Introduction to Monte Carlo in Finance

1 - Background and Basic Ideas

Giovanni Della Lunga

WORKSHOP IN QUANTITATIVE FINANCE

Bologna - May 2-3, 2019

Outline

1. Background
   - Models in Continuous Time
   - Why we need Monte Carlo Methods
   - What is Monte Carlo?

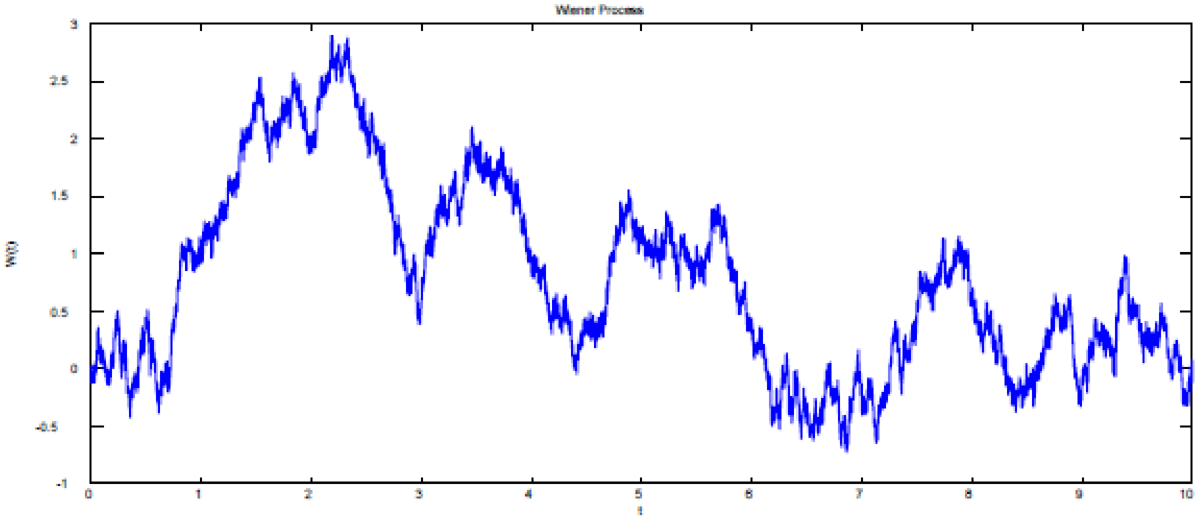
2. Basic Monte Carlo
   - Numerical Integration of SDE
   - The Brownian Bridge

Subsection 1

Models in Continuous Time

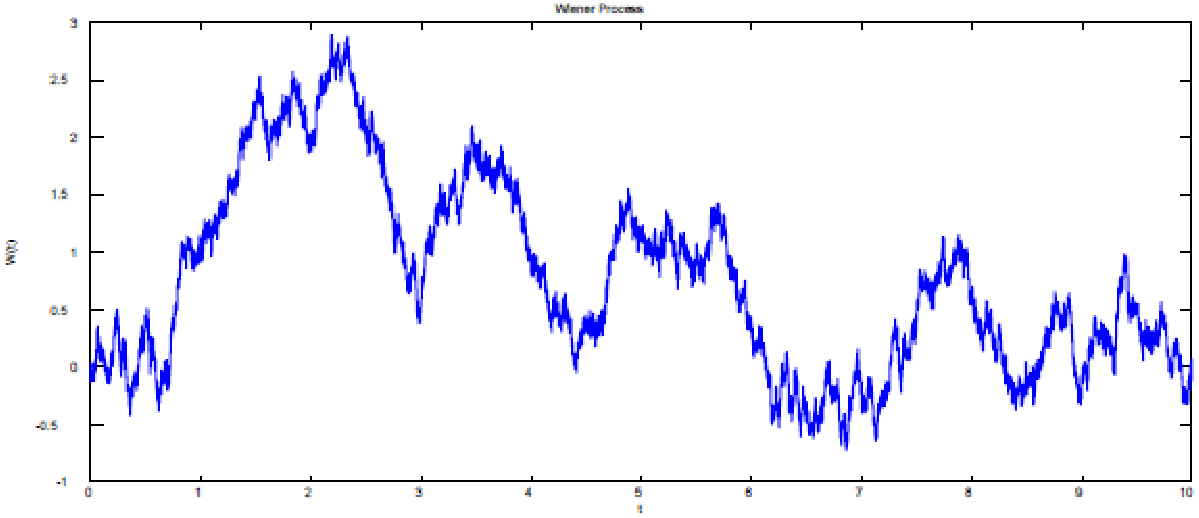
Models in Continuous Time

- We begin with some oversimplified rules of stochastic calculus;
- The very erratic path which is a specific feature of the Brownian motion is in general associated with the observation that the phenomenon, although very disordered, has a certain time homogeneity, i.e. the origin of time does not have importance to describe the time evolution.
- These properties underly the next definition...



Models in Continuous Time

- We define a stochastic process W_t called the *Standard Brownian Motion* or *Wiener Process* having the following properties:
  - For each $h > 0$, the increment $W(t+h) - W(t)$ has a $N(0, h)$ distribution and is independent of all preceding increments $W(u) - W(v)$, $t > u > v > 0$
  - $W(0) = 0$



Models in Continuous Time

- The fact that such a process exists is by no means easy to see.
- It has been an important part of the literature in Physics, Probability and Finance at least since the papers of Bachelier and Einstein, more than 100 years ago.
- A Brownian motion process also has some interesting and remarkable theoretical properties:
    - it is continuous with probability one but the probability that the process has finite variation in any interval is 0.
    - with probability one it is nowhere differentiable.

Models in Continuous Time

- Continuous time process are usually built one small increment at a time and defined to be the limit as the size of the time increment is reduced to zero.
- Let us consider for example how we might define a stochastic (Ito) integral of the form

$$\int_0^T h(t)\, dW_t$$

- An approximating sum takes the form

$$\int_0^T h(t) dW_t \sim \sum_{i=0}^{n-1} h(t_i)[W(t_{i+1}) - W(t_i)] \quad 0 = t_0 < t_1 < \cdots < t_n = T$$

Models in Continuous Time

- Note that the function $h(t)$ is evaluated at the left hand end-point of the intervals $[t_i, t_{i+1}]$, and this is characteristic of the Ito calculus, and an important feature distinguishing it from the usual Riemann calculus.

$$\int_0^T h(t)dW_t \sim \sum_{i=0}^{n-1} h(t_i)[W(t_{i+1})-W(t_i)] \quad 0 = t_0 < t_1 < \cdots < t_n = T$$

Models in Continuous Time

- There are some simple reasons why evaluating the function at the left hand end-point is necessary for stochastic models in finance.
- For example let us suppose that the function $h(t)$ measures how many shares of a stock we possess and $W(t)$ is the price of one share of stock at time t. It is clear that we cannot predict precisely future stock prices and our decision about investment over a possibly short time interval $[t_i, t_{i+1}]$ must be made at the beginning of this interval, not at the end or in the middle.
- Second, in the case of a Brownian motion process $W(t)$, it makes a difference where in the interval $[t_i, t_{i+1}]$ we evaluate the function h to approximate the integral, whereas it makes no difference for Riemann integrals.This difference is essentially due to the fact that $W(t)$, unlike those functions studied before in calculus, is of infinite variation.

