Monte Carlo in Action

Prof. Americo Cunha Jr

americocunha.org

@AmericoCunhaJr

@AmericoCunhaJr @AmericoCunhaJr

Unit square sampling

Sample at random an unit square according to the distributions:

- Uniform
- Normal
- Gamma
- Chi-squared

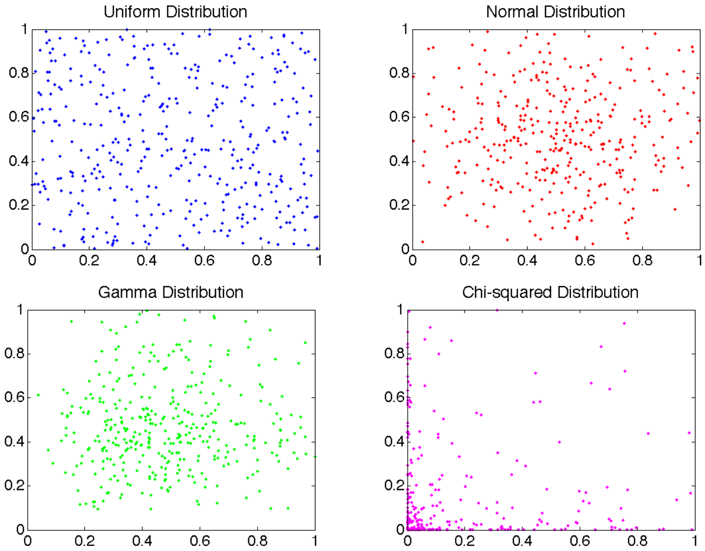
Matlab code for square sampling (1/2)

```
clc; clear; close all;

Ns = 1000; xmin = 0.0; xmax = 1.0; ymin = 0.0; ymax = 1.0;

mu = 0.5; coefvar = 0.5; sigma = mu*coefvar;

% uniform sample
Xu = rand(Ns); Yu = rand(Ns);

% normal sample
Xn = mu + sigma*randn(Ns); Yn = mu + sigma*randn(Ns);

% gamma sample
Xg = gamrnd(1/coefvar^2,mu*coefvar^2,Ns);
Yg = gamrnd(1/coefvar^2,mu*coefvar^2,Ns);

% Chi-squared sample
Xc2 = chi2rnd(mu,Ns); Yc2 = chi2rnd(mu,Ns);
```

Matlab code for square sampling (2/2)

```
figure(1)
plot(Xu,Yu,'.b');
axis([xmin xmax ymin ymax]);
title(' Uniform Distribution','FontSize',20)

figure(2)
plot(Xn,Yn,'.r');
axis([xmin xmax ymin ymax]);
title(' Normal Distribution','FontSize',20)

figure(3)
plot(Xg,Yg,'.g');
axis([xmin xmax ymin ymax]);
title(' Gamma Distribution','FontSize',20)

figure(4)
plot(Xc2,Yc2,'.m');
axis([xmin xmax ymin ymax]);
title(' Chi-squared Distribution','FontSize',20)
```

Different samplings of an unit square



MC for integration

Consider the integral

$$\begin{aligned} \alpha &= \int_0^{2\pi} \sin^2(x)\, dx \\ &= \int_0^{2\pi} 2\pi \, \sin^2(x) \, \frac{1}{2\pi}\, dx \\ &= \int_{\mathbb{R}} h(y)\, p(y) dy \end{aligned}$$

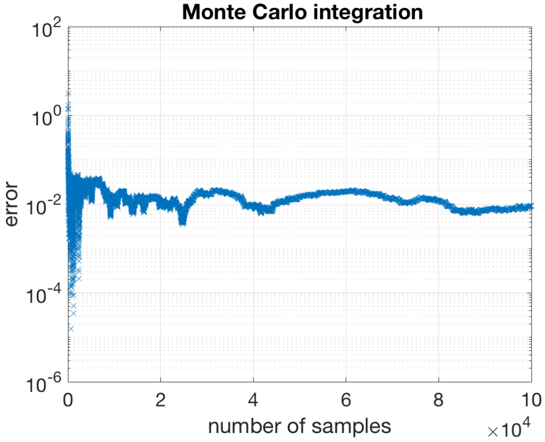
where

- $h(y) = 2\pi \sin^2(y)$
- $p(y) = \dfrac{1}{2\pi} \mathbb{1}_{(0,2\pi)}(y)$ i.e., $Y \sim \mathcal{U}(0, 2\pi)$.

