



MODELO MATEMÁTICO DE PROPAGACIÓN DE COVID-19



FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES

GRADO EN ECONOMÍA

TRABAJO DE FIN DE GRADO

Autor: Mario Pajares García

Tutor: Miguel Jerez Méndez

Curso Académico: 2020-2021

Convocatoria Ordinaria Junio

Índice

1. Resumen	2
2. Introducción	3
3. Metodología	5
4. Datos de movilidad utilizados para el modelo	7
6. Simulaciones	14
6.1. Simulación del modelo sin ningún tipo de evento	14
6.2. Simulación del modelo con evento de cuarentena	16
6.3. Simulación del modelo con vacunación	20
7. Conclusión y propuestas de futuro	22
8. Referencias bibliográficas	24
9. Anexos.....	26

1. Resumen

El objetivo de este trabajo ha sido desde el primer momento, contribuir a la elaboración de una metodología que pueda mejorar la futura toma de decisiones por parte de las autoridades sanitarias para reducir las consecuencias del Covid-19. Para ello vamos a elaborar un modelo al que llamaremos SIRHDQ, el cual incluye el tránsito de personas entre los diferentes territorios de la Comunidad de Madrid.

Este modelo trabaja con parámetros como la proporción de infectados que son hospitalizados, el porcentaje de personas que fallecen o el número de personas a las que retransmite la enfermedad una persona infectada. Este último parámetro es la base de cualquier estudio epidemiológico ya que permite predecir el riesgo de que la enfermedad avance con más fuerza. Estos parámetros los hemos ido calculando a través de las series de infectados, hospitalizados y defunciones que nos proporciona el Ministerio de Sanidad.

Una vez planteados los datos que vamos a utilizar en el modelo, procederemos a elaborar distintas simulaciones con el fin de entender qué medidas podrían ser más eficaces a la hora de mitigar el virus. Estas simulaciones van desde plantear una situación inicial en la que no aplicamos medidas restrictivas, hasta la elaboración de una estrategia de cuarentena que combine el tiempo de cuarentena con la proporción de personas cuarentenadas.

Quizás estas estrategias no sean factibles en la sociedad actual, pero confío en que la metodología de trabajo pueda servir de ayuda a la hora de abarcar crisis epidemiológicas como la sufrida con el coronavirus.

2. Introducción

Durante el transcurso de la pandemia ha habido una discusión acerca de qué medidas restrictivas se tenían que tomar y cuando era mejor tomarlas.

El objetivo de este trabajo es desarrollar un modelo epidemiológico compartimental que permita estudiar la evolución de los distintos estados de la Comunidad de Madrid, para así optimizar la toma de decisiones. La población se divide en los siguientes compartimentos:

- Susceptibles (S): Población susceptible de contraer el virus.
- Infectados (I): Población contagiada por el virus.
- Hospitalizados (H): Población que tiene que ser hospitalizada debido a las consecuencias de Covid-19.
- Recuperados (R): Personas que han conseguido recuperarse del virus sin consecuencias graves.
- Defunciones (D): Población fallecida debido al Covid-19.
- Infectados en cuarentena (QI): Población infectada que se encuentra en cuarentena y por lo tanto no puede transmitir la enfermedad.
- Susceptibles en cuarentena (QS): Población susceptible que se encuentra en cuarentena y por lo tanto no puede contraer la enfermedad.
- Recuperados en cuarentena (QR): Población recuperada que se encuentra en cuarentena y por lo tanto no puede reinfectarse.

El objetivo de este modelo es conseguir estudiar la interacción entre los compartimentos de distintos nodos, siempre y cuando no se encuentren aislados, y de esta manera observar cómo afecta el flujo de movimientos entre S, I y R a la dinámica del modelo.

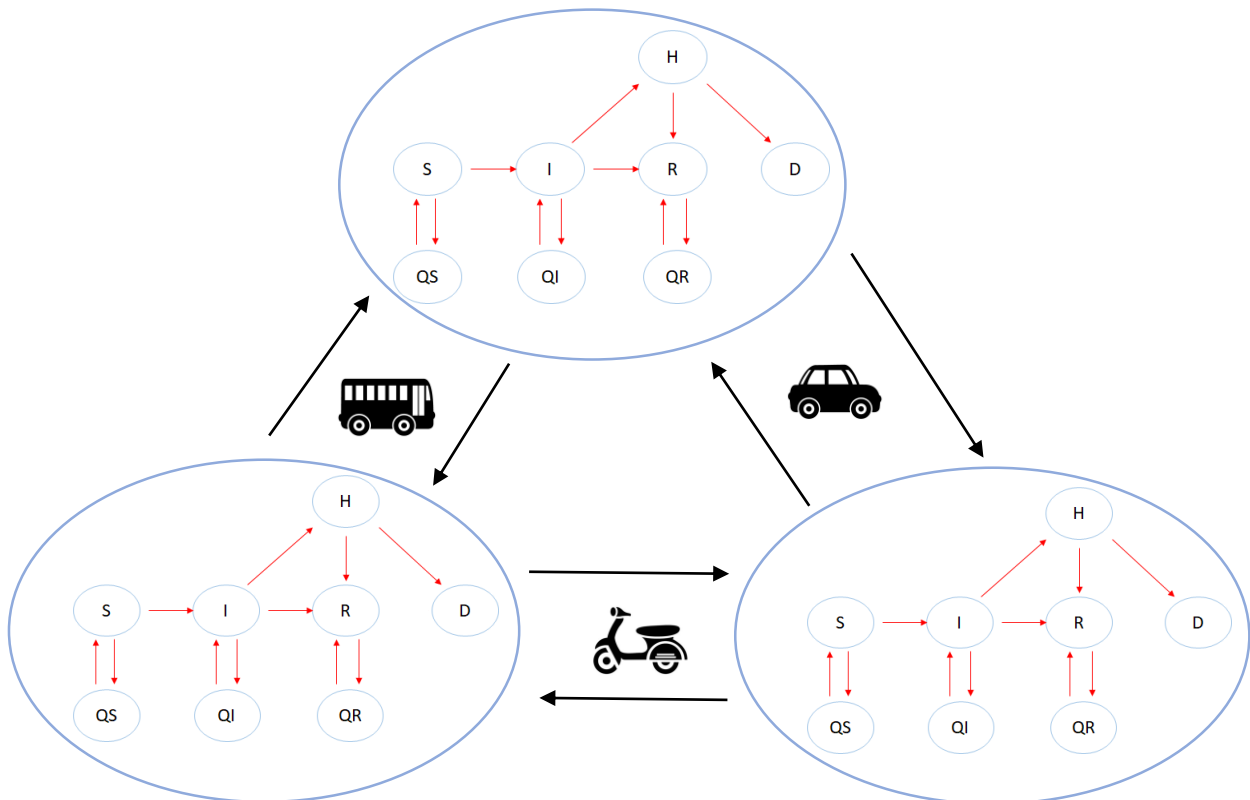


Figura 1: Esquema del modelo donde se representa la movilidad de las personas.

3. Metodología

El núcleo del modelo llevado a cabo ha sido el paquete del software de análisis de datos R SimInf. Esta librería es una de las más eficientes y flexibles a la hora de realizar modelos epidemiológicos basados en datos de enfermedades reales.

El marco de referencia integra la dinámica de la infección en subpoblaciones de tiempo continuo, utilizando el “*algoritmo de simulación estocástica de Gillespie (1976, 1977)*”, el cual genera una trayectoria estadísticamente correcta de un sistema de ecuaciones estocásticas para las que se conocen las velocidades de reacción.

Este algoritmo puede incorporar datos como nacimientos, muertes o movimiento de personas entre los nodos que estemos utilizando en un periodo de tiempo determinado.

En cuanto a los datos que necesita el modelo para su funcionamiento son los siguientes:

Espacio/Argumento	Descripción
Matriz E	Cada fila de esta matriz corresponde a un comportamiento en el modelo (susceptibles, infectados, recuperados, etc.). Las entradas distintas de 0 en una columna indican de qué comportamiento se debe tomar muestra de los individuos al procesar un evento, por lo que dependiendo de la columna que seleccionemos estaremos utilizando un grupo u otro para llevar a cabo dicho evento.
Matriz N	Determina como los individuos en los eventos de transferencia interna y externa son cambiados para entrar en otro compartimento. Cada fila, al igual que la matriz anterior, corresponde a un compartimento en el modelo. Los valores que no son 0 de una columna indican cómo nos tenemos que mover de compartimentos en orden descendente. Por ejemplo, un valor 1 en una entrada significa que los individuos muestreados se mueven al siguiente compartimento, un valor 2 significa que los individuos muestreados se mueven dos compartimentos, y así sucesivamente.
Eventos	Para codificar los eventos es necesario la utilización de las matrices anteriormente mencionadas. Una vez sepamos qué compartimento queremos coger y hacia cuál lo queremos mover, es hora de especificar qué tipo de transferencia queremos realizar, ya sea interna o externa, y en qué periodo de tiempo va a surgir dicho evento.
Ecuaciones de transición	Son las ecuaciones mediante las cuales se va a producir el cambio de compartimentos entre un estado de una muestra poblacional y otro.
Matriz de población inicial	Es una matriz que especifica la situación inicial de un conjunto de nodos. Las filas indican cada uno de los nodos o, en nuestro caso, las áreas telefónicas, y cada columna los compartimentos que utilizaremos. Esta matriz se rellena con el número de personas que se encuentran en cada compartimento en la situación inicial, es decir, justo antes de que empiece la simulación.
Tiempo	Periodo de tiempo en el que transcurre la simulación. Este lo determinamos a través de un vector entero.

Compartimentos	Es cada uno de los estados en los que se puede encontrar la población del modelo. En nuestro caso serán: Susceptibles, infectados, hospitalizados, recuperados, defunciones, individuos en cuarentena infectados, individuos en cuarentena susceptibles y por último individuos en cuarentena recuperados (S, I, H, R, D, QI, QS, QR).
Vector de datos	Es el vector de datos global que es común a todos los nodos. En nuestro caso se utilizarán los parámetros beta, gamma, proporción de infectados que son hospitalizados y proporción de hospitalizados que se recuperan.

Tabla 1: Elementos utilizados para el funcionamiento del modelo.

Para que se entienda mejor cuál es el funcionamiento tanto de la matriz E como la N, vamos a mostrar qué matrices hemos utilizado y como se interpretan.

	1	2	3	4	5	6	7	8	9	10		1	2	3	4	5	6	7	8
E =	S	1	0	0	0	0	0	0	0	1	1	S	0	6	0	0	0	0	3
	I	0	1	0	0	0	0	0	0	1	1	I	0	0	0	4	0	0	0
	H	0	0	1	0	0	0	0	0	1	0	H	2	0	0	0	0	0	0
	R	0	0	0	1	0	0	0	0	1	1	R	0	0	0	0	4	0	0
	D	0	0	0	0	1	0	0	0	1	0	D	0	0	0	0	0	0	0
	QI	0	0	0	0	0	1	0	0	1	0	QI	0	0	0	0	-4	0	0
	QS	0	0	0	0	0	0	1	0	1	0	QS	0	0	-6	0	0	0	0
	QR	0	0	0	0	0	0	0	1	1	0	QR	0	0	0	0	0	-4	0

La matriz E, como hemos mencionado anteriormente, sirve para seleccionar el compartimento con el que queremos trabajar, por lo que si escogemos la columna 3 estamos seleccionando al grupo poblacional de hospitalizados, mientras que si escogemos la 9 utilizamos todos los compartimentos que hay disponibles.

La matriz N indica cómo movemos a un grupo poblacional de un compartimento a otro. De esta manera, si seleccionamos la columna 1 lo que estamos haciendo es mover a esa muestra poblacional de hospitalizados a defunciones, mientras que si elegimos la columna 7 movemos a los individuos recuperados que se encuentran en cuarentena a simplemente individuos recuperados.

4. Datos de movilidad utilizados para el modelo

Para tener una visión más realista de cómo es el tránsito de personas, para a su vez observar cómo se podría propagar la enfermedad, vamos a utilizar los datos de movilidad de la Comunidad de Madrid que proporciona el INE, ya que nuestro trabajo estará acotado a dicha Comunidad Autónoma.

