Monte Carlo Simulations

September 10, 2026

- With *Empirical Probability*, we perform an experiment many times n and count the number of occurrences n_a of an event $\mathcal{A}$
  - The *relative frequency* of occurrence of event $\mathcal{A}$ is n_a/n
  - The *frequency theory of probability* asserts that the relative frequency converges as $n \to \infty$

$$\Pr(\mathcal{A}) = \lim_{n \to \infty} \frac{n_a}{n}$$

- *Axiomatic Probability* is a formal, set-theoretic approach
  - Mathematically construct the sample space and calculate the number of events $\mathcal{A}$
- The two are complementary!

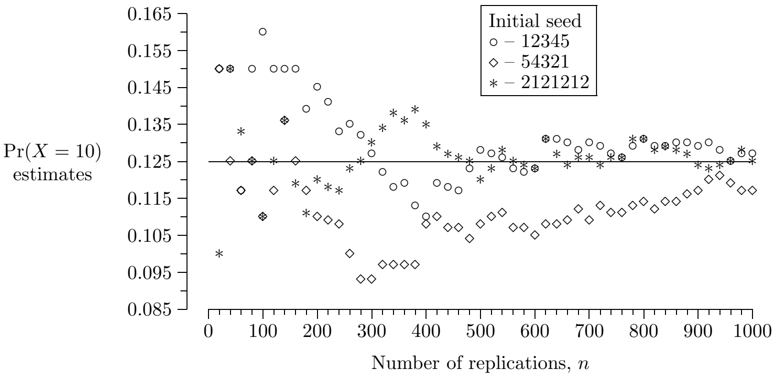
Galileo's Dice

- If three fair dice are rolled, which sum is more likely, a 9 or a 10?
  - There are $6^3 = 216$ possible outcomes

$$\Pr(X = 9) = \frac{25}{216} \cong 0.116 \qquad \text{and} \qquad \Pr(X = 10) = \frac{27}{216} = 0.125$$

- Program `galileo` calculates the probability of each possible sum between 3 and 18
- The drawback of Monte Carlo simulation is that it only produces an estimate
  - Larger n does not guarantee a more accurate estimate

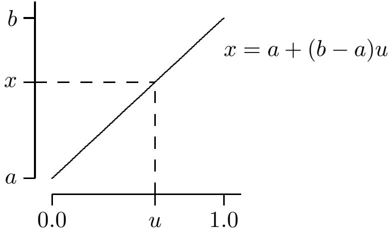
- Frequency probability estimates converge slowly and somewhat erratically



- You should *always* run a Monte Carlo simulation with multiple initial seeds

Random Variates

- A *Random Variate* is an algorithmically generated realization of a random variable
- $u =$ `Random()` generates a *Uniform*$(0, 1)$ random variate
- How can we generate a *Uniform*(a, b) variate?



Generating a Uniform Random Variate

```
double Uniform(double a, double b)      /* use a < b */ {
    return (a + (b - a) * Random());
}
```

Equilikely Random Variates

- *Uniform*$(0, 1)$ random variates can also be used to generate an *Equilikely*(a, b) random variate

$$\begin{aligned}
0 < u < 1 &\iff 0 < (b - a + 1)u < b - a + 1 \\
&\iff 0 \le \lfloor (b - a + 1)u \rfloor \le b - a \\
&\iff a \le a + \lfloor (b - a + 1)u \rfloor \le b \\
&\iff a \le x \le b
\end{aligned}$$

- Specifically, $x = a + \lfloor (b - a + 1)\, u \rfloor$

Generating an Equilikely Random Variate

```
long Equilikely(long a, long b)      /* use a < b */ {
    return (a + (long) ((b - a + 1) * Random()));
}
```

Examples

- **Example 2.3.3** To generate a random variate x that simulates rolling two fair dice and summing the resulting up faces, use

  ```
  x = Equilikely(1, 6) + Equilikely(1, 6);
  ```

  Note that this is *not* equivalent to

  ```
  x = Equilikely(2, 12);
  ```

- **Example 2.3.4** To select an element x at random from the array $a[0]$, $a[1]$, ..., $a[n-1]$ use

  ```
  i = Equilikely(0, n - 1);
  x = a[i];
  ```

$Equilikely(a,b)$ Geometry and $Uniform(a,b+1)$ Composition

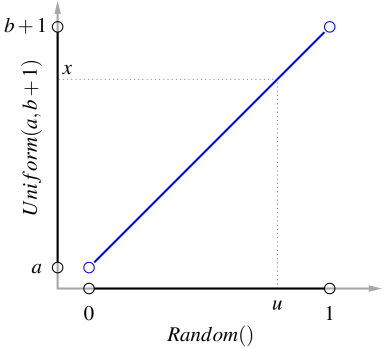
Our definition of

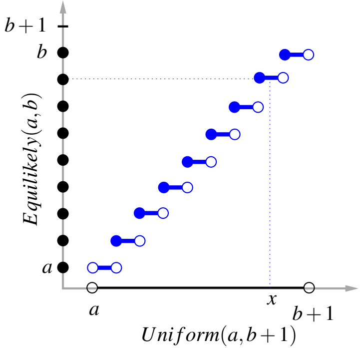
$$Equilikely(a,b) \to a + \lfloor ((b-a+1) \cdot Random()) \rfloor$$

could also be written using $Uniform()$:

$$Equilikely(a,b) \to \lfloor Uniform(a,b+1) \rfloor$$

which hints at the geometry happening with the pRNG values...





Whew! You're ready!

`lga-monte-carlo-probs.pdf`