

Variance Reduction Methods I

1 Simulation Efficiency

Suppose as usual that we wish to estimate $\theta := E[h(\mathbf{X})]$. Then the standard simulation algorithm is:

1. Generate $\mathbf{X}_1, \dots, \mathbf{X}_n$
2. Estimate θ with $\hat{\theta}_n = \sum_{j=1}^n Y_j / n$ where $Y_j := h(\mathbf{X}_j)$.
3. Approximate $100(1 - \alpha)\%$ confidence intervals are then given by

$$\left[\hat{\theta}_n - z_{1-\alpha/2} \frac{\hat{\sigma}_n}{\sqrt{n}}, \hat{\theta}_n + z_{1-\alpha/2} \frac{\hat{\sigma}_n}{\sqrt{n}} \right]$$

where $\hat{\sigma}_n$ is the usual estimate of $\text{Var}(Y)$ based on Y_1, \dots, Y_n .

One way to measure the quality of the estimator, $\hat{\theta}_n$, is by the *half-width*, HW , of the confidence interval. For a fixed α , we have

$$HW = z_{1-\alpha/2} \sqrt{\frac{\text{Var}(Y)}{n}}.$$

We would like HW to be small, but sometimes this is difficult to achieve. This may be because $\text{Var}(Y)$ is too large, or too much computational effort is required to simulate each Y_j so that n is necessarily small, or some combination of the two. As a result, it is often imperative to address the issue of *simulation efficiency*.¹ There are a number of things we can do:

1. Develop a good simulation algorithm
2. Program carefully to minimize storage requirements. For example we do not need to store all the Y_j 's: we only need to keep track of $\sum Y_j$ and $\sum Y_j^2$ to compute $\hat{\theta}_n$ and approximate CI's
3. Program carefully to minimize execution time
4. Decrease the variability of the simulation output that we use to estimate θ . The techniques used to do this are usually called *variance reduction* techniques

We will now study various variance reduction techniques, and assume that we are doing items (1) to (3) as well as possible. Before proceeding to study these techniques, however, we should first describe a *measure of simulation efficiency*.

¹Recall that as a tool, simulation is often used to solve problems that are too hard to solve either explicitly or numerically. These problems are therefore often very demanding from a computational point of view. The argument that we will not have to care about simulation efficiency as computers become ever faster is spurious: our goals will simply adapt so that we will want to solve ever harder problems.

1.1 Measuring Simulation Efficiency

Suppose there are two random variables, W and Y , such that $E[W] = E[Y] = \theta$. Then we could choose to either simulate W_1, \dots, W_n or Y_1, \dots, Y_n in order to estimate θ . Let M_w denote the method of estimating θ by simulating the W_i 's. M_y is similarly defined. Which method is more efficient, M_w or M_y ? To answer this, let n_w and n_y be the number of samples of W and Y , respectively, that are needed to achieve a half-width, HW . Then it is easy to see that

$$\begin{aligned} n_w &= \left(\frac{z_{1-\alpha/2}}{HW} \right)^2 \text{Var}(W) \\ n_y &= \left(\frac{z_{1-\alpha/2}}{HW} \right)^2 \text{Var}(Y) \end{aligned}$$

Let E_w and E_y denote the amount of computational effort required to produce one sample of W and Y , respectively. Then the total effort expended by M_w to achieve a half width HW is

$$TE_w = \left(\frac{z_{1-\alpha/2}}{HW} \right)^2 \text{Var}(W) E_w.$$

Similarly, the effort expended by M_y to achieve the same half-width HW , is given by

$$TE_y = \left(\frac{z_{1-\alpha/2}}{HW} \right)^2 \text{Var}(Y) E_y.$$

We say that M_w is *more efficient* than M_y if

$$TE_w < TE_y.$$

Note that $TE_w < TE_y$ if and only if

$$\text{Var}(W)E_w < \text{Var}(Y)E_y. \quad (1)$$

We will use the quantity $\text{Var}(W)E_w$ as a measure of the efficiency of the simulator, M_w .² Note that expression (1) implies that we cannot conclude that one simulation algorithm, M_w , is better than another, M_y , simply because $\text{Var}(W) < \text{Var}(Y)$; we also need to take E_w and E_y into consideration. However, it is often the case that we have two simulators available to us, M_w and M_y , where $E_w \approx E_y$ and $\text{Var}(W) \ll \text{Var}(Y)$. In such cases it is clear that using M_w provides a substantial improvement over using M_y .³

2 Common Random Numbers

The common random numbers technique is different from the other methods we will study in that it is used for simulation studies where two or more systems are being compared. The easiest way to explain the idea of common random numbers is by way of example.

Example 1 (Comparing Two Queueing Systems)

Consider a queueing system where customers arrive according to a Poisson process, $N(t)$. The operator of the system needs to install a server to service the arrivals and he has a choice of two possible servers, M and N . In the event that M is chosen, let S_i^m denote the *service time* of the i^{th} customer, and let X^m denote the total waiting time in the system of all the customers who arrive before time T . That is,

$$X^m = \sum_{i=1}^{N(T)} W_i^m$$

²Note that it does not depend on α .

³In what follows, we usually have in mind that M_y represents the usual simulation algorithm that we have been using up to now.

where W_i^m is the *total time* in the system of the i^{th} customer. This implies $W_i^m = S_i^m + Q_i^m$ where Q_i^m is the waiting time *before* being served for the i^{th} customer. S_i^n , X_i^n , W_i^n and Q_i^n are all defined in the same way for server N . The operator wants to estimate

$$\theta = E[X^m] - E[X^n].$$

Perhaps the obvious way to estimate θ is to estimate $\theta_m := E[X^m]$ and $\theta_n := E[X^n]$ *independently* of one another, and then set $\hat{\theta} = \hat{\theta}_m - \hat{\theta}_n$. The variance of $\hat{\theta}$ is then given by

$$\text{Var}(\hat{\theta}) = \text{Var}(\hat{\theta}_m) + \text{Var}(\hat{\theta}_n).$$

But maybe we can do better by computing $\hat{\theta}_m := E[X^m]$ and $\hat{\theta}_n := E[X^n]$ in such a way that $\hat{\theta}_m$ and $\hat{\theta}_n$ are *dependent*. If this is the case, then

$$\text{Var}(\hat{\theta}) = \text{Var}(\hat{\theta}_m) + \text{Var}(\hat{\theta}_n) - 2\text{Cov}(\hat{\theta}_m, \hat{\theta}_n).$$

So if we can arrange it that $\text{Cov}(\hat{\theta}_m, \hat{\theta}_n) > 0$, then we can achieve a variance reduction. Sometimes we can achieve a significant variance reduction using the following general idea.

