JDEP 384H: Numerical Methods in Business

Instructor: Thomas Shores
Department of Mathematics

Lecture 23, April 10, 2007
110 Kaufmann Center

Outline

Outline

Outline

Outline

Outline

Outline

Variance Reduction 1: Antithetic Variates

Basic Idea:

To estimate $E[X] = \mu$, select r.v.'s X_1 and X_2 with

- the same distribution as X, but require that they be negatively correlated. Then X and $Y = (X_1 + X_2)/2$ have the same mean μ.
- However, we have $\mathrm{Var}(Y)$ is given by $\frac{\mathrm{Var}(X_1)+\mathrm{Var}(X_2)+2\,\mathrm{Cov}(X_1,X_2)}{4} = \mathrm{Var}(X) + \frac{1}{2}\mathrm{Cov}(X_1, X_2)$.
- Generate paired random samples $\left(X_1^{(i)}, X_2^{(i)}\right)$, $i = 1, \ldots, n$, and obtain pair-averaged samples $Y^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$.
- The resulting sample variance is expected to be be smaller than that of the random sample $X_1^{(i)}$ of X.
- Hope this reduces the variance of the sample.
- Practical pointer: If $X = g(U)$ are supposed to be generated by uniform $U(0,1)$ samples U_i , try $X_1^{(i)} = g(U_i)$ and $X_2^{(i)} = g(1 - U_i)$. IF $g(u)$ is monotone increasing, this works!

Variance Reduction 1: Antithetic Variates

Basic Idea:

To estimate $E[X] = \mu$, select r.v.'s X_1 and X_2 with

- the same distribution as X, but require that they be negatively correlated. Then X and $Y = (X_1 + X_2)/2$ have the same mean μ.
- However, we have $\mathrm{Var}(Y)$ is given by $\frac{\mathrm{Var}(X_1)+\mathrm{Var}(X_2)+2\,\mathrm{Cov}(X_1,X_2)}{4} = \mathrm{Var}(X) + \frac{1}{2}\mathrm{Cov}(X_1, X_2)$.
- Generate paired random samples $\left(X_1^{(i)}, X_2^{(i)}\right)$, $i = 1, \ldots, n$, and obtain pair-averaged samples $Y^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$.
- The resulting sample variance is expected to be be smaller than that of the random sample $X_1^{(i)}$ of X.
- Hope this reduces the variance of the sample.
- Practical pointer: If $X = g(U)$ are supposed to be generated by uniform $U(0,1)$ samples U_i , try $X_1^{(i)} = g(U_i)$ and $X_2^{(i)} = g(1 - U_i)$. IF $g(u)$ is monotone increasing, this works!

Variance Reduction 1: Antithetic Variates

Basic Idea:

To estimate $E[X] = \mu$, select r.v.'s X_1 and X_2 with

- the same distribution as X, but require that they be negatively correlated. Then X and $Y = (X_1 + X_2)/2$ have the same mean μ.
- However, we have $\mathrm{Var}(Y)$ is given by $\frac{\mathrm{Var}(X_1)+\mathrm{Var}(X_2)+2\,\mathrm{Cov}(X_1,X_2)}{4} = \mathrm{Var}(X) + \frac{1}{2}\,\mathrm{Cov}(X_1, X_2)$.
- Generate paired random samples $\left(X_1^{(i)}, X_2^{(i)}\right)$, $i = 1, \ldots, n$, and obtain pair-averaged samples $Y^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$.
- The resulting sample variance is expected to be be smaller than that of the random sample $X_1^{(i)}$ of X.
- Hope this reduces the variance of the sample.
- Practical pointer: If $X = g(U)$ are supposed to be generated by uniform $U(0,1)$ samples U_i , try $X_1^{(i)} = g(U_i)$ and $X_2^{(i)} = g(1 - U_i)$. IF $g(u)$ is monotone increasing, this works!

