

UNIVERSIDAD AUTÓNOMA DE MADRID  
ESCUELA POLITÉCNICA SUPERIOR



Doble Grado en Ingeniería Informática y Matemáticas

## TRABAJO FIN DE GRADO

Determinación de una estrategia óptima  
de vacunación basada en redes sociales.

Autor: Litzy Tatiana Rocha Avila

Tutor: Simone Santini

junio 2021



# Índice general

---

<b>Resumen</b>	<b>IX</b>
<b>Abstract</b>	<b>XI</b>
<b>1 Introducción</b>	<b>1</b>
<b>2 Estado del arte</b>	<b>3</b>
2.1 El modelo SIR . . . . .	3
2.1.1 Información que aporta el modelo . . . . .	5
2.1.2 Limitaciones del modelo . . . . .	5
2.2 Grafos Aleatorios y redes sociales . . . . .	6
2.2.1 Definiciones básicas . . . . .	6
2.2.2 Creación del grafo: Modelo Barabási–Albert . . . . .	6
2.3 Difusión de un virus en un grafo . . . . .	8
2.4 Programación Dinámica . . . . .	9
2.4.1 El Problema de Decisión en Múltiples Pasos . . . . .	9
2.4.2 Principio de optimalidad . . . . .	10
2.5 Estrategias de vacunación . . . . .	11
<b>3 Modelo de vacunación</b>	<b>15</b>
3.1 El problema . . . . .	15
3.2 Grafos . . . . .	16
3.2.1 Creación de grafos . . . . .	16
3.2.2 Grafos utilizados . . . . .	18
3.2.3 Difusión de la enfermedad en el grafo . . . . .	18
3.3 Parámetros epidemiológicos . . . . .	20
3.4 Optimización . . . . .	22
3.4.1 Formalización . . . . .	23
3.4.2 Modos de vacunación . . . . .	25
3.5 Implementación . . . . .	25
3.5.1 Almacenamiento de resultados . . . . .	29
3.5.2 Valores utilizados en la simulación . . . . .	30
<b>4 Resultados</b>	<b>31</b>
4.1 Resultados de la Programación Dinámica . . . . .	31
4.1.1 Topología $A\alpha$ . . . . .	32
4.1.2 Topología $B\alpha$ . . . . .	33
4.1.3 Topología $A\beta$ . . . . .	34
4.1.4 Topología $B\beta$ . . . . .	35
4.2 Comparación con otras estrategias . . . . .	36

<b>5 Conclusiones</b>	<b>39</b>
5.1 Ventajas . . . . .	39
5.2 Limitaciones . . . . .	40
5.3 Posibles mejoras . . . . .	40
<b>Bibliografía</b>	<b>41</b>

# Índice de figuras

3.1	Algoritmo que genera un grafo utilizando el modelo Barabási–Albert. . . . .	16
3.2	Algoritmo que une dos grafos usando <i>preferential attachment</i> . . . . .	17
3.3	Pseudocódigo que describe la evolución de la epidemia en un día. . . . .	26
3.4	Pseudocódigo que describe el proceso de vacunación. . . . .	27
3.5	Pseudocódigo que describe el proceso de vacunación y evolución de la epidemia en $T$ días. . . . .	27
3.6	Pseudocódigo que describe la resolución del problema planteado usando Programación Dinámica . . . . .	29
4.1	Resultados topología $A\alpha$ en 49 días de simulación . . . . .	32
4.2	Resultados topología $A\alpha$ en 98 días de simulación . . . . .	32
4.3	Resultados topología $B\alpha$ en 49 días de simulación . . . . .	33
4.4	Resultados topología $B\alpha$ en 98 días de simulación . . . . .	33
4.5	Resultados topología $A\beta$ en 49 días de simulación . . . . .	34
4.6	Resultados topología $A\beta$ en 98 días de simulación . . . . .	34
4.7	Resultados topología $B\beta$ en 49 días de simulación . . . . .	35
4.8	Resultados topología $B\beta$ en 98 días de simulación . . . . .	35



# Índice de tablas

3.1	Parámetros y código utilizado para generar grafos . . . . .	18
3.2	Probabilidades de transición entre los estados $S, I, R, V$ . . . . .	19
3.3	Valores de los parámetros epidemiológicos utilizados en la simulación. . . . .	22
4.1	Comparación del número de víctimas con otros modelos partiendo de un infectado joven . . . . .	37
4.2	Comparación del número de víctimas con otros modelos partiendo de 70 infectados jóvenes y 15 infectados mayores . . . . .	38





# Resumen

---

El trabajo está dividido en cinco capítulos. El primer capítulo consta de una breve introducción al tema mediante la motivación que nos llevó a hacer este trabajo y un breve planteamiento de la solución.

El segundo capítulo contiene una explicación de las herramientas necesarias para el desarrollo del trabajo. Entre ellas encontramos el modelo epidemiológico SIR para derivar los parámetros epidemiológicos de una enfermedad, el Modelo Barabási-Albert para crear grafos aleatorios que poseen una distribución de vecino libre de escala, la difusión de un virus en un grafo para las posibles estrategias a desarrollar en el capítulo tres, la introducción a Programación Dinámica para resolver el Problema de Decisión en Múltiples Pasos que será la base de nuestra optimización, y por último la explicación detallada de varias estrategias de vacunación que darán pie a nuestro estudio.

El tercer capítulo consiste en la descripción detallada de nuestro modelo utilizando las herramientas descritas en el capítulo anterior. Consta del planteamiento del problema, la descripción de los grafos utilizados, la derivación de los parámetros epidemiológicos, la optimización realizada mediante el uso de Programación Dinámica, y la implementación del modelo.

En el cuarto capítulo presentamos los resultados de la aplicación de nuestro trabajo mediante el análisis de gráficas generadas a partir de simulaciones que utilizan nuestro algoritmo de optimización y comparamos los resultados con otras tres posibles estrategias de vacunación.

En el último capítulo presentamos las conclusiones, donde detallamos las ventajas, las limitaciones y las posibles mejoras del modelo.

## Palabras clave

Programación Dinámica, grafos, redes sociales, SIR, difusión en grafo, optimización, Problema de Decisión en Múltiples Pasos, vacunación, virus.



# Abstract

---

The work is divided into five chapters. The first chapter consists of a brief introduction to the subject through the motivation that led us to do this work and a brief approach to the solution.

The second chapter contains an explanation of the necessary tools for the development of the work. Among them we find the epidemiological model SIR to derive the epidemiological parameters of a disease, the Barabási-Albert Model to create random graphs that have a scale-free neighbor distribution, the diffusion of a virus in a graph for the possible strategies to be developed in the chapter three, the introduction to Dynamic Programming to solve the Multistage Decision Process Problem that will be the basis of our optimization, and finally the detailed explanation of various vaccination strategies that will give rise to our study.

The third chapter consists of the detailed description of our model using the tools described in the previous chapter. It consists of the statement of the problem, the description of the graphs used, the derivation of the epidemiological parameters, the optimization carried out using Dynamic Programming, and the implementation of the model.

In the fourth chapter we present the results of the application of our work through the analysis of graphs generated from simulations that use our optimization algorithm and we compare the results with three other possible vaccination strategies.

In the last chapter we present the conclusions, where we detail the advantages, limitations and possible improvements of the model.

## Keywords

Dynamic Programming, graphs, social networks, SIR, graph spreading, optimization, Multistage Decision Process Problem, vaccination, virus.



# CAPÍTULO 1

## Introducción

---

Tras la llegada de la inesperada epidemia causada por la Covid-19 uno de los problemas que se nos planteó es como distribuir las vacunas entre la población debido a la limitación de dosis disponibles.

La estrategia que se está utilizando en la mayoría de países es realizar una vacunación por edades priorizando la población más vulnerable, es decir, los mayores. Una de las preguntas que nos surgió a raíz de esto fue si esta estrategia es la más óptima. Con el objetivo de resolver esta duda presentamos nuestro estudio.

El problema planteado es el siguiente: poseemos una cantidad de vacunas totales, pero o bien llega por lotes o bien nuestro sistema sanitario tan solo es capaz de vacunar a una cierta cantidad de personas al día. Nuestro objetivo es asignar el porcentaje de dosis que recibirán la población mayor y la población joven cada día con el objetivo de minimizar la cantidad de victimas totales.

La solución que presentamos consiste en utilizar Programación Dinámica para optimizar el reparto. Para ello el primer paso que realizamos consiste en dividir la población en dos grupos: el grupo de los jóvenes y el grupo de los mayores. Esta división se debe a que la edad es un factor fundamental a la hora de estudiar el número de víctimas.

Para simular la expansión de la epidemia a través de la población por una parte hemos decidido utilizar grafos, cada nodo del grafo representa un individuo de la población y cada enlace las interacciones sociales entre los individuos. Por otra parte, otra clave para determinar la evolución de la epidemia son los parámetros epidemiológicos, para derivarlos hemos utilizado el modelo SIR y los datos suministrados por el Ministerio de Sanidad español.

En cuanto a la optimización realizada, hemos dividido nuestro problema en varios posibles escenarios determinados por el número de infectados y vacunados jóvenes y mayores en un determinado intervalo temporal, de esta manera independientemente de la situación en la que nos encontremos el modelo nos proporcionará un camino óptimo para minimizar las victimas totales. Para encontrar los porcentajes diarios óptimos que debemos asignar a cada grupo hemos utilizado Programación Dinámica empezando por el último día y yendo hacia atrás para encontrar el camino que nos da el número de víctimas mínimo desde el primer día hasta el último.



## CAPÍTULO 2

# Estado del arte

---

En este capítulo se sentarán las bases y herramientas necesarias para abarcar el desarrollo del trabajo. La primera sección consiste en una introducción al modelo epidemilógico SIR. En la segunda sección nos encontramos con algunas nociones básicas de grafos y cómo construir un grafo que sirva como base para estudiar la evolución de una epidemia a través de sus nodos. Esta sección está fuertemente ligada con la siguiente: el estudio de la difusión de una enfermedad en un grafo. En la cuarta sección se encuentran las nociones básicas de Programación Dinámica que juega un papel muy importante en problemas de optimización complejos y que es el método de optimización que aplicaremos en este trabajo, La última parte del capítulo está centrada en la presentación de algunas estrategias de vacunación que darán pie a nuestro estudio.

### 2.1. El modelo SIR

El modelo SIR [8] es de uno de los modelos matemáticos epidemiológicos más simples utilizado para analizar la magnitud y el comportamiento de las enfermedades infecciosas de transmisión directa.

El modelo divide a la población en tres grupos disjuntos: la población susceptible (S), la población infectada (I) y la población recuperada (R). Aunque no tiene en cuenta a las víctimas, a efectos prácticos de las ecuaciones que lo describen, las víctimas se pueden considerar como parte del grupo de los recuperados. Para cada instante de tiempo  $t$ , denotamos la cantidad de personas pertenecientes a cada grupo como  $S(t)$ ,  $I(t)$  y  $R(t)$ . De este modo, el modelo describe como los individuos transitan entre los distintos grupos mediante un sistema no lineal de tres ecuaciones diferenciales ordinarias:

$$(2.1) \quad \frac{dS}{dt} = -\beta SI$$

$$(2.2) \quad \frac{dI}{dt} = \beta SI - \alpha I$$

$$(2.3) \quad \frac{dR}{dt} = \alpha I$$

Donde  $\beta$  es el ratio de transmisión y  $\alpha$  el ratio de recuperación.

La ecuación (2.1) expresa la variación instantánea de individuos susceptibles en un determinado momento. Dado que el modelo SIR no considera que puedan existir re-infecciones (una vez superada la enfermedad obtenemos anticuerpos de por vida), el número de individuos susceptibles tan solo puede disminuir con el tiempo, a eso se debe el signo negativo de esta ecuación. Por otra parte, el número de individuos que dejan de ser susceptibles se debe a una única razón: se han infectado. El término  $\beta SI$  indica el número de nuevos infectados por unidad de tiempo. Suponiendo que cada individuo infectado tenga  $k$  contactos por unidad de tiempo y  $k$  es independiente del tamaño de la población. Entonces  $k\frac{S}{N}$  de esos contactos son susceptibles. Si  $\tau$  representa la fracción de esos contactos que resultan contagiados, entonces cada individuo infectado contagia a  $k\tau\frac{S}{N}$  individuos. Por lo tanto,  $\beta = \frac{k\tau}{N}$ , donde  $\tau$  recibe el nombre de *transmisibilidad*.

La ecuación (2.2) expresa la variación instantánea de individuos infectados en un determinado momento. Como habíamos dicho previamente, el término  $\beta SI$  de esta ecuación indica el número de nuevos infectados por unidad de tiempo. Por otra parte, un porcentaje  $\alpha$  de los individuos que se encontraban infectados o bien se recuperarán o bien serán víctimas, por lo que pasarán al grupo de recuperados.

La ecuación (2.3) expresa la variación instantánea de individuos recuperados en un determinado momento. Como la única manera de que un individuo llegue a esta etapa es después de haber sido infectado, esta variación depende únicamente del número de infectados y el ratio de recuperación. El modelo considera que una vez hayamos superado la enfermedad, obtenemos anticuerpos de por vida, lo que implica que una vez llegados a esta etapa, no volveremos a cambiar de grupo.

El modelo SIR considera que la población total permanece constante y cada persona pertenece únicamente a uno de los tres grupos. Por lo tanto, suponiendo una población total  $N$ , en todo momento  $t$  se cumple que:

$$(2.4) \quad N = S(t) + I(t) + R(t)$$

Esto reduce nuestro sistema de 3 ecuaciones diferenciales ordinarias a las dos primeras, puesto que despejando de (2.4) podemos expresar  $R(t)$  como  $R(t) = N - S(t) - I(t)$ .

No se puede calcular la solución explícita del sistema planteado. Sin embargo, existen herramientas matemáticas que nos permiten extraer información y propiedades cualitativas de las soluciones.

