



FACULTADE DE MATEMÁTICAS

Traballo Fin de Grao

Modelización matemática de la dosificación óptima en la administración de fármacos

Imanol Artola Olaizola

2024-2025

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

GRADO DE MATEMÁTICAS

Trabajo Fin de Grado

Modelización matemática de la dosificación óptima en la administración de fármacos

Imanol Artola Olaizola

Julio, 2025

UNIVERSIDAD DE SANTIAGO DE COMPOSTELA

Trabajo propuesto/ Traballo proposto

Área de Coñecemento: Matemática Aplicada
Título: Modelización matemática da dosificación óptima na administración de fármacos
Breve descripción do contido
Tratase de modelar con un sistema de ecuacións diferenciais ordinarias a evolución da concentración do fármaco en pacientes, considerando como parámetro a optimizar a distribución en doses da totalidade do fármaco a recibir en quimioterapia. A optimización farase en base ao seu impacto no crecemento de células tumorais.
Recomendacións
Outras observacións

Índice

Resumen	VIII
Introducción	XI
1. Descripción del modelo matemático	1
1.1. Farmacocinética	1
1.1.1. Distribuciones y derivadas distribucionales	2
1.1.2. Definición del término fuerte del fármaco	4
1.2. Farmacodinámica	6
1.2.1. Obtención de la solución del modelo de Gompertz	8
1.3. Combinación de los modelos de farmacocinética y farmacodinámica	9
2. Validación del modelo	13
2.1. Implementación en MATLAB	13
2.2. Comparación con un resultado de la bibliografía	15
2.3. Gráficas, normas y orden de convergencia	17
2.3.1. Gráficas	18
2.3.2. Normas y errores	19
2.3.3. Orden de convergencia	20
3. Optimización de la distribución de dosis	23

3.1. Introducción y parámetros	23
3.2. Problema de optimización	24
3.2.1. Aproximaciones del problema	25
3.3. Resultados teóricos	26
4. Resolución numérica del problema de optimización	29
4.1. Contexto clínico y farmacológico	29
4.1.1. Fármacos citotóxicos	30
4.1.2. Parámetros usados	30
4.2. Implementación de la optimización en MATLAB	31
4.3. Reproducción de los resultados	32
4.3.1. Tratamiento óptimo para $5/28d$	32
4.3.2. Tratamiento óptimo para $D = 5750mg/m^2$	35
5. Conclusiones	37
Bibliografía	39
I. Código de MATLAB del Capítulo 2	III
I.1. Código de MATLAB de la Sección 2.2	V
I.2. Código de MATLAB de la Sección 2.3	VII
II. Código de MATLAB del Capítulo 4	XI
II.1. Código de MATLAB de la Sección 4.3.1	XI
II.2. Código de MATLAB de la Sección 4.3.2	XIV

Resumen

Este trabajo desarrolla un modelo farmacocinético/farmacodinámico (PK/PD) con el fin de optimizar la administración de fármacos en tratamientos de quimioterapia. El objetivo es encontrar una distribución de dosis que minimice el volumen del tumor, manteniendo constante la cantidad total de fármaco administrado.

Se considera un modelo basado en ecuaciones diferenciales, utilizando una versión modificada del modelo Gompertz, que posteriormente se implementa en MATLAB. Para el modelo propuesto, se contrastan los resultados numéricos con las soluciones exactas y se reproducen resultados de la literatura.

A continuación, se presenta el problema de optimización, con restricciones clínicas, vinculado al modelo descrito. Los resultados teóricos indican que la solución óptima consiste en administrar un mayor número de dosis de forma equitativa, siempre que se cumplan las restricciones impuestas.

Los resultados bibliográficos reproducidos coinciden con las predicciones del modelo, lo que valida su implementación. Se concluye que existen estrategias de tratamiento más eficaces que las utilizadas normalmente, y se subraya la importancia de seguir avanzando hacia aplicaciones clínicas más realistas, destacando el potencial de las herramientas matemáticas en la planificación terapéutica personalizada.

Palabras clave: Farmacocinética/farmacodinámica (PK/PD), optimización de dosis, quimioterapia, temozolomida, MATLAB.

Abstract

This work develops a pharmacokinetic/pharmacodynamic (PK/PD) model to optimize drug administration in chemotherapy treatments. The aim is to find a dosing strategy that minimizes tumor volume while keeping the total amount of drug administered constant.

A differential equation-based model is considered, using a modified version of the Gompertz model, which is then implemented in MATLAB. Here, numerical results are compared with exact solutions and results from the literature are reproduced.

Subsequently, an optimization problem with clinical constraints, linked to the described model, is presented. The theoretical results indicate that the optimal solution is to administer a greater number of doses evenly, provided that the imposed constraints are satisfied.

The reproduced literature results match the model's predictions, validating its implementation. It is concluded that there are more effective treatments strategies than those normally used, and the importance of moving towards more realistic clinical applications is emphasized, highlighting the potential of mathematical tools in personalized therapeutic planning.

Keywords: Pharmacokinetics/pharmacodynamics (PK/PD), dose optimization, chemotherapy, temozolomide, MATLAB.

Introducción

El cáncer representa uno de los principales problemas de salud a nivel mundial. En 2020 se diagnosticaron más de 18 millones de nuevos casos y se registraron cerca de 10 millones de muertes asociadas a ello [16]. En España, solo en 2024 se diagnosticaron 286.664 nuevos casos y en 2022 se produjeron aproximadamente 115.000 fallecimientos relacionados con el cáncer, lo que equivale a uno de cada cuatro fallecimientos en el país [16].

Las previsiones epidemiológicas indican que estos datos van a ir aumentando durante los próximos años, tanto las cifras de incidencia como las de mortalidad [16].

Más allá de la gravedad de las cifras, uno de los principales desafíos del cáncer es que aún no existe una cura definitiva. No obstante, existen algunos tratamientos como las intervenciones quirúrgicas, la radioterapia o la quimioterapia, entre otros [17]. Este trabajo se centra en ésta última, la quimioterapia.

La quimioterapia consiste en el uso de medicamentos específicos para tratar cualquier enfermedad, en este caso, el cáncer. Tiene como objetivos la curación, control y paliación dependiendo del estado y tipo de tumor [14]. A diferencia de otros tratamientos más localizados, la quimioterapia se distingue por el hecho de que el medicamento circula por todo el cuerpo, atacando las células tumorales propagadas por el cuerpo.

Uno de los aspectos más críticos de este tratamiento es la determinación de la dosis adecuada. Si la dosis es demasiado baja, puede no generar el efecto deseado sobre el tumor, pero el cuerpo sí que sufrirá las consecuencias. Al contrario, si la dosis es demasiado grande, sí que tendrá efecto sobre el tumor, pero los efectos adversos sobre el individuo pueden ser graves. Por ello, es esencial encontrar la dosis adecuada para cada paciente.

Para abordar este problema, se recurre al uso de modelos matemáticos aplicados a la farmacología. Estos modelos proporcionan herramientas para describir algunos procesos mediante ecuaciones diferenciales, permitiendo predecir su evolución y optimizar las estrategias. En este caso, se analizan las evoluciones de la concentración del fármaco en el cuerpo y del volumen del tumor. Todo eso se recoge en los modelos farmacocinéticos y farmacodinámicos.

La combinación de ambos modelos proporciona una herramienta fundamental para evaluar la eficacia de un tratamiento, simulando tratamientos sin tener que exponer al paciente a riesgos, lo cual facilita la toma de decisiones.

En este trabajo se desarrolla un modelo matemático PK/PD que describa la dinámica del tratamiento quimioterápico. El objetivo principal es optimizar la distribución temporal de la dosis total del fármaco administrado, con el fin de minimizar el volumen tumoral al finalizar el tratamiento.

Para ello, se construirá el modelo paso a paso, resolviendo las ecuaciones diferenciales asociadas. Posteriormente, se implementará en MATLAB y se validará su comportamiento mediante diferentes comparaciones. Después, se formulará un problema de optimización basado en este modelo, para que, al final, se reproduzcan algunos resultados de la bibliografía utilizando los algoritmos de MATLAB.

De este modo, este trabajo busca aportar, desde una perspectiva matemática, herramientas que puedan contribuir a mejorar la planificación de tratamientos oncológicos basados en quimioterapia.

Capítulo 1

Descripción del modelo matemático

Habiendo múltiples modelos matemáticos que se pueden emplear en diferentes ámbitos, el objetivo de este capítulo es encontrar el modelo adecuado para los objetivos propuestos en el resumen de este trabajo. Para ello, se hace uso de la farmacología.

La farmacología es la ciencia que estudia los fármacos y, entre otras muchas cosas, los efectos que provocan esos fármacos en el organismo. Dentro de la farmacología hay diferentes ramas, y dos de ellas son la farmacocinética y la farmacodinámica.

Este capítulo se centra en explicar qué es cada una de las ramas comentadas, y en mostrar qué modelos y EDOs usa cada uno. Después de deducir las soluciones analíticas de ambos modelos, se exponen las ventajas que hay en combinar ambas partes. Finalmente, se introduce el modelo matemático que se va a utilizar en este trabajo, logrando su solución analítica para terminar el capítulo.

1.1. Farmacocinética

La farmacocinética estudia las concentraciones del fármaco en el cuerpo a lo largo del tiempo. Dentro de este estudio, se incluyen los procesos por los cuales el fármaco se absorbe, se distribuye o se metaboliza. Dicho de una forma coloquial, la farmacocinética estudia lo que el cuerpo le hace al fármaco [9].

Para estudiar todo lo anterior, se considera el siguiente problema de Cauchy [8]:

$$\begin{cases} c'(t) = -\lambda c(t) + u(t), \\ c(0) = 0, \end{cases} \quad (1.1)$$

donde $c(t)$ nos indica la concentración en sangre del fármaco en el cuerpo en el instante t , dada

en mg/L . Además, el problema de Cauchy tiene una condición inicial en el instante $t_1 = 0$, donde el valor de la concentración en ese momento es $0mg/L$. Por otra parte, el parámetro λ es la constante de tasa de eliminación, dada por h^{-1} o d^{-1} , según el caso. En este trabajo, consideramos como unidad de tiempo d^{-1} , es decir, días.

El término $-\lambda c(t)$ quiere decir que el cuerpo está eliminando el fármaco. Asimismo, $c'(t)$ indica la evolución que tiene el fármaco a lo largo del tiempo.

Por último, $u(t)$, representa el perfil de administración del fármaco, es decir, indica cómo entra el fármaco en el cuerpo. Para definir $u(t)$, se supone que el fármaco se administra en N momentos, que puede ser que no sean equiespaciadas en el tiempo. Dicho de otra forma, se administra el fármaco en los tiempos $t_1 < t_2 < \dots < t_N$. En cada momento, se administra una dosis del fármaco, $\{d_i\}_{i=1}^N$, donde d_i será la dosis que se administra en el paciente en el instante t_i , siendo sus unidades mg/L . Vamos a suponer que $t_1 = 0$, es decir, que en el primer instante ya se administra una dosis.

Además, se supone que el fármaco se administra instantáneamente, es decir, que en el instante que se administra, el fármaco se extiende por todo el cuerpo. Sabiendo todo esto, para definir el término $u(t)$, en $mgL^{-1}d^{-1}$, se necesitan las siguientes herramientas [1].

1.1.1. Distribuciones y derivadas distribucionales

Definición 1.1. Sea Ω un conjunto abierto de \mathbb{R}^n . El espacio $C_c^\infty(\Omega)$ es el conjunto de las funciones de $C^\infty(\Omega)$ y de soporte compacto. Es decir, es el conjunto de las funciones que tienen derivadas parciales de todos los órdenes en Ω y que su soporte es un conjunto compacto contenido en Ω . Obsérvese entonces que las funciones $C_c^\infty(\Omega)$ son nulas fuera de Ω . Además, si Ω es un dominio acotado, estas funciones serán nulas sobre su frontera.

Definición 1.2. Sea Ω un conjunto abierto de \mathbb{R}^n . Una *distribución* en Ω es una función lineal continua en $C_c^\infty(\Omega)$. El espacio vectorial que recoge todas las distribuciones de Ω es denotado como $\mathcal{D}'(\Omega)$.

Se denota por $\langle T, \phi \rangle$ la actuación de la distribución T sobre la función ϕ .

Ejemplo 1.3 (Delta de Dirac). Para todo $\phi \in C_c^\infty(\mathbb{R}^n)$, la *distribución Delta de Dirac* es la siguiente:

$$\langle \delta, \phi \rangle = \phi(0), \quad \phi \in C_c^\infty(\mathbb{R}^n). \quad (1.2)$$

Esto indica que es una distribución que se concentra en 0, y que no tiene efecto en cualquier otro punto. Por ello, en general, para considerar la distribución delta concentrada en un punto distinto del cero, se define $\delta(x - a)$ que concentra su acción en el punto a :

$$\langle \delta(x - a), \phi \rangle = \phi(a). \quad (1.3)$$

Definición 1.4. Sea Ω un subconjunto abierto de \mathbb{R}^n y T un elemento de $\mathcal{D}'(\Omega)$. Las derivadas parciales de una distribución T respecto a la variable x_k para todo $k = 1, \dots, n$, es la distribución $\frac{\partial T}{\partial x_k}$ definida así:

$$\left\langle \frac{\partial T}{\partial x_k}, \phi \right\rangle = -\left\langle T, \frac{\partial \phi}{\partial x_k} \right\rangle, \text{ para todo } \phi \in C_c^\infty(\Omega).$$

Este resultado se puede interpretar fácilmente cuando la distribución T es la asociada a una función $f \in C^1(\Omega)$, siendo Ω un dominio, en cuyo caso la derivada en el sentido de las distribuciones coincide con la derivada clásica. En efecto, utilizando la fórmula de Green y teniendo en cuenta que las funciones de $C_c^\infty(\Omega)$ son nulas sobre su frontera,

$$\int_{\Omega} \frac{\partial f}{\partial x_k} \phi dx_k = - \int_{\Omega} f \frac{\partial \phi}{\partial x_k} dx_k, \text{ para todo } \phi \in C_c^\infty(\Omega). \quad (1.4)$$

Observación 1.5. De (1.4) se deduce que cuando la distribución está asociado a una función diferenciable, la fórmula de la Definición 1.4 coincide con la derivada clásica.

Ejemplo 1.6 (Función de Heaviside). Sea $\Omega = \mathbb{R}$. La función de Heaviside es la siguiente:

$$H(x) = \begin{cases} 1 & \text{si } x \geq 0, \\ 0 & \text{si } x < 0. \end{cases} \quad (1.5)$$

Entonces, su derivada en el sentido de las distribuciones, para cada $\phi \in C_c^\infty$, utilizando la Definición 1.4, se obtiene como

$$\left\langle \frac{dH}{dx}, \phi \right\rangle = -\left\langle H, \frac{d\phi}{dx} \right\rangle = - \int_{\mathbb{R}} H \frac{d\phi}{dx} = - \int_0^{\infty} \frac{d\phi}{dx} = \phi(0).$$

Considerando (1.2), concluimos que

$$\frac{dH}{dx}(x) = \delta(x). \quad (1.6)$$

Observación 1.7. La derivada clásica de la función de Heaviside no se puede obtener en todo su dominio, ya que tiene un salto en $x = 0$.

Obsérvese que si el salto de Heaviside se produce en $x = a$, también se verifica que

$$\frac{dH(x-a)}{dx} = \delta(x-a). \quad (1.7)$$

Por otro lado, la interpretación de la distribución Delta de Dirac, desde un punto de vista físico, es un impulso localizado en su punto de actuación, representándose formalmente como:

$$\delta(x) = \begin{cases} 0 & \text{si } x \neq 0, \\ \infty & \text{si } x = 0, \end{cases} \text{ y además, } \int_{-\infty}^{\infty} \delta(x) dx = 1 \quad (1.8)$$

Volviendo a nuestro trabajo, considerando que los puntos de salto de concentración de los fármacos serán los t_i , también consideramos la distribución delta con impulso en estos puntos. Así, la distribución (1.8), permite interpretar que la administración de dosis es instantáneo en los momentos t_i para $i \in \{1, \dots, N\}$. Es decir,

$$\delta(t - t_i) = \begin{cases} 0 & \text{si } t \neq t_i, \\ \infty & \text{si } t = t_i, \end{cases} \quad \text{para } i \in \{1, \dots, N\}. \quad (1.9)$$

1.1.2. Definición del término fuerte del fármaco

Ahora ya, utilizando las distribuciones conocidas en (1.9), podemos escribir la fuente farmacológica, $u(t)$, en forma de sumatorio.

$$u(t) = \sum_{i=1}^N d_i \delta(t - t_i). \quad (1.10)$$

Con esto, se indica que, para un tiempo t general, sólo se va a administrar una dosis siempre y cuando $t = t_i$, ya que, si no, la distribución de delta será 0; además la cantidad de dosis administrada es de valor d_i .

Obsérvese que, gracias a (1.7), $u(t)$ también podría escribirse como:

$$u(t) = \sum_{i=1}^N d_i \frac{dH}{dt}(t - t_i). \quad (1.11)$$

Puesto que H se considera adimensional, las unidades de $\frac{dH}{dt}(t - t_i)$ son d^{-1} . Entonces, sabiendo que las unidades de d_i son mg/L , las unidades de $u(t)$ serán $mgL^{-1}d^{-1}$, tal y como se ha definido al principio.

Sabiendo cómo funciona cada variable, se puede lograr la solución del problema de Cauchy (1.1) definido al principio. Aun así, no podemos usar la derivada clásica para lograr la solución fuerte, sino que debemos utilizar (1.3), y usar la derivada en el sentido de las distribuciones introducida en la Definición 1.4. Por ello, se puede construir una solución fuerte a trozos, para cada subintervalo $I_i = [t_i, t_{i+1})$, ya que en el instante t_i se administra otra dosis, y eso genera una discontinuidad en la cantidad de fármaco en el cuerpo.

