Monte Carlo Simulation

What is Monte Carlo

- It is a tool for combining *distributions*, and thereby propagating more than just summary statistics
- It uses random number generation, rather than analytic calculations
- It is increasingly popular due to high speed personal computers
- The Monte Carlo method is a numerical method for statistical simulation which utilizes sequences of random numbers to perform the simulation

Background/History

- “Monte Carlo” from the gambling town of the same name (no surprise)
- First applied in 1947 to model diffusion of neutrons through fissile materials
- Limited use because time consuming
- Much more common since late 80’s
- Too easy now?

Steps of Monte Carlo Simulation

- Step 1: Clearly identify the problem
  - Identify the objective of problem
  - Identify the main factor which have the greatest effect on the objective of the problem.
- Step 2:Construct a suitable model
  - Specify the variable and parameters
  - Formulate decision rules
  - Identify the type of distribution (Theoretical or analytical)
  - Specify in what way time will change.

Steps of Monte Carlo Simulation

- Step 3: Prepare model for experiment
  - Specify the starting condition
  - Number of rums of simulation to be made
- Step 4:Using 1-3 perform simulation
  - Coding system, random number generator
- Step 5: Examine & summarize the results obtained in step 4
- Step 6: Evaluate the results of simulation.

Find the Value of π

- consider a circle inscribed in a unit square. Given that the circle and the square have a ratio of areas that is π/4, the value of π can be approximated using a Monte Carlo method
- Draw a square on the ground, then inscribe a circle within it.
- Uniformly scatter some objects of uniform size (grains of rice or sand) over the square.
- Count the number of objects inside the circle and the total number of objects.
- The ratio of the two counts is an estimate of the ratio of the two areas, which is π/4. Multiply the result by 4 to estimate π.

Random v. Pseudo-random

- **Random numbers** have no defined sequence or formulation. Thus, for any *n* random numbers, each appears with equal probability.
- If we restrict ourselves to the set of 32-bit integers, then our numbers will start to repeat after some very large *n*. The numbers thus clump within this range and around these integers.
- Due to this limitation, computer algorithms are restricted to generating what we call **pseudo-random numbers**.

Definition

- *Ripley* defines most probabilistic modeling as stochastic simulation, with *Monte Carlo* being reserved for Monte Carlo integration and Monte Carlo statistical tests.
- *Sawilowsky define as* a simulation is a fictitious representation of reality, a Monte Carlo method is a technique that can be used to solve a mathematical or statistical problem, and a Monte Carlo simulation uses repeated sampling to determine the properties of some phenomenon (or behavior).
- Examples:
  - ✓ Simulation: Drawing **one** pseudo-random uniform variable from the interval [0,1] can be used to simulate the tossing of a coin, If the value is less than or equal to 0.50 designate the outcome as heads, but if the value is greater than 0.50 designate the outcome as tails. This is a simulation, but not a Monte Carlo simulation.
  - ✓ Monte Carlo method: Pouring out a box of coins on a table, and then computing the ratio of coins that land heads versus tails is a Monte Carlo method of determining the behavior of repeated coin tosses, but it is not a simulation.
  - ✓ Monte Carlo simulation: Drawing **a large number** of pseudo-random uniform variables from the interval [0,1], and assigning values less than or equal to 0.50 as heads and greater than 0.50 as tails, is a *Monte Carlo simulation* of the behavior of repeatedly tossing a coin.

Characteristics of a high quality Monte Carlo simulation

- The (pseudo-random) number generator has certain characteristics (*e.g.*, a long "period" before the sequence repeats)
- The (pseudo-random) number generator produces values that pass tests for randomness
- There are enough samples to ensure accurate results
- The proper sampling technique is used
- The algorithm used is valid for what is being modeled

Applications

- Physical sciences: Monte Carlo methods are very important in computational physics, physical chemistry, and related applied fields, and have diverse applications from complicated quantum chromodynamics calculations to designing heat shields and aerodynamic forms as well as in modeling radiation transport for radiation dosimetry calculations.
- Statistical physics Monte Carlo molecular modeling is an alternative to computational molecular dynamics, and Monte Carlo methods are used to compute statistical field theories of simple particle and polymer systems.
- Quantum Monte Carlo methods solve the many-body problem for quantum systems.
- Experimental particle physics, Monte Carlo methods are used for designing detectors, understanding their behavior and comparing experimental data to theory.
- Astrophysics, they are used in such diverse manners as to model both galaxy evolution and microwave radiation transmission through a rough planetary surface. Monte Carlo methods are also used in the ensemble models that form the basis of modern weather forecasting.

