Repeated Random Sampling: Approximating Distributions & Estimating Pi

Using Monte Carlo Techniques in Python

C. Odhiambo[1]

[1]Institute of Mathematical Sciences
Strathmore University

12th May, 2025

Introduction to Random Sampling

- Random sampling is a statistical technique used to select a subset of data from a larger population.
- Each sample is chosen randomly and independently.
- It's a core method in statistics, simulations, and data science because it provides unbiased insights into the population.
- Whether drawing from a normal distribution or shuffling cards, randomness helps us model real-world uncertainty.

Why Repeated Random Sampling?

- One random sample might not represent the population well.
- By repeating the sampling process many times, we reduce variability and improve our approximation of true population parameters.
- This approach allows us to create empirical distributions, estimate standard errors, and test hypotheses more robustly.
- It forms the basis for bootstrapping and Monte Carlo simulations.
- Here's why repeated random sampling is important: Unbiased Estimates and Generalization: Repeated random sampling helps ensure that the sample mean, on average, equals the population mean, leading to unbiased estimates. Assessing Variability and Reliability: By examining the results of multiple random samples, researchers can assess the variability of their data and the reliability of their conclusions and Model Robustness and Performance

Monte Carlo Simulation

- Monte Carlo methods are a class of computational algorithms that use random sampling to obtain numerical results.
- These methods are often used when deterministic methods are too complex or intractable.
- Monte Carlo is especially useful in scenarios involving uncertainty, such as financial modeling, physics, engineering, and estimating mathematical constants.
- Concept: Random Sampling-Monte Carlo simulations rely on generating random numbers to simulate the inputs of a model. Repeated Simulations-The simulation is run many times, each time with a different set of random inputs. Probability Distributions-Uncertainty in input variables is represented as probability distributions. Output Analysis-The results of the simulations are then analyzed to understand the range of possible outcomes, their probabilities, and the impact of uncertainty.

Approximating Distributions

- We can simulate thousands of random samples from a known or unknown distribution to approximate its behavior.
- This includes visualizing histograms, calculating means, medians, standard deviations, and even constructing confidence intervals.
- These simulated distributions help researchers understand the likely outcomes of random processes and guide decision-making.
- In Monte Carlo simulation, approximating distributions involves drawing samples from a probability distribution to estimate its characteristics.
- This is often done by using pseudo-random number generation to transform uniformly distributed numbers into values that follow a desired distribution.
- The generated samples can then be used to estimate quantities like expected values, marginal likelihoods, or other properties of the

Key Concepts in Sampling

- Population: The complete set of data or outcomes.
- Sample: A subset drawn from the population.
- Sampling with/without Replacement: Whether each draw is returned to the pool.
- Random Number Generators: Used to simulate randomness from uniform, normal, or other distributions. Repeated sampling helps identify patterns and converge to population characteristics.

Hypothesis Testing And Estimation

Hypothesis testing and estimation are both crucial tools in statistical inference, but they serve different purposes. Estimation aims to find a value for an unknown population parameter based on sample data, while hypothesis testing evaluates a specific claim about a population parameter.

Example in Sampling

Sampling from a Normal Distribution

Suppose we want to simulate data from a standard normal distribution (mean $= 0$, std $= 1$). By drawing 10,000 samples and plotting them, we expect a bell-shaped curve. Repeating the sampling and calculating the mean each time shows that the average of the sample means will tend to zero as sample size increases, demonstrating the Law of Large Numbers.

Estimating Pi Using Random Sampling

A famous Monte Carlo application is to estimate π. By simulating points in a unit square and checking how many fall inside a quarter circle, we can use the proportion to approximate π. This experiment relies on geometric probability and showcases how randomness can be used to estimate constants with high precision.

Geometric Setup for Estimating Pi

Visualize a square with side length 1 and a quarter circle inscribed in it. The area of the square is 1, and the area of the quarter circle is $\pi/4$. Generate random points (x, y) in the square. If a point satisfies $x^2 + y^2 \leq 1$, it's inside the quarter circle. The ratio of points inside the circle to total points approximates $\pi/4$.

Step-by-Step Process for Estimating Pi

- Generate N random (x,y) points where each is in [0,1].
- Count how many satisfy $x^2 + y^2 \leq 1$
- Let that count be M.
- Estimate $\pi \approx 4 \times \frac{M}{N}$
- As N increases, the estimate becomes more accurate.

This is a classic case of using repeated random sampling for geometric estimation.

Visualizing Simulations

Visualizing simulation results involves using various methods to represent simulation data in a way that's easy to understand and analyze. This can include using graphs, charts, and even interactive 3D visualizations to explore the simulated system's behavior.

- Points inside the circle are plotted in one color (e.g., blue).
- Points outside are plotted in another (e.g., red).
- As the number of points increases (e.g., N = 1000, 10,000, 100,000), the visual estimate of π improves, and the quarter circle becomes clearly defined by the blue points.

Error Analysis and Convergence

Repeated sampling produces estimates close to the true value of π, but with some error.

- $Error = |Estimated\pi - True\pi|$
- The error decreases as the number of samples increases.
- Convergence follows the law:

$$Error \propto \frac{1}{N}$$

This explains why increasing N gives better precision but with diminishing returns.

Beyond Pi-Other Applications

Monte Carlo and random sampling methods are not limited to estimating π.

They are used in:

- Numerical integration over complex domains
- Risk modeling in finance (e.g., Value at Risk)
- Physics simulations (e.g., particle interactions)
- Predictive analytics and machine learning (e.g., bagging)
- Reliability testing and rare event simulation