Los datos que nos suministra el INE son para el periodo que va desde el 16/03/2020 hasta 20/06/2020, en los cuales podemos observar cómo se desarrolló la movilidad de las personas durante la cuarentena. Además, tenemos información acerca de dos días delimitados fuera de la cuarentena, los cuales nos podrían aportar información verídica acerca del tránsito de personas entre regiones en condiciones normales. Estos días son el 01/01/2020 y el 18/11/2019.

Estos datos de movilidad se elaboran a través de la monitorización de los teléfonos móviles de las tres principales compañías telefónicas, como son, según el Ministerio de Economía y Empresa:

Operador	Cuota de mercado Comunidad de Madrid (% del total)
Movistar:	42,8%
Orange:	26,6%
Vodafone:	23,6%

Tabla 2: Cuota de mercado compañías telefónicas Comunidad de Madrid (2019).

Fuente: Ministerio de economía y empresa.

Como se observa, todas estas compañías poseen una cuota de mercado superior al 20%, por lo que se puede considerar más que aceptable para los objetivos del proyecto.

El ámbito poblacional está constituido por los teléfonos móviles de la población residente en la Comunidad de Madrid de los tres operadores mencionados anteriormente. La información aportada se refiere solo a teléfonos móviles, no a todos los dispositivos susceptibles de tener una tarjeta SIM.

El ámbito geográfico es todo el territorio de la Comunidad de Madrid. El territorio se va a dividir en 293 celdas telefónicas.

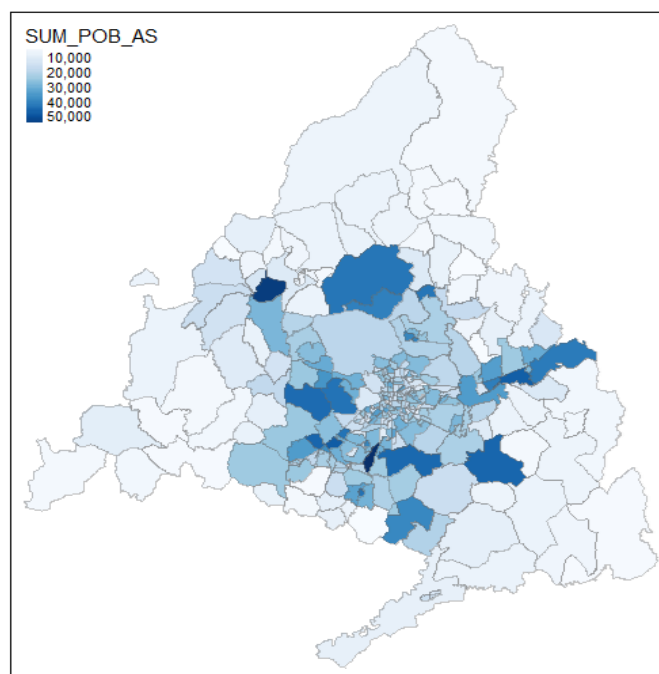


Figura 2: Densidad de población Comunidad de Madrid.

Nota: Cada territorio del mapa representa una celda telefónica, en la que, según la saturación del color, la densidad de población será mayor o menor.

Se debe tener en cuenta que cada uno de los municipios de entre 5.000 y 50.000 habitantes constituye una celda.

Los municipios de más de 50.000 habitantes se pueden desagregar en distritos o barrios, siempre con un umbral superior a esos 5.000 habitantes. En el caso de Madrid se divide el municipio en los 128 barrios que lo componen, debido a que todos ellos superan el umbral de 5.000 habitantes. El tamaño medio de cada celda en la Comunidad de Madrid es de 22.737 habitantes y teniendo en cuenta que la cuota de mercado de las tres compañías telefónicas muestreadas supera el 90% podemos concluir que los datos utilizados suponen prácticamente toda la muestra poblacional.

Una vez dicho esto, es necesario conocer a qué celda telefónica pertenece cada teléfono móvil. Para ello se toman como mínimo 4 días laborales consecutivos de noviembre de 2019. La celda de residencia de cada teléfono móvil es aquella en donde éste se encuentra durante más tiempo entre las 00:01 y las 6:00 de la mañana de estos cuatro días observados.

Una vez se conoce para cada dispositivo su celda de referencia, se intenta determinar la celda de destino cotidiana.

El método es parecido al mencionado anteriormente. Se deberán tomar todas las celdas (incluida la de residencia) donde se encuentra el terminal entre las 9:00 y las 18:00 siempre y cuando se encuentre a ese terminal al menos durante cuatro horas al día y, al menos, en dos días de los cuatro observados anteriormente. Si hay más de una celda, se tomará aquella más frecuente, que no sea la de residencia.

Con esta información ya sabemos qué tipo de datos tenemos para implementarlos en el modelo y como han sido tipificados.

5. Parámetros utilizados

Como hemos mencionado anteriormente, el modelo va a utilizar una serie de parámetros que van a hacer que la simulación funcione de la mejor manera posible. Es muy importante poder ajustar bien estos resultados, dado que el valor de éstos va a hacer que el modelo se acerque con mayor precisión a la situación real de la Comunidad de Madrid. Además, la serie temporal que vamos a utilizar comenzará a partir de septiembre de 2020, debido al desfase de datos comunicados por la Comunidad de Madrid durante el periodo que va del comienzo de la pandemia hasta el verano.

Los parámetros que vamos a utilizar son los siguientes:

- $a \longrightarrow$ Proporción de infectados que son hospitalizados.

$$a = \frac{\text{nº de hospitalizados por Covid}}{\text{nº de infectados}} \quad (1)$$

Este dato es fundamental para que se lleve a cabo correctamente la ecuación de transición que cambia de compartimento una parte de la muestra de infectados a hospitalizados:

$$I \longrightarrow a \times I \longrightarrow H$$

Para el cálculo de este valor, se han recogido los datos que proporciona Datadista. Este organismo es un medio de comunicación especializado en periodismo de investigación, que extrae de los diferentes organismos públicos como el Ministerio de Sanidad, BOE o el Departamento de Seguridad Nacional entre otros, datos diarios acerca del Covid-19 sobre infectados, hospitalizados y defunciones en la Comunidad de Madrid.

Con esta información se puede elaborar una serie semanal que nos va a indicar si la media es estable a lo largo del tiempo.



Gráfico 1: Proporción de infectados que son hospitalizados.

Nota: La línea roja representa la media de la serie.

Gráficamente se puede observar que esta serie no es estacionaria en media. Para verificar esto vamos a realizar el test de Dickey-Fuller Aumentado (ADF), que indica si una serie es estacionaria en media o no.

Augmented Dickey-Fuller Test

```
data: prop_hosp
```

```
Dickey-Fuller = -1.4751, Lag order = 0, p-value = 0.7743
```

```
alternative hypothesis: stationary
```

Este test nos confirma lo mencionado anteriormente, ya que no se rechaza la hipótesis nula de no estacionariedad. Aun así, debemos tener en cuenta que, primero, el periodo de tiempo de la serie elaborada es corto; y segundo, que tanto los datos de infectados y defunciones por Covid-19 dependen del número de test que se realicen. Según los registros de Datadista para parte del periodo de datos que hemos utilizado, y teniendo en cuenta que las pruebas que se registran son PCR, anticuerpos y antígenos, podemos observar como el número de test ha ido creciendo en consonancia con la proporción de infectados que fallecen. Esto demuestra que existe cierta correlación entre las series. Con esto nos podemos hacer una idea de que, si el periodo de tiempo de la muestra fuera mayor, y el número de test realizados fuera estable a lo largo del tiempo,

probablemente la serie de proporción de infectados que fallecen sería estacionaria en media. Aunque a largo plazo se podría esperar que la media sufriera un descenso, dado que la vacunación se prioriza a grupos sociales más vulnerables al virus. Además de que el número de infectados va a disminuir de forma significativa, estos individuos portadores del virus tienen menos probabilidad de sufrir complicaciones. También se debe tener en cuenta los medicamentos que se patenten para combatir la enfermedad, que harán que disminuyan los fallecidos por covid-19.

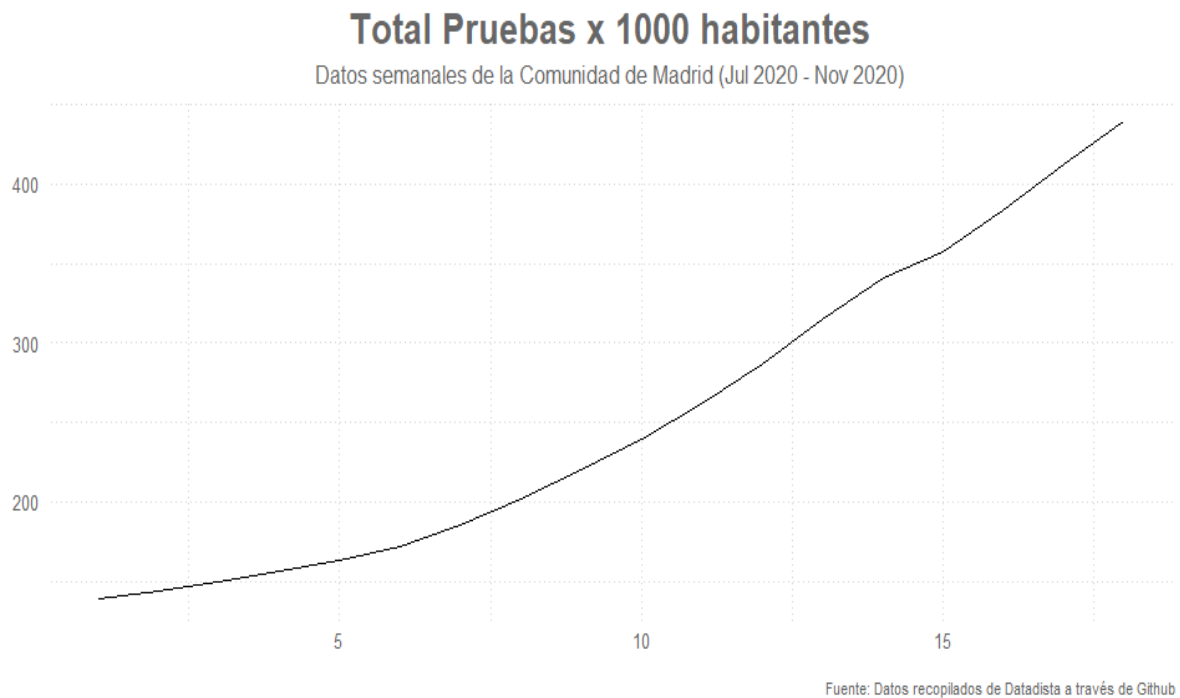


Gráfico 2: N° de pruebas Covid-19 x 1000 habitantes.

Con todo esto, aun no rechazándose la hipótesis nula de no estacionariedad, utilizaremos el valor de su media para el modelo que es $a = 0,096$.

➤ $m \longrightarrow$ Proporción de infectados que fallecen.

$$m = \frac{n^{\circ} \text{ de defunciones por Covid}}{n^{\circ} \text{ de infectados}} \quad (2)$$

Este parámetro se va a utilizar en la ecuación de transición que cambia de compartimento una parte de la muestra de infectados a defunciones.

$I \longrightarrow m \times I \longrightarrow D$

Al igual que en el caso anterior, vamos a proceder a verificar si la media es estable a lo largo del tiempo para así poder utilizarla en el modelo.

