

Práctica 6: Sistema multiagente

Simulación de sistemas

Marco Antonio Guajardo Vigil 2095

5 de marzo, 2019

1. Introducción

Un sistema multiagente es parecido a un autómata celular, hay un conjunto de entidades con estados internos que pueden observar estados de otras entidades y reaccionar cambiando su propio estado. La diferencia es que un sistema multiagente es un concepto más general y permite que estos agentes se muevan y varíen su vecindad, entre otras cosas [3].

Para esta práctica se implementa un sistema multiagente con una aplicación en epidemiología. Los agentes podrán estar en uno de estos tres estados: **Susceptibles**, **Infectados** o **Recuperados**, esto se conoce como el modelo *SIR*. Los parámetros son el número de agentes n y la probabilidad de infección al inicio p_i . La infección produce inmunidad en los recuperados, por lo cual solamente los susceptibles podrán ser infectados. La probabilidad de contagio es proporcional a la distancia euclidiana entre dos agentes $d(i, j)$ de la siguiente manera:

$$p_c = \begin{cases} 0, & \text{si } d(i, j) \geq r, \\ \frac{r - d}{r}, & \text{en otro caso,} \end{cases}$$

donde r es un umbral.

Nuestros agentes tienen coordenadas x y y , una dirección y una velocidad (expresadas en términos de Δx y Δy). Se posicionan los agentes uniformemente al azar en un torus formado por doblar un rectángulo de $l \times l$ en dos dimensiones, se observa un ejemplo de un sistema multiagente en la figura 1.

2. Objetivo

Se vacuna con probabilidad p_v a los agentes al momento de crearlos de tal forma que estos inicien en estado R y no pueden contagiarse ni propagar la infección. Se estudia el efecto estadístico del valor p_v en el porcentaje máximo de infectados durante la simulación.

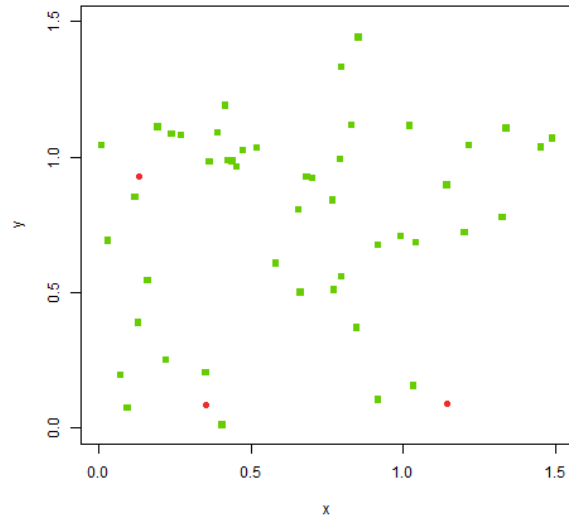


Figura 1: Ejemplo de un sistema multiagente de cincuenta agentes, tres infectados y cuarenta y siete susceptibles.

2.1. Implementación de R

Para la elaboración de este experimento, se hace uso de un software libre para computación estadística y gráficos llamado R [1], el cual nos permite realizar los cálculos necesarios para dicho experimento. Con él, se pueden controlar los datos estadísticos que se ocupan para dar seguimiento con la práctica, se necesita graficarlos para así poder compararlos mejor, ya que se maneja una cantidad de datos considerable y trabajaremos con ellos en forma estadística.

2.2. Experimentación

La simulación contiene cincuenta agentes representado en la variable `n`, se crea un `data.frame()` llamado `Rvacunas` (en el cual se almacenan los valores de las probabilidades, réplicas, máximo de infectados y porcentaje de esos máximos). La probabilidad de vacunación se varía en secuencia desde el cero al uno, en pasos de 0.1 y se hacen cincuenta réplicas por cada probabilidad como se muestra en el código:

```

1 l <- 1.5
2 n <- 50 #numero de agentes
3 pi <- 0.05 #probabilidad de infeccion al inicio
4 pr <- 0.02 #probabilidad de recuperacion
5 v <- 1 / 30 #velocidad del agente
6 PV <- seq(0,1,0.1) #probabilidad de vacunacion al inicio
7 r <- 0.1 #umbral
8 replicas <- 50
9 Rvacunas <- data.frame()
```