Primero de todo, se resuelve la ecuación diferencial del problema, sin tener en cuenta el término de aportación de fármaco, $u(t)$, ya que, éste se utilizará para las condiciones iniciales de cada subproblema. Se supone que $t > t_i$, es decir, que ya hay alguna dosis dentro del cuerpo. Con

esto logramos que $c(t) > 0$, en todo momento y que se pueda la siguiente cadena de expresiones:

$$\begin{aligned}
 c'(t) = -\lambda c(t) &\iff \frac{c'(t)}{c(t)} = -\lambda \Rightarrow \int \frac{dc(t)}{c(t)} = - \int \lambda dt \\
 &\iff \ln(|c(t)|) = \ln(c(t)) = -\lambda t + C \\
 &\iff c(t) = \tilde{C}_i e^{-\lambda t}, \text{ siendo } \tilde{C}_i \text{ una constante real positiva.} \quad (1.12)
 \end{aligned}$$

Entonces, habrá una constante \tilde{C}_i para cada intervalo I_i . Si $t \in I_1 = [t_1, t_2)$, en el instante t_1 , la dosis d_1 está dentro del cuerpo. Como se supone que antes de administrar la primera dosis, la concentración del fármaco es nula, entonces en $t < 0$ $c(t) = 0$. En $t_1 = 0$ se administra la primera dosis, de valor d_1 , por lo que la condición inicial en el intervalo I_1 , será d_1 . Utilizando (1.12), se logra la constante \tilde{C}_1 asociada a la función $c(t)$ en el intervalo I_1 : $c(t_1) = \tilde{C}_1 e^{-\lambda t_1} = d_1$, es decir, $\tilde{C}_1 = d_1 e^{\lambda t_1}$. Entonces, $c(t) = d_1 e^{-\lambda(t-t_1)}$ para cada $t \in I_1 = [t_1, t_2)$.

Si $t \in I_2 = [t_2, t_3)$, en el momento justo cuando $t \uparrow t_2$, en el cuerpo habrá una concentración de valor $e^{-\lambda(t_2-t_1)}$, y justo en $t = t_2$ se añade la dosis d_2 , por lo que la concentración inicial de este subintervalo I_2 será $c(t_2) = d_1 e^{-\lambda(t_2-t_1)} + d_2$. Utilizando esta condición inicial de nuevo en (1.12), se logra la constante \tilde{C}_2 de este intervalo I_2 : $c(t_2) = \tilde{C}_2 e^{-\lambda t_2} = d_1 e^{-\lambda(t_2-t_1)} + d_2$, es decir, $\tilde{C}_2 = d_1 e^{\lambda t_1} + d_2 e^{\lambda t_2}$. Entonces, $c(t) = d_1 e^{-\lambda(t-t_1)} + d_2 e^{-\lambda(t-t_2)}$ para cada $t \in I_2 = [t_2, t_3)$. Utilizando el razonamiento por inducción, se ve claramente que, haciendo lo mismo en cada subintervalo, se puede llegar a la solución general:

$$c(t) = \begin{cases} d_1 e^{-\lambda(t-t_1)} & \text{si } t \in I_1 = [t_1, t_2), \\ d_1 e^{-\lambda(t-t_1)} + d_2 e^{-\lambda(t-t_2)} & \text{si } t \in I_2 = [t_2, t_3), \\ \vdots & \\ d_1 e^{-\lambda(t-t_1)} + d_2 e^{-\lambda(t-t_2)} + \dots + d_{N-1} e^{-\lambda(t-t_{N-1})} & \text{si } t \in I_{N-1} = [t_{N-1}, t_N), \\ d_1 e^{-\lambda(t-t_1)} + d_2 e^{-\lambda(t-t_2)} + \dots + d_{N-1} e^{-\lambda(t-t_{N-1})} + d_N e^{-\lambda(t-t_N)} & \text{si } t \geq t_N. \end{cases} \quad (1.13)$$

En general, en el intervalo I_i para cada $i = 1, \dots, N$, se puede escribir en la siguiente forma:

$$c(t) = \sum_{j=1}^i d_j e^{-\lambda(t-t_j)}. \quad (1.14)$$

Se puede reescribir la expresión de la solución (1.13), utilizando (1.14) y la función de Heaviside introducida en el Ejemplo (1.6). Primero de todo, hay que considerar una traslación de la función (1.5):

$$H(t_i - t) = \begin{cases} 1 & \text{si } t \geq t_i, \\ 0 & \text{si } t < t_i. \end{cases} \quad (1.15)$$

Entonces, logramos reescribir la solución del problema (1.1) como:

$$c(t) = \sum_{i=1}^N d_i e^{-\lambda(t-t_i)} H(t-t_i). \quad (1.16)$$

Esto tiene sentido, utilizando la igualdad (1.6) lograda en el Ejemplo 1.6, $c(t)$ sería solución de (1.1) en el sentido de las distribuciones.

1.2. Farmacodinámica

La farmacodinámica estudia la relación entre la concentración del fármaco y sus efectos biológicos. Coloquialmente hablando, se puede decir que la farmacodinámica estudia lo que el fármaco le hace al cuerpo [9].

Para describir la ecuación diferencial que se va a usar en esta sección, primero de todo, se describe el modelo de una manera más sencilla. En este caso, se usa el modelo Gompertz.

El modelo Gompertz

Se puede decir que el modelo más apropiado va a depender de los detalles del experimento [11]. Sin embargo, a pesar de que exista una amplia variedad de modelos que se pueden utilizar, el modelo Gompertz es ampliamente reconocido como el mejor modelo para este ámbito [2], al ser el modelo más apropiado para analizar el crecimiento de tumores [13], suele dar la mejor descripción de su crecimiento e, incluso, las mejores predicciones [12].

La EDO del modelo Gompertz [3] (Pág. 2) es la siguiente:

$$\frac{dV(t)}{dt} = rV(t) \ln\left(\frac{H}{V(t)}\right), \quad (1.17)$$

donde la variable $V(t)$ representa el volumen tumoral, es decir, la población de células tumorales en el tiempo t , que es dada en dm^3 o, lo que es lo mismo, litros, L . Por otra parte, su derivada, $\frac{dV(t)}{dt}$, expresa la evolución que tiene el volumen tumoral a lo largo del tiempo. Además, el modelo incluye dos parámetros: el primero es r , que es la tasa de crecimiento del tumor, que se da en h^{-1} o d^{-1} . En este trabajo se utilizará d^{-1} . El segundo parámetro que se incluye en (1.17) es H , que define el tamaño máximo que puede tener el tumor, teniendo las mismas unidades que $V(t)$.

Por otra parte, en la ecuación (1.17) tenemos el logaritmo neperiano del cociente de H entre $V(t)$. Sabiendo cuál es la definición de H , es fácil entender que $V(t) \leq H$ para todo t . Esto significa que su división nunca va a ser menor que 1, por lo que su logaritmo siempre será positivo. Cuando el volumen tumoral sea parecido a H , el logaritmo se acercará al 0, lo que

indica, que la evolución del crecimiento va a ralentizarse. Por otro lado, si hay mucha diferencia entre $V(t)$ y H , la evolución que pueda tener el crecimiento, puede ser muy grande.

Asimismo, se considera una condición inicial para el tiempo $t = 0$, denotado como $V_0 > 0$, que representa el volumen inicial del tumor. Utilizando todos los elementos descritos, se obtiene el siguiente problema de Cauchy para describir el crecimiento del tumor si no hay intervenciones externas al mismo:

$$\begin{cases} \frac{dV(t)}{dt} = rV(t) \ln\left(\frac{H}{V(t)}\right), \\ V(0) = V_0. \end{cases} \quad (1.18)$$

Por otra parte, volviendo a la farmacodinámica, el modelo que se suele utilizar es el modelo que se acaba de describir, pero incluyendo algún término de pérdida (o destrucción de tejido tumoral) que va a variar según la farmacodinámica. Éste nuevo término representa la reducción del tamaño del tumor causada por el tratamiento. Entonces, la EDO que representa la reducción del tumor considerando exclusivamente el tratamiento farmacológico es la siguiente:

$$\frac{dV(t)}{dt} = -rV(t) \ln\left(\frac{H}{V(t)}\right) G(c(t)), \quad (1.19)$$

donde $G(c(t))$ representa el tipo de pérdida que se va a considerar, y $c(t)$ es la concentración del fármaco en el cuerpo.

La EDO que se va a utilizar en este caso, va a ser la suma entre el modelo que mide el crecimiento natural del modelo (1.17), y el modelo que mide la reducción del tumor debido al fármaco (1.19):

$$\frac{dV(t)}{dt} = rV(t) \ln\left(\frac{H}{V(t)}\right) (1 - G(c(t))). \quad (1.20)$$

En la bibliografía hay diferentes tipos de pérdidas que se puedan considerar. Las más usadas en la farmacología son la pérdida de Skipper y la de E_{max} (ver [10]). Como la pérdida de E_{max} contempla la saturación del efecto cuando la concentración de fármaco es muy grande, esta función es la más recomendada [8]. Por ello, vamos a utilizar esa pérdida, que se representa como:

$$G(c(t)) = \frac{E_{max}c(t)}{c_{50} + c(t)}, \quad (1.21)$$

donde E_{max} representa la máxima tasa posible de reducción del tumor. Además, c_{50} denota la concentración del fármaco que se necesita para que la reducción del tumor sea del 50% de E_{max} . Ambos parámetros son constantes en el tiempo y positivos. $G(c(t))$ es adimensional. Obsérvese que el límite,

$$\lim_{x \rightarrow \infty} \frac{E_{max}x}{c_{50} + x} = E_{max},$$

y que si en un instante t^* , $c(t^*) = c_{50}$, entonces, $G(c(t^*)) = \frac{E_{max}}{2}$

Como ya se han descrito todas las variables, todas las EDOs, y todos los parámetros, ya se puede describir el problema de Cauchy de la farmacodinámica, utilizando (1.18), (1.20) y (1.21):

$$\begin{cases} \frac{dV(t)}{dt} = rV(t) \ln\left(\frac{H}{V(t)}\right) \left(1 - \frac{E_{max}c(t)}{c_{50}+c(t)}\right), \\ V(0) = V_0. \end{cases} \quad (1.22)$$

1.2.1. Obtención de la solución del modelo de Gompertz

Después de definir el problema de Cauchy (1.22), se puede tratar de obtener la solución. Para ello, y para evitar confusiones, se usa de nuevo la función general $G(c(t))$, tal y como aparece en (1.20), en vez de la función particular definida en (1.21).

$$\frac{dV(t)}{dt} = rV(t) \ln\left(\frac{H}{V(t)}\right) (1 - G(c(t))) \iff \frac{dV(t)}{V(t) \ln\left(\frac{H}{V(t)}\right)} = r(1 - G(c(t))) dt. \quad (1.23)$$

Ahora, se integran los dos términos de la última igualdad en el intervalo $[0, t]$. Primero de todo, véase la integral del segundo término:

$$\int_0^t r(1 - G(c(s))) ds = r \left(t - \int_0^t G(c(s)) ds \right). \quad (1.24)$$

Como se desconoce la función $c(t)$, no se puede resolver la integral, por lo que se deja así indicada.

Ahora, volviendo al primer miembro de la igualdad (1.23), se usa un cambio de variable para poder resolver la integral. El cambio que se propone es el siguiente:

$$v(s) = \ln\left(\frac{H}{V(s)}\right), \quad dv(s) = -\frac{1}{V(s)} dV(s). \quad (1.25)$$

Utilizando esto, se puede resolver más fácilmente la integral:

$$\int_0^t \frac{dV(s)}{V(s) \ln\left(\frac{H}{V(s)}\right)} \stackrel{(1.25)}{=} - \int_0^t \frac{dv(s)}{v(s)} = - [\ln(|v|)]_{s=0}^{s=t} \stackrel{(1.25)}{=} - \left[\ln\left(\left| \ln\left(\frac{H}{V(s)}\right) \right| \right) \right]_{s=0}^{s=t}.$$

Tal y como se ha indicado en el apartado del modelo de Gompertz, el logaritmo del cociente de H entre $V(t)$ no va a ser nunca negativo, por la forma en la que está definida H , por lo que se puede quitar el valor absoluto en la última expresión. Por otra parte, utilizando la Regla de Barrow, y teniendo en cuenta la condición inicial del problema de Cauchy (1.22), se puede apreciar que $V(0) = V_0$, por lo que se hace ese cambio:

$$\int_0^t \frac{dV(s)}{V(s) \ln\left(\frac{H}{V(s)}\right)} = -\ln\left(\ln\left(\frac{H}{V(t)}\right)\right) + \ln\left(\ln\left(\frac{H}{V_0}\right)\right) = -\ln\left(\frac{\ln\left(\frac{H}{V(t)}\right)}{\ln\left(\frac{H}{V_0}\right)}\right). \quad (1.26)$$

Después de hacer las dos integrales, juntando los resultados de (1.26) y (1.24), se deduce de (1.23):

$$\begin{aligned}
\ln\left(\frac{H}{V(t)}\right) &= \ln\left(\frac{H}{V_0}\right) \exp\left(-r\left(t - \int_0^t G(c(s))ds\right)\right) \\
\iff \frac{H}{V(t)} &= \exp\left(\ln\left(\frac{H}{V_0}\right) \exp\left(-r\left(t - \int_0^t G(c(s))ds\right)\right)\right) \\
\iff V(t) &= H \exp\left(-\ln\left(\frac{H}{V_0}\right) \exp\left(-r\left(t - \int_0^t G(c(s))ds\right)\right)\right) \\
\iff V(t) &= H \exp\left(\ln\left(\frac{V_0}{H}\right) \exp\left(-r\left(t - \int_0^t G(c(s))ds\right)\right)\right). \tag{1.27}
\end{aligned}$$

Resumiendo, después de hacer todas las cuentas, la solución del problema de Cauchy (1.22), será (1.27).

1.3. Combinación de los modelos de farmacocinética y farmacodinámica

El modelo combinado PK/PD (farmacocinética/farmacodinámica) puede ayudar a comprender mejor la relación entre la exposición y la respuesta, es decir, la relación entre la farmacocinética y farmacodinámica, y también el cambio entre estas relaciones según qué fármacos se utilizan [9].

En esta última sección del capítulo se combinan los problemas de Cauchy (1.1) y (1.22) y las soluciones obtenidas, introduciendo así otro problema más completo. Es fácil darse cuenta de que las soluciones que se han logrado en las secciones 1.1.2 y 1.2.1 están relacionadas, puesto que el problema de farmacodinámica (1.22) depende de $c(t)$ que justamente está definida en la parte de la farmacocinética por el modelo (1.1). Por ello, el problema de Cauchy del modelo combinado PK/PD es una fusión entre (1.1) y (1.22) [8]:

$$\begin{cases} V'(t) = rV(t) \ln\left(\frac{H}{V(t)}\right) \left(1 - \frac{E_{max}c(t)}{c_{50}+c(t)}\right) & , V(0) = V_0, \\ c'(t) = -\lambda c(t) + u(t) & , c(0) = 0. \end{cases} \tag{1.28}$$

Una vez introducido el problema de Cauchy del modelo combinado vamos a tratar de obtener su solución. En la subsección 1.2.1 se logró la solución de la EDO (1.22), pero hay una parte que no se pudo integrar. Ahora, ya que $c(s)$ está definido por (1.13), su acoplamiento de la EDO (1.22) que incluye la evolución de la concentración, $c(t)$, se puede calcular a través de la integral (1.24), y con ello, la solución del problema de Cauchy (1.28). Se considera la siguiente integral:

$$\int_0^t G(c(s))ds \stackrel{(1.21)}{=} \int_{t_1}^t \frac{E_{max}c(s)}{c_{50} + c(s)} ds, \tag{1.29}$$

dado que tal y como se ha definido, $t_1 = 0$.

