Uniform Distribution

Arabin Kumar Dey

Assistant Professor

Department of Mathematics
Indian Institute of Technology Guwahati

14-th January, 2013

Uniform(0,1) random numbers are the key to random variate generation in simulation.

Goal : Give an algorithm that produces a sequence of pseudo-random numbers (PRN's) $R_1, R_2, \cdots$ that "appear" to be iid Unif(0,1).

Desired properties of algorithm

- Output appears to be iid Unif(0,1)
- Very fast
- ability to reproduce any sequence it generates

Classes of Unif(0,1) Generators

- table of random numbers
- midsquare (not very useful)
- Fibonacci (not very useful)
- linear congruential (most commonly used in practice)

Random Number Tables

List of digits supplied in tables.

Cumbersome and slow - not very useful.

Once tabled no longer random.

Mid-Square Method (J. von Neumann)

- Idea: Take the middle part of the square of the previous random number. John von Neumann was a brilliant and fun-loving guy, but method is lousy!
- Example: Take $R_i = X_i/10000, \forall i$, where the X_i's are positive integers < 10000.
  Set seed $X_0 = 6632$; then $6632^2 \rightarrow 43\mathbf{9834}24$;
  So $X_1 = 9834$; then $9834^2 \rightarrow 96\mathbf{7075}56$;
  So $X_2 = 7075$, etc,$\cdots$
- Unfortunately, positive serial correlation in R_i's. Also, occasionally degenerates; e.g., consider $X_i = 0003$.

- Fibonacci and Additive Congruential Generators
  These methods are also no good!!
- Take

$$X_i = X_{i-1} + X_{i-2} \mod m, \quad i = 2, \cdots$$

  where $R_i = X_i/m$, m is the modulus, X_0, X_1 are seeds, and $a = b \ \ mod \ \ m$ iff a is the remainder of b/m, e.g., 6 = 13 mod 7.
- **Problems: Small numbers follow small numbers.**
  Also, it's not possible to get $X_{i-1} < X_{i+1} < X_i$ or $X_i < X_{i+1} < X_{i-1}$ (which should occur w.p. 1/3).

Outline

- LCG's are the most widely used generators. These are pretty good when implemented properly.
- $X_i = (aX_{i-1} + c) \mod m$, where X_0 is the seed.
- $R_i = X_i/m, i = 1, 2, \cdots$
- Choose a, c, m carefully to get good statistical quality and long period or cycle length, i.e., time until LCG starts to repeat itself.
- If c = 0, LCG is called a **multiplicative** generator.

Example

Example: For purposes of illustration, consider the LCG

$$X_i = (5X_{i-1} + 3) \mod 8$$

If $X_0 = 0$, we have $X_1 = (5X_0 + 3) \mod 8 = 3$; continuing,

i	0	1	2	3	4	5	6	7	8	9
X_i	0	3	2	5	4	7	6	1	0	3
R_i	0	$\frac{3}{8}$	$\frac{2}{8}$	$\frac{5}{8}$	$\frac{4}{8}$	$\frac{7}{8}$	$\frac{6}{8}$	$\frac{1}{8}$	0	$\frac{3}{8}$

so that the sequence starts repeating with $X_8 = 0$.
This is a *full-period* generator, since it has cycle length m = 8. Generally speaking, full-period is the desired criteria.

Outline

So what can go wrong with LCG's?

- Something like $X_i = (4X_{i-1} + 2) \mod 8$ is not full-period, since it only produces even integers.
- Something like $X_i = (X_{i-1} + 1) \mod 8$ is full-period, but it produces very non-random output: $X_1 = 1$, $X_2 = 2$, $X_3 = 3$, etc.
- In any case, if m is small, you'll get quick cycling whether or not the generator is full period. "Small" could mean anything less than **2 billion** or so!
- And just because m is big, you still have to be careful. In addition above first two points, some subtle problems can arise. Take a look at RANDU $\cdots$.

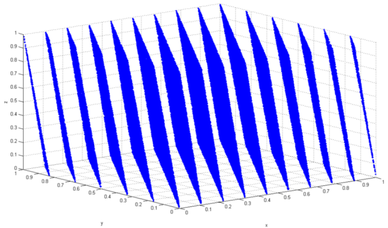
Example

The infamous RANDU generator,

$$X_i = 65539X_{i-1} \mod 2^{31}$$

was popular during the 1960's.

Here's what (R_{i-2}, R_{i-1}, R_i) look like if you plot them in 3-D (stolen from Wikipedia). If they were truly iid Unif(0,1), you'd see the dots randomly dispersed in the unit cube. But instead, the random numbers fall entirely on 15 hyperplanes (not good).



Choosing a Good Generator - Some Theory

Here are some miscellaneous results due to Knuth and others that are helpful in determining the quality of a PRN generator.

Theorem :

The generator $X_{i+1} = aX_i \mod 2^n$ $(n > 3)$ can have cycle length of at most 2^{n-2}. This is achieved when X_0 is odd and $a = 8k + 3$ or $a = 8k + 5$ for some k.

Theorem :

$X_{i+1} = (aX_i + c) \mod m, c > 0$ has full cycle if (i) c and m are relatively prime; (ii) a - 1 is a multiple of every prime which divides m; and (iii) a - 1 is a multiple of 4 if 4 divides m.

