Chapter 2. Monte Carlo Simulations (v.A.1)

The following two results from Statistics will prove useful in justifying the uses of Monte Carlo Simulations and understanding their convergence properties.

2.1 Law of Large Numbers (LLN)

Suppose $\{X_i\}^n_{i=1} \sim i.i.d.$ sample and $E(X_i) = \mu,\ Var(X_i) = \sigma^2 < \infty$.

Define $S_n = \frac{X_1 + X_2 + \ldots + X_n}{n}$. Then, $\lim_{n\to\infty} E(S_n - \mu)^2 = 0$, that is, $S_n \approx \mu$ for large enough n.

2.2 Central Limit Theorem (CLT)

Suppose $\{X_i\}^n_{i=1} \sim i.i.d.$ sample and $E(X_i) = \mu,\ Var(X_i) = \sigma^2 < \infty$. Then,

$Z_n = \frac{S_n - \mu}{\sigma\sqrt{n}} \underset{n\to\infty}{\longrightarrow} Z \sim N(0,1)$. The convergence is in distribution.

Below we describe another method for simulating continuously distributed random variates.

2.3 Acceptance-Rejection Method

Suppose we want to generate random variates that come from a distribution with density function $f(x)$ and that it is difficult or impossible to invert the corresponding CDF. That is, the inverse transformation method is not applicable. Assume there exists a function $g(x)$ such that

$g(x) \geq f(x)$ for any x for which $f(x) \neq 0$. Assume $\int_{-\infty}^{\infty} g(x) < \infty$.

Define $h(x) = \frac{g(x)}{\int_{-\infty}^{\infty} g(x)}$. Then, $h(\cdot)$ is a probability density function.

The idea behind the Acceptance-Rejection method is to select a function g in such a way that it is relatively easy to simulate a random variate that has the density of h. Then, the following steps will result in a random variate X that has the desired distribution, with density function f:

STEP 1. Generate Y from the $h(\cdot)$ distribution,

STEP 2. Generate $U \sim U[0,1]$ that is independent of Y,

STEP 3. If $U \leq \frac{f(Y)}{g(Y)}$ then **ACCEPT** Y and set $X = Y$ and exit,

else, if $U > \frac{f(Y)}{g(Y)}$ **REJECT** Y and go to the next STEP,

STEP 4. Repeat Steps 1-3.

Comment: If the support of the function f is a finite interval, say $[a, b]$, then we can take

$$g(x) = \begin{cases} 0, & if \quad x < a, \ or \ x > b \\ \max\limits_{x \in [a,b]} f(x), & if \quad a \leq x \leq b \end{cases}$$ and thus,

$$h(x) = \begin{cases} 0, & if \ \ x < a, \ or \ x > b \\ \frac{1}{(b-a)}, & if \quad a < x \leq b \end{cases}.$$

Proof of the Acceptance-Rejection Algorithm for continuous random variables.

It is necessary to show that the conditional distribution of Y given that $U \leq \frac{f(Y)}{g(Y)}$ is F.

Using the Bayes' Theorem, we can show:

$$P\left(Y \leq y \left| U \leq \frac{f(Y)}{g(Y)}\right.\right) = \frac{P\left(U \leq \frac{f(Y)}{g(Y)} \left| Y \leq y\right.\right) P(Y \leq y)}{P\left(U \leq \frac{f(Y)}{g(Y)}\right)}$$

Notice that

$$P\left(U \leq \frac{f(Y)}{g(Y)}\right) = \int_{-\infty}^{+\infty} P\left(U \leq \frac{f(Y)}{g(Y)} \left| Y = y\right.\right) h(y)dy = \int_{-\infty}^{+\infty} \frac{f(y)}{g(y)} h(y)dy = \frac{1}{\int_{-\infty}^{+\infty} g(x)dx}.$$

For notational convenience define $c = \int_{-\infty}^{+\infty} g(x)dx$. Observe further that

$$P\left(U \leq \frac{f(Y)}{g(Y)} \left| Y \leq y\right.\right) = \frac{P\left(U \leq \frac{f(Y)}{g(Y)}, Y \leq y\right)}{P(Y \leq y)}$$

$$= \frac{1}{H(y)} \int_{-\infty}^{y} P\left(U \leq \frac{f(Y)}{g(Y)} \left| Y = w\right.\right) h(w)dw = \frac{1}{H(y)} \int_{-\infty}^{y} \frac{f(w)}{g(w)} h(w)dw = \frac{F(y)}{cH(y)}.$$

Therefore it follows that

$$P\left(Y \le y \,\middle|\, U \le \frac{f(Y)}{g(Y)}\right) = \frac{\frac{F(y)}{cH(y)}H(y)}{\frac{1}{c}} = F(y).$$

This completes the proof.

Remarks:

In the Acceptance-Rejection algorithm we accepts only a fraction of all generated variates.

Therefore one needs to evaluate the efficiency of the algorithm. The answers to the following questions will shed light on the efficiency of the algorithm:

(a) What is the probability of acceptance of this algorithm?

(b) Define X to be the time of the first "success", defined as accepting the generated number. What is E(X)?

(c) How to choose g(x) to minimize the computational cost of the algorithm?

Suggested Exercises:

1. Use the Acceptance-Rejection method to generate N(0, 1)-distributed random variates. Use Double-Exponential distribution for g: $g(x) = \frac{1}{2}e^{-|x|}$. Notice that, $\frac{f(x)}{g(x)} < c = 1.32$.

2. The answers to the following questions will shed further light on the efficiency of the Acceptance-Rejection algorithm. Define N to be the time of the first "success" – accepting the generated number.

   (a) What is the probability distribution of N?

   (b) Compute EN and show that E$N = c$.

   (c) How to choose the function $g(x)$ to minimize the computational cost of the algorithm? Ideally, we would like to choose g 'close' to f to minimize the computational load. In such cases, c will be close to 1 and we will need to perform fewer iterations. However, there is a tradeoff: g being 'close' to f helps to have

fewer iterations, but if it is difficult to generate variates from the pdf f, then it must be difficult to generate from variates from the pdf g as well.

Examples:

1. Generate $Beta(a,b)$-distributed random variates using the Acceptance-Rejection (AR) algorithm.

   The density function of $Beta(a,b)$ is given by:

$$f(x)=\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}x^{a-1}(1-x)^{b-1}\quad for\ 0\le x\le 1, where\ \Gamma(a)=\int_0^\infty x^{a-1}e^{-x}dx$$

We will use a specific example for demonstration: $Beta(4,3)$.

Then, the density function is $f(x)=60x^3(1-x)^2\ for\ 0\le x\le 1$.

It is easy to see that $f(x)\le 4$ for $0\le x\le 1$ (verify this). We will take

$$g(x)=\begin{cases}4, & for \quad 0\le x\le 1\\ 0, & else\end{cases}$$

and follow the AR algorithm.

Comments:

1. Notice that, the function $g(x)=\begin{cases}0, & else\\ 3, & for \quad 0\le x\le 1\end{cases}$ would also work in this case. So, the obvious question is: is there an "optimal" choice of g?

The answer is: the optimal g (among constants) is the following function:

$$g(x)=\begin{cases}max\dfrac{f(x)}{g(x)}, & for \quad 0\le x\le 1\\ 0, & else\end{cases}$$

The reasoning of this is that, the average number of trials in the algorithm until acceptance is the number that we choose in g (3 or 4 above), therefore $maxf(x)$ would be the best choice to minimize that number.

