

## Práctica Nº 2

### Autómata Celular

**Nombre:** José Adrián García Fuentes  
**Fecha:** 23/Febrero/2021

**Profesor:** Satu Elisa Schaeffer

---

## 1. Objetivo

- Diseñar y ejecutar un experimento para determinar el efecto de la regla de supervivencia (por lo menos cinco reglas) en la vida de la colonia en una malla de 12 por 12 celdas hasta que se mueran todas o que se hayan cumplido 30 iteraciones, teniendo cada celda o viva o muerta con la misma probabilidad al inicio [1].
- Graficar y tabular los hallazgos [1].

## 2. Metodología

La metodología empleada se realizó a través de RStudio [2] llevando a cabo los pasos señalados en la *Práctica 2 Autómata Celular* [1], la secuencia realizada de forma paralela fue basada en el código en R [3] encontrada en el repositorio de GitHub [4] para diseñar y ejecutar el experimento se estableció una colonia a la que llamaremos nido dentro del código con dimensiones de 12 por 12 celdas, se generó un punto de origen con una rutina en donde intervienen los números enteros más cercanos quienes determinara si nuestro origen está vivo o muerto en nuestro caso se acopló esta regla imaginando a un ave fénix por tanto se determinara si está hecho cenizas o se encuentra vivo, sin embargo, para la supervivencia se efectuaron distintas reglas que dependerán del número de vecinos vivos, el código completo de la metodología empleada se encuentra en el repositorio [5] del autor.

## 3. Reglas de supervivencia

Las 5 reglas de supervivencia aplicadas son las siguientes:

- Nuestro punto se encontrara vivo siempre y cuando 6 de sus vecinos se encuentre vivo.
- Nuestro punto se encontrara vivo siempre y cuando 3 de sus vecinos se encuentre vivo.
- Nuestro punto se encontrara vivo siempre y cuando 1 de sus vecinos se encuentre vivo.
- Nuestro punto se encontrara vivo siempre y cuando 5 de sus vecinos se encuentre vivo.
- Nuestro punto se encontrara vivo siempre y cuando 2 de sus vecinos se encuentre vivo.

- Nuestro punto se encontrara vivo siempre y cuando 4 de sus vecinos se encuentre vivo.

Sin embargo para cada regla se deberá contar con un par de condiciones a cumplir de manera que el porcentaje de supervivencia cambiara.

- El número de vecinos tendrá que ser dado por una condición aleatoria.
- Si el número de sus vecinos es mayor a 5 cambiara la regla de supervivencia en una cantidad de vecinos entre 2 y 5
- Si el número de sus vecinos es menor a 5 cambiara la regla de supervivencia en una cantidad de vecinos entre 1 y 7

Por tanto las cantidades de números de vecinos serán de forma aleatoria y pueden reducirse o incrementar, para realizar una única regla de supervivencia podríamos revisar el código de GitHub [6] en donde se subió el código por separado con números que dicten un solo valor de supervivencia.

## 4. Resultados

A continuación se muestra parte del código realizado en RStudio [2] para ver el código completo consultar la base de datos de GitHub [3].

```

1 library(parallel)
2
3 terreno <- 12
4 hectareascuadradas <- terreno^2
5 expansion<- terreno + 2
6 nido <- matrix(round(runif(hectareascuadradas)), nrow=terreno, ncol
  =terreno, byrow=TRUE)
7 suppressMessages(library("sna"))
8 png("practica2_t0_r.png")
9 plot.sociomatrix(nido, diaglab=FALSE, main="Inicio")
10 graphics.off()
11
12 Ano <- function(posicionvector) {
13   fila <- floor((posicionvector - 1) / terreno) + 1
14   columna <- ((posicionvector - 1) %% terreno) + 1
15   vecinos <- nido[max(fila - 1, 1) : min(fila + 1, terreno),
16                 max(columna - 1, 1) : min(columna + 1, terreno)
17   ]
18   return(1 * ((sum(vecinos) - nido[fila, columna]) == sample(1:6,1))
19   )
20   if (((1 * ((sum(vecinos) - nido[fila, columna]) == sample(1:6,1)))
21   >5) {(1 * ((sum(vecinos) - nido[fila, columna]) == sample
22   (1:6,1)))==(2:5)}
23   else {(1 * ((sum(vecinos) - nido[fila, columna]) == sample(1:6,1))
24   )==(1 * ((sum(vecinos) - nido[fila, columna]) == sample(1:7,1))
25   )+1)}
26 }
27 print(nido)