Models in Continuous Time

- Let us suppose that the increment dW is used to denote small increments $W(t_{i+1}) - W(t_i)$ involved in the construction of the integral.
- If we denote the interval of time $t_{i+1} - t_i$ by dt, we can loosely assert that dW has a normal distribution with mean 0 and variance dt.
- If we add up a large number of independent such increments, since the variances add, the sum has variance the sum of the values dt and standard deviation the square root.
- Loosely, dW is normal with mean 0 and standard deviation $dt^{1/2}$ and so dW is non-negligible compared with dt as $dt \rightarrow 0$

Models in Continuous Time

- We can define each of the differentials dW and dt essentially by reference to the result when we integrate.
- Remember that if we write an equation in differential form

$$dX_t = h(t)dW_t$$

then this only has real meaning through its integrated version

$$X_t = X_0 + \int_0^t h(t)\, dW_t$$

- What about the terms involving $(dW)^2$? What meaning should we assign to a term like $\int h(t)(dW)^2$?

Models in Continuous Time

- Consider the approximating function $\sum h(t_i)(W(t_{i+1}) - W(t_i))^2$.
- Notice that, at least in the case that the function h is non-random we are adding up independent random variables

  $$h(t_i)(W(t_{i+1}) - W(t_i))^2$$

  each with expected value $h(t_i)(t_{i+1} - t_i)$ and when we add up these quantities the limit is $\int h(t)dt$ by the law of large numbers.
- Roughly speaking, as differentials, we should interpret $(dW)^2$ as dt because that is the way it acts in an integral.
- Subsequent terms such as $(dW)^3$ or $(dt)(dW)^2$ are all $o(dt)$, i.e. they all approach 0 faster than does dt as $dt \to 0$.

Models in Continuous Time

- Now consider defining a process as a function both of the Brownian motion and of time, say $V_t = g(W_t, t)$.
- If W_t represented the price of a stock or a bond, V_t might be the price of a derivative on this stock or bond.
- Expanding the increment dV using a Taylor series expansion gives

$$dV_t = \frac{\partial}{\partial W} g(W_t, t) dW + \frac{\partial}{\partial t} g(W_t, t) dt + \frac{1}{2} \frac{\partial^2}{\partial W^2} g(W_t, t)(dW)^2 + \ldots$$

- So substituting for $(dW)^2$ and ignoring all terms that are $o(dt)$, we obtain a simple version of Ito's lemma

$$dg(W_t, t) = \frac{\partial}{\partial W} g(W_t, t)\, dW + \left[\frac{\partial}{\partial t} g(W_t, t) + \frac{1}{2} \frac{\partial^2}{\partial W^2} g(W_t, t) \right] dt$$

Models in Continuous Time

- Now we begin with an attempt to construct the model for an Ito process or diffusion process in continuous time.
- We construct the price process one increment at a time and it seems reasonable to expect that both the mean and the variance of the increment in price may depend on the current price but does not depend on the process before it arrived at that price.
- This is a loose description of a Markov property.
- The conditional distribution of the future of the process depends only on the current time t and the current price of the process.

Models in Continuous Time

- Let us suppose in addition that the increments in the process are, conditional on the past, normally distributed.
- Thus we assume that for small values of h, conditional on the current time t and the current value of the process X_t, the increment $X_{t+h} - X_t$ can be generated from a normal distribution with mean $a(X_t, t)h$ and with variance $\sigma^2(X_t, t)h$ for some functions a and σ^2 called the **drift** and **diffusion** coefficients respectively.
- Such a normal random variable can be formally written as

$$dX_t = a(X_t, t)dt + \sigma(X_t, t)dWt$$

Models in Continuous Time

- Many of the continuous time models used in finance are described as Markov diffusions or Ito processes which permits the mean and the variance of the increments to depend more generally on the present value of the process and the time.
- The integral version of this relation is of the form

$$X_T = X_0 + \int_0^T a(X_t, t)\, dt + \int_0^T \sigma(X_t, t)\, dW_t$$

  the last integral being an *Ito's Integral*
- The coefficient $a(X_t, t)$ and $\sigma(X_t, t)$ vary with the choice of model.

Models in Continuous Time

- Various choices for the functions $a(X_t, t)$ and $\sigma(X_t, t)$ are possible.
- For example, for the Black-Scholes model or Geometric Brownian Motion:

$$a(X_t, t) = aX_t, \quad \sigma(X_t, t) = \sigma X_t$$

  for constant drift and volatility parameters a and σ;
- The Cox-Ingersoll-Ross (1985) model, used to model spot interest rates, corresponds to

$$a(X_t, t) = A(b - X_t), \quad \sigma(X_t, t) = c\sqrt{X_t}$$

- Many other choices are possible, there are a large number of models for most continuous-time processes observed in finance;

Subsection 2

Why we need Monte Carlo Methods

Valuing a derivative contract

- A derivative can be perfectly replicated by means of a self-financing dynamic portfolio whose value exactly matches all of the derivative flows in every state of the world. This approach shows that the values of the derivative (and of the portofolio) solves the following PDE

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial S} rS + \frac{1}{2}\frac{\partial f}{\partial S}\sigma^2 S^2 = fr \tag{1}$$

with the terminal condition at T that is the derivative's payoff

$$f(T, S(T)) = payoff$$

Feynman–Kac formula

- The **Feynman–Kac formula** named after Richard Feynman and Mark Kac, establishes a link between parabolic partial differential equations (PDEs) and stochastic processes.
- It offers a method of solving certain PDEs by simulating random paths of a stochastic process. Conversely, an important class of expectations of random processes can be computed by deterministic methods.
- Consider the PDE

$$\frac{\partial u}{\partial t}(x,t)+\mu(x,t)\frac{\partial u}{\partial x}(x,t)+\tfrac{1}{2}\sigma^2(x,t)\frac{\partial^2 u}{\partial x^2}(x,t)-V(x,t)u(x,t)+f(x,t)=0$$

subject to the terminal condition

$$u(x,T)=\psi(x)$$

Feynman–Kac formula

- Then the Feynman–Kac formula tells us that the solution can be written as a conditional expectation

$$u(x,t) = E^{Q}\left[\int_t^T e^{-\int_t^r V(X_\tau,\tau)\,d\tau} f(X_r,r)dr + e^{-\int_t^T V(X_\tau,\tau)\,d\tau}\psi(X_T)\middle| X_t = x\right]$$

under the probability measure $\mathbb{Q}$ such that X' is an Ito process driven by the equation

$$dX = \mu(X,t)\,dt + \sigma(X,t)\,dW^{\mathbb{Q}}$$

Valuing a derivative contract

- According to the Feynmann-Kac formula, if $f(t_0, S(t_0))$ solves the B-S PDE, then it is also solution of

$$f(t_0, S(t_0)) = \mathbb{E}\left[e^{-r(T-t_0)} f(T, S(T)|\mathcal{F}_{t_0}\right]$$

- i.e. it's the expected value of the discounted payoff in a probability measure where the evolution of the asset is

$$dS = rSdt + \sigma Sdw$$

- This probability measure is the **Risk Neutral Measure**

Valuing a derivative contract

- Since there exist such an equivalence, we can compute option prices by means of two numerical methods
- PDE: finite difference (explicit, implicit, crank-nicholson)suitable for optimal exercise derivatives;
- Integration
  - Quadrature Methods;
  - Monte Carlo Methods

