Numerical Methods II

Prof. Mike Giles

mike.giles@maths.ox.ac.uk

Oxford University Mathematical Institute

Variance Reduction

Monte Carlo starts as a very simple method; much of the complexity in practice comes from trying to reduce the variance, to reduce the number of samples that have to be simulated to achieve a given accuracy.

- antithetic variables (lecture 3)
- control variates (lecture 3)
- importance sampling (lecture 3)
- stratified sampling
- Latin hypercube
- quasi-Monte Carlo (lecture 5)

Stratified Sampling

The key idea is to achieve a more regular sampling of the most "important" dimension in the uncertainty.

Start by considering a one-dimensional problem:

$$I = \int_0^1 f(U)\, \mathrm{d}U.$$

Instead of taking N samples, drawn from uniform distribution on $[0,1]$, instead break the interval into M strata of equal width and take L samples from each.

Stratified Sampling

Define U_{ij} to be the value of i^{th} sample from strata j,

$$\begin{aligned}\overline{F}_j &= L^{-1} \sum_i f(U_{ij}) = \text{average from strata } j, \\ \overline{F} &= M^{-1} \sum_j \overline{F}_j = \text{overall average}\end{aligned}$$

and similarly let

$$\begin{aligned}\mu_j &= \mathbb{E}[f(U) \,|\, U \in \text{strata } j], \\ \sigma_j^2 &= \mathbb{V}[f(U) \,|\, U \in \text{strata } j], \\ \mu &= \mathbb{E}[f], \\ \sigma^2 &= \mathbb{V}[f].\end{aligned}$$

Stratified Sampling

With stratified sampling,

$$\mathbb{E}[\overline{F}] = M^{-1} \sum_j \mathbb{E}[\overline{F}_j] = M^{-1} \sum_j \mu_j = \mu$$

so it is unbiased.

The variance is

$$\begin{aligned} \mathbb{V}[\overline{F}] = M^{-2} \sum_j \mathbb{V}[\overline{F}_j] &= M^{-2} L^{-1} \sum_j \sigma_j^2 \\ &= N^{-1} M^{-1} \sum_j \sigma_j^2 \end{aligned}$$

where $N = LM$ is the total number of samples.

Stratified Sampling

Without stratified sampling, $\mathbb{V}[\overline{F}] = N^{-1}\sigma^2$ with

$$\begin{aligned}
\sigma^2 &= \mathbb{E}[f^2] - \mu^2 \\
&= M^{-1} \sum_j \mathbb{E}[f(U)^2 \,|\, U \in \text{ strata } j] \; - \; \mu^2 \\
&= M^{-1} \sum_j (\mu_j^2 + \sigma_j^2) \; - \; \mu^2 \\
&= M^{-1} \sum_j \left((\mu_j - \mu)^2 + \sigma_j^2 \right) \\
&\geq M^{-1} \sum_j \sigma_j^2
\end{aligned}$$

Thus stratified sampling reduces the variance.

Stratified Sampling

How do we use this for MC simulations?

For a one-dimensional application:

- Break $[0,1]$ into M strata
- For each stratum, take L samples U with uniform probability distribution
- Define $X=\Phi^{-1}(U)$ and use this for $W(T)$
- Compute average within each stratum, and overall average.

Stratified Sampling

Test case: European call

$r=0.05,\ \sigma=0.5,\ T=1,\ S_0=110,\ K=100,\ N=10^4$ samples

M	L	MC error bound
1	10000	1.39
10	1000	0.55
100	100	0.21
1000	10	0.07

Application

MATLAB code:

```
for M = [1 10 100 1000]
  L = N/M;   ave=0; var=0;
  for m = 1:M
    U = (m-1+rand(1,L))/M;
    Y = norminv(U);
    S = S0*exp((r-sig^2/2)*T + sig*sqrt(T)*Y);
    F = exp(-r*T)*max(0,S-K);
    ave1 =  sum(F)/L;
    var1 = (sum(F.^2)/L - ave1^2)/(L-1);
    ave = ave + ave1/M;
    var = var + var1/M^2;
  end
end
```

Stratified Sampling

Sub-dividing a stratum always reduces the variance, so the optimum choice is to use 1 sample per stratum

However, need multiple samples in each stratum to estimate the variance and obtain a confidence interval.

This tradeoff between efficiency and confidence/reliability happens in other contexts – e.g. QMC in next lecture

Despite this, interesting to analyse what happens with 1 sample per stratum.

Stratified Sampling

For j^{th} stratum, if $f(U)$ is differentiable then

$$f(U) \approx f(U_j) + f'(U_j)\,(U - U_j)$$

where U_j is midpoint of stratum, and hence

$$\begin{aligned} \mathbb{V}[f(U)\,|\,U \in \text{stratum } j] &\approx \left(f'(U_j)\right)^2 \mathbb{V}[U - U_j\,|\,U \in \text{stratum } j] \\ &= \frac{1}{12N^2}\left(f'(U_j)\right)^2 \end{aligned}$$

since the stratum has width $\dfrac{1}{N}$.