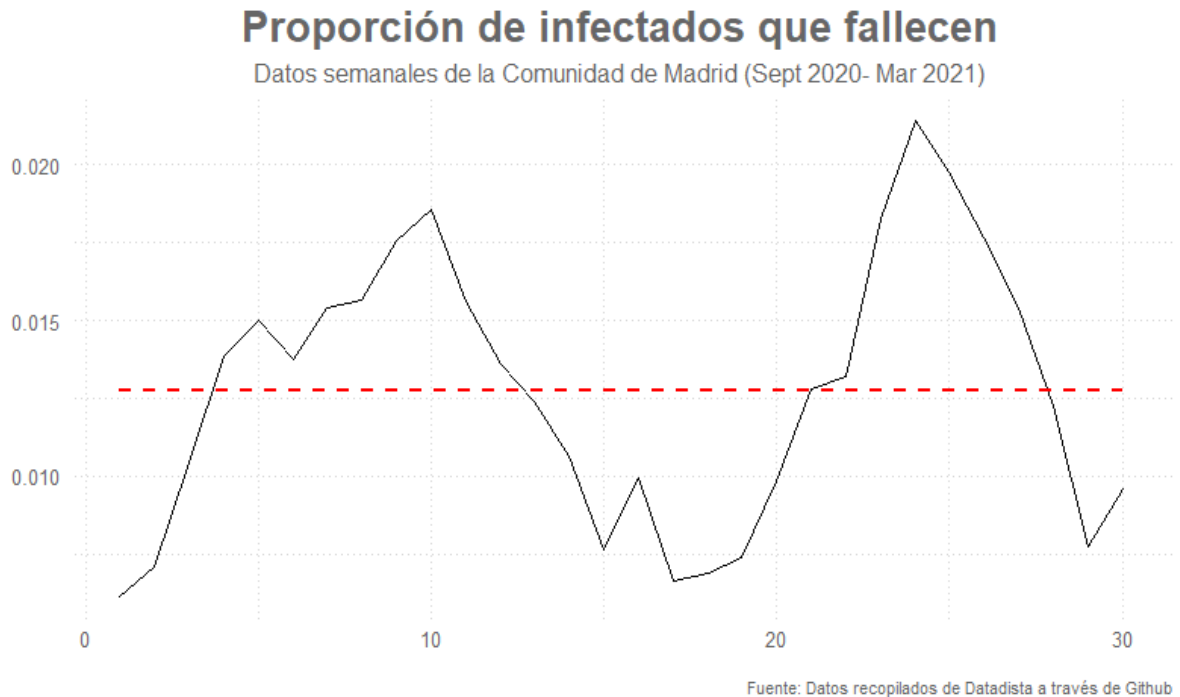


Gráfico 3: *Proporción de infectados que fallecen.*

Nota: La línea roja representa la media de la serie.

En este caso hay dudas sobre si la serie es estacionaria en media; se puede intuir tendencia, por lo que también es conveniente elaborar el test ADF:

Augmented Dickey-Fuller Test

data: prop_def

Dickey-Fuller = -1.6978, Lag order = 0, p-value = 0.6888

alternative hypothesis: stationary

No se rechaza la hipótesis nula de no estacionariedad, lo que deja entrever que la serie pueda tener cierta tendencia.

Aun no rechazándose la hipótesis nula de no estacionariedad, utilizaremos el valor de su media para el modelo, que es $m = 0,013$, por los motivos anteriormente mencionados.

➤ $d \longrightarrow$ Proporción de hospitalizados que se recuperan.

$$d = 1 - \frac{\text{Defunciones por Covid}}{\text{Hospitalizados por Covid}} \quad (3)$$

El valor de d se usará para la ecuación de transición que cambia de compartimento una parte de la muestra de hospitalizados a recuperados.

$$H \longrightarrow d \times H \longrightarrow R$$

Viendo el gráfico de proporción de hospitalizados que se recuperan respecto al tiempo:



Gráfico 4: *Proporción de hospitalizados que se recuperan.*

Nota: La línea roja representa la media de la serie.

Se puede intuir que los datos tampoco van a ser estacionarios en media. El test ADF de nuevo nos va a despejar las dudas sobre la estabilidad de la media.

Augmented Dickey-Fuller Test

```
data: prop_rec
```

```
Dickey-Fuller = -1.5019, Lag order = 0, p-value = 0.764
```

```
alternative hypothesis: stationary
```

En este caso tampoco se va a rechazar la hipótesis nula de no estacionariedad, por lo que la conclusión a la que podemos llegar es la misma que en los casos anteriores. Esta serie se va a ver afectada por el número de test realizados. A largo plazo también se puede esperar que la media de proporción de hospitalizados que se recuperan cambie por los motivos anteriormente mencionados.

Aun así, conociendo las limitaciones que hay en nuestra muestra, también procederemos a utilizar la media de esta serie para el modelo, cuyo valor es $d = 0.889112$.

➤ $\gamma \longrightarrow$ Tasa de recuperación.

$$\gamma = \frac{1}{\text{promedio del periodo infeccioso}} \quad (4)$$

Según el Ministerio de Sanidad, los pacientes que tienen un curso leve de infección, el pico de la carga viral ocurre durante los primeros 5-6 días tras el inicio de síntomas y prácticamente desaparece al décimo día. Dado que la mayor parte de personas que desarrolla Covid-19 es con sintomatología leve, utilizaremos un valor de 5 días para el periodo infeccioso, lo que hace que el parámetro γ sea 0,2.

Con esto concluimos con el desarrollo de todos los parámetros fijos del modelo.

A continuación, vamos a proceder a explicar el último valor que necesitamos, que por sus características es el más difícil de estimar.

➤ $\beta \longrightarrow$ Tasa de infección

El valor de este parámetro dependerá de si la enfermedad es más o menos contagiosa e indica la probabilidad de que una persona susceptible sea infectada cuando entra en contacto con un individuo portador del virus.

El cálculo de este dato se lleva a cabo a través de la siguiente fórmula:

$$Ro = \frac{\beta}{\gamma} \quad (5)$$

Para conseguir el valor de β es necesario conocer la tasa básica de reproducción del virus (Ro). Este indicador es especialmente relevante porque expresa la capacidad de contagio de la enfermedad con la que queramos trabajar. Formalmente se define como el número medio de individuos que se infectan a partir de un infectado. De esta manera, Ro nos indica cuántas veces es mayor o menor la velocidad de transmisión de la enfermedad con respecto a la de recuperación. En el caso de que $Ro < 1$ la infección desaparecerá en un periodo de tiempo relativamente corto; pero si $Ro > 1$ se propagará la infección, por lo que cuanto mayor sea el valor de este parámetro, más rápido progresará la enfermedad.

Como hemos mencionado anteriormente Ro , y en consecuencia β , es difícil de estimar dado que primero, según va pasando el tiempo, el número de personas que ha pasado el virus y en consecuencia que presenta anticuerpos frente a este es mayor, por lo que el número de nuevos infectados por un solo infectado disminuye; y segundo, medidas como la implantación obligatoria de las mascarillas también hace que este valor varíe.

Antes de darle valores a β veamos cuál es la ecuación de transición que mueve una parte de la muestra de susceptibles a infectados:

$$S \longrightarrow \frac{\beta \times S \times I}{\underbrace{(S + I + R + H + QS + QI + QR)}_{\text{(población total)}}} \longrightarrow I$$

Para comenzar con la estimación de la tasa de reproducción del virus a lo largo del tiempo, vamos a utilizar el paquete de R EpiEstim. Esta librería la ha puesto a disposición la propia OMS para poder estimar el impacto de las políticas de salud pública, utilizando una serie diaria de la incidencia de un país o región. Con esta herramienta y utilizando los datos disponibles de infectados de la Comunidad de Madrid, vamos a poder observar cómo ha ido evolucionando esta tasa a lo largo del tiempo.

Se debe tener en cuenta que el parámetro R_0 , en nuestro caso, varía en el tiempo, por lo que a la hora de simular el modelo plantearemos una serie de escenarios con diferentes valores de este parámetro.

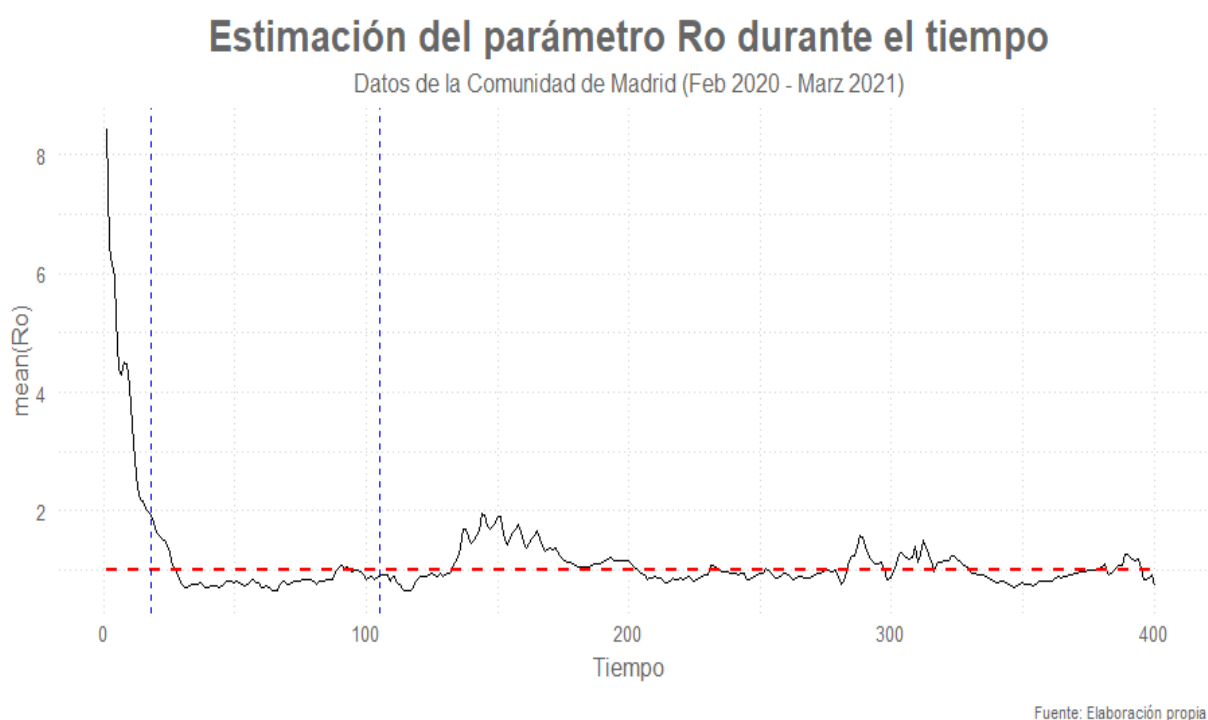


Gráfico 5: Estimación del parámetro R_0 a lo largo del tiempo.

***Nota:** La línea roja del gráfico marca el valor de $R_0 = 1$. Como hemos mencionado anteriormente los valores que fluctúan por encima de ella significan que la epidemia está en proceso expansivo, mientras que, si están por debajo la expansión del virus podría estar controlada.*

Las líneas azules, indican el periodo en el que comenzó el confinamiento domiciliario hasta cuando la Comunidad de Madrid pasó a la fase 2 del proceso de desescalada.

Este gráfico es de suma importancia, dado que muestra evidencia empírica de que medidas que regulan la libre circulación de las personas hacen que la tasa de reproducción del virus sea inferior a 1 y, en consecuencia, que la pandemia se pueda controlar mejor.

Se debe tener en cuenta para el siguiente punto que vamos a estudiar, que los valores de R_0 vienen condicionados por la situación de la comunidad de Madrid y en consecuencia a sus medidas restrictivas, como la implementación de la mascarilla, toque de queda o restricciones a la hora de reunirse grupos de personas.

Teniendo en cuenta esto, cabe decir que, según estudios como el de Juan Ignacio Barrios Arce (mencionado en la bibliografía), el número básico de reproducción promedio del Covid-19 es de 2,5. Este valor se calcula en la situación inicial y dejando que la enfermedad siga su cauce, es decir, este valor promedio se da cuando no existen medidas restrictivas. Este trabajo no va a estar enfocado en esa dirección, sino que a partir de la situación real de la Comunidad de Madrid y de valores reales del parámetro R_0 que se han dado durante la pandemia, estudiaremos diferentes situaciones.

6. Simulaciones

Llegado a este punto, ya tenemos el material suficiente como para empezar a simular el modelo planteando diferentes escenarios. Los parámetros fijos que introduciremos son los mencionados en el punto anterior. En cuando al parámetro β , vamos a plantear 3 situaciones.

En estos escenarios utilizaremos el valor de R_0 más alto (obviando los valores iniciales del gráfico 3.5), más pequeño y $R_0 = 1$ del gráfico 3.5, para observar cómo se desarrollarían las curvas del modelo si mantuviéramos estables esos valores de R_0 a largo plazo. Como hemos explicado anteriormente, el modelo trabaja con el parámetro β , por lo que a partir de esos valores de R_0 calcularemos β .

- $R_0 = 1,958 \longrightarrow \beta = R_0 \times \gamma = 1,958 \times 0,2 = 0,392$
- $R_0 = 0,648 \longrightarrow \beta = 0,1296$
- $R_0 = 1 \longrightarrow \beta = 0,2$

Una vez planteados estos tres supuestos vamos a proceder a simular el modelo partiendo de diferentes situaciones.

