

---

# *Introduction to Monte Carlo Methods & Random Variable Generation*

*Prof. Nicholas Zabaras  
University of Notre Dame  
Notre Dame, IN, USA*

*Email: [nzabaras@gmail.com](mailto:nzabaras@gmail.com)  
URL: <https://www.zabaras.com/>*

*September 25, 2017*



# Contents

---

- Review of the Bayesian inference framework, Introducing the Monte Carlo simulation, reviewing the central Limit Theorem and Law of Large Numbers, indicator functions, error approximations
- Examples showing convergence of the MC simulator, Generalization of the MC Estimator in High Dimensions
- Sample Representation of the Monte Carlo Estimator
- Deterministic vc MC integration, Using MC for Computing integrals, expectations and Bayes factors
- Need for sampling methods, Sampling from a discrete distribution, Reverse sampling for continuous distributions, Transformation Methods, Box-Muller Algorithm, sample from the Gaussian

Following closely:

- C. Robert, G. Casella, Monte Carlo Statistical Methods (Ch.. 1, 2, 3.1, & 3.2) ([google books](#), [slides](#), [video](#))
- J. S. Liu, MC Strategies in Scientific Computing (Chapters 1 & 2)
- J-M Marin and C. P. Robert, Bayesian Core (Chapter 2)
- Statistical Computing & Monte Carlo Methods, A. Doucet (course notes, 2007)



# The Bayesian Model

---

- Let us revisit our Bayesian model

$$\pi(\theta | x) = \frac{\pi(\theta)f(x | \theta)}{\int_{\Theta} \pi(\theta)f(x | \theta)d\theta}$$

- In most problems of interest there are no analytical closed forms for the posterior (using conjugate priors is one exception).
- Also note that the denominator in the Bayes' equation above implies the calculation of the following high-dimensional integral:

$$\int_{\Theta} \pi(\theta)f(x | \theta)d\theta$$

# The Bayesian Model

---

- Consider the posterior model

$$\pi(\theta | x) = \frac{\pi(\theta)f(x | \theta)}{\int_{\Theta} \pi(\theta)f(x | \theta)d\theta}$$

- Typical **point estimates** based on the posterior include the following:

$$\mathbb{E}[\theta | x] = \int_{\Theta} \theta \pi(\theta | x) d\theta$$

$$Var[\theta | x] = \int_{\Theta} \theta^2 \pi(\theta | x) d\theta - (\mathbb{E}[\theta | x])^2$$

- We are often interested in **the mode of the posterior**  $\pi(\theta | x)$

$$\theta^{MAP} = \arg \max_{\theta} \pi(\theta | x)$$

- If  $\theta = (\theta_1, \theta_2)$  and  $\theta_2$  is a nuisance parameter, **marginal distributions** are also often needed:

$$\pi(\theta_1 | x) = \int \pi(\theta_1, \theta_2 | x) d\theta_2$$

# The Bayesian Model

---

- Consider the posterior model

$$\pi(\theta | x) = \frac{\pi(\theta)f(x | \theta)}{\int_{\Theta} \pi(\theta)f(x | \theta)d\theta}$$

- The predictive distribution of  $Y$ ,  $Y \sim g(y|x)$ , and its mean  $\mathbb{E}[Y | x]$ , are:

$$g(y | x) = \int f(y | \theta)\pi(\theta | x)d\theta$$

$$\mathbb{E}[Y | x] = \iint yf(y | \theta)\pi(\theta | x)d\theta dy$$

- Similarly, for **model selection**, we have seen that the posterior takes the form:

$$\pi(k, \theta_k | x) = \frac{\pi(k)\pi_k(\theta_k)f_k(x | k, \theta_k)}{\sum_{k=1}^{\infty} \pi(k) \int_{\Theta_k} \pi_k(\theta_k)f_k(x | k, \theta_k)d\theta_k}$$



# Monte Carlo Simulation

---

- All of the calculations reviewed earlier require computing high-dimensional integrals.
- Monte Carlo simulation provides the means for effective calculation of these integrals and for resolving many more issues.

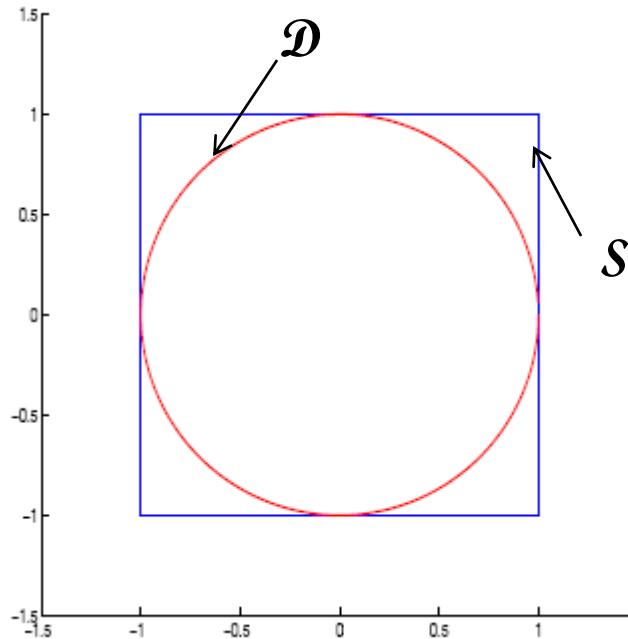
Some historical (early) references on Monte Carlo Methods:

- N. Metropolis and S. Ulam, [The Monte Carlo Method](#), [J Amer. Statist. Ass.](#), Vol. 44, pp. 335-341 (1949)
- N. Metropolis, [The Beginning of the Monte Carlo Method](#), [Los Alamos Science Special Issue](#) (1987)



# Introducing Monte Carlo Simulation

- Consider a  $2 \times 2$  square  $\mathcal{S} \subset \mathbb{R}^2$  as shown in the Figure below.



- Let an inscribed circle  $\mathcal{D}$  of radius 1 in the square  $\mathcal{S}$ .

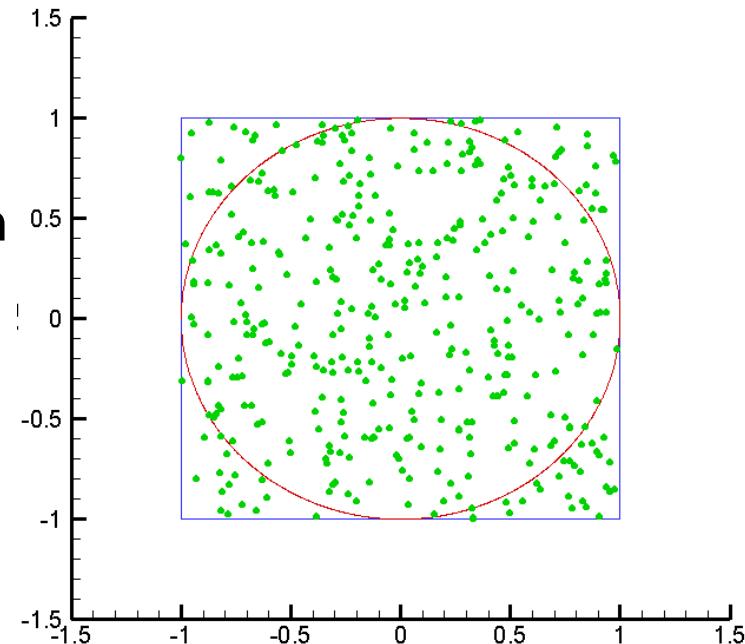
# Introducing Monte Carlo Simulation

- Let us consider dropping darts uniformly on the square  $\mathcal{S}$ . This means that the probability of the dart falling in a subdomain  $A$  of  $\mathcal{S}$  is proportional to the area of  $A$ .
- Let  $D=(x,y)$  define a random variable on  $\Theta = \mathcal{S}$  that represents the location of the drop of the dart. We have:

$$P(D \in A) = \frac{\int_A dx dy}{\int_{\mathcal{S}} dx dy}$$

- Assume  $N$  independent drops of the dart on the square  $\mathcal{S}$ , i.e.

$$\{D_1, D_2, \dots, D_N\}$$



# Introducing Monte Carlo Simulation

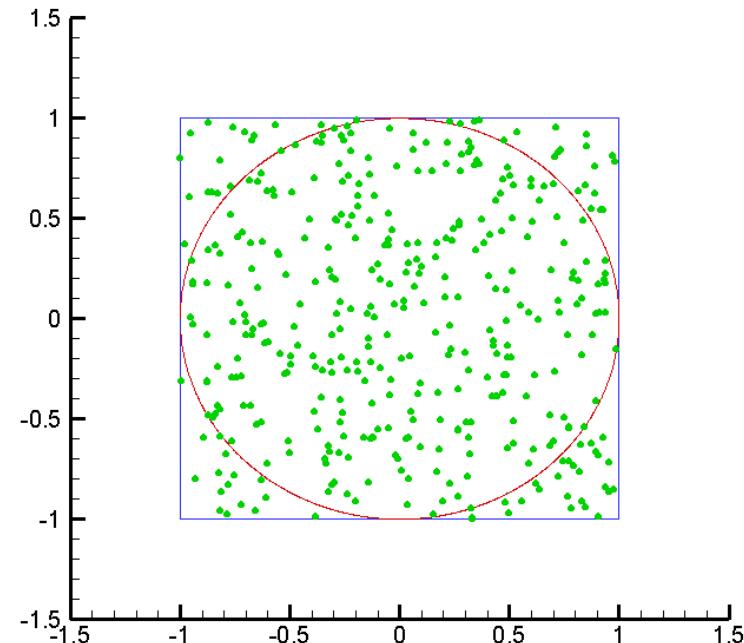
$$P(D \in A) = \frac{\int_A dxdy}{\int_S dxdy}$$

- A common sense estimation of the above probability would be

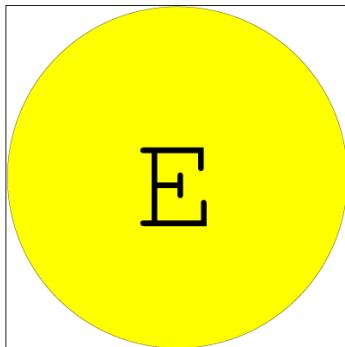
$$P(D \in A) \simeq \frac{\text{Times the dart fell in } A}{N}$$

where  $N$  is the total number of dart falls.

- Can we give a statistical justification of this result?



# Calculate $\Pr(E)$ in Monte Carlo using R



```
> N=2000  
> x<-runif(N)-0.5  
> y<-runif(N)-0.5  
> A<-ifelse((sqrt(x^2+y^2)-0.5<0),1,0)  
> mean(A)  
[1] 0.7825
```

# Indicator Functions

- Let us introduce the indicator function for the event  $A$ , i.e. let

$$\mathbb{I}_A(x, y) = \begin{cases} 1 & \text{if drop point of dart } d = (x, y) \in A, \\ 0, & \text{otherwise} \end{cases}$$

- Let us compute the probability for  $D \in A$ .

$$P(D \in A) = \frac{\int_S \mathbb{I}_A(x, y) dx dy}{\int_S dx dy} = \frac{\int_S \mathbb{I}_A(x, y) dx dy}{4} = \int_S \mathbb{I}_A(x, y) \frac{1}{4} dx dy$$

- The above results come immediately noticing that

$$\int_S \mathbb{I}_A(x, y) dx dy = \int_A \mathbb{I}_A(x, y) dx dy + \int_{S \setminus A} \mathbb{I}_A(x, y) dx dy = \int_A 1 dx dy + \int_{S \setminus A} 0 dx dy = \int_A 1 dx dy$$

# Indicator Functions

$$P(D \in A) = \frac{\int_A \mathbb{I}_A(x, y) dx dy}{\int_S dx dy} = \frac{\int_A \mathbb{I}_A(x, y) dx dy}{4} = \int_S \mathbb{I}_A(x, y) \frac{1}{4} dx dy$$

- The probability density associated to  $P$  is  $\frac{1}{4}$  - this is the density of the uniform distribution on  $S$  denoted as  $\mathcal{U}_S$
- Let us introduce the random variable  $V(D) := \mathbb{I}_A(D) := \mathbb{I}_A(X, Y)$ , where  $X, Y$  are the random variables representing the Cartesian coordinates of a uniformly distributed point  $d=(x,y)$  on  $S$ ,  $D \sim \mathcal{U}_S$ , where a dart falls.
- With this notation

$$P(d \in A) = \int_S \mathbb{I}_A(x, y) \frac{1}{4} dx dy = \mathbb{E}_{\mathcal{U}_S}(V)$$



# **Strong Law of Large Numbers**

---

- Let  $X_i$  for  $i = 1, 2, \dots, N$  be independent and identically distributed random variables (i.i.d.) with mean  $\mathbb{E}(X_i) = \mu$  and variance  $V(X_i) = \sigma^2 < \infty$ , the strong LLN states for the sample mean  $\bar{X}_N$  the following:

$$\lim_{N \rightarrow \infty} \bar{X}_N = \mu \text{ almost surely}$$



# Weak Law of Large Numbers

- Let  $X_i$  for  $i = 1, 2, \dots, N$  be independent and identically distributed random variables (i.i.d.) with mean  $\mathbb{E}(X_i) = \mu$  and variance  $V(X_i) = \sigma^2 < \infty$ , the weak LLN states that **the sample mean**  $\bar{X}_N$  is a random variable that converges to the **true mean** as  $N \rightarrow \infty$ , i.e.

$$\bar{X}_N = \frac{\sum_{i=1}^N X_i}{N} \rightarrow \mu \text{ as } N \rightarrow \infty$$

- More formally:

$$\lim_{N \rightarrow \infty} \Pr\left[\left|\bar{X}_N - \mu\right| \geq \varepsilon\right] = 0 \quad \forall \varepsilon > 0$$

- Note that

$$\mathbb{E}[\bar{X}_N] = \frac{\sum_{i=1}^N \mathbb{E}[X_i]}{n} = \frac{\sum_{i=1}^N \mu}{n} = \mu \quad \text{and} \quad \text{var}[\bar{X}_N] = \frac{\sum_{i=1}^N \text{var}[X_i]}{N^2} = \frac{\sum_{i=1}^N \sigma^2}{N^2} = \frac{\sigma^2}{N}$$

# The Central Limit Theorem

- Let  $X_i$  for  $i = 1, 2, \dots, N$  be independent and identically distributed (i.i.d) random variables each with expectation  $\mu$  and variances  $\sigma^2$ . Then:

$$\mathbb{E}\left[\bar{X}_N\right] = \mathbb{E}\left[\frac{\sum_{i=1}^N X_i}{N}\right] = \frac{1}{N} \sum_{i=1}^N \mathbb{E}(X_i) = \frac{1}{N} \sum_{i=1}^N \mu_i = \frac{1}{N} N \mu = \mu$$

$$Var\left[\bar{X}_N\right] = Var\left[\frac{\sum_{i=1}^N X_i}{N}\right] = \frac{1}{N^2} \sum_{i=1}^N Var(X_i) = \frac{1}{N^2} \sum_{i=1}^N \sigma^2 = \frac{1}{N} N \sigma^2 = \frac{\sigma^2}{N}$$

- Under some weak conditions (such as finite variance), the sample

mean  $\bar{X}_N = \frac{\sum_{i=1}^N X_i}{N}$  has a limiting normal distribution:

$$\lim_{N \rightarrow \infty} \frac{\bar{X}_N - \mu}{\sqrt{\frac{\sigma^2}{N}}} \sim \mathcal{N}(0,1)$$

$$or \text{ as } N \rightarrow \infty: \quad \bar{X}_N = \mathcal{N}\left(\mu, \frac{\sigma^2}{N}\right)$$



# Back to Indicator Functions

---

- Let  $X_i$  for  $i = 1, 2, \dots, N$  be an indicator function for an event  $F$ , i.e. let

$$X_i = \begin{cases} 1 & \text{if the outcome of the experiment } i \text{ is } F, \\ 0 & \text{otherwise} \end{cases}$$

Then as a result of the law of large numbers,

$$\overline{X}_N = \sum_{i=1}^N \frac{X_i}{N} \Rightarrow \Pr(F) \text{ as } N \rightarrow \infty$$

- Note that herein we used that

$$\mathbb{E}[X_i] = 1 \times \Pr(F) + 0 \times \Pr(\overline{F}) = \Pr(F)$$



# Returning to Our Dart Drop Experiment

- Let us introduce the random variables  $\{V_i := V(D_i) := \mathbb{I}_A(D_i), i = 1, 2, \dots, N\}$  associated to the drops  $\{D_i, i = 1, 2, \dots, N\}$  and consider the sum

$$S_N = \frac{\sum_{i=1}^N V_i}{N} = \frac{\text{number of drops that fell in } A}{N}$$

*Empirical average  
of i.i.d.  $\{V_i := V(D_i), i = 1, 2, \dots, N\}$*

- Assuming  $(N \rightarrow \infty)$ , then the law of large numbers (since  $\mathbb{E}_{\mu_s}(V) < +\infty$ ) yields

$$\lim_{N \rightarrow \infty} S_N = \mathbb{E}_{\mu_s}(V) \text{ (almost surely)}$$

where we already proved that  $\Pr(D \in A) = \mathbb{E}_{\mu_s}(V)$

- When  $N \rightarrow \infty$ , this justifies our intuitive result introduced earlier.