Si se toma para $c(s)$ la solución de (1.1) definida por (1.16), la solución de la integral (1.29) para $t \in I_k$, $k = 1, \dots, N$ es,

$$\int_{t_1}^t \frac{E_{max}c(s)}{c_{50} + c(s)} ds = \frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1) \cdots (c_{50} + d_1 e^{-\lambda(t_k - t_1)} + \cdots + d_k)}{(c_{50} + d_1 e^{-\lambda(t_2 - t_1)}) \cdots (c_{50} + d_1 e^{-\lambda(t - t_1)} + \cdots + d_N e^{-\lambda(t - t_k)})} \right). \quad (1.30)$$

Entonces, la integral (1.29) quedaría así para un tiempo t generalizado:

$$\int_{t_1}^t \frac{E_{max}c(s)}{c_{50} + c(s)} ds = \begin{cases} \frac{E_{max}}{\lambda} \ln \left(\frac{c_{50} + d_1}{c_{50} + d_1 e^{-\lambda(t - t_1)}} \right) & \text{si } t \in [t_1, t_2), \\ \frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1)(c_{50} + d_1 e^{-\lambda(t_2 - t_1)} + d_2)}{(c_{50} + d_1 e^{-\lambda(t_2 - t_1)})(c_{50} + d_1 e^{-\lambda(t - t_1)} + d_2 e^{-\lambda(t - t_2)})} \right) & \text{si } t \in [t_2, t_3), \\ \vdots & \\ \frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1) \cdots (c_{50} + d_1 e^{-\lambda(t_{N-1} - t_1)} + d_2 e^{-\lambda(t_{N-1} - t_2)} + \cdots + d_{N-1})}{(c_{50} + d_1 e^{-\lambda(t_2 - t_1)}) \cdots (c_{50} + d_1 e^{-\lambda(t - t_1)} + \cdots + d_{N-1} e^{-\lambda(t - t_{N-1}})})} \right) & \text{si } t \in [t_{N-1}, t_N), \\ \frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1) \cdots (c_{50} + d_1 e^{-\lambda(t_N - t_1)} + \cdots + d_N)}{(c_{50} + d_1 e^{-\lambda(t_2 - t_1)}) \cdots (c_{50} + d_1 e^{-\lambda(t - t_1)} + \cdots + d_N e^{-\lambda(t - t_N)})} \right) & \text{si } t \geq t_N. \end{cases} \quad (1.31)$$

Se usa el método de la inducción para demostrar la igualdad (1.30). Para ello, primero, hay que demostrar que la integral (1.30) es correcta para $t \in I_1$ ($k = 1$). Es decir, tenemos que demostrar que para todo $t \in I_1$, la integral (1.29) es la siguiente:

$$\int_{t_1}^t \frac{E_{max}d_1 e^{-\lambda(s-t_1)}}{c_{50} + d_1 e^{-\lambda(s-t_1)}} ds = \frac{E_{max}}{\lambda} \ln \left(\frac{c_{50} + d_1}{c_{50} + d_1 e^{-\lambda(t-t_1)}} \right). \quad (1.32)$$

Para ello, primero de todo se define un cambio de variable,

$$v(s) = c_{50} + d_1 e^{-\lambda(s-t_1)}; dv(s) = -\lambda d_1 e^{-\lambda(s-t_1)} ds. \quad (1.33)$$

Entonces, la solución se logra de esta manera:

$$\begin{aligned} \int_{t_1}^t \frac{E_{max}d_1 e^{-\lambda(s-t_1)}}{c_{50} + d_1 e^{-\lambda(s-t_1)}} ds &\stackrel{(1.33)}{=} -\frac{E_{max}}{\lambda} \int_{t_1}^t \frac{dv(s)}{v(s)} = -\frac{E_{max}}{\lambda} [\ln(|v(s)|)]_{s=t_1}^{s=t} \\ &\stackrel{(1.33)}{=} -\frac{E_{max}}{\lambda} \left[\ln \left(\left| c_{50} + d_1 e^{-\lambda(s-t_1)} \right| \right) \right]_{s=t_1}^{s=t} \\ &= \frac{E_{max}}{\lambda} \ln \left(\frac{c_{50} + d_1}{c_{50} + d_1 e^{-\lambda(t-t_1)}} \right), \end{aligned}$$

tal y como indica (1.30) para los puntos t de I_1 (ver (1.31) y (1.32)). En estas operaciones, ya que lo de dentro del logaritmo es positivo, es entendible que se puedan quitar los valores absolutos.

Dicho esto, se supone que para $k = j$, $1 \leq j < N$ es cierta la expresión (1.30), es decir, que para todo $t \in I_j$ se cumple,

$$\int_{t_1}^t \frac{E_{max}c(s)}{c_{50} + c(s)} ds = \frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1) \cdots (c_{50} + d_1 e^{-\lambda(t_j - t_1)} + \cdots + d_j)}{(c_{50} + d_1 e^{-\lambda(t_2 - t_1)}) \cdots (c_{50} + d_1 e^{-\lambda(t - t_1)} + \cdots + d_j e^{-\lambda(t - t_j)})} \right), \quad (1.34)$$

y se demuestra que se cumple para $k = j + 1$, es decir, que la integral para todo $t \in I_{j+1}$ es la siguiente,

$$\int_{t_1}^t \frac{E_{max}c(s)}{c_{50} + c(s)} ds = \frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1) \cdots (c_{50} + d_1 e^{-\lambda(t_{j+1}-t_1)} + \cdots + d_{j+1})}{(c_{50} + d_1 e^{-\lambda(t_2-t_1)}) \cdots (c_{50} + d_1 e^{-\lambda(t-t_1)} + \cdots + d_{j+1} e^{-\lambda(t-t_{j+1})})} \right). \quad (1.35)$$

Para ello, basta con aplicar la propiedad aditiva de la integral respecto al dominio y que, por tanto, la integral se puede escribir a trozos del siguiente modo:

$$\int_{t_1}^t \frac{E_{max}c(s)}{c_{50} + c(s)} ds = \int_{t_1}^{t_{j+1}} \frac{E_{max}c(s)}{c_{50} + c(s)} + \int_{t_{j+1}}^t \frac{E_{max}c(s)}{c_{50} + c(s)}. \quad (1.36)$$

Para la primera integral del segundo miembro aplicamos la hipótesis de inducción, a la que tenemos que añadir el resultado de hacer la segunda integral. Para ello, se tiene en cuenta la expresión de $c(s)$ para $t \in I_{j+1}$, y se utiliza el mismo cambio de variable que para el caso $k = 1$:

$$v(s) = c_{50} + d_1 e^{-\lambda(s-t_1)} + \cdots + d_{j+1} e^{-\lambda(s-t_{j+1})}; dv(s) = -\lambda(d_1 e^{-\lambda(s-t_1)} + \cdots + d_{j+1} e^{-\lambda(s-t_{j+1})}) ds,$$

dicho de otra forma,

$$v(s) = c_{50} + c(s); dv(s) = -\lambda c(s) ds, s \in I_{j+1}. \quad (1.37)$$

En consecuencia, la solución de la segunda integral del segundo miembro es:

$$\begin{aligned} \int_{t_{j+1}}^t \frac{E_{max}c(s)}{c_{50} + c(s)} &\stackrel{(1.37)}{=} -\frac{E_{max}}{\lambda} \int_{t_{j+1}}^t \frac{dv(s)}{v(s)} = -\frac{E_{max}}{\lambda} [\ln(|v(s)|)]_{s=t_{j+1}}^{s=t} \stackrel{(1.37)}{=} -\frac{E_{max}}{\lambda} [\ln(|c_{50} + c(s)|)]_{s=t_{j+1}}^{s=t} \\ &= \frac{E_{max}}{\lambda} \ln \left(\frac{c_{50} + d_1 e^{-\lambda(t_{j+1}-t_1)} + d_2 e^{-\lambda(t_{j+1}-t_2)} + \cdots + d_j e^{-\lambda(t_{j+1}-t_j)} + d_{j+1}}{c_{50} + d_1 e^{-\lambda(t-t_1)} + d_2 e^{-\lambda(t-t_2)} + \cdots + d_j e^{-\lambda(t-t_j)} + d_{j+1} e^{-\lambda(t-t_{j+1})}} \right). \end{aligned} \quad (1.38)$$

Aquí, es fácil entender que lo que está dentro del logaritmo es algo positivo, teniendo en cuenta que en esta integral, $t \geq t_j$, por lo que $c(s) > 0$. Por lo tanto, aplicando la hipótesis de inducción a la integral entre t_1 y t_j , y sumando la expresión lograda en (1.38), obtenemos la solución propuesta (1.35), utilizando la propiedad del logaritmo de que la suma de los logaritmos es igual al logaritmo de productos.

De este modo, se ha demostrado que la solución de la integral (1.29) es nuestra hipótesis (1.30) para cada $t \in I_k, k = 1, \dots, N$. En consecuencia, la solución del modelo PK/PD definido en (1.28) es la solución (1.27) lograda en la Sección 1.2, pero poniendo la correspondiente expresión dada en (1.31) sobre cada intervalo I_k , en lugar de la integral.

Capítulo 2

Validación del modelo

Después de describir cada modelo por separado y el modelo completo en el Capítulo 1, es necesario implementar este último en MATLAB. El primer paso será obtener una solución aproximada usando comandos propios de MATLAB. Posteriormente, se implementará la solución exacta obtenida en el capítulo anterior (ver (1.16) y (1.31)).

Este capítulo se centra precisamente en esa implementación en MATLAB. Sin embargo, lo más importante no es solo programarlo, sino validar que la implementación sea correcta. Es fundamental verificar que el código está bien escrito y libre de errores, para que, al llegar a la etapa de optimización, todos los cálculos y procedimientos previos sean fiables.

Todo el código relacionado con este apartado está en el Anexo I. En la Sección 2.1 se explican las dos primeras funciones del Anexo I y se realiza una breve explicación de los demás elementos.

Por otro lado, en la Sección 2.2 se desarrollan los códigos correspondientes a la Sección I.1 del Anexo I, mientras que en la Sección 2.3, se detallan los códigos y las funciones de la Sección I.2 del mismo anexo.

2.1. Implementación en MATLAB

Tal y como se muestra en el Anexo I, todo lo desarrollado en el Capítulo 1 ha sido implementado en MATLAB.

En primer lugar, logramos una aproximación del modelo PK/PD (1.28) usando `ode45`. Se sabe que el comando `dsolve` es una función simbólica y que proporciona la solución exacta de ecuaciones diferenciales. Sin embargo, `dsolve` no es capaz de resolver cualquier tipo de ecuación. Dado que el modelo PK/PD es un modelo no lineal, `dsolve` no consigue resolverlo, por lo que se

utiliza para lograr una aproximación de su solución la función propia de MATLAB, `ode45`. Este comando permite resolver ecuaciones diferenciales ordinarias mediante métodos de Runge-Kutta de orden 4 o 5, proporcionando una solución numérica eficiente.

Como argumento de entrada de la función `pkpd_ode45.m` está la estructura de parámetros `par`, donde dentro se guardan los valores de los parámetros $\lambda, r, E_{max}, c_{50}, T, V_0, c_0$ y H . Por otra parte, los valores de las dosis que se administran y los momentos en los que se administran se guardan en `doses` y `doses_time`, respectivamente. Además, la función tiene el argumento `options`. Si este argumento está vacío, se supone que la tolerancia es la predeterminada. Si no, deberá tener esta información:

```
options = odeset('RelTol', xxxx, 'AbsTol', yyyy)
```

La tolerancia relativa, `RelTol`, controla el error relativo máximo de cada paso, mientras que la tolerancia absoluta, `AbsTol`, controla el error absoluto máximo de cada paso, es decir, controla que el error no sea demasiado grande alrededor de cero.

Por otro lado, como argumento de salida tenemos `T_full` e `y_full` que son el vector de los tiempos evaluados en la simulación y la matriz de dos columnas que recoge los resultados obtenidos de la concentración $c(t)$ (`y_full(:,1)`) y el volumen del tumor $V(t)$ (`y_full(:,2)`). Con esta función se logra una aproximación a la solución del modelo definido en (1.28).

En paralelo, se ha implementado `pkpd_exact.m`, donde se aproximan directamente las soluciones deducidas en el capítulo anterior (1.16) y (1.27) con la solución de la integral lograda en (1.31).

Tanto en la función `pkpd_ode45.m` como en `pkpd_exact.m` hay que tener mucho cuidado con las condiciones iniciales, ya que, las soluciones se logran para cada subintervalo. Entonces, para cada subintervalo, se cambian las condiciones iniciales. En las dos variables, es decir, en la concentración del fármaco y el volumen del tumor, la condición inicial de un intervalo va a ser el último valor conseguido en el intervalo anterior. La distinción es que, en cuanto a la concentración, se incorpora la dosificación correspondiente al inicio de cada intervalo.

Además de lo indicado anteriormente, en la función `pkpd_exact.m` deben tenerse en cuenta dos aspectos adicionales. El primero es que, además de la estructura `par` y los vectores `doses` y `doses_time`, se va a considerar el vector `Th`, correspondiente al mallado de tiempo generado en `pkpd_ode45.m`, ya que esto facilitará realizar las normas correspondientes para el cálculo de errores en los casos de comprobación de la implementación considerados. Por ello, se tomará como argumento de entrada de la función `pkpd_exact.m` el vector `Th` y lo primero que se debe hacer es identificar en el mallado que define `Th` los intervalos de dosificación. Lo segundo que hay que tener en cuenta es que hay un solo argumento de salida, `Ye`, que es una matriz con dos

columnas, una para la solución exacta de la concentración $c(t)$ (obtenida por (1.16)) y la otra para la solución exacta del volumen del tumor, $V(t)$, obtenida a través de (1.31). En esta función no se incluye el vector de los tiempos evaluados como argumento de salida, ya que se usa el vector `Th` obtenido con la función `pkpd_ode45.m`.

Todos los parámetros de nuestro modelo PK/PD están guardados en la estructura `par`, las dosificaciones están en un vector llamado `doses` y los tiempos de dosificación en otro llamado `doses_time`.

2.2. Comparación con un resultado de la bibliografía

Para validar el modelo implementado, primero de todo, vamos a tratar de reproducir un resultado del artículo [7], en particular, el resultado representado con el color negro en su Gráfica 1 en la Figura 1 (reproducida también con la línea en color negro de la Figura 2.1). Todos los resultados se explicarán mejor en el Capítulo 4, por lo que ahora no se van a explicar en profundidad.

El resultado recogido en [7] se logra con $N = 30$ dosis, repartidas en 6 ciclos. Cada ciclo es de 28 días, y se administran las dosis en los 5 primeros días de cada ciclo. Los parámetros que se utilizan para reproducir la Figura 2.1 son las que están definidas en la Tabla 2.1.

Parámetro	Descripción	Valor
N	Cantidad total de dosis	30
T	Periodo del tratamiento	$210d$
V_0	Volumen inicial del tumor	$0,25dm^3 = 0,25L$
λ	Tasa de eliminación del fármaco	$9,242d^{-1}$
H	Tamaño máximo del tumor	$1dm^3 = 1L$
r	Tasa de crecimiento del tumor	$5,51e-3d^{-1}$
E_{max}	Máxima tasa posible de reducción del tumor	60
c_{50}	Conc. del fármaco para que la reducción sea el 50 % de E_{max}	$3,6e-1m^2/L$
d_i	Dosis administrada en el momento t_i	$0,6mg/L$ o $0,8mg/L$
\hat{d}_i	Dosis de la bibliografía para cada momento t_i	$150mg/m^2$ o $200mg/m^2$
σ	BSA	$4e-3m^2/l$

Tabla 2.1: Parámetros correspondientes al caso presentado en [7] y a la Figura 2.1.

Se puede observar que aparecen dos parámetros que no se han definido en este trabajo, pero sí aparecen en la bibliografía: \hat{d}_i y σ . Tal y como se ve en la Figura 2.1, \hat{d}_i son las dosificaciones

que se administran en la bibliografía y σ es la BSA ¹. Hay que remarcar que las dosis que están definidas en la bibliografía son diferentes a las definidas en este trabajo. Aun así, se puede apreciar que, por las coherencias de las unidades, las dosificaciones definidas en este trabajo son el resultado del producto entre las dosis de la bibliografía y la BSA. Por eso, en la tabla, al definir d_i , se puede ver que es el producto entre los dos valores siguientes.

Por otro lado, también se puede observar que hay dos valores diferentes para las dosis. La primera sería para la utilizada en el primer ciclo, y la segunda para los siguientes. Además, el artículo no hace referencia al parámetro H , ya que el resultado obtenido es $V(t)/H$ y el volumen inicial $V_0(t)/H$. Entonces, para reproducir el resultado de la bibliografía, se considera $H = 1dm^3$. Como las dosificaciones están dadas en unidades de mg/L , se asume que el volumen tumoral $V(t)$, el volumen inicial $V_0(t)$ y el tamaño máximo del tumor H se expresan en litros (L o dm^3), garantizando así la coherencia de las unidades del trabajo. En el Capítulo 4 se explica más detalladamente la razón por la que consideran la variable adimensional $V(t)/H$.

Para reproducir en este trabajo la gráfica del artículo [7] se ha usado el programa Engauge Digitizer para obtener los puntos de la curva. Los puntos para reproducir la curva están en un archivo CSV llamado `datos_curva.csv`, y los puntos en los que se administran las dosis están en otro archivo `datos_puntos.csv`.

Sabiendo todo esto, el script `plotter1.m` reproduce las gráficas de la bibliografía. Superpuesta a ella, se representan las logradas utilizando las funciones `pkpd_ode45.m` y `pkpd_exacta.m`. Los momentos en los que se administran las dosis están dibujados con círculos negros para la curva de la bibliografía, círculos azules para la aproximada y asteriscos rojos para la exacta, tal y como se puede ver en la Figura 2.1.

Hay que tener en cuenta que, los datos logrados en Engauge Digitizer no son exactos, por lo que puede incorporar algunos errores. Sin embargo, se puede apreciar en la Figura 2.1 que los gráficos obtenidos con `pkpd_ode45.m` y `pkpd_exact.m` son prácticamente iguales. Dado que a simple vista no se aprecian muy bien las diferencias que hay entre ellas, a continuación se hace una revisión más cuantitativa.

Para ello, se utiliza el archivo `diferencia.m`. En este archivo, primero de todo se interpolan las soluciones del volumen del tumor de la bibliografía, para poder calcular los errores, y hacer una norma L^2 . Esta norma y algunos otros más indicadores mejor definidos en la Subsección 2.3.2.

