STATS102C Lecture Notes

Ying Nian Wu

November 27, 2015

Contents

1 Introduction

Monte Carlo: European Las Vegas
S. Ulam: complained about his uncle going to Monte Carlo
N. Metropolis (colleague): named sampling method as Monte Carlo method

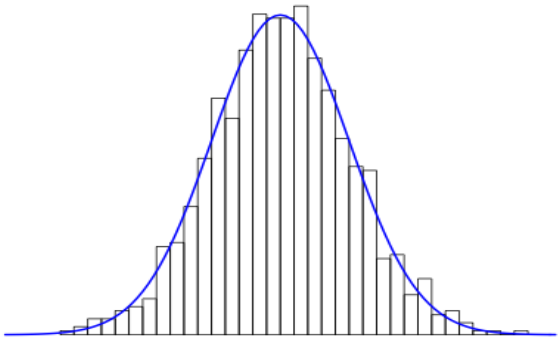
1.1 Three important inventions in 1940s and 1950s

Bombs
- atomic / hydrogen bombs
nucleon movement calculation is huge (people acting like computer)
Computers
Computers doing Monte Carlo calculations
- Eniac (Electronic Numeric Integrator and Computer)
- Maniac (Math Analyzer Numerical Integrator and Computer)
Monte Carlo
Applied areas:
physics, statistical physics
engineering, computer science
statistics, machine learning
Players
E. Fermi, J. Von Neumann, E. Teller, S. Ulam, N. Metopolis

1.2 Integration

$$E(h(x)) = \int h(x)f(x)dx = \int h(x_1...x_p)f(x_1...x_p)dx_1...dx_p$$

$$x \sim f(x)$$



Note: $f(x)$ is density function.
Here generates the curse of dimensionality. But randomness comes to rescue.

$$X_1, X_2, ...X_n \sim f(x)$$

$$X_i = \begin{pmatrix} X_{i1} \\ X_{i2} \\ ... \\ X_{ip} \end{pmatrix}$$

$$X = \begin{pmatrix} X_1 \\ X_2 \\ ... \\ X_p \end{pmatrix}$$

Now we estimate the integration:

$$\hat{I} = \frac{1}{n} \sum_{i=1}^{n} h(X_i)$$

$$E(\hat{I}) = I$$

This means the random variable centered at I.

$$Var(\hat{I}) = \frac{Var(I)}{n}$$

Here $\frac{Var(I)}{n}$ is a fixed number, so there is no dimensionality in error because error only depends on the number of samples drawed. So Monte Carlo gets around the curse of dimensionality, under the assumption that we could draw samples from high dimensional population.

1.3 Markov Chain Monte Carlo (MCMC)

Example Card Shuffling
52! permutations

random transposition → uniform distribution
7 shuffles to get close to the uniform distribution
Example Travelling Salesman Problem
To find the shortest path to travel to each city once.

$$P(x) = \frac{1}{z} e^{\frac{-\text{length(x)}}{\text{Temperature}}}$$

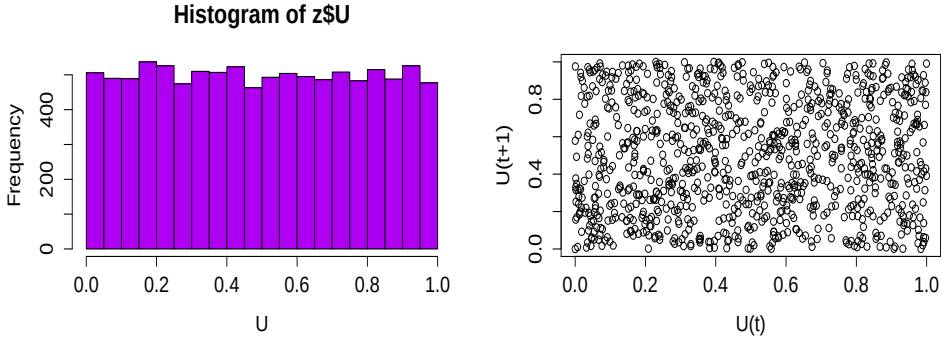
short length → high probability
reduce temperature→ simulated annealing → converges to the shortest path
Note: annealing: to start from high temperature and then to reduce temperature gradually.

1.4 Analysis by Synthesis

one type of machine learning
imagined / memorized $\xrightarrow{\text{compare and update until matched}}$ reality

2 Random Number Generators

There are some roots $U_1, U_2, ...U_t$ are random variable following a uniform distribution, Unif$[0, 1]$.
i.i.d: independent and identically distributed.



we can see the distribution of U_t by following histogram of U_t and the plot of U_t vs U_{t+1}.
In computer, if $X_1, X_2, ...X_t \overset{iid}{\sim} \text{Unif}\{0, 1, 2, ...M-1\}$, then

$$U_t = \frac{X_t}{M} \sim \text{Unif}\{0, \frac{1}{M}, \frac{2}{M}, ...\frac{M-1}{M}\} \overset{\text{approx.}}{\sim} \text{Unif}[0, 1]$$

There are several ways to generate random numbers, such as Linear Congruential Method, Inversion Method, etc.

2.1 Linear Congruential Method

Start from an integer X_0, and an random seed. The iteration is

$$X_{t+1} = aX_t + b \ \ (mod\ M)$$

Note: a, b, M are carefully chosen integers.

Example:

$$X_0 = 1, X_{t+1} = 5X_t + 0 \ \ (mod\ 7)$$
$$X_1 = 5 \times 1 + 0 \ \ (mod\ 7) = 5$$
$$X_2 = 5 \times 5 + 0 \ \ (mod\ 7) = 4$$
$$X_3 = 4 \times 5 + 0 \ \ (mod\ 7) = 6$$
$$X_4 = 6 \times 5 + 0 \ \ (mod\ 7) = 2$$
$$X_5 = 2 \times 5 + 0 \ \ (mod\ 7) = 3$$
$$X_6 = 3 \times 5 + 0 \ \ (mod\ 7) = 1$$

These is a Pseudo random number generator, and circulate these random numbers.
A better Pseudo random number generator function is

$$X_{t+1} = 7^5 X_t + 0 \ \ (mod\ 2^{31} - 1)$$

It can go through all numbers from 0 to $2^{31} - 1$. And then,

$$U_t = \frac{X_t}{M} \sim \text{Unif}[0, 1]$$

2.2 Inversion Method

2.2.1 Discrete Random Variables

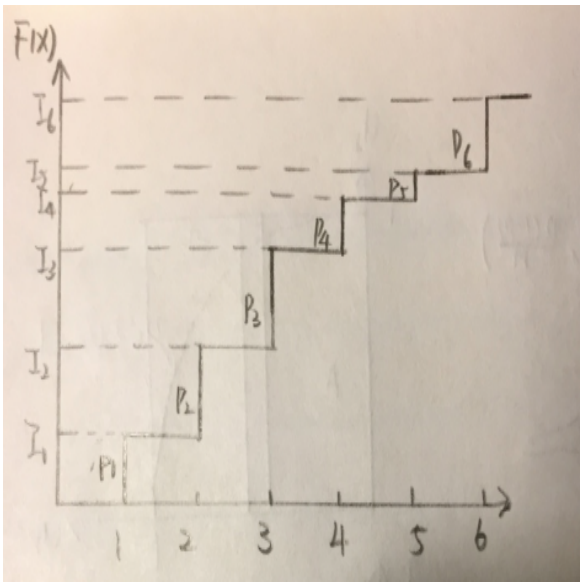
If we want to simulate coin flipping. $U \sim \text{Unif}[0, 1]$,it will return head if $U \geq \frac{1}{2}$, or it will return tail if $U < \frac{1}{2}$. If it is biased, it will return head if $U \geq \frac{2}{3}$ and return tail if $U < \frac{2}{3}$.
If we want to simulate die, $U \sim \text{Unif}[0, 1]$. It will return k if $U \in I_k$.
For non-biased die,

Number	1	2	3	4	5	6
Probability	1/6	1/6	1/6	1/6	1/6	1/6

For biased die,

Number	1	2	3	4	5	6
Probability	P1	P2	P3	P4	P5	P6

Then consider the cumulative probability function(cdf), $F(x) = P(X \leq x)$.
$F(x) =$ 0 $x < 1$
$P_1 + P_2$ $1 \leq x < 2$
$P_1 + P_2 + P_3$ $2 \leq x < 3$
$P_1 + P_2 + P_3 + P_4$ $3 \leq x < 4$
$P_1 + P_2 + P_3 + P_4 + P_5$ $4 \leq x < 5$
$P_1 + P_2 + P_3 + P_4 + P_5 + P_6$ $5 \leq x < 6$
The following is the plot of $F(x)$.

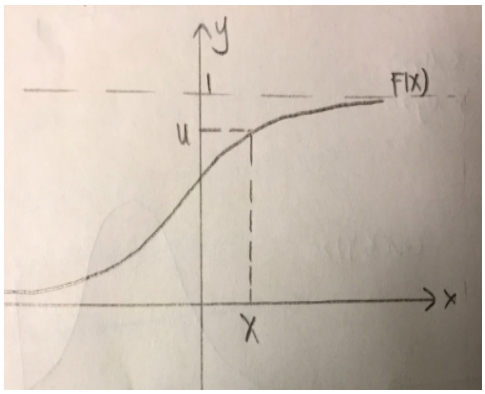


$F(x)$ follows $U \sim \text{Unif}[0,1]$, so we can get $x = F^{-1}(u)$.

2.2.2 Continuous Random Variables

There is a continuous random variable X with probably density function (pdf) $f(x)$.

$$f(x) = \lim_{\triangle x \to 0} \frac{P(X \in (x, x + \triangle x))}{\triangle x}$$

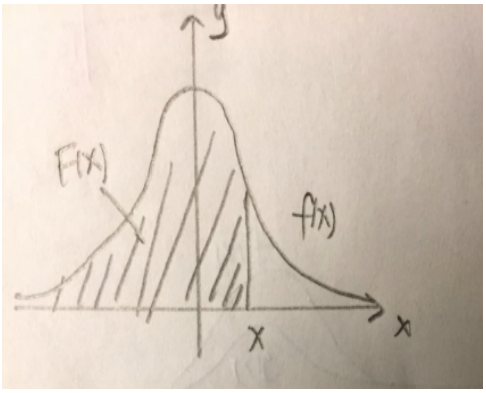


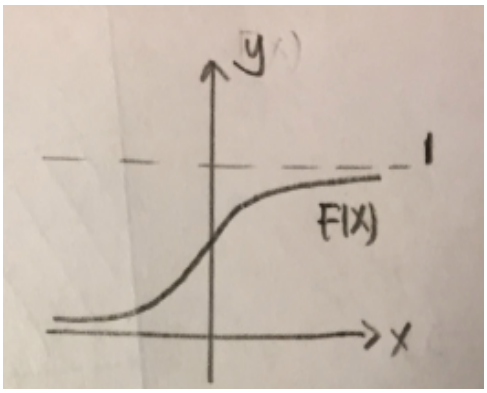
$P(X \in (x, x + \triangle x)$ means how often $X \in (x, x + \triangle x)$, so the frequency of $X \in (x, x + \triangle x) = f(x)\ \triangle x$.
$F(x)$ is the commutative density function (*cdf*) of X.

$$F(x) = P(X \leq x) = \int_{-\infty}^{x} f(x)\, dx.$$

The relationship between *pdf* $f(x)$ and *cdf* $F(x)$ is

$$\frac{dF(x)}{dx} = f(x)$$





Proof

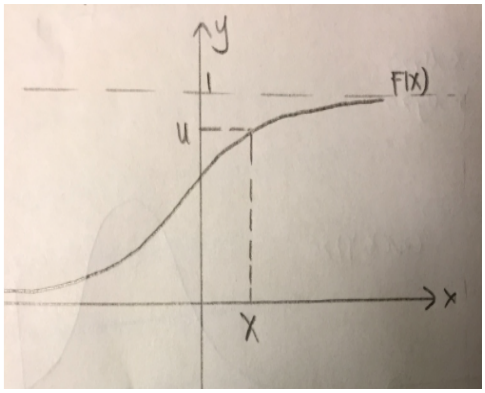
$$f(x) = \frac{dF(x)}{dx} = \lim_{\triangle x \to 0} \frac{F(x+\triangle x) - F(x)}{\triangle x} = \lim_{\triangle x \to 0} \frac{P(X \in (x, x+\triangle x))}{\triangle x} = f(x)$$

Note: Geometrically, $f(x)$ is the slope of $F(x)$.

The basic idea of Inversion method is $U \in \text{Unif}\{0, \frac{1}{M}, \frac{2}{M}, ...\frac{M-1}{M}\}$, and

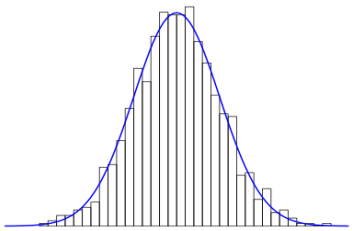
$x \in \text{Unif}\{F^{-1}(0), F^{-1}(\frac{1}{M}), F^{-1}(\frac{2}{M}), ...F^{-1}(\frac{M-1}{M})\}$, so $P(\text{returned} X \leq x) = P(u \leq F(x))$.

Therefore, if we want to generate random X which follows $f(x)$, we can let $U \sim \text{Unif}[0,1]$, and get $x = F^{-1}(u)$ by solving $F(x) = u$, then we can get some random $X \sim f(x)$



We can also understand this method by the plot and histogram. Thinking about a population of numbers or points, and we get a large sample of numbers or points. The population size is 300 million points, denoted as M.

