



Investigación Administrativa

Vol.54, núm. 136, 2025

ISSN: 1870-6614

ISSN: 2448-7678

ria@ipn.mx

Viabilidad Económica del Software de Facturación en la Nube

Economic Viability of Software Billing in the Cloud



Esta obra está bajo una [Licencia Creative Commons Atribución-NoComercial 4.0 Internacional](https://creativecommons.org/licenses/by-nc/4.0/).



INSTITUTO POLITÉCNICO NACIONAL, ESCUELA SUPERIOR DE COMERCIO Y ADMINISTRACIÓN, SECCIÓN DE POSGRADO E INVESTIGACIÓN




Viabilidad Económica del Software de Facturación en la Nube

Economic Viability of Software Billing in the Cloud

 10.35426/IAv54n136.07

Eduardo Pérez Carrillo
Tecnológico Nacional de México, México.
eduardo.car.perez@gmail.com
 0009-0007-4937-7123

Diego Adiel Sandoval Chávez
Tecnológico Nacional de México, México.
diego.sc@cdjuarez.tecnm.mx
 0000-0002-2536-1844

Recepción: 15 de agosto de 2025
Aprobación: 08 diciembre de 2025
 Acceso abierto diamante

Resumen

El objetivo de esta investigación es evaluar la viabilidad económica de implementar tecnologías de cómputo en la nube en proyectos de software de facturación, aprovechando las horas de inactividad de los desarrolladores durante la fase de diseño. El método consistió en un análisis prospectivo mediante el cálculo de flujos de efectivo y la tasa interna de retorno en 13 casos reales, complementados con 2000 simulaciones generadas por bootstrapping. Además, se desarrolló un modelo de red neuronal multicapa para automatizar la decisión de viabilidad. Los resultados evidencian que los proyectos con más de diez desarrolladores son económicamente viables. Los hallazgos confirman una correlación significativa entre el tamaño del equipo y la rentabilidad. La originalidad radica en la aplicación de inteligencia artificial a la evaluación económica de proyectos tecnológicos. Las limitaciones incluyen el reducido número de casos y la necesidad de ampliar la base de datos en futuras investigaciones.

Palabras clave: Evaluación Económica, Proyectos de Software, Cómputo en la Nube, Clasificación Binaria, Red Neuronal Multicapa.

Clasificación JEL: M54, O36

Investigación Administrativa, 2025, vol. 54, núm. 136, julio-diciembre (publicación continua)

ISSN: 1870-6614 / 2448-7678

Abstract

The objective of this research is to assess the economic feasibility of implementing cloud computing technologies in billing software projects by strategically utilizing developers' idle hours during the design phase. The methodology involved a prospective financial analysis based on cash flow projections and the internal rate of return (IRR) applied to 13 real-world cases, complemented by 2,000 additional simulations generated through bootstrapping. Furthermore, a multilayer neural network model was developed to automate the evaluation of project viability. The results indicate that projects with teams of more than 10 developers are economically sustainable, revealing a strong and significant correlation between team size and profitability. The originality of this work lies in the integration of artificial intelligence into the economic assessment of technological projects. The main limitations include the relatively small sample size and the need to expand the dataset in future studies to enhance the model's generalizability.

Keywords: Economic evaluation, Software projects, Cloud computing, Bootstrapping, Binary Classification, Multilayer Neural Network.

JEL Classification: M54, O36

Introducción

La determinación de la viabilidad económica de un proyecto es un aspecto crucial en el proceso de toma de decisiones. Para asegurar su éxito, resulta esencial realizar un análisis integral de flujos de efectivo que permita evaluar si el proyecto cumplirá con las expectativas de beneficios económicos en el corto plazo. Tras este estudio, es necesario confirmar o descartar la viabilidad económica de una alternativa de negocio en el largo plazo o en condiciones diferentes. En resumen, es necesario determinar si la inversión será rentable, lo que requerirá una evaluación fundamentada, tendiente a lograr mejores resultados y una gestión más clara y eficiente (Quiñonez, Monserrate, & López, 2018).

El desarrollo de un proyecto de software se compone de una serie de fases estructuradas que permiten transformar los requerimientos iniciales del usuario en un producto tecnológico funcional. Las etapas típicas incluyen el análisis de requisitos, en el que se identifican y documentan las necesidades del cliente; el diseño técnico, en el cual se define la arquitectura del sistema; la codificación, que traduce el diseño en líneas de código ejecutables; las pruebas, destinadas a validar la calidad y funcionalidad del software; y el mantenimiento, que asegura su evolución y adaptación a lo largo del tiempo (Gupta, Rawal, & Barge, 2021). En el marco de las metodologías ágiles, especialmente Scrum, estas fases se integran dentro de ciclos iterativos y flexibles llamados sprints, que consisten en periodos cortos y definidos en los que el equipo de desarrollo trabaja sobre un conjunto específico de tareas seleccionadas del Product Backlog. Cada sprint finaliza con la entrega de un incremento funcional del producto, permitiendo así retroalimentación constante, mejora continua y una mayor alineación con las expectativas de los clientes (Schwaber & Sutherland, 2020; Valpadasu Hema et al., 2020).

Investigación Administrativa, 2025, vol. 54, núm. 136, julio-diciembre (publicación continua)

La empresa en la que se desarrolla la investigación, situada en Ciudad Juárez, México, se dedica al desarrollo de proyectos de software personalizados para clientes en la región binacional México-Estados Unidos. Durante el periodo del 2019 al 2024, la compañía completó 32 proyectos, de los cuales el 40% corresponde al área de facturación, consolidando esta área como su principal fuente de ingresos. Sin embargo, la gestión del tiempo muerto del equipo de desarrollo de software representa un reto importante. El tiempo muerto toma lugar durante las fases de diseño de cada sprint, ya que los desarrolladores no inician su trabajo en el proyecto hasta contar con la información recolectada en esta etapa. Esta información se recaba presencialmente en las oficinas de los clientes y no puede ser compartida al equipo por medios electrónicos debido a un acuerdo de confidencialidad. Así, los desarrolladores permanecen sin actividades durante este periodo. Además, por acuerdo entre la empresa y sus clientes, los desarrolladores deben estar asignados completamente al proyecto, por lo que no se les permite involucrarse en actividades de otros proyectos durante ese tiempo. Esta limitación genera inactividad en los desarrolladores durante esta etapa e incrementa considerablemente los costos de los proyectos, ya que los salarios de los desarrolladores deben ser cubiertos aun cuando no se tengan tareas facturables asignadas a estos. El costo semanal del equipo de desarrolladores representa más del 60% del costo total del proyecto, impactando significativamente la rentabilidad de la empresa. La tendencia en la cantidad de proyectos durante 2022 y 2024, muestra un aumento del 225% en relación con años anteriores. Las pérdidas asociadas al tiempo muerto han crecido exponencialmente, alcanzando incrementos del 500% en el periodo.

El desarrollo de software está en línea con la cuarta revolución industrial que ha transformado los sistemas productivos globales, destacando la integración de tecnologías en áreas como la inteligencia artificial, Big Data y la digitalización de procesos. Según Schwab (2015), esta revolución se caracteriza por su velocidad y capacidad para fusionar los mundos físico, digital y biológico. Morera-Carballo (2022) subraya que las empresas se han adaptado a esta era mediante la adopción de resiliencia, versatilidad y la gestión de grandes volúmenes de datos, lo que ha potenciado la innovación tecnológica.

Por otra parte, la coyuntura de la pandemia de COVID-19 impulsó la digitalización como estrategia de recuperación y crecimiento empresarial (de Lucas Ancillo y Gavrila, 2023; Cheshmehzangi, 2021; Almeida et al., 2020). En este sentido, López, Niembro y Ramos (2014) evidencian el crecimiento acelerado de la industria del software en Latinoamérica, destacando la mano de obra calificada y competitiva en el mercado global. Por otro lado, investigaciones en Colombia (Díaz-Vega & Ospina-Ospina, 2023) identificaron retos en la sostenibilidad y gestión empresarial de las pequeñas y medianas empresas del sector, enfatizando la necesidad de metodologías ágiles para mejorar la calidad, reducir costos y acortar tiempos de entrega. De igual manera, destacan la importancia de una evaluación económica precisa para garantizar la viabilidad económica de proyectos de software, utilizando herramientas como el valor actual neto, la tasa interna de retorno y análisis de sensibilidad (Lin, 2023; Camacho Angulo, Banchón Jiménez, Barcia Villacreses, & Allauca Amaguaya, 2023; Fonseca Varela y Chalita Tovar, 2021).