Variance Reduction 1: Antithetic Variates

Basic Idea:

To estimate $E[X] = \mu$, select r.v.'s X_1 and X_2 with

- the same distribution as X, but require that they be negatively correlated. Then X and $Y = (X_1 + X_2)/2$ have the same mean μ.
- However, we have $\mathrm{Var}(Y)$ is given by $\frac{\mathrm{Var}(X_1)+\mathrm{Var}(X_2)+2\,\mathrm{Cov}(X_1,X_2)}{4} = \mathrm{Var}(X) + \frac{1}{2}\,\mathrm{Cov}(X_1, X_2)$.
- Generate paired random samples $\left(X_1^{(i)}, X_2^{(i)}\right)$, $i = 1, \ldots, n$, and obtain pair-averaged samples $Y^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$.
- The resulting sample variance is expected to be be smaller than that of the random sample $X_1^{(i)}$ of X.
- Hope this reduces the variance of the sample.
- Practical pointer: If $X = g(U)$ are supposed to be generated by uniform $U(0,1)$ samples U_i , try $X_1^{(i)} = g(U_i)$ and $X_2^{(i)} = g(1 - U_i)$. IF $g(u)$ is monotone increasing, this works!

Variance Reduction 1: Antithetic Variates

Basic Idea:

To estimate $E[X] = \mu$, select r.v.'s X_1 and X_2 with

- the same distribution as X, but require that they be negatively correlated. Then X and $Y = (X_1 + X_2)/2$ have the same mean μ.
- However, we have $\mathrm{Var}(Y)$ is given by $\frac{\mathrm{Var}(X_1)+\mathrm{Var}(X_2)+2\,\mathrm{Cov}(X_1,X_2)}{4} = \mathrm{Var}(X) + \frac{1}{2}\,\mathrm{Cov}(X_1, X_2)$.
- Generate paired random samples $\left(X_1^{(i)}, X_2^{(i)}\right)$, $i = 1, \ldots, n$, and obtain pair-averaged samples $Y^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$.
- The resulting sample variance is expected to be be smaller than that of the random sample $X_1^{(i)}$ of X.
- Hope this reduces the variance of the sample.
- Practical pointer: If $X = g(U)$ are supposed to be generated by uniform $U(0,1)$ samples U_i , try $X_1^{(i)} = g(U_i)$ and $X_2^{(i)} = g(1 - U_i)$. IF $g(u)$ is monotone increasing, this works!

Variance Reduction 1: Antithetic Variates

Basic Idea:

To estimate $E[X] = \mu$, select r.v.'s X_1 and X_2 with

- the same distribution as X, but require that they be negatively correlated. Then X and $Y = (X_1 + X_2)/2$ have the same mean μ.
- However, we have $\mathrm{Var}(Y)$ is given by $\frac{\mathrm{Var}(X_1)+\mathrm{Var}(X_2)+2\,\mathrm{Cov}(X_1,X_2)}{4} = \mathrm{Var}(X) + \frac{1}{2}\,\mathrm{Cov}(X_1, X_2)$.
- Generate paired random samples $\left(X_1^{(i)}, X_2^{(i)}\right)$, $i = 1, \ldots, n$, and obtain pair-averaged samples $Y^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$.
- The resulting sample variance is expected to be be smaller than that of the random sample $X_1^{(i)}$ of X.
- Hope this reduces the variance of the sample.
- Practical pointer: If $X = g(U)$ are supposed to be generated by uniform $U(0,1)$ samples U_i , try $X_1^{(i)} = g(U_i)$ and $X_2^{(i)} = g(1 - U_i)$. IF $g(u)$ is monotone increasing, this works!