The following is the plot of distribution of points. we can find x follows $f(x)$.



$$f(x) = \frac{\text{number of points in}(x, x+\triangle x)/M}{\triangle x}$$

This is a histogram of M points. Each bins includes discrete points. The total area of all bins equal to 1. Population in $(x, x+\triangle x) = f(x) = x$
In conclusion, if we want to get $X \sim f(x)$,the algorithm of inversion method is
(1). try to get several u and $U \sim \text{Unif}[0,1]$
(2). $X = F^{-1}(u)$ since $u = F(x)$.(u is the proportion of points in front of u, and $F(x)$ is the proportion of points in front of x.)

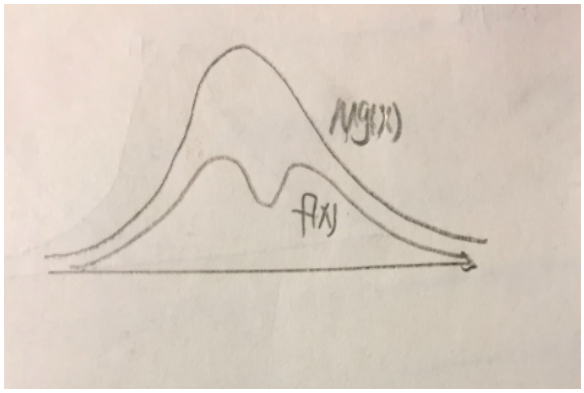
Example:$X \sim \exp(\lambda)$,$f(x) = \lambda e^{-\lambda x}$ when $x \geq 0$; otherwise,$f(x) = 0$.
Solution:$F(x) = \int_0^x \lambda e^{-\lambda x}\, dx = -e^{-\lambda x}|_x^0 = 1 - e^{-\lambda x}$. Solve for $F(x) = u$,and then

$$x = \frac{\log(1-u)}{-\lambda}$$

By inversion method algorithm, firstly, we should generate $U \sim \text{Unif}[0,1]$. Secondly, return $x = \frac{\log(1-u)}{-\lambda}$.
If $x = 1$, $x = -\log(u)$, which follows exp(1).

2.3 Acceptance-Rejection Method

The goal of Acceptance-rejection method is to generate X following target distribution $f(x)$ Sometimes, it is easy to generate X following the envelope distribution $g(x)$, that $f(x) \leq Mg(x)\ \forall x$.



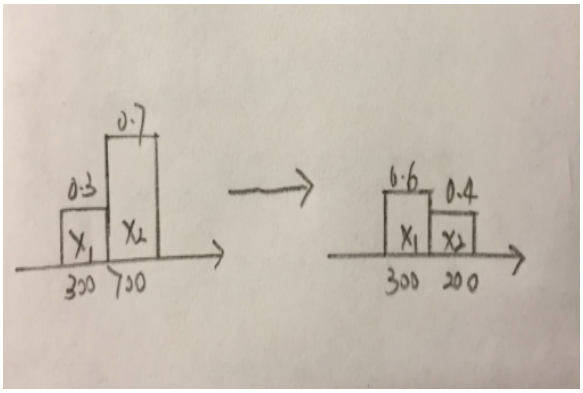
Let us consider an example. Assume we have 1000 points which follows $f(x)$ but we cannot generate $X \sim f(x)$ directly. Therefore, we try an alternative way to generate some $X \sim g(x)$, the following is the pdf

of $g(x)$.

$$g(X_1) = 0.3$$

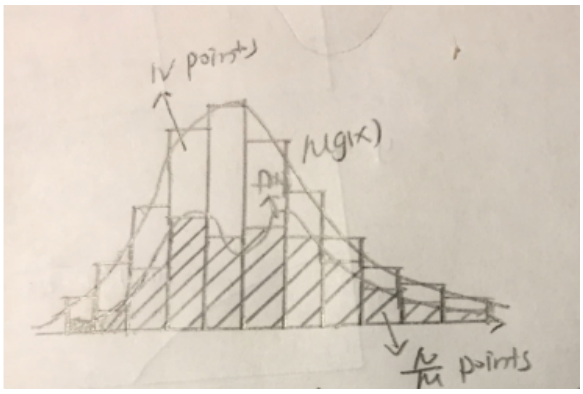
$$g(X_2) = 0.7$$

Based on the *pdf* of the $g(x)$, we can get that there are 300 points equals to X_1 and 700 points equals to X_2. let us assume M=2 so that $Mg(x) \geq f(x) \forall x$. So $Mg(x) = (0.6, 1.4)$. Further, we can also assume that the probability for $X1$ and $X2$ are 0.6 and 0.4, respectively.



Then we will accept x with probability $\frac{f(x)}{Mg(x)}$. so we will get 300 points that equals to X_1 and 200 points equals to X_2.

The algorithm of acceptance-rejection method for this example is
(1) generate $X \sim g(x)$
(2)if $x = X_1$, accept and return X_1. If $x = X_2$, accept X_2 with probability $\frac{2}{7}$ (reject X_2 with probability $\frac{5}{7}$). If accept, return X_2; if reject, back to (1).
By calculating the acceptance rate, the acceptance rate $= \frac{500}{1000} = \frac{1}{2} = \frac{1}{M}$
More general for the second step.
(2) Accept x with probability $= \frac{f(x)}{Mg(x)}$. If accept, return x; if reject, go back to (1), or we can generate $U \sim \text{Unif}[0, 1]$, return X if $U \leq \frac{f(x)}{Mg(x)}$; otherwise, go back to (1).
Note: since $M \geq \frac{f(x)}{g(x)}$ for all x, $M \geq \max \frac{f(x)}{g(x)}$, so the smallest $M = \max \frac{f(x)}{g(x)}$
Let repeat sample $X \sim g(x)$ for n times to get $X_1, ...X_n$.
(1)The total number of points in $(x, x + \triangle x) = Ng(x) \triangle x$
and $f(x) = \frac{P(X \in (x, x + \triangle x))}{\triangle x}$
(2)How many points are accepted in $(x, x + \triangle x)$.See the following graph:



Answer: probability of acceptance for $X \in (x, x+\triangle x) = \dfrac{f(x)}{Mg(x)}$.

Then, the number of points survived in $(x, x+\triangle x) = Ng(x) \triangle x \times \dfrac{f(x)}{Mg(x)} = \dfrac{N}{M} f(x) \triangle x$

Across all bins, number of points accepted $= \sum\limits_{\text{bins}} \dfrac{N}{M} f(x) \triangle x = \dfrac{N}{M} \sum\limits_{\text{bins}} f(x) \triangle x = \dfrac{N}{M}$

So the acceptance rate $\dfrac{\dfrac{N}{M}}{N} = \dfrac{1}{M}$.

Among all the accepted points, what is the frequency of points in $(x, x+\triangle x)$ is

$$\frac{\dfrac{N}{M} f(x) \triangle x}{\dfrac{N}{M}} = f(x) \triangle x$$

, which means our accepted points X following $f(x)$ as desired.

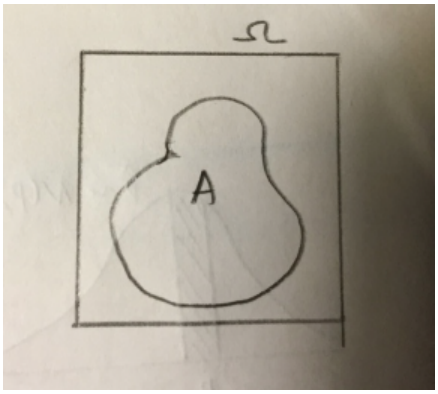
Then we will use another method to proof that those accepted x follows $f(x)$.
proof:Assume $X \sim g(x)$ such that $P(X \in (x, x+\triangle x)) = g(x) \triangle x$.

$P(X \in (x, x+\triangle x) \mid \text{Acceptance}) = P(A \mid B) = \dfrac{P(A \cap B)}{P(B)} = \dfrac{P(A)P(B \mid A)}{P(B)}$

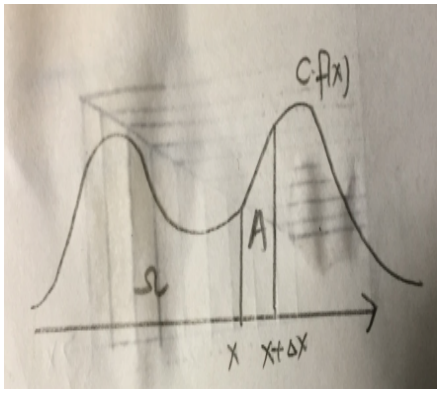
$$= \frac{g(x) \triangle x \dfrac{f(x)}{Mg(x)}}{\sum\limits_{bins} g(x) \triangle x \dfrac{f(x)}{Mg(x)}} = \frac{\dfrac{1}{M} f(x) \triangle x}{\dfrac{1}{M} \int f(x) dx} = f(x) \triangle x$$

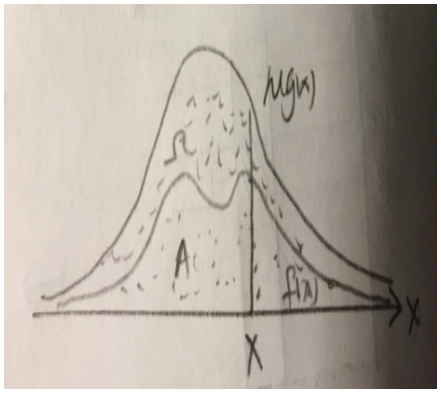
Note: A is $X \in (x, x+\triangle x)$, B is accept X.
Therefore, we have proved that $P(\text{accepted} X \in (x, x+\triangle x)) = f(x) \triangle x$

There is another example, our target distribution is Unif(A). The algorithm is
(1) generate $(X, Y) \sim \text{Unif}(\Omega)$
(2) accept (X, Y) if $(X, Y) \in A$; otherwise, reject (X, Y) and go back to (1)



Assume C is a constant. $(X, Y) \sim \text{Unif}(\Omega)$ and $X \sim f(x)$, then $P(X \in (x, x + \triangle x)) = \frac{|\ A\ |}{|\ \Omega\ |} = \frac{cf(x) \triangle x}{\int cf(x)dx} = \frac{cf(x) \triangle x}{c} = f(x) \triangle x$





compared with the original algorithm, the algorithm here is
(1) $(X, Y) \sim \text{Unif}(\Omega)$, and $X \sim g(x)$, then $Y \mid x \sim \text{Unif}[0, Mg(x)]$
(2)$f(x) \cdot U * Mg(x)$, so the acceptance rate $= \frac{|\ A\ |}{|\ \Omega\ |} = \frac{\int f(x)dx}{M \int g(x)dx} = \frac{1}{M}$

2.4 Transformation Method

There is one special case for inversion method:

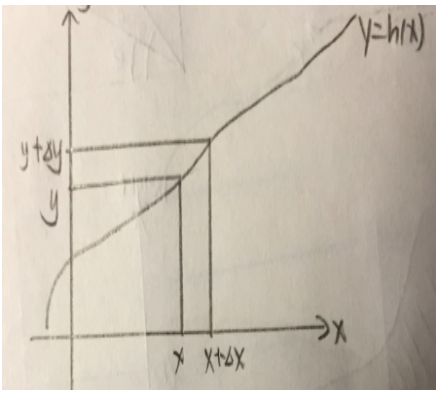
$$U \sim \text{Unif}(0, 1)$$

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

However, sometimes $F^{-1}(U)$ is difficult to compute sometimes$F^{-1}(U)$.Therefore, we can let $X \sim f_X(x)$ and $Y \sim h(x)$, then $Y \sim f_Y(y)$. But what is f_Y is? we will introduce two transformation methods to explain this question.

2.4.1 Non-linear and Linear Transformation

Suppose $y = h(x)$ is monotone increasing, so $x = g(y) = h^{-1}(y)$.



Then,

$$f_X(x) = \frac{P(X \in (x, x + \triangle xx))}{\triangle x}$$

$$f_Y(y) = \frac{P(Y \in (y, y + \triangle y))}{\triangle y} = \frac{P(h(x) \in (y, y + \triangle y))}{\triangle y} = \frac{P(X \in (g(y), g(y + \triangle y)))}{\triangle y}$$

$$= \frac{f_X(g(y))(g(y + \triangle y) - g(y))}{\triangle y} = f_X(g(y))g^{-1}(y) = f_Y(y)$$

There is another way to proof.