**Definición 2.1.** Suponiendo que toda la población, es inicialmente susceptible excepto un individuo infectado, es decir,  $S(0) = N - 1$ ,  $I(0) = 1$ , y  $R(0) = 0$ . Se define el número básico de reproducción,  $R_0$ , como el número promedio de personas que se infectan a partir de un individuo infectado a lo largo del periodo de duración de la enfermedad. Matemáticamente está expresado por:

$$(2.5) \quad R_0 = \frac{\beta N}{\alpha}$$



**Teorema 2.2.** *Dependiendo del valor de  $R_0$  tenemos dos casos:*

1. *Si  $R_0 < 1$ , entonces  $I(t)$  decrece monótonamente hacia 0 cuando  $t \rightarrow \infty$ . Es decir, la enfermedad terminará desapareciendo.*
2. *Si  $R_0 > 1$ , entonces  $I(t)$  empezará a crecer rápidamente hasta alcanzar un máximo y después decrecerá hacia 0 cuando  $t \rightarrow \infty$ . Es, decir, cuanto mayor sea  $R_0$  más rápido se extenderá la enfermedad.*

El Teorema 2.2 nos lleva a la conclusión de que una enfermedad puede causar una epidemia en una población totalmente susceptible si nos encontramos en el caso  $R_0 > 1$ , es decir, si  $\beta N > \alpha$ .

### 2.1.1. Información que aporta el modelo

Para frenar la propagación de la enfermedad según el teorema 2.2 debemos reducir el valor de  $\beta$  y aumentar el valor de  $\alpha$ . Algunas de las medidas que ayudarán a frenar la propagación de la enfermedad son:

1. Utilización de medicamentos para acortar la duración de la enfermedad con el objetivo de aumentar  $\alpha$ .
2. Reducir los contactos entre personas susceptibles e infectadas mediante medidas de aislamiento con el objetivo de reducir  $\beta$ .
3. Reducir la transmisibilidad mediante medidas higiénicas con el objetivo de reducir  $\beta$ .
4. Reducir la cantidad de personas susceptibles mediante el uso de vacunas con el objetivo de reducir  $N$  y por tanto reducir también  $S$ , dado que solo las personas no vacunadas participarán en la infección.

### 2.1.2. Limitaciones del modelo

El modelo SIR, al igual que otros muchos modelos matemáticos, simplifica mucho la realidad, entre algunas de estas simplificaciones, las más destacables son:

- No tiene en cuenta las conexiones entre la población. Asume que todas las personas tienen la misma cantidad de contactos y que cualquier persona puede encontrarse con otra de la población.
- No tiene en cuenta las distintas características individuales de cada persona (sexo, edad, otras enfermedades..) que repercuten directamente en los valores de  $\beta$  y  $\alpha$ . Por ejemplo, en el caso del Covid-19, el virus afecta de manera distinta a la población anciana.
- No tiene en cuenta las re-infecciones.
- Asume una población de tamaño fijo sin tener en cuenta los posibles nacimientos y muertes por otras causas en ese periodo de tiempo. Esto implica que no es apto para estudiar la evolución de infecciones que duran muchos años.

## 2.2. Grafos Aleatorios y redes sociales

Una de las limitaciones del modelo SIR es no tener en cuenta las conexiones entre las personas de la población. Este problema puede ser abordado mediante la utilización de grafos. Para poder representar la conexión entre personas lo más real posible a través de un grafo hay que tener en cuenta muchos factores a la hora de crearlo. Varios científicos han abordado este problema dando una variedad de modelos. Aquí nos centraremos en el modelo Barabási-Albert [1].

### 2.2.1. Definiciones básicas

**Definición 2.3.** Un grafo es una estructura formada por un conjunto de objetos también llamados nodos que se relacionan entre sí formando enlaces. Matemáticamente un grafo  $G$  es un par ordenado  $G = (V, E)$  donde:

- $V = v_1, \dots, v_n$  es un conjunto de elementos llamados nodos o vértices.
- $E \subseteq V \times V$  es un conjunto de pares de nodos cuyos elementos son llamados enlaces o aristas.

**Definición 2.4.** Un grafo no dirigido se define como un grafo donde si el par  $(v_1, v_2) \in V$ , entonces  $(v_2, v_1) \in V$ .

**Definición 2.5.** El orden de un grafo se define como número total de vértices que posee y se expresa como  $|V|$ .

**Definición 2.6.** El grado de un nodo se define como el número de enlaces que posee ese nodo, formalmente, sea  $G = (V, E)$ ,  $v \in V$ , entonces  $|v| = |N_v| = |\{u : (v, u) \in E\}|$  donde  $N_v = \{u : (v, u) \in E\}$  se define como el conjunto de vecinos del nodo  $v$ .

**Definición 2.7.** Un bucle se define como el enlace que conecta un nodo consigo mismo, es decir, sea  $G = (V, E)$ ,  $v \in V$ , entonces  $(v_1, v_2) \in E$  es un bucle  $\iff v_1 = v_2$ .

### 2.2.2. Creación del grafo: Modelo Barabási–Albert

Una vez entendidas las definiciones básicas de un grafo, se nos plantea el problema de crear un grafo que sea capaz de representar las relaciones sociales en la población. Representar exactamente las relaciones entre un grupo de personas es imposible. Y siendo siempre un ejemplo específico, tampoco sería muy útil. Normalmente se prefiere trabajar con grafos aleatorios que representen las características estadísticas del conjunto de relaciones sociales. El modelo Erdős–Rényi [3] fue uno de los primeros modelos de grafo aleatorio que se propuso. Sin embargo, los grafos generados siguiendo este modelo se alejan bastante de las redes reales puesto que no tienen en cuenta la distribución del número de vecinos que encontramos en las redes sociales, es decir, relativamente pocas personas tienen muchas conexiones y hay una “cola larga” de personas con pocas conexiones. Estas características se tienen en cuenta en el modelo propuesto en 1999 por Albert-László Barabási y Réka Albert [1].

El algoritmo de Barabási–Albert genera grafos aleatorios que poseen una distribución de vecino “libre de escala” (*scale-free*), es decir, redes que poseen unos pocos nodos con muchas conexiones y una cola larga de nodos con pocas conexiones. En concreto, los autores demostraron que independientemente del sistema y las identidades que forman parte del grafo, la probabilidad  $P(k)$  de que un nodo en una red interactúe con otros  $k$  nodos decae con una *power-law*, siguiendo  $P(k) \sim k^{-\gamma}$  donde  $\gamma > 1$  es un exponente que determina que tan “ancha” es la cola larga de personas con pocas conexiones y que depende específicamente de la red.

El algoritmo que sigue este modelo consiste en lo siguiente:

1. Partimos de una red aleatoria (generada por el modelo Erdős–Rényi [3] por ejemplo) que posee un número pequeño ( $m_0$ ) de nodos con  $M_0$  enlaces.
2. Para cada tiempo  $t$ , añadimos un nuevo nodo  $n$  a la red, este nodo tendrá  $m(\leq m_0)$  enlaces con diferentes nodos que estaban previamente en la red. La probabilidad  $P$  de que el nuevo nodo esté conectado con un nodo  $i$  de la red depende del grado (número de enlaces) que el nodo  $i$  posea, es decir:

$$P(n \rightarrow i) = \frac{k_i}{\sum_j k_j}$$

donde  $k_i$  representa el grado del nodo  $i$ . Según esta fórmula, los nodos que poseen más enlaces tienen más posibilidades de ser elegidos que aquellos que poseen pocos enlaces.

3. Después de  $N$  pasos, el algoritmo conduce a una red aleatoria con  $N + m_0$  nodos y  $mN + M_0$  enlaces. Esta red evoluciona a un estado libre de escala donde la probabilidad de que un nodo tenga  $k$  enlaces sigue una *power-law* con  $P(k) \sim k^{-\gamma}$ , donde  $\gamma$  depende del parámetro  $m$  y  $P(k)$  es independiente del tiempo (y por lo tanto independiente del tamaño de la red) indicando que a pesar de su continuo crecimiento la red se organiza a sí misma en un estado estacionario libre de escala.

La diferencia entre este algoritmo y los anteriores a él recae en dos características que poseen las redes reales y que no se habían tenido en cuenta previamente:

- El crecimiento de la red: Añadiendo constantemente nuevos nodos a la red.
- La conexión preferencial: Los nodos que poseen más conexiones tienen mayor probabilidad de crear conexiones nuevas.

Ambas características son imprescindibles y teniéndolas en cuenta, las redes generadas por este algoritmo se ajustan mucho mejor a redes reales. Algunos ejemplos son: internet, la world wide web (www), redes de citas y algunas redes sociales.

### 2.3. Difusión de un virus en un grafo

En [7] Yang Wang *et al.* nos presentan un modelo que nos permite extraer información de la difusión de un virus en un grafo independientemente de la conectividad o topología del grafo.

Asumimos una red conectada  $G = (N, E)$  donde  $N$  es el número de nodos en la red y  $E$  es el conjunto de enlaces. Sea  $\beta$  la tasa de contagio con la que un nodo es infectado por un vecino,  $\delta$  la tasa de recuperación de un nodo infectado y  $t$  un instante de tiempo fijado.

El modelo asume un tiempo discreto, durante cada intervalo de tiempo un nodo infectado  $i$  intentará infectar a sus vecinos con probabilidad  $\beta$ . Al mismo tiempo  $i$  podría curarse con probabilidad  $\delta$ . Denotamos  $p_{i,t}$  a la probabilidad de que un nodo  $i$  esté infectado en tiempo  $t$ .

**Definición 2.8.** Definimos  $C_{k,t}$  como la probabilidad de que un nodo que tiene  $k$  enlaces no se infecte en tiempo  $t$  como

$$C_{k,t} = \prod_{j:\text{vecino de } i} (1 - \beta * p_{j,t-1})$$

En este modelo, un nodo  $i$  está sano en tiempo  $t$  si:

1.  $i$  estaba sano antes de  $t$  y no ha sido infectado.
2.  $i$  estaba infectado en  $t - 1$ , se curó en  $t$  y no ha sido infectado en  $t$ .
3.  $i$  estaba en tiempo  $t - 1$  infectado pero se curó inmediatamente.

La probabilidad de que un nodo  $i$  esté sano a tiempo  $t$  se define como:

$$(2.6) \quad 1 - p_{i,t} = (1 - p_{i,t-1})C_{k,t} + \delta p_{i,t-1}C_{k,t} + \frac{1}{2}\delta p_{i,t-1}(1 - C_{k,t})$$

Donde en el último termino de la derecha asumimos que la probabilidad de que el nodo  $i$  se cure antes de la infección de uno de sus vecinos es del 50%.

Dada la topología de una red y los valores  $\beta$  y  $\delta$  podemos resolver la ecuación (2.6) y obtener la evolución temporal de la población infectada  $\nu_t = \sum_{i=1}^N p_{i,t}$ .

Utilizando este modelo [7] llegamos al Teorema 2.9.

**Teorema 2.9.** *Cuando una epidemia desaparece, debe cumplirse que  $\frac{\beta}{\delta} < \frac{1}{\lambda_{i,A}}$  donde  $\beta$  es el ratio de infección,  $\delta$  el ratio de recuperación y  $\lambda_{i,A}$  Es el mayor valor propio de la matriz de adyacencia  $A$ .*

Para los detalles acerca de la demostración consultar [7]. Este teorema nos indica que cuanto más pequeño sea el mayor valor propio de la matriz de adyacencia más nos acercamos al umbral en el que la epidemia terminará desapareciendo. Como consecuencia de este teorema, una estrategia óptima para combatir la epidemia podría ser vacunar los nodos que posean más conexiones para reducir el mayor valor propio de la matriz de adyacencia.

## 2.4. Programación Dinámica

En 1952, Richard Bellman presentó en [2] una técnica matemática e informática que recibió el nombre de Programación Dinámica. Esta técnica se utiliza para resolver problemas complejos que requieren una serie de pasos secuenciales para alcanzar el resultado deseado. En [5] se pueden encontrar varios ejemplos de aplicaciones de esta técnica, sin embargo, destaca especialmente en problemas de optimización.

El algoritmo que utiliza esta técnica consiste en dividir el problema inicial en subproblemas y guardar los resultados calculados en una tabla con el objetivo de poder reutilizarlos más adelante sin tener que recalcular operaciones realizadas previamente ahorrando así tiempo de ejecución. Es decir, se trata de mejorar el tiempo de ejecución a cambio de ceder memoria.

### 2.4.1. El Problema de Decisión en Múltiples Pasos

Uno de los problemas que aborda esta técnica es el “Problema de Decisión en Múltiples Pasos”. Matemáticamente podemos expresarlo como:

Para cada tiempo  $t = 0, 1, 2, \dots$  elegido según nos convenga, observamos un proceso que tiene asociado un estado  $u = (t, x(t))$ , donde  $x(t)$  es el estado del proceso a tiempo  $t$ . En cada momento temporal debemos tomar una decisión del conjunto de decisiones posibles  $D(u)$  asociadas al estado  $u$ . La elección de una decisión  $d$  genera un estado nuevo a través de una transformación  $T$  tal que  $v = (t + 1, \tilde{x}) = T(u, d)$  y que tiene asociado un costo  $c(u, d)$ . Definimos una *política* como una función que para cada estado  $u$  asigna una decisión específica dentro de las posibles decisiones. Nuestro objetivo consiste en hallar una política óptima de manera que la suma de los costos de las sucesivas decisiones sea mínima. Si llamamos  $p$  a nuestra política tenemos que partiendo del estado  $u_0$ :

$$\begin{aligned}u_1 &= T(u_0, p(u_0)) \\u_2 &= T(u_1, p(u_1)) \\u_3 &= T(u_2, p(u_2)) \\u_n &= T(u_{n-1}, p(u_{n-1}))\end{aligned}$$

Por lo tanto nuestro problema se reduce a encontrar una política óptima, pero ¿cómo la encontramos? En 1952 Richard Bellman publicó el Principio de optimalidad que nos permite encontrar la política óptima que buscamos y la función de costo asociada a esa política.

