

Práctica N° 6

Sistema multiagente

Nombre: José Adrián García Fuentes
Fecha: 21/Marzo/2021

Profesor: Satu Elisa Schaeffer

1. Introducción

Un sistema multiagente es un poco como un autómata celular: hay un conjunto de entidades con estados internos que pueden observar estados de los otros 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. En la sexta práctica se implementará un sistema multiagente con una aplicación en epidemiología. Los agentes podrán estar en uno de tres estados: susceptibles, infectados o recuperados (esto se conoce como el modelo SIR) [1].

Los parámetros serán el número de agentes n y la probabilidad de infección al inicio P_i . Vamos a suponer, por simplicidad, que la infección produce inmunidad en los recuperados, por lo cual solamente los susceptibles podrán ser infectados. La probabilidad de contagio será en nuestro caso proporcional a la distancia euclidiana entre dos agentes $d(i, j)$ de la siguiente manera (1):

$$P_c = \begin{cases} 0, & \text{si } d(i, j) \geq r, \\ \frac{r - d}{r}, & \text{en otro caso,} \end{cases} \quad (1)$$

donde r es un umbral [1].

Nuestros agentes tendrán coordenadas x y y , una dirección y una velocidad (expresadas las dos últimas simplemente en términos de Δx y Δy). Vamos a posicionar los agentes, por ahora, uniformemente al azar en un torus formado por doblar un rectángulo de $l \times l$, visualizando en todo momento el rectángulo en dos dimensiones [1].

2. Objetivo

- Vacunar con probabilidad P_v a los agentes al momento de crearlos de tal forma que están desde el inicio en el estado R y ya no podrán contagiarse ni propagar la infección [1].
- Estudiar el efecto estadístico del valor de P_v en el porcentaje máximo de infectados durante la simulación y el momento en el cual se alcanza ese máximo [1].

3. Metodología

La metodología empleada se realizó a través de RStudio [2] llevando a cabo los pasos señalados en la *Práctica 6: Sistema multiagente* [1], a partir del código en el repositorio de Schaeffer [3], se hicieron modificaciones para tener un sistema multiagente en el que nuestro agente no viviera en una posición fija en la celda sino que pudiera moverse de posición en el cuadro, cada agente estará en uno de 3 posibles estados (susceptibles, infectados o recuperados) la probabilidad de la infección inicial es 0,05 % y la velocidad máxima es 1/30 la proporción de recuperación será 0,02 %, el código completo de la metodología empleada se encuentra en el repositorio de GitHub [4].

```
1 pi <- 0.05
2 pr <- 0.02
3 v <- 1 / 30
4 pv <- 0
5 pv_paso <- 0.10
6 for (vac in 1:11) {
7   agentes <- data.frame(x = double(), y = double(),
8                         dx = double(), dy = double(),
9                         estado = character())
10  rnd<-runif(1)
11  for (i in 1:n) {
12    rnd<-runif(1)
13    if (rnd < pv) {
14      e <- "R"
15    } else {
16      e <- "S"
17      if (rnd < pi) {
18        e <- "I"
19      }
20    }
21    agentes <- rbind(agentes, data.frame(x = runif(1, 0, 1),
22                                         y = runif(1, 0, 1),
23                                         dx = runif(1, -v, v),
24                                         dy = runif(1, -v, v),
25                                         estado = e))
26  }
27  levels(agentes$estado) <- c("S", "I", "R")
28  epidemia <- integer()
29  r <- 0.1
30  rm <- 0.3
31  pm <- 0.2
32  ka <- 5
33  tmax <- 100
34  digitos <- floor(log(tmax, 10)) + 1
35  for (tiempo in 1:tmax) {
36    infectados <- dim(agentes[agentes$estado == "I",])[1]
37    epidemia <- c(epidemia, infectados)
38    if (infectados == 0) {
39      break
40    }
41    contagios <- rep(FALSE, n)
42    for (i in 1:n) {
43      a1 <- agentes[i, ]
44      if (a1$estado == "I") {
45        for (j in 1:n) {
46          if (!contagios[j]) {
47            a2 <- agentes[j, ]
48            if (a2$estado == "S") {
49              dx <- a1$x - a2$x
```

```

50         dy <- a1$y - a2$y
51         d <- sqrt(dx^2 + dy^2)
52         if (d < r) {
53             p <- (r - d) / r
54             if (runif(1) < p) {
55                 contagios[j] <- TRUE
56             }
57         }
58     }
59 }
60 }
61 }
62 }
63 for (i in 1:n) {
64     a <- agentes[i, ]
65     if (contagios[i]) {
66         a$estado <- "I"
67     } else if (a$estado == "I") {
68         if (runif(1) < pr) {
69             a$estado <- "R"
70         }
71     }
72     a$x <- a$x + a$dx
73     a$y <- a$y + a$dy
74     if (a$x > 1) {
75         a$x <- a$x - 1
76     }
77     if (a$y > 1) {
78         a$y <- a$y - 1
79     }
80     if (a$x < 0) {
81         a$x <- a$x + 1
82     }
83     if (a$y < 0) {
84         a$y <- a$y + 1
85     }
86     agentes[i, ] <- a
87 }
88 aS <- agentes[agentes$estado == "S",]
89 aI <- agentes[agentes$estado == "I",]
90 aR <- agentes[agentes$estado == "R",]
91 t1 <- paste(tiempo, "", sep="")
92 while (nchar(t1) < digitos) {
93     t1 <- paste("0", t1, sep="")
94 }
95 salida <- paste("p6_t", t1, ".png", sep="")
96 tiempo <- paste("Paso", tiempo)
97 png(salida)
98 plot(1, type="n", main=tiempo, xlim=c(0, 1), ylim=c(0, 1), xlab="x", ylab="y")
99 if (dim(aS)[1] > 0) {
100     points(aS$x, aS$y, pch=15, col="chartreuse3", bg="chartreuse3")
101 }
102 if (dim(aI)[1] > 0) {
103     points(aI$x, aI$y, pch=16, col="firebrick2", bg="firebrick2")
104 }
105 if (dim(aR)[1] > 0) {
106     points(aR$x, aR$y, pch=17, col="goldenrod", bg="goldenrod")
107 }
108 graphics.off()
109 }

