of steps, repeat the following:
     (a) Randomly select two distinct indices i and j in the range $[0, n-1]$.
     (b) Swap the elements at positions i and j in the sequence A.
3. **Output the Result**:
   - After completing the set number of steps, return the sequence A, which should now be a randomly shuffled permutation of the initial sequence.

Take 4 numbers example (1,2,3,4)

1. **Initialize the Sequence**:
   - Start with an ordered sequence $A = (1, 2, 3, 4)$ (or any initial arrangement of n elements).
2. **Repeat for Some Number of Steps**:
   - For a fixed number of steps, repeat the following:
     (a) Randomly select two distinct indices i and j in the range $[0, n-1]$.
     (b) Swap the elements at positions i and j in the sequence A.
3. **Output the Result**:
   - After completing the set number of steps, return the sequence A, which should now be a randomly shuffled permutation of the initial sequence.

Applying the Algorithm to the Example Sequence $(1, 2, 3, 4)$

Here is an example of applying three steps of the algorithm to the sequence $(1, 2, 3, 4)$.

1. **Step 1**:
   - Start with the initial sequence $A = (1, 2, 3, 4)$.
   - Randomly select indices $i = 1$ and $j = 2$.
   - Swap $A[1]$ with $A[2]$:
     $$A = (1, 3, 2, 4)$$
2. **Step 2**:
   - Current sequence: $A = (1, 3, 2, 4)$.
   - Randomly select indices $i = 0$ and $j = 3$.
   - Swap $A[0]$ with $A[3]$:
     $$A = (4, 3, 2, 1)$$
3. **Step 3**:
   - Current sequence: $A = (4, 3, 2, 1)$.
   - Randomly select indices $i = 1$ and $j = 2$.
   - Swap $A[1]$ with $A[2]$:
     $$A = (4, 2, 3, 1)$$

After these three steps, the sequence A has been shuffled to $(4, 2, 3, 1)$.

4 Metropolis Process

Given an existing distribution π, how would you make a Markov Chain? Design a MC with a "unique" stat distribution as π^t.

Algorithm 1: Metropolis Process

Start at state x
Sample y from MC, $P(x)$
Move to y with Probability $A[x, y]$
Else, stay at x

In the above algorithm, $P(x)$ is a probability of transition, while $A[x, y]$ is the probability of acceptance. To update P to it's next iteration P', we use:

$$P' = PA$$

To ensure that our MC has a distribution close to π, we need to properly derive A. From reversibility, we know that:

$$\pi(x)P'(x, y) = \pi(y)P'(y, x)$$

Substituting {P'}, we get:

$$\pi(x)P(x, y)A[x, y] = \pi(y)P(y, x)A[y, x]$$

Rearranging, we get:

$$\frac{A[x, y]}{A[y, x]} = \frac{\pi(y)P(y, x)}{\pi(x)P(x, y)}$$

So, the value of A can be derived as:

$$A[x, y] = min\left(\frac{\pi(y)P(y, x)}{\pi(x)P(x, y)}, 1\right)$$

and

$$A[y, x] = min\left(\frac{\pi(x)P(x, y)}{\pi(y)P(y, x)}, 1\right)$$

5 Comparison with Rejection Sampling

Metropolis processes and rejection sampling are both methods for sampling from a target distribution, but they differ significantly in their approaches and effectiveness, especially in high-dimensional spaces.

In rejection sampling, we draw samples from a proposal distribution $g(x)$ and accept each sample x with probability:

$$\frac{f(x)}{\mu \times g(x)}$$

where $f(x) < \mu \times g(x)$ for all x, and $\mu \geq 1$ is a constant ensuring the acceptance probability is at most 1.

This method is straightforward and works well when $g(x)$ closely approximates the target distribution $f(x)$, which is feasible in low-dimensional settings. However, as the dimensionality increases, designing such a $g(x)$ becomes challenging, leading to very low acceptance rates and inefficiency due to the high number of rejected samples.

Metropolis processes address this limitation by constructing a Markov chain where each new sample depends on the current state. They accept moves based on an acceptance probability that depends on the ratio of weights and proposal probabilities, allowing the chain to explore the state space efficiently without requiring a globally similar proposal distribution. This makes Metropolis methods more suitable for high-dimensional problems, as they can incorporate domain knowledge and maintain reasonable acceptance rates even when the target distribution is complex.