Simulation 2

MX

2025-12-30

Chap 6 Monte Carlo Integration and Variance Reduction

6.2 Monte Carlo Integration

6.2.1 simple Monte Carlo Estimator

Recall that if X is an r.v. with density $f(x)$, then the expectation of X is $E(X) = \int_{-\infty}^{\infty} x f(x) dx$. The expectation of the r.v. $Y = g(X)$ is $E[g(X)] = \int_{-\infty}^{\infty} g(x) f(x) dx$.

If a random sample $X_1, X_2, ..., X_m$ is taken from the distribution of X, an unbiased estimator of $E[g(X)]$ is the sample mean (Note: the "sample size" m in $X_1, X_2, ..., X_m$ should be taken as the number of replications in Monte Carlo simulation, and n will be used to denote the sample size in a replicate. See 6.3)

Now suppose we want to compute $\int_a^b g(x)dx$. We also have $X \sim$ Uniform(a,b) with density $f(x) = \frac{1}{b-a}, a < x < b$ and $= 0$ otherwise. $\int_a^b g(x)dx = (b-a)\int_a^b g(x)\frac{1}{b-a}dx = (b-a)\int_a^b g(x)f(x)dx = (b-a)\int_{-\infty}^{\infty} g(x)f(x)dx = (b-a)E[g(X)]$. That is,

$$\int_a^b g(x)dx = (b-a)E[g(X)]$$

Therefore, integrate $g(x)$ becomes finding its expectation $E[g(X)] := \theta$. The simple Monte Carlo estimator of θ is $\hat{\theta} = \frac{1}{m}\sum_{j=1}^{m} g(X_j)$, which converges to the mean with prob 1 by the Strong LLN. $X_1, X_2, ..., X_j$ **iid are to be generated from Uniform(a,b).**

Comment: much like the Integral Mean Value Theorem, which says if a function $f(x)$ is continuous on $[a, b]$ then there exists some point $c \in (a, b)$ such that $\int_a^b f(x)dx = f(c)(b-a)$. $f(c)$ is the average value of f on $[a, b]$

Example 6.2. Compute an MC estimate of $\int_2^4 e^{-x}dx$. The exact value is $\int_2^4 e^{-x}dx = -e^{-x}|_2^4 = -e^{-4} + e^{-2}$.

In this example, $g(X) = e^{-X}$, $\theta = E[e^{-X}]$, $\hat{\theta} = \frac{1}{m}\sum_{j=1}^{m} e^{-X_j}$. $X_1, X_2, ..., X_j$ iid are to be generated from Uniform(2,4).

```r
m = 10000
x = runif(m, min=2, max=4) # generate X1, ..., Xm
gx = exp(-x)
sample_mean = mean(gx)  # MC estimate of the expectation E(g(X))
integral = (4-2) * sample_mean  # the integral value

cat('MC estimate:', integral, 'exact value:', exp(-2) - exp(-4))
```

```
## MC estimate: 0.1168462 exact value: 0.1170196
```

Example 6.3. Compute $\Phi(a) = \int_{-\infty}^{a} \frac{1}{\sqrt{2\pi}} e^{-t^2/2}dt$ (standard normal cdf) using MC approach. Note the interval is unbounded.

Due to symmetry, consider the case where $a \geq 0$ and estimating a simplified version $\int_0^a e^{-t^2/2}dt := \theta$. (If $a > 0$, $\Phi(a) = 0.5 + \frac{1}{\sqrt{2\pi}}\theta$)

This means generating Uniform(0,a) random numbers for each different a of $\Phi(a)$. But suppose we want to always sample from Uniform(0,1). This can be accomplished through the change of variable: let $x = \frac{1}{a} \cdot t$ and so $t = ax$, then $dt = adx$, $e^{-t^2/2} = e^{-(ax)^2/2}$, $t = 0 \Rightarrow x = 0$, and $t = a \Rightarrow x = 1$.

