6.1 Randomness: quantifying uncertainty

- The concepts of uncertainty and randomness have intrigued humanity for a long time.
- The world around us is not deterministic and we are faced continually with chance occurrences.
- Uncertainty is inherent in nature; for example, the behaviour of fundamental physical particles, genes and chromosomes in biology, and individuals in society under stress or strain.
- The methodology for exploring uncertainty involves the use of *random numbers*.

6.2 Pseudo–random numbers

Suppose we need to obtain a list of random digits $0, 1, 2, \ldots, 9$. How might we go about this? There are several options:

Fair ten–sided die

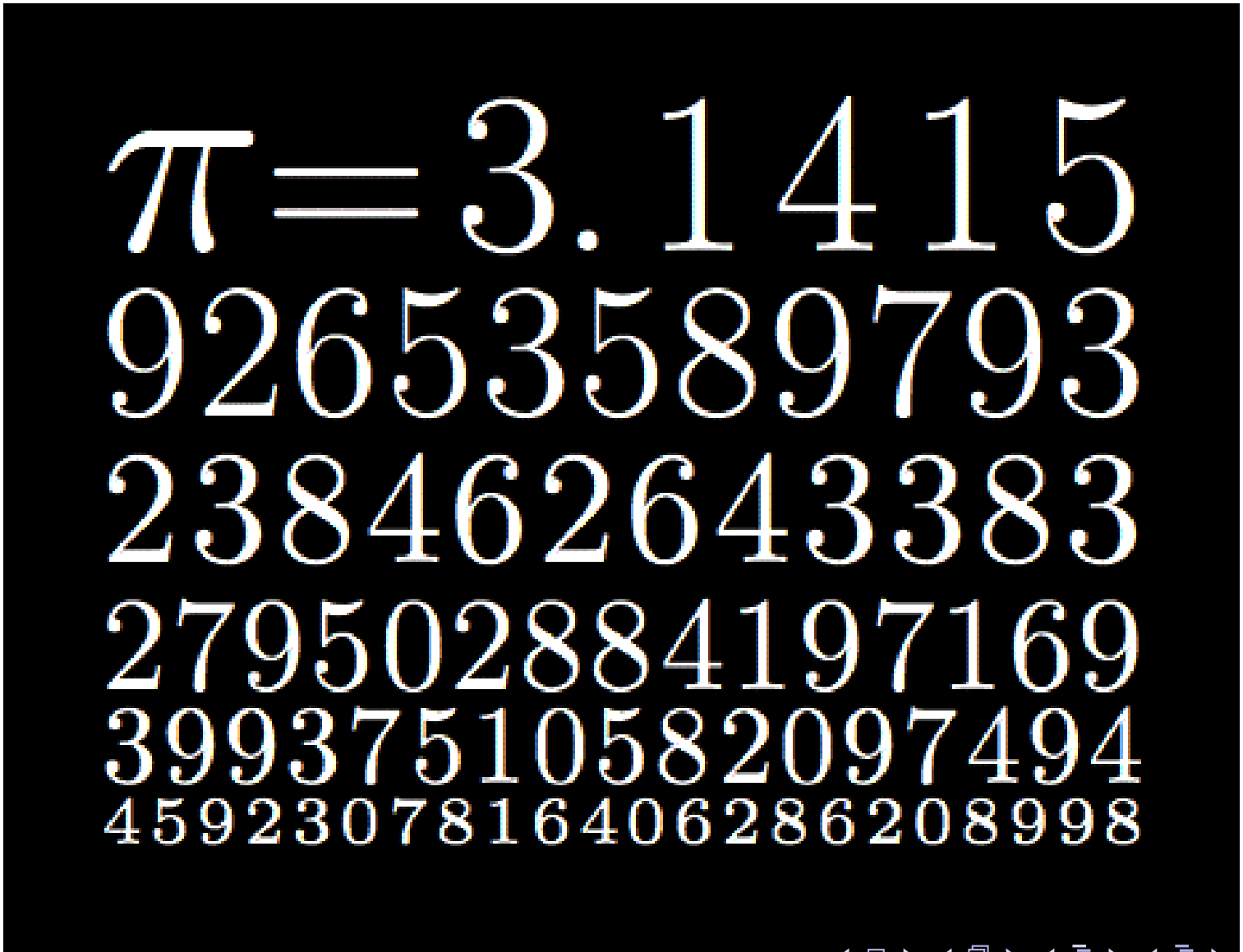
If the sides are labelled from 0 to 9 then tosses of this die will yield the required digits.

Decimal expansion of π

Irrational number – decimal expansion goes on forever, with no pattern!

Decimal expansion of π

Irrational number – decimal expansion goes on forever, with no pattern!



Tosses of a fair coin

Toss a fair coin four times. The following equally likely outcomes could correspond to the integers shown:

Tosses of a fair coin

Toss a fair coin four times. The following equally likely outcomes could correspond to the integers shown:

H H H H	0	H T H T	5
H H H T	1	H T T H	6
H H T H	2	H T T T	7
H H T T	3	T H H H	8
H T H H	4	T H H T	9

The coin has no ‘memory’, and so each block of 4 tosses is independent of any other. If any outcome other than those listed occurs, we can ignore and toss a new set of 4.

Tosses of a fair coin

Toss a fair coin four times. The following equally likely outcomes could correspond to the integers shown:

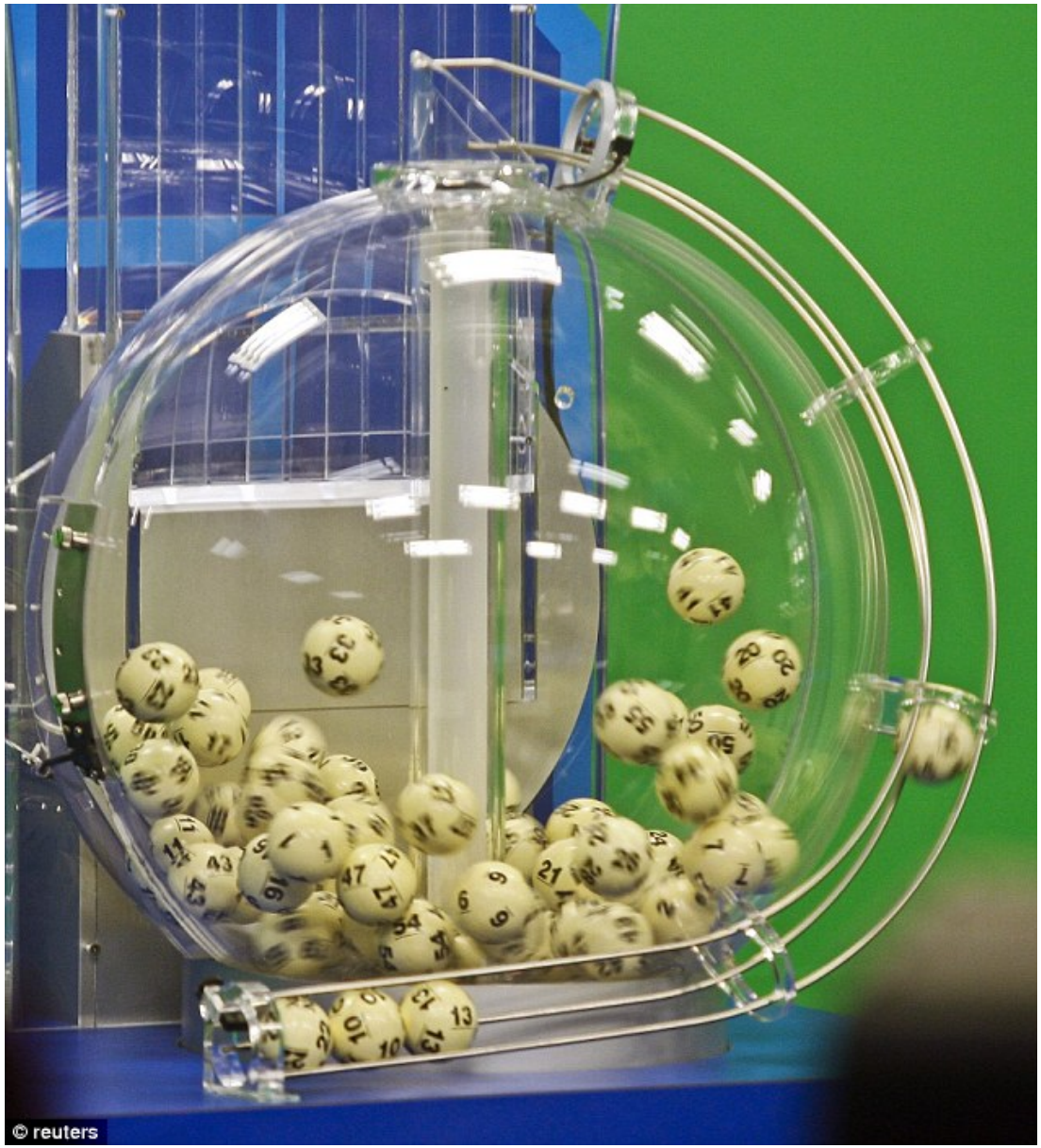
H H H H	0	H T H T	5
H H H T	1	H T T H	6
H H T H	2	H T T T	7
H H T T	3	T H H H	8
H T H H	4	T H H T	9

The coin has no ‘memory’, and so each block of 4 tosses is independent of any other. If any outcome other than those listed occurs, we can ignore and toss a new set of 4.

This method is rather inefficient as a lot of the time a combination of outcomes is rejected!

Other physical devices: Wheels of fortune

Other physical devices: Lottery machines



Other physical devices: Gamma ray counters

- Quantum mechanics predicts that the nuclear decay of atoms is random

Other physical devices: Gamma ray counters

- Quantum mechanics predicts that the nuclear decay of atoms is random
- **Idea:** Use a geiger counter to generate random numbers!

6.2 Pseudo–random numbers

- Mechanical and electronic devices are not reproducible...