### 2.4.2. Principio de optimalidad

Una clase importante de problemas de optimización tiene la siguiente característica: una política óptima tiene la propiedad de que cualquiera que sea el estado y las decisiones iniciales que hayamos tomado, las próximas decisiones constituyen una política óptima con respecto al estado resultante de la primera decisión. Es decir, supongamos que hemos obtenido el camino  $u_0 \rightarrow u_1 \rightarrow u_2 \rightarrow u_3 \rightarrow \dots \rightarrow u_n$  partiendo del estado  $u_0$  y siguiendo una política óptima, entonces,  $u_3 \rightarrow \dots \rightarrow u_n$  es también un camino óptimo a partir del nodo  $u_3$ . Por lo tanto, las colas generadas por una política óptima son también óptimas.

Si llamamos  $f$  a la función de costo correspondiente a la política óptima elegida, podemos expresar el principio de optimalidad matemáticamente como:

$$(2.7) \quad f(u_t) = c(u_t, p(u_t)) + f(T(u_t, p(u_t)))$$

El Principio de optimalidad nos permite establecer la siguiente ecuación funcional a partir de (2.7):

$$(2.8) \quad f(u) = \min_{u \in D(u)} c(u, p(u)) + f(T(u, p(u)))$$

A partir de esta ecuación podemos encontrar  $f$  y  $p$  de manera recursiva empezando por el final, es decir, de atrás hacia adelante. Asignamos a  $f(u_n) = 0$  y calculamos:

$$\begin{aligned} f(u_{n-1}) &= \min_{u_{n-1} \in D(u_{n-1})} c(u_{n-1}, p(u_{n-1})) + f(T(u_{n-1}, p(u_{n-1}))) \\ f(u_{n-1}) &= \min_{u_{n-1} \in D(u_{n-1})} c(u_{n-1}, p(u_{n-1})) + f(u_n) \\ f(u_{n-1}) &= \min_{u_{n-1} \in D(u_{n-1})} c(u_{n-1}, p(u_{n-1})) + 0 \end{aligned}$$

Supongamos que el estado  $u'_{n-1} \in D(u_{n-1})$  es el que minimiza (2.7), entonces,

$$p(u'_{n-1}) = u_n$$

$$\begin{aligned} f(u_{n-2}) &= \min_{u_{n-2} \in D(u_{n-2})} c(u_{n-2}, p(u_{n-2})) + f(T(u_{n-2}, p(u_{n-2}))) \\ f(u_{n-2}) &= \min_{u_{n-2} \in D(u_{n-2})} c(u_{n-2}, p(u_{n-2})) + f(u_{n-1}) \end{aligned}$$

Supongamos que el estado  $u'_{n-2} \in D(u_{n-2})$  es el que minimiza (2.7), entonces,

$$p(u'_{n-2}) = u'_{n-1}$$

...

$$\begin{aligned} f(u_0) &= \min_{u_0 \in D(u_0)} c(u_0, p(u_0)) + f(T(u_0, p(u_0))) \\ f(u_0) &= \min_{u_0 \in D(u_0)} c(u_0, p(u_0)) + f(u_1) \end{aligned}$$

Supongamos que el estado  $u'_0 \in D(u_0)$  es el que minimiza (2.7), entonces,

$$p(u'_0) = u'_1$$

El problema de la vacunación posee las características necesarias para aplicar Programación Dinámica; la parte de la política óptima que empieza en tiempo  $t$  es óptima para la situación epidemiológica en tiempo  $t$ . Por esto, utilizar el algoritmo de Programación Dinámica es viable para optimizar la política de vacunación, y es el que usaremos en el próximo capítulo.

## 2.5. Estrategias de vacunación

Todos los indicadores y los estudios epidemiológicos coinciden en que la medida más eficaz para combatir una epidemia es la utilización de vacunas para reducir la cantidad de personas susceptibles en la población. En el caso ideal tendríamos vacunas para toda la población y podríamos vacunar a todos a la vez. Sin embargo, la realidad es muy distinta, por lo general la cantidad de vacunas será menor que el total de personas, por lo que uno de los problemas que se nos plantea es como repartir las vacunas de manera que minimicemos el impacto de la enfermedad.

En [6] se estudia algunas estrategias de vacunación para la Covid-19 teniendo en cuenta la existencia de un grupo vulnerable (los mayores) y la variabilidad de las interacciones y círculos sociales. Dada la relación muy estrecha entre [6] y este trabajo, analizamos aquí detenidamente los resultados que se consiguieron ahí.

Algunas de las herramientas utilizadas para la realización del estudio fueron:

1. La creación de grafos no dirigidos aleatorios (utilizando el modelo Barabási–Albert 2.2.2), que representan la población: cada nodo representa un individuo y cada individuo puede pertenecer a la población joven (80%) o a la población mayor (20%). Los enlaces del grafo representan las interacciones entre los individuos.
2. La difusión de la enfermedad en el grafo siguiendo un modelo de evolución estocástico en el que para cada tiempo  $t$ , los individuos de la población transitan entre cuatro estados (susceptibles, infectados, recuperados y víctimas) siguiendo:
  - Para cada individuo susceptible se escoge uno de sus vecinos de manera aleatoria, si el vecino escogido está infectado, la enfermedad se transmite con una probabilidad  $\beta$  a la persona susceptible.
  - Cada infectado se recupera con una probabilidad  $\phi$ .
  - Cada infectado se convierte en víctima con una probabilidad  $\rho$ .

Las probabilidades  $\beta$ ,  $\phi$  y  $\rho$  se obtienen a partir de los datos epidemiológicos proporcionados por el Ministerio de Sanidad español <sup>1</sup> y el modelo SIR.

<sup>1</sup>Los datos epidemiológicos están derivados de los reportes diarios del Ministerio de Sanidad español publicado en su página web <https://www.mscbs.gob.es/profesionales/saludPublica/ccayes/alertasActual/nCov/home.htm>

Entre los factores a tener en cuenta encontramos:

1. La cantidad de vacunas disponibles.
2. La cantidad de vacunas que asignaremos al grupo de los mayores.
3. La densidad de relaciones entre los jóvenes y mayores. Para ello se han tenido en cuenta 4 posibles casos:
  - Pocas interacciones entre mayores y pocas interacciones entre jóvenes y mayores.
  - Muchas interacciones entre mayores y pocas interacciones entre jóvenes y mayores.
  - Pocas interacciones entre mayores y muchas interacciones entre jóvenes y mayores.
  - Muchas interacciones entre mayores y muchas interacciones entre jóvenes y mayores.
4. La manera de vacunar a los jóvenes, para ello disponemos de dos métodos:
  - Vacunar a los jóvenes que tienen más vecinos en general.
  - Vacunar a los jóvenes que tienen más contacto con la población vulnerable, es decir los mayores.

Veamos ahora algunos de los resultados. Dividiremos los resultados bajo dos hipótesis.

### **Caso 1: Suponiendo capacidad de vacunación infinita**

En este caso tenemos una cantidad de vacunas limitadas que se aplicará a la población susceptible antes del comienzo de la epidemia, es decir, en  $t = 0$ . En la práctica, este modelo corresponde a casos en que se vacuna antes del comienzo de la epidemia (p. ej. la campaña anual contra la gripe) o se vacuna en un tiempo muy corto comparado con la evolución de la epidemia.

Las conclusiones obtenidas variando los factores previamente mencionados son:

1. Mantener la distancia social durante las primeras fases de vacunación es la clave para obtener los mejores resultados.
2. La manera de vacunar a los jóvenes dependerá de la cantidad de vacunas disponibles y la densidad de relaciones entre los jóvenes y los mayores.
  - Suponiendo una cantidad de vacunas suficiente para vacunar un 5% de la población total: La mejor estrategia sería mantener aislada a la población más vulnerable y asignar las vacunas a los jóvenes más conectados con la población mayor. A corto plazo una buena estrategia sería asignar todas las vacunas a los mayores, sin embargo, a largo plazo resulta en una rápida expansión de la epidemia.



- Suponiendo una cantidad de vacunas suficiente para vacunar un 10% de la población total: El número de víctimas en todos los casos disminuye. La mejor estrategia sería mantener a la población más vulnerable aislada y asignar la mayoría de las vacunas a los jóvenes que tienen más conexiones en general. En el caso en el que no sea posible aislar correctamente a la población más vulnerable, la mejor estrategia es asignar más vacunas a la población mayor independientemente del modo de vacunación de los jóvenes.
  - Suponiendo una cantidad de vacunas suficiente para vacunar un 20% de la población total: El número de víctimas en todos los casos disminuye. Al tener un número de dosis más alto, la estrategia más óptima sería en todos los casos mantener la población más vulnerable aislada. En el caso en el que no sea posible aislar correctamente a la población más vulnerable, la mejor estrategia es asignar más vacunas a la población mayor independientemente del modo de vacunación de los jóvenes.
3. En todos los casos encontramos un fenómeno peculiar llamado *inversión de riesgo*: El número de víctimas decrece conforme la cantidad de vacunas administradas a los mayores aumenta. Sin embargo, si asignamos todas las vacunas a la población más vulnerable, dejamos a la población joven, que está mucho más conectada, totalmente expuesta, lo que conduce a una rápida expansión de la epidemia. Es decir, si no podemos aislar a la población vulnerable correctamente terminará causando más muertes a largo plazo. Por otra parte, si asignamos una mayor cantidad de vacunas a los jóvenes frenará la expansión de la epidemia pero la cantidad de víctimas totales (y sobretodo dentro de la población vulnerable) aumenta. Esto sugiere que en el caso de tener la vacuna disponible por lotes la mejor estrategia sería llegar a un balance entre vacunar a los jóvenes para frenar la transmisión de la enfermedad y a los mayores para protegerlos.

### Caso 2: Suponiendo capacidad de vacunación limitada

En este caso tenemos una cantidad limitada de vacunas y una capacidad diaria de vacunación limitada, es decir, podemos vacunar a  $p$  personas en un día. Al igual que antes uno de los factores determinantes es la cantidad de vacunas que asignaremos al grupo más vulnerable, pero en este caso no será un valor fijo, sino que irá cambiando cada día. Ver todas las posibilidades para cada día es una tarea inviable, por lo que se ha utilizado un algoritmo genético [6]. Los resultados se han obtenido usando 500 generaciones de 800 genes cada una. Los resultados son los mejores genes de varias generaciones.

En esta parte del estudio se ha fijado una cantidad de vacunas totales suficientes para vacunar al 20% de la población, este porcentaje de vacunas se irá administrando a lo largo de los días, vacunando  $p$  personas cada día.

Teniendo en cuenta ambos modos de vacunación y la densidad de relaciones entre los jóvenes y mayores obtenemos los siguientes resultados:

1. Independientemente del método utilizado para vacunar a los jóvenes, todos los resultados apuntan a asignar una cantidad alta de vacunas a la población mayor en los primeros días seguida de un incremento de las vacunas asignadas a los jóvenes conforme el número de víctimas empieza a incrementar.
2. Vacunando a los jóvenes que tienen más conexiones en general: Si la población mayor está correctamente aislada, después de vacunar a la población más vulnerable durante los primeros días, el foco se mueve a la población joven para detener la expansión de la epidemia. En el caso de que la población mayor no esté correctamente aislada la estrategia óptima consiste en alternar el foco de vacunación entre ambas poblaciones.
3. Vacunando a los jóvenes más conectados con mayores: Si la población mayor está correctamente aislada, los primeros días la vacunación se centra en la población más vulnerable durante un periodo de tiempo más amplio que en el caso anterior y posteriormente cambia el foco de vacunación a la población joven paulatinamente. En el caso de que la población mayor no esté correctamente aislada, el foco de vacunación cambia antes a la población joven y oscila entre ambas poblaciones.

## CAPÍTULO 3

# Modelo de vacunación

---

En este capítulo propondremos un modelo de vacunación con el objetivo de minimizar el número de víctimas causado por la epidemia. En la primera parte enunciaremos el problema que queremos resolver. En la segunda sección presentaremos los grafos que representarán las relaciones sociales de la población y la evolución de la epidemia a través de sus nodos. En la tercera sección explicaremos el cálculo de los parámetros epidemiológicos utilizados. En la sección 3.4 explicaremos la optimización realizada utilizando Programación Dinámica. Finalmente, en la sección 3.5 detallaremos la implementación realizada para obtener los resultados que se presentan en el capítulo siguiente.

### 3.1. El problema

Se nos plantea el siguiente problema: Una enfermedad infecciosa de transmisión directa ha empezado a difundirse a través de una población en su mayoría susceptible salvo unos pocos infectados causando una epidemia. Disponemos de una cantidad de vacunas totales, pero o bien esta vacuna llegó en lotes de  $p$  dosis diarias, o bien nuestro sistema sanitario tan solo es capaz de vacunar a  $p$  personas cada día. Nuestro objetivo consiste en asignar correctamente el porcentaje de vacunas que recibirán la población joven y mayor cada semana con el objetivo de minimizar las víctimas totales causadas por la epidemia en  $T$  días.

Para resolver el problema planteado hemos estudiado la evolución de la enfermedad en cuatro grafos con diferente topología y creados según el modelo Barabási–Albert 2.2.2. Al tratarse de una enfermedad infecciosa necesitamos obtener los parámetros epidemiológicos de la misma, para ello hemos utilizado datos proporcionados por el Ministerio de Sanidad español y el modelo SIR 2.1. Una vez sentadas las bases para resolver el problema, hemos utilizado Programación Dinámica para tomar las decisiones que optimizan el problema.