# Mean Square Error

- Since  $P(d \in \mathcal{D}) = \int_{\mathcal{D}} \frac{1}{4} dx dy = \frac{\pi}{4}$ ,  $S_N$  is an unbiased estimator of  $\frac{\pi}{4}$ .

$$S_N \text{ is a random variable : } S_N = \frac{\pi}{4} + E_N$$

Error term

- To characterize the precision of the estimator, we can use the following:

$$\text{Var}(E_N) = \text{Var}(S_N) = \frac{1}{N^2} \sum_{i=1}^N \text{Var}(V_i) = \frac{1}{N} \text{Var}(V_1) \quad (\text{recall the } V_i \text{'s are i.i.d.})$$

- This means that

$$\sqrt{\text{Var}(S_N)} = \sqrt{\mathbb{E}[(S_N - \mathbb{E}(S_N))^2]} = \sqrt{\mathbb{E}[(S_N - \Pr(D \in \mathcal{D}))^2]} = \frac{1}{\sqrt{N}} \sqrt{\text{Var}(V_1)}$$

Thus the mean square error between  $S_N$  and  $\Pr(D \in \mathcal{D})$  decreases as  $\frac{1}{\sqrt{N}}$

- For our problem  $\text{Var}(V_1) = P(D \in \mathcal{D}) - (P(D \in \mathcal{D}))^2 = \frac{\pi}{4} - \left(\frac{\pi}{4}\right)^2 = 0.1685$ .

# Properties of the Estimator

- Applying the central limit theorem (since  $\text{Var}(V) < +\infty$  ) :

$$S_N \xrightarrow{d} \mathcal{N}\left(\frac{\pi}{4}, \frac{\text{Var}(V)}{N}\right)$$

- The probability of the error being larger than  $\varepsilon = 2\sqrt{\frac{\text{var}(V)}{N}}$  is given as:

$$\Pr\left[\left|S_N - \frac{\pi}{4}\right| \geq 2\sqrt{\frac{\text{var}(V)}{N}}\right] = 2\left\{1 - \Phi\left(\frac{2\sqrt{\frac{\text{var}(V)}{N}}}{\sqrt{\frac{\text{var}(V)}{N}}}\right)\right\} = 2(1 - \Phi(2)) = 0.0456$$

where  $\Phi(x)$  the CDF of  $\mathcal{N}(0,1)$ .

- One can similarly prove that for any integer  $N \geq 1$  and  $\varepsilon > 0$ ,

$$\Pr\left[\left|S_N - \frac{\pi}{4}\right| \geq \varepsilon\right] = 2\left\{1 - \Phi\left(\frac{\varepsilon}{\sqrt{\frac{\text{var}(V)}{N}}}\right)\right\} = \text{erfc}\frac{\varepsilon}{\sqrt{2\frac{\text{var}(V)}{N}}} < Ce^{-\varepsilon^2 \frac{N}{2\text{var}(V)}} \sqrt{\frac{\text{var}(V)}{N\varepsilon^2}}$$

where  $\Phi(x) = \frac{1}{2}\left(1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right)$  and for large  $x$ :  $\text{erfc}(x) \approx \frac{e^{-x^2}}{x\sqrt{\pi}}$  (see here).

# Coefficient of Variation

---

- The coefficient of variation for our Monte Carlo estimator is given as:

$$\text{Coeff. of Var.} \equiv COV(S_N) = \frac{\sqrt{\text{var}(S_N)}}{\mathbb{E}(S_N)} = \frac{\sqrt{\pi/4(1-\pi/4)}}{\pi/4\sqrt{N}} = \sqrt{\frac{4(1-\pi/4)}{N\pi}} = \frac{0.5233}{\sqrt{N}}$$

- For different COVs, the number of samples needed is given as:

$$COV = 10\% \Rightarrow N = 28$$

$$COV = 5\% \Rightarrow N = 110$$

$$COV = 1\% \Rightarrow N = 2739$$

- With all of the above error estimates, we conclude that the approximation error varies as  $O\left(\frac{1}{\sqrt{N}}\right)$ .



# Properties of the Estimator

- An alternative non-asymptotic error estimate<sup>a</sup> using a Bernstein type inequality is:

$$\forall N \geq 1 \text{ and } \varepsilon > 0, P\left(\left|S_N - \frac{\pi}{4}\right| > \varepsilon\right) \leq 2e^{-2N\varepsilon^2}$$

- Using this result, we can show that:

$$\forall \alpha \in (0,1], P\left(\left|S_N - \frac{\pi}{4}\right| > \varepsilon\right) < \alpha \text{ for } N \geq \frac{\log(2/\alpha)}{2\varepsilon^2}$$

- Alternatively, for any  $N \geq 1$ , using the Eq. above we can write:

$$\Pr\left[\left|S_N - \frac{\pi}{4}\right| > \sqrt{\frac{\log(40)}{2N}}\right] \leq 0.05$$

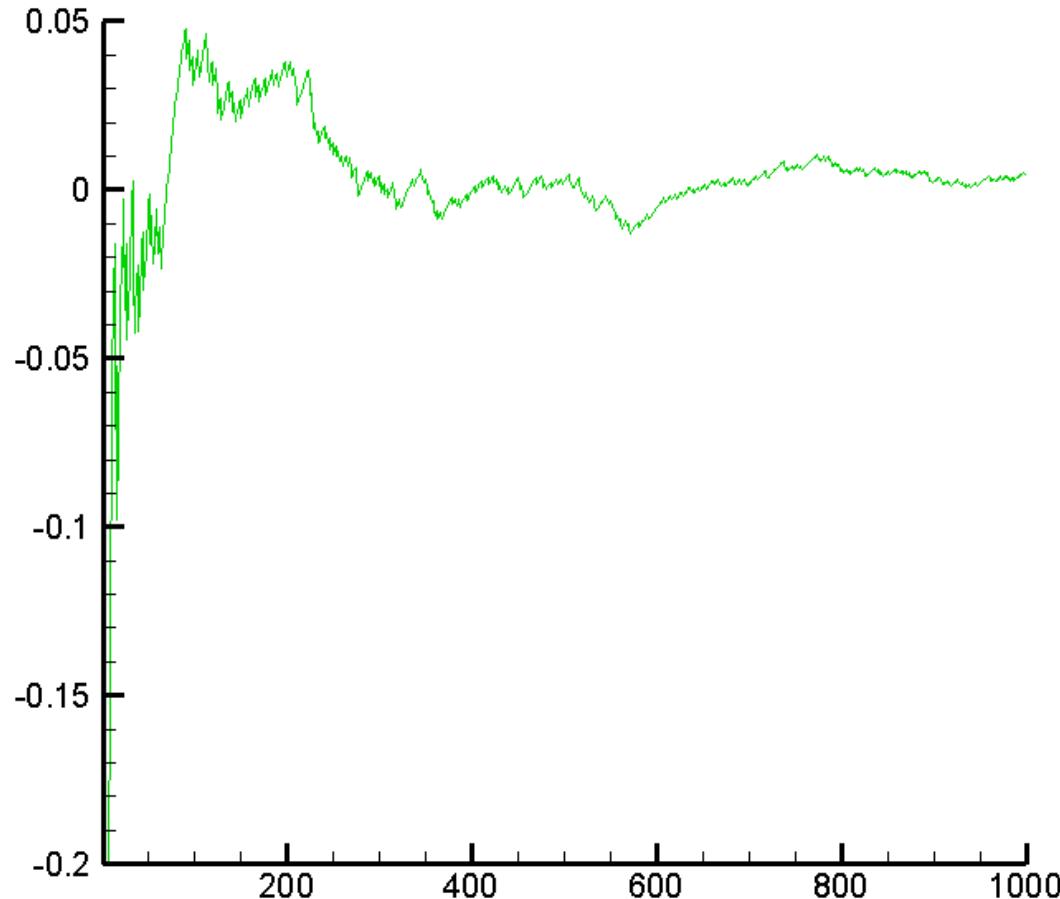
- These expressions for the approximation error show that it is inversely proportional to  $\sqrt{N}$ .

<sup>a</sup>From Statistical Computing & Monte Carlo Methods, A. Doucet.



# Convergence of the Simulator

- Convergence of  $S_N - \frac{\pi}{4}$  as a function of  $N$  (one realization)

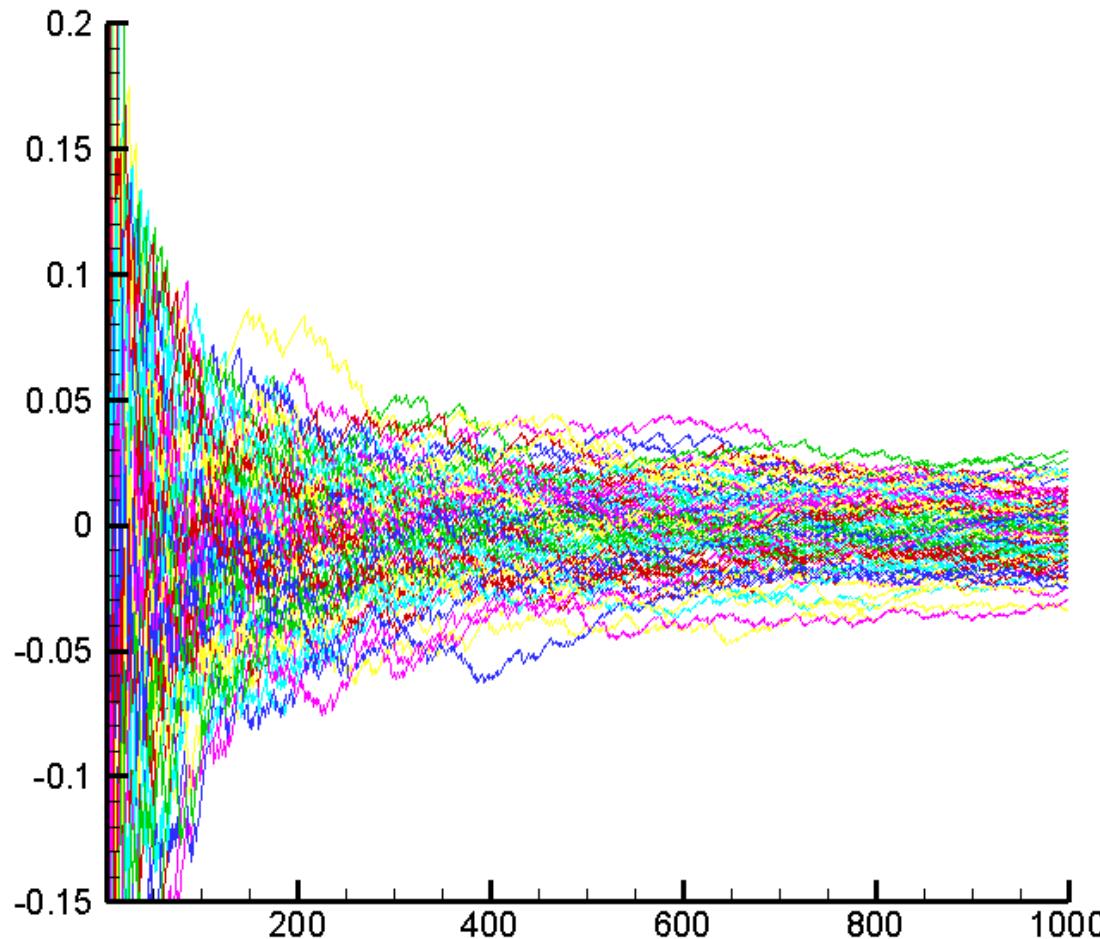


See [here](#) for a  
C++  
implementation  
(Meanerror\_plt)

# Convergence of the Simulator

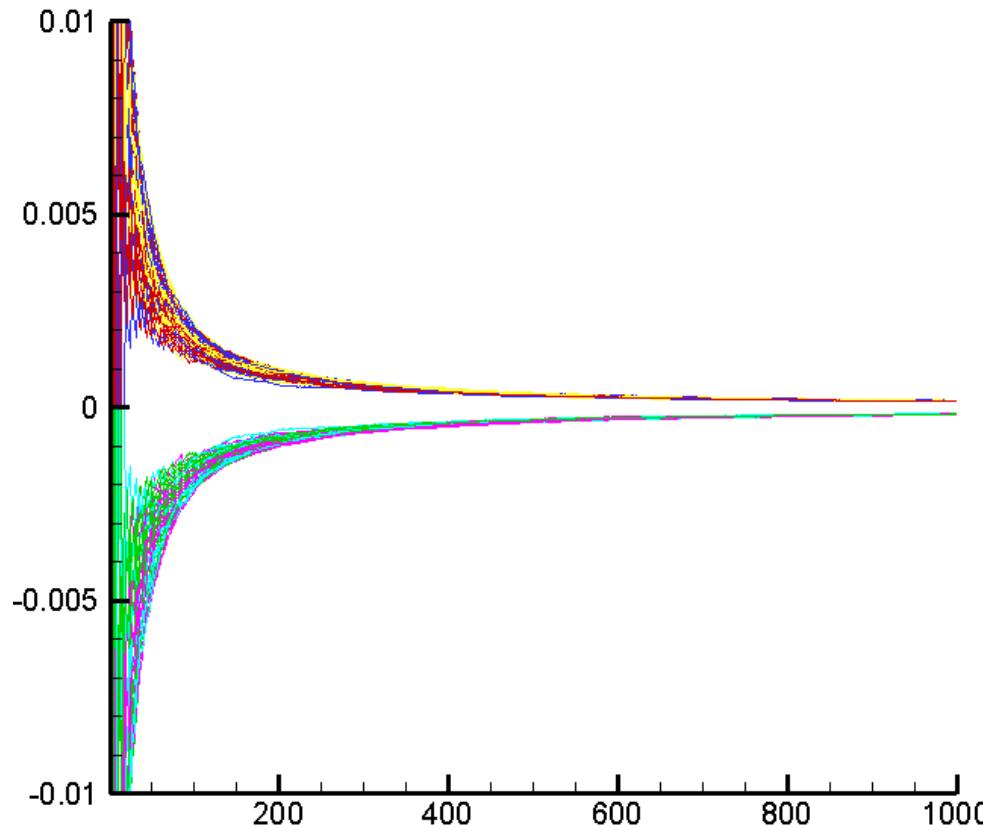
- Convergence of  $S_N - \frac{\pi}{4}$  as a function of  $N$  (one hundred realizations)

See [here](#) for a  
C++  
implementation  
(Meanerror.plt)



