

Simulación del proceso de recarga para Aguas Subterráneas utilizando Redes Neuronales Artificiales como Método de Aproximación en el Acuífero Las Sierras, Nicaragua.

Simulation of the recharge process for groundwater using Artificial Neural Networks as an Approximation Method in the Las Sierras Aquifer, Nicaragua.

Chevez, Carlos R.; Pinell, Freydell; Mejía Quiroz, Álvaro Antonio

 Carlos R. Chevez

carlos.chevez@ineter.gob.ni

Instituto Nicaragüense de Estudios Territoriales, Managua, Nicaragua. Universidad Nacional Autónoma de Nicaragua, Managua, Nicaragua, Nicaragua

 Freydell Pinell

freydell.pinell@ineter.gob.ni

Instituto Nicaragüense de Estudios Territoriales, Managua, Nicaragua. Universidad Nacional Autónoma de Nicaragua, Managua, Nicaragua, Nicaragua

 Álvaro Antonio Mejía Quiroz

amejia@unan.edu.ni

Universidad Nacional Autónoma de Nicaragua, Managua, Nicaragua, Nicaragua

Revista Torreón Universitario

Universidad Nacional Autónoma de Nicaragua-Managua, Nicaragua

ISSN: 2410-5708

ISSN-e: 2313-7215

Periodicidad: Cuatrimestral

vol. 12, núm. 33, 2023

revis.torreon.faremc@unan.edu.ni

Recepción: 30 Junio 2022

Aprobación: 13 Diciembre 2022

URL: <http://portal.amelica.org/ameli/journal/387/3873792010/>

DOI: <https://doi.org/10.5377/rtu.v12i33.15896>

El autor o los autores de los artículos, ensayos o investigaciones conceden a la Universidad Nacional Autónoma de Nicaragua, Managua (UNAN-Managua) los derechos de edición (copyright) del trabajo enviado, por consiguiente la Universidad cuenta con el derecho exclusivo para publicar el artículo durante el periodo completo de los derechos de autor.



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

Resumen: El conocimiento de la funcionalidad de un sistema hidrogeológico es de vital importancia para su gestión y conservación. Una de las variables y entrada principal que alimenta al sistema es la Recarga Potencial (Rp) producto de la precipitación (P). La finalidad del presente trabajo es el diseño de un modelo regresor no lineal usando Redes Neuronales Artificiales (RNA). Para lo anterior, se utilizó los datos recolectados por el INETER para estimar la Rp a través de las variables de entrada como: precipitación (P), texturas de suelo y demás variables ambientales en el Acuífero ubicado en Managua. Con la información recopilada, se procedió a la exploración de datos o 'Data mining' por medio de la estadística descriptiva, la cual permite presentar, interpretar y analizar los datos de forma comprensiva. Utilizando el lenguaje de programación Python (Rossum, 1991) y el entorno de trabajo JupyterLab, se procedió a desarrollar los elementos de la RNA a través de la librería Scikit -Learn o mejor conocida como Sklearn (Cournapeau, 2010). Posterior a las iteraciones y arreglos de los Hiperparámetros de la RNA, se logró un mejor ajuste utilizando la función de coste o 'cost function', la cual determina el error entre el valor estimado y el observado. Finalmente, se indican las configuraciones finales del modelo regresor para cada una de las texturas en la zona.

Palabras clave: Hidrogeología, Redes Neuronales Artificiales (RNA), Aprendizaje Automatizado (AA).

Abstract: The knowledge of hydrogeological system functionality, is a vital importance for its management and sustainable development. One of the variables and main input it feeds this system is Recharge (R) product of precipitation (P). The purpose of this study is desing a No Lineal Regresor Model using Artificial Neural Networks (ANN). For the above, with INETER data collected, it was estimate R using input variables for instance: precipitation (P), soil textures and other known environmental variables in Managua Aquifer. With the information collected, data exploration or 'Data mining' was carried out through descriptive statistics, which allows presenting, interpreting and analyzing the data in a comprehensive way. Using the Python programming language

(Rossum, 1991) and the JupyterLab work environment, the ANN elements were developed through the Scikit-Learn library or better known as Sklearn (Cournapeau, 2010). After the iterations and settings of hyperparameters, a better fit will be improved using the cost function, which determines the error between the estimated value and the observed value, in order to optimize the parameters of ANN. At the end, the final configurations of ANN are indicated for each soil texture.