Para lograr la variación se realizan dos `for()`, uno para las probabilidades de vacunación y otro para las réplicas, se agrega un `if()` para vacunar a los agentes de acuerdo a la probabilidad que tengan y un `while()` que para la simulación una vez alcance la mayor cantidad de infectados posible, para eso se toma en cuenta la cantidad de infectados actuales la cual se almacena en la variable `actual` y se compara con lo que este almacenado en la variable `mayor` que inicialmente esta en cero, si la variable mayor es menor a actual, mayor toma el valor de actual, así hasta que actual llegue a ser menor que mayor, ya que en ese caso, alcanzo el mayor número de infectados posibles y se sale del `while()` con un `break`, como se muestra en el código:

```

1 epidemia <- integer()
2 mayor <- 0
3 actual <- 0
4 for(pv in PV){
5   for(rep in 1:replicas){
6     if(runif(1) < pv){ #vacunados al inicio con probabilidad pv
7       e <- "R"
8     } else if(runif(1) < pi){
9       e <- "I"
10    } else {
11      e <- "S"
12    }
13
14    while(TRUE){
15      ...
16      actual <- infectados
17      if(mayor < actual){
18        mayor <- actual
19      } else if(actual < mayor){
20        break
21      }
22    }
23    maximo_infectados <- max(epidemia)
24    porcentaje <- 100 * maximo_infectados / n
25    Rvacunas <- rbind(Rvacunas, c(pv, rep, maximo_infectados, porcentaje))
26  }
27 }

```

Se crea una gráfica con ayuda de la librería `ggplot2` [4] que muestra el porcentaje de infectados máximos que alcanzo la simulación con tal probabilidad de ser vacunados, esto es para hacer observaciones acerca de lo que llega a influir la vacuna en el sistema:

```

1 tema <- theme(
2   panel.background = element_rect(fill = "lightblue", colour = "lightblue", size = 0.5,
3     linetype = "solid"),
4   panel.grid.major = element_line(size = 0.5, linetype = 'solid', colour = "white"),
5   panel.grid.minor = element_line(size = 0.25, linetype = 'solid', colour = "white")
6 )
7 p <- ggplot(Rvacunas, aes(x = Probabilidad, y = Porcentaje, fill = Probabilidad)) + geom_violin()
8 p <- p + geom_violin(scale = "width", alpha = 0.6) + geom_violin(trim = F) + geom_boxplot(width=0.3,
9   alpha=0.8)
10 p <- p + labs(x="Probabilidades de vacunaci\u{F3}n", y = "Porcentaje m\u{E1}ximo de infectados") + tema
11 ggsave("violin.png")