6.2 Pseudo–random numbers

- Mechanical and electronic devices are not reproducible...
- So we use *Pseudo–random numbers generators* (RNG)

6.2 Pseudo–random numbers

The German Federal Office for Information Security (*Bundesamt für Sicherheit in der Informationstechnik*, or BSI) has established criteria for quality RNG:

6.2 Pseudo–random numbers

The German Federal Office for Information Security (*Bundesamt für Sicherheit in der Informationstechnik*, or BSI) has established criteria for quality RNG:

1. A sequence of random numbers has a high probability of containing no identical consecutive elements

6.2 Pseudo–random numbers

The German Federal Office for Information Security (*Bundesamt für Sicherheit in der Informationstechnik*, or BSI) has established criteria for quality RNG:

1. A sequence of random numbers has a high probability of containing no identical consecutive elements
2. A sequence of numbers which is indistinguishable from ‘true random’ numbers (tested using statistical tests)

6.2 Pseudo–random numbers

The German Federal Office for Information Security (*Bundesamt für Sicherheit in der Informationstechnik*, or BSI) has established criteria for quality RNG:

1. A sequence of random numbers has a high probability of containing no identical consecutive elements
2. A sequence of numbers which is indistinguishable from ‘true random’ numbers (tested using statistical tests)
3. It should be impossible to calculate – or guess – from any given sub-sequence, any previous or future values in the sequence

6.2 Pseudo–random numbers

The German Federal Office for Information Security (*Bundesamt für Sicherheit in der Informationstechnik*, or BSI) has established criteria for quality RNG:

1. A sequence of random numbers has a high probability of containing no identical consecutive elements
2. A sequence of numbers which is indistinguishable from ‘true random’ numbers (tested using statistical tests)
3. It should be impossible to calculate – or guess – from any given sub-sequence, any previous or future values in the sequence
4. It should be impossible, for all practical purposes, for an attacker to calculate, or guess, the values used in the random number algorithm

6.2 Pseudo–random numbers

The German Federal Office for Information Security (*Bundesamt für Sicherheit in der Informationstechnik*, or BSI) has established criteria for quality RNG:

1. A sequence of random numbers has a high probability of containing no identical consecutive elements
2. A sequence of numbers which is indistinguishable from ‘true random’ numbers (tested using statistical tests)
3. It should be impossible to calculate – or guess – from any given sub-sequence, any previous or future values in the sequence
4. It should be impossible, for all practical purposes, for an attacker to calculate, or guess, the values used in the random number algorithm

Points 3 and 4 are crucial for many applications.

6.3 Congruential generators

- Consider the set $\mathbb{N}^0$ of non–negative integers

6.3 Congruential generators

- Consider the set $\mathbb{N}^0$ of non–negative integers
- That is, $\mathbb{N}^0 = 0, 1, 2, \ldots$
- Let 'mod' represent the *modulo* operation, so that, for $x, m \in \mathbb{N}^0$, $x \neq 0$, (x) mod m means that x is divided by m and the remainder is taken as the result

6.3.1 Example: modulo operations

1. What is 13 mod 4? Answer =
2. What is 19 mod 5? Answer =
3. What is 2008 mod 3? Answer =
4. What is 10,008 mod 11? Answer =

6.3.1 Example: modulo operations

1. What is 13 mod 4? Answer = 1.
2. What is 19 mod 5? Answer =
3. What is 2008 mod 3? Answer =
4. What is 10,008 mod 11? Answer =

6.3.1 Example: modulo operations

1. What is 13 mod 4? Answer = 1.
2. What is 19 mod 5? Answer = 4.
3. What is 2008 mod 3? Answer =
4. What is 10,008 mod 11? Answer =

6.3.1 Example: modulo operations

1. What is 13 mod 4? Answer = 1.
2. What is 19 mod 5? Answer = 4.
3. What is 2008 mod 3? Answer = 1.
4. What is 10,008 mod 11? Answer =

6.3.1 Example: modulo operations

1. What is 13 mod 4? Answer = 1.
2. What is 19 mod 5? Answer = 4.
3. What is 2008 mod 3? Answer = 1.
4. What is 10,008 mod 11? Answer = 9.

6.3.1 Example: modulo operations

Now consider the relation

$$r_i = (ar_{i-1} + b) \bmod m, \quad i = 1, 2, \ldots, m, \tag{6.1}$$

where r_0 is the initial number, known as the *seed*, and a, b, $m \in \mathbb{N}^0$ are the *multiplier*, *additive constant* and *modulo* respectively.

6.3.1 Example: modulo operations

- The modulo operation means that at most m different numbers can be generated before the sequence must repeat – namely the integers $0, 1, 2, \ldots, m-1$.

6.3.1 Example: modulo operations

- The modulo operation means that at most m different numbers can be generated before the sequence must repeat – namely the integers $0, 1, 2, \ldots, m - 1$.
- The actual number of generated numbers is $h \leq m$, called the *period* of the generator.

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57			41			33	61	37	29	93	81	77			1	17	

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57	69		41			33	61	37	29	93	81	77			1	17	

Let's try this in R.

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57	69	73	41			33	61	37	29	93	81	77			1	17	

Let's try this in R.

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57	69	73	41	97		33	61	37	29	93	81	77			1	17	

Let's try this in R.

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57	69	73	41	97	49	33	61	37	29	93	81	77			1	17	

Let's try this in R.

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57	69	73	41	97	49	33	61	37	29	93	81	77	9		1	17	

Let's try this in R.

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57	69	73	41	97	49	33	61	37	29	93	81	77	9	53	1	17	

Let's try this in R.

6.3.2 Example: Congruential generators

Selecting $a = 17$, $b = 0$, $m = 100$, $r_0 = 13$ in relation (6.1) generates the following sequence:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r_i	13	21	57	69	73	41	97	49	33	61	37	29	93	81	77	9	53	1	17	89

Let's try this in R.

6.3.3 Example: Bad random number generators

- In the 1970's a popular random generator used was `RANDU`, where $M = 2^{31}$, $a = 65539$ and $b = 0$.
- Unfortunately this is a spectacularly bad choice of parameters!

6.3.3 Example: Bad random number generators

- In the 1970's a popular random generator used was `RANDU`, where $M = 2^{31}$, $a = 65539$ and $b = 0$.
- Unfortunately this is a spectacularly bad choice of parameters!
- On noting that $a = 65539 = 2^{16} + 3$, then

$$r_{i+1} = ar_i = 65539 \times r_i = (2^{16} + 3)r_i \ .$$

6.3.3 Example: Bad random number generators

So

$$r_{i+2} = a\, r_{i+1} = (2^{16} + 3) \times r_{i+1} = (2^{16} + 3)^2 r_i \ .$$

On expanding the square, we get

6.3.3 Example: Bad random number generators

So

$$r_{i+2} = a\, r_{i+1} = (2^{16} + 3) \times r_{i+1} = (2^{16} + 3)^2 r_i \ .$$

On expanding the square, we get

$$r_{i+2} = (2^{32} + 6 \times 2^{16} + 9) r_i = [6(2^{16} + 3) - 9] r_i = 6 r_{i+1} - 9 r_i \ .$$

Note: all these calculations should be to the mod 2^{31}.

6.3.3 Example: Bad random number generators

So

$$r_{i+2} = a\, r_{i+1} = (2^{16} + 3) \times r_{i+1} = (2^{16} + 3)^2 r_i \ .$$

On expanding the square, we get

$$r_{i+2} = (2^{32} + 6 \times 2^{16} + 9) r_i = [6(2^{16} + 3) - 9] r_i = 6 r_{i+1} - 9 r_i \ .$$

Note: all these calculations should be to the mod 2^{31}. So there is a large correlation between the three points! What does this mean in practice?

6.3.3 Example: Bad random number generators

So

$$r_{i+2} = a\, r_{i+1} = (2^{16} + 3) \times r_{i+1} = (2^{16} + 3)^2 r_i \ .$$

On expanding the square, we get

$$r_{i+2} = (2^{32} + 6 \times 2^{16} + 9) r_i = [6(2^{16} + 3) - 9] r_i = 6 r_{i+1} - 9 r_i \ .$$

Note: all these calculations should be to the mod 2^{31}. So there is a large correlation between the three points! What does this mean in practice? Well, let's consider triplets from this random generator, **as illustrated in R:**