Los errores que se calculan en `diferencias.m` tratan de comparar la curva de la bibliografía

¹BSA (*Body surface area*) es una fórmula utilizada para calcular el índice cardíaco y estimar el porcentaje de grasa corporal, entre otras aplicaciones. Existen distintas fórmulas según el propósito del cálculo, la edad del paciente, etc. [4]

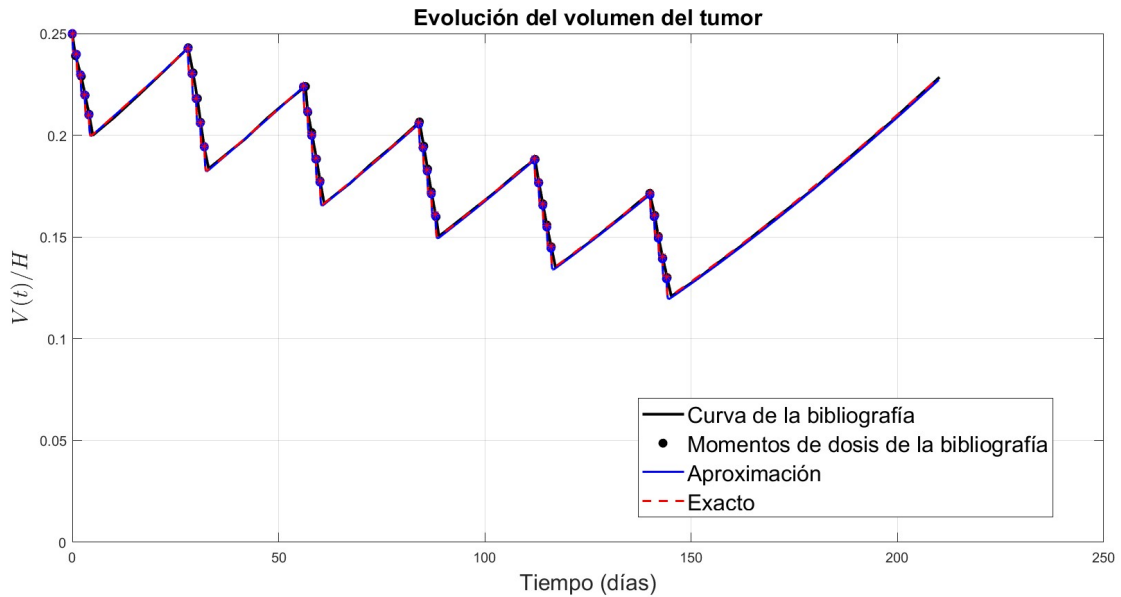


Figura 2.1: Representación de la evolución del volumen del tumor con $N = 30$ dosis de la Gráfica 1 de la figura 1 del artículo [7]. Comparación con las metodologías propias propuestas.

con la curva lograda con `pkpd_ode45.m`, y con la curva lograda con `pkpd_exacta.m`. Las normas L^2 discretas del error absoluto son 0,044261 y 0,041621, respectivamente. Podemos concluir que las curvas logradas se aproximan bastante a la referencia. Además, se deduce que la solución exacta proporciona una mejor aproximación, como era de esperar.

Aun así, se podría ver si el orden de la solución es de 4 o 5, ya que la función `pkpd_ode45.m` utiliza métodos de Runge-Kutta con orden 4 o 5. Pero como este ejemplo tiene demasiados saltos, debido a los momentos en que se administran las dosis, no es tan suave como para que se cumplan las hipótesis de Runge-Kutta, por lo que no se verifica que tenga orden 4 o 5.

2.3. Gráficas, normas y orden de convergencia

Teniendo en cuenta que en el ejemplo anterior no se pudo verificar la orden de la solución, en esta sección se modifican algunos parámetros de la Tabla 2.1 con el objetivo de obtener una solución lo más suave posible y así poder analizar su orden de convergencia.

Para ello, se realizan los siguientes cambios: $H = 100dm^3$, $T = 15d$ y $V_0 = 10dm^3$. Por otra parte, las dosificaciones que se cogen en esta sección son 50, 80 y 80 mg/L , y se administran en los días 0, 6 y 12.

Aunque estos cambios no tienen una lógica física realista, tienen un gran interés desde el

punto de vista matemático, ya que permite analizar con mayor claridad las normas y el orden.

2.3.1. Gráficas

Ya se han explicado los códigos de la solución aproximada, calculada con `ode45`, y de la solución exacta en la Sección 2.1. Con ellas vamos a generar los gráficos de ambas soluciones, y analizar si hay alguna diferencia visualmente. Para ello, se utiliza la función `plotter2.m` para representar los gráficos, y marcar los puntos en los que se administra las dosis.

En este caso, además del gráfico de la evolución del volumen del tumor, se incluye otro gráfico que muestra la concentración del fármaco en el cuerpo. Esto permite identificar, en caso de alguna irregularidad, si el problema está en la solución de la concentración o en la del volumen del tumor. Según la leyenda, la línea roja es discontinua y la azul es continua. En los gráficos de la Figura

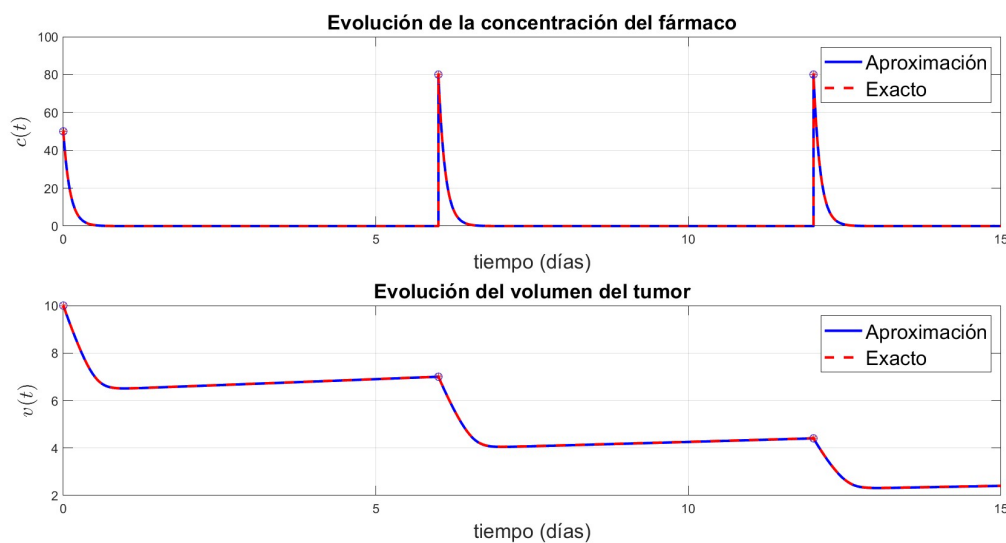


Figura 2.2: Representaciones de la solución exacta y aproximada en la evolución de la concentración del fármaco y en la evolución del volumen del tumor.

2.2, ambas líneas coinciden tan estrechamente que solo se aprecia una única línea rojiazul, lo que sugiere que la aproximación es muy precisa. No obstante, esta coincidencia visual no basta para asegurar rigurosamente la validez de la aproximación. Es decir, es necesario acompañarla con pruebas adicionales.

2.3.2. Normas y errores

Otra forma de validar el modelo es calculando las normas y los errores. En este caso, se evalúan la norma L^∞ discreta del error absoluto y del relativo, así como la norma L^2 discreta del error absoluto.

Antes de continua, es importante aclarar que, de aquí en adelante, los cálculos se realizarán únicamente con el volumen del tumor, ya que es el aspecto que nos interesa.

Error L^∞ absoluto

Este valor es el máximo error absoluto en todos los puntos evaluados. Es decir, la norma L^∞ mide la mayor diferencia entre la solución exacta y la aproximada. Matemáticamente, se define como:

$$\|y_e - y_h\|_{L^\infty} = \max_i |y_e^i - y_h^i|, \quad (2.1)$$

donde y_e es la solución exacta, con componentes y_e^i para cada $i = 1, \dots, M$, y y_h es la solución aproximada, con componentes y_h^i para cada $i = 1, \dots, M$, siendo M la dimensión de la malla \mathcal{T}_h .

Error L^∞ relativo

Este valor mide el máximo error relativo entre los valores aproximados y exactos. Se obtiene normalizando el error L^∞ absoluto, dividiéndolo por el máximo valor absoluto de la solución exacta. Esto permite una comparación más objetiva, especialmente cuando los valores de las soluciones son muy grandes o muy pequeñas. Dicho de otra forma,

$$\frac{\|y_e - y_h\|_{L^\infty}}{\|y_e\|_{L^\infty}} = \frac{\max_i |y_e^i - y_h^i|}{\max_i |y_e^i|}, \quad (2.2)$$

donde y_e , y_h y sus componentes y_e^i , y_h^i son los definidos anteriormente.

Error L^2 absoluto

La norma L^2 mide la diferencia entre las dos soluciones en promedio. Se puede calcular de diversas formas, pero en esta ocasión se utiliza la regla del trapecio para aproximar la integral:

$$\|y_e - y_h\|_{L^2} = \left(\sum_{i=1}^{M-1} \frac{|y_e^i - y_h^i|^2 + |y_e^{i+1} - y_h^{i+1}|^2}{2} (t_{i+1} - t_i) \right)^{1/2}, \quad (2.3)$$

donde t_i será el instante de tiempo asociado al punto i , y M, y_e, y_h y sus respectivos componentes y_e^i, y_h^i son los definidos anteriormente.

Todo esto se implementa en MATLAB en el script `norma.m`, donde dentro de ella se logran las normas (2.1), (2.2) y (2.3) haciendo uso de las funciones `normaInfAbs`, `normaInfRel` y `normaL2`, respectivamente. Utilizando el ejemplo descrito al inicio de la Sección 2.3, se obtienen los siguientes resultados:

- La norma L^∞ discreta del error absoluto: 0.0054994
- La norma L^∞ discreta del error relativo: 0.0048099
- La norma L^2 discreta del error absoluto: 0.017483

La interpretación del primer resultado es que el mayor error entre dos soluciones es de 0.0055 aproximadamente. Por otra parte, el segundo resultado indica que, en el peor de los casos, la diferencia entre las dos soluciones es del 0.48 %. Para terminar, como la última norma es 0.017483, esto indica que la diferencia entre las dos soluciones es bastante pequeña.

Resumiendo, todo esto indica que la aproximación que hemos hecho es bastante precisa, ya que los errores son muy pequeños tanto en términos absolutos como relativos. Aun así, vamos a hacer un gráfico para ver dónde están los errores.

Al final del script `norma.m`, se genera un gráfico que visualiza los errores en cada punto de la malla. Para ello, se utiliza el comando `errorbar`. Este comando agrega barras de error para representar la magnitud del error en cada uno de esos puntos. Sin embargo, mostrar barras de error en todos los puntos de la malla no sería práctico, ya que no se podrían visualizar los errores. Por esta razón, se introduce el parámetro `tramo`, que permite mostrar las barras de error solo en unos puntos específicos de la malla, facilitando así su visualización. Además, dado que los errores son muy pequeños, se utiliza el parámetro `escala` para ajustar la visualización. Esto ayuda a resaltar el error y mejora la comprensión de la precisión de la aproximación en el gráfico.

Se puede observar claramente en la Figura 2.3 que el error aumenta a medida que pasan los días. Al principio no hay ningún error debido a la condición inicial, lo que provoca que la aproximación sea exacta al inicio, alcanzando su valor máximo al final de la simulación. Esto se debe a que las aproximaciones numéricas tienden a acumular errores a medida que avanzan en el tiempo.

2.3.3. Orden de convergencia

Por último, para validar nuestro modelo es necesario comprobar que tiene un buen orden de convergencia. Dado que para obtener la aproximación se utiliza `ode45`, el orden de convergencia debería de situarse alrededor de 4 o 5.

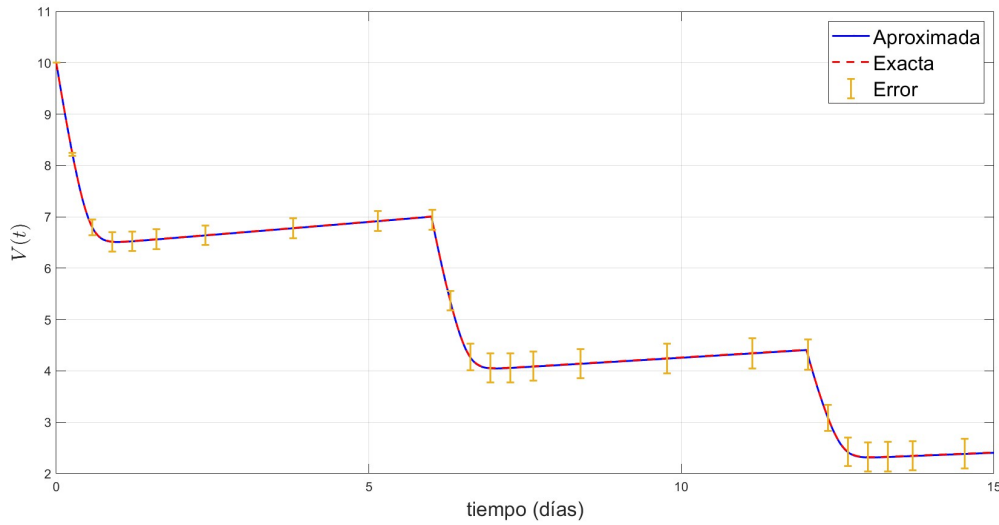


Figura 2.3: Visualización del error.

Definición 2.1. Sea h un parámetro de discretización suficientemente pequeño y E su error correspondiente. Entonces, el *orden de convergencia* p se logra de esta manera:

$$E = Ch^p,$$

siendo C una constante cualquiera.

Observación 2.2. Si consideramos dos parámetros de discretización, h_i y h_{i+1} con sus respectivos errores, e_i y e_{i+1} , entonces, logramos estas dos ecuaciones: $e_i = Ch_i^p$ y $e_{i+1} = Ch_{i+1}^p$. Dividiendo ambas ecuaciones,

$$\frac{e_i}{e_{i+1}} \approx \left(\frac{h_i}{h_{i+1}} \right)^p,$$

y aplicando logaritmos en ambos lados, podemos despejar el orden de convergencia:

$$p \approx \frac{\log\left(\frac{e_i}{e_{i+1}}\right)}{\log\left(\frac{h_i}{h_{i+1}}\right)}. \quad (2.4)$$

Es importante tener en cuenta que, para que haya convergencia el paso h_{i+1} y su respectivo error e_{i+1} deben de ser menores que h_i y e_i .

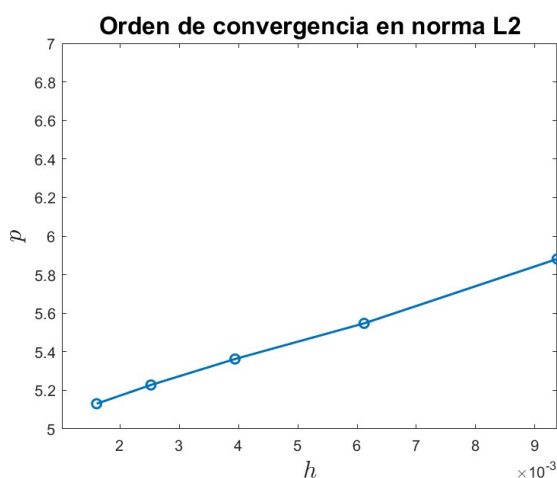
En nuestro caso, el valor de h_i se tomará como la media de todas las diferencias entre los pasos, lo que permite obtener una medida representativa del tamaño de discretización utilizado en la simulación.

Lo que vamos a modificar en este proceso va a ser la tolerancia. Tal y como se observa en la función `orden.m`, tenemos varias tolerancias, las cuales se utilizan para ajustar las opciones

del comando `ode45`. En cada caso, la tolerancia de `ode45` se modifica. Esto quiere decir que la solución aproximada será más refinada a medida que se reduzca la tolerancia. Como consecuencia, la media de las diferencias entre los pasos disminuirá, y con ello, el error.

Por otro lado, el error se calcula utilizando una de las normas descritas en la Sección 2.3.2, tal y como se aprecia en la función `orden.m`. En nuestro ejemplo, se emplea la norma (2.3), pero podríamos usar cualquiera.

Una vez obtenidos los valores necesarios, el orden de convergencia se determina mediante la ecuación (2.4), cuyos resultados pueden verse en la Tabla 2.2.



tol	$h(i)$	$e(i)$	$p(i)$
1e-7	0.009375	2.019674e-7	5.881121
1e-8	0.006117	1.640283e-8	5.547392
1e-9	0.003939	1.426882e-9	5.362311
1e-10	0.002520	1.301014e-10	5.227732
1e-11	0.001604	1.225698e-11	5.130386
1e-12	0.001018	1.192532e-12	

Tabla 2.2: Resumen de los resultados obtenidos utilizando la función `orden.m`.

Figura 2.4: Gráfico del orden de convergencia con la norma L^2 .

Además, se genera un gráfico que representa la evolución del orden de convergencia. Como el paso medio es mayor cuando la tolerancia es mayor, la primera aproximación del orden de convergencia es la más alta, y también su paso. Por esta razón, el gráfico se representa de derecha a izquierda, tal y como se muestra en la Figura 2.4. Aunque pueda parecer que el orden aumenta, en realidad está disminuyendo, lo que se confirma en la Tabla 2.2.

Observando la Figura 2.4 y la Tabla 2.2, podemos concluir que el orden de convergencia tiende a 5. Es decir, el modelo se comporta conforme a lo esperado y su implementación es correcta.

Capítulo 3

Optimización de la distribución de dosis

A la hora de optimizar el modelo, es posible plantear distintos enfoques. Por ejemplo, el número total de dosis, la cantidad administrada en cada una, los momentos de administración, la duración total del tratamiento o la dosis total administrada. También puede haber diferentes objetivos como minimizar la toxicidad acumulada o el volumen del tumor.

En este trabajo nos centramos específicamente en la optimización de la distribución de la dosis, es decir, en determinar el número total de dosis y la cantidad de fármaco de cada una de ellas, con el objetivo de minimizar el volumen del tumor al final del tratamiento.

En este capítulo, se expone en profundidad el problema de optimización asociado al modelo desarrollado. Comenzando con una función general, se introducen los resultados obtenidos en los capítulos anteriores para lograr una versión más concreta y adaptada a este trabajo. Después, teniendo en cuenta el fármaco que se utiliza, se hacen algunas aproximaciones para simplificar los cálculos y para que sea más fácil de entender. El objetivo será alcanzar algunas conclusiones muy útiles para el Capítulo 4, donde pondremos en práctica todo lo desarrollado en este capítulo.

3.1. Introducción y parámetros

Tal y como se explicó anteriormente, se va a realizar una optimización de la distribución de dosis del fármaco, manteniendo fija la cantidad total administrada.

Para ello, se parte de los parámetros definidos en los capítulos anteriores, como el tiempo total del tratamiento, $T > t_N$, los instantes de administración t_i para $i = 1, \dots, N$ y las dosis que se administran en cada instante t_i , d_i . Además, hay otros datos que necesitaremos, d_{min} y d_{max} , que son las dosis mínimas y máximas que se podrán administrar, respectivamente. Tal y como es de esperar, $0 < d_{min} < d_{max}$.

