

MATH 556: MATHEMATICAL STATISTICS I

RANDOM NUMBER GENERATION AND MONTE CARLO

Many functions exist in R to produce random samples from probability distributions.

```
n<-10
args(rnorm)

+ function (n, mean = 0, sd = 1)
+ NULL

rnorm(n,5,sqrt(4))      #sample from Normal(5,4)

+ [1] 6.502293 1.473405 2.702801 5.487267 9.760693 6.604750 3.156596 4.308689
+ [9] 8.060313 3.283245

args(rgamma)

+ function (n, shape, rate = 1, scale = 1/rate)
+ NULL

rgamma(n,2,0.5)        #sample from Gamma(2,0.5)

+ [1] 3.1156507 8.3305039 3.9334308 4.9887674 0.6771500 3.0315139 3.5519852
+ [8] 6.7265197 0.5367732 2.1656573

args(rpois)

+ function (n, lambda)
+ NULL

rpois(n,3.5)           #sample from Poisson(3.5)

+ [1] 5 4 4 4 4 2 4 0 8 1

args(sample)

+ function (x, size, replace = FALSE, prob = NULL)
+ NULL

sample(c(1:100),n,replace=T)  #sample discrete uniform on {1,2,...,100} with replacement

+ [1] 20 49 78 48 13 90 59 86 63 29

sample(c(1:100),n,replace=F) #sample discrete uniform on {1,2,...,100} without replacement

+ [1] 22 1 52 73 40 6 72 48 36 24

sample(c(1:5),n,replace=T,prob=c(0.2,0.1,0.4,0.1,0.2)) #sample discrete distn on {1,2,...,5}

+ [1] 5 3 4 3 5 2 5 1 3 2
```

The set `.seed()` function can be used to make the random draws reproducible

```
set.seed(8910)
rnorm(5);rnorm(5)

+ [1] 0.8214213 0.6982410 -0.3845740 1.3006786 1.4397298
+ [1] 0.02488985 0.69435700 0.32589669 -1.19616616 0.61756331

set.seed(8910)
rnorm(5)

+ [1] 0.8214213 0.6982410 -0.3845740 1.3006786 1.4397298
```

Random samples can be used for several purposes in basic distribution theory.

- **Visualization:** $X \sim Weibull(\alpha, \beta)$

$$f_X(x) = \alpha \beta x^{\alpha-1} \exp\{-\beta x^\alpha\} \quad x > 0$$

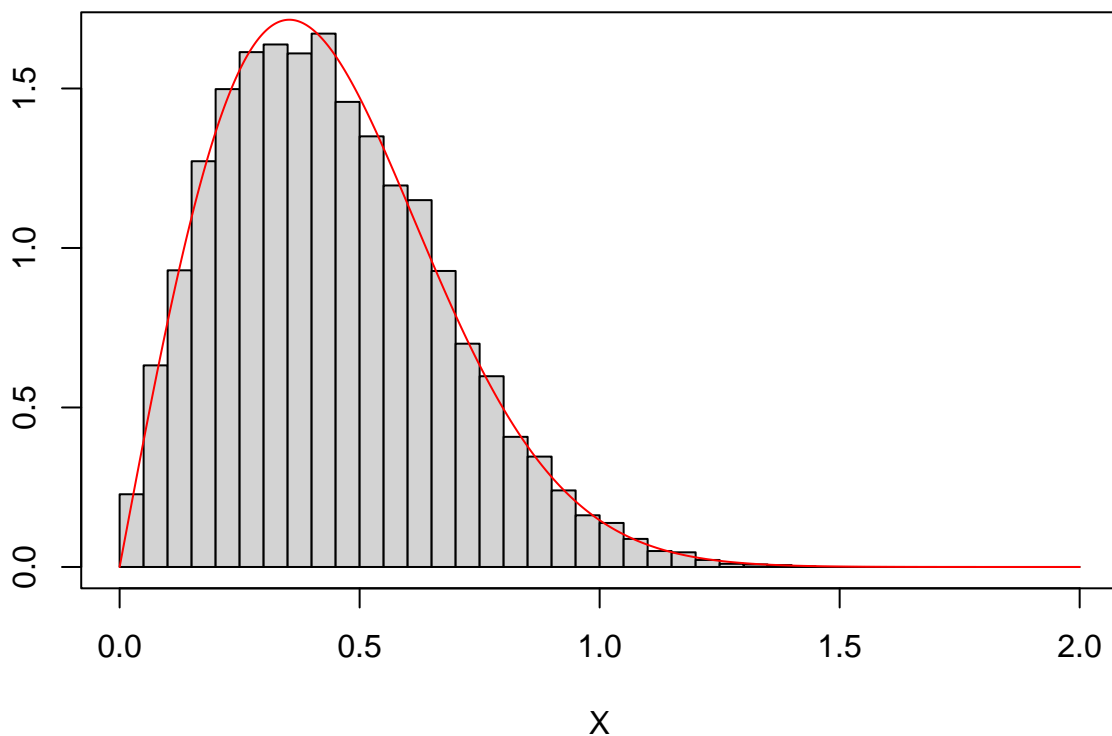
and zero otherwise, for $\alpha > 0$ and $\beta > 0$. Parameter α is the *shape* parameter, and β controls the dispersion of the distribution. An alternative parameterization utilizes the *scale* parameter $\sigma = 1/\beta^{1/\alpha}$ yielding pdf