Therefore

$$\int_0^a e^{-t^2/2}dt = \int_0^1 e^{-(ax)^2/2}adx$$

In this example, $g(X) = ae^{-(aX)^2/2}$, $\theta = E[ae^{-(aX)^2/2}]$, $\hat{\theta} = \frac{1}{m}\sum_{j=1}^m ae^{-(aX_j)^2/2}$. $X_1, X_2, ..., X_j$ iid are to be generated from Uniform(0,1).

```
a = seq(0.1, 2.5, length=10)
m = 10000
x = runif(m)  # generate X1, ..., Xm
cdf = numeric(length = length(a))
for (i in 1:length(a)){
  g = a[i] * exp(-(a[i] * x)^2 / 2)
  sample_mean = mean(g)
  integral = (1 - 0)*sample_mean
  cdf[i] = 0.5 + integral / sqrt(2*pi)  # get the cdf
}

# compare results
phi = pnorm(q=a)
round(rbind(a, cdf, phi), 3)
```

```
##     [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
## a   0.10 0.367 0.633 0.900 1.167 1.433 1.700 1.967 2.233 2.500
## cdf 0.54 0.643 0.737 0.816 0.878 0.924 0.955 0.975 0.987 0.993
## phi 0.54 0.643 0.737 0.816 0.878 0.924 0.955 0.975 0.987 0.994
```

The MC estimates are close to the pnorm() but look worse in the upper tail of the distribution.

The "hit-or-miss" approach to Monte Carlo integration

Suppose X has the density $f(x)$ and cdf $F(x)$. Consider a random variable $Y = I(X \leq x)$, $I()$ being the indicator function defined as $I(X \leq x) = \begin{cases} 1, & X \leq x \\ 0, & X > x \end{cases}$. Y follows a Bernoulli distribution: $\begin{pmatrix} 1 & 0 \\ P(X \leq x) & P(X > x) \end{pmatrix}$. Thus $E(Y) = P(X \leq x) = F(x) = \int_{-\infty}^x f(x)dx$. That is,

$$\int_{-\infty}^x f(x)dx = E(Y)$$

Therefore, to estimate the integral (cdf), find the expectation $E(Y) := \theta$. The Monte Carlo estimator of θ is $\hat{\theta} = \frac{1}{m}\sum_{j=1}^m Y_j$. $Y_j = I(X_j \leq x)$, with $X_1, ..., X_m$ distributed according to $f(x)$.

Note that the estimator $\frac{1}{m}\sum_{j=1}^m Y_j$ here is actually a proportion $\hat{p}$. We essentially approximate the $\int_{-\infty}^x f(x)dx$ with a *proportion* of samples that fall in the "shaded" area (area of integration).

Example 6.4. Ex 6.3 cont. Use the approach above to estimate the std. normal cdf $\Phi(a)$

Sol. Let $Z \sim$ N(0,1) with density $f(z)$. And note $\Phi(a)$ is integral $\int_{-\infty}^a f(z)dz$. Apply the procedure above: estimate the expectation $\int_{-\infty}^a f(z)dz = E(Y)$ using the Monte Carlo estimator $\frac{1}{m}\sum_{j=1}^m Y_j$ where $Y_j = I(Z_j \leq x)$.

```
a = seq(0.1, 2.5, length = 10)
m = 10000
z = rnorm(m)  # generate Z1,..., Zm
dim(a) = length(a)  # turn a, a vector, into a 10x1 matrix

est = apply(a, MARGIN = 1,  # apply FUN row wise (call FUN for each row of a)
     FUN = function(a,z){mean(z <= a)}, # calculate the proportion
    z=z)  # supply additional argument

phi = pnorm(a)
round(rbind(a, est, phi), 3)
```

```
##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
## a   0.100 0.367 0.633 0.900 1.167 1.433 1.700 1.967 2.233 2.500
## est 0.542 0.651 0.744 0.821 0.887 0.930 0.959 0.976 0.988 0.995
## phi 0.540 0.643 0.737 0.816 0.878 0.924 0.955 0.975 0.987 0.994
```

