Introduction to Sampling Model

Li Mingchang

School of Mathematics, South China University of Technology

malimingchang@mail.scut.edu.cn

May 21, 2021

Overview

1. Introduction
2. Law of large numbers
   - Weak law of large numbers (Khinchin's law)
   - Borel's law of large numbers
3. Monte Carlo method
   - Buffon's needle problem
   - Monte Carlo integration
4. Specific sampling method
   - Inverse transform sampling
   - Rejection sampling
   - Importance sampling
   - Self-normalized importance sampling
5. Markov chain Monte Carlo
   - Metropolis Hasting algorithm

What's Sampling Model?

In the probabilistic graph model, the Sampling Method is one of the inference methods, and it is approximate inference. In fact, the process of inference is to deduce the probability of some variables based on the known probability of some other variables. In many machine learning algorithms, the expectation value is required. There are some complicated distributions that can not be obtained directly by integral. So, we can get enough samples by sampling, then we can get an approximate expectation by averaging the sample; the more samples we take, the closer the approximate expectation is to the true expectation. In fact, this is the Monte Carlo method.

Law of large numbers

Before we get started, let's review law of large numbers in probability theory which lays a theoretical foundation for sampling model algorithms.

Weak law of large numbers

The weak law of large numbers (also called Khinchin's law) states that the sample average converges in probability towards the expected value. Assume that $X_i, i = 1, 2, \ldots, n$ is i.i.d random variables with expected value μ, then

$$\overline{X}_n \xrightarrow{P} \mu, \text{ when } n \to \infty. \tag{1}$$

That is, for any positive number ε,

$$\lim_{n \to +\infty} P|(\overline{X}_n - \mu| > \varepsilon) = 0. \tag{2}$$

Borel's law of large numbers

Borel's law of large numbers, named after Émile Borel, states that if an experiment is repeated a large number of times, independently under identical conditions, then the proportion of times that any specified event occurs approximately equals the probability of the event's occurrence on any particular trial; the larger the number of repetitions, the better the approximation tends to be. More precisely, if E denotes the event in question, p its probability of occurrence, and $N_n(E)$ the number of times E occurs in the first n trials, then with probability one,

$$\frac{N_n(E)}{n} \xrightarrow{P} p, \text{ when } n \to \infty. \tag{3}$$

It also comes in the form of

$$\forall \varepsilon > 0, \lim_{n \to +\infty} P(|\frac{N_n(E)}{n} - p| < \varepsilon) = 1. \tag{4}$$

Monte Carlo method

The most famous application of the Monte Carlo method is to find π.

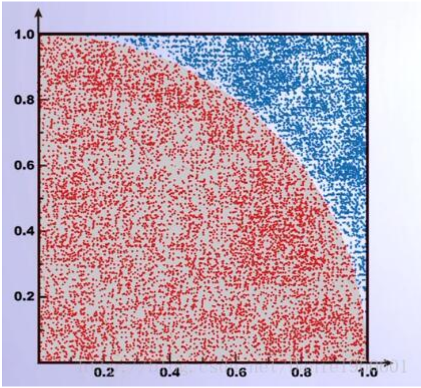


Figure: The most common method for estimating π

Buffon's needle problem

Also, we have another way to estimate π. Suppose we have a floor made of parallel strips of wood, each the same width, and we drop a needle onto the floor. What is the probability that the needle will lie across a line between two strips? Buffon's needle was the earliest problem in geometric probability to be solved. It can be solved using integral geometry. The solution for the sought probability p, in the case where the needle length l is not greater than the width d of the strips, is

$$p = \frac{2}{\pi}\frac{l}{d}. \tag{5}$$

Buffon's needle problem

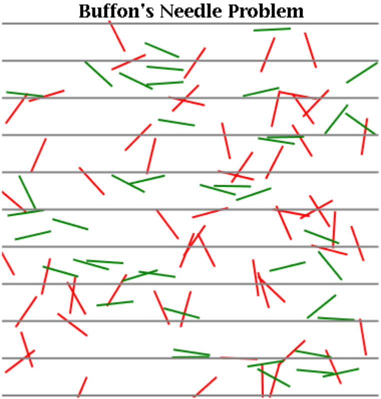


Figure: Buffon's needle problem

Monte Carlo integration

Monte Carlo integration is a numerical method for calculating integration. It can be used to calculate some integrals in machine learning.
For example, if we want to calculate $S = \int_a^b f(x)dx$, what can we do? We start from a simple transform. Let

$$S = \int_a^b \frac{f(x)}{p(x)} p(x) dx, \tag{6}$$

where $p(x)$ is a PDF(probability density function).