2.4 Multidimensional Normal Generation

Definition: A random vector $X = (X_1, X_2, \dots, X_n)$ is said to have an n-dimensional Normal Distribution with a mean vector $\mu = (\mu_1, \mu_2, \dots, \mu_n)$, and a variance-covariance matrix $\Sigma = (\sigma_{i,j})_{i,j=1,\dots,n}$ if the probability density function of X is given by

$$\phi(x) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(x-\mu)'\Sigma^{-1}(x-\mu)\right)$$

To generate a realization of an n-variate Normal Vector vector $X = (X_1, X_2, \dots, X_n)$, we will follow these steps:

1. Use Cholesky decomposition of matrixthe variance-covariance matrix Σ to write $\Sigma = \mathrm{L}\,\mathrm{L}'$, where L is an $n \times n$ lower-diagonal matrix, and L' is its transpose matrix.
2. Generate a vector $Z = (Z_1, Z_2, \dots, Z_n)$ of independent standard normally distributed variates,
3. Define $X = \mu + \mathrm{L}\,\mathrm{Z}$. Then, for this vector, $X \sim N(\mu, \Sigma)$.

To prove the statement of Step 3, note that:

(a) $X = \mu + \mathrm{L}\,\mathrm{Z}$ is a linear combination of normals, so it is a normal variate,

(b) $EX = \mu + \mathrm{L}\,\mathbb{E}\mathrm{Z} = \mu$,

(c) $Var(X) = Var(\mu + \mathrm{L}\,\mathrm{Z}) = Var(\mathrm{L}\,\mathrm{Z}) = \mathrm{L}\,\mathrm{Var}(\mathrm{ZZ}')\,\mathrm{L}' = \mathrm{L}\,\mathrm{L}' = \Sigma$.

Since normal variates are determined by their first two moments, then, the statements of normality are proved.

Comment:

When $n = 2$, then a Bivariate Normal $X = (X_1, X_2)$ is generated as follows:

Assume $\Sigma = \begin{pmatrix} {\sigma_1}^2 & \sigma_1\sigma_2\rho \\ \sigma_1\sigma_2\rho & {\sigma_2}^2 \end{pmatrix}$, then we can take L $= \begin{pmatrix} \sigma_1 & 0 \\ \sigma_2\rho & \sqrt{1-\rho^2}\sigma_2 \end{pmatrix}$, and the generation of the two normal (correlated) variates is given by the following algorithm:

Step 1: Generate a pair $Z = (Z_1, Z_2)$ of independent standard normally distributed variates

Step 2: define $= (X_1, X_2)$ as fllows:

$$\begin{cases} X_1 = \mu_1 + \sigma_1 Z_1 \\ X_2 = \mu_2 + \sigma_2\rho Z_1 + \sigma_2\sqrt{1-\rho^2}Z_2 \end{cases}$$

It is easy to see that the means and variance-covariances of the pair are as desired.

Suggested Exercise: Derive the matrix L for the *n=3* case.

2.5 Monte Carlo Simulation

The most common use of Monte Carlo Simulation in finance is when one needs to calculate the expected value of a functional of a random variable: $\mathbb{E}f(X)$. Assume $g(\cdot)$ is the density function of the random variable X. Then, we can express the expectation as an integral:

$$\mathbb{E}f(X) = \int_{-\infty}^{\infty} f(t)g(t)dt$$

If this integral can't be computed explicitly, then Monte Carlo Simulation techniques are adopted to estimate it. The estimation idea is to use the Law of Large Numbers (LLN) to estimate the integral.

Suppose $\{X_i\}^n{}_{i=1}$ is a sample of $i.i.d.$ random variables with the same distribution as X and $\mathbb{E}(X) = \mu,\ Var(X) = \sigma^2 < \infty$.

For the Sample Mean of the random sample $\{f(X_1), f(X_2), \dots, f(X_n)\}$ defined as

$$\overline{fX}_n = \frac{f(X_1)+f(X_2)+\dots+f(X_n)}{n}$$

we have:

$$\lim_{n\to\infty} \mathbb{E}\left(\frac{f(X_1)+f(X_2)+\ \ldots+\ f(X_n)}{n} - \mathbb{E}f(X)\right)^2 = \lim_{n\to\infty} \mathbb{E}\big(\ \overline{fX}_n - \mathbb{E}f(X)\big)^2 = 0.$$

That is, $\overline{fX}_n \approx\ \mathbb{E}f(X)$ for large enough n.

Thus, the simple Monte Carlo method can be used to estimate expected values of complicated functionals of random variables. The method becomes even more appealing as the dimensionality of the problem increases.

Naturally, the larger the size of the sample (n), the closer the approximation would be to the actual (unknown) parameter value. A natural question arises then: what is the relative error of the Monte Carlo estimation method? Or, what is the convergence rate of the method?

In this section, we will study the error of the Monte Carlo estimation method, and a few ideas/techniques that would help to reduce those errors and thus make the numerical algorithms converge faster.

Define the Sample Mean $\bar{X}_n$ and the Sample Variance S_n^2 of the sample $\{X_i\}^n{}_{i=1} \sim i.i.d.$ with the same distribution as the random variable X, where $\mathbb{E}(X) = \mu,\ Var(X) = \sigma^2 < \infty,$ as follows:

$$\bar{X}_n = \frac{X_1+X_2+\ \ldots+\ X_n}{n}, \qquad S_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}\ (X_i - \bar{X}_n)^2.$$

Assume we are trying to estimate the unknown mean μ.

Notice that, the Sample Mean is an unbiased estimator for μ: $\mathbb{E}(\ \bar{X}_n) = \mu$

Then, we have that $Var(\ \bar{X}_n\ ) = \mathbb{E}(\ \bar{X}_n - \mu)^2 = \frac{\sigma^2}{n}$.

It can be seen from the latter that the square-distance of the Sample Mean from the actual (unknown) expected value converges to 0 as n gets larger.

Assume that Z is a standard-normally distributed random variable. z_α is defined as the number so that $\boldsymbol{P}(Z > z_\alpha) = \alpha$.

We have that $\boldsymbol{P}\left(-z_{\frac{\alpha}{2}} < Z < z_{\frac{\alpha}{2}}\right) = 1 - \alpha$ and we know that $\frac{\bar{X}_n - \mu}{\frac{\sigma}{\sqrt{n}}} \to Z \sim N(0,1)$ as $n \to \infty$, due to the Central Limit Theorem.

Therefore, we can write that $\boldsymbol{P}\left(-z_{\frac{\alpha}{2}} < \frac{\bar{X}_n - \mu}{\frac{\sigma}{\sqrt{n}}} < z_{\frac{\alpha}{2}}\right) \approx 1 - \alpha$.

This will give us the $(1 - \alpha)$ confidence interval for the unknown mean μ:

$$\bar{X}_n \pm \frac{z_{\alpha/2}}{\sqrt{n}}\sigma$$

In the above case, it is assumed that the variance σ^2 is known, but the expectation μ is unknown.

However, in most realistic cases σ is unknown and it needs to be estimated. Replacing the standard deviation σ by the Sample Standard Deviation S_n is one potential solution. By doing so, we can derive the $(1 - \alpha)$ confidence interval for the unknown mean μ, which will be given by:

$$\bar{X}_n \pm \frac{t_{\alpha/2}}{\sqrt{n}}S_n$$

Here $t_{\alpha/2}$ is the t-value for the t-distribution, defined as $\boldsymbol{P}(t > t_\alpha) = \alpha$.