```

2.3. Resultados y conclusiones

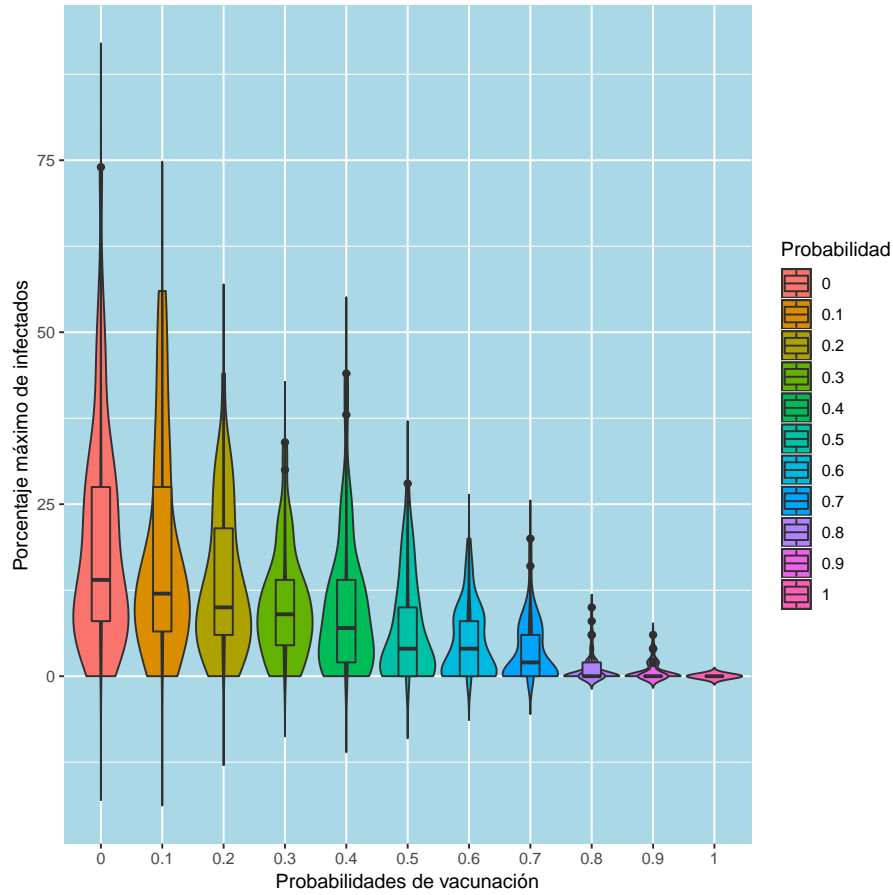


Figura 2: Porcentaje máximo de agentes infectados con relación a las probabilidades de ser vacunado.

En la gráfica de la figura 2 se muestra que las probabilidades que más afectan en la epidemia son $p_v > 0,2$ aproximadamente, estas probabilidades empiezan a bajar el máximo de infectados considerablemente, por lo tanto, sí afecta la probabilidad en el crecimiento de agentes infectados. Esto se comprueba con un test probabilístico de *Kruskal-Wallis*, el cual fue realizado con ayuda del paquete *nortest*, con el cual obtenemos un valor p que nos garantiza si afectan esas probabilidades.

```
1 kruskal.test(Rvacunas$Porcentaje ~ Rvacunas$Probabilidad)
```

Por lo general, un nivel de significación (indicado como α o alfa) de 0,05 funciona bien. Este nivel indica que no hay una diferencia real ó afectación considerable. El valor de $p \leq \alpha$, muestra que las diferencias entre algunas de las medianas son estadísticamente significativas [2].

Si el valor de p es menor o igual que el nivel de significación, podemos concluir que las probabilidades si afectaron en los infectados de la epidemia. Con la prueba se obtuvo $p - value < 2,2e - 16$, esto es menor al nivel de significación, por

lo tanto, las probabilidades si afectan considerablemente en la epidemia, el número de infectados baja exponencialmente conforme la probabilidad de vacunacion es más alta.

3. Reto 1

El objetivo de este primer reto es cambiar los patrones de movimiento de modo que no tengan una trayectoria fija utilizando el modelo de *punto medio aleatorio* (En inglés: random waypoint model). Cada agente tiene una posición meta (x, y) hacia la cual se mueve con cierta velocidad v ; al alcanzar o superar su meta, se elige una nueva meta uniformemente al azar.

La velocidad de cada agente es una constante, normalmente distribuido sobre la población de agentes. Se examina si surgen cambios en el efecto de p_v con esta modificación.

3.1. Experimentación

Se modifica el código de manera que los agentes cambien sus patrones de movimiento, estableciendo nuevas metas para (x, y) , se asigna una probabilidad de cambio de meta p_c para que el movimiento del agente cambie cada cierto tiempo, esto se realiza sumando a la posición siguiente (meta) un número aleatorio dado en rangos de velocidad: $-v < m < v$, en donde m puede tomar valores establecidos entre ese rango:

```
1 pc <- 0.1
2 if(retol){
3   if(runif(1) < pc){ #probabilidad de nueva meta
4     a$x <- a$x + runif(1,-v,v)
5     a$y <- a$y + runif(1,-v,v)
6   } else{
7     a$x <- a$x + a$dx
8     a$y <- a$y + a$dy
9   }
10 } else{
11   a$x <- a$x + a$dx
12   a$y <- a$y + a$dy
13 }
```

Ya que se obtienen variaciones en los movimientos de los agentes, se procede a realizar de nuevo el experimento anterior, vacunando inicialmente a los agentes con probabilidad p_v , siendo esta probabilidad la secuencia desde el cero al uno, en pasos de 0.1. Se realizan cincuenta réplicas para mejores estimaciones.