La condición inicial del modelo es que la celda 130 del mapa de la Comunidad de Madrid, que pertenece a Pozuelo de Alarcón (distrito 01), va a ser el origen de la epidemia. Se van a introducir 5 infectados en esta celda y a partir de ahí veremos cómo se extiende por toda la Comunidad de Madrid el virus.

El motivo por el que hemos escogido esta celda telefónica es debido a que es una de las que más densidad de población tiene, lo que nos da un escenario propicio para que el virus avance.

6.1. Simulación del modelo sin ningún tipo de evento

Desde la aparición de los primeros casos de gripe en 1918, en las principales ciudades de Europa y en EEUU, los gobernantes de San Luis decidieron llevar a cabo medidas que restringían la libre circulación de las personas, como el cierre de colegios e iglesias o la prohibición de manifestaciones y desfiles. Gracias a esta actuación rápida y contundente se pudieron salvar miles de vidas.

En este punto veremos cómo evoluciona la epidemia en el caso de que no se tomara ningún tipo de medida para frenarla. Estudiaremos las curvas de infectados y hospitalizados para los diferentes valores del parámetro R_0 en un periodo de tiempo de 300 días y valoraremos si la definición de la tasa de reproducción básica del virus se cumple en nuestro caso.

➤ $R_0 = 1,958$

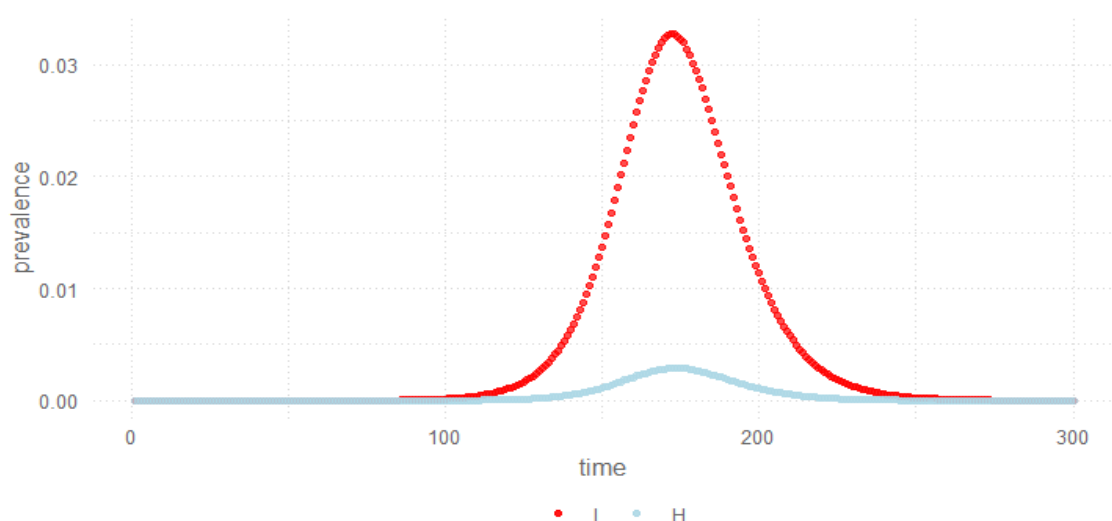


Gráfico 6: *Proporción de Infectados y hospitalizados.*

Nota: Representación de la evolución de la proporción de infectados y hospitalizados a lo largo de los 300 días de la simulación, manteniendo constante el número básico de reproducción del virus en 1,958.

Como observamos, tanto el pico de prevalencia de infectados como el de hospitalizados, llegaría el día 173 de la simulación y tendrían unos valores respectivos del 3,27% y el 0,29%. Para poner en perspectiva estos datos vamos a ver cuáles serían los valores máximos, tanto de infectados como de hospitalizados, en la Comunidad de Madrid, según los datos que nos aporta el Ministerio de Sanidad. En el caso de los infectados, el valor máximo fue de 7738 casos reportados en un día, lo que equivale a un 0,12% de la población, mientras que el valor máximo de hospitalizados fue de 1930, que equivale a un 0,03%.

Esto nos hace ver que, si hubiéramos mantenido ese nivel de R_0 a lo largo del tiempo, tendríamos una situación totalmente insostenible, tanto en número de infectados como en nivel de saturación hospitalaria.

➤ $R_0 = 0,648$

Como hemos mencionado anteriormente, para valores de R_0 menores que 1, la enfermedad irá desapareciendo a lo largo del tiempo. En nuestra simulación, debido a las condiciones iniciales impuestas, no da pie a que se desarrolle el virus.

Si la Comunidad de Madrid hubiera conseguido mantener este valor en un periodo de tiempo largo, es más que probable que la enfermedad se hubiera ido diluyendo. Debemos tener en cuenta que este valor mínimo de R_0 se encuentra en el periodo de cuarentena y sabemos que esta medida es un parche para mitigar el virus durante un periodo de tiempo corto, pero sostenerla a largo plazo sería inviable.

➤ $R_0 = 1$

Por último, falta despejar la duda de si con un R_0 que tome como valor la unidad se desarrollaría la enfermedad. Estableciendo esta condición en nuestro modelo, vemos que el resultado es muy similar al caso anterior: la enfermedad perduraría durante los primeros 5 días de la simulación, sin llegar a propagarse de manera clara.

Esto refuerza la definición del número de reproducción básica del virus mencionado en el apartado anterior de nuestro trabajo. Para valores inferiores o iguales a 1 de R_0 , la enfermedad se deja de propagar, hasta diluirse, por lo que si se pudiera llevar un registro exhaustivo de este parámetro, las autoridades sanitarias tendrían capacidad de tomar medidas más eficaces para así minimizar los daños causados por el Covid-19.

El problema es que para el cálculo de este parámetro se necesita tener datos claros del número de casos diarios, y aquí es donde entran en juego otros factores, como el número de test que se realizan.

Dicho todo esto, en los apartados siguientes utilizaremos únicamente el valor de $R_0 = 1,958$, para que podamos trabajar sobre las diferentes simulaciones.

6.2. Simulación del modelo con evento de cuarentena

El confinamiento domiciliario ha afectado a más de la mitad de la población mundial, y ha provocado que todos los sectores, en especial el sector servicio, reduzcan e incluso frenen en seco su actividad habitual. Esto ha provocado un descenso muy brusco del número de ocupados desde que empezó la pandemia, pasando en el caso de España de 19.804,9 en el segundo trimestre de 2019 a 18.607,2 en el mismo trimestre de 2020, lo que supone un descenso del 6,44% de la tasa de ocupados según el INE.

Si bien existen distintos tipos de cuarentena, ya sea la nacional o la parcial, todas ellas, por lo general, implican un confinamiento severo, además de la suspensión o limitación de las actividades no esenciales, para a continuación ir desescalando las medidas y poco a poco recuperar la normalidad.

La primera cuarentena por Covid-19 tuvo lugar en China, cuando el gobierno ordenó el cierre de la provincia de Hubei el 23 de enero de 2020, que incluía a Wuhan, ciudad de origen de la pandemia. A partir de aquí, según se fue extendiendo el virus, prácticamente todos los países, incluido España, se unieron a esta política de cuarentenas.

Ya desde el día que se decidió que había que tomar medidas restrictivas para frenar la propagación del virus, ha habido un debate sobre cómo hacerlo sin que nuestra economía se vea demasiado resentida.

La idea de este apartado es plantear una estrategia de cuarentena mediante la cual se consiga frenar las curvas de infectados y de hospitalizados, sin que sea necesario parar de manera absoluta la libre circulación de las personas.

Vamos a partir de la base de que R_0 toma el valor mencionado en el apartado anterior (1,958). Además, el inicio de la cuarentena lo vamos a fijar en el periodo 140, que es cuando la proporción de infectados se está acercando al 1%. A partir de aquí vamos a ver cómo afectaría tanto a la prevalencia de infectados como a la de hospitalizados, estableciendo diferentes periodos para el final de la cuarentena y diferentes proporciones de personas cuarentenadas.

➤ Estrategia de cuarentena en función de la prevalencia de infectados

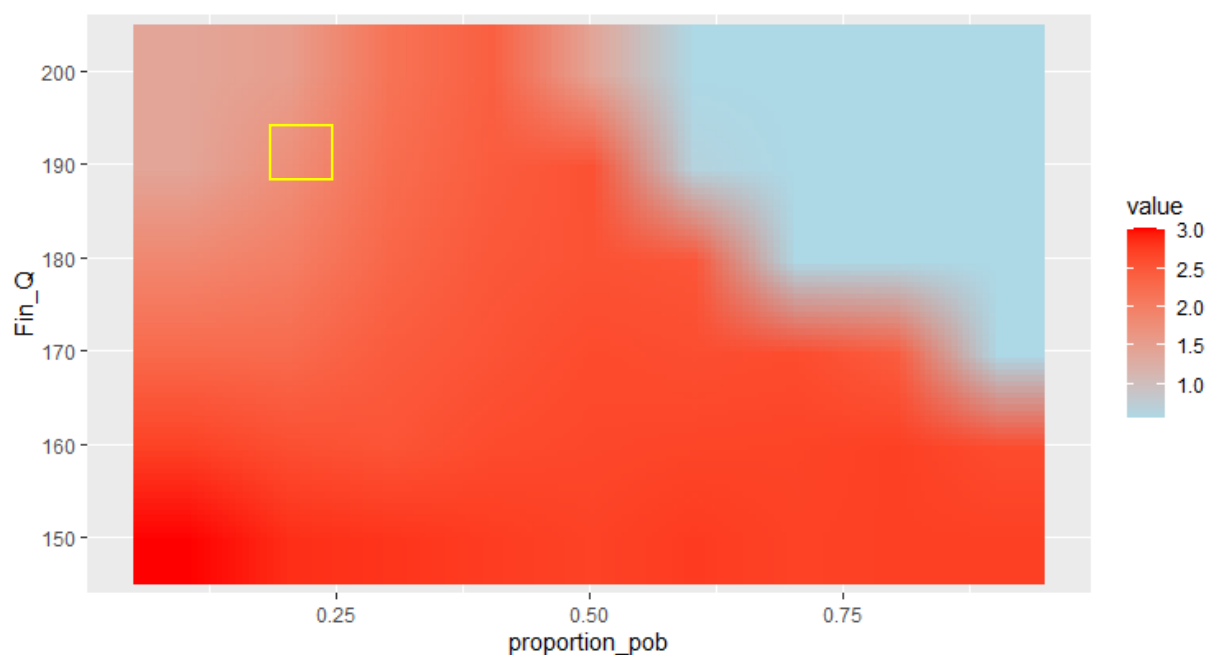


Figura 3: Mapa de calor, picos prevalencia infectados.

Nota: Representación de los valores máximos de la proporción de infectados en todo el periodo de la simulación, en función de la duración de la cuarentena y de la proporción de personas que se mantienen en cuarentena absoluta de toda la Comunidad de Madrid.

Podemos ver de manera clara lo que parecía evidente desde un principio. Según vamos alargando la cuarentena y aumentando la proporción de personas que están en cuarentena el pico de la curva de infectados disminuye.

Una vez dicho eso, intentaremos escoger una estrategia que consiga que el pico de prevalencia de infectados durante toda la simulación no sea superior a un 2%; y que además permita mantener la actividad económica en la medida de lo posible. Evitando los valores de la derecha de la figura 3, los cuales podemos llegar a ellos si el periodo de cuarentena es largo además de que una gran parte de la población se mantenga sin estar en contacto con otras personas, vemos en el recuadro amarillo que si marcamos un periodo de cuarentena de 50 días, y una proporción de cuarentenados del 20% obtendríamos una cifra máxima de prevalencia del 1,76%.

Esta cifra, comparándola con la del apartado 6.1, es claramente inferior, lo que nos da certeza de que manejando datos que realmente reflejen la situación epidemiológica de una zona, podríamos llevar a cabo una estrategia de cuarentena que mejore la curva de infectados.