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

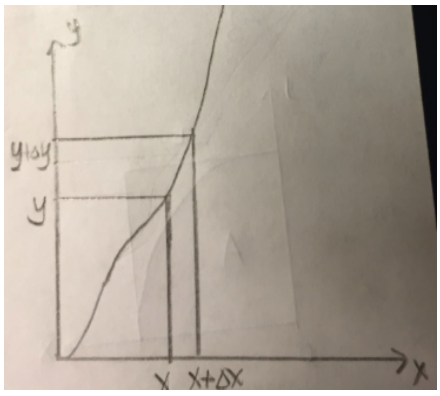
$$\frac{dF_Y(y)}{dy} = f_Y(y)$$

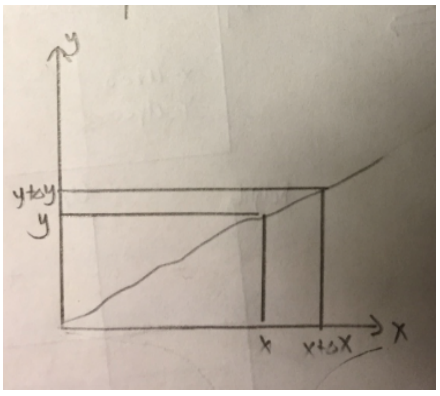
$$f_Y(y) = P(Y \leq y) = P(h(x) \leq y) = P(X \leq g(y)) = F_X(g(y))$$

Using chain rule,

$$f_Y(y) = \frac{dF_Y(g(y))}{dy} = f_X(g(y))g^{-1}(y)$$

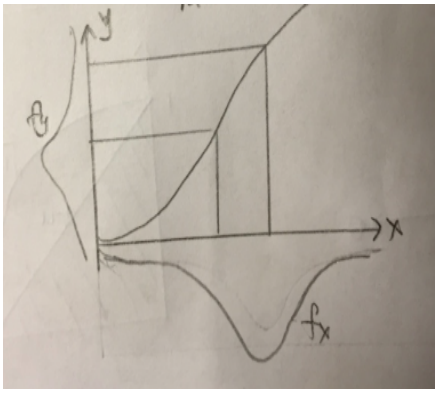
Space mapping: see the following two graphs, one is stretching and another one is squeezing after the linear transformation. When stretching, the density becomes smaller, and when squeezing, the density becomes larger.





Let us introduce the space mapping factor: $\frac{\triangle y}{\triangle x} = h^{'}(x)$ and $\frac{\triangle x}{\triangle y} = g^{'}(y)$.

Then suppose we have many population points. If we do the linear transformation by $y = h(x)$, what happened to the density?



$$f_Y(y) = \frac{\text{number of people in}(y, y+\triangle y)}{\triangle y}$$

$$f_X(x) = \frac{\text{number of people in}(x, x+\triangle x)}{\triangle x}$$

$$\frac{f_Y(y)}{f_X(x)} = \frac{\triangle x}{\triangle y} = g^{'}(y)$$

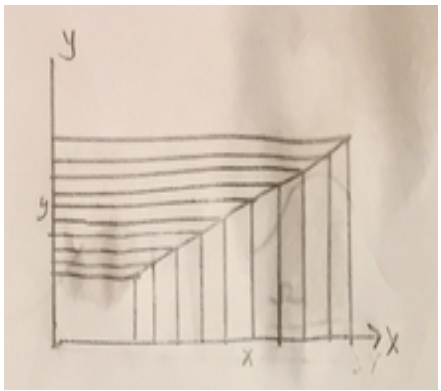
So we can say that the change of density only determined by the size of neighbourhood. $f_Y(y)f_X(g(y))g^{-1}(y)$, if $g^{'}(y)$ is large, the graph is squeezing to make the bins smaller and the density becomes larger; it $g^{'}(y)$ is small, the graph is sketching to make the bins bigger and the density becomes smaller.
Symbolically, $X \sim f_X(x)dx \sim f_X(g(y))d(g(y)) \sim f_X(g(y))g^{'}(y)dy \sim f_Y(y)$
There is a simple **example**.
$T \sim \exp(1)$ and $R = \sqrt{2T}$, so what is the distribution of R?
Solution:

$$R = \sqrt{2T}, T = \frac{R^2}{2}$$

$$T \sim e^{-t}dt \sim e^{-\frac{r^2}{2}}d(\frac{r^2}{2}) \sim e^{-\frac{r^2}{2}}rdr$$

so the density function of R is $f_R(r) = e^{-\frac{r^2}{2}}r$

Then consider a **linear transformation**.
If we project x to y, the density of points on y becomes bigger if we have a small slope of y vs x; the density of points on y becomes smaller if we have a big slope of y vs x. See the following graphs.



$f_Y(y)$ = density at y = density at x $\dfrac{1}{\text{slope at x}} = f_X(x)\dfrac{\triangle x}{\triangle y}$

If we have a linear transformation $y = ax + b$, then $f_Y(y) = f_X(x)\dfrac{\triangle x}{\triangle y} = f_X(\dfrac{y-b}{a}) \times \dfrac{1}{a}$, so we can see the density change under linear transformation.

Actually, the non-linear transformation and linear transformation is same, since the tangent line of the points on the curve is linear.

2.4.2 Polar Transformation

First of all, let us generate two copies of independent random variable, X,$Y \overset{iid}{\sim} N(0,1)$. $f_X(x) = \dfrac{1}{\sqrt{2}\pi}e^{-\dfrac{x^2}{2}}$ and $f_Y(y) = \dfrac{1}{\sqrt{2}\pi}e^{-\dfrac{y^2}{2}}$

If A and B are independent, the joint distribution of A and B equals the product of marginal probability. $P(A \cap B) = P(A)P(B)$

So, the disjoint distribution for X, Y is

$$f(x,y) = \frac{P(X \in (x, x+\triangle x, Y \in (y, y+\triangle y))}{\triangle x \triangle y} = \frac{P((x,y) \in D)}{\mid D \mid}$$

$$= \frac{P(X \in (x, x+\triangle x))}{\triangle x}\frac{P(Y \in (y, y+\triangle y))}{\triangle y} = f_X(x)f_Y(y)$$

$$= \frac{1}{\sqrt{2}\pi}e^{-\frac{x^2}{2}}\ \frac{1}{\sqrt{2}\pi}e^{-\frac{y^2}{2}} = \frac{1}{2\pi}e^{-\frac{x^2+y^2}{2}}$$

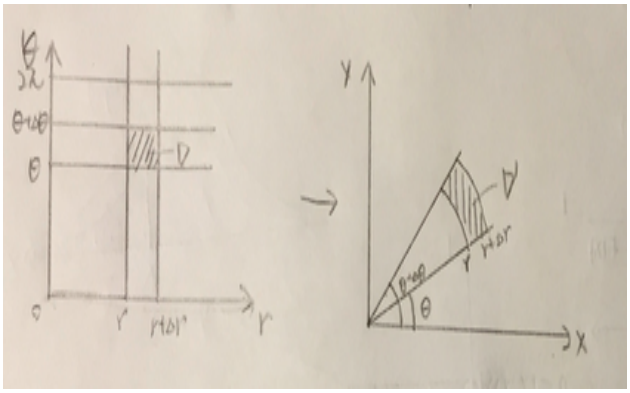
Then we will do the polar transformation. We assume that $X = R\cos(\theta)$ and $Y = R\sin(\theta)$,then

$$f(r,\theta) = f(x,y)\frac{dxdy}{drd\theta} = \frac{1}{2\pi}e^{-\frac{r^2}{2}}\frac{dxdy}{drd\theta}$$

Then we will calculate the Jocobian

$$r = \begin{vmatrix} \frac{dx}{dr} & \frac{dx}{d\theta} \\ \frac{dy}{dr} & \frac{dy}{d\theta} \end{vmatrix} = \begin{vmatrix} \cos(\theta) & -r\sin(\theta) \\ \sin(\theta) & r\cos(\theta) \end{vmatrix} = r\cos^2(\theta) + r\sin^2(\theta) = r$$

So, the $f(r,\theta) = \dfrac{1}{2\pi}e^{-\dfrac{r^2}{2}}r$



There is another way to think about it.

$$f(r,\theta)=\frac{P((R,\theta)\in(r,r+\triangle r)\times(\theta,\theta+\triangle\theta))}{\mid D\mid}=\frac{P((R,\theta)\in D)}{\mid D\mid}$$

$$=\frac{P((X,Y)\in D^{'})}{\mid D\mid}=\frac{r\times P((X,Y)\in D^{'})}{\mid D^{'}\mid}=rf(x,y)=\frac{1}{2\pi}e^{-\frac{r^2}{2}}r$$

Note:$\frac{\mid D^{'}\mid}{\mid D\mid}=\frac{r\triangle\theta\triangle r}{\triangle r\triangle d}=r$

Now consider an **example** about how we can use polar transformation to generate $N(0,1)$ random variable.

$\theta\sim$ Unif$[0,2\pi]$, So $\theta=2\pi u_1$.Let $R\sim\theta^{-\frac{r^2}{2}}d\frac{r^2}{2}$,then let $T=\frac{r^2}{2}$ such that $T\sim\exp(1)$. So $T=-\log(u_2)$ and $R=\sqrt{2T}=\sqrt{-2\log(u_2)}$. Then we can use our R and θ to generate random variable $X,Y\sim N(0,1)$ by using $X=R\cos(\theta),Y=R\sin(\theta)$.

Note:$u_1,u_2\sim$ Unif$(0,1)$

Then we can prove the polar transformation method we use for this example.

$$(R,\theta)\sim\frac{1}{2\pi}e^{-\frac{r^2}{2}}rdrd\theta=\frac{1}{2\pi}d\theta\times e^{-\frac{r^2}{2}}rdr$$

$$\text{let}u_1=d(\frac{\theta}{2\pi}),u_2=d(e^{\frac{-r^2}{2}})$$

$$\int\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}dx\times\int\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}dy=\int\frac{1}{2\pi}e^{-\frac{x^2+y^2}{2}}dxdy$$

$$=\int\frac{1}{2\pi}e^{-\frac{r^2}{2}}rdrd\theta=\int_0^{2\pi}\frac{1}{2\pi}d\theta\int_0^{\infty}e^{\frac{-r^2}{2}}rdr$$

$$=\int_0^{\infty}e^{\frac{-r^2}{2}}rd\frac{r^2}{2}=-e^{-\frac{r^2}{2}}\mid_0^{\infty}=1$$

3 Monte Carlo Integration Introduction

$I=\int h(x)f(x)dx=E(h(X))$ where $X\sim f(x)$

Generate $X_1,X_2,...,X_n\overset{i.i.d}{\sim}f(x)$

$$\hat{I}=\frac{1}{n}\sum_{i=1}^{n}h(Xi)\xrightarrow{n\to\infty}I$$

sample average $\xrightarrow{\text{Law of Large Number}}$ expectation

example 1

e.g. $I=\int_{\infty}^{-\infty}\sqrt{|x|}\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}dx$

$$h(x)=\sqrt{|x|}$$

$$f(x)\sim\text{N}(0,1)$$

Generate $X_1, X_2, ..., X_n \overset{i.i.d}{\sim} \mathrm{N}(0,1)$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n}\sqrt{|x|}$$

example 2

$I = \int_0^1 (\cos 20x + \sin 50x)^2 dx$

$$f(x) \sim \mathrm{Unif}[0,1]$$

Generate $X_1, X_2, ..., X_n \overset{i.i.d}{\sim} \mathrm{Unif}[0,1]$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n}(\cos 20x + \sin 50x)^2$$

3.1 Monte Carlo Method Essential

To simulate from a certain distribution and to use sample average to approximate true expectation.

example 3

$$I_{|x|>3} = \begin{cases} 1 & |x| > 3; \\ 0 & \text{otherwise.} \end{cases}$$

where $X \sim \mathrm{N}(0,1)$

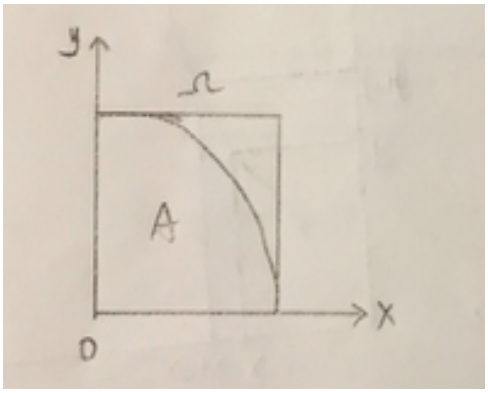
$$I = \int I_{|x|>3}\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}dx = E(I_{|x|>3}) = P(|x|>3) = \int_{|x|>3} f(x)dx$$

Algorithm

Generate $X_1, X_2, ..., X_n \overset{i.i.d}{\sim} \mathrm{N}(0,1)$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n}|Xi| > 3 = \frac{1}{n}(\text{number of times } |Xi| > 3 = \text{frequency } |Xi| > 3 \xrightarrow[n\to\infty]{\text{converge}} P(|x|>3)$$

3.2 Calculating π



Ω = unit square randomly throw a point intoΩ

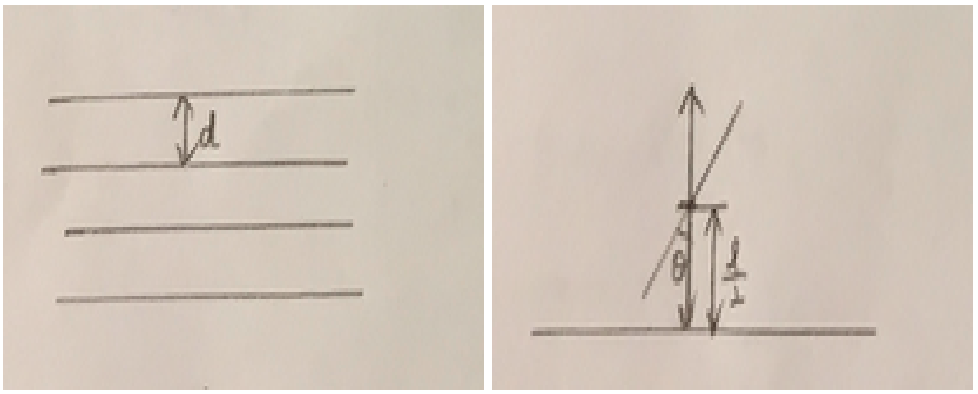
$$P(\text{point falls into A}) = \frac{|A|}{|\Omega|} = \frac{\pi/4}{1} = \frac{\pi}{4}$$

Algorithm
Generate $(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n) \overset{i.i.d}{\sim} \text{Unif}[0, 1]$

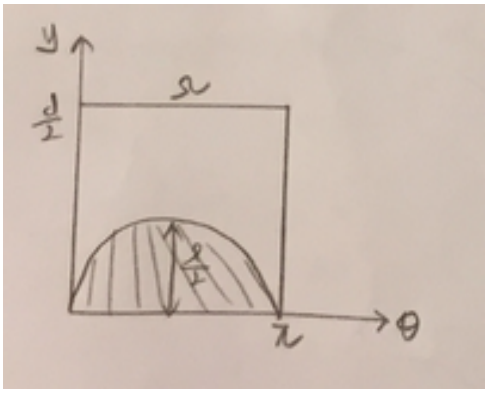
$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} 1_{(X,Y)\in A} = \frac{1}{n}\sum_{i=1}^{n} 1_{(X_1^2+Y_1^2\leqslant 1)} \approx \frac{\pi}{4}$$

$$\pi \approx 4\hat{I}$$

3.3 Buffon's Needle



l is the length of the needle; d is the width of line. Given $l < d$, how often would the needles cross the line? Calculating the distance between the center of the needle and the closest line. distance $Y \in [0, \frac{d}{2}]$ angle $\Theta \in [0, \pi]$ $Y \sim \text{Unif}[0, \frac{d}{2}]$ verticle drop $= \frac{l}{2}\sin\theta$ $\Theta \sim \text{Unif}[0, \pi]$ touches line when $y \leqslant \frac{l}{2}\sin\theta$



$$P(A) = \frac{|A|}{|\Omega|} = \frac{\int_0^\pi \frac{l}{2}\sin\theta d\theta}{\pi\frac{d}{2}} = \frac{2l}{\pi d} \approx \text{frequency}$$

$$\pi \approx \frac{d \cdot \text{freqency}}{2l}$$

Mano Luzzarin's Experiment He did an experiment that he claimed he threw 3408 times and get $\pi \approx \frac{355}{113}$ within $3 * 10_{-7}$. However, it is too good to be true.