3.2. Resultados y conclusiones

Se realizó la prueba probabilística de *Kruskal-Wallis* para comparar si se presentan cambios en los efectos de las probabilidades. El valor de significación fue el mismo a como cuando cada uno de los agentes tenían el mismo patrón de movimiento durante la duración de la epidemia, lo cual significa que las fijaciones de nuevas metas no afectaron al valor significativo de las probabilidades de vacunación.

Con respecto a la gráfica de la figura 3 se observa que el cambio de patrones afecta a la cantidad máxima de infectados siendo de un 55 %, anteriormente de un 75 %, por lo tanto, tener nuevos patrones de movimiento hacen que la epidemia sea más difícil de propagarse.

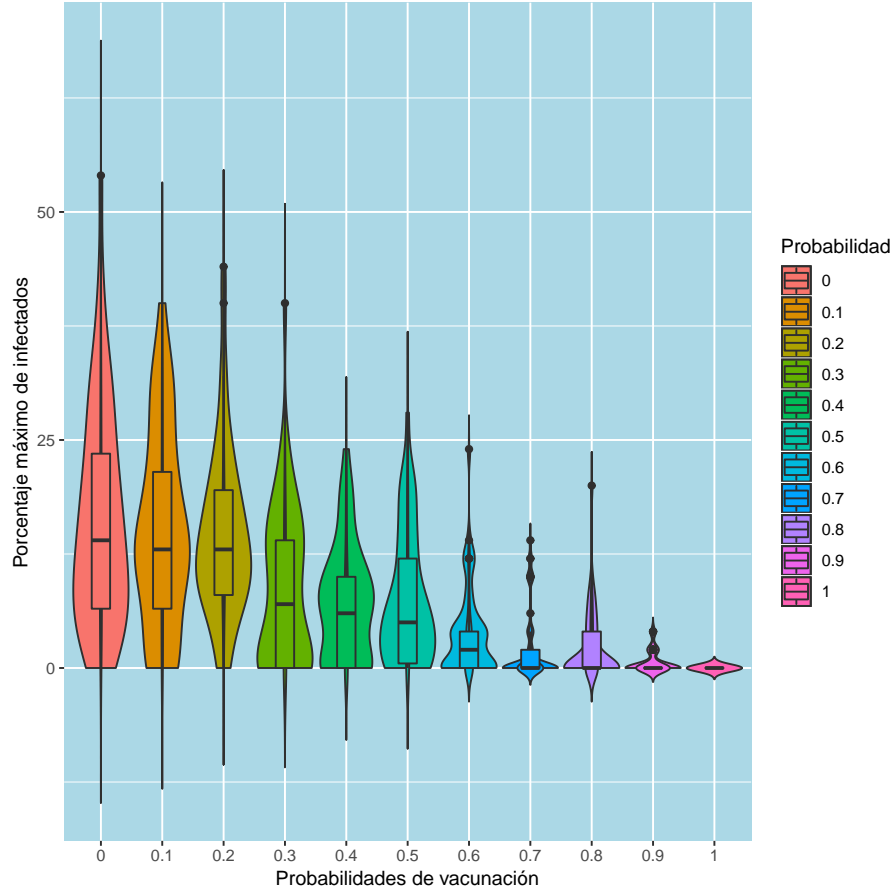


Figura 3: Porcentaje máximo de agentes infectados con relación a las probabilidades de ser vacunado respecto al efecto punto medio aleatorio.

4. Reto 2

El objetivo de este segundo reto es asignar amistades a los agentes, de manera que si se encuentran en una distancia euclídeana no mayor a r_a de un amigo suyo, disminuyen su velocidad a la mitad por k_a iteraciones (para saludar a su amigo).