Let X_1^m, \dots, X_r^m and X_1^n, \dots, X_r^n be the sets of r samples that we use to estimate $E[X^m]$ and $E[X^n]$, respectively. Now set

$$Z_i := X_i^m - X_i^n, \quad i = 1, \dots, r.$$

If the Z_i 's are IID, then

$$\begin{aligned} \hat{\theta} &= \frac{\sum_{i=1}^r Z_i}{r} \\ \text{Var}(\hat{\theta}) &= \frac{\text{Var}(X_i^m) + \text{Var}(X_i^n) - 2\text{Cov}(X_i^m, X_i^n)}{r}. \end{aligned}$$

So to reduce $\text{Var}(\hat{\theta})$, we would like to make $\text{Cov}(X_i^m, X_i^n)$ as large as possible. We can generally achieve this by using the *same* set of random numbers to generate X_i^m and X_i^n . For example, we should use the same arrival sequences for each possible server. We can do more: while S_i^m and S_i^n will generally have different distributions we might still be able to arrange it so that S_i^m and S_i^n are positively correlated. For example, if they are generated using the inverse transform method, we should use the same $U_i \sim U(0, 1)$ to generate both S_i^m and S_i^n . Since the inverse of the CDF is monotonic, this means that S_i^m and S_i^n will in fact be positively correlated. By using *common* random numbers in this manner and *synchronizing* them correctly as we have described, it should be the case that X_i^m and X_i^n are strongly positively correlated. For example, if X_i^m is large, then that would suggest that there have been many arrivals in $[0, T]$ and / or service times have been very long. But then the same should be true for the system when N is the server, implying that X_i^n should also be large. ■

This example demonstrates the idea of common random numbers. While in general it cannot always be guaranteed to work, i.e. decrease the variance, it is often very effective, sometimes decreasing the variance by orders of magnitude. The philosophy of the method is that comparisons of the two systems should be made “under similar experimental conditions”.

2.1 Common Random Numbers in Financial Engineering

Common random numbers can also be used to estimate the *delta* of an option when it cannot be computed explicitly. For example, let $C(S)$ be the price of a particular option when the value of an underlying security is S . Then

$$\Delta := \frac{\partial C}{\partial S}$$

is the *delta* of the option and it measures the sensitivity of the option price to changes in the price, S_t , of the underlying security. If, for example, C is the value of a standard European call option in the Black-Scholes

framework then we can compute Δ explicitly. There are times, however, when this is not possible, and in such circumstances we might instead use simulation.

Let $\epsilon > 0$ be given. Then the finite-difference ratio

$$\Delta_\epsilon := \frac{C(S + \epsilon) - C(S - \epsilon)}{2\epsilon}$$

should be close to Δ when ϵ is small. We could therefore estimate Δ by estimating Δ_ϵ . In that case, $C(S + \epsilon)$ and $C(S - \epsilon)$ should **not** be estimated independently but instead, they should be estimated using the same set of *common random numbers*. For small ϵ , the variance reduction from using common random numbers can be dramatic.

Remark 1 *Despite the improvements that you obtain from using common random numbers, the finite-difference method we have just described is usually not the best way to estimate Δ . Noting that $C = E[h(S)]$ for some function, $h(\cdot)$, we see that computing Δ amounts to computing*

$$\Delta = E \left[\frac{\partial h(S)}{\partial S} \right]$$

if we can justify switching the order of expectation and differentiation. In practice, we can often do this, and as a result, estimate Δ more efficiently and without the bias of the finite-difference method. See Glasserman (2004) for further details as well as a survey of Monte Carlo methods applied to security pricing.

3 Control Variates

Suppose you wish to determine the mean midday temperature, θ , in Gagaland and that your data consists of

$$\{(T_i, R_i) : i = 1, \dots, n\},$$

where T_i and R_i are the midday temperature and daily rainfall, respectively, on some random day, D_i . Then $\theta = E[T]$ is the mean midday temperature.

Question: What estimator would you use for θ ?

Answer: If the D_i 's are drawn uniformly from $\{1, \dots, 365\}$, then the obvious choice is to set

$$\hat{\theta}_n = \frac{\sum_{i=1}^n T_i}{n}$$

and we then know that $E[\hat{\theta}_n] = \theta$. Suppose, however, that we also know:

1. $E[R]$, the mean daily rainfall in Gagaland
2. R_i and T_i are *dependent*; in particular, it tends to rain more in the cold season

Is there any way we can exploit this information to obtain a better estimate of θ ? The answer of course, is yes.

Let $\bar{R}_n := \sum_{i=1}^n R_i / n$ and now suppose $\bar{R}_n > E[R]$. Then this implies that the D_i 's over-represent the rainy season in comparison to the dry season. But since the rainy season tends to coincide with the cold season, it also means that the D_i 's over-represent the cold season in comparison to the warm season. As a result, we expect

$$\hat{\theta}_n < \theta.$$

Therefore, to improve our estimate, we should increase $\hat{\theta}_n$. Similarly, if $\bar{R}_n < E[R]$, we should decrease $\hat{\theta}_n$.

In this example, rainfall is the *control variate* since it enables us to better control our estimate of θ . The principle idea behind many variance reduction techniques (including control variates) is to “use what you know” about the system. In this example, the system is Gagaland’s climate, and what we know is $E[R]$, the average daily rainfall. We will now study control variates more formally, and in particular, we will determine by *how much* we should increase or decrease $\hat{\theta}_n$.