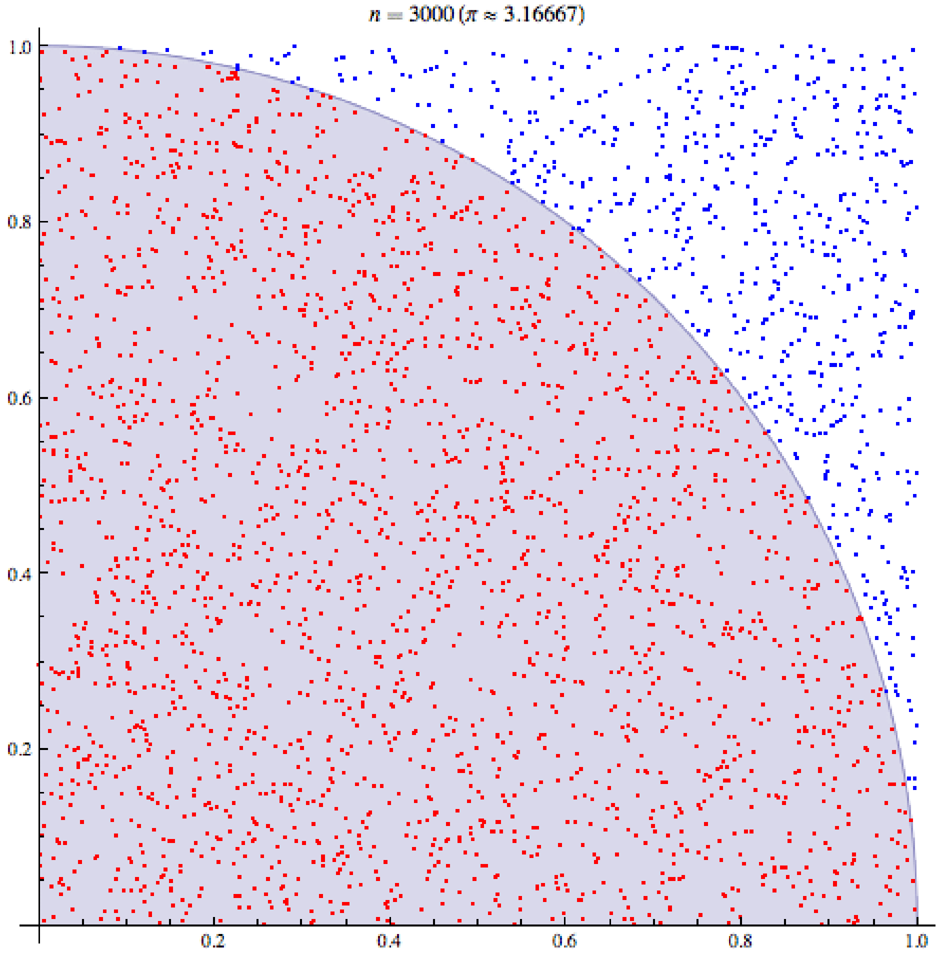
Subsection 3

What is Monte Carlo?

What is Monte Carlo?

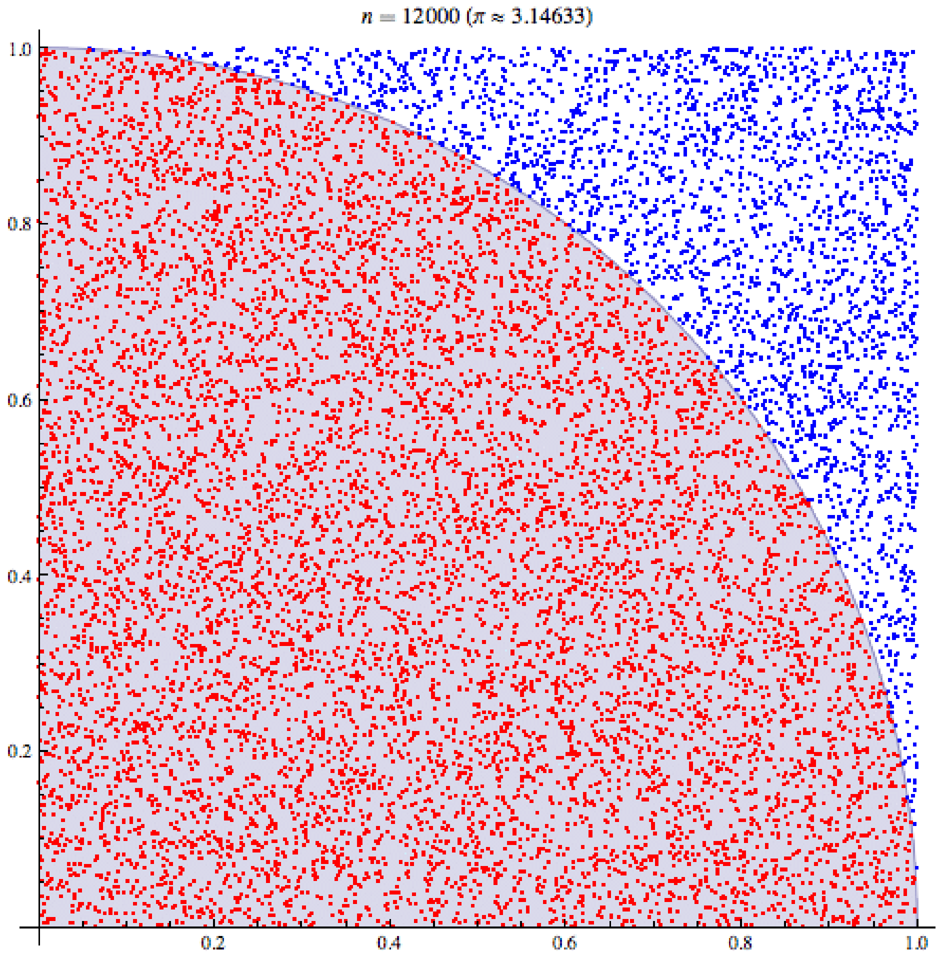
- From a quite general point of view (not a really precise one, actually) with the term Monte Carlo usually one means a numerical technique which makes use of random numbers for solving a problem.
- For the moment we assume that you can understand, at least intuitively, what a random number is.
- Later we will return to the definition of a random number, and, as we shall see, this will lead to absolutely not trivial issues.
- Let's start with a very simple example...

What is Monte Carlo?