➤ Estrategia de cuarentena en función de la prevalencia de hospitalizados

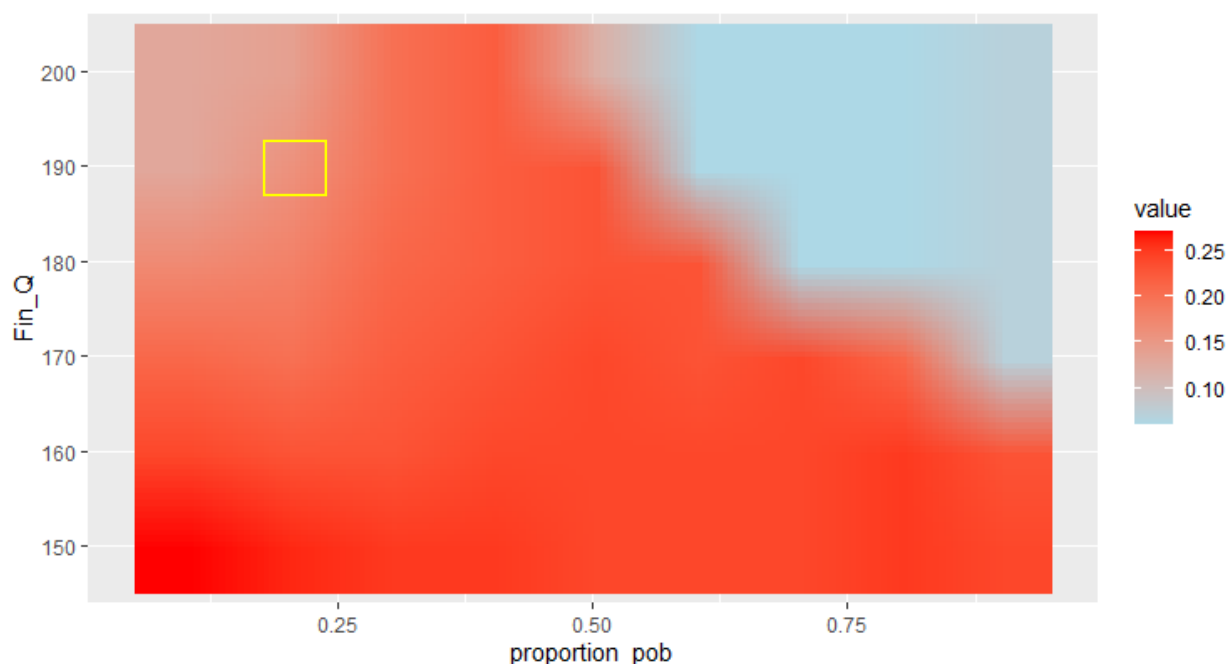


Figura 4: Mapa de calor, picos prevalencia hospitalizados

Nota: Representación de los valores máximos de la proporción de hospitalizados en todo el periodo de la simulación, en función de la duración de la cuarentena y de la proporción de personas que se mantienen en cuarentena absoluta de toda la Comunidad de Madrid.

La epidemia del coronavirus en España ha mantenido al límite los hospitales, que se han visto desbordados principalmente en la primera ola. El umbral que marca riesgo alto es del 15% de ocupación en la UCI, punto que ha superado con creces la Comunidad de Madrid durante las diferentes olas. Debemos tener en cuenta que Madrid tienen una capacidad asistencial por encima de la media en relación con otras comunidades autónomas, hecho que ha propiciado que las consecuencias de la pandemia no sean demasiado desfavorables en comparación con otras comunidades autónomas de la península.

Comunidad Autónoma	Porcentaje de fallecidos por covid-19
Andalucía	1,47 %
Aragón	2,20%
Asturias	3,25 %
C. Valenciana	1,55 %
Cantabria	1,26 %
Castilla la Mancha	1,69 %
Castilla y León	1,96 %
Cataluña	1,42 %
Extremadura	1,77 %

Galicia	1,52 %
La Rioja	1,49 %
Madrid	1,02 %
Murcia	1,30 %
Navarra	1,14 %
País Vasco	1,57 %

Tabla 3: Porcentaje de fallecidos por covid-19 en cada comunidad autónoma entre julio de 2020 y mayo de 2021.

Fuente: Elaboración propia a través de datos recopilados de Datadista.

Según los datos recogidos de la Comunidad de Madrid en la serie temporal de hospitalizados, podemos suponer que el nivel de saturación hospitalaria viene marcado por el pico de hospitalizados durante el periodo estudiado. Este valor lo hemos mencionado anteriormente y es del 0,03%. Debemos tener en cuenta que estamos manteniendo a lo largo del tiempo el valor máximo de R_0 , por lo que en este caso no se va a poder plantear una estrategia viable de cuarentenas que haga que no se llegue a ese nivel de saturación.

Aun así y siguiendo con la casuística del punto anterior, vemos que la mejor estrategia de cuarentena coincide en ambos puntos. Llevándose a cabo una cuarentena de 50 días y una proporción de cuarentenados del 20%, el pico de hospitalizados quedaría en un 0,16%.

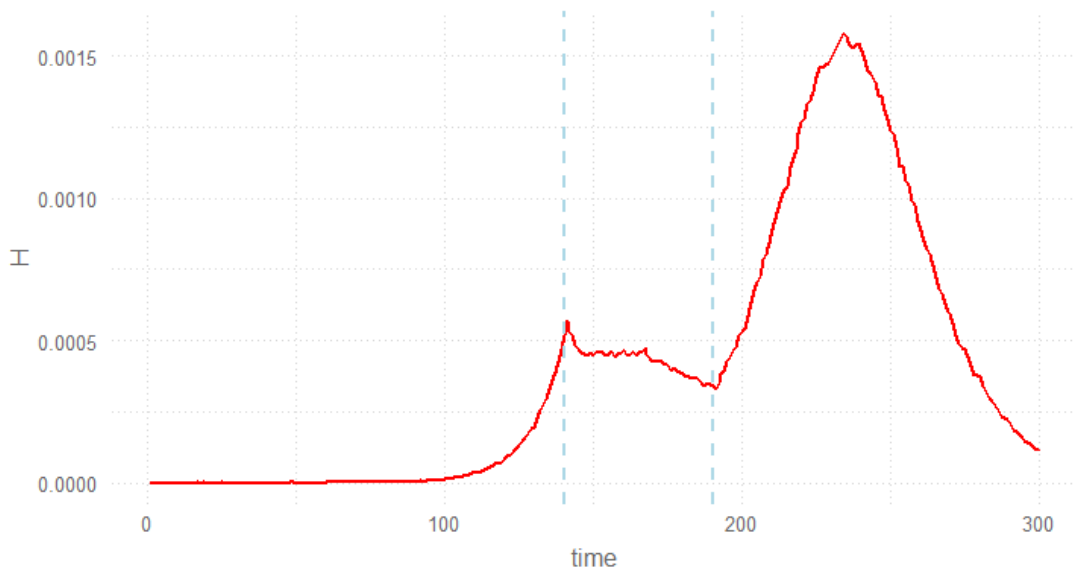


Gráfico 7: Proporción de hospitalizados

Nota: Las líneas azules representan el periodo de cuarentena entre los días 140 y 190.

Viendo la evolución temporal de la proporción de personas que son hospitalizadas por Covid-19 en toda la Comunidad de Madrid, observamos como en el periodo de cuarentena consigue mitigar la curva hasta el punto de dejarla estable alrededor del nivel de saturación mencionado

anteriormente (0,03%). En cuanto esas restricciones de movilidad se desvanecen, permitiendo de nuevo la libre circulación de todo el grupo poblacional, la proporción de hospitalizados vuelve a crecer de manera exponencial, eso sí, llegando a un pico inferior al que llegaríamos si no se tomase ningún tipo de medida.

Con esto queda evidenciado que, primero, las medidas restrictivas de movilidad funcionan a la hora de reducir la presión hospitalaria; y segundo, que con una proporción reducida de la población que quede en cuarentena, se pueden llegar a tener resultados eficientes.

Viendo el resultado del número de hospitalizados para cada una de las celdas telefónicas estudiadas, podemos llegar a una conclusión parecida que en el caso anterior.

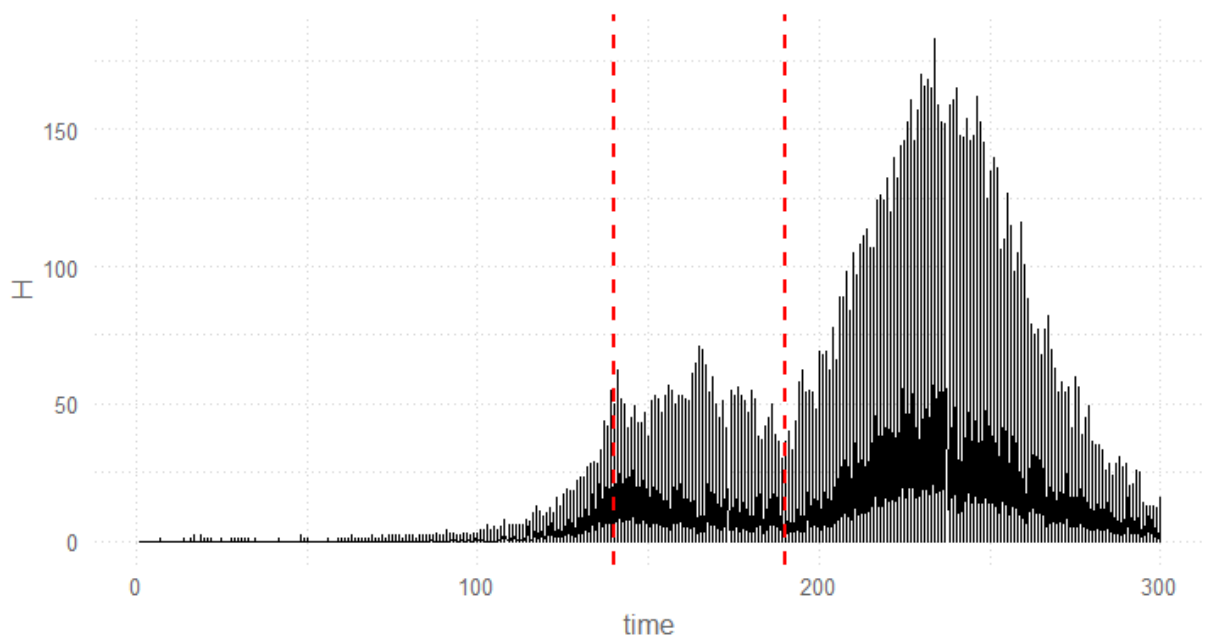


Gráfico 8: Gráfico de densidad hospitalizados.

Nota: Representación del número de hospitalizados para cada celda telefónica en su periodo de tiempo correspondiente. Las líneas rojas representan el periodo de cuarentena entre los días 140 y 190 de la simulación.

Como vemos, la evolución de la serie es calcada a la del gráfico anterior, pero cabe destacar que, aunque el pico se vea reflejado por encima de los 150 hospitalizados en un día, esto no representa a la mayoría de los nodos estudiados. Vemos que, para la mayor parte de los territorios, el pico de hospitalizados no es muy superior a los 50, además de que la variación de esta curva no es tan agresiva cuando se levanta la cuarentena.

En este apartado hemos conseguido demostrar que se pueden llevar a cabo estrategias que a nivel sanitario son eficaces. Queda pendiente compaginar esos planteamientos con un estudio económico acerca de las repercusiones que tendrían las estrategias mencionadas sobre el tejido económico y productivo de la Comunidad de Madrid.

6.3. Simulación del modelo con vacunación

Nadie discute la importancia que tiene la vacunación del Covid-19 para nuestro futuro.

Investigadores de todas las nacionalidades han trabajado duramente para dar con la fórmula de la vacuna y de esta manera evitar que el coronavirus vuelva a hacer mella en nuestra sociedad.

Actualmente tenemos un abanico de vacunas lo suficientemente amplio como para atacar el problema de raíz, aunque queda un arduo trabajo por delante a la hora de elaborar una estrategia para suministrarlas. Actualmente existen una gran cantidad de países como EEUU, Hungría o Israel que tienen una gran parte de la población vacunada, lo que les ha permitido controlar la pandemia.

En este punto del trabajo vamos a estudiar cómo afectaría la vacunación a las curvas de infectados y hospitalizados. Para ello, vamos a elaborar un evento en nuestro modelo que mueva de compartimento a una parte poblacional de susceptibles a recuperados.

Las condiciones iniciales de este evento son que la vacunación empieza el día 120 de la simulación y cada día se vacunará a un 1% de la población, hasta que todo el mundo esté inmunizado. Debido a la complejidad de la estrategia que se está llevando a cabo en España, es imposible emular esa situación, por lo que debemos tener en cuenta que, primero, nuestro modelo no tiene en cuenta los grupos de edad a los que se les suministra la vacuna; y segundo, la vacuna sería de una sola dosis y desde el momento en el que se le suministra, este grupo poblacional queda inmunizado.

