

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**Aplicación para predecir la viabilidad de los huevos de
mosquito con inteligencia artificial**

Trabajo de graduación presentado por Gabriel Rolando Quiroz
Albizures para optar al grado académico de Licenciado en Ingeniería en
Ciencias de la Computación y TI

Guatemala,

2023

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



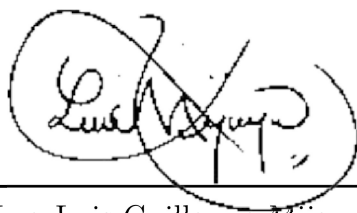
**Aplicación para predecir la viabilidad de los huevos de
mosquito con inteligencia artificial**

Trabajo de graduación presentado por Gabriel Rolando Quiroz
Albizures para optar al grado académico de Licenciado en Ingeniería en
Ciencias de la Computación y TI

Guatemala,

2023

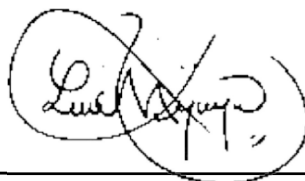
Vo.Bo.:



(f)

Ing. Luis Guillermo Mijangos

Tribunal Examinador:



(f)

Ing. Luis Guillermo Mijangos



(f)

Ing. Douglas Barrios



(f)

Ing. Sebastian Galindo

Fecha de aprobación: Guatemala, 7 de diciembre de 2023.

El monitoreo de las poblaciones de mosquitos es crucial para obtener más información sobre ellos y sus enfermedades. Tradicionalmente, el conteo y análisis de la viabilidad de los huevos de mosquito ha sido un proceso manual laborioso. Este proyecto surge de una colaboración con el Centro de Estudios en Salud, de la Unidad de Entomología Médica y Malaria, del Instituto de Investigaciones de la Universidad del Valle de Guatemala, con la intención de automatizar y agilizar este proceso.

El objetivo principal es desarrollar una aplicación web que permita realizar y almacenar predicciones y conteos de huevos de mosquito de manera eficiente. Se persiguen dos metas específicas: identificar un modelo que pueda predecir la viabilidad de los huevos, y crear una plataforma digital que facilite la captura, almacenamiento y análisis de los datos recolectados.

A través de esta iniciativa, se espera proporcionar una herramienta valiosa para los biólogos y otros profesionales involucrados en el estudio y control de las poblaciones de mosquitos, además de fomentar la colaboración interdisciplinaria entre las áreas de biología y ciencias de la computación.

Además, quiero expresar mi profundo agradecimiento a Ana Cristina Benedict Villagrán, Adriana Echeverría y la Dra. Norma Renée Padilla por su confianza y ayuda con los sustratos e ingresos al Centro de Estudios en Salud, de la Unidad de Entomología Médica y Malaria, del Instituto de Investigaciones de la Universidad del Valle de Guatemala. Un especial agradecimiento a Erlin Ayala por proporcionar el equipo de fotografía y sus valiosos consejos. Asimismo, mi gratitud al Ingeniero Luis Guillermo Mijangos por su guía y consejos en la elaboración de este proyecto. Finalmente, mi gratitud a mis compañeros y a mi familia, cuyo apoyo incondicional y aliento han sido fundamentales siempre.

Prefacio	v
Lista de figuras	xii
Lista de cuadros	xiii
Resumen	xv
Abstract	xvii
1. Introducción	1
2. Antecedentes	3
3. Justificación	5
4. Objetivos	7
4.1. Objetivos generales	7
4.2. Objetivos específicos	7
5. Marco teórico	9
5.1. Biología y comportamiento de los mosquitos	9
5.1.1. Ciclo de vida	9
5.1.2. Puesta de huevos	11
5.1.3. Puesta de huevos en condiciones de laboratorio	11
5.1.4. Biología y factores de reproducción	12
5.2. Modelos de Inteligencia Artificial para predicción	13
5.2.1. Máquinas de Soporte Vectorial	13
5.2.2. Random Forest	14
5.2.3. Redes Neuronales Artificiales	14
5.2.4. Regresión lineal	15
5.3. Validación y evaluación del modelo de predicción	15
5.3.1. Evaluación	15
5.3.2. Validación cruzada	15

5.3.3.	Análisis de residuos	15
5.4.	Desarrollo de aplicación web y modelo	16
5.4.1.	Lenguajes de desarrollo	16
5.5.	Manejo de Usuarios	17
5.5.1.	Firebase	17
5.6.	Frontend	18
5.6.1.	HTML y CSS	18
5.6.2.	Angular	18
5.6.3.	NodeJS	18
5.7.	Backend	18
5.7.1.	Express	19
5.7.2.	TensorFlowJS	19
5.7.3.	Postman	19
5.7.4.	Uptime Robot	19
5.7.5.	Flask	19
5.8.	Base de Datos	20
5.8.1.	MySQL	20
5.9.	Hosting Web	20
5.9.1.	Vercel	20
5.10.	Responsible AI	20
5.10.1.	Origen de los datos	21
5.10.2.	Limitaciones	21
6.	Metodología	23
6.1.	Gestión de proyecto	23
6.1.1.	Adaptación de la metodología Scrum	23
6.2.	Proceso de conteo de huevos	25
6.3.	Toma de fotografías	26
6.3.1.	Métodos	27
6.4.	Creación del Dataset	29
6.5.	Preprocesamiento del Dataset	31
6.6.	Desarrollo y evaluación de los modelos	32
6.6.1.	Máquina de soporte vectorial	32
6.6.2.	Random Forest	33
6.6.3.	Redes Neuronales Artificiales	33
6.6.4.	Regresión Lineal	34
6.7.	Implementación de Base de Datos	35
6.7.1.	Tabla de usuarios	35
6.7.2.	Tabla de datos generales	35
6.7.3.	Tabla de conteos	35
6.8.	Desarrollo del backend	36
6.8.1.	Gestión de usuarios	36
6.8.2.	Gestión de conteos	37
6.8.3.	Integración con el modelo	37
6.8.4.	Base de Datos y seguridad	37
6.8.5.	Monitoreo y pruebas de eficiencia	37
6.9.	Desarrollo del registro de usuarios	38
6.10.	Cálculo de la densidad poblacional de huevos en un sustrato	38

6.10.1. Densidad poblacional de cada tipo de huevo como variable de entrada en los modelos	39
6.11. Validación cruzada para los modelos destacados	40
6.12. Análisis de residuos	40
7. Resultados	43
7.1. Enfoque utilizando porcentaje	43
7.1.1. Random Forest	43
7.1.2. Máquina de Vectores (SVM)	45
7.1.3. Red Neuronal Artificial (ANN)	47
7.1.4. Regresión Lineal	49
7.2. Enfoque utilizando densidad poblacional	50
7.2.1. Random Forest	50
7.2.2. Máquina de Vectores (SVM)	51
7.2.3. Red Neuronal Artificial (ANN)	53
7.2.4. Regresión Lineal	55
7.3. Justificación de modelo elegido	56
7.3.1. Validación cruzada	56
7.3.2. Análisis de residuos	57
7.4. Métricas del backend	58
7.5. Inicio de sesión y manejo de usuarios	59
7.6. Base de datos	62
8. Conclusiones	63
9. Recomendaciones	65
10. Bibliografía	67
11. Anexos	71
11.1. Manual de Usuario para la aplicación web	71
11.1.1. Acceso a la plataforma	71
11.1.2. Navegación principal	71
11.1.3. Realizar conteo	71
11.1.4. Historial:	72

Lista de figuras

1.	Ciclo de vida del mosquito	10
2.	Funcionamiento de la SVM	13
3.	Funcionamiento de Random Forest	14
4.	Tablero en Azure	24
5.	Reunión periódica con el cliente	25
6.	Inspección y conteo	26
7.	Foto tomada usando iPhone 12 Mini y estereoscopio	27
8.	Foto tomada usando cámara Canon y estereoscopio	28
9.	Montaje de cámara y estereoscopio	29
10.	Hoja de registro de conteos	30
11.	Dataset inicial	31
12.	Dataset preprocesado	32
13.	Diagrama de la Base de Datos	36
14.	Sustratos en laboratorio	39
15.	Dataset con la densidad de cada tipo de huevo	39
16.	Reales vs predicciones	44
17.	Error en predicciones	44
18.	Reales vs predicciones	46
19.	Error en predicciones	46
20.	Reales vs predicciones	47
21.	Error en predicciones	47
22.	Historial de entrenamiento	48
23.	Reales vs predicciones	49
24.	Error en predicciones	49
25.	Reales vs predicciones	51
26.	Error en predicciones	51
27.	Reales vs predicciones	52
28.	Error en predicciones	52
29.	Reales vs predicciones	53
30.	Error en predicciones	54
31.	Historial de entrenamiento	54

32.	Reales vs predicciones	55
33.	Error en predicciones	56
34.	Residuos vs valores predichos	57
35.	Gráfico Q-Q	57
36.	Login de usuario	59
37.	Login de usuario responsive	60
38.	Registro de usuarios en MySQL	61
39.	Registro de usuarios en firebase	61
40.	Diagrama entidad-relación de la base de datos	62

Lista de cuadros

1.	Métricas del modelo utilizando Random Forest	44
2.	Métricas del modelo utilizando SVM	45
3.	Métricas del modelo utilizando ANN	47
4.	Métricas del modelo utilizando Regresión Lineal	49
5.	Métricas del modelo utilizando Random Forest	50
6.	Métricas del modelo utilizando SVM	52
7.	Métricas del modelo utilizando ANN	53
8.	Métricas del modelo utilizando Regresión Lineal	55
9.	Métricas utilizando validación cruzada para los modelos de regresión lineal . .	56
10.	Prueba de Shapiro-Wilk	58
11.	Métricas de tiempo de respuesta a distintas peticiones e iteraciones	58

El presente informe describe un proyecto centrado en la creación de una aplicación web y un modelo de inteligencia artificial diseñados para predecir la cantidad de mosquitos que habrá en un sustrato a partir de información como la temperatura, días de secado de los huevos y generación de los mosquitos. La aplicación permite almacenar y gestionar los datos de los conteos y predicciones realizadas. El objetivo principal de este proyecto es obtener una estimación precisa del número de huevos viables de mosquito para poder preparar el material necesario en los procesos que se llevan a cabo posteriormente.

La aplicación web creada permite a los usuarios ingresar datos sobre las condiciones ambientales actuales, la cantidad de huevos en una pequeña zona del sustrato y las medidas del área del sustrato que tienen esa cantidad de huevos. Posteriormente, el modelo de IA procesa esta información y ofrece una estimación de la cantidad de mosquitos que se espera encontrar en el sustrato. Además, la aplicación permite almacenar y gestionar los datos de los conteos y predicciones realizadas, proporcionando un acceso fácil y desde cualquier lugar a través de la web para todos los usuarios.

El modelo utilizado fue el de regresión lineal con métricas muy buenas en cuanto al error, presentando un RMSE de $3.85e^{-13}$ y un R^2 de 0.98.

The present report describes a project focused on the creation of a web application and an artificial intelligence model designed to predict the number of mosquitoes that will be present in a substrate based on information such as temperature, egg drying days, and mosquito generation. The application allows for the storage and management of the data from the counts and predictions made. The main goal of this project is to obtain an accurate estimate of the number of viable mosquito eggs in order to prepare the necessary material for subsequent processes.

The created web application allows users to input data on current environmental conditions, the number of eggs in a small area of the substrate, and the measurements of the area of the substrate that contains that number of eggs. Subsequently, the AI model processes this information and provides an estimate of the number of mosquitoes expected to be found in the substrate. In addition, the application allows for the storage and management of the data from the counts and predictions made, providing easy access from anywhere via the web for all users.

The model used was linear regression with very good metrics in terms of error, presenting an RMSE of $3.85e^{-13}$ and an R^2 of 0.98.

CAPÍTULO 1

Introducción

El estudio de los mosquitos es de gran importancia debido a su papel en la salud humana y el medio ambiente. Los mosquitos son transmisores de enfermedades como el dengue, la malaria, el virus del Zika y la fiebre amarilla, que afectan a millones de personas en todo el mundo. Comprender la biología, el comportamiento y la ecología de los mosquitos es fundamental para el desarrollo de estrategias efectivas de control y prevención de enfermedades transmitidas por mosquitos. Además, los mosquitos desempeñan un papel en los ecosistemas como polinizadores y como fuente de alimento para otros organismos. El estudio de los mosquitos nos permite comprender mejor estos aspectos y tomar medidas para proteger la salud humana y mantener el equilibrio en los ecosistemas.

En los estudios de mosquitos, se establecen criaderos controlados para criar y observar la reproducción de los mosquitos en condiciones controladas. Dichos criaderos cuentan con gran cantidad de huevos de mosquitos que pasan por distintos procesos para llegar finalmente a ser mosquitos.

En el proceso de secado de los huevos, existen tres tipos: huevos viables, eclosionados y en forma de canoa. Es fundamental conocer la cantidad de cada uno ya que conociendo eso se cambian y adecuan distintos factores como la temperatura y la humedad para obtener la mayor cantidad de huevos viables posibles.

En el desarrollo de modelos predictivos para analizar hábitats de larvas de mosquito y su relación con variables climáticas, el trabajo de Sun et al. Se destaca por emplear técnicas de aprendizaje profundo y análisis de series temporales para predecir la abundancia de larvas en diferentes localidades de Estados Unidos, enfocando en cómo el cambio climático podría expandir los hábitats de mosquitos [1]. Además, otro estudio relevante, también resalta el uso de inteligencia artificial para predecir aspectos relacionados con poblaciones de mosquitos [2]. Estos trabajos previos proporcionan un fundamento sólido y una motivación clara para explorar cómo la inteligencia artificial y el aprendizaje automático pueden aplicarse de manera efectiva en la predicción y gestión de datos relacionados con poblaciones de mosquitos.

El conteo y monitoreo de huevos es fundamental para mantener control de las poblaciones y poder preparar materiales para los estudios posteriores que se hacen con los mosquitos adultos, por lo que surge la necesidad de agilizar y mejorar la eficiencia de los procesos de estudio y control de mosquitos. Contar manualmente todo el sustrato en donde se ubican los huevos de mosquitos para ver cuáles están en condiciones viables es un trabajo tedioso y propenso a errores humanos, lo que puede afectar la precisión de los datos y retrasar el progreso de la investigación.

Actualmente está creciendo el uso de inteligencia artificial en todos los campos y en la biología no es diferente. Esta se utiliza para monitoreo de plagas, optimización, conteo de insectos, predicción de especies y para muchas cosas más. La implementación de un sistema de inteligencia artificial que pueda predecir la viabilidad de los huevos de manera automatizada y precisa tiene varias ventajas. En primer lugar, ahorra tiempo y recursos al eliminar la necesidad de contar por completo cada sustrato, permitiendo que los investigadores se enfoquen en tareas más complejas y analíticas. Además, se puede ir comparando las predicciones con los resultados reales para mejorar el modelo y así reducir el error humano que hay al contar manualmente los huevos.

Finalmente, cabe señalar que los mosquitos también son importantes para la polinización, por lo que el conocimiento de sus habitantes puede ser útil para estudiar y preservar la diversidad vegetal.

4.1. Objetivos generales

- Desarrollar una aplicación que ayude a predecir la cantidad de huevos de mosquito bajo condiciones de laboratorio.
- Desarrollar un modelo que permita predecir la cantidad de huevos de mosquito y almacenar las predicciones.

4.2. Objetivos específicos

- Evaluar distintos algoritmos que permitan la predicción de la viabilidad de los huevos de mosquito y seleccionar el más adecuado.
- Determinar una manera óptima para fotografiar los huevos de mosquito.
- Implementar una base de datos que permita almacenar los conteos.
- Desarrollar un backend eficiente para gestionar las operaciones de guardado, consulta y edición de conteos.
- Implementar el inicio de sesión y manejo de usuarios de la aplicación para que tengan acceso a realizar y visualizar conteos.

Esta investigación aborda diversos aspectos fundamentales relacionados con la predicción de la cantidad de huevos de mosquito en un sustrato, el sustrato se refiere al material o medio en el que los mosquitos depositan sus huevos. En primer lugar, se analiza la biología y el comportamiento de los mosquitos, centrándose en su ciclo de vida y su capacidad para seleccionar y poner huevos en distintos sustratos. Se examina la influencia de factores ambientales y climáticos en la viabilidad de los huevos, proporcionando una comprensión sólida de la ecología reproductiva de estos insectos.

En el ámbito de la inteligencia artificial, se estudian los modelos de aprendizaje automático aplicados en la predicción de la cantidad de huevos de mosquito. Se describen los algoritmos de regresión utilizados para identificar patrones y realizar estimaciones anticipadas, lo que permite el desarrollo de un modelo predictivo efectivo.

Por último, se abordan aspectos metodológicos, como los métodos de muestreo y recopilación de datos, el procesamiento y preparación de la información para el modelo de inteligencia artificial, y las estrategias de validación y evaluación del modelo. También, se abordan aspectos del desarrollo web y de base de datos, métodos de validación y monitoreo de servidores.

5.1. Biología y comportamiento de los mosquitos

5.1.1. Ciclo de vida

El ciclo de vida de los mosquitos consta de cuatro etapas principales: huevo, larva, pupa y adulto. [3]

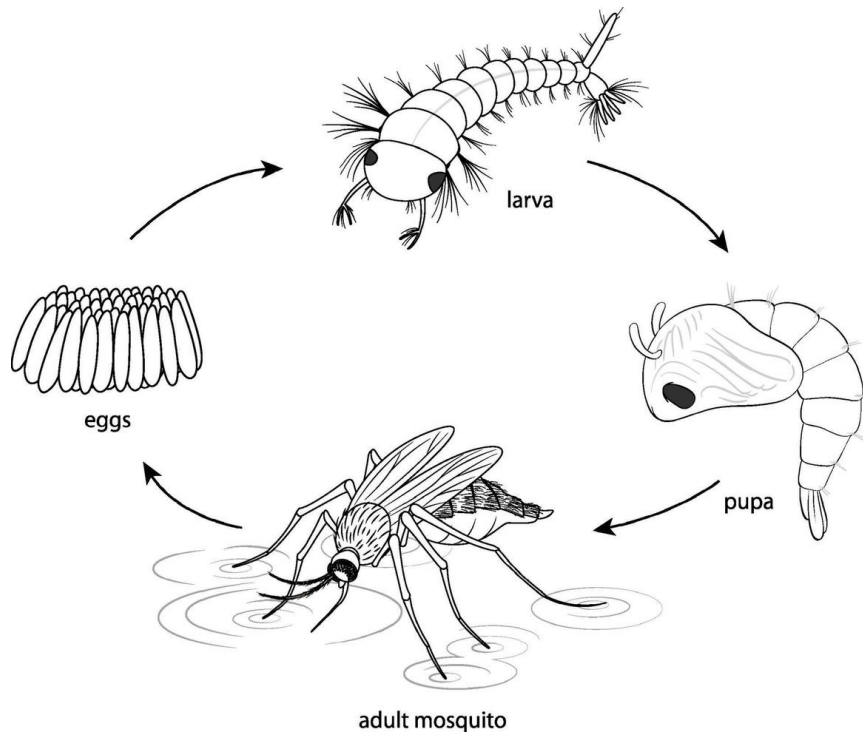


Figura 1: Ciclo de vida del mosquito

Fuente: del, (2021) [4]

1. Huevos

Los mosquitos hembras depositan sus huevos en o cerca del agua, ya que la mayoría de las especies de mosquitos requieren agua para completar su ciclo de vida. Los huevos son generalmente dispuestos en racimos o flotan individualmente en la superficie del agua. [3]

2. Larvas

De los huevos emergen las larvas, que son acuáticas y se desarrollan en el agua. Son conocidas comúnmente como gusanos. Las larvas se alimentan de materia orgánica presente en el agua y respiran a través de tubos respiratorios en la parte superior del abdomen, que les permiten obtener oxígeno de la atmósfera. [3]

3. Pupas

Después de pasar por varias etapas de crecimiento, las larvas se convierten en pupas. La etapa de pupa es inmóvil y está contenida en una carcasa protectora llamada "pupario". Durante esta etapa, el mosquito se transforma internamente en el adulto. [3]

4. Adulto

Finalmente, el mosquito adulto emerge de la pupa. Los mosquitos adultos son insectos alados que vuelan en busca de alimento y parejas para la reproducción. Las hembras necesitan una fuente de sangre para desarrollar sus huevos, mientras que los machos se alimentan principalmente de néctar y otras sustancias azucaradas. [3]