```

Los resultados obtenidos tras correr el código en RStudio [2] fueron convertidos a archivos .png, cada uno de estos representa la cantidad de vivos que se encuentran en cada interacción, las imágenes obtenidas fueron convertidas en un

archivo .gif [7] el cual se encuentra en el repositorio de GitHub [5], en la Fig. 1 se muestra el inicio de la secuencia experimental así como el paso final dado durante el experimento.

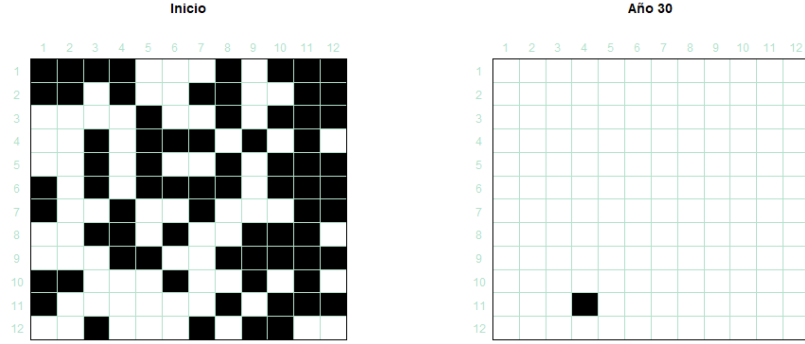


Figura 1:  
Secuencia de pasos de autómata celular

De acuerdo a la interacción de pasos dados al cual llamaremos años se realizó un grafico que representa el numero de celdas vivas dependientes del paso de los años ver Fig. 2.

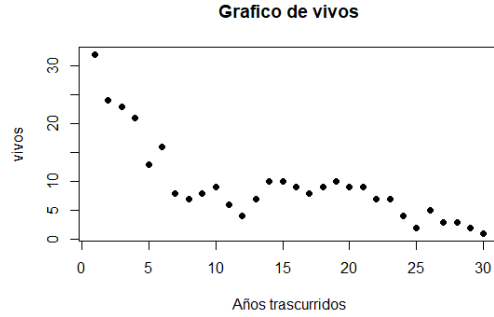


Figura 2:  
Grafico representativo de vivos en el transcurso de los 30 pasos

## 5. Conclusión

Es de suma importancia comprender el diseño y ejecución de un experimento poniendo grados de complejidad, en este caso reglas de supervivencia con la finalidad de modelar un experimento de manera in situ antes de pasar al modo in vitro.

## Referencias

- [1] E. Schaeffer, “Práctica 2: autómatas celulares,” Febrero 2021. <https://elisa.dyndns-web.com/teaching/comp/par/p2.html>.
- [2] J. J. Allaire, “Rstudio,” Febrero 2021. <https://rstudio.com>.
- [3] E. Schaeffer, “gameoflife.r,” Febrero 2021. <https://github.com/fuentesadrian/Simulation/blob/master/CellularAutomata/gameOfLife.R>.
- [4] E. Schaeffer, “Cellular automata,” Febrero 2021. <https://github.com/fuentesadrian/Simulation/tree/master/CellularAutomata>.
- [5] J. A. Garcia Febrero 2021. <https://github.com/fuentesadrian/SIMULACION-DE-NANOMATERIALES/tree/main/Tarea%202>.
- [6] J. A. Garcia Febrero 2021. <https://github.com/fuentesadrian/SIMULACION-DE-NANOMATERIALES/blob/main/Tarea%202/codigoseparado.r>.
- [7] J. A. Garcia Febrero 2021. <https://github.com/fuentesadrian/SIMULACION-DE-NANOMATERIALES/blob/main/Tarea%202/automatacellular.gif>.