Monte Carlo integration

Suppose that $X_1, X_2, \ldots, X_n \sim p(x)$ and X_i is independent, x_i is a sample of X_i. Then

$$\hat{S}_n = \frac{1}{n}\sum_{k=1}^{n}\frac{f(x_k)}{p(x_k)} \tag{7}$$

is a estimate of S. Obviously, $\hat{S}$ is an unbiased estimate of S with

$$E(\hat{S}_n) = S \quad \text{and} \quad Var(\hat{S}_n) = \frac{1}{n}\{\int_a^b \frac{[f(x)]^2}{p(x)}dx - S^2\}. \tag{8}$$

Monte Carlo integration

Specifically, if

$$p(x) = \begin{cases} \frac{1}{b-a} & \text{a<x<b} \\ 0 & \text{otherwise} \end{cases}$$

which means $p(x)$ is uniform distribution of (a, b).
Then

$$\hat{S}_n = \frac{b-a}{n} \sum_{k=1}^{n} f(x_k) \tag{9}$$

is the well-known method of calculating definite integral.

Inverse transform sampling

From this section, we discuss some specific Sampling Methods. Firstly, we We'll start by introducing inverse transform sampling.
Inverse transform sampling is a basic method for pseudo-random number sampling, i.e., for generating sample numbers at random from any probability distribution given its cumulative distribution function.

Inverse of distribution function

General function inversion can not be applied to CDF. For general cases, define

$$F^{-1}(u) = inf\{x|F(x) \geq u\}, 0 < u < 1. \tag{10}$$

Inverse of distribution function

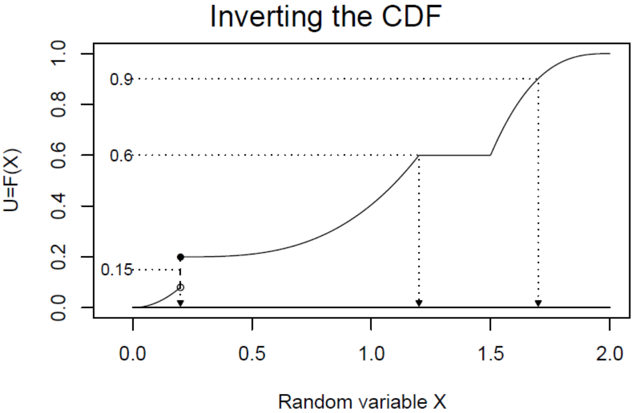
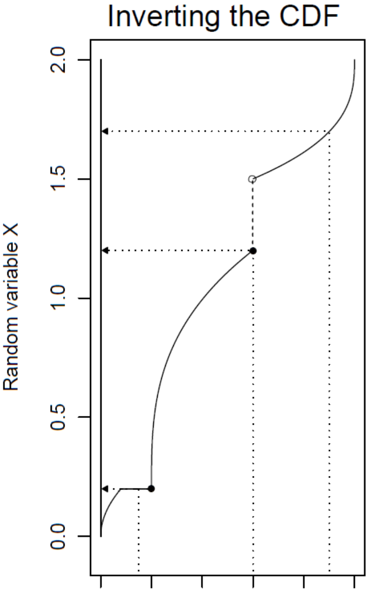