5.1.2. Puesta de huevos

La elección del sustrato para la puesta de huevos puede variar según la especie de mosquito. Algunas especies prefieren aguas estancadas, como charcos, estanques, jarras con agua o recipientes abandonados, mientras que otras pueden poner sus huevos en aguas corrientes, como arroyos y ríos. Las hembras buscan lugares adecuados para poner sus huevos, ya que las larvas dependen del agua para sobrevivir. [5]

5.1.3. Puesta de huevos en condiciones de laboratorio

1. Sustrato

En el laboratorio, los recipientes utilizados para la oviposición suelen ser bandejas, platos, tubos de ensayo llenos de agua o pedazos de papel especial. El agua puede ser agua destilada, agua de grifo envejecida o incluso agua mezclada con fermentos orgánicos para simular aguas estancadas y atraer a las hembras a poner huevos. [6]

2. Variables

Algunas especies de mosquitos requieren estímulos específicos para ovipositar. Por ejemplo, ciertos mosquitos pueden responder a estímulos visuales, como el contraste entre el recipiente y el agua, o a estímulos químicos, como los derivados de fermentos orgánicos. Además, los estímulos táctiles, como la textura del sustrato o la presencia de vegetación, y los estímulos olfativos, como los olores específicos en el agua o en las cercanías, también pueden influir en la elección del sitio de oviposición. Las condiciones ambientales, como la temperatura, la humedad y la luz, pueden también alterar la probabilidad y el éxito de la oviposición. [7]

3. Control de condiciones

La temperatura, la humedad y la luz se controlan con precisión en el laboratorio para asegurar una tasa de oviposición óptima. Por lo general, se mantienen condiciones similares a las que los mosquitos encontrarían en su ambiente natural durante la temporada de reproducción. [7]

4. Alimentación

Las hembras suelen requerir una comida de sangre antes de la oviposición. En el laboratorio, esto se logra a menudo alimentando a los mosquitos con sangre de animales (como ratones o conejos) o utilizando sistemas artificiales de alimentación con sangre. [7]

5. Manejo de huevos

Una vez que los huevos son depositados por los mosquitos, es posible recolectarlos y trasladarlos a otros recipientes si es necesario, dependiendo del objetivo del estudio. El proceso de recolección puede requerir cuidado y herramientas especializadas para asegurar que los huevos no sean dañados durante el proceso. Posteriormente, pueden ser trasladados a otros recipientes que estén preparados con las condiciones adecuadas para la incubación y el desarrollo posterior de los huevos. [7]

5.1.4. Biología y factores de reproducción

La biología y reproducción de los mosquitos están influenciadas por varios factores, incluyendo la temperatura y los cambios climáticos. Algunos aspectos importantes son:

1. Temperatura

La temperatura afecta significativamente el desarrollo y la tasa de reproducción de los mosquitos. En general, los mosquitos se desarrollan más rápidamente en climas cálidos, y la velocidad de desarrollo de los huevos, larvas y pupas puede aumentar con temperaturas más altas. Las temperaturas extremas, tanto frías como calurosas, pueden limitar la actividad y la reproducción de los mosquitos. Cuando se encuentran en la fase de huevos la temperatura regula su tasa de desarrollo. En temperaturas más cálidas, los huevos pueden eclosionar más rápidamente, mientras que en temperaturas bajas, la eclosión puede retrasarse o incluso detenerse. Sin embargo, temperaturas extremadamente altas o bajas pueden ser letales para los huevos. [8]

2. Humedad

La humedad es un factor crucial para la reproducción de los mosquitos, ya que la mayoría de las especies requieren agua para completar su ciclo de vida. Las hembras adultas necesitan depositar sus huevos en o cerca del agua para que las larvas puedan desarrollarse. Sin un hábitat acuático adecuado, los mosquitos no pueden reproducirse. Si bien la mayoría de los huevos de mosquito necesitan estar en o cerca del agua para eclosionar, la humedad relativa del ambiente también puede ser crítica. En condiciones de baja humedad, los huevos pueden desecarse y perder viabilidad. Algunas especies de mosquitos, como los del género *Aedes*, pueden desovar en lugares secos y sus huevos pueden resistir la desecación, esperando la inundación para comenzar su desarrollo. [9]

3. Generación

La duración del ciclo de vida de los mosquitos también afecta su reproducción. La velocidad de desarrollo desde el huevo hasta el adulto puede variar según la especie y las condiciones ambientales, como la temperatura y la disponibilidad de alimentos. Los huevos son la etapa del ciclo de vida que permite a los mosquitos superar condiciones adversas. Dependiendo de la generación y de las condiciones ambientales, algunos huevos pueden entrar en un estado de quiescencia (una especie de "suspensión" del desarrollo) y esperar condiciones más favorables para eclosionar. [10]

4. Estacionalidad

La presencia de mosquitos puede variar a lo largo del año, dependiendo de las estaciones y las condiciones climáticas. En climas más cálidos, los mosquitos pueden estar presentes durante todo el año, mientras que en climas más fríos, pueden ser más abundantes durante los meses más cálidos. Los huevos de algunas especies pueden ser más resistentes a las condiciones adversas de invierno, permitiendo que la población de mosquitos perdure de una temporada a otra. La estacionalidad también puede influir en la cantidad y la frecuencia de la oviposición, con algunas especies produciendo más huevos durante ciertas estaciones del año. [11]

5. Días de secado

En condiciones de laboratorio, el período durante el cual se dejan secar los huevos de mosquito, denominado "días de secado", es un factor esencial para la reproducción y eclosión de estos insectos. Esta desecación controlada simula las condiciones naturales que enfrentan algunos huevos, preparándolos para una eclosión exitosa. Sin embargo, la duración óptima puede variar según la especie; mientras que un breve período de secado puede resultar insuficiente para algunos, una desecación prolongada podría comprometer la viabilidad de otros. Además, las condiciones ambientales, como la humedad y la temperatura durante este proceso, pueden afectar la integridad de los huevos. Por ello, en el contexto experimental, es crucial determinar y optimizar el número de "días de secado" para asegurar tasas de viabilidad elevadas y consistentes. [12]

5.2. Modelos de Inteligencia Artificial para predicción

5.2.1. Máquinas de Soporte Vectorial

Las Máquinas de Soporte Vectorial (SVM por sus siglas en inglés) son un tipo de algoritmo de aprendizaje supervisado utilizado tanto para clasificación como para regresión. La idea central detrás de las SVM es encontrar el hiperplano óptimo que mejor separa las clases de datos en un espacio multidimensional. En el caso de clasificación binaria, el hiperplano separa dos clases diferentes, mientras que en la clasificación multiclase, se utilizan múltiples hiperplanos. SVM busca maximizar el margen entre las clases, lo que aumenta su capacidad de generalización a nuevos datos. Además, las SVM pueden usar funciones del kernel para mapear datos a espacios de mayor dimensión, lo que les permite manejar datos no linealmente separables. [13]

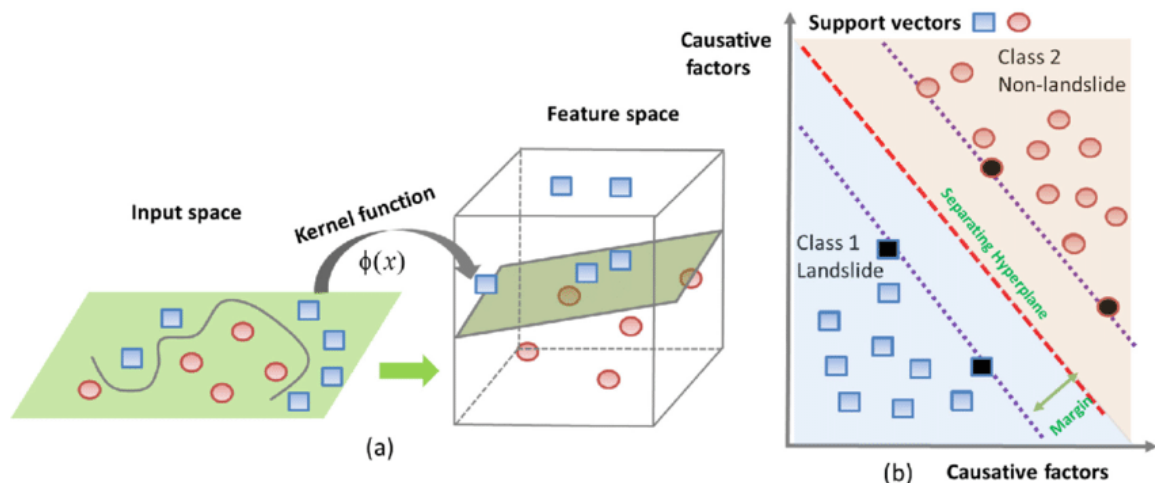


Figura 2: Funcionamiento de la SVM

Fuente: ResearchGate, (2019) [14]

5.2.2. Random Forest

Random Forest es un algoritmo de aprendizaje conjunto basado en el método de "bagging". Consiste en la combinación de múltiples árboles de decisión, donde cada árbol se entrena con una muestra aleatoria de datos de entrada y utiliza un subconjunto aleatorio de características para tomar decisiones en cada nodo. Luego, para realizar una predicción, cada árbol del bosque emite su propia predicción y se lleva a cabo un proceso de votación o promedio para obtener la predicción final. Los modelos Random Forest son robustos, fáciles de usar, y tienden a evitar el sobreajuste, lo que los convierte en una opción popular para problemas de clasificación y regresión. [15]

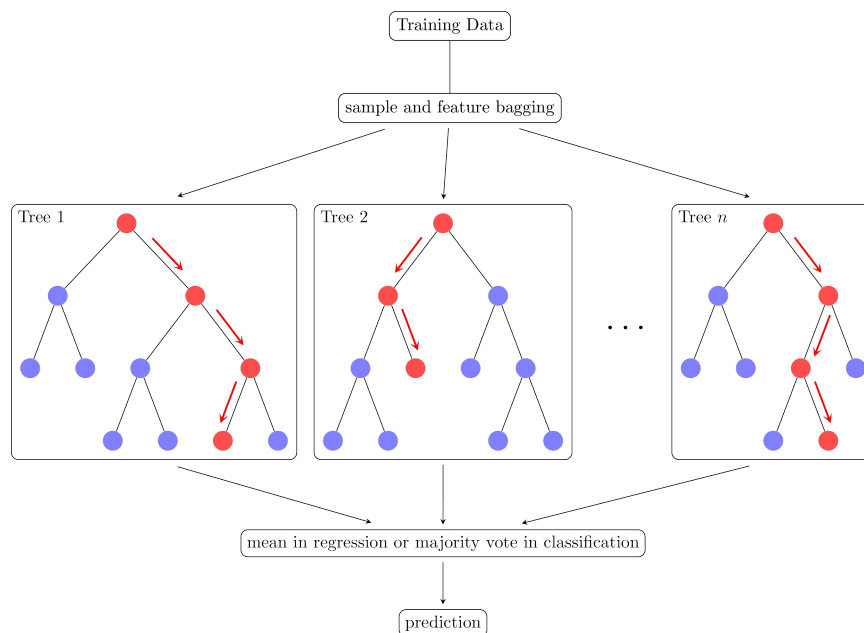


Figura 3: Funcionamiento de Random Forest

Fuente: Riebesell, (2022) [16]

5.2.3. Redes Neuronales Artificiales

Las Redes Neuronales Artificiales (ANN por sus siglas en inglés) son modelos de aprendizaje profundo inspirados en el funcionamiento del cerebro humano. Consisten en una red interconectada de unidades llamadas neuronas o nodos, organizadas en capas. Cada neurona recibe entradas ponderadas, las procesa mediante una función de activación no lineal y produce una salida. Las redes neuronales se entrenan mediante un proceso iterativo llamado descenso del gradiente para ajustar los pesos y aprender a realizar tareas específicas, como clasificación o regresión. A medida que la red tiene más capas y neuronas, se le conoce como una red neuronal profunda o "deep learning". Las ANN han demostrado un rendimiento sobresaliente en una amplia gama de tareas de predicción, especialmente cuando se cuenta con grandes conjuntos de datos. [17]

5.2.4. Regresión lineal

La regresión lineal es un método que se utiliza tanto en estadística como en aprendizaje automático para predecir un valor numérico en base a una o más variables de entrada. Este método asume que existe una relación lineal entre las variables de entrada y la variable que se quiere predecir. Se busca encontrar la "línea de mejor ajuste" que minimice la diferencia entre los valores predichos por el modelo y los valores reales en los datos de entrenamiento. El término "de mejor ajuste" se refiere a la línea que pasa lo más cerca posible de todos los puntos de datos. [18]

Una vez que se ha entrenado el modelo, se pueden hacer predicciones sobre nuevos datos utilizando la línea de mejor ajuste determinada. La regresión lineal es un modelo simple y a menudo se utiliza como punto de partida en el aprendizaje automático antes de avanzar hacia modelos más complejos. [18]

5.3. Validación y evaluación del modelo de predicción

5.3.1. Evaluación

La evaluación implica medir el rendimiento de cada modelo individual en un conjunto de datos de prueba o validación. Para ello se utilizan dos tipos de métricas.

- Clasificación: Se utilizan métricas como precisión, recall, F1-score, matriz de confusión y área bajo la curva ROC (AUC-ROC).
- Regresión: Se utilizan métricas como raíz de error cuadrático medio (RMSE), el error cuadrático medio (MSE) y el coeficiente de determinación (R^2).

5.3.2. Validación cruzada

La validación cruzada es una técnica que te permite obtener una estimación más confiable del rendimiento del modelo al entrenarlo y evaluarlo en diferentes subconjuntos del conjunto de datos. La validación cruzada puede ayudar a evitar problemas de sobreajuste y proporcionar una estimación más realista del rendimiento del modelo en datos no vistos. La validación cruzada k-fold es comúnmente utilizada, donde el conjunto de datos se divide en k partes iguales, y el modelo se entrena y evalúa k veces, utilizando diferentes combinaciones de datos de entrenamiento y prueba.

5.3.3. Análisis de residuos

El análisis de residuos implica examinar la diferencia entre los valores observados y los valores predichos por el modelo, conocidos como residuos. Este análisis es fundamental para diagnosticar la adecuación del modelo, detectar patrones no capturados e identificar posibles outliers o datos atípicos. [19]

Gráfico Q-Q

Un gráfico Q-Q (quantile-quantile) es una herramienta gráfica para evaluar si un conjunto de datos sigue una distribución específica, generalmente la normal. Compara los cuantiles de los datos con los cuantiles esperados de la distribución teórica. Si los puntos en el gráfico Q-Q se alinean aproximadamente a lo largo de una línea recta, sugiere que los datos siguen la distribución en cuestión. [20]

Prueba Shapiro-Wilk

La prueba de Shapiro-Wilk es un test estadístico utilizado para determinar si una muestra de datos proviene de una población con una distribución normal. La prueba calcula un estadístico W basado en la correlación entre los datos y los cuantiles correspondientes de la distribución normal. Valores pequeños del estadístico W indican evidencia contra la normalidad. [21]

5.4. Desarrollo de aplicación web y modelo

5.4.1. Lenguajes de desarrollo

Son las herramientas fundamentales que permiten a los desarrolladores construir y estructurar aplicaciones, definir la lógica de negocio y establecer interacciones en la web y otros entornos. Cada lenguaje tiene sus propias características, ventajas y aplicaciones ideales, siendo seleccionados en función de las necesidades específicas del proyecto. Estos lenguajes, junto con sus respectivos frameworks y bibliotecas, facilitan la creación, optimización y mantenimiento de soluciones digitales. [22]

JavaScript

Es uno de los lenguajes de programación más populares y ampliamente adoptados en el mundo. Se utiliza principalmente en el desarrollo web para agregar interactividad y dinamismo a las páginas web. Su capacidad para ejecutarse en el lado del cliente ha convertido a JavaScript en un pilar esencial para la web moderna. Con la introducción de NodeJS, JavaScript también ha encontrado su lugar en el desarrollo de servidor, permitiendo a los desarrolladores usar un lenguaje coherente en todo el stack de desarrollo. [23]

TypeScript

Es un superconjunto tipado de JavaScript desarrollado y mantenido por Microsoft. TypeScript introduce características de programación orientada a objetos y un sistema de tipos estático opcional. Estas características hacen que TypeScript sea especialmente adecuado para proyectos de desarrollo a gran escala, ya que pueden ayudar a identificar errores en

tiempo de compilación antes de que el código se ejecute. Angular, el framework que se utilizó para el frontend, adopta TypeScript como lenguaje principal debido a sus ventajas. [24]

Python

Es un lenguaje de programación de alto nivel conocido por su sintaxis limpia y legible. Su versatilidad lo ha hecho popular en diversas aplicaciones, desde desarrollo web hasta ciencia de datos y aprendizaje automático. Python cuenta con un ecosistema amplio de bibliotecas y frameworks, lo que facilita la rápida implementación de soluciones complejas. [25]

Keras

Es una interfaz de alto nivel para redes neuronales que se ejecuta sobre varios backends de procesamiento tensorial, siendo TensorFlow uno de los más populares. Keras simplifica el proceso de construcción y entrenamiento de modelos de aprendizaje automático, ofreciendo bloques modulares y reutilizables que facilitan la definición y optimización de arquitecturas complejas. [26]

TensorFlow

Es una biblioteca de código abierto desarrollada por Google para computación numérica y aprendizaje automático. TensorFlow proporciona un conjunto completo de herramientas para construir y entrenar modelos de machine learning a gran escala. Su capacidad para funcionar tanto en CPUs como en GPUs hace que el entrenamiento de modelos complejos sea más eficiente y rápido. [27]

5.5. Manejo de Usuarios

El manejo de usuarios es esencial para cualquier aplicación o plataforma que busque ofrecer experiencias personalizadas o que requiera controlar el acceso a ciertas áreas o funciones.

