Mathematics for Machine Learning (AI 512): MCMC Sampling

Amit Chattopadhyay

IIIT-Bangalore

Sampling: Motivation and Classical Methods

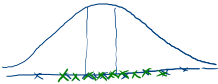
Random Sampling following a distribution

Q: What do we mean by generating samples following a *probability mass function* (p.m.f.) or *probability distribution function* (p.d.f.)?

Sample of size n : $x_1, x_2, \ldots, x_n$ from some population d.f. $F_X(x)$

Define a Fake r.v. $\mathring{X}$: takes $x_1, x_2, \ldots, x_n$ with probability

$$P(\mathring{X} = x_i) = \frac{1}{n} \quad , \quad i = 1, 2, \ldots n$$



$$F_{\mathring{X}}(x) = P(\mathring{X} \leq x) = \frac{\nu}{n}$$: empirical distribution

Th: For $n \to \infty$, $F_{\mathring{X}}(x) \longrightarrow F_X(x)$. (empirical distribution is statistical image of population distribution)

$$P(\mathring{X}=1) = \frac{\nu_1}{n}, \; P(\mathring{X}=2) = \frac{\nu_2}{n}, \ldots, P(\mathring{X}=6) = \frac{\nu_6}{n}$$

Motivation

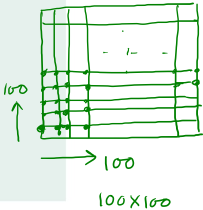
Q: Why do we need samples?

- Calculating **marginal distributions** (over d-dimensional domain)

$$p(x_1) = \sum_{x_2,\ldots,x_d} p(x_1, x_2, \ldots, x_d)$$

- Calculating **expectation** of some function (over d-dimensional domain)

$$\mathbb{E}(f(\underline{X})) = \sum_{x_1,\ldots,x_d} f(x_1, x_2, \ldots, x_d)p(x_1, \ldots, x_d)$$



Difficulty: Computation requires summation over an **exponential** number of values

Motivation

Monte Carlo Simulation:

Finding **approximate answer** by generating a sample of size n following the distribution of $p(x_1, \ldots, x_d)$

1. Generate a **sample** of size n: $\left\{ \underline{x}^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)}); \ i = 1, 2, \ldots, n \right\}$
2. Compute: $y_i = f(\underline{x}^{(i)})$
3. $\displaystyle \lim_{n \to \infty} \frac{y_1 + y_2 + \ldots + y_n}{n} = \mathbb{E}(f(\underline{X}))$

Markov Chain Monte Carlo (MCMC):

- Generate samples via designing a Markov Chain (MC) whose states are possible values of $\underline{x} = (x_1, \ldots, x_d)$
- Ensure **stationary probabilities** of the states are exactly $p(x_1, x_2, \ldots, x_d)$

$\boxed{\pi = \pi P}$

- Under some mild conditions, number of steps in RW grows **polynomially**, whereas, number of states grows **exponentially**

Few Classical Sampling Methods

- Inverse transformation method
- Rejection sampling
- Importance sampling

S.M. Ross.

Ch 11.

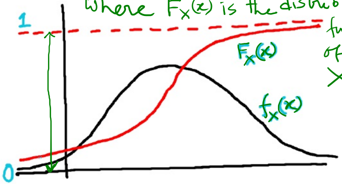
Inverse Transformation Method

uniform

Simulating a random variable X having continuous distribution

Prop 1: $Y = F_X(X)$ has $U(0,1)$ distribution.

where $F_X(x)$ is the distribution function of X



Transformation: $\boxed{y = F_X(x)} \Rightarrow$ $F_X(x)$ is mon. inc. fn.

$\frac{dy}{dx} = f_X(x)$

$$
\begin{aligned}
G_Y(y) &= \text{d.f. of the r.v. } Y \\
&= P(Y \leq y) \\
&= P(F_X(X) \leq F_X(x)) \\
&= P(X \leq x) \\
&= F_X(x)
\end{aligned}
$$

when x varies from $-\infty$ to ∞

y varies from 0 to 1.

$$g_Y(y) = \text{p.d.f. of } Y = G_Y'(y) = \frac{d}{dx} F_X(x) \frac{dx}{dy} = f_X(x) \frac{1}{f_X(x)} = 1$$

$$\forall y \in (0,1)$$

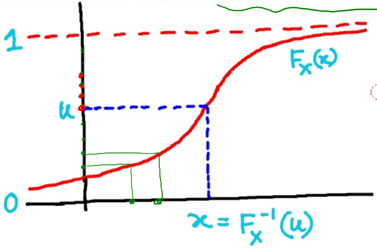
Inverse Transformation Method

Prop 2: Let U be a uniform $(0,1)$ random variable. For any continuous CDF F if we define the random variable X by

$$\boxed{X = F^{-1}(U)}$$

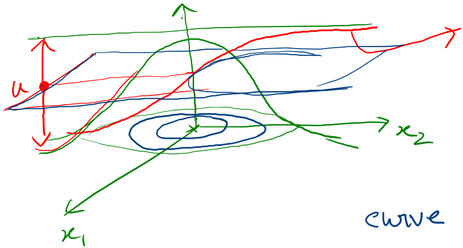
then random variable X has distribution function F.

$$f_U(u) = 1, \quad u \in (0,1)$$
$$= 0, \quad \text{else}$$



$G_X(x)$ = c.d.f. of X

$$= P(X \le x)$$
$$= P(F^{-1}(U) \le x)$$
$$= P(U \le F(x))$$
$$= \int_0^{F(x)} 1 . \, du = F(x)$$



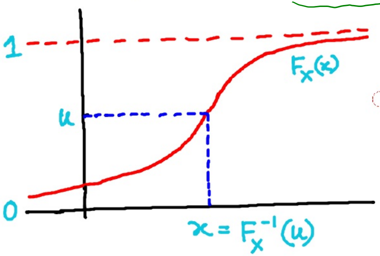
$\Phi_{X,Y} : \mathbb{R}^2 \to \mathbb{R}$ $\phi_{X,Y}$

$Z = \Phi(X,Y) \sim U(0,1)$

curve $\sim \Phi_{X,Y}(u)$

Sampling using Inverse Transformation Method

1. Generate a random number $u \sim U(0,1)$ using a pseudorandom number generator
2. Compute: $x = F^{-1}(u)$
3. This yields the desired samples: x that follow $f_X(x)$



Drawback: 1. Computing F^{-1} for a general function in d-dimensional domain is difficult or may not be possible.
2. The level set: $F^{-1}(u)$ when $d > 1$ is not a single point, but an isosurface (infinite set of points).

A Pseudorandom Number Generator

1. Start initial number X_0 (**seed**)
2. Choose positive integers: a, c and m
3. Compute recursively:

$$X_{n+1} = (aX_n + c) \text{ module } m, \text{ for } n \geq 0$$

$\{0, 1, 2, \ldots, m-1\}$

4. Consider $U_n = X_n/m$ samples from $U(0,1)$ distribution