Partiendo de esto, vamos a comparar la evolución de las curvas de infectados y hospitalizados entre el modelo sin ningún tipo de evento y el que implementa la vacunación.

➤ Prevalencia infectados

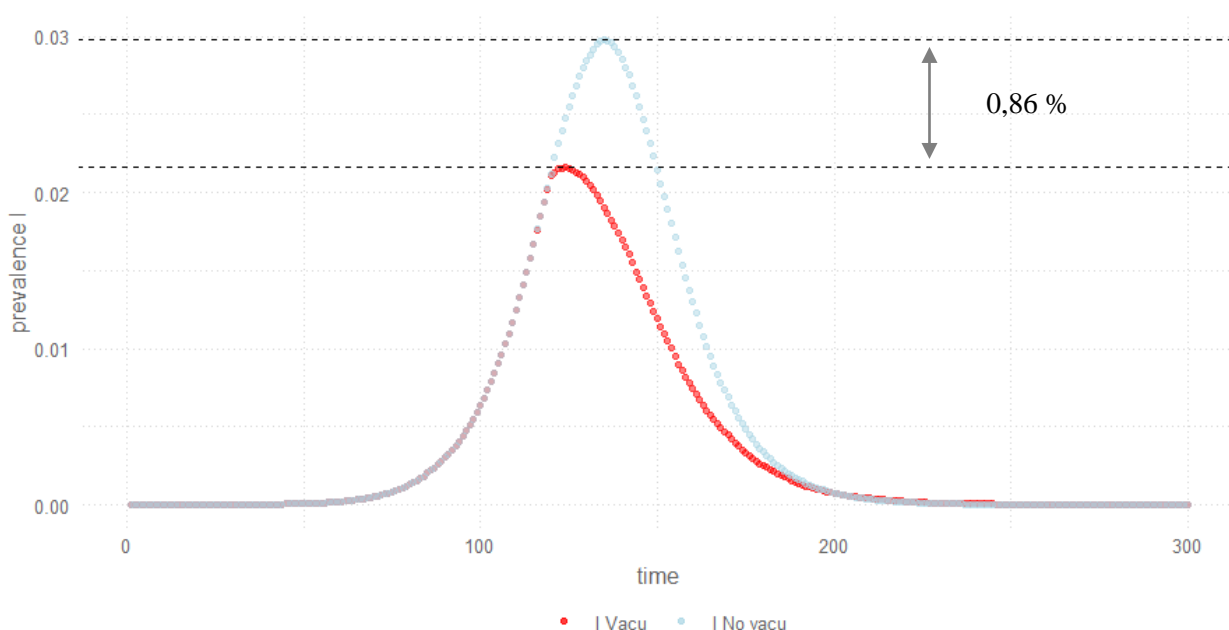


Gráfico 9: Proporción de infectados con evento de vacunación y sin él.

Nota: Las líneas negras marcan los picos de cada curva de infectados.

Podemos observar, como era de esperar, que la vacunación reduce el pico de infectados. Concretamente la variación de esta curva es del -27,4 %, además, el punto máximo se produce antes.

En el modelo con vacunación el pico de infectados llega el día 124 de la simulación, es decir, en nuestro caso con un 4 % de la población vacunada la curva de infectados ya empieza a descender. En cuanto al pico llega 9 días antes que en el modelo sin vacunación.

➤ Prevalencia hospitalizados

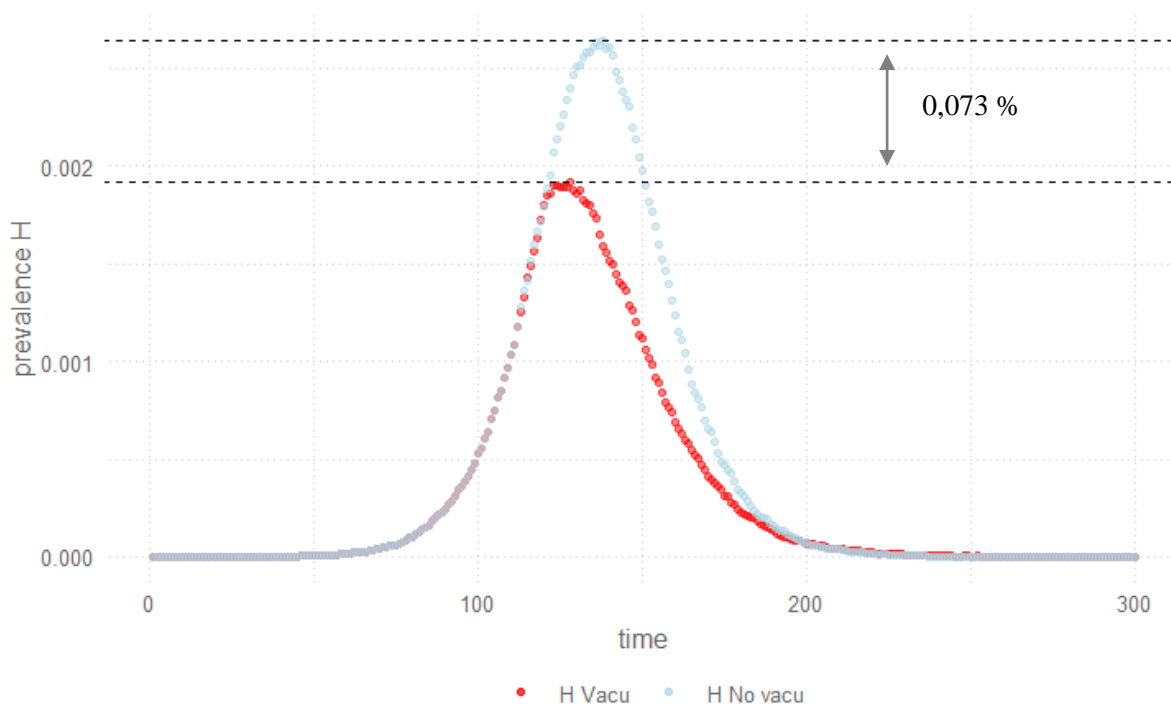


Gráfico 10: Proporción de hospitalizados con evento de vacunación y sin él.

Nota: Las líneas negras marcan los picos de cada curva de hospitalizados.

En este caso, el pico de hospitalizados llega 4 días después que el de infectados, lo que se debe al tiempo que transcurre desde que contraes la enfermedad hasta que desarrollas síntomas graves.

La variación de la curva es prácticamente la misma que en el caso de infectados. Con este ritmo de vacunación conseguimos reducir el pico de hospitalizados un 27,5 % y con un 8% de población inmunizada debido a la vacuna vemos que esta curva empieza a descender.

Esta información es de gran utilidad para las autoridades sanitarias, ya que permite predecir en qué punto las curvas, tanto de infectados como de hospitalizados, empiezan a retroceder y de esta manera plantear medidas restrictivas más laxas para los días venideros.

Ya hemos demostrado que con buena información es posible predecir el punto en el que la vacuna hace que las curvas cambien de tendencia. Queda pendiente el estudio de los costes en cuestión de defunciones y hospitalizados por día en función de los días que se retrase la administración de la vacuna.

7. Conclusión y propuestas de futuro

Emular las características de la pandemia en la Comunidad de Madrid es una tarea ardua, aun así, en este trabajo hemos intentado demostrar que se puede hacer un pequeño esbozo de cómo puede desarrollarse la epidemia, en función de las diferentes medidas sanitarias que se tomen.

Por supuesto, existen limitaciones, ya que primero, al incluir la movilidad de las personas en nuestro modelo solo hemos encontrado datos relativos a dos días. Esta muestra es pequeña para interpretar la movilidad real de las personas fuera de la cuarentena.

En cuanto a los parámetros, su elaboración depende única y exclusivamente, como ya sabemos, de los datos que suministra la Comunidad de Madrid al Ministerio de Sanidad. Estos datos ya hemos comentado en apartados anteriores, hay que tomarlos con cautela, debido al desfase de datos al comienzo de la pandemia y a los casos que no quedan registrados por falta de test.

Conociendo con detalle todos los datos, incluido el número básico de reproducción del virus (R_0), que es el parámetro que más repercute en el modelo, podemos ser capaces de predecir la evolución de todas las curvas y de cómo afectarían las medidas restrictivas a éstas.

El objetivo de este trabajo ha sido plantear una metodología que pueda ayudar a mejorar la toma de decisiones acerca de las medidas que se pueden aplicar respecto al coronavirus. Aun así, este modelo tiene la flexibilidad suficiente como abordar el estudio de otras pandemias.

Queda pendiente para futuros trabajos el estudio de cada una de las regiones de la Comunidad de Madrid, comparándolas en función de su densidad de población, estructura social y capacidad de adaptarse a ciertas medidas restrictivas. También queda pendiente el estudio del coste económico que supondrían estas medidas, que a nivel sanitario parecen eficaces, pero realmente desconocemos su impacto en nuestra economía, por lo que convendría hacer un análisis que optimice las medidas restrictivas y de esta manera encontrar un equilibrio entre economía y salud.

8. Referencias bibliográficas

[1] Instituto de investigación Hospital de la Princesa, Unidad de Análisis de Datos, (2021), “Modelo numérico de propagación de Covid-19 en la Comunidad de Madrid”:

<https://www.iis-princesa.org/unidad-de-analisis-de-datos/>

[2] Gillespie, Daniel T. (1976). "A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reactions". Journal of Computational Physics. 22 (4): 403–434.

[3] Gillespie, Daniel T. (1977). "Exact Stochastic Simulation of Coupled Chemical Reactions". The Journal of Physical Chemistry. 81 (25): 2340–2361.

[4] SimInf, (2019), “An R Package for Data-Driven Stochastic Disease Spread Simulations”:

<https://cran.r-project.org/web/packages/SimInf/vignettes/SimInf.pdf>

[5] INE, (2020), “Movimiento de personas por áreas de movilidad”:

https://www.ine.es/covid/covid_movilidad.htm#tablas_resultados

[6] Ministerio de economía y empresa, (2019), “Informe sobre la Sociedad de la Información y las Telecomunicaciones y el Sector TIC y los Contenidos por Comunidades Autónomas Ed. 2019”:

<https://www.ontsi.red.es/sites/ontsi/files/2019-10/Informe%20Comunidad%20de%20Madrid.pdf>

[7] Datadista, (2021), “ccaa_covid19_datos_sanidad_nueva_serie.csv”:

https://github.com/datadista/datasets/blob/master/COVID%2019/ccaa_covid19_datos_sanidad_nueva_serie.csv

[8] Ministerio de Sanidad, (2021), “Información Científica-Técnica, enfermedad por coronavirus, Covid-19”:

<https://www.mscbs.gob.es/profesionales/saludPublica/ccayes/alertasActual/nCov/documentos/ITCoronavirus.pdf>

[9] OMS, (2020), “Estimador Covid-19”: <https://harvardanalytics.shinyapps.io/covid19/>

[10] EpiEstim, (2021), “A demonstration”:

<https://cran.r-project.org/web/packages/EpiEstim/vignettes/demo.html>

[11] EpiEstim, (2021), “Estimate Time Varying Reproduction Numbers from Epidemic Curves”:
<https://cran.r-project.org/web/packages/EpiEstim/EpiEstim.pdf>

[12] Pan American Health Organisation, World Health Organisation, (2021), “Ejercicio de modelaje de covid-19, una guía paso a paso para calcular el Rt con EpiEstim”:

[file:///C:/Users/HP/Dropbox/Mi%20PC%20\(DESKTOP-KE4BQ28\)/Downloads/Guia-paso-a-paso-EpiEstim%20\(Rt\)-es-actualizado.pdf](file:///C:/Users/HP/Dropbox/Mi%20PC%20(DESKTOP-KE4BQ28)/Downloads/Guia-paso-a-paso-EpiEstim%20(Rt)-es-actualizado.pdf)

[13] Dr. Juan Ignacio Barrios Arce, (2020), “El índice Rt: el indicador para administrar la Pandemia del Covid-19”: <https://www.juanbarrios.com/el-indice-r0-el-indicador-para-administrar-la-pandemia-del-covid19/>

