Numerical Methods II
M. Giles

## Practical 1

1. (a) Generate $10^6$ uniform random variables using `rand`, then convert them into $10^6$ unit Normal variables using `norminv`. Check that they have the expected mean and variance.

   (b) Given a covariance matrix

   $$\Sigma = \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix}$$

   perform a Cholesky factorisation (using Matlab function `chol`) to obtain a lower-triangular matrix $L$ such that

   $$\Sigma = L\, L^T$$

   Use this matrix $L$ to convert $2 \times 10^6$ independent unit Normals (generated using Matlab function `randn`) into $10^6$ pairs of Normals with the desired covariance. Check that they have the expected mean and covariance.

   (c) Repeat the previous item using the PCA factorisation of $\Sigma$ (using Matlab function `eig`).

2. Let $U$ be uniformly distributed on $[0,1]$. You are to use Monte Carlo simulation to estimate the value of

   $$\overline{f} = \mathbb{E}[f(U)] = \int_0^1 f(U)\ \mathrm{d}U$$

   where

   $$f(x) = x\ \cos \pi x.$$

   (a) Calculate analytically the exact value for $\overline{f}$ and

   $$\sigma^2 = \mathbb{E}[(f(U) - \overline{f})^2] = \int_0^1 (f(U) - \overline{f})^2\ \mathrm{d}U$$

   (b) Using the Matlab `rand` function, write a Matlab program which computes

   $$Y_m = N^{-1} \sum_{n=1}^{N} f(U^{(m,n)})$$

   for 1000 different sets of 1000 independent random variables $U^{(m,n)}$.

(c) Sort the $Y_m$ into ascending order, and then plot $C_m = (m - 1/2)/1000$ versus $Y_m$ – this is the numerical cumulative distribution function.

Superimpose on the same plot the cumulative distribution function you would expect from the Central Limit Theorem (using the `normcdf` or `norminv` functions) and comment on your results.

You may like to experiment by trying larger or smaller sets of points to improve your understanding of the asymptotic behaviour described by the CLT.

(d) Modify your code to use a single set of $10^6$ random numbers, and plot

$$Y_N = N^{-1} \sum_{n=1}^{N} f(U^{(n)})$$

versus $N$ for $N = 10^3 - 10^6$. This should demonstrate the convergence to the true value predicted by the Strong Law of Large Numbers.

For each $N$, also compute an unbiased estimate for the variance $\sigma^2$ (in lecture 3 I'll show some Matlab code for doing this conveniently using the `cumsum` function) and hence add to the plot upper and lower confidence bounds based on 3 standard deviations of the variation in the mean.

Add a line corresponding to the true value. Does this lie inside the bounds?

3. Repeat Question 2 for a European call option in which the final value of the underlying is

$$S(T) = S(0)\ \exp\left((r - \tfrac{1}{2}\sigma^2)T + \sigma W(T)\right)$$

where

$$W(T) = \sqrt{T}\ X = \sqrt{T}\ \Phi^{-1}(U)$$

with $X$ being a unit Normal (produced by `randn`) or $U$ a uniform $(0, 1)$ random variable (produced by `rand`).

The payoff function is

$$f(S) = \exp(-rT)\ (S(T) - K)^+$$

and the constants are $r = 0.05, \sigma = 0.2, S(0) = 100, K = 100$.

The analytic value is given by the routine `european_call` available from my webpage; read its header to see how to call it.

There is no need to compute the analytic variance in part a); just use the unbiased estimator when plotting the CLT prediction in part c).