# Convergence of the Simulator

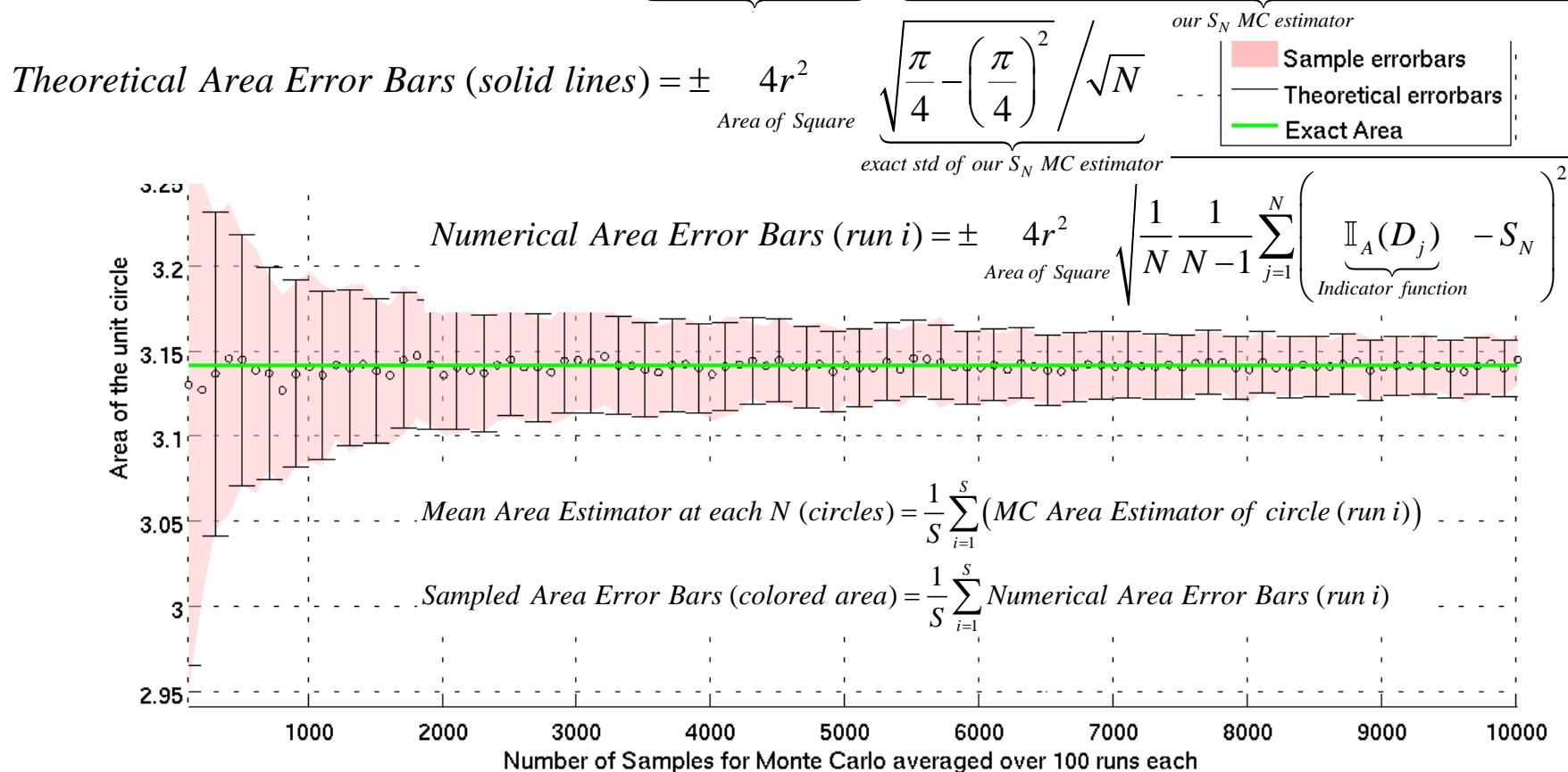
- Square root empirical mean square error  $S_N - \frac{\pi}{4}$  across 100 realizations as a function of  $N$  and  $\pm \sqrt{\frac{Var(V)}{N}}$  (dotted)



See [here](#) for a C++ implementation (Var\_error.plt)

# Convergence of the Estimator

MC Estimator of the area  $A$  of circle (run  $i$ ) =  $\underbrace{\text{Area of square} \times \text{Number of counts of samples falling in circle}}_{\text{our } S_N \text{ MC estimator}} / N$



- Monte Carlo estimator of the area of a unit circle: (a) Circles: MC mean averaged over the  $S=100$  runs; (b) Theoretical error bars using the estimator discussed earlier; (c) Average of the MC numerical error bars (from the  $S=100$  MC runs for each  $N$ ). [MatLab implementation](#) is given here.

# Generalization of the Dart Drop Experiment

---

- Consider now the case where  $\Theta = \mathbb{R}^{n_\theta}$ ,  $n_\theta \gg 1$ . We assume a hypercube  $S^{n_\theta}$  and the inscribed hyperball  $\mathcal{D}^{n_\theta}$  in  $\Theta$ .
- We can build the same estimator as in 2D. Our indicator function now becomes  $\mathbb{I}_{\mathcal{D}^{n_\theta}}(D)$ .
- All of our earlier derivations are still applicable. The rate of convergence of the estimator in the mean square sense is independent of  $n_\theta$  and equal to  $1/\sqrt{N}$ .
- This is not the case using a deterministic method on a grid of regularly spaced points where the convergence rate is typically of the form  $1/N^{r/n_\theta}$  where  $r$  is related to the smoothness of the contours of  $\mathcal{D}^{n_\theta}$ .
- Monte Carlo methods are thus attractive when  $n_\theta$  is large.



# Generalization of the Dart Drop Experiment

- Assume you are interested in computing the volume of the hypersphere of radius  $R = 1$  in  $n_\theta$ -dimensions:

$$vol(S_{n_\theta}) = \frac{\pi^{\frac{n_\theta}{2}}}{\Gamma\left(\frac{n_\theta}{2} + 1\right)} \rightarrow 0 \text{ as } n_\theta \rightarrow \infty$$

- We want to do so using samples from the hypercube  $[-1, 1]^{n_\theta}$  of volume  $2^{n_\theta}$ .
- Using  $N$  samples, the variance of our MC estimator is

$$\frac{Var(X)}{N} = \frac{p_{n_\theta}(1 - p_{n_\theta})}{N} \approx \frac{p_{n_\theta}}{N} \text{ as } n_\theta \rightarrow \infty$$

where:  $X \sim Bernoulli(p_{n_\theta})$  with  $p_{n_\theta} = \frac{vol(S_{n_\theta})}{2^{n_\theta}}$ .

# Monte Carlo and the Curse of Dimensionality

- The coefficient of variation is then:

$$COV = \frac{\sqrt{\frac{Var(X)}{N}}}{\mathbb{E}(X)} \approx \frac{\sqrt{\frac{p_{n_\theta}}{N}}}{p_{n_\theta}} = \frac{1}{\sqrt{Np_{n_\theta}}} \text{ as } n_\theta \rightarrow \infty$$

- To obtain a reasonable relative error, we would need  $N \approx 100 p_{n_\theta}^{-1}$ .
- For this choice:  $COV \approx 0.1$
- For  $n_\theta = 20, N \approx 100 p_{n_\theta}^{-1} = 100 \times 2^{20} \frac{\Gamma(11)}{\pi^{10}} = 100 \times 2^{20} \frac{10!}{\pi^{10}} \approx 4.06 \times 10^9$
- Clearly a plain Monte Carlo approach is not as helpful in 20 and higher dimensions! You are trying in a huge volume of the hypercube to hit the infinitesimal hypersphere!



# Generalization of the Dart Drop Experiment

---

Consider :  $\mathbb{E}(X) = 1$ , and  $\text{Var}(X) = C\alpha^{n_\theta}$ , with  $\alpha > 1$ .

Then : For  $\frac{\text{Var}(X)}{N} \leq \varepsilon \Rightarrow N \geq \frac{C\alpha^{n_\theta}}{\varepsilon}$ .

- To obtain a fixed precision  $\varepsilon$  in the variance, a number of samples that is exponential in the dimension is needed!
- So using a plain Monte Carlo method still suffers from the curse of dimensionality
- ✓ MC not appropriate for rare events modeling.



# Generalization

---

- Assume  $N \gg 1$  i.i.d. samples  $\theta^{(i)} \sim \pi (i = 1, 2, \dots, N)$
- Now consider any set  $A \subset \Theta$  and assume that we are interested in  $\pi(A) = P(\theta \in A)$ , for  $\theta \sim \pi$ . We naturally choose the following estimator:

$$\pi(A) \simeq \frac{\text{number of samples in } A}{\text{total number of samples}}$$

which by the law of large numbers is a consistent estimator of  $\pi(A)$  since

$$\lim_{N \rightarrow +\infty} \frac{1}{N} \sum_{i=1}^N \mathbb{I}_A(\theta^{(i)}) = \mathbb{E}_{\pi}(\mathbb{I}_A(\theta)) = \pi(A)$$



# Generalization

---

- Now we generalize the idea to tackle the generic problem of estimating

$$\mathbb{E}_\pi(f(\theta)) = \int_{\Theta} f(\theta) \pi(\theta) d\theta$$

where  $f : \Theta \rightarrow \mathbb{R}^{n_f}$  and  $\pi$  is a probability distribution on  $\Theta \subset \mathbb{R}^{n_x}$ .

- We assume that  $\mathbb{E}_\pi(|f(\theta)|) < +\infty$  and that  $\mathbb{E}_\pi(f(\theta))$  cannot be calculated analytically.



# Generalization

- To evaluate  $\mathbb{E}_\pi(f(\theta))$ , we consider the unbiased estimator

$$S_N(f) = \frac{1}{N} \sum_{i=1}^N f(\theta^{(i)})$$

- From the law of large numbers  $S_N(f)$  will converge and

$$\lim_{N \rightarrow +\infty} \frac{1}{N} \sum_{i=1}^N f(\theta^{(i)}) = \mathbb{E}_\pi(f(\theta)) \text{ (a.s.)}$$

- A good measure of the approximation is the variance of  $S_N(f)$ ,

$$\text{Var}_\pi[S_N(f)] = \text{Var}_\pi\left[\frac{1}{N} \sum_{i=1}^N f(\theta^{(i)})\right] = \frac{\text{Var}_\pi[f(\theta)]}{N}$$

- From the central limit (if  $\text{var}_\pi[f(\theta)] < +\infty$ ), we have:

$$\begin{aligned} \sqrt{N}(S_N(f) - \mathbb{E}_\pi(f(\theta))) &\xrightarrow[d]{N \rightarrow +\infty} \mathcal{N}(0, \text{Var}_\pi[f(\theta)]) \text{ or} \\ S_N(f) &\xrightarrow[d]{N \rightarrow +\infty} \mathcal{N}\left(\mathbb{E}_\pi(f(\theta)), \frac{\text{Var}_\pi[f(\theta)]}{N}\right) \end{aligned}$$



# Generalization

---

- The rate of convergence is independent of the dimension of  $\Theta$
- Integration in complex domains is now not a problem.
- The method is easy to implement and rather general. You will need
  - to be able to evaluate  $f(\theta) \forall \theta \in \Theta$
  - to be able to produce samples distributed according to  $\pi$

Some historical references on Monte Carlo Methods:

- N. Metropolis and S. Ulam, [The Monte Carlo Method](#), [J Amer. Statist. Ass.](#), Vol. 44, pp. 335-341 (1949)
- N. Metropolis, [The Beginning of the Monte Carlo Method](#), [Los Alamos Science Special Issue](#) (1987)



# Sample Representation of the MC Estimator

- Let us introduce the Dirac-Delta function  $\delta_{\theta_0}(\theta)$  for  $\theta_0 \in \Theta$  defined for any  $f : \Theta \leftarrow \mathbb{R}^{n_f}$  as follows:

$$\int_{\Theta} f(\theta) \delta_{\theta_0}(\theta) d\theta = f(\theta_0)$$

- Note that this implies in particular that for  $A \subset \Theta$

$$\int_{\Theta} \mathbb{I}_A(\theta) \delta_{\theta_0}(\theta) d\theta = \int_A \delta_{\theta_0}(\theta) d\theta = \mathbb{I}_A(\theta_0)$$

- For  $\theta^{(i)} \sim \pi, i = 1, \dots, N$ , we can introduce the following mixture of Delta-Dirac functions:

$$\hat{\pi}_N(\theta) := \frac{1}{N} \sum_{i=1}^N \delta_{\theta^{(i)}}(\theta)$$

which is the empirical measure, and consider for any  $A \subset \Theta$

$$\hat{\pi}_N(A) \triangleq \int_A \hat{\pi}_N(\theta) d\theta = \sum_{i=1}^N \int_A \frac{1}{N} \delta_{\theta^{(i)}}(\theta) d\theta = \frac{1}{N} \sum_{i=1}^N \mathbb{I}_A(\theta^{(i)}) = S_N(A) \text{ (# of samples in } A)$$



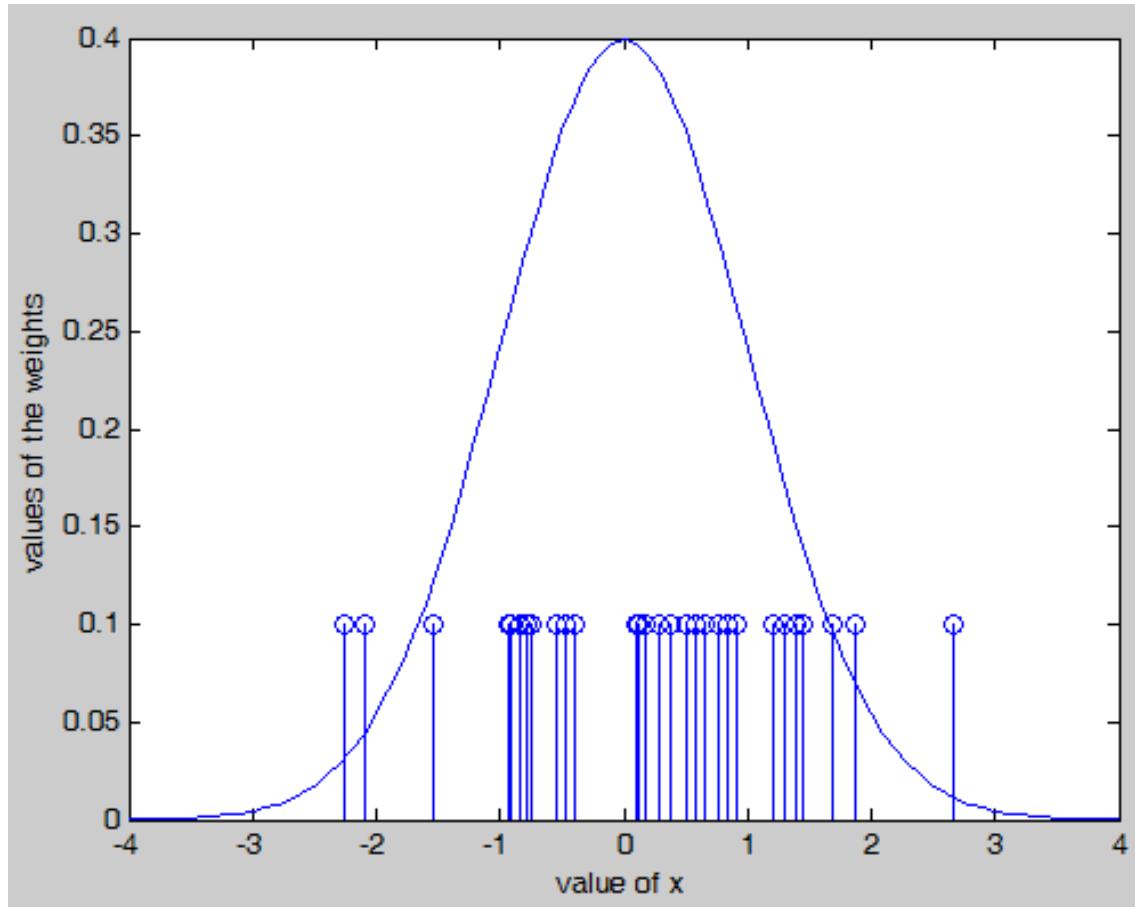
# *Sample Representation of $\pi$*

---

- The concentration of points in a given region of the space represents  $\pi$
- This is contrast with parametric statistics where one starts with samples and then introduces a distribution with an algebraic representation of the underlying population.
- Note that each sample  $\theta^{(i)}$  has a weight of  $1/N$ , but that it is also possible to consider weighted sample representations of  $\pi$ .