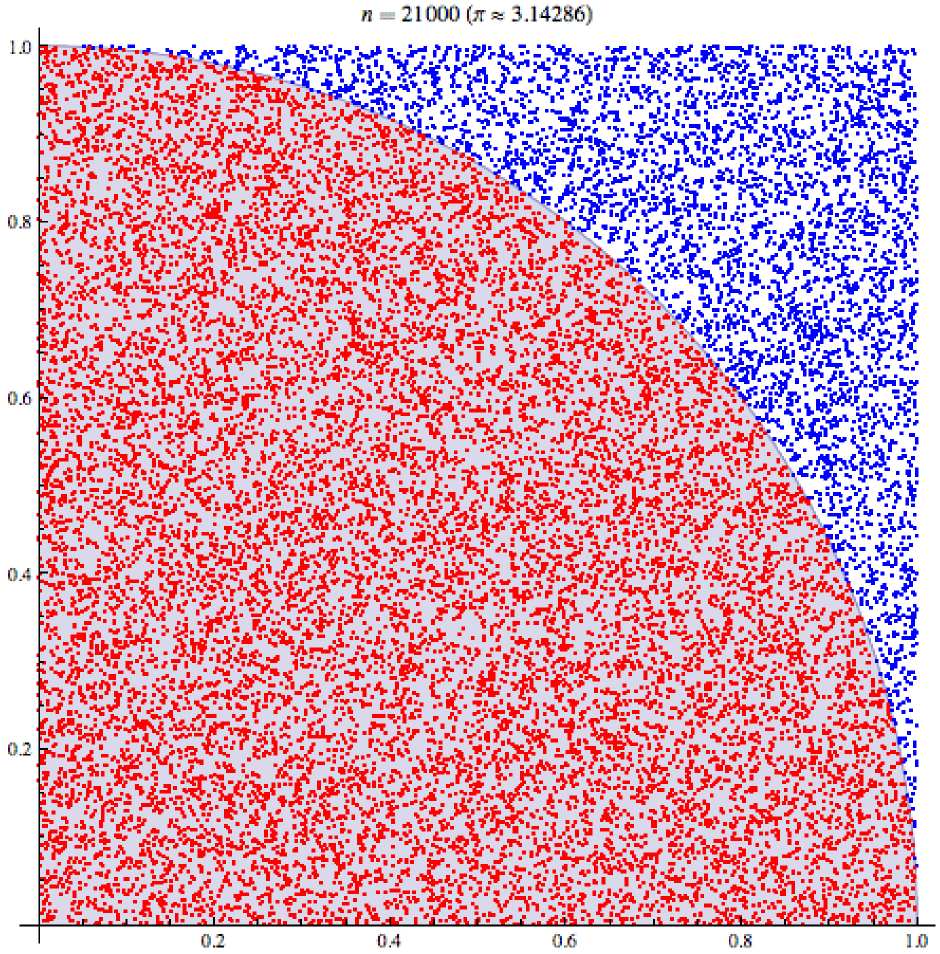
- How to estimate a value of Pi using the Monte Carlo method?
- Generate a large number of random points and see how many fall in the circle enclosed by the unit square;
- Take a circle inscribed in a unit square. Given that the circle and the square have a ratio of areas that is $\pi/4$, the value of π can be approximated using a Monte Carlo method:

What is Monte Carlo?



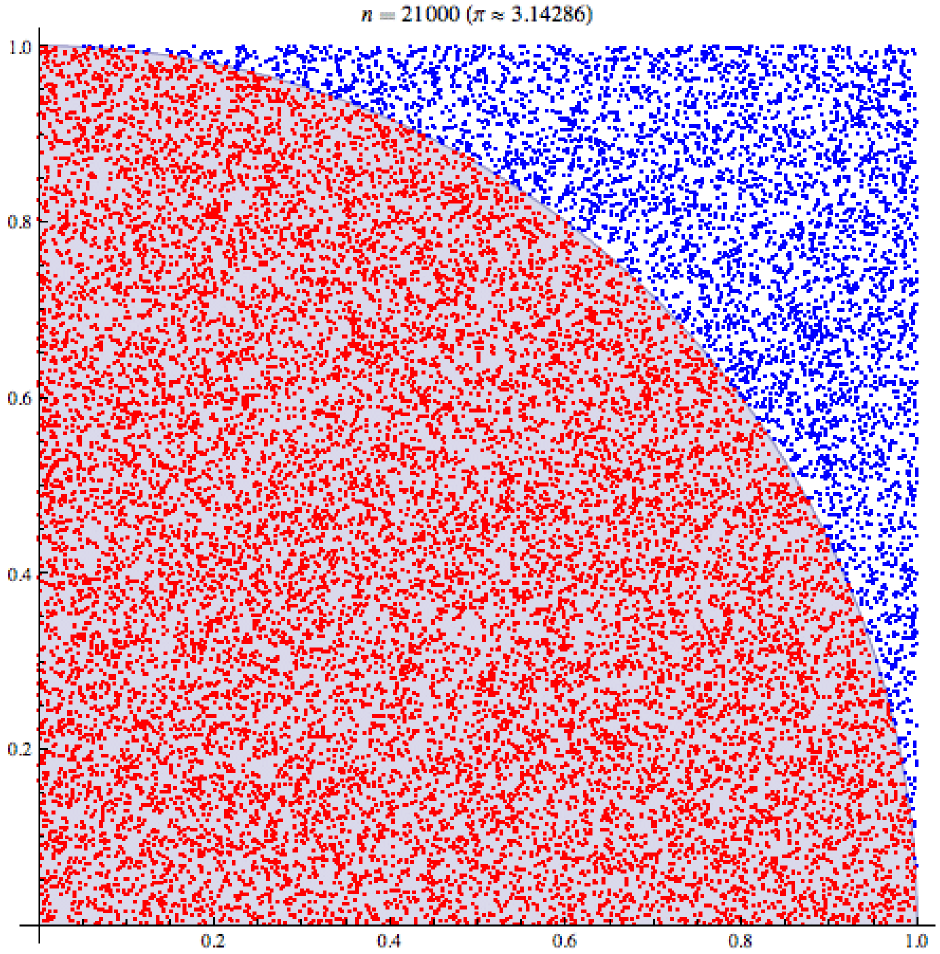
1. Draw a square on the ground, then inscribe a circle within it.
2. Uniformly scatter some objects of uniform size (grains of rice or sand) over the square.
3. Count the number of objects inside the circle and the total number of objects.
4. The ratio of the two counts is an estimate of the ratio of the two areas, which is $\pi/4$. Multiply the result by 4 to estimate π.

What is Monte Carlo?



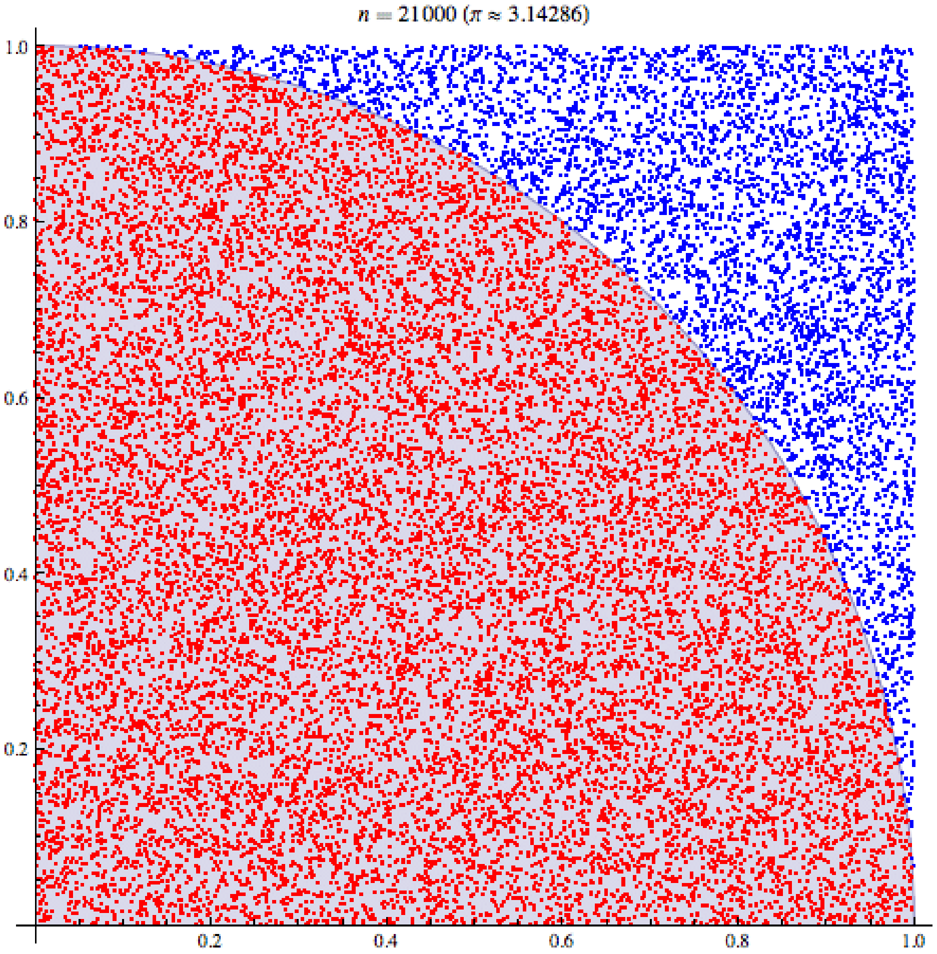
- In this procedure the domain of inputs is the square that circumscribes our circle.
- We generate random inputs by scattering grains over the square then perform a computation on each input (test whether it falls within the circle).
- Finally, we aggregate the results to obtain our final result, the approximation of π.