5.5.1. Firebase

Es una plataforma desarrollada por Google que ofrece diversos servicios para el desarrollo de aplicaciones web y móviles. Dentro de Firebase, el servicio de autenticación es ampliamente reconocido por su facilidad de integración y seguridad. Permite a los desarrolladores gestionar el proceso de registro e inicio de sesión de los usuarios sin tener que implementar desde cero sistemas de autenticación, ofreciendo múltiples métodos de autenticación como correo electrónico, autenticación con Google, Facebook, entre otros. [28]

5.6. Frontend

Es la interfaz con la que interactúa el usuario, constituyendo la parte visible y tangible de una aplicación o sitio web. El frontend se encarga de presentar la información de manera clara y atractiva, permitiendo al usuario llevar a cabo acciones e interactuar con el sistema. Utilizando lenguajes como HTML, CSS y JavaScript, el frontend traduce la lógica y los datos provenientes del backend en experiencias visuales interactivas, asegurando que sean intuitivas y amigables para el usuario. [29]

5.6.1. HTML y CSS

Son las tecnologías base para el desarrollo web. HTML (Hypertext Markup Language) es el estándar para crear estructuras de páginas web [30], mientras que CSS (Cascading Style Sheets) define el estilo y diseño visual de esos elementos. [31]

5.6.2. Angular

Es un framework de desarrollo web desarrollado y mantenido por Google. Se utiliza para construir aplicaciones web de una sola página (SPA) y ofrece herramientas para el enlace de datos bidireccional, inyección de dependencias y modularidad. Angular permite una rápida iteración de desarrollo y tiene un ecosistema amplio de librerías y herramientas. [32]

5.6.3. NodeJS

Es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Aunque es ampliamente conocido por su uso en el desarrollo backend, en el contexto del frontend, NodeJS es esencial para herramientas como npm (gestor de paquetes) que facilitan la gestión de dependencias y scripts de desarrollo.[33]

5.7. Backend

Es la columna vertebral digital de una aplicación o sitio web, encargándose de toda la lógica de negocio, procesamiento de datos y comunicación con la base de datos. Operando en el servidor, el backend recibe las solicitudes del frontend, procesa la información según lo necesario y devuelve una respuesta. Es responsable de garantizar la seguridad, eficiencia y correcto funcionamiento de la aplicación, así como de gestionar las operaciones de lectura y escritura en la base de datos. [34]

5.7.1. Express

Es un framework para NodeJS que facilita la creación de aplicaciones web y APIs de forma rápida y sencilla. Proporciona un conjunto mínimo de características para el desarrollo del servidor, permitiendo al desarrollador optar por las herramientas y middlewares que mejor se adapten a sus necesidades. [35]

5.7.2. TensorFlowJS

Para cargar modelos realizados con TensorFlow en Python existe TensorFlowJS que es una biblioteca de JavaScript para entrenamiento y despliegue de modelos de aprendizaje automático en navegadores y en Node.js. Permite ejecutar modelos previamente entrenados y realizar inferencias directamente en el navegador o en el servidor, aprovechando la aceleración por hardware. [36]

5.7.3. Postman

Postman es una plataforma colaborativa que facilita el desarrollo de APIs, proporcionando herramientas para diseñar, probar, documentar y monitorear interfaces de programación de aplicaciones. Ofrece una interfaz gráfica de usuario intuitiva que permite a los desarrolladores crear, enviar y analizar solicitudes HTTP, examinar respuestas y compartir colecciones de solicitudes. Su amplio conjunto de características, que incluye pruebas automatizadas y entornos variables, lo convierte en una herramienta esencial para el desarrollo moderno de APIs. [37]

5.7.4. Uptime Robot

Uptime Robot es un servicio de monitoreo de sitios web y servidores que proporciona información en tiempo real sobre la disponibilidad y rendimiento de las aplicaciones en línea. Permite a los usuarios configurar alertas y recibir notificaciones en caso de inactividad, ofreciendo una visión continua del tiempo de actividad y la salud del sistema. Con su interfaz fácil de usar y múltiples opciones de configuración, Uptime Robot es una herramienta valiosa para asegurar la confiabilidad y accesibilidad constante de los servicios en línea. [38]

5.7.5. Flask

Flask es un microframework para Python que se utiliza para construir aplicaciones web. Se caracteriza por su simplicidad y flexibilidad, permitiendo a los desarrolladores crear rápidamente aplicaciones web con un núcleo mínimo pero extensible. Flask proporciona herramientas y librerías esenciales para el enrutamiento de URL, gestión de solicitudes y respuestas, y plantillas HTML, entre otras funcionalidades. A pesar de su naturaleza ligera, Flask es altamente escalable y se adapta tanto a proyectos pequeños como a aplicaciones web complejas, lo que lo convierte en una opción popular en el desarrollo web moderno. [39]

5.8. Base de Datos

Es el núcleo estructurado donde se almacenan, organizan y recuperan los datos de una aplicación o sistema. La elección de un sistema de gestión de bases de datos es crucial, ya que determina cómo se guardarán los datos, la velocidad y eficiencia de las consultas, y las capacidades de escalabilidad y seguridad. Un buen sistema de base de datos garantiza la integridad, disponibilidad y consistencia de la información, sirviendo de pilar para cualquier solución tecnológica. [40]

5.8.1. MySQL

Es uno de los sistemas de gestión de bases de datos relacionales más populares y confiables en el mundo. Open-source y altamente customizable, MySQL proporciona características para garantizar la integridad y confiabilidad de los datos, así como herramientas para optimizar el rendimiento de las consultas y administración de la base de datos. [41]

5.9. Hosting Web

Es el servicio que proporciona el espacio necesario en servidores de internet para que una aplicación o sitio web sea accesible a los usuarios. Más allá de simplemente alojar archivos, un buen proveedor de hosting garantiza la disponibilidad, velocidad y seguridad del sitio web o aplicación, permitiendo una experiencia de usuario óptima. La elección del hosting puede influir en aspectos como el tiempo de carga, la escalabilidad y la capacidad de respuesta ante picos de tráfico. [42]

5.9.1. Vercel

Es una plataforma de hosting orientada al despliegue continuo y la automatización. Optimizada para aplicaciones front-end y back-end construidas con JavaScript y otros lenguajes relacionados, Vercel proporciona características como despliegues inmediatos, HTTPS por defecto y escalabilidad automática. [43]

5.10. Responsible AI

En el desarrollo de este proyecto, se prestó especial atención a los principios de la Inteligencia Artificial Responsable (Responsible AI). Esto implicó garantizar que el modelo de IA desarrollado se basara en datos precisos y representativos, y que se diseñara y evaluara con un enfoque en la equidad, la transparencia y la rendición de cuentas. En esta sección se aborda el origen de los datos, los desafíos del modelo y sus limitaciones.

5.10.1. Origen de los datos

La recolección de datos se llevó a cabo de forma manual, siguiendo estrictos protocolos de seguridad y bioseguridad, en colaboración con el Centro de Estudios en Salud, de la Unidad de Entomología Médica y Malaria, del Instituto de Investigaciones de la Universidad del Valle de Guatemala. Durante todo el proceso, se aseguró la integridad de las muestras biológicas y se contó con la aprobación y consentimiento de las autoridades pertinentes de la universidad.

Se estudió el proceso de puesta de huevos y su posterior conteo, en cada conteo se registraban los datos en una hoja escrita, por lo que se procedió a transcribir todos estos datos en excel para poder crear el dataset para el modelo.

5.10.2. Limitaciones

El modelo de regresión lineal utilizado en este proyecto presenta ciertas limitaciones inherentes a su naturaleza y a las condiciones de los datos. Primero, cabe destacar que el modelo no incluye variables como la humedad o la luz. Estas variables ambientales pueden tener un impacto significativo en el desarrollo y la viabilidad de los huevos, por lo que su exclusión puede limitar la precisión del modelo en ciertos contextos.

Otra limitación es la dependencia del modelo en los datos históricos para realizar predicciones. Esto implica que cualquier cambio en las condiciones o en los patrones de puesta de huevos que no estén reflejados en los datos históricos podrían afectar la precisión de las predicciones.

Finalmente, aunque se realizaron esfuerzos para garantizar la integridad y representatividad de los datos, siempre existe el riesgo de sesgos inadvertidos o errores en los datos que podrían influir en los resultados del modelo. Estos sesgos podrían provenir de la manera en que se recopilaban o procesaban los datos, o de factores no observados que influyen en la puesta de huevos.

6.1. Gestión de proyecto

En esta sección, se detalla la metodología empleada para la coordinación y ejecución eficiente del proyecto. Se describe una adaptación personalizada de Scrum, una metodología ágil, para manejar las diversas fases del desarrollo. Esta adaptación incluye la implementación de Sprints y reuniones periódicas con el cliente. Este enfoque permitió una gestión flexible y ágil, adaptándose a cambios imprevistos y garantizando un progreso constante hacia los objetivos del proyecto.

6.1.1. Adaptación de la metodología Scrum

Para llevar a cabo este proyecto se utilizó una adaptación de Scrum, para manejar las diversas fases del desarrollo. Esta adaptación incluyó la implementación de Sprints y reuniones periódicas con el cliente. Se utilizó Azure DevOps para implementar esta metodología, donde se llevó el backlog con todas las tareas y se trabajó por sprints. Aunque no se tuvieron reuniones diarias (dailys) en el equipo de desarrollo, se mantuvo una comunicación constante.

Tablero

Para la gestión y visualización del progreso del proyecto, se utilizó Azure DevOps, que ofreció una plataforma integral para implementar la metodología ágil. El tablero en Azure DevOps proporcionó una visión clara de todas las tareas, permitiendo un seguimiento detallado del avance de cada elemento y del proyecto en su conjunto.

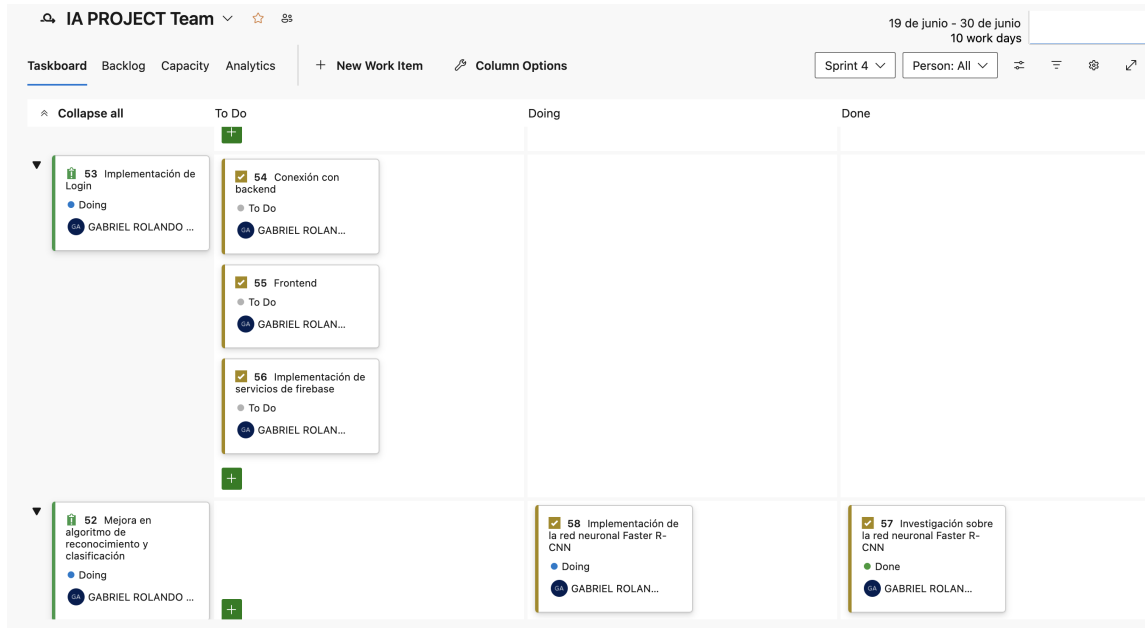


Figura 4: Tablero en Azure

Sprints

Los sprints fueron implementados como ciclos de trabajo intensivos y enfocados, con una duración estándar de dos semanas. Sin embargo, se adoptó un enfoque flexible para la duración de los Sprints, permitiendo que algunos se extendieran más allá de las dos semanas según las necesidades y requerimientos específicos del proyecto. Al final de cada Sprint, o en algunos casos después de dos Sprints, se organizaban reuniones para mostrar los avances al cliente. Estas reuniones eran cruciales para obtener retroalimentación, ajustar planes y garantizar que el proyecto se mantuviera en el camino correcto hacia el cumplimiento de los objetivos establecidos.

Reuniones

Se llevaron a cabo reuniones periódicas con el cliente para mostrarle avances. Estas reuniones se llevaban a cabo cada dos semanas o cada mes dependiendo del requerimiento específico del proyecto.

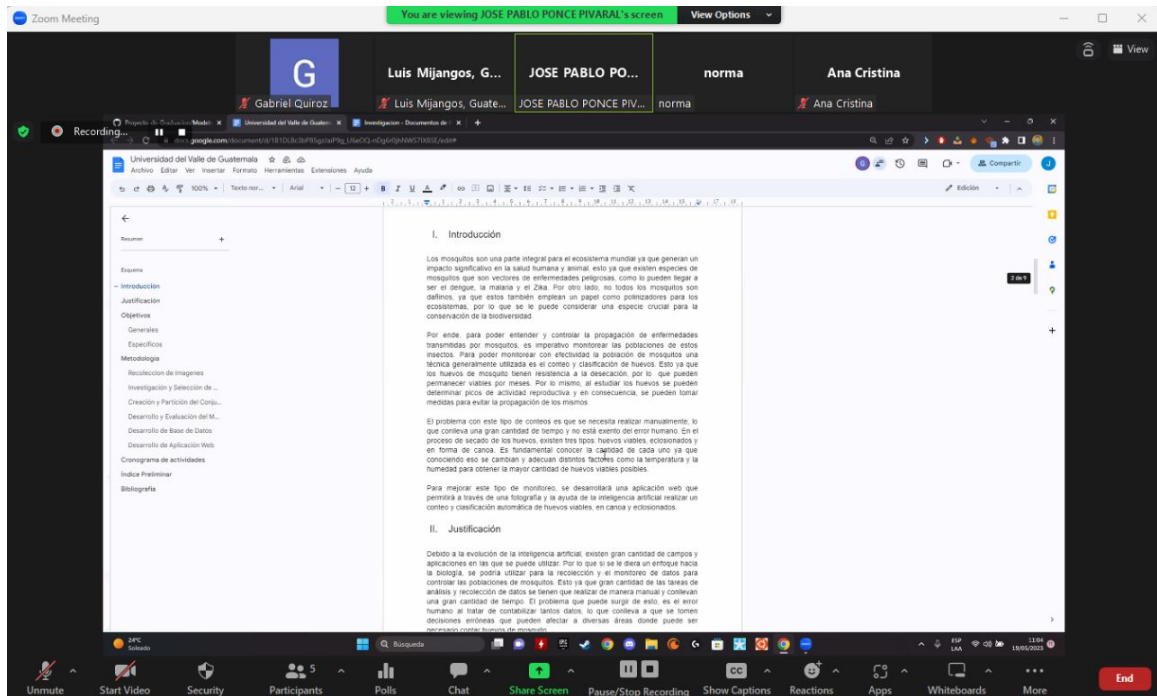


Figura 5: Reunión periódica con el cliente

6.2. Proceso de conteo de huevos

El primer paso crucial en este proyecto fue entender el proceso actual de conteo de huevos y las condiciones del laboratorio. Esto implicó analizar las condiciones ambientales, como la temperatura y la humedad, esenciales para la crianza de los huevos, y las prácticas de higiene para prevenir contaminaciones. Se revisaron detenidamente los procedimientos desde la recolección de los huevos, su traslado para secado, hasta su inspección y conteo final utilizando un estereoscopio. Esta revisión inicial proporcionó una base sólida para identificar desafíos y posibles mejoras en el proceso de conteo de huevos.

1. Preparación de la sala de laboratorio

Se asegura que la sala cuente con una temperatura y humedad adecuadas para la crianza de los huevos de mosquitos.

2. Equipamiento adecuado

Antes de ingresar a la sala, es necesario utilizar una bata de laboratorio, guantes y zapatos especiales para mantener la higiene y prevenir la contaminación.

3. Recolección de los huevos

Los huevos de mosquito se encuentran depositados en sustratos especiales. Estos sustratos se recolectan cuidadosamente y se trasladan a otro estante para el proceso de secado, el cual dura aproximadamente 1 o 2 días.

4. Inspección y conteo

Una vez que los sustratos se han secado, se trasladan a otro salón donde se lleva a cabo la inspección y el conteo de los huevos. Se utiliza un estereoscopio para observar los huevos con mayor detalle y la persona encargada del conteo va moviendo el sustrato mientras registra la cantidad de huevos y su clasificación.

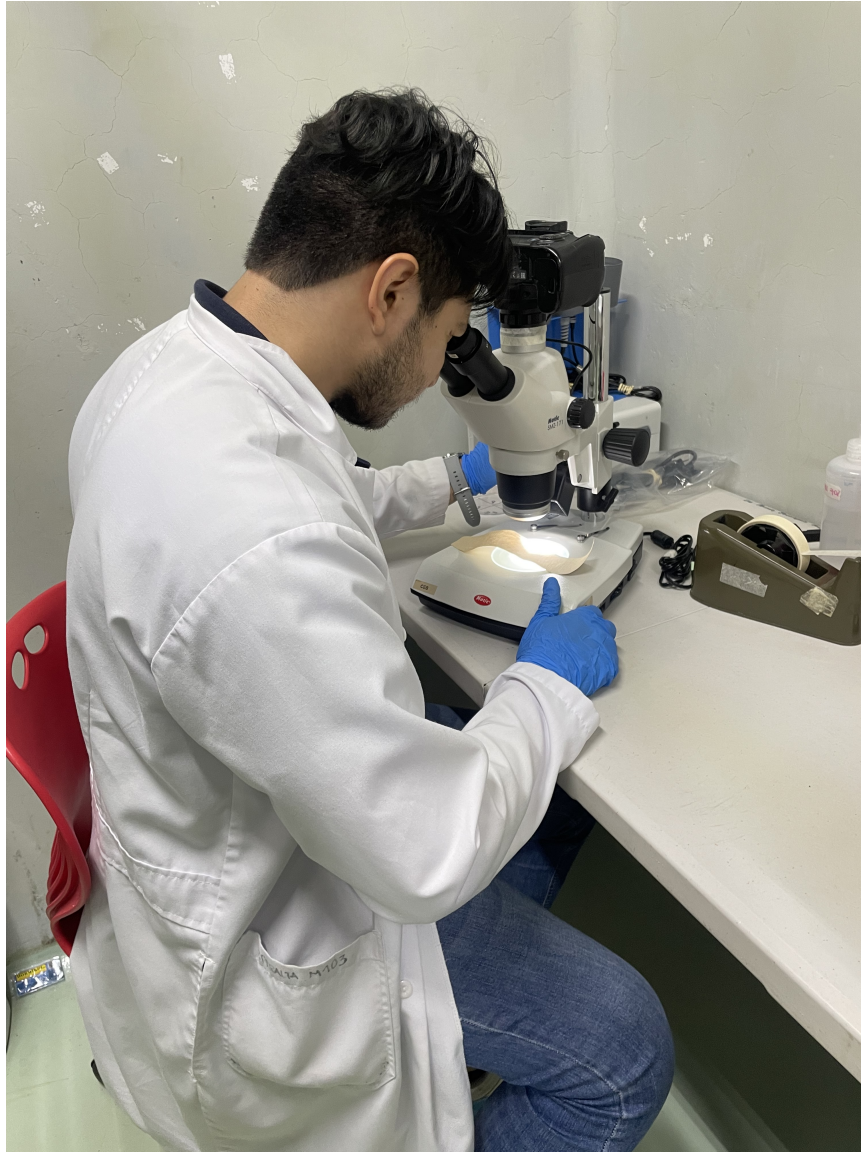


Figura 6: Inspección y conteo

6.3. Toma de fotografías

En el desarrollo de este proyecto, la toma de fotografías de los huevos de mosquito representa una etapa importante, y se lleva a cabo en colaboración con un colega investigador encargado del proceso de captura y clasificación de imágenes. La calidad y precisión de estas fotografías son fundamentales, ya que constituyen la base sobre la cual se desarrolla un

modelo de clasificación de imágenes que identifica y categoriza los huevos. Aunque este proyecto no emplea directamente estas imágenes, la clasificación resultante de los huevos es esencial para la creación del dataset que alimenta este modelo.

6.3.1. Métodos

1. Teléfono celular y estereoscopio

El primer enfoque explorado implicó el uso de la cámara de un teléfono celular iPhone 12 mini, capturando imágenes a través del lente de un estereoscopio con un aumento de 2x. Aunque este método permitió obtener fotografías legibles donde se podían distinguir los huevos con cierta claridad, presentaba desafíos significativos. La estabilidad del teléfono era difícil de mantener, y el ángulo de captura resultaba bastante limitado, lo que complicaba la obtención de imágenes de alta calidad. Por lo tanto, se decidió descartar este método debido a estas limitaciones.



Figura 7: Foto tomada usando iPhone 12 Mini y estereoscopio

2. Lente con aumento para celular

En un intento por simplificar el proceso y evitar el uso del estereoscopio, se adquirió un lente con aumento para el celular iPhone 12 mini. Sin embargo, los resultados fueron insatisfactorios, ya que las imágenes resultaron ser borrosas y no proporcionaban la claridad necesaria para visualizar los huevos de manera adecuada. Este método no cumplió

con las expectativas de resolución y nitidez requeridas, por lo que se optó por explorar alternativas.

3. Cámara profesional

Se exploró la utilización de una cámara profesional de la marca Canon junto con un lente especial de gran aumento. Aunque este enfoque permitió visualizar algunos huevos, la legibilidad era insuficiente y la identificación del tipo de huevo resultaba problemática. La calidad de las imágenes no era la adecuada para los propósitos del proyecto, lo que llevó a la búsqueda de una solución más efectiva.

4. Cámara profesional y estereoscopio

Finalmente, se implementó una combinación de una cámara profesional de la marca Canon y un estereoscopio. La cámara se montó en el orificio designado para cámaras en el estereoscopio, lo que permitió una integración efectiva entre ambos equipos. Este montaje resultó en la obtención de imágenes de alta calidad, donde la visualización y diferenciación de cada tipo de huevo se lograba de manera correcta y clara. La estabilidad y la resolución alcanzada con este método demostraron ser adecuadas para las necesidades del proyecto, estableciéndose así como el método preferido para la captura de imágenes en esta investigación.