# Sample Representation of $\pi$

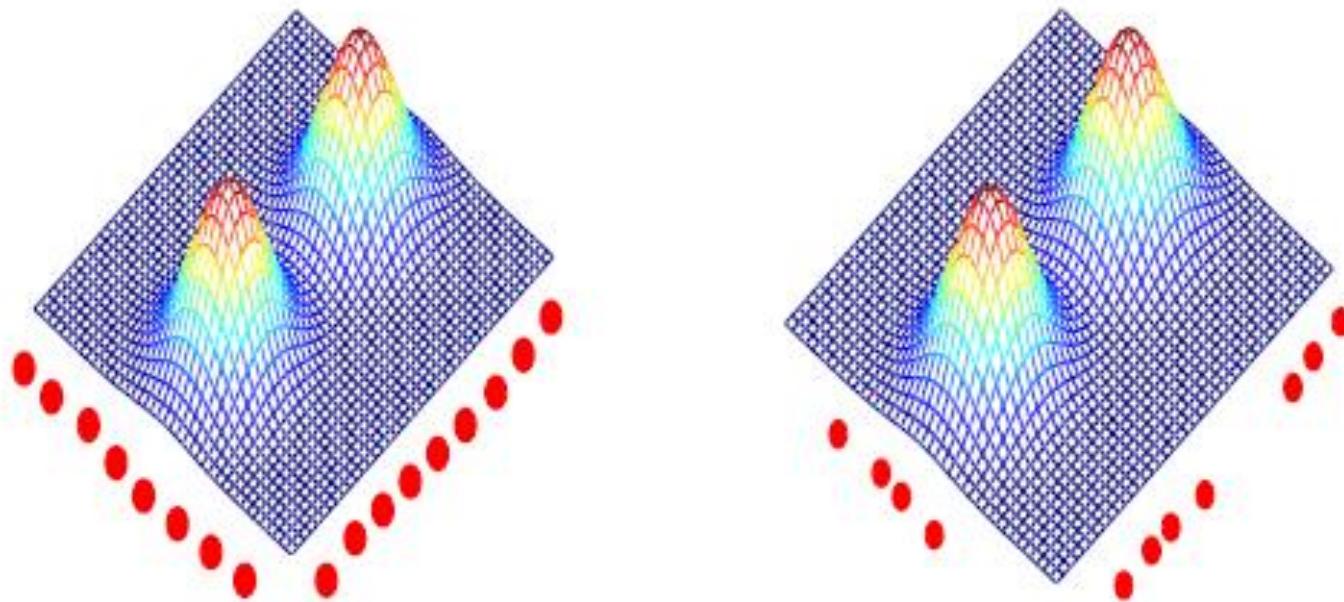
- Sample representation of a Gaussian distribution is shown below.



See [here](#) for a MatLab Implementation

# Deterministic Vs. Monte Carlo Integration

---



# Sample Representation of the MC Estimator

- Now consider the problem of estimating  $\mathbb{E}_\pi(f)$ . We simply replace  $\pi$  with its sample representation  $\hat{\pi}_N(\theta)$  and obtain:

$$\mathbb{E}_\pi(f) \simeq \int_{\Theta} f(\theta) \sum_{i=1}^N \int_A \frac{1}{N} \delta_{\theta^{(i)}}(\theta) d\theta = \sum_{i=1}^N \int_{\Theta} \frac{1}{N} f(\theta) \delta_{\theta^{(i)}}(\theta) d\theta = \frac{1}{N} \sum_{i=1}^N f(\theta^{(i)})$$

- This is precisely  $S_N(f)$ , the Monte Carlo estimator suggested earlier.
- Clearly based on  $\hat{\pi}_N(\theta)$ , we can easily estimate  $\mathbb{E}_\pi(f)$  for any  $f$ .
- For example, the variance of  $f$  is approximated as:

$$Var_\pi(f) = \mathbb{E}_\pi(f^2) - \mathbb{E}_\pi^2(f) \simeq \frac{1}{N} \sum_{i=1}^N f^2(\theta^{(i)}) - \left( \frac{1}{N} \sum_{i=1}^N f(\theta^{(i)}) \right)^2$$

# From the Algebraic to the Sample Representation

- Similarly, if we have:

$$\hat{\pi}_N(\theta_1, \theta_2) = \frac{1}{N} \sum_{i=1}^N \delta_{\theta_1^{(i)} \theta_2^{(i)}}(\theta_1, \theta_2)$$

- The marginal distribution is given as:

$$\hat{\pi}_N(\theta_1) = \frac{1}{N} \sum_{i=1}^N \delta_{\theta_1^{(i)}}(\theta_1)$$

- If we want to estimate  $\arg \max_{\theta} \pi(\theta)$  and  $\pi(\theta)$  is only known up to a normalizing constant, then a reasonable estimate is the following:

$$\arg \max_{\{\theta^{(i)}\}} \pi(\theta^{(i)})$$

# *Sampling from an Arbitrary Distribution*

---

- We can now see that if we can sample easily from an arbitrary distribution, then we could easily compute any quantities of interest.
- But how do we sample from an arbitrary distribution?
  - We will discuss about MCMC and other methods in follow up lectures



# Monte Carlo Integration

- We can use the same estimation for performing integration. Consider computing the following:

$$I = \int_D f(x)dx$$

- We can write this integration as  $I = \int_D f(x)dx = \mathbb{E}_{\pi_D} [f(x)]$  where the mean is wrt the uniform distribution in  $D$ :

$$\pi(x) = \begin{cases} 1 & \text{if } x \in D \\ 0, & \text{otherwise} \end{cases}$$

- To compute the integral  $I$ , we thus need to draw samples from this distribution. Our estimator will then be as follows:

$$\hat{I} = \frac{1}{n} \sum_{i=1}^n f(x_i)$$

- The estimator converges as  $O(\frac{1}{\sqrt{N}})$ .



# Monte Carlo Integration

- We can extend this to high ( $d$ ) dimensions:

$$I = \int_D f(\mathbf{x}) d\mathbf{x}$$

- The estimator converges as  $\mathcal{O}(\frac{1}{\sqrt{N}})$  regardless of the dimensionality  $d$ .
- For a Riemann integration of this, the rate of convergence is better  $\mathcal{O}(\frac{1}{N})$  where a grid of equally spaced points is used (e.g. for  $D=[0, 1]$ ,  $\Delta x=1/N$ ).
- However, in let us say ten dimensions,  $D=[0, 1]^{10}$ , you will need  $\mathcal{O}(N^{10})$  grid points to achieve  $\mathcal{O}(\frac{1}{N})$  the same rate of convergence as in 1D!
- Generally, the rate of convergence of the Riemann approximation of the above integral is  $\mathcal{O}(1/N^{r/d})$  where  $N$  is the total number of grid points and  $r$  depends on the smoothness of the domain  $D$ .

J. S. Liu, [MC Strategies in Scientific Computing](#) (Chapter 2, Introduction)



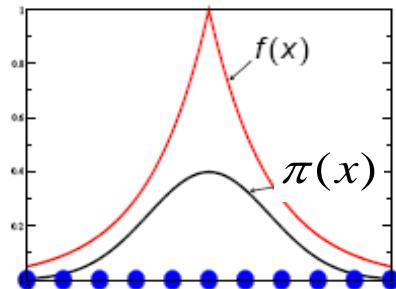
# Monte Carlo for Integration

- In computing  $I = \int_{D=[0,1]} f(x)dx$  with Riemann integration, one e.g.

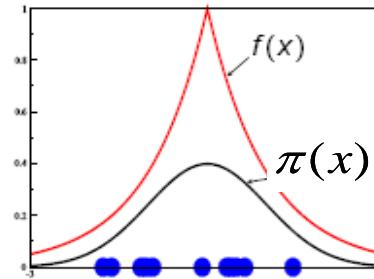
constructs a grid with  $\Delta x=1/N$ , computes  $f(x_i) = f(i\Delta x)$  and with approximation error  $\mathcal{O}(\Delta x) \propto \mathcal{O}(1/N)$  evaluates:

$$\hat{I}_N \approx \sum_{i=1}^N f(x_i) \Delta x = \frac{1}{N} \sum_{i=1}^N f(x_i)$$

- In  $d$ -dimensions, we need  $N^d$  evaluations  $f(x_i)$  to keep an error  $\mathcal{O}(1/N)$
- Clearly the MC estimator is slower in convergence  $\mathcal{O}(1/\sqrt{N})$  but it converges independently of the dimension  $d$ . It requires no grid of points. But you need to be able to sample from  $\pi(x)$ .



Riemann integration



MC integration

# Monte Carlo Integration

- Consider calculating for any function  $f(x)$ , the following integral

$$\mu(f) = \int_0^1 f(x)dx$$

- We can write this integral as:

$$\mu(f) = \int_{-\infty}^{+\infty} f(x)\mathbb{I}_{[0,1]}(x)dx = \mathbb{E}[f(X)]$$

- Here

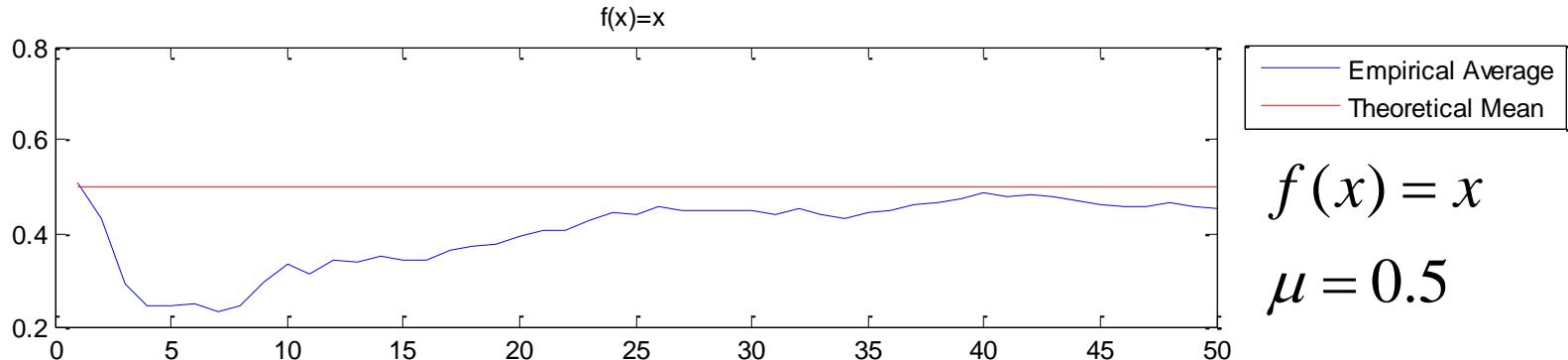
$$X \sim \mathcal{U}_{[0,1]}$$

- The Monte Carlo estimator of the integral is now:

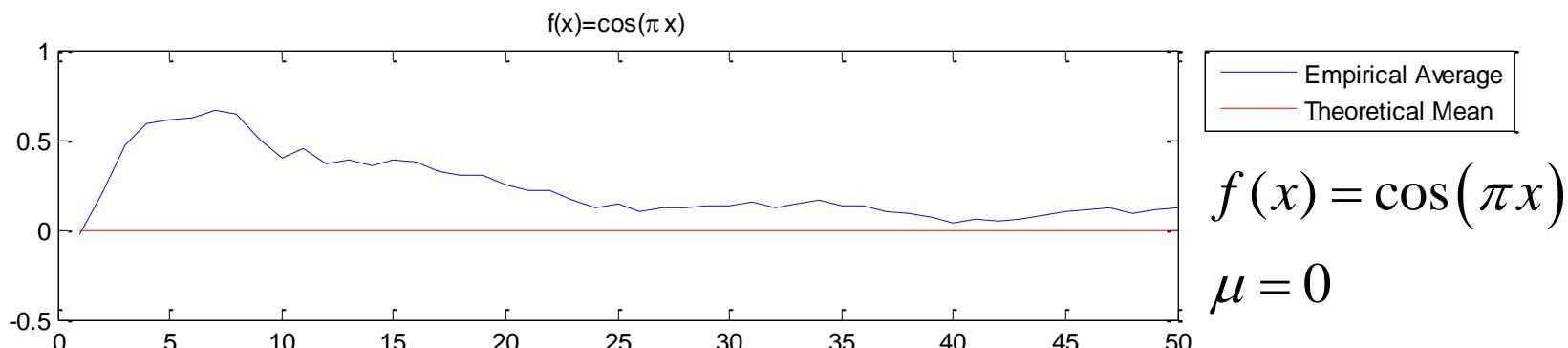
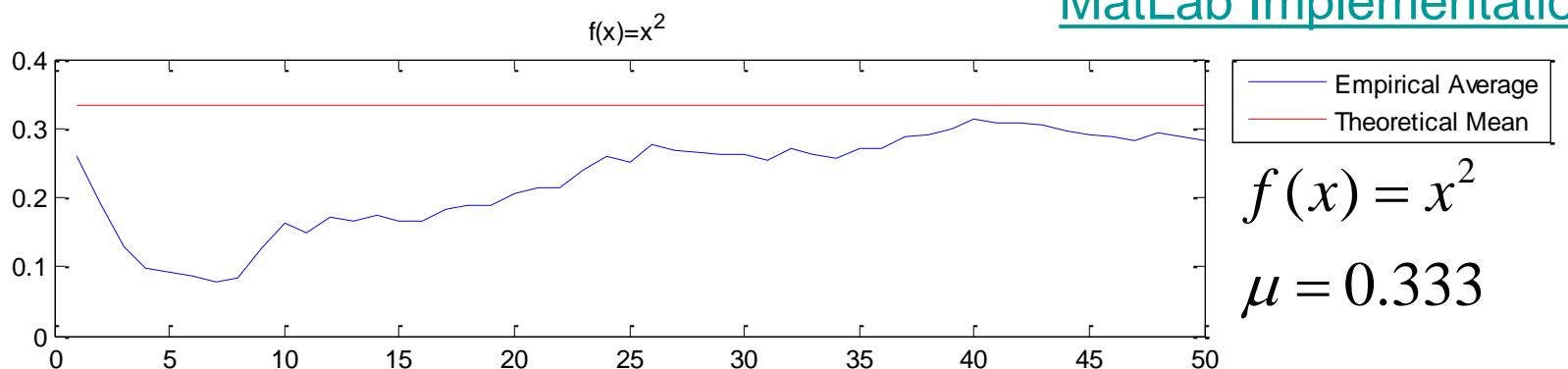
$$\hat{\mu}_N(f) = \frac{1}{N} \sum_{i=1}^N f(X_i), X_i \stackrel{i.i.d.}{\sim} U_{[0,1]}$$



# Monte Carlo Integration



MatLab Implementation



# Monte Carlo Integration: Variance

- From the earlier discussed properties of the MC estimator:

$$\mathbb{E}[\hat{\mu}_N] = \mu$$

$$Var[\hat{\mu}_N] = \frac{1}{N} Var[f(X)] = \frac{1}{N} \sigma^2(f), X \sim \mathcal{U}_{[0,1]}$$