Las redes neuronales artificiales conforman una herramienta esencial dentro del área del aprendizaje automático, estas se encuentran inspiradas en la estructura y funcionamiento del cerebro humano, buscando emular su proceso de análisis para la toma de decisiones. Se

componen de nodos interconectados, conocidos como neuronas, que se encargan de procesar, transmitir y dirigir información a través de diversas capas de una manera jerárquica. Este modelo permite a los sistemas computacionales aprender a partir de ejemplos, lo que resulta útil para tareas complejas como la clasificación, regresión o predicción de datos (Walczak, 2018; Dastres & Soori, 2021). Su fortaleza de estas es la capacidad para la detección de patrones no lineales, haciendo uso de grandes volúmenes de datos, lo que ha impulsado su adopción en múltiples áreas, tales como: automatización de procesos, reconocimiento de voz, clasificación y predicción de datos o análisis de imágenes, todo esto orientado a la toma de decisiones (García, 2021).

Debido a la importancia que representa el proceso de toma de decisiones utilizando redes neuronales, la matriz de confusión es un recurso fundamental para el proceso de evaluación de su desempeño, particularmente en modelos de clasificación. La matriz permite analizar las predicciones generadas por la red y contrastarlos con valores reales, identificando los aciertos y errores cometidos durante el proceso de clasificación, permitiendo cuantificar su magnitud. La información obtenida proporcionada por la matriz de confusión se organiza los siguientes casos: verdaderos positivos (VP), verdaderos negativos (VN), falsos positivos (FP) y falsos negativos (FN), los cuales permiten comprender con una mayor precisión el tipo de decisiones que está tomando el modelo, ofreciendo una visión más completa que la simple exactitud del modelo (Chicco, Warrens & Jurman, 2021). La matriz de confusión es útil en contextos en los que existen clases desbalanceadas o cuando es necesario minimizar los errores de algún tipo específico, como lo pueden ser los falsos negativos o positivos (Tharwat, 2021). De esta manera, la matriz de confusión permite cuantificar el rendimiento de las predicciones del modelo y proporciona información para gestionar la red en función de las decisiones esperadas.

Diversos estudios confirman la eficacia del uso de redes neuronales en tareas de predicción financiera en diversos rubros. Omole y Enke (2024) estudiaron la predicción del precio futuro del Bitcoin implementando y comparando diversos modelos de redes neuronales con un enfoque de clasificación binario para determinar subidas o caídas y así tomar decisiones de compra o de venta. El modelo de red neuronal utilizado alcanzó una precisión de 82.44% y un retorno anual del 6654%, evidenciando la efectividad que puede llegar a tener el uso de las redes neuronales en contextos reales de inversión. Por otro lado, Etim, Milke y Luca (2025) abordaron la predicción de movimientos intradía en el precio de las acciones de Apple, haciendo uso de redes neuronales de tipo perceptrón multicapa en tareas de clasificación buscando sustentar las decisiones de comprar o vender acciones. El modelo logro reducir la tasa de errores respecto al método convencional de analizar personalmente cada operación, incrementando la robustez y eficacia del proceso.

Debido a la relevancia del tiempo muerto en los proyectos de la empresa y su impacto negativo en los costos y la rentabilidad, es necesario abordar esta problemática desde una perspectiva analítica y propositiva. Este estudio busca evaluar económicamente la viabilidad de implementar tecnologías de cómputo en la nube en proyectos de facturación, a partir del aprovechamiento de las horas de inactividad de los desarrolladores, mediante un análisis prospectivo soportado por un modelo de red neuronal entrenado para predecir escenarios viables e inviables.

Método

En esta investigación se emplea un diseño dual retrospectivo-prospectivo, de temporalidad longitudinal. Este enfoque permite analizar la evolución y tendencias de los proyectos de software en relación con su viabilidad económica. El diseño longitudinal facilita la recolección de datos en múltiples momentos, permitiendo identificar patrones en las variables de interés. Con datos históricos de 13 proyectos, se examinó retrospectivamente la viabilidad económica y se generó una prospectiva de 2000 nuevos casos basados en estos proyectos, simulando escenarios económicos y ampliando el análisis. Se adoptan también enfoques descriptivos y exploratorios para evaluar la viabilidad económica en proyectos de software.

La investigación también describe las características de las herramientas de evaluación y la nube elegidas, mientras que un abordaje exploratorio adopta redes neuronales para determinar la evaluación económica mediante la Tasa Interna de Retorno (TIR), para así generar preguntas para futuros análisis. La TIR es la tasa de descuento que hace que el Valor Actual Neto (VAN) de todos los flujos de efectivo sea igual a cero. La ecuación 1 es la conformación genérica para calcular el flujo de efectivo neto.

$$0 = \sum_{t=1}^n \frac{\text{Flujo de efectivo en el periodo } t}{(1 + \text{Tasa de descuento})^t} - \text{Inversión Inicial} \quad (1)$$

La TIR ofrece un criterio que simplifica la comparación entre proyectos de distintas escalas y plazos. Según Ross, Westerfield y Jordan (2018), la TIR permite evaluar la rentabilidad comparativa y para decidir cuál proyecto resulta más atractivo desde el punto de vista financiero.

El criterio de selección-rechazo de proyectos establece que, si el VAN es igual a cero, lo que implica que la TIR del flujo analizado es igual al costo del capital utilizado para el descuento, el proyecto será considerado aceptable, ya que satisfará las demandas contractuales de los aportantes de los recursos de deuda y de aquellos que suministran recursos de riesgo. Por lo contrario, si el VAN es negativo, la TIR será inferior al tipo de descuento empleado, lo que hará que el proyecto sea financieramente inviable a la tasa de evaluación considerada. Es importante señalar que no se debe referir a estos proyectos como "no rentables", ya que, aunque puedan no serlo, el beneficio obtenido podría no ser suficiente para cumplir con las exigencias de los inversores respecto de una tasa mínima aceptable (Fernández, 2022).

La investigación utiliza datos de campo recolectados de proyectos reales y terminados de una empresa de desarrollo de software en el área de facturación, aplicando a estos la evaluación económica. Además, se emplea un enfoque sintético para analizar el uso de tecnologías de cómputo en la nube y aprovechar el tiempo muerto de los desarrolladores durante la fase de diseño de cada sprint, evaluando la viabilidad económica de este recurso.

Para la muestra de 13 proyectos de software, se recopiló información sobre costos, ingresos, tiempo muerto, duración de proyectos y sprints, así como el número de desarrolladores involucrados. La investigación, de carácter aplicado y cuantitativo, utilizó la TIR, comparándola con la Tasa Mínima Aceptable de Rendimiento (TMAR), y condujo un análisis prospectivo

mediante redes neuronales para evaluar la viabilidad económica de proyectos futuros. La TMAR representa el rendimiento mínimo que debe exigirse al proyecto por la renuncia a invertir los recursos financieros en alternativas con un nivel de riesgo similar. Una estimación adecuada de esta tasa es fundamental para calcular correctamente el VAN a partir de los flujos de efectivo esperados del proyecto (Rebollar-Rebollar, Posadas, Rebollar-Rebollar, Hernández, & González, 2020).

La investigación se desarrolló siguiendo una serie de fases que abarcan la elección del proveedor de herramientas de tecnologías de la nube, la cual es una fase crítica del proceso, debido a que es necesario conocer el costo del servicio de cómputo en la nube a utilizar para proceder con la creación de los flujos de efectivo de los proyectos. El cliente tomó la decisión de utilizar Microsoft Azure®, ya que dentro de la empresa su infraestructura está enfocada en Microsoft®, que va desde el sistema operativo instalado en las computadoras hasta cuentas de correo de dominio Outlook®, por lo que resulta más sencilla la integración y el uso de tecnologías de Microsoft® dentro de la empresa.