Corollary :

$X_{i+1} = (aX_i + c) \mod 2^n$ $(c, n > 1)$ has full cycle if c is odd and $a = 4k + 1$ for some k.

Lots of cycle length results like these.

Example (Banks et al.):

$$X_i = 13X_{i-1} \mod 64.$$

i	X_i	X_i	X_i	X_i
0	1	2	3	4
1	13	26	39	52
2	41	18	56	36
3	21	42	63	20
4	17	34	51	4
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
8	33	2	35	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
16	**1**		**3**	

The minimum period = 4... terrible random numbers! Why does cycling occur so soon? See first theorem.

And here's a theorem that gives a condition for multiplicative generators to be full period.

Therorem

The multiplicative generator $X_i = aX_{i-1} \mod m$, with prime m has full period (m - 1) if and only if
(a) m divides $a^{m-1} - 1$.
(b) For all integers $i < m - 1$, m does not divide $a^i - 1$.

How many such multipliers exist ?
For $m = 2^{31} - 1$, it can be shown that 534,600,000 multipliers yield full period.
Remark: The "best" multiplier with $m = 2^{31} - 1$ is a = 950,706,376 (Fishman and Moore 1986)

Geometric Considerations

The k-tuples $(R_i, \cdots, R_{i+k-1}), i \geq 1$, from multiplicative generators lie on parallel hyperplanes in $[0, 1]^k$.
The following geometric quantities are of interest.

- Minimum number of hyperplanes (in all directions). Find the multiplier that maximizes this number.
- Maximum distance between parallel hyperplanes. Find the multiplier that minimizes this number.
- Minimum Euclidean distance between adjacent k-tuples. Find the multiplier that maximizes this number.

Remark: The RANDU generator is particularly bad since it lies on only 15 hyperplanes.

Choosing a Good Generator - Statistical Tests

We'll look at two classes of tests:

- Goodness-of-fit tests - are the PRN's approximately Unif(0,1) ?
- Independent tests - are the PRN's approximately independent ?

If a particular generator passes both types of tests (in addition to other tests that we won't tell you about), we'll be happy to use the PRN's it generates.
All tests are of the form H_0 (our null hypothesis) vs. H_1 (the alternative hypothesis).

- We regard H_0 as the status quo, so we'll only reject H_0 if we have "ample" evidence against it.
- In fact, we want to avoid incorrect rejections of the null hypothesis. Thus, when we design the test, we'll set the level of significance

$$\alpha \equiv P(Reject\ H_0 | H_0\ is\ true) = P(Type - I\ error)$$

- (typically, $\alpha = 0.05$ or 0.1). We won't worry about Type II error at this point.

χ^2 Goodness of fit Test

- Test $H_0 : R_1, R_2, \cdots R_n \sim Unif(0, 1)$.
- Divide the unit interval into k cells (subintervals). If you choose equi-probable cells $[0, \frac{1}{k}), [\frac{1}{k}, \frac{2}{k}), \cdots, [\frac{k-1}{k}, 1]$, then a particular observation R_j will fall in a particular cell with prob 1/k.
- Tally how many of the n observations fall into the k cells. If $O_i \equiv \#\text{of} R_j$'s in cell i, then (since the R_j's are iid), we can easily see that $O_i \sim Bin(n, \frac{1}{k}), \quad i = 1, 2, \cdots, k$
- Thus, the expected number of R_j's to fall in cell i will be $E_i \equiv E[O_i] = n/k, \quad i = 1, 2, \cdots, k.$

- **We'll reject the null hypothesis H_0 if the O_i's don't match well with the E_i's.**
- The χ^2 goodness of fit statistics is $\chi_0^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$.
- A large value of this statistic indicates a bad fit.
- In fact, we reject the null hypothesis H0 (that the observations are uniform) if

$$\chi_0^2 > \chi_{\alpha,k-1}^2,$$

where $\chi_{\alpha,k-1}^2$ is the appropriate $(1-\alpha)$ quantile from a χ^2 table, i.e.,

$$P(\chi_{k-1}^2 < \chi_{\alpha,k-1}^2) = 1 - \alpha$$

If $\chi_0^2 \leq \chi_{\alpha,k-1}^2$, we fail to reject H_0.

Usual recommendation from baby stats class: For the χ^2 g-o-f test to work, pick k, n such that $E_i \leq 5$ and n at least 30.

Illustrative Example (Banks et al.): n = 100 observations, k = 10 intervals. Thus, $E_i = 10$ for $i = 1, 2, \cdots, 10$. Further, suppose that $O_1 = 13, O_2 = 8, \cdots, O_{10} = 11$. (In other words, 13 observations fell in the cell [0,0.1), etc.)

- Turns out that

$$\chi_0^2 \equiv \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} = 3.4$$

- Let's take $\alpha = 0.05$. Then from χ^2 tables, we have

$$\chi^2_{\alpha,k-1} = \chi^2_{0.05,9}$$

- Since $\chi_0^2 < \chi^2_{\alpha,k-1}$, We fail to reject H_0, and so we'll assume that the observations are approximately uniform.