```

4. Resultados

Se obtuvo el código secuencia de GitHub de Schaeffer [3] y se añadió probabilidad de vacunación a los agentes al momento de crearlos de tal forma que desde el inicio están en el estado recuperado y ya no podrán contagiarse ni propagar la infección se obtuvo el efecto estadístico del valor de probabilidad de vacunación (cuadro 1) en el porcentaje máximo de infectados durante la simulación y el momento en el cual se alcanza el máximo se generó un archivo .gif [5] de la simulación en la figura 1 se muestra la posición de los agentes cada uno al momento de iniciar tuvo una probabilidad de vacunación.

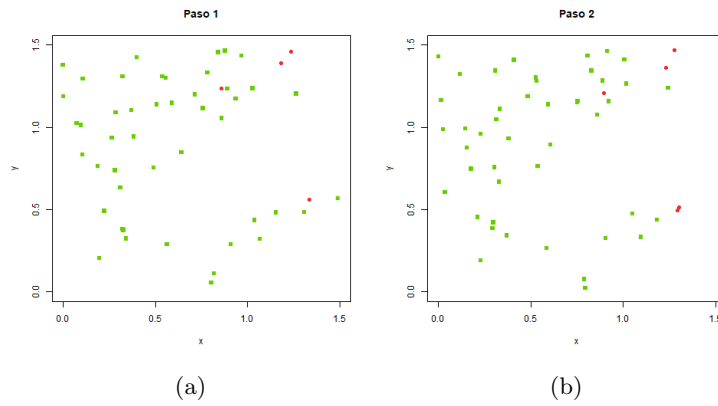


Figura 1: Inicio simulación de agentes

Debido a que cada agente ya no propaga la infección y no puede contagiarse al final de la simulación el porcentaje de agentes se encuentra en un estado recuperado para mejor entendimiento ver figura 2.