Figura 8: Foto tomada usando cámara Canon y estereoscopio



Figura 9: Montaje de cámara y estereoscopio

6.4. Creación del Dataset

Es necesario obtener las variables que afectan a la viabilidad de los huevos, como lo es la temperatura, humedad, tiempo de secado y generación. Con esa información y la clasificación obtenida con las imágenes en cuanto a si son huevos viables o no, es posible crear el dataset. Con la colaboración del Centro de Estudios en Salud, de la Unidad de Entomología Médica y Malaria, del Instituto de Investigaciones de la Universidad del Valle de Guatemala, se obtuvieron datos de distintos conteos que se han realizado en el pasado. Estos datos contienen información sobre fechas, población a la que pertenecen, generación, cantidad de huevos viables (HV), eclosionados (HE) y en canoa (HEC). Dicha información se encontraba registrada en papel, por lo que se procedió a transcribir todos los datos a una

	Codigo Papel	Fecha Colocacion	Fecha Retiro	HV	HE	HEC	Total	F
0	A	2023-05-26	2023-05-29	839.0	20.0	436.0	1295	3
1	B	2023-05-26	2023-05-29	281.0	0.0	144.0	569	3
2	C	2023-05-26	2023-05-29	1045.0	333.0	92.0	1470	3
3	D	2023-05-26	2023-05-29	1833.0	242.0	230.0	2305	3
4	E	2023-05-26	2023-05-29	1238.0	239.0	98.0	1575	3

Figura 11: Dataset inicial

6.5. Preprocesamiento del Dataset

Una vez creado el dataset, el siguiente paso fue llevar a cabo un meticuloso preprocesamiento de los datos, asegurando así su calidad y facilitando análisis subsecuentes. Se emplearon las librerías pandas, numpy, seaborn, matplotlib.pyplot, ProfileReport y meteostat en un entorno Python para este propósito. Las etapas del preprocesamiento se detallan a continuación:

1. Verificación y limpieza de datos

Se realizó una revisión exhaustiva en busca de valores nulos, duplicados o inconsistentes. Posteriormente, se procedió a su corrección o eliminación según fuese pertinente. Específicamente, se detectaron datos vacíos en las columnas Fecha de Colocación y Fecha Retiro. Dada la naturaleza temporal de estos datos, se optó por el método de eliminar esas filas para resolver dichas inconsistencias.

2. Análisis exploratorio y correlación entre variables

Se llevó a cabo un análisis exploratorio inicial, enfocándose en la correlación entre las variables para identificar aquellas de mayor relevancia. Se determinó que variables como el total de huevos y la generación tenían una relación significativa con la viabilidad de los mismos.

3. Incorporación de datos externos

Con la ayuda de la librería meteostat, y aprovechando las fechas proporcionadas en el dataset, se extrajeron datos de temperatura correspondientes a cada conteo. Dado que la temperatura es un factor determinante en la viabilidad de los huevos de mosquito, se añadió al dataset una columna representando esta variable. Adicionalmente, se introdujo una columna indicando los días de secado, calculados a partir de las fechas de colocación y retiro de cada conteo. Posteriormente, estas fechas se eliminaron del dataset al considerarse no esenciales para el modelo.

4. Muestras del total por tipo de huevo

Para que el modelo pueda recibir datos adecuados en sus entradas, es necesario proporcionar una muestra representativa del total de cada tipo de huevo. Se estimó que un 15 % del sustrato total sería suficiente para estas muestras. En consecuencia, se añadieron columnas para representar muestras de huevos viables, eclosionados y en canoa, cada una con el 15 % del total correspondiente.

	HV	HE	HEC	Total	F	Temperatura	Días	Sample_HV	Sample_HE	Sample_HEC
0	839.0	20.0	436.0	1295	3	21.7	3	126	3	65
1	281.0	0.0	144.0	569	3	21.7	3	42	0	22
2	1045.0	333.0	92.0	1470	3	21.7	3	157	50	14
3	1833.0	242.0	230.0	2305	3	21.7	3	275	36	34
4	1238.0	239.0	98.0	1575	3	21.7	3	186	36	15

Figura 12: Dataset preprocesado

6.6. Desarrollo y evaluación de los modelos

Una vez preprocesado el dataset, se procedió a trabajar con tres algoritmos seleccionados para realizar pruebas. El dataset se compuso de las siguientes columnas o variables: Código Papel, Días, HV, HE, HEC, Total, F, SampleHV, SampleHE, SampleHEC y Temperatura. A continuación, se describen los algoritmos utilizados y su proceso de implementación:

6.6.1. Máquina de soporte vectorial

Una vez preprocesados los datos, se comenzó con la implementación del modelo de Máquina de Soporte Vectorial (SVM).

1. Definición de entradas y salidas

Las variables de entrada seleccionadas fueron SampleHV, SampleHE, SampleHEC, F, Temperatura y Días. Las variables de salida, por otro lado, fueron HV, HE y HEC.

2. División de datos

Se dividió el conjunto de datos en subconjuntos de entrenamiento y prueba, utilizando un 20% de los datos para pruebas.

3. Escalado de características

Dado que SVM es sensible a la escala de las características, se aplicó una estandarización utilizando StandardScaler de scikit-learn. Las características de entrenamiento se transformaron con el método fitTransform, mientras que las características de prueba se transformaron utilizando el método transform.

4. Entrenamiento y evaluación del modelo SVM

Para cada una de las variables objetivo (HV, HE y HEC), se entrenó un modelo SVM individual. Una vez entrenados los modelos SVM, se evaluó su rendimiento en el conjunto de prueba. Se calcularon métricas de error como la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación.

5. Generación de gráficas

Se generaron gráficos de dispersión para visualizar la relación entre los valores reales y los valores predichos por el modelo SVM para cada variable objetivo y también se trazaron

histogramas para visualizar la distribución de los errores de predicción del modelo SVM para cada variable objetivo.

6.6.2. Random Forest

Posteriormente se procedió con la implementación del modelo Random Forest.

1. Definición de entradas y salidas

Las variables de entrada seleccionadas fueron SampleHV, SampleHE, SampleHEC, F, Temperatura y Días. Las variables de salida, por otro lado, fueron HV, HE y HEC.

2. División de datos

Se dividió el conjunto de datos en subconjuntos de entrenamiento y prueba, utilizando un 20 % de los datos para pruebas.

3. Escalado de características

Se aplicó un escalado a las características usando el StandardScaler de scikit-learn, con el fin de estandarizarlas. Las características de entrenamiento se transformaron con el método fitTransform, mientras que las características de prueba se transformaron utilizando el método transform.

4. Entrenamiento y evaluación del modelo Random Forest

Para cada una de las variables objetivo (HV, HE y HEC), se entrenó un modelo de Bosques Aleatorios individual. Una vez que los modelos de Bosques Aleatorios fueron entrenados, se procedió a evaluar su desempeño en el conjunto de prueba. Se calcularon las mismas métricas de error que se emplearon con SVM: Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación (R^2).

5. Generación de gráficas

Se generaron gráficos de dispersión para visualizar la relación entre los valores reales y los valores predichos por el modelo Random Forest para cada variable objetivo y también se trazaron histogramas para visualizar la distribución de los errores de predicción del modelo Random Forest para cada variable objetivo.

6.6.3. Redes Neuronales Artificiales

Posteriormente se procedió con la implementación del modelo ANN.

1. Definición de entradas y salidas

Las variables de entrada seleccionadas fueron SampleHV, SampleHE, SampleHEC, F, Temperatura y Días. Las variables de salida, por otro lado, fueron HV, HE y HEC.

2. División de datos

Se dividió el conjunto de datos en subconjuntos de entrenamiento y prueba, utilizando un 20 % de los datos para pruebas.

3. Escalado de características

Se realizó la estandarización tanto de las entradas como de las salidas utilizando StandardScaler de scikit-learn.

4. Configuración del modelo

Se configuró una arquitectura secuencial utilizando Keras. La red neuronal consta de una capa de entrada con 64 neuronas y función de activación ReLU. Luego, se añadió otra capa oculta con 64 neuronas y función de activación ReLU. Finalmente, se configuró una capa de salida con 3 neuronas (correspondientes a las tres variables objetivo) y función de activación ReLU.

5. Entrenamiento y evaluación del modelo ANN

Para compilar el modelo se utilizó el optimizador adam. Luego, se estableció la función de pérdida personalizada que se definió previamente y se seleccionaron métricas adicionales para monitorizar durante el entrenamiento, que incluyen el RMSE y R^2 . Se entrenó el modelo utilizando el conjunto de entrenamiento. Se especificaron 50 épocas para el entrenamiento y también se proporcionó un conjunto de validación para monitorizar el desempeño del modelo en datos no vistos durante el entrenamiento. Para evaluar y poder calcular las métricas se desestandarizaron las predicciones y las etiquetas verdaderas a su escala original.

6. Generación de gráficas

Se generaron gráficos de dispersión para visualizar la relación entre los valores reales y los valores predichos por el modelo ANN para cada variable objetivo y también se trazaron histogramas para visualizar la distribución de los errores de predicción del modelo para cada variable objetivo. De igual forma se trazó una gráfica que muestra la evolución de la función de pérdida durante el entrenamiento para los conjuntos de entrenamiento y validación a lo largo de las épocas. Esta gráfica es esencial para entender cómo el modelo se desempeñó a lo largo del tiempo y si hubo algún signo de sobreajuste.

6.6.4. Regresión Lineal

Posteriormente se procedió con la implementación del modelo de Regresión Lineal.

1. Definición de entradas y salidas

Las variables de entrada seleccionadas fueron SampleHV, SampleHE, SampleHEC, F, Temperatura y Días. Las variables de salida, por otro lado, fueron HV, HE y HEC.

2. División de datos

Se dividió el conjunto de datos en subconjuntos de entrenamiento y prueba, utilizando un 20% de los datos para pruebas.

3. Escalado de características

Se estandarizaron tanto las entradas como las salidas utilizando el StandardScaler de

scikit-learn. Es importante notar que, en este caso, tanto las características como las salidas se han escalado, lo que es un enfoque común al trabajar con algunas técnicas de regresión

4. Entrenamiento y evaluación del modelo de Regresión Lineal

Para las variables objetivo (HV, HE y HEC), se entrenó un modelo de Regresión Lineal. Dado que tanto las características como las salidas se habían escalado previamente, fue necesario desescalar las predicciones y los valores actuales a su escala original. Se calcularon las métricas de error, la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación R^2 , para las predicciones de cada variable objetivo.

5. Generación de gráficas

Se generaron gráficos de dispersión para visualizar la relación entre los valores reales y los valores predichos por el modelo de Regresión Lineal para cada variable objetivo y también se trazaron histogramas para visualizar la distribución de los errores de predicción del modelo para cada variable objetivo.

6.7. Implementación de Base de Datos

Se estableció una base de datos utilizando MySQL con el objetivo principal de almacenar y gestionar los datos de los conteos realizados tanto por el modelo de clasificación como por el modelo de predicción. Dado que cada conteo está asociado a datos generales específicos, y un solo conjunto de datos generales puede estar relacionado con múltiples conteos, se estructuró la base de datos en tres tablas principales para albergar esta información.

6.7.1. Tabla de usuarios

Esta tabla se diseñó para registrar a los usuarios. Almacena información crucial como el correo electrónico, el nombre y el ID proporcionado por Firebase. Esta estructura facilita la identificación y gestión de los usuarios que interactúan con el sistema.

6.7.2. Tabla de datos generales

En esta tabla se conserva información relacionada con la especie, el código del responsable, la generación, el código de crianza, la especie en sí y el total de huevos viables. Sirve como una referencia principal para los detalles generales asociados a cada conteo.

6.7.3. Tabla de conteos

Esta tabla está vinculada con la tabla de datos generales. Por ello, tiene un campo designado para indicar el ID del dato general al cual está asociado cada conteo. Adicionalmente,

esta tabla alberga campos para los huevos viables, huevos eclosionados, huevos en canoa, fechas de colocación y retiro del sustrato, total general de huevos y un campo de ID de usuario. Este último permite identificar al usuario logueado que efectuó el conteo.

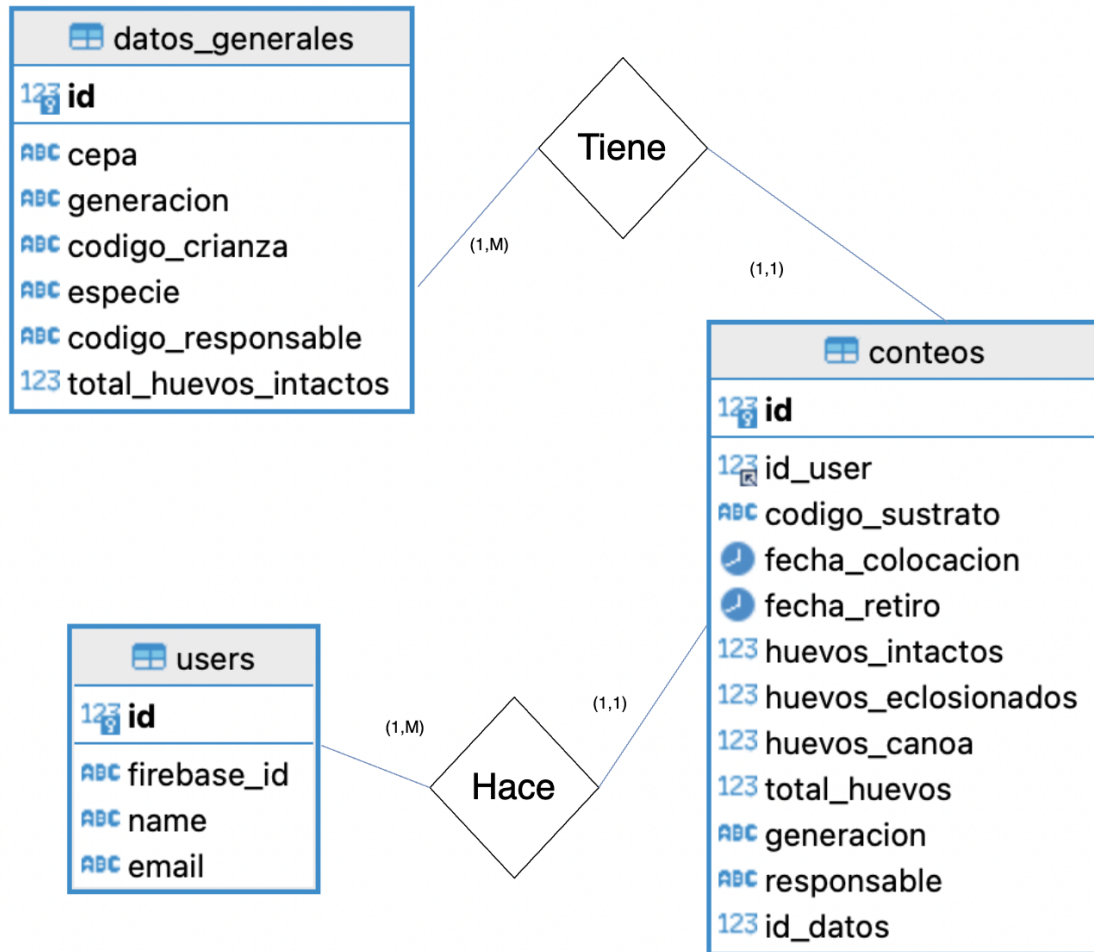


Figura 13: Diagrama de la Base de Datos

6.8. Desarrollo del backend

Posteriormente a tener lista la base de datos, se procedió a realizar el backend, para ello se utilizó el framework Express. La estructura principal del backend se estableció en el archivo `index.js`, donde se configuró el servidor, se establecieron las rutas y se gestionaron las solicitudes entrantes.

6.8.1. Gestión de usuarios

Para administrar las operaciones relacionadas con los usuarios, se crearon controladores específicos en el archivo controlador de usuario. Estos controladores gestionan tareas como

recuperar un usuario por correo electrónico y crear nuevos usuarios. La lógica detrás de estas operaciones se encuentra en el archivo de servicio de usuario. Aquí, se implementaron funciones para interactuar directamente con la base de datos y realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para los usuarios.

6.8.2. Gestión de conteos

De manera similar, se crearon controladores específicos para manejar conteos en el archivo controlador de conteo. Estos controladores gestionan una variedad de tareas, desde obtener un conteo específico por ID hasta entrenar y hacer predicciones con un modelo de aprendizaje automático. La lógica de servicio detrás de estas operaciones se implementó en el servicio de conteo. Este archivo contiene funciones que interactúan con la base de datos para gestionar conteos y también funciones que trabajan con TensorFlowJs para operaciones con el modelo de predicción.

6.8.3. Integración con el modelo

El modelo se desplegó como un microservicio utilizando Flask, el framework de Python. A través del servicio principal, desarrollado con Express, se consume este microservicio para permitir que el usuario interactúe con el modelo y obtener las predicciones correspondientes.

6.8.4. Base de Datos y seguridad

Se utilizó el paquete mysql para interactuar con la base de datos MySQL creada anteriormente. La conexión se configuró en un archivo de configuración, y las operaciones de base de datos se manejaron principalmente en los archivos de servicio. Además, se implementaron medidas para garantizar la seguridad de la base de datos y evitar exposiciones no deseadas.

6.8.5. Monitoreo y pruebas de eficiencia

El servidor desarrollado está siendo monitoreado constantemente, las 24 horas del día, con revisiones programadas cada 5 minutos. Este monitoreo ininterrumpido se realiza utilizando el servicio de Uptime Robot, una herramienta que nos permite asegurar la máxima disponibilidad y detectar rápidamente cualquier tiempo de inactividad o anomalía en el funcionamiento del servidor.

Para evaluar la eficiencia y rapidez de respuesta del servidor, se llevaron a cabo diversas pruebas de velocidad. Estas pruebas se realizaron utilizando Postman, una plataforma ampliamente reconocida por su capacidad para enviar solicitudes HTTP y analizar las respuestas de servidores. Con Postman utilizando la función Runner, se simulaban diferentes escenarios de uso, desde solicitudes simples hasta procesos más complejos que involucran múltiples interacciones con la base de datos.

6.9. Desarrollo del registro de usuarios

Para el registro de usuarios, se optó por utilizar el servicio de autenticación de Firebase debido a su robustez y versatilidad. Esta decisión permitió integrar eficientemente la autenticación con el correo institucional de la Universidad del Valle de Guatemala.

El proceso de autenticación se complementó con una interfaz desarrollada en Angular. La elección de Angular no solo proporcionó una estructura sólida y escalable para el desarrollo frontend, sino que también permitió crear una experiencia de usuario fluida y atractiva. Se puso especial énfasis en diseñar una interfaz limpia e intuitiva, facilitando a los usuarios el proceso de registro y acceso al sistema. Esta interfaz no solo mejora la experiencia del usuario, sino que también refuerza la confianza y seguridad, al ofrecer un entorno familiar y coherente con la identidad de la Universidad del Valle de Guatemala.

6.10. Cálculo de la densidad poblacional de huevos en un sustrato

Con el propósito de incrementar la precisión en las predicciones, se optó por calcular la densidad poblacional de huevos por centímetro cuadrado de sustrato (Huevos/cm²). Esta medida ofrece una métrica más detallada para anticipar la viabilidad de los huevos de mosquito en lugar de simplemente usar un porcentaje. Al examinar diversos sustratos en el laboratorio, se determinó el área de cada uno de ellos.

$$A = longitud \times ancho \tag{1}$$

A partir de estos datos, se calculó la mediana del área de todos los sustratos analizados. Esta mediana se estableció como el área estándar del sustrato y se incorporó a cada registro del conjunto de datos. Adicionalmente, se añadieron tres columnas al dataset que indican la cantidad de cada tipo de huevo por centímetro cuadrado (Huevos/cm²), proporcionando así una representación de la densidad poblacional de huevos.

$$D = \frac{huevos}{A} \tag{2}$$



Figura 14: Sustratos en laboratorio

	HV	HE	HEC	Total	F	Temperatura	Dias	Sample_HV	Sample_HE	Sample_HEC	Area	Density_HV	Density_HE	Density_HEC
0	839.0	20.0	436.0	1295	3	21.7	3	126	3	65	62.64	13.393997	0.319285	6.960409
1	281.0	0.0	144.0	569	3	21.7	3	42	0	22	62.64	4.485951	0.000000	2.298851
2	1045.0	333.0	92.0	1470	3	21.7	3	157	50	14	62.64	16.682631	5.316092	1.468710
3	1833.0	242.0	230.0	2305	3	21.7	3	275	36	34	62.64	29.262452	3.863346	3.671775
4	1238.0	239.0	98.0	1575	3	21.7	3	186	36	15	62.64	19.763729	3.815453	1.564496

Figura 15: Dataset con la densidad de cada tipo de huevo

6.10.1. Densidad poblacional de cada tipo de huevo como variable de entrada en los modelos

Después de haber utilizado las muestras de los huevos como variables de entrada en los modelos, se obtuvo la densidad poblacional de cada tipo de huevo en el sustrato como una alternativa para obtener predicciones más precisas. Por lo tanto, se realizó el mismo proceso de modelado, pero reemplazando las variables de entrada originales por la densidad poblacional de los sustratos. Todos los demás aspectos del proceso, incluido el preprocesamiento de datos, la división de datos, la configuración, el entrenamiento y la evaluación de los modelos, se mantuvieron constantes.