3.1 The Control Variate Method

Suppose again that we wish to estimate $\theta := E[Y]$ where $Y = h(\mathbf{X})$ is the output of a simulation experiment. Suppose that Z is also an output of the simulation or that we can easily output it if we wish. Finally, we assume that we know $E[Z]$. Then we can construct many unbiased estimators of θ :

1. $\hat{\theta} = Y$, our usual estimator
2. $\hat{\theta}_c = Y + c(Z - E[Z])$

where c is some real number. It is clear that $E[\hat{\theta}_c] = \theta$. The question is whether or not $\hat{\theta}_c$ has a lower variance than $\hat{\theta}$. To answer this question, we compute $\text{Var}(\hat{\theta}_c)$ and get

$$\text{Var}(\hat{\theta}_c) = \text{Var}(Y) + c^2 \text{Var}(Z) + 2c \text{Cov}(Y, Z). \quad (2)$$

Since we are free to choose c , we should choose it to minimize $\text{Var}(\hat{\theta}_c)$. Simple calculus then implies that the optimal value of c is given by

$$c^* = -\frac{\text{Cov}(Y, Z)}{\text{Var}(Z)}.$$

Substituting for c^* in (2) we see that

$$\begin{aligned} \text{Var}(\hat{\theta}_{c^*}) &= \text{Var}(Y) - \frac{\text{Cov}(Y, Z)^2}{\text{Var}(Z)} \\ &= \text{Var}(\hat{\theta}) - \frac{\text{Cov}(Y, Z)^2}{\text{Var}(Z)}. \end{aligned}$$

So we see that in order to achieve a variance reduction it is only necessary that $\text{Cov}(Y, Z) \neq 0$. In this case, Z is called a *control variate* for Y . To use the control variate Z in our simulation we would like to modify our algorithm so that after generating n samples of Y and Z we would simply set

$$\hat{\theta}_{c^*} = \frac{\sum_{i=1}^n (Y_i + c^*(Z_i - E[Z]))}{n}.$$

There is a problem with this, however, as we usually do not know $\text{Cov}(Y, Z)$. We overcome this problem by doing p *pilot* simulations and setting

$$\widehat{\text{Cov}}(Y, Z) = \frac{\sum_{j=1}^p (Y_j - \bar{Y}_p)(Z_j - E[Z])}{p-1}.$$

If it is also the case that $\text{Var}(Z)$ is unknown, then we also estimate it with

$$\widehat{\text{Var}}(Z) = \frac{\sum_{j=1}^p (Z_j - E[Z])^2}{p-1}$$

and finally set

$$\hat{c}^* = -\frac{\widehat{\text{Cov}}(Y, Z)}{\widehat{\text{Var}}(Z)}.$$

Assuming we can find a control variate, our control variate simulation algorithm is as follows. Note that the V_i 's are IID, so we can compute approximate confidence intervals as before.

Control Variate Simulation Algorithm for Estimating $E[Y]$

```

/* Do pilot simulation first */
for i = 1 to p
    generate (Yi, Zi)
end for
compute c*
/* Now do main simulation */
for i = 1 to n
    generate (Yi, Zi)
    set Vi = Yi + c*(Zi - E[Z])
end for
set  $\hat{\theta}_{c^*} = \bar{V}_n = \sum_{i=1}^n V_i / n$ 
set  $\hat{\sigma}_{n,v}^2 = \sum (V_i - \hat{\theta}_{c^*})^2 / (n - 1)$ 
set 100(1 -  $\alpha$ ) % CI =  $\left[ \hat{\theta}_{c^*} - z_{1-\alpha/2} \frac{\hat{\sigma}_{n,v}}{\sqrt{n}}, \hat{\theta}_{c^*} + z_{1-\alpha/2} \frac{\hat{\sigma}_{n,v}}{\sqrt{n}} \right]$ 

```

Example 2

Suppose we wish to estimate $\theta = E[e^{(U+W)^2}]$ where $U, W \sim U(0, 1)$ and IID. In our notation we then have $Y := e^{(U+W)^2}$. The usual method is:

1. Generate U_1, \dots, U_n and W_1, \dots, W_n , all IID
2. Compute $Y_1 = e^{(U_1+W_1)^2}, \dots, Y_n = e^{(U_n+W_n)^2}$
3. Estimate θ with

$$\hat{\theta}_{n,y} = \frac{\sum_{j=1}^n Y_j}{n}$$

4. Build confidence intervals for θ :

$$\hat{\theta}_{n,y} \pm z_{1-\alpha/2} \frac{\hat{\sigma}_{n,y}}{\sqrt{n}}$$

where $\hat{\sigma}_{n,y}^2$ is the usual estimate of $\text{Var}(Y)$.

Now consider using the control variate technique. First we have to choose an appropriate control variate, Z .⁴ One possible control variate is $Z_1 := U + W$. We know $E[U + W] = 1$ and it is also apparent that in this case $\text{Cov}(Y, Z_1)$ should be greater than 0. Alternatively, we could use $Z_2 := (U + W)^2$. Again, we would expect $\text{Cov}(Y, Z_2) > 0$ and we know $E[Z_2]$ since

$$\begin{aligned}
 E[Z_2] &= E[U^2 + 2UW + W^2] \\
 &= 1/3 + 1/2 + 1/3 \\
 &= 7/6.
 \end{aligned}$$

So let us use Z_2 as our control variate. (Note that there are usually many possible covariates. For example, we could also use $Z_3 := e^{U+W}$. You should check that $E[Z_3]$ is easily computed.)

⁴Recall that for a control variate Z , we should be able to compute $E[Z]$, and $\text{Cov}(Y, Z)$ should not equal 0.

Matlab Code for Estimating $\theta = E[e^{(U+W)^2}]$

```

> % Do pilot simulation first
> p=100; n=1000;
> u=rand(p,1); w=rand(p,1); y=exp((u+w).^2); z2 = (u+w).^2;
> cov_est=cov([y z2])

cov_est =
    31.3877    4.1800
    4.1800    0.6960

> c_est = - cov_est(1,2)/cov_est(2,2)

c_est = -6.0061

> % Now do main simulation
> u = rand(n,1); w=rand(n,1); y = exp((u+w).^2); z2 = (u+w).^2;
> v = y + c_est*(z2 - 7/6);
> mean(y), std(y) % Check the mean and std of usual estimator

ans =    5.0340    6.2507

> mean (v), std(v) % Check the mean and std of new estimator

ans =    4.9329    3.1218

> % Now compute confidence intervals
> CI = [mean(v) - 1.96*std(v)/sqrt(n), mean(v) + 1.96*std(v)/sqrt(n)]

CI =    4.7394    5.1263

```

Note that the variance reduction here is reasonable: we go from a variance of 6.2507^2 to a variance of 3.1218^2 which implies we have reduced the variance by approximately a factor of 4. Had we used Y_3 above as our control variate, then we probably would have reduced the variance by a larger factor. In general, a good rule of thumb is that we should not be satisfied unless we have a variance reduction on the order of a factor of 5 to 10, though often we will achieve much more.⁵ ■

Example 3 (Pricing an Asian Call Option)

Recall that the payoff of an Asian call option is given by

$$h(\mathbf{X}) := \max \left(0, \frac{\sum_{i=1}^m S_{iT/m}}{m} - K \right)$$

where $\mathbf{X} := \{S_{iT/m} : i = 1, \dots, m\}$. The price of the option is then given by

$$C_a = E[e^{-rT} h(\mathbf{X})]$$

⁵Recall that using the control variate estimator increases the amount of work we must do to generate each sample. As we said earlier though, this increase in workload is often very small and is more than offset by the variance reduction that accompanies it.

where r is the interest rate, K is the strike price and T is the expiration date. We will assume as usual that $S_t \sim GBM(r, \sigma)$.