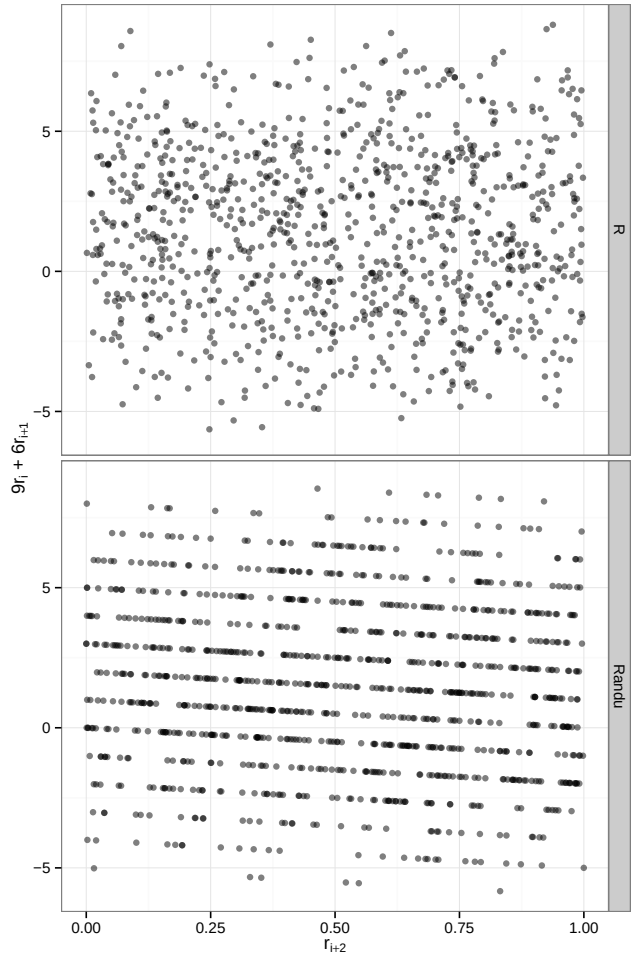


Figure: Comparison of the Randu algorithm and a standard R algorithm

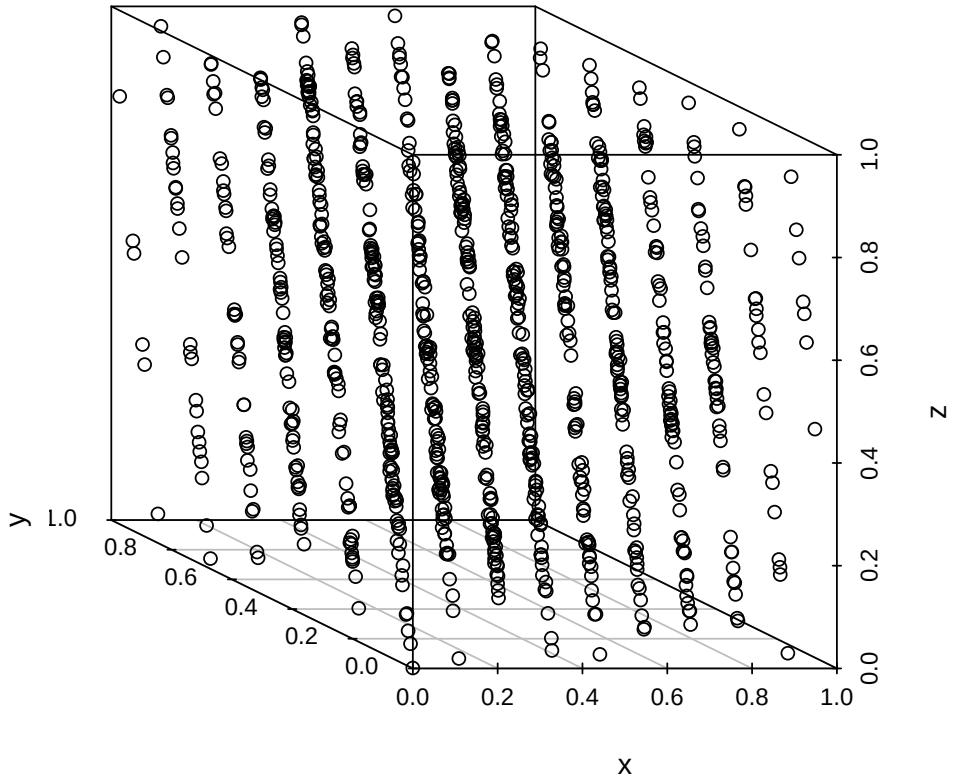


Figure: 3d scatterplot of Randu triples.

6.3.3 Example: Bad random number generators

Some more modern pseudo-RNGs:

6.3.3 Example: Bad random number generators

Some more modern pseudo-RNGs:

- `Mersenne-Twister`

6.3.3 Example: Bad random number generators

Some more modern pseudo-RNGs:

- `Mersenne-Twister`
- `Super-Duper`

Generation of pseudo-random numbers – remarks

1. As computers essentially use numbers to base 2, generators generally use $m = 2^k$, where k is a very large number ($k \in \mathbb{N}$).
2. We want the period of the sequence to be as large as possible.

Generation of pseudo-random numbers – remarks

1. As computers essentially use numbers to base 2, generators generally use $m = 2^k$, where k is a very large number ($k \in \mathbb{N}$).
2. We want the period of the sequence to be as large as possible.

6.4 Assessing random number generators

For the relation

$$r_i = (ar_{i-1} + b) \bmod m, \quad i = 1, 2, \ldots, m\ , \tag{6.2}$$

the maximum period, m, is achieved for $b > 0$ if, and only if:

6.4 Assessing random number generators

For the relation

$$r_i = (ar_{i-1} + b) \bmod m, \quad i = 1, 2, \ldots, m\,, \tag{6.2}$$

the maximum period, m, is achieved for $b > 0$ if, and only if:

(i) b and m have no common factors other than 1;

6.4 Assessing random number generators

For the relation

$$r_i = (ar_{i-1} + b) \bmod m, \quad i = 1, 2, \ldots, m\ , \tag{6.2}$$

the maximum period, m, is achieved for $b > 0$ if, and only if:

(i) b and m have no common factors other than 1;

(ii) $(a - 1)$ is a multiple of every prime number that divides m;

6.4 Assessing random number generators

For the relation

$$r_i = (ar_{i-1} + b) \bmod m, \quad i = 1, 2, \ldots, m\ , \tag{6.2}$$

the maximum period, m, is achieved for $b > 0$ if, and only if:

(i) b and m have no common factors other than 1;

(ii) $(a - 1)$ is a multiple of every prime number that divides m;

(iii) $(a - 1)$ is a multiple of 4 if m is a multiple of 4.

Remarks

1. If $m = 2^k$, then if $a = 4c + 1$ for some positive integer c, (ii) and (iii) will hold.
2. Similarly, for (i) to be true then b must be a positive odd integer if $m = 2^k$.
3. As a real example, the Numerical Algorithms Group (NAG) `Fortran` library uses $k = 59$, $b = 0$ and $a = 13^{13}$ in one of it's random number generator.

6.4.1 Example: Maximum periods of RNG

Check to see if the maximum period can be achieved if the Congruential method with the following parameters is used to generate a sequence of pseudo–random numbers:

$a = 16$, $b = 5$, $m = 20$

All three conditions must be satisfied for the maximum period to be obtained, so we check each in turn.

- Condition (i): False. $b = 5$ and $m = 20$ have a common factor, 5. Hence the maximum period is *not* achieved.

$a = 16$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.

$a = 16$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): False. $\left(a - 1\right) = 15$, and this is *not* divisible by 2. But 20 *is* divisible by 2.

$a = 16$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): False. $(a - 1) = 15$, and this is *not* divisible by 2. But 20 *is* divisible by 2.
- Hence the maximum period is *not* achieved.

$a = 11, b = 3, m = 20$

$a = 11$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.

$a = 11$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): True. $(a - 1) = 10$, which is divisible by both 2 and 5, which are the only primes which divide 20.

$a = 11$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): True. $(a - 1) = 10$, which is divisible by both 2 and 5, which are the only primes which divide 20.
- Condition (iii): False. $m = 20$ is a multiple of 4, but $(a - 1) = 10$ isn't.

$a = 11$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): True. $(a-1) = 10$, which is divisible by both 2 and 5, which are the only primes which divide 20.
- Condition (iii): False. $m = 20$ is a multiple of 4, but $(a-1) = 10$ isn't.
- Hence the maximum period is *not* achieved.

$a = 21, b = 3, m = 20$

$a = 21$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.

$a = 21$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): True. $(a - 1) = 20$, which is divisible by both 2 and 5, which are the only primes which divide 20.

$a = 21$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): True. $(a - 1) = 20$, which is divisible by both 2 and 5, which are the only primes which divide 20.
- Condition (iii): True. $m = 20$ is a multiple of 4, and $(a - 1) = 20$ is too.

$a = 21$, $b = 3$, $m = 20$

- Condition (i): True. b and m have no common factors other than 1.
- Condition (ii): True. $(a - 1) = 20$, which is divisible by both 2 and 5, which are the only primes which divide 20.
- Condition (iii): True. $m = 20$ is a multiple of 4, and $(a - 1) = 20$ is too.
- Hence the maximum period of 20 *is* achieved.

6.5.1 The `runif` function

```
> runif(n, min=0, max=1)
```

- This function will generate `n` random numbers between the values of `min` and `max`.
- If the arguments `min` or `max` are omitted, then the default values are 0 and 1 respectively.

6.5.1 The runif function

For example,

```
> runif(1)
[1] 0.2729965
> runif(1)
[1] 0.9990863
> runif(5)
[1] 0.8122783 0.9069950 0.1731072 0.3454292 0.6102412
> runif(1, 6, 7)
[1] 6.427512
```

6.5.1 The runif function

```
> set.seed(12345)
> runif(1)
[1] 0.7209039
> runif(1)
[1] 0.8757732
```

6.5.1 The runif function

```
> set.seed(12345)
> runif(1)
[1] 0.7209039
> runif(1)
[1] 0.8757732
```

```
> set.seed(12345)
> runif(1)
[1] 0.7209039
```

6.5.2 The `sample` function

Another important R function that we will use is the `sample` function:

```
> sample(x, size, replace = FALSE, prob = NULL)
```

6.5.2 The `sample` function

This takes the following arguments

- `x`: a list of values
- `size`: non-negative integer giving the number of items to choose.
- `replace`: Should sampling be with replacement? Default: `FALSE`.
- `prob`: A vector of probability weights. Default: All values equally likely.

Example usage of `sample`

Suppose we wish to sample five numbers from $\{1, 2, 3, 4, 5, 6\}$, then

```
> set.seed(1)
> x = c(1, 2, 3, 4, 5, 6)
> sample(x, 5)
[1] 2 6 3 4 1
```

We can also sample with replacement:

```
> sample(x, 5, replace=TRUE)
[1] 6 6 4 4 1
```

This means that values **may** appear more than once.

6.5.3 Simulating the Captial One Cup draw

We are in the semi-finals of the *Capital One* Cup, and need to organise the draw for the final stage. The remaining teams are:

Manchester Utd, Mancehster City, Sunderland, West Ham.

6.5.3 Simulating the Captial One Cup draw

Here's how we do this in R:

```
> set.seed(3)
> teams = c("Man Utd", "Man City", "Sunderland", "West
Ham")
> sample(teams, 4)
[1] "Man Utd"    "Sunderland" "West Ham"   "Man City"
```

So we have 'Man Utd vs Sunderland' and 'West Ham vs Man City'.

6.5.3 Simulating the Capital One Cup draw

However, if we think Sunderland are likely to get beaten by Man Utd, we can rig the voting:

```
> prob_weights = c(0.4, 0.4, 0.05, 0.2)
> sample(teams, 4, prob=prob_weights)
[1] "Man Utd"    "Man City"   "West Ham"
"Sunderland"
```

That's better!

6.5.3 Simulating the Capital One Cup draw

Tomorrow's practical

Part VII

Simulating Discrete Random Numbers

7.1 Simulating a Bernoulli random variable

The Bernoulli random variable $I \sim \mathrm{Bern}(p)$ has already been encountered in MAS1341. It is perhaps the simplest random variable and has the probability mass function

i	0	1
$\Pr[I = i]$	$1 - p$	p

7.1 Simulating a Bernoulli random variable

To simulate such a quantity, we generate an observation u from a uniform $U(0,1)$ distribution and set

$$I = \begin{cases} 0 & \text{if } u < 1-p \\ 1 & \text{if } u \geq 1-p. \end{cases} .$$

7.1.1 Example: Bernoulli random numbers

Suppose we generate a number from a uniform $U(0, 1)$ distribution

$$0.332, 0.739, 0.653, 0.110, 0.587, 0.144$$

and we wish to use these to simulate six independent observations from I, a Bern(0.8) random variable.

7.1.1 Example: Bernoulli random numbers

Suppose we generate a number from a uniform $U(0, 1)$ distribution

$$0.332, 0.739, 0.653, 0.110, 0.587, 0.144$$

and we wish to use these to simulate six independent observations from I, a Bern(0.8) random variable. Here $p = 0.8$, so we convert using:

$$I = \begin{cases} 0 & \text{if } u < 1 - p = 0.2 \\ 1 & \text{if } u \geq 1 - p = 0.8. \end{cases}$$

to obtain the sequence 1, 1, 1, 0, 1, 0 as our Bernoulli random sample.