$$f_X(x) = \frac{\alpha}{\sigma} \left(\frac{x}{\sigma}\right)^{\alpha-1} \exp\left\{-\left(\frac{x}{\sigma}\right)^\alpha\right\} \quad x > 0$$

This is the parameterization that R uses. The expectation of this distribution is

$$\mathbb{E}_X[X] = \frac{\Gamma\left(1 + \frac{1}{\alpha}\right)}{\beta^{1/\alpha}} = \sigma \Gamma\left(1 + \frac{1}{\alpha}\right).$$

```
n<-10000
al<-2;be<-4; sig<-1/be^{1/al}
X<-rweibull(n,al,sig)
par(mar=c(4,3,1,0))
hist(X,breaks=seq(0,2,by=0.05),main='',freq=FALSE);box()
x<-seq(0,2,by=0.001)
fx<-dweibull(x,al,sig)
lines(x,fx,col='red')
```



```
sig*gamma(1+1/al) #Expectation
+ [1] 0.4431135

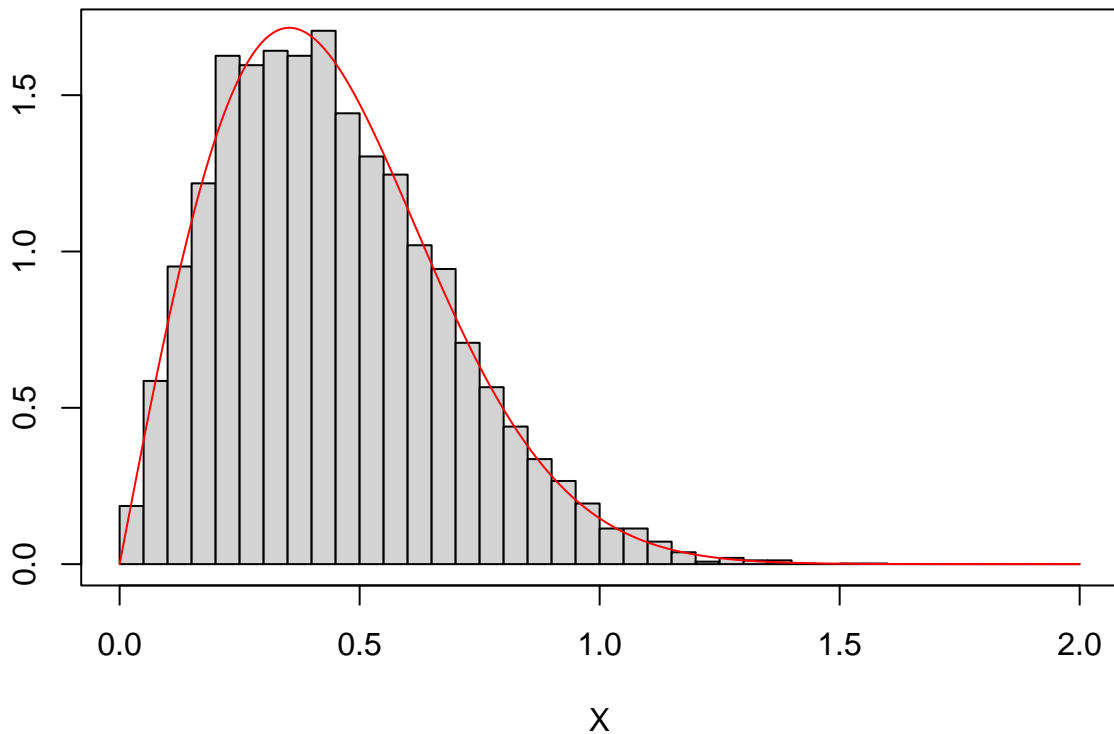
mean(X) #Sample mean
+ [1] 0.4426044
```