Algorithm $\hat{I} = \frac{1}{n}\sum_{i=1}^{n} h(Xi) \xrightarrow{n\to\infty} I$ (Law of Large Number)

Question: how do we quantify the fluctuations?
Note: the randomness of $\hat{I}$ comes from the randomness of X_i.
We first get a single run $X_1, X_2, ..., X_n \to \hat{I}_1$.
Then we get another run $X_1, X_2, ..., X_n \to \hat{I}_2$.
Under hypothetical repeated runs, $\hat{I}$ fluctuates.
The center of fluctuation is $E(\hat{I}) = E(\frac{1}{n}\sum_{i=1}^{n} h(Xi)) = \frac{1}{n}\sum_{i=1}^{n} E(h(Xi)) = \frac{1}{n}\sum_{i=1}^{n} E(h(X)) = E(h(x)) = I$.
So $\hat{I}$ is unbiased.

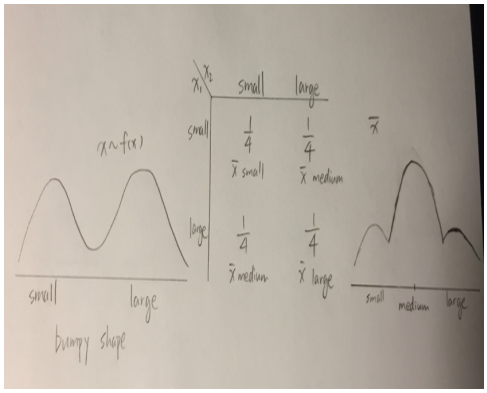
$$Var(\hat{I}) = Var(\frac{1}{n}\sum_{i=1}^{n} h(Xi)) = \frac{Var(\sum_{i=1}^{n} h(Xi))}{n^2}$$

Because $h(Xi)$'s are independent.

$$Var(\hat{I}) = \frac{\sum_{i=1}^{n} Var(h(Xi))}{n^2} = \frac{\sum_{i=1}^{n} Var(h(X))}{n^2} = \frac{n \cdot Var(h(x))}{n^2} = \frac{Var(h(x))}{n} \xrightarrow{n\to\infty} 0$$

This implies $\hat{I} \to I$.

Explanation
$x \sim f(x)$



bumpy shape → smoother shape
variances become smaller

3.4 Central Limit Theorem

$$E(\hat{I}) = I$$

$$Var(\hat{I}) = \frac{Var(h(x))}{n} = \frac{v}{n}$$

error has nothing to do with the dimensionality of X C.L.T. $\hat{I} \sim N(I, \frac{v}{n})$ I's fluctuate around true values $\sqrt{n}(\hat{I} - I) \to N(0, v)$

Buffon's needle
$I = P(\text{needle touches line}) = \frac{2l}{d\pi} = E(1_{touch}) = P \sim$

Bernoulli distribution
$v = p(1-p)$

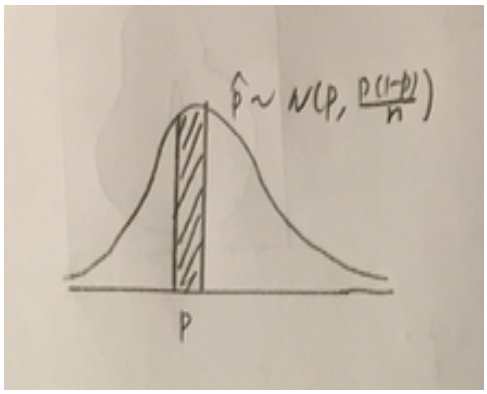
$$\hat{I} \to \mathrm{N}(p, \frac{p(1-p)}{n})$$

$$sd(\hat{I}) = \sqrt{\frac{p(1-p)}{n}}$$

$$se(\hat{I}) = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

95 percent C.I. for $p = \frac{2l}{d\pi} = I$ is $\hat{P} \pm 1.96 \cdot se(\hat{I})$ meaning the probability of the confidence interval captures p is 0.95.

Manu Luzzarin's Experiment



p-value = P(observe more extreme values) = $P(\left|\hat{P} - P\right| < \varepsilon_{ds})$
The probability of getting Manu Luzzarin's experiment result is too small.

3.5 Variance Reduction methods

1) Variances determine the efficiency of Monte Carlo Methods.
2) Variances only depend on the samples size, not the dimensionality of X.

3.5.1 Probability Preparation

$(X, Y) \sim f(x, y)$

$$X \sim f_X(x) = \int f(x, y)dy$$

$$Y \sim f_Y(y) = \int f(x, y)dx$$

Conditioning $f(y \mid x) = \frac{f(x,y)}{f_x(x)}$ Recall Probability $P(A \mid B) = \frac{P(A \cap B)}{P(B)}$

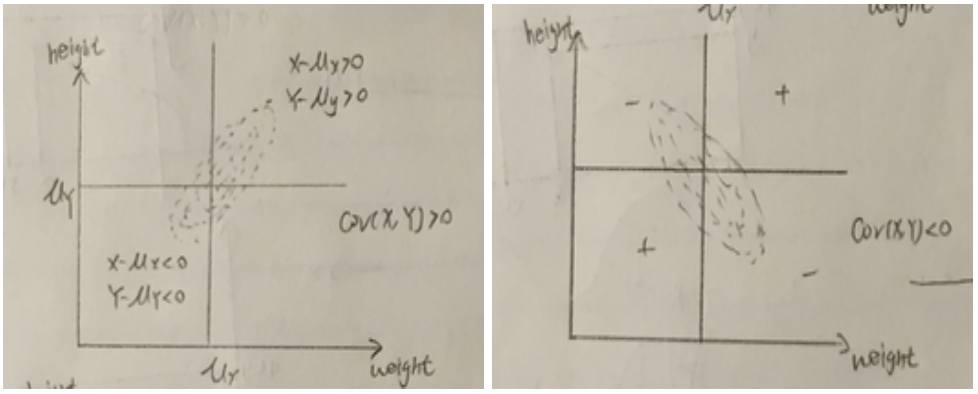
$$E(Y \mid X = x) = \int yf(y \mid x)dy$$

denote

$$E(Y \mid X = x) = h(x)$$

$Cov(X, Y) = E[(X - \mu_X)(Y - \mu_Y)]$ where $\mu_X = E(X)$ and $\mu_Y = E(Y)$.

$$Var(X + Y) = Var(X) + Var(Y) + 2Cov(X, Y)$$



Proof

$$Var(X + Y) = E[(X + Y) - E(X + Y)]^2 = E[(X + Y) - [E(x) + E(Y)]] = E([X - E(X)] + [Y - E(Y)])^2$$

$$= E([X - E(X)]^2 + [Y - E(Y)]^2 + 2[X - E(X)][Y - E(Y)]) = Var(X) + Var(Y) + 2Cov(X, Y)$$

3.5.2 Conditional Expectation and Variance

Adam Formula

$$E(E(Y \mid X0) = E(Y) \Leftrightarrow E(h(X)) = E(Y)$$

i.e. change Y to h(X) for each person but does not change the average (sum within all age groups).
Let $Y = h(x) + \varepsilon$ $E(\varepsilon) = 0$ overall deviation is 0 because $E(\varepsilon \mid X = x) = 0$

$$\sum_{i=1}^{n}(X_i - \bar{X}) = (X_1 - \bar{X}) + (X_2 - \bar{X}) + ... + (X_n - \bar{X}) = \sum_{i=1}^{n} X_i - n\bar{X} = 0$$

$$E(\varepsilon h(x)) = E(\varepsilon h(x) \mid X) = 0$$

$h(x)$ is a constant so that $E(\varepsilon h(x) \mid X = x) = h(x)E(\varepsilon \mid X = x) = 0$ where $E(\varepsilon \mid X = x) = 0$

$$Var(Y) = Var(h(x) + \varepsilon) = Var(h(x)) + Var(\varepsilon) + 2Cov(h(x), \varepsilon)$$

where $Var(\varepsilon) = (E(\varepsilon))^2 + E(\varepsilon^2) = E(\varepsilon^2) = E(E(\varepsilon^2 \mid X)) = E(Var(Y \mid X))$
$Cov(h(x), \varepsilon) = E(h(x)\varepsilon) - E(h(x))E(\varepsilon) = 0$ because $E(h(x)\varepsilon) = E(\varepsilon) = 0$

EVE Formula

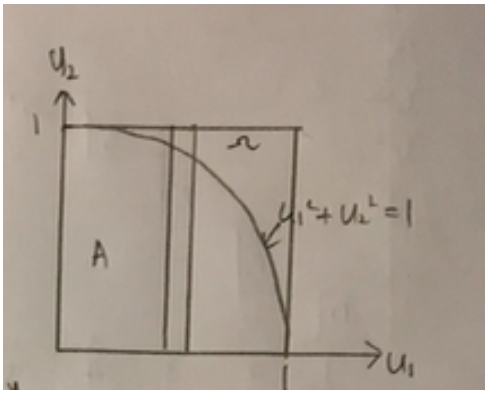
$$Var(Y) = E(Var(Y \mid X)) + Var(E(Y \mid X))$$

where $E(Y \mid X) = h(x)$
total variance = within group variances + between group variances

$$Var(h(x)) \leqslant Var(Y)$$

In this way, we reduce variances by changing Y into expectation.

Example



$$1_A(u_1,u_2)=\begin{cases}1 & u_1^2+u_2^2\leqslant 1;\\ 0 & \text{otherwise.}\end{cases}$$

where $u_1,u_2 \overset{i.i.d}{\sim} \text{Unif}[0,1]$

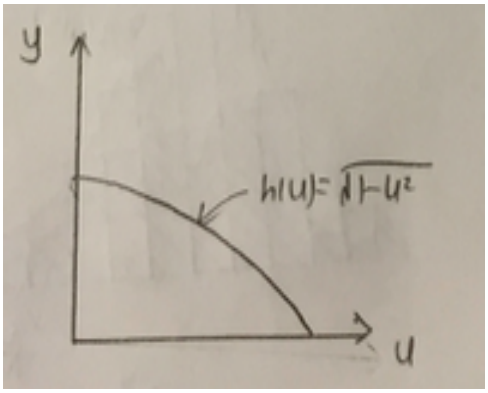
$$u_1^i,u_2^i \overset{i.i.d}{\sim} \text{Unif}[0,1]$$

$$\frac{\pi}{4}=P((u_1^i,u_2^i)\in A)=E(1_A(u_1,u_2))\approx\frac{1}{n}\sum_{i=1}^{n}1_A(u_1^i,u_2)$$

Conditioning

$$E(1_A(u_1,u_2)\mid U_1=u^1)=P((u_1,u_2)\in A\mid U_1=u^1)=P(u_1\leqslant\sqrt{1-u_1^2})=\sqrt{1-u_1^2}$$

3.5.3 Different Monte Carlo Method



$$u_1^i \overset{i.i.d}{\sim} \text{Unif}[0,1]$$

$$\frac{\pi}{4}\approx E(\sqrt{1-u_1^2})\approx\frac{1}{n}\sum_{i=1}^{n}\sqrt{1-u_1^{i2}}$$

$$\frac{\pi}{4}\approx E(\sqrt{1-u^2})\approx\frac{1}{n}\sum_{i=1}^{n}\sqrt{1-u_i^2}$$

$$E(\sqrt{1-u^2})=\int_0^1\sqrt{1-u^2}du=\frac{\pi}{4}$$

Advange:
1) Variance is smaller.

2) Generating only one random variables requires less time.
3) If we know the exact formula then we could reduce randonmization. In other words, we throw away within group variances.