The variance of the simple MC estimator $\hat{\theta}$ (ref. 6.2.1)

The variance is $Var(\hat{\theta}) = Var(\frac{1}{m}\sum_{i=1}^m g(X_j)) = \frac{1}{m^2} \cdot Var(\sum_{i=1}^m g(X_j)) = \frac{1}{m^2} \cdot m \cdot Var(g(X)) = \frac{1}{m}Var(g(X))$. That is,

$$Var(\hat{\theta}) = \frac{1}{m}Var(g(X))$$

$Var(g(X))$ can be estimated using the sample variance S^2: $S^2 = \frac{1}{m}\sum_{i=1}^m [g(x_j) - \overline{g(x)}]^2$

Therefore

$$\widehat{Var(\hat{\theta})} = \frac{1}{m}S^2 = \frac{1}{m^2}\sum_{j=1}^m [g(x_j) - \overline{g(x)}]^2$$

CLT implies that $\frac{\hat{\theta} - E(\hat{\theta})}{\sqrt{Var(\hat{\theta})}}$ converges in distr to $N(0,1)$ as $n \to \infty$. ... Thus can compute error bounds on the MC estimate of the integral.

6.2.2 Variance and Efficiency

Looking at $\widehat{Integral} = (b-a)\hat{\theta}$, it is apparent that $Var(\widehat{Int}) = (b-a)^2 \cdot Var(\hat{\theta}) = \frac{(b-a)^2}{m}Var(g(X))$

In the hit-or-miss approach $Y = I(X \leq x)$ is a Bern(p) rv with $p = P(X \leq x) = F(x)$, and the variance of the estimator $\hat{\theta} = \frac{1}{m}\sum_{j=1}^m Y_j$ is $Var(\hat{\theta}) = Var(\frac{1}{m}\sum_{j=1}^m Y_j) = \frac{1}{m}Var(Y)$. Because $Y \sim Bern(p)$, $Var(\hat{\theta}) = \frac{1}{m}p(1-p)$, which is the greatest when $p = 1/2$, that is, $F(x) = 1/2$.

Ex 6.5. Estimate the variance of the estimator in Example 6.4

```
a = seq(from = 0.1, to = 1, by = 0.1)
m = 1000
cdf = numeric(length(a))
v_sample = numeric(length(a))

z = rnorm(m)  # z1, z2,..., zn
for (i in 1:length(a)){
  y = (z <= a[i])  # y ~ Bern
  cdf[i] = mean(y)  # compute proportion / mean
  v_sample[i] = mean((y - mean(y))^2) / m  # variance of the estimator
}

phi = pnorm(a)  # theoretical cdf values, = p in Bern(p)
v_theory = phi * (1-phi) / m

round(rbind(phi, v_theory * 100, v_sample * 100), 4)
```

```
##       [,1]   [,2]   [,3]   [,4]   [,5]   [,6]   [,7]   [,8]   [,9]  [,10]
## phi 0.5398 0.5793 0.6179 0.6554 0.6915 0.7257 0.7580 0.7881 0.8159 0.8413
##     0.0248 0.0244 0.0236 0.0226 0.0213 0.0199 0.0183 0.0167 0.0150 0.0133
##     0.0248 0.0242 0.0235 0.0223 0.0210 0.0194 0.0176 0.0154 0.0136 0.0119
```

```
round(rbind(a, phi, cdf), 3)
```

```
##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
## a   0.100 0.200 0.300 0.400 0.500 0.600 0.700 0.800 0.900 1.000
## phi 0.540 0.579 0.618 0.655 0.691 0.726 0.758 0.788 0.816 0.841
## cdf 0.548 0.587 0.623 0.664 0.700 0.737 0.772 0.810 0.838 0.862
```

6.3 Variance Reduction

The Monte Carlo approach to estimating $\theta = E[g(X_1, ..., X_n)]$ is to compute the sample mean $\hat{\theta} = \frac{1}{m}\sum_{j=1}^{m} g(X_1^{(j)}, ..., X_n^{(j)})$ for a large number m of replicates. The function $g()$ is often a statistic, an n-variate function of a sample.

