PRACTICA 3

1. **IDENTIFICACIÓN: MODELO MÍNIMO GLUCOSA-INSULINA**

|  |  |  |
| --- | --- | --- |
| **COMPETENCIAS** | **CONTENIDO TEMÁTICO** | **INDICADOR DE LOGRO** |
| Comprender la modelación del proceso Glucosa-Insulina.  Desarrollar habilidades en la aplicación del Modelo mínimo Glucosa-Insulina. | Modelo mínimo Glucosa-Insulina (modificado)  Solución de sistemas de ecuaciones diferenciales (ODEs) mediante Euler. | Desarrollar en Python mediante Google Colab, un programa que permita simular el proceso Glucosa-Insulina mediante el Modelo mínimo. |

1. **RECURSOS REQUERIDOS**

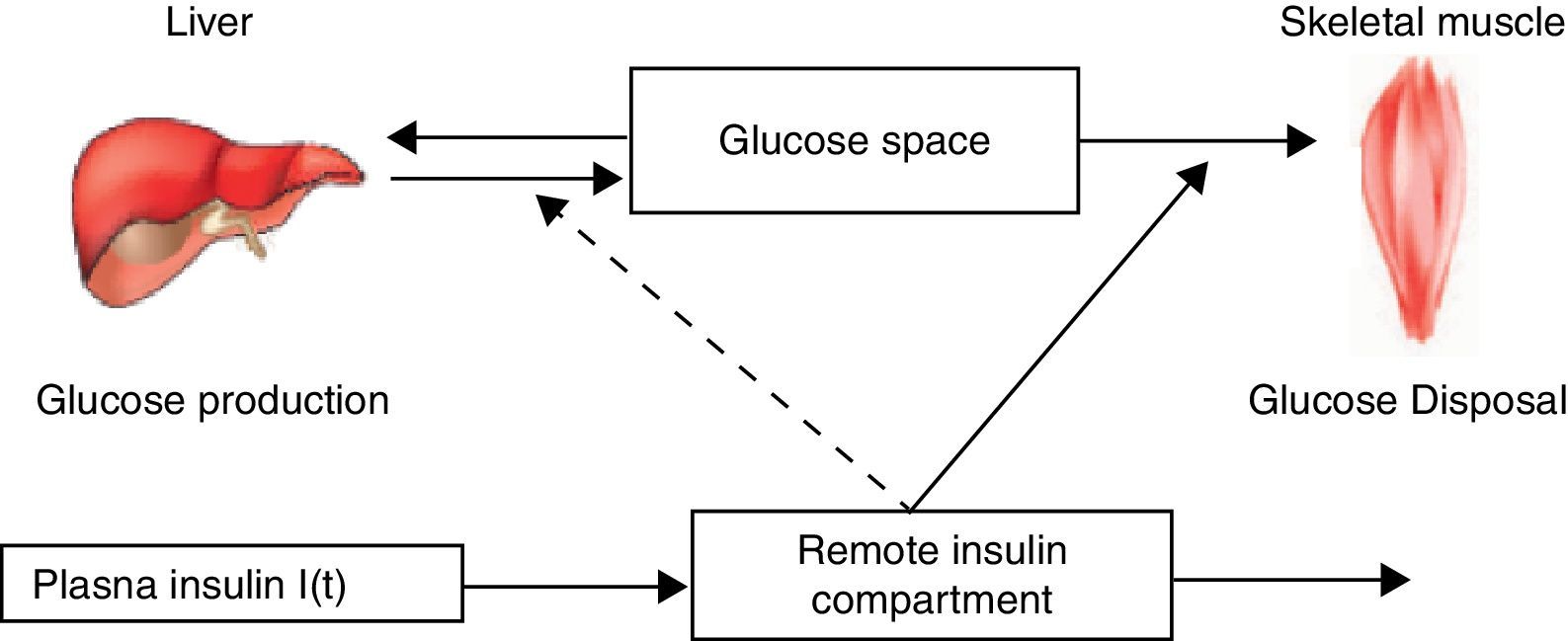
Requeridos: Python y Google Colab.