Figure: CDF

Inverse of distribution function



Inverse transform sampling

The Inverse transform sampling procedure is as follows:
1. Generate a random number u_i from a uniform distribution $U(0,1)$.
2. Calculate $x_i = F_X^{-1}(u_i)$ as sample result of random varible X, where $F_X(x)$ is a CDF(cumulative distribution function) with

$$F_X(x) = \int_{-\infty}^{x} f_X(u)du = P(X \leq x). \tag{11}$$

The proof of inverse transform sampling

Now we proof that x_i can be a sample of random varible X.
Since $U \sim U(0,1)$, we have

$$F_U(u) = \begin{cases} 0 & u \leq 0 \\ u & 0 < u < 1 \\ 1 & u \geq 1 \end{cases}$$

Then

$$\begin{aligned} F_{X_i}(x) &= P(X_i \leq x) \\ &= P(F_X^{-1}(U) \leq x) \\ &= P(F_X(F_X^{-1}(U)) \leq F_X(x)) \\ &= P(U \leq F_X(x)) \\ &= F_U(F_X(x)) = F_X(x) \end{aligned}$$

Hence, X and X_i follow the same distribution. x_i is a sample of X.

Example of inverse transform sampling

Try to generate a random variable $X \sim Exp(\lambda)$. In this case,

$$F(x) = 1 - e^{-\lambda x}, x \geq 0 \tag{12}$$

and

$$F^{-1}(u) = -\frac{1}{\lambda} ln(1-u), 0 < u < 1. \tag{13}$$

Hence, to generate a random variable $X \sim Exp(\lambda)$, we need to generate rv $U \sim U(0,1)$ and let

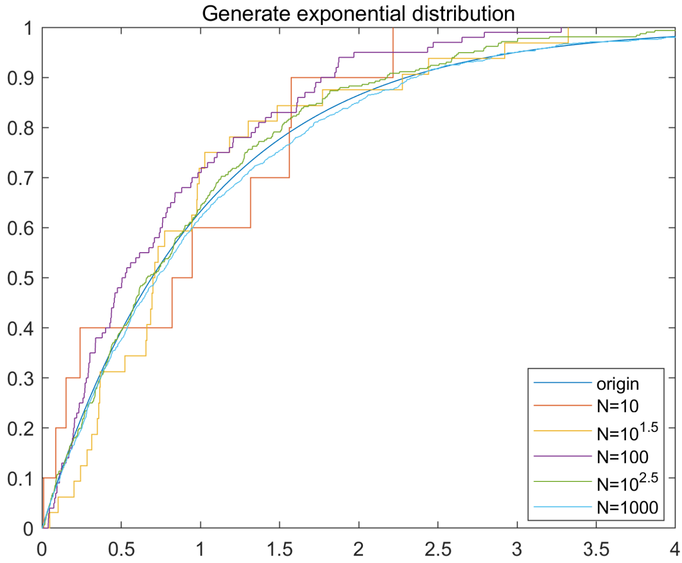
$$X = -\frac{1}{\lambda} ln(1-U). \tag{14}$$

Now, we get a rv $X \sim Exp(\lambda)$.

Numerical simulation in MATLAB

```
lambda=1;
plot(y,1-exp(-lambda.*y));
hold on
for i=1:0.5:3
datasize=round(10^i);
data=rand(1, datasize);
X=-log(data)./lambda;
stairs([0,sort(X)],0:1/datasize:1);
end
xlim([0,4]);
legend('origin','N=10','N=10^{1.5}','N=100','N=10^{2.5}',
'N=1000','location','best')
title('Generate exponential distribution')
```

Visualization of results



The pros and cons of inverse transform sampling

As a basic way to sample, inverse transform sampling has its advantange and disadvantange.
Pros: Easy.
Cons: Usually, it's hard to calculate CDF and its inverse.

Rejection sampling

In order to solve shortcomings of inverse transform sampling, rejection sampling has been proposed.

The basic idea is that by covering the “Lower probability distribution” with a “Higher probability distribution”. The “Higher probability distribution” is easier to sample (standard distribution). The samples can be considered as samples from the “Lower probability distribution” at certain probability.

Rejection sampling

We aim to get a sample z (corresponds to random variable Z) from $p(z)$. To sample $p(z)$, we need a proposal distribution $q(z)$ and a constant k with

$$\forall z, p(z) \leq kq(z). \tag{15}$$

$kq(z)$ is called the comparison function.