3.5.4 Stratified Sampling

$$(u_1^i, u_2^i) \overset{i.i.d}{\sim} \text{Unif}[0,1] i = 1, ...300$$

$$P(u_1 \in I_k) = \frac{1}{3} \approx \text{freq}(u_1^i \in I) \text{where} k = 1, 2, 3$$

$$i = 1, ...100 (u_1^i \sim \text{Unif}[0, \frac{1}{3}]$$

$$i = 101, ...200 (u_1^i \sim \text{Unif}[\frac{1}{3}, \frac{2}{3}]$$

$$i = 201, ...300 (u_1^i \sim \text{Unif}[\frac{1}{3}, 1]$$

$$\frac{\pi}{4} \approx \text{freq}[(u_1^i, u_2^i) \in A]$$

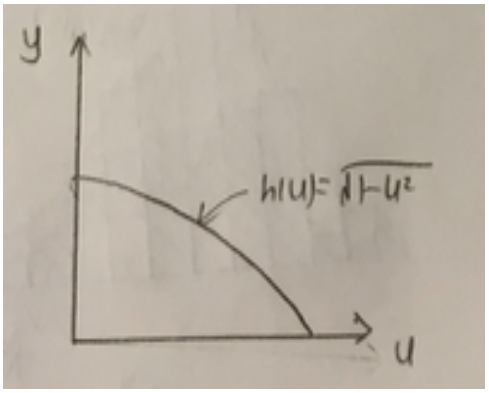
$$Y = I_A(u_1, u_2)$$

$X = k$ if $(u_1, u_2) \in I_k$

$$E(Y) = E(E(Y \mid X)) = E(Y \mid X = 1) \times \frac{1}{3} + E(Y \mid X = 2) \times \frac{1}{3} + E(Y \mid X = 3) \times \frac{1}{3}$$

This is to say to maintain the probability of each stratum.
Advantage: $Var(Y) = E(Var(Y \mid X) + Var(E(Y \mid X))$
In this way, we reduce variability by fixing things to their theoretical value.

3.5.5 Antithetic Variables



$$\frac{\pi}{4} = \int_0^1 \sqrt{1 - u^2} du \approx \frac{1}{n} \sum_{i=1}^{n} \sqrt{1 - u_1^2}$$

$$\text{if} u_i \sim \text{Unif}[0,1]$$

$$\text{then} \tilde{u_i} \sim \text{Unif}[0,1]$$

The integral $\approx \frac{1}{2n}\sum_{i=1}^{n}(\sqrt{1-u_i^2}+\sqrt{1-\tilde{u_i}^2})$
This is to say we generate one copy and then flip it to get another copy.

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} h(Xi)$$

$$E(\hat{I}) = E(h(x))$$

$$Var(\hat{I}) = \frac{Var(h(x))}{n}$$

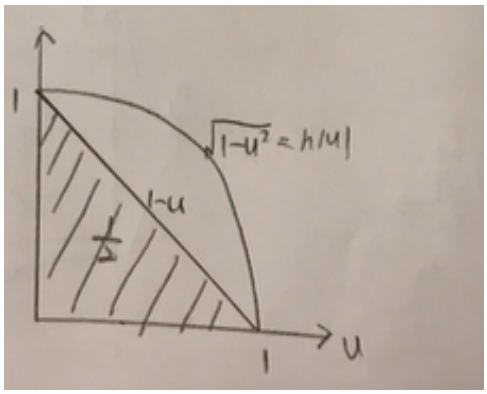
$$\hat{I_2} = \frac{1}{2n}\sum_{i=1}^{n}[h(Xi)+h(\tilde{X}i)]$$

$$E(\hat{I_2}) = \frac{1}{2n}\sum_{i=1}^{n}[E(h(Xi))+E(h(\tilde{X}i))] = E(h(x))$$

$$Var(\hat{I_2}) = \frac{1}{4n^2}\sum_{i=1}^{n}[Var(h(Xi))+Var(h(\tilde{X}i))+2Cov((h(Xi),h(\tilde{X}i))] = \frac{Var(h(x))}{2n}+\frac{\sum_{i=1}^{n} Cov((h(Xi),h(\tilde{X}i))}{4n^2}$$

$Var(h(Xi))+Var(h(\tilde{X}i)) = 2Var(h(Xi))$ because they have the same distribution.
$Cov < 0$
because when one is big, the other is small.
So $Var(\hat{I_2}) \leqslant \frac{Var(h(x))}{2n}$
In this way, we use correlated samples instead of i.i.d. to offset variability. (e.g. stock market)

3.5.6 Control Variate



$$\frac{\pi}{4} = \int_0^1 [h(u)-h_0(u)]du + \frac{1}{2}$$

where there is no fluctuation in $\frac{1}{2}$

$$\frac{\pi}{4} \approx \sum_{i=1}^{n}[h(u_i)-h_0(u_i)] = \hat{I_3}$$

$$Var(\hat{I_3}) = \frac{Var[h(u)-h_0(u)]}{n} = \frac{Var(h(u))+Var(h(u_0))-2Cov[h(u)-h_0(u)]}{n}$$

where $Cov[h(u) - h_0(u)]n$ is positive because when one is big, the other is also big Thus we can regress $h(u)$ on $h_0(u)$ to find optimal β to minimize $Var[h(u) - \beta h_0(u)]$.
In this way, we reduce variances by making the flucutation in difference smaller.

4 Importance Sampling

Goal: $I = E_f[h(X)] = \int h(x)f(x)dx\bigstar$
Premise:

f is difficult to sample from e.g. $f(x) = \sqrt{\frac{2}{\pi}}e^{-\frac{x^2}{2}}$

g is easy to sample from e.g. $g(x) = 2e^{-2x}$

Sample $X_1, X_2, ..., X_n \overset{i.i.d.}{\sim} g(x)$

$$I = E_g[h(x)\frac{f(x)}{g(x)}]\bigstar\bigstar$$

$$= \int h(x)\frac{f(x)}{g(x)}g(x)dx = E_g[\tilde{h}(x)]$$

where we denote $\tilde{h}(x) = h(x)\frac{f(x)}{g(x)} = h(x)w(x)$ $\hat{I} \xrightarrow[\text{regular monte carlo}]{} \frac{1}{n}\sum_{i=1}^{n}\tilde{h}(X_i)$

$\xrightarrow[\text{important sampling}]{} \frac{1}{n}\sum_{i=1}^{n}h(X_i)w(X_i)$

where $w_i = w(X_i) = \frac{f(X_i)}{Y_i}$

For each X_i, we attach weight w_i.

$\hat{I} \approx \sum_{i=1}^{n} h(X_i)\frac{w_i}{\sum_{i=1}^{n} w_i}$ where $\frac{1}{n}\sum_{i=1}^{n} w_i \approx 1$

$$E(\hat{I}) = I$$

$Var(\hat{I}) = \frac{Var_g(\tilde{h}(X))}{n} = \frac{Var_g[h(x)w(x)]}{n}$ is the accuracy of importance sampling

Assume $h(X)$ does not change too much, then $Var(\hat{I})$ depends on $Var(w(X))$.

4.1 Property

$$I = E_f(h(x)) = \int h(x)f(x)dx = E_g(h(x)\frac{f(x)}{g(x)}) = \int h(x)\frac{f(x)}{g(x)}dx = Eg(h(x)w(x))$$

where $w(x) = \frac{f(x)}{g(x)}$.

Sample from $X_1, X_2, ..., X_n \sim g(x)$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} h(x_i)w_i$$

where $w_i = w(x_i)$ adjusts over-representation or under-representation.

4.2 Specialization

Let $h(x) = 1$

$$1 = E(1) = E_f(1) = E_g(\frac{f(x)}{g(x)}) = E_g(w(x))$$

$$E_g(w(x)) = 1$$

This means the weight fluctuates around 1.

$$\frac{1}{n}\sum_{i=1}^{n} w_i \approx 1$$

4.3 Normalizing Constant

$$f(x) = \frac{1}{z_f}\tilde{f}(x)$$

$$g(x) = \frac{1}{z_g}\tilde{g}(x)$$

Sometimes, we only know $\tilde{f}(x)$ and $\tilde{g}(x)$ but don't know the normalizing constants z_f and z_g, or one normalizing constant is difficult to calculate.

$$z_f = \int \tilde{f}(x)dx$$

$$z_g = \int \tilde{g}(x)dx$$

Example

$f(x) = \sqrt{\frac{\pi}{2}}e^{-\frac{x^2}{2}}$ so that $\tilde{f}(x) = e^{-\frac{x^2}{2}}$ and $z_f = \sqrt{\frac{\pi}{2}}$

$g(x) = 2e^{-2x}$ so that $\tilde{g}(x) = e^{-2x}$ andt $z_g = \frac{1}{2}$

Normalizing Constant

We know $E_g(w(x)) = 1$ where $w(x)$ is unknown.

$$E_g(\frac{\tilde{f}(x)}{\tilde{g}(x)})\frac{z_g}{z_f}) = 1$$

Define $\tilde{w}(x) = \frac{\tilde{f}(x)}{\tilde{g}(x)}$ which is unknown.

$$E_g(\tilde{w}(x)\frac{z_g}{z_f}) = 1$$

$$\frac{\tilde{f}(x)}{\tilde{g}(x)}E_g(\tilde{w}(x)) = 1$$

$$E_g(\tilde{w}(x)) = \frac{z_f}{z_g}$$

$$z_f = z_g E_g(\tilde{w}(x)) \approx z_g \frac{1}{n}\sum_{i=1}^{n}\tilde{w}(x_i)$$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} h(x_i)w(x_i) = \frac{1}{n}\sum_{i=1}^{n} h(x_i)\tilde{w}(x_i)\frac{z_g}{z_f} \approx \frac{1}{n}\sum_{i=1}^{n} h(x_i)\frac{\tilde{w}(x)}{\frac{1}{n}\sum\limits_{i=1}^{n}\tilde{w}(x)} = \sum_{i=1}^{n} h(x_i)\frac{\tilde{w}(x_i)}{\sum\limits_{i=1}^{n}\tilde{w}_i(x)}$$

Example

$$\tilde{f}(x) = e^{-\frac{x^2}{2}}(x \geqslant 0)$$

$$\tilde{f}(x) = e^{-2x}(x \geqslant 0), z_g = \frac{1}{2}$$

Generate $X_1, X_2, ..., X_n \sim g(x)$

$$\hat{z_f} = \frac{1}{2}\frac{1}{n}\sum_{i=1}^{n}\frac{e^{-\frac{x^2}{2}}}{e^{-2x}}$$

where $\frac{e^{-\frac{x^2}{2}}}{e^{-2x}} = \tilde{w}(x)$

$$I = \int_0^\infty \sqrt{x}f(x)dx = E_f(\sqrt{x})$$

$$\hat{I} = \sum_{i=1}^{n}\sqrt{x_i}\frac{\tilde{w}(x_i)}{\sum\limits_{i=1}^{n}\tilde{w}_i(x)}$$

Gibbs / Boltzmain Distribution

$$P(x) = \frac{1}{z}e^{-\frac{\mathcal{E}(X)}{T}}$$

$T \sim$ Temperature
$\mathcal{E} \sim$ Energy
$X \sim$ Configuration
Z is the unknown constant which we need Monte Carlo methods to calculate.

back to importance sampling

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} h(X_i)w(X_i)$$

$$w(x) = \frac{f(x)}{g(x)} = \frac{\frac{1}{z_f}\tilde{f}(x)}{\frac{1}{z_g}\tilde{f}(x)} = \frac{z_g}{z_f}\frac{\tilde{f}(x)}{\tilde{g}(x)} = \frac{z_g}{z_f}\tilde{w}(x)$$

where $\frac{z_g}{z_f}$ is unknown and $tildew(x)$ is an unormalized density function ratio.

$$E_f w(x) = 1$$

$$E_g[\frac{z_g}{z_f}\tilde{w}(x)] = 1$$

$\frac{z_g}{z_f}E(\tilde{w}(x)) = 1$ where $\frac{z_g}{z_f}$ is a constant and we denote it as ρ.

$$\hat{\rho} = \frac{1}{\frac{1}{n}\sum_{i=1}^{n}\tilde{w}(X_i)}$$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n}\tilde{h}(X_i)\tilde{w}(X_i) = \frac{1}{n}\sum_{i=1}^{n}h(x)\rho\tilde{w}(X_i) \approx \frac{1}{n}\sum_{i=1}^{n}h(x)\frac{1}{\frac{1}{n}\sum_{i=1}^{n}\tilde{w}(X_i)}\tilde{w}(X_i)$$

More general, $\hat{I} = \sum_{i=1}^{n}h(x)\frac{\tilde{w}(X_i)}{\sum_{i=1}^{n}\tilde{w}(X_i)}$

Example

$$\tilde{w}(x) = \frac{e^{-\frac{x^2}{2}}}{e^{-2x}}$$

$$\hat{I} = \sum_{i=1}^{n}h(x)\frac{\tilde{w}(X_i)}{\sum_{i=1}^{n}\tilde{w}(X_i)}$$

4.4 Specialization to uniform

$$f(x) = \begin{cases} \frac{1}{|A|} & \text{if } x \in A; \\ 0 & \text{otherwise.} \end{cases}$$

Suppose we have a uniform distribution but don't know how to calculate the volumn of A.

$$\tilde{f}(x) = 1_A(x), z_f = |A|$$

Assume $g(x) > 0, x \in A$
Generate $X_1, X_2, ..., X_n \sim g(x)$

$$|A| = z_g E_g(\tilde{w}(x))$$

Because $z_g = 1$.