Variance Reduction 1: Antithetic Variates

Basic Idea:

To estimate $E[X] = \mu$, select r.v.'s X_1 and X_2 with

- the same distribution as X, but require that they be negatively correlated. Then X and $Y = (X_1 + X_2)/2$ have the same mean μ.
- However, we have $\mathrm{Var}(Y)$ is given by $\frac{\mathrm{Var}(X_1)+\mathrm{Var}(X_2)+2\,\mathrm{Cov}(X_1,X_2)}{4} = \mathrm{Var}(X) + \frac{1}{2}\,\mathrm{Cov}(X_1, X_2)$.
- Generate paired random samples $\left(X_1^{(i)}, X_2^{(i)}\right)$, $i = 1, \ldots, n$, and obtain pair-averaged samples $Y^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$.
- The resulting sample variance is expected to be be smaller than that of the random sample $X_1^{(i)}$ of X.
- Hope this reduces the variance of the sample.
- Practical pointer: If $X = g(U)$ are supposed to be generated by uniform $U(0,1)$ samples U_i , try $X_1^{(i)} = g(U_i)$ and $X_2^{(i)} = g(1 - U_i)$. IF $g(u)$ is monotone increasing, this works!

Calculations

Returning to our Monte Carlo integration example, recall that to bound the (absolute) error by γ with the confidence $1-\alpha$, require that $z_{1-\alpha/2}\frac{S(n)}{\sqrt{n}} \leq \gamma$ (assuming normal distribution.) Experiment with this Matlab code.

```
> mu = exp(1)-1
> rand('state',0)
> alpha = 0.05 % 95 percent confidence level
> zalpha = stdn_inv(1-alpha/2)
> n = 200
> U = rand(n,1);
> X1 = exp(U);
> X2 = exp(1-U);
> Xn = 0.5*(X1+X2);
> [smplmu,smplstdv,muci,] = norm_fit(X1,alpha)
> abs(mu-smplmu), gmma = zalpha*sqrt(smplstdv/n)
> [smplmu,smplstdv,muci] = norm_fit(X2,alpha)
> abs(mu-smplmu), gmma = zalpha*sqrt(smplstdv/n)
> [smplmu,smplstdv,muci] = norm_fit(Xn,alpha)
> abs(mu-smplmu), gmma = zalpha*sqrt(smplstdv/n)
```

Variance Reduction 2: Control Variates

Basic Idea:

To estimate $E[X] = \mu$:

- Find a random variable C, with known mean μ_C and form r.v. $X_C = X + \beta(C - \mu)$.
- Have $E[X_C] = E[X] = \mu$.
- Have $\text{Var}([X_C]) = \text{Var}(X) + \beta^2 \text{Var}(C) + 2\beta \text{Cov}(X, C)$.
- So if $2\beta \text{Cov}(X, C) + \beta^2 \text{Var}(C) < 0$, we get reduction with optimum at $\beta = \beta^* = -\text{Cov}(Y, C) / \text{Var}(C)$ (why?) , with variance $\left(1 - \rho^2(X, C)\right) \text{Var}(X)$. In practice, we estimate β^* experimentally.

Variance Reduction 2: Control Variates

Basic Idea:

To estimate $E[X] = \mu$:

- Find a random variable C, with known mean μ_C and form r.v. $X_C = X + \beta(C - \mu)$.
- Have $E[X_C] = E[X] = \mu$.
- Have $\text{Var}([X_C]) = \text{Var}(X) + \beta^2 \text{Var}(C) + 2\beta \text{Cov}(X, C)$.
- So if $2\beta \text{Cov}(X, C) + \beta^2 \text{Var}(C) < 0$, we get reduction with optimum at $\beta = \beta^* = -\text{Cov}(Y, C) / \text{Var}(C)$ (why?) , with variance $\left(1 - \rho^2(X, C)\right) \text{Var}(X)$. In practice, we estimate β^* experimentally.

Variance Reduction 2: Control Variates

Basic Idea:

To estimate $E[X] = \mu$:

- Find a random variable C, with known mean μ_C and form r.v. $X_C = X + \beta(C - \mu)$.
- Have $E[X_C] = E[X] = \mu$.
- Have $\text{Var}([X_C]) = \text{Var}(X) + \beta^2 \text{Var}(C) + 2\beta \text{Cov}(X, C)$.
- So if $2\beta \text{Cov}(X, C) + \beta^2 \text{Var}(C) < 0$, we get reduction with optimum at $\beta = \beta^* = -\text{Cov}(Y, C) / \text{Var}(C)$ (why?) , with variance $\left(1 - \rho^2(X, C)\right) \text{Var}(X)$. In practice, we estimate β^* experimentally.

