

Práctica 5: Sistema Multiagente

Raul L.

22 de marzo de 2022

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 esta práctica vamos a implementar 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].

2. Objetivo

Estudia el efecto de contención en el sentido de que agentes infectados reduzcan su velocidad de movimiento a la mitad durante su infección. Determina con pruebas estadísticas adecuadas si este cambio produce un efecto significativo en la magnitud de la epidemia (la altura del pico en la curva del porcentaje de infectados por iteración) y en la velocidad de ella la iteración en la cual se llega por la primera vez al valor pico [1].

3. Código

Para este código se utilizó como base el código de la doctora donde se hicieron modificaciones variando la velocidad para comprobar si afectaría en el porcentaje de contagios en el pico como también se agregaron un numero de repeticiones para poder tener una estimación considerable y poder hacer una correcta prueba estadística.

Código en Python

<https://github.com/satuelisa/Simulation/blob/master/MultiAgent/movement.py>

Código creado en Python

https://github.com/Raullr28/Resultados/blob/main/P4/codigo_celdas.py

```

if a.estado == 'I':
    if div != 0:
        x = a.x + (a.dx/div)
        y = a.y + (a.dy/div)
    elif div == 0:
        x = a.x + (a.dx*div)
        y = a.y + (a.dy*div)
    else:
        x = a.x + a.dx
        y = a.y + a.dy

```

Código 1: Representa la automatización para variar la velocidad de los infectados.

```

for div in 1,2,0:
    print("##### velocidad 1/",div,"#####")
    pico, tmp= [], []
    for rep in range(repeticiones):
        agentes = pd.DataFrame()
        agentes['x'] = [uniform(0, 1) for i in range(n)]
        agentes['y'] = [uniform(0, 1) for i in range(n)]
        agentes['dx'] = [uniform(-v, v) for i in range(n)]
        agentes['dy'] = [uniform(-v, v) for i in range(n)]
        agentes['estado'] = ['S' if random() > pi else 'I' for i in range(n)]
        epidemia = []
        for tiempo in range(tmax):
            conteos = agentes.estado.value_counts()
            infectados = conteos.get('I', 0)
            epidemia.append(infectados)
            if infectados == 0:
                break

```

Código 2: Representación función cambio de velocidad.

4. Resultados

En las 3 figuras principales se muestra el comportamiento del pico con respecto a la reducción de la velocidad para demostrar como puede influir la velocidad de propagación en el pico de la pandemia, como prueba estadística se realizaron comparaciones con 30 repeticiones para cada caso de velocidad en graficas de caja-bigote.

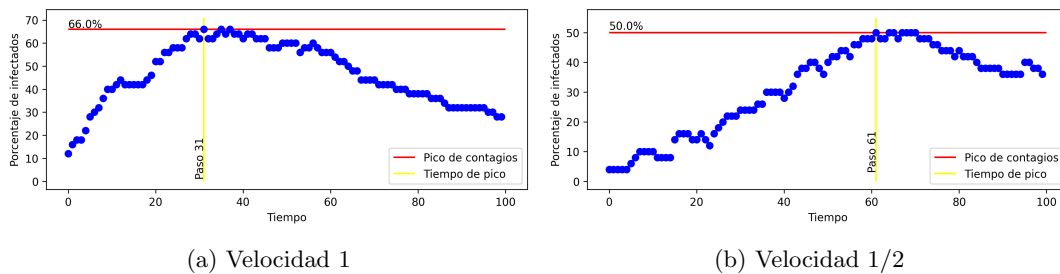


Figura 1: Gráfica de picos a diferentes velocidades.