7.1.2 Using R to simulate a Bernoulli R.V.

To simulate a Bernoulli variable using R is straightforward. The easiest way is just to use the sample command:

```
> p = 0.5
> sample(c(0, 1), 1, prob=c(1-p, p), replace=TRUE)
[1] 0
> sample(c(0, 1), 10, prob=c(1-p, p), replace=TRUE)
 [1] 1 1 0 1 0 1 0 1 1 1
```

7.2 Simulation of discrete random variables

Suppose we want to simulate a discrete random variable X which has probability mass function

$$\Pr[X = x_j] = p_j \quad \text{for } j = 1, 2, \ldots,$$

and where $\sum_{\text{all}} p_j = 1$.

7.2 Simulation of discrete random variables

To accomplish this we first simulate a value u from a $U(0, 1)$ distribution and set

7.2 Simulation of discrete random variables

To accomplish this we first simulate a value u from a $U(0,1)$ distribution and set

$$X = \begin{cases} x_1 & \text{if } u < p_1 \end{cases}$$

7.2 Simulation of discrete random variables

To accomplish this we first simulate a value u from a $U(0,1)$ distribution and set

$$X = \begin{cases} x_1 & \text{if } u < p_1 \\ x_2 & \text{if } p_1 \leq u < p_1 + p_2 \end{cases}$$

7.2 Simulation of discrete random variables

To accomplish this we first simulate a value u from a $U(0,1)$ distribution and set

$$X = \begin{cases} x_1 & \text{if } u < p_1 \\ x_2 & \text{if } p_1 \leq u < p_1 + p_2 \\ x_3 & \text{if } p_1 + p_2 \leq u < p_1 + p_2 + p_3 \end{cases}$$

7.2 Simulation of discrete random variables

To accomplish this we first simulate a value u from a $U(0,1)$ distribution and set

$$X = \begin{cases} x_1 & \text{if } u < p_1 \\ x_2 & \text{if } p_1 \le u < p_1 + p_2 \\ x_3 & \text{if } p_1 + p_2 \le u < p_1 + p_2 + p_3 \\ \vdots & \vdots \end{cases}$$

7.2 Simulation of discrete random variables

To accomplish this we first simulate a value u from a $U(0,1)$ distribution and set

$$X = \begin{cases} x_1 & \text{if } u < p_1 \\ x_2 & \text{if } p_1 \le u < p_1 + p_2 \\ x_3 & \text{if } p_1 + p_2 \le u < p_1 + p_2 + p_3 \\ \vdots & \vdots \\ x_j & \text{if } \sum_{i=1}^{j-1} p_i \le u < \sum_{i=1}^{j} p_i \end{cases}$$

7.2 Simulation of discrete random variables

To accomplish this we first simulate a value u from a $U(0,1)$ distribution and set

$$X = \begin{cases} x_1 & \text{if } u < p_1 \\ x_2 & \text{if } p_1 \le u < p_1 + p_2 \\ x_3 & \text{if } p_1 + p_2 \le u < p_1 + p_2 + p_3 \\ \vdots & \vdots \\ x_j & \text{if } \sum_{i=1}^{j-1} p_i \le u < \sum_{i=1}^{j} p_i \\ \vdots & \vdots \end{cases}$$

7.2 Simulation of discrete random variables

Now, for a Uniform random variable u, and for $0 < a < b < 1$, it is the case that $\Pr[a \leq U < b] = b - a$.

7.2 Simulation of discrete random variables

Now, for a Uniform random variable u, and for $0 < a < b < 1$, it is the case that $\Pr[a \leq U < b] = b - a$.Thus,

$$\Pr[X = x_j] \quad = \quad \Pr\left[\sum_{i=1}^{j-1} p_i \leq U < \sum_{i=1}^{j} p_i\right]$$

7.2 Simulation of discrete random variables

Now, for a Uniform random variable u, and for $0 < a < b < 1$, it is the case that $\Pr[a \leq U < b] = b - a$.Thus,

$$\begin{aligned} \Pr[X = x_j] &= \Pr\left[\sum_{i=1}^{j-1} p_i \leq U < \sum_{i=1}^{j} p_i\right] \\ &= \sum_{i=1}^{j} p_i - \sum_{i=1}^{j-1} p_i \end{aligned}$$

7.2 Simulation of discrete random variables

Now, for a Uniform random variable u, and for $0 < a < b < 1$, it is the case that $\Pr[a \leq U < b] = b - a$.Thus,

$$\begin{aligned} \Pr[X = x_j] &= \Pr\left[\sum_{i=1}^{j-1} p_i \leq U < \sum_{i=1}^{j} p_i\right] \\ &= \sum_{i=1}^{j} p_i - \sum_{i=1}^{j-1} p_i \\ &= p_j \,. \end{aligned}$$

as required.

7.2.1 Example: discrete random numbers

Simulate a random variable with the following probability mass function:

x	1	2	3	4
$\Pr[X = x]$	0.2	0.15	0.25	0.4

Solution

We calcaluate the CDF of the distribution

x	1	2	3	4
$\Pr[X = x]$	0.2	0.15	0.25	0.4

Solution

We calcaluate the CDF of the distribution

x	1	2	3	4
$\Pr[X = x]$	0.2	0.15	0.25	0.4
$\Pr[X \leq x]$	0.2	0.35	0.60	1.0

Solution

We calcaluate the CDF of the distribution

x	1	2	3	4
$\Pr[X = x]$	0.2	0.15	0.25	0.4
$\Pr[X \leq x]$	0.2	0.35	0.60	1.0

Then we generate an observation u from $U \sim \mathrm{Uniform}(0, 1)$, so

- if $u < 0.2$, set $X = 1$;

Solution

We calcaluate the CDF of the distribution

x	1	2	3	4
$\Pr[X = x]$	0.2	0.15	0.25	0.4
$\Pr[X \leq x]$	0.2	0.35	0.60	1.0

Then we generate an observation u from $U \sim \mathrm{Uniform}(0, 1)$, so

- if $u < 0.2$, set $X = 1$;
- if $0.2 \leq u < (0.2 + 0.15) = 0.35$, set $X = 2$;

Solution

We calcaluate the CDF of the distribution

x	1	2	3	4
$\Pr[X = x]$	0.2	0.15	0.25	0.4
$\Pr[X \leq x]$	0.2	0.35	0.60	1.0

Then we generate an observation u from $U \sim \mathrm{Uniform}(0, 1)$, so

- if $u < 0.2$, set $X = 1$;
- if $0.2 \leq u < (0.2 + 0.15) = 0.35$, set $X = 2$;
- if $0.35 \leq u < (0.2 + 0.15 + 0.25) = 0.6$, set $X = 3$;

Solution

We calcaluate the CDF of the distribution

x	1	2	3	4
$\Pr[X = x]$	0.2	0.15	0.25	0.4
$\Pr[X \leq x]$	0.2	0.35	0.60	1.0

Then we generate an observation u from $U \sim \text{Uniform}(0, 1)$, so

- if $u < 0.2$, set $X = 1$;
- if $0.2 \leq u < (0.2 + 0.15) = 0.35$, set $X = 2$;
- if $0.35 \leq u < (0.2 + 0.15 + 0.25) = 0.6$, set $X = 3$;
- if $u \geq 0.6$, set $X = 4$.

Solution

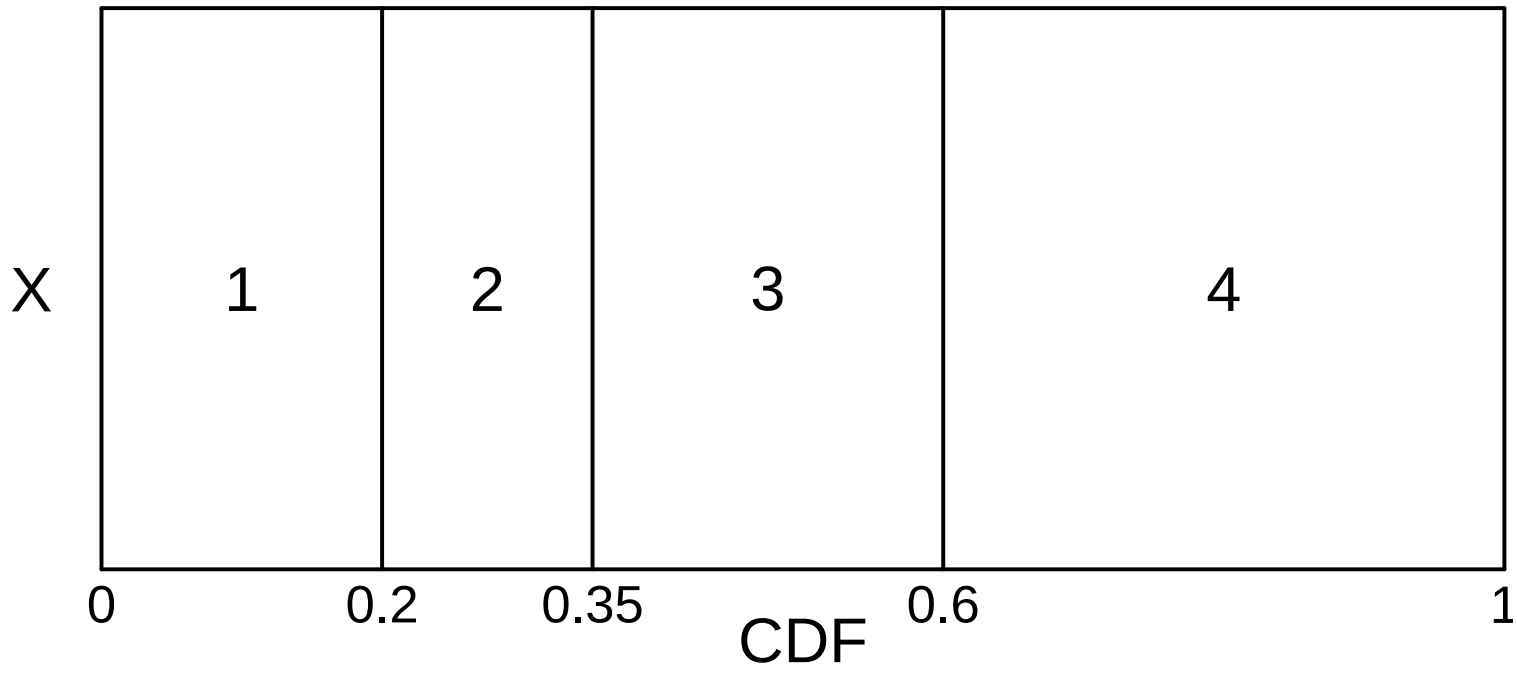


Figure: Simulating discrete random numbers.