- **Transformations:** If $U \sim Uniform(0, 1)$, then

$$X = \left(-\frac{1}{\beta} \log(1 - U) \right)^{1/\alpha}$$

ensures that $X \sim Weibull(\alpha, \beta)$

```
set.seed(8910)
n<-10000
U<-runif(n)
X<-(-log(1-U)/be)^(1/al)
par(mar=c(4,3,1,0))
hist(X,breaks=seq(0,2,by=0.05),main='',freq=FALSE);box()
lines(x,fx,col='red')
```



```
sig*gamma(1+1/al) #Expectation
+ [1] 0.4431135
mean(X)           #Sample mean
+ [1] 0.4448699
```

Multivariate transformations can also be studied. Suppose that U_1 and U_2 are independent $Uniform(0, 1)$ variables, and consider

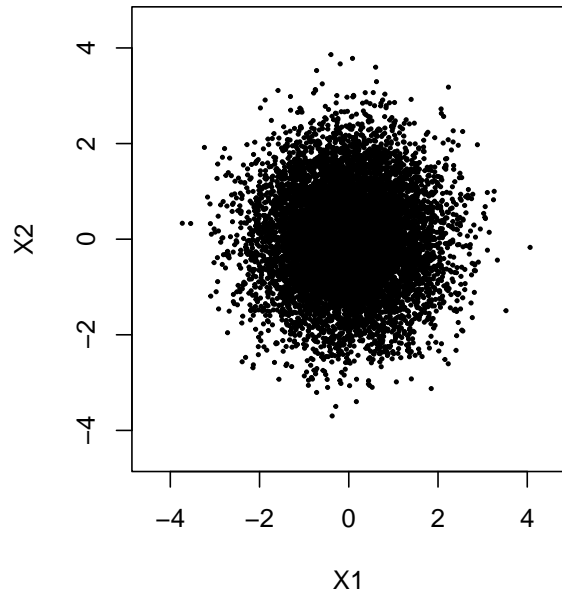
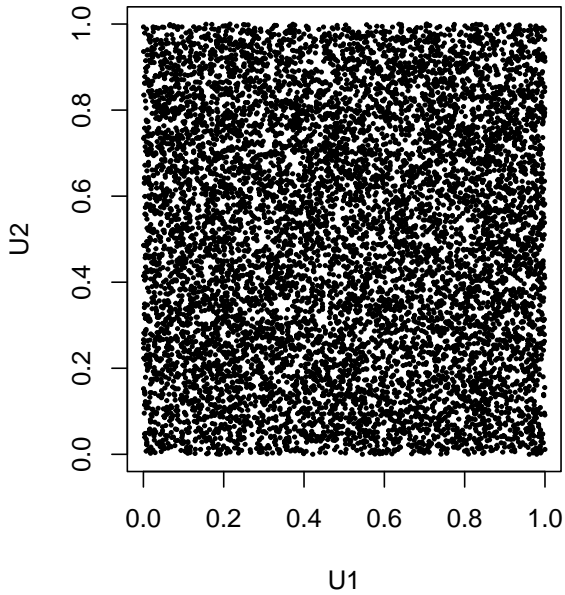
$$X_1 = \sqrt{-2 \log U_1} \cos(2\pi U_2) \quad X_2 = \sqrt{-2 \log U_1} \sin(2\pi U_2).$$

