Informe de Simulación Estocástica
Análisis del proceso de producción de chupetines

Autor: Bryan Cutipa Carcasi
Docente: Fred Torres Cruz

Introducción

En este informe se presenta un modelo estocástico que simula un proceso de producción de chupetines a partir de tres tipos de dulces (denotados x, y, z). El objetivo es determinar cuántas interacciones (acciones del sistema) son necesarias, en promedio, para producir 10 chupetines, partiendo de una distribución aleatoria de 30 dulces. El análisis combina razonamiento probabilístico con simulación Monte Carlo, una técnica fundamental en **Estadística Computacional**.

Modelo matemático

El sistema evoluciona según dos reglas deterministas con componentes aleatorias:

1. **Producción**: Si $x \geq 1$, $y \geq 1$ y $z \geq 1$, se consume una unidad de cada tipo y se produce un chupetín ($c \leftarrow c + 1$).
2. **Reciclaje**: Si no se puede producir pero $c \geq 1$ y al menos un tipo de dulce está disponible, se consume un chupetín y un dulce al azar, y se generan 4 nuevos dulces distribuidos uniformemente entre x, y, z.

El estado del sistema en cualquier momento es el vector $(x, y, z, c) \in \mathbb{N}^4$. El proceso termina cuando $c = 10$ o cuando se alcanza un número máximo de iteraciones sin lograr el objetivo (lo que se considera un *fracaso*).

Este es un proceso de Markov con espacio de estados finito (aunque grande), y la probabilidad de éxito depende de la dinámica estocástica de reciclaje.

Simulación computacional

Se realizaron 10 000 réplicas independientes del proceso, usando `set.seed(123)` para reproducibilidad. Cada simulación comienza con una partición aleatoria de 30 dulces generada mediante una distribución multinomial:

$$(x, y, z) \sim \text{Multinomial}(n = 30, p = (1/3, 1/3, 1/3)).$$

El código R implementa las reglas descritas y registra el número de interacciones necesarias para alcanzar 10 chupetines. Las simulaciones que no logran el objetivo en 1000 pasos se descartan como fracasos.

Resultados

De las 10 000 simulaciones:

Simulaciones exitosas: `r length(exitosas)`.

Tasa de éxito: `r round(length(exitosas)/10000*100, 2)` %.

Para los casos exitosos, se obtuvieron las siguientes estadísticas descriptivas del número de interacciones:

Estadístico	Valor
Mínimo	`r min(exitosas)`
Mediana	`r median(exitosas)`
Promedio	`r round(mean(exitosas), 2)`
Máximo	`r max(exitosas)`

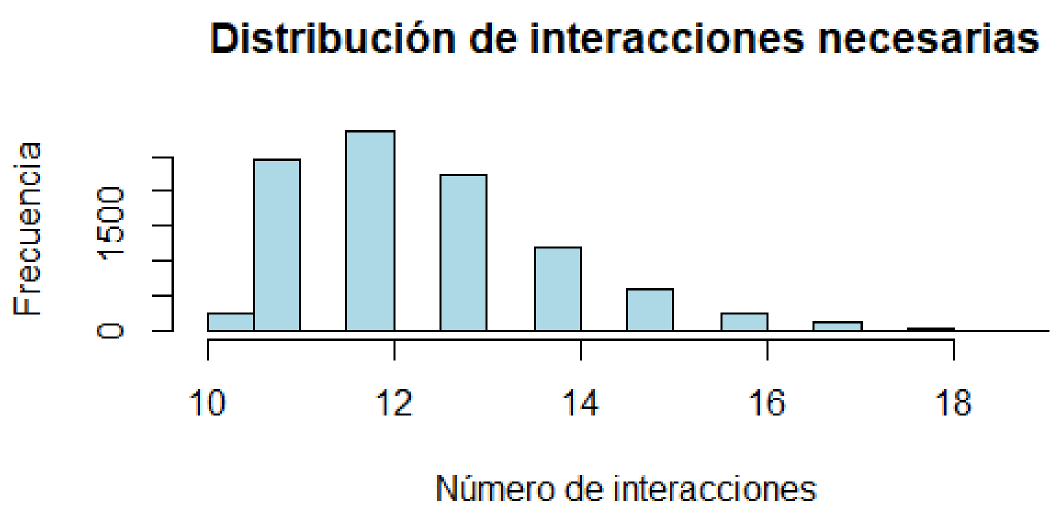

Figura 1: Distribución del número de interacciones necesarias para producir 10 chupetines (solo casos exitosos).