Stratified Sampling

Summing all of the variances (due to independence) and dividing by N^2 (due to averaging) the variance of the average over all strata is then

$$\frac{1}{12N^4} \sum_j \left(f'(U_j)\right)^2 \approx \frac{1}{12N^3} \int \left(f'(U)\right)^2 \, \mathrm{d}U$$

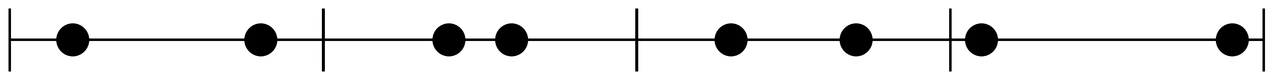
so the r.m.s. error is $O(N^{-3/2})$, provided $f'(U)$ is square integrable.

This is much better than the usual $O(N^{-1/2})$ r.m.s. error – shows how powerful stratified sampling can be.

Stratified Sampling

This analysis suggest another improvement – since the function is almost linear within each stratum, it is a near-ideal situation in which to use antithetic variables.

For each sample point, take a second which is mirror image within that stratum.



Stratified Sampling

So far, have assumed an equal number of samples in each stratum. If we now take L_j samples in stratum j, the overall variance is

$$M^{-2} \sum_j L_j^{-1} \sigma_j^2 .$$

Using a Lagrange multiplier to minimise this while holding fixed the total number of samples

$$\sum_j L_j$$

leads to the optimal choice

$$L_j \;\propto\; \sigma_j$$

Stratified Sampling

Hence, it is best to use more samples for strata with high variability.

σ_j is usually not known beforehand, but one can use a small number of pilot runs to estimate σ_j and choose L_j.

Important to throw away the results from the pilot runs and not use them in the final estimate – otherwise can introduce a bias into the results.

Stratified Sampling

For a multivariate application, one approach is to:

- Break $[0,1]$ into M strata
- For each stratum, take L samples U with uniform probability distribution
- Define $X_1 = \Phi^{-1}(U)$
- Simulate other elements of X using standard Normal random number generation
- Multiply X by matrix C to get $Y = CX$ with desired covariance
- Compute average within each stratum, and overall average

Stratified Sampling

The effectiveness of this depends on a good choice of C.

Ideally, want the function $f(Y)$ to depend solely on the value of X_1 so it reduces to a one-dimensional application.

Not easy in practice, requires good insight or a complex optimisation, so instead generalise stratified sampling approach to multiple dimensions.

Stratified Sampling

For a d-dimensional application, can split each dimension of the $[0,1]^d$ hypercube into M strata producing M^d sub-cubes.

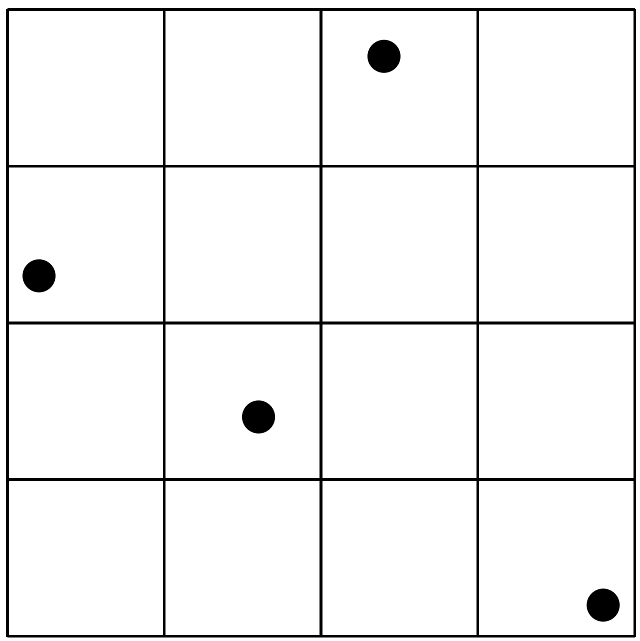
One generalisation of stratified sampling is to generate L points in each of these hypercubes

However, the total number of points is $L\,M^d$ which for large d would force M to be very small in practice.

Instead, use a method called Latin Hypercube sampling

Latin Hypercube

Generate M points, dimension-by-dimension, using 1D stratified sampling with 1 value per stratum, assigning them randomly to the M points to give precisely one point in each stratum



Latin Hypercube

This gives one set of M points, with average

$$\overline{f} = M^{-1} \sum_{m=1}^{M} f(U_m)$$

Since each of the points U_m is uniformly distributed over the hypercube,

$$\mathbb{E}[\overline{f}] = \mathbb{E}[f]$$

The fact that the points are not independently generated does not affect the expectation, only the (reduced) variance

Latin Hypercube

We now take L independently-generated set of points, each giving an average $\overline{f}_l$.

Averaging these

$$L^{-1} \sum_{l=1}^{L} \overline{f}_l$$

gives an unbiased estimate for $\mathbb{E}[f]$, and the empirical variance for $\overline{f}_l$ gives a confidence interval in the usual way.

Latin Hypercube

Note: in the special case in which the function $f(U)$ is a sum of one-dimensional functions:

$$f(U) = \sum_i f_i(U_i)$$

where U_i is the i^{th} component of U, then Latin Hypercube sampling reduces to 1D stratified sampling in each dimension.

In this case, potential for very large variance reduction by using large sample size M.

Much harder to analyse in general case.

Final comments

- Stratified sampling is very effective in 1D, but not so clear how to use it in multiple dimensions
- Latin Hypercube is one generalisation – very effective when function can be decomposed into a sum of 1D functions
- Hard to predict which variance reduction approach will be most effective
- Advice: when facing a new class of applications, try each one, and don't forget you can sometimes combine different techniques
  (e.g. stratified sampling with antithetic variables, or Latin Hypercube with importance sampling)