We may use the multivariate transformation theorem to demonstrate that X_1 and X_2 are independent $Normal(0, 1)$ variables.

```

set.seed(8910)
n<-10000
U1<-runif(n);U2<-runif(n)
X1<-sqrt(-2*log(U1))*cos(2*pi*U2)
X2<-sqrt(-2*log(U1))*sin(2*pi*U2)
par(mar=c(4,4,1,2),mfrow=c(1,2))
plot(U1,U2,xlim=range(0,1),ylim=range(0,1),pch=19,cex=0.3)
plot(X1,X2,xlim=range(-4.5,4.5),ylim=range(-4.5,4.5),pch=19,cex=0.3)

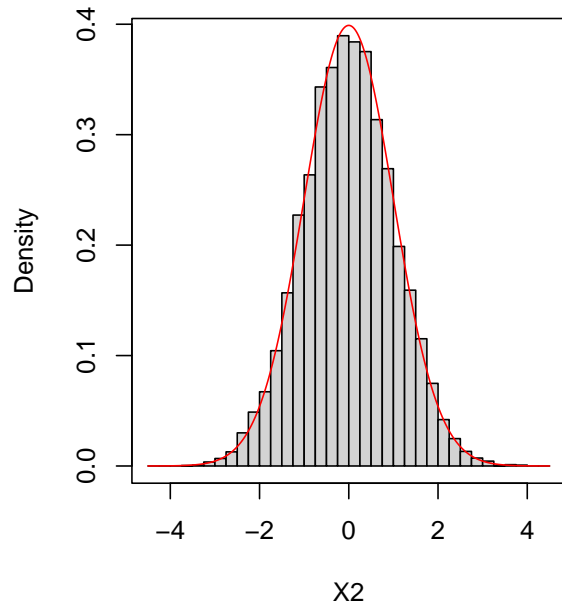
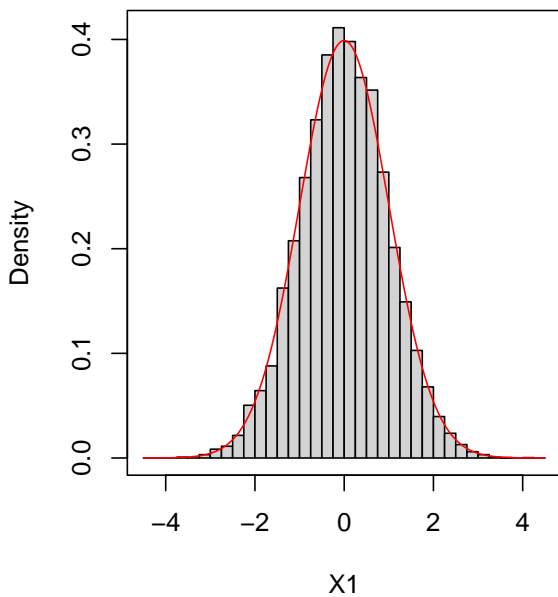
```



```

x<-seq(-4.5,4.5,by=0.001)
fx<-dnorm(x)
hist(X1,breaks=seq(-4.5,4.5,by=0.25),main='',freq=FALSE);box()
lines(x,fx,col='red')
hist(X2,breaks=seq(-4.5,4.5,by=0.25),main='',freq=FALSE);box()
lines(x,fx,col='red')