El objetivo de esta optimización es determinar el número óptimo de dosis N y el valor de cada dosis d_i , de forma que manteniendo constante la cantidad total que se administra, D , se minimice el volumen tumoral al final del tratamiento.

3.2. Problema de optimización

Haciendo uso del artículo [7], el problema de optimización que necesitamos en este trabajo tiene como base la siguiente modelización:

$$(P_1) = \begin{cases} \text{Encontrar } N \in \mathbb{N}, & d = (d_1, \dots, d_N) \in \mathbb{R}^N \\ \text{sujeto a } \sum_{i=1}^N d_i = D \text{ tales que} & \\ & d_{min} \leq d_i \leq d_{max}, \quad i = 1, \dots, N, \text{ y se minimice el valor } V(t), \end{cases} \quad (3.1)$$

siendo $T > t_N$.

Utilizando los resultados que se han logrado en el Capítulo 1, se puede concretar algo más. Así, recuperando la solución de $V(T)$ de la expresión (1.27), obtenida en la Sección 1.2, el mínimo de $V(t)$ se puede escribir como:

$$\text{mín } V(T) = \text{mín} \left(H \exp \left(\ln \left(\frac{V_0}{H} \right) \exp \left(-r \left(T - \int_0^T G(c(s)) ds \right) \right) \right) \right).$$

Entonces, tal y como se explica en la Sección 1.2, dado que $V_0 < H$, se verifica que $\ln(\frac{V_0}{H}) < 0$. Así pues, para que el volumen del tumor en el tiempo final, $V(T)$, sea mínimo, necesitaremos que la segunda parte del exponente sea máximo, es decir,

$$\begin{aligned} \text{mín } V(T) &= \text{máx} \left(\exp \left(-r \left(T - \int_0^T G(c(s)) ds \right) \right) \right) \\ &= \text{máx} \left(\exp \left(-rT + r \int_0^T G(c(s)) ds \right) \right) = \text{máx} \left(\int_0^T G(c(s)) ds \right), \end{aligned}$$

ya que r y T son constantes. Por ello, el problema de optimización quedaría así,

$$\begin{cases} \text{Encontrar } N \in \mathbb{N}, & d = (d_1, \dots, d_N) \in \mathbb{R}^N \\ \text{sujeto a } \sum_{i=1}^N d_i = D, \text{ tales que} & \\ & d_{min} \leq d_i \leq d_{max}, \quad i = 1, \dots, N, \text{ y se maximice } \int_0^T G(c(s)) ds. \end{cases} \quad (3.2)$$

En la Sección 1.3, se logra la solución de la integral de (3.2), que corresponde a la última línea de (1.31). Poniéndolo en una forma más compacta, se logra,

$$\int_0^T G(c(s)) ds = \frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1) \prod_{i=2}^N (c_{50} + \sum_{j=1}^{i-1} d_j e^{-\lambda(t_i - t_j)} + d_i)}{\prod_{i=1}^N (c_{50} + \sum_{j=1}^i d_j e^{-\lambda(t_{i+1} - t_j)})} \right), \quad (3.3)$$

suponiendo que $t_{N+1} = T$. Entonces,

$$\begin{aligned} \text{máx} \left(\int_0^T G(c(s)) ds \right) &= \text{máx} \left(\frac{E_{max}}{\lambda} \ln \left(\frac{(c_{50} + d_1) \prod_{i=2}^N (c_{50} + \sum_{j=1}^{i-1} d_j e^{-\lambda(t_i - t_j)} + d_i)}{\prod_{i=1}^N (c_{50} + \sum_{j=1}^i d_j e^{-\lambda(t_{i+1} - t_j)})} \right) \right) \\ &= \text{máx} \left(\frac{(c_{50} + d_1) \prod_{i=2}^N (c_{50} + \sum_{j=1}^{i-1} d_j e^{-\lambda(t_i - t_j)} + d_i)}{\prod_{i=1}^N (c_{50} + \sum_{j=1}^i d_j e^{-\lambda(t_{i+1} - t_j)})} \right), \end{aligned}$$

ya que E_{max} y λ son constantes, y maximizar el logaritmo es igual que maximizar su argumento. Para ver esto, se consideran $A_i = c_{50} + \sum_{j=1}^{i-1} d_j e^{-\lambda(t_i - t_j)} + d_i$ y $B_i = c_{50} + \sum_{j=1}^i d_j e^{-\lambda(t_{i+1} - t_j)}$ para cada $i = 2, \dots, N$. De este modo, el argumento del logaritmo se puede escribir como

$$\frac{(c_{50} + d_1) \prod_{i=2}^N A_i}{(c_{50} + d_1 e^{-\lambda(t_2 - t_1)}) \prod_{i=2}^N B_i}. \quad (3.4)$$

Obsérvese que $e^{-\lambda(t_i - t_j)}, e^{-\lambda(t_{i+1} - t_j)} > 0$, y además, como $t_{i+1} > t_i$, se sabe que $(t_i - t_j) < (t_{i+1} - t_j)$ para todo $i > j$, por lo que $e^{-\lambda(t_i - t_j)} > e^{-\lambda(t_{i+1} - t_j)}$. Sabiendo esto, se puede apreciar que $d_j e^{-\lambda(t_i - t_j)} > d_j e^{-\lambda(t_{i+1} - t_j)}$. Es decir, $A_i > B_i$ para todo $i = 2, \dots, N$ y también se deduce que $(c_{50} + d_1) > (c_{50} + d_1 e^{-\lambda(t_2 - t_1)})$, por lo que la fracción (3.4) va a ser mayor que 1, es decir, el logaritmo va a ser positivo. Esto garantiza que el máximo del logaritmo coincida con el máximo del cociente, y por tanto se pueda trabajar directamente con la expresión racional.

Por tanto, el problema se reduce a maximizar la expresión contenida dentro del logaritmo. Es decir,

$$\left\{ \begin{array}{l} \text{Encontrar } N \in \mathbb{N}, \quad d = (d_1, \dots, d_N) \in \mathbb{R}^N \\ \text{sujeto a } \sum_{i=1}^N d_i = D, \text{ tales que} \\ \quad d_{min} \leq d_i \leq d_{max}, \quad i = 1, \dots, N, \text{ y se maximice } \frac{(c_{50} + d_1) \prod_{i=2}^N (c_{50} + \sum_{j=1}^{i-1} d_j e^{-\lambda(t_i - t_j)} + d_i)}{\prod_{i=1}^N (c_{50} + \sum_{j=1}^i d_j e^{-\lambda(t_{i+1} - t_j)})}. \end{array} \right. \quad (3.5)$$

3.2.1. Aproximaciones del problema

El artículo [7] utiliza como ejemplo la temozolomida, un fármaco empleado en el tratamiento de ciertos tipos de cáncer. Se trata de un fármaco citotóxico con vida media corta, lo que implica una rápida eliminación del fármaco entre dosis consecutivas. Las características y el funcionamiento de los fármacos citotóxicos, y en concreto de la temozolomida, se analizan con más detalle en el Capítulo 4.

Gracias a la vida media corta de la temozolomida, el artículo [7] utiliza una aproximación para que el resultado compacto (3.3) se pueda simplificar. Así, nuestro problema será más sencillo

para resolver. La aproximación que se considera es la siguiente:

$$d_{max}e^{-\lambda s} \ll c_{50}, \text{ donde } s = \min_{i=1, \dots, N} (t_{i+1} - t_i). \quad (3.6)$$

Esta desigualdad se cumple siempre y cuando la vida media del fármaco es suficientemente pequeña, o lo que es lo mismo, la tasa de eliminación, λ , es alta. En el caso de la temozolomida, la vida media es muy pequeña, por lo que, generalmente, se va a cumplir esta hipótesis. Así, podremos utilizarlo para simplificar el resultado que tenemos. Analizando la expresión (3.3), bajo la hipótesis (3.6), se pueden despreciar los sumatorios incluidos en el argumento del logaritmo. Entonces, quedaría algo así:

$$\begin{aligned} \frac{(c_{50} + d_1) \prod_{i=2}^N (c_{50} + \sum_{j=1}^{i-1} d_j e^{-\lambda(t_i - t_j)} + d_i)}{\prod_{i=1}^N (c_{50} + \sum_{j=1}^i d_j e^{-\lambda(t_{i+1} - t_j)})} &\approx \frac{(c_{50} + d_1) \prod_{i=2}^N (c_{50} + d_i)}{\prod_{i=1}^N c_{50}} \\ &= \frac{\prod_{i=1}^N (c_{50} + d_i)}{\prod_{i=1}^N c_{50}} = \prod_{i=1}^N \left(\frac{d_i}{c_{50}} + 1 \right), \end{aligned}$$

es decir, el problema de optimización será,

$$(P_2) = \begin{cases} \text{Encontrar } N \in \mathbb{N}, \quad d = (d_1, \dots, d_N) \in \mathbb{R}^N \\ \text{sujeto a } \sum_{i=1}^N d_i = D, \text{ tales que} \\ \quad d_{min} \leq d_i \leq d_{max}, \quad i = 1, \dots, N, \text{ y se maximice } \prod_{i=1}^N \left(\frac{d_i}{c_{50}} + 1 \right). \end{cases} \quad (3.7)$$

3.3. Resultados teóricos

A partir del Problema 3.7 se sabe que $\sum_{i=1}^N d_i = D$, lo que equivale a $\frac{\sum_{i=1}^N d_i}{D} = 1$. De aquí se puede deducir que, si no se aplica todo el fármaco en una sola dosis, entonces $\frac{d_{min}}{D}, \frac{d_{max}}{D} < 1$, es decir,

$$\left[\frac{d_{min}}{D}, \frac{d_{max}}{D} \right] \subset (0, 1).$$

Esta observación permite expresar las dosis en términos de porcentajes respecto a la dosis total D .

Por otra parte, sabiendo el porcentaje mínimo y máximo de D a dar en cada dosis, el objetivo es encontrar N tal que

$$\frac{1}{N} \in \left[\frac{d_{min}}{D}, \frac{d_{max}}{D} \right],$$

equivalentemente,

$$N \in \left[\frac{D}{d_{max}}, \frac{D}{d_{min}} \right].$$

Como N es la cantidad total de dosis, debe ser un número natural, por lo que,

$$\left[\frac{D}{d_{max}}, \frac{D}{d_{min}} \right] \cap \mathbb{N} \neq \emptyset, \quad (3.8)$$

es decir, dentro del intervalo tiene que haber un número natural.

Esta hipótesis garantiza la existencia de un valor $N \in \mathbb{N}$, tal que, al dividir la dosificación total D en N partes iguales, cada dosis d_i esté dentro del intervalo $[d_{min}, d_{max}]$, para todo $i = 1, \dots, N$. Por tanto, los valores válidos de N son aquellos números naturales que pertenecen al conjunto:

$$N \in \left\{ \left\lceil \frac{D}{d_{max}} \right\rceil, \dots, \left\lfloor \frac{D}{d_{min}} \right\rfloor \right\}, \quad (3.9)$$

donde $\lceil x \rceil$ y $\lfloor x \rfloor$ denotan las funciones techo y suelo enteros de x , respectivamente. Fijando uno de estos valores posibles para N , se plantea a continuación el siguiente problema:

$$(P_2^N) = \begin{cases} \text{Encontrar } N \in \mathbb{N}, \quad d = (d_1, \dots, d_N) \in \mathbb{R}^N \\ \text{sujeto a } \sum_{i=1}^N d_i = D, \text{ tales que} \\ \quad d_{min} \leq d_i \leq d_{max}, \quad i = 1, \dots, N, \text{ y se maximice } \prod_{i=1}^N \left(\frac{d_i}{c_{50}} + 1 \right). \end{cases} \quad (3.10)$$

Con todo esto, podemos presentar un par de resultados, demostrados en [7]:

Teorema 3.1. *Sea $d_{min} > 0$ y asumimos que se cumple (3.8). Entonces, para cada N verificando (3.9), el problema (P_2^N) tiene una única solución óptima \hat{d} ,*

$$\hat{d}_i = D/N, \text{ para todo } i = 1, \dots, N.$$

Demostración. Se sabe que la relación entre la media aritmética y geométrica es la siguiente:

$$\sqrt[N]{\prod_{i=1}^N x_i} \leq \frac{1}{N} \sum_{i=1}^N x_i.$$

Cambiando x_i por $\frac{d_i}{c_{50}} + 1$,

$$\sqrt[N]{\prod_{i=1}^N \left(\frac{d_i}{c_{50}} + 1 \right)} \leq \frac{1}{N} \sum_{i=1}^N \left(\frac{d_i}{c_{50}} + 1 \right) = \frac{1}{N} \left(\frac{D}{c_{50}} + N \right) = \frac{D}{Nc_{50}} + 1.$$

Así, utilizando el problema (P_2^N) y la desigualdad de arriba, el valor máximo de $\prod_{i=1}^N \left(\frac{d_i}{c_{50}} + 1 \right)$ se alcanzará cuando se alcance el valor $\left(\frac{D}{Nc_{50}} + 1 \right)^N$. Es decir,

$$\frac{d_i}{c_{50}} + 1 = \frac{D}{Nc_{50}} + 1 \iff d_i = \frac{D}{N}, \text{ para todo } i = 1, \dots, N.$$

Entonces, queda demostrado que hay una única solución óptima \hat{d} . □

Utilizando el Teorema 3.1, podemos lograr otro resultado:

Corolario 3.2. *Supongamos que $d_{min} > 0$ y que se cumple (3.8). Entonces, la solución óptima de (P_2) está dada por (\hat{N}, \hat{d}) , donde $\hat{N} = \lfloor \frac{D}{d_{min}} \rfloor$ y $\hat{d}_i = \frac{D}{\hat{N}}$, para todo $i = 1, \dots, \hat{N}$.*

Demostración. Utilizando el Teorema 3.1, podemos deducir que, para un número fijo de dosis \hat{N} , la distribución óptima es $\hat{d}_i = \frac{D}{\hat{N}}$.

Por otra parte, utilizaremos la siguiente función auxiliar:

$$\varphi_1(x) = \left(\frac{1}{x} + 1 \right)^x.$$

Es fácil ver que esta función es estrictamente creciente en $(0, \infty)$. Entonces, utilizando la igualdad 3.3 para \hat{d} y (P_2^N) ,

$$\prod_{i=1}^{\hat{N}} \left(\frac{d_i}{c_{50}} + 1 \right) = \left(\frac{D}{\hat{N}c_{50}} + 1 \right)^{\hat{N}} = \left(\varphi_1 \left(\frac{\hat{N}c_{50}}{D} \right) \right)^{D/c_{50}}.$$

Con esto, se puede ver fácilmente que el valor máximo de este producto se logrará cuando \hat{N} sea el máximo posible, puesto que la función auxiliar es estrictamente creciente y todos los demás valores son constantes.

Entonces, cuanto más grande sea \hat{N} , \hat{d}_i será más pequeño. Pero como estamos en el problema (P_2) , la dosis no puede ser más pequeña que d_{min} . Es decir,

$$\hat{d}_i \geq d_{min} \iff \frac{D}{\hat{N}} \geq d_{min} \iff \hat{N} \leq \frac{D}{d_{min}}. \quad (3.11)$$

Dicho de otra forma, $\hat{N} = \lfloor \frac{D}{d_{min}} \rfloor$. □

Con estos resultados, se llega a la conclusión de que para este tipo de fármacos, con vida media corta, el método curativo óptimo es el tratamiento más duradero y con dosis individuales iguales.

Entonces, se podría pensar que para que el tratamiento sea óptimo, N tiene que ser muy grande, o lo que esto implica, que la d_{min} sea lo más pequeño posible, pero hay que tener en cuenta que para que el medicamento sea eficiente, la dosis mínima tiene que ser de valor positivo. Por eso, habría que encontrar un equilibrio y determinar la d_{min} óptima.

Capítulo 4

Resolución numérica del problema de optimización

En este capítulo se implementa en MATLAB el problema de optimización desarrollado en el capítulo anterior, con el objetivo de obtener una estrategia óptima para la distribución de dosis de un fármaco en tratamientos quimioterápicos. Todo lo relacionado con este capítulo es puramente práctico, poniendo en práctica todos los resultados teóricos vistos en los capítulos anteriores.

En primer lugar, para contextualizar lo siguiente se introduce brevemente el concepto de los fármacos citotóxicos, ya que, los resultados que se reproducen posteriormente están relacionados con tratamientos que utilizan este tipo de fármacos.

A continuación, se detalla la función de optimización utilizada. Se explican los argumentos de entrada y salida, así como su función dentro del algoritmo, con el objetivo de facilitar su comprensión. Además, se explica cómo se aplican las restricciones del problema (P_2) (ver 3.7) dentro del algoritmo.

Finalmente, se reproducen varias tablas y figuras del artículo [7] con el fin de contrastar los resultados obtenidos en este trabajo con los de la bibliografía. Así, se puede comprobar la validez del modelo implementado y verificar que los resultados teóricos y prácticos coincidan.

4.1. Contexto clínico y farmacológico

Tal y como se ha dicho en capítulos anteriores, en el artículo [7] se consideran los fármacos citotóxicos.

4.1.1. Fármacos citotóxicos

Los fármacos citotóxicos son fármacos que actúan en las células que se dividen muy rápido [6]. Las células cancerígenas, los folículos pilosos y algunas células del estómago e intestino son ejemplo de este tipo de células. Los fármacos de quimioterapia son de este tipo, es decir, la quimioterapia mata a todas las células que se propagan rápidamente.

Aun así, estos fármacos matan a las células en un cierto momento de su división. Si no están en ese punto en el momento en el que el fármaco está actuando, el fármaco no tiene efecto sobre ellas. Por eso, en los tratamientos se usan diferentes tipos de fármacos y se repiten ciclos. Por ejemplo, un fármaco de este tipo es la temozolomida.