Notice that, when the Sample Standard Deviation is used instead of σ, the distribution of the standardized Sample Mean is no longer Normal, but it is t-distributed with (n-1) degrees of

freedom. However, for large sample sizes (and due to the Central Limit Theorem) we can still use the z-values (instead of t-values) in the confidence interval derivations.

Thus, when using the LLN to estimate the unknown mean, we have an error: the difference between our estimate and the actual value. This error is given by this formula: $\frac{z_{\alpha/2}}{\sqrt{n}} S_n$. The following result allows users to control the size of the error, by choosing the sample size.

Lemma: Assume ε is the desired error of the Monte Carlo simulation. Then, the size of the sample $\hat{n}$ should be chosen so that it satisfies this condition: $\frac{z_{\alpha/2}}{\sqrt{n}} S_n \leq \varepsilon$.

Assume that S_n is bounded. Then, the error term of the approximation is bounded by $\frac{K}{\sqrt{n}}$, where $K < \infty$ is a constant. That is, the convergence rate of Monte Carlo approximation method is $n^{-1/2}$.

This implies that, if we want to achieve an accuracy of 0.01 in our approximation, then the number of simulations *n* should be about $n = 10^4$.

One can increase n and achieve any desired accuracy in the Monte Carlo approximation. However, this may be computationally costly and not applicable in some cases. There are a few alternative ways to increase the accuracy of the approximation that result in faster convergence of the algorithm. These methods, which are designed to reduce the variance of the approximation and make the convergence faster, are titled variance reduction techniques in the literature and will be covered in the next section.

Example: Suppose we want to price a European Call option by Monte Carlo Simulations. We are assuming to be in the Black-Scholes framework.

We have that $c = e^{-rT}\mathbb{E}^*(S_T - X)^+$, where $S_T = S_0 exp\left(\left(r - \frac{\sigma^2}{2}\right)T + \sigma W_T\right)$.

In order to simulate W_T, notice that $W_T \approx \sqrt{T}\,Z$ where $Z \sim N(0,1)$.

Then, we would follow these steps to estimate *c* by Monte Carlo Simulations:

STEP 1. Generate $\{Z_i\}_{i=1}^{n} \sim i.i.d.\ N(0,1)$,

STEP 2. Compute the payoff of the call option price in scenario i:

$$payoff_i = \left(S_0\ e^{\left(\left(r-\frac{\sigma^2}{2}\right)T+\sigma\sqrt{T}Z_i\right)} - X \right)^+$$

STEP 3. Take $\tilde{c} = e^{-rT}\ \frac{1}{n}\sum_{i=1}^{n} payoff_i = e^{-rT}\ \frac{1}{n}\sum_{i=1}^{n}\left(S_0\ e^{\left(\left(r-\frac{\sigma^2}{2}\right)T+\sigma\sqrt{T}Z_i\right)} - X \right)^+$

Then, $\tilde{c}$ is an estimated value of c.

While this method will provide an estimate for the option price (and we have an explicit formula for the price), the convergence rate is in the order of $n^{-1/2}$.

In the next section, we will discuss variance reduction techniques that will help in reducing the computational time of the estimations that are performed by using Monte Carlo Simulations.

2.6 Variance Reduction Techniques

In this section we will discuss a few techniques for improving the speed and efficiency of Monte Carlo simulation algorithms, usually referred to as "Variance Reduction Techniques".

The success of any variance reduction technique is dependent on the problem of interest. Since there is no theory to show which method should be used for what type of problems, preliminary runs are necessary to assess a variance reduction technique's effectiveness.

Suppose we want to estimate an unknown parameter θ. Assume $\{X_i\}^{n}_{\ i=1} \sim i.i.d.$ with $\mathbb{E}(X_i) = \theta$ and $Var(X_i) = \sigma^2$.

If we take the Sample Mean $\bar{X}_n = \frac{X_1+X_2+\ldots+X_n}{n}$ of the sequence, then, we know that $\mathbb{E}\bar{X}_n = \theta$.

That is, the Sample Mean is an *unbiased estimator* of θ.

It follows from the Central Limit Theorem (CLT) that for large enough n we have:

$$\frac{\bar{X}_n - \theta}{\sigma/\sqrt{n}} \sim N(0,1)$$

That is, the Sample Mean not only is an unbiased estimator for the unknown parameter θ, but it converges (in distribution) to it. It would be desired to have as low variance (variation) in our estimates (for θ) as possible. That is, we would want the estimate to be as "close" to the unknown parameter as possible, where "close" is to be interpreted in a probabilistic sense: having low variance.

Thus, having lower variance of the estimators would help the algorithm converge faster. That is, we have a criterion to compare two unbiased estimators of unknown parameters: if we have two unbiased estimators of θ, then we'll prefer the one that has a smaller variance.

2.6.1 Antithetic Variates

This method is widely used in problems of pricing financial instruments. Suppose we want to estimate $\theta = \mathbb{E}(X)$ and that we can generate random variates X_1 and X_2, so that $\mathbb{E}X_1 = \mathbb{E}X_2 = \theta$. Both X_1 and X_2 are unbiased estimators for θ, and so is $\frac{X_1+X_2}{2}$. We have

$$Var\left(\frac{X_1 + X_2}{2}\right) = \frac{1}{4}(VarX_1 + VarX_2 + 2Cov(X_1, X_2))$$

The variance of the estimator $\frac{X_1+X_2}{2}$ would be reduced if the variates X_1 and X_2 were negatively correlated (when X_1 and X_2 are not independent.)

That is, if $Cov(X_1, X_2) < 0$ then $\frac{X_1 + X_2}{2}$ will have lower variance than either X_1 or X_2. This is the basic idea behind the Antithetic Variates variance reduction technique.

Example: Suppose we want to estimate $\theta = \mathbb{E}(e^U)$ for $U \sim U[0,1]$. The actual value of the integral is known to be $(e - 1)$.

Define $X = e^U$. Take $X_1 = e^{U_1}$, $X_2 = e^{U_2}$.

If U_1 is independent of U_2, then, $Var\left(\frac{e^{U_1} + e^{U_2}}{2}\right) = 0.1210$.

If $U_2 = 1 - U_1$ then, $Cov(X_1, X_2) = -0.2342 < 0$. Also, $Var\, e^U = 0.2420$ and

$Var\left(\frac{e^U + e^{1-U}}{2}\right) = 0.0039$.

This is a reduction of a variance by about 97%. It is because U_1 and $U_2 = 1 - U_1$ are negatively correlated and have the lowest correlation possible.

In general, suppose we are interested in estimating $\mathbb{E}\, h(X_1, X_2, \ldots, X_n)$ where $\{X_i\}_{i=1}^n$ is a sequence of independent random variates.

Assume, that we are able to simulate $\{X_i\}_{i=1}^n$ by the inverse-transformation method: $\{X_i\}_{i=1}^n$ are generated by $\{U_i\}_{i=1}^n \sim i.i.d.\ U[0,1]$, so that $F^{-1}(U_i)$ has the same distribution as X_i.

Define $Y_1 = h(F^{-1}(U_1), \ldots, F^{-1}(U_n))$ and $Y_2 = h(F^{-1}(1 - U_1), \ldots, F^{-1}(1 - U_n))$.