What is Monte Carlo?



- There are two important points to consider here...
- First, if the grains are not uniformly distributed, then our approximation will be a poor one;

What is Monte Carlo?



- Second, there should be a large number of inputs;
- the approximation is generally very poor if only a few grains are randomly dropped into the whole square;
- on average the approximation improves (slowly!) as more grains are dropped.

Notebook

- **GitHub :** polyhedron-gdl;
- **Notebooks :** mcs_2;

Monte Carlo is Integration!

- There is a formal connection between the use of the Monte Carlo method and the concept of integration of a function.
- The case related to the area of the circle is evident
- More in general we can state that each extraction of a sample of random numbers can be used as an estimator of an integral.
- As an example consider the case relating to the integration of a function of a real variable;
- by a suitable change of variable, we can always bring us back to the simplest case in which the integration interval is between 0 and 1:

$$I = \int_0^1 f(x)dx = \int_0^1 1 \cdot f(x)dx$$

Monte Carlo is Integration!

- The key point of our argument is to recognize that the expression written above is also the expectation value of the function f at values of a random variable uniformly distributed in the range $[0, 1]$.
- It becomes possible to estimate the value of our integral using an arithmetic mean of n values of $f(U_i)$ where each U_i is a sample from a uniform distribution in $[0, 1]$.
- In other words we can say that the quantity

$$\tilde{I}_n = \frac{1}{n}\sum_{i=1}^{n} f(U_i)$$

is an unbiased estimator of I.

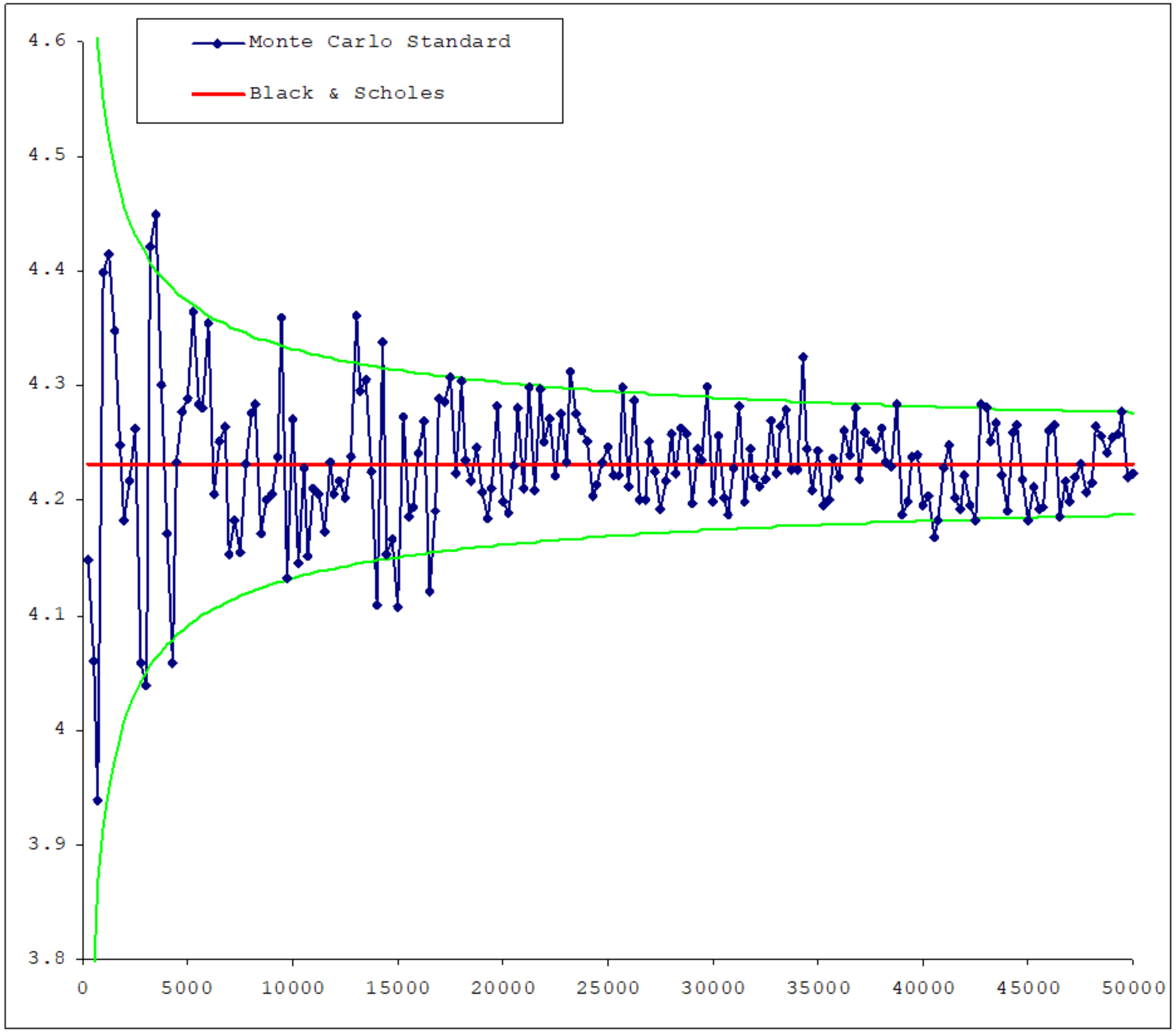
Monte Carlo is Integration!

- The variance of the estimator is

$$var\left(\tilde{I}_n\right) = \frac{var(f(U_i))}{n}$$

- the mean square error of the estimator, which can be interpreted as the mean square error of the Monte Carlo simulation, decreases with increasing n.
- This result is completely independent of the dimensionality of the problem.
- It's this last characteristic that makes attractive the Monte Carlo method for solving problems with a large number of dimensions.
- In this case typically the Monte Carlo method converge to the final value faster than the traditional numerical methods.

Monte Carlo is Integration!



What is Monte Carlo?

Regardless the many different definitions, Montecarlo methods share a common procedural pattern;

1. Define a domain of possible inputs;
2. Generate inputs randomly from a probability distribution over the domain;
3. Perform a deterministic computation on the inputs;
4. Aggregate the results;

This is particularly evident in the next example...