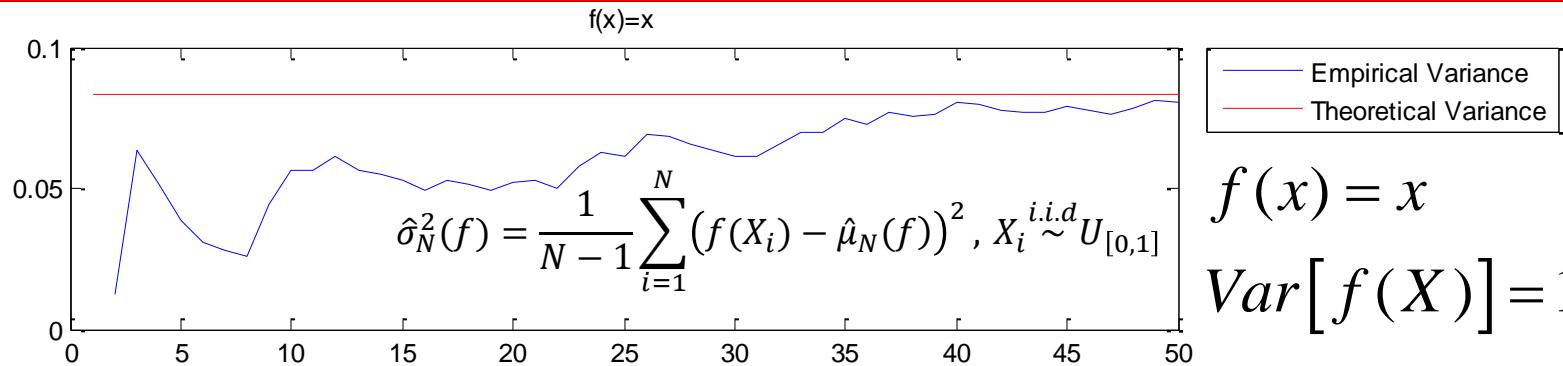
where:

$$\sigma^2(f) \equiv Var[f(X)] = \int_{-\infty}^{+\infty} (f(x) - \mu)^2 \mathbb{I}_{[0,1]}(x) dx$$

- When the variance  $\sigma^2(f) \equiv Var[f(X)]$  is unknown, we can use its MC estimator instead:

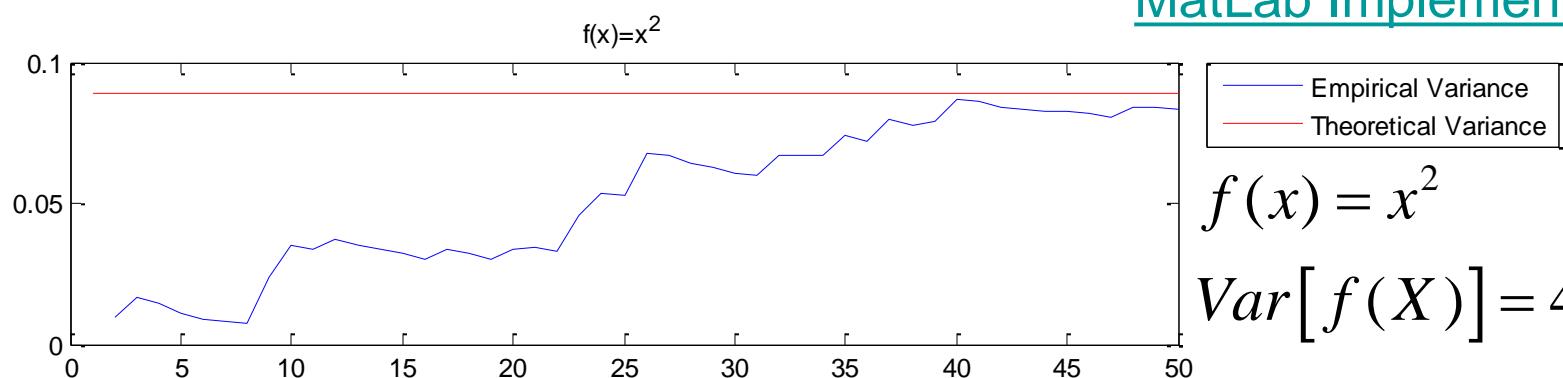
$$\hat{\sigma}_N^2(f) = \frac{1}{N-1} \sum_{i=1}^N (f(X_i) - \hat{\mu}_N(f))^2, \hat{\mu}_N(f) = \frac{1}{N} \sum_{j=1}^N f(X_j), X_i \stackrel{i.i.d.}{\sim} \mathcal{U}_{[0,1]}$$

# Monte Carlo Integration: Empirical Variance

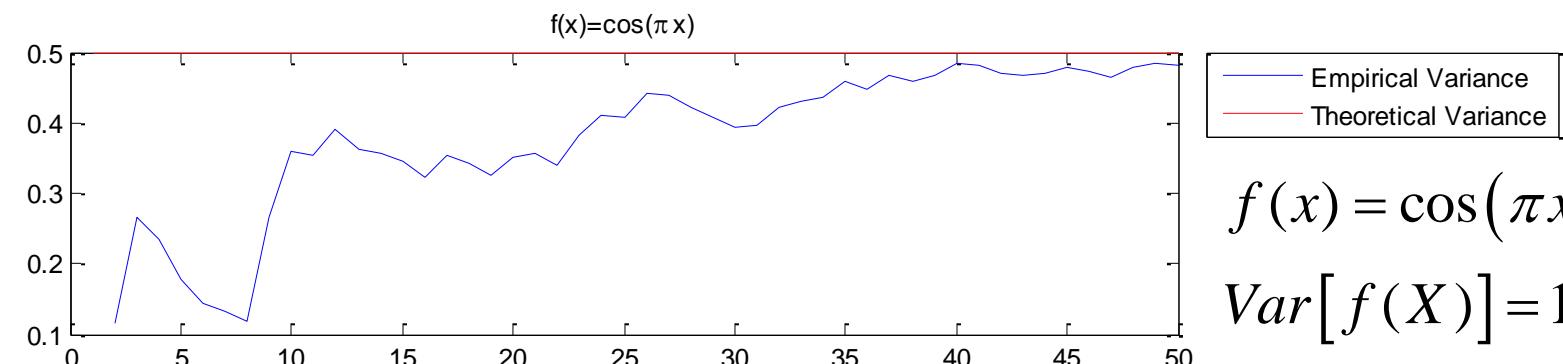


$$f(x) = x$$
$$\text{Var}[f(X)] = 1/12$$

MatLab Implementation



$$f(x) = x^2$$
$$\text{Var}[f(X)] = 4/45$$



$$f(x) = \cos(\pi x)$$
$$\text{Var}[f(X)] = 1/2$$



# Optimal Number of MC Samples

- We can compute the optimal number of MC samples for a desired accuracy of the estimator:

$$\Pr(|\hat{\mu}_N(f) - \mu(f)| \leq \varepsilon) = 1 - \alpha \Rightarrow \Pr\left(\frac{|\hat{\mu}_N(f) - \mu(f)|}{\sigma/\sqrt{N}} \leq \frac{\varepsilon\sqrt{N}}{\sigma}\right) = 1 - \alpha$$

- From the asymptotic properties  $\sqrt{N}(\hat{\mu}_N - \mu) \xrightarrow[N \rightarrow \infty]{D} \mathcal{N}(0, \sigma^2(f))$  we can estimate:

$$X_\alpha = \varepsilon \sqrt{\frac{N}{\sigma^2(f)}} \Rightarrow N = \left(\frac{X_\alpha}{\varepsilon}\right)^2 \sigma^2(f)$$

where:

$$X_\alpha = \Phi^{-1}\left(1 - \frac{\alpha}{2}\right), \text{ } \Phi^{-1} \text{ inverse cdf of } \mathcal{N}[0,1]$$

- Approximating the variance  $\sigma^2(f)$ , the number of needed  $N$  should satisfy:

$$\hat{\sigma}_N^2(f) \leq \frac{N\varepsilon^2}{X_\alpha^2}$$



# Optimal Number of MC Samples

- The condition  $\hat{\sigma}_N^2(f) \leq \frac{N\varepsilon^2}{X_\alpha^2}$  can be satisfied iteratively
  - 1. Start with  $n_1$  MC samples from  $X_1, \dots, X_{n_1} \sim \mathcal{U}[0,1]$
  - 2. If  $\hat{\sigma}_{n_1}^2(f) \leq \frac{n_1\varepsilon^2}{X_\alpha^2}$  then stop; otherwise
  - 3. Evaluate  $k_1 = \left\lfloor \frac{X_\alpha^2 \hat{\sigma}_{n_1}^2(f)}{\varepsilon^2} - n_1 \right\rfloor$  and generate  $k_1$  samples

$X_{n_1+1}, \dots, X_{n_1+k} \sim \mathcal{U}[0,1]$  where here  $\lfloor x \rfloor$  indicates the integer

part of x.



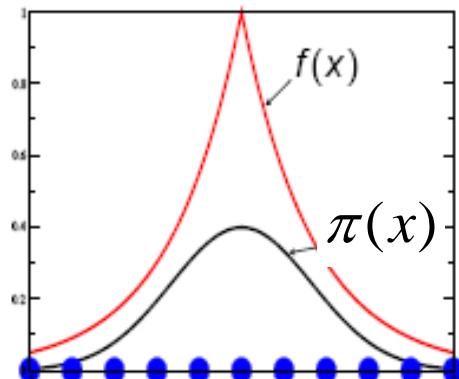
# Computing Expectations

- Expectations with respect to any distribution also involve high-dimensional integration:

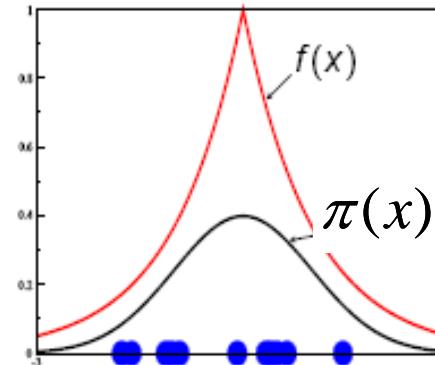
$$I = \int_D f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$$

If we can draw samples from  $\pi(\mathbf{x})$ , then we can use the estimator

$$\hat{I} = \frac{1}{N} \sum_{i=1}^N f(x_i), \text{ with Coeff. of Variat. } (\hat{I}) = \frac{\sqrt{\text{Var}(f(x))}}{\sqrt{N} \mathbb{E}_{\pi(x)}[f(x)]}$$



Riemann integration



MC integration

# Bayes Factor Approximation

- For the [CMBdata](#), consider 2 subsamples  $(x_1, \dots, x_n)$  and  $(y_1, \dots, y_n)$ .
- We consider that both come from distributions  $x_1, \dots, x_n \sim \mathcal{N}(\mu_x, \sigma^2)$  and  $y_1, \dots, y_n \sim \mathcal{N}(\mu_y, \sigma^2)$ .
- We want to decide if both means are the same, i.e. test  $H_0: \mu_x = \mu_y$ .
- We assume that the variance (error)  $\sigma^2$  is the same for both models and use a prior
$$\pi_\sigma(\sigma^2) = 1/\sigma^2$$
- The Bayes factor  $B_{10}^\pi$  can then be computed as follows:

$$B_{10}^\pi = \frac{\int \ell(\mu_x, \mu_y, \sigma^2 | \mathcal{D}_n) \pi(\mu_x, \mu_y) \pi_\sigma(\sigma^2) d\sigma^2 d\mu_x d\mu_y}{\int \ell(\mu, \sigma^2 | \mathcal{D}_n) \pi_\mu(\mu) \pi_\sigma(\sigma^2) d\sigma^2 d\mu}$$

- Note that the Bayes factor does not depend on the normalizing constant of  $\pi_\sigma(\sigma^2)$  and thus the use of an improper prior  $\pi_\sigma(\sigma^2) = 1/\sigma^2$  is fine.

J-M Marin and C. P. Robert, [Bayesian Core](#), Chapter 2

 Statistical Computing, University of Notre Dame, Notre Dame, IN, USA (Fall 2017, N. Zabaras)

# Bayes Factor Approximation

- We introduce a new parametrization by writing:

$$\mu_x = \mu - \xi, \mu_y = \mu + \xi$$

with the prior

$$\pi(\mu, \xi) = \pi_\mu(\mu)\pi_\xi(\xi)$$

This allows the same prior  $\pi_\mu(\mu)$  to be used in both  $H_0$  and the alternative  $H_1: \mu_x \neq \mu_y$ .

- We choose the improper prior  $\pi_\mu(\mu) = 1$  and  $\xi \sim \mathcal{N}(0, 1)$ .
- The Bayes factor can now be re-written as:

$$B_{10}^\pi = \frac{\int \ell(\mu_x, \mu_y, \sigma^2 | \mathcal{D}_n) \pi(\mu_x, \mu_y) \pi_\sigma(\sigma^2) d\sigma^2 d\mu_x d\mu_y}{\int \ell(\mu, \sigma^2 | \mathcal{D}_n) \pi_\mu(\mu) \pi_\sigma(\sigma^2) d\sigma^2 d\mu} = \\ = \frac{\int (\sigma^2)^{-n/2} \exp\left\{-\left(n(\mu - \xi - \bar{x})^2 + s_x^2\right)/2\sigma^2\right\} (\sigma^2)^{-n/2} \exp\left\{-\left(n(\mu + \xi - \bar{y})^2 + s_y^2\right)/2\sigma^2\right\} \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} (\sigma^2)^{-1} d\sigma^2 d\mu d\xi}{\int (\sigma^2)^{-n/2} \exp\left\{-\left(n(\mu - \bar{x})^2 + s_x^2\right)/2\sigma^2\right\} (\sigma^2)^{-n/2} \exp\left\{-\left(n(\mu - \bar{y})^2 + s_y^2\right)/2\sigma^2\right\} (\sigma^2)^{-1} d\sigma^2 d\mu}$$

# Bayes Factor Approximation

- To simplify the Bayes factor, let us define:

$$S^2 = \frac{s_x^2}{n} + \frac{s_y^2}{n} = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n} + \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n}$$

- We can then write:

$$B_{10}^\pi = \frac{\int (\sigma^2)^{-n-1} e^{-\frac{n}{2\sigma^2}((\mu-\xi-\bar{x})^2 + (\mu+\xi-\bar{y})^2 + S^2)} \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} d\sigma^2 d\mu d\xi}{\int (\sigma^2)^{-n-1} e^{-\frac{n}{2\sigma^2}((\mu-\bar{x})^2 + (\mu-\bar{y})^2 + S^2)} d\sigma^2 d\mu}$$

- Performing the integrals in  $\sigma^2$ :

$$B_{10}^\pi = \frac{\int \left[ (\mu-\xi-\bar{x})^2 + (\mu+\xi-\bar{y})^2 + S^2 \right]^{-n} \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} d\mu d\xi}{\int \left[ (\mu-\bar{x})^2 + (\mu-\bar{y})^2 + S^2 \right]^{-n} d\mu}$$

- Lets now integrate in  $\mu$ . Start with the denominator. Note that:

$$(\mu-\bar{x})^2 + (\mu-\bar{y})^2 = 2 \left( \mu - \frac{\bar{x} + \bar{y}}{2} \right)^2 + \frac{(\bar{x} - \bar{y})^2}{2}$$

# Bayes Factor Approximation

$$B_{10}^{\pi} = \frac{\int \left[ (\mu - \xi - \bar{x})^2 + (\mu + \xi - \bar{y})^2 + S^2 \right]^{-n} \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} d\mu d\xi}{\int \left[ (\mu - \bar{x})^2 + (\mu - \bar{y})^2 + S^2 \right]^{-n} d\mu}$$