## 3.2. Grafos

Uno de los factores más importantes a la hora de estudiar la evolución de una epidemia es la topología de la red que forman los individuos de la población. Hemos generado varios grafos con el objetivo de obtener resultados en función de las diferentes posibilidades que podemos encontrarnos en la realidad.

### 3.2.1. Creación de grafos

Sea  $G = (V, E)$  un grafo con  $V = \{1, \dots, N\}$  y  $E \subseteq V \times V$  donde cada nodo  $v \in V$  representa a un individuo de la población y cada enlace  $(u, v) \in E$  representa las interacciones sociales entre individuos.

Sea  $N = |V|$ , denotamos  $d(u) = |\{v : (u, v) \in E\}|$  como el grado del nodo  $u$ . Definimos  $\bar{d} = \frac{\sum_k d(k)}{N}$  como el número medio de vecinos de los nodos del grafo.

Nuestro objetivo es poder representar las interacciones sociales de la población lo más real posible, con este objetivo en mente hemos decidido utilizar el modelo Barabási–Albert a la hora de crear los grafos que utilizaremos para el estudio. Hemos visto en 2.2.2 que el algoritmo para crear el grafo requiere  $N$  pasos y en cada paso se añade un nuevo nodo que se enlazará a otros  $m$  nodos que se encontraban previamente en la red con *preferential attachment*, es decir, los nodos que poseen más enlaces tienen más posibilidades de ser elegidos que aquellos que poseen pocos enlaces. El algoritmo utilizado para generar el grafo se encuentra en la figura 3.1.

```
(V, E) = crear_grafo(N, m)
1.  V ← {1}
2.  E ← ∅
3.  d ← [0 | k = 1, ..., t]
4.  for k = 2, ..., N do
5.    V ← V ∪ {k}
6.    for c = 1, ..., min(grado(V), m) do
7.      u ← prefRnd(d[1], ..., d[k-1])
8.      E ← E ∪ {(k, u), (u, k)}
9.      d[u] ← d[u] + 1
10.   od
11.   d[k] ← c
12. od
13. return (V, E)
```

Figura 3.1: Algoritmo que genera un grafo utilizando el modelo Barabási–Albert, donde  $N$  representa el número de nodos totales que se desea añadir al grafo y  $m$  determina la conectividad del grafo. La función  $\text{grado}(V)$  devuelve el número de nodos actuales en el grafo. La función  $\text{prefRnd}$  elige un elemento de  $V$  usando *preferential attachment* y descartando los elegidos previamente. La lista  $[d[1], \dots, d[k-1]]$  se utiliza para calcular las probabilidades de conexión correspondientes a cada elemento.

Una de las limitaciones que posee el modelo SIR 2.1 es no tener en cuenta las distintas características que posee cada individuo que repercuten directamente en los parámetros epidemiológicos. En el caso del Covid-19 uno de los factores determinantes se encuentra en la edad de los individuos, pues la mortalidad es mayor en la población envejecida.

A la hora de realizar nuestro estudio hemos dividido a la población total en dos grupos: el grupo vulnerable (los mayores) y el grupo de los jóvenes. Puesto que las interacciones sociales en cada grupo varían es necesario crear dos grafos:

- Un grafo que represente a los jóvenes  $G_y = (V_y, E_y)$  creado con los parámetros  $N_y$  y  $m_y$  que generará un grafo con exponente  $\gamma_y$  y media de interacciones  $\bar{d}_y$ .
- Un grafo que represente a los mayores  $G_e = (V_e, E_e)$  creado con los parámetros  $N_e$  y  $m_e$  que generará un grafo con exponente  $\gamma_e$  y media de interacciones  $\bar{d}_e$ .

El modelo completo está compuesto de ambos grafos interconectados mediante *preferential attachment*, es decir, los nodos que poseen más enlaces tienen más posibilidades de ser elegidos que aquellos que poseen pocos enlaces.

Para conectar ambos grafos primero se debe de seleccionar un número  $\bar{d}_{ey}$  que representa la media de interacciones sociales con jóvenes que tiene una persona mayor. Entonces, se generan  $N_e \cdot \bar{d}_{ey}$  enlaces entre ambos grafos seleccionando una persona mayor al azar, una persona joven con *preferential attachment* y conectándolas. La razón por la que utilizamos *preferential attachment* a la hora de conectar ambos grupos es debido a que algunas personas jóvenes (por ejemplo, trabajadores sociales, personales de limpieza, repartidores, etc.) tienen contacto con muchos mayores mientras que la mayoría de jóvenes tienen poco contacto con los mayores. El algoritmo para la unión de ambos grafos se encuentra en la figura 3.2.

```

(V, E) = unir_grafos( $G_y = (V_y, E_y)$ ,  $G_e = (V_e, E_e)$ ,  $\bar{d}_{ey}$ )
1.  V ←  $V_y \cup V_e$ 
2.  E ←  $E_y \cup E_e$ 
3.  N ←  $|V_e| \cdot \bar{d}_{ey}$ 
4.  d ←  $[1|n \in V_y]$ 
5.  for  $k \leftarrow 1, \dots, N$  do
6.    u ← unif( $V_e$ )
7.    v ← prefRnd( $V_y$ , d)
8.    d[u] ← d[u] + 1
9.    d[v] ← d[v] + 1
10.  E ←  $E \cup \{(u,v), (v,u)\}$ 
11. od
12. return(V, E)

```

Figura 3.2: Algoritmo que une dos grafos usando *preferential attachment*. Asume que los identificadores de los nodos de la población joven y mayor son diferentes. Asume que  $V_y$  y  $V_e$  son listas ordenadas que mantienen la correspondencia entre nodos y valores en  $d$ . La función **unif**(V) devuelve un elemento de V elegido al azar con probabilidad uniforme.

### 3.2.2. Grafos utilizados

Para las simulaciones realizadas que se encuentran en el capítulo 4 se han utilizado grafos con  $N = 5000$  nodos, el 80 % de la población es joven ( $N_y = 4000$ ) y el 20 % mayor ( $N_e = 1000$ ).

El grafo de interacciones sociales de los jóvenes es fijo y se ha generado tomando  $N = 4000$  y  $m = 20$ , como resultado hemos obtenido un grafo con  $\gamma_y = 1,77$  y  $\bar{d}_y = 16,9$ . Hemos creado un total de cuatro grafos dando distintos valores a las interacciones sociales entre la población mayor  $\bar{d}_e$  y a las interacciones sociales entre jóvenes y mayores  $\bar{d}_{ey}$ . Estos datos están identificados en la tabla 3.1.

Tenemos cuatro tipos de grafo según los distintos parámetros:

- Grafos del tipo **A**: caracterizado por tener escasas interacciones entre la población mayor, es decir, cada persona mayor interactúa con pocas personas mayores.
- Grafos del tipo **B**: caracterizado por tener muchas interacciones entre la población mayor, es decir, cada persona mayor interactúa mucho con otras personas mayores.
- Grafos del tipo  $\alpha$ : caracterizado por tener pocas interacciones entre la población joven y mayor, es decir, cada persona mayor interactúa poco con la población joven.
- Grafos del tipo  $\beta$ : caracterizados por tener muchas interacciones entre la población joven y mayor, es decir, cada persona mayor interactúa mucho con la población joven.

		$\bar{d}_{ey}$	
		0,5	3,0
$\bar{d}_e$		$\alpha$	$\beta$
1,0	A	$A\alpha$	$A\beta$
7,0	B	$B\alpha$	$B\beta$

Tabla 3.1: Parámetros y código utilizado para generar cuatro grafos variando las interacciones sociales entre la población mayor  $\bar{d}_e$  y a las interacciones sociales entre jóvenes y mayores  $\bar{d}_{ey}$ .

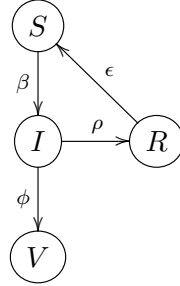
### 3.2.3. Difusión de la enfermedad en el grafo

Ahora que ya tenemos generados los grafos necesarios para nuestro estudio, pasaremos a explicar la evolución de la epidemia en nuestro modelo a través de los nodos y enlaces del grafo.

Sea  $G = (V, E)$  un grafo generado como se ha descrito previamente. Sea  $v \in V$  un nodo del grafo que representa a un individuo de la población, dado un tiempo  $t$  el individuo puede estar en uno de los cuatro estados: susceptible (S), infectado (I), recuperado (R) o victima (V).

Las personas transitan entre los cuatro estados dependiendo de sus interacciones sociales con otros individuos y su probabilidad de recuperación siguiendo el siguiente diagrama donde asumimos que no hay re-infecciones en el tiempo de estudio.

(3.1)



La única transición que depende de la interacción social entre los nodos del grafo es  $S \rightarrow I$  donde un individuo susceptible pasa a estar infectado. La probabilidad de que ocurra esta transición depende del parámetro  $\beta$  y del número de vecinos infectados de un nodo dado. El modelo asume que cada individuo interactúa uniformemente con sus vecinos, esta es la asunción más fuerte que hacemos en nuestro modelo. En realidad, cada nodo interactúa con algunos nodos más frecuentemente que con otros, un modelo que puede servir como solución a este problema se encuentra en [4], donde utilizan un grafo no dirigido cuyos enlaces tienen un peso que representa la frecuencia con la que dos individuos interactúan. Otra posible solución sería utilizar un multi-grafo, donde el número de enlaces entre los nodos represente la densidad de interacciones entre ellos.

Por lo tanto, dado un nodo  $u$  en el  $estado_{t-1}[u] = S$  en el tiempo  $t - 1$ . Cada vecino  $v \in N_u$  que se encuentre en el  $estado_{t-1}[v] = I$  infectará con probabilidad  $\beta$  al nodo  $u$  de forma que pasará al  $estado_t[u] = I$  y con probabilidad  $1 - \beta$  el nodo  $u$  permanecerá en el  $estado_t[u] = S$ .

Los otros estados transitan independientemente de las interacciones entre los nodos de acuerdo con el esquema de la tabla 3.2.

$estado_{t-1}[u]$	$estado_t[u]$	con probabilidad
$S$	$I$	$1 - (1 - \beta)^{N_I(u)}$
	$S$	$(1 - \beta)^{N_I(u)}$
$I$	$R$	$\rho$
	$V$	$\phi$
	$I$	$1 - \phi - \rho$
$R$	$S$	$\epsilon$
	$R$	$1 - \epsilon$

Tabla 3.2: Probabilidades de transición entre los estados  $S, I, R, V$ . La transición de  $S$  a  $I$  depende del contacto de un nodo con sus vecinos donde  $N_I(u)$  representa los vecinos del nodo  $u$ . El resto de transiciones dependen únicamente de los parámetros epidemiológicos.

Los nodos no vacunados son los únicos que transitan entre estados, un indicador independiente nos indica si el nodo está vacunado o no.

Solo los nodos susceptibles pueden ser vacunados. Cuando un nodo es vacunado permanece en el estado  $R$ , es decir, asumimos que la vacuna es 100% efectiva y aporta inmediatamente desde el momento de aplicación anticuerpos de por vida.

### 3.3. Parámetros epidemiológicos

Ahora que sabemos como transita un individuo entre los distintos estados queda por determinar los parámetros epidemiológicos  $\beta, \phi, \rho$  y  $\epsilon$ . Para ello hemos utilizado una modificación del modelo SIR 2.1 como una primera aproximación:

- Hemos dividido la población recuperada ( $R$ ) del modelo original SIR, que no consideraba víctimas, en población recuperada ( $R$ ) y víctimas ( $V$ ).
- Asumimos que población infectada está aislada para su estudio, lo que se traduce en partir de las condiciones iniciales  $S(0) = N, I(0) = 0, R(0) = 0$  y  $V(0) = 0$ .

Además, hemos asumido que la inmunidad tras superar la enfermedad es permanente en el tiempo de estudio considerado en la simulación, por lo tanto  $\epsilon_y = \epsilon_e = 0$ .

La evolución de la población aislada está descrita por:

$$(3.2) \quad \frac{dI}{dt} = -\rho I - \phi I$$

$$(3.3) \quad \frac{dR}{dt} = \rho I$$

$$(3.4) \quad \frac{dV}{dt} = \phi I$$

La ecuación (3.2) se trata de una EDO de primer orden de variables separadas, por lo tanto,

$$\begin{aligned} \frac{dI}{dt} = -\rho I - \phi I &\implies \frac{dI}{dt} = I(-\rho - \phi) \\ \frac{dI}{I} = (-\rho - \phi)dt &\implies \int \frac{dI}{I} = \int (-\rho - \phi)dt \\ \ln(I) = -(\rho + \phi)t + C &\implies I = e^{-(\rho+\phi)t+C} \end{aligned}$$

Teniendo en cuenta las condiciones iniciales del problema, la solución para  $I(t)$  está dada por:

$$(3.5) \quad I(t) = I_0 e^{-(\rho+\phi)t}$$

donde  $I_0$  es el número inicial de infectados, en nuestro caso  $N$ .