6.11. Validación cruzada para los modelos destacados

Tras identificar a la Regresión Lineal como el modelo óptimo utilizando tanto las muestras de los huevos como la densidad poblacional de cada tipo de huevo con RMSE y R^2 superiores a los otros modelos, se buscó realizar una última comprobación mediante la técnica de validación cruzada. Esta técnica, ampliamente reconocida por su capacidad para evaluar el desempeño y la robustez de un modelo, fue aplicada a ambos modelos para determinar su eficacia y confiabilidad.

Se optó por una validación cruzada de 22 folds, adecuándose al tamaño del conjunto de datos, que consta de 1078 muestras. Esta configuración garantiza que cada fold tenga exactamente 49 datos, permitiendo una distribución uniforme y equitativa de los datos en cada iteración. Durante este proceso, los modelos fueron entrenados y evaluados repetidamente en distintos subconjuntos del dataset.

Las métricas de evaluación seleccionadas para esta comparativa fueron la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación R^2 . Estas métricas proporcionan una visión integral del rendimiento del modelo, desde la magnitud del error hasta la proporción de la variabilidad de la respuesta que es explicada por el modelo.

Mediante la validación cruzada, se buscó no solo confirmar la superioridad de un modelo sobre el otro, sino también asegurar que el modelo seleccionado posea una capacidad de generalización consistente y fiable en diferentes muestras del conjunto de datos.

6.12. Análisis de residuos

Tras confirmar la eficacia y robustez del modelo de regresión lineal con enfoque de densidad poblacional mediante la técnica de validación cruzada con un RMSE de $3.85e^{-13}$ y un R^2 de 0.98, se procedió a realizar un análisis de residuos. Se procedió con los siguientes pasos:

1. Cálculo de residuos

Se calcularon los residuos como la diferencia entre los valores observados y los valores predichos por el modelo de regresión lineal.

2. Análisis gráfico de residuos

Se crearon gráficos de dispersión de los residuos versus los valores predichos para visualizar patrones o estructuras no aleatorias. La ausencia de patrones en estos gráficos es indicativa de un buen ajuste del modelo.

3. Prueba de normalidad

Se aplicó la prueba de Shapiro-Wilk a los residuos para evaluar la normalidad de su distribución. Una distribución normal de los residuos es una suposición clave en la regresión lineal, ya que afecta la validez de las pruebas de hipótesis y los intervalos de confianza.

4. Gráfico Q-Q (Quantile-Quantile)

Se elaboró un gráfico Q-Q para comparar la distribución de los residuos con una distribución normal teórica. Este gráfico es una herramienta visual poderosa para evaluar la normalidad de los residuos.

Es importante destacar que el RMSE en este análisis representa la desviación de los huevos en términos de predicción. Para otorgar un contexto más claro, a continuación se muestra el rango de datos de cada tipo de huevo.

- **Huevos Viables (HV):** 3686.0
- **Huevos Eclosionados (HE):** 957.0
- **Huevos en Canoa (HEC):** 2126.0

Estos rangos permiten una interpretación más profunda del RMSE, no solo como una cifra aislada, sino en relación con la variabilidad inherente de los datos de huevos, proporcionando una comprensión más precisa y aplicada a las condiciones reales del estudio.

7.1. Enfoque utilizando porcentaje

7.1.1. Random Forest

Resultados obtenidos del Modelo Random Forest. Se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran dos gráficas: una comparando valores reales con predicciones y otra mostrando la distribución del error.

Cuadro 1: Métricas del modelo utilizando Random Forest

	RMSE	R ²
Huevos Viables	10.76	1.00
Huevos Eclosionados	9.95	0.99
Huevos en Canoa	11.85	1.00

Fuente: Elaboración propia

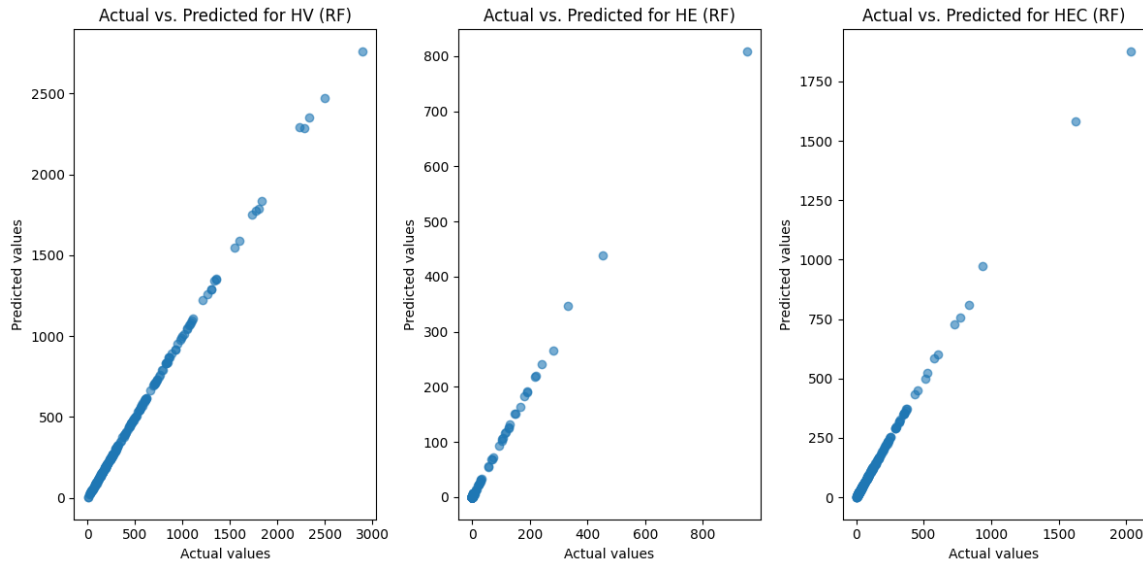


Figura 16: Reales vs predicciones

Fuente: Elaboración propia

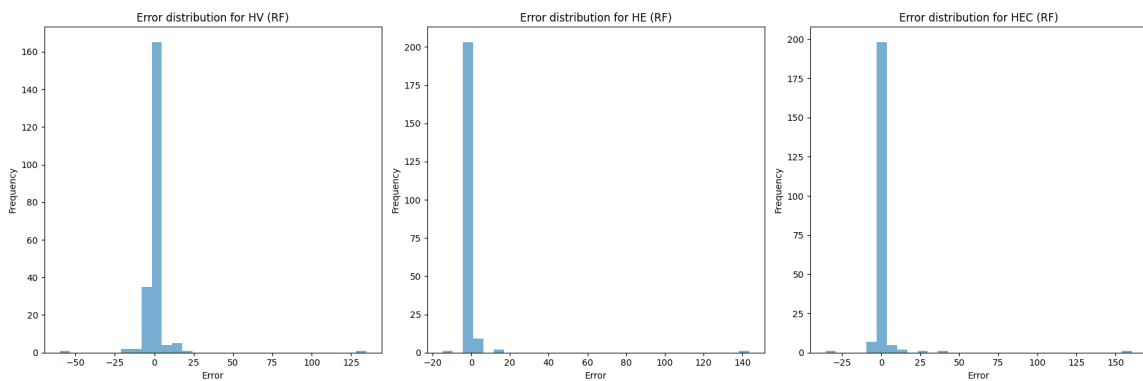


Figura 17: Error en predicciones

Fuente: Elaboración propia

El modelo Random Forest mostró un desempeño notable en la predicción de los tres tipos de huevos: viables, eclosionados y en canoa. Para huevos viables, la raíz del error cuadrático (RMSE) fue de 10.76 y un coeficiente de determinación (R²) de 1.00, sugiriendo un ajuste

casi perfecto. En cuanto a los huevos eclosionados, el modelo alcanzó un RMSE de 9.95 y un R^2 de 0.99, indicando también un alto grado de precisión. Finalmente, para los huevos en canoa, el RMSE fue de 11.85 con un R^2 de 1.00. Aunque los valores de R^2 cercanos o iguales a 1 sugieren que el modelo explica casi toda la variabilidad de los datos, es inusual obtener valores tan altos, lo que podría indicar un ajuste excesivo. En general, el Random Forest demostró ser un modelo poderoso para esta tarea.

La Figura 16 demuestra que el modelo generalmente predice con precisión, aunque hay desviaciones en valores más altos. La densidad de puntos en rangos inferiores sugiere una mayor precisión o más datos en esas áreas. La Figura 17 revela que la mayoría de los errores están centrados cerca de cero, con algunas excepciones más grandes, indicando un buen desempeño global pero con áreas específicas que requieren atención. En conjunto, las gráficas sugieren que el modelo es efectivo pero con espacio para mejoras.

7.1.2. Máquina de Vectores (SVM)

Resultados obtenidos del Modelo SVM. De igual forma se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran dos gráficas: una comparando valores reales con predicciones y otra mostrando la distribución del error.

Cuadro 2: Métricas del modelo utilizando SVM

	RMSE	R^2
Huevos Viables	476.73	0.06
Huevos Eclosionados	85.86	0.03
Huevos en Canoa	216.10	0.06

Fuente: Elaboración propia

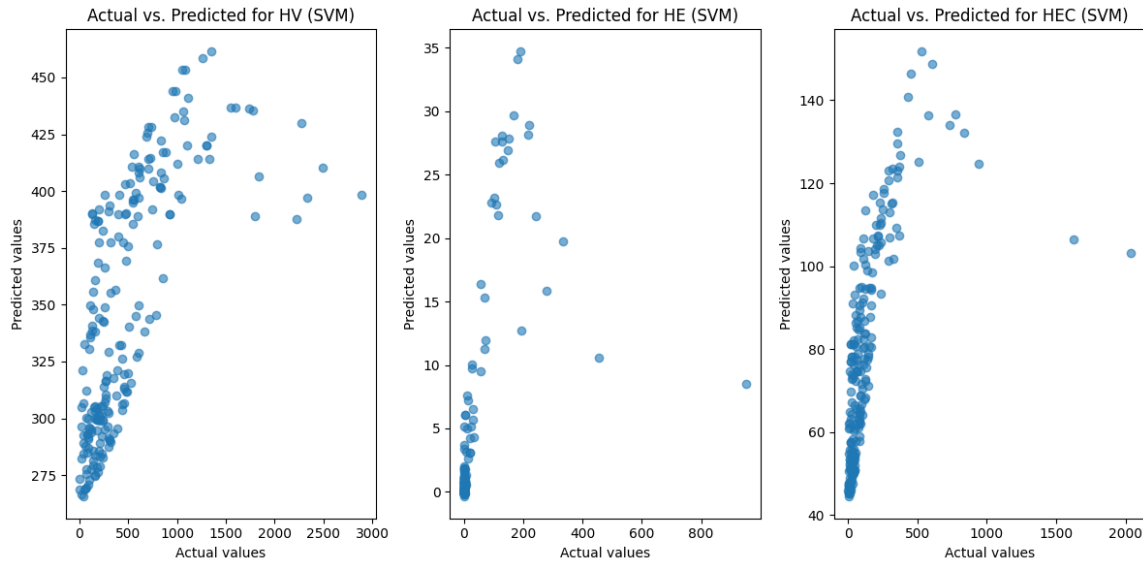


Figura 18: Reales vs predicciones

Fuente: Elaboración propia

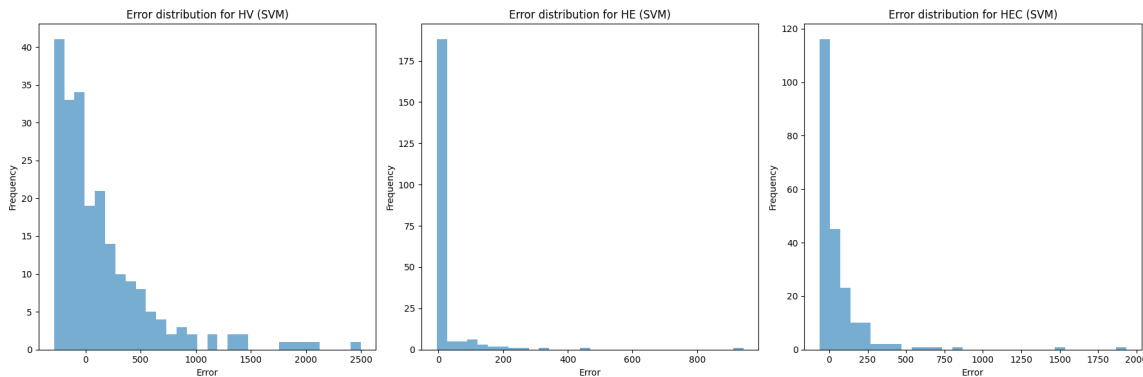


Figura 19: Error en predicciones

Fuente: Elaboración propia

El modelo SVM mostró un desempeño subóptimo en la predicción de los tres tipos de huevos: viables, eclosionados y en canoa. Las raíces de los errores cuadráticos medios (RMSE) son altos, particularmente para los huevos viables con un RMSE de 476.73. Además, los coeficientes de determinación R^2 para las tres categorías son muy bajos, siendo 0.06 para huevos viables y en canoa y 0.03 para huevos eclosionados, lo que indica que el modelo solo explica una pequeña fracción de la variabilidad de los datos. Estos valores sugieren que el SVM no captura adecuadamente las relaciones subyacentes en el conjunto de datos, lo que lleva a predicciones imprecisas.

En la Figura 18, HV, HE y HEC muestran que el modelo SVM tiende a hacer predicciones precisas, especialmente en rangos inferiores de valores. Sin embargo, hay una variabilidad observable a medida que los valores reales aumentan, especialmente en HV y HEC. La Figura 19 revelan que la mayoría de los errores para HV y HEC están concentrados cerca de cero, pero HE muestra una dispersión más amplia de errores, indicando posibles áreas de mejora

en esa métrica. En conjunto, estas gráficas sugieren que, en ciertos rangos de datos el modelo no funciona correctamente.

7.1.3. Red Neuronal Artificial (ANN)

Continuando con los resultados obtenidos del Modelo ANN. Se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran tres gráficas: una comparando valores reales con predicciones, otra mostrando la distribución del error y otra mostrando el historial de entrenamiento.

Cuadro 3: Métricas del modelo utilizando ANN

	RMSE	R^2
Huevos Viables	250.55	0.74
Huevos Ecllosionados	18.30	0.95
Huevos en Canoa	79.70	0.87

Fuente: Elaboración propia

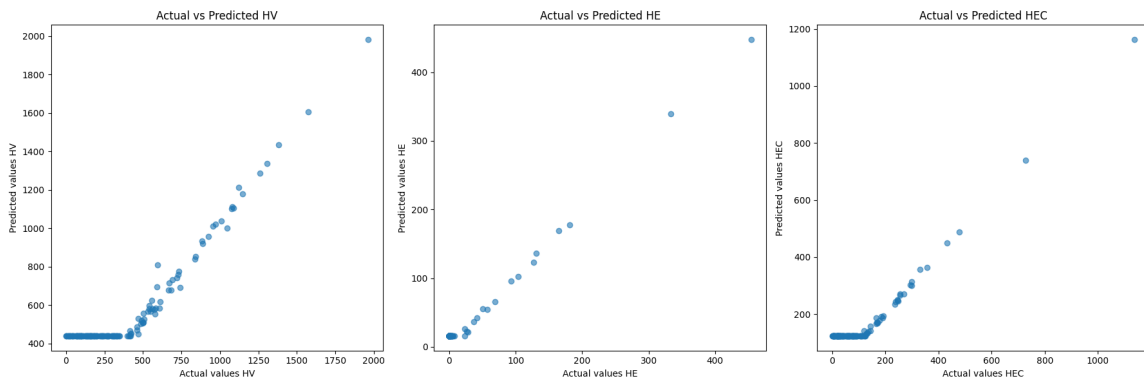


Figura 20: Reales vs predicciones

Fuente: Elaboración propia

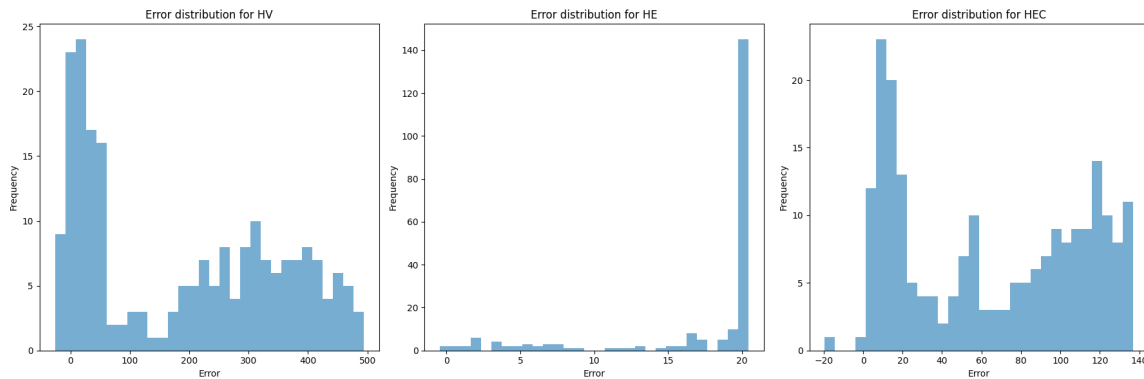


Figura 21: Error en predicciones

Fuente: Elaboración propia

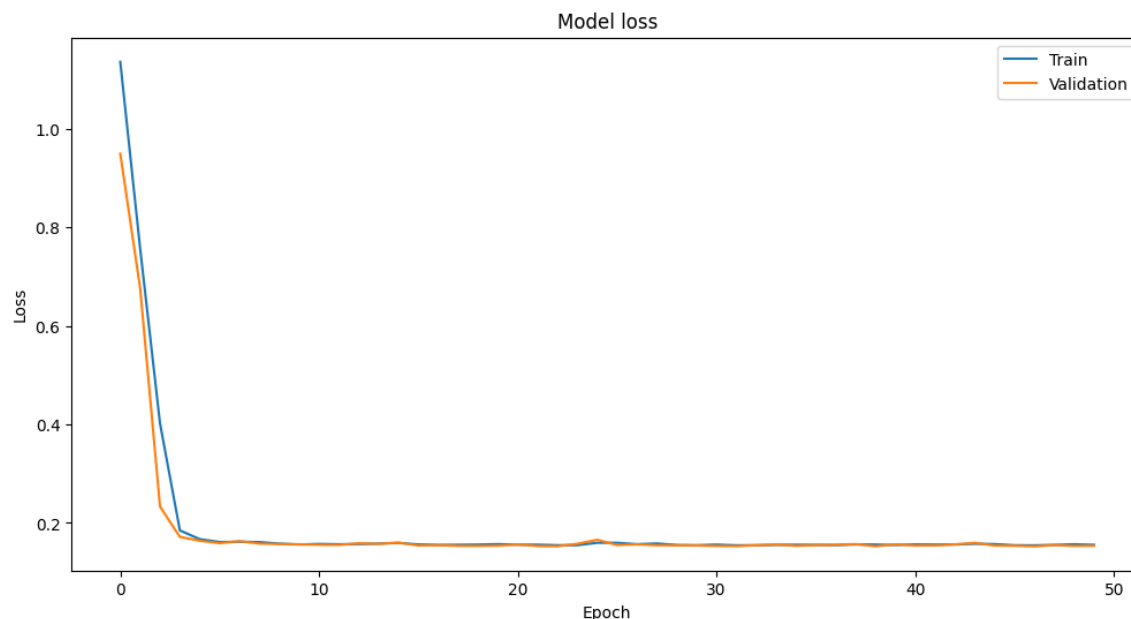


Figura 22: Historial de entrenamiento

Fuente: Elaboración propia

El modelo de Red Neuronal Artificial (ANN) para la predicción de los tres tipos de huevos mostró un desempeño robusto, con un RMSE de 250.55 para huevos viables, 18.30 para huevos eclosionados y 79.70 para huevos en canoa. Aunque hay margen de mejora en las predicciones de huevos viables, las métricas son particularmente prometedoras para los huevos eclosionados y en canoa, con coeficientes de determinación R^2 de 0.95 y 0.87, respectivamente. Estos resultados sugieren que, a diferencia del SVM, el ANN captura de manera efectiva las relaciones en el conjunto de datos, aunque sería beneficioso continuar ajustando el modelo para optimizar aún más su precisión.