Rejection sampling

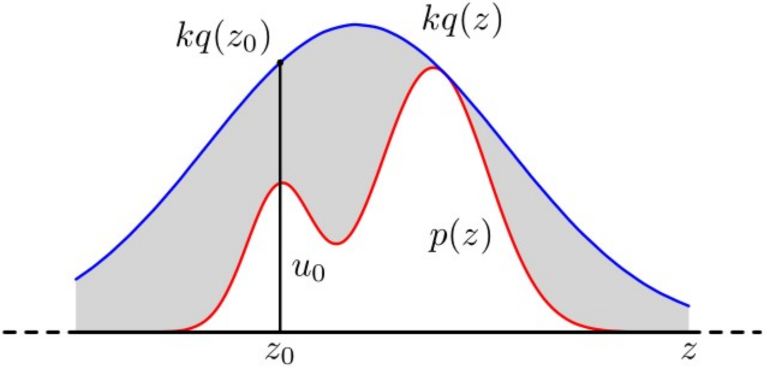


Figure: Rejection sampling

Step of rejection sampling

Now we describe the step of rejection sampling:
Step 1: Get a sample y (corresponds to random variable Y) from proposal distribution $q(z)$.
Step 2: Calculate acceptance probability

$$\alpha = \frac{p(y)}{kq(y)}. \tag{16}$$

step 3: Get a sample u (corresponds to random variable U independent with Y) from $U(0,1)$.
Step 4: If $\alpha \geq u$, accept y as a sample of $p(z)$; else return step 1.

Accept region

By introducing accept region we can get a deeper understanding of the step.

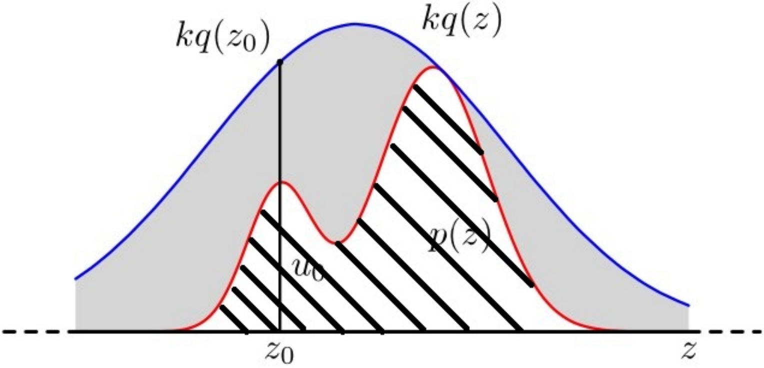


Figure: Shaded part is accept region

Properties of rejection sampling

We have some properties about rejection sampling.

- $p(Y), q(Y)$ are rv independent with U.
- $0 \leq \frac{p(Y)}{kq(Y)} \leq 1$.
- $k \geq 1$.
- N the number of times required to get a sample is a rv with $N \sim Ge(\frac{1}{k})$.

Acceptance ratio

To calculate the acceptance ratio of each sample, we need the following conculsion:

$$\begin{aligned} P(U \leq \frac{p(Y)}{kq(Y)} | Y = y) &= P(U \leq \frac{p(y)}{kq(y)} | Y = y) \\ &= P(U \leq \frac{p(y)}{kq(y)}) \\ &= \frac{p(y)}{kq(y)}. \end{aligned} \tag{17}$$

Acceptance ratio

By total probability theorem and equation 17, acceptance ratio

$$\begin{aligned} p &= P(U \leq \frac{p(Y)}{kq(Y)}) \\ &= \int_{-\infty}^{+\infty} P(U \leq \frac{p(y)}{kq(y)} | Y = y) q(y) dy \\ &= \int_{-\infty}^{+\infty} \frac{p(y)}{kq(y)} q(y) dy \\ &= \frac{1}{k} \int_{-\infty}^{+\infty} p(y) dy \\ &= \frac{1}{k}. \end{aligned} \tag{18}$$

That's why the smaller the k is, the better.