$$|A| = E_g(\tilde{w}(x)) = E_g(\frac{1_A(x)}{g(x)}) = \frac{1}{n}\sum_{i=1}^{n}\frac{1_A(x_i)}{g(x_i)}$$

$$\hat{I} = \sum_{i=1}^{n}h(x_i)\frac{\frac{1_A(x_i)}{g(x_i)}}{\sum_{i=1}^{n}\frac{1_A(x_i)}{g(x_i)}}$$

If $g(x) \sim \text{Unif}[0,1]$
$\tilde{f}(x) = 1$ so that $z_f = |A|$

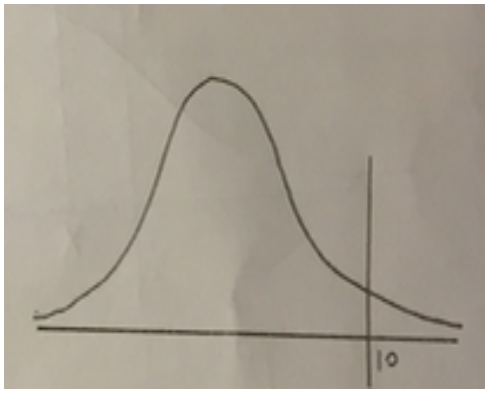
$$\rho = \frac{z_g}{z_f} = \frac{z_g}{|A|}$$

$$|A| = \frac{z_g}{\rho} = z_g \frac{1}{n} \sum_{i=1}^{n} \tilde{w}(X_i) = z_g \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\tilde{g}(x)}$$

4.5 Application

4.5.1 Tail Area

$$X_i \sim \mathrm{N}(0,1)$$
$$P(X > 10) = ?$$



Regular Monte Carlo

Generate $X_1, X_2, ..., X_n \overset{i.i.d}{\sim} \mathrm{N}(0,1)$

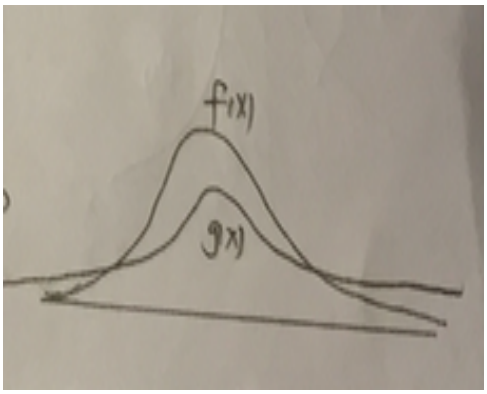
$$\hat{I} = \frac{1}{n} \sum_{i=1}^{n} 1_{X_i > 10} \xrightarrow[n]{\infty} P(X > 10) = E(1_{X>10})$$

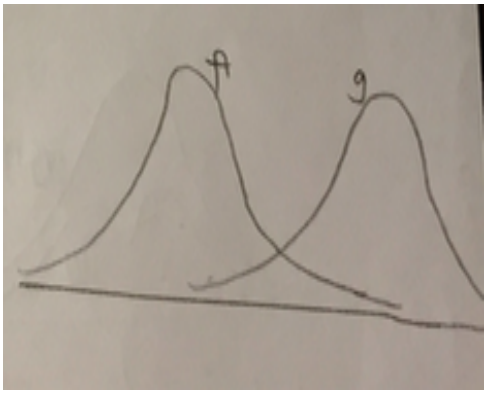
Most times, $\hat{I}$ is zero. So we need to have large number of samples to get the result which is very inefficient.

Importance Sampling

$$f(x) \sim \mathrm{N}(0,1)$$

$g(x) \sim \mathrm{N}(0, 10^2)$ or $\mathrm{N}(10, 1)$





Whichever has the samller variance is the better simulation to get the probability to go beyond 10.

$$X_1, X_2, ..., X_n \overset{i.i.d}{\sim} g(x)$$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} 1_{X_i>10} \cdot \frac{f(X_i)}{g(X_i)}$$

where $1_{X_i>10} = h(x)$ and $\dfrac{f(X_i)}{g(X_i)} = w(x)$.

4.5.2 Insurance

Calculating Small probability is useful in insurance. e.g. If we run the company for 3 days, what is the probability of going bankrupt? Loss: $X_1, X_2, ..., X_n \overset{i.i.d.}{\sim} \mathrm{N}(-10, 1)$ where $f(x) \sim \mathrm{N}(-10, 1)$ and $\mu < 0$ because in general, companies make money every day. Suppose the probability of declaring bankrupcy is $P(\max(x_1, x_1 + x_2, x_1 + x_2 + 3) > 50)$ where 50 accounts for the deposit in bank. $g(x) : X_1, X_2, ..., X_n \overset{i.i.d.}{\sim}$ $\mathrm{N}(20, 1)$ in doing so, we shift the distribution

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} 1_{\max_i>c} \cdot \frac{f(x_1^i, x_2^i, x_3^i)}{g(x_1^i, x_2^i, x_3^i)}$$

$$f(x_1, x_2, x_3) = f(x_1)f(x_2)f(x_3)$$
$$g(x_1, x_2, x_3) = g(x_1)g(x_2)g(x_3)$$

4.5.3 Self-avoiding Walk

Length $= N$
Each point is presented by a set of coordinates.

$$x_1 = (0, 0)$$

$(X_1, X_2, ..., X_n) = \vec{x}$ can either be a configuration of a molecule or a path/trajectory of a self-avoiding walk.
Constraints
X_{t+1} and X_t are neighbors.
$X_{t+1} \neq X_t$ or X_{t-1} or ... X_1.
Idea
Think of $\vec{x}$ as a draw from a uniform distribution.
Define $A =$ all possible configurations/paths

$\vec{x} \sim \text{unif}(A)$ this is $f(\vec{x})$.

$$I = E_f(|x_N - x_1|^2)$$

where $h(x) = |x_N - x_1|^2$ is the squared distance between the starting and ending points among all possible paths.
find $g(x)$ / random walk $\rightarrow$ simulate walks $\rightarrow$ calculate $\tilde{w}(x)$ or $w(x) \rightarrow \hat{I}$
$g(x)$ is the random self-avoiding walk.

Algorithm

$x_1 = (0, 0)$ for $t = 1, ...N - 1$
Given $x_1, x_2, ..., x_t$
Choose x_{t+1} randomly from untraveled neighbors of x_t (locally uniform)

$$m_t \in 4, 3, 2, 1, 0$$

If $m_t > 0$, choose one neighbor with probability $= \dfrac{1}{m_t}$ regardless history;
If $m_t = 0$, stop and return$x_1, x_2, ..., x_t$.
It is possible to stop before N steps.
Repeat $i = 1, ..., n$ times.

$$g(\vec{x}) = P(x_1)P(x_2|x_1)...P(x_{t+1}|x_1...x_t)...P(x_T|x_1...x_{T-1}), T \leqslant N$$

$$g(\vec{x}) = \frac{1}{m_1}\frac{1}{m_2}...\frac{1}{m_T} = \frac{1}{\prod_{t=1}^{T} m_t}$$

$\vec{x}$ is not uniform because there exist different m_t for different paths.

$$|A| = E_g(\frac{1_A(x)}{g(x)}) = E_g(\frac{1_A(x)}{1/\prod_{t=1}^{T} m_t}) = E_g(1_A(\vec{x})\prod_{t=1}^{T} m_t)$$

Repeat n times

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n}\prod_{t=1}^{T^{(i)}} m_t^{(i)} 1_A(\vec{x}^{(i)})$$

$1_A = 1$ if it reaches N steps; $1_A = 0$ if it stops early or $T < N$.

$$\hat{I} = \sum_{i=1}^{n}\left|x_N^{(i)} - x_1^{(i)}\right|^2 \frac{1_A(\vec{x}^{(i)})\prod_{t=1}^{T^{(i)}} m_t^{(i)}}{\sum_{t=1}^{T^{(i)}} 1_A(\vec{x}^{(i)})\prod_{t=1}^{T^{(i)}} m_t^{(i)}}$$

where m_t^i is the number of choices at step i.

5 Markov Chain

5.1 Introduction

Markov Chain is a preparation for Markov Chain Monte Carlo(MCMC). The goal for Markov Chain is try to sample $\vec{x} \sim \pi(\vec{x})$ and also minimize $\varepsilon(\vec{x})$. $\vec{x}$ can be high dimensional or with difficult configuration.

Example:

$$\pi(\vec{x}) = \frac{1}{z} e^{\frac{-\varepsilon(\vec{x})}{T}}$$

T is the temperature, and z is normalizing constant or partition function.
Then,

$$z = \sum_{x} e^{\frac{-\varepsilon(\vec{(x)})}{T}}$$

$\pi(\vec{(x)})$ is called the Gibbs or Boltzmann distribution, and it is difficult to calculate the z.

Iterative algorithm

$$x_0 \to x_1 \to x_2 \to ...x_t \to ...$$

$$\text{for large t, } x_t \sim \pi(x)$$

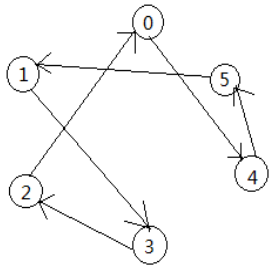
Real life example: card shuffling

$$\vec{x} = \text{permutation of (1,2,...,52)}$$
$$\vec{x} \sim \pi(x) = \text{uniform over all 52 permutations.}$$

Markov Chain: Start from (1,2,...,52)
Iterate: at iterate t, randomly pick 2 card in x_t, then we will exchange their positions and get x_{t+1}. This process is called random transposition. We modify the current configuration, regardless of previous configuration.
For big t, $x_t \sim$ uniform.
For instance, n=5. $x_0 = (1, 2, 3, 4, 5)$
Randomly pick 2,4.$(1, 2^{\downarrow}, 3, 4^{\downarrow}, 5)$ and exchange their positions, then $x_1 = (1, 4, 3, 2, 5)$
Randomly pick 1,4.$(1^{\downarrow}, 4^{\downarrow}, 3, 2, 5)$ and exchange their positions, then $x_2 = (4, 1, 3, 2, 5)$
Randomly pick 4,5.$(4^{\downarrow}, 1, 3, 2, 5^{\downarrow})$ and exchange their positions, then $x_3 = (5, 1, 3, 2, 4)$
$x_t \xrightarrow{\text{random mapping}} x_{t+1}$, and we do not care about $x_0, x_1, ...x_{t-1}$

Riffle shuffle
Riffle shuffle is we randomly split card into 2 groups, and then we randomly drop the cards. so we get x_t(currenct iterator) $\to x_{_{t+1}}$(next iterator). Riffle Shuffle shows that 7 shuffles are enough.

Travelling salesman problem
There are 6 cities (in general, n cities), and we always leave from city 0 (hometown) and each city can travel once.



Look at the graph, we can see that $\vec{x} = (4, 5, 1, 3, 2)$, $\varepsilon(\vec{x})$ = whole distance of the path. If we want to find shortest path, which is min$\varepsilon(\vec{x})$

If we sample $\vec{x} \sim \pi(\vec{x}) = \frac{1}{2} e^{\frac{-\varepsilon(\vec{x})}{T}}$

If $\varepsilon(\vec{x})$ is small, $\pi(\vec{x})$ is large.

If $\varepsilon(\vec{x})$ is large, $\pi(\vec{x})$ is small.

More likely, we can get $\vec{x}$ with small $\varepsilon(\vec{x})$ using random transposition.

5.2 Markov Chain

We have random number variables $x_0, x_1, ...x_t....$

Markov have following property: future $\perp$past | present

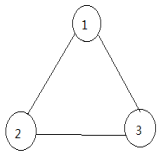
It means that if we know the present, we can know the future, and the future does not related to the past. Future and past are independent.

$P(x_{t+1} = y \mid x_t = x, x_{t-a}, ...x_0) = k(x, y)$.← transition probability.

The transition probability means if $x_t = x$(current), what is the probability of $x_{t+1} = y$

Note:$x_{t-1}, ...x_0$ is the past history.

Then, let us consider an **example** to show how Markov Chain works.