```



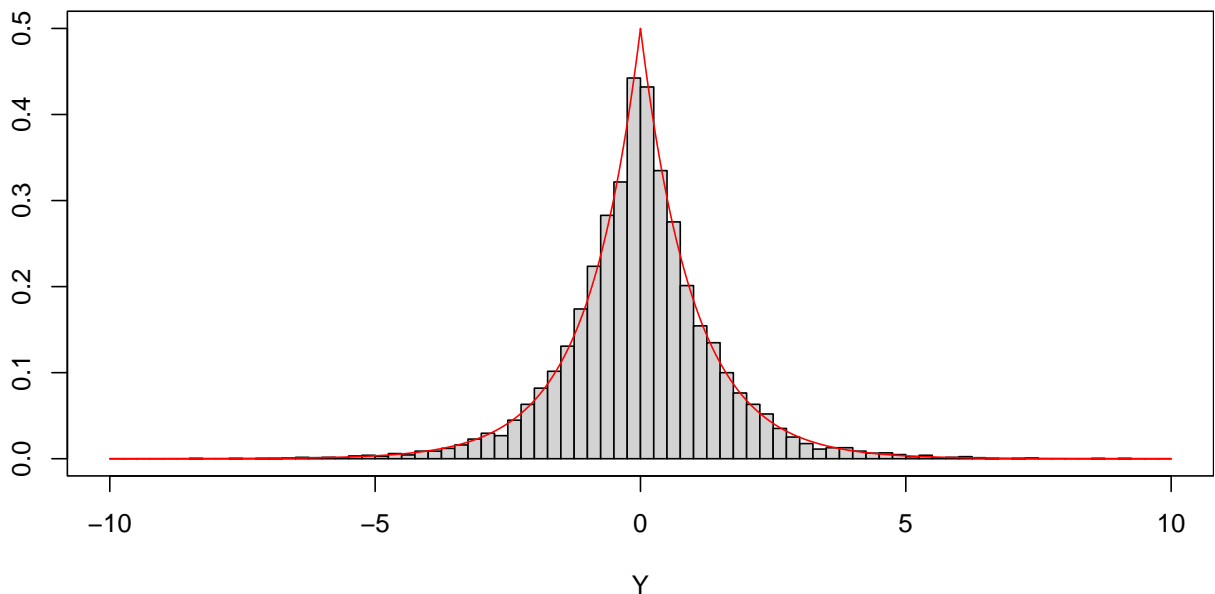
It is also straightforward to study non 1-1 transformations: suppose $X_1, X_2 \sim \text{Exponential}(1)$ are independent. We have seen that

$$Y = X_1 - X_2$$

has a Double Exponential (or Laplace) distribution, with

$$f_Y(y) = \frac{1}{2} \exp\{-|y|\} \quad y \in \mathbb{R}.$$

```
set.seed(8910)
n<-10000
X1<-rexp(n)
X2<-rexp(n)
Y<-X1-X2
par(mar=c(4,3,1,0))
y<-seq(-10,10,by=0.001)
fy<-0.5*exp(-abs(y))
hist(Y,breaks=seq(-10,10,by=0.25),main='',freq=FALSE,ylim=range(0,0.5));box()
lines(y,fy,col='red')
```



This can be useful when the analytical calculation is less straightforward: suppose $X_1, X_2 \sim \text{Gamma}(\alpha, 1)$ are independent. We can study the distribution of

$$Y = X_1 - X_2$$

easily using simulation. By direct but more involved calculation, we can demonstrate that, if α is a positive integer, say $\alpha = r + 1$ for $r = 0, 1, 2, \dots$, then

$$f_Y(y) = \sum_{j=0}^r \binom{r}{j} \frac{\Gamma(r+j+1)}{\{\Gamma(r+1)\}^2} \frac{1}{2^{r+j+1}} |y|^{r-j} \exp\{-|y|\} \quad y \in \mathbb{R}$$