The usual simulation method for estimating C_a is to generate n independent samples of the payoff, $Y_i := e^{-rT} h(\mathbf{X}_i)$, for $i = 1, \dots, n$, and to set

$$\hat{C}_a = \frac{\sum_{i=1}^n Y_i}{n}.$$

But we could also estimate C_a using control variates and there are many possible choices that we could use:

1. $Z_1 = S_T$
2. $Z_2 = e^{-rt} \max(0, S_T - K)$
3. $Z_3 = \sum_{j=1}^m S_{iT/m}/m$

In each of the three cases, it is easy to compute $E[Z]$. Indeed, $E[Z_2]$ is the well-studied Black-Scholes option price. We would also expect Z_1 , Z_2 and Z_3 to have a positive covariance with Y , so that each one would be a suitable candidate for use as a control variate. Which one would lead to the greatest variance reduction? ■

Example 4 (The Barbershop)

Many application of variance reduction techniques can be found in the study of queuing systems. As a simple exam, consider the case of a barbershop where the barber opens for business every day at 9am and closes at 6pm. He is the only barber in the shop and he is considering hiring another barber to share the workload. First, however, he would like to estimate the mean total time that customers spend waiting on a given day.

Assume customers arrive at the barbershop according to a non-homogeneous Poisson process, $N(t)$, with intensity $\lambda(t)$, and let W_i denote the waiting time of the i^{th} customer. Then, noting that the barber has a 9-hour work day, the quantity that he wants to estimate is $\theta := E[Y]$ where

$$Y := \sum_{j=1}^{N(9)} W_j.$$

Assume also that the service times of customers are IID with CDF, $F(\cdot)$, and that they are also independent of the arrival process, $N(t)$.

The usual simulation method for estimating θ would be to simulate n days of operation in the barbershop, thereby obtaining n samples, Y_1, \dots, Y_n , and then setting

$$\hat{\theta}_n = \frac{\sum_{j=1}^n Y_j}{n}.$$

However, a better estimate could be obtained by using a control variate. In particular, let Z denote the total time customers on a given day spend *in service* so that

$$Z := \sum_{j=1}^{N(9)} S_j$$

where S_j is the service time of the j^{th} customer. Then, since services times are IID and independent of the arrival process, it is easy to see that

$$E[Z] = E[S]E[N(9)]$$

which should be easily computable. Intuition suggests that Z should be positively correlated with Y and therefore it would also be a good candidate to use as a control variate. ■

3.2 Multiple Control Variates

Up until now we have only considered the possibility of using one control variate. However, there is no reason why we should not use more than one and we will now describe how to do this. Suppose again that we wish to estimate $\theta := E[Y]$ where Y is the output of a simulation experiment. We also suppose that for $i = 1, \dots, m$, Z_i is an output or that we can easily output it if we wish. Finally, we assume that $E[Z_i]$ is known for each i .

We can then construct unbiased estimators of θ by setting

$$\hat{\theta}_c = Y + c_1(Z_1 - E[Z_1]) + \dots + c_m(Z_m - E[Z_m])$$

where each $c_i \in \mathbf{R}$. It is clear again that $E[\hat{\theta}_c] = \theta$.

The question is whether we can choose the c_i 's so that $\hat{\theta}_c$ has a lower variance than $\hat{\theta}$, the usual estimator. Of course, we can always set each $c_i = 0$, thereby obtaining $\hat{\theta}$, so that if we choose a good set of c_i 's we should not do worse than $\hat{\theta}$. To choose a good set of c_i 's, we compute $\text{Var}(\hat{\theta}_c)$ and obtain

$$\text{Var}(\hat{\theta}_c) = \text{Var}(Y) + 2 \sum_{i=1}^m c_i \text{Cov}(Y, Z_i) + \sum_{i=1}^m \sum_{j=1}^m c_i c_j \text{Cov}(Z_i, Z_j). \quad (3)$$

As before, we could use calculus to find the optimal set of c_i 's: take partial derivatives with respect to each c_i , set them equal to 0 and solve the resulting system of m linear equations in m unknowns. This is easy to do but as before, however, the optimal solutions c_i^* will involve unknown quantities so that they will need to be estimated using a pilot simulation. A convenient way of estimating the \hat{c}_i^* 's is to observe that $\hat{c}_i^* = -b_i$ where the b_i 's are the (least squares) solution to the following linear regression:

$$Y = a + b_1 Z_1 + \dots + b_m Z_m + \epsilon \quad (4)$$

where ϵ is an error term.⁶ Many software packages have commands that will automatically output the b_i 's. In Matlab for example, the *regress* command will work. The simulation algorithm with multiple control variates is exactly analogous to the simulation algorithm with just a single control variate.

Multiple Control Variate Simulation Algorithm for Estimating $E[Y]$

```

/* Do pilot simulation first */
for i = 1 to p
    generate (Y_i, Z_{1,i}, ..., Z_{m,i})
end for
compute c_i^*, i = 1, ..., m
/* Now do main simulation */
for i = 1 to n
    generate (Y_i, Z_{1,i}, ..., Z_{m,i})
    set V_i = Y_i + \sum_{j=1}^m \hat{c}_j^* (Z_{j,i} - E[Z_j])
end for
set \hat{\theta}_{c^*} = \bar{V}_n = \sum_{i=1}^n V_i / n
set \hat{\sigma}_{n,v}^2 = \sum (V_i - \hat{\theta}_{c^*})^2 / (n - 1)
set 100(1 - \alpha) % CI = [\hat{\theta}_{c^*} - z_{1-\alpha/2} \frac{\hat{\sigma}_{n,v}}{\sqrt{n}}, \hat{\theta}_{c^*} + z_{1-\alpha/2} \frac{\hat{\sigma}_{n,v}}{\sqrt{n}}]

```

Suppose, for example, that we wished to use three control variates, Z_1 , Z_2 , and Z_3 . Using Matlab, we could compute the \hat{c}_i^* 's as follows:

⁶If you like, you can check for yourself that the c_i 's that minimize the variance in (3) multiplied by -1 , equal the b_i 's that solve the regression in (4).