Then, Y_1 and Y_2 have the same distribution. If $h(\cdot)$ is a monotone function, then Y_1 and Y_2 are negatively correlated and the antithetic variates estimate $\frac{Y_1 + Y_2}{2}$ has a lower variance than either one of Y_i's. The following result helps to justify the use of the method in a more general case.

Lemma: If $h(x_1, x_2, \ldots, x_n)$ is a monotone function of its arguments, then for $\{U_i\}^n{}_{i=1} \sim i.i.d.\ U[0,1]$, we have $cov(\ h(U_1, U_2, \ldots, U_n),\ h(1-U_1, 1-U_2, \ldots, 1-U_n)) \leq 0.$

In general, for each simulated sample, where $U_i^{'}s$ were used, we obtain the second sample by using $(1-U_i)'s$

Example: Suppose we want to price a European Call option by simulation and by implementing the Antithetic Variates technique of variance reduction.

Using the risk-neutral option pricing framework, one can say that the price of a European Call option is the present value of the expected future payoff at maturity, under the risk-neutral measure.

That is, $c = e^{-rT}\mathbb{E}^*(S_T - X)^+$, where $S_T = S_0 exp\left(\left(r - \frac{\sigma^2}{2}\right)T + \sigma W_T\right)$.

Then, we would follow these steps to estimate the price:

STEP 1. Generate $\{Z_i\}_{i=1}^n \sim i.i.d.\ N(0,1)$,

STEP 2. Compute the following two realizations of the call option price (using Z_i in one and $-Z_i$ in the second):

$$c^+{}_i = e^{-rT}\left(S_0\ e^{\left(\left(r-\frac{\sigma^2}{2}\right)T+\sigma\sqrt{T}Z_i\right)} - X\right)^+ \text{ and}$$

$$c^-{}_i = e^{-rT}\left(S_0\ e^{\left(\left(r-\frac{\sigma^2}{2}\right)T+\sigma\sqrt{T}(-Z_i)\right)} - X\right)^+$$

STEP 3. Define $c_i = \frac{c^+{}_i + c^-{}_i}{2}$, and take $\tilde{c} = \frac{1}{n}\sum_{i=1}^n c_i$.

Then, $\tilde{c}$ approximates the actual value of c.

Comment:

Notice that, if we generate $\{Z_i\}_{i=1}^{n} \sim i.i.d.N(0,1)$, then the odd moments of the random sample $\{\{Z_i\}_{i=1}^{n}, \{-Z_i\}_{i=1}^{n}\}$ are exactly equal to zero, by construction. However, if we had generated a random sample $\{Z_i\}_{i=1}^{2n} \sim i.i.d.\ N(0,1)$, then, the odd (empirical) moments are not exactly equal to zero.

2.6.2 Control Variates

Suppose we want to estimate the unknown parameter $\theta = \mathbb{E}(X)$ and that we know for a random variable Y, $\mathbb{E}(Y) = \delta$, where δ is known. Y is called a control variate, and normally it is correlated with X.. For any constant γ define another random variable $T = X - \gamma(Y - \delta)$. Then, T is an unbiased estimator for θ: $\mathbb{E}T = \mathbb{E}X - \gamma(\mathbb{E}Y - \delta) = \theta$.

The idea of this technique is to choose the free parameter γ in such a way that the estimator T has a lower variance than X. We have:

$$Var(T) = \gamma^2 Var(Y) - 2\gamma Cov(X,Y) + Var(X).$$

The variance of T is the lowest if we choose $\gamma = \frac{Cov(X,Y)}{Var(Y)}$. With this choice of γ, we get $Var(T) = (1 - \rho^2)Var(X)$, where ρ is the correlation between X and Y. It is easy to see that the greater the correlation ρ between X and Y, the more the variance reduction will be.

In practice, it is not known what $Cov(X,Y)$ and $Var(Y)$ are, so a few test simulations must be run to estimate them, then only we can choose the optimal γ.

$Cov(X,Y)$ will be estimated as follows:

$$\widehat{Cov(X,Y)} = \frac{\sum(Y_i - \bar{Y})(X_i - \bar{X})}{n-1}$$

Comment: One may use more than one control variates in simulations as follows:

$$T = X - \sum_{i=1}^{m} \gamma_i \, (Y_i - \delta_i)$$

where $\mathbb{E}Y_i = \delta_i$ (δ_i are known).

Example. Suppose we want to estimate $\theta = \mathbb{E}\left[e^{(U+W)^2}\right]$, where $U, W \sim \ i.i.d. \, U(0,1)$.

We can use the Monte Carlo simulation technique to estimate the parameter θ as follows:

1. Generate $U_1, \dots, U_n, W_1, \dots, W_n$ all *i.i.d.* $U(0,1)$;
2. Compute $X_1 = e^{(U_1+W_1)^2}, \dots, X_n = e^{(U_n+W_n)^2}$
3. Take $\hat{\theta}_n = \frac{\sum_{j=1}^{n} X_j}{n}$ to estimate θ.

To use the Control Variate technique in the above estimation, the first step is to choose an appropriate control variate Z.

Good control variates have two main properties:

1. Their expected values are known,
2. The correlation between the control variate and the original random variable is high.

In this example, there are several candidates. For example, we could choose

1. $Z_1 = U + W, \quad \mathbb{E}[Z_1] = 1$
2. $Z_2 = (U + W)^2, \ \mathbb{E}[Z_2] = \frac{7}{6}$
3. $Z_3 = e^{U+W}, \ \mathbb{E}[Z_3] = (e-1)^2$

In all these cases we expect the covariance between our control variate Z and $Y = e^{(U+W)^2}$ to be positive. Numerical calculations show that the correlation coefficient between Y and Z_3 is the highest, closely followed by the correlation coefficient between Y and Z_2.

Suggested exercises

1. Estimate $\mathbb{E}[Y]$ with and without control variates.
2. Determine the variance reduction in the example above. That is, compare the variance of T_i to the variance of Y, where $T_i = Y + \gamma(Z_i - E[Z_i],\ i = 1, \dots, 3.$
3. Can you find other control variates which result in a bigger variance reduction?
4. Assume you want to price an Asian Call option. What are some possible control variate candidates?

2.6.3 Conditioning

We now consider a simple, but powerful generalization of the Control Variate technique. Computing expected values by conditioning is a common technique in probability theory. When we want to estimate $\mathbb{E}(X)$, it is sometimes useful to condition the random variable X on another random variable Y. The following result will help us justify the use of conditioning in reducing the variance of our estimation:

Theorem.

(a) $\mathbb{E}(X) = \mathbb{E}\{\mathbb{E}\,[\,X|Y]\}$

(b) $Cov(X,Y) = \mathbb{E}\{cov(X,Y|Z)\} + \ cov\{\,\mathbb{E}\,[X|Z], \mathbb{E}[Y|Z]\}$

(c) $Var(X) = \ \mathbb{E}\{Var[X|Y]\} + \ Var\{\mathbb{E}[X|Y]\}$

Comment: In case of discrete distributions , the double expectation can be expanded to be written as follows:

$$\mathbb{E}\{\mathbb{E}\,[\,X|Y]\} = \sum_i \mathbb{E}(X|Y = y_i)P(\,Y = y_i) = \sum_i \sum_k x_k\, P(X = x_k|Y = y_i)P(\,Y = y_i)$$

The above theorem can be used as follows to reduce the variance of estimation.