Adicionales: Octave o Matlab®.

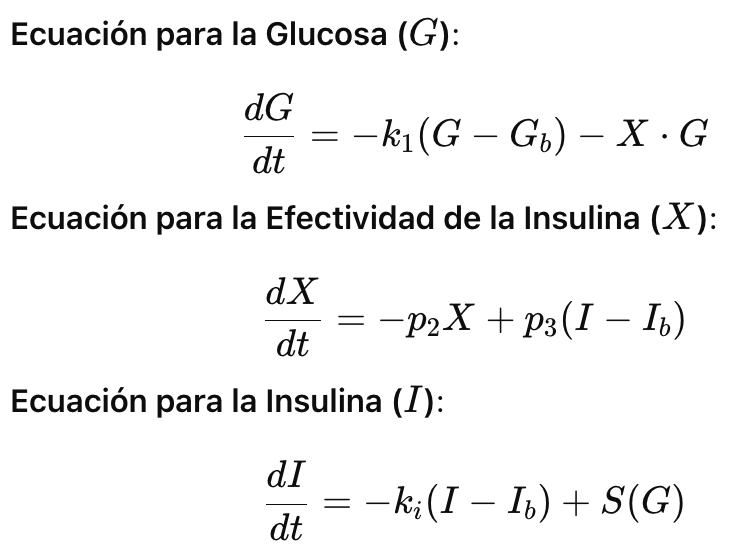
1. **CONTENIDO TEMÁTICO**

**3.1 Modelo mínimo Glucosa-Insulina (modificado)**

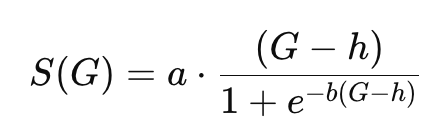
Simula cómo las concentraciones de glucosa e insulina, así como la efectividad de la insulina, cambian en el tiempo en respuesta a ciertas condiciones iniciales y parámetros específicos.



El siguiente sistema de ecuaciones captura la interacción dinámica entre la glucosa y la insulina en el cuerpo, incluyendo la eliminación de glucosa, la captación de glucosa mediada por la insulina, la secreción de insulina en respuesta a la glucosa y la degradación tanto de la insulina como de su efectividad.



Donde la función de secreción de insulina *S(G)* está definida como:



**Variables de Estado:**

*G* : Concentración de glucosa en sangre (en mg/dL).

*X* : Efectividad de la insulina, que refleja cómo la insulina facilita la captación de glucosa por los tejidos.

*I* : Concentración de insulina en sangre (en μU/mL).

**Parámetros:**

*k1​* :Tasa de eliminación de la glucosa independiente de la insulina.

*Gb* :Nivel basal de glucosa (concentración de glucosa en ayuno).

*P2* : Tasa de decaimiento de la efectividad de la insulina.

*p3* : Tasa de aumento de la efectividad de la insulina en respuesta a la insulina.

*Ib* : Nivel basal de insulina (concentración de insulina en ayuno).

*ki​* : Tasa de eliminación de la insulina.

*a y b* : Parámetros que controlan la secreción de insulina en respuesta a los niveles de glucosa.

*h* : Umbral de glucosa que activa la secreción de insulina.

**Explicación:**

*k1​⋅(G−Gb​)* : Representa la tasa de eliminación de glucosa en ausencia de insulina. Si *G > Gb,* la glucosa se elimina más rápidamente.

*X⋅G* :Término que describe cómo la efectividad de lainsulina *(X)* incrementa la captación de glucosa, reduciendo la concentración de glucosa en sangre.