To see this, consider the case $y > 0$ and note that

$$P_Y[Y > y] = P_{X_1, X_2}[X_1 - X_2 > y] = \int_0^\infty \int_{x_2+y}^\infty f_{X_1}(x_1) f_{X_2}(x_2) dx_1 dx_2.$$

From this we compute the density by noting that $F_Y(y) = 1 - P_Y[Y > y]$, and differentiating with respect to y under the first integral: the differentiation is facilitated using the fundamental law of calculus. This yields, for $y > 0$

$$f_Y(y) = \frac{1}{\{\Gamma(\alpha)\}^2} e^{-y} \int_0^\infty (x_2(x_2 + y))^{\alpha-1} \exp\{-2x_2\} dx_2.$$

To compute this integral for $\alpha = r + 1$ an integer, we use the binomial expansion

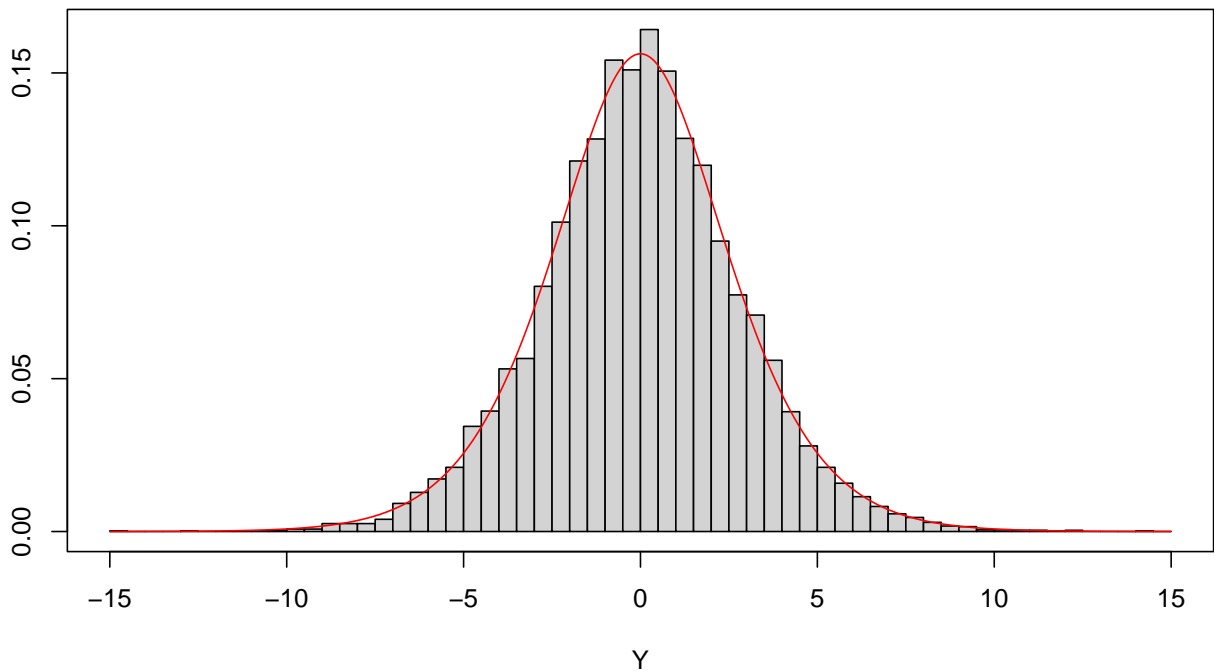
$$(x_2 + y)^r = \sum_{j=0}^r \binom{r}{j} x_2^j y^{r-j}$$

and then integrate term-by-term. We have that

$$\int_0^{\infty} x_2^{r+j} \exp\{-2x_2\} dx_2 = \frac{\Gamma(r+j+1)}{2^{r+j+1}}$$