The proof of rejection sampling

Now, we verify rejection sampling by CDF. We only need to proof

$$P(Y \leq x | y \text{ is accepted}) = F_Z(x). \tag{19}$$

That is

$$P(Y \leq x | U \leq \frac{p(y)}{kq(y)}) = F_Z(x). \tag{20}$$

The proof of rejection sampling

Using the basic fact $P(A|B) = \frac{P(B|A)P(A))}{P(B)}$ that

$$\begin{aligned}
P(Y \le x | U \le \frac{p(y)}{kq(y)}) &= \frac{P(Y \le x)P(U \le \frac{p(y)}{kq(y)}|Y \le x)}{P(U \le \frac{p(y)}{kq(y)})} \\
&= \frac{F_Y(x)P(U \le \frac{p(y)}{kq(y)}|Y \le x)}{\frac{1}{k}} \\
&= kF_Y(x)P(U \le \frac{p(y)}{kq(y)}|Y \le x). \qquad (21)
\end{aligned}$$

The proof of rejection sampling

Notice that

$$\begin{aligned} P(U \leq \frac{p(y)}{kq(y)} | Y \leq x) &= \frac{P(U \leq \frac{p(y)}{kq(y)}, Y \leq x)}{F_Y(x)} \\ &= \frac{1}{F_Y(x)} \int_{-\infty}^{x} P(U \leq \frac{p(y)}{kq(y)} | Y = w) q(w) dw \\ &= \frac{1}{F_Y(x)} \int_{-\infty}^{x} \frac{p(w)}{kq(w)} q(w) dw \\ &= \frac{1}{F_Y(x)} \frac{1}{k} F_X(x). \end{aligned} \tag{22}$$

By equation 21 and 22 we get the conclusion

$$P(Y \leq x | U \leq \frac{p(y)}{kq(y)}) = F_X(x). \tag{23}$$

Adaptive rejection sampling

For many distributions, finding a proposal distribution that includes the given distribution without a lot of wasted space is difficult. An extension of rejection sampling that can be used to overcome this difficulty and efficiently sample from a wide variety of distributions (provided that they have log-concave density functions, which is in fact the case for most of the common distributions—even those whose density functions are not concave themselves!) is known as adaptive rejection sampling (ARS).

Log-concave function

In convex analysis, a non-negative function $f: R^n \to R^+$ is log-concave if its domain is a convex set, and if it satisfies the inequality

$$Inf(\theta x + (1-\theta)y) \leq \theta Inf(x) + (1-\theta)Inf(y) \tag{24}$$

or

$$f(\theta x + (1-\theta)y) \leq f(x)^{\theta} + logf(y)^{1-\theta} \tag{25}$$

for all $x, y \in domf$ and $0 < \theta < 1$.
These are also equivalent to $Inf(x)$ is concave.

Adaptive rejection sampling

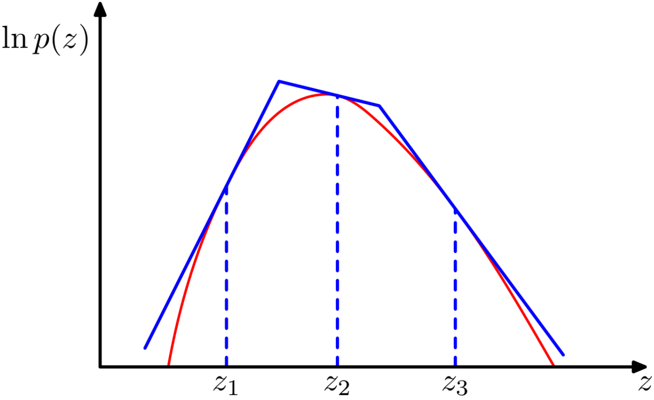


Figure: Adaptive rejection sampling

Importance sampling

In statistics, importance sampling is a general technique for estimating properties of a particular distribution, while only having samples generated from a different distribution than the distribution of interest.

Importance sampling

Suppose that we want to calculate $\mu = E[f(z)] = \int f(z)p(z)dz$ by Monte Carlo integration. When f is large and p is small, the probability of sampling in this part is very low leading the variance very large.