Se examina nuevamente si surgieron cambios en el efecto de p_v por esta modificación, se eligen los valores con las siguientes condiciones: $0 < r_a < 1$, $k_a > 1$ y $0 < p_a < 1$.

4.1. Experimentación

La asignación de amistades se logra con la probabilidad $p_a = 0,6$, esto se realiza al comienzo junto a la asignación de estados de los agentes. Para que un agente detecte que un agente amigo esta cerca de él con una distancia euclidea menor a la de su amigo $r_a = 0,8$, se realiza un `if()` que permita evaluar esa condición y también se debe de cumplir que exista una amistad entre esos dos agentes, cuando esto suceda, se reducira la velocidad de ambos a la mitad hasta que termine el número de iteraciones o pasos k_a . Todo esto se ve representado en el siguiente código:

```
1 ka <- 6
2 ra <- 0.8 #distancia de amigo
3 pa <- 0.6 #probabilidad de amistad
4
5 agentes <- data.frame(x = double(), y = double(), dx = double(), dy = double(), estado = character(),
6   amigo = NULL)
7 #asignar amistad
8 if(runif(1) < pa){
9   amistad <- TRUE
10 } else{
11   amistad <- FALSE
12 }
13
14 agentes <- rbind(agentes, data.frame(x = runif(1, 0, 1), y = runif(1, 0, 1),
15   dx = runif(1, -v, v), dy = runif(1, -v, v),
16   estado = e, amigo = amistad))
17
18 for(pv in PV){
19   for(rep in 1:replicas){
20     ...
21     pasos <- ka
22     #bajar la velocidad a la mitad cuando se encuentre a un amigo
23     if(d < ra && a1$amigo == a2$amigo){
24       if(pasos == ka){
25         dx <- dx / 2
26         dy <- dy / 2
27         pasos <- pasos - 1
28         #print("baja")
29       } else if(pasos > 0){
30         pasos <- pasos - 1
31       } else{ #regresa a su velocidad normal
32         dx <- dx * 2
33         dy <- dy * 2
34         #print("normaliza")
35         pasos <- ka
36       }
37     }
38   }
39 }
```

Se procede a realizar la simulación variando la probabilidad de las vacunas al comienzo y se grafican los datos obtenidos para llegar a una conclusión y definir si las amistades entre agentes llegan a tener algún afecto significativo ó no en la proporción de los máximos infectados con relación a la probabilidad de vacunación.

4.2. Resultados y conclusiones

Estos resultados, visibles en la figura 4, muestran un porcentaje de infectados mayor a los anteriores debido al momento en que los agentes bajan su velocidad tienen más riesgo de encontrarse con algún agente infectado, además de que el agente amigo puede estar también infectado. Esto solo afecta al porcentaje de infectados pero sigue el mismo comportamiento dependiendo de la probabilidad de vacunación, mientras más alta sea, menos infectados hay. La gráfica ahora se asemeja más a la de una exponencial.

En la figura 5 se observa mejor como llega a ser el porcentaje de infectados en los tres casos.