Los flujos de efectivos se enfocaron en los costos e ingresos generados por los desarrolladores de software semanalmente, utilizando el costo e ingreso por hora de estos profesionales, el número de semanas del proyecto y el número de semanas trabajadas para el cálculo de los flujos de efectivo. Por cada proyecto se crearon dos flujos de efectivo, uno de ellos es el flujo original, el cual solo toma en cuenta los costos e ingresos reportados, el segundo es el flujo de efectivo modificado tomando en cuenta los costos del uso de herramientas de cómputo en la nube calculado en la fase anterior y el ingreso esperado por el uso de estas tecnologías, que es equivalente a cobrar el costo semanal por los desarrolladores en las semanas de tiempo muerto.

Una vez determinados los flujos de efectivo, se procedió a realizar el cálculo de la TIR para los flujos de efectivo modificados, esto con la finalidad de observar el comportamiento económico de los proyectos con los nuevos gastos e ingresos al hacer uso de las herramientas de cómputo en la nube previamente cotizadas.

El cálculo de la TIR se llevó a cabo utilizando Python® como base y la función IRR de la librería numpy-financial, la cual solo requiere los flujos de efectivo para realizar el cálculo. La TMAR propuesta por la empresa es de 20%, con lo cual, si el valor que arroja la TIR es mayor al valor TMAR propuesto, el uso de tecnologías de cómputo en la nube se aprueba. Se generaron nuevos casos tomando como base los 13 casos originales. Para este proceso se utilizaron los principios del muestreo a posteriori sobre la muestra, remuestreo o bootstrapping (Bruch, 2022).

El siguiente paso fue determinar las características para el diseño de la red neuronal. La investigación busca determinar si un proyecto es viable económicamente para el uso de tecnologías de la nube con tan solo conocer la cantidad de desarrolladores, sprint y la cantidad de semanas de cada fase del sprint. El problema es uno comúnmente conocido como clasificación binaria (Vieira & Rodríguez, 2024), en el contexto del concepto de Machine Learning. Por lo que determinar la viabilidad económica de un proyecto utilizando herramientas de cómputo en la nube solo tiene dos posibles resultados, los cuales serían 'Viable' o 'No Viable', 'SI' o 'NO', 'Aceptado' o 'No Aceptado' o cualquier otro que designe dos estados.

La arquitectura de red neuronal elegida fue un perceptrón multicapa dentro del contexto de aprendizaje supervisado, con cinco neuronas en la capa de entrada, una única capa oculta con cinco neuronas y una capa de salida con una neurona. El propósito de la red consiste en convertir datos de entrada que no pueden separarse de forma lineal en representaciones abstractas que sí lo sean, a través de una estructura jerárquica de capas. Estas capas combinan funciones tanto lineales como no lineales. Entre las capas no lineales más utilizadas se encuentran las funciones de activación, tales como sigmoide logística, tangente hiperbólica (tanh), ReLU, ELU, Swish y Mish (Dubey, Singh, & Chaudhuri, 2022).

Por su parte las funciones de activación seleccionadas fueron ReLU para las entradas de la capa oculta y sigmoide para la entrada de la capa de salida. Para el problema de clasificación binaria la función de pérdida elegida fue entropía cruzada binaria debido a su naturaleza probabilística y su capacidad para optimizar el modelo de manera eficiente (Kim & Lim, 2023). La función principal de esta conformación es minimizar la diferencia entre las probabilidades predichas y las verdaderas, lo que resulta en una mejora de la precisión en problemas de clasificación binaria (Kumari & Srivastava, 2017).

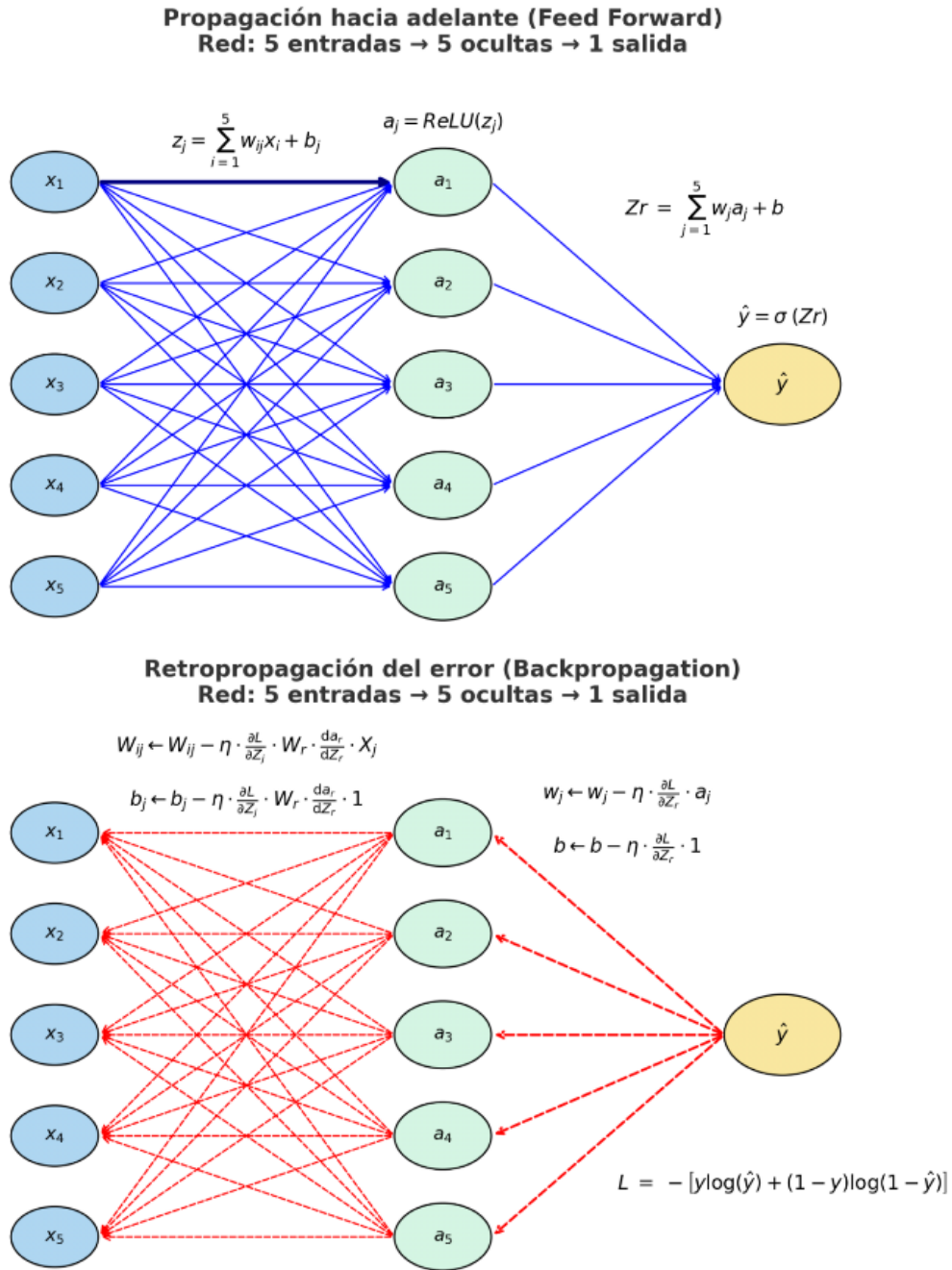
La función de optimización tiene como objetivo ajustar los parámetros del modelo para minimizar la diferencia entre las predicciones realizadas y los valores reales, es decir, disminuir el error o la pérdida (Abdulkadirov, Lyakhov, & Nagornov, 2023). Para lograr lo anterior, se utilizan distintas técnicas matemáticas, siendo el descenso de gradiente estocástico la elegida para esta tarea.

En el descenso de gradiente, el algoritmo sigue varios pasos esenciales (Tian, Zhang, & Zhang, 2023; Amari, 1993). En primer lugar, el modelo hace una predicción y calcula el error utilizando una función que mide la diferencia entre la predicción y el valor real. Luego, se calcula el gradiente de la función de pérdida en relación con los parámetros del modelo, lo que muestra cómo cambia la pérdida al modificar los parámetros. Después, los parámetros se ajustan en la dirección contraria al gradiente para reducir la pérdida. Este proceso se repite continuamente hasta que se llega a un punto donde la función de pérdida es mínima, lo que indica que el modelo ha aprendido de forma óptima. Al ser un problema en el que no se cuenta con una gran cantidad de datos, este algoritmo de optimización permite evaluar cada iteración de manera individual, lo que se ajusta al contexto de la investigación. El número de épocas se determinó en 100 y el batch size en uno, ya que el algoritmo del descenso del gradiente estocástico lo requiere. De esta manera se logró un entrenamiento extensivo y con modificaciones a los valores después de evaluar cada caso (Lin R., 2022).