- So the denominator takes the form:

$$\int \left[ (\mu - \bar{x})^2 + (\mu - \bar{y})^2 + S^2 \right]^{-n} d\mu = 2^{-n} \int \left[ \left( \mu - \frac{\bar{x} + \bar{y}}{2} \right)^2 + \frac{(\bar{x} - \bar{y})^2}{4} + \frac{S^2}{2} \right]^{-n} d\mu$$

$$\frac{(\bar{x} - \bar{y})^2}{4} + \frac{S^2}{2}$$

- With the definitions  $\sigma^2 = \frac{4}{2n-1}$  and  $\nu = 2n-1$ , we can simplify as:

$$\int \left[ (\mu - \bar{x})^2 + (\mu - \bar{y})^2 + S^2 \right]^{-n} d\mu = (2\sigma^2\nu)^{-n} \int \left[ 1 + \left( \mu - \frac{\bar{x} + \bar{y}}{2} \right)^2 / \sigma^2\nu \right]^{-(\nu+1)/2} d\mu = (2\sigma^2\nu)^{-n} \sigma \frac{\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}{\Gamma\left(\frac{\nu+1}{2}\right)} =$$

$$\frac{\sqrt{\nu\pi}}{(2\nu)^n \sigma^{2n-1} \Gamma\left(\frac{\nu+1}{2}\right) \Gamma\left(\frac{\nu}{2}\right)} = \frac{\sqrt{\pi}}{2^n \Gamma\left(\frac{\nu+1}{2}\right) \Gamma\left(\frac{\nu}{2}\right)} \frac{1}{\left( \left( \frac{\bar{x} - \bar{y}}{2} \right)^2 + \frac{S^2}{2} \right)^{n-1/2}}$$

Common Factor  
in the denomin.  
and numer.of  
 $B_{10}^{\pi}$

Here we use the normalizing constant  
of the [t-distribution](#)

$$p(\theta) = \frac{\Gamma((\nu+1)/2)}{\Gamma(\nu/2)\sqrt{\nu\pi\sigma}} (1 + \frac{1}{\nu}(\frac{\theta-\mu}{\sigma})^2)^{-(\nu+1)/2}$$



# Bayes Factor Approximation

□ Thus for the normal case with

$$x_1, \dots, x_n \sim \mathcal{N}(\mu + \xi, \sigma^2)$$

$$y_1, \dots, y_n \sim \mathcal{N}(\mu - \xi, \sigma^2) \quad \text{and} \quad H_0 : \xi = 0$$

under prior  $\pi(\mu, \sigma^2) = 1/\sigma^2$  and  $\xi \sim \mathcal{N}(0, 1)$

$$B_{01}^\pi = \frac{1}{B_{10}^\pi} = \frac{\left[ (\bar{x} - \bar{y})^2 + 2S^2 \right]^{-n+1/2}}{\int \left[ (2\xi + \bar{x} - \bar{y})^2 + 2S^2 \right]^{-n+1/2} e^{-\xi^2/2} d\xi / \sqrt{2\pi}}$$

For [CMBdata](#), we simulate  $\xi_1, \dots, \xi_{1000} \sim \mathcal{N}(0, 1)$  and approximate  $B_{01}^\pi$  with (for one simulation that we run)

$$\widehat{B}_{01}^\pi = \frac{[(\bar{x} - \bar{y})^2 + 2S^2]^{-n+1/2}}{\sum_{i=1}^{1000} [(2\xi_i + \bar{x} - \bar{y})^2 + 2S^2]^{-n+1/2} / 1000} = 43.3309$$

when  $\bar{x} = 0.0888$ ,  $\bar{y} = 0.1078$ ,  $S^2 = 0.00875$ ,  $n = 100$ . Thus  $H_0$  is much more likely with the data available.

# Precision Evaluation in Bayes Factor

- For an estimator,  $I = \int h(x)\pi(x)dx$  with  $x_i \sim \pi(x)$ , the variance can be computed (in terms of the empirical mean) as

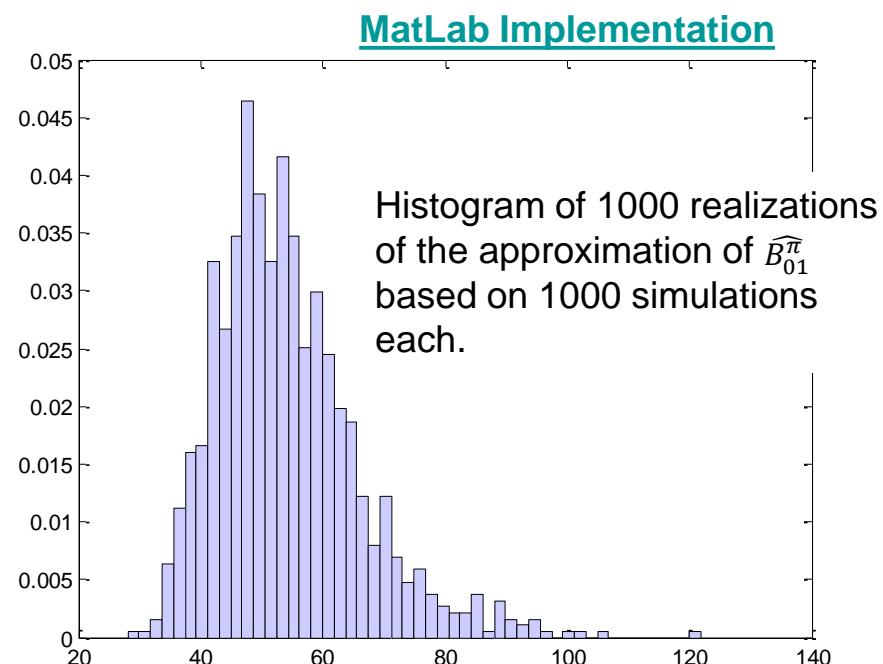
$$\nu_N = \frac{1}{N} \frac{1}{N-1} \sum_{j=1}^N [h(x_j) - \bar{h}_N]^2$$

and for  $N$  large,

$$\{\bar{h}_N - \mathbb{E}_f[h(X)]\}/\sqrt{\nu_N} \sim \mathcal{N}(0,1)$$

We can thus find the variability in the Bayes factor estimation.

We construct a convergence test and of confidence bounds on the approximation of  $\widehat{B}_{01}^\pi$



# Example (Cauchy-Normal)

- For estimating a normal mean, a robust prior is a Cauchy prior

$$x \sim \mathcal{N}(\theta, 1), \quad \theta \sim \mathcal{C}(0, 1)$$

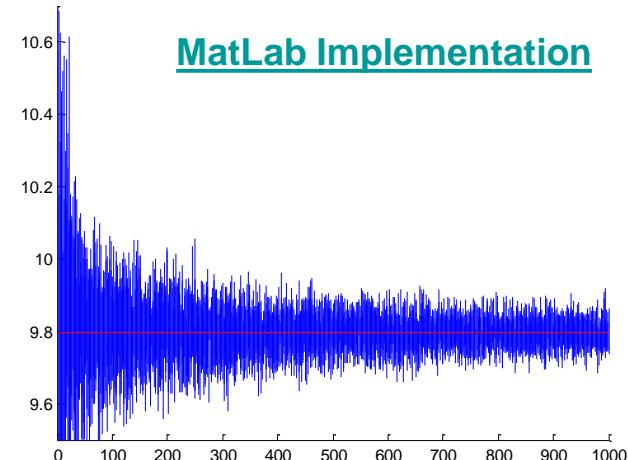
Under squared error loss, the posterior mean is given as:

$$\delta^\pi(x) = \frac{\int_{-\infty}^{+\infty} \frac{\theta}{1+\theta^2} e^{-(x-\theta)^2/2} d\theta}{\int_{-\infty}^{+\infty} \frac{1}{1+\theta^2} e^{-(x-\theta)^2/2} d\theta}$$

Form of  $\delta^\pi$  suggests simulating i.i.d. variables  $\theta_1, \dots, \theta_m \sim \mathcal{N}(x, 1)$  and calculate

$$\widehat{\delta}_m^\pi(x) = \frac{\sum_{i=1}^m \frac{\theta_i}{1+\theta_i^2}}{\sum_{i=1}^m \frac{1}{1+\theta_i^2}}$$

LLN implies  $\widehat{\delta}_m^\pi(x) \rightarrow \delta^\pi(x)$  as  $m \rightarrow \infty$



# ***Random Variable Generation***



# The Problem of Interest

---

- $x \equiv$  vector of  $k$ -random variables.
- $\pi(x) \equiv$  distribution function. In the context of Bayesian analysis, this will be the posterior distribution.
- Goal is to evaluate  $\mathbb{E}_\pi \{f(x)\}$

$$\mathbb{E}_\pi \{f(x)\} \equiv \frac{\int f(x)\pi(x)dx}{\int \pi(x)dx}$$

- $\int \pi(x)dx$  is the normalizing constant.
- In most algorithms to be discussed in this and the following lectures, we do not need to know this normalization constant.

# Monte Carlo Integration: Review

- Monte Carlo integration evaluates  $\mathbb{E}_\pi \{f(\mathbf{x})\}$  by drawing samples  $\{\mathbf{x}_t; t = 1, \dots, N\}$  from  $\pi(\mathbf{x})$  and then approximating

$$\mathbb{E}_\pi (f(\mathbf{x})) \equiv \frac{1}{N} \sum_{t=1}^N f(\mathbf{x}_t), \mathbf{x}_t \text{ i.i.d.}$$

- That is, the population mean of  $f(X)$  is estimated by a sample mean.
- When  $\mathbf{x}_t$  are i.i.d., LLN ensures that the approximation can be accurate as desired by increasing  $N$ .
- $N$  is the number of samples we use to approximate  $\mathbb{E}_\pi (f(\mathbf{x}))$
- As we have seen, the convergence rate is  $\frac{1}{\sqrt{N}}$ .



# *Sampling From an Arbitrary Distribution*

- Let us assume that we need to estimate

$$\mathbb{E}_{\pi}(f(\mathbf{x})) = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}, \mathbf{x} \in \mathcal{X}$$

- The basic steps in MC are the following:

- Generate i.i.d samples

$$\mathbf{x}_i \sim \pi(\mathbf{x}) \leftarrow$$

**But how do  
we sample from  
an arbitrary  
distribution?**

- Evaluate

$$\hat{f} = \frac{1}{N} \sum_{i=1}^N f(\mathbf{x}_i)$$

- We have shown that this estimator is unbiased:

$$\mathbb{E}(\hat{f}) = \mathbb{E}_{\pi}(f(\mathbf{x}))$$

- The variance of the estimator is:

$$Var(\hat{f}) = \frac{Var(f(\mathbf{x}))}{N}$$



# Sampling From an Arbitrary Distribution

- Consider an arbitrary probability density  $\pi(\mathbf{x})$
- Monte Carlo approximation is given by

$$\hat{\pi}_N(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \delta_{X^{(i)}}(\mathbf{x}), \text{ where } X^{(i)} \stackrel{i.i.d.}{\sim} \pi$$

- For any function  $f : \mathcal{X} \rightarrow \mathbb{R}$

$$\mathbb{E}_{\hat{\pi}_N}(f) = \frac{1}{N} \sum_{i=1}^N f(X^{(i)}) \cong \mathbb{E}_{\pi}(f)$$

But how do  
we sample from  
an arbitrary  
distribution?



or more precisely:

$$\mathbb{E}_X[\mathbb{E}_{\hat{\pi}_N}(f)] = \mathbb{E}_{\pi}(f) \text{ and } \text{Var}_X(\mathbb{E}_{\hat{\pi}_N}(f)) = \frac{\text{Var}_{\pi}(f)}{N}$$



# *Sampling Uniform Random Variables*

---

- Most sampling algorithms rely on generating uniform random variables in  $[0, 1]$ .
- We only have algorithms for generating pseudo-random numbers which look like they are i.i.d.  $\mathcal{U}[0, 1]$ .
- There are several standard uniform random number generators available.



# Sampling From a Discrete Distribution

- Consider  $\mathcal{X} = \{1, 2, 3\}$  and

$$\pi(X = 1) = 1/6, \pi(X = 2) = 2/6, \pi(X = 3) = 1/2$$

- Define the cdf of  $X$  for  $x \in [0, 3]$  as

$$F_X(x) = \sum_{i=1}^3 \pi(X = i) \mathbb{I}(i \leq x)$$

and its inverse for  $u \in [0, 1]$

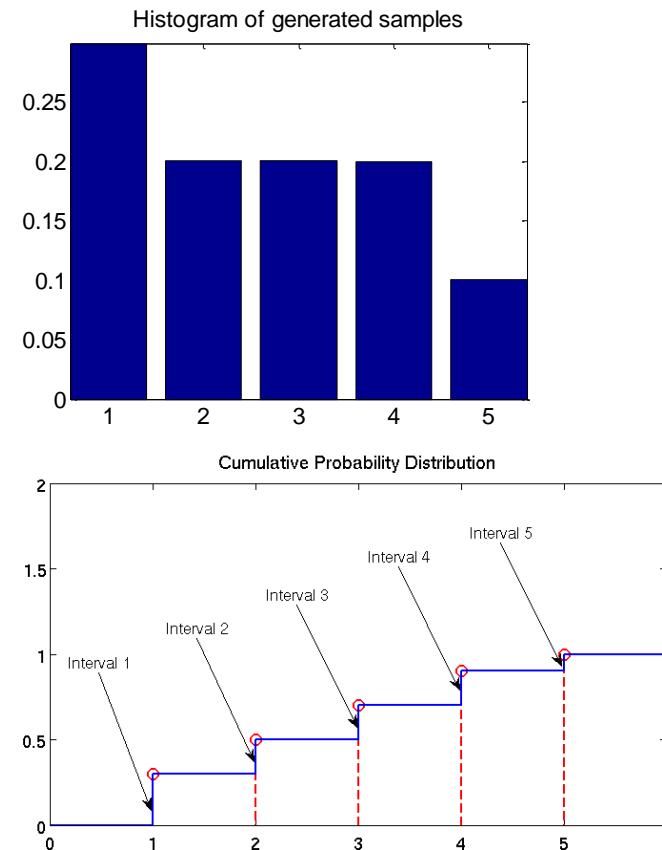
$$F_X^{-1}(u) = \inf \{x \in \mathcal{X}; F_X(x) \geq u\}$$

# Sampling from a Discrete Distribution

- Consider the distribution of a discrete random variable
- To sample from this discrete distribution, sample  $u \sim U[0, 1]$
- Find  $X = F_X^{-1}(u)$
- The probability of  $u$  falling in the vertical interval  $i$  is equal to the probability  $\pi(X = i)$ .
- We will show next:

$$\Pr[x = k] = \Pr[P_{k-1} < u \leq P_k],$$

$$\text{where } P_k = \sum_{j=1}^k p_j$$



[MatLab Implementation](#)

# Sampling from a Discrete Distribution

- Consider a discrete random variable  $X$  taking values  $\{1, 2, 3, \dots, \infty\}$  with probability  $\Pr[X = k] = p_k$ , with  $\sum_{i=1}^{\infty} p_i = 1$
- We want by sampling  $u \sim \mathcal{U}[0, 1]$  to generate samples  $x \sim p(X)$
- Let  $P_k \in [0, 1]$ , s.t.  $P_k = \Pr[X \leq k] = \sum_{j=1}^k p_j$
- Draw  $u \sim \mathcal{U}[0, 1]$ .

- Then set  $x = k$ , where  $k = \min \{k \in \{1, 2, 3, \dots, +\infty\} \text{ such that } u \leq P_k\}$

Note :  $\Pr[x = k] = \Pr[P_{k-1} < u \leq P_k] = \int_{P_{k-1}}^{P_k} 1 du$

$$= P_k - P_{k-1} = \sum_{j=1}^k p_j - \sum_{j=1}^{k-1} p_j = p_k, \text{i.e. the desired distribution}$$

# **Sampling from a Continuous Distribution**

- Assume the distribution has a density, then the CDF takes the form:

$$\text{Note : } F_X(x) = \Pr(X \leq x) = \int_{-\infty}^{+\infty} \pi(u) \mathbb{I}(u \leq x) du = \int_{-\infty}^x \pi(u) du$$

- **Algorithm:**  $u \sim \mathcal{U}[0,1]$  and then set  $X = F_X^{-1}(u)$ . Do we have  $X \sim \pi$ ?