Engineering

- Monte Carlo methods are widely used in engineering for sensitivity analysis and quantitative probabilistic analysis in process design. The need arises from the interactive, co-linear and non-linear behavior of typical process simulations.
  - ✓ Microelectronics engineering, Monte Carlo methods are applied to analyze correlated and uncorrelated variations in analog and digital integrated circuits.
  - ✓ Geostatistics and Geometallurgy, Monte Carlo methods underpin the design of mineral processing flowsheets and contribute to quantitative risk analysis
  - ✓ Wind energy yield analysis, the predicted energy output of a wind farm during its lifetime is calculated giving different levels of uncertainty
  - ✓ impacts of pollution are simulated and diesel compared with petrol.
  - ✓ Autonomous robotics, Monte Carlo localization can determine the position of a robot. It is often applied to stochastic filters such as the Kalman filter or particle filter that forms the heart of the SLAM (simultaneous localization and mapping) algorithm.
  - ✓ Telecommunications, when planning a wireless network, design must be proved to work for a wide variety of scenarios that depend mainly on the number of users, their locations and the services they want to use. Monte Carlo methods are typically used to generate these users and their states. The network performance is then evaluated and, if results are not satisfactory, the network design goes through an optimization process.
  - ✓ Reliability engineering, one can use Monte Carlo simulation to generate mean time between failures and mean time to repair for components.

Computer graphics

- Path tracing, occasionally referred to as Monte Carlo ray tracing, renders a 3D scene by randomly tracing samples of possible light paths. Repeated sampling of any given pixel will eventually cause the average of the samples to converge on the correct solution of the rendering equation, making it one of the most physically accurate 3D graphics rendering methods in existence.

Search and rescue

- The US Coast Guard utilizes Monte Carlo methods within its computer modeling software SAROPS in order to calculate the probable locations of vessels during search and rescue operations. Each simulation can generate as many as ten thousand data points which are randomly distributed based upon provided variables. Search patterns are then generated based upon extrapolations of these data in order to optimize the probability of containment (POC) and the probability of detection (POD), which together will equal an overall probability of success (POS). Ultimately this serves as a practical application of probability distribution in order to provide the swiftest and most expedient method of rescue, saving both lives and resources.

Design and visuals

- Monte Carlo methods are also efficient in solving coupled integral differential equations of radiation fields and energy transport, and thus these methods have been used in global illumination computations that produce photo-realistic images of virtual 3D models, with applications in video games, architecture, design, computer generated films, and cinematic special effects.

Monte Carlo example

- Let us consider a game in which an unbiased coin is repeatedly flipped. For each Flip you had to pay Rs. 1and when the difference between H & T tossed become 3, you get Rs. 8.
- If the required result obtained in less than 8 flips, you win some money and vice versa.
- Probability of tossing of coin is 0.5 in each case so we consider half number for head and remain for Tail, Head(0,1,2,3,4) for Tail (5,6,7,8,9).
- Following is a string of random no and suitable flips for these are

  5, 9, 3, 6, 4, 8, 6, 8, 1,… and T, T, H, T, H, T, T, T, H, ……

Game No.	S. No	Random No.	H or T	Cumulative		Difference
				Head	Tail	
1	1	5	T	0	1	1
	2	9	T	0	2	2
	3	3	H	1	2	1
	4	6	T	1	3	2
	5	4	H	2	3	1
	6	8	T	2	4	2
	7	6	T	2	5	3
						Win Rs. 1

Game No.	S. No	Random No.	H or T	Cumulative		Difference
				Head	Tail	
2	1	8	T	0	1	1
	2	1	H	1	1	0
	3	5	T	1	2	1
	4	2	H	2	2	0
	5	4	H	3	2	1
	6	0	H	4	2	2
	7	6	T	4	3	1
	8	3	H	5	3	2
	9	1	H	6	3	3
						Lose Rs. 1