En la Figura 1 se muestra el entrenamiento de una red diseñada para capturar la complejidad del problema de predicción, la cual primero calcula una salida con los datos y la compara con la respuesta real, y posteriormente ajusta pesos y sesgos mediante retropropagación para reducir el error en las siguientes iteraciones.

Figura 1.

Ejemplificación del proceso de entrenamiento de la red neuronal.



Fuente: Elaboración propia.

Existen cinco estadísticas utilizadas en el ámbito del aprendizaje automático para determinar el nivel de eficiencia de una red neuronal. Estas estadísticas son: precisión, precisión de la clase positiva, sensibilidad, especificidad y F1-Score, todas estas estadísticas se calculan a partir de los resultados mostrados en la matriz de confusión (Chicco & Jurman, 2020). La matriz de confusión se encarga de mostrar la cantidad de casos verdaderos positivos (VP), falsos positivos (FP), falsos negativos (FN) y verdaderos negativos (VN) (Liang, 2022).

La precisión es la encargada de medir el porcentaje de predicciones correctas sobre el total de predicciones realizadas y se calcula mediante la siguiente fórmula:

$$\text{Precisión} = \frac{VP+VN}{VP+FP+FN+VN} \quad (2)$$

Precisión positiva mide el porcentaje de predicciones realizadas de manera correcta sobre el total de predicciones realizadas por la red neuronal. Su fórmula es la siguiente:

$$\text{Precisión Positiva} = \frac{VP}{VP+FP} \quad (3)$$

La estadística de sensibilidad se encarga de medir la capacidad que tiene el modelo para detectar los casos positivos y se aplica la siguiente fórmula para obtener el valor:

$$\text{Sensibilidad} = \frac{VP}{VP+FN} \quad (4)$$

La especificidad mide la capacidad del modelo para detectar los casos negativos encontrados durante la ejecución de la predicción, el cálculo de sensibilidad se realiza aplicando la siguiente fórmula:

$$\text{Especificidad} = \frac{VN}{VN+FP} \quad (5)$$

El F1-Score es la media armónica entre el porcentaje de precisión positiva y el valor de la sensibilidad, se calcula de la siguiente manera:

$$F1 - Score = 2 \cdot \frac{\text{Precisión Positiva} \cdot \text{Sensibilidad}}{\text{Precisión Positiva} + \text{Sensibilidad}} \quad (6)$$

De acuerdo con Alzubaidi et al. (2021), existen diversas herramientas de software para desarrollar redes neuronales, por lo que se deben elegir las herramientas que mejor se adapten a las capacidades de los desarrolladores por su simplicidad. Las herramientas basadas en el lenguaje de programación Python se han posicionado como las mejores opciones para el desarrollo de redes neuronales (Raschka & Mirjalili, 2019). Para el desarrollo de la red neuronal se seleccionaron las librerías de *TensorFlow*® y *Keras*®, que por medio del lenguaje de programación *Python*® se integraron llevando a cabo el desarrollo de la estructura de la red, el entrenamiento y la ejecución de la misma.

Resultados

A continuación, se exponen los principales hallazgos derivados del procesamiento de los datos, así como las predicciones generadas por la red neuronal, las cuales fueron clasificadas en escenarios viables y no viables. Adicionalmente, se analizan los indicadores económicos relevantes que respaldan las predicciones y se comparan con métricas de referencia para validar la consistencia del modelo.

De los 13 casos reportados inicialmente se obtuvieron ocho proyectos con una TIR superior a la TMAR (20%), lo que representa una tendencia positiva hacia la viabilidad. En contraste, cinco proyectos no superaron el umbral, de los cuales tres registraron una TIR negativa, lo que indica un posible riesgo económico al implementar herramientas de cómputo en la nube. Al aplicar bootstrapping y generar un total de 2013 casos (13 originales y 2000 adicionales), se observó que el 54.2% de los proyectos resultaron viables (1092 casos), mientras que el 45.8% no lo fueron (921 casos). Este resultado refleja un balance significativo ($p < .001$) hacia la viabilidad.

Se calculó la distribución triangular para el valor de la TIR en los nuevos casos generados, en la cual se obtuvo un valor mínimo de -41.67, máximo de 61.7 y más frecuente de 24.2.

Previa verificación de la condición de normalidad ($p < .260$), se calculó el Coeficiente de Correlación de Pearson (ρ) mediante el uso de Python® y la librería Pandas para los datos generados. De acuerdo con Fiallos (2021), ρ es el índice numérico más utilizado para estimar una correlación, el cual mostró un valor de correlación lineal positiva y significativa de 0.93 entre los valores de la TIR y la cantidad de desarrolladores de software asignados al proyecto. De igual manera, se calculó el valor p utilizando la librería scipy de Python®, en la cual se obtuvo un valor $p < .001$ para la correlación.

En relación con el tamaño de los equipos, se identificó una tendencia clara:

- Con menos de 10 desarrolladores, ningún proyecto alcanzó viabilidad económica.
- A partir de 10 desarrolladores, la aceptación se incrementó considerablemente (alrededor del 87%).
- Con 14 o más desarrolladores, la viabilidad fue del 100% de los casos, lo que sugiere un punto de inflexión crítico en la rentabilidad de los proyectos.

Por otra parte, como ejercicio de validación se calculó la información mostrada anteriormente modificando el valor de la TMAR a 16% y 14%, obteniendo que al disminuir la TMAR al 16% aumenta los proyectos aceptados en un 21.97% y disminuirla al 14% aumenta en un 25.90% la aceptación de proyectos con respecto al 20% de la TMAR aceptada originalmente.

El proceso de entrenamiento de la red neuronal se ejecuta mediante una cantidad de iteraciones definidas por el número de casos y la cantidad de épocas, para este caso se ejecutaron 201,300 iteraciones, mediante las cuales se ajustan los pesos y los sesgos de la red para minimizar el error y obtener predicciones de calidad. En la Tabla 1 se observan los parámetros de entrada de la primera iteración, la cual será utilizada para ejemplificar el proceso, del entrenamiento de la red neuronal.

Tabla 1.*Valores de entrada para la primera iteración del entrenamiento de la red neuronal.*

<i>Descripción</i>	<i>Valor</i>	<i>Entrada en la red neuronal</i>
<i>Miembros por equipo de desarrollo</i>	10	X_1
<i>Cantidad de Sprint</i>	8	X_2
<i>Fase de diseño (Semanas)</i>	1	X_3
<i>Fase de desarrollo (Semanas)</i>	2	X_4
<i>Fase de control (Semanas)</i>	1	X_5

Fuente: Elaboración propia.

