

Lecture 3: Functions

Ciaran Evans

Last time

```
n <- 1000
m = 2^32
a = 1664525
c = 1013904223

x0 <- 1

x <- rep(NA, n)
x[1] <- x0
for(i in 2:n){
  x[i] <- (a * x[i-1] + c) %% m
}

u <- x/m
```

What if I wanted to run the LCG with a different seed, or a different number of samples, or different parameters?

LCG code changing the settings

```
n <- 1000  
m = 2^32  
a = 1664525  
c = 1013904223
```

```
x0 <- 1
```

```
...
```

```
n <- 5000  
m = 1024  
a = 1  
c = 1
```

```
x0 <- 12
```

```
...
```

Functions

name of function → define a function → function inputs (arguments)

```
my_lcg <- function(n, x0, m, a, c){  
  x <- rep(NA, n)  
  x[1] <- x0  
  for(i in 2:n){  
    x[i] <- (a * x[i-1] + c) %% m  
  }  
  return(x/m) ← output  
}
```

right now,
all the
arguments
need to
be specified
when calling
the function

```
head(my_lcg(n = 1000, x0 = 1, m = 2^32,  
            a = 1664525, c = 1013904223))
```

```
## [1] 2.328306e-10 2.364555e-01 3.692707e-01 5.042420e-01  
## [6] 5.054363e-02
```

```
head(my_lcg(n = 5000, x0 = 12, m = 1024, a = 1, c = 1))
```

```
## [1] 0.01171875 0.01269531 0.01367188 0.01464844 0.015625
```

Function defaults

default values

```
my_lcg <- function(n, x0, m = 2^32,  
                  a = 1664525, c = 1013904223){  
  x <- rep(NA, n)  
  x[1] <- x0  
  for(i in 2:n){  
    x[i] <- (a * x[i-1] + c) %% m  
  }  
  return(x/m)  
}
```

(if the user
doesn't specify an
input, then we
use the default
value)

```
my_lcg(n = 5, x0 = 1, m = 2^32,  
       a = 1664525, c = 1013904223)
```

```
## [1] 2.328306e-10 2.364555e-01 3.692707e-01 5.042420e-01
```

```
my_lcg(n = 5, x0 = 1)
```

```
## [1] 2.328306e-10 2.364555e-01 3.692707e-01 5.042420e-01
```

What if we don't want to specify a seed?

- ▶ In R, functions for simulating random variables don't *require* us to specify a seed:

```
runif(5)
```

```
## [1] 0.7896950 0.1879501 0.4728500 0.1790625 0.3693306
```

- ▶ But even if we don't manually set a seed, R's random number generators still require one!

How might R choose a seed, if we don't specify one ourselves?

R gets information in part from
current time

Getting the system time

```
Sys.time()
```

```
## [1] "2025-01-11 09:24:48 EST"
```

Getting the system time

```
Sys.time() |>  
  lubridate::second()
```

```
## [1] 48.57291
```

```
(Sys.time() |> lubridate::second()) * 100000
```

```
## [1] 4857634
```

↑ use this as an initial seed

What if we don't want to specify a seed?

users don't have
to specify a
seed (but can if
they want to)

```
my_lcg <- function(n, x0 = NA, m = 2^32,  
                   a = 1664525, c = 1013904223){  
  if(is.na(x0)){  
    x0 <- ((Sys.time() |> lubridate::second()) * 100000) %% m  
  }
```

```
  x <- rep(NA, n)  
  x[1] <- x0  
  for(i in 2:n){  
    x[i] <- (a * x[i-1] + c) %% m  
  }  
  return(x/m)  
}
```

```
my_lcg(n = 5)
```

```
## [1] 0.001131162 0.083768899 0.663071837 0.884992848 0.95
```

What if we don't want to specify a seed?

```
my_lcg(n = 5)
```

```
## [1] 0.001131225 0.188753437 0.051198487 0.396858916 0.82
```

```
my_lcg(n = 5)
```

```
## [1] 0.001201529 0.211493830 0.002825943 0.089415424 0.44
```

Homework 1

<https://sta379-s25.github.io/homework/hw1.html>

- ▶ Function practice and probability review
- ▶ Accept and submit coding portion of assignment on GitHub Classroom
- ▶ Collaboration encouraged on homework, but everyone must submit their own work and acknowledge collaborators