Variance Reduction 2: Control Variates

Basic Idea:

To estimate $E[X] = \mu$:

- Find a random variable C, with known mean μ_C and form r.v. $X_C = X + \beta(C - \mu)$.
- Have $E[X_C] = E[X] = \mu$.
- Have $\text{Var}([X_C]) = \text{Var}(X) + \beta^2 \text{Var}(C) + 2\beta \text{Cov}(X, C)$.
- So if $2\beta \text{Cov}(X, C) + \beta^2 \text{Var}(C) < 0$, we get reduction with optimum at $\beta = \beta^* = -\text{Cov}(Y, C) / \text{Var}(C)$ (why?) , with variance $\left(1 - \rho^2(X, C)\right) \text{Var}(X)$. In practice, we estimate β^* experimentally.

Variance Reduction 2: Control Variates

Basic Idea:

To estimate $E[X] = \mu$:

- Find a random variable C, with known mean μ_C and form r.v. $X_C = X + \beta(C - \mu)$.
- Have $E[X_C] = E[X] = \mu$.
- Have $\text{Var}([X_C]) = \text{Var}(X) + \beta^2 \text{Var}(C) + 2\beta \text{Cov}(X, C)$.
- So if $2\beta \text{Cov}(X, C) + \beta^2 \text{Var}(C) < 0$, we get reduction with optimum at $\beta = \beta^* = -\text{Cov}(Y, C) / \text{Var}(C)$ (why?) , with variance $\left(1 - \rho^2(X, C)\right) \text{Var}(X)$. In practice, we estimate β^* experimentally.

Calculations

Returning to our Monte Carlo integration example, find a bound γ for the (absolute) error by γ with the confidence $1 - \alpha$. This Matlab code uses a linear approximation as control variate.

```
> mu = exp(1)-1
> rand('state',0)
> alpha = 0.05 % 95 percent confidence level
> zalpha = stdn_inv(1 - alpha/2)
> n = 100
> Un = rand(n,1);
> Xn = exp(Un);
> Cn = 1+(exp(1)-1)*Un; % Control variate
> muC = 1+(exp(1)-1)*0.5 % Expected value of C
> S = -cov([Cn,Xn]); % get covariance matrix
> bta = S(2,1)/S(2,2) % guess at optimum beta
> XC = Xn + bta*(Cn - muC);
> [smplmu,smplstdv,muci] = norm_fit(Xn,alpha)
> abs(mu-smplmu), gmma = zalpha*sqrt(smplstdv/n)
> [smplmu,smplstdv,muci] = norm_fit(XC,alpha)
> abs(mu-smplmu), gmma = zalpha*sqrt(smplstdv/n)
```

Outline

Path Generation (Asset Dynamics)

Basic Idea:

Given an Ito stochastic differential equation $dS_t = a(S_t, t)\,dt + b(S_t, t)\,dW_t$, how do we model a path of the underlying stochastic process $S(t)$?

- Simple discretization might lead to what we used in Exercise 3.5 for geometric Brownian motion $dS = \mu S\,\delta t + \sigma S\,dX$: $\Delta S = S_{k+1} - S_k \approx \mu S_k\,\delta t + \sigma S_k\,dX$, where $S_k = S(t_k)$.
- But this makes the random variable S_{k+1} normally distributed, given S_k, which is wrong! (Why?)
- Reason: we saw in the ProbStatLectures section on stochastic integrals that we can actually solve for S and obtain $S(t) = S(0)\,e^{\nu t + \sigma\sqrt{t}\cdot z}$, so that with a little work we get $S_{k+1} = S_k\,e^{\nu\,\delta t + \sigma\sqrt{\delta t}\cdot z}$, and S_{k+1} is lognormally distributed, given S_k. This gives a better strategy for simulating paths.

Path Generation (Asset Dynamics)

Basic Idea:

Given an Ito stochastic differential equation $dS_t = a(S_t, t)\,dt + b(S_t, t)\,dW_t$, how do we model a path of the underlying stochastic process $S(t)$?