Una persona permanece infectada durante un tiempo medio de  $\frac{1}{\rho+\phi}$  antes de recuperarse o morir. Asumiendo que tomamos los días como unidad de tiempo, y una media de  $D$  días de duración de la enfermedad, independientemente de la edad obtenemos

$$(3.6) \quad D = \frac{1}{\rho + \phi} = \frac{1}{\rho_y + \phi_y} = \frac{1}{\rho_e + \phi_e}$$



y por tanto despejando de (3.6),

$$(3.7) \quad \rho_y + \phi_y = \rho_e + \phi_e = \frac{1}{D}$$

Los valores de los parámetros individuales se determinan usando la letalidad, que es el número de víctimas dividido del número de casos. Sea  $L_y$  y  $L_e$  la letalidad de la población joven y mayor respectivamente. A partir de (3.4) y utilizando (3.5) podemos obtener  $V(t)$

$$\begin{aligned} \frac{dV}{dt} = \phi I &\implies \frac{dV}{dt} = \phi I_0 e^{-(\rho+\phi)t} \implies \\ dV = \phi I_0 e^{-(\rho+\phi)t} dt &\implies \int dV = \int \phi I_0 e^{-(\rho+\phi)t} dt \implies \\ V(t) = \phi I_0 \int_0^t e^{-(\rho+\phi)\tau} d\tau &\implies V(t) = \frac{\phi I_0}{-(\rho+\phi)} \int_0^t -(\rho+\phi) e^{-(\rho+\phi)\tau} d\tau \implies \\ V(t) = \frac{\phi I_0}{-(\rho+\phi)} [e^{-(\rho+\phi)t}]_0^t &\implies V(t) = \frac{\phi I_0}{-(\rho+\phi)} [e^{-(\rho+\phi)t} - 1] \end{aligned}$$

Por lo tanto,

$$(3.8) \quad V(t) = \frac{\phi I_0}{\rho + \phi} [1 - e^{-(\rho+\phi)t}]$$

Para un tiempo  $t$  muy grande en (3.8) se cumple que

$$(3.9) \quad V(t) \approx \frac{\phi I_0}{\rho + \phi}$$

Por lo tanto, utilizando (3.9) podemos expresar la letalidad para la población joven y mayor como,

$$(3.10) \quad L_y = \frac{\phi_y}{\rho_y + \phi_y}$$

$$(3.11) \quad L_e = \frac{\phi_e}{\rho_e + \phi_e}$$

Juntando las ecuaciones (3.10), (3.11) y (3.7) obtenemos

$$(3.12) \quad \phi_y = \frac{L_y}{D}, \quad \phi_e = \frac{L_e}{D}$$

Sustituyendo (3.12) en (3.10) y (3.11) y despejando  $\rho$  obtenemos

$$(3.13) \quad \rho_y = \frac{1}{D}(1 - L_y), \quad \rho_e = \frac{1}{D}(1 - L_e)$$

El valor  $\beta$  se deriva de su relación con el factor unitario de infección  $R_0$ , cuya relación viene dada por

$$(3.14) \quad R_0 = \frac{\beta}{\phi + \rho}$$

Nótese que siguiendo el modelo SIR descrito en 2.1  $\beta = k\tau$ . Por lo tanto, utilizando (3.12) y (3.13) y (3.14) podemos expresar  $R_{0,y}$  y  $R_{0,e}$  como

$$(3.15) \quad R_{0,y} = \frac{\beta_y}{\phi_y + \rho_y} = \frac{\beta_y}{\frac{L_y}{D} + \frac{1-L_y}{D}} = \frac{\beta_y}{\frac{1}{D}}$$

$$(3.16) \quad R_{0,e} = \frac{\beta_e}{\phi_e + \rho_e} = \frac{\beta_e}{\frac{L_e}{D} + \frac{1-L_e}{D}} = \frac{\beta_e}{\frac{1}{D}}$$

Despejando de (3.15) y (3.16) obtenemos

$$\beta_y = \frac{R_{0,y}}{D}, \quad \beta_e = \frac{R_{0,e}}{D}$$

Los valores utilizados en la simulación se encuentran en la tabla 3.3.

Observable		Modelo	
Parámetro	Valor	Parámetro	Valor
$D$	18 días	$\beta_y$	0.1333
$R_{0,y}$	2.4	$\beta_e$	0.1333
$R_{0,e}$	2.4	$\phi_y$	0.0011
$L_y$	0.08	$\phi_e$	0.0111
$L_e$	0.2	$\rho_y$	0.0544
		$\rho_e$	0.044

Tabla 3.3: Valores de los parámetros utilizados en la simulación. Los parámetros observables de las dos primeras columnas están derivados de los datos epidémicos de España durante la primera ola del Covid-19 (antes del 30 de marzo). Los parámetros no observables del modelo que se encuentran en las dos últimas columnas están derivados de los parámetros observables y son los que usaremos en nuestro modelo.

### 3.4. Optimización

En este modelo partimos de un grafo que representa la población y un número pequeño de personas infectadas, seguiremos el mismo algoritmo que [6] para estudiar la evolución de la epidemia, pero nos centraremos únicamente en el caso en el que disponemos de una capacidad de vacunación limitada por cada intervalo de tiempo, por lo que optimizaremos la cantidad de vacunas asignadas a cada población (mayores y jóvenes) en cada instante de tiempo  $t$  designado. Para ello, en lugar de utilizar un algoritmo genético como el que se utilizó en el estudio previamente mencionado [6], haremos uso de Programación Dinámica, dividiendo así una tarea muy compleja en tareas más simples.

Al tratarse de un Problema de Decisión en Múltiples Pasos, basta con definir nuestras variables de estado y la función de costo para hallar la política óptima y el costo asociado a esa política haciendo uso del principio de optimalidad.

### 3.4.1. Formalización

Fijado un tiempo total  $T$ , lo dividimos en varios intervalos temporales según nos convenga  $[t_1, t_2, \dots, t_T]$ .

**Definición 3.1.** Definimos el vector  $e(t)$  como

$$e(t) = (Iy(t), Ie(t), Ve(t))$$

Donde  $Iy(t)$  representa la cantidad de población joven infectada,  $Ie(t)$  la cantidad de población mayor infectada y  $Ve(t)$  la cantidad de población mayor vacunada en tiempo  $t$ .

A primera vista, el vector  $e(t)$  no representaba todo el estado, en cuanto faltan el número de jóvenes vacunados ( $Vy(t)$ ) y el número de recuperados,  $Re(t)$  y  $Ry(t)$ .

El número de jóvenes vacunados no aparece en el estado debido a que no es una variable independiente. El número total de vacunados al tiempo  $t$ ,  $V(t)$ , se puede calcular dado que conocemos la cantidad de vacunas que se administran cada día. El número  $Vy(t)$  por tanto se puede calcular como

$$Vy(t) = V(t) - Ve(t)$$

La situación para  $Re(t)$  y  $Ry(t)$  es diferente. En este caso, estas variables deberían ser parte del estado, pero las eliminamos estimándolas por razones de eficiencia. Si el estado contiene  $e$  variables y cada una puede asumir  $L$  valores, el algoritmo tiene una complejidad  $O(L^e)$ , por lo tanto cada nueva variable supone un aumento exponencial de la complejidad.

Aproximamos  $Ry(t)$  (resp.  $Re(t)$ ) como una función de  $Iy(t)$  (resp.  $Ie(t)$ ). Consideramos  $Ry(t)$  (el cálculo para  $Re(t)$  es análogo).

Si la diferencia  $b_{i+1} - b_i = \Delta t$  no es muy grande y  $\beta$  no es muy grande,  $S(t)$  variará poco, y lo podemos considerar constante. Por lo tanto tenemos en el modelo SIR:

$$(3.17) \quad \frac{DIy}{dt} = \beta_y \frac{S}{N} Iy - \rho_y Iy - \phi_y Iy = \lambda Iy$$

con

$$\lambda = \beta_y \frac{S}{N} - \rho_y - \phi_y$$

La ecuación (3.17) se trata de una EDO de primer orden de variables separadas, que se puede resolver de la siguiente manera,

$$\begin{aligned} \frac{DIy}{dt} &= \lambda Iy \\ \frac{DIy}{Iy} &= \lambda dt \\ \int \frac{DIy}{Iy} &= \int \lambda dt \\ \ln(Iy) &= \lambda t \end{aligned}$$

Por tanto,

$$Iy(t) = I_0 e^{\lambda t}$$

Sustituyendo este resultado en la ecuación de recuperados,

$$\begin{aligned} \frac{dR}{dt} &= \rho I \\ \frac{dRy}{dt} &= \rho_y I_0 e^{\lambda t} \\ Ry(t) &= \int_0^t \rho_y I_0 e^{\lambda \tau} d\tau \\ Ry(t) &= \frac{\rho_y I_0}{\lambda} [e^{\lambda t} - 1] \end{aligned}$$

aproximando

$$e^{\lambda t} \sim 1 + \lambda \Delta t$$

tenemos

$$Ry(t) \sim \rho_y I_0 \Delta t$$

Con esta aproximación  $Ry$  y  $Re$  son funciones de proporcionales a  $\rho_y$  y  $\rho_e$  y se pueden eliminar del estado. Por lo tanto, a partir de  $e(t)$  podemos definir el estado  $u_t = (t, e(t))$ .

**Definición 3.2.** Definimos  $P(u_t) = (p, u_{t+1})$  donde  $p$  el porcentaje de vacunas suministradas a la población mayor en función de  $u_t$ , y  $u_{t+1}$  es el estado al que nos llevará administrar  $p$  vacunas a la población mayor a partir del estado  $u_t$ , es decir, se trata nuestra política puesto que  $u_{t+1} = P(u_t)[1]$ , para simplificar notación, a esta política la llamaremos  $p(u_t)$ , por lo tanto,  $u_{t+1} = P(u_t)[1] = p(u_t)$ .

**Definición 3.3.** Definimos  $f(u_t)$  como la cantidad de victimas en tiempo  $t$  en función del estado  $u_t$ .

**Definición 3.4.** Definimos  $v(u_t, p(u_t)) = v(u_t, u_{t+1})$  como la cantidad de victimas entre los estados  $u_t$  y  $u_{t+1}$ . Esta es nuestra función costo.

Nuestro objetivo consiste en minimizar  $f(u_0)$ , donde  $u_0 = (0, e(0))$  y  $e(0)$  puede tomar los valores de cualquier escenario inicial, el cálculo se realiza para todos los escenarios posibles, pero en nuestro estudio nos centraremos en el caso inicial donde hay un único infectado joven, es decir,  $e(0) = (1, 0, 0)$ . Para minimizar  $f(u_0)$  debemos encontrar los porcentajes  $P(u_0)[0], \dots, P(u_T)[0]$  óptimos de vacunas que debemos suministrar a la población mayor en cada intervalo temporal para reducir el número de victimas totales.

Para el tiempo  $t$ , la función  $f(u_t)$  cumple

$$(3.18) \quad f(u_t) = v(u_t, p(u_t)) + f(u_{t+1})$$

Por lo que para minimizar  $f(u_0)$  basta con encontrar  $p(u_t) = u_{t+1}$  en función de las vacunas asignadas a los mayores en tiempo  $t$  y las variables de estado de  $u_t$  que minimiza  $f(u_t)$  en cada punto  $t \in (0, T)$ .

Al ser una función recursiva que depende del valor siguiente, una manera de encontrar el mínimo consiste en calcular el resultado desde el final, es decir empezando en  $t = T$  y terminando en  $t = 0$ . En  $t = T$  asignamos  $f(u_T) = 0 \forall u_T$ , y en cada instante  $t < T$  calculamos  $p(u_t)$  óptimo en función de los distintos valores que puede tomar  $e(t)$  y  $P(u_t)[0]$ . Para ello en cada instante  $t$  dispondremos de un conjunto de posibles políticas  $p(u_t)$ , escogeremos la que minimice la función  $f$  en cada paso siguiendo el principio de optimalidad.

### 3.4.2. Modos de vacunación

Una vez hemos determinado el porcentaje de vacunas asignadas a cada población para cada intervalo temporal se nos plantea el problema de elegir a quien vacunar. En nuestro estudio las vacunas asignadas a los mayores se distribuyen uniformemente, sin embargo, hemos elegido dos modos de vacunación para los jóvenes.

- **Modo popular:** La vacuna se administra en orden de popularidad, es decir, los nodos que poseen más vecinos son vacunados antes. Este modo se utiliza con el objetivo de reducir el mayor valor propio de la matriz de adyacencia del grafo, puesto que según el Teorema 2.9 el umbral en el que la epidemia termina desapareciendo depende directamente de este parámetro. Vacunando a los nodos más conectados la enfermedad se transmitirá de manera más lenta.
- **Modo conectado:** La vacuna se administra en orden de conexión con los mayores, es decir, la gente joven que interactúa más con mayores son vacunados primero. Este modo de vacunación tiene como objetivo proteger a la población más vulnerable frenando la transmisión de la enfermedad desde la población joven, que está mucho más conectada, a la población mayor, que posee menos conexiones.

## 3.5. Implementación

La implementación se ha realizado íntegramente en Python, desde la creación de los grafos hasta la optimización. La creación de grafos se ha realizado utilizando el algoritmo descrito en la sección 3.2.

Una vez creado el grafo, a la hora de inicializarlo infectamos a una pequeña parte de la población. Para simular el comportamiento de la epidemia hemos tomado los días como unidad de tiempo, es decir, cada día vacunamos a una cierta cantidad de personas y estudiamos que ocurre con cada nodo del grafo. El pseudocódigo de la evolución se encuentra en la figura 3.3.

```

G = simular_un_dia(G)
1.  for  $k$  in  $G$  do
2.    if estado( $k$ ) == Vacunado then
3.      siguiente_estado( $k$ , Vacunado)
4.    elif estado( $k$ ) == Susceptible then
5.      for  $k_i$  in vecinos( $k$ ) do
6.        if estado( $k_i$ ) == Infectado then
7.           $w \leftarrow \text{unif}(0, 1)$ 
8.          if  $w < \beta$  then
9.            siguiente_estado( $k$ , Infectado)
10.         fi
11.       fi
12.     od
13.   elif estado( $k$ ) == Infectado then
14.      $w \leftarrow \text{unif}(0, 1)$ 
15.     if  $w < \rho$  then
16.       siguiente_estado( $k$ , Recuperado)
17.     else
18.        $w \leftarrow \text{unif}(0, 1)$ 
19.       if  $w < \phi$  then
20.         siguiente_estado( $k$ , Victima)
21.       fi
22.     fi
23.   fi
24. od
25. return  $G$ 

```

Figura 3.3: Pseudocódigo que describe la evolución de la epidemia en un día. La función siguiente\_estado recibe como argumento un nodo y un estado y actualiza el estado del nodo. La función unif elige un número al azar con probabilidad uniforme entre los parámetros dados, en nuestro caso entre los números 0 y 1.