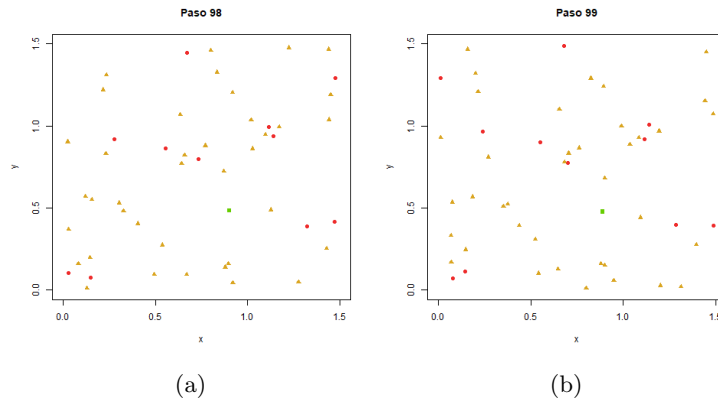


Figura 2: Fin de simulación de agentes

El efecto estadístico de la probabilidad de vacunación aunque es muy poco variable se observaría que dependiendo de la probabilidad de vacunación existe una ligera disminución en el porcentaje de contagiados salvo algunas excepciones 30 % y 50 %.

Efecto estadístico del valor de probabilidad de vacunación.		
Probabilidad de vacunación	Máximo de infectados	Momento máximo de infectados (t)
0 %	65 %	49
10 %	60 %	52
20 %	58 %	19 y 38
30 %	82 %	17
40 %	56 %	20
50 %	80 %	17 y 22
60 %	76 %	20
70 %	68 %	30
80 %	53 %	26
90 %	50 %	60
100 %	0 %	0

Al comparar que la población de agentes se encuentra completamente vacunada (figura 3) con una vacunación nula (figura 3a) se observa un porcentaje alto de agentes infectados (ver figura 3b)

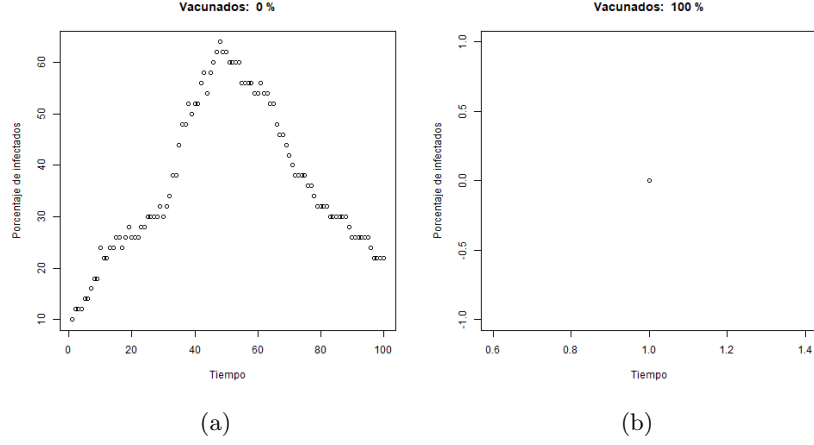


Figura 3: Comparación de agentes completamente vacunados con ningun agente vacunado.

Aunque se esperaba que a mayor porcentaje de vacunados el porcentaje máximo de infectados disminuyera gradualmente existen algunas variantes como ejemplo se muestra la figura 4a donde una probabilidad del 20 % de agentes vacunados mostró un pico más alto que la probabilidad del 10 % y se observa un repunte en el contagio en los tiempos 19 y 38 o bien en la figura 4b donde ocurre lo mismo sin embargo se observa una baja en el porcentaje de infectados con respecto al tiempo (figura 4).

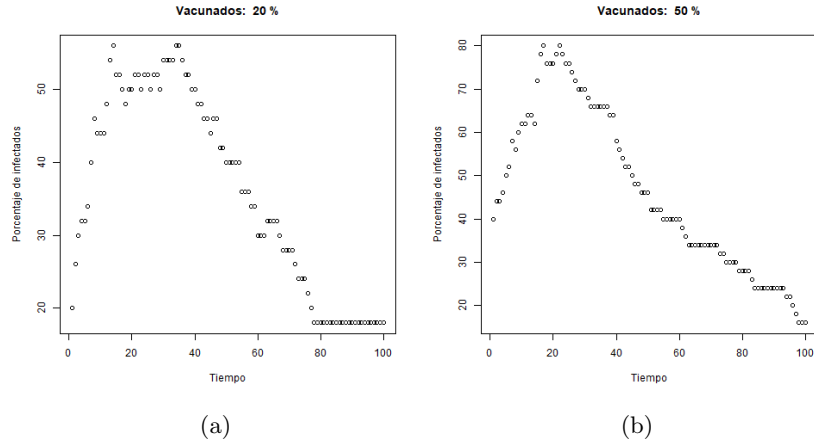


Figura 4: Comparación de porcentaje de vacunacion con inconsistencias.