La temozolomida se usa para el tratamiento de algunos tumores cerebrales como, por ejemplo, el glioblastoma uniforme. Se usa normalmente cuando el glioblastoma está recién diagnosticado como tratamiento de monoterapia [15].

Tal y como se explica en el Capítulo 3, la temozolomida es un fármaco con vida media corta, por lo que es interesante al hacer este trabajo, ya que se pueden usar algunas aproximaciones para obtener resultados muy útiles.

4.1.2. Parámetros usados

Tal y como se explica en el Capítulo 2, en este capítulo se van a usar los mismos parámetros definidos en la Tabla 2.1. Los resultados de este capítulo también aparecen divididos por H en el artículo [7], por lo que haremos lo mismo que en el Capítulo 2, es decir, $H = 1dm^3$. Lo único que cambian son las dosis.

En este capítulo cuando se exponen los tratamientos, las dosis que se administran en cada ciclo serán las que se han definido en la Tabla 2.1 como \hat{d}_i , es decir, esas dosis no son las dosis que se han definido en este trabajo, pues, para ello, se tiene que multiplicar \hat{d}_i por σ , tal y como se ha explicado en el Capítulo 2.

El volumen del tumor se divide por H para comparar sin preocuparse por las unidades y para facilitar la interpretación teórica. Por otro lado, al obtener las soluciones también aparece el término $V(t)/V_0$, donde V_0 es el volumen del tumor al inicio del tratamiento. Esta relación permite saber cuánto crece o decrece el tumor respecto a su valor inicial. Por ejemplo, si $V(T)/V_0 = 0,7$, significa que el volumen del tumor se ha reducido al 70% del tamaño que tenía al inicio del tratamiento.

4.2. Implementación de la optimización en MATLAB

MATLAB dispone de múltiples funciones para poder optimizar distintos problemas, según las características de cada uno. Todas estas funciones de optimización están en su *Optimization Toolbox*. La función que mejor se adapta a nuestro problema es la función `fmincon`, puesto que es una muy buena opción para minimizar funciones no lineales con restricciones. Para encontrar información de esta función lo que se ha usado ha sido la ayuda propia de MATLAB.

Función `fmincon`

Tal y como aparece en la ayuda de MATLAB, el esquema de esta función es:

```
[x_opt, fval] = fmincon(fun, x0, A, b, Aeq, beq, lb, ub, nonlcon, options)
```

El primer argumento de entrada de esta función es `fun`. Este argumento debe de ser una función anónima que tenga como variable el vector de las dosificaciones del tratamiento, y dentro de ella, esta función anónima llama a la función que contenga al modelo matemático y devuelve el valor de la función objetivo. En nuestro caso, la función objetivo está explicado en el código II.1. Como la función objetivo (1.28) está implementado dentro de `pkpd_ode45.m`, los argumentos de `funcion_objetivo` serán los mismos que los de `pkpd_ode45.m`, es decir, los parámetros, dosis y momentos en los que se administran esas dosis.

Por otra parte, `x0` es el punto inicial. En este problema, será el tratamiento inicial, para luego optimizarlo. Este argumento es bastante relevante, porque es el punto de partida del algoritmo, y si el punto inicial no encaja con el problema, el algoritmo tarda en converger, o incluso, no converge adecuadamente. Tal y como se ve en el capítulo anterior, en el Corolario 3.2, la mejor opción para el punto inicial es $x_0 = (\frac{D}{N}, \dots, \frac{D}{N})$.

Los argumentos de entrada `A` y `b` son las restricciones lineales de desigualdad, es decir, $Ax \leq b$. En este problema no tenemos esta restricción, por lo que estos argumentos estarán vacíos.

Además, se incluyen los argumentos `Aeq` y `beq`, que son las restricciones lineales de igualdad, $Aeqx = beq$. En este problema, tenemos la restricción $\sum_{i=1}^N d_i = D$, por lo que `Aeq` va a ser un vector de unos, y `beq` será `D`. Asimismo, están los argumentos `lb` y `ub` que son las cotas inferiores y superiores de las variables, es decir, $lb \leq x \leq ub$. En el caso de nuestro problema, `lb` y `ub` son los vectores d_{min} y d_{max} .

Para finalizar, tenemos los argumentos de entrada `nonlcon` y `options`. El argumento `nonlcon` se utiliza para las restricciones no lineales que tiene el problema. En nuestro problema no hay restricciones no lineales por lo que este argumento estará vacío. Por último, el argumento `options` es una estructura con diferentes opciones para definir cómo se comporta el algoritmo. Lo más

común es que sea de este tipo:

```
options = optimoptions('fmincon','Display','iter','Algorithm','sqp')
```

El argumento `iter` se usa para observar el proceso de resolución, es decir, para ver cómo evoluciona, y el `sqp` es el método recomendado para problemas suaves con igualdades. En ocasiones es más rápido que el algoritmo `interior_point`, que es el predeterminado.

Por otra parte, normalmente se utilizan como argumentos de salida los valores `x_opt` y `fval`. El primero, `x_opt`, es el vector donde se alcanza el mínimo de la función objetivo, mientras que `fval` es el valor que dicha función toma en ese punto. En este trabajo, `x_opt` corresponde al tratamiento óptimo, es decir, al vector de dosis d_i , y `fval` será el volumen tumoral final, V , obtenido al aplicar esa distribución de dosis.

4.3. Reproducción de los resultados

Tal y como se indica en [7], el tratamiento habitual para un paciente con glioblastoma uniforme que recibe temozolomida consiste en 6 ciclos, cada uno compuesto por 5 días consecutivos de tratamiento, seguido de 23 días de descanso, completando así un ciclo de 28 días. Esto se denota como $5/28d$. La dosis administrada en el primer ciclo suele ser de $150mg/m^2$, mientras que en los ciclos posteriores se incrementa a $200mg/m^2$. El objetivo es determinar si este tratamiento es óptimo o hay alguna otra estrategia más eficaz.

Además de esta estrategia para el tratamiento, también puede haber otros tratamientos con ciclos de $7/14d$ con dosis de $150mg/m^2$, $21/28d$ con dosis de $100mg/m^2$ o de $75mg/m^2$ o también ciclos de $28/28d$, es decir, dosis diaria de $50mg/m^2$, entre otros.

En esta sección, se reproducen algunas de las figuras y tablas del artículo [7], específicamente las Tablas 2 y 3, y las primeras gráficas de las Figuras 1 y 2, con el objetivo de verificar la validez de los resultados obtenidos y comprobar si existe alguna discrepancia con respecto al modelo implementado.

4.3.1. Tratamiento óptimo para $5/28d$

Se considera el tratamiento de $5/28d$, con dosis mínima de $150mg/m^2$ y dosis máxima de $200mg/m^2$. La dosis total, en toda la sección va a ser $D = 5750mg/m^2$, reutilizando todos los demás parámetros de la Tabla 2.1.

En esta sección se quiere ver cuáles son las dosis y cantidad de dosis óptimas. Es decir, siendo $N \in [29, 40]$, cuál es el valor óptimo N . Tal y como se vio en el Corolario 3.2 de la Sección 3.3,

N	(P_2)		(P_1)		
	$\frac{D}{N}$	$\frac{V(T)}{V_0}$	\hat{d}_{min}	\hat{d}_{max}	$\frac{V(T)}{V_0}$
29	198.27586	0.92788	198.25603	198.32483	0.92788
30	191.66667	0.90524	191.64680	191.71387	0.90524
31	185.48387	0.88352	185.46121	185.54979	0.88352
32	179.68750	0.86268	179.66763	179.72724	0.86268
33	174.24242	0.84267	174.22279	174.28336	0.84267
34	169.11765	0.82344	169.09979	169.15596	0.82344
35	164.28571	0.80496	164.26876	164.32554	0.80496
36	159.72222	0.78717	159.70287	159.77536	0.78717
37	155.40541	0.77006	155.38866	155.43935	0.77006
38	151.31579	0.75359	151.30993	151.32733	0.75359

Tabla 4.1: Resultados de las optimizaciones de los problemas (P_1) y (P_2) .

lo óptimo sería que las dosis fuesen todas iguales con $d_i = \frac{D}{N}$, con $N = \lfloor \frac{D}{d_{min}} \rfloor$.

Entonces, lo más útil sería coger como punto inicial $\mathbf{x}_0 = (\frac{D}{N}, \dots, \frac{D}{N})$, ya que se puede considerar que se cumplen las hipótesis del Corolario 3.2. Para utilizar el corolario, hay que verificar que se cumple la desigualdad (3.6), ya que, si no, no se podría verificar que las soluciones teóricas sean ciertas.

Siendo $d_{max} = 200$, $\lambda = 9,242$, $c_{50} = 0,36$ y $s = 1$,

$$200e^{-9,242} = 0,0194 \ll 0,36,$$

se verifica la desigualdad, por lo que se puede hacer la aproximación. Entonces, primero de todo, se reproduce la Tabla 2 del artículo [7] en la Tabla 4.1, utilizando el código del Anexo II.2.

Tal y como se explica en la Subsección 4.1.2, se divide el volumen tumoral al final del tratamiento con el volumen inicial, para poder deducir si el tumor decrece o no. Por otra parte, \hat{d}_{min} y \hat{d}_{max} son valores que se obtienen después de hacer la optimización.

Se puede ver que los resultados de la Tabla 4.1 y de la Tabla 2 del artículo [7] son idénticos. Además, es importante destacar que en esta tabla no se incluyen los valores $N = 39$ y $N = 40$. Tal y como se comentó en el apartado 3.3, la dosis administrada no puede ser menor que d_{min} , ya que el fármaco podría no ser efectivo. Sin embargo, tomando $N = 39$ o $N = 40$, las dosis $\frac{D}{N}$ son menores que $d_{min} = 150mg/m^2$, lo cual no cumple las restricciones marcadas.

Técnicamente, podemos ponerlas en la tabla, pero no es correcto afirmar que lo óptimo es coger $N = 40$, ya que no cumple las hipótesis del problema (P_2) . Por eso, se ve perfectamente

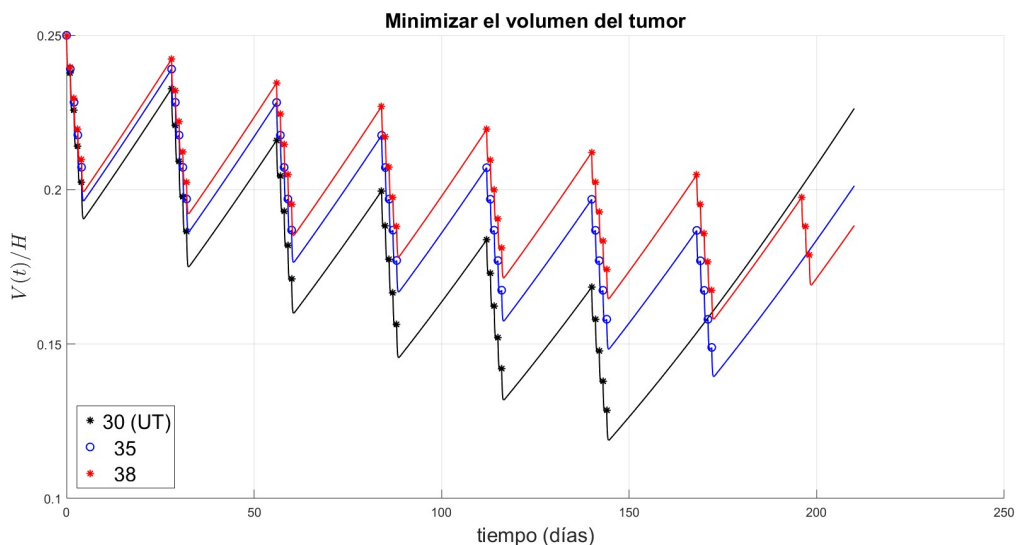


Figura 4.1: Evolución del volumen del tumor con tratamientos con $N = 30, 35$ y 38 dosis.

en la Tabla 4.1 que lo óptimo sería tener un tratamiento con $N = 38$ dosis, ya que el tumor se rebaja al 75%, cosa que no logran las otras distribuciones de la dosis total de fármaco.

En la Figura 4.1 se representan las evoluciones del volumen tumoral para tres estrategias correspondientes a $N = 30$, $N = 35$ y $N = 38$. Los valores de esta figura no se extraen de la última columna de la Tabla 4.1, sino que lo que se muestra es la evolución del volumen normalizado. Aún así, ambas representaciones se pueden relacionar: por ejemplo, el 75,359% de $0,25dm^3$ es $0,1884dm^3$, que corresponde con el valor que se obtiene en la curva roja ($N = 38$) en $T = 210d$.

En el artículo [7] se realiza una comparación con N igual a 40 en vez de 38 en su gráfica 1 de la Figura 1. Sin embargo, dado que un tratamiento con $N = 40$ no está dentro de nuestro problema, resulta más adecuado utilizar $N = 38$ que representa verdaderamente el caso óptimo.

En la Figura 4.1 se observa que los tratamientos con un menor número de dosis (y, por tanto, dosis más altas por administración) provocan una reducción inicial más brusca del volumen tumoral. Sin embargo, al tener una duración menor, una vez finalizado el tratamiento, el volumen tumoral vuelve a incrementarse de forma más notable en comparación con la estrategia óptima, $N = 38$. Esta estrategia muestra una disminución más progresiva y sostenida del volumen. Esta evolución más estable justifica su carácter óptimo, ya que logra un mejor control del crecimiento tumoral a lo largo del tiempo.

d_{min}	\hat{N}	$\frac{D}{\hat{N}}$	Ciclo	$\frac{V(T)}{V_0}$	Intensidad de dosis
150	38	151.31579	7/14d	0.75363	78.76712
100	57	100.87719	21/28d	0.52918	80.98596
75	76	75.65789	21/28d	0.40372	59.27835
50	115	50.00000	28/28d	0.27223	50.00000

Tabla 4.2: Resultados de las optimizaciones para los tratamientos 7/14d con dosis de 150mg/m², 21/28d con dosis de 100mg/m² y 75mg/m² y dosis diaria con 50mg/m².

4.3.2. Tratamiento óptimo para $D = 5750\text{mg}/\text{m}^2$

En este caso tenemos algunas estrategias de tratamientos, como por ejemplo, 7/14d con dosis de 150mg/m², 21/28d con dosis de 100mg/m² o de 75mg/m² y 28/28d con dosis de 50mg/m², es decir, dosis diaria. Así, en este apartado, lo que queremos ver es cuál es la mejor estrategia entre ellas manteniendo la dosis total en para $D = 5750\text{mg}/\text{m}^2$.

Primero de todo, utilizando el código del Anexo II.3, se reproduce la Tabla 3 del artículo [7] tal y como se ve en la Tabla 4.2.

Para lograr los resultados, en el primer bloque se utiliza el Corolario 3.2, puesto que se dice que $\hat{N} = \lfloor \frac{D}{d_{min}} \rfloor$ y la dosis óptima se toma como $d_i = \frac{D}{\hat{N}}$. Por eso, no se necesita ninguna función para lograr la optimización, es decir, ya se logran las dosis óptimas. Además, como se verifican la condición (3.6), se pueden hacer las aproximaciones y verificar el corolario.

La relación entre el volumen final y el volumen inicial se hace como en el apartado anterior, siendo lo único destacable la intensidad de dosis. La intensidad de dosis analiza la cantidad de fármaco administrado por unidad de tiempo [5]. Dicho de otra forma,

$$\text{Intensidad de dosis} = \frac{\text{Dosis total administrada}}{\text{Duración total del tratamiento}}. \quad (4.1)$$

Por tanto, utilizando la expresión (4.1), se obtiene la última columna de la Tabla 4.2. En el código que se incluye en el Anexo II.3, se puede observar que es necesario sumar una unidad a la duración total del tratamiento, ya que en MATLAB se comienza en el instante inicial $t_0 = 0$.

En la Tabla 4.2 también se puede ver que se cumplen los resultados teóricos obtenidos en el capítulo anterior, puesto que el tratamiento óptimo es aquel que tiene la mayor cantidad de dosis y las dosis más pequeñas. Esto se puede apreciar en que la evolución del tumor respecto al volumen inicial del tumor baja a un 27% con la dosis diaria, mientras que con 7/14d baja solamente al 75%.

Estos resultados se ven reflejados en la Figura 4.2, siendo la reproducción de la gráfica 1 de

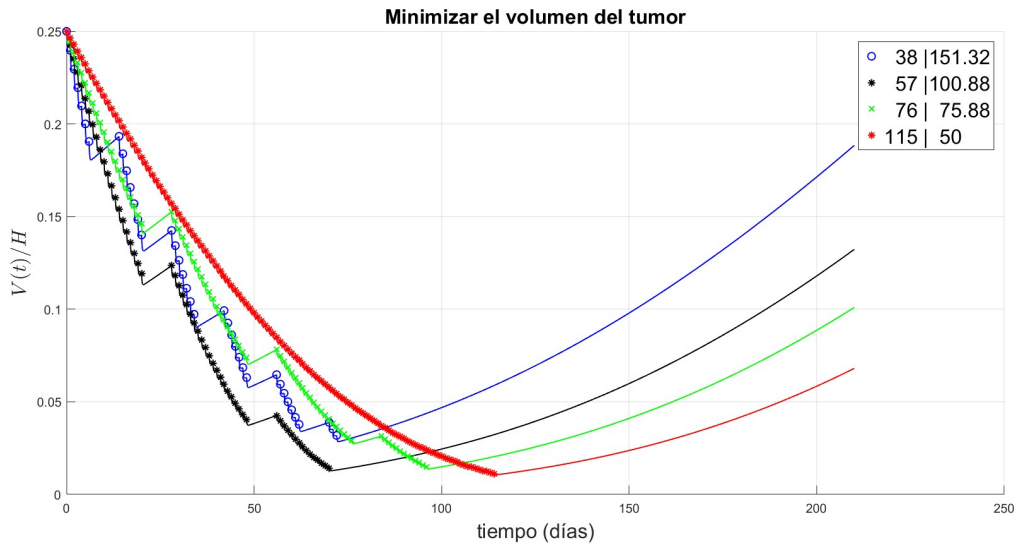


Figura 4.2: Evolución del volumen del tumor con tratamientos $7/14d$ con dosis de $150mg/m^2$, $21/28d$ con dosis de $100mg/m^2$ y $75mg/m^2$ y dosis diaria con $50mg/m^2$.

la Figura 2 de [7], donde aparecen los distintos tratamientos. Al igual que se comentó en relación con la Figura 4.1, la estrategia óptima presenta una pendiente más suave durante el tratamiento y, una vez finalizado, el aumento del volumen tumoral también es más moderado en comparación con el resto de las estrategias.