Suppose we are estimating $\theta = \mathbb{E}(X)$. Property (a) of the above result implies that for any random variable Y, $\mathbb{E}\,[\,X|Y]$ is an unbiased estimator for θ. It follows from part (c) of the Theorem that $Var(X) = \mathbb{E}\{Var[X|Y]\} + Var\{\mathbb{E}[X|Y]\} \geq Var\{\mathbb{E}[X|Y]\}$.

That, is $\mathbb{E}\,[\,X|Y]$ and X are both unbiased estimators for θ , but $\mathbb{E}\,[\,X|Y]$ has a lower variance then X, and therefore, it is preferable as an estimator.

In general, the search for an appropriate Y to condition on requires searching for a random variable Y such that:

(a) The conditional expectation $\mathbb{E}\,[\,X|Y]$ is computable, and

(b) $Var\{\mathbb{E}[X|Y]\}$ is substantially smaller than $Var(X)$.

Example.

We want to estimate $\theta = P(U + Z > 4)$ where $\mathrm{U} \sim \mathrm{Exp}(1)$ and $\mathrm{Z} \sim \mathrm{Exp}(0.5)$.

Notice that $\mathrm{P}(\mathrm{U} + \mathrm{Z} > 4) = \mathbb{E}[1_{(\mathrm{U}+\mathrm{Z}>4)}]$. Define $\mathrm{Y} = 1_{(\mathrm{U}+\mathrm{Z}>4)}$.

Then, we can use the standard Monte Carlo simulation method to estimate θ as follows:

1. Generate $\mathrm{U}_1, \dots, \mathrm{U}_\mathrm{n} \sim i.i.d.\ \mathrm{Exp}(1)$, $\mathrm{Z}_1, \dots, \mathrm{Z}_\mathrm{n} \sim i.i.d.\ \mathrm{Exp}(0.5)$
2. Define $\mathrm{Y_i} = 1_{(\mathrm{U_i}+\mathrm{Z_i}>4)}$ for $\mathrm{i} = 1, \dots, \mathrm{n}$.

3. Use the following estimate: $\hat{\theta}_n = \frac{1}{n}\sum_{i=1}^{n} Y_i$

If we want to use the conditioning approach, define $V = \mathbb{E}(Y|Z)$. Then,

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

where $F_u(\cdot)$ is the cumulative distribution function of U. Therefore

$$1 - F_u(4 - z) = \begin{cases} e^{-(4-z)} \; if \; 0 \le z \le 4, \\ 1 \quad if \; z > 4. \end{cases}$$

Hence

$$V = \mathbb{E}[Y|Z] = \begin{cases} e^{-(4-Z)} \; if \; 0 \le Z \le 4, \\ 1 \; if \; Z > 4. \end{cases}$$

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

1. Generate $Z_1, \dots, Z_n$ all independent, $Exp(0.5) -$ distributed,,
2. Set $V_i = \mathbb{E}[Y|Z_i]$ for $i = 1, \dots, n$,
3. Compute $\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} V_i$ and use it as an estimator for the unknown parameter.

Suggested problems.

1. Implement the two variance reduction algorithms in the example above. Is there a difference in their convergence rates?
2. How can we use the conditioning technique to price a chooser option? A chooser option is a European style option with maturity T_2. At time $T_1 < T_2$ you may choose if the option is a call or a put; the options' strike prices are fixed at K at time $t = 0$.

2.6.4 Stratified Sampling

Suppose we are interested in estimating $\theta = \mathbb{E}(X)$ and that X is somehow "dependent" on the value of another random variable Y, which takes a finite set of values with known probabilities: $P(Y = y_k) = p_k$ for $k = 1, \dots, N$.

Then, we can write the expression for θ in terms of the distribution of Y:

$$\theta = \mathbb{E}\,X = \sum_{k=1}^{N} \mathbb{E}(X|Y = y_k)P(Y = y_k)$$

If Y is chosen in such a way that the conditional expectations are relatively easy to calculate, then, based on the conditional variance results of the last Theorem, we have an estimator with lower variance.

More generally, if a population can be separated into several sub-populations ("strata") according to their variability, then we can take more samples in the sub-populations with high variability than in the sub-populations with low variability, to estimate the population mean. This idea is illustrated by the following example.

Example. Suppose we want to estimate the following double integral using the idea of stratified sampling.

$$\mu = \int_{-1}^{1}\int_{-1}^{1} \exp\left(\sqrt[3]{x_1} + \sqrt[3]{x_2}\right) dx_1 dx_2$$

In order to be able to use $U(0,1) -$ distributed random variables to estimate the integral, we apply the following scaling transformation:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 2\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} - \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

This transformation defines a function $x = g(u)$. The determinant of its Jacobian is given by

$$|Jg(u)| = \begin{vmatrix} \frac{\partial g_1}{\partial u_1} & \frac{\partial g_1}{\partial u_2} \\ \frac{\partial g_2}{\partial u_2} & \frac{\partial g_2}{\partial u_2} \end{vmatrix} = \begin{vmatrix} 2 & 0 \\ 0 & 2 \end{vmatrix} = 4$$

Thus, we have

$$\mu = \int_{-1}^{1}\int_{-1}^{1} \exp\left(\sqrt[3]{x_1} + \sqrt[3]{x_2}\right) dx_1 dx_2 = \int_{0}^{1}\int_{0}^{1} 4\exp\left(\sqrt[3]{2u_1 - 1} + \sqrt[3]{2u_2 - 1}\right) du_1 du_2.$$

To determine the strata, we plot the function $h(u_1, u_2) = 4\ \exp\left(\sqrt[3]{2u_1 - 1} + \sqrt[3]{2u_2 - 1}\right)$ $for\ 0 \le u_1 \le 1, 0 \le u_2 \le 1.$

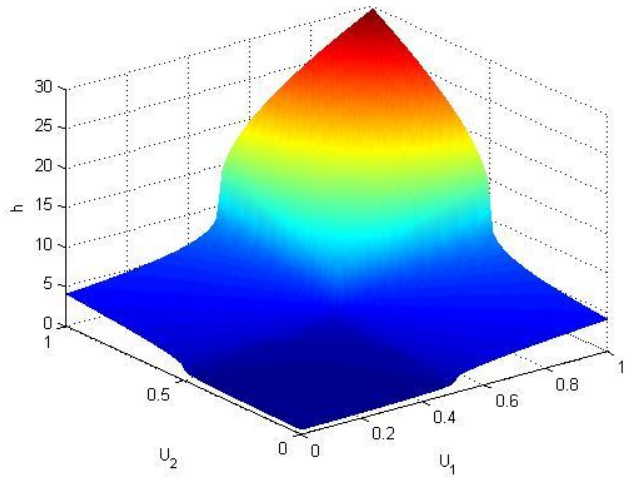

Figure 1: *Plot of Function* ***h***.

The graph shows us that the function $h(u_1, u_2)$ is most variable for $0.5 \le u_1, u_2 \le 1$.

For a given sample size, a Monte Carlo estimation in that sub-region would produce a higher standard error than a Monte Carlo estimation in the remaining area. In order to produce a more accurate result, we use stratified sampling.