Game No.	S. No	Random No.	H or T	Cumulative		Difference
				Head	Tail	
3	1	3	H	1	0	1
	2	3	H	2	0	2
	3	7	T	2	1	1
	4	1	H	3	1	2
	5	8	T	3	2	1
	6	0	H	4	2	2
	7	6	T	4	3	1
	8	2	H	5	3	2
	9	9	T	5	4	1
	10	3	H	6	4	2
	11	4	H	7	4	3
						Lose Rs. 3

Game No.	S. No	Random No.	H or T	Cumulative		Difference
				Head	Tail	
4	1	2	H	1	0	1
	2	9	T	1	1	0
	3	4	H	2	1	1
	4	2	H	3	1	2
	5	3	H	4	1	3
						Win Rs. 1

Game No.	S. No	Random No.	H or T	Cumulative		Difference
				Head	Tail	
5	1	3	H	1	0	1
	2	2	H	2	0	2
	3	9	T	2	1	1
	4	6	T	2	2	0
	5	1	H	3	2	1
	6	8	T	3	3	0
	7	7	T	3	4	1
	8	0	H	4	4	0
	9	8	T	4	5	1
	10	6	T	4	6	2
	11	7	T	4	7	3
						Lose Rs. 3

Game No.	S. No	Random No.	H or T	Cumulative		Difference
				Head	Tail	
6	1	4	H	1	0	1
	2	1	H	2	0	2
	3	8	T	2	1	1
	4	2	H	3	1	2
	5	6	T	3	2	1
	6	0	H	4	2	2
	7	9	T	4	3	1
	8	2	H	5	3	2
	9	4	H	6	3	3
						Lose Rs. 1

Game No.	S. No	Random No.	H or T	Cumulative		Difference
				Head	Tail	
7	1	9	T	0	1	1
	2	8	T	0	2	2
	3	3	H	1	2	1
	4	9	T	1	3	2
	5	3	H	2	3	1
	6	7	T	2	4	2
	7	5	T	2	5	3
						Win Rs. 1

Application of Monte Carlo for integration of a function

Random No	X=0.1*r	Random no	Y=140*r	X3	M	N
22	2.2	0.57	79.8	10.65	0	1
25	2.5	0.18	25.2	15.63	0	2
18	1.8	0.00	0	5.83	1	3
45	4.5	0.90	126.0	91.13	1	4
25	2.5	0.05	7	15.63	2	5
27	2.7	0.77	107.8	19.68	2	6
48	4.8	0.66	92.4	110.60	3	7
43	4.3	0.10	14	79.51	4	8
40	4.0	0.76	106.4	64	4	9
47	4.7	0.42	58.8	103.82	5	10
38	3.8	0.78	109.2	54.87	5	11
33	3.3	0.88	123.2	35.94	5	12
24	2.4	0.03	4.2	13.82	6	13

Normally Distributed Random Numbers

- There are large number of real life situation, where the behavior of system is described by normal probability distribution.
  - Marks obtained by the student in a class, number of defective parts produced, dimension of parts made on a machine etc.
- The value of variate (y) is calculated by

$$y = \mu + \sigma \sum_{r=1}^{12} r_i - 6$$

$$While\,mean\,is\,zero$$
$$y = \mu + \sigma z$$

Application of Random No

Bomb Strike	RNN (random normal no)	X	RNN	Y	Result
1	0.23	115	0.24	72	Hit
2	-1.16	-580	-0.02	-6	Miss
3	0.39	195	0.64	192	Hit
4	-1.90	-950	-1.04	-312	Miss
5	-0.78	-390	0.68	204	Hit
6	-0.02	-10	-0.47	-141	Hit
7	-0.40	-200	-0.75	-225	Hit
8	-0.66	-330	-0.44	-132	Hit
9	1.41	705	1.21	363	Miss
10	0.07	35	-0.08	-24	Hit

Monte Carlo vs Stochastic Simulation

$$X=d-(a+b+c)$$

Part	Min	Max
a	1.95	2.05
b	1.95	2.05
c	29.50	30.50
d	34.00	35.00

Assignment 1