Cabe señalar que en [7], en el eje y de la figura correspondiente, se utiliza la notación $V(t)/V_0$ en lugar de $V(t)/H$. Aunque no supone un grave error, es un detalle que conviene tener en cuenta, ya que puede generar confusión con respecto a la normalización utilizada en el resto del modelo.

Capítulo 5

Conclusiones

A lo largo de este trabajo se ha desarrollado un modelo matemático PK/PD para describir la evolución de la concentración de un fármaco y el consiguiente volumen del tumor, empleando para ello una versión modificada del modelo de Gompertz. Seguidamente, se ha validado el modelo descrito, implementándolo en MATLAB.

La validación se ha hecho de dos formas: la primera ha sido comparando los resultados obtenidos con resultados extraídos de la bibliografía, y la segunda, contrastando la solución numérica calculada con la solución exacta lograda analíticamente. La gráfica lograda es bastante parecida a la original y todos los resultados cuantitativos son bastante buenas, tanto las diversas normas consideradas como el orden de convergencia, lo que permite asumir que tanto la solución del modelo matemático como su implementación en MATLAB son consistentes.

Posteriormente, se ha planteado un problema de optimización con el objetivo de obtener la distribución de dosis óptima. Después de varios resultados teóricos analizados en el Capítulo 3, se concluye que la estrategia óptima consiste en repartir la dosis total de forma equitativa entre el mayor número de administraciones posibles, siempre que se respete la restricción de dosis mínima para que el tratamiento sea efectivo. Estos resultados teóricos se han verificado mediante la reproducción de tablas y gráficas del artículo [7], logrando resultados prácticamente idénticos.

Como conclusión de todos estos análisis, hay que destacar que la elección del tratamiento puede ayudar o perjudicar a la salud del paciente, por lo que tiene mucha importancia.

También hay que enfatizar la relevancia de este tipo de herramientas matemáticas para la planificación terapéutica, ya que, durante mucho tiempo ha sido común utilizar estrategias en el tratamiento que ahora se ve que no son las óptimas. Por ejemplo, en el artículo [7] se expone que lo usual es dar un tratamiento con ciclos de $5/28d$, pero se demuestra que lo óptimo es el

tratamiento diario. No obstante, este tipo de tratamientos implican un mayor coste logístico y una mayor carga para el sistema sanitario.

Al fin y al cabo, tener a una persona yendo diariamente al hospital a recibir el tratamiento es más costoso que tenerla allí cinco días al mes. Por el contrario, como el tratamiento diario es lo óptimo, se obtienen mejores resultados al finalizar el tratamiento, por lo que, en muchos casos no se necesitaría otro tratamiento o incluso pasar por quirófano, cosa que es probable que ocurra después del tratamiento de $5/28d$.

Conviene también señalar que es muy importante revisar la bibliografía utilizada. En este trabajo, en la Tabla 4.1 que reproduce resultados similares a los del artículo [7], se observa que lo óptimo sería que hubiese $N = 40$ dosis, pero en [7] no tienen en cuenta la restricción del problema (P_2), esto es, que la dosis tiene que estar entre d_{min} y d_{max} . No es una errata de gran relevancia, puesto que no cambia mucho la solución, pero resalta la necesidad de interpretar cuidadosamente los resultados publicados.

Finalmente, es necesario reconocer las limitaciones del modelo desarrollado. Este modelo es muy simple, todo está muy simplificado, no se consideran efectos secundarios ni múltiples comportamientos, por ejemplo. Para avanzar hacia aplicaciones clínicas más realistas, sería conveniente utilizar otros modelos aparte del Gompertz, incorporar más restricciones, utilizar datos reales de pacientes, etc.

En conclusión, este trabajo demuestra la importancia de la modelización matemática como herramienta de apoyo a la toma de decisiones terapéuticas. Aunque los modelos actuales son simplificados, su desarrollo y validación logran un primer paso hacia la mejora de los tratamientos personalizados en oncología.

Bibliografía

- [1] J. Barros-Neto. *An introduction to the theory of distributions* (1973), 35-62.
- [2] S. Benzekry et al. Classical Mathematical Models for Description and Prediction of Experimental Tumor Growth. *Public Library of Science* (2014) **10**, 1-19.
- [3] M. Bodnar, M.J. Piotrowska, U. Foryś. Gompertz model with delays and treatment: mathematical analysis. *Mathematical biosciences and engineering : MBE* (2013) **10** (3), 551-563.
- [4] B. Flint, J.M. Das, C.A. Hall. Body Surface Area. [Updated 2025 Feb 6] In: *StatPearls*[Internet]. Disponible en: <https://www.ncbi.nlm.nih.gov/books/NBK559005/>.
- [5] M.L. Citron. Dose-Dense Chemotherapy: Principles, Clinical Results and Future Perspectives. *Breast Care (Basel)* (2008) **3** (4), 251-255.
- [6] L. Eldridge. What does cytotoxic mean? (2024) Disponible en: <https://www.verywellhealth.com/cytotoxic-definition-and-examples-2249082>
- [7] L.A. Fernandez, C. Pola, J. Sáinz-Pardo Díaz. A mathematical justification for metronomic chemotherapy in oncology. *Math. Model. Nat. Phenom* (2022) **17** (12)
- [8] L. A. Fernandez, C. Pola. Optimal control problems for the Gompertz model under the Norton-Simon hypothesis in chemotherapy. *Discrete and Continuous Dynamical Systems -B* (2019) **24** (6), 2577-2612.
- [9] E. Garralda, R. Dienstmann, J. Tabernero. Pharmacokinetic/Pharmacodynamic Modeling for Drug Development in Oncology. *Am Soc Clin Oncol Educ Book* (2017) (37), 210-215.
- [10] N. Holford. Pharmacodynamic principles and the time course of immediate drug effects. *Translational and clinical pharmacology* (2017) **25** (4), 157-161.
- [11] H. Murphy, H. Jaafari, H.M. Dobrovolny. Differences in predictions of ODE models of tumor growth: a cautionary example. *BMC Cancer* (2016) **16**, 1471-2407.

-
- [12] S. Ruiz-Arrebola, D. Guirado, M. Villalobos, A. Lallena. Evaluation of Classical Mathematical Models of Tumor Growth Using an On-Lattice Agent-Based Monte Carlo Model. *Applied Sciences* (2021) **11**.
- [13] C. Vaghi et al. Population modeling of tumor growth curves and the reduced Gompertz model improve prediction of the age of experimental tumors. *PLoS computational biology* (2020) **16** (2).
- [14] <https://www.cancer.org/es/cancer/como-sobrellevar-el-cancer/tipos-de-tratamiento/quimioterapia.html>
- [15] <https://es.wikipedia.org/wiki/Temozolomida>
- [16] https://seom.org/images/publicaciones/informes-seom-de-evaluacion-de-farmacos/LAS_CIFRAS_2024.pdf
- [17] <https://www.who.int/es/news-room/fact-sheets/detail/cancer>

Anexo I

Código de MATLAB del Capítulo 2

En este anexo se explican las funciones utilizadas en el Capítulo 2, así como los códigos para obtener los gráficos y los resultados que se incluyen en el mencionado capítulo. En primer lugar, se detallan las funciones `pkpd_exact.m` y `pkpd_ode45.m` que son las funciones que se emplean recurrentemente en el trabajo.

```
1 function [T_full,y_full] = pkpd_ode45(par, doses, doses_time, options)
2 % La función pkpd_ode45.m se usa para lograr una solución aproximada del
3 % problema:
4 %  $c'(t) = -\lambda c(t) + u(t)$ ;  $c(0) = 0$ 
5 %  $V'(t) = r \cdot V(t) \cdot \ln(H/V(t)) \cdot (1 - E_{\max} \cdot c(t) / (c_{50} + c(t)))$ ;  $V(0) = V_0$ 
6 % Para ello, se utiliza el comando ode45.
7 % options solo se usa en el apartado del orden. Si no, se usan las
8 % tolerancias que se usan por defecto.
9 if isempty(options)
10     options= odeset('RelTol', 1e-3, 'AbsTol', 1e-6);
11 end
12 % Estos vectores serán los que se utilizarán después. Dentro de y_full, la
13 % primera columna será la aproximación de la concentración, y la segunda,
14 % la aproximación del volumen del tumor.
15 T_full = [];
16 y_full = [];
17
18 % Hay que hacer la discretización para cada intervalo. Para ello, hay que
19 % tener muy claro cual es el primer momento de ese intervalo, y cuales son
20 % sus condiciones iniciales en ese momento.
21 t_start = doses_time(1);
22 C_in = par.c0;
23 V_in = par.v0;
24 doses_time_2 = [doses_time,par.T];
25 for i = 1:length(doses_time)
26     % En el primer momento del intervalo entra una dosis, por lo que hay
```

```

27 % que modificar la condición inicial de la concentración.
28 C_in = C_in + doses(i);
29 t_end = doses_time_2(i+1);
30
31 [T,y] = ode45 (@(t,y) [-par.lambda*y(1); par.r*y(2).*log(par.H/y(2)).* ...
32 (1-(par.Emax*y(1)./(par.c50 + y(1))))], [t_start t_end], [C_in;V_in],
33 options);
34
35 T_full = [T_full; T];
36 y_full = [y_full; y];
37
38 % Hay que modificar las condiciones iniciales para el siguiente
39 % intervalo.
40 C_in = y(end,1);
41 V_in = y(end,2);
42 t_start = t_end;
43 end
44 end

```

Código I.1: Código en la función pkpd_ode45.m.

```

1 function Ye = pkpd_exact(Th, par, doses, doses_time)
2 % Esta función pkpd_exact se utiliza para conseguir la solución exacta
3 % utilizando los resultados obtenidos en la Sección 1. Para ello, primero
4 % de todo se reescriben los parámetros para que sea más facil utilizarlos.
5 K = par.Emax / par.lambda;
6 c50 = par.c50; H = par.H; r = par.r; lambda = par.lambda; v0 = par.v0;
7
8 % Nuestra solución exacta será la matriz Ye, donde en su primera columna
9 % tendremos la solución exacta de la concentración, y en la segunda, la
10 % solución exacta del volumen del tumor.
11 Ye = zeros(length(Th), 2);
12 nn = 0;
13 C_prev = 0;
14
15 for i = 1:length(doses)
16 % Como ya tenemos el vector del tiempo Th, lo que hay que hacer es
17 % identificar qué mallado tiene cada intervalo. Para ello se distingue
18 % el último intervalo de los otros, ya que se logra de diferente modo.
19 if i < length(doses)
20 tt = (Th > doses_time(i) & Th < doses_time(i+1));
21 T_i = [doses_time(i); Th(tt); doses_time(i+1)];
22 else
23 tt = (Th > doses_time(i));
24 T_i = [doses_time(i); Th(tt)];
25 end
26

```

```

27 % Concentración de cada tramo.
28 C = C_prev + doses(i) * exp(-lambda * (T_i - doses_time(i)));
29
30 % Volumen en cada tramo.
31 if i == 1
32     int = K * log((c50 + doses(1))./(c50 + C));
33 else
34     int = int_prev + K .* log((c50 + C_prev + doses(i))./ (c50 + C));
35 end
36 V = H .* exp(log(v0/H).*exp(-r.*(T_i - int)));
37
38 % Meter las concentraciones y volúmenes logradas dentro de la matriz.
39 Ye(nn + (1:length(T_i)),1) = C;
40 Ye(nn + (1:length(T_i)),2) = V;
41
42 % Modificar las condiciones iniciales para el siguiente intervalo.
43 nn = nn + length(T_i);
44 C_prev = C(end);
45 int_prev = int(end);
46 end
47 end

```

Código I.2: Código de la función pkpd_exact.m.

I.1. Código de MATLAB de la Sección 2.2

En esta sección se detallan los archivos utilizados en la Sección 2.2 para hacer la Figura 2.1 y poder calcular las diferencias.

```

1 % Limpiar todo y cerrar todas las figuras abiertas.
2 clc;
3 clear;
4 close all;
5
6 % Parámetros, dosis y tiempos de dosis.
7 par = struct('lambda', 9.242, 'r', 5.51e-3, 'Emax', 60, 'c50', 3.6e-1, 'T', 210, ...
8     'v0', 2.5e-1, 'c0', 0, 'H', 1);
9 doses = 4e-3.*[150*ones(1,5), 200*ones(1,25)];
10 doses_time = [0:4,28:32,56:60,84:88,112:116,140:144];
11
12 % Lograr las soluciones aproximadas, exactas y el mallado del tiempo.
13 [Th,Yh] = pkpd_ode45(par,doses,doses_time,[]);
14 Ye = pkpd_exact(Th,par,doses,doses_time);
15
16 % Gráficos de los datos conseguidos.

```

```

17 plotter1;
18
19 % Diferencias entre los datos de la bibliografía y los logrados aquí.
20 diferencias;

```

Código I.3: Código de PRIN1pkpd.m.

```

1 % En este script se hace el gráfico para los datos de PRIN1pkpd.m
2 % Primero de todo, hay que lograr los valores de los momentos en los que se
3 % administra una dosis, para poder dibujar ese momento.
4 Yh_doses = zeros(length(doses_time),2);
5 Ye_doses = zeros(length(doses_time),2);
6 for i = 1:length(doses_time)
7     all_yh_doses = Yh(Th==doses_time(i),:);
8     all_ye_doses = Ye(Th==doses_time(i),:);
9     if size(all_yh_doses)== [2 2]
10         Yh_doses(i,:) = all_yh_doses(2,:);
11         Ye_doses(i,:) = all_ye_doses(2,:);
12     else
13         Yh_doses(i,:) = all_yh_doses(:);
14         Ye_doses(i,:) = all_ye_doses(:);
15     end
16 end
17
18 % Después, hay que obtener los valores de la bibliografía, que están en los
19 % ficheros datos_curva.csv y datos_puntos.csv, logrados utilizando el
20 % programa Engauge Digitizer.
21
22 % Para ello, primero de todo, hay que obtener los datos de los puntos y
23 % limpiarlos.
24 curva = readtable('datos_curva.csv', 'TextType', 'string', 'Delimiter', ',', '
    DecimalSeparator', ',');
25 puntos = readtable('datos_puntos.csv', 'TextType', 'string', 'Delimiter', ',', '
    DecimalSeparator', ',');
26
27 % Después, hay que lograr los valores (t,y) de la curva y de los puntos.
28 t1 = curva.x; y1 = curva.Curve1;
29 t2 = puntos.x; y2 = puntos.Curve1;
30
31 % Ya cuando tenemos todos los datos para hacer los gráficos, se hacen
32 % plot-s de todos los valores que hay, ajustando el color y tamaño de las
33 % curvas, y el color y la forma de los puntos.
34 figure;
35 plot(t1, y1, 'k-', 'LineWidth',2); hold on
36 plot(t2, y2, 'ko','MarkerFaceColor', 'k');
37 plot(Th,Yh(:,2),'b-', Th, Ye(:,2), 'r--', 'LineWidth',1.5);
38 plot(doses_time,Yh_doses(:,2),'bo', doses_time, Ye_doses(:,2), 'r*', '

```

```

    MarkerFaceColor', 'b')
39
40 % Para finalizar, se pone la leyenda, los ejes y el título, ajustando los ejes.
41 axis([0 250 0 0.25])
42 xlabel('Tiempo (días)', 'FontSize',16)
43 ylabel('$V(t)/H$', 'Interpreter','latex', 'FontSize',16)
44 title('Evolución del volumen del tumor', 'FontSize',16)
45 legend('Curva de la bibliografía','Momentos de dosis de la bibliografía','
    Aproximación', 'Exacto', 'location', 'best', 'FontSize', 16)
46 grid on
47 hold off

```

Código I.4: Código de plotter1.m.

```

1 % Para ver las diferencias, hay que quitar los valores repetidos de Th, y
2 % eso conlleva a quitar esos valores de Yh e Ye.
3 [Th_unique, pos] = unique(Th, 'stable');
4 Yh_unique = Yh(pos,:);
5 Ye_unique = Ye(pos,:);
6
7 % Entonces, ya se puede interpolar.
8 y_interp_h = interp1(Th_unique, Yh_unique(:,2), t1);
9 y_interp_e = interp1(Th_unique, Ye_unique(:,2), t1);
10
11 % Se hace la norma de y1, es decir, de los puntos de la gráfica, y su
12 % interpolación.
13 normal2(t1, y1, y_interp_h)
14 normal2(t1, y1, y_interp_e)

```

Código I.5: Código de diferencias.m.