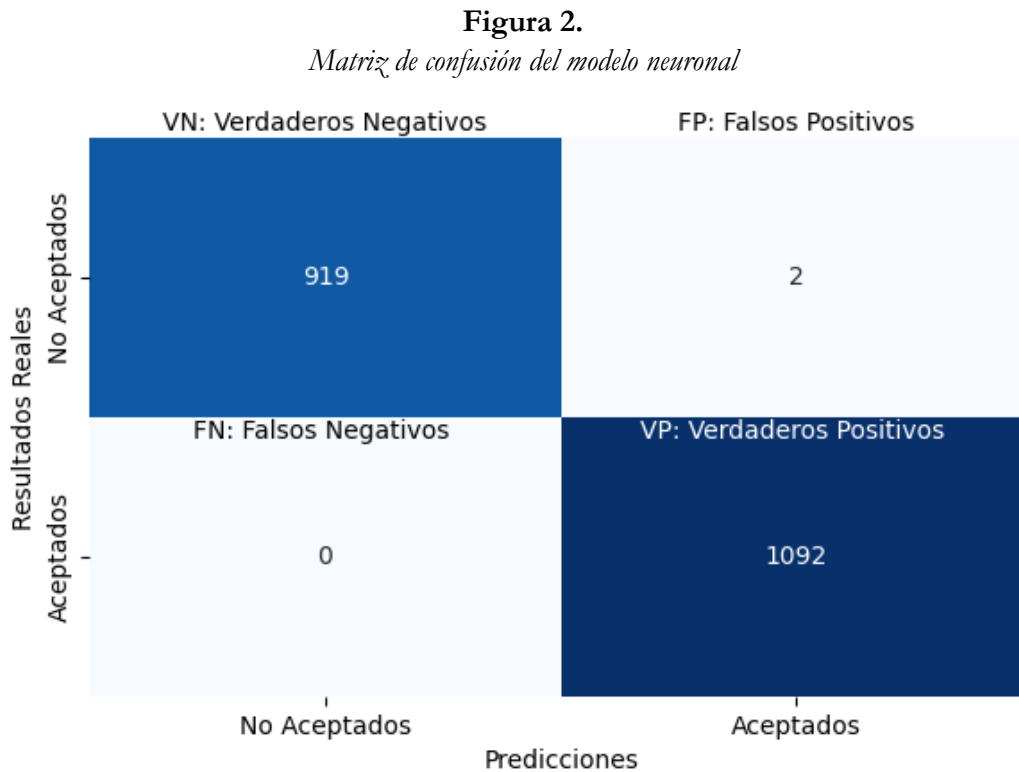
A continuación, se muestran los valores asignados a los pesos y los sesgos de cada capa de la red, antes y después del ajuste correspondiente al entrenamiento. Concluida la primera iteración, se prosigue con la segunda utilizando los valores del siguiente caso en la lista hasta completar los 2013 casos; posteriormente, se inicia una nueva iteración con el primer caso y así sucesivamente hasta terminar las 100 épocas.

Tabla 2.*Pesos y sesgos de la red antes y después de la retropropagación.*

	<i>Capa Entrada</i>							<i>Capa oculta</i>		
		W_1	W_2	W_3	W_4	W_5	B		W_6	b
<i>Inicialización aleatoria de pesos y sesgos (propagación hacia adelante)</i>	X_1	0.23	-0.12	0.45	0.38	0.01	-0.25	a_1	-0.08	-0.30
	X_2	-0.44	0.19	-0.07	0.32	-0.26	0.17	a_2	-0.42	-0.30
	X_3	0.48	0.21	0.03	0.27	-0.34	-0.48	a_3	0.24	-0.30
	X_4	-0.18	0.05	0.47	0.33	0.29	-0.18	a_4	-0.03	-0.30
	X_5	0.41	-0.09	0.14	0.49	0.22	0.14	a_5	0.15	-0.30
		W_1	W_2	W_3	W_4	W_5	B		W_6	b
<i>Ajuste de pesos y sesgos (propagación hacia atrás)</i>	X_1	0.23	-0.4417	0.4809	-0.1801	0.41	0.23	a_1	-0.08	-0.2996
	X_2	-0.12	0.1886	0.2108	0.0499	0.09	-0.12	a_2	-0.41965	-0.2996
	X_3	0.45	-0.0702	0.0301	0.4699	0.14	0.45	a_3	0.24211	-0.2996
	X_4	0.38	0.3197	0.2702	0.3299	0.49	0.38	a_4	-0.0269	-0.2996
	X_5	0.01	-0.2602	-0.3399	0.2899	0.22	0.01	a_5	0.15008	-0.2996
		W_1	W_2	W_3	W_4	W_5	B		W_6	b

Fuente: elaboración propia.

Para analizar la información del funcionamiento de la red neuronal se hizo uso de las estadísticas de precisión, precisión de la clase positiva, sensibilidad, especificidad y F1-Score las cuales se calcularon tomando los resultados en la matriz de confusión obteniendo los siguientes resultados.



Fuente: Elaboración propia.

Con respecto al desempeño del modelo neuronal, la matriz de confusión mostró una clasificación sobresaliente, con muy pocos falsos positivos y ningún falso negativo. Las métricas obtenidas mostraron una precisión global del 99.9%, una precisión de la clase positiva del 99.8%, sensibilidad de 1.0, especificidad del 99.8% y un F1-Score de 0.999, lo que garantiza no solo una excelente exactitud general, sino también una alta capacidad para la detección de casos positivos, minimizar errores de clasificación y mantener un equilibrio entre precisión y sensibilidad. En conjunto, estos hallazgos sugieren que la viabilidad económica del uso de tecnologías de cómputo en la nube depende del tamaño del equipo de desarrollo.

Discusión

Los resultados son de relevancia particular para la empresa bajo estudio, ya que responden directamente a la necesidad de optimizar el aprovechamiento de los desarrolladores de software mediante la adopción de tecnologías de cómputo en la nube, con el fin de reducir los periodos de inactividad y mejorar la rentabilidad de los proyectos. Esta alternativa fue evaluada para determinar su impacto económico, identificándose una correlación muy alta y significativa entre la cantidad de desarrolladores y la viabilidad del uso de dichas herramientas. Lo anterior coincide con los hallazgos de Garousi, Tarhan, Pfahl, Coşkunçay y Demirörs (2019), Zhang et al. (2015) y Jiarpakdee, Tantithamthavorn y Treude (2018), quienes evidencian la influencia de los factores críticos de éxito, como el tamaño del equipo y la gestión del conocimiento, en el desempeño de los proyectos de software. Estudios en sectores distintos al tecnológico, como el educativo (Salazar, Díaz, & Macías, 2023) y el económico (Ramírez-Sánchez, Valdez-Luna, & Gutiérrez-Pallares, 2023), reportan asociaciones similares entre el número de recursos y la eficiencia operativa.

Se determinó que la adopción de tecnologías de cómputo en la nube resulta económicamente válida a partir de equipos conformados por 10 desarrolladores y totalmente viable a partir de 14. Bajo este escenario, la implementación de la nube se considera una estrategia financieramente justificable para maximizar el aprovechamiento del capital humano y disminuir los costos asociados al tiempo improductivo. Sin embargo, en equipos más pequeños su aplicación no resultaría recomendable, ya que los beneficios no compensarían los costos de implementación. En estos casos, se observó que los equipos pequeños que mantienen ciertos periodos de inactividad pueden alcanzar mejores niveles de rentabilidad que aquellos que incurren en gastos adicionales para eliminar completamente los tiempos muertos, pues las estrategias de optimización también representan un costo operativo que debe evaluarse con base en la relación inversión y retorno.

Khayer et al. (2020) evidenciaron que la adopción del cómputo en la nube dentro de pequeñas y medianas empresas requiere un equilibrio entre los beneficios de escalabilidad y los costos de implementación, destacando que la rentabilidad depende de la capacidad de la organización para integrar dicha tecnología en función de su madurez operativa.

El análisis económico prospectivo también mostró que gran parte del costo total de los proyectos recae en los recursos humanos y tecnológicos empleados, por lo que es indispensable considerar todos los componentes que integran el proyecto al momento de estimar su rentabilidad (Fernández Macedo & Peyerl, 2022; Solís González, Guzmán Prince, Mediana Álvarez, & Galván Rodríguez, 2024). Si bien algunos autores sostienen que el tamaño del equipo incide directamente en la estructura de costos y en la eficiencia del proceso (Pendharkar & Rodger, 2009), otros, como Rodríguez et al. (2012), argumentan que esta relación no es necesariamente proporcional ni constante, pues la eficiencia depende más de la diversidad de funciones, la especialización del personal y la estructura de coordinación interna. Desde esta perspectiva, el tamaño del equipo se considera un factor determinante en la rentabilidad o la viabilidad económica para la empresa de estudio, ya que, al tratarse de una PyME dedicada al desarrollo de software ad hoc para cada cliente, la conformación e integración del equipo de

trabajo se convierten en elementos críticos que condicionan directamente el costo operativo y la eficiencia del proyecto.