Using R

To simulate the above distribution in R, there a two (obvious) methods that we can use: the `sample` command

```
> x = c(1, 2, 3, 4)
> prob = c(0.2, 0.15, 0.25, 0.4)
> sum(prob)
[1] 1
> sample(x, 1, prob, replace=TRUE)
[1] 4
```

Using R

To simulate the above distribution in R, there a two (obvious) methods that we can use: the `sample` command

```
> x = c(1, 2, 3, 4)
> prob = c(0.2, 0.15, 0.25, 0.4)
> sum(prob)
[1] 1
> sample(x, 1, prob, replace=TRUE)
[1] 4
```

or use a bunch of `if` statements:

```
> u = runif(1)
> if(u <= 0.2) {
+   X = 1
+ } else if(u <= (0.2+0.15)) {
+   X = 2
+ } else if(u <= (0.2+0.15+0.25)) {
+   X = 3
```

7.2.2 Simulating a Poisson random variable

For the uniform random numbers,

$$0.253,\ 0.588,\ 0.789$$

simulate three random numbers from a Poisson distribution with mean $\lambda = 2$.

Solution

The pdf of the poisson distribution is

$$Pr[X = x] = \frac{e^{-\lambda}\lambda^{x}}{x!}$$

where $\lambda > 0$.

Solution

The pdf of the poisson distribution is

$$Pr[X = x] = \frac{e^{-\lambda}\lambda^{x}}{x!}$$

where $\lambda > 0$. So we have

x	0	1	2	3	4
$\Pr[X = x]$	0.135	0.271	0.271	0.180	0.090

Solution

The pdf of the poisson distribution is

$$Pr[X = x] = \frac{e^{-\lambda}\lambda^{x}}{x!}$$

where $\lambda > 0$. So we have

x	0	1	2	3	4
$\Pr[X = x]$	0.135	0.271	0.271	0.180	0.090
$\Pr[X \leq x]$	0.135	0.406	0.677	0.857	0.947

Solution

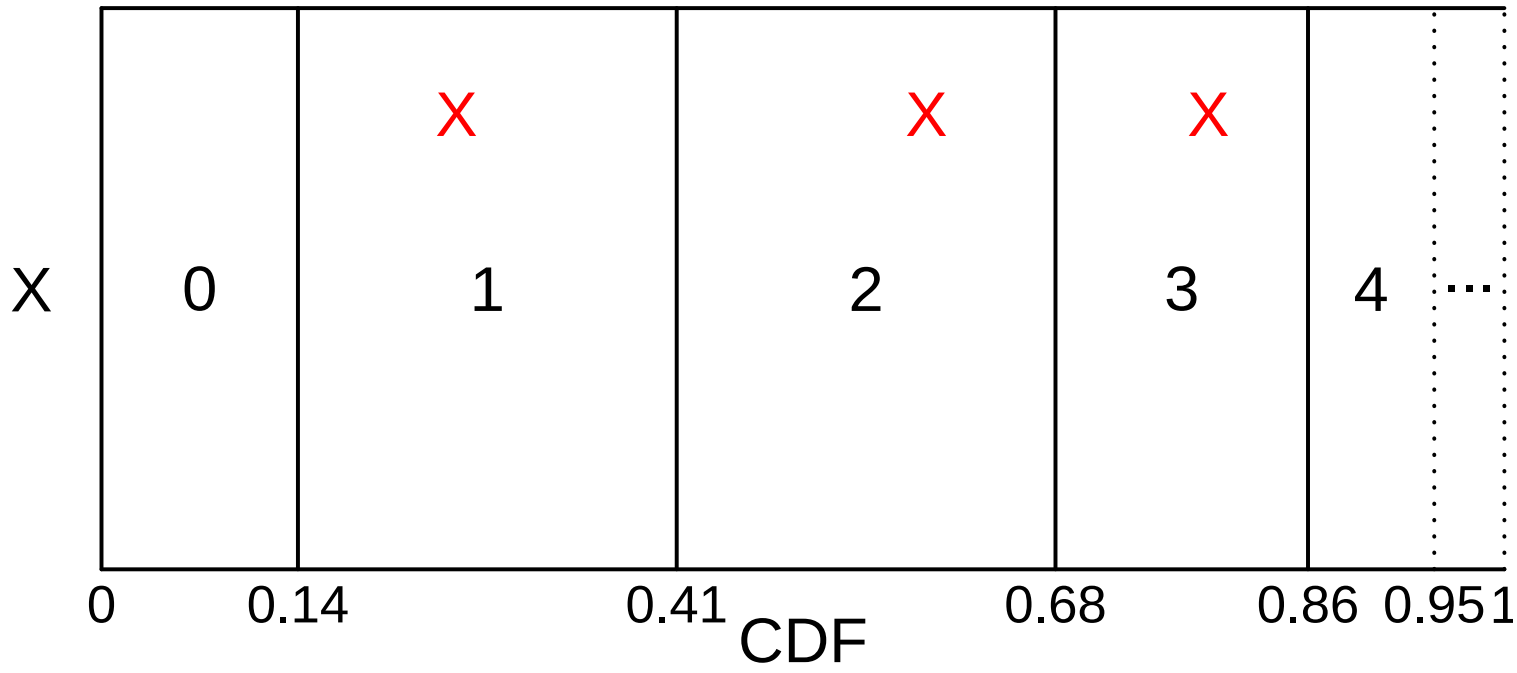


Figure: Diagram illustrating on simulating discrete random numbers from the Poisson distribution.

Hence our random numbers are 1, 2 and 3.

7.3 Simulating a Geometric random variable

Let $X \sim \mathrm{Geom}(p)$, so that X takes the random values $1, 2, 3, \ldots$ with probabilities p, $(1-p)p$, $(1-p)^2 p$, ...

7.3 Simulating a Geometric random variable

Let $X \sim \mathrm{Geom}(p)$, so that X takes the random values $1, 2, 3, \ldots$ with probabilities p, $(1-p)p$, $(1-p)^2 p$, ... Hence,

$$\Pr[X = k] = p(1-p)^{k-1} \ .$$

The Geometric distribution is the distribution of the number of independent Bernoulli trials until the first success is encountered. For each individual trial, the probability of success is p.

7.3 Simulating a Geometric random variable

We know that we will obtain the value $X = k$ (say) when simulating an observation from X using a value of U if

$$\Pr[X = 1] + \Pr[X = 2] + \ldots + \Pr[X = k - 1] \leq U < \Pr[X = 1] + \ldots + \Pr[X = k],$$

i.e.

7.3 Simulating a Geometric random variable

We know that we will obtain the value $X = k$ (say) when simulating an observation from X using a value of U if

$$\Pr[X = 1] + \Pr[X = 2] + \ldots + \Pr[X = k-1] \leq U < \Pr[X = 1] + \ldots + \Pr[X = k],$$

i.e.

$$\sum_{j=1}^{k-1} \Pr[X = j] \leq U < \sum_{j=1}^{k} \Pr[X = j],$$

i.e.

7.3 Simulating a Geometric random variable

We know that we will obtain the value $X = k$ (say) when simulating an observation from X using a value of U if

$$\Pr[X = 1] + \Pr[X = 2] + \ldots + \Pr[X = k-1] \leq U < \Pr[X = 1] + \ldots + \Pr[X = k],$$

i.e.

$$\sum_{j=1}^{k-1} \Pr[X = j] \leq U < \sum_{j=1}^{k} \Pr[X = j],$$

i.e.

7.3 Simulating a Geometric random variable

We know that we will obtain the value $X = k$ (say) when simulating an observation from X using a value of U if

$$\Pr[X = 1] + \Pr[X = 2] + \ldots + \Pr[X = k-1] \leq U < \Pr[X = 1] + \ldots + \Pr[X = k],$$

i.e.

$$\sum_{j=1}^{k-1} \Pr[X = j] \leq U < \sum_{j=1}^{k} \Pr[X = j],$$

i.e.

$$\sum_{j=1}^{k-1} (1-p)^{j-1} p \leq U < \sum_{j=1}^{k} (1-p)^{j-1} p\,. \tag{7.1}$$

since

$$\Pr[X = k] = p(1-p)^{k-1}$$

7.3 Simulating a Geometric random variable

We know that we will obtain the value $X = k$ (say) when simulating an observation from X using a value of U if

$$\Pr[X = 1] + \Pr[X = 2] + \ldots + \Pr[X = k-1] \leq U < \Pr[X = 1] + \ldots + \Pr[X = k],$$

i.e.

$$\sum_{j=1}^{k-1} \Pr[X = j] \leq U < \sum_{j=1}^{k} \Pr[X = j],$$

i.e.

$$\sum_{j=1}^{k-1} (1-p)^{j-1} p \leq U < \sum_{j=1}^{k} (1-p)^{j-1} p \,. \tag{7.1}$$

since

$$\Pr[X = k] = p(1-p)^{k-1}$$

7.3 Simulating a Geometric random variable

We know (for $0 < r < 1$) that

$$r \sum_{k=0}^{n} (1-r)^k = 1 - (1-r)^{n+1}$$

i.e. a Geometric progression.

7.3 Simulating a Geometric random variable

So

$$\sum_{j=1}^{k-1} (1-p)^{j-1} p \quad =$$

7.3 Simulating a Geometric random variable

So

$$\sum_{j=1}^{k-1}(1-p)^{j-1}p \quad = \quad p\sum_{j=1}^{k-1}(1-p)^{j-1}$$

7.3 Simulating a Geometric random variable

So

$$\begin{aligned}\sum_{j=1}^{k-1}(1-p)^{j-1}p &= p\sum_{j=1}^{k-1}(1-p)^{j-1} \\ &= p\sum_{j=0}^{k-2}(1-p)^{j}\end{aligned}$$

7.3 Simulating a Geometric random variable

So

$$\begin{aligned}\sum_{j=1}^{k-1}(1-p)^{j-1}p &= p\sum_{j=1}^{k-1}(1-p)^{j-1} \\ &= p\sum_{j=0}^{k-2}(1-p)^{j} \\ &= 1-(1-p)^{k-1}\,.\end{aligned}$$

7.3 Simulating a Geometric random variable

Hence for expression 7.1 we have

$$1 - (1-p)^{k-1} \leq U < 1 - (1-p)^{k},$$

7.3 Simulating a Geometric random variable

Hence for expression 7.1 we have

$$1 - (1-p)^{k-1} \leq U < 1 - (1-p)^{k},$$

i.e.

$$(1-p)^{k-1} \geq 1 - U > (1-p)^{k},$$

7.3 Simulating a Geometric random variable

Hence for expression 7.1 we have

$$1 - (1-p)^{k-1} \leq U < 1 - (1-p)^{k},$$

i.e.

$$(1-p)^{k-1} \geq 1 - U > (1-p)^{k},$$

i.e.

$$(k-1)\ln(1-p) \geq \ln(1-U) > k\ln(1-p),$$

7.3 Simulating a Geometric random variable

Hence for expression 7.1 we have

$$1 - (1-p)^{k-1} \leq U < 1 - (1-p)^{k},$$

i.e.

$$(1-p)^{k-1} \geq 1 - U > (1-p)^{k},$$

i.e.

$$(k-1)\ln(1-p) \geq \ln(1-U) > k\ln(1-p),$$

i.e.

$$k - 1 \leq \frac{\ln(1-U)}{\ln(1-p)} < k,$$

with the inequality sign reversing since $\ln(1-p) < 0$.