*Proof :*  $\Pr(X \leq x) = \Pr(F_X^{-1}(u) \leq x) = \Pr(u \leq F_X(x))$  since  $F_X$  is non-decreasing

$$= \int_0^1 \mathbb{I}(u \leq F_X(x)) du \text{ since } u \sim \mathcal{U}[0,1]$$

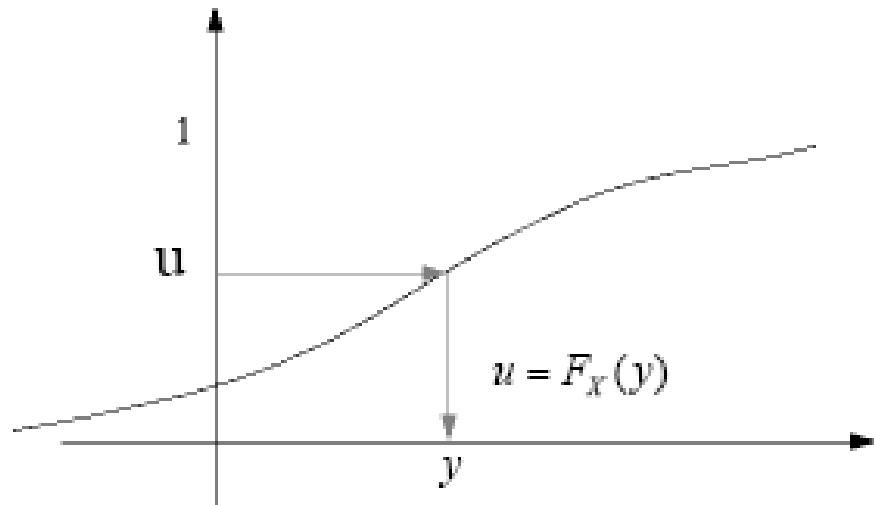
$= F_X(x)$  which is precisely the CDF of  $\pi$



# Reverse Sampling for Continuous Distributions

- Simplest method
- Used frequently for non-uniform random number generation
- Sample a random number  $u$  from  $\mathcal{U}[0, 1]$
- Set  $x=y$ , where  $u=F_X(y)$  (i.e.  $x=F_X^{-1}(u)$ )
- Simple, but you need the functional form of  $F$ .

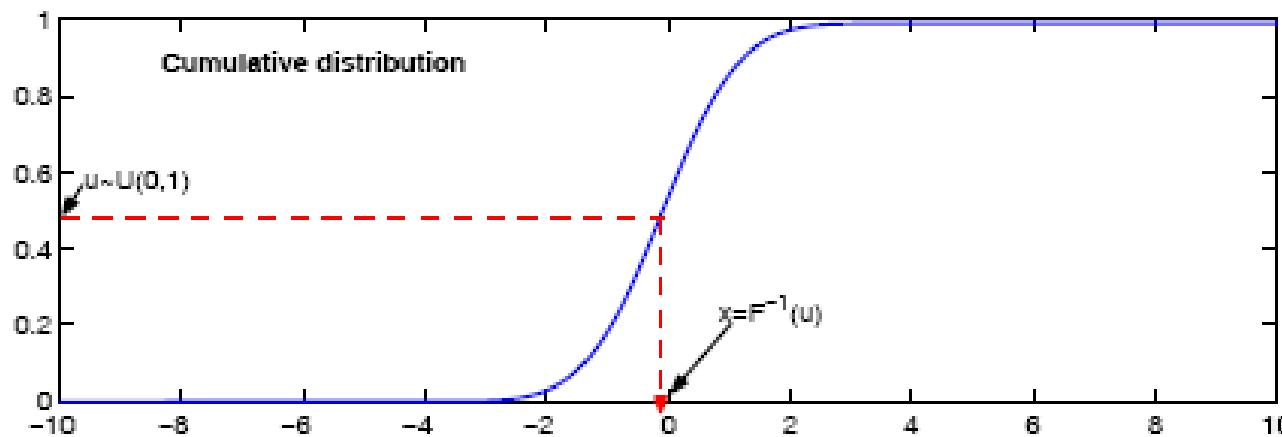
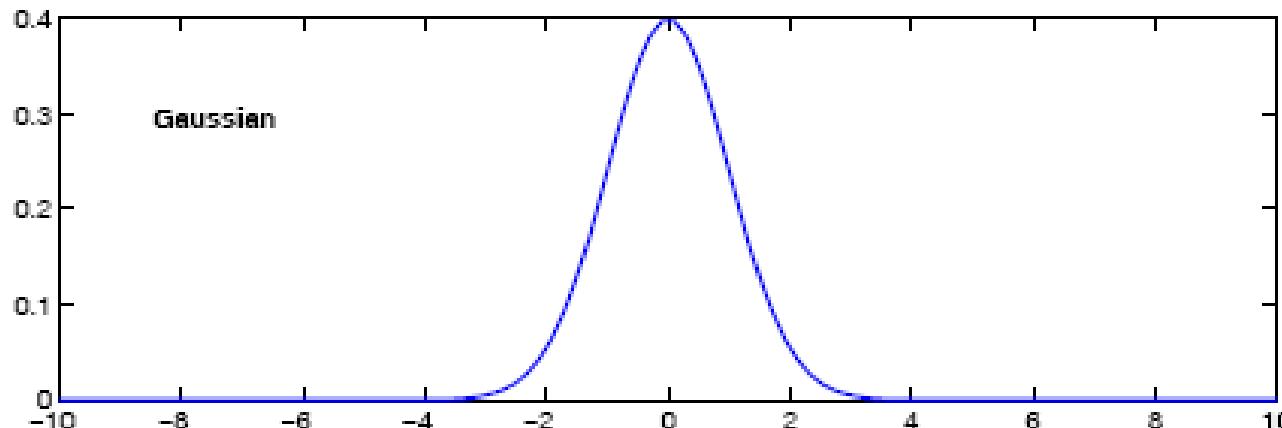
$$\begin{aligned} \text{Indeed : } \Pr[x \leq y] &= \Pr[u \leq F_X(y)] \\ &= \int_0^{F_X(y)} 1 du = F_X(y) \end{aligned}$$



Sampling using the inverse of the CDF

# Sampling from a Continuous Distribution

- The distribution and the CDF of a normal distribution



# Inverse Method: Exponential Distribution

- Consider the exponential distribution with parameter  $\lambda$ :

$$\pi(x) = \lambda e^{-\lambda x} \mathbb{I}_{[0, +\infty)}$$

- The CDF of  $X$  is:

$$F_X(x) = \int_{-\infty}^x \pi(z) dz = \begin{cases} 0 & \text{if } x \leq 0 \\ 1 - e^{-\lambda x} & \text{if } x > 0 \end{cases}$$

- Thus the inverse CDF is:

$$1 - e^{-\lambda x} = u \Leftrightarrow x = -\frac{1}{\lambda} \log(1 - u) = F_X^{-1}(u)$$

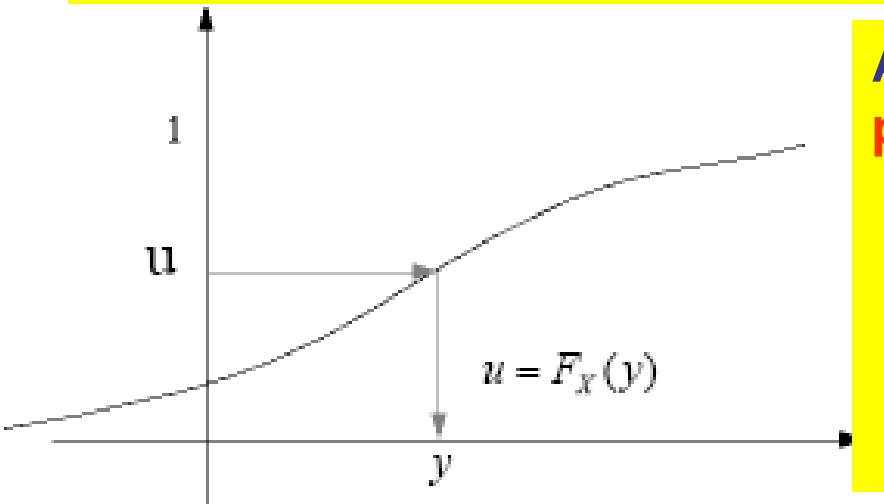
- Inverse method:  $u \sim \mathcal{U}[0, 1]$ , then  $X = -\frac{1}{\lambda} \log(1 - U) \sim \pi \equiv \text{Exp}(\lambda)$

- Similarly if  $u \sim \mathcal{U}[0, 1]$ , then  $X = -\frac{1}{\lambda} \log U \sim \text{Exp}(\lambda)$ . Indeed:

$$\Pr(X \leq x) = \Pr(-\log U \leq \lambda x) = \Pr(U \geq e^{-\lambda x}) = 1 - e^{-\lambda x} \Rightarrow X \sim \text{Exp}(\lambda)$$

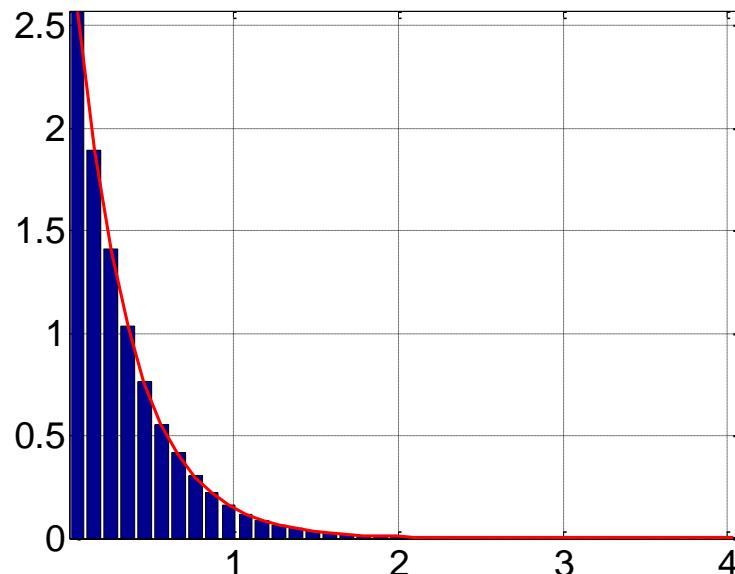
# Inverse Method: Exponential Distribution

Exponential( $\lambda$ ):  $\pi(x) = \lambda e^{-\lambda x}$ ,  $F_X(y) = 1 - e^{-\lambda y}$ ,  $x = -\ln(1-u)/\lambda$



As  $F(y)$  is area under  $\pi(y)$ ,  $y = F^{-1}(u)$  prescribes that

- Choose  $u=(0,1]$ , then find value  $y$  that has that *fraction  $u$  of area to the left of  $y$ , or  $u=F(y)$*
- Return that value of  $x=y$ .



[MatLab Implementation](#)



# Inverse Method: Example

- Consider  $N$  i.i.d. random variables  $X_i \sim f_X$  (with CDF  $F_X$ ). We are interested to sample from the following distribution:

$$Z = \max(X_1, \dots, X_N)$$

- You can use an inverse approach as follows:

$$\begin{aligned} F_Z(z) &= \Pr(X_1 \leq z, \dots, X_N \leq z) \\ &= \prod_{i=1}^N \Pr(X_i \leq z) = [F_X(z)]^N \end{aligned}$$

- Thus for any  $U \sim \mathcal{U}[0, 1]$ , we can sample  $Z \sim f_Z$  as follows:

$$Z = F_Z^{-1}(U) = F_X^{-1}(U^{1/N})$$



# **Inverse Method: Limitations**

---

- Practical and simple method for univariate distributions.
- Limited to cases where the inverse cdf has an analytical form that can be tabulated.
- Its practical use is very limited.



# Transformation Methods

- We have seen an example of the transformation method:

$$\text{If } u \sim \mathcal{U}[0,1] \Rightarrow X = -\frac{1}{\lambda} \log(1-U) \sim \mathcal{Exp}(\lambda)$$

$$\text{If } u \sim \mathcal{U}[0,1] \Rightarrow X = -\frac{1}{\lambda} \log U \sim \mathcal{Exp}(\lambda)$$

- We use the fact that  $\pi$  is related to other transformations easier to sample from.
- These methods are specific to some distributions, e.g.

If  $X_i \sim \mathcal{Exp}(1)$  then :  $Y = 2 \sum_{i=1}^{\nu} X_i \sim \chi_{2\nu}^2$ ,  $Y = \beta \sum_{i=1}^{\alpha} X_i \sim \mathcal{Gamma}(\alpha, \beta)$ ,

$$Y = \frac{\sum_{i=1}^{\alpha} X_i}{\sum_{i=1}^{\alpha+\beta} X_i} \sim \mathcal{Be}(\alpha, \beta)$$



# Transformation Methods

---

- Starting with samples from the uniform distribution, these transformations are very simple to apply:

$$Y = -2 \sum_{i=1}^{\nu} \log(U_i) \sim \chi_{2\nu}^2,$$

$$Y = -\beta \sum_{i=1}^{\alpha} \log(U_i) \sim \text{Gamma}(\alpha, \beta)$$

$$Y = \frac{\sum_{i=1}^{\alpha} \log(U_i)}{\sum_{i=1}^{\alpha+\beta} \log(U_i)} \sim \text{Be}(\alpha, \beta)$$

- With this approach, we cannot generate Gamma random variables with a non-integer shape parameter  $\alpha$ .
- Cannot generate  $\chi_1^2$  which will give us a  $\mathcal{N}(0, 1)$  variable.

# Box Muller Algorithm to Sample Gaussians

- Consider  $(x,y)$  coordinates  $X_1 \sim \mathcal{N}(0,1)$  and  $X_2 \sim \mathcal{N}(0,1)$ , then the polar coordinates  $(R,\theta)$  of this point are independent and distributed according to

$$R^2 = X_1^2 + X_2^2 \sim \text{Exp}\left(\frac{1}{2}\right)$$
$$\theta \sim \mathcal{U}[0, 2\pi]$$

If  $X_1 \sim \mathcal{N}(0,1), \dots, X_n \sim \mathcal{N}(0,1)$ ,  
then  $X_1^2 + \dots + X_n^2 \sim \chi_n^2$ . For n=2,  
 $x_{n=2}^2 \sim \text{Exp}\left(\frac{1}{2}\right)$ .

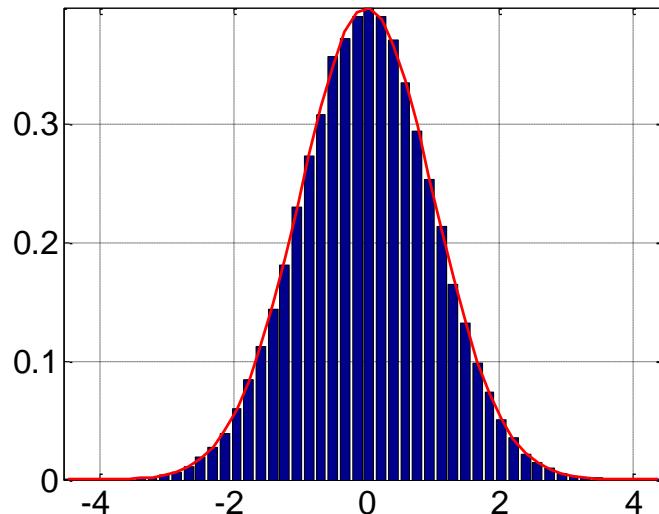
- It is simple to simulate  $R = \sqrt{-2 \log(U_1)}$  and  $\theta \sim 2\pi U_2$ , where

$U_1 \sim \mathcal{U}[0,1]$  and  $U_2 \sim \mathcal{U}[0,1]$ . Then

$$X_1 = R \cos \theta = \sqrt{-2 \log(U_1)} \cos(2\pi U_2)$$

$$X_2 = R \sin \theta = \sqrt{-2 \log(U_1)} \sin(2\pi U_2)$$

- By construction  $X_1$  and  $X_2$  are two independent  $\mathcal{N}(0, 1)$  random variables. A plot of one of them is shown here.



