Computing Practical 1 - Monte Carlo integration

In this practical we are going to calculate the area of a 2-D shape using Monte Carlo integration. For a shape Ω that is contained within the square,$[-1,1]\times[1,1]$ it is clear that

$$\text{Area of } \Omega \ = I[f] = \int_{-1}^{1}\int_{-1}^{1} f(x,y)dxdy$$

where $f(x,y)$ is the indicator function

$$f(x,y) = \begin{cases} 1 & \text{if } (x,y)\in\Omega \\ 0 & \text{otherwise} \end{cases} .$$

Exercise 1: Produce a plot in matlab of the indicator function for the unit circle.

a. Set up a grid for (x,y) using the meshgrid command

```
>> [x,y]=meshgrid(-1:0.1:1, -1:0.1:1);
```

b. Compute the values of the function f on the grid

```
>> f=zeros(size(x)); I=find((x.^2 + y.^2)<=1); f(I)=ones(size(I));
```

c. Plot f

```
>> pcolor(x,y,f); colorbar;
```

To use Monte Carlo integration to compute $I[f]$ we first have to interpret the integral as a probabilistic quantity. Note that if we take $p(x,y)$ as the 2-D uniform density on the square $[-1,1]\times[-1,1]$ then $p(x,y)=1/4$ for (x,y) inside the square. Thus,

$$I[f] = 4\int_{-1}^{1}\int_{-1}^{1} f(x,y)p(x,y)dxdy.$$

Exercise 2: Compute the Monte Carlo approximation to the integral for $N = 50$

a. Generate a random sample from the uniform [0,1] distribution.

```
>> XY= rand(50, 2);
```

b. Rescale the sample to be from the uniform [-1, 1] distribution and allocate X and Y arrays

```
>> XY2 = -1.0+2.0*XY; X=XY2(:,1); Y=XY2(:,2);
```

c. Plot the sample and superimpose the unit circle

```
>> plot(X, Y, 'o'); grid on; hold on; t=0:0.01:  2*pi; plot(cos(t), sin(t)); axis equal;
```

d. Compute the Monte Carlo estimate of the integral and compare with the value of pi.

```
>> Idx=find((X.^2 + Y.^2) <= 1); FN=zeros(50,1); FN(Idx)=ones(size(Idx));
INFN=4.0e0*sum(FN)/50.0e0
>> err=pi-INFN
```

Exercise 3: Repeat Exercise 2 for a range of values of N. Tabulate your results. Produce a plot to verify that the RMSE goes like $O(N^{-1/2})$.

Exercise 4: By repeating Exercise 2 for N=50 and tabulating your answers for a number of realizations, verify that the Monte Carlo quadrature rule is unbiased i.e., that $E[I_N[f]] = I[f]$.