as the integrand is proportional to a $Gamma(r+j+1, 2)$ pdf. The result follows. Note that the pdf is symmetric about zero as we must have that the distribution of $X_1 - X_2$ is identical to the distribution of $X_2 - X_1$.

```
set.seed(8910)
n<-10000
al<-4
r<-al-1
X1<-rgamma(n,al,1)
X2<-rgamma(n,al,1)
Y<-X1-X2
par(mar=c(4,3,1,0))
y<-seq(-15,15,by=0.001)
ay<-abs(y)
fy<-y*0
for(j in 0:r){
  fy<-fy+choose(r,j)*(gamma(r+j+1)/gamma(r+1)^2)*(2^(-(r+j+1)))*ay^(r-j)*exp(-ay)
}
hist(Y,breaks=seq(-15,15,by=0.5),main='',freq=FALSE);box()
lines(y,fy,col='red')
```



- **Monte Carlo:** The general principle of Monte Carlo is that the availability of random samples from a distribution $F_X(x)$ is essentially equivalent to knowledge of F_X itself if the number of samples is large enough. Suppose X_1, \dots, X_n are independent rvs having the same cdf F_X . We may approximate F_X itself using the *empirical distribution function*, \hat{F}_n , defined by

$$\hat{F}_n(x) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{(-\infty, x]}(X_i) \equiv \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{[X_i, \infty)}(x) \quad x \in \mathbb{R}$$

which, for any fixed x , records the fraction of the X_i s that are less than or equal to x . If n is large enough, we can show that $\hat{F}_n(x)$, when computed with simulated sample values x_1, \dots, x_n , closely approximates $F_X(x)$. Thus we may also approximate the expectation

$$\mathbb{E}_X[g(X)] = \int_{-\infty}^{\infty} g(x) dF_X(x)$$

by the sample average

$$\hat{\mathbb{E}}_X[g(X)] = \int_{-\infty}^{\infty} g(x) d\hat{F}_n(x) = \frac{1}{n} \sum_{i=1}^n g(x_i).$$

This is the same principle that we use in statistical inference when we use the sample mean, sample variance etc. from a data sample to estimate the theoretical mean and variance etc. The discrete distribution

$$\hat{f}_n(x) = \sum_{i=1}^n \frac{1}{n} \mathbb{1}_{\{x\}}(X_i) = \sum_{i=1}^n \frac{1}{n} \mathbb{1}_{\{X_i\}}(x)$$

with masses $1/n$ placed at the X_i s is used to approximate the pmf/pdf of X .

In the previous example, with X_1 and X_2 independent $Gamma(\alpha, 1)$, we may compute the variance of $Y = X_1 - X_2$ directly as $\text{Var}_Y[Y] = \text{Var}_{X_1}[X_1] + \text{Var}_{X_2}[X_2] = \alpha + \alpha = 2\alpha$, and approximate it by Monte Carlo as follows:

```
var(Y) #sample variance
+ [1] 7.809462

2*a1 #theoretical variance.
+ [1] 8
```

Note that because the values are randomly drawn, we will get slightly different numerical results each time. For 10 replicate runs, we get

```
vvec<-replicate(10, var(rgamma(n, a1, 1)-rgamma(n, a1, 1)))
vvec
+ [1] 8.007298 7.952201 8.030130 8.125803 7.950684 7.839052 7.800716 7.868360
+ [9] 8.265122 8.025022
```

If the sample size is increased, the variation becomes smaller.

```
n<-n*10
vvec<-replicate(10, var(rgamma(n, a1, 1)-rgamma(n, a1, 1)))
vvec
+ [1] 7.995375 8.042749 8.109641 7.949860 8.025333 7.962438 8.046584 8.060480
+ [9] 7.964145 7.999656
```

```
n<-n*10
vvec<-replicate(10,var(rgamma(n,a1,1)-rgamma(n,a1,1)))
vvec

+ [1] 7.978249 8.001125 8.017460 7.996612 7.992621 7.995561 7.998518 7.987549
+ [9] 8.002257 8.016414
```