Define the three sub-regions of $[0,1] \times [0,1]$:

1. $\Omega_1 = \{(u_1, u_2): 0 \leq u_1 \leq .5 \; and \; 0 \leq u_2 \leq .5\}$
2. $\Omega_2 = \{(u_1, u_2): (0 \leq u_1 \leq .5 \; and \; .5 \leq u_2 \leq 1) \; or \; (.5 \leq u_1 \leq 1 \; and \; 0 \leq u_2 \leq .5)\}$
3. $\Omega_3 = \{(u_1, u_2): 0.5 \leq u_1 \leq 1 \; and \; 0.5 \leq u_2 \leq 1\}$

From the graph it can be observed that we want to sample more often in Ω_3 than in Ω_1 or Ω_2. The number of samples in a region should depend on its size (probability weight) and the variability of h in that region relative to the overall (weighted) variability.

Define the probability weight $p_i = P(\Omega_i)$. In order to measure the variability in the different regions, let U_{Ω_i} be the restriction of $U = (U_1, U_2)$ on the set Ω_i. Thus, the variability of h restricted to the set Ω_i can be measured by $\sigma_i = \sqrt{Var\left(h\left(U_{\Omega_i}\right)\right)}$. Assuming we want to take n overall samples, the sample size n_i for $\Omega_i, 1 \leq i \leq 3$ is given by

$$n_i = \frac{p_i \times \sigma_i}{p_1 \times \sigma_1 + \; p_2 \times \sigma_2 + \; p_3 \times \sigma_3}$$

Notice that the standard deviations $\sigma_i, 1 \leq i \leq 3$ need to be estimated. Let $U^k{}_{\Omega_i}$ be the k-th realization of the random variable U_{Ω_i}. For example, for $i = 1, \dots, 3$ generate 50 U_{Ω_i} distributed random variables and compute $h\left(U^1{}_{\Omega_i}\right), \dots, h\left(U^{50}{}_{\Omega_i}\right)$. Then, compute the sample standard deviation s_i of this sample. Thus,

$$n_i = \frac{p_i \times s_i}{p_1 \times s_1 + \; p_2 \times s_2 + \; p_3 \times s_3}.$$

Estimating the standard deviation results in $s_1 = \; .281, s_2 = .893, s_3 = 4.478$. Since we will need estimates for $E\left[h\left(U_{\Omega_i}\right)\right], i = 1, \dots, 3$, we compute these as well: $\hat{\mu}_1 \; = \; .930$, $\hat{\mu}_2 \; = \; 4.119$ and $\hat{\mu}_3 \; = \; 18.040$. The probability weights are $p_1 = .25, p_2 = .5$ and $p_3 = .25$.

Assuming that the overall sample size is $n = 1000$, the sample sizes for the three regions are given by $n_1 = 43,\ n_2 = 273$ and $n_3 = 684$. Since our derivation involved a Monte Carlo simulation to estimate the standard deviations, your result may differ slightly. Without stratified sampling we would on average sample 250 times from Ω_1, 500 times from Ω_2 and 250 times from Ω_3.

We can now construct our stratified sampling estimator. It is given by

$$\hat{\theta}_{st} = \frac{\sum_{k=1}^{m_1} h\left(U^k{}_{\Omega_1}\right)}{m_1}\, p_1 + \frac{\sum_{k=1}^{m_2} h\left(U^k{}_{\Omega_2}\right)}{m_2} p_2 + \frac{\sum_{k=1}^{m_3} h\left(U^k{}_{\Omega_3}\right)}{m_3} p_3$$

By how much does the stratified sampling approach reduce the variance? We can compare the standard deviation of the stratified sampling estimator with the standard deviation of a crude Monte-Carlo estimator. Since the random variables

$$U^1{}_{\Omega_1}, \dots, U^{m_1}{}_{\Omega_1},\ U^1{}_{\Omega_2}, \dots, U^{m_2}{}_{\Omega_2}, U^1{}_{\Omega_3}, \dots, U^{m_3}{}_{\Omega_3}$$

are all independent, the standard deviation of $\hat{\theta}_{st}$ is given by

$$\sigma_{\hat{\theta}_{st}} = \sqrt{\left(\frac{p_1}{m_1}\right)^2 \sum_{k=1}^{m_1} \sigma_1^2 + \left(\frac{p_2}{m_2}\right)^2 \sum_{k=1}^{m_2} \sigma_2^2 + \left(\frac{p_3}{m_3}\right)^2 \sum_{k=1}^{m_3} \sigma_3^2}$$

Replacing the standard deviations σ_i by the sample standard deviations s_i, we obtain

$$s_{\hat{\theta}_{st}} = \sqrt{\left(\frac{p_1}{m_1}\right)^2 \sum_{k=1}^{m_1} s_1^2 + \left(\frac{p_2}{m_2}\right)^2 \sum_{k=1}^{m_2} s_2^2 + \left(\frac{p_3}{m_3}\right)^2 \sum_{k=1}^{m_3} s_3^2}$$

If we use the values calculated above, we have

$$s_{\hat{\theta}_{st}} = 0.517$$

The standard deviation of the crude Monte Carlo estimator is given by

$$\sigma_{MC} = \frac{\sigma}{\sqrt{n}},$$

where $\sigma$is the population standard deviation of $h(U)$. We could now run another Monte-Carlo Simulation to find an estimate for σ. This, however, but it is not computationally efficient. We can also treat $h(U)$ as a mixture of the distributions of the random variables $h(U_{\Omega_1}), h(U_{\Omega_2})$ and $h(U_{\Omega_3})$. This means that the probability density function $f_{h(U)}$ of $h(U)$ is given by

$$f_{h(U)}(u_1, u_2) = \sum_{i=1}^{3} f_{h(U_{\Omega_i})}(u_1, u_2) p_i.$$

Then, $\hat{\mu} = p_1\ \hat{\mu}_1 + p_2\ \hat{\mu}_2 + p_3\ \hat{\mu}_3 = 6.802$ and

$$s = \sqrt{(s_1^2 + \hat{\mu}_1^2)p_1 + (s_2^2 + \hat{\mu}_2^2)p_1 + (s_3^2 + \hat{\mu}_3^2)p_1 - \hat{\mu}^2} = 7.016$$

Thus the standard deviation of the crude Monte Carlo estimator is $\sigma_{MC} = \frac{7.016}{\sqrt{1000}} = .222.$

This means, if we wanted to achieve a standard error of the same size as the stratified sampling estimator for the crude Monte Carlo simulation, we would have to sample

$$n = \left(\frac{s}{s_{\widehat{\theta}_{st}}}\right)^2 = 18{,}417$$

realizations of the random variable U.

Back to our original task, the stratified sampling estimator returns 6.928 as the value of the integral.

Suggested Exercise.

1. Implement the previous example to analyze the stratified sampling method.

2.5.6 Importance Sampling

The idea here is to distort the underlying probability measure to estimate the parameter under consideration, with less variance. This is achieved by using the Girsanov's Theorem to "change" the underlying measure in such a way that is computationally less expensive to estimate the unknown parameter under another measure. That is, we will attempt to reduce the variance by changing the probability distribution from which the random variates are generated.

Suppose we are interested in estimating the unknown parameter θ:

$$\theta = \mathbb{E}\, g(X) = \int_{-\infty}^{\infty} g(x)f(x)dx,$$

where $f(\cdot)$ is the probability density function (pdf) of the random variable X.