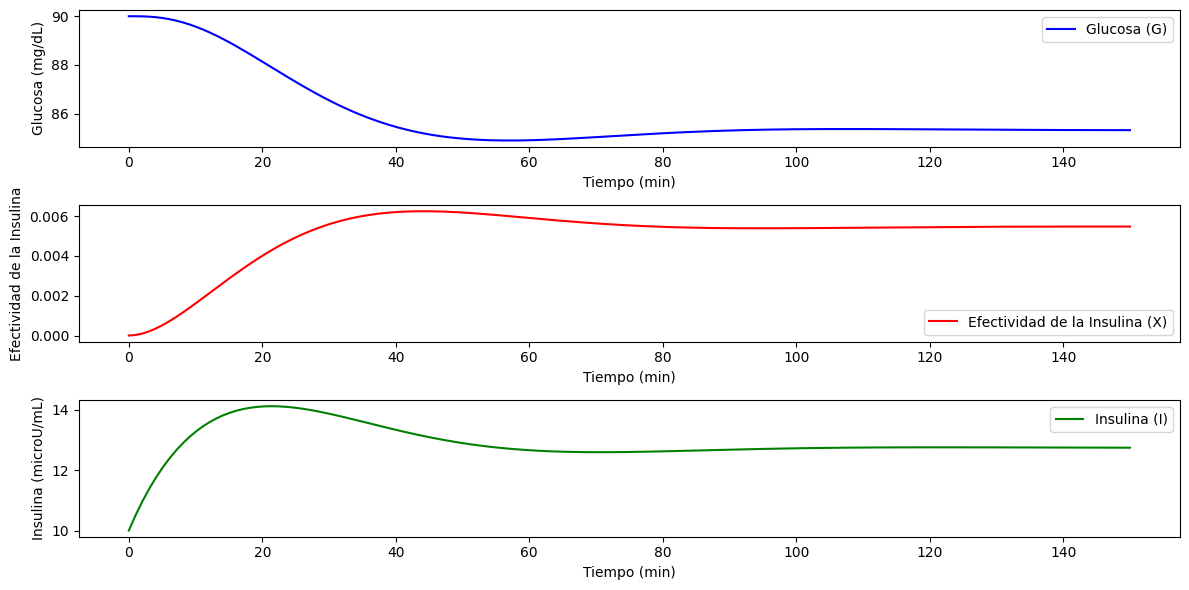
*−p2⋅X* :Tasa de decaimiento de la efectividad de la insulina. Si no hay insulina adicional, la efectividad disminuye con el tiempo.

*p3⋅(𝐼−𝐼b)* :Incremento en la efectividad de la insulina debido a la presencia de insulina. Si *𝐼 >𝐼b,* la efectividad aumenta.

*−ki⋅(I−Ib)* : Tasa de eliminación de la insulina. Si *𝐼 >𝐼b,* la insulina se elimina más rápido.

*S(G)*: Secreción de insulina dependiente de la glucosa. Describe cómo el páncreas secreta insulina en respuesta a la concentración de glucosa *G*, de manera más activa cuando *G* está por encima del umbral *h*. Es una función sigmoidal (logística) que asegura que la secreción de insulina crezca más rápidamente cuando *G* excede *h*, pero se estabilice a niveles altos de glucosa.

El siguiente es un ejemplo de la solución al sistema de ecuaciones:



**3.2 Sistemas de EDOs**

Un sistema de EDOs (Sistema de Ecuaciones Diferenciales Ordinarias) es un conjunto de ecuaciones diferenciales ordinarias que describe cómo varias variables cambian con respecto a una variable independiente, generalmente el tiempo. En un sistema de EDOs, cada ecuación relaciona una de las variables dependientes con la variable independiente y posiblemente con otras variables dependientes del sistema. Tales sistemas en general se representan como:

*dy*1 = *f*1(*x*, *y*1, *y*2, …, *yn*)

*dx*

*dy*2 = *f*2(*x*, *y*1, *y*2, …, *yn*)

*dx*

*…*

*dyn* = *fn*(*x*, *y*1, *y*2, …, *yn*)

*dx*

La solución de este sistema requiere que se conozcan *n* condiciones iniciales en el valor inicial de *x*.