En este estudio, la TMAR se fijó inicialmente en un 20%; sin embargo, los resultados mostraron que al reducirla al 16% se incrementa la cantidad de proyectos económicamente viables en un 21.97%. Este comportamiento refuerza la necesidad de analizar la TMAR de manera crítica, considerando variables macroeconómicas como el mercado, la duración del proyecto, la inflación y el riesgo asociado a la inversión (Sánchez Márquez, 2019; Orozco Álvarez, Hernández Islas, Cruz Vázquez, & Lira Martínez, 2024; Vizuete Montero, 2023). Dutta y Fan (2009) señalan que la TMAR óptima puede ubicarse incluso por debajo del costo de capital, ya que su función esencial es incentivar la inversión y equilibrar el esfuerzo de desarrollo con la rentabilidad esperada. De manera complementaria, Barry, Carlin, Crane y Graham (2024) destacan que las organizaciones aplican tasas de referencia flexibles o ajustables según la naturaleza del proyecto, las condiciones del mercado y el nivel de riesgo asumido. En conjunto, ambos estudios evidencian que una TMAR flexible permite realizar evaluaciones más precisas y adaptadas al contexto, evitando que la aplicación de umbrales fijos limite la identificación de oportunidades que podrían resultar financieramente viables bajo condiciones controladas de riesgo.

Asimismo, los resultados ponen de manifiesto el valor de la inteligencia artificial como herramienta de apoyo en la toma de decisiones financieras, al permitir modelar escenarios de rentabilidad y estimar la viabilidad de los proyectos con base en datos históricos. En esta línea, Park, Kim y Lee (2023) destacan que la integración de modelos de inteligencia artificial en la gestión financiera permite realizar estimaciones más precisas y optimizar la asignación de recursos. Por su parte, el uso de una red neuronal para la predicción de la viabilidad económica demostró ser pertinente, ya que el modelo alcanzó una precisión del 99.9%, lo que confirma su utilidad como instrumento de apoyo para la toma de decisiones estratégicas en entornos empresariales de base tecnológica.

A pesar del alto nivel de precisión obtenido por la red neuronal, es importante reconocer la posibilidad de sobreajuste derivada del tamaño limitado de la muestra original. Aunque el método de bootstrapping permitió ampliar el conjunto de datos a 2000 casos, estos mantienen la misma estructura estadística de los 13 proyectos iniciales, lo que pudo propiciar que el modelo aprendiera patrones específicos en lugar de generalizarlos. Por ello, la capacidad predictiva del clasificador debe interpretarse con precaución y ser verificada en futuras investigaciones que incluyan una base de datos más amplia y diversa. En trabajos posteriores, la implementación de técnicas de validación cruzada o el uso de regularizadores podría contribuir a reducir este sesgo y fortalecer la estabilidad y confiabilidad del modelo.

Lo anterior, coincide con lo expuesto por Minderer et al. (2021) y Carvajal Chávez (2024), quienes destacan que una elevada precisión en redes neuronales no implica necesariamente una adecuada capacidad de generalización, ya que esta depende tanto de la arquitectura del modelo como del volumen y la calidad de los datos empleados durante el entrenamiento. De forma

complementaria, Olisah et al. (2024) señalan que la confiabilidad de los resultados está directamente relacionada con la diversidad de los datos utilizados, por lo que un conjunto limitado puede favorecer el sobreajuste y producir métricas de desempeño infladas que no reflejan el comportamiento real del modelo en nuevos escenarios.

Cabe destacar que, aunque la elevada precisión del 99 % pueda generar dudas respecto a su capacidad de generalización, existen investigaciones en ámbitos médicos e industriales que reportan resultados similares y estadísticamente válidos. Un estudio orientado a la clasificación de componentes electrónicos en entornos industriales registró un 99.03 % de acierto (Hozyń, 2023). De manera complementaria, Chen y Lin (2021) demostraron que un modelo neuronal difuso puede predecir con alta exactitud la productividad de una planta manufacturera, destacando que en escenarios con datos homogéneos y bien estructurados estas cifras pueden ser realistas y no necesariamente reflejar sobreajuste. Sin embargo, los propios autores advierten que, cuando el volumen de datos es reducido, el modelo tiende a memorizar patrones específicos, por lo que recomiendan ampliar la muestra y diversificar los casos analizados. En este sentido, aunque valores de precisión elevados pueden considerarse aceptables en contextos de alta estabilidad y control de datos, en el presente estudio el tamaño limitado de la muestra inicial sigue representando una restricción importante para asumir tal condición .

Aunque el método de bootstrapping permitió ampliar considerablemente la cantidad de casos analizados, es importante reconocer que este procedimiento depende directamente de la muestra original. Por ello, los registros generados conservan la misma estructura y distribución estadística de los proyectos iniciales, lo que limita la diversidad de escenarios y restringe la validez externa de los resultados. Sin embargo, debe tomarse en cuenta que la empresa donde se desarrolló la investigación es una PyME dedicada al desarrollo de software a la medida, por lo que su volumen de proyectos es reducido y la conformación de una muestra más amplia requeriría un periodo prolongado de tiempo. Se recomienda que futuras investigaciones incorporen un mayor número de proyectos reales que permitan fortalecer la base de conocimiento y validar el desempeño del modelo bajo diferentes condiciones. En caso de no disponer de nueva información, resulta necesario ajustar el proceso de generación de casos mediante vectores de datos más heterogéneos, utilizando valores máximos y mínimos para crear secuencias ampliadas que incrementen la combinación de escenarios, manteniendo al mismo tiempo las condiciones observadas en la muestra original. De esta manera, será posible conservar la validez interna del análisis obtener una mayor solidez metodológica del modelo propuesto.

Es importante señalar que los hallazgos en este estudio no pueden replicarse de manera literal en otras organizaciones, ya que la investigación fue diseñada específicamente en función de las características, procesos y estructura operativa de la empresa analizada. No obstante, los procedimientos metodológicos aplicados pueden servir como referencia para otras empresas que busquen evaluar la viabilidad económica de soluciones tecnológicas orientadas a optimizar el tiempo muerto y mejorar la rentabilidad operativa. En este sentido, resulta fundamental considerar el tamaño del equipo de trabajo como un factor relevante en la estimación de costos y beneficios, puesto que, a mayor número de desarrolladores, mayor capacidad existe para

absorber costos fijos y generar márgenes de ganancia sostenidos. De igual forma, mantener una Tasa Mínima Aceptable de Rendimiento (TMAR) flexible permite adaptar las decisiones financieras a las condiciones particulares de cada organización, facilitando la identificación de escenarios económicamente viables y sostenibles en el tiempo. La incorporación de técnicas de inteligencia artificial en el proceso de análisis puede fortalecer la precisión de las estimaciones y agilizar la toma de decisiones estratégicas, al proporcionar una visión más objetiva y basada en datos sobre la rentabilidad y el desempeño esperado de las soluciones implementadas.

Finalmente, el código fuente, los scripts de simulación y los modelos empleados en este estudio se encuentran disponibles en un repositorio público de GitHub®, donde se incluyen las etapas base para reproducir los experimentos: <https://github.com/E-debug-lab/EvaluacionEconomicaIA>. Por motivos de confidencialidad y conforme al permiso otorgado por la empresa colaboradora, la versión pública del código presenta ciertas secciones ofuscadas; no obstante, conserva la lógica operativa y los procedimientos esenciales que permiten replicar el flujo metodológico.

La población analizada estuvo conformada por proyectos de desarrollo de software orientados al área de facturación, lo que permitió identificar comportamientos particulares en su estructura de costos y en la manera en que se distribuye el trabajo durante las distintas fases del proceso. En este tipo de proyectos, la etapa de diseño requiere un análisis detallado de reglas de negocio y reuniones presenciales con el cliente, lo que provoca periodos de inactividad en los que los desarrolladores permanecen asignados sin actividades productivas. Bajo el esquema tradicional estos tiempos muertos se consideran un costo fijo inevitable que reduce de manera directa la rentabilidad del proyecto y limita el margen de maniobra para optimizar los recursos.

Los resultados de esta investigación muestran que la incorporación de servicios de cómputo en la nube permite transformar esas horas improductivas en actividades facturables, lo que modifica de forma significativa el comportamiento económico respecto al modelo convencional. En los proyectos de facturación que siguen un enfoque tradicional la rentabilidad suele verse comprometida debido al impacto del tiempo muerto, mientras que la propuesta basada en la nube incrementa la TIR en escenarios donde los equipos cuentan con diez desarrolladores o más. Este cambio evidencia que, para este tipo de proyectos, el uso de tecnologías en la nube constituye una alternativa más eficiente para gestionar los costos asociados al desarrollo y mejorar los indicadores económicos sin alterar las características operativas propias del dominio.