MatLab Implementation

# Box Muller Algorithm to Sample Gaussians

- Let us see a simpler approach to sample from a spherical Gaussian. We sample  $z_1, z_2$  uniformly from a unit circle:  $p(z_1, z_2) = \frac{1}{\pi} \mathbb{I}(z \text{ inside unit circle})$ . This can be done by sampling uniformly on the cube  $[-1, 1]^2$  and disregarding the samples outside the circle.
- Consider polar coordinates  $(r, \theta)$ . You can show that  $\left| \frac{\partial(z_1, z_2)}{\partial(r, \theta)} \right| = \left| \frac{\partial(r \cos \theta, r \sin \theta)}{\partial(r, \theta)} \right| = r$ .
- Define  $y_1 = (-2 \ln r^2)^{1/2} \cos \theta, y_2 = (-2 \ln r^2)^{1/2} \sin \theta$ . With simple algebra you can show that  $\left| \frac{\partial(y_1, y_2)}{\partial(r, \theta)} \right| = -\frac{2}{r}$ .
- The distribution of  $y_1, y_2$  can now be computed as:

$$p(y_1, y_2) = p(z_1, z_2) \left| \frac{\partial(z_1, z_2)}{\partial(r, \theta)} \right| \left| \frac{\partial(y_1, y_2)}{\partial(r, \theta)} \right|^{-1} = \frac{1}{\pi} r \frac{r}{2} = \frac{r^2}{2\pi} = \frac{1}{2\pi} \exp\left(-\frac{1}{2}(y_1^2 + y_2^2)\right)$$

- Thus we obtain the desired result:

$$p(y_1, y_2) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}y_1^2\right) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}y_2^2\right)$$



# *Sampling from the Bivariate Normal Density*

---

To sample from the bivariate density,

- 1) Generate two, uncorrelated, standard normal variates,  $z_1$  and  $z_2$ .

$$z \sim \mathcal{N}(\mu = 0, \sigma^2 = 1)$$

- 2) Compute the correlated  $X_1$  and  $X_2$

$$X_1 = \mu_1 + \sigma_1 z_1$$

$$X_2 = \mu_2 + \sigma_2 \left[ z_1 \rho + z_2 \sqrt{1 - \rho^2} \right]$$

- 3)  $X_1$  and  $X_2$  will have means  $\mu_1$  and  $\mu_2$ , standard deviations  $\sigma_1$  and  $\sigma_2$ , and correlation  $\rho$ .



# *Sampling from the Bivariate Normal Density*

---

Example 1. data are sampled from the standard bivariate normal distribution

Example 2. data are sampled from the bivariate normal distribution with

Mean vector                     $\{\mu_1, \mu_2\} = \{-1, 1\}$

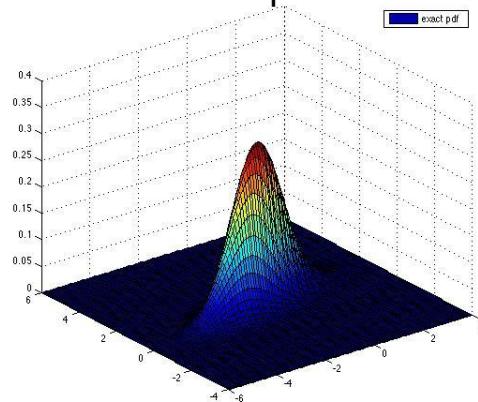
Covariance matrix               $\Sigma = \begin{bmatrix} 1.44 & 0.3 \\ 0.3 & 0.25 \end{bmatrix}$



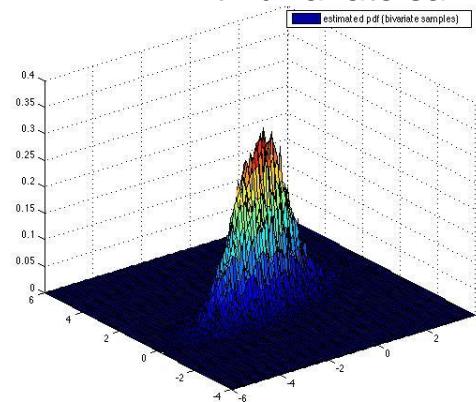
# Sampling from the Bivariate Normal Density

Validation of the algorithm

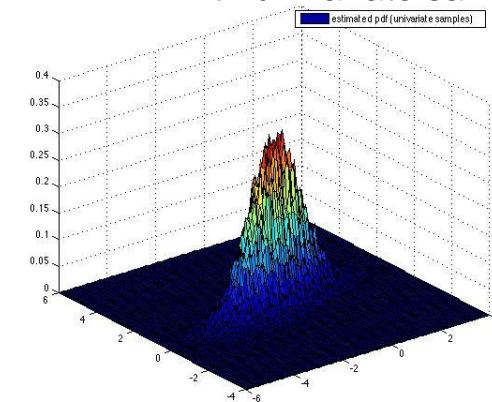
Exact pdf



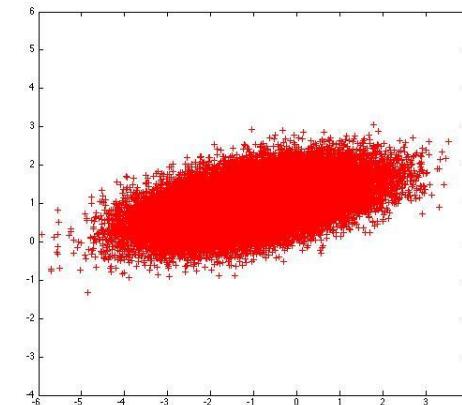
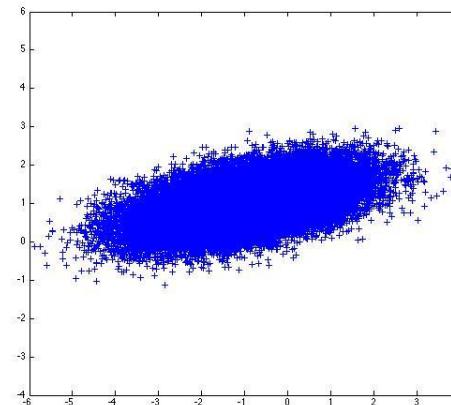
PDF with bivariate samples



PDF with univariate samples



The pdf generated from samples compared with the true standard bivariate Gaussian pdf

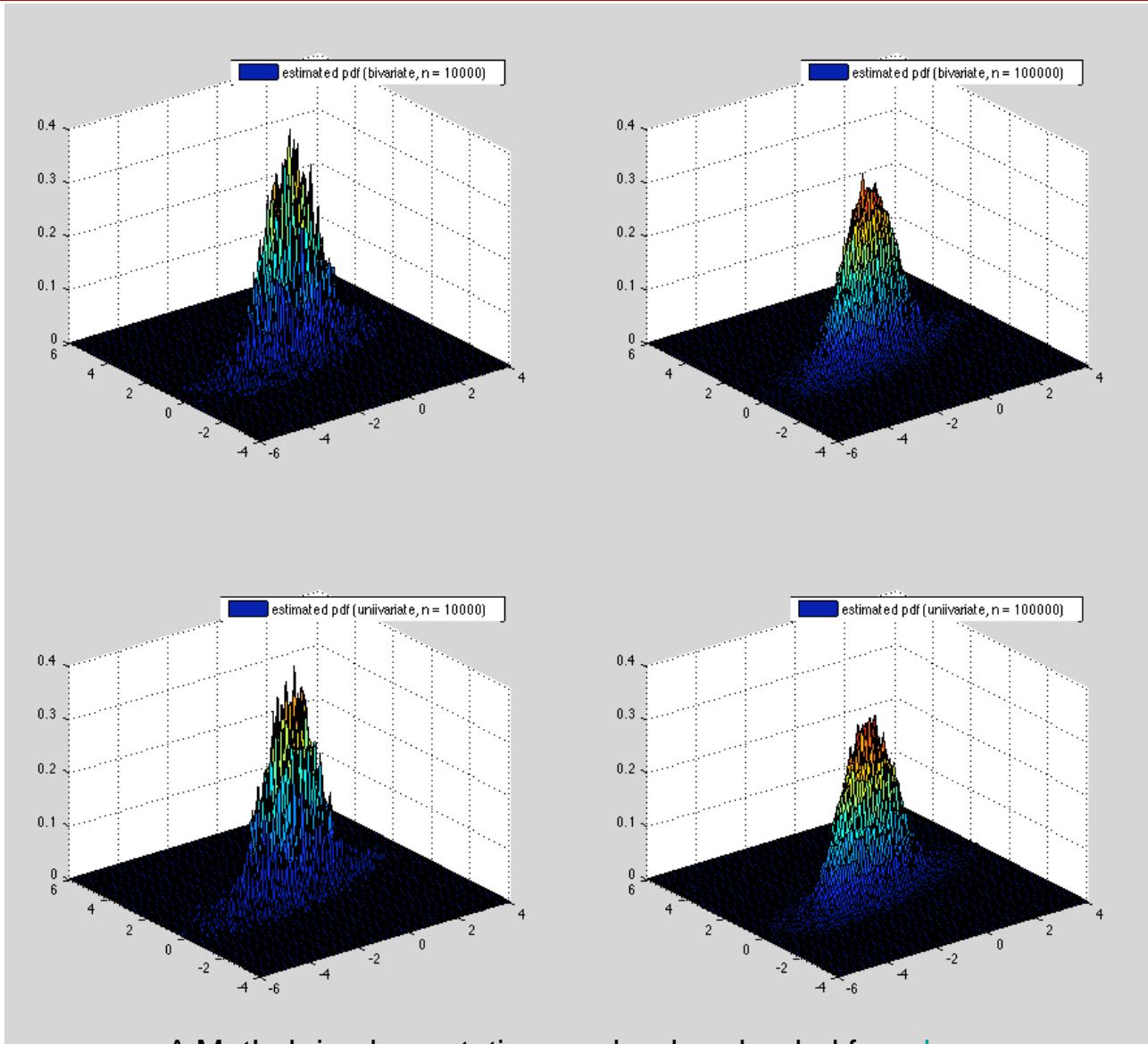


A MatLab implementation can be downloaded from [here](#).

Statistical Computing, University of Notre Dame, Notre Dame, IN, USA (Fall 2017, N. Zabaras)



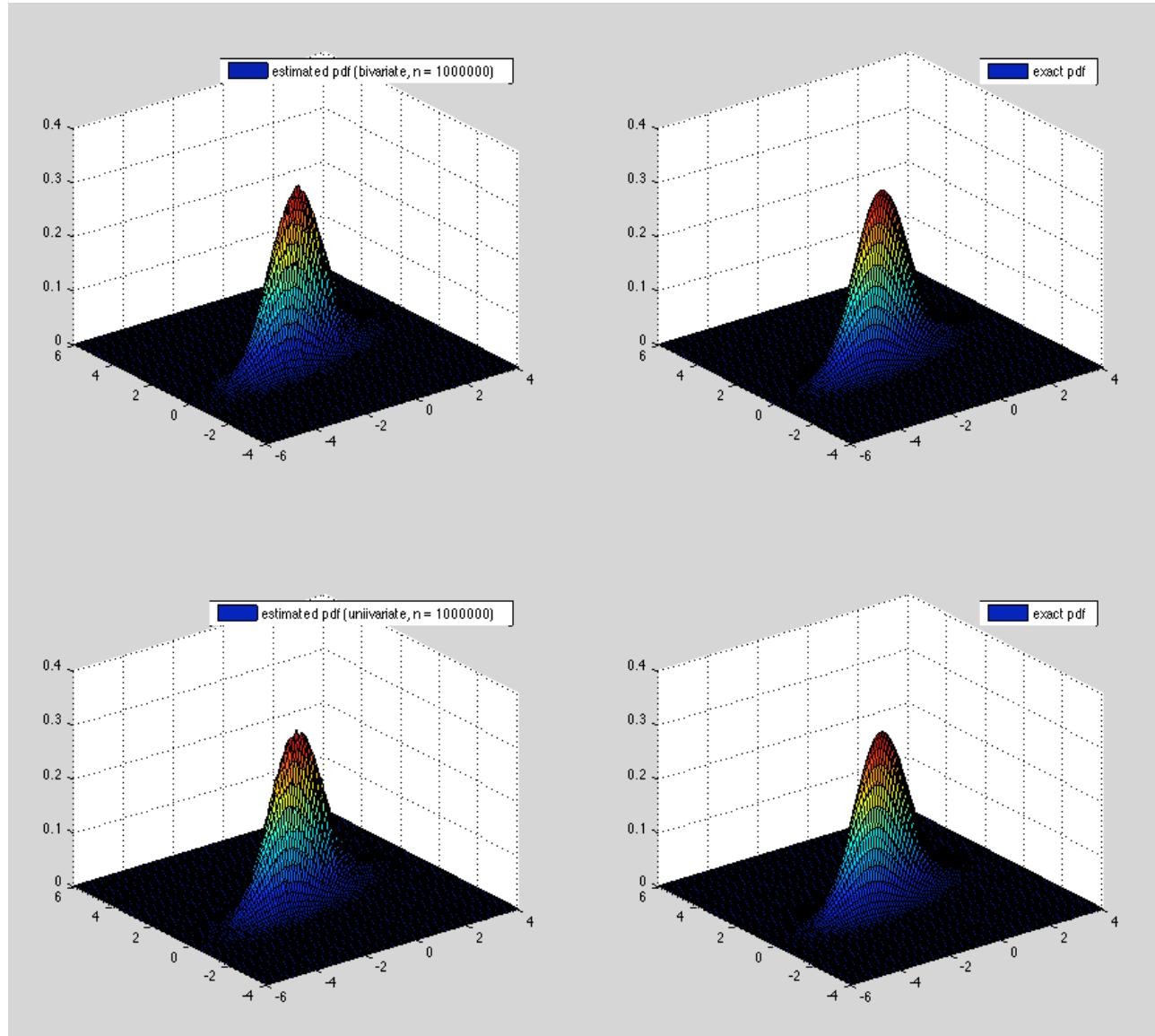
# Sampling from the Bivariate Normal Density



A MatLab implementation can be downloaded from [here](#).



# Sampling from the Bivariate Normal Density



A MatLab implementation can be downloaded from [here](#).



# Sampling from the Multivariate Gaussian

- In principle, the same approach can work:
  - Sample  $\mathbf{u} \sim \mathcal{U}[0, 1]^d$
  - Set  $\mathbf{x} = F_X^{-1}(\mathbf{u})$
- How about for the multivariate Gaussian:  $\mathcal{N}(\mu, \Sigma)$ 
  - Draw  $z_i \sim \mathcal{N}(0, 1)$  i.i.d.
  - Introduce the Cholesky decomposition of  $\Sigma$ :  $\Sigma = SS^T$
  - Set

$$\mathbf{x} = \mu + S\mathbf{z}$$

$$\begin{aligned} \text{Indeed : } \mathbb{E}\left[\left(\mathbf{x} - \mu\right)\left(\mathbf{x} - \mu\right)^T\right] &= \mathbb{E}\left[S\mathbf{z}\mathbf{z}^T S^T\right] = \\ &= S\mathbb{E}\left[\mathbf{z}\mathbf{z}^T\right]S^T = SS^T = \Sigma \end{aligned}$$