Pseudo-random Number Sampling

Wantee Wang

2015-05-13T18:10:05+08:00

Contents

Pseudo-random number sampling is the numerical practice of generating pseudo-random numbers that are distributed according to a given probability distribution.

It is hard to design a algorithm to directly sample a random variable from a pdf in a high-dimension space. So We first restrict ourself to the simpler problem: drawing a sample from uniform distribution on $[0, 1]$. Once we solve that, we can easily extends it to uniform distribution on any interval by scaling.

1 Uniform distribution on [0,1]

Unfortunately, it is impossible to draw a rational uniformly at random. According to measure theory, on the interval $[0, 1]$, the measure on the set of all rational numbers is 0, and all irrational numbers is 1. In other words, compared to irrationals, rationals can be ignored, so one cannot sample a rational number in finite steps. However, in real computers, irrational number can't be represented, thus our sampling must be in a pseudo way.

Further, we know that the set of all positive integers and the set of all rational numbers have the same cardinality. Then, we can simplify the problem to sample positive integers.

The classic approach is the *Linear Congruential Generator* (LCG). The generator is defined by the recurrence relation:

$$X_{n+1} = (aX_n + c) \mod m$$

where X is the sequence of pseudorandom values, $m > 0$ is the "modulus", $0 < a < m$ is the "multiplier", $\$ \ 0 \leq c < m \$ is the "increment" and $0 \leq X_0 < m$ is the "seed".

Most C compliers uses this algorithm to implement the `rand()` function.

2 Inverse Transform Sampling

Next, we extend the problem to arbitrary distribution of one dimension.

This method is called inverse transform sampling, because it plays on inverse functions. It works as follows:

1. Generate a random number u from the uniform distribution in the interval $[0, 1]$.
2. Compute the value x such that $F(x) = u$.
3. Take x to be the random number drawn from the distribution described by F.

We show its correctness in the following. If the probability distribution has a cumulative distribution function (CDF):

$$F(x) = Pr(X \leq x) = \int_{-\infty}^{+\infty} f(t)dt.$$

Where f(t) is the probability density function. Since the CDF is monotone non-decreasing, it has an inverse function $F^{-1}(u)$.

Let U be the number generated on $[0, 1]$. Then,

$$\begin{aligned} Pr(F^{-1}(U) \leq x) &= Pr(U \leq F(x)) \\ \text{(applying } F\text{, which is monotonic, to both sides)} & \\ &= F(x) \\ \text{(because } Pr(U \leq y) = y\text{)} & \end{aligned}$$

Therefore, $X = F^{-1}(U)$ follows the distribution $F(x)$.

This is shown clearly in this figure (Figure 1) (from here).

The whole process can be seen as retrieving the x value which has the y value equal to some sample u from $U(0, 1)$.

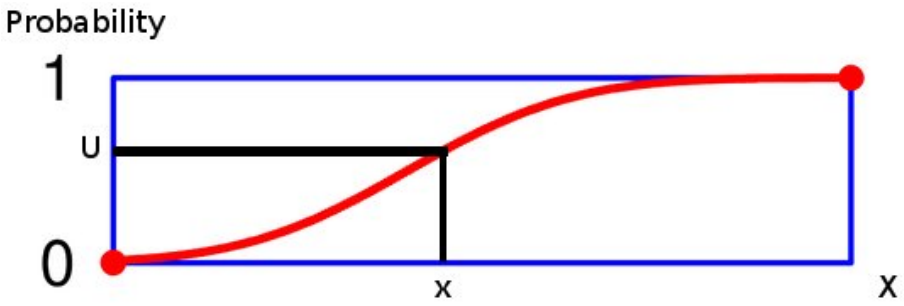

Figure 1: A cumulative distribution function.

3 Gibbs Sampling

For multivariate probability distribution, there is Gibbs sampling.

Gibbs sampling is a Markov chain Monte Carlo (MCMC) algorithm. The point of Gibbs sampling is that given a multivariate distribution it is simpler to sample from a conditional distribution than to marginalize by integrating over a joint distribution.

Suppose we want to obtain k samples of $\mathbf{X} = (x_1, \ldots, x_n)$ from a joint distribution $p(x_1, \ldots, x_n)$. Denote the i-th sample by $\mathbf{X}^{(i)} = (x_1^{(i)}, \ldots, x_n^{(i)})$. We proceed as follows:

1. We begin with some initial value $\mathbf{X}^{(0)}$.
2. for $j = 1 : n$; sample $x_j^{(i+1)} \sim p(x_j^{(i)} \,|\, x_{-j}^{(i)})$ where $p(x_j^{(i)} \,|\, x_{-j}^{(i)}) = p(x_j^{(i)} \,|\, x_1^{(i+1)}, \ldots, x_{j-1}^{(i+1)}, x_{j+1}^{(i)}, \ldots, x_n^{(i)})$
3. Repeat the above step k times.

In this dimension-to-dimension way, we can sample a multi-dimension distribution.

Online version at http://wantee.github.io/2015/05/13/pseudo-random-number-sampling/