La Figura 20, muestra la relación entre los valores reales y predichos de tres categorías diferentes: Huevos Viables (HV), Huevos Eclosionados (HE) y Huevos en Canoa (HEC). En la gráfica de HV, la mayoría de las predicciones están alineadas con los valores reales, mostrando una tendencia creciente hacia el extremo superior. Por otro lado, en la gráfica de HE, las predicciones también siguen una tendencia creciente, pero hay menos puntos y están más dispersos. Finalmente, en la gráfica de HEC, las predicciones están mayormente agrupadas en el extremo inferior, con algunos puntos dispersos en valores más altos. Es evidente que el modelo tiene una precisión variable en las diferentes categorías, siendo más preciso en HV y menos en HEC. En la gráfica de HV y HEC muestra puntos alineados horizontalmente esto indica que para diferentes valores reales, el modelo está ofreciendo la misma predicción o predicciones muy similares. Esto puede indicar que el modelo tiene dificultades para distinguir entre ciertas observaciones y generaliza al predecir un valor constante para esas observaciones. La Figura 21 muestra la distribución del error en las diferentes categorías de huevos. Esta figura revela cómo se distribuyen los errores en las predicciones, con HE mostrando una distribución de error muy centrada y HEC con un rango más amplio. Finalmente, la Figura 22 muestra la pérdida del modelo durante el entrenamiento y validación a lo largo de las épocas. Se observa una rápida disminución inicial de la pérdida, que se estabiliza hacia

el final, indicando posiblemente la convergencia del modelo.

7.1.4. Regresión Lineal

Resultados obtenidos del Modelo de Regresión Lineal. Se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran dos gráficas: una comparando valores reales con predicciones y otra mostrando la distribución del error.

Cuadro 4: Métricas del modelo utilizando Regresión Lineal

	RMSE	R^2
Huevos Viables	1.89	0.99
Huevos Eclosionados	1.10	0.99
Huevos en Canoa	1.91	0.99

Fuente: Elaboración propia

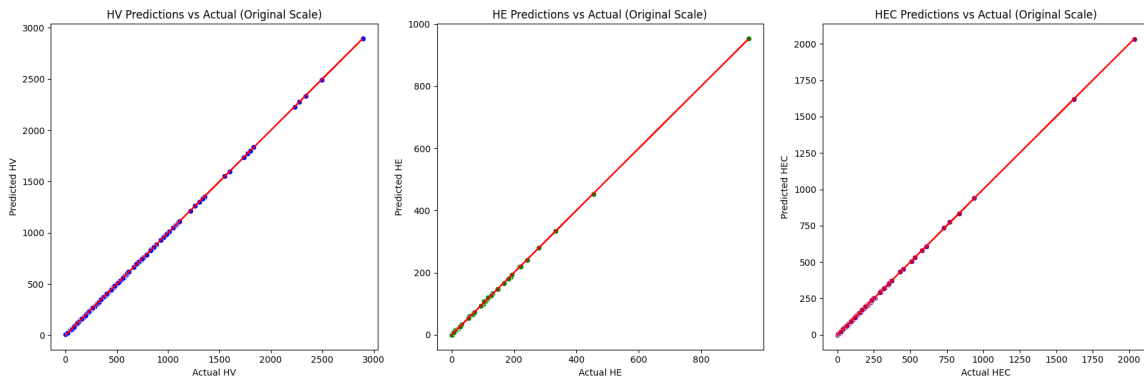


Figura 23: Reales vs predicciones

Fuente: Elaboración propia

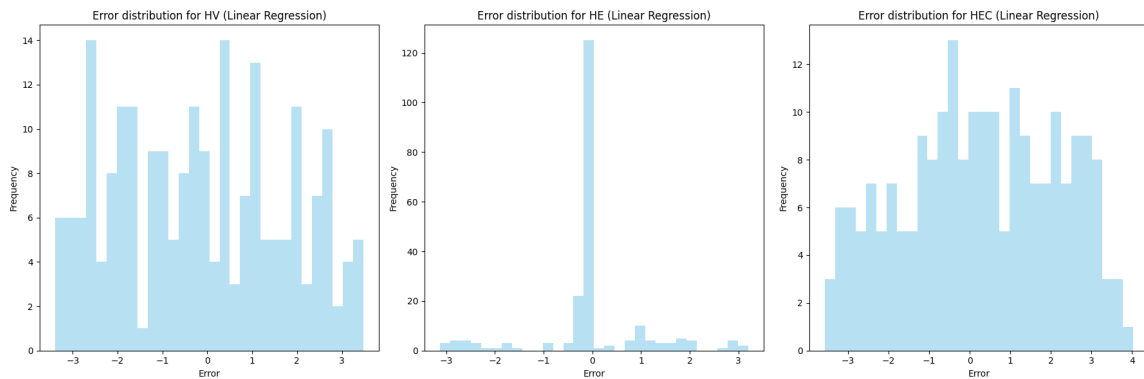


Figura 24: Error en predicciones

Fuente: Elaboración propia

El modelo de Regresión Lineal para la predicción de los tres tipos de huevos ha mostrado un desempeño excepcionalmente bueno. El RMSE para huevos viables, huevos eclosionados y huevos en canoa son 1.89, 1.10 y 1.91 respectivamente, valores que son considerablemente bajos. Indicando que, en promedio, las predicciones del modelo se desvían de los valores reales por menos de dos huevos en todas las categorías. Lo más impresionante es el coeficiente de determinación R^2 para las tres categorías, que es de 0.99. Esto sugiere que el modelo de Regresión Lineal explica el 99% de la variabilidad en los datos en todas las categorías, lo que es indicativo de un ajuste casi perfecto. Estos resultados sugieren que, para este conjunto de datos en particular, un modelo de Regresión Lineal puede ser más que suficiente para capturar las relaciones subyacentes y hacer predicciones precisas.

Las figuras presentadas muestran las distribuciones de error y las comparaciones de predicciones versus valores reales para los tres tipos de huevo: HV, HE y HEC utilizando regresión lineal. En la Figura 24, HE muestra una gran concentración cerca de 0, mientras que HV y HEC tienen una distribución más variada, esto podría deberse a que HE tiene mucho menor ruido y menor variabilidad en cuanto a su rango de datos por lo que es propenso a menos errores en la predicción. La Figura 23 ilustra la precisión de las predicciones. HV y HEC muestran una correspondencia cercana entre las predicciones y los valores reales, indicando un buen ajuste del modelo. Sin embargo, la gráfica de HE muestra una leve dispersión, lo que sugiere que las predicciones no son tan precisas para esa métrica.

7.2. Enfoque utilizando densidad poblacional

7.2.1. Random Forest

Resultados obtenidos del Modelo Random Forest. Se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran dos gráficas: una comparando valores reales con predicciones y otra mostrando la distribución del error.

Cuadro 5: Métricas del modelo utilizando Random Forest

	RMSE	R^2
Huevos Viables	14.65	0.99
Huevos Eclosionados	20.99	0.94
Huevos en Canoa	38.81	0.96

Fuente: Elaboración propia

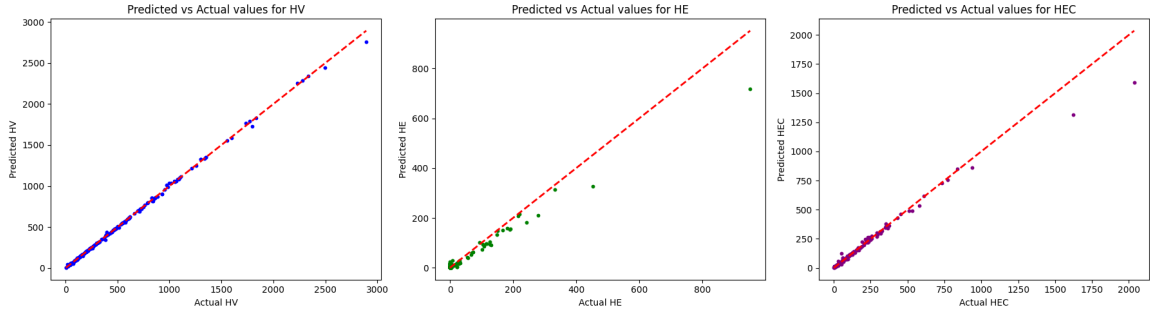


Figura 25: Reales vs predicciones

Fuente: *Elaboración propia*

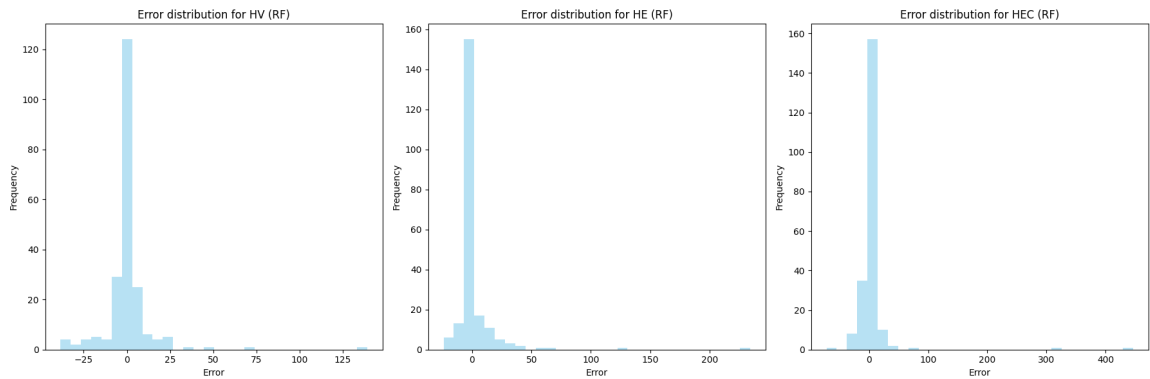


Figura 26: Error en predicciones

Fuente: *Elaboración propia*

El modelo Random Forest para la predicción de los tres tipos de huevos presenta RMSE de 14.65, 20.99 y 38.81 huevos para las categorías de huevos viables, eclosionados y en canoa, respectivamente. A pesar de estas desviaciones, el modelo captura eficientemente la variabilidad en los datos, con coeficientes de determinación R^2 de 0.99, 0.94 y 0.96 para cada categoría. Estos valores indican un desempeño sólido del modelo, aunque hay diferencias en el error entre las categorías.

La Figura 26 muestra la distribución del error en las predicciones para HV, HE y HEC. En HV y HE, la mayoría de los errores están agrupados cerca de cero, indicando una precisión general alta, aunque HEC muestra errores más dispersos en valores grandes. La Figura 25 muestra los valores predichos y reales. HV tiene una alineación cercana a la línea de identidad, especialmente en valores bajos, indicando precisión. HE, aunque sigue una tendencia similar, presenta desviaciones en algunos puntos. HEC muestra un patrón coherente pero tiene puntos atípicos importantes que sugieren una tendencia no lineal.

7.2.2. Máquina de Vectores (SVM)

Resultados obtenidos del SVM. Se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran dos gráficas: una comparando valores reales con predicciones y otra mostrando la distribución del error.

Cuadro 6: Métricas del modelo utilizando SVM

	RMSE	R ²
Huevos Viables	29.12	0.99
Huevos Eclosionados	0.10	0.99
Huevos en Canoa	0.11	0.99

Fuente: Elaboración propia

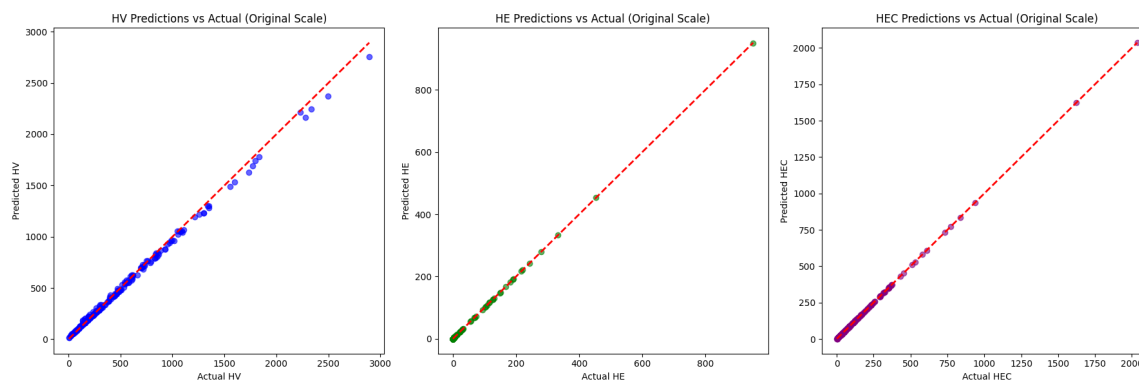


Figura 27: Reales vs predicciones

Fuente: Elaboración propia

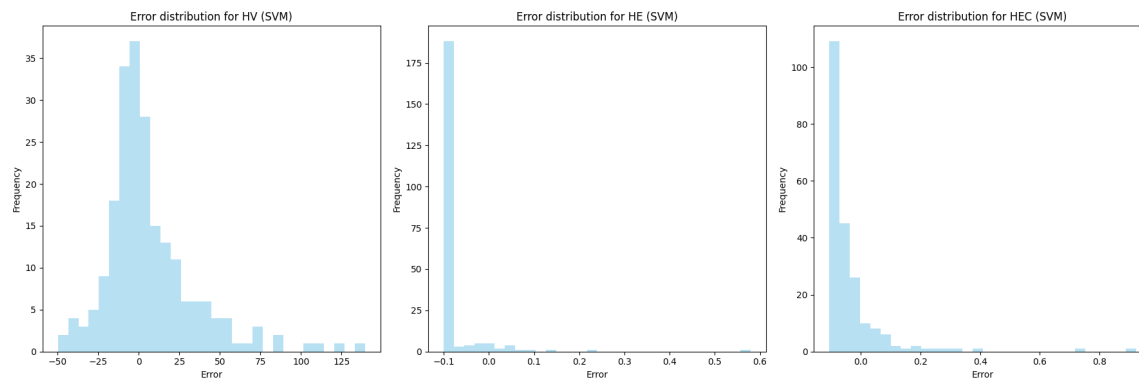


Figura 28: Error en predicciones

Fuente: Elaboración propia

El modelo SVM para la predicción de los tipos de huevos muestra RMSE de 29.12, 0.10 y 0.11 huevos para las categorías de huevos viables, eclosionados y en canoa, respectivamente. Aunque para los huevos viables el RMSE es mayor, sigue siendo razonablemente bajo. Para huevos eclosionados y en canoa, es extremadamente bajo, lo que sugiere una precisión casi perfecta en las predicciones. Los coeficientes de determinación son de 0.99 para cada categoría, lo que indica que el modelo explica el 99% de la variabilidad en los datos, reflejando un ajuste casi perfecto a los datos en todas las categorías.

En la Figura 28, vemos variaciones en las magnitudes y frecuencias de los errores. La

gráfica de HV muestra una distribución más amplia y centrada en torno a cero, mientras que HE y HEC tienen errores más concentrados cerca de cero. La Figura 27 indica la precisión del modelo: la cercanía de los puntos a la línea diagonal roja (línea ideal) sugiere cuán precisas son las predicciones. HV y HEC muestran una alineación cercana con la línea ideal, indicando predicciones precisas, mientras que HE presenta una dispersión moderada, señalando ciertas imprecisiones en el modelo.

7.2.3. Red Neuronal Artificial (ANN)

Los resultados obtenidos del Modelo ANN. Se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran tres gráficas: una comparando valores reales con predicciones, otra mostrando la distribución del error y otra mostrando el historial de entrenamiento.

Cuadro 7: Métricas del modelo utilizando ANN

	RMSE	R^2
Huevos Viables	21.22	0.99
Huevos Eclosionados	4.96	0.99
Huevos en Canoa	8.51	0.99

Fuente: Elaboración propia

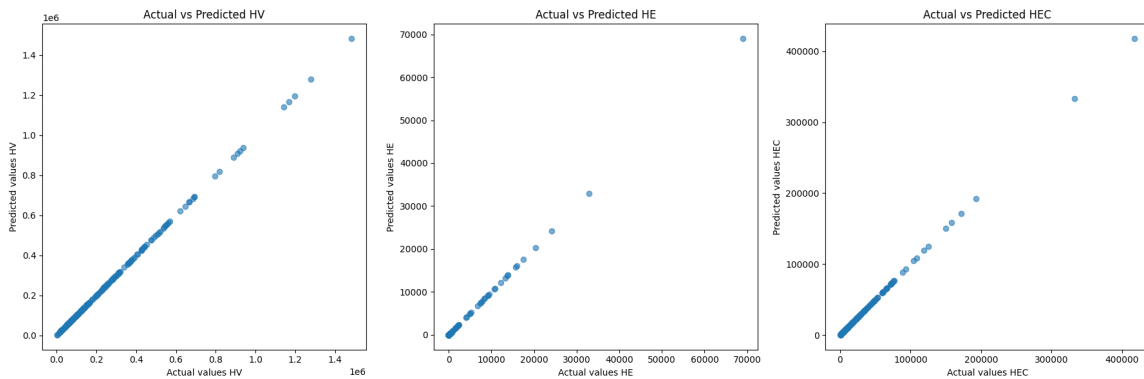


Figura 29: Reales vs predicciones

Fuente: Elaboración propia

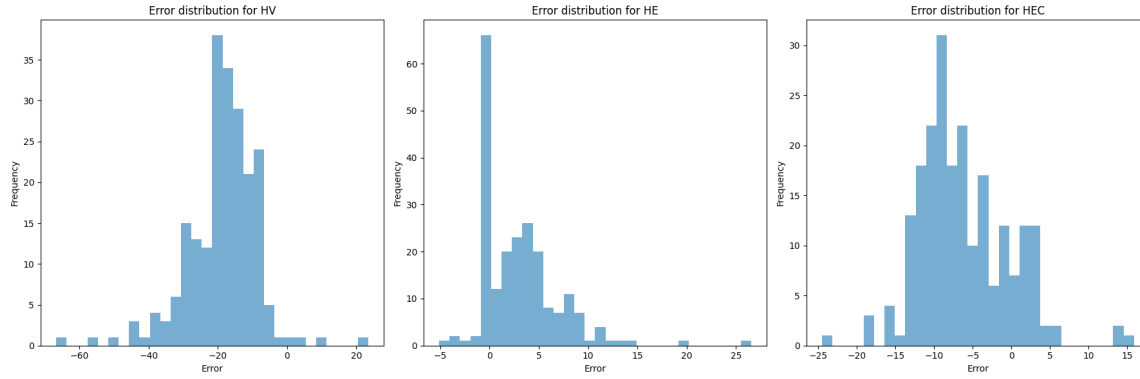


Figura 30: Error en predicciones

Fuente: *Elaboración propia*

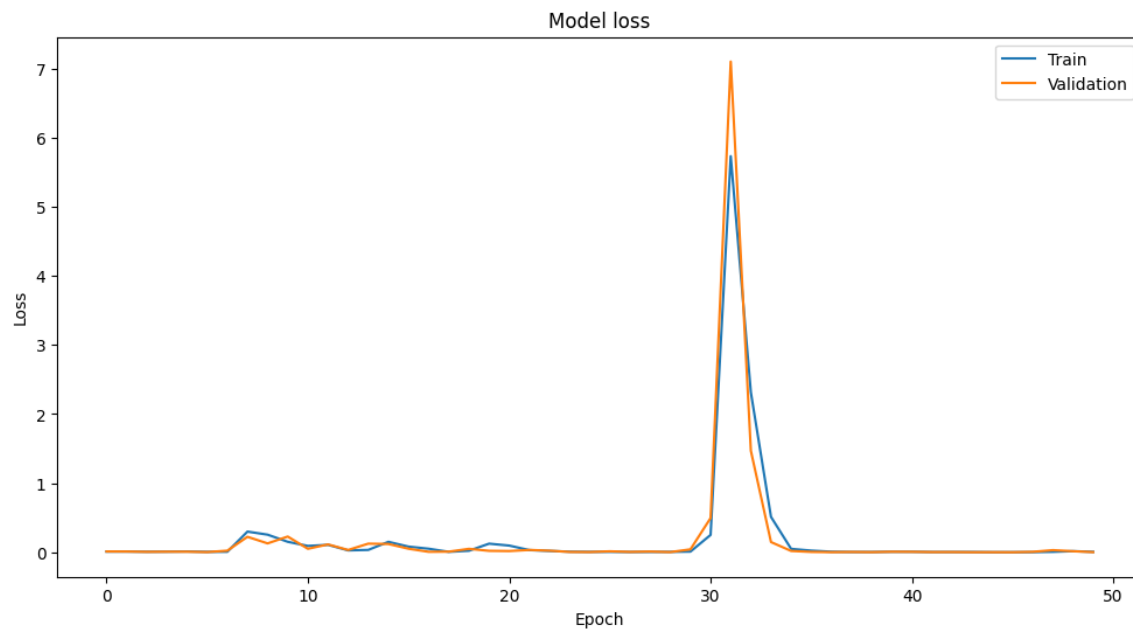


Figura 31: Historial de entrenamiento

Fuente: *Elaboración propia*

El modelo de Red Neuronal Artificial (ANN) para la predicción de los tres tipos de huevos presenta RMSE de 21.22, 4.96 y 8.51 huevos para las categorías de huevos viables, eclosionados y en canoa, respectivamente. Aunque esta métrica varía entre categorías, es razonablemente baja. Lo más destacado es que el modelo captura de manera efectiva la variabilidad en los datos, con coeficientes de determinación R^2 de 0.99 para cada categoría, lo que sugiere un ajuste casi perfecto del modelo a los datos en todas las categorías.