Matlab Code for Computing the \hat{c}_i^* 's

```
> b=regress(Y,[e Z1 Z2 Z3]); % e is a column of 1's
                        % e, Y, Z1, Z2 and Z3 are (n x 1) vectors
> c=-b;                % This is the vector of c values
```

Example 5 (Portfolio Evaluation)

Consider an agent or a fund that has a large number of assets in its portfolio. These assets could include stocks, bonds and derivative securities. Let W_t denote the value of the portfolio at time t . The fund is interested in computing

$$\theta = \mathbf{P} \left(\frac{W_T}{W_0} \leq q \right)$$

where q is a fixed constant. Of course we could just estimate θ using the standard simulation algorithm. Very often, however, the event in question, i.e. $W_T/W_0 \leq q$, is a *rare event* and the usual simulation algorithm will be very inefficient.

One possible remedy is to identify the most important assets in the portfolio and use their values as control variates. This is often possible since in many financial models, the values of the primitive assets (stocks, bonds) as well as some option prices, are known and will be correlated with the value of the portfolio. ■

Remark 2 *When estimating a rare event probability in practice, it is usually preferable to do so using importance sampling, possibly combined with other variance reduction techniques such as stratified sampling or control variates.*

4 Antithetic Variates

Suppose as usual that we would like to estimate $\theta = E[h(\mathbf{X})] = E[Y]$, and that we have generated two samples, Y_1 and Y_2 . Then an unbiased estimate of θ is given by

$$\hat{\theta} = \frac{Y_1 + Y_2}{2}$$

and

$$\text{Var}(\hat{\theta}) = \frac{\text{Var}(Y_1) + \text{Var}(Y_2) + 2\text{Cov}(Y_1, Y_2)}{4}. \quad (5)$$

If Y_1 and Y_2 are IID, then $\text{Var}(\hat{\theta}) = \text{Var}(Y)/2$. However, we could reduce $\text{Var}(\hat{\theta})$ if we could arrange it so that $\text{Cov}(Y_1, Y_2) < 0$. We now describe the method of antithetic variates for doing this. We will begin with the case where Y is a function of IID $U(0, 1)$ random variables so that

$$\theta = E[h(\mathbf{U})]$$

where $\mathbf{U} = (U_1, \dots, U_m)$ and the U_i 's are IID $\sim U(0, 1)$. The usual Monte Carlo algorithm, assuming we use $2n$ samples, is as follows.

Usual Simulation Algorithm for Estimating θ

```

for  $i = 1$  to  $2n$ 
    generate  $\mathbf{U}_i$ 
    set  $Y_i = h(\mathbf{U}_i)$ 
end for
set  $\hat{\theta}_{2n} = \bar{Y}_{2n} = \sum_{i=1}^{2n} Y_i / 2n$ 
set  $\hat{\sigma}_{2n}^2 = \sum_{i=1}^{2n} (Y_i - \bar{Y}_{2n})^2 / (2n - 1)$ 
set approx.  $100(1 - \alpha) \%$  CI  $= \hat{\theta}_{2n} \pm z_{1-\alpha/2} \frac{\hat{\sigma}_{2n}}{\sqrt{2n}}$ 

```

In the above algorithm, however, we could also have used the $1 - U_i$'s to generate sample Y values, since if $U_i \sim U(0, 1)$, then so too is $1 - U_i$. We can use this fact to construct another estimator of θ as follows.

As before, we set $Y_i = h(\mathbf{U}_i)$, where $\mathbf{U}_i = (U_1^{(i)}, \dots, U_m^{(i)})$. We now also set $\tilde{Y}_i = h(1 - \mathbf{U}_i)$, where we use $1 - \mathbf{U}_i$ to denote $(1 - U_1^{(i)}, \dots, 1 - U_m^{(i)})$. Note that $E[Y_i] = E[\tilde{Y}_i] = \theta$ so that in particular, if

$$Z_i := \frac{Y_i + \tilde{Y}_i}{2},$$

then $E[Z_i] = \theta$. This means that Z_i is also an unbiased estimator of θ . If the U_i 's are IID, then so too are the Z_i 's and we can use them as usual to compute approximate confidence intervals for θ .

We say that U_i and $1 - U_i$ are *antithetic variates* and we then have the following *antithetic variate* simulation algorithm.

Antithetic Variate Simulation Algorithm for Estimating θ

```

for  $i = 1$  to  $n$ 
    generate  $\mathbf{U}_i$ 
    set  $Y_i = h(\mathbf{U}_i)$  and  $\tilde{Y}_i = h(1 - \mathbf{U}_i)$ 
    set  $Z_i = (Y_i + \tilde{Y}_i) / 2$ 
end for
set  $\hat{\theta}_{n,a} = \bar{Z}_n = \sum_{i=1}^n Z_i / n$ 
set  $\hat{\sigma}_{n,a}^2 = \sum_{i=1}^n (Z_i - \bar{Z}_n)^2 / (n - 1)$ 
set approx.  $100(1 - \alpha) \%$  CI  $= \hat{\theta}_{n,a} \pm z_{1-\alpha/2} \frac{\hat{\sigma}_{n,a}}{\sqrt{n}}$ 

```

As usual, $\hat{\theta}_{n,a}$ is an unbiased estimator of θ and the Strong Law of Large Numbers implies $\hat{\theta}_{n,a} \rightarrow \theta$ almost surely as $n \rightarrow \infty$.

Each of the two algorithms we have described above uses $2n$ samples so the question naturally arises as to which algorithm is better. Note that both algorithms require approximately the same amount of effort⁷ so that comparing the two algorithms amounts to computing which estimator has a smaller variance.

⁷In fact, we could argue that the antithetic algorithm requires marginally less effort since it only needs to generate n independent uniform random vectors instead of $2n$.