Monte Carlo methods are most often used when there are no straightforward analytic results available. Suppose we have that

$$g(x) = \max \{2|x| \log |x|, 5\}.$$

Then

$$\widehat{\mathbb{E}}_X[g(X)] = \frac{1}{n} \sum_{i=1}^n \max \{2|x_i| \log |x_i|, 5\}$$

where x_1, \dots, x_n are drawn from f_X .

```
n<-100000
ecalc<-function(nv,av){
  xv<-rgamma(nv,av,1)-rgamma(nv,av,1)
  gxv<-pmax(2*abs(xv)*log(abs(xv)),5)
  return(mean(gxv))
}
gvec<-replicate(10,ecalc(n,a1))
gvec

+ [1] 7.582499 7.545870 7.584154 7.558610 7.548019 7.568509 7.548397 7.592875
+ [9] 7.595055 7.542022
```

An extension to basic Monte Carlo is *importance sampling*: we have that

$$\mathbb{E}_X[g(X)] = \int_{-\infty}^{\infty} g(x) \frac{dF_X(x)}{dF_0(x)} dF_0(x)$$

where F_0 is some other distribution, provided the quantity

$$\frac{dF_X(x)}{dF_0(x)} \equiv \frac{f_X(x)}{f_0(x)}$$

is well-defined and finite on the support of X . Thus we may write

$$\mathbb{E}_X[g(X)] = \mathbb{E}_0 \left[g(X) \frac{f_X(X)}{f_0(X)} \right]$$

that is, as an expectation with respect to f_0 .

Cautionary note: Monte Carlo is a powerful technique, but it is not suitable for solving all problems. Consider computing the (Riemann) integral

$$\int_0^1 \frac{1}{x} \sin \left(\frac{2\pi}{x} \right) dx$$

by Monte Carlo. This involves sampling $X_1, \dots, X_n \sim \text{Uniform}(0, 1)$ independently, and then computing

$$\frac{1}{n} \sum_{i=1}^n \frac{1}{x_i} \sin \left(\frac{2\pi}{x_i} \right).$$

The Riemann integral can be computed as

$$\begin{aligned} \int_0^1 \frac{1}{x} \sin \left(\frac{2\pi}{x} \right) dx &= \int_1^{\infty} \frac{\sin(2\pi t)}{t} dt = \int_0^{\infty} \frac{\sin(t)}{t} dt - \int_0^{2\pi} \frac{\sin(t)}{t} dt \\ &= \text{Si}(\infty) - \text{Si}(2\pi) \end{aligned}$$

where $\text{Si}(\cdot)$ is a special function (the *sine integral*) with $\text{Si}(\infty) = \pi/2$. The numerical value of the integral is 0.1526.


```

library(pracma)
Si(10^6)-Si(2*pi) #Riemann integral result

+ [1] 0.1526438

sincalc<-function(nv){
  xv<-runif(nv)
  return(mean(sin(2*pi/xv)/xv))
}
svec<-replicate(20,sincalc(n)) #Monte Carlo replicates
svec

+ [1] 6.2222875 -5.3297809 1.1591927 -1.0311998 1.0686836 -0.2439774
+ [7] -1.5354914 0.4194717 0.9593114 -0.5984831 15.6722360 6.8136477
+ [13] 1.1831975 0.4768715 -1.0018140 1.8673244 0.4550503 -0.0316202
+ [19] 0.3104977 -4.7483663

```

The problem is that if $X \sim \text{Uniform}(0,1)$

$$\mathbb{E}_X \left[\frac{1}{X} \sin \left(\frac{2\pi}{X} \right) \right]$$

is not defined. Specifically

$$\mathbb{E}_X \left[\left| \frac{1}{X} \sin \left(\frac{2\pi}{X} \right) \right| \right]$$

is not finite.