- Simple discretization might lead to what we used in Exercise 3.5 for geometric Brownian motion $dS = \mu S\,\delta t + \sigma S\,dX$: $\Delta S = S_{k+1} - S_k \approx \mu S_k\,\delta t + \sigma S_k\,dX$, where $S_k = S(t_k)$.
- But this makes the random variable S_{k+1} normally distributed, given S_k, which is wrong! (Why?)
- Reason: we saw in the ProbStatLectures section on stochastic integrals that we can actually solve for S and obtain $S(t) = S(0)\,e^{\nu t + \sigma\sqrt{t}\cdot z}$, so that with a little work we get $S_{k+1} = S_k e^{\nu\,\delta t + \sigma\sqrt{\delta t}\cdot z}$, and S_{k+1} is lognormally distributed, given S_k. This gives a better strategy for simulating paths.

Path Generation (Asset Dynamics)

Basic Idea:

Given an Ito stochastic differential equation $dS_t = a(S_t, t)\,dt + b(S_t, t)\,dW_t$, how do we model a path of the underlying stochastic process $S(t)$?

- Simple discretization might lead to what we used in Exercise 3.5 for geometric Brownian motion $dS = \mu S\,\delta t + \sigma S\,dX$: $\Delta S = S_{k+1} - S_k \approx \mu S_k\,\delta t + \sigma S_k\,dX$, where $S_k = S(t_k)$.
- But this makes the random variable S_{k+1} normally distributed, given S_k, which is wrong! (Why?)
- Reason: we saw in the ProbStatLectures section on stochastic integrals that we can actually solve for S and obtain $S(t) = S(0)\,e^{\nu t + \sigma\sqrt{t}\cdot z}$, so that with a little work we get $S_{k+1} = S_k e^{\nu\,\delta t + \sigma\sqrt{\delta t}\cdot z}$, and S_{k+1} is lognormally distributed, given S_k. This gives a better strategy for simulating paths.

Path Generation (Asset Dynamics)

Basic Idea:

Given an Ito stochastic differential equation $dS_t = a(S_t, t)\,dt + b(S_t, t)\,dW_t$, how do we model a path of the underlying stochastic process $S(t)$?

- Simple discretization might lead to what we used in Exercise 3.5 for geometric Brownian motion $dS = \mu S\,\delta t + \sigma S\,dX$: $\Delta S = S_{k+1} - S_k \approx \mu S_k\,\delta t + \sigma S_k\,dX$, where $S_k = S(t_k)$.
- But this makes the random variable S_{k+1} normally distributed, given S_k, which is wrong! (Why?)
- Reason: we saw in the ProbStatLectures section on stochastic integrals that we can actually solve for S and obtain $S(t) = S(0)\,e^{\nu t + \sigma\sqrt{t}\cdot z}$, so that with a little work we get $S_{k+1} = S_k\,e^{\nu\,\delta t + \sigma\sqrt{\delta t}\cdot z}$, and S_{k+1} is lognormally distributed, given S_k. This gives a better strategy for simulating paths.

Some Path Calculations

```
> mu = 0.1, sigma = 0.3, S0 = 100
> randn('state',0)
> nsteps = 52, T=1, dt = T/nsteps, nreps = 100
> S = zeros(nsteps+1,1); S(1) = S0;
> S2 = zeros(nreps,1);
> truemean = S(1)*exp(mu*T) % according to p.  99
> truestdv = sqrt(exp(2*(log(S(1))+(mu-sigma^2/2)*T) + ...
> sigma*sqrt(T))*(exp(sigma^2*T)-1)) % according to p.
632
> for j = 1:nreps
> for k = 1:nsteps, S(k+1) = S(k)*(1 + mu*dt + ...
> sigma*sqrt(dt)*randn()); end
> S2(j) = S(25);
> end % store up results at 1 year
> [smplmu,smplstdv,muci] = norm_fit(S2,alpha)
> A = AssetPath(S0,mu,sigma,T,nsteps,nreps);
> [smplmu,smplstdv,muci] = norm_fit(A(:,nsteps+1),alpha)
```