Interpretación: La distribución es asimétrica a la derecha, lo que indica que, aunque la mayoría de los procesos convergen en menos de 30 pasos, algunos requieren muchos más debido a la naturaleza aleatoria del reciclaje. La alta tasa de éxito sugiere que el mecanismo de reciclaje es eficaz para recuperar estados productivos.

Conclusión

Este ejercicio ilustra cómo los modelos estocásticos pueden representar sistemas con reglas simples pero comportamiento emergente complejo. La simulación Monte Carlo permite aproximar propiedades del sistema (como la tasa de éxito o el número esperado de pasos) que serían difíciles de derivar analíticamente. Además, refuerza conceptos clave de la estadística computacional: reproducibilidad, muestreo aleatorio, y análisis de resultados mediante resúmenes numéricos y gráficos.

Código R

```r
# Cargar librer as necesarias
if (!require("dplyr")) install.packages("dplyr")
library(dplyr)

simular_una_vez <- function(total_dulces = 30, objetivo_chupetines = 10, max_iter = 1000) {
  dulces_iniciales <- rmultinom(1, size = total_dulces, prob = c(1/3, 1/3, 1/3))
  x <- dulces_iniciales[1]; y <- dulces_iniciales[2]; z <- dulces_iniciales[3]
  c <- 0; chupetines_producidos <- 0; interacciones <- 0

  while (chupetines_producidos < objetivo_chupetines && interacciones < max_iter) {
    if (x > 0 && y > 0 && z > 0) {
      x <- x - 1; y <- y - 1; z <- z - 1
      c <- c + 1; chupetines_producidos <- chupetines_producidos + 1
      interacciones <- interacciones + 1
    } else if (c > 0 && (x > 0 || y > 0 || z > 0)) {
      disponibles <- c()
      if (x > 0) disponibles <- c(disponibles, "x")
      if (y > 0) disponibles <- c(disponibles, "y")
      if (z > 0) disponibles <- c(disponibles, "z")
      dulce_usado <- sample(disponibles, 1)
      if (dulce_usado == "x") x <- x - 1
      if (dulce_usado == "y") y <- y - 1
      if (dulce_usado == "z") z <- z - 1
      c <- c - 1
      nuevos_dulces <- sample(c("x", "y", "z"), size = 4, replace = TRUE)
      x <- x + sum(nuevos_dulces == "x")
      y <- y + sum(nuevos_dulces == "y")
      z <- z + sum(nuevos_dulces == "z")
      interacciones <- interacciones + 1
    } else {
```

```
      return(NA)
    }
  }

  if (chupetines_producidos >= objetivo_chupetines) return(
   interacciones) else return(NA)
}

set.seed(123)
n_sim <- 10000
resultados <- replicate(n_sim, simular_una_vez())
exitosas <- resultados[!is.na(resultados)]

cat("Simulaciones exitosas:", length(exitosas), "de", n_sim, "\n")
cat("Tasa de  xito :", round(length(exitosas) / n_sim * 100, 2), "%\
   n\n")

if (length(exitosas) > 0) {
  cat("Estad sticas de interacciones (solo casos exitosos):\n")
  cat("M nimo:", min(exitosas), "\n")
  cat("Mediana:", median(exitosas), "\n")
  cat("Promedio:", round(mean(exitosas), 2), "\n")
  cat("M ximo:", max(exitosas), "\n")
  png("histograma_interacciones.png", width = 800, height = 500)
  hist(exitosas, breaks = 20, col = "lightblue",
       main = "Distribuci n de interacciones necesarias",
       xlab = "N mero de interacciones", ylab = "Frecuencia")
  dev.off()
}
```

Listing 1: Código de simulación