What is Monte Carlo?

- The pricing of an option is usually dealt with in the context of so-called risk-neutral valuation.
- Indicating with $f[S(T)]$ where S is the value of the underlying asset, the value of the option at maturity T, the value today, $f[S(t)]$, is given by

$$f(S(t)) = \mathbb{E}^{\mathbb{Q}}\left[P(t,T)f[S(T)]\right]$$

- $\mathbb{E}^{\mathbb{Q}}$ being the risk-neutral expectation value and $P(t,T)$ the discount function between t and T.

What is Monte Carlo?

- Let's assume, for simplicity, to know with certainty the value of the discount function so the problem can be put in the form

$$f(S(t)) = e^{-r(T-t)}\mathbb{E}^{\mathbb{Q}}\left[f[S(T)]\right]$$

- The application of the Monte Carlo method in this case is reduced essentially to the generation of a sufficiently high number of estimates of $f[S(T)]$ from which to extract the average value.

⚠ To this end it is necessary first to introduce a hypothesis on how the underlying stock price evolves over time;

What is Monte Carlo?

- Let's suppose for example that the asset price follows a geometric Brownian motion, according to this hypothesis the rate of change of the price in a range of infinitesimal time is described by

$$dS_t = rS_t dt + S_t \sigma dZ_t$$

where r is the risk free rate, σ is the volatility of S returns and dw is a brownian motion;

- Remember the generic form of the Ito's Lemma for a single random variable

$$df(x(t), t) = \left(a(x, t)\frac{\partial f}{\partial x} + \frac{\partial f}{\partial t} + \frac{1}{2}(b(x, t))^2 \frac{\partial^2 f}{\partial x^2} \right) dt + b(x, t)\frac{\partial f}{\partial x} dW_t$$

What is Monte Carlo?

- In this case we have:

$$x(t) = S_t \quad a(x(t), t) = rS_t, \quad b(x(t), t) = \sigma S_t$$

and we choose

$$f(x(t), t) = \log(S_t)$$

- Direct computation of the derivatives give us:

$$\frac{\partial f}{\partial x} = \frac{1}{S_t}, \quad \frac{\partial f}{\partial t} = 0, \quad \frac{\partial^2 f}{\partial S^2} = -\frac{1}{S_t^2}$$

- We obtain

$$\begin{aligned} d\log(S_t) &= \left(rS_t \frac{1}{S_t} - \frac{1}{2}\sigma^2 S_t^2 \frac{1}{S_t^2} \right) dt + \sigma S_t \frac{1}{S_t} dW_t \\ &= \left(r - \frac{1}{2}\sigma^2 \right) dt + \sigma dW_t \end{aligned}$$

What is Monte Carlo?

- ⚠ A discrete version, which can easily be simulated is given by the difference equation

$$\Delta \log(S_t) = \log\left(\frac{S_t}{S_{t+\Delta t}}\right) = \left(r - \frac{1}{2}\sigma^2\right)\Delta t + \sigma \Delta W_t \Rightarrow$$

$$\Rightarrow S_t = S_{t-\Delta t} exp\left[\left(r - \frac{1}{2}\sigma^2\right)\Delta t + \sigma \Delta W_t\right]$$

- 🎯 How obtain a sequence of Wiener process? Remember that by definition

$$w(t_i) = w(t_{i-1}) + \sqrt{t_i - t_{i-1}} Z \quad Z \sim N(0,1)$$

for times $t \in (\Delta t, 2\Delta t, \ldots, T)$ and z_t being standard normally distributed random numbers;

What is Monte Carlo?

- Finally, the evolution rule is

$$S_t = S_{t-\Delta t} exp\left[\left(r - \frac{1}{2}\sigma^2\right)\Delta t + \sigma\sqrt{\Delta t} z_t\right]$$

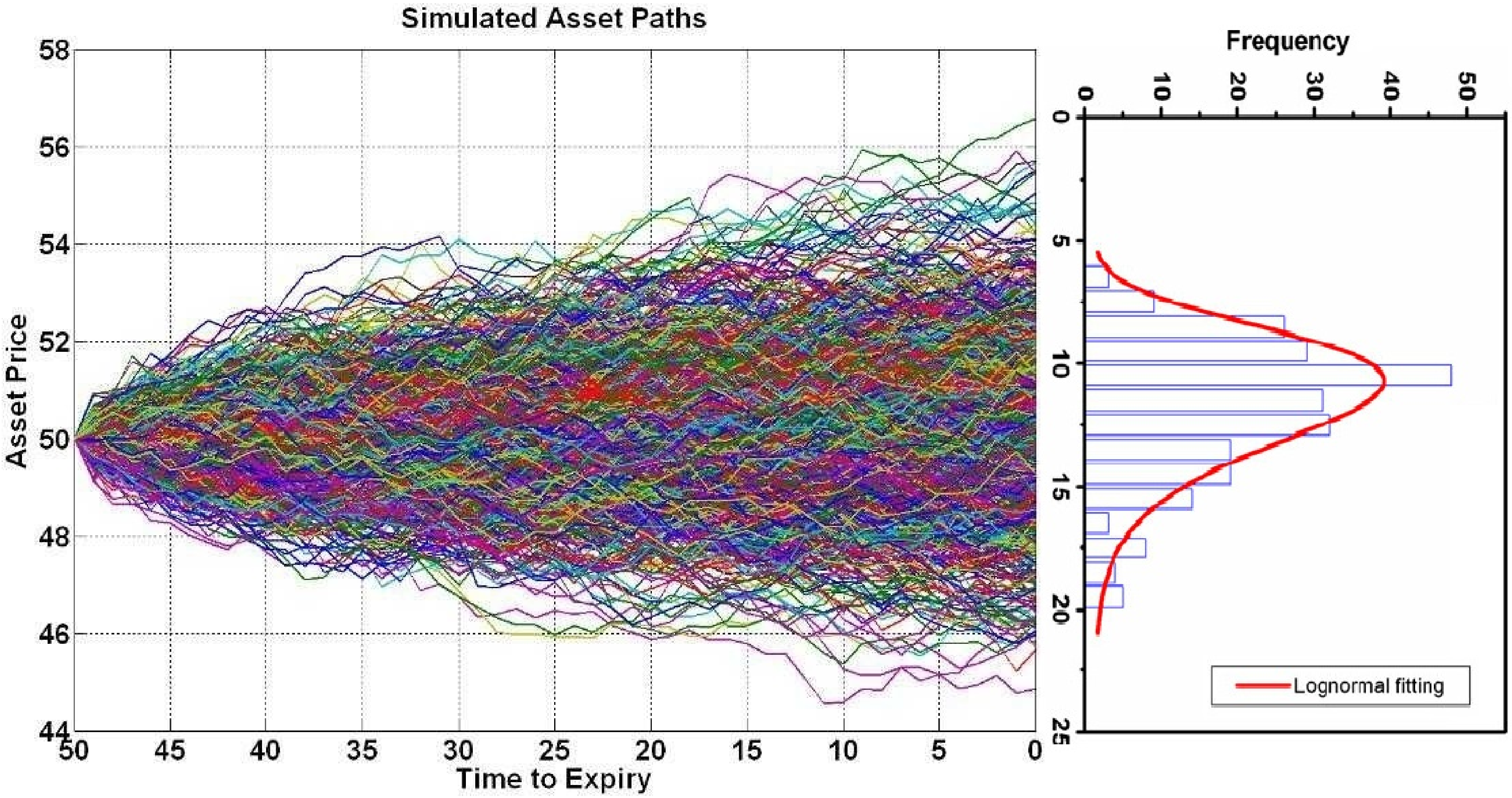
- Once we have the simulated value of the underlying at time T, we are able to derive the value of the option at the same date; For a call option, for example, we simply write

$$f[S(T)] = \max(S(T) - K, 0)$$

  where K is the strike price.