4.1 Comparing Estimator Variances

It is easy to see that

$$\text{Var}(\hat{\theta}_{2n}) = \text{Var}\left(\frac{\sum_{i=1}^{2n} Y_i}{2n}\right) = \frac{\text{Var}(Y)}{2n}$$

and

$$\begin{aligned} \text{Var}(\hat{\theta}_{n,a}) &= \text{Var}\left(\frac{\sum_{i=1}^n Z_i}{n}\right) = \frac{\text{Var}(Z)}{n} \\ &= \frac{\text{Var}(Y + \tilde{Y})}{4n} = \frac{\text{Var}(Y)}{2n} + \frac{\text{Cov}(Y, \tilde{Y})}{2n} \\ &= \text{Var}(\hat{\theta}_{2n}) + \frac{\text{Cov}(Y, \tilde{Y})}{2n}. \end{aligned}$$

Therefore $\text{Var}(\hat{\theta}_{n,a}) < \text{Var}(\hat{\theta}_{2n})$ if and only if $\text{Cov}(Y, \tilde{Y}) < 0$. Recalling that $Y = h(\mathbf{U})$ and $\tilde{Y} = h(1 - \mathbf{U})$, this means that

$$\text{Var}(\hat{\theta}_{n,a}) < \text{Var}(\hat{\theta}_{2n}) \iff \text{Cov}(h(\mathbf{U}), h(1 - \mathbf{U})) < 0.$$

We will soon discuss conditions that are sufficient to *guarantee* that $\text{Cov}(h(\mathbf{U}), h(1 - \mathbf{U})) < 0$, but first let's consider an example.

Example 6 (Monte Carlo Integration)

Consider the problem of estimating

$$\theta = \int_0^1 e^{x^2} dx.$$

As usual, we may then write $\theta = \mathbb{E}[e^{U^2}]$ where $U \sim U(0, 1)$. We can compare the usual raw simulation algorithm with the simulation algorithm that uses antithetic variates.

Matlab Code for Estimating $\theta = \int_0^1 e^{x^2} dx$

```
> n=1000;
> U=rand(2*n,1);    % Original simulation method first
> Y=exp(U.^2);
> [mean(Y) std(Y)^2]

ans =      1.4696      0.2315

> u=rand(n,1);      % Now use n antithetic pairs
> v=1-u;
> Z=(exp(u.^2) + exp(v.^2))/2;
> [mean(Z) std(Z)^2]

ans =      1.4668      0.0293

> % Now compute percentage variance reduction
> Variance_reduction = (0.2315/(2*n) - 0.0293/n)*100/(0.2315/(2*n))

Variance_reduction =      74.6868
```

In the above example we achieved a variance reduction of 75%. Had we been trying to estimate $\theta = E[e^u]$, then we would have achieved an even greater variance reduction.⁸ In general, however, it is possible that only a small variance reduction, or even a variance *increase* is obtained. We will now examine the conditions under which a variance reduction can be guaranteed.

Consider first the case where \mathbf{U} is a uniform random variable so that $m = 1$, $\mathbf{U} = U$ and $\theta = E[h(U)]$. Suppose now that $h(\cdot)$ is a *non-decreasing* function of u over $[0, 1]$. Then if U is large, $h(U)$ will also tend to be large while $1 - U$ and $h(1 - U)$ will tend to be small. That is, $\text{Cov}(h(U), h(1 - U)) < 0$. We can similarly conclude that if $h(\cdot)$ is a *non-increasing* function of u then once again, $\text{Cov}(h(U), h(1 - U)) < 0$. So for the case where $m = 1$, a sufficient condition to guarantee a variance reduction is for $h(\cdot)$ to be a *monotonic*⁹ function of u on $[0, 1]$.

Let us now consider the more general case where $m > 1$, $\mathbf{U} = (U_1, \dots, U_m)$ and $\theta = E[h(\mathbf{U})]$. We say $h(u_1, \dots, u_m)$ is a monotonic function of each of its m arguments if, in each of its arguments, it is non-increasing or non-decreasing. We have the following theorem which generalizes the $m = 1$ case above.

Theorem 1 *If $h(u_1, \dots, u_m)$ is a monotonic function of each of its arguments on $[0, 1]^m$, then for a set $\mathbf{U} := (U_1, \dots, U_m)$ of IID $U(0, 1)$ random variables*

$$\text{Cov}(h(\mathbf{U}), h(1 - \mathbf{U})) < 0$$

where $\text{Cov}(h(\mathbf{U}), h(1 - \mathbf{U})) := \text{Cov}(h(U_1, \dots, U_m), h(1 - U_1, \dots, 1 - U_m))$.

Proof See appendix of Chapter 8 in Ross (2002)..

Note that the theorem specifies *sufficient* conditions for a variance reduction, but not *necessary* conditions. This means that it is still possible to obtain a variance reduction even if the conditions of the theorem are not satisfied. For example, if $h(\cdot)$ is “mostly ” monotonic, then a variance reduction, and possibly a substantial one, might be still be obtained.

4.2 Non-Uniform Antithetic Variates

So far we have only considered problems where $\theta = E[h(\mathbf{U})]$, for \mathbf{U} a vector of IID $U(0, 1)$ random variables. Of course in practice, it is often the case that $\theta = E[Y]$ where $Y = h(X_1, \dots, X_m)$, and where (X_1, \dots, X_m) is a vector of independent¹⁰ random variables. We can still use the antithetic variable method for such problems if we can use the inverse transform method to generate the X_i 's.

To see this, suppose $F_i(\cdot)$ is the CDF of X_i . If $U_i \sim U(0, 1)$ then $F_i^{-1}(U_i)$ has the same distribution as X_i . This implies that we can generate a sample of Y by generating $U_1, \dots, U_m \sim \text{IID } U(0, 1)$ and setting

$$Z = h(F_1^{-1}(U_1), \dots, F_m^{-1}(U_m)).$$

Since the CDF of any random variable is non-decreasing, it follows that the inverse CDFs, $F_i^{-1}(\cdot)$, are also non-decreasing. This means that if $h(x_1, \dots, x_m)$ is a monotonic function of each of its arguments, then $h(F_1^{-1}(U_1), \dots, F_m^{-1}(U_m))$ is also a monotonic function of the U_i 's. Theorem 1 therefore applies.

Antithetic variables are often very useful when studying queueing systems and we briefly describe why this is the case.

⁸In that case, the true variance reduction is easy to compute explicitly.

⁹A monotonic function, $f(x)$, is one that is either non-increasing or non-decreasing.