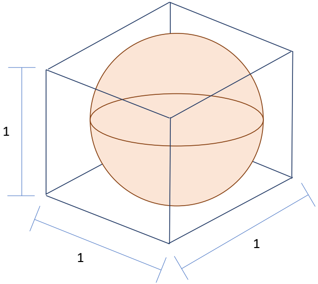
MC for integration

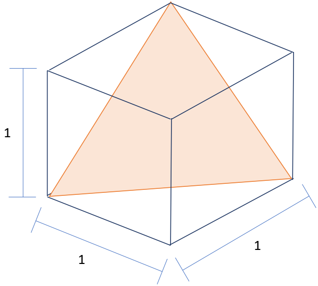
```
clc; clear; close all


Ns = 1e5;

ymin = 0.0; ymax = 2*pi;

Y  = ymin + (ymax-ymin)*rand(Ns,1);
hY = (ymax-ymin)*(sin(Y).^2);

alpha_hat = cumsum(hY)./(1:Ns)';
error     = abs(pi-alpha_hat);

semilogy(1:Ns,error,'x')
title(' Monte Carlo integration','FontSize',20)
set(gca,'fontsize',18)
xlabel('number of samples')
ylabel('error')
grid
```

MC for integration



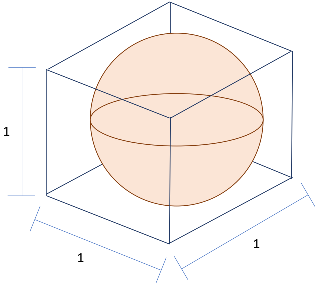
MC for estimation of statistics

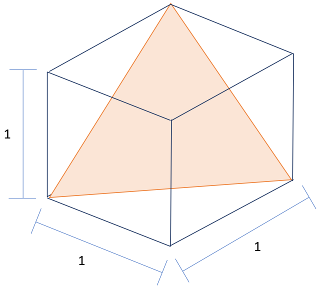




(X, Y, Z) is a random vector defined on the unit cube in $\mathbb{R}^3$

MC for estimation of statistics

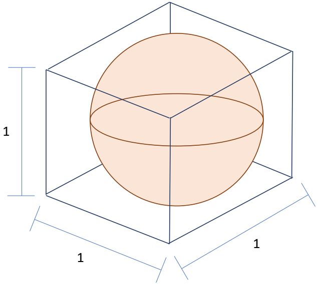


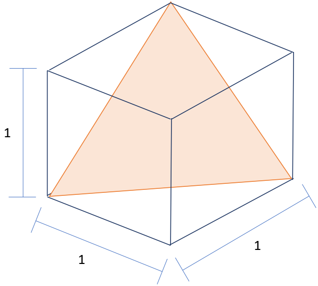


(X, Y, Z) is a random vector defined on the unit cube in $\mathbb{R}^3$

$$\mathcal{P}\left\{(X, Y, Z) \in x^2 + y^2 + z^3 \leq 1/4\right\} = \frac{1}{6}\,\pi$$

MC for estimation of statistics





(X, Y, Z) is a random vector defined on the unit cube in $\mathbb{R}^3$

$$\mathcal{P}\left\{(X, Y, Z) \in x^2 + y^2 + z^3 \leq 1/4\right\} = \frac{1}{6}\,\pi$$

$$\mathcal{P}\left\{(X, Y, Z) \in x + y + z = 1\right\} = 0$$

MC for estimation of statistics (sphere)

```
clear; close all; clc

% sample size
Ns = 100000;

% draw the samples
X = -0.5 + rand(Ns,3);

% distance from the origin for each sample
R = sqrt(sum(X.^2,2));

% count the number of hits
count = sum(R <= 1/2)

% compute the estimated volume
P_sphere = count/Ns
P_true   = pi/6
rel_error = (abs(P_sphere-P_true)/abs(P_true))*100
```

MC for estimation of statistics (sphere)

```
count =

       52241


P_sphere =

    0.5224


P_true =

    0.5236


rel_error =

    0.2270
```

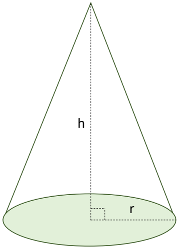
MC for estimation of statistics (plane)

```
clear; close all; clc

% sample size
Ns = 100000;

% draw the samples
X = -0.5 + rand(Ns,3);

% sum of the coordinates
SumCoord = sum(X,2);

% count the number of hits
count = sum(SumCoord == 1)

% compute the estimated volume
P_plane = count/Ns
P_true  = 0
```

MC for estimation of statistics (plane)

```
count =

     0


P_plane =

     0


P_true =

     0
```

MC for uncertainty propagation



- Parameters support:

$$(r, h) \in [0,1] \times (0, \infty)$$

- Volume:

$$v(r,h) = \frac{1}{3}\,\pi\, r^2\, h$$

- Input joint-distribution:

  $R \sim Beta(a,b)$ and $H \sim Exp(1/\mu_H)$ independent

$$p_{RH}\,(r,h) = \mathbb{1}_{[0,1]}(r)\,\frac{\Gamma\,(a+b)}{\Gamma\,(a)\,\Gamma\,(b)}\,r^{a-1}\,(1-r)^{b-1} \times \mathbb{1}_{(0,\infty)}(h)\,\frac{\exp\left(-h/\mu_H\right)}{\mu_H}$$

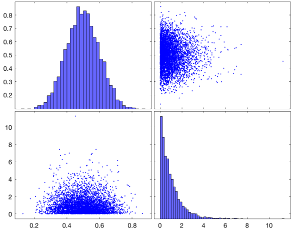
- Output distribution:

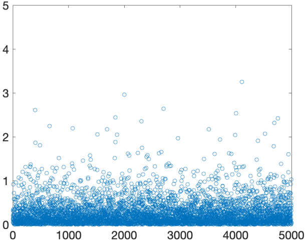
$$p_V\,(v) = \;???$$

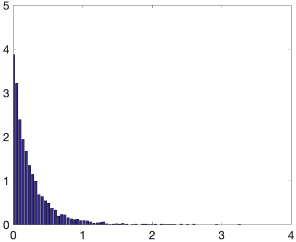
MC for uncertainty propagation

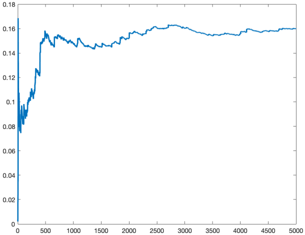
```
clear; close all; clc
% random number generator (fix the seed for reproducibility)
rng_stream = RandStream('mt19937ar','Seed',30081984);
RandStream.setGlobalStream(rng_stream);
Ns = 5000;

muR = 0.5; sigmaR = 0.1; nuR = muR*(1-muR)/(sigmaR^2)-1;
a = muR*nuR; b = (1-muR)*nuR; muH = 1.0;
R = betarnd(a,b,Ns,1); H = exprnd(muH,Ns,1);

V_cone      = (1/3)*pi*R.^2.*H;
[bins,freq] = randvar_pdf(V_cone,round(sqrt(Ns)));
alpha_2     = cumsum(V_cone.^2)'./(1:Ns);

figure(1)
gplotmatrix([R,H]);

figure(2)
plot(V_cone,'o'); ylim([0 5])

figure(3)
bar(bins,freq);
xlim([0 4]); ylim([0 5])

figure(4)
plot(1:Ns,alpha_2,'LineWidth',2)
```

MC for uncertainty propagation







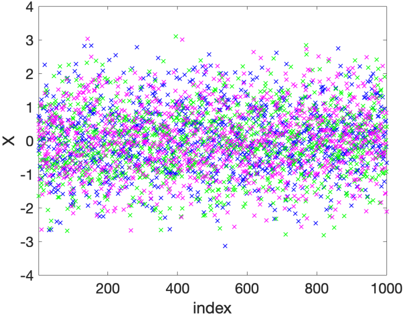


MC convergence analysis (finite mean)

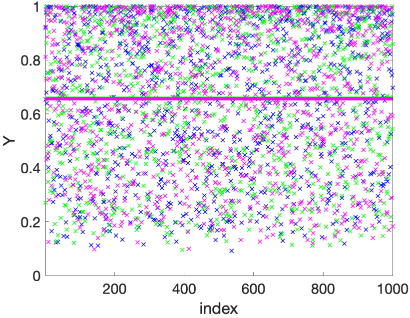
```
clc; clear; close all

Ns = 10^3;
X1 = randn(Ns,1); X2 = randn(Ns,1); X3 = randn(Ns,1);
Y1 = 1./(1+X1.^2); Y2 = 1./(1+X2.^2); Y3 = 1./(1+X3.^2);

meanY1 = mean(Y1)
meanY2 = mean(Y2)
meanY3 = mean(Y3)

figure(1)
plot(1:Ns,X1,'xb',1:Ns,X2,'xg',1:Ns,X3,'xm','LineWidth',1)
set(gca,'fontsize',18)
xlabel('index'); ylabel('X'); xlim([1,Ns]); ylim([-4 4]);

figure(2)
plot(1:Ns,Y1,'xb',1:Ns,Y2,'xg',1:Ns,Y3,'xm','LineWidth',1)
hold on
plot([1 Ns],[meanY1 meanY1],'b-', 'LineWidth', 4);
plot([1 Ns],[meanY2 meanY2],'g-', 'LineWidth', 4);
plot([1 Ns],[meanY3 meanY3],'m-', 'LineWidth', 4);
hold off
xlabel('index'); ylabel('Y'); xlim([1,Ns]); ylim([0 1]);
```

