

Modelado Metabólico de *Hermetia illucens* mediante Redes Neuronales Informadas por la Física (PINNs)

Metodología de Análisis y Resultados

3 de enero de 2026

1. Introducción

El objetivo de este estudio fue inferir las variables metabólicas latentes (no observables) del crecimiento larvario de la Mosca Soldado Negra (*Hermetia illucens*) bajo dos dietas distintas (D1 y D4). Para ello, se utilizó una Red Neuronal Informada por la Física (PINN) basada en el modelo dinámico de Eriksen (2022), utilizando datos experimentales de respiración.

2. Metodología de Procesamiento de Datos

El flujo de trabajo para transformar las mediciones crudas de los sensores en datos aptos para el modelado computacional se diseñó para gestionar la variabilidad experimental y el ruido instrumental. El proceso constó de tres etapas secuenciales: extracción recursiva, cálculo de tasas metabólicas y corrección de línea base.

2.1. Estructura de Directorios y Extracción de Archivos

Los datos experimentales primarios no se encontraban centralizados, sino distribuidos en una estructura jerárquica de directorios basada en el tiempo de desarrollo larvario. Se desarrolló un algoritmo de búsqueda recursiva en Python para iterar sobre esta estructura.

El conjunto de datos abarcó cuatro puntos temporales críticos, correspondientes a las carpetas raíz: **Día 9, Día 11, Día 13 y Día 17**. Dentro de cada carpeta diaria, el algoritmo procesó múltiples subdirectorios correspondientes a las réplicas biológicas, siguiendo esta estructura lógica:

- **Nivel 1 (Tiempo):** Carpeta del día (e.g., /9, /11).
- **Nivel 2 (Unidad Experimental):** Subcarpetas con nomenclatura variable (e.g., `experimento1_D1_voraz`, `control_sustrato_D4`). Un algoritmo de procesamiento de texto normalizó estos nombres en cuatro categorías estandarizadas:
 - `D1_Tratamiento` y `D4_Tratamiento`
 - `Control_Alimento_D1` y `Control_Alimento_D4`

- **Nivel 3 (Datos Crudos):** En cada unidad experimental se extrajeron dos archivos específicos:
 - `experimento_co2.csv`: Dinámica de CO₂.
 - `experimento_ch4.csv`: Dinámica de CH₄.

Cabe destacar que, a pesar de la extensión `.csv`, los archivos crudos poseían un formato de valores separados por tabuladores (TSV) sin encabezados. De cada archivo se extrajo la serie temporal completa (t, C) , donde t representa el tiempo de medición y C la concentración acumulada en ppm.

2.2. Análisis de Curvas y Cálculo de Tasas

La variable de interés biológico para el modelo de Eriksen no es la concentración absoluta, sino la **tasa de producción** (la velocidad a la que las larvas generan gas). Para obtener este valor escalar a partir de las series temporales extraídas, se aplicó el siguiente protocolo a cada archivo:

1. **Filtrado de Saturación del Sensor:** Se identificó una limitación instrumental en el sensor de CO₂ con un techo de medición cercano a los 6000 ppm. Para evitar subestimar la pendiente debido al “planamiento” de la curva por saturación, se aplicó un filtro de corte duro (hard threshold):

$$Datos_{validos} = \{(t, C) \mid C < 5850 \text{ ppm}\} \quad (1)$$

2. **Regresión Lineal (Cálculo de Pendiente):** Sobre los datos válidos, se realizó una regresión lineal por mínimos cuadrados para obtener la pendiente m , que representa la tasa de producción bruta:

$$\frac{d[Gas]}{dt} \approx m = \frac{\sum(t_i - \bar{t})(C_i - \bar{C})}{\sum(t_i - \bar{t})^2} \quad (2)$$

El resultado es una tasa en unidades de ppm/min para cada réplica experimental.

Este proceso redujo miles de puntos de datos temporales a un único valor representativo por archivo, consolidándose todo en el archivo intermedio `datos_finales_PINN_corregidos.csv`.

2.3. Corrección de Línea Base (Resta de Controles)

Finalmente, para aislar la respiración neta de las larvas de la emisión de fondo del sustrato (fermentación microbiana), se aplicó una sustracción dinámica pareada.

A diferencia de restar una constante, se restó el promedio de los controles correspondientes al **mismo día** y **mismo gas**. Sea R_{total} la tasa calculada para un tratamiento con larvas y $\bar{R}_{control}$ el promedio de las tasas de los sustratos solos para ese día:

$$R_{neta} = \max(0, R_{total} - \bar{R}_{control}) \quad (3)$$

El uso de la función $\max(0, \cdot)$ asegura la consistencia física (la respiración no puede ser negativa). El resultado final se exportó como `datos_PINN_netos.csv`, conteniendo exclusivamente las tasas metabólicas netas de CO₂ listas para el entrenamiento de la red neuronal.