En la figura 5 se encuentra un gráfico de caja-bigote que determina el número de infectados por cada porcentaje de vacunados de manera descendente.

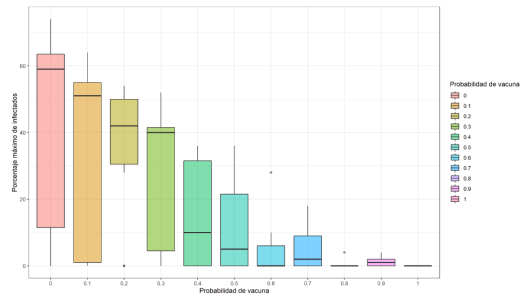


Figura 5: Comparación de porcentaje de vacunación

5. Conclusión

Al estudiar el efecto estadístico del valor de la probabilidad de vacunación con el porcentaje de agentes infectados en primera instancia se esperaría que el porcentaje de infectados disminuyera de forma gradual hasta que el total de la población de agentes estuviera completamente vacunado sin embargo en dos ocasiones se obtiene inconsistencias que tal vez estén dadas por las direcciones en las que los agentes se desplazan, un ejemplo simple sería la pandemia por Covid-19 en la que aunque la población no esté vacunada se tienen repuntes de contagios.

6. Reto 1

Cambiar los patrones de movimiento para que no tengan una trayectoria fija, cambiar los agentes a que utilice el modelo de punto intermedio aleatorio (inglés: random waypoint model): cada agente tiene una posición meta (x, y) hacia al cual se mueve con una velocidad v ; al alcanzar (o superar) su meta, elige al azar una nueva meta uniformemente al azar. La velocidad de cada agente es una constante, normalmente distribuido sobre la población de agentes. Examina si surgieron cambios en el efecto de P_v por esta modificación [1].

Se obtuvo un código similar de GitHub de Vázquez [6] y se adaptó una meta cambiante para cada agente el objetivo es concluir si este nuevo tipo de movimiento de los agentes afecta o no la conclusión anterior.

```
1 xc<-runif(1,0,1)
2 yc<-runif(1,0,1)
3 px<-runif(1,0,1)
4 py<-runif(1,0,1)
```

Al declarar los puntos iniciales en los que aparecen los agentes y sus puntos meta, se genera un número aleatorio de pasos con el que se calcula la velocidad con la que avanza el punto meta cada agente en la figura 6 se muestra un diagrama caja-bigote de manera descendente aunque en comparación del gráfico de la tarea base los valores en el porcentaje de infectados no son tan separados se concentran en un rango ligeramente alto.

```
1 vx<-((px-xc)/pasos)
2 vy<-((py-yc)/pasos)
```

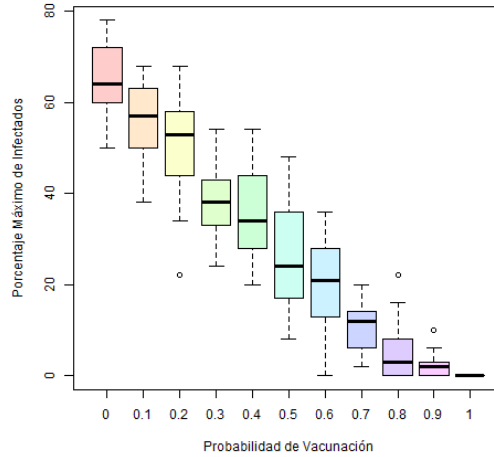


Figura 6: Gráfico caja-bigote porcentaje de infectados al establecer un punto meta aleatorio.

7. Conclusión

Se obtiene el mismo efecto en el que el valor de P_v afecta el porcentaje de infectados sin embargo debido a que las interacciones entre agentes son muy variables el porcentaje de infectados por tiempo es mayor para cada valor de P_v .