European Call with Simple Monte Carlo

Basic Idea:

- If r is the risk-free interest rate and option price f_0 at time $t = 0$ is risk-free with price f_T at time $t = T$, then the value of f_0 should be the discounted expected payoff $f = e^{-rT} E[f_T]$ under a risk-neutral probability measure.
- Of course, f_T is a r.v. But the drift for this asset should be the risk-free rate r. So all we have to do is average the payoffs over various stock price paths to time T, then discount the average to obtain an approximation for f_0.
- For example, with a European call, the payoff curve gives

$$f_T = \max\left\{0, S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z} - K\right\}$$

where K is the strike price. So we need the final value of random walks of stock prices $S_T = S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z}$.

European Call with Simple Monte Carlo

Basic Idea:

- If r is the risk-free interest rate and option price f_0 at time $t = 0$ is risk-free with price f_T at time $t = T$, then the value of f_0 should be the discounted expected payoff $f = e^{-rT} E[f_T]$ under a risk-neutral probability measure.
- Of course, f_T is a r.v. But the drift for this asset should be the risk-free rate r. So all we have to do is average the payoffs over various stock price paths to time T, then discount the average to obtain an approximation for f_0.
- For example, with a European call, the payoff curve gives

$$f_T = \max\left\{0, S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z} - K\right\}$$

where K is the strike price. So we need the final value of random walks of stock prices $S_T = S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z}$.

European Call with Simple Monte Carlo

Basic Idea:

- If r is the risk-free interest rate and option price f_0 at time $t = 0$ is risk-free with price f_T at time $t = T$, then the value of f_0 should be the discounted expected payoff $f = e^{-rT} E[f_T]$ under a risk-neutral probability measure.
- Of course, f_T is a r.v. But the drift for this asset should be the risk-free rate r. So all we have to do is average the payoffs over various stock price paths to time T, then discount the average to obtain an approximation for f_0.
- For example, with a European call, the payoff curve gives

$$f_T = \max\left\{0, S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z} - K\right\}$$

where K is the strike price. So we need the final value of random walks of stock prices $S_T = S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z}$.

European Call with Simple Monte Carlo

Basic Idea:

- If r is the risk-free interest rate and option price f_0 at time $t = 0$ is risk-free with price f_T at time $t = T$, then the value of f_0 should be the discounted expected payoff $f = e^{-rT} E[f_T]$ under a risk-neutral probability measure.
- Of course, f_T is a r.v. But the drift for this asset should be the risk-free rate r. So all we have to do is average the payoffs over various stock price paths to time T, then discount the average to obtain an approximation for f_0.
- For example, with a European call, the payoff curve gives

$$f_T = \max\left\{0, S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z} - K\right\}$$

where K is the strike price. So we need the final value of random walks of stock prices $S_T = S_0 e^{\left(r-\sigma^2/2\right)T+\sigma\sqrt{T}z}$.

Example Calculations

Example Calculations:

Use Monte Carlo and antithetic variates generated by z and $-z$ to estimate the value of a European call with same data as in previous example and strike price of $K = 110$. Take risk-free interest rate to be $r = 0.06$.

```
> alpha = 0.05, randn('state',0)
> sigma = 0.3, S0 = 100, r = 0.06, K = 110
> nsteps = 52, T=1, dt = T/nsteps, nreps = 100
> nuT = (mu-0.5*sigma^2)*T;
> siT = sigma*sqrt(T);
> Veps = randn(nreps,1);
> payoff1 = max(0,S0*exp(nuT+siT*Veps) - K);
> payoff2 = max(0,S0*exp(nuT+siT*(-Veps)) - K);
> prices = exp(-r*T)*0.5*(payoff1 + payoff2);
> trueprice = bseurcall(S0,K,r,T,0,sigma,0)
> [price, V, CI] = norm_fit(prices) % compare
> [price, V, CI] = norm_fit(exp(-r*T)*payoff1) % compare
```