- By repeating the above procedure a very large number of times we are able to obtain a distribution of values for $f[S(T)]$ from which it is possible to extract the expectation value ...

What is Monte Carlo?



Pricing a Call Option: A Different View

- It's worth to recast the pricing problem into a simple integral formulation in order to gain some insight into the general problem;
- So let's consider again the payoff of a simple plain vanilla option

$$e^{-rT}\mathbb{E}^{\mathbb{Q}}[h(S_T)] = e^{-rT}\mathbb{E}^{\mathbb{Q}}\left[h\left(S_0 e^{\log(S_T/S_0)}\right)\right]$$

- By a simple application of Ito's lemma is easy to demonstrate that the variable $X = \log(S_T/S_0)$ has a normal distribution with mean $m = (r - \frac{1}{2}\sigma^2)T$ and variance $s = \sigma^2 T$.
- So we can write

$$C(S,t) = e^{-rT}\int_{-\infty}^{+\infty} \max[S_0 e^X - K, 0] e^{-\frac{(X-m)^2}{2s^2}} dX$$

Pricing a Call Option: A Different View

- It is possible to generate a normally distributed random variable $X = \Phi^{-1}(U; (r - \frac{1}{2}\sigma^2)T; \sigma^2 T)$ using the inverse transform method, where $\Phi^{-1}(\dots)$ is the inverse of the normal cumulative distribution function evaluated at U;
- U is a uniform $[0, 1]$ random variable.

$$U = \Phi[X; m, u], \quad u \to 1\, when\, X \to +\infty, \quad u \to 0\, when\, X \to -\infty$$

- From the previous relation we find (within a normalization factor)

$$du = \frac{d\Phi[X; m, u]}{dX} dX \Rightarrow dX = \frac{1}{e^{-\frac{(X-m)^2}{2s^2}}} du$$

- and ...

Pricing a Call Option: A Different View

- … finally we can write our integral in the form

$$C(S,t) = \int_0^1 f(u)du$$

where $f(u) = e^{-rT}\max[S_0 \exp(\Phi^{-1}(u;m,s)) - K, 0]$

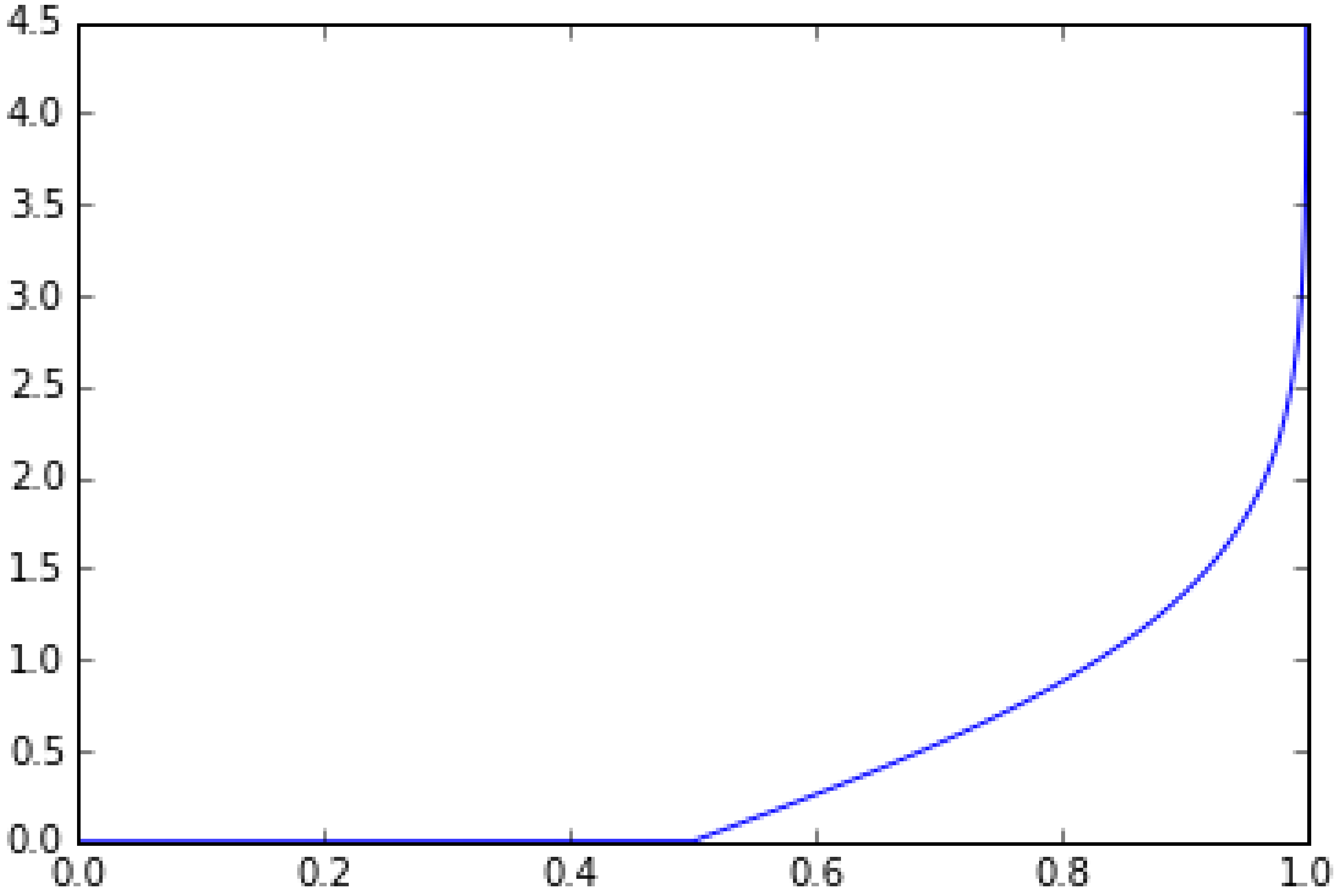
Pricing a Call Option - The Python Code

```
def f(u, S0, K, r, sigma, T):
    m        = (r - .5*sigma*sigma)*T
    s        = sigma*sqrt(T)
    f_u      = exp(-r*T) *
               np.maximum(S0*exp(scnorm.ppf(u, m, s))-K,0)
    return f_u

u        = rand(1000000)
f_u      = f(u,S0,K,r,sigma,T)

print mean(f_u)
```

Pricing a Call Option - The Integrand Function



Notebook

- **GitHub :** polyhedron-gdl;
- **Notebooks :** mcs_3;

Outline

1. Background
   - Models in Continuous Time
   - Why we need Monte Carlo Methods
   - What is Monte Carlo?

2. Basic Monte Carlo
   - Numerical Integration of SDE
   - The Brownian Bridge

Subsection 1

Numerical Integration of SDE

Scenario Contruction

- There are several ways to construct scenario for pricing
- Constructing a path of the solution to the SDE at times t_i by exact advancement of the solution;
    - This method is only possible if we have an analytical expression for the solution of the stochastic differential equation as in the case of Geometric Brownian Motion;
- Approximate numerical solution of the stochastic differential equation;
    - This is the method of choice if we cannot use the previous one; Just as in the case of ODE there are numerical techniques for discretizing and solving SDE.