8. Reto 2

Los agentes tienen amistades: si se encuentran a una distancia euclidiana no mayor a r_a de un amigo suyo, se disminuye su velocidad a la mitad por k_a iteraciones (para saludar a su amigo). Cada par de agentes tiene una amistad con una probabilidad P_a . Examina nuevamente si surgieron cambios en el efecto de P_v por esta modificación, con valores $0 < r_a < 1$, $k_a > 1$ y $0 < P_a \ll 1$ de tu elección [1].

```
1
2 library(gifski)
3
4 setwd("C:/Users/ADRIAN GARCIA/Documents/practica6")
5
6 l <- 1.5
7 n <- 50
8 pi <- 0.05
9 pr <- 0.02
10 v <- 1 / 30
11 pa<-0.02
12 pv <- 0
13 pv_paso <- 0.10
14
15 for (vac in 1:11) {
16   agentes <- data.frame(x = double(), y = double(),
17                         dx = double(), dy = double(),
18                         estado = character())
19   rnd<-runif(1)
20   for (i in 1:n) {
21     rnd<-runif(1)
22     if (rnd < pv) {#vacunados
23       e <- "R"
24     } else {
25       e <- "S"
26       if (rnd < pi) {
27         e <- "I"
28       }
29     }
30     agentes <- rbind(agentes, data.frame(x = runif(1, 0, 1),
31                                           y = runif(1, 0, 1),
32                                           dx = runif(1, -v, v),
33                                           dy = runif(1, -v, v),
34                                           estado = e))
35   }
36
37 levels(agentes$estado) <- c("S", "I", "R")
38 epidemia <- integer()
39 r <- 0.1
40 rm <- 0.3
41 pm <- 0.2
42 ka <- 5
43 tmax <- 100
```

```

44 digitos <- floor(log(tmax, 10)) + 1
45
46 for (tiempo in 1:tmax) {
47   infectados <- dim(agentes[agentes$estado == "I",,])[1]
48   epidemia <- c(epidemia, infectados)
49   if (infectados == 0) {
50     break
51   }
52   contagios <- rep(FALSE, n)
53   for (i in 1:n) {
54     a1 <- agentes[i, ]
55     if (a1$estado == "I") {
56       for (j in 1:n) {
57         if (!contagios[j]) {
58           a2 <- agentes[j, ]
59           if (a2$estado == "S") {
60             dx <- a1$x - a2$x
61             dy <- a1$y - a2$y
62             d <- sqrt(dx^2 + dy^2)
63             if (d < rm) { # umbral amistad
64               p <- (r - d) / r
65               if (runif(1) < pm) {
66                 for (i in 1:ka) {
67                   dx <- (a1$x - a2$x)/2
68                   dy <- (a1$y - a2$y)/2
69                 }
70               }
71             }
72             if (d < r) {
73               p <- (r - d) / r
74               if (runif(1) < p) {
75                 contagios[j] <- TRUE
76               }
77             }
78           }
79         }
80       }
81     }
82   }
83   for (i in 1:n) {
84     a <- agentes[i, ]
85     if (contagios[i]) {
86       a$estado <- "I"
87     } else if (a$estado == "I") {
88       if (runif(1) < pr) {
89         a$estado <- "R"
90       }
91     }
92     a$x <- a$x + a$dx
93     a$y <- a$y + a$dy
94     if (a$x > 1) {
95       a$x <- a$x - 1
96     }
97     if (a$y > 1) {
98       a$y <- a$y - 1
99     }
100    if (a$x < 0) {
101      a$x <- a$x + 1
102    }
103    if (a$y < 0) {
104      a$y <- a$y + 1
105    }