MC convergence analysis (finite mean)

$$X \sim \mathcal{N}(0,1)$$



$$Y = \frac{1}{1+X^2}$$

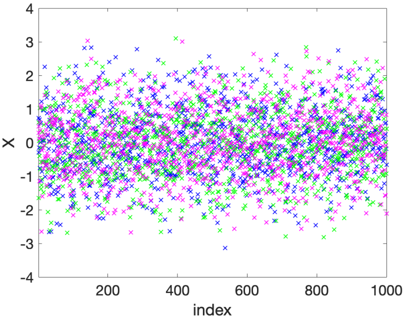


MC convergence analysis (infinite mean)

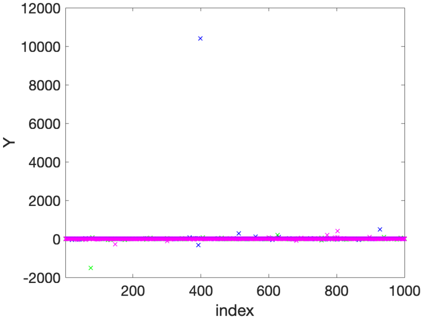
```
clc; clear; close all;

Ns = 10^3;
X1 = randn(Ns,1); X2 = randn(Ns,1); X3 = randn(Ns,1);
Z1 = 1./(1+X1); Z2 = 1./(1+X2); Z3 = 1./(1+X3);

meanZ1 = mean(Z1)
meanZ2 = mean(Z2)
meanZ3 = mean(Z3)

figure(1)
plot(1:Ns,X1,'xb',1:Ns,X2,'xg',1:Ns,X3,'xm','LineWidth',1)
xlabel('index'); ylabel('X'); xlim([1,Ns]); ylim([-4 4]);

figure(2)
plot(1:Ns,Z1,'xb',1:Ns,Z2,'xg',1:Ns,Z3,'xm','LineWidth',1)
hold on
plot([1 Ns],[meanZ1 meanZ1],'b-', 'LineWidth', 4);
plot([1 Ns],[meanZ2 meanZ2],'g-', 'LineWidth', 4);
plot([1 Ns],[meanZ3 meanZ3],'m-', 'LineWidth', 4);
hold off
xlabel('index'); ylabel('Y'); xlim([1,Ns]); ylim([-2000 12000]);
```

MC convergence analysis (infinite mean)

$$X \sim \mathcal{N}(0,1)$$

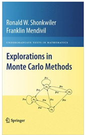


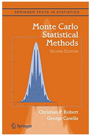
$$Z = \frac{1}{1+X}$$

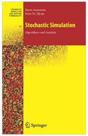


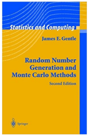
References

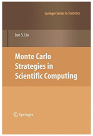
- S. B. Volchan, *What Is a Random Sequence?* **The American Mathematical Monthly**, 109: 46–63, 2002.
- T. Homem-de-Mello and G. Bayraksan, *Monte Carlo sampling-based methods for stochastic optimization*, **Surveys in Operations Research and Management Science**, 19: 56–85, 2014.
- A. Cunha Jr, R. Sampaio, R. Nasser, H. Lopes, and K. Breitman, *Uncertainty quantification through Monte Carlo method in a cloud computing setting*, **Computer Physics Communications**, 185: 1355-1363, 2014.
- R. W. Shonkwiler, and F. Mendivil, **Explorations in Monte Carlo Methods**, Springer, 2009.
- C. Robert, and G. Casella, **Monte Carlo Statistical Methods**, Springer, 2nd edition, 2005.
- S. Asmussen, and P. W. Glynn, **Stochastic Simulation: Algorithms and Analysis**, Springer, 2007.
- J. E. Gentle, **Random Number Generation and Monte Carlo Methods**, Springer, 2nd Edition, 2004.
- J. S. Liu, **Monte Carlo Strategies in Scientific Computing**, Springer, 2009.
- R. Y. Rubinstein, and Dirk P. Kroese, **Simulation and the Monte Carlo Method**, Wiley, 3rd Edition, 2016.
- D. P. Kroese, T. Taimre, and Z. I. Botev, **Handbook of Monte Carlo Methods**, Wiley, 2011.

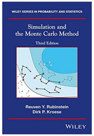


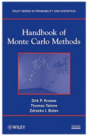












How to cite this material?

A. Cunha Jr, *Monte Carlo in Action*, 2021.

@AmericoCunhaJr

@AmericoCunhaJr @AmericoCunhaJr