Los métodos para una EDO pueden extenderse al sistema. Las aplicaciones en la ingeniería llegan a considerar miles de ecuaciones simultáneas. En todo caso, el procedimiento para resolver un sistema de ecuaciones consiste únicamente en aplicar la técnica simple por ecuación en cada paso, antes de proceder con el siguiente.

Supongamos que se han calculado los valores *y*1,*j*, *y*2,*j*, ..., *ym,j*, donde *j* es el contador del tiempo, si tiene mediante el método de Euler las siguientes ecuaciones para resolver el sistemas de EDOs:

*k*1,*i* =  *fi* (*tj*, *y*1,*j*, *y*2,*j*, ..., *ym*,*j*), para cada *i* = 1, 2, ..., *m*

*yi*,*j*+1 = *yi*,*j* + *h* (*k*1,*i*) *,*para cada *i* = 1, 2, ..., *m*

Euler es un método explícito para la solución de ecuaciones diferenciales. Este método estima el valor de la solución en el instante *j+*1 a partir de la exploración del valor de la ecuación diferencial en el instante *j*, tangente de la solución, por un paso de integración *h*. La aproximación depende del tamaño del paso del problema.

1. BIBLIOGRAFÍA

* Ewart Carson and Claudio Cobelli. Modelling Methodology for Physiology and Medicine. Acedemic Press: San Diego, 2001.
* Steven C. Chapra, Raymond P. Canale. Métodos Numéricos para ingenieros. 5ª ed. Mc Graw Hill: México 2007.

1. **PROCEDIMIENTO SUGERIDO**

Desarrollar una rutina en Paython mediante Google Colab para encontrar la solución a las ecuaciones diferenciales que describen el Modelo mínimo Glucosa-Insulina.

Procedimiento sugerido:

* Iniciar sesión en Google Colab, inicializar un código nuevo y asignarle nombre.
* Importar las librerías necesarias (numpy, matplotlib)
* Crear una función para la secreción de insulina, la cual debe retornar el valor de la evaluación de la ecuación *S(G)*, para lo cual se requiere del valor de la Glucosa *G* en cada instante de tiempo. Implementar los siguientes valores de los parámetros:

*a* = 0.1

*b* = 0.01

*h* = 80

* Crear una función para resolver el sistema de 3 EDOs mediante Euler, para lo cual se requiere de los valores iniciales de *G*, *X* e *I*, y del vector tiempo *t*. Implementar los siguientes valores de los parámetros:

*k1 = 0.1*

*p2 = 0.05*

*p3 = 0.0001*

*ki = 0.1*

*Gb = 90*

*Ib = 10*

Posteriormente, crear un bucle *for* donde *i* es un contador desde 0 hasta la longitud del vector *t -1* para aplicar el método de Euler asumiendo un tamaño de paso de 1 minuto y conocer así el valor de *G*, *X* e *I* en los instantes de tiempo *i+1*

* El vector de tiempo *t* se puede crear mediante la instrucción np.linspace, simular hasta el minuto 150.
* Los vectores de *G, X* e *I* pueden inicializarse mediante la instrucción np.zeros, para posteriormente asignar los siguientes valores iniciales:

*G*[0] = 90

*X*[0] = 0

*I*[0] = 10

* Graficar la Glucosa *G*, Efectividad de la Insulina *X* y la Insulina *I* versus tiempo *t*, para lo cual se recomiendan las instrucciones plt.subplot, plt.plot y plt.legend()

**Informe**

* Realizar tres simulaciones modificando ya sea alguna condición inicial ó algún valor de los parámetros.
* Entregar el código del programa desarrollado y para cada uno de los ensayos entregar los datos utilizados y las figuras obtenidas.
* Entregar conclusiones de los resultados.
* Nota: la práctica puede ser realizada en su totalidad en Google Colab.

|  |  |
| --- | --- |
| ***Elaborado por:*** | ***Catalina Tobón Zuluaga*** |

NOTA

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**HOJA DE DESARROLLO**