Keywords: Hydrogeology, Artificial Neural Networks (ANN), Machine Learning (ML).

1. INTRODUCCIÓN

Los avances en el modelado informático, la potencia informática y el procesamiento de la información, han dado como resultado herramientas prácticas y mejoradas para comprender mejor los sistemas naturales altamente complejos. Se ha centrado una gran cantidad de trabajo en la aplicabilidad de los métodos de Aprendizaje Automático (AA) en la ciencia de la hidrología (Shortridge J.E, 2019); e hidrogeología (Tao y otros, 2022) sin embargo, no se utiliza una técnica única, ya que los datos y el escenario disponible determinarán el método más adecuado para un problema en cuestión (S. Sahoo y otros, 2017). Los enfoques basados en el AA y en el modelado hidrológico son un área importante de investigación. Estos modelos son más adecuados para estudios de recursos hídricos que los modelos físicos (Kenda K & Klemen, 2019). El método más común para la predicción de los recursos de agua subterránea usando técnicas de Aprendizaje Automático (AA) es la Red Neuronal Artificial (RNA). Los métodos de AA reconocen patrones ocultos en datos históricos y luego aplica estos patrones para predecir escenarios futuros. (Pham y otros, 2022)

En Nicaragua, la principal fuente de abastecimiento de agua potable es de origen subterráneo, representando el 70% (FAO, 2013). No obstante, son unas de las fuentes poco estudiadas en Nicaragua y en Centroamérica; por ello se ha desarrollado a través de los años nuevas tecnologías de investigación que promuevan una adecuada gestión de los recursos hídricos, contribuyendo a mejorar el acceso de agua para la población. En los últimos años se han desarrollado estudios enfocados al cálculo de las recargas subterráneas (INETER, 2020) y de manera detallada las características hidrogeológicas del acuífero Las Sierras (INETER, 2020), con el propósito de obtener un dato aproximado de la recarga subterránea.

La importancia del presente documento radica en el diseño de un modelo regresor no lineal de Aprendizaje Automático (AA) que estime o simule Recarga Potencial (Rp), a partir de los datos histórico de precipitación, información de campo, levantamiento geológico y pruebas de infiltración dentro del acuífero en estudio. En este sentido, no existe bibliografía nacional dedicada al uso del AA en la hidrología e hidrogeología. Inicialmente se pretende preparar la RNA usando las variables de entrada (inputs), generación de las capas ocultas y outputs. Posteriormente, se realizará el 'backpropagacion' que básicamente es la sección de entrenamiento de la red usando set de entrenamiento. Finalmente se obtiene la configuración, la cual brindará los outputs estimados con el menor error respecto a los datos observados o de prueba.

2. MATERIALES Y MÉTODOS

2.1 Área de estudio

El acuífero de Las Sierras se encuentra ubicado en la ciudad de Managua, con un extensión 568.51 km^2 cubriendo en los municipios de Ticuantepe, Masaya, La Concepción, El Crucero, San Marcos, Nindirí,

Managua, Ciudad Sandino y Tipitapa. (Figura 1). Desde el punto de vista geomorfológico, se localiza en el borde centro occidental de una estructura regional llamada Depresión de Nicaragua, en la cual yace y aflora material litológico predominantemente volcánico terciario y cuaternario (INETER, Mapa de las Provincias Geológicas de Nicaragua, 2004) .

Según la metodología de Köppen modificado (Köppen, 1918) , en Managua existe el sub-tipo climático dominante: Awo (w) igw, Cálido Sub- húmedo de menor humedad, con estación canicular. El registro histórico 1971-2010, indica que la zona presenta un período lluvioso muy bien definido de mayo a octubre. En el transcurso de este, se presentan 2 máximos de precipitación, en junio con 164.87 mm y en septiembre con 220.53 mm de promedio mensual. A partir de noviembre se presenta la estación seca, donde se observa que las precipitaciones disminuyen significativamente con valores de 53.45 mm en noviembre a 2.93 mm en abril, siendo este mes el más seco. El comportamiento de la temperatura oscila entre el valor máximo de 29.04°C en abril y el mínimo valor de 25.96°C diciembre. A partir de junio (27.09°C), disminuye, prolongándose hasta diciembre (25.96°C). Esta disminución de la temperatura, coincide con la estación invernal de los países situados en el Hemisferio Norte y con las incursiones de masas de aire frío de procedencia polar, propias de la época. (INETER, Atlas Climáticos. Período 1971 - 2010, 2022)