7.3 Simulating a Geometric random variable

Thus, we observe $X = k$ if

1. $X > \frac{\ln(1-U)}{\ln(1-p)}$

7.3 Simulating a Geometric random variable

Thus, we observe $X = k$ if

1. $X > \frac{\ln(1-U)}{\ln(1-p)}$
2. $X - 1 \leq \frac{\ln(1-U)}{\ln(1-p)}$, i.e. $X \leq 1 + \frac{\ln(1-U)}{\ln(1-p)}$.

Both (1) and (2) are satisfied by $X = 1 + \left\lfloor \frac{\ln(1-U)}{\ln(1-p)} \right\rfloor$.

7.3.1 Examples: geometric random numbers

Given $u = 0.2179$, simulate a $\mathrm{Geom}(0.2)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.2$, we have

$$X = 1 + \left\lfloor \frac{\ln(1-U)}{\ln(1-p)} \right\rfloor$$

7.3.1 Examples: geometric random numbers

Given $u = 0.2179$, simulate a $\mathrm{Geom}(0.2)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.2$, we have

$$\begin{aligned} X &= 1 + \left\lfloor \frac{\ln(1-U)}{\ln(1-p)} \right\rfloor \\ &= 1 + \left\lfloor \frac{\ln(1-0.2179)}{\ln(1-0.2)} \right\rfloor \end{aligned}$$

7.3.1 Examples: geometric random numbers

Given $u = 0.2179$, simulate a $\mathrm{Geom}(0.2)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.2$, we have

$$\begin{aligned} X &= 1 + \left\lfloor \frac{\ln(1-U)}{\ln(1-p)} \right\rfloor \\ &= 1 + \left\lfloor \frac{\ln(1-0.2179)}{\ln(1-0.2)} \right\rfloor \\ &= 1 + \left\lfloor \frac{-0.2458}{-0.2231} \right\rfloor \end{aligned}$$

7.3.1 Examples: geometric random numbers

Given $u = 0.2179$, simulate a $\mathrm{Geom}(0.2)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.2$, we have

$$\begin{aligned} X &= 1 + \left\lfloor \frac{\ln(1-U)}{\ln(1-p)} \right\rfloor \\ &= 1 + \left\lfloor \frac{\ln(1-0.2179)}{\ln(1-0.2)} \right\rfloor \\ &= 1 + \left\lfloor \frac{-0.2458}{-0.2231} \right\rfloor \\ &= 1 + \lfloor 1.1014 \rfloor \end{aligned}$$

7.3.1 Examples: geometric random numbers

Given $u = 0.2179$, simulate a $\mathrm{Geom}(0.2)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.2$, we have

$$\begin{aligned} X &= 1 + \left\lfloor \frac{\ln(1-U)}{\ln(1-p)} \right\rfloor \\ &= 1 + \left\lfloor \frac{\ln(1-0.2179)}{\ln(1-0.2)} \right\rfloor \\ &= 1 + \left\lfloor \frac{-0.2458}{-0.2231} \right\rfloor \\ &= 1 + \lfloor 1.1014 \rfloor \\ &= 1 + 1 = 2. \end{aligned}$$

7.3.1 Question 2

Given $u = 0.8923$, simulate a $\mathrm{Geom}(0.4)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.4$, we have

$$X = 1 + \left\lfloor \frac{\ln(1 - 0.8923)}{\ln(1 - 0.4)} \right\rfloor$$

7.3.1 Question 2

Given $u = 0.8923$, simulate a $\mathrm{Geom}(0.4)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.4$, we have

$$\begin{aligned} X &= 1 + \left\lfloor \frac{\ln(1 - 0.8923)}{\ln(1 - 0.4)} \right\rfloor \\ &= 1 + \left\lfloor \frac{-2.228406}{-0.5108256} \right\rfloor \end{aligned}$$

7.3.1 Question 2

Given $u = 0.8923$, simulate a $\mathrm{Geom}(0.4)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.4$, we have

$$\begin{aligned} X &= 1 + \left\lfloor \frac{\ln(1 - 0.8923)}{\ln(1 - 0.4)} \right\rfloor \\ &= 1 + \left\lfloor \frac{-2.228406}{-0.5108256} \right\rfloor \\ &= 1 + \lfloor 4.362361 \rfloor \end{aligned}$$

7.3.1 Question 2

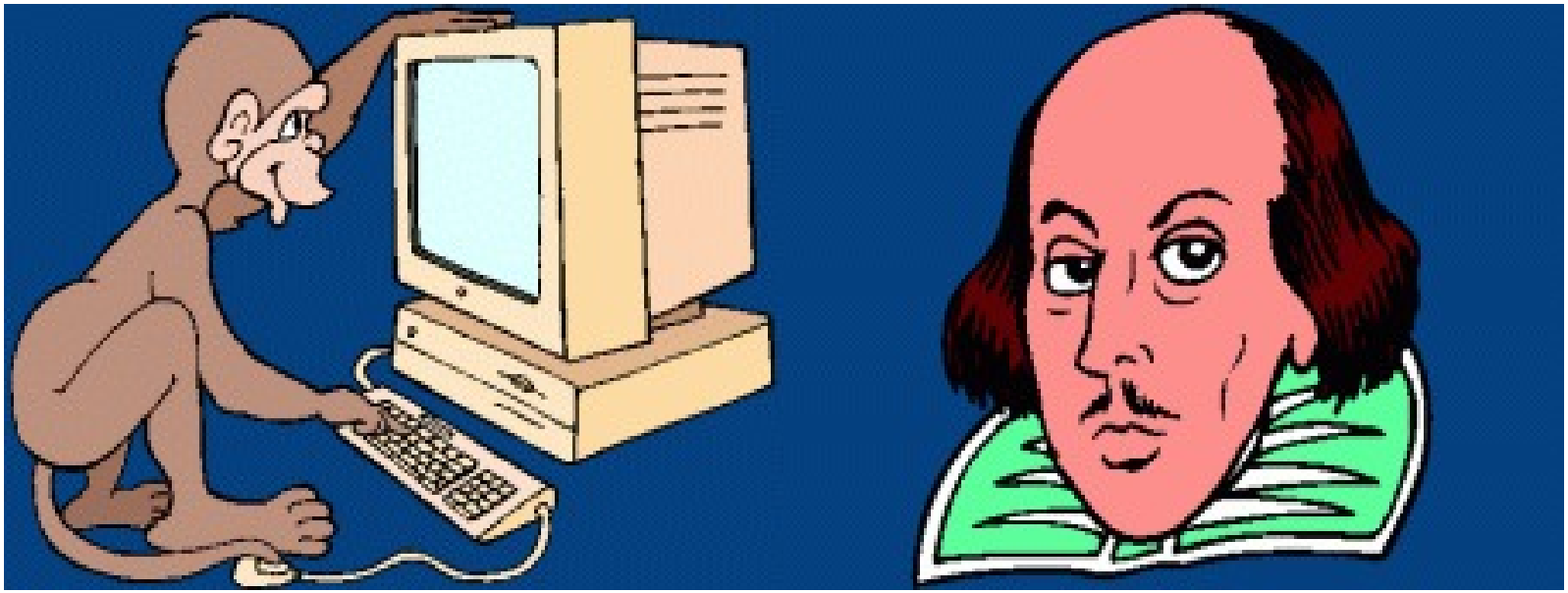
Given $u = 0.8923$, simulate a $\mathrm{Geom}(0.4)$ random variable X.

Solution

Now, for a geometric random variable with $p = 0.4$, we have

$$\begin{aligned} X &= 1 + \left\lfloor \frac{\ln(1 - 0.8923)}{\ln(1 - 0.4)} \right\rfloor \\ &= 1 + \left\lfloor \frac{-2.228406}{-0.5108256} \right\rfloor \\ &= 1 + \lfloor 4.362361 \rfloor \\ &= 1 + 4 = 5. \end{aligned}$$

The infinite monkey theorem



This theorem states:

“If you have enough monkeys banging randomly on typewriters, they will, with absolute certainty, eventually type the complete works of William Shakespeare”

The infinite monkey theorem

This theorem was first discussed in a book by Emile Borel in 1909, and can be proven, subject to the following assumptions:

- there are an infinite number of bananas to feed the monkeys;
- the monkeys type constantly, with no rest periods;
- each monkey is equally as intelligent as every other monkey.

The infinite monkey theorem

We can 'prove' the infinite monkey theorem by assuming a geometric distribution for the number of attempts until a monkey is successful.

Let's assume there's just one monkey, and let there be M keys on the monkey's typewriter. Thus,

$$\Pr(\text{monkey hits some particular key}) = \frac{1}{M}.$$

Let there be T characters in the complete works of Shakespeare. Thus,

$$\begin{aligned} \Pr(\text{complete works of shakespeare}) &= \frac{1}{M^T} \\ &= a > 0, \qquad \text{say.} \end{aligned}$$

The infinite monkey theorem

Define an *attempt* to be T consecutive independent key presses by the monkey.

$$\begin{aligned} \Pr(\text{attempt fails}) &= 1 - a, \qquad \text{and} \\ \Pr(\text{attempt succeeds}) &= a. \end{aligned}$$

Clearly, all attempts are independent (unless a monkey reads while it goes along!). Let

X : number of attempts until the first success is encountered.