Random walk on state space ($\bigstar$).
$\bigstar = \{1, 2, 3\}$. We have three states, and all of them are connected. We can see the following graph.
First,$x_0 = 1$. At each iterator, with probability $\frac{1}{2}$ to stay. If we do not stay, with probability $\frac{1}{4}$ respectively to go to one of the neighbours.
We can flip a fair coin to generate this probability. If head, stay; if tail, flip the coin again: if head, move clock-wise; if tail, move counter clock-wise.
The following is a transition matrix. It shows the probability for $x_{t+1} = y \mid x_t = x$. Sometimes, people use k_{ij}, a 3×3 matrix. Using i for x, and j for y.

x/y	1	2	3
1	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$
2	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$
3	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$

Assume $x_0 = 1$,then we have the following table to show the probability for each state at $t = 1$.

x_1	1	2	3
prob	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$

Then we need to calculate the probability for each state at $t = 2$.
$P(x_2 = 1) = P(x_2 = 1, x_1 = 1) + P(x_2 = 1, x_1 = 2) + P(x_2 = 1, x_1 = 3)$
$= P(x_1 = 1)P(x_2 = 1 \mid x_1 = 1) + P(x_1 = 2)P(x_2 = 1 \mid x_1 = 2) + P(x_1 = 3)P(x_2 = 1 \mid x_1 = 3)$
$= \frac{1}{2} \times \frac{1}{2} + \frac{1}{4} \times \frac{1}{4} + \frac{1}{4} \times \frac{1}{4} = \frac{3}{8}$
Following the same method, we can calculate that $P(x_2 = 2) = \frac{5}{16}$ and $P(x_2 = 3) = \frac{5}{16}$. The following table is about the probability for each state at $t = 2$.

x_1	1	2	3
prob	$\frac{3}{8}$	$\frac{5}{16}$	$\frac{5}{16}$

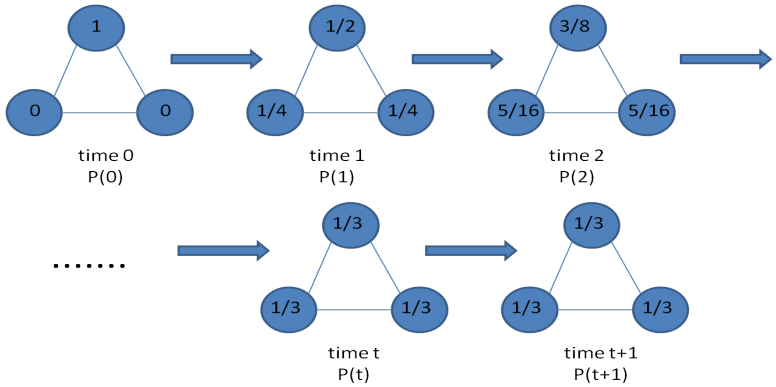
We see a pattern that

$$P^{(t)}(x) = Pr(x_t = x)$$

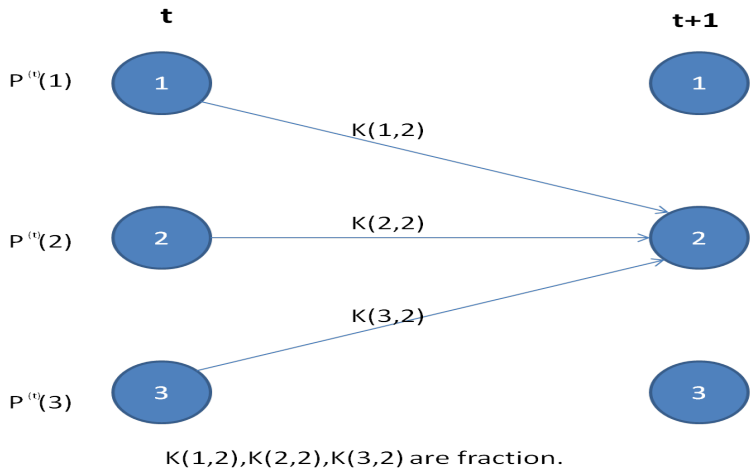
$$P^{(t+1)}(y) = pr(x_{t+1} = y) = \sum_x Pr(x_{t+1} = y \mid x_t = x)Pr(x_t = x) = \sum_x k(x, y)P^t(x)$$

Then we can go back to the population immigration problem. 1 million people start from state 1. At each step, for each state, half people choose to stay, and one quarter of the people choose immigrate clockwise, and one quarter of the people choose immigrate counter clockwise. Then we want to know the distribution of population at time t.$P^t(x) = P(x_i = x) =$ number of people in state x at time t.
The following is a table for the population in each state for time $t = 0, 1, 2$, and a graph for how the population changes in each state as time changing.

time	1	2	3
0	1 million	0 million	0 million
1	$\frac{1}{2}$ million	$\frac{1}{4}$ million	$\frac{1}{4}$ million
2	$\frac{3}{8}$ million	$\frac{5}{16}$ million	$\frac{5}{16}$ million

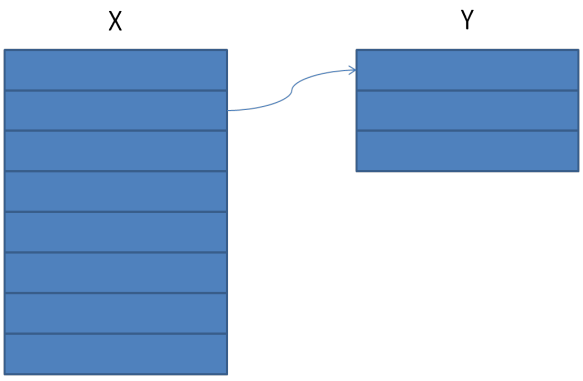


From the graph, we can see that after reach the stationary distribution $\pi(x)$, the distribution does not change with time t change. Therefore, $\pi(y) = \sum_x \pi(x)k(x,y)$, $\pi(x)$ represents the number of people in x. $k(x,y)$ represents the fraction of these in x who will move to y. Moreover, $P_{(t+1)}(y) = \sum_x P^t(x)k(x,y)$. The following graph is to explain how this formula works.



Google page rank

Suppose we have already know k according to the model. We want to know $\pi(x)$, the number of people end up in page x. The rank x is based on $\pi(x)$.



Assume we have the probability of 85% to click a link related to the current page randomly, and we have the probability of 15% to jump to a page not linked.
We start from $P^0 \to P^1 \to P^2$ using $P^{t+1}(y) = \sum_x P^t(x)k(x,y)$, then it achieve a uniform distribution π.

When people search a certain keyword, we can output to the page by order and the most relevant pages on rank.
For MCMC, we know π, we want to design the Markov Chain.

5.3 Markov Chain and sampling

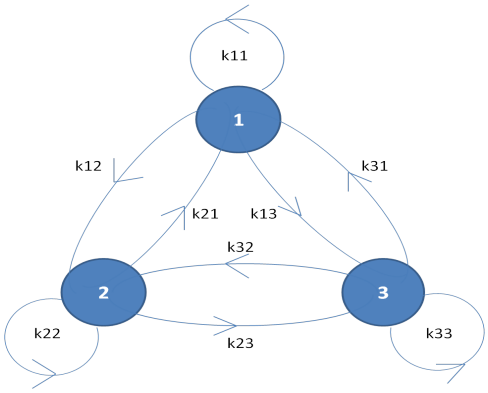
We have ever learnt about some sampling method, like inversion method, acceptance-rejection method, importance sampling. Now we want to learn how the Markov Chain related to the sampling process.
A Markov Chain is defined by (X, k, π), X is state space. $X = \{1, 2, 3\}$, k is the transition matrix, and π is the stationary distribution.

5.3.1 Transition matrix

Look at the three states, $p^{(t)}(i)$ is the number of people in state i at time t. $i = \{1, 2, 3\}$; and $k(i, j)$ or k_{ij} is the fraction of those in state who will go to state j.

$$k = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}_{3\times 3}$$

The following is a directed graph about the transition matrix.



Let $x_1, x_2, ...x_t, ...$ be a sequence of states visited by the Markov Chain. x_t is the state at time t.
$P(x_{t+1} = y \mid x_t = x, x_{t-1}, ...x_0)$
$x_{t+1} = y$ is the future state, $x_t = x$ is the current state, and $x_{t-1}, ...x_0$ is the past state.
Finite state: $k_{ij} = k(i,j) = P(x_{t+1} = j \mid x_t = i, x_{t-1}, ...x_0)$
By the Markov Chain property: future $\perp$ past $\mid$ present ("Memoryless"), so $k(x,y) = P(x_{t+1} = y \mid x_t = x)$.
Let $P^{(t)}(x) = P(x_t = x)$
$P^{(t+1)}(y) = P(x_{t+1} = y) = \sum_x P(x_{t+1} = y \mid x_t = x) = \sum_x P(x_t = x)P(x_{t+1} = y \mid x_t = x) = \sum_x P^{(t)}(x)k(x,y)$
In vector/matrix form, the finite state:

$$P_j^{t+1} = P_j^{t+1} = P(x_{t+1} = j) = \sum_i P_i^{(t)} k_{ij}$$

Denote $P^t = (P_0^t, P_1^t, P_2^t, ...P_t^t, ...P_n^t)$,

$$P_j^{(t+1)} = P^{(t)} \begin{pmatrix} k_{1j} \\ k_{2j} \\ ... \\ k_{nj} \end{pmatrix}$$

By induction, $P^t = P^{t-1}k = P^{t-2}kk = P^0k^t$.

$$P_{1\times n}^{t+1} = P_{1\times n}^t k_{n\times n}$$

$$P^t = P^0 k^t$$

5.3.2 t-step transition

$$k^t(x,y) = P(x_{s+t} = y \mid x_s = x)$$

$$P^t(y) = P(x_t = y) = \sum_x P(x_t = y, x_0 = x)$$

$$= \sum_x P(x_0 = x)P(x_t = y \mid x_0 = x)$$

$$= \sum_x P^{(0)}(x)k^{(t)}(x,y)$$

$$P^{(t)} = P^{(0)}(x)k^{(t)}$$

Using $P^{(t)} = P^{(0)}k^t$, so that $k^{(t)} = k^t$.
For example,

$$k^{(2)}(x, y) = P(x_{t+2} = y \mid x_t = x)$$
$$= \sum_z P(x_{t+2} = y, x_{t+1} - 2 \mid x_t = x)$$

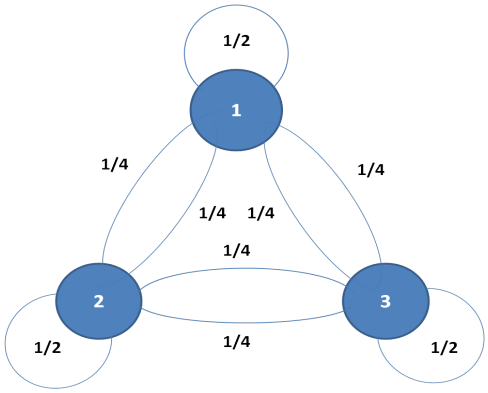
Since $P(A, B) = P(A)P(B \mid A) = P(B)P(B \mid A)$, $P(A, B \mid C) = P(A \mid C)P(B \mid A, C)$
$= \sum_z P(x_{t+1} = z \mid x_t = x)P(x_{t+2} = y \mid x_{t+1} = z, x_t = x) = \sum_z k(x, z)k(z, y)$.
Therefore, $k^{(2)} = kk = k^2$. By induction, we can get $k^{(t)} = k^t$.

5.3.3 Stationary distribution

Definition for stationary distribution π: $\pi k = \pi$. $\pi(y) = \sum_x \pi(x)k(x, y)$

Detailed balance condition $\pi(x)k(x, y) = \pi(y)k(y, x)$. If it satisfies detailed balance, it can satisfy the overall balance $\pi = \pi k$. $\pi(y) = \sum_x \pi(x)k(x, y) = \sum_x \pi(y)k(y, x) = \pi(y) \sum_x k(y, x) = \pi(y)$

5.3.4 Simulation



$P^{(0)} = (1, 0, 0)$ or $(0, 1, 0)$ or $(0, 0, 1) \rightarrow P^{(t)} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, which is π.

5.3.5 Conclusion

$$\pi = \pi k$$
$$\pi(x)k(x, y) = \pi(y)k(y, x)$$
$$k^{(t)} = k^t$$
$$P^{(t)} = p^{(0)}k^t = p^{(0)}k^{(t)}$$

MCMC can guratee have iid sample.

5.4 Metropolis-Hastings Algorithm

5.4.1 introduction

Considering the **example** we have discussed in the Markov Chain before, the immigration among three states, we let $\pi(x)B(x,y)$ represents the number of people who will move to y and $B(x,y) = B(y,x)$, then we will get that $\pi(x)B(x,y) = \pi(y)B(y,x)$ for all x and y, and π is a uniform distribution.

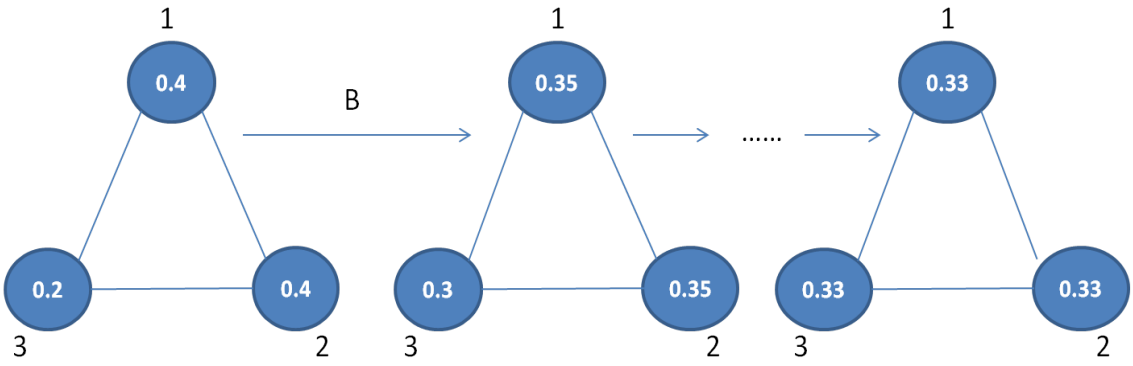
If we want sample from uniform π, we should design a symmetric chain: $B(x,y) = B(y,x)$.

Another **example** is Card shuffling by random transposition. $p(\vec{x})$ = uniform over 52! permutations.

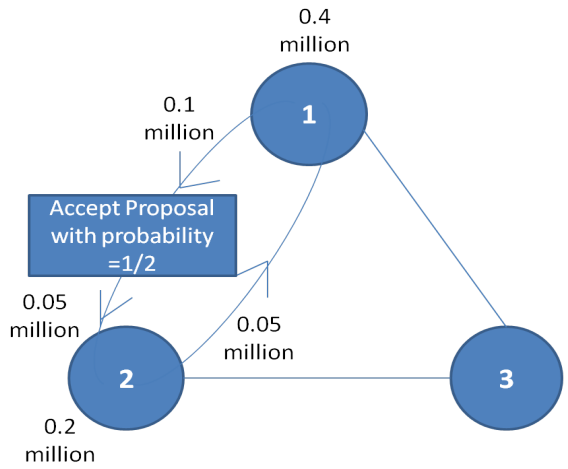
5.4.2 Metropolis Algorithm

Question:What if π is not uniform?