La Figura 29 muestra la comparación entre los valores reales y predichos de huevos en tres categorías: Huevos Viables (HV), Huevos Eclosionados (HE) y Huevos en Canoa (HEC). En general, las predicciones siguen de cerca los valores reales, especialmente en el caso de HV. Sin embargo, hay una mayor variabilidad en las predicciones de HE y HEC, particularmente en valores más altos. Esto sugiere que el modelo tiene una precisión razonable pero puede

mejorar, especialmente al predecir grandes cantidades de huevos eclosionados y en canoa. La Figura 30 representa las distribuciones de error para HV, HE y HEC. Es evidente que los errores para HE están centrados alrededor de cero, mientras que HV tiene una distribución más sesgada hacia la izquierda y HEC muestra un ligero sesgo hacia la derecha. Por último, la Figura 31 muestra la evolución de la pérdida durante las épocas de entrenamiento para ambos conjuntos, de entrenamiento y validación. Una observación interesante es cerca de la época 30, hay un pico muy pronunciado en la pérdida para ambos conjuntos (entrenamiento y validación), esto puede indicar algún tipo de problema o perturbación, como un cambio abrupto en la tasa de aprendizaje o un error en los datos. Sin embargo, tras el pico la pérdida de ambos conjuntos disminuye rápidamente, acercándose a los valores que tenía antes del pico. Esto sugiere que, tras el evento perturbador, el modelo fue capaz de corregirse y volver a una trayectoria de aprendizaje adecuada.

7.2.4. Regresión Lineal

Por último, los resultados obtenidos del modelo de Regresión Lineal. Se muestra una tabla con métricas clave como el RMSE y R^2 , proporcionando una visión de su precisión. Además, se muestran dos gráficas: una comparando valores reales con predicciones y otra mostrando la distribución del error.

Cuadro 8: Métricas del modelo utilizando Regresión Lineal

	RMSE	R^2
Huevos Viables	$8.35e^{-13}$	0.97
Huevos Eclosionados	$8.51e^{-14}$	0.96
Huevos en Canoa	$1.94e^{-13}$	0.98

Fuente: Elaboración propia

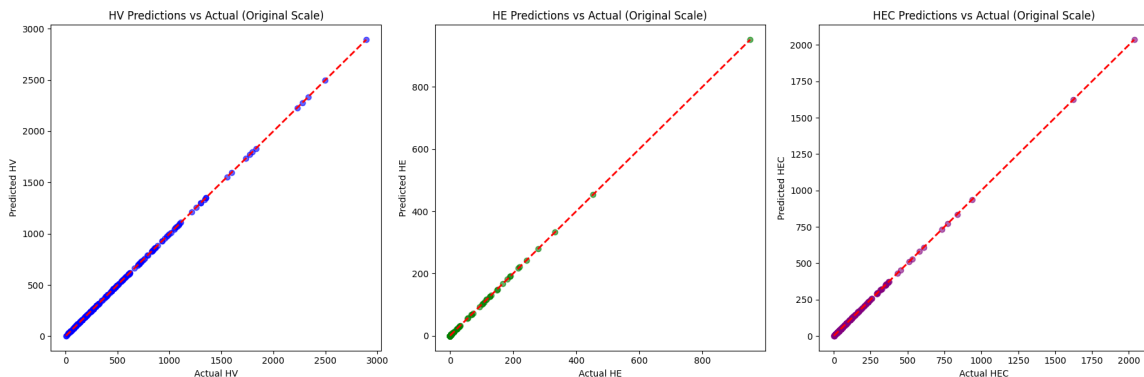


Figura 32: Reales vs predicciones

Fuente: Elaboración propia

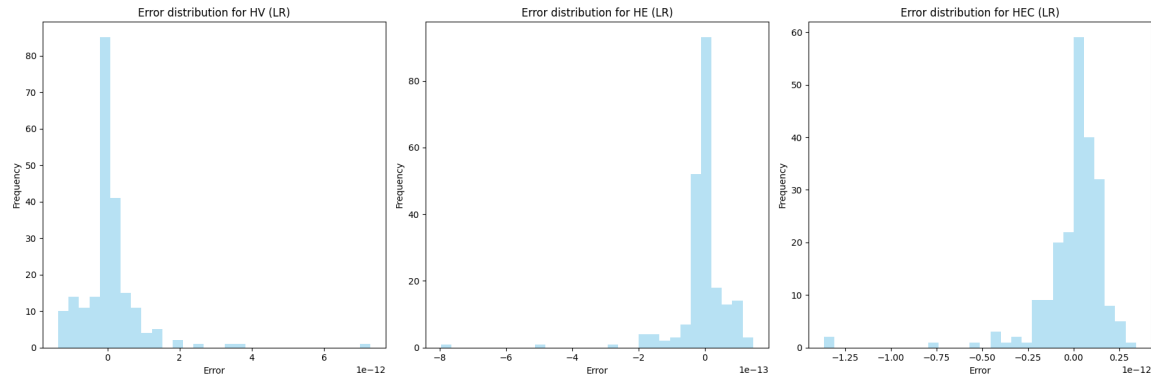


Figura 33: Error en predicciones

Fuente: *Elaboración propia*

El modelo de Regresión Lineal para la predicción de los tres tipos de huevos muestra errores extremadamente bajos, con RMSE de $8.35e^{-13}$, $8.51e^{-14}$ y $1.94e^{-13}$ huevos para las categorías de huevos viables, eclosionados y en canoa, respectivamente. Estos valores de RMSE son prácticamente cero, lo que sugiere una precisión casi perfecta en las predicciones. Adicionalmente, los coeficientes de determinación R^2 son altos, con valores de 0.97, 0.96 y 0.98 para cada categoría, lo que indica que el modelo explica entre el 96% y el 98% de la variabilidad en los datos. El modelo de Regresión Lineal presenta un desempeño excepcional en la predicción de los tres tipos de huevos, con errores extremadamente bajos y un ajuste casi perfecto a los datos.

En la Figura 33, HV y HE muestran un pico predominante cerca del error cero, indicando predicciones precisas para la mayoría de los puntos de datos, mientras que HEC presenta una distribución más amplia, lo que sugiere una mayor dispersión del error sin llegar a ser muy grande. La Figura 32 indica que las predicciones están en general alineadas con los valores reales, habiendo muy pocas desviaciones.

7.3. Justificación de modelo elegido

7.3.1. Validación cruzada

Cuadro 9: Métricas utilizando validación cruzada para los modelos de regresión lineal

	RMSE	R^2
RL Muestras de Huevos	1.69	0.96
RL Densidad Poblacional	$3.85e^{-13}$	0.98

Fuente: *Elaboración propia*

Después de una revisión meticulosa de las métricas y las visualizaciones correspondientes a todos los modelos implementados, se observó una tendencia clara: los modelos de Regresión

Lineal demostraron ser superiores en términos de métricas como el RMSE y R^2 . Además, las gráficas asociadas a estos modelos reflejaron un ajuste más coherente y cercano a los datos reales en comparación con otros enfoques. Esta superioridad se mantuvo tanto en el enfoque basado en muestras de huevos como en el enfoque orientado a la densidad poblacional de los sustratos.

Dada esta observación, se decidió concentrar los esfuerzos en una comparación más detallada entre los dos modelos de Regresión Lineal. La validación cruzada, una herramienta robusta en la evaluación de modelos, se empleó para esta tarea. Ambos modelos presentan errores bastante pequeños, pero cabe destacar que el enfoque de densidad poblacional tiene errores mínimos por lo que es el que mejor desempeño muestra para realizar este tipo de predicciones.

7.3.2. Análisis de residuos

Resultados del análisis de residuos. Se presentan dos gráficas que presentan los residuos vs los valores predichos y un gráfico Q-Q. Además, muestra métricas de la prueba de Shapiro-Wilk para las distintas categorías de huevo.

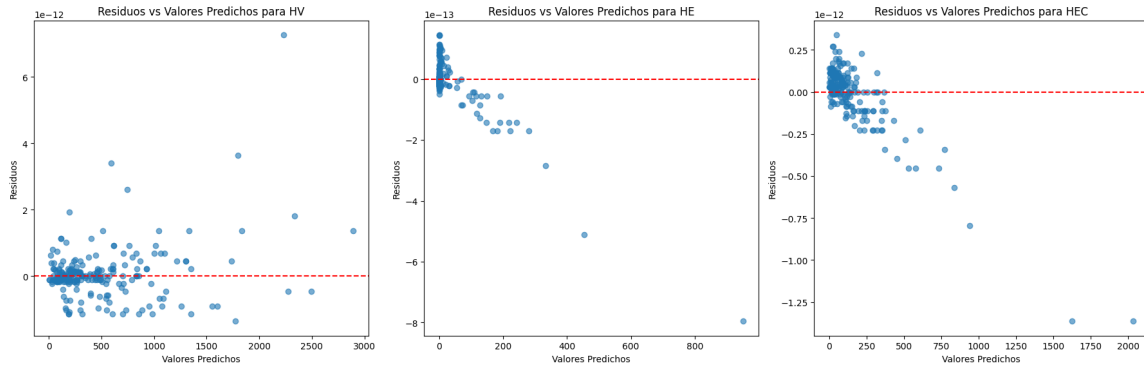


Figura 34: Residuos vs valores predichos

Fuente: *Elaboración propia*

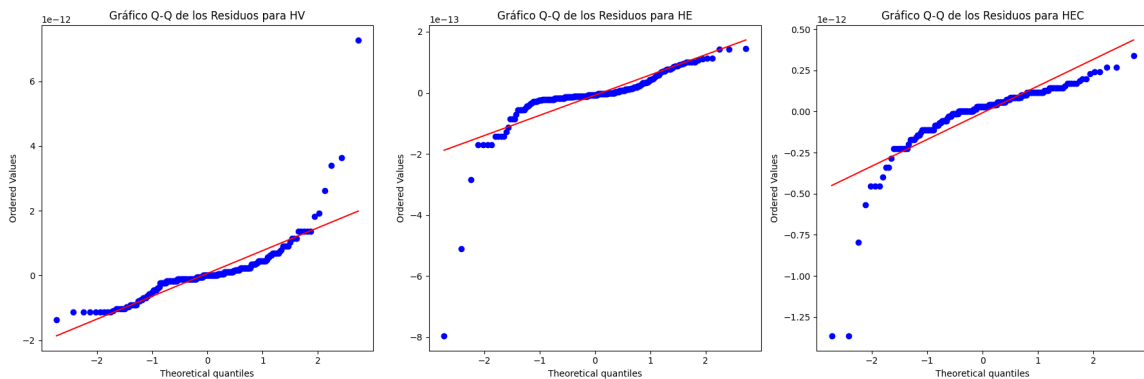


Figura 35: Gráfico Q-Q

Fuente: *Elaboración propia*

Cuadro 10: Prueba de Shapiro-Wilk

	Estadístico	p-valor
Huevos Viables	0.82	$5.98e^{-9}$
Huevos Eclosionados	0.71	$7.22e^{-12}$
Huevos en Canoa	0.78	$1.22e^{-9}$

Fuente: Elaboración propia

La Figura 34 muestra los residuos frente a los valores predichos. Parece que los datos están bastante dispersos alrededor de la línea de cero, lo cual es un buen signo de que los residuos son aleatorios y no hay patrones evidentes, sin embargo tiene varios datos alejados de 0 lo que indica que hay un área de mejora al modelo. La Figura 35, por otro lado, muestra una visualización de cómo los residuos se comparan con una distribución normal teórica. La mayoría de los puntos se adhieren cercanamente a la línea roja, aunque hay algunas desviaciones, especialmente para "Huevos Eclosionados". La prueba de Shapiro-Wilk evalúa la normalidad de los datos, y los valores de estadístico (0.82 para Huevos Viables, 0.71 para Huevos Eclosionados, y 0.78 para Huevos en Canoa) sugieren que los datos se desvían ligeramente de una distribución normal.

A pesar de estas observaciones, el análisis de residuos sugiere áreas donde el modelo podría ser refinado, en comparación con otros modelos probados, el modelo de Regresión Lineal todavía presenta un rendimiento superior.

Por lo tanto, aunque hay espacio para mejoras en el modelo, en este contexto particular, la Regresión Lineal demostró ser la más adecuada. Es vital considerar tanto las métricas estándar como el análisis de residuos para obtener una imagen completa del rendimiento del modelo. En el futuro, estos insights del análisis de residuos pueden ser utilizados para ajustar y mejorar aún más el modelo.

7.4. Métricas del backend

Resultados obtenidos del backend. Se muestra una tabla con métricas de tiempo promedio a distinta cantidad de iteraciones.

Cuadro 11: Métricas de tiempo de respuesta a distintas peticiones e iteraciones

	10 iteraciones	100 iteraciones	1000 iteraciones	10000 iteraciones
GET	157 ms	150 ms	152 ms	154 ms
POST	147 ms	152 ms	150 ms	156 ms
PUT	140 ms	144 ms	146 ms	142 ms

Fuente: Elaboración propia

El análisis de las métricas de tiempo de respuesta para las distintas peticiones (GET,

POST, PUT) e iteraciones (10, 100, 1000, 10000) revela una consistencia y eficiencia notable en el rendimiento del backend. Los tiempos promedio de respuesta oscilan entre 140 ms y 156 ms, demostrando una estabilidad en la respuesta del servidor incluso bajo cargas crecientes de solicitudes. Para las peticiones GET, se observa una variación mínima de 157 ms a 154 ms, sugiriendo una gestión eficaz de las operaciones de lectura. En las solicitudes POST, los tiempos fluctúan ligeramente entre 147 ms y 156 ms, indicando un manejo estable de las operaciones de escritura. De manera similar, las solicitudes PUT mantienen tiempos de respuesta consistentemente bajos, desde 140 ms hasta 146 ms, destacando una eficiencia particular en las actualizaciones de datos. Estos resultados reflejan una optimización sobresaliente del backend, capaz de manejar diferentes tipos y volúmenes de solicitudes con una eficiencia y consistencia impresionantes.

7.5. Inicio de sesión y manejo de usuarios

Resultados obtenidos del manejo de usuarios. Se muestran capturas del login de la aplicación web. Además, se muestran datos de la base de datos MySQL y de firebase en donde se muestra el almacenamiento de los usuarios.

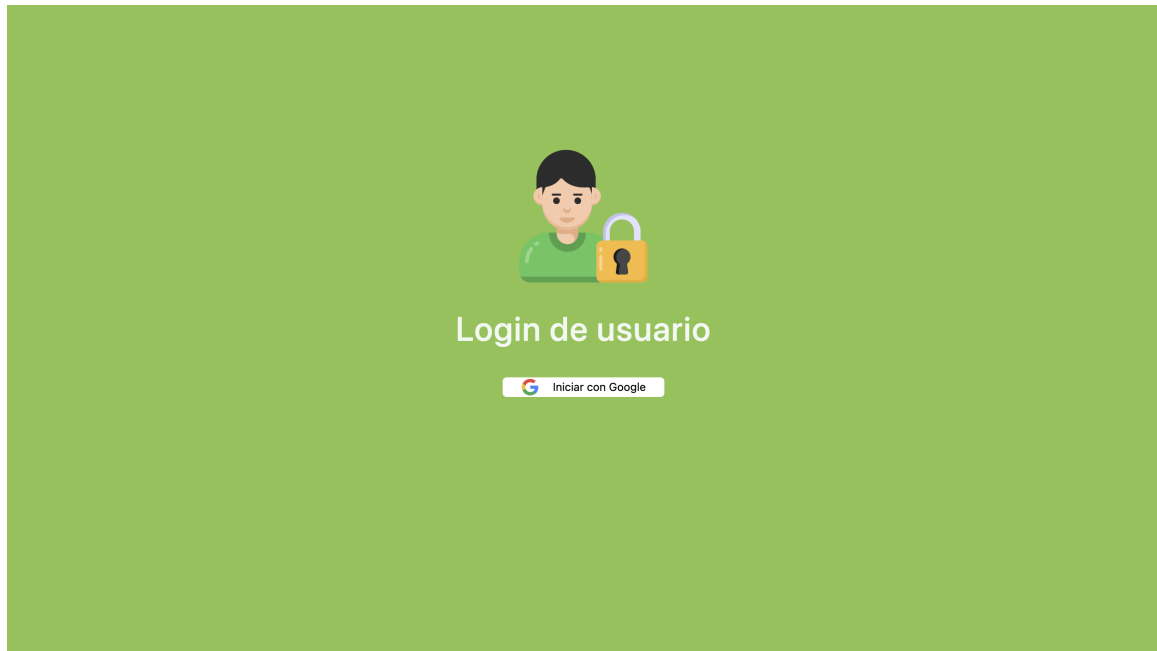


Figura 36: Login de usuario

Fuente: Elaboración propia

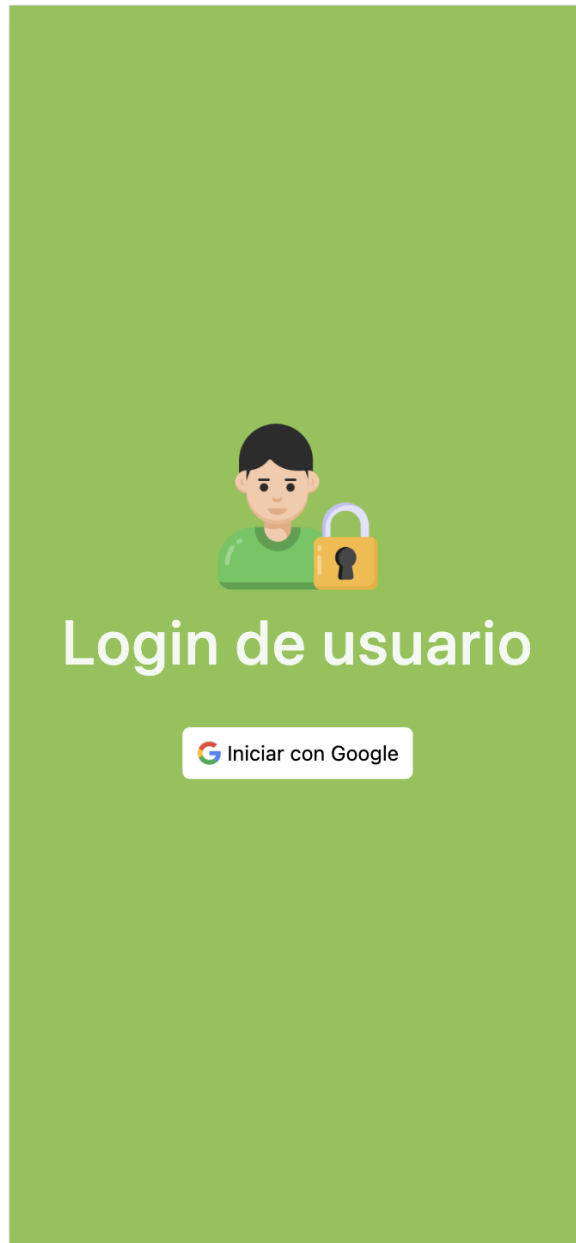


Figura 37: Login de usuario responsive

Fuente: Elaboración propia

	id	firebase_id	name	email
1	21	01httk46fTPHoqcBv0vnloHas6E3	GABRIEL ROLANDO QUIROZ ALBIZURES	qui19255@uvg.edu.gt
2	22	iMp3TX1sZJYr2VNHKshanx8JkVz2	José Pablo Ponce	jopaponcepivo@gmail.com
3	23	OrEyallknkNahOWaKgJ4g8HFefV2	JOSE PABLO PONCE PIVARAL	pon19092@uvg.edu.gt
4	24	NeTPmMqrBsXXwIZim7UStBp4hfE2	Gabriel Quiroz	gabrielquirozr@gmail.com
5	25			juandavid123@gmail.com
6	26			joel123@gmail.com
7	27			juan123@gmail.com
8	28	fE7qDtPsWAbBXO2k4HG6oNWlxol2	Isabella Sequeira	isabellasequeira88@gmail.com

Figura 38: Registro de usuarios en MySQL

Fuente: *Elaboración propia*

Identificador	Proveedores	Fecha de creación	Fecha de acceso	UID de usuario
isabellasequeira88@gmail...		26 oct 2023	26 oct 2023	fE7qDtPsWAbBXO2k4HG6oNWlxol2
juan123@gmail.com		25 oct 2023	25 oct 2023	usd6QbbNA9SalqS4OW99PqQA1...
joel123@gmail.com		25 oct 2023	25 oct 2023	t7x88050s3V5Cnk0CNPEaAtPjOE2
juandavid123@gmail.com		25 oct 2023	25 oct 2023	hiw74FuyAkdN0wv3pYRoHGI29sQ2
qui19255@uvg.edu.gt		31 ago 2023	25 oct 2023	01httk46fTPHoqcBv0vnloHas6E3
jopaponcepivo@gmail.com		21 jul 2023	19 oct 2023	iMp3TX1sZJYr2VNHKshanx8JkVz2
pon19092@uvg.edu.gt		14 jul 2023	27 oct 2023	OrEyallknkNahOWaKgJ4g8HFefV2
gabrielquirozr@gmail.com		10 jul 2023	26 oct 2023	NeTPmMqrBsXXwIZim7UStBp4hf...