En conjunto estos hallazgos permiten concluir que la integración de cómputo en la nube proporciona una vía para aprovechar los periodos de inactividad y fortalecer la rentabilidad, lo que convierte a esta propuesta en una opción económicamente justificable frente a los esquemas convencionales previamente utilizados en la empresa.

Contribuciones de los autores: Conceptualización: Eduardo Perez Carrillo, Diego Adiel Sandoval Chávez; Curación y análisis formal de datos: Eduardo Perez Carrillo; Investigación y metodología: Eduardo Perez Carrillo, Diego Adiel Sandoval Chávez; Validación: Eduardo Perez Carrillo, Diego Adiel Sandoval-Chávez; Visualización: Eduardo Perez Carrillo, Diego Adiel Sandoval-Chávez; Redacción del borrador original: Eduardo Perez Carrillo, Diego Adiel Sandoval-Chávez.

Financiamiento: Este estudio recibió apoyo de la SECIHTI mediante una beca nacional a estudiantes de posgrado.

Referencias

- Abdulkadriov, R., Lyakhov, P., & Nagornov, N. (2023). Survey of optimization algorithms in modern neural networks. *Mathematics*, 11(11), 2466. <https://doi.org/10.3390/math11112466>
- Almeida, F., Duarte Santos, J., & Augusto Monteiro, J. (2020). The challenges and opportunities in the digitalization of companies in a post-COVID-19 world. *IEEE Engineering Management Review*, 48(3), 97–103. <https://doi.org/10.1109/EMR.2020.3013206>
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). *Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions*. *Journal of Big Data*, 8(1), 53. <https://doi.org/10.1186/s40537-021-00444-8>
- Amari, S. I. (1993). Backpropagation and stochastic gradient descent method. *Neurocomputing*, 5(4–5), 185–196. [https://doi.org/10.1016/0925-2312\(93\)90006-O](https://doi.org/10.1016/0925-2312(93)90006-O)
- Barry, J. W., Carlin, B. I., Crane, A. D., & Graham, J. (2024). *The role of elevated hurdle rates*. National Bureau of Economic Research (NBER) Working Paper No. 32283. <https://doi.org/10.3386/w32283>
- Bruch, C. (2022). Applying the rescaling bootstrap under imputation for a multistage sampling design. *Computational Statistics*, 37, 1461–1494. <https://doi.org/10.1007/s00180-021-01164-6>
- Camacho Angulo, M., Banchón Jiménez, S., Barcia Villacreses, K. F., & Allauca Amaguaya, M. (2023). Aplicación de la teoría de restricciones en un proceso productivo con enfoque a la industria 4.0. *Reciamuc*, 7(2), 281–304. [https://doi.org/10.26820/reciamuc/7.\(2\).abril.2023.281-304](https://doi.org/10.26820/reciamuc/7.(2).abril.2023.281-304)
- Carvajal Chávez, C. A. (2024). Uso de técnicas como la regresión y redes neuronales para anticipar el rendimiento del maíz. *Recimundo*, 8(4), 126–135. [https://doi.org/10.26820/recimundo/8.\(4\).diciembre.2024.126-135](https://doi.org/10.26820/recimundo/8.(4).diciembre.2024.126-135)
- Chen, C.-H., & Lin, Y.-H. (2021). *Enhancing the accuracy and precision of forecasting the productivity of a factory: A fuzzyfied feedforward neural network approach*. *Complex & Intelligent Systems*, 7(6), 3023–3038. <https://doi.org/10.1007/s40747-021-00416-8>
- Cheshmehzangi, A. (2021). From transitions to transformation: A brief review of the potential impacts of COVID-19 on boosting digitization, digitalization, and systems thinking in the built environment. *Journal of Building Construction and Planning Research*, 9(1), 26–39. <https://doi.org/10.4236/jbcpr.2021.91003>

- Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21, 1–13. <https://doi.org/10.1186/s12864-019-6413-7>
- Chicco, D., Warrens, M. J., & Jurman, G. (2021). The Matthews correlation coefficient (MCC) is more informative than Cohen's kappa and Brier score in binary classification assessment. *IEEE Access*, 9, 78368–78381. <https://doi.org/10.1109/ACCESS.2021.3084050>
- Dastres, R., & Soori, M. (2021). Artificial neural network systems. *International Journal of Imaging and Robotics (IJIR)*, 21(2), 13–25. https://www.researchgate.net/publication/350486076_Artificial_Neural_Network_Systems
- de Lucas Ancillo, A., & Gavrila, S. G. (2023). The impact of research and development on entrepreneurship, innovation, digitization and digital transformation. *Journal of Business Research*, 157, 113566. <https://doi.org/10.1016/j.jbusres.2022.113566>
- Díaz-Vega, M., & Ospina-Ospina, M. (2023). *Conceptos económicos de innovación en la valoración tecnológica de MiPymes. Informador Técnico*, 87(2), 177–194. <https://doi.org/10.23850/22565035.5233>
- Dubey, S. R., Singh, S. K., & Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*, 503, 92–108. <https://doi.org/10.1016/j.neucom.2022.06.111>
- Dutta, S., & Fan, Q. (2009). Hurdle rates and project development efforts. *The Accounting Review*, 84(2), 405–432. <https://doi.org/10.2308/accr.2009.84.2.405>
- Etim, U. E., Milke, V., & Luca, C. (2025). Ensemble of neural networks to forecast stock price by analysis of three short timeframes. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence – Volume 3: ICAART* (pp. 813–820). SciTePress. <https://doi.org/10.5220/0013183600003890>
- Fernandes Macedo, S., & Peyerl, D. (2022). Prospects and economic feasibility analysis of wind and solar photovoltaic hybrid systems for hydrogen production and storage: A case study of the Brazilian electric power sector. *International Journal of Hydrogen Energy*, 47(19), 10460-10473. <https://doi.org/10.1016/j.ijhydene.2022.01.133>
- Fernández, A. L. (2022). *Introducción a la evaluación de proyectos de inversión: el VAN y la TIR*. (U. N. Matanza, Ed.) *Cálculo financiero de las operaciones simples y complejas*, pp. 367–394. <https://repositoriocyt.unlam.edu.ar/handle/123456789/2115>
- Fiallos, G. (2021). La correlación de Pearson y el proceso de regresión por el método de mínimos cuadrados. *Ciencia Latina Revista Científica Multidisciplinar*, 5(3), 2491–2509. https://doi.org/10.37811/cl_rcm.v5i3.466
- Fonseca Varela, M., & Chalita Tovar, L. E. (2021). Evaluación financiera de producción de agave y mezcal: caso de estudio Caltepec, Puebla. *Revista Mexicana de Ciencias Agrícolas*, 12(2), 263–273. <https://doi.org/10.29312/remexca.v12i2.2583>
- García, R. (2021). El perceptrón: una red neuronal artificial para clasificar datos. *Revista de Investigación en Modelos Matemáticos Aplicados a la Gestión y la Economía*, 8(1), 1–14. <https://www.economicas.uba.ar/wp-content/uploads/2016/04/Garcia-Roberto-1.pdf>