In travelling salesman problem, $\pi(\vec{x}) = \frac{1}{z}e^{-\frac{\varepsilon(\vec{x})}{T}}$ is not uniform.($\varepsilon(\vec{x})$ is the total length of $\vec{x}$ (Path permutation)).
Seeing the following graph, assume there are 0.4 million, 0.2 million, 0.4 million people in state 1,2,3, respectively.



If we use the transition matrix of base chain, the population in these three states will become $\frac{1}{3}$ million.
Now consider **Metropolis-Hastings algorithm**. See the following graph, there are 0.1 million people will immigrate from state 1 to state 2, while there are 0.05 million people will immigrate from state 2 to state 1. If we want to keep the population in state 2 unchanged, we cold randomly choose 0.05 million people out of 0.1 million to get permissions to get into state 2. The office for the immigration from state 1 to state 2 accept the proposal with probability $\frac{1}{2}$, while the office for the immigration from state 2 to state 1 accept the proposal with probability 1 . This is the basic idea behind the Metropolis algorithm.



Suppose state space is ★, the base chain $B(x, y)$ make proposal.

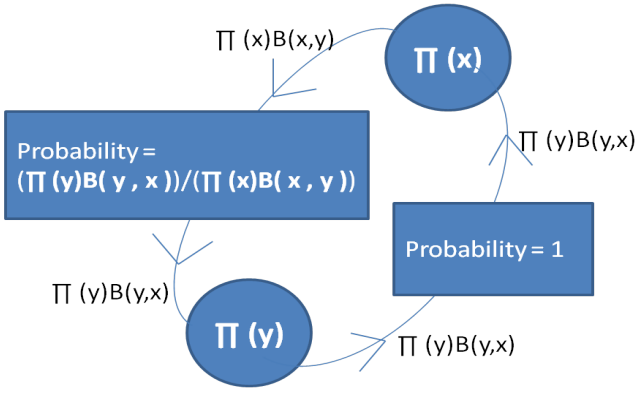
$$P(X_{\text{proposal}} = y | X_t = x) = B(x, y)$$

What is the probability to accept X_{proposal}?

$$\text{If we accept, } X_{t+1} = X_{\text{proposal}}$$

$$\text{it we reject, } X_{t+1} = X_t$$

More generally, see the following graphs for the immigration between state x and state y. Totally there are $\pi(x)B(x, y)$ make proposal to immigrate from state x to state y, and there are $\pi(y)B(y, x)$ make proposal to immigrate from state y to state x.($\pi(x)$ is the number of people in state x, and $B(x, y)$ is the fraction of people in state x will move to state y)



Case 1: if $\pi(x)B(x, y) \geq \pi(y)B(y, x)$, the office for the immigration from state x to state y will accept x proposal with probability $= \dfrac{\pi(y)B(y, x)}{\pi(x)B(x, y)}$

Case 2: if $\pi(x)B(x, y) \leq \pi(y)B(y, x)$, the office for the immigration from state x to state y will accept x proposal with probability =1.

Therefore, the probability to accept x proposal $= \min(1, \dfrac{\pi(y)B(y, x)}{\pi(x)B(x, y)})$.

5.4.3 Implementation

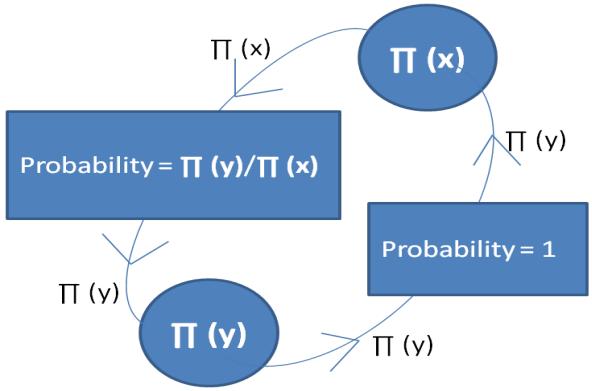
How to write R code to implement it?

We generate $U \sim \text{unif}(0,1)$, if $U \leq \dfrac{\pi(y)B(y,x)}{\pi(x)B(x,y)}$, accept the proposal and then $X_{t+1} = X_{\text{proposal}}$, otherwise, reject the proposal and $X_{t+1} = X_t$

This algorithm always called the Metropolis Hastings algorithm.

If we assume $B(x,y) = B(y,x)$, then prob $= \min(1, \dfrac{\pi(y)}{\pi(x)})$.

Looking at the following graph, $\pi(x)$ is the number of people in a big country, $\pi(y)$ is the number of people in small country and the base chain is more symmetric. The proposals by people from big country to small country will not always be accepted, while the proposals by people from small country to big country will always be accepted.

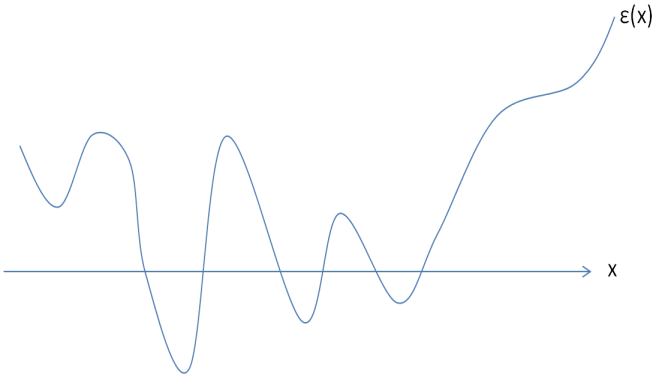


If we want to maintain the overall population in different countries, we need the Metropolis-Hastings algorithm. If we do not have this algorithm, then the population in different country will be a uniform distribution.

5.4.4 More Examples

Energy state example

Consider the following graphs. $\varepsilon(x)$ is the energy.



$\pi(x) = \dfrac{1}{z}e^{-\varepsilon(x)}$ and $\pi(y) = \dfrac{1}{z}e^{-\varepsilon(y)}$.

Prob $= \min(1, \dfrac{\pi(y)}{\pi(x)}) = \min(1, e^{\varepsilon(x)-\varepsilon(y)})$.

If $\varepsilon(x) \geq \varepsilon(y)$, the prob = 1 to move to low energy state; if $\varepsilon(x) \leq \varepsilon(y)$, the prob = $e^{\varepsilon(x)-\varepsilon(y)}$ to move to high energy state.
High energy state will always go to low energy state. This algorithm allows the small ball jump from a local minimum to a better local minimum. If we do not allow the small ball from low energy state to high energy state, the small ball will get stuck in the local minimum.
Then adding a temperature T, $\pi(x) = \frac{1}{z}e^{-\frac{\varepsilon(x)}{T}}$ and $\pi(y) = \frac{1}{z}e^{-\frac{\varepsilon(y)}{T}}$.
prob = $\min(1, e^{\frac{\varepsilon(x)-\varepsilon(y)}{T}})$.
If $\varepsilon(x) \geq \varepsilon(y)$, prob = 1; if $\varepsilon(x) \leq \varepsilon(y)$, prob = $e^{\frac{\varepsilon(x)-\varepsilon(y)}{T}}$.
If T is large, the prob close to 1; if T is small, the prob close to 0.
Simulated annually, the process is starting from very high temperature T, then gradually lower the temperature T, which is as the same as putting a hot metal in water in our real life.
Travelling salesman
For the travelling salesman problem. each state $\vec{x}$ is a path. We will randomly pick two cities and switch their positions. The accept prob is $\min(1, \frac{\varepsilon(x)-\varepsilon(y)}{T})$. $\varepsilon(x) - \varepsilon(y)$ is reduction in total distance.
If we want to reach our target distribution π and we have a base chain $B(x, y)$. The algorithm for this problem is:
At time t, $X_t = x$

$$P(X_{\text{proposal}} = y \mid X_t = x) = B(x, y)$$

We accept X_{proposal} with prob $\min(1, \frac{\pi(y)B(y,x)}{\pi(x)B(x,y)})$

$$M(x, y) = P(X_{t+1} = y \mid X_t = x) \text{ if}(x \neq y)$$

$$= P(X_{\text{proposal}} = y, \text{ accept} \mid X_t = x)$$

Using Chain Rule,

$$= P(\text{accept} \mid X_t = x, X_{\text{proposal}} = y)P(X_{\text{proposal}} = y \mid X_t = x)$$

$$= \min(1, \frac{\pi(y)B(y,x)}{\pi(x)B(x,y)})B(x, y)$$

$$\pi(x)M(x, y) = \pi(x)\min(1, \frac{\pi(y)B(y,x)}{\pi(x)B(x,y)})B(x, y)$$

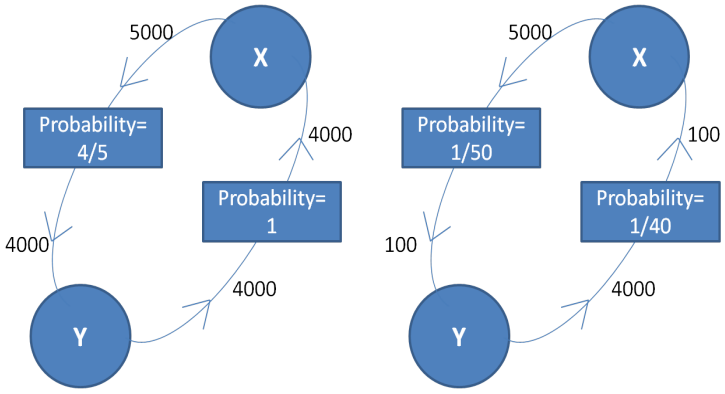
$$= \min(\pi(x)B(x, y), \pi(y)B(y, x))$$

$$= \pi(y)M(y, x), \text{ which is symmetrix in x and y}$$

if $x = y$, $M(x, y) = 1 - \sum_{y \neq x} M(x, y)$.

Through this algorithm, we can maintain a detailed balance, $\pi(x)M(x, y) = \pi(y)M(y, x)$, which means that once in π distribution, it will stay in π distribution all the time.

There always have alternative way to maintain the balance.(seeing the following graph)



We choose the first one because maximum flow of the population converges to target distribution faster.

Immigration
Going back to the immigration problem we have discussed in the Markov Chain, $\pi(1) = 0.4$, $\pi(2) = 0.2$ and $\pi(3) = 0.4$

x/y	1	2	3
1	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$
2	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$
3	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$

$$M(1,2) = B(1,2)\min(1, \frac{\pi(2)}{\pi(1)}) = \frac{1}{4} \times \frac{1}{2} = \frac{1}{8}$$

$$M(1,3) = B(1,3)\min(1, \frac{\pi(3)}{\pi(1)}) = \frac{1}{4} \times 1 = \frac{1}{4}$$

$$M(1,1) = 1 - M(1,2) - M(1,3) = \frac{5}{8}$$

Then, we can get the transition matrix under Metropolis Algorithm.

x/y	1	2	3
1	$\frac{5}{8}$	$\frac{1}{8}$	$\frac{1}{4}$
2	$\frac{1}{4}$	$\frac{5}{8}$	$\frac{1}{8}$
3	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$

Another example
$\pi(x) = \frac{1}{z} e^{-\frac{x^2}{T}}$, x is an integer. $z = \sum_{x=-\infty}^{\infty} e^{-\frac{x^2}{2}}$.
Using for loop iterator algorithm,
(1) design base chain

Base chain $B(x,y)$, $P(X_{\text{proposal}} = y \mid X_t = x) = \begin{cases} \frac{1}{2} \text{ when } y = x-1 \\ \frac{1}{2} \text{ when } y = x+1 \end{cases}$

So, if $X_t = x$, we sample $U_1 \sim \text{unif}(0,1)$, and $X_{t+1} = \begin{cases} x+1 \text{ if } U_1 \geq 0.5 \\ x-1 \text{ if } U_1 \leq 0.5 \end{cases}$

(2)accept y with prob $= \min(1, \frac{\pi(y)}{\pi(x)}) = \min(1, \frac{e^{-\frac{y^2}{2}}}{e^{-\frac{x^2}{2}}})$

We generate $U_2 \sim$ unif, if U_2
$leqp, X_{t+1} = y$; otherwise, $X_{t+1} = x$.

(3)$\pi(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2}}$, $x \in R$. Base Chain: $X_t = x, y = x + \triangle$ and $\triangle \sim \text{unif}[-a, a]$

More about Travelling salesman problem

Each state is a solution, a solution is a path (a permutation of n cities).(1,2,...n) and the hometown = 0.A state is not a city, is a path.

We let $X_t = x$, which is a permutation. $X_{\text{proposal}} = y$.

we sample i from unif$\{1, 2, ...n\}$ and sample j from unif$\{1, 2, ...n\}$.

If $i \neq j$, $y = x$ ($y_j = x_i$, $y_i = x_j$), the acceptance probability $= \min(1, e^{\frac{\varepsilon(x) - \varepsilon(y)}{T}})$

x: 2 $1^{\downarrow i}$ 3 4 $5^{\downarrow j}$. y: 2 5 3 4 1.

Then $x = 2 \to 1+1 \to 3+3 \to 4+4 \to 5+5 \to 2$,

and $y = 2 \to 5+5 \to 3+3 \to 4+4 \to 1+1 \to 2$.