I.2. Código de MATLAB de la Sección 2.3

En esta sección se detallan los archivos y funciones utilizados en la Sección 2.3 para hacer las Figuras 2.2, 2.3 y 2.4 y para calcular los errores.

```

1 % Limpiar todo y cerrar todas las figuras abiertas.
2 clc;
3 clear;
4 close all;
5
6 % Parámetros, dosis y tiempos de dosis.
7 par = struct('lambda', 9.242, 'Emax', 60, 'c50', 3.6e-1, 'r', 5.51e-3,...
8             'H', 100, 'T', 15, 'c0', 0, 'v0', 10);
9 doses = [50,80,80];

```

```

10 doses_time = [0, 6, 12];
11
12 [Th,Yh] = pkpd_ode45(par,doses,doses_time,[]);
13 Ye = pkpd_exact(Th,par,doses,doses_time);
14
15 % Gráficos de los datos conseguidos.
16 plotter2;
17
18 % Normas y errores de los datos.
19 norma;
20
21 % El orden de convergencia de la aproximación.
22 orden(par, doses, doses_time, @normaL2)

```

Código I.6: Código de PRIN2pkpd.m.

```

1 % En este script se hace el gráfico para los datos de PRIN12pkpd.m
2 % Primero de todo, hay que lograr los valores de los momentos en los que se
3 % administra una dosis, para poder dibujar ese momento, tal y como se hace
4 % en el fichero plotter1.m, pero en este caso dos veces, ya que hay que
5 % hacer dos gráficos diferentes.
6 Yh_doses = zeros(length(doses_time),2);
7 Ye_doses = zeros(length(doses_time),2);
8 for i = 1:length(doses_time)
9     all_yh_doses = Yh(Th==doses_time(i),:);
10    all_ye_doses = Ye(Th==doses_time(i),:);
11    if size(all_yh_doses)== [2 2]
12        Yh_doses(i,:) = all_yh_doses(2,:);
13        Ye_doses(i,:) = all_ye_doses(2,:);
14    else
15        Yh_doses(i,:) = all_yh_doses(:);
16        Ye_doses(i,:) = all_ye_doses(:);
17    end
18 end
19 % Se hacen los gráficos. Utilizando subplot, se logra que haya un gráfico
20 % arriba y otro abajo. Se ajustan los colores y el tipo de línea de los
21 % gráficos, como su grosor. También se ajustan los colores, formas y
22 % grosores de los puntos. Además se pone una leyenda, nombres de los ejes y
23 % título.
24 texto = ['$c(t)$' ; '$v(t)$'];
25 titulo = ['Evolución de la concentración del fármaco'; ...
26          'Evolución del volumen del tumor'];
27 figure;
28 for i = 1:2
29     subplot(2,1,i)
30     plot(Th,Yh(:,i),'b-', Th, Ye(:,i), 'r--', 'LineWidth',2); hold on
31     plot(doses_time,Yh_doses(:,i),'bo', doses_time, Ye_doses(:,i), 'r*')

```

```

32     xlabel('tiempo_(días)', 'FontSize',16)
33     ylabel(texto(i,:), 'Interpreter','latex', 'FontSize',16)
34     title(titulo(i,:), 'FontSize',16)
35     legend('Aproximación', 'Exacto', 'FontSize', 16)
36     grid on
37     hold off
38 end

```

Código I.7: Código de plotter2.m.

```

1  % En este script se logran algunas normas, para verificar que el método no
2  % tiene errores. Primero de todo, utilizando las funciones normaInfAbs.m,
3  % normaInfRes.m y normal2.m, se logran los resultados.
4
5  norm_Loo_abs = normaInfAbs(Th, Yh(:,2), Ye(:,2));
6  norm_Loo_rel = normaInfRel(Th, Yh(:,2),Ye(:,2));
7  norm_L2 = normal2(Th,Yh(:,2),Ye(:,2));
8
9  disp(['Norma_Loo_discreta_del_error_absoluto:_' num2str(norm_Loo_abs)])
10 disp(['Norma_Loo_discreta_del_error_relativo:_' num2str(norm_Loo_rel)])
11 disp(['Norma_L2_discreta_del_error_absoluto:_' num2str(norm_L2)])
12
13 % Cuando ya se saben los resultados, se hace un gráfico para saber cómo
14 % varía el error mientras avanza el tiempo. Para ello se usa el comando
15 % errorbar.
16 escala = 50;
17 tramo = 15;
18
19 figure;
20 plot(Th,Yh(:,2),'b-', Th, Ye(:,2), 'r--', 'LineWidth',1.5); hold on
21 errorbar(Th(1:tramo:end), Ye(1:tramo:end,2), escala * abs(Ye(1:tramo:end,2) -
    ...
22     Yh(1:tramo:end,2)),linestyle = 'None',LineWidth=1.5); hold off
23 xlabel('tiempo_(días)', 'FontSize',16); ylabel('$V(t)$', 'Interpreter', ...
24     'latex', 'FontSize',16)
25 legend('Aproximada', 'Exacta', 'Error', 'FontSize',16)
26 grid on

```

Código I.8: Código de norma.m.

```

1 function norma = normaInfAbs(~,Yh,Ye)
2 % Norma  $L^{\infty}$  discreta del error absoluto.
3 norma = norm(Ye-Yh, Inf);
4 end

```

Código I.9: Código de la función normaInfAbs.m.

```

1 function norma = normaInfRel(~, Yh, Ye)
2 % Norma L\infty discreta del error relativo.
3 norma = norm((Ye-Yh)./max(abs(Yh),eps), Inf);
4 end

```

Código I.10: Código de la función normaInfRel.m.

```

1 function norma = normaL2(Th, Yh, Ye)
2 % Norma L2 discreta del error absoluto.
3 % Tal y como se explica en la SubSección 2.3.2., este error se consigue
4 % utilizando la regla del trapecio, es decir, utilizando el comando trapz.
5 norma = sqrt(trapz(Th, (Ye-Yh).^2));
6 end

```

Código I.11: Código de la función normaL2.m.

```

1 function orden(par, doses, doses_time, norma)
2 % El orden de convergencia se logra haciendo la aproximación
3 %  $p = \log(e(i)/e(i+1))/\log(h(i)/h(i+1))$ . Para ello, utilizamos diferentes
4 % tolerancias.
5 tolerancias = [1e-7 1e-8 1e-9 1e-10 1e-11 1e-12];
6 h = zeros(size(tolerancias));
7 e = zeros(size(tolerancias));
8
9 for i = 1:length(tolerancias)
10     % Las tolerancias van a cambiar las opciones de ode45, y va a hacer que
11     % sea más refinado el resultado.
12     options = odeset('RelTol', tolerancias(i), 'AbsTol', tolerancias(i)*1e-3);
13     [Th, Yh] = pkpd_ode45(par, doses, doses_time, options);
14     Ye = pkpd_exact(Th, par, doses, doses_time);
15     % Como h no es uniforme, se coje una media.
16     h(i) = mean(diff(unique(Th)));
17     % La norma que se elija se utilizará para calcular el error.
18     e(i) = norma(Th, Yh, Ye);
19 end
20 p = log(e(2:end)./e(1:end-1))./log(h(2:end)./h(1:end-1));
21
22 % Se hace el gráfico, utilizando el p logrado y los pasos h.
23 figure;
24 plot(h(1:end-1), p, 'o-', 'LineWidth', 1.5)
25 axis([h(end) h(1) 5 7])
26 title('Orden de convergencia en norma L2', 'FontSize', 16)
27 xlabel('$h$', 'Interpreter', 'latex', 'FontSize', 16)
28 ylabel('$p$', 'Interpreter', 'latex', 'FontSize', 16)
29 end

```

Código I.12: Código de la función orden.m.

Anexo II

Código de MATLAB del Capítulo 4

En este anexo se explica con más detalle el código que se usa en el Capítulo 4. Primero de todo, se presenta la función objetivo, seguido por los códigos que se utilizarán en las Secciones 4.3.1 y 4.3.2, respectivamente.

En primer lugar, tenemos la función `funcion_objetivo.m` que se necesita en `fmincon`.

```
1 function V = funcion_objetivo(par, doses, doses_time)
2     [~, y] = pkpd_ode45(par, doses, doses_time, []);
3     V = y(end, 2);
4 end
```

Código II.1: Código de la función objetivo.

II.1. Código de MATLAB de la Sección 4.3.1

En esta sección se explica con detalle el código que se utiliza en la Sección 4.3.1 para hacer la Figura 4.1 y la Tabla 4.1.

```
1 % Limpiar todo y cerrar todas las figuras abiertas.
2 clc;
3 clear;
4 close all;
5
6 % Características del tratamiento.
7 n_ini = 29;
8 n_fin = 40;
9 diff = n_fin - n_ini + 1;
10 cant_dosis_ciclo = 5;
11 ciclos = 28;
```

```

12
13 % Parámetros del tratamiento.
14 par = struct('lambda', 9.242, 'r', 5.51e-3, 'Emax', 60, 'c50', 3.6e-1, 'T', 210, ...
15     'v0', 2.5e-1, 'c0', 0, 'H', 1, 'sigma', 4e-3, 'D', 5750);
16 dmin = par.sigma * 150;
17 dmax = par.sigma * 200;
18
19 % Vectores varios para obtener la Tabla 2.
20 n = n_ini:n_fin;
21 d_n = zeros(diff,1);
22 v_v0 = zeros(diff,1);
23 d_min = zeros(diff,1);
24 d_max = zeros(diff,1);
25 v_v0_opt = zeros(diff,1);
26
27 % Lo necesario para hacer la gráfica 1 de la Figura 1.
28 figure;
29 hold on;
30 k = 1;
31 col = 'kbr';
32 point = ['k*'; 'bo'; 'r*'];
33 names = ['30□(UT)'; '□□35□□□'; '□□38□□□'];
34 % col y point son para darle el color y forma de los puntos a los gráficos.
35 % names es para poner en la leyenda.
36
37 % Bucle para N \in [29,40].
38 for i = n_ini:n_fin
39
40     % Restricción que logramos en el Capítulo 3: D/N tiene que estar entre
41     % d_min y d_max.
42     if par.sigma*(par.D/i)<dmin || par.sigma*(par.D/i)>dmax
43         continue
44     end
45
46     % Dosis para cada N.
47     doses = (par.D/i)*ones(1,i)*par.sigma;
48
49     % Cálculo de momentos en el que se administran las dosis para cada N.
50     doses_time = [];
51     ciclo_entero = floor(i/cant_dosis_ciclo);
52     ciclo_sin_completar = mod(i,cant_dosis_ciclo);
53     for j = 1:ciclo_entero
54         doses_time = [doses_time, (0:4) + (j-1)*ciclos];
55     end
56     if ciclo_sin_completar > 0
57         doses_time = [doses_time, 28*ciclo_entero+(0:ciclo_sin_completar-1)];
58     end

```

```

59
60 % Argumentos de la función fmincon, como se han explicado en la
61 % Sección 4.2. Hay que multiplicar por sigma cada vez que tenemos
62 % D, o alguna dosificación, para lograr los mismos resultados que en la
63 % bibliografía.
64 fun = @(x) funcion_objetivo(par,x,doses_time);
65 Aeq = ones(1,i);
66 beq = par.D*par.sigma;
67 lb = dmin*ones(1,i);
68 ub = dmax*ones(1,i);
69 options = optimoptions('fmincon','Algorithm','sqp');
70 % En este caso no se usa 'Display','Iter' por comodidad.
71
72 % Optimización de nuestro problema.
73 [x_opt,fval] = fmincon(fun,doses,[],[],Aeq,beq,dmin,dmax,[],options);
74
75
76 % Soluciones para la Tabla 2.
77 d_n(i-n_ini+1) = par.D/i;
78 [~, Yh] = pkpd_ode45(par, doses, doses_time, []);
79 v_v0(i-n_ini+1) = Yh(end,2)/par.v0;
80 d_min(i-n_ini+1) = min(x_opt)/par.sigma;
81 d_max(i-n_ini+1) = max(x_opt)/par.sigma;
82 [Th, Yh_opt] = pkpd_ode45(par, x_opt, doses_time, []);
83 v_v0_opt(i-n_ini+1) = Yh_opt(end,2)/par.v0;
84
85 % Gráfico.
86
87 % Se van a hacer los gráficos de N = 30, 35 y 38 solamente.
88 if ismember(i,[30,35,38])
89     % Para lograr el volumen que tiene en el momento que se administran
90     % las dosis, para poder poner los puntos.
91     Yh_doses = zeros(1,length(doses_time));
92     for i = 1:length(doses_time)
93         all_yh_doses = Yh_opt(Th==doses_time(i),2);
94         if length(all_yh_doses)==2
95             Yh_doses(i) = all_yh_doses(2);
96         else
97             Yh_doses(i) = all_yh_doses;
98         end
99     end
100
101     % Gráfico de la evolución del volumen, y los puntos.
102     plot(Th,Yh_opt(:,2), col(k), 'LineWidth',1,'HandleVisibility','off');
103     plot(doses_time, Yh_doses, point(k,:), 'LineWidth',1, 'LineStyle', ...
104         'none','DisplayName',names(k,:));
105     % En el plot de la evolución del volumen aparece

```

```

106     % 'handleVisibility','off', porque en la leyenda no queremos que
107     % aparezcan. Es suficiente con la leyenda de los puntos.
108     % En el plot de los puntos aparece 'LineStyle', 'none', porque solo
109     % queremos que aparezcan puntos, no lineas.
110     k = k+1;
111     end
112 end
113
114 % Matriz para visualizar mejor los datos obtenidos, más como en una tabla.
115 tabla = [n',d_n,v_v0,d_min,d_max,v_v0_opt];
116
117 % Leyenda, título, nombre de cada eje...
118 hold off
119 grid on
120 legend('show', 'location', 'southwest','FontSize',16)
121 xlabel('tiempo (días)', 'FontSize',16)
122 ylabel('$V(t)/H$', 'Interpreter','latex', 'FontSize',16)
123 title('Minimizar el volumen del tumor', 'FontSize',16)

```

Código II.2: Código de apartado4_3_1.m.

II.2. Código de MATLAB de la Sección 4.3.2

En esta sección se explica con detalle el código que se utiliza en la Sección 4.3.2, para hacer la Figura 4.2 y la Tabla 4.2.

```

1 % Limpiar todo y cerrar todas las figuras abiertas.
2 clc;
3 clear;
4 close all;
5
6 % Características de los tratamientos.
7 cant_dosis_ciclo = [7, 21, 21, 28];
8 ciclos = [14, 28, 28, 28];
9
10 % Parámetros de los tratamientos.
11 par = struct('lambda', 9.242, 'r', 5.51e-3, 'Emax', 60, 'c50', 3.6e-1, 'T', 210, ...
12     'v0', 2.5e-1, 'c0', 0, 'H', 1, 'sigma', 4e-3, 'D', 5750);
13 dmin = [150, 100, 75, 50];
14
15 % Vectores varios para obtener la Tabla 3.
16 v_v0 = zeros(4,1);
17 dose_intensity = zeros(4,1);
18
19 % Vectores obtenidos para la Tabla 3.

```

```

20 N_hat = floor(par.D./dmin);
21 D_N_hat = par.D ./ N_hat;
22
23 % Lo necesario para hacer la gráfica 1 de la Figura 1.
24 figure;
25 hold on
26 k = 1;
27 col = 'bkgr';
28 point = ['bo','k*','gx','r*'];
29 names = ['38|151.32','57|100.88','76|75.88','115|50'];
30 % col y point son para darle el color y forma de los puntos a los gráficos.
31 % names es para poner en la leyenda.
32
33 % Bucle para cada tratamiento.
34 for i = 1:length(dmin)
35
36     % Dosis para cada tratamiento.
37     doses = D_N_hat(i)*ones(1,N_hat(i))*par.sigma;
38
39     % Cálculo de momentos en el que se administran las dosis para cada N.
40     doses_time = [];
41     ciclo_entero = floor(N_hat(i)/cant_dosis_ciclo(i));
42     ciclo_sin_completar = mod(N_hat(i),cant_dosis_ciclo(i));
43     for j = 1:ciclo_entero
44         doses_time = [doses_time, (0:cant_dosis_ciclo(i)-1) + (j-1)*ciclos(i)];
45     end
46     if ciclo_sin_completar > 0
47         doses_time = [doses_time, ciclos(i)*ciclo_entero+(0:ciclo_sin_completar
48             -1)];
49     end
50
51     % Resultados de ode45.
52     [Th,Yh] = pkpd_ode45(par, doses, doses_time, []);
53
54     % Soluciones que faltaban para la Tabla 3.
55     v_v0(i) = Yh(end,2)/par.v0;
56     dose_intensity(i) = par.D/(doses_time(end)+1);
57     % En la teoría se explica mejor lo que significa la intensidad de
58     % dosis.
59
60     % Gráfico.
61
62     % Para lograr el volumen que tiene en el momento que se administran las
63     % dosis, para poder poner los puntos.
64     Yh_doses = zeros(1,length(doses_time));
65     for i = 1:length(doses_time)
66         all_yh_doses = Yh(Th==doses_time(i),2);

```

```

66         if length(all_yh_doses)==2
67             Yh_doses(i) = all_yh_doses(2);
68         else
69             Yh_doses(i) = all_yh_doses;
70         end
71     end
72
73     % Gráfico de la evolución del volumen, y los puntos.
74     plot(Th,Yh(:,2), col(k), 'LineWidth',1,'HandleVisibility', 'off');
75     plot(doses_time, Yh_doses, point(k,:), 'LineWidth',1, 'LineStyle', 'none',
76         ...
77         'DisplayName',names(k,:));
78     % En el plot de la evolución del volumen aparece
79     % 'handleVisibility','off', porque en la leyenda no queremos que
80     % aparezcan. Es suficiente con la leyenda de los puntos.
81     % En el plot de los puntos aparece 'LineStyle', 'none', porque solo
82     % queremos que aparezcan puntos, no líneas.
83     k = k+1;
84 end
85
86 % Matriz para visualizar mejor los datos obtenidos, más como en una tabla.
87 tabla = [N_hat',D_N_hat',v_v0,dose_intensity];
88
89 % Leyenda, título, nombre de cada eje...
90 hold off
91 grid on
92 legend('show', 'location', 'northeast','FontSize',16)
93 xlabel('tiempo (días)', 'FontSize',16)
94 ylabel('$V(t)/H$', 'Interpreter','latex', 'FontSize',16)
95 title('Minimizar el volumen del tumor', 'FontSize',16)

```

Código II.3: Código de apartado4_3_2.m.