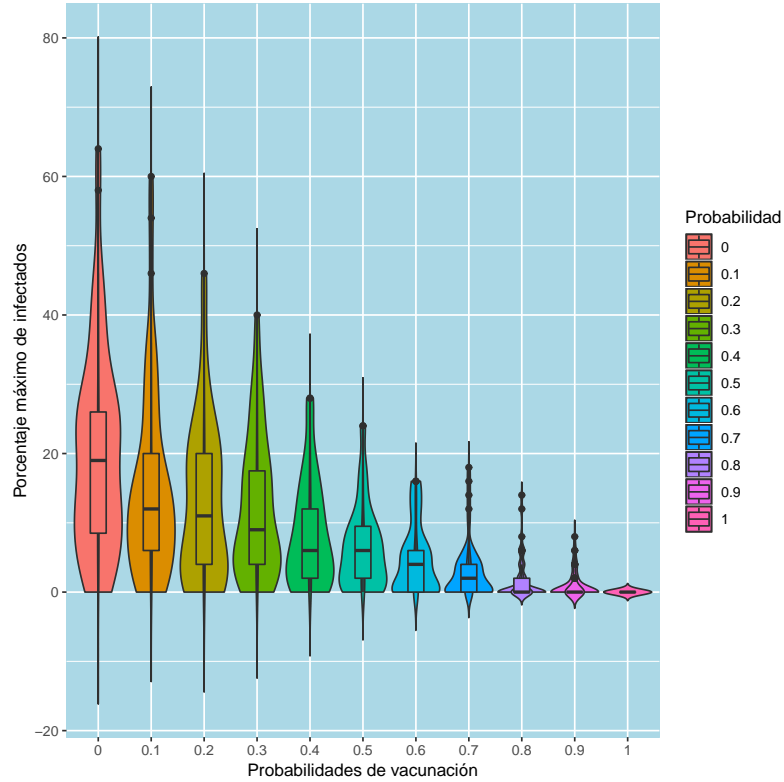
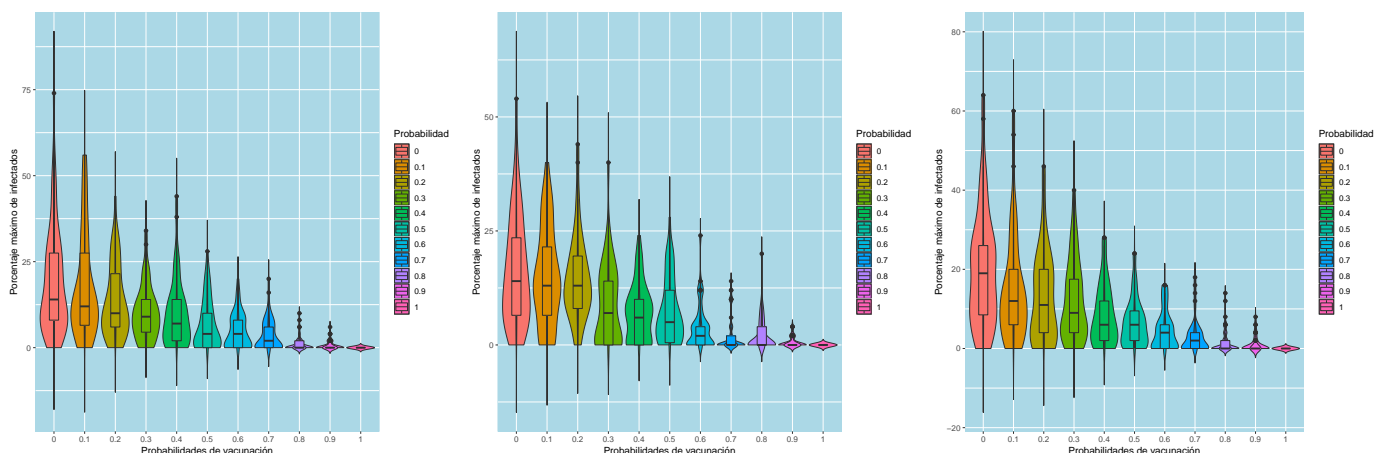


Figura 4: Porcentaje máximo de agentes infectados con relación a las probabilidades de ser vacunado respecto al efecto de las amistades que tienen los agentes.



(a) Probabilidades de vacunación con relación al mayor porcentaje de infectados.

(b) Efecto de punto medio aleatorio.

(c) Efecto de agentes con amistades.

Figura 5: Comparación de los efectos realizados en cada reto de acuerdo a la tarea principal.

Referencias

- [1] The R Project for Statistical Computing. 2019. URL <https://www.r-project.org/>.
- [2] Minitab Inc. Interpret the key results for Kruskal-Wallis Test. 2019. URL <https://support.minitab.com/en-us/minitab-express/1/help-and-how-to/modeling-statistics/anova/how-to/kruskal-wallis-test/interpret-the-results/key-results/>.
- [3] Schaeffer, E. Práctica 6: Sistema multiagente. 2019. URL <https://elisa.dyndns-web.com/teaching/comp/par/p6.html>.
- [4] Wickham, H. Ggplot2: Elegant graphics for data analysis. 2016. URL <https://ggplot2.tidyverse.org/>.