¹⁰Note that we do not assume that the X_i 's are identically distributed since in general this will not be the case.

Example 7 (The Barbershop Revisited)

Consider again the barbershop example and suppose that the barber now wants to estimate the average total waiting time, θ , of the first 100 customers. Then $\theta = E[Y]$ where

$$Y = \sum_{j=1}^{100} W_j$$

and where W_j is the waiting time of the j^{th} customer. Now for each customer, j , there is an inter-arrival time, I_j , which is the time between the $(j-1)^{th}$ and j^{th} arrivals. There is also a service time, S_j , which is the amount of time the barber spends cutting the j^{th} customer's hair. It is therefore the case that Y may be written as

$$Y = h(I_1, \dots, I_{100}, S_1, \dots, S_{100})$$

for some function, $h(\cdot)$. Now for many queueing systems, $h(\cdot)$ will be a monotonic function of its arguments since we would typically expect Y to increase as service times increase, and decrease as inter-arrival times increase. As a result, it might be advantageous¹¹ to use antithetic variates to estimate θ . ■

4.3 Normal Antithetic Variates

We can also generate antithetic normal random variates without using the inverse transform technique. For if $X \sim N(\mu, \sigma^2)$ then $\tilde{X} \sim N(\mu, \sigma^2)$ also, where $\tilde{X} := 2\mu - X$. Clearly X and \tilde{X} are negatively correlated. So if $\theta = E[h(X_1, \dots, X_m)]$ where the X_i 's are IID $N(\mu, \sigma^2)$ and $h(\cdot)$ is monotonic in its arguments, then we can again achieve a variance reduction by using antithetic variates.

Example 8

Suppose we want to estimate $\theta = E[X^2]$ where $X \sim N(2, 1)$. Then it is easy to see that $\theta = 5$, but we can also estimate it using antithetic variates.

Question 1: Is a variance reduction guaranteed?

Matlab Code for Estimating $\theta = E[X^2]$

```
> U=2 + randn(2*n,1);    % First use the raw simulation method
> h=U.^2;
> [mean(h) std(h)^2]

ans =    4.9564    16.8349

> U=2 + randn(n,1);    % Now use antithetic variates
> V=4-U;
> h=(U.^2 + V.^2)/2;
> [mean(h) std(h)^2]

ans =    5.0092    1.7966

> % Now compute the percentage variance reduction
> var_reduction = (16.8349/(2*n) - 1.7966/n)*100/(16.8349/(2*n))

var_reduction =    78.6562
```

¹¹We are assuming that the I_j 's and S_j 's can be generated using the inverse transform method.

We therefore obtain a substantial variance reduction.

Question 2: Why might this have occurred?

Question 3: What would you expect if $Z \sim N(10, 1)$?

Question 4: What would you expect if $Z \sim N(0.5, 1)$? ■

Example 9 (Knock-In Option)

Suppose we wish to price a knock-in option where the payoff is given by

$$h(S_T) = \max(0, S_T - K)I_{\{S_T > B\}}$$

where B is some fixed constant. We assume that r is the risk-free interest rate, $S_t \sim GBM(r, \sigma^2)$ and that S_0 is the initial stock price. Then the option price may be written as

$$C_0 = E[e^{-rT} \max(0, S_T - K)I_{\{S_T > B\}}]$$

so we can estimate it using simulation. Since we can write $S_T = S_0 e^{(r - \sigma^2/2)T + \sigma\sqrt{T}X}$ where $X \sim N(0, 1)$ we can use antithetic variates to estimate C_0 . Are we sure to get a variance reduction? ■

Example 10

We can also use antithetic variates to price Asian options. In this case, however, we need to be a little careful generating the stock prices if we are to achieve a variance reduction. You might want to think about this yourself. ■

5 Conditional Monte Carlo

We will now consider a new variance reduction technique, namely *conditional* Monte Carlo. The idea here is very simple. As was the case with control variates, we use our knowledge about the system being studied to reduce the variance of our estimator.

As usual, suppose we wish to estimate $\theta = E[h(\mathbf{X})]$ where $\mathbf{X} = (X_1, \dots, X_m)$. If we could compute θ analytically, then this would be equivalent to solving an m -dimensional integral. However, maybe it is possible to evaluate *part* of the integral analytically. If so, then we might be able to use simulation to estimate the other part and thereby obtain a variance reduction. The vehicle that we use to do part of the integration analytically is the concept of *conditional expectation*, which of course we have seen before. Before we describe the method in detail, we will briefly review conditional expectations and variances.

5.1 Review of Conditional Expectations and Variances

Let \mathbf{X} and \mathbf{Z} be random vectors, and let $Y = h(\mathbf{X})$ be a random variable. Suppose we set

$$V = E[Y|\mathbf{Z}].$$

Then V is itself a random variable¹² that *depends* on \mathbf{Z} , so that we may write $V = g(\mathbf{Z})$ for some function, $g(\cdot)$. We also know that

$$E[V] = E[E[Y|\mathbf{Z}]] = E[Y]$$

¹²It is very important that you view V as a random variable. In particular, it is a function of \mathbf{Z} which is random.

so that if we are trying to estimate $\theta = E[Y]$, one possibility¹³ would be to simulate V 's instead of simulating Y 's. In order to decide which would be a better estimator of θ , it is necessary to compare the variances of Y and $E[Y|\mathbf{Z}]$. To do this, recall the *conditional variance* formula:

$$\text{Var}(Y) = E[\text{Var}(Y|\mathbf{Z})] + \text{Var}(E[Y|\mathbf{Z}]). \quad (6)$$

Now $\text{Var}(Y|\mathbf{Z})$ is also a random variable that depends on \mathbf{Z} , and since a variance is always non-negative, it must follow that $E[\text{Var}(Y|\mathbf{Z})] \geq 0$. But then (6) implies

$$\text{Var}(Y) \geq \text{Var}(E[Y|\mathbf{Z}])$$

so we can conclude that V is a better¹⁴ estimator of θ than Y .

To see this from another perspective, suppose that to estimate θ we first have to simulate \mathbf{Z} and then simulate Y given \mathbf{Z} . If we can compute $E[Y|\mathbf{Z}]$ exactly, then we can eliminate the additional noise that comes from simulating Y given \mathbf{Z} , thereby obtaining a variance reduction. Finally, we point out that in order for the conditional expectation method to be worthwhile, it must be the case that Y and \mathbf{Z} are *dependent*.