[14] INE, (2021), “Total Nacional. Ambos sexos. 16 y más años. Ocupados. Personas.”:
<https://www.ine.es/consul/serie.do?d=true&s=EPA87>

Anexo I

En este anexo mostraremos el código utilizado para la implementación de los datos de movilidad en el modelo.

```
library(igraph)
library(stringdist)
library(FactoMineR)
library(factoextra)
library(corrplot)
library(tidyverse)

mapa<-
readOGR("C:/Users/mario/Dropbox/Mario/movilidad/shapefiles_celdas_marzo2020",
"celdas_marzo_2020",use_iconv=TRUE,encoding="UTF-8")

CRS("+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0")

mapa_df<-as.data.frame(mapa)

datos<-
read_excel("C:/Users/mario/Dropbox/Mario/movilidad/areas_de_movilidad_y_pobla
cion_a_1_ene_2019.xlsx")

class(datos)

nmadr<-grep("Madrid",as.data.frame(datos)$NPRO)

datos_madr<-as.data.frame(datos[nmadr,])

class(datos_madr)

cels_madr<-unique(datos_madr$ID_GRUPO)

ind_madr<-match(cels_madr,mapa$ID_GRUPO)

mapa_madr<-mapa[ind_madr,]

mapa_madr_df<-as.data.frame(mapa_madr)

tm_shape(mapa_madr) + tm_borders(alpha=.4) +
tm_fill("SUM_POB_AS",style="cont",palette="Blues")

mapa_madr_df$NOMBRE_CEL[130]
mapa_madr_df$SUM_POB_AS[130]
mapa_madr_df$SUM_POB_AS
```

```

### mapa_madr
### datos_madr

##### movilidad de las personas (x moviles)
##### datos reales de la cuarentena
##### origen-destino

ficheros<-
dir("C:/Users/mario/Dropbox/Mario/movilidad/datos_disponibles",pattern="csv")

tmp<-grep("FlujosOrigen",ficheros)
fluxorig<-ficheros[tmp]
tmp<-substr(fluxorig,21,25)
tmp<-str_replace_all(tmp,"MAR","_03")
tmp<-str_replace_all(tmp,"ABR","_04")
tmp<-str_replace_all(tmp,"MAY","_05")
tmp<-str_replace_all(tmp,"JUN","_06")
tmp<-paste(substr(tmp,4,5),"_",substr(tmp,1,2),sep="")
tmp<-tmp[1:(length(tmp)-1)]
ind<-sort(tmp,index.return=TRUE)$ix
fluxori_ord<-fluxorig[ind]
fechas<-paste(substr(fluxori_ord,21,22),substr(fluxori_ord,23,25))

#localeToCharset(locale = Sys.getlocale("LC_CTYPE")) # determina el encoding
correcto
#iconv(AP_Direcs,"UTF-8","ISO8859-1")
tt_cua<-length(fluxori_ord)
general_dest_porc=general_dest=general_ind_dest<-list()
for (i in 1:tt_cua) {
# ojo con el encoding
dia<-
read.csv(paste("C:/Users/mario/Dropbox/Mario/movilidad/datos_disponibles","/"
,fluxori_ord[i],sep=""),sep=";",fileEncoding ="ISO8859-1")
flux_porc_list<-list()
flux_dest_list<-list()
flux_dest_ind_list<-list()

```

```

for (j in 1:length(cels_madr)) {
  ind<-grep(cels_madr[j],trimws(as.character(dia$CELDA_ORIGEN)))
  if(length(ind) > 0) {
    pobla<-datos_madr[which(datos_madr$ID_GRUPO==cels_madr[j]),10][1]
    flux_num<-dia$FLUJO[ind]
    flux_dest<-trimws(as.character(dia$CELDA_DESTINO[ind]))
    porc<-round(flux_num/pobla,4)
    flux_porc_list[[j]]<-porc
    flux_dest_list[[j]]<-flux_dest
    flux_dest_ind_list[[j]]<-match(flux_dest,cels_madr)
  }}
  general_dest_porc[[i]]<-flux_porc_list
  general_dest[[i]]<-flux_dest_list
  general_ind_dest[[i]]<-flux_dest_ind_list
}
cuarentena_dest_porc<-general_dest_porc
cuarentena_dest<-general_dest
cuarentena_ind_dest<-general_ind_dest

##### destino-origen

ficheros<-
dir("C:/Users/mario/Dropbox/Mario/movilidad/datos_disponibles",pattern="csv")
tmp<-grep("FlujosDestino",ficheros)
fluxdest<-ficheros[tmp]
tmp<-substr(fluxdest,22,26)
tmp<-str_replace_all(tmp,"MAR","_03")
tmp<-str_replace_all(tmp,"ABR","_04")
tmp<-str_replace_all(tmp,"MAY","_05")
tmp<-str_replace_all(tmp,"JUN","_06")
tmp<-paste(substr(tmp,4,5),"_",substr(tmp,1,2),sep="")
tmp<-tmp[1:(length(tmp)-1)]
ind<-sort(tmp,index.return=TRUE)$ix
fluxdest_ord<-fluxorig[ind]

```

```

fechas<-paste(substr(fluxdest_ord,21,22),substr(fluxdest_ord,23,26))

#localeToCharset(locale = Sys.getlocale("LC_CTYPE")) # determina el encoding
correcto

#iconv(AP_Direcs,"UTF-8","ISO8859-1")

general_orig_porc=general_orig=general_ind_orig<-list()
for (i in 1:tt_cua) {
# ojo con el encoding

dia<-
read.csv(paste("C:/Users/mario/Dropbox/Mario/movilidad/datos_disponibles","/"
,fluxdest_ord[i],sep=""),sep=";",fileEncoding ="ISO8859-1")

flux_porc_list<-list()
flux_orig_list<-list()
flux_orig_ind_list<-list()
for (j in 1:length(cels_madr)) {
ind<-grep(cels_madr[j],trimws(as.character(dia$CELDA_DESTINO)))
if(length(ind) > 0) {
pobla<-datos_madr[which(datos_madr$ID_GRUPO==cels_madr[j]),10][1]
flux_num<-dia$FLUJO[ind]
flux_orig<-trimws(as.character(dia$CELDA_ORIGEN[ind]))
porc<-round(flux_num/pobla,4)
flux_porc_list[[j]]<-porc
flux_orig_list[[j]]<-flux_orig
flux_orig_ind_list[[j]]<-match(flux_orig,cels_madr)
}}
general_orig_porc[[i]]<-flux_porc_list
general_orig[[i]]<-flux_orig_list
general_ind_orig[[i]]<-flux_orig_ind_list
}

cuarentena_orig_porc<-general_orig_porc
cuarentena_orig<-general_orig
cuarentena_ind_orig<-general_ind_orig

```

```
##### grafico
##### grafica para la cuarentena
cent<-51 # Pozuelo de Alarcón
cent2<-269
cent3<-100
mapa_madr_por<-mapa_madr
mapa_madr_por$PORC<-0

#tmap_mode("view")
Pr<-list()
for (j in 1:tt_cua) {
  dest<-data.frame(general_dest[[j]][[cent]],general_dest_por[[j]][[cent]])
  dest2<-data.frame(general_dest[[j]][[cent2]],general_dest_por[[j]][[cent2]])
  dest3<-data.frame(general_dest[[j]][[cent3]],general_dest_por[[j]][[cent3]])
  if( (length(dest) > 0) & (length(dest2)) > 0 & (length(dest3) > 0)) {
    colnames(dest)=colnames(dest2)=colnames(dest3)<-c("ID_GRUPO","PORC")
    dest$ID_GRUPO<-trimws(as.character(dest$ID_GRUPO),"right")
    dest_flux<-merge(mapa_madr,dest,by.x="ID_GRUPO",by.y="ID_GRUPO")
    dest_flux<-dest_flux[which(!is.na(dest_flux$PORC)),]
    dest_flux$PORC<-round(dest_flux$PORC*100,0)

    dest2$ID_GRUPO<-trimws(as.character(dest2$ID_GRUPO),"right")
    dest_flux2<-merge(mapa_madr,dest2,by.x="ID_GRUPO",by.y="ID_GRUPO")
    dest_flux2<-dest_flux2[which(!is.na(dest_flux2$PORC)),]
    dest_flux2$PORC<-round(dest_flux2$PORC*100,0)

    dest3$ID_GRUPO<-trimws(as.character(dest3$ID_GRUPO),"right")
    dest_flux3<-merge(mapa_madr,dest3,by.x="ID_GRUPO",by.y="ID_GRUPO")
    dest_flux3<-dest_flux3[which(!is.na(dest_flux3$PORC)),]
    dest_flux3$PORC<-round(dest_flux3$PORC*100,0)

    origen<-mapa_madr[which(mapa_madr$ID_GRUPO==cels_madr[[cent]]),]
    origen_nomb<-
    as.character(as.data.frame(mapa_madr[which(mapa_madr$ID_GRUPO==cels_madr[[cent]]),])$NOMBRE_CEL)
  }
}
```

```

origen2<-mapa_madr[which(mapa_madr$ID_GRUPO==cels_madr[[cent2]]),]
origen2_nombre<-
as.character(as.data.frame(mapa_madr[which(mapa_madr$ID_GRUPO==cels_madr[[cen
t2]]),])$NOMBRE_CEL)
origen3<-mapa_madr[which(mapa_madr$ID_GRUPO==cels_madr[[cent3]]),]
origen3_nombre<-
as.character(as.data.frame(mapa_madr[which(mapa_madr$ID_GRUPO==cels_madr[[cen
t3]]),])$NOMBRE_CEL)

##### surrogados para dias normales fuera
confinamiento

##### referencia de noviembre de 2019

##### origen-destino

nov<-
read.csv(paste("C:/Users/mario/Dropbox/Mario/movilidad/datos_disponibles", "/"
,"FlujosOrigen100+_M1_NOV.csv", sep=""), sep=";", fileEncoding = "ISO8859-1")
class(nov)

# hago los surrogados del movimineto de personas

tt_sur=800

std=0.1

nov_dest_porc=nov_dest=nov_ind_dest<-list()
for (kk in 1:tt_sur) {
nov_porc_list<-list()
nov_dest_list<-list()
nov_dest_ind_list<-list()
for (j in 1:length(cels_madr)) {
ind<-grep(cels_madr[j], trimws(as.character(nov$CELDA_ORIGEN)))
if(length(ind) > 0) {
pobla<-datos_madr[which(datos_madr$ID_GRUPO==cels_madr[j]),10][1]
#flux_num<-nov$FLUJO[ind]

flux_num<-
round(rnorm(length(nov$FLUJO[ind]), nov$FLUJO[ind], nov$FLUJO[ind]*std))

flux_dest<-trimws(as.character(nov$CELDA_DESTINO[ind]))

porc<-round(flux_num/pobla,4)

nov_porc_list[[j]]<-porc

```



```

nov_dest_list[[j]]<-flux_dest
nov_dest_ind_list[[j]]<-match(flux_dest,cels_madr)
}}
nov_dest_porc[[kk]]<-nov_porc_list
nov_dest[[kk]]<-nov_dest_list
nov_ind_dest[[kk]]<-nov_dest_ind_list
}
surrogados_dest_porc<-general_dest_porc<-nov_dest_porc
surrogados_dest<-general_dest<-nov_dest
surrogados_ind_dest<-general_ind_dest<-nov_ind_dest

##### destino-origen

nov<-
read.csv(paste("C:/Users/mario/Dropbox/Mario/movilidad/datos_disponibles","/"
,"Flujosdestino100+_M1_NOV.csv",sep=""),sep=";",fileEncoding ="ISO8859-1")
# hago los surrogados del movimineto de personas

nov_orig_porc=nov_orig=nov_ind_orig<-list()
for (kk in 1:tt_sur) {
nov_porc_list<-list()
nov_orig_list<-list()
nov_orig_ind_list<-list()
for (j in 1:length(cels_madr)) {
ind<-grep(cels_madr[j],trimws(as.character(nov$CELDA_DESTINO)))
if(length(ind) > 0) {
pobla<-datos_madr[which(datos_madr$ID_GRUPO==cels_madr[j]),10][1]
flux_num<-
round(rnorm(length(nov$FLUJO[ind]),nov$FLUJO[ind],nov$FLUJO[ind]*std))
flux_orig<-trimws(as.character(nov$CELDA_ORIGEN[ind]))
porc<-round(flux_num/pobla,4)
nov_porc_list[[j]]<-porc
nov_orig_list[[j]]<-flux_orig
nov_orig_ind_list[[j]]<-match(flux_orig,cels_madr)
}}
nov_orig_porc[[kk]]<-nov_porc_list