A la hora de vacunar a la población mayor, los individuos son elegidos al azar entre aquellos que son susceptibles con probabilidad uniforme, mientras que la población joven posee dos modos, la implementación de esta vacunación la hemos llevado a cabo mediante la utilización de una lista de prioridad que será pasada como argumento a la función de vacunación según el modo utilizado:

- **Modo popular:** Creamos una lista de los nodos jóvenes ordenada según el número de vecinos que poseen, los nodos que poseen más vecinos se vacunan antes que los que poseen menos.
- **Modo conectado:** Creamos una lista de los nodos jóvenes ordenada según el número de vecinos mayores que poseen, los nodos que poseen más vecinos mayores se vacunan antes que los que poseen menos.

A la hora de elegir a que nodo vacunar se busca dentro de la lista ordenada correspondiente los nodos susceptibles que están disponibles para ser vacunados y se vacunan según el orden de la lista. El pseudocódigo para la vacunación se encuentra en la figura 3.4

```

G = vacunar(G, I, R, vace, vacy, lista_jovenes)
1.  vacunas_restantes ← vacunar_mayores(G, I, R, vace)
2.  extra_jovenes ← vace - vacunas_restantes
3.  vacunar_jovenes(G, I, R, vacy + extra_jovenes, lista_jovenes)
4.  return G

```

Figura 3.4: Pseudocódigo que describe el proceso de vacunación. Primero se vacuna a la población mayor, donde  $I$  es una lista de los nodos infectados, y  $R$  es una lista de los nodos recuperados. La función `vacunar_mayores` vacuna a la población mayor susceptible (es decir los nodos que no estén en la lista de vacunados ni recuperados) eligiendo nodos al azar con probabilidad uniforme, si no hay suficientes susceptibles disponibles, las vacunas restantes se asignan a los jóvenes. La función `vacunar_jovenes` vacuna a la población joven susceptible según el orden dado por la lista `lista_jovenes` que se obtiene según el modo de vacunación utilizado en jóvenes.

A la hora de realizar una simulación para  $t$  días, por cada día ejecutamos primero la vacunación siguiendo el algoritmo descrito en 3.4 y posteriormente la evolución de la epidemia siguiendo el algoritmo descrito en 3.3. El pseudocódigo de la simulación se encuentra en 3.5.

```

G = simular_t_dias(T, G, Vacy, Vace, lista_jovenes)
1.  for  $k \leftarrow 1, \dots, T$  do
2.     $G \leftarrow$  vacunar( $G, I, R, Vac_y, Vac_e, lista_jovenes$ )
3.     $G \leftarrow$  simular_un_dia( $G$ )
4.  od
5.  return G

```

Figura 3.5: Pseudocódigo que describe el proceso de vacunación y evolución de la epidemia en  $T$  días.  $Vac_y$  y  $Vac_e$  es la cantidad de vacunas diarias que reciben la población joven y mayor respectivamente. `lista_jovenes` es el orden de vacunación de la población joven.  $I$  es la lista de los nodos infectados, y  $R$  es la lista de los nodos recuperados. Para cada día primero se vacuna a ambas poblaciones utilizando la función `vacunar` y posteriormente se estudia la evolución de cada nodo mediante la función `simular_un_dia`.

Una vez tenemos disponible todas las herramientas necesarias para simular la evolución de la epidemia, veamos la implementación del algoritmo de optimización utilizado.

Partimos de una cantidad de vacunas totales  $V\_total$  que serán aplicadas en  $T$  días divididos en intervalos de  $m$  días dando un total de  $N$  momentos temporales a estudiar  $[t_0, \dots, t_N]$ , por lo que las vacunas diarias utilizadas tendrán un valor fijo  $v\_diarias = V\_total/T$ .

Para poder realizar la optimización hemos dividido nuestro conjunto de posibles escenarios en los distintos casos que podemos encontrarnos en cada tiempo  $t$ . El conjunto de posibles escenarios está determinado por los valores de las variables de estado  $(Iy, Ie, Vy, Ve)$ , sin embargo, como aplicamos la misma cantidad de vacunas en cada momento temporal podemos determinar  $Vy$  a partir de los días transcurridos y  $Ve$ , por lo que nuestro conjunto de posibles escenarios estará determinado por  $(Iy, Ie, Ve)$ . Dividimos el conjunto  $(Iy, Ie, Ve)$  en  $(MAX\_Iy, MAX\_Ie, MAX\_Ve)$  intervalos para su estudio abarcando todos los posibles valores que pueden tomar. Para de facilitar el guardado y acceso a los datos hemos creado dos funciones para indexar los posibles valores de  $(Iy, Ie, Ve)$ , la primera, `val2idx`, asocia un valor dado a un índice, es decir, asocia los valores de  $(Iy, Ie, Ve)$  a sus correspondientes índices  $(iyx, iex, vex)$ , y la segunda, `idx2val`, asocia un índice  $(iyx, iex, vex)$  a un valor  $(Iy, Ie, Ve)$ . Los índices  $(iyx, iex, vex)$  tomarán valores comprendidos entre 0 y  $MAX\_Iy - 1$ ,  $MAX\_Ie - 1$  y  $MAX\_Ve - 1$  respectivamente.

Nuestro objetivo es encontrar para cada posible valor de  $(Iy, Ie, Ve)$  la cantidad de vacunas óptima que suministrar a la población mayor y joven, por ello hemos definido  $l$  posibles porcentajes a probar  $Pp = [p_0, \dots, p_l]$ . Este es nuestro conjunto de decisiones, en cada tiempo  $[t_0, \dots, t_N]$  debemos elegir el  $p$  que minimiza las muertes totales.

Como explicamos en la sección 3.4, empezaremos por el final, es decir por  $t_N$  y asignamos inicialmente a  $Victimas[t_N, iyx, iex, vex] = 0 \forall iyx, iex, vex$ , a partir de aquí calcularemos las víctimas y el porcentaje de  $Vacunas[t, iyx, iex, vex]$  asignadas a la población mayor en cada instante  $t \in [t_0, \dots, t_N]$ .

Se debe de tener en cuenta que en cada instante temporal se calcula la política óptima dependiendo del costo acumulado del siguiente instante temporal, es decir, para calcular la política óptima del instante  $t_n$  se utilizan los costos del instante  $t_{n+1}$  (previamente calculados puesto que empezamos por el final) probando todos los posibles valores de  $Pp$  hasta encontrar el que minimiza  $Victimas[t_n, iyx, iex, vex]$ . Por lo tanto, una vez fijado un número  $p$  de vacunas asignadas al grupo de los mayores a partir del estado  $(Iy, Ie, Ve)$  en tiempo  $t_n$ , el estado siguiente  $t_{n+1}$  queda fijado. Para la obtención del estado siguiente hemos utilizado la función `obtener_estado_siguiente`, que nos devuelve las variables del estado  $(Iyn, Ien, Ven)$  al que iremos a parar después de que se haya simulado asignar  $p$  vacunas a los mayores. De esta forma podemos acumular el costo total del camino desde  $t_N$  hasta  $t_n$  para elegir la  $p$  que minimiza el número de víctimas a partir del instante  $t_n$ , repitiendo este proceso hasta el instante  $t_0$  obtenemos el resultado deseado.

El pseudocódigo de la optimización se encuentra en la figura 3.6.



```

[Victimas, Vacunas] = optimizacion(G, lista_jovenes)
1.  for  $t \leftarrow [t_N - 1, \dots, t_0]$  do
2.    for  $iyx \leftarrow [iyx_0, \dots, iyx_{MAX_{Iy}-1}]$  do
3.      for  $iex \leftarrow [iex_0, \dots, iex_{MAX_{Ie}-1}]$  do
4.        for  $vex \leftarrow [vex_0, \dots, vex_{MAX_{Ve}-1}]$  do
5.          [Iy, Ie, Ve]  $\leftarrow$  idx2val([iyx, iex, vex])
6.          Vy  $\leftarrow$  v_diarias  $\cdot$  t - Ve
7.          for  $p \leftarrow [p_0, \dots, p_l]$  do
8.            G  $\leftarrow$  simular_t_dias(dias, G, dias  $\cdot$  p, dias  $\cdot$  (1 - p), lista_jovenes)
9.            v  $\leftarrow$  obtener_victimas(G)
10.           [Iyn, Ien, Ven]  $\leftarrow$  obtener_estado_siguiete(G)
11.           [iyxn, iexn, vexn]  $\leftarrow$  val2idx([Iyn, Ien, Ven])
12.           if v + Victimas[t+1, iyxn, iexn, vexn] < costo_min then
13.             costo_min  $\leftarrow$  v + Victimas[t+1, iyxn, iexn, vexn]
14.             p_min  $\leftarrow$  p
15.           fi
16.         od
17.       Victimas[t, iyx, iex, vex]  $\leftarrow$  costo_min
18.       Vacunas[t, iyx, iex, vex]  $\leftarrow$  [p_min, iynx, iexn, vexn]
19.     od
20.   od
21. return [Victimas, Vacunas]

```

Figura 3.6: Pseudocódigo que describe la resolución del problema planteado usando Programación Dinámica. Donde lista\_jovenes define el modo que se está usando.

### 3.5.1. Almacenamiento de resultados

En cuanto al almacenamiento de los resultados consiste en dos estructuras paralelas, que poseerán el tamaño de los posibles valores de las variables de estado  $e(t)$  para cada instante de tiempo  $t$ . Ambas estructuras son arrays de 4 dimensiones: una para el tiempo y tres para los posibles valores de  $(Iy, Ie, Ve)$ . El tamaño de cada dimensión dependerá únicamente de los índices que utilizemos, es decir,  $MAX_{Iy}$ ,  $MAX_{Ie}$  y  $MAX_{Ve}$ .

- La estructura *Victimas* almacena el número víctimas acumulado.
- La estructura *Vacunas* almacena los porcentajes  $p$  óptimos de vacunas que asignamos a los mayores. Además, en *Vacunas* almacenamos también el siguiente estado al que iremos a parar tras asignar las vacunas a los mayores, es decir, en *Vacunas* se encuentra nuestra política óptima.

Nuestro objetivo consistía en minimizar el número total de víctimas hasta tiempo  $N$  partiendo de una pequeña población infectada, por lo tanto la cantidad minimizada de víctimas se encontrará guardada la posición  $Victimas[0, 0, 0, 0]$ . Para encontrar el porcentaje de vacunas asignadas a los mayores en cada tiempo  $[t_0, \dots, t_N]$  basta con recorrer los índices guardados en  $Vacunas$  partiendo de  $Vacunas[0, 0, 0, 0]$ . Nótese que ambas estructuras poseen el camino óptimo para cualquier escenario inicial, tan solo debemos situarnos en el índice correspondiente al estado inicial y seguir el camino que indica  $Vacunas$ . Por otra parte, gracias al principio de optimalidad, el modelo se puede retroalimentar con datos reales y posicionarse en los índices de estado correspondiente en todo momento para encontrar un camino óptimo a partir de ese punto sin necesidad de hacer cálculos extra.

### 3.5.2. Valores utilizados en la simulación

A la hora de simular el comportamiento de la epidemia hemos utilizado los cuatro grafos descritos en 3.2.

Por otro lado hemos fijado el porcentaje de vacunas totales a 0,2, ya que si tomamos un porcentaje más grande terminaremos vacunando a los mayores muy rápido y no habrá nada que optimizar.

En cuanto al tiempo total de la simulación, hemos elegido dos valores: vacunación intensiva en 49 días y vacunación más lenta en 98 días. Los intervalos de tiempo tomados son cada 7 días, de forma que cada semana decidiremos que porcentaje de vacunas asignar a cada grupo de la población. Quedando así nuestros intervalos de la siguiente manera:

- 98 días: [0, 7, 14, 21, 28, 35, 42, 49, 56, 63, 70, 77, 84, 91].
- 49 días: [0, 7, 14, 21, 28, 35, 42, 49].

Hemos elegido 11 posibles porcentajes de vacunas asignadas a los mayores a probar, según lo descrito en 3.6 tenemos que  $Pp = [0, 0'1, 0'2, 0'3, 0'4, 0'5, 0'6, 0'7, 0'8, 0'9, 1]$ . Este es el valor  $p$  que debe ser optimizado para cada instante temporal  $t$  y cada posible valor de nuestras variables de estado  $(Iy, Ie, Ve)$ .

Por último, hemos dividido los posibles valores de  $(Iy, Ie, Ve)$  en 6 intervalos indexados. Según lo descrito en 3.6 tenemos que  $MAX\_Iy = 6$ ,  $MAX\_Ie = 6$ , y  $MAX\_Ve = 6$ . Por lo tanto tendremos que el conjunto de valores que pueden tomar nuestro índices son,  $iyx \in [0, 1, 2, 3, 4, 5]$ ,  $iey \in [0, 1, 2, 3, 4, 5]$  y  $vex \in [0, 1, 2, 3, 4, 5]$ . Hemos elegido estos valores debido a que si elegimos unos más pequeños perdemos precisión, pero si elegimos unos más grandes el tiempo computacional es excesivamente grande. Cuanto mayor sea este valor más precisión tendremos.

# CAPÍTULO 4

## Resultados

---

En este capítulo presentamos los resultados del estudio mediante la realización de simulaciones. Por una parte averiguamos el camino óptimo a partir de unos pocos nodos infectados siguiendo el algoritmo descrito en 3.4 y posteriormente calculamos las víctimas realizando una simulación que utiliza el porcentaje de vacunas a aplicar en cada intervalo temporal hallado previamente.

Este capítulo consta de dos secciones. La primera consiste en la presentación y análisis de las gráficas del reparto de vacunas asociado a cada semana y la cantidad de víctimas diarias para cada valor temporal total (49 días y 98 días), modo (popular o co y topología). Y la segunda parte consiste en una comparación de nuestro modelo con otras 3 estrategias: vacunar primero a los mayores, vacunar primero a los jóvenes y asignar la mitad de dosis disponibles a cada grupo.