Filas por página: 50 1 - 8 of 8

Figura 39: Registro de usuarios en firebase

Fuente: *Elaboración propia*

En las figuras 36 y 37, se presenta la interfaz de inicio de sesión de la aplicación tanto en su versión de escritorio (PC) como en su versión responsive, diseñada para dispositivos móviles. La versión de escritorio muestra una disposición clara y accesible, optimizada para una experiencia de usuario fluida en pantallas más grandes. Por otro lado, la versión responsive demuestra la adaptabilidad de la aplicación, manteniendo la funcionalidad y la estética en una presentación compacta y eficiente para pantallas más pequeñas. Este enfoque de diseño garantiza una experiencia de usuario coherente y de alta calidad, independientemente del dispositivo utilizado para acceder a la aplicación.

Adicionalmente, las figuras 38 y 39 evidencian la eficacia del sistema de registro de usuarios. Estas imágenes muestran que los usuarios que se registran en la aplicación son almacenados de forma segura tanto en Firebase como en MySQL, asegurando la integridad y la confiabilidad de la gestión de usuarios. Este sistema de almacenamiento dual refuerza

la seguridad y proporciona una base sólida para la administración de datos de usuario.

7.6. Base de datos

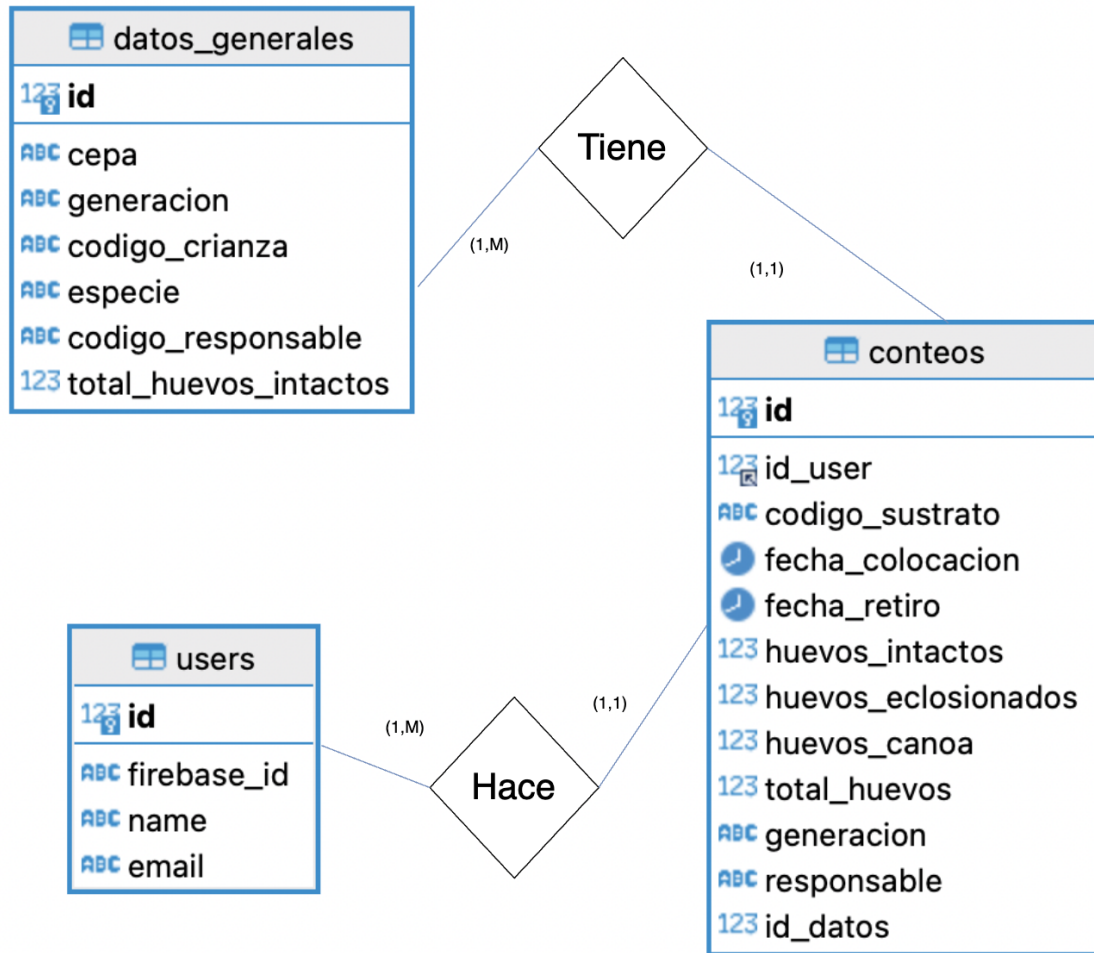


Figura 40: Diagrama entidad-relación de la base de datos

Fuente: *Elaboración propia.*

La Figura 40 presenta el diagrama entidad-relación de la base de datos utilizada en el proyecto. Este diagrama ilustra la estructura organizada y las relaciones entre las diferentes tablas de la base de datos. La tabla conteos almacena información detallada sobre los conteos de huevos realizados, mientras que la tabla de datos generales recopila datos ambientales y de condiciones relevantes para el análisis. Por su parte, la tabla usuarios gestiona la información de los usuarios registrados en la aplicación. Dadas las relaciones existentes (Uno a muchos) de las tablas de datos generales y usuarios con los conteos, esta última contiene el id de esas dos tablas.

Se desarrolló una aplicación que facilita la predicción de la cantidad de huevos de mosquito bajo condiciones de laboratorio. Esta herramienta contribuye a la agilización del proceso de conteo de huevos.

Se implementó un modelo de regresión lineal que predice la cantidad de huevos de mosquito, y almacena los resultados para su posterior análisis y manipulación. Los resultados obtenidos en el RMSE ($3.85e^{-13}$), R^2 (0.98) y la validación cruzada indican que el modelo funciona bien. Sin embargo, el análisis de residuos realizado reveló que, aunque el modelo proporciona un punto de partida útil, existen algunos aspectos que requieren atención y refinamiento. El análisis de residuos mostró patrones inesperados y desviaciones de la normalidad, lo que sugiere que el modelo actual tiene puntos de mejora. Estos hallazgos resaltan la necesidad de la inclusión de variables adicionales o la exploración de enfoques de modelado alternativos.

De los modelos evaluados se identificó que el modelo de regresión lineal es el más adecuado para predecir la viabilidad de los huevos de mosquito presentando un RMSE de $3.85e^{-13}$ y un R^2 de 0.98, además de mostrar un gran rendimiento en la validación cruzada.

Se determinó una técnica efectiva para fotografiar los huevos de mosquito, garantizando imágenes claras y de alta calidad que permiten un análisis más preciso. Esta técnica estandarizada puede ser aplicada en futuros estudios para garantizar consistencia y precisión.

Se implementó una base de datos adecuada que permite almacenar datos de los conteos y se desarrolló un backend eficiente con tiempos de respuesta medios que no pasan de 160ms. Las operaciones de lectura, actualización y creación se gestionan sin problemas, lo que facilita la interacción con los datos y asegura que la información esté siempre actualizada y sea accesible.

Se introdujeron características de inicio de sesión y registro de usuarios. Esto no solo mejora la seguridad de la aplicación sino que también permite la personalización y el seguimiento de las contribuciones individuales.

Recomendaciones

Se recomienda entrenar y probar los modelos con un dataset más extenso. Al hacerlo, se puede mejorar la precisión del modelo y asegurar que sea generalizable a diferentes situaciones y condiciones.

Se recomienda añadir más variables al dataset que puedan influir en la viabilidad de los huevos de mosquito. Factores como la humedad, el origen geográfico de los huevos, la luz y otras condiciones ambientales pueden ser cruciales para obtener predicciones más precisas. Añadir más variables puede mejorar la capacidad de generalización del modelo a nuevos datos o condiciones y mejorar las métricas del análisis de residuos.

Para asegurar la calidad y consistencia en la toma de fotografías, es recomendable invertir en un estereoscopio con cámara incorporada. Esta herramienta no solo facilitará la captura de imágenes sino que también garantizará que estas sean de la mejor calidad posible.

-
- [1] C. Sun, J. Nimbalkar y R. Bedi, *Predicting Future Mosquito Larval Habitats Using Time Series Climate Forecasting and Deep Learning*, 2022. arXiv: 2208.01436 [cs.LG].
- [2] K. Y. Lee, N. Chung y S. Hwang, “Application of an artificial neural network (ANN) model for predicting mosquito abundances in urban areas,” *Ecological Informatics*, vol. 36, págs. 172-180, 2016, ISSN: 1574-9541. DOI: <https://doi.org/10.1016/j.ecoinf.2015.08.011>. dirección: <https://www.sciencedirect.com/science/article/pii/S1574954115001521>.
- [3] C. for Disease Control y Prevention, *Life Cycles of Aedes aegypti and Aedes albopictus Mosquitoes*, 2023. dirección: <https://www.cdc.gov/mosquitoes/about/life-cycles/aedes.html>.
- [4] del, *Diagrama del ciclo de vida del mosquito gratis*, feb. de 2021. dirección: <https://es.vecteezy.com/arte-vectorial/2046831-doodle-diagrama-del-ciclo-de-vida-del-mosquito>.
- [5] *The Mosquito - Mosquitopia*. NCBI Bookshelf, A gravid female mosquito aims to hatch her eggs in a water resource that is sufficiently rich in nutrients and long-lasting to allow the larvae to grow, develop and produce pupae from which adult mosquitoes can successfully emerge. dirección: <https://www.ncbi.nlm.nih.gov/books/NBK585164/>.
- [6] K. P. Velo E, “Enhancement of *Aedes albopictus* collections by ovitrap and sticky adult trap,” 2016. dirección: <https://pubmed.ncbi.nlm.nih.gov/27102015/>.
- [7] A. A.N, “Life Cycle, Bio-ecology and DNA Barcoding of mosquitoes *Aedes aegypti* (Linnaeus) and *Aedes albopictus* (Skuse),” Life Cycle, Bio-ecology and DNA Barcoding of mosquitoes *Aedes aegypti* (Linnaeus) and *Aedes albopictus* (Skuse). dirección: https://www.researchgate.net/publication/320144961_Life_Cycle_Bio-ecology_and_DNA_Barcoding_of_mosquitoes_Aedes_aegypti_Linnaeus_and_Aedes_albopictus_Skuse.
- [8] L. M. Beck-Johnson, W. A. Nelson, K. P. Paaijmans, A. F. Read, M. B. Thomas y O. N. Bjørnstad, “Temperature-Mediated Effects on Mosquito Population Dynamics in the Tropics: Implications for Malaria Risk,” *Journal of The Royal Society Interface*, vol. 8, n.º 62, págs. 311-318, 2011.

- [9] D. E. Impoinvil, J. O. Kongere y W. A. Foster, “Effects of larval growth condition and water availability on desiccation resistance and its physiological basis in adult *Anopheles gambiae* sensu stricto,” *Malaria journal*, vol. 15, n.º 1, págs. 1-13, 2016. dirección: <https://malariajournal.biomedcentral.com/articles/10.1186/1475-2875-9-225>.
- [10] J. E. Gentile, S. S. C. Rund y G. R. Madey, “Analysis of a mosquito life cycle model,” *Malaria Journal*, vol. 14, n.º 1, págs. 1-14, 2015. dirección: <https://malariajournal.biomedcentral.com/articles/10.1186/s12936-015-0775-1>.
- [11] D. A. Ewing, B. V. Purse, C. A. Cobbold, S. M. Schäfer y S. M. White, “Uncovering mechanisms behind mosquito seasonality by integrating mathematical models and daily empirical population data: *Culex pipiens* in the UK,” *Parasites & Vectors*, vol. 12, n.º 74, 2019. dirección: <https://parasitesandvectors.biomedcentral.com/articles/10.1186/s13071-019-3321-2>.
- [12] B. B. Anjana Prasad Sreesa Sreedharan, “Eggs of the mosquito *Aedes aegypti* survive desiccation by rewiring their polyamine and lipid metabolism,” *PLOS Biology*, 2023. dirección: <https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.3002342>.
- [13] B. Schölkopf y A. J. Smola, *Support Vector Machines for Pattern Classification*. Springer, 2002. dirección: <https://link.springer.com/book/10.1007/978-1-84996-098-4>.
- [14] *illustration of support vector machine (svm)*, 2019. dirección: https://www.researchgate.net/figure/Illustration-of-support-vector-machine-SVM-principle-a-Input-space-is-mapped-to-the_fig4_331801455.
- [15] L. Breiman, “Random forests,” *Machine learning*, vol. 45, n.º 1, págs. 5-32, 2001. dirección: <https://link.springer.com/article/10.1023/A:1010933404324>.
- [16] J. Riebesell, *Random Forest*, abr. de 2022. dirección: <https://tikz.net/random-forest/>.
- [17] I. Goodfellow, Y. Bengio y A. Courville, *Deep Learning*. MIT Press, 2016. dirección: <http://www.deeplearningbook.org/>.
- [18] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
- [19] N. R. Draper y H. Smith, *Applied Regression Analysis*. John Wiley Sons, 1998.
- [20] R. E. Wilk, R. H. Myers y S. L. Myers, *Probability and Statistics for Engineers and Scientists*, 9.ª ed. Pearson, 2011.
- [21] S. S. Shapiro y M. B. Wilk, “An analysis of variance test for normality (complete samples),” *Biometrika*, vol. 52, n.º 3/4, págs. 591-611, 1965.
- [22] N. specified, “Programming languages: history and future,” *ACM Digital Library*, No specified. dirección: <https://dl.acm.org/doi/10.1145/361454.361485>.
- [23] M. W. Docs, *JavaScript*, 2023. dirección: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>.
- [24] “TypeScript: The starting point for learning TypeScript.” (2023), dirección: <https://www.typescriptlang.org/>.
- [25] “Welcome to Python.org.” (2023), dirección: <https://www.python.org/>.

- [26] *Keras documentation: Developer guides*, <https://keras.io/>.
- [27] *TensorFlow - An end-to-end open source machine learning platform*, <https://www.tensorflow.org/>.
- [28] Google, *Firebase Documentation*, 2023. dirección: <https://firebase.google.com/docs>.
- [29] R. Lanciaux, *Modern Front-end Architecture: Optimize Your Front-end Development with ...* 2023. dirección: <https://link.springer.com/book/10.1007/978-1-4842-6625-0>.
- [30] W. (H. A. T. W. Group), *HTML Living Standard*, 2023. dirección: <https://html.spec.whatwg.org/>.
- [31] W. (W. W. Consortium), *Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification*, 2011. dirección: <https://www.w3.org/TR/CSS2/>.
- [32] A. Team, *Angular Documentation*, 2023. dirección: <https://angular.io/docs>.
- [33] Node.js, *Node.js Documentation*, 2023. dirección: <https://nodejs.org/en/docs/>.
- [34] B. Jellinek, *Backend Development*. 2023. dirección: <https://backend-development.github.io/>.
- [35] *Express - Fast, unopinionated, minimalist web framework for Node.js*, <https://expressjs.com/>.
- [36] *TensorFlow.js - TensorFlow.js: Machine Learning for the Web and Beyond*, <https://www.tensorflow.org/js>.
- [37] *Postman*, <https://www.postman.com/>, Plataforma para el desarrollo de APIs, 2023.
- [38] *Uptime Robot*, <https://uptimerobot.com/>, Servicio de monitoreo de tiempo de actividad, 2023.
- [39] *Flask*, <https://flask.palletsprojects.com/>, Microframework para aplicaciones web en Python, 2023.
- [40] M. L. Gillenson, *Fundamentals of Database Management Systems, 3rd Edition*. John Wiley & Sons, Inc, 2011. dirección: <https://www.wiley.com/en-us/Fundamentals+of+Database+Management+Systems%2C+3rd+Edition-p-9781119907442>.
- [41] *MySQL :: The world's most popular open source database*, <https://www.mysql.com/>.
- [42] P. Pollock, *Web Hosting For Dummies*. John Wiley & Sons, 2013. dirección: <https://www.wiley.com/en-us/Web+Hosting+For+Dummies-p-9781118540572>.
- [43] *Vercel - Develop. Preview. Ship*, <https://vercel.com/>.

11.1. Manual de Usuario para la aplicación web

11.1.1. Acceso a la plataforma

1. Ingrese a la plataforma a través del siguiente enlace: <https://mosquitodetector.vercel.app/login>.
2. Inicie sesión utilizando la opción de Google con su correo.

11.1.2. Navegación principal

- Una vez iniciada la sesión, será redirigido automáticamente a la página de inicio.
- En la parte superior, encontrará una barra de navegación con las siguientes opciones:
 - **Realizar conteo:** Página inicial donde se realiza el conteo de huevos de mosquito.
 - **Historial:** Muestra una tabla con la información general de los conteos realizados.
 - **Acerca de:** Proporciona información general sobre la página y sus desarrolladores.
 - **Desconectar:** Permite cerrar la sesión en la plataforma.

11.1.3. Realizar conteo

1. En la página ‘Realizar Conteo’, podrá cargar imágenes de huevos de mosquito, ya sea arrastrando y soltando las imágenes o seleccionándolas desde su explorador de archivos.
2. Tras cargar las imágenes, aparecerá un botón de **Continuar**. Al presionarlo, las imágenes serán enviadas para su análisis.

3. Se mostrará un pop-up con los resultados del análisis, incluyendo la cantidad de huevos totales, clasificados como viables, en canoa y eclosionados. También se proporcionará un espacio para llenar información adicional. En este pop-up encontrará tres botones:
 - **Guardar:** Al presionarlo, se guardará la información ingresada en las tablas y se cerrará el pop-up.
 - **Regresar:** Cierra el pop-up sin guardar la información.
 - **Predicción:** Abre un nuevo pop-up donde podrá ingresar variables para un modelo de predicción.
4. Si seleccionó **Predicción**, ingrese las variables solicitadas en el nuevo pop-up y luego presione **Predecir** para obtener y visualizar las predicciones. Si desea guardar esta predicción, presione **Guardar**. En caso contrario, puede presionar **Regresar** para cerrar el pop-up e ingresar nuevas imágenes si lo desea.
 - **Guardar:** Al presionarlo, se redijira al pop-up inicial en donde se podran ingresar los datos generales del conteo y guardarlos en la base de datos.
 - **Regresar:** Cierra el pop-up sin guardar la información.
 - **Predecir:** Llena la nueva tabla con los datos de los huevos viables, huevos eclosionados y huevos en canoa.

11.1.4. Historial:

1. En la sección ‘Historial’, encontrará una tabla con todos los conteos realizados.
2. Cada entrada de la tabla tiene un enlace **Más**, que al ser presionado, lo redirigirá a ‘Historial Detallado’. Aquí se mostrará la misma información de la tabla anterior junto con una tabla adicional de ‘Conteo de Huevecillos’ con información adicional del conteo seleccionado.
3. En esta sección, podrá editar la información en las tablas si lo considera necesario.

Con estos pasos, podrá hacer uso completo de la plataforma Mosquito Detector para cargar, analizar y revisar el conteo de huevos de mosquito.