Exercise 1 Why must Y and \mathbf{Z} be dependent for the conditional Monte Carlo method to be worthwhile?

5.2 The Conditional Monte Carlo Simulation Algorithm

We want to estimate $\theta := E[h(\mathbf{X})] = E[Y]$ using conditional Monte Carlo. To do so, we must have another variable or vector, \mathbf{Z} , that satisfies the following requirements:

1. \mathbf{Z} can be easily simulated
2. $V := g(\mathbf{Z}) := E[Y|\mathbf{Z}]$ can be computed exactly.

This means that we can simulate a value of V by first simulating a value of \mathbf{Z} and then setting $V = g(\mathbf{Z}) = E[Y|\mathbf{Z}]$.¹⁵ We then have the following algorithm for estimating θ .

Conditional Monte Carlo Algorithm for Estimating θ

```

for  $i = 1$  to  $n$ 
    generate  $\mathbf{Z}_i$ 
    compute  $g(\mathbf{Z}_i) = E[Y|\mathbf{Z}_i]$ 
    set  $V_i = g(\mathbf{Z}_i)$ 
end for
set  $\hat{\theta}_{n,cm} = \bar{V}_n = \sum_{i=1}^n V_i / n$ 
set  $\hat{\sigma}_{n,cm}^2 = \sum_{i=1}^n (V_i - \bar{V}_n)^2 / (n - 1)$ 
set approx.  $100(1 - \alpha) \%$  CI  $= \hat{\theta}_{n,cm} \pm z_{1-\alpha/2} \frac{\hat{\sigma}_{n,cm}}{\sqrt{n}}$ 

```

It is also possible that other variance reduction methods could be used in conjunction with conditioning. For example, it might be possible to use control variates, or if $g(\cdot)$ is a monotonic function of its arguments, then antithetic variates might be useful.

¹³Of course this assumes that we can actually simulate the V 's.

¹⁴This assumes that the work required to generate V is not too much greater than the work required to generate Y .

¹⁵It may also be possible to identify the distribution of $V = g(\mathbf{Z})$, so that we could then simulate V directly.

Example 11

Suppose we wish to estimate $\theta := \mathbf{P}(U + Z > 4)$ where $U \sim \text{Exp}(1)$ and $Z \sim \text{Exp}(1/2)$. If we set $Y := I_{\{U+Z>4\}}$ then we see that $\theta = \mathbf{E}[Y]$. Then the usual simulation method is:

1. Generate $U_1, \dots, U_n, Z_1, \dots, Z_n$ all independent
2. Set $Y_i = I_{\{U_i+Z_i>4\}}$ for $i = 1, \dots, n$
3. Set $\hat{\theta}_n = \sum_{i=1}^n Y_i/n$
4. Compute approximate CI's as usual

However, we can also use the conditioning method, which works as follows. Set $V = \mathbf{E}[Y|Z]$. Then

$$\mathbf{E}[Y|Z = z] = \mathbf{P}(U + Z > 4|Z = z) = \mathbf{P}(U > 4 - z) = 1 - F_u(4 - z)$$

where $F_u(\cdot)$ is the CDF of U which we know is exponentially distributed with mean 1. Therefore

$$1 - F_u(4 - z) = \begin{cases} e^{-(4-z)} & \text{if } 0 \leq z \leq 4, \\ 1 & \text{if } z > 4. \end{cases}$$

which implies

$$V = \mathbf{E}[Y|Z] = \begin{cases} e^{-(4-Z)} & \text{if } 0 \leq Z \leq 4, \\ 1 & \text{if } Z > 4. \end{cases}$$

Now the conditional Monte Carlo algorithm for estimating $\theta = \mathbf{E}[V]$ is:

1. Generate Z_1, \dots, Z_n all independent
2. Set $V_i = \mathbf{E}[Y|Z_i]$ for $i = 1, \dots, n$
3. Set $\hat{\theta}_{n,cm} = \sum_{i=1}^n V_i/n$
4. Compute approximate CI's as usual using the V_i 's

Note that we might also want to use other variance reduction methods in conjunction with conditioning when we are finding $\hat{\theta}_{n,cm}$. ■

5.3 The Black-Scholes Formula

Before we proceed with a final example, we first describe the Black-Scholes formula for the price of a European call option. Suppose we have a stock with price S_t at time t , and where $S_t \sim GBM(\mu, \sigma^2)$. If r is the risk-free interest rate, K the strike, T the expiration date and S_0 the initial stock price, then the price of the option, C_0 , may be expressed as

$$C_0 = \mathbf{E}[e^{-rT} \max(0, S_T - K)] \quad (7)$$

where we assume in (7) that $S_t \sim GBM(r, \sigma^2)$. The price of the option can be computed analytically and is given by

$$C_0 = S_0 \Phi(d_1) - K e^{-rT} \Phi(d_2) \quad (8)$$

where $\Phi(\cdot)$ is the CDF of a standard normal random variable and

$$\begin{aligned} d_1 &= \frac{\log(S_0/K) + (r + \sigma^2/2)T}{\sigma\sqrt{T}} \\ d_2 &= \frac{\log(S_0/K) + (r - \sigma^2/2)T}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}. \end{aligned}$$

Definition 1 Let $c(x, t, K, r, \sigma)$ be the Black-Scholes price of a European call option when the current stock price is x , the time to maturity is t , the strike is K , the risk-free interest rate is r and the volatility is σ .

Example 12 (An Exotic Option)

Suppose we want to find the price of a European option that has payoff at expiration given by

$$h(\mathbf{X}) = \begin{cases} \max(0, S_T - K_1) & \text{if } S_{T/2} \leq L, \\ \max(0, S_T - K_2) & \text{otherwise.} \end{cases}$$

where $\mathbf{X} = (S_{T/2}, S_T)$. We can write the price of the option as

$$C_0 = \mathbb{E} \left[e^{-rT} \left(\max(0, S_T - K_1) I_{\{S_{T/2} \leq L\}} + \max(0, S_T - K_2) I_{\{S_{T/2} > L\}} \right) \right]$$

where as usual, we assume that $S_t \sim GBM(r, \sigma^2)$.

Question 1: How would you estimate the price of this option using simulation and only *one* normal random variable per sample payoff?

Question 2: Could you use antithetic variates as well? Would they be guaranteed to produce a variance reduction? ■