### 4.1. Resultados de la Programación Dinámica

Veremos un total de dieciséis gráficas. En cada topología tendremos 4 gráficas: Las dos primeras corresponden a 49 días de vacunación, las dos segundas a 98 días de vacunación. Dentro de cada par de gráficas la primera corresponde al modo popular y la segunda al modo conectado. El código utilizado para describir la topología utilizada para los nombres de las gráficas se encuentra en la tabla 3.1. En todas las simulaciones hemos empezado con 1 nodo joven infectado en tiempo 0.

### 4.1.1. Topología $A\alpha$

En este caso hay pocas interacciones con la población mayor, es decir, cada persona mayor interactúa con pocas personas mayores y también hay pocas interacciones entre la población mayor y la población joven. Por lo tanto estamos en el caso en el que la población mayor está aislada del resto de individuos.

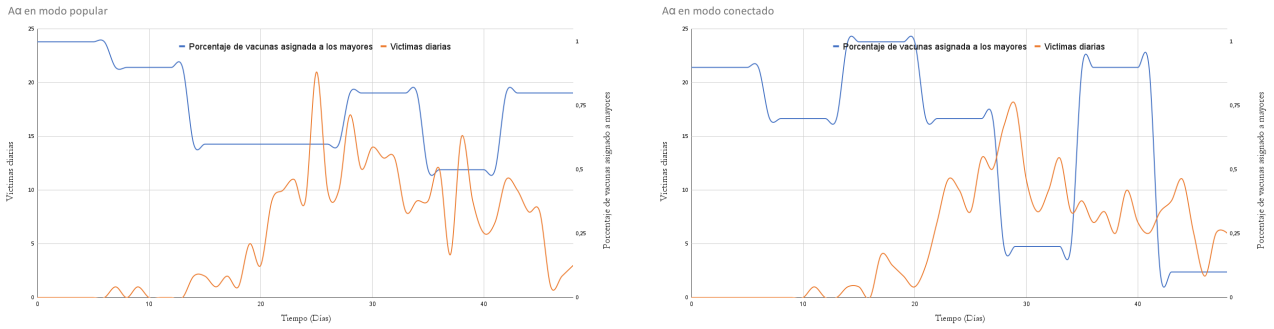


Figura 4.1: Resultados topología  $A\alpha$  en 49 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

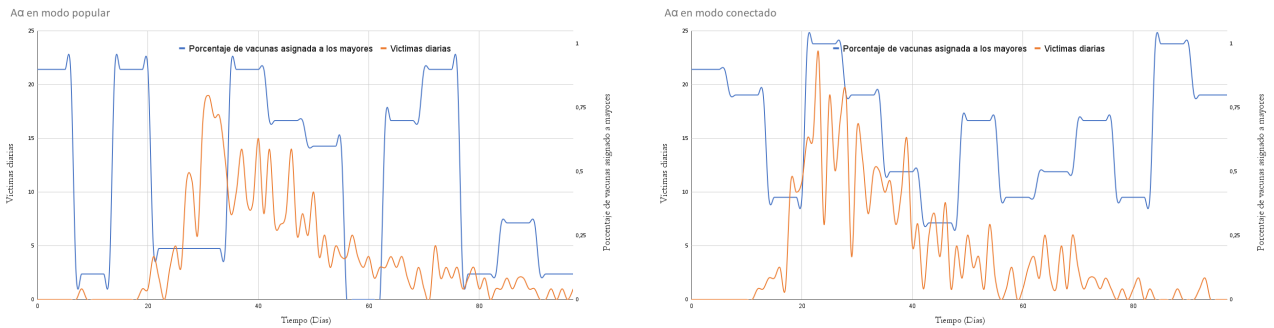


Figura 4.2: Resultados topología  $A\alpha$  en 98 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

Las cuatro gráficas empiezan con una vacunación intensiva de los mayores seguida de una alternación del foco de vacunación por intervalos temporales. Algo a destacar de estas gráficas es que tienden a subir el número de vacunas asignadas a los mayores cuando ha habido previamente una subida en la cantidad de víctimas diarias y posteriormente tiende a asignar más vacunas a los jóvenes conforme decrece la cantidad de víctimas. Las caídas abruptas de asignación de vacunas como en el caso del modo popular en la figura 4.2 en  $t = 7$  son debidas a pequeños errores de nuestro algoritmo y deben corregirse posteriormente.

### 4.1.2. Topología $B\alpha$

En este caso hay muchas interacciones entre mayores pero hay pocas interacciones de la población mayor con la población joven.

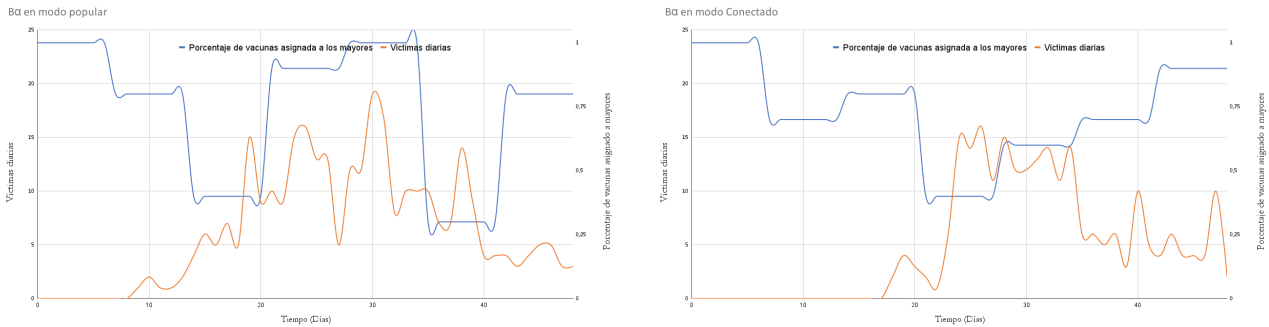


Figura 4.3: Resultados topología  $B\alpha$  en 49 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

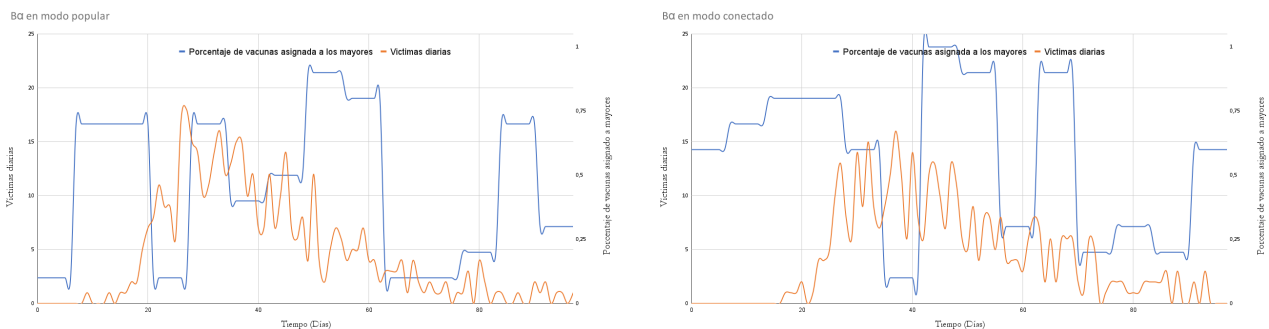


Figura 4.4: Resultados topología  $B\alpha$  en 98 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

Tres de las cuatro gráficas empiezan con una vacunación intensiva a los mayores y posteriormente alternan el foco de vacunación por intervalos temporales. En estas gráficas también se puede observar el fenómeno previamente mencionado, tienden a subir el número de vacunas asignadas a los mayores cuando ha habido previamente una subida en la cantidad de víctimas diarias y posteriormente tiende a asignar más vacunas a los jóvenes conforme decrece la cantidad de víctimas. En la gráfica correspondiente al modo popular en la figura 4.4, empezar con una vacunación intensiva en el grupo de los jóvenes ayuda a disminuir la velocidad de propagación de la enfermedad en el grafo de los jóvenes. Esta estrategia puede deberse a que los mayores están relativamente aislados del grupo de los jóvenes y por lo tanto el algoritmo se puede centrar en frenar la difusión de la enfermedad en el grafo de los jóvenes.

### 4.1.3. Topología $A\beta$

En este caso hay pocas interacciones entre mayores pero hay muchas interacciones de la población mayor con la población joven.

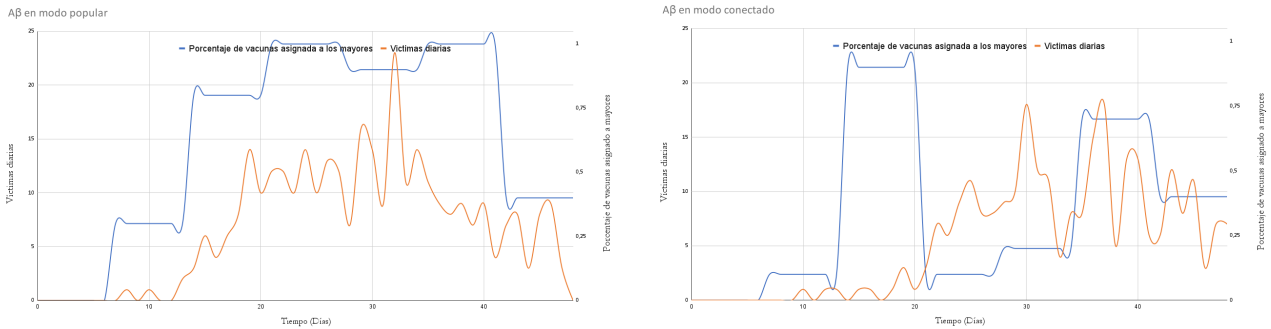


Figura 4.5: Resultados topología  $A\beta$  en 49 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

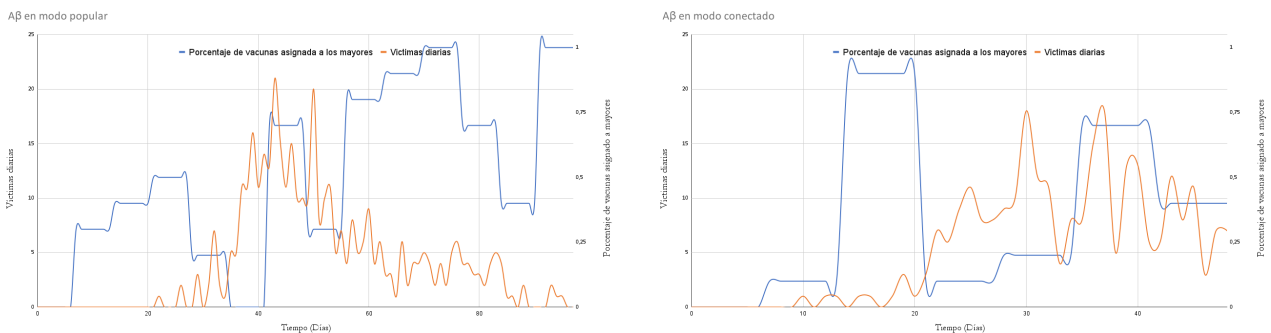


Figura 4.6: Resultados topología  $A\beta$  en 98 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

Tres de las cuatro gráficas empiezan con una vacunación intensiva a los jóvenes. Esto puede deberse a que en este caso los jóvenes están muy conectados a los mayores y la enfermedad se extiende más rápido en la población joven ya que poseen más vecinos, por lo tanto la estrategia a usar en este caso consiste en una primera vacunación intensiva a los jóvenes para frenar la expansión de la epidemia a través del grafo, seguida de una tendencia a subir la dosis de vacunas asignadas a los mayores en el modo popular para proteger a la población vulnerable, y alternar el foco de vacunación en el modo conectado puesto que los nodos conectados con mayores hacen de barrera para evitar que la enfermedad se transmita a la población mayor. Aquí también podemos apreciar el fenómeno visto en las gráficas anteriores, sin embargo no es tan claro como en los casos anteriores.

#### 4.1.4. Topología $B\beta$

En este caso hay muchas interacciones entre mayores y hay muchas interacciones de la población mayor con la población joven.

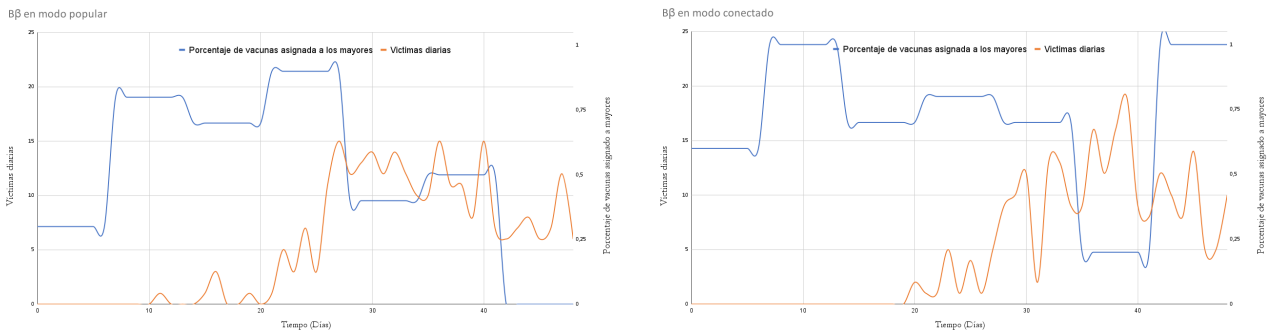


Figura 4.7: Resultados topología  $B\beta$  en 49 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

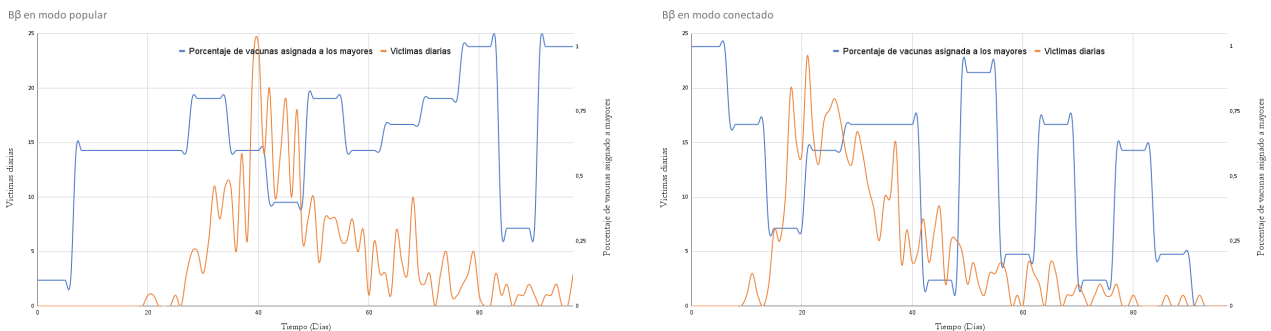


Figura 4.8: Resultados topología  $B\beta$  en 98 días de simulación. La primera gráfica corresponde al modo popular y la segunda al modo conectado. La línea azul representa el porcentaje de vacunas asignado a la población mayor y la línea naranja las víctimas diarias.