Let

$$Y_j = g(X_1^{(j)}, ..., X_n^{(j)}),\ j = 1, 2, ..., m$$

and it is a replicate. $Y_1, ..., Y_m$ are iid with distribution of $g(X_1, ..., X_n)$.

The MC estimator

$$\hat{\theta} = \frac{1}{m}\sum_{i=1}^{m} g(X_1^{(j)}, ..., X_n^{(j)}) = \frac{1}{m}\sum_{i=1}^{m} Y_j = \bar{Y}$$

has variance $Var(\hat{\theta}) = Var(\bar{Y}) = \frac{1}{m}Var(Y) = \frac{1}{m}Var(g(X_1, ..., X_n))$

Increasing the number of replicates m can reduce the variance of the Monte Carlo estimator. However, a large increase is needed to get even a small improvement in variance. Other methods are shown as follows.

6.4 Antithetic Variables

Suppose that $(X_1, ..., X_n)$ are simulated via the inverse transform method. For each of the m replicates we generate $U_i \sim$ Uniform(0,1) and compute $X_i^{(j)} = F_X^{-1}(U_i^{(j)})$, $i = 1, ..., n$. If U is generated on (0,1) then $1 - U$ has the same distr as U and they are negatively correlated. Then

$$Y_j = g(X^{(j)}) = g(F_X^{-1}(U^{(j)}))$$

has the same distr as

$$Y_j^{'} = g(F_X^{-1}(1 - U^{(j)}))$$

(We consider $n = 1$ case for clarity and simplicity.) This is because $U \overset{d}{=} 1 - U$, $F^{-1}(U) \overset{d}{=} F^{-1}(1 - U)$, applying g gives $g(F^{-1}(U)) \overset{d}{=} g(F^{-1}(1 - U))$, that is $Y_j \overset{d}{=} Y_j^{'}$

Further, it is proved (see last page) that Y_j and $Y_j^{'}$ are negatively correlated when the function g is *monotone*.

The antithetic variable approach is easy to apply. **If m MC replications are required, generate m/2 replicates $Y_j = g(F_X^{-1}(U^{(j)}))$, $j = 1, ..., m/2$, and the remaining m/2 replicates** $Y_j^{'} = g(F_X^{-1}(1 - U^{(j)}))$. And **the antithetic estimator of $\theta = E[g(X)]$ is**

$$\hat{\theta} = \frac{1}{m}(Y_1 + Y_1^{'} + ... + Y_{m/2} + Y_{m/2}^{'}) = \frac{1}{m}\sum_{j=1}^{m/2}(Y_j + Y_j^{'})$$

Note, because $E(Y_j) = E(g(F_X^{-1}(U^{(j)})) = E(g(X))$, and $E(Y_j^{'}) = E(g(F_X^{-1}(1 - U^{(j)})) = E(g(X))$, Y_j and $Y_j^{'}$ are unbiased. The variance of $\hat{\theta}$ is reduced due to negative correlation between Y_j and $Y_j^{'}$. Further, $\frac{m}{2} \cdot n$ rather than mn uniform variates are required.

Example 6.6. Refer to Example 6.3 illustrating MC integration applied to estimate the standard normal cdf $\Phi(x)$. Estimate using the antithetic variables and find the approximate reduction in standard error.

After change of variables and some simplification, we were estimating $\int_0^1 e^{-(ax)^2/2}adx$. And $g(X) = ae^{-(aX)^2/2}, X \sim U(0, 1)$, $\theta = E[ae^{-(aX)^2/2}]$, $\hat{\theta} = \frac{1}{m}\sum_{j=1}^{m} ae^{-(aX^{(j)})^2/2}, X^{(1)}, ..., X^{(m)} \sim U(0, 1)$.