The idea is to choose a function $t(.)$ so that $\frac{g(x)f(x)}{t(x)}$ is as smooth as possible (ideally, we would like it to be a constant, but in reality, it is impossible).

Assume $Support(f) \in Support\,(t)$, where $t(\cdot)$ is a density function. Then,

$\theta = \int_{-\infty}^{\infty} \frac{g(x)f(x)}{t(x)} t(x)dx = \mathbb{E}_t\left(\frac{g(Y)f(Y)}{t(Y)}\right)$, where Y is a random variable with density function $t(\cdot)$.

The algorithm for estimating θ is as follows:

1. Generate $Y_1, Y_2, \ldots, Y_n$ from the density $t(\cdot)$,
2. Compute $\hat{\theta}_n = \frac{1}{n}\sum_{i=1}^{n} \frac{g(Y_i)f(Y_i)}{t(Y_i)}$ and use it as an estimator for θ.

We can claim that:

(a) $\hat{\theta}_n$ is an unbiased estimator for θ: $\mathbb{E}\hat{\theta}_n = \theta$.

(b) $Var\hat{\theta}_n = \frac{1}{n}\left(\mathbb{E}_t\left(\frac{g(Y)f(Y)}{t(Y)}\right)^2 - \theta^2\right)$.

Exercise:

Relate this algorithm to the Acceptance-Rejection algorithm.

We will illustrate the above-technique by the following examples.

Example 1. Assume that the stock price at time T has $N(100,1)$ distribution. You want to price a digital call option, struck at \$110, by Monte Carlo simulation. Let the initial stock price be $S_0 =$ \$100.

The main problem here is that, if we simulate paths of the stock price, almost all paths will end up out of the money for the option we are trying to price. The probability of a ten standard deviation move of the stock price is very small. Therefore, the use of a Monte-Carlo Simulation will result in a call price of zero. We can apply importance sampling to this problem.

Our goal is to evaluate

$$\int_{-\infty}^{\infty} f(S_0 + z)\phi(z)dz$$

where f is the payoff function of the digital call:

$$f(x) = \begin{cases} 0, & x < 110 \\ 1, & x \geq 110 \end{cases}$$

and $\phi(z)$ is the probability density function of a standard normal random variable. Let ψ be the probability density function of a normal random variable with mean 10 and standard deviation 1. Then, we can write the above integral as

$$\int_{-\infty}^{\infty} f(S_0 + z)\frac{\phi(z)}{\psi(z)}\psi(z)dz = \mathbb{E}_\psi\left[f(S_0 + Z)\frac{\phi(Z)}{\psi(Z)}\right]$$

If we run a Monte Carlo simulation to estimate the expected value, then we draw realizations from a normal distributed random variable with mean 10 and standard deviation 1 and evaluate

$$f(S_0 + z)\frac{\phi(z)}{\psi(z)} = f(S_0 + z)e^{\frac{10^2}{2} - 10z}$$

Notice that, we sample paths where the payoff of the digital call jumps. Using importance sampling we actually have paths which end up in the money, whereas a crude Monte Carlo simulation does not produce any paths which end up in the money.

An implementation of this problem shows that a crude Monte Carlo Simulation results in a price of zero, whereas the Monte Carlo simulation using the importance sampling procedure gives a price of 7.6485e-024. This is still very small, but it is more accurate.

Although very stylized, this example demonstrates the power of importance sampling.

It is now a very natural question to ask which choice of $t(\cdot)$ most reduces the variance of the importance sampling estimator?

$$\hat{\theta}_{IS} = \frac{1}{n}\sum_{i=1}^{n}\frac{g(X_i)f(X_i)}{t(X_i)}$$

Consider the variance of the importance sampling estimator, given by

$$Var\left(\hat{\theta}_{IS}\right) = \frac{1}{n}\mathbb{E}\left[\frac{g(X)f(X)}{t(X)} - \mathbb{E}\left[\frac{g(X)f(X)}{t(X)}\right]\right]^2$$

Looking at this expression and assuming that g is non-negative, we see that we can find an importance sampling estimator with zero variance if we choose $t(x) = \alpha g(x)f(x)$, where α is a normalizing constant that is chosen so that, t is a probability density function. Unfortunately, the normalizing constant α is nothing else than the expected value we are trying to estimate in the first place.

But, it illustrates that it is desirable to choose $t(\cdot)$ as close as possible to being proportional to $|g(\cdot)|f(\cdot)$.

Furthermore, a good importance sampling function $t(x)$ should possess the following additional properties:

I. $t(x) \geq 0$ for $\forall x$ for which $g(x)f(x) \neq 0$.

II. It should be easy to simulate variates that have pdf of $t(x)$.

III. It should be easy to compute the density $t(x)$ for any value x which can be realized.

General examples of $t(\cdot)$ (to be used in the Importance Sampling technique) are given by the **Esscher Transform** as follows:

$$t(x) = \frac{e^{ux}f(x)}{E_f(e^{uX})}$$

Examples.

1. **Exponential:** $f(x) = \alpha e^{\alpha x}$. Then, $t(x) = (\alpha - u)e^{(\alpha-u)x}$, which ha exponential distribution, provided that $\alpha > u$.
2. **Bernoulli:** $f(x) = p^x(1-p)^{1-x}$. for $x = 0, 1$. Then, $t(x) = \left(\frac{pe^u}{pe^u+1-p}\right)^x \left(\frac{1-p}{pe^u+1-p}\right)^{1-x}$

Example 3 (By Wei Cai. used by permission).

The Importance Sampling idea is demonstrated by using one of the classical integral calculations:

$$\int_0^1 \sqrt{1-x^2}dx = \frac{\pi}{4}.$$

To illustrate the use of the class of functions in importance sampling, consider the following family of density functions:

$$t(x) = \begin{cases} \dfrac{1 - ax^2}{1 - \dfrac{a}{3}} & \text{for } 0 \le x \le 1 \\ 0 & \text{otherwise} \end{cases}$$

It can be shown that with this function $t(.)$, the variance of $\frac{g(Y)f(Y)}{t(Y)}$ will be given by:

$$Var\left(\frac{g(Y)f(Y)}{t(Y)}\right) = \left(1 - \frac{a}{3}\right)\left(\frac{1}{a} - \frac{1-a}{a^{3/2}}\, tanh^{-1}(a^{-\frac{1}{2}})\right) - \left(\frac{\pi}{4}\right)^2.$$

Since our goal is to find $t(.)$ that makes the variance of our estimator as small as possible, we will choose the constant *a* in such a way that the above variance is the smallest. It is easy to see that $a = 0.74$ is the number (numerical calculations) that minimizes the variance. Notice that, $a = 0$ is the case in which we just use the uniform distribution.

Suggested exercises.

Try the following choices of $g(x)$ in implementing the Importance Sampling technique.

(1) $g(x) = \frac{(1-\alpha x)}{(1-\frac{\alpha}{2})}$. What is the optimal α to minimize the computational cost here?

(2) $g(x) = \frac{(1-\alpha x^2)}{(1-\frac{\alpha}{3})}$. What is the optimal α to minimize the computational cost here? (Try $\alpha = 0, 0.35, 0.48, 0.65, 0.74, 0.86, 1$).

Comments.

1. To gain some more intuitive insight about Importance Sampling, refer to the paper on "Importance sampling: an illustrative introduction" by P H Borcherds.
2. For an application of Importance Sampling in pricing barrier options, see the book *"Using Monte Carlo Simulation and Importance Sampling to Rapidly Obtain Jump-Diffusion Prices of Continuous Barrier Options"*, by Mark S. Joshi and Terrence Leung.