3. Modelo Matemático (Eriksen 2022)

La PINN se fundamentó en la ecuación diferencial ordinaria (EDO) propuesta por Eriksen et al. (2022), que descompone la respiración en tres costos energéticos:

$$\frac{dCO_2}{dt} = Y_g \cdot \frac{dB}{dt} + Y_l \cdot \frac{dL}{dt} + m \cdot B \quad (4)$$

Donde:

- $B(t)$: Biomasa estructural.
- $L(t)$: Reservas de lípidos.
- Y_g, Y_l, m : Parámetros de rendimiento y mantenimiento a ser inferidos por la IA.

4. Implementación de la PINN

Se implementó una red neuronal profunda en PyTorch (clase `LarvaNet`) con la siguiente arquitectura:

- **Entrada:** Tiempo (t) normalizado (escalado Min-Max).
- **Capas Ocultas:** 2 capas densas de 32 neuronas con activación Tanh.
- **Salida:** 2 neuronas representando $B(t)$ y $L(t)$.

La función de pérdida incluyó el residuo de la EDO física y una regularización para asegurar positividad biológica.

5. Resultados y Discusión

5.1. Parámetros Metabólicos Descubiertos

La red neuronal convergió exitosamente (Loss ≈ 0 para D4), permitiendo extraer los siguientes coeficientes bioenergéticos:

Cuadro 1: Parámetros Bioenergéticos Inferidos por la IA

Parámetro	Tratamiento D1 (Estrés)	Tratamiento D4 (Óptimo)
m (Mantenimiento)	0.69581	0.58429
Y_g (Costo Crecimiento)	0.96518	0.85961
Y_l (Costo Lípidos)	0.85845	0.75689

5.2. Interpretación Biológica

1. **Costo de Mantenimiento:** El tratamiento D1 presentó un costo basal (m) un **19.1 % mayor** que el D4, indicando un estado de estrés metabólico.
2. **Eficiencia de Crecimiento:** El parámetro Y_g fue menor en D4, lo que implica una mayor eficiencia en la conversión de sustrato a biomasa estructural.

3. Dinámica de Lípidos: La reconstrucción de variables latentes mostró que el pico de CO₂ en el día 11 del tratamiento D4 correspondió a una alta tasa de síntesis de lípidos y crecimiento, mientras que en D1 se observó una movilización (consumo) de reservas.

A. Código Fuente Relevante

A.1. Extracción de Datos y Cálculo de Pendientes

```
1 def procesar_archivo_tsv(ruta_completa, dia, etiqueta, gas):
2     # Lectura de TSV sin encabezado
3     df = pd.read_csv(ruta_completa, sep='\t', header=None)
4     df_numericos = df.iloc[:, 1:1].copy() # Omitir columna de hora
5
6     # ... (Logica de iteracion) ...
7
8     # Filtro de Saturacion
9     mask_validos = y_raw < 5850
10    y_final = y_raw[mask_validos]
11
12    # Calculo de pendiente (Tasa de Produccion)
13    slope, _, _, _, _ = linregress(x_final, y_final)
14
15    return resultados
```

A.2. Definición de la PINN

```
1 class LarvaNet(nn.Module):
2     def __init__(self):
3         super().__init__()
4         self.net = nn.Sequential(
5             nn.Linear(1, 32), nn.Tanh(),
6             nn.Linear(32, 32), nn.Tanh(),
7             nn.Linear(32, 2)
8         )
9         # Parametros iniciales
10        self.raw_params = nn.Parameter(torch.tensor([0.5, 0.3, 0.1]))
11
12    def physics_loss(model, t, co2_medido):
13        B, L = model(t)
14        # Diferenciacion Automatica para obtener tasas dB/dt y dL/dt
15        dB_dt = torch.autograd.grad(B, t, grad_outputs=torch.ones_like(B),
16                                    create_graph=True)[0]
16        dL_dt = torch.autograd.grad(L, t, grad_outputs=torch.ones_like(L),
17                                    create_graph=True)[0]
17
18        Y_g, Y_l, m = model.get_physics_params()
19
20        # Ecuacion de Eriksen
21        co2_pred = (Y_g * dB_dt) + (Y_l * dL_dt) + (m * B)
22
23    return torch.mean((co2_pred - co2_medido) ** 2), co2_pred
```