Now since g is monotone when $x > 0$, Corollary 6.1 is satisfied and we can use the **antithetic variable approach**: generate $U^{(1)}, ..., U^{(m/2)}$ from Uniform(0,1). Because $X \sim U(0, 1)$ so $F_X^{-1} = F_U^{-1}$, inverse transform is not needed, and we can compute half of the replicates $Y_j = g(U^{(j)}) = ae^{-(aU^{(j)})^2/2}$ and the other half $Y_j^{'} = ae^{-(a(1-U^{(j)}))^2/2}$.

The antithetic approach estimator is $\hat{\theta} = \frac{1}{m}\sum_{j=1}^{m/2}(Y_j + Y_j^{'})$.

```
MC_Phi = function(x, R = 10000, antithetic = T){
  u = runif(R/2)
  if (antithetic){
    v = 1 - u
    } else{
```

```r
      v = runif(R/2)
    }
  u = c(u, v)
  cdf = numeric(length(x))
  for (i in 1:length(x)){
    g = x[i] * exp(- (x[i]*u)^2 / 2)
    sample_mean = mean(g)
    cdf[i] = sample_mean / sqrt(2 * pi) + 0.5
  }
  return(cdf)  # return estimated cdf value Phi(x)
  }

# compare results
x = seq(0.1, 2.5, length = 5)
Phi = pnorm(x)
set.seed(123)
mc_noanti = MC_Phi(x, antithetic = F)
set.seed(123)
mc_anti = MC_Phi(x, antithetic = T)
round(rbind(x, mc_noanti, mc_anti, Phi), 5)
```

```
##              [,1]    [,2]    [,3]    [,4]    [,5]
## x         0.10000 0.70000 1.30000 1.90000 2.50000
## mc_noanti 0.53983 0.75825 0.90418 0.97311 0.99594
## mc_anti   0.53983 0.75805 0.90325 0.97132 0.99370
## Phi       0.53983 0.75804 0.90320 0.97128 0.99379
```

```r
# compute variance reduction using simulation, at given x
m = 1000
MC_noanti = MC_anti = numeric(m)
x = 1.95  # given x, as in cdf F(x)

for (i in 1:m){  # run function m times, to get a series of estimated Phi(x)
  MC_noanti[i] = MC_Phi(x, antithetic = F)
  MC_anti[i] = MC_Phi(x, antithetic = T)
}

cat('antithetic approach SD:', sd(MC_anti), 'simple approach SD:', sd(MC_noanti))
```

```
## antithetic approach SD: 0.0001486924 simple approach SD: 0.002132755
```

```r
cat('\n variance reduction:', (var(MC_noanti)-var(MC_anti))/var(MC_noanti))
```

```
##
##  variance reduction: 0.9951393
```

6.5 Control Variates

Another approach to reducing the variance in a MC estimator of $\theta = E[g(X)]$ is using control variates. Suppose there is a function f, such that $\mu = E[f(X)]$ is known and $f(X)$ is correlated with $g(X)$. Then for any constant c, the *per-replication* estimator

$$\hat{\theta}_c = g(X) + c(f(X) - \mu)$$

$\hat{\theta}_c$ is unbiased for θ: $E(\hat{\theta}_c) = E[g(X)+c(f(X)-\mu)] = E[g(X)]+c(E[f(X)]-\mu)) = \theta$. (Note this is more of a random variable $Z_c = g(X)+c(f(X)-\mu)$ than an estimator $\hat{\theta}_c = \frac{1}{m}\sum_{j=1}^{m}[g(X^{(j)})+c(f(X^{(j)})-\mu)]$. In Monte Carlo, first define a single-run estimator Z_c and then the MC estimator is its sample mean; variance reduction works by modifying each replication not the average afterward.)