Simulation Efficiency

As was seen earlier, we can build confidence intervals for parameter estimators as follows:

$$\bar{X}_n \pm \frac{z_{1-\alpha/2}}{\sqrt{n}} S_n.$$

One way to measure the efficiency of our estimator $\bar{X}_n$, is to consider the length of the confidence interval: $\frac{2z_{1-\alpha/2}}{\sqrt{n}} S_n = 2z_{1-\alpha/2}\sqrt{\frac{S_n}{n}}$.

We would like to have a relatively small interval (which means less error), but it is not always possible to achieve that goal, with a "reasonable computational cost". Of course, by increasing the number of simulations (which means higher computational cost), one can decrease the error term and achieve higher accuracy. Therefore, the question is: how to estimate the quality of the estimation, by taking into account the computational cost and the accuracy of the approximation.

One measure of simulation efficiency would be the combination of these two: the computational cost and accuracy of the clagorithm.

Let cc be the computational cost to simulate <u>one</u> realization of a random variable under consideration; n be the number of simulations; and let w be the error term, measured by the width of the confidence interval. We have:

$$n = \left(\frac{2S_n z_{1-\alpha/2}}{w}\right)^2$$

Then, the total computational cost of simulations –TCC - will be computed as

$$TCC = n * cc = cc * \left(\frac{2S_n z_{1-\alpha/2}}{w}\right)^2$$

Thus, two algorithms, i and j, will be compared to each other in terms of their efficiency as follows:

If:

$$TCC_i = cc_i * \left(\frac{2S_n^i z_{1-\alpha/2}}{w}\right)^2 < TCC_j = cc_j * \left(\frac{2S_n^j z_{1-\alpha/2}}{w}\right)^2$$

then, the i is more efficient algorithm han j.

In other words, if $cc_i * \left(S_n^i\right)^2 < cc_j * \left(S_n^j\right)^2$ then, i is more efficient than j.

Notice, that we can replace $\boldsymbol{cc}$ by t (computing time) and measure and compare the efficiency of simulation methods.

2.6 Exercises

1. Generate a series (X_i, Y_i) for $i = 1, \dots, n$ Bivariate-Normally distributed random vectors, with mean of $(0,0)$ and the variance – covariance matrix of $\begin{pmatrix} 1 & 0.7 \\ 0.7 & 1 \end{pmatrix}$. Compute the following:

$$\rho = \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

where $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$, and $\bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$, and $n = 1000$.

2. Evaluate the following expected value by using Monte Carlo simulation:

$$E(X^3 + \sin(Y) + X^2 Y)$$

where X and Y have $N(0,1)$ distribution and a covariance of 0.65.

3. (a) Estimate the following expected values:

$$E({W_5}^2 + \sin(W_5),$$

$$E\left(e^{\frac{t}{2}}\cos(W_t)\right) for\ t = 0.5, 4, 8.$$

where W_t is a Wiener Process.

(b) How are the values of the last three integrals (for the cases $t = 0.5, 4, 8$) related?

(c) Now use a variance reduction technique (whichever you prefer) to compute the expected value in part (a). Comment on potential improvements.

4. Let S_t be a Geometric Brownian Motion process: .

$$S_t = S_0\ e^{\left(\sigma W_t + \left(r - \frac{\sigma^2}{2}\right)t\right)}$$

where $r = 0.04, \sigma = 0.2, S_0 = 88$, and W_t is a Standard Brownian Motion process (Wiener process).

(a) Estimate the price c of a European Call option on the stock with $T = 10, X = 100$ *by using Monte Carlo simulation*.

(b) Now use variance reduction techniques to compute the price in part (a) again. Did the accuracy improve? You may compute the exact value of the option c by the Black-Scholes formula, by using Excel. Now estimate c by crude Monte Carlo simulation, and then by using different variance reduction techniques to see if there are any improvements in convergence.

5. Simulate 4 paths of $S_t for\ 0 \leq t \leq 10$ (defined in Problem 4) by dividing up the interval [0, 10] into 1,000 equal parts. Then, for each integer number n from 1 to 10, estimate ES_n. Plot all of them in one graph.

6. Consider the following integral for computing the number π: $\int_0^1 \sqrt{1-x^2}dx = \frac{\pi}{4}$

(a) The integral can be estimated by simple numerical integration using, say Euler's (or any other) scheme. Estimate the integral numerically. Then, estimate it by Monte Carlo simulation.

(b) Now try the Importance Sampling method (described earlier) to improve the estimate of π of part (a).

7. (*Popular interview question on stochastic calculus*) What can you say about the following integrals?

$$\int_0^T W_t dt, \text{ and } \int_0^T W_t dW_t$$

Use the simulation techniques learned to simulate the integrals and empirically find their distributions.

8. Evaluate the following expected values and probabilities:

$$E\left(X_2^{\frac{1}{3}}\right), \qquad E(Y_3), \qquad E\big(X_2 Y_2\, \mathbf{1}(X_2 > 1)\big), \qquad P(Y_2 > 5).$$

where the Ito's processes X and Y evolve according to the following SDEs:

$$dX_t = \left(\frac{1}{5} - \frac{1}{2}X_t\right)dt + \frac{2}{3}dW_t,\ X_0 = 1,$$

$$dY_t = \left(\left(\frac{2}{1+t}\right)Y_t + \frac{1+t^3}{3}\right)dt + \frac{1+t^3}{3}dZ_t,\ Y_0 = \frac{3}{4}$$

$W\ and\ Z$ are independent Wiener processes, and $\mathbf{1}(X_2 > 1) = 1\ \ if\ X_2 > 1, and\ 0\ if\ X_2 \leq 1.$

9. Estimate the following expected values and compare them to each other.

$$E(1+X_3)^{1/3},\ and\ \ E(1+Y_3)^{1/3}$$

where $dX_t = \frac{1}{4}X_t dt + \frac{1}{3}X_t dW_t - \frac{3}{4}X_t dZ_t,\ X_0 = 1$ and $Y_t = e^{-0.08t + \frac{1}{3}W_t + -\frac{3}{4}Z_t}$

and $W\ and\ Z$ are independent Wiener processes.

Compute the probability (by simulation) that a European *put* option will expire in the money. Use these parameters: $S_0 = 20, X = 20, \sigma = 0.25, r = 0.04$ and $T = 0.5$ years.

10. Compare the pseudorandom sample with the quasi Monte Carlo sample of $Uniform[0,1]x[0,1]$:

a) Generate 100 vectors of $Uniform\ [0,1]x[0,1]$ by using MATLAB (or the software you are using) random number generator.

b) Generate 100 points of the 2-dimentional Halton sequences, using bases 2 and 7.

c) Generate 100 points of the 2-dimentional Halton sequences, using bases 2 and 4. (4 is non-prime!).

d) Draw all on separate graphs and see if there are differences in the three (visual test only).

e) Use 2-dimensional Halton sequences and compute the following integral:

(use $N = 10{,}000$. Try different pairs of bases: $(2,4), (2,7), (5,7)$.)

$$\int_0^1 \int_0^1 e^{-xy} \left(\sin(6\pi x) + cos^{\frac{1}{3}} (2\pi y) \right) dxdy$$