Las gráficas correspondientes al modo popular empiezan con una vacunación intensiva a los jóvenes para frenar la expansión de la epidemia en la población que está fuertemente conectada seguida de una alternancia del foco entre jóvenes y mayores. Por otro lado en el modo conectado el foco al inicio son los mayores, esto puede deberse a que unas pocas dosis asignadas a los nodos más conectados con los mayores frena la transmisión de la enfermedad entre ambos grupos siendo suficiente para disminuir las víctimas, posteriormente alterna el foco entre ambos grupos. Aquí también podemos apreciar en menor medida el fenómeno visto en las gráficas anteriores.

### Características generales

En la mayoría de los casos las víctimas totales en el modo conectado son inferiores a las del modo popular. Esto nos da la pista de que la vacunación debería estar centrada en estos nodos. Por otra parte, hemos podido observar que el algoritmo tiende a aumentar las dosis asignadas a los mayores conforme suben las víctimas y cuando empieza decrecer el número de víctimas va alternando entre ambas poblaciones. Esto puede deberse a que el modelo se centra en frenar la difusión de la enfermedad vacunando a la población joven, pero a la vez cambia de foco para reducir el número de víctimas en la población mayor cuando hay un aumento significativo de víctimas diarias.

Algo a destacar en el modo conectado es que en la mayoría de casos se asigna una gran cantidad de vacunas a la población mayor y posteriormente se alterna el foco de vacunación. Esto puede deberse a que vacunar a unos pocos nodos jóvenes conectados con la población mayor permite que la enfermedad no se transmita entre la población joven y mayor puesto que estos nodos actúan como barrera.

En cuanto al modo popular, por lo general tiende a iniciar con una vacunación intensiva a los jóvenes seguida de una alternancia del foco de vacunación. Esto puede deberse a que al no poder aislar correctamente a los mayores, la enfermedad empieza a difundirse rápidamente en el grafo correspondiente a los jóvenes y por lo tanto vacunar a los jóvenes que poseen más vecinos repercute en una disminución de la velocidad de propagación en la población general, y por tanto tardará más en propagarse al grafo de los mayores. Posteriormente va alternando el foco de vacunación entre jóvenes y mayores para disminuir el número de víctimas y evitar que la velocidad de propagación de la enfermedad aumente.

## 4.2. Comparación con otras estrategias

Con el objetivo de verificar que tan óptimo es nuestro algoritmo hemos comparado los resultados obtenidos con otras tres estrategias:

1. Asignar todas las dosis disponibles a los mayores, es decir,  $p = 1 \forall t$ .
2. Asignar todas las dosis disponibles a los jóvenes, es decir,  $p = 0 \forall t$ .
3. Asignar la mitad de las dosis a los mayores y la otra mitad a los jóvenes, es decir,  $p = 0,5 \forall t$ .

Para comparar las estrategias hemos realizado 100 simulaciones para cada estrategia y hemos hallado la media de víctimas totales en cada una. Hemos hecho dos pruebas, en una hemos empezado con 1 nodo joven infectado, y en otra con 70 infectados jóvenes y 15 infectados mayores.

En la tabla 4.1 encontramos los resultados de las distintas estrategias aplicadas a las diferentes topologías con ambos modos partiendo de un único infectado joven.



	$p = 1 \forall t$	$p = 0,5 \forall t$	$p = 0 \forall t$	Nuestro modelo
Días	Victimas de la topología $A\alpha$ en modo popular			
49	257	231	213	250
98	383	388	417	387
Días	Victimas de la topología $A\alpha$ en modo conectado			
49	263	223	180	250
98	385	372	373	378
	Victimas de la topología $B\alpha$ en modo popular			
49	255	248	266	247
98	411	427	453	431
	Victimas de la topología $B\alpha$ en modo conectado			
49	258	255	247	253
98	405	431	454	420
	Victimas de la topología $A\beta$ en modo popular			
49	280	269	243	251
98	421	429	437	433
	Victimas de la topología $A\beta$ en modo conectado			
49	279	272	246	263
98	420	428	443	430
	Victimas de la topología $B\beta$ en modo popular			
49	275	265	275	261
98	421	434	449	430
	Victimas de la topología $B\beta$ en modo conectado			
49	275	281	296	273
98	421	437	459	430

Tabla 4.1: Comparación del número de víctimas con otros modelos partiendo de un infectado joven.

La estrategia de vacunar primero a los mayores ( $p = 1 \forall t$ ) es muy eficiente a largo plazo pero muy poco eficiente a corto. Ocurre lo contrario con el caso de vacunar primero a los jóvenes ( $p = 0 \forall t$ ). Por otro lado vacunar a la mitad de la población joven y a la mitad de la población mayor ( $p = 0,5 \forall t$ ) ofrece una media de las estrategias anteriores sin llegar a ser la más óptima en ningún caso. Nuestro modelo es uno de los más estables tanto a corto plazo como a largo plazo y en algunas ocasiones como en el caso de 49 días en el grafo  $B\alpha$  y  $B\beta$  llega a ser la estrategia más óptima.

Algo destacable que encontramos en todas las estrategias en ambos modos de vacunación es que en general la cantidad de víctimas principalmente en la topología  $A\alpha$  pero también en la topología  $B\alpha$  es inferior a las otras dos topologías. Esto implica que tomar medidas de aislamiento como el confinamiento de la población mayor al menos durante las primeras fases de vacunación incidirán muy positivamente reduciendo el número de víctimas totales.

En la tabla 4.2 encontramos los resultados de las distintas estrategias aplicadas a las diferentes topologías con ambos modos partiendo de 70 infectados jóvenes y 15 infectados mayores. Los valores de esta tabla para nuestro modelo los hemos obtenido incrementando el número de máximo de índices de las variables de estado a 8.

	$p = 1 \forall t$	$p = 0,5 \forall t$	$p = 0 \forall t$	Nuestro modelo
Días	Victimas de la topología $A\alpha$ en modo popular			
49	328	330	350	328
98	407	412	434	408
Días	Victimas de la topología $A\alpha$ en modo conectado			
49	328	318	320	315
98	402	407	421	402
	Victimas de la topología $B\alpha$ en modo popular			
49	360	374	392	360
98	438	450	464	450
	Victimas de la topología $B\alpha$ en modo conectado			
49	359	380	396	358
98	440	453	463	451
	Victimas de la topología $A\beta$ en modo popular			
49	361	365	379	360
98	439	445	458	441
	Victimas de la topología $A\beta$ en modo conectado			
49	361	366	378	365
98	439	442	454	445
	Victimas de la topología $B\beta$ en modo popular			
49	360	373	395	368
98	440	453	465	444
	Victimas de la topología $B\beta$ en modo conectado			
49	363	374	396	374
98	438	451	464	451

Tabla 4.2: Comparación del número de víctimas con otros modelos partiendo de 70 infectados jóvenes y 15 infectados mayores.

Tanto la estrategia de vacunar primero a los mayores ( $p = 1 \forall t$ ) como nuestra estrategia toman los valores más bajos de víctimas en todos los casos. Pese a que ambas estrategias toman valores muy similares la más óptima en la mayoría de casos es la de vacunar primero a los mayores ( $p = 1 \forall t$ ). Nótese que en esta tabla nuestro modelo toma valores más bajos en comparación con las otras estrategias en más ocasiones que en la tabla 4.1 gracias a que hemos tomado 8 índices en lugar de 6. Esto indica que cuanto más índices tomemos obtendremos mejores resultados.

## CAPÍTULO 5

# Conclusiones

---

Como hemos podido observar, a primera vista la estrategia actual que están utilizando muchos países es una de las más óptimas en la mayoría de los casos. Sin embargo, nuestro modelo es capaz de competir con la estrategia actual incluso llegando a reducir las víctimas en casos específicos. El cálculo de los porcentajes óptimos se puede ajustar mucho mejor aumentando el número máximo de índices correspondientes a las variables de estado de la simulación, de esta forma los posibles escenarios estarán divididos en intervalos más pequeños y por lo tanto será más preciso a la hora de calcular los porcentajes. Sin embargo, esto repercute en un aumento significativo del tiempo computacional, por lo que para que sea viable necesitamos disponer de un buen *hardware*.

Por otro lado una estrategia común que hemos podido observar en la mayoría gráficas del capítulo anterior es que nuestro modelo tiende a subir la cantidad de vacunas asignadas a los mayores cada vez que hay un aumento significativo del número de víctimas y posteriormente va disminuyendo esta asignación conforme decrece este número. Esto nos da a entender que el algoritmo busca un equilibrio entre frenar la expansión de la enfermedad en el grafo de los jóvenes y proteger a los mayores para disminuir la mortalidad.

### 5.1. Ventajas

La ventaja principal que encontramos en nuestro modelo es la capacidad que tiene de adaptarse a cualquier situación gracias a que calculamos todos los caminos óptimos para cada posible situación de infectados y vacunados que podemos encontrarnos en la realidad. Esto permite retroalimentar al modelo con los datos reales para volver a hallar el camino óptimo en base a esos datos sin tener que realizar ningún cálculo extra puesto que guardamos toda la información obtenida para cada posible situación en cada instante temporal. Esto permite que nuestro modelo sea muy flexible.

## 5.2. Limitaciones

La limitación principal que encontramos en el modelo es la dificultad para identificar los nodos correspondientes a la población joven que debemos vacunar. En nuestra simulación las relaciones sociales de los individuos están muy simplificadas y podemos saber con precisión la cantidad de vecinos que tiene cada nodo en todo momento. En el caso de usar el modo conectado, podemos vacunar a los individuos según su profesión (p. ej. personal que trabaja en las residencias de ancianos). Sin embargo, identificar los individuos a vacunar en el modo popular es más complicado puesto que supondría una invasión en la privacidad de los individuos.

Otro inconveniente se encuentra en identificar la topología de la red. Pese a que una de las medidas a tomar es el confinamiento de la población, y en ese caso podríamos suponer que estamos en la topología  $A\alpha$ , en el resto de los casos no está tan claro ante que topología nos encontramos.

Otra de las limitaciones que posee el modelo es el suponer que los individuos interactúan uniformemente con sus vecinos.

## 5.3. Posibles mejoras

Una de las mejoras que se desearía hacer en el modelo consiste en ser capaces de representar la densidad de interacciones entre los nodos, esto puede hacerse utilizando pesos que representen la frecuencia con la que dos personas se relacionan en los enlaces de los nodos y tenerlos en cuenta a la hora de simular la evolución de la enfermedad.

Por otro lado, sería muy interesante poder intercalar ambos modos de vacunación en la población joven para poder mejorar la optimización de la cantidad de vacunas asignadas en cada momento temporal teniendo la posibilidad de cambiar el foco dentro la población joven.

Otra mejora deseable sería encontrar la cantidad óptima de índices para obtener resultados fiables sin aumentar excesivamente el tiempo de cómputo.

Por último, si se dispone de un *hardware* lo suficientemente potente para poder añadir la variable de estado de los recuperados, el algoritmo se podría ajustar aún más a cada caso específico obteniendo así mejores resultados.

# Bibliografía

---

- [1] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.
- [2] Richard Bellman. The theory of dynamic programming. Technical report, Rand corp santa monica ca, 1954.
- [3] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.
- [4] Ronald Manríquez, Camilo Guerrero-Nancuante, Felipe Martínez, and Carla Taramasco. Spread of epidemic disease on edge-weighted graphs from a database: A case study of covid-19. *International Journal of Environmental Research and Public Health*, 18(9):4432, 2021.
- [5] Manuel Maurette and Ignacio Ojea. Programación dinámica. *Universidad de Buenos Aires Sitio Web: [http://cms. dm. uba. ar/materias/1ercuat2009/optimizacion/Maurette\\_Ojea. pdf](http://cms.dm.uba.ar/materias/1ercuat2009/optimizacion/Maurette_Ojea.pdf)*, 2006.
- [6] Simone Santini. Optimal covid-19 vaccination strategies with limited vaccine and delivery capabilities; a model based on social networks. *ACM Transaction on Computing for Healthcare*, 2021. (in press).
- [7] Yang Wang, Deepayan Chakrabarti, Chenxi Wang, and Christos Faloutsos. Epidemic spreading in real networks: An eigenvalue viewpoint. In *22nd International Symposium on Reliable Distributed Systems, 2003. Proceedings.*, pages 25–34. IEEE, 2003.
- [8] Howard Howie Weiss. The SIR model and the foundations of public health. *Materials mathematics*, pages 0001–17, 2013.