The infinite monkey theorem

Define an *attempt* to be T consecutive independent key presses by the monkey.

$$\begin{aligned} \Pr(\text{attempt fails}) &= 1 - a, \qquad \text{and} \\ \Pr(\text{attempt succeeds}) &= a. \end{aligned}$$

Clearly, all attempts are independent (unless a monkey reads while it goes along!). Let

$$X : \text{number of attempts until the first success is encountered.}$$

Then

The infinite monkey theorem

Define an *attempt* to be T consecutive independent key presses by the monkey.

$$\begin{aligned} \Pr(\text{attempt fails}) &= 1-a, \qquad \text{and} \\ \Pr(\text{attempt succeeds}) &= a. \end{aligned}$$

Clearly, all attempts are independent (unless a monkey reads while it goes along!). Let

$$X : \text{number of attempts until the first success is encountered.}$$

Then

$$\Pr(X=i) = (1-a)^{i-1}a.$$

Thus, X has a geometric distribution, with probability a, i.e.

$$X \sim \text{geom}(a).$$

The infinite monkey theorem

Let A_i be the event "attempt i is the first success". Then A_1, A_2, ... are all disjoint events, so that

$$\Pr(A_1 \cup A_2 \cup A_3 \cup \ldots) \quad = \quad \Pr(A_1) + \Pr(A_2) + \Pr(A_3) + \ldots,$$

The infinite monkey theorem

Let A_i be the event "attempt i is the first success". Then $A_1, A_2, \ldots$ are all disjoint events, so that

$$\Pr(A_1 \cup A_2 \cup A_3 \cup \ldots) = \Pr(A_1) + \Pr(A_2) + \Pr(A_3) + \ldots,$$

i.e.

$$\begin{aligned} \Pr(\text{S'ful attempt at some stage}) &= \Pr(A_1) + \Pr(A_2) + \Pr(A_3) + \ldots \\ &= a + (1-a)a + (1-a)^2 a + \ldots \qquad (*) \end{aligned}$$

The infinite monkey theorem

Now (*) is the sum to infinity of a geometric progression (G.P.). Recall that the sum of the first n terms of a G.P. is given by

$$\frac{\text{first term} \times (1 - (\text{common ratio})^n)}{1 - (\text{common ratio})}.$$

Our first term is a, and the common ratio is $(1 - a)$.

The infinite monkey theorem

Thus, the sum to infinity is given by

$$\sum_{i=1}^{\infty}(1-a)^{i-1}a \quad = \quad \frac{a(1-(1-a)^{\infty})}{1-(1-a)}.$$

Now $(1-a)^k \to 0$ as $k \to 0$, and so

$$\begin{aligned} \Pr(\text{successful attempt at some stage}) &= \frac{a(1-a)}{a} \\ &= 1, \end{aligned}$$

i.e. with certainty the monkey will eventually type out the complete works of Shakespeare.

The infinite monkey theorem

Many people have tried to simulate this problem using random numbers in order to prove (or disprove!) this theorem (though the simulation methods involved are beyond the scope of this course).

The infinite monkey theorem

Many people have tried to simulate this problem using random numbers in order to prove (or disprove!) this theorem (though the simulation methods involved are beyond the scope of this course).

If you're interested, have a look at

`http://infinite-monkey-theorem.wikiverse.org/`

and follow the link for the `Monkey Shakespeare Simulator`.

The infinite monkey theorem

Many people have tried to simulate this problem using random numbers in order to prove (or disprove!) this theorem (though the simulation methods involved are beyond the scope of this course).

If you're interested, have a look at

`http://infinite-monkey-theorem.wikiverse.org/`

and follow the link for the `Monkey Shakespeare Simulator`.

The simulator was launched on 1 July 2003 (and is still running), and simulates a large population of monkeys typing randomly.

The infinite monkey theorem

Many people have tried to simulate this problem using random numbers in order to prove (or disprove!) this theorem (though the simulation methods involved are beyond the scope of this course).

If you're interested, have a look at

`http://infinite-monkey-theorem.wikiverse.org/`

and follow the link for the `Monkey Shakespeare Simulator`.

The simulator was launched on 1 July 2003 (and is still running), and simulates a large population of monkeys typing randomly.

Currently (18th March 2014), the best attempt has given the first 22 letters from Romeo and Juliet!

Part VIII

Monte Carlo Methods

Quiz: Distributions

1. An express coach is due to arrive in Newcastle from London at 23.00. However, in practice, it is equally likely to arrive anywhere between 15 minutes ear ly to 45 minutes late, depending on traffic conditions. Let the random variable X denote the amount of time (in minutes) that the coach is delayed.

2. Customers arrive at the drive-thru window of a fast food restaurant at a rate of 2 per minute during the lunch hour. Let Y be the number of customers that arrive during the lunch hour.

3. Let H: height of students taking MAS1343.

4. Let G: the number of bad eggs in a box of 12. The probability of an egg being bad is 0.1.

8.1 The continuous uniform U(0,1) distribution

$X \sim U(0,1)$ denotes the random variable X which has probability density function (PDF):

$$f_X(x) = \begin{cases} 1 & 0 < x < 1 \\ 0 & \text{otherwise.} \end{cases}$$

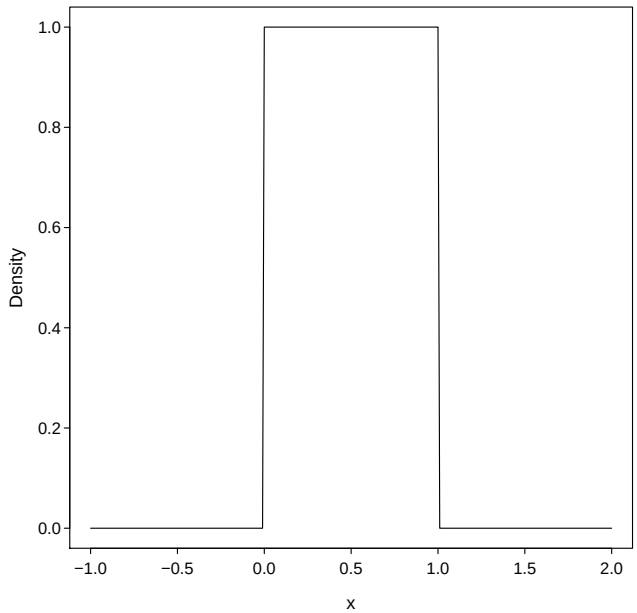


Figure: PDF of the uniform distribution.

How do we simulate $U(0, 1)$?

How do we simulate $U(0,1)$?

- Generating $U(0,1)$ random variables *precisely* would be very difficult, but we can get three decimal places by generating a random integer x from the set

$$\{0, 1, 2, \ldots, 999\},$$

with all outcomes equally likely, and then put $u = x/1000$.

- Then u is a simulation from a $U(0,1)$ random variable recorded to three decimals.
- So for a full period Congruential generator with $m = 2^{32}$ we get set of integers:

$$\{0, 1, 2, \ldots, 2^{32} - 1\} .$$

Setting $u = x/2^{32}$ would give a pseudo–random number from the $U(0,1)$ distribution, recorded to about 8 decimal places.

8.2 Monte Carlo

- The term “Monte Carlo” is used to describe any simulation study which involves random numbers.
- The name is a reference to the famous Monte Carlo Casino in Monaco, where repetition of random events is the order of the day!

8.2.1 What is a simulation study?

- For our purposes, a simulation study is any study where we study the properties of a system using random numbers.
- We have already seen simulation studies, for example the monopoly practical.

8.2.1 What is a simulation study?

- For our purposes, a simulation study is any study where we study the properties of a system using random numbers.
- We have already seen simulation studies, for example the monopoly practical.
- In general, suppose we do an experiment which has the event A as one possible outcome.
- We would like to estimate the probability of A, denoted by $\Pr[A]$.

8.2.1 What is a simulation study?

- Then by repeatedly simulating the experiment, it is simple to estimate $\Pr[A]$, the probability of the event A, using $P_F(A)$, where:

$$P_F(A) = \frac{\text{No. of times } A \text{ occurs}}{\text{Number of times experiment simulated}}.$$

- Here $P_F(A)$ is the *frequency estimate* of $\Pr[A]$.
- Why does this work?

8.2.1 What is a simulation study?

- Well suppose we denote the number of times we simulate the experiment by n, then $P_F(A)$ has the following important property:

$$\text{as } n \to \infty, \text{ then } P_F(A) \to \Pr[A].$$

- This means that the more simulations we do, the more *accurate* our estimate of $\Pr[A]$.

Monte-Carlo Integration

Monte-Carlo Integration

- Draw a simple a function for: $0 \leq x \leq 1$ and $0 \leq y \leq 1$
- Just make it a simple collection of rectangles

Monte-Carlo Integration

- Draw a simple a function for: $0 \leq x \leq 1$ and $0 \leq y \leq 1$
- Just make it a simple collection of rectangles
- Calculate the area of your function - write it down.

Monte-Carlo Integration

- Draw a simple a function for: $0 \leq x \leq 1$ and $0 \leq y \leq 1$
- Just make it a simple collection of rectangles
- Calculate the area of your function - write it down.
- Now I will generate points (x, y) using **R**. Mark these points roughly on your plot.

8.3 Approximation of integrals

- We would like to evaluate $\int_0^1 f(x)\,dx$, but the function may be too complicated to integrate.
- We can find an approximate answer by noting that the integral is equal to the area under the curve, and using Monte Carlo methods.
- We design an experiment which would work in general, even if the function was defined on a general range (a, b), and if $f(x) \in (0, c)$, for any positive value c.