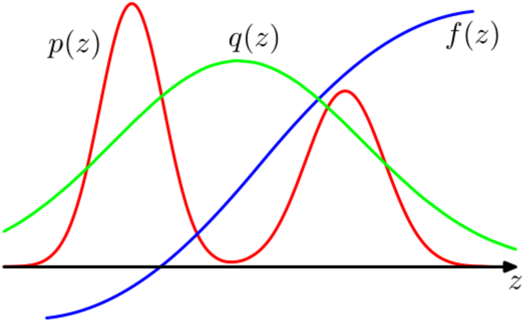


Figure: Deficiencies in the basic approach

Importance sampling

To reduce variance, importance distribution $q(z)$ is proposed. By simple transform,

$$\mu = \int f(z)p(z)dz = \int \frac{f(z)p(z)}{q(z)}q(z)dz = E_q[\frac{f(z)p(z)}{q(z)}]. \tag{26}$$

Then generate samples $z_1, z_2, \ldots, z_n$ according to distribution $q(z)$. One of estimates of μ is

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n}\frac{f(z_i)p(z_i)}{q(z_i)} \tag{27}$$

with

$$E_q[\hat{\mu}] = \mu \tag{28}$$

and

$$Var_q[\hat{\mu}] = \frac{1}{n}[\int \frac{f^2(z)p^2(z)}{q(z)}dz - \mu^2] = \frac{1}{n}\int \frac{(f(z)p(z) - \mu q(z))^2}{q(z)}dz. \tag{29}$$

Importance sampling

In particular, $q(z)$ must statisfies the following basic conditions:

- $q(z)$ is a PDF.
- When $f(z)p(z) \neq 0$, $q(z) > 0$. This means we can't ignore any z where $f(z)p(z) \neq 0$.
- To prevent $Var_q[\hat{\mu}]$ from being infinity, $\frac{p(z)}{q(z)}$ must be finite.

Importance sampling

Now, we want to find opitmal $q(z)$.

By equation 29, if $q(z) = \frac{f(z)p(z)}{\mu}$, then $Var_q[\hat{\mu}] = 0$. But there are two problems:

- $q(z)$ must be a PDF but $f(z)$ might be negetive.
- μ is exactly what we want to calculate.

Importance sampling

However, we can prove that $\tilde{q}(z) = \frac{|f(z)|p(z)}{c}$ can minimize $Var_q[\hat{\mu}]$ where c is a constant making $\tilde{q}(z)$ be a PDF. In this way

$$c = \int |f(z)|p(z)dz \tag{30}$$

and

$$\tilde{q}(z) = \frac{|f(z)|p(z)}{\int |f(z)|p(z)dz}. \tag{31}$$

Importance sampling

Notice that

$$\begin{aligned} Var_{\tilde{q}}[\hat{\mu}] &= \int \frac{f^2 p^2}{\tilde{q}} dz - \mu^2 \\ &= (\int |f| p dz)^2 - \mu^2 \\ &= (\int \frac{|f| p}{q} q dz)^2 - \mu^2 \\ &\leq \int \frac{f^2 p^2}{q^2} q dz - \mu^2 \\ &= \int \frac{f^2 p^2}{q} dz - \mu^2 \\ &= Var_q[\hat{\mu}]. \end{aligned} \tag{32}$$

Hence, for any $q(z)$, $Var_{\tilde{q}}[\hat{\mu}] \leq Var_q[\hat{\mu}]$ by 43.

Self-normalized importance sampling

In some problems, we don't actually know the exact form of target distribution $p(z)$ but we know nonstandardized form $p_0(z) = cp(z)$ where $c > 0$ is an unknown constant. For importance distribution, assume that we know the non-standard distribution $q_0(z) = bq(z)$ where $b > 0$ is an unknown constant. In this case, first we compute ratio

$$w(z) = \frac{p_0(z)}{q_0(z)} = \frac{c}{b} * \frac{p(z)}{q(z)}, \tag{33}$$

and consider the self-normalized importance sampling estimate

$$\tilde{\mu} = \frac{\sum_{i=1}^{n} f(Z_i)w(Z_i)}{\sum_{i=1}^{n} w(Z_i)}, Z_i \sim q(z). \tag{34}$$

