

Reporte práctica 2

Autómata celular

1. INTRODUCCIÓN

El objetivo principal de la práctica es analizar una simulación del juego de la vida (autómatas celulares) en donde se representan “células” en una matriz booleana con el 0 como una célula muerta y el 1 una célula viva. Lo que determina si una célula está viva o muerta es la condición de que exactamente tres de sus vecinos estén vivos.

Se analiza el número de pasos o iteraciones hasta antes de que todas las células mueran. Lo anterior en función de la probabilidad de células vivas al inicio de la simulación.

2. SIMULACIÓN

Para la simulación se determinan las células vivas al azar mediante la función *sample* en donde podemos variar la probabilidad de los sujetos vivos al inicio de la simulación.

Primero se determinan las variables “*dim*”, que se refiere al número de filas y columnas de la matriz booleana “*num*” que determina la cantidad de valores dentro de la matriz.

Posteriormente se crea una matriz “*datos*” en donde serán guardadas las iteraciones en las que rompe el ciclo para cada probabilidad y repetición.

Para determinar qué células vivirán y/o morirán se utiliza la función “*paso*” de “*pos*”, la cual recorre la matriz y en cada celda verifica cuántos de sus vecinos están vivos, siendo tres la célula vive; más o menos de tres, muere.

La variación en la probabilidad se determinó del 0.1 al 0.9, con una función *for* que fuera de 1 a 9 y la variable interna “*prb*” se divide entre 10 para convertirlo a probabilidad. Se utilizó un *for* para repetir lo anterior el número de veces que se elija mediante la variable “*corridas*” para cada probabilidad

Se utiliza un *for* para romper el ciclo cuando todas las células mueran que suma los números de cada casilla y compara si es igual a 0, siendo así “*todos están muertos*” y el ciclo rompe.

Se exportan los valores de la matriz “*datos*”, creada al inicio del programa, en una gráfica caja-bigote para analizar la distribución de las iteraciones en función de las probabilidades.

2.1 CÓDIGO

```

1 library(parallel)
2 dim <- 10
3 num <- dim^2
4 corridas=8
5 cluster <- makeCluster(detectCores() - 1)
6 clusterExport(cluster, "dim")
7 clusterExport(cluster, "paso")
8 datos<-matrix(rep(0),nrow=corridas, ncol=9)
9 paso <- function(pos) {
10   fila <- floor((pos - 1) / dim) + 1
11   columna <- ((pos - 1) %% dim) + 1
12   vecindad <- actual[max(fila - 1, 1) : min(fila + 1, dim),
13                     max(columna - 1, 1) : min(columna + 1, dim)]
14   return(1 * ((sum(vecindad) - actual[fila, columna]) == 3))
15 }
16 for(expe in 1:corridas){
17   for(prb in 1:9){
18     p=prb/10
19     actual <- matrix(sample(c(0,1),num,prob=c(1-p,p), replace=TRUE), nrow=dim, ncol=dim)
20     suppressMessages(library("sna"))
21     for (iteracion in 1:9){
22       clusterExport(cluster, "actual")
23       siguiente <- parSapply(cluster, 1:num, paso)
24       if (sum(siguiente) == 0) { # todos murieron
25         print("Ya no queda nadie vivo.")
26         break;
27       }
28       actual <- matrix(siguiente, nrow=dim, ncol=dim, byrow=TRUE)
29       datos[expe,prb]=iteracion
30     }
31   }
32 }
33 png("t2.png", width=1200, height=900)
34 stopCluster(cluster)
35 boxplot(datos, xlab="probabilidad", ylab="corridas",main="Análisis" )
36 graphics.off()

```

3. RESULTADOS

En la gráfica a continuación se muestra la distribución de las corridas antes de romper el ciclo en función de la probabilidad. Se puede observar que la distribución da una campana de Gauss lo que nos indica que los datos tienen una distribución normal, en dónde denota que la probabilidad que alcanzó mayor número de pasos o iteraciones fue la de 0.4 vivas al inicio. Siendo también esta probabilidad la que alcanza una mayor variabilidad en el número de iteración en la que rompe, siendo de 3 hasta 9.

Los valores más pequeños de iteraciones se obtuvieron en las probabilidades extremas, 0.1 y 0.9.

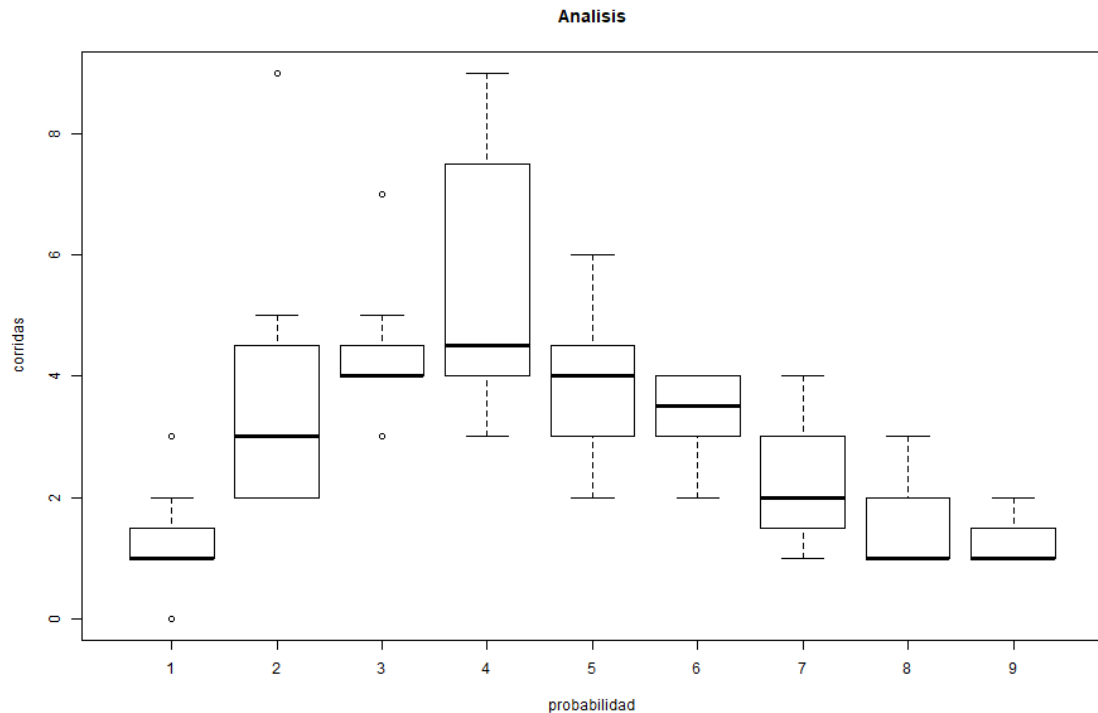


Figura 1. Gráfica caja-bigote de las iteraciones obtenidas en 9 corridas para las probabilidades de 0.1 a 0.9 de células vivas al inicio.

4. CONCLUSIONES

Los valores de iteración antes de que todas las células mueran en función de la variación de la probabilidad se distribuyen de manera normal. Por tanto, los valores extremos tomarán el menor número de iteraciones mientras que los valores medios del rango de probabilidad serán los más óptimos si se busca llegar a un mayor número de iteraciones.

De acuerdo con los resultados obtenidos el 0.4 de probabilidad es en la que durarán más tiempo las células vivas.