- Garousi, V., Tarhan, A., Pfahl, D., Coşkunçay, A., & Demirörs, O. (2019). Correlation of critical success factors with success of software projects: An empirical investigation. *Software Quality Journal*, 27(1), 429–493. <https://doi.org/10.1007/s11219-018-9419-5>
- Gupta, A., Rawal, A., & Barge, Y. (2021). Comparative study of different SDLC models. *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, 9(11), 73–80. <https://doi.org/10.22214/ijraset.2021.38736>
- Hożyń, S. (2023). *Convolutional neural networks for classifying electronic components in industrial applications*. *Energies*, 16(2), 887. <https://doi.org/10.3390/en16020887>
- Jiarpakdee, J., Tantithamthavorn, C., & Treude, C. (2018). *AutoSpearman: Automatically mitigating correlated metrics for interpreting defect models* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.1806.09791>
- Khayer, A., Talukder, M. S., Bao, Y., & Hossain, M. N. (2020). *Cloud computing adoption and its impact on SMEs' performance for cloud-supported operations: A dual-stage analytical approach*. *Technology in Society*, 60, 101225. <https://doi.org/10.1016/j.techsoc.2019.101225>
- Kim, J., & Lim, H. S. (2023). Neural network with binary cross entropy for antenna selection in massive MIMO systems: Convolutional neural network versus fully connected network. *IEEE Access*, 11, 111410–111421. <https://doi.org/10.1109/ACCESS.2023.3322679>
- Kumari, R., & Srivastava, S. K. (2017). Machine learning: A review on binary classification. *International Journal of Computer Applications*, 160(7), 11–15. <https://doi.org/10.5120/ijca2017913083>
- Liang, J. (2022). *Confusion matrix: Machine learning*. POGIL Activity Clearinghouse, 3(4). <https://pac.pogil.org/index.php/pac/article/view/304>
- Lin, J. (2023). A comparative study on the application of NPV and IRR in financial market investment decision. *Academic Journal of Business & Management*, 5(4), 51–54. <https://doi.org/10.25236/AJBM.2023.050409>
- Lin, R. (2022). Analysis on the selection of the appropriate batch size in CNN neural network. In *Proceedings of the 2022 International Conference on Machine Learning and Knowledge Engineering (MLKE)* (pp. 106–109). IEEE. <https://doi.org/10.1109/MLKE55170.2022.00026>
- López, A., Niembro, A., & Ramos, D. (2014). Latin America's competitive position in knowledge-intensive services trade. *CEPAL Review*, 113, 21–39. <https://doi.org/10.18356/30e45128-en>
- Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., Tran, D., & Lucic, M. (2021). Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems*, 34, 15682–15694. <https://doi.org/10.48550/arXiv.2106.07998>
- Morera-Carballo, M. (2022). Los sistemas de información gerencial y su evolución hacia la cuarta revolución industrial. *Revista Nacional de Administración*, 13(1), 95–103. <http://dx.doi.org/10.22458/rna.v13i1.4236>
- Olisah, C. C., Smith, L., Smith, M., Lawrence, M. O., & Ojukwu, O. (2024). *Corn yield prediction model with deep neural networks for smallholder farmer decision support system* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2401.03768>

- Omole, O., & Enke, D. (2024). Deep learning for Bitcoin price direction prediction: Models and trading strategies empirically compared. *Financial Innovation*, 10(1), Article 117. <https://doi.org/10.1186/s40854-024-00643-1>
- Orozco Álvarez, C., Hernández Islas, S., Cruz Vázquez, M. N., & Lira Martínez, M. M. (2024). Plan de negocio para la producción de alginato de sodio a partir del aprovechamiento del alga “Sargassum”, en las playas de Quintana Roo, México. *LATAM: Revista Latinoamericana de Ciencias Sociales y Humanidades*, 5(2), 1093–1112. <https://doi.org/10.56712/latam.v5i2.1938>
- Quiñonez, É., Monserrate, R., & López, S. (2018). *La viabilidad de un proyecto, el valor actual neto (VAN) y la tasa interna de retorno (TIR)*. *Pro Sciences: Revista de Producción, Ciencias e Investigación*, 2(17), 9–15. <https://journalprosciences.com/index.php/ps/article/view/165>
- Pendharkar, P. C., & Rodger, J. A. (2009). The relationship between software development team size and software development cost. *Communications of the ACM*, 52(1), 141–144. <https://doi.org/10.1145/1435417.1435449>
- Ramírez-Sánchez, M. Y., Valdez-Luna, G. G., & Gutiérrez-Pallares, E. (2023). Prospectiva para la exportación de la fresa mexicana con un modelo de regresión lineal múltiple. *Revista Espacios*, 44(07). <https://doi.org/10.48082/espacios-a23v44n07p03>
- Raschka, S., & Mirjalili, V. (2019). *Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2* (3rd ed.). Packt Publishing Ltd.
- Rebollar-Rebollar, S., Posadas, R., Rebollar-Rebollar, E., Hernández, J., & González, F. (2020). Aportes a indicadores de evaluación privada. *Revista Mexicana de Agronegocios*, 19(38), 1187–1196. <https://doi.org/10.22004/ag.econ.303885>
- Rivas-Tovar, L. A. (2025). *Normas APA 7ª edición: Estructura, citas y referencias* (12ª ed.). Instituto Politécnico Nacional. https://www.researchgate.net/publication/357046089_NORMAS_APA_7_EDICION_ESTRUCTURA_CITAS_Y_REFERENCIAS
- Rodríguez, D., Sicilia, M. A., García, E., & Harrison, R. (2012). Empirical findings on team size and productivity in software development. *Journal of Systems and Software*, 85(3), 562–570. <https://doi.org/10.1016/j.jss.2011.09.009>
- Ross, S. A., Westerfield, R. W., & Jordan, B. D. (2018). *Fundamentos de finanzas corporativas* (11.ª ed.) [Traducción al español]. México: McGraw-Hill/Interamericana. ISBN 978-1-4562-6177-1.
- Salazar, J. R., Díaz, M., & Macías, R. (2023). El impacto de la educación superior en la competitividad económica de México. *Revista Espacios*, 44(09), Article 02. <https://doi.org/10.48082/espacios-a23v44n09p02>
- Schwab, K. (2015). *The Fourth Industrial Revolution: What it means and how to respond*. *Foreign Affairs*. <https://doi.org/10.4337/9781802208818.00008>
- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum – The rules of the game*. Scrum.org. <https://scrumguides.org/>
- Solís González, B. A., Guzmán Prince, I., Mediana Álvarez, M. Á., & Galván Rodríguez, D. G. (2024). Impacto de la rotación de personal en los costos y la rentabilidad empresarial. *Ciencia Latina Revista Científica Multidisciplinar*, 8(6), 6615–6627. https://doi.org/10.37811/cl_rcm.v8i6.15347

Investigación Administrativa, 2025, vol. 54, núm. 136, julio-diciembre (publicación continua)

- Sutrisno, S., Khairina, N., Syah, R. B., & Eftekhari-Zadeh, E. (2023). Improved artificial neural network with high precision for predicting burnout among managers and employees of start-ups during COVID-19 pandemic. *Electronics*, 12(5), Article 1109. <https://doi.org/10.3390/electronics12051109>
- Tharwat, A. (2021). Classification assessment methods. *Applied Computing and Informatics*, 17(1), 168–192. <https://doi.org/10.1016/j.aci.2018.08.003>
- Tian, Y., Zhang, Y., & Zhang, H. (2023). Recent advances in stochastic gradient descent in deep learning. *Mathematics*, 11(3), Article 682. <https://doi.org/10.3390/math11030682>
- Tung, C.-T., Kao, M.-Y., & Hu, C. (2022). Neural network-based modeling with high accuracy and potential model speed. *IEEE Transactions on Electron Devices*, 69(11), 6476–6479. <https://doi.org/10.1109/TED.2022.3208514>
- Valpadasu, H., Rajalakshmi, P., & Valli, S. (2020). Scrum: An effective software development agile tool. *IOP Conference Series: Materials Science and Engineering*, 981(2), 022060. <https://doi.org/10.1088/1757-899X/981/2/022060>
- Vieira, P. M., & Rodrigues, F. (2024). An automated approach for binary classification on imbalanced data. *Knowledge and Information Systems*, 66, 2747–2767. <https://doi.org/10.1007/s10115-023-02046-7>
- Vizuite Montero, M. O. (2023). Factibilidad económica para elaborar y comercializar muebles artesanales a base de neumáticos fuera de uso. *LATAM: Revista Latinoamericana de Ciencias Sociales y Humanidades*, 4(2), 259–272. <https://doi.org/10.56712/latam.v4i2.609>
- Walczak, S. (2018). Artificial neural networks. In M. Khosrow-Pour (Ed.), *Encyclopedia of information science and technology* (4th ed., pp. 120–131). IGI Global. <https://doi.org/10.4018/978-1-5225-2255-3.ch009>
- Zhang, W., Ma, Z. Y., Zhang, W. G., Lu, Q. L., & Nie, X. B. (2015). Correlation Analysis of Software Defects Density and Metrics. *Applied Mechanics and Materials*, 713–715, 2225–2228. <https://doi.org/10.4028/www.scientific.net/AMM.713-715.2225>