```

```

nov_orig[[kk]]<-nov_orig_list
nov_ind_orig[[kk]]<-nov_orig_ind_list
}
surrogados_orig_por<-general_orig_por<-nov_orig_por
surrogados_orig<-general_orig<-nov_orig
surrogados_ind_orig<-general_ind_orig<-nov_ind_orig

```

Anexo II

En este anexo veremos el código que hemos utilizado para calcular el parámetro Ro a lo largo del tiempo:

```

library(EpiEstim)
library(ggplot2)

Ro_mad <- estimate_R(mad_infects$Casos, method = "parametric_si",
                     config = make_config(list(
                         mean_si = 4.8,
                         std_si = 2.3)))

## Los datos de mean_si y std_si se refieren al intervalo de tiempo que
## existe entre el momento en el que se desarrollan síntomas en un caso y cuando
## se ha tenido el contacto con la persona infectada. Asumiendo una distribución
## gamma, la OMS nos proporciona los datos de 4,8 días de media y una desviación
## típica de 2,3 días.

Ro <- Ro_mad$R

mean_Ro <- as.data.frame(Ro$`Mean(R)` )

mean_Ro$Time <- 1:400
mean_Ro$normal_Ro <- 1

colnames(mean_Ro)[1] <- "mean_Ro"

## Representamos la evolución del parámetro Ro

ggplot(mean_Ro, aes(x = Time, y = mean_Ro)) +
  geom_path() + theme_elegante() +
  geom_line(aes(y = normal_Ro), col = "red", size = 0.8, linetype = 2) +
  geom_vline(xintercept = c(18, 105), col = "lightblue", linetype = 2) +
  theme(plot.title = element_text (face = "bold")) +
  labs(x = "Tiempo", y = "mean(Ro)",

```

```

title = "Estimación del parámetro Ro durante el tiempo",
subtitle = "Datos de la Comunidad de Madrid (Feb 2020 - Marz 2021)",
caption = "\n Fuente: Elaboración propia")

```

Anexo III

En el último anexo vamos a mostrar el código que hemos utilizado para la elaboración del modelo. Este proceso es la base de todo el trabajo realizado:

```

library(tidyverse)
library(SimInf)

tt <- 1:300

exists("tt") #Es el tiempo total de la simulación, debe estar definido antes
print(tt)

centro=130 # Origen
ncities<-dim(mapa_madr)[1]
tmp<-mapa_madr$SUM_POB_AS # 293 celdas
n <- sum(tmp)
## condiciones iniciales
ndat<-length(tmp)

u0<-cbind(tmp,0,0,0,0,0,0,0)
u0[centro,]<-c(tmp[centro]-5,5,0,0,0,0,0,0) # 5 infectados
#u0<-cbind(tmp-50,50,0,0,0,0,0) # Infectado en cada pueblo

E<-array(0,dim=c(8,8))
diag(E)<-1 # Esto se selecciona en la columna "select" de eventos
E<-cbind(E,rep(1,8))
colnames(E)<-c("1","2","3","4","5","6","7","8","9")
rownames(E)<-c("S","I","H","R","D","QI","QS","QR")

E1 <- matrix(c(1, 1, 0, 1,0,0,0,0), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "10"))

```

```
E<-cbind(E,E1)
```

```
# Esta columna se selecciona en la columna "shift" de eventos
```

```
N <- matrix(c(0, 0, 2, 0,0,0,0,0), nrow = 8, ncol = 1,
dimnames = list(c("S","I","H","R","D","QI","QS","QR"), "1"))
```

```
##### A partir de aquí es para las cuarentenas
```

```
N1 <- matrix(c(6, 0, 0, 0,0,0,0,0), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "2"))
```

```
N<-cbind(N,N1)
```

```
N2 <- matrix(c(0, 0, 0, 0,0,0,-6,0), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "3"))
```

```
N<-cbind(N,N2)
```

```
N3 <- matrix(c(0, 4, 0, 0,0,0,0,0), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "4"))
```

```
N<-cbind(N,N3)
```

```
N4 <- matrix(c(0, 0, 0, 0,0,-4,0,0), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "5"))
```

```
N<-cbind(N,N4)
```

```
N5 <- matrix(c(0, 0, 0, 4,0,0,0,0), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "6"))
```

```
N<-cbind(N,N5)
```

```
N6 <- matrix(c(0, 0, 0, 0,0,0,0,-4), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "7"))
```

```
N<-cbind(N,N6)
```

```
N7 <- matrix(c(3, 0, 0, 0,0,0,0,0), nrow = 8, ncol = 1, dimnames =
list(c("S","I","H","R","D","QI","QS","QR"), "8"))
```

```
N <- cbind(N, N7)
```

```
##### Ecuaciones de transiciones
```

```
colnames(u0)<-c("S","I","H","R","D","QI","QS","QR")
```

```
transitions <- c("S -> b*S*I/(S+I+R+H+QI+QS+QR) -> I", "I -> g*I -> R", "I ->
a*I -> H", "H -> d*H -> R", "QI->a*QI->H", "QI->g*QI->R", "QI -> g*QI -> QR",
"QI -> a*QI -> H")
```

```
compartments <- c("S","I","H","R","D","QI","QS","QR")
```

```
g_data<-c(b = 0.39, g = 0.2, a=0.096, d=0.89)
```

```

#https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/index.html
#g_data<-c(b = 0.14, g = 0.05, a=0, d=0      #desacoplado, solo SIR

print(g_data[1]/g_data[2]) #Valor parámetro Ro

##### events -----> event time node dest n proportion select
shift

##### Parámetros generales de los eventos:
ini_V <- 120
ini_Q <- 90      # Inicio Cuarentena
fin_Q <- 240     # Fin Cuarentena
prop_pob_q <- 0 # Proporción de población de este compartimiento que pasa a Q
prop_pob_v <- 0.1 # Proporción de población que se vacuna diariamente

tiempo<-rep(seq(1,tail(tt,1)), each=ndat)

##### Esto hace un 11,1% pase de compartimiento H -> D (muertes en el
hospital)
HtoD <- data.frame(event = "intTrans", time = tiempo,
node = seq(1,ndat), dest = 0, n = 0, proportion = 0.11, select = 3, shift =
1)

##### Susceptibles
##### saca un % de la población de susceptibles y los pasa a
quarentine a t = ini_Q
#tiempo1<-rep(20,tail(tt,1))
StoQS <- data.frame(event = "intTrans", time = rep(ini_Q,each=ndat),
node = seq(1,ndat), dest = 0, n = 0, proportion = prop_pob_q, select = 1,
shift = 2)

##### saca % de cuarentenados susceptibles y los vuelve susceptibles
#tiempo1<-rep(20,tail(tt,1))
QStoS <- data.frame(event = "intTrans", time = rep(fin_Q,each=ndat),
node = seq(1,ndat), dest = 0, n = 0, proportion = 1, select = 7, shift = 3)

```

```

##### Infectados

##### Sacar un % de la población de infectados y los pasa a cuarentena
#tiempo1<-rep(20,tail(tt,1))

ItoQI <- data.frame(event = "intTrans", time = rep(ini_Q,each=ndat),
node = seq(1,ndat), dest = 0, n = 0, proportion = prop_pob_q, select = 2,
shift = 4)

##### Sacar % de cuarentenados susceptibles y los vuelve susceptibles
#tiempo1<-rep(20,tail(tt,1))

QItOI <- data.frame(event = "intTrans", time = rep(fin_Q,each=ndat),
node = seq(1,ndat), dest = 0, n = 0, proportion = 1, select = 6, shift = 5)

##### No hace falta "sacarlos" de la cuarentena porque se recuperan o van al
hospital

##### Recuperados

##### Sacar un % de la población de recuperados y los pasa a cuarentena
#tiempo1<-rep(20,tail(tt,1))

RtoQR <- data.frame(event = "intTrans", time = rep(ini_Q,each=ndat),
node = seq(1,ndat), dest = 0, n = 0, proportion = prop_pob_q, select = 4,
shift = 6)

##### Sacar % de cuarentenados susceptibles y los vuelve susceptibles
#tiempo1<-rep(20,tail(tt,1))

QRtoR <- data.frame(event = "intTrans", time = rep(fin_Q,each=ndat),
node = seq(1,ndat), dest = 0, n = 0, proportion = 1, select = 8, shift = 7)

### Evento de vacunación

StoR <- data.frame(event = "intTrans", time = rep(ini_V,each=ndat), node =
seq(1,ndat), dest = 0, n = 0, proportion = prop_pob_v, select = 1, shift = 8)

### Evento de movilidad

tmp<-NULL

transf_ext_out<-NULL

for (kk in tt) {

```

```

nn<-lengths(general_ind_dest[[kk]])
tmp_dest_list<-general_ind_dest[[kk]][-which(nn==0)]
tmp_porc_list<-general_dest_porc[[kk]][-which(nn==0)]
nn1<-nn[-which(nn==0)]
ncities_p<-(1:ncities)[-which(nn==0)]

tmp <- data.frame(event = "extTrans", time = kk, node = rep(ncities_p,nn1),
dest = unlist(tmp_dest_list), n=0, proportion =unlist(tmp_porc_list)/3,
select = 1, shift = 0)

tmp1 <- data.frame(event = "extTrans", time = kk, node = rep(ncities_p,nn1),
dest = unlist(tmp_dest_list), n=0, proportion =unlist(tmp_porc_list)/3,
select = 2, shift = 0)

tmp2 <- data.frame(event = "extTrans", time = kk, node = rep(ncities_p,nn1),
dest = unlist(tmp_dest_list), n=0, proportion =unlist(tmp_porc_list)/3,
select = 4, shift = 0)

transf_ext_out<-rbind(transf_ext_out,tmp,tmp1,tmp2)
}

tmp<-NULL
transf_ext_in<-NULL

for (kk in tt) {
nn<-lengths(general_ind_orig[[kk]])
tmp_orig_list<-general_ind_orig[[kk]][-which(nn==0)]
tmp_porc_list<-general_orig_porc[[kk]][-which(nn==0)]
nn1<-nn[-which(nn==0)]
ncities_p<-(1:ncities)[-which(nn==0)]

tmp <- data.frame(event = "extTrans", time = kk,
node = rep(ncities_p,nn1), dest = unlist(tmp_orig_list), n=0, proportion
=unlist(tmp_porc_list), select = 1, shift = 0)

tmp1 <- data.frame(event = "extTrans", time = kk,
node = rep(ncities_p,nn1), dest = unlist(tmp_orig_list), n=0, proportion
=unlist(tmp_porc_list), select = 2, shift = 0)

tmp2 <- data.frame(event = "extTrans", time = kk,
node = rep(ncities_p,nn1), dest = unlist(tmp_orig_list), n=0, proportion
=unlist(tmp_porc_list), select = 4, shift = 0)

transf_ext_in<-rbind(transf_ext_in,tmp,tmp1,tmp2)
}

```

```

eventos<-rbind(transf_ext_in,transf_ext_out,HtoD)

eventos_Q <-
rbind(transf_ext_in,transf_ext_out,HtoD,StoQS,ItoQI,RtoQR,QStoS,QRtoR,QItoI)

eventos_V <-
rbind(transf_ext_in,transf_ext_out,HtoD,StoQS,ItoQI,RtoQR,QStoS,QRtoR,QItoI,
StoR)

eventos_Q <- na.omit(eventos_Q)
eventos_V <- na.omit(eventos_V)


#eventos<-rbind(transf_ext_rand,StoQ,QtoS_10p)
#eventos<-rbind(transf_ext_rand)
#eventos<-rbind(transf_ext_rand,HtoD) # sin cuarentena
#eventos<-rbind(transf_ext_rand,StoQ,QtoS,HtoD) # dos cuarentenas
#eventos<-rbind(transf_ext_rand,StoQ,QtoS,HtoD,StoQ_2,QtoS_2) # dos
cuarentenas


SIHRDQ <- mparse(transitions = transitions, events=eventos_V, E=E, N=N,
compartments = compartments, gdata = g_data, u0 = u0, tspan = tt)

set.seed(123)

result <- run(model = SIHRDQ, threads = NULL) #threads=NULL usar todos los
procesadores

dat <- trajectory(result)

```