Self-normalized importance sampling

Let $\tilde{\mu}$ be the self-normalized importance sampling estimate as 34 shown. Then

$$P(\lim_{n \to +\infty} \tilde{\mu} = \mu) = 1, \tag{35}$$

and

$$Var_q[\tilde{\mu}] = \frac{1}{n} E_q[w(X)^2 (f(X) - \mu)]. \tag{36}$$

The self-normalized importance sampler $\tilde{\mu}$ requires a stronger condition on q than the unbiased importance sampler $\hat{\mu}$ does. We now need $q(x) > 0$ whenever $p(x) > 0$ even if $f(x)$ is zero with high probability.

Markov chain Monte Carlo

The sampling methods we have known before are all parallelizable sampling methods. Next, we introduce a serialized sampling method named Markov Chain Monte Carlo.

A first-order Markov chain is defined to be a series of random variables $X_0, X_1, \ldots, X_{n+1}$ with

$$P(X_{n+1}|X_0, \ldots, X_n) = P(X_{n+1}|X_n). \tag{37}$$

Markov chain Monte Carlo

Let $\pi(x)$ be the target distribution and $k(x^*|x)$ be the Chapman-Kologronvo equation, where in general, in a markov process:

$$\pi_t(x^*) = \int \pi_{t-1}(x) k(x^*|x) dx. \tag{38}$$

Obviously, at equilibrium, that stationary distribution satisfies

$$\pi(x^*) = \int \pi(x) k(x^*|x) dx. \tag{39}$$

Markov chain Monte Carlo

In some cases, equation 39 is difficult to verify. One of its sufficient condition named detailed balanced condition holds when for any x, x^*,

$$\pi(x)k(x^*|x) = \pi(x^*)k(x|x^*). \tag{40}$$

By 40, 39 holds for

$$\int \pi(x)k(x^*|x)dx = \int \pi(x^*)k(x|x^*)dx = \pi(x^*)\int k(x|x^*)dx = \pi(x^*). \tag{41}$$

Metropolis Hasting algorithm

The Metropolis Hasting algorithm is as follows:

```
1. Initialise $x^{(0)}$.
2. for $i = 0$ to $N - 1$
       $u \sim U(0,1)$
       $x^* \sim q(x^*|x^{(i)})$
       if $u < min(1, \frac{\pi(x^*)q(x|x^*)}{\pi(x)q(x^*|x)})$
           $x^{(i+1)} = x^*$
       else
           $x^{(i+1)} = x^{(i)}$
```

Metropolis Hasting algorithm

Propose x^* from $q(x^*|x)$, then accept x^* with ratio $\alpha(x^*)$, where

$$\alpha(x^*) = min(1, \frac{\pi(x^*)q(x|x^*)}{\pi(x)q(x^*|x)}). \tag{42}$$

Why it works?

We only need to verify it satisfy detailed balance:

$$\begin{aligned}\pi(x)k(x^*|x) &= \pi(x)q(x^*|x)min(1, \frac{\pi(x^*)q(x|x^*)}{\pi(x)q(x^*|x)}) \\ &= min(\pi(x)q(x^*|x), \pi(x^*)q(x|x^*)) \\ &= min(\pi(x)q(x^*|x), \pi(x^*)q(x|x^*)) * \frac{\pi(x^*)q(x|x^*)}{\pi(x^*)q(x|x^*)} \\ &= \pi(x^*)q(x|x^*)min(1, \frac{\pi(x)q(x^*|x)}{\pi(x^*)q(x|x^*)}) \\ &= \pi(x^*)k(x|x^*). \end{aligned} \tag{43}$$

References and Acknowledgements

The references in this report are as follows:

- *Bishop, Christopher M. Pattern Recognition and Machine Learning. New York :Springer, 2006.*
- `https://github.com/roboticcam`
- `https://space.bilibili.com/244865478`
- `https://statweb.stanford.edu/~owen/mc/`
- `https://towardsdatascience.com/what-is-rejection-sampling-1f6aff92330d`

In addition, thank you *Xiong Yifei(SCUT)* for your contribution to this report.

The End