```

```

106     agentes[i, ] <- a
107   }
108   aS <- agentes[agentes$estado == "S",]
109   aI <- agentes[agentes$estado == "I",]
110   aR <- agentes[agentes$estado == "R",]
111   tl <- paste(tiempo, "", sep="")
112   while (nchar(tl) < digitos) {
113     tl <- paste("0", tl, sep="")
114   }
115   salida <- paste("p6_t", tl, ".png", sep="")
116   tiempo <- paste("Paso", tiempo)
117   png(salida)
118   plot(1, type="n", main=tiempo, xlim=c(0, 1), ylim=c(0, 1), xlab="
x", ylab="y")
119   if (dim(aS)[1] > 0) {
120     points(aS$x, aS$y, pch=15, col="chartreuse3", bg="chartreuse3")
121   }
122   if (dim(aI)[1] > 0) {
123     points(aI$x, aI$y, pch=16, col="firebrick2", bg="firebrick2")
124   }
125   if (dim(aR)[1] > 0) {
126     points(aR$x, aR$y, pch=17, col="goldenrod", bg="goldenrod")
127   }
128   graphics.off()
129 }
130 salida <- paste("vacuna=", pv, ".png", sep="")
131 png(salida)
132 plot(1:length(epidemia), 100 * epidemia / n, xlab="Tiempo", ylab="
Porcentaje de infectados",
133     main=paste("Vacunados: ",pv * 100, "%"))
134 graphics.off()
135 pv <- pv + pv_paso
136 }
137 png_files <- list.files( pattern = "p6_t.*png$", full.names = TRUE)
138 gifski(png_files, gif_file = "animation.gif", width = 800, height =
600, delay = 1)

```

A partir del código de la tarea base se añadió a cada agente amistades con (P_a de 0,02 %) y si estos se encontraban a una distancia euclídeana de un amigo suyo este disminuiría su velocidad se añadió probabilidad de vacunación a los agentes al momento de crearlos, se obtuvo el efecto estadístico del valor de probabilidad de vacunación (cuadro 2) en el porcentaje máximo de infectados durante la simulación y el momento en el cual se alcanza el máximo se generó un archivo .gif [5] de la simulación.

En la figura 7 se muestran los primeros 4 pasos de una simulación si observamos en el paso uno (7a) se encuentra un agente contagiado en la parte central, un poco en la parte superior derecha, en el paso 2 (7b) se encuentra con un amigo provocando disminución en su velocidad a la mitad, en el paso 3 (7c) se observa que el mismo agente no se separó mucho en distancia del agente con el que se encontró en comparación del movimiento de los otros agentes, observe el paso 4 (7d) la distancia entre todos los agentes es mayor en comparación con el agente antes mencionado.