Numerical Integration of SDE

- The numerical integration of the SDE by finite difference is another way of generating scenarios for pricing;
- In the case of the numerical integration of ordinary differential equations by finite differences the numerical scheme introduces a discretization error that translates into the numerical solution differing from the exact solution by an amount proportional to a power of the time step.
- This amount is the truncation error of the numerical scheme.

Numerical Integration of SDE

- In the case of the numerical integration of SDE by finite differences the interpretation of the numerical error introduced by the discretization scheme is more complicated;
- Unlike the case of ODE where the only thing we are interested in computing is the solution itself, when dealing with SDE there are two aspects that interest us:
  - One aspect is the accuracy with which we compute the trajectories or paths of a realization of the solution
  - The other aspect is the accuracy with which we compute functions of the process such as expectations and moments.

Numerical Integration of SDE

- The order of accuracy with which a given scheme can approximate trajectories of the solution is not the same as the accuracy with which the same scheme can approximate expectations and moments of functions of the trajectories;
- The convergence of the numerically computed trajectories to the exact trajectories is called strong convergence and the order of the corresponding numerical scheme is called order of strong convergence;
- The convergence of numerically computed functions of the stochastic process to the exact values is called weak convergence and the related order is called order of weak convergence.

Numerical Integration of SDE

- We assume that the stock price S_t is driven by the stochastic differential equation (SDE)

$$dS(t) = \mu(S, t)dt + \sigma(S, t)dW_t \tag{2}$$

  where W_t is, as usual, Brownian motion.
- We simulate S_t over the time interval $[0; T]$, which we assume to be is discretized as $0 = t_1 < t_2 < \cdots < t_m = T$, where the time increments are equally spaced with width dt.
- Equally-spaced time increments is primarily used for notational convenience, because it allows us to write $t_i - t_{i-1}$ as simply dt. All the results derived with equally-spaced increments are easily generalized to unequal spacing.

Numerical Integration of SDE

- **Euler Scheme**
- The simplest way to discretize the process in Equation (2) is to use Euler discretization

$$\text{EULER} \Rightarrow \hat{S}(t_{i+1}) = \hat{S}(t_i) + \mu[\hat{S}(t_i), t_i]\Delta t + \sigma[\hat{S}(t_i), t_i]\left(w(t_{i+1}) - w(t_i)\right)$$

- **Milshstein Scheme**

$$\text{MILSHSTEIN} \Rightarrow \text{EULER} + \frac{1}{2}\sigma[\hat{S}(t_i)]\frac{\partial\sigma[\hat{S}(t_i)]}{\partial S}\left[\left(w(t_{i+1}) - w(t_i)\right)^2 - \Delta t\right]$$

This scheme is described in Glasserman and in Kloeden and Platen for general processes, and in Kahl and Jackel for stochastic volatility models. The scheme works for SDEs for which the coefficients $\mu(S_t)$ and $\sigma(S_t)$ depend only on S, and do not depend on t directly

Numerical Integration of SDE

- For an intuitive derivation, we will only look at GBM:

$$d\ ln\, x_t = \left(\mu - \frac{1}{2}\sigma^2\right) dt + \sigma dW_t$$

- The solution to the GBM SDE is

$$x_{t+\Delta t} = x_t exp\left\{ \int_t^{t+\Delta t} \left(\mu - \frac{1}{2}\sigma^2\right) dt + \int_t^{t+\Delta t} \sigma dW_u \right\}$$

$$\sim x_t \left(1 + \mu\Delta t - \frac{1}{2}\sigma^2\Delta t + \sigma\Delta W_t + \frac{1}{2}\sigma^2(\Delta W_t)^2\right)$$

$$\Rightarrow x_{t+\Delta t} \sim x_t + a(x_t)\Delta t + b(x_t)\Delta w_t + \frac{1}{2}b(x_t)\frac{\partial b(x_t)}{\partial x_t}((\Delta W_t)^2 - \Delta t)$$

Notebook

- **GitHub :** polyhedron-gdl;
- **Notebooks :** mcs_sde_solution;
- **Code :** mcs_sde_solution.py;

Subsection 2

The Brownian Bridge

The Brownian Bridge

- Assume you have a Wiener process defined by a set of time-indexed random variables $W(t_1), W(t_2), ..., W(t_n)$.
- How do you insert a random variable $W(t_k)$ where $t_i \leq t_k \leq t_{i+1}$ into the set in such a manner that the resulting set still constitutes a Wiener process?
- The answer is: with a Brownian Bridge!
- The Brownian Bridge is a sort of interpolation that allows you to introduce intermediate points in the trajectory of a Wiener process.

The Brownian Bridge

- Brownian Bridge Construction
- Given $W(t)$ and $W(t+\delta t_1+\delta t_2)$ we want to find $W(t+\delta t_1)$;
- We assume that we can get the middle point by a weighted average of the two end points plus an independent normal random variable:

$$W(t+\delta t_1)=\alpha W(t)+\beta W(t+\delta t_1+\delta t_2)+\gamma Z$$

where α, β and λ are constants to be determined and Z is a standard normal random variable.

The Brownian Bridge

- We have to satisfy the following conditions:

$$\begin{cases} cov[W(t+\Delta t_1), W(t)] = min(t+\Delta t_1, t) = t \\ cov[W(t+\Delta t_1), W(t+\Delta t_1+\Delta t_2)] = t+\Delta t_1 \\ var[W(t+\Delta t_1)] = t+\Delta t_1 \end{cases}$$

$$\begin{cases} \alpha+\beta = 1 \\ \alpha t + \beta(t+\Delta t_1+\Delta t_2) = t+\Delta t_1 \\ \alpha^2 t + 2\alpha\beta t + \beta^2(t+\Delta t_1+\Delta t_2) + \lambda^2 = t+\Delta t_1 \end{cases}$$

- which are equivalent to:

$$\begin{cases} \alpha = \frac{\Delta t_2}{\Delta t_1+\Delta t_2} \\ \beta = 1-\alpha = \frac{\Delta t_1}{\Delta t_1+\Delta t_2} \\ \gamma = \sqrt{\Delta t_1 \alpha} \end{cases}$$

The Brownian Bridge

- We can use the brownian bridge to generate a Wiener path and then use the Wiener path to produce a trajectory of the process we are interested in;
- The simplest strategy for generating a Wiener path using the brownian bridge is to divide the time span of the trajectory into two equal parts and apply the brownian bridge construction to the middle point. We then repeat the procedure for the left and right sides of the time interval.

The Brownian Bridge

- Notice that as we fill in the Wiener path, the additional variance of the normal components we add has decreasing value;
- Of course the total variance of all the Wiener increments does not depend on how we construct the path, however the fact that in the brownian bridge approach we use random variables that are multiplied by a factor of decreasing magnitude means that the importance of those variables also decreases as we fill in the path;
- The dimension of the random variables with larger variance need to be sampled more efficiently than the dimension with smaller variance;

Notebook

- **GitHub :** polyhedron-gdl;
- **Code :** mcs_brownian_bridge.py;