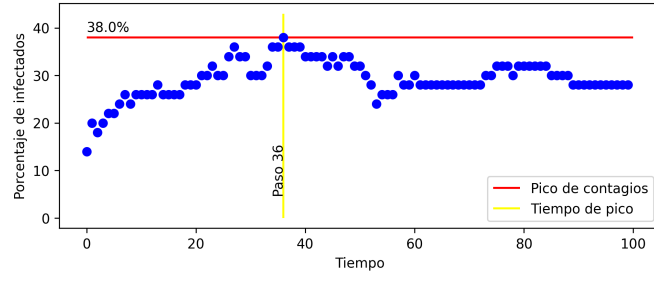
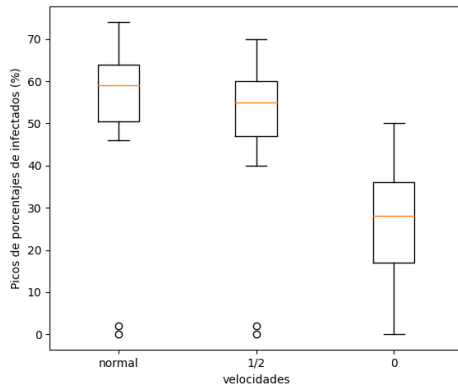
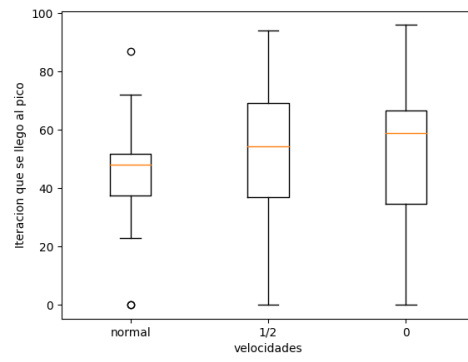


Figura 2: Velocidad nula 0.



(a) Velocidad



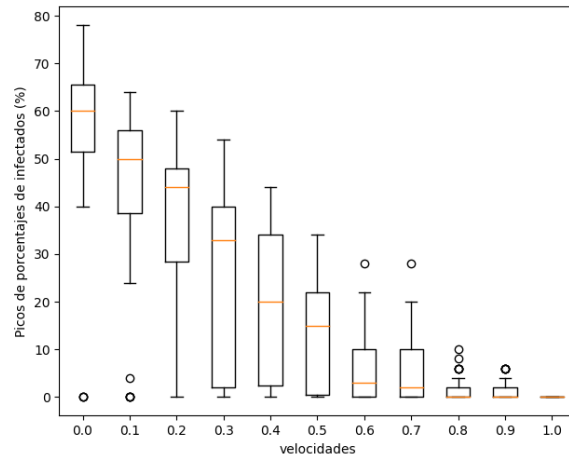
(b) Picos

Figura 3: Gráficas Estadísticas.

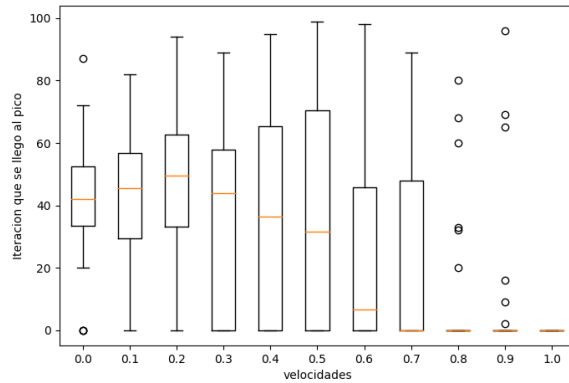
5. Reto 1

En el primer reto, vacuna 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. Estudia el efecto estadístico del valor de p_v en (de cero a uno en pasos de 0.1) el porcentaje máximo de infectados durante la simulación y el momento (iteración) en el cual se alcanza ese máximo..

6. Resultados



(a) porcentaje



(b) Pico

Figura 4: Gráfica comparativa.

7. Conclusión

Se demostró que disminuyendo la velocidad de los puntos como a su vez vacunando, hace que se reduzca la probabilidad de infectados y hace que arde en llegar más rápido el pico.

Referencias

- [1] E. Schaeffer. Práctica 6: sistema multiagente. 2022. URL <https://satuelisa.github.io/simulation/p6.html>.