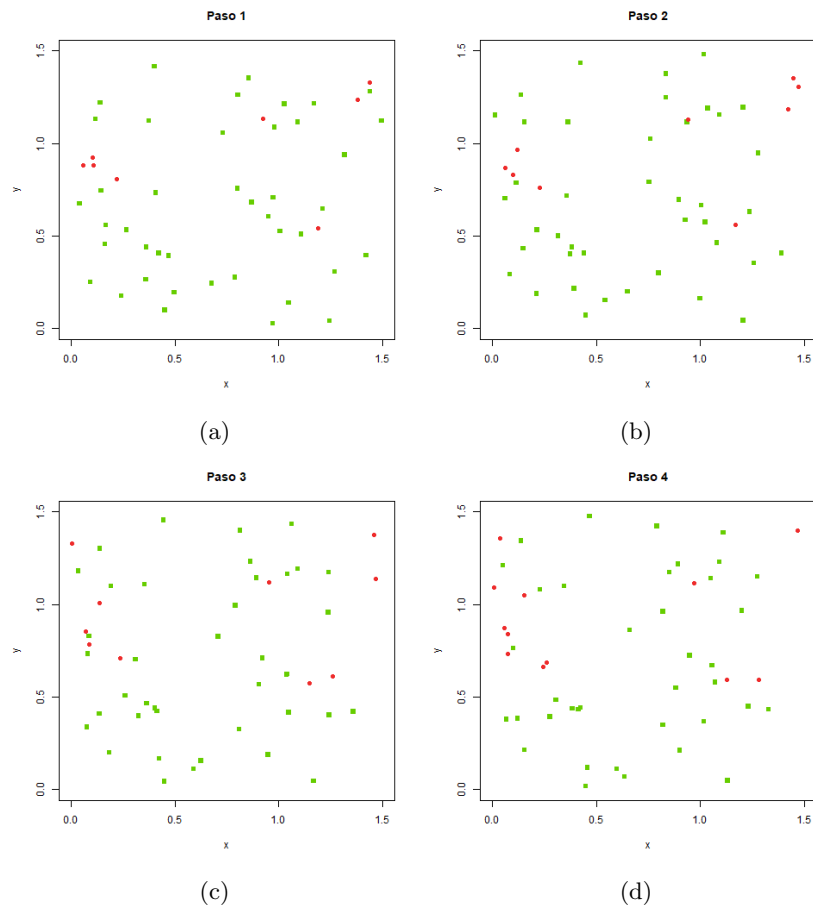


Figura 7: Inicio simulación de agentes

Efecto estadístico del valor de probabilidad de vacunación.		
Probabilidad de vacunación	Máximo de infectados	Momento máximo de infectados (t)
0 %	55 %	60
10 %	50 %	60
20 %	65 %	30
30 %	56 %	60
40 %	86 %	17
50 %	78 %	22
60 %	78 %	17 y 30
70 %	70 %	18
80 %	82 %	17
90 %	2 %	3
100 %	0 %	0

Al realizar una comparación con el grafico de la tarea base (figura 8a) y el efecto que tiene la probabilidad de amistad (figura 8b) en los agentes que no se encuentran vacunados (figura 8) se observa un porcentaje alto de agentes infectados (ver figura 8).

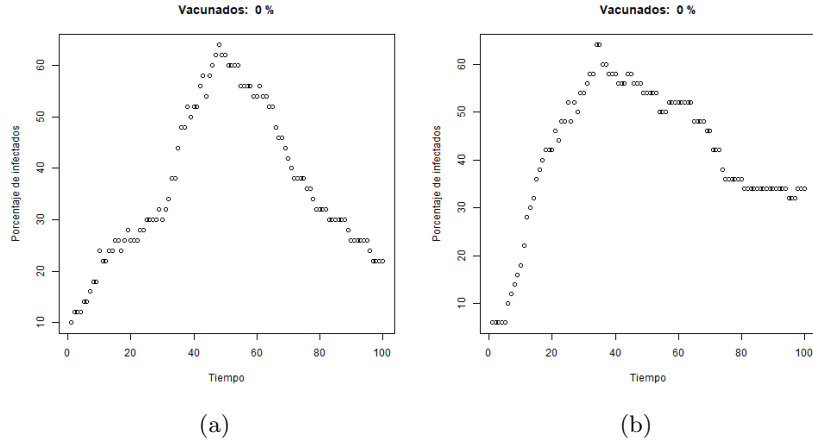


Figura 8: Comparación de gráficos tarea base y reto 2.

Se obtienen gráficos muy similares sin embargo el porcentaje de infectados por tiempo es ligeramente mayor tal vez esto sea dado por el tiempo en que se desplazan los agentes cuando interactúan con un amigo y las probabilidades en que otro agente se pueda contagiar dependerán de si se topan con él.

9. Conclusión.

El efecto estadístico de la probabilidad de vacunación en comparación de la tarea base cambia aunque en algunas ocasiones cambia muy poco se podría decir que se mantiene en un porcentaje de infectados en un rango de 55 a 80 % a excepción del 90 % de vacunados que solo se infectó un 2 % y el tiempo de contagio fue rapido tal vez estos cambios en la variacion de infectados es debido al tiempo en el que se retrasan los agentes al estar con un amigo y acorta de cierta manera la probabilidad de contagio.

Referencias

- [1] E. Schaeffer, “Práctica 6: Sistema multiagente,” Marzo 2021. <https://https://elisa.dyndns-web.com/teaching/comp/par/p6.html>.
- [2] J. J. Allaire, “Rstudio,” Marzo 2021. <https://rstudio.com>.
- [3] E. Schaeffer, “Práctica 6: Sistema multiagente,” Marzo 2021. <https://github.com/fuentesadrian/Simulation/tree/master/MultiAgent>.
- [4] J. A. Garcia Marzo 2021. <https://github.com/fuentesadrian/SIMULACION-DE-NANOMATERIALES/tree/main/Tarea%206>.
- [5] A. Garcia, “archivo gif,” 2021. <https://github.com/fuentesadrian/SIMULACION-DE-NANOMATERIALES/blob/main/Tarea%206/animation.gif>.
- [6] f. Vazquez, “Práctica 6: Sistema multiagente,” Marzo 2021. <https://github.com/fvzqa/Simulacion>.