The variance is $Var(\hat{\theta}_c) = Var[g(X) + c(f(X) - \mu)] = Var[g(X)] + c^2 Var[f(X)] + 2c \cdot Cov(g(X), f(X)) = Var[f(X)]c^2 + 2Cov(g(X), f(X)) \cdot c + Var[g(X)]$. That is, the variance of the estimator is is a quadratic function of c, minimized at

$$c = c^* = -\frac{Cov(g(X), f(X))}{Var[f(X)]}$$

and minimum variance is

$$Var(\hat{\theta}_{c^*}) = \frac{-Cov^2(g(X), f(X))}{Var[f(X)]} + Var[g(X)] = Var[g(X)] \cdot (1 - \rho^2_{f,g})$$

The rv $f(X)$ is called a *control variate* for $g(X)$. $Var[g(X)]$ is reduced by $\frac{-Cov^2(g(X),f(X))}{Var[f(X)]}$.

The percent reduction is

$$\frac{Var[g(X)] - Var(\hat{\theta}_{c^*})}{Var[g(X)]} = \frac{Cov^2(g(X), f(X))}{Var[f(X)] \cdot Var[g(X)]} = \rho^2_{f,g}$$

Thus it is advantageous if $f(X)$ and $g(X)$ are strongly correlated.

To compute the constant c^* we need $Cov(g(X), f(X))$ and $Var[f(X)]$. These parameters can be estimated from a preliminary Monte Carlo experiment.

Ex 6.7. Apply the control variate approach to compute $\int_0^1 e^u du = E[e^U] = \theta$, $U \sim U(0,1)$

$g(U) = e^U$, $\theta = E[e^U]$. By integration $\theta = e^u|_0^1 = e - 1$. However this example will allow us to verify the control variate approach is correctly implemented.

Under the **simple** MC approach with m replicates, the estimator is $\hat{\theta} = \frac{1}{m}\sum_{j=1}^m g(U^{(j)})$. Its variance is $\frac{1}{m}Var(g(U))$ where $Var(g(U)) = Var(e^U) = E(e^{2U}) - (Ee^U)^2 = \frac{1}{2}e^{2u}|_0^1 - (e-1)^2 = \frac{1}{2}(e^2 - 1) - (e-1)^2 = \mathbf{0.242035}$.

Now let $f(U) = U$ be the control variate. The new estimator is $\hat{\theta}_c = g(X) + c(f(X) - E[(f(X)]) = e^U + c(U - \frac{1}{2})$, whose variance is minimized at $c^* = -\frac{Cov(U,e^U)}{Var(U)}$. We can calculate that $E(U) = 1/2$, $Var(U) = 1/12$, and $Cov(U, e^U) = E(Ue^U) - E(U)E(e^U) = \int_0^1 ue^u du - \frac{1}{2}(e-1) = 1 - \frac{1}{2}(e-1) = \frac{3}{2} - \frac{1}{2}e = \mathbf{0.140859}$. Thus $c^* = -\frac{\frac{3}{2} - \frac{1}{2}e}{1/12} = -18 + 6e = -\mathbf{1.69030}$.

Therefore the controlled estimator with minimum variance is $\hat{\theta}_{c^*} = e^U + c^*(U - \frac{1}{2})$, and $Var(\hat{\theta}_{c^*}) = \frac{-Cov^2(e^U,U)}{Var(U)} + Var(e^U) = \frac{-0.140859^2}{1/12} + 0.242035 = \mathbf{0.003940}$. The percentage reduction is $\frac{Cov^2(e^U,U)}{Var(U) \cdot Var(e^U)} = \frac{0.140859^2}{\frac{1}{12}0.242035} = 0.9837 = 98.37\%$.

```r
m = 10000
c_star = -1.69030
set.seed(1)
u = runif(m)
g = exp(u)  # simple estimator (per-replication)
g_controlled =  exp(u) + c_star * (u - 1/2)  # control variate estimator

cat('exact value:', exp(1)-1, 'simple MC:', mean(g), 'control variate MC:', mean(g_controlled))
```

```
## exact value: 1.718282 simple MC: 1.719776 control variate MC: 1.719492
```

```r
cat('\nvariance reduction:', (var(g) - var(g_controlled)) / var(g) )
```

```
##
## variance reduction: 0.9837743
```

The reduction in variance is 98.377% close to the derived value 98.37%.