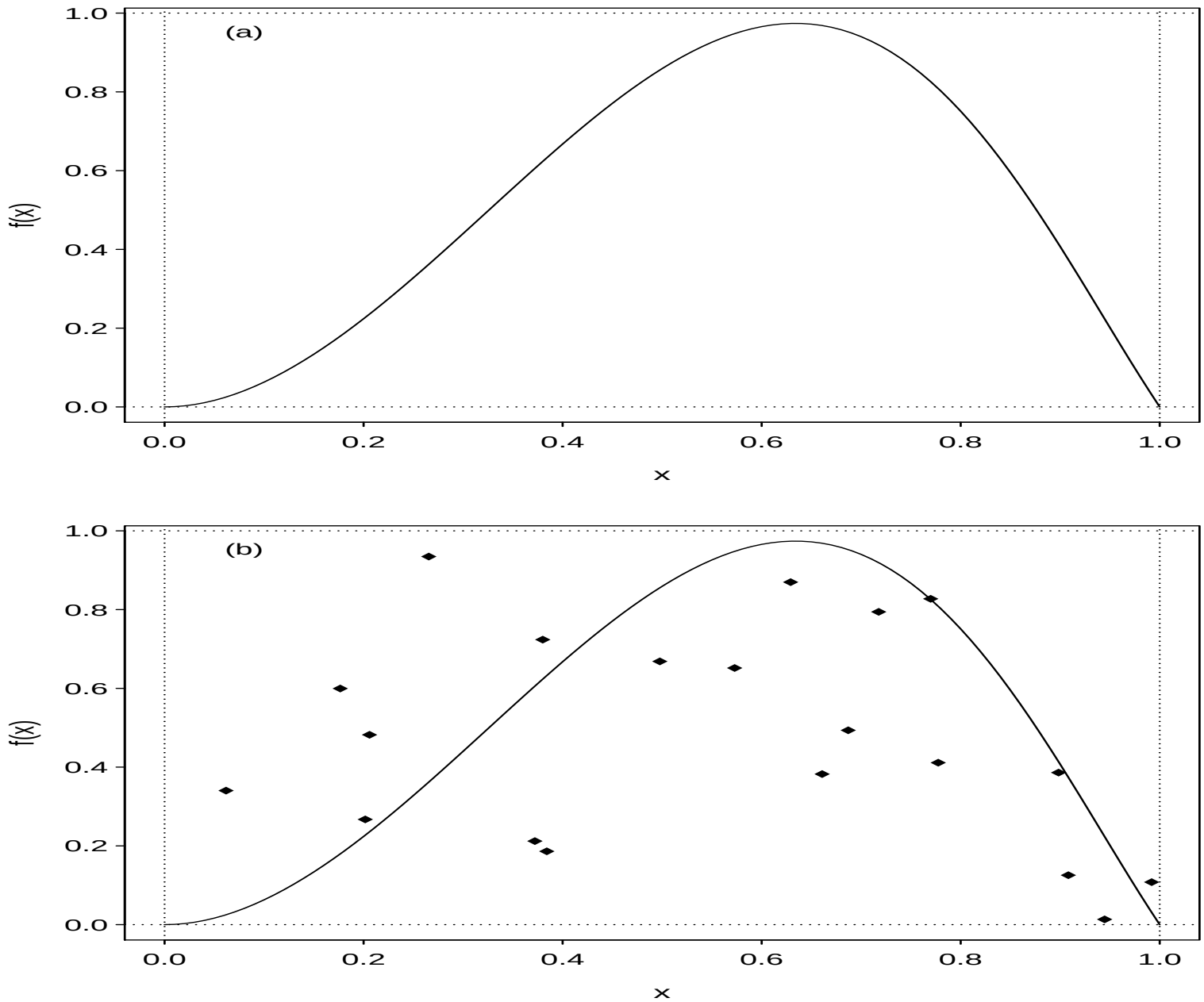


Figure: (a) An example function. (b) Twenty points randomly placed on the graph.

8.3 Approximation of integrals

We generate a random data point from a *simulation grid*. Let

$$A = \{\text{the data point lies below the curve}\}.$$

8.3 Approximation of integrals

We generate a random data point from a *simulation grid*. Let

$$A = \{\text{the data point lies below the curve}\}.$$

Then

$$\Pr[A] = \frac{\text{area under curve}}{\text{area of simulation grid}},$$

so that

8.3 Approximation of integrals

We generate a random data point from a *simulation grid*. Let

$$A = \{\text{the data point lies below the curve}\}.$$

Then

$$\Pr[A] = \frac{\text{area under curve}}{\text{area of simulation grid}},$$

so that

$$\begin{aligned}\int_a^b f(x)\,dx &= [\text{area under curve}] \\ &= \Pr[A] \times [\text{area of simulation grid}].\end{aligned}$$

8.3.1 Example

- Consider the function we saw earlier in Figure 8.2, defined on $(0, 1)$, and with $f(x) \in (0, 1)$.
- Estimate $\int_0^1 f(x)dx$ using Monte Carlo methods.

Solution: Step 1

We simulate n data points from the simulation grid;

Solution: Step 1

We simulate n data points from the simulation grid; here this is the unit square $(0, 1) \times (0, 1)$.

Solution: Step 2

Each of the coordinates x and y are generated using a $U(0, 1)$ random variable.

Solution: Step 2

Each of the coordinates x and y are generated using a $U(0,1)$ random variable. Note the fact that

$$A =$$

Solution: Step 2

Each of the coordinates x and y are generated using a $U(0,1)$ random variable. Note the fact that

$$A = \{\text{data point } (x, y) \text{ lies below the curve}\} =$$

Solution: Step 2

Each of the coordinates x and y are generated using a $U(0,1)$ random variable. Note the fact that

$$A = \{\text{data point } (x, y) \text{ lies below the curve}\} = \{y < f(x)\}.$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$,

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\int_0^1 f(x)\,dx \quad =$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\int_0^1 f(x)\,dx \quad = \quad \Pr[A]$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\int_0^1 f(x)\,dx \quad = \quad \Pr[A] \times$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\int_0^1 f(x)\,dx \quad = \quad \Pr[A] \times [\text{area of simulation grid}]$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\begin{aligned} \int_0^1 f(x)\,dx &= \Pr[A] \times [\text{area of simulation grid}] \\ &= \Pr[A] \end{aligned}$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\begin{aligned} \int_0^1 f(x)\,dx &= \Pr[A] \times [\text{area of simulation grid}] \\ &= \Pr[A] \times 1 \end{aligned}$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\begin{aligned} \int_0^1 f(x)\,dx &= \Pr[A] \times [\text{area of simulation grid}] \\ &= \Pr[A] \times 1 \\ &= \Pr[A] \end{aligned}$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\begin{aligned}
\int_0^1 f(x)\,dx &= \Pr[A] \times [\text{area of simulation grid}] \\
&= \Pr[A] \times 1 \\
&= \Pr[A] \\
&\simeq P_F(A)
\end{aligned}$$

Solution: Step 3

If r points lie below the curve, then $\Pr[A] \simeq P_F(a) = r/n$, and thus

$$\begin{aligned}
\int_0^1 f(x)\,dx &= \Pr[A] \times [\text{area of simulation grid}] \\
&= \Pr[A] \times 1 \\
&= \Pr[A] \\
&\simeq P_F(A) \\
&= r/n.
\end{aligned}$$

Solution: Step 4

In our case, from the figure above we estimate:

Solution: Step 4

In our case, from the figure above we estimate:

$$\int_0^1 f(x)\,dx \simeq \frac{r}{n} =$$

Solution: Step 4

In our case, from the figure above we estimate:

$$\int_0^1 f(x)\,dx \simeq \frac{r}{n} = \frac{12}{20}$$

Solution: Step 4

In our case, from the figure above we estimate:

$$\int_0^1 f(x)\,dx \simeq \frac{r}{n} = \frac{12}{20} = 0.6\ .$$

Using R

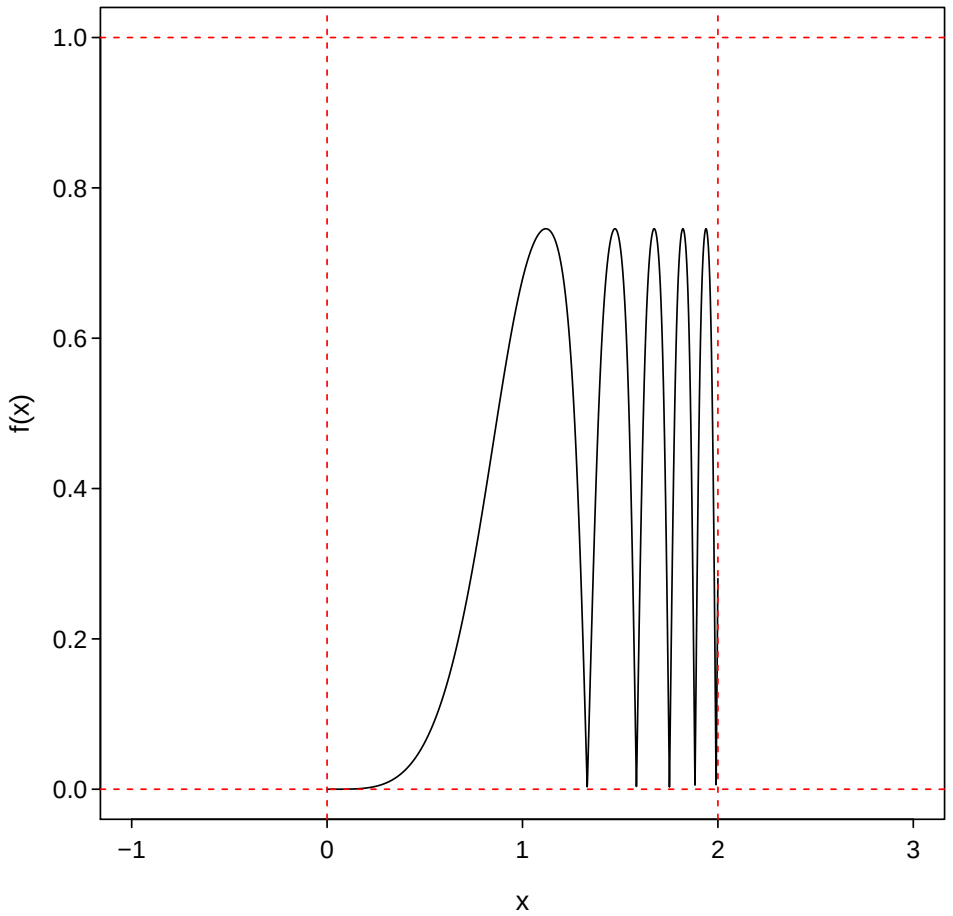


Figure: A plot of $\left|\sin\{\sin[\sin(x^4)]\}\right|$ with the sampling region.

Using R

We can also easily implement the above algorithm in R. As an extreme example, consider the function

$$f(x) = \left| \sin\{\sin[\sin(x^4)]\}\right| .$$

Using R

We can also easily implement the above algorithm in R. As an extreme example, consider the function

$$f(x) = \left| \sin\{\sin[\sin(x^4)]\} \right| .$$

We wish to calculate

$$\int_0^2 f(x)\, dx = \int_0^2 \left| \sin\{\sin[\sin(x^4)]\} \right| dx ,$$

which according to maple evaluates to approximately 0.71875. First plot the function to determine the necessary region:

Using R

The plot generated in the code above is shown in figure 8.2. Then we use a `for` loop to simulate lots of random numbers

```
> set.seed(1)
> N = 100000
> no_of_hits = 0
> for(i in 1:N) {
+   x = runif(1, 0, 2); y = runif(1)
+   if(abs(sin(sin(sin(x^4)))) > y) {
+     no_of_hits = no_of_hits + 1
+   }
+ }
> area_under_curve = no_of_hits/N*2
> area_under_curve
```