6.5.1 Antithetic Variate as Control Variate

First notice that the control variate estimator is a linear combination of unbiased estimators ($\hat{\theta}_1$ and $\hat{\theta}_2$) of θ:

$$\hat{\theta}_c = g(X) + c(f(X) - \mu) = (1-c)g(X) + c \cdot (g(X) + f(X) - \mu) = (1-c)\hat{\theta}_1 + c\hat{\theta}_2$$

where c is a constant. $\hat{\theta}_c$ is unbiased. In general if two estimators of θ are unbiased, then their linear combination is also unbiased.

In Section 6.4 it was shown that $Y_j = g(F_X^{-1}(U^{(j)}))$ and $Y_j^{'} = g(F_X^{-1}(1 - U^{(j)})))$ are unbiased. Therefore the antithetic estimator is a special case of control variates where $c = 0.5$:

$$\hat{\theta} = \frac{1}{2}(Y_j + Y_j^{'}) = 0.5\hat{\theta}_1 + 0.5\hat{\theta}_2$$

(Here we use per-replication estimator, the original estimator is $\hat{\theta} = \frac{2}{m}\sum_{j=1}^{m}\frac{1}{2}(Y_j + Y_j^{'}))$

The variance of $\hat{\theta}_c$ is

$$Var(\hat{\theta}_c) = Var[(1-c)\hat{\theta}_1 + c\hat{\theta}_2] = Var[c(\hat{\theta}_2 - \hat{\theta}_1) + \hat{\theta}_1] = Var(\hat{\theta}_1) + c^2 Var(\hat{\theta}_2 - \hat{\theta}_1) + 2cCov(\hat{\theta}_2 - \hat{\theta}_1, \hat{\theta}_1)$$

6.5.2 Several Control Variates

6.5.3 Control Variates and Regression

6.6 Importance Sampling

6.7 Statified Sampling

6.8 Statified Importance Sampling

Skipped in 6.4

Definitions:
1. If $x_i \leq y_i$, $i = 1, ..., n$, define $(x_1, ..., x_n) \leq (y_1, ..., y_n)$. 2. An n-variate function $g = g(X_1, ..., X_n)$ is *increasing* if it is increasing in its coordinates/arguments. That is, g is increasing if $g(x_1, ..., x_n) \leq g(y_1, ..., y_n)$ whenever $(x_1, ..., x_n) \leq (y_1, ..., y_n)$. g preserves the ordering. Similarly, g is *decreasing* if it is decreasing in its coordinates. 3. g is *monotone* if it is decreasing or increasing.

Proposition 6.1. If $\mathbf{X} = (X_1, ..., X_n)$ are independent, and f and g are increasing functions, then $E[f(\mathbf{X})g(\mathbf{X})] \geq E[f(\mathbf{X})] \cdot E[g(\mathbf{X})]$

Corollary 6.1. If $g = g(X_1, ..., X_n)$ is monotone, then

$$Y = g(F_X^{-1}(U_1), ..., F_X^{-1}(U_n))$$

and

$$Y' = g(F_X^{-1}(1 - U_1), ..., F_X^{-1}(1 - U_n))$$

are negatively correlated.

One-dimension intuition: let $Y = g(F_X^{-1}(U))$ and $Y' = g(F_X^{-1}(1 - U))$. When U is large and $1 - U$ is small, $F_X^{-1}(U)$ is large and $F_X^{-1}(1 - U)$ is small, so if g is monotone for example increasing, $Y = g(F_X^{-1}(U))$ is large and $Y' = g(F_X^{-1}(1 - U))$ is small. Hence the negative correlation.