Hidrogeológicamente, el acuífero es libre aunque se han registrado condiciones localizadas de semi-confinamiento debido a la presencia de lentes limo-arcillosos y de flujos de lava densas con fracturamiento bajo. (Losilla y otros, 2001) . Las principales unidades geológicas son las Formación La Sierra (TpQs) y los materiales del Cuaternario Aluvial (Qal). (INAA/JICA, 1993) La recarga natural se origina por agua de lluvia que se infiltra hasta el acuífero. Dado que la precipitación es directamente proporcional a la elevación, en general se considera que esta aumenta con esa misma tendencia, mayor en la zona alta (sur) y menor en la zona baja (norte); sin embargo estudios recientes señalan que también existe recarga a cotas intermedias, esta depende también del coeficiente de infiltración de las unidades geológicas, de los suelos y uso de la tierra. (Losilla y otros, 2001)

Los suelos son de origen volcánico y las texturas de suelo que predominan en la zona son: Franco arenoso, Franco Arcilloso, Arenoso y el área impermeabilizada producto del desarrollo urbano. (INETER, 2021)

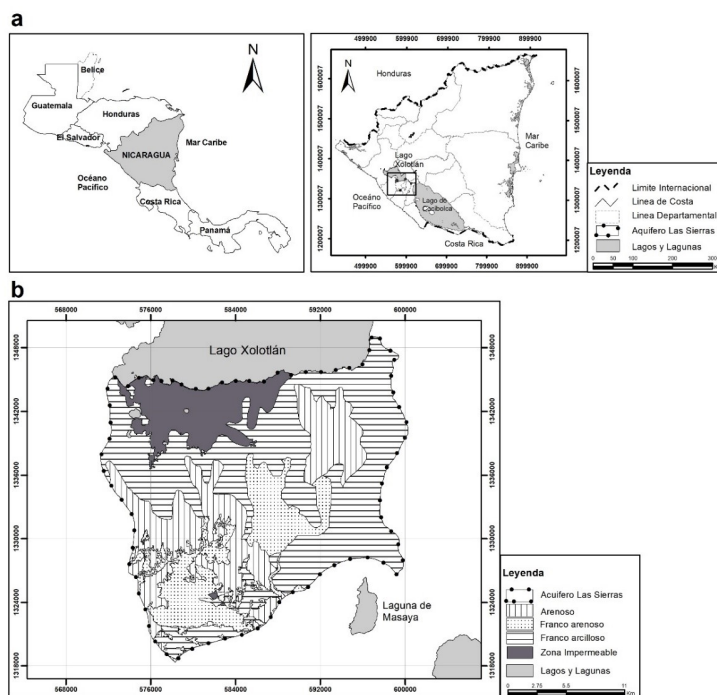


FIGURA 1

a: Localización del Acuífero, b: Textura de suelo en el Acuífero La Sierras.

Fuente: Elaboración Propia

2.2 Datos

La selección de variables de entrada es crucial para el desarrollo de modelos basados en datos y es particularmente relevante en modelado de recursos hídricos. (Quilty y otros, 2016) (Galelli y otros, 62, 33–51) La base de datos se compone de 576 observaciones y 33 variables ($M > N$), dando un total de 19, 000 datos. El origen de estos son principalmente son las observaciones del INETER. Las variables se categorizan en 3 grupos:

- Tiempo: Comprende un registro de observaciones del 2005 hasta 2020 (15 años), con una resolución mensual
- Características del suelo: capacidad de infiltración, capacidad de campo, punto de marchitez, densidad del suelo, pendiente, tipo de vegetación y profundidad de raíces.
- Variables ambientales: precipitación, temperatura del aire, precipitación infiltrada, escorrentía superficial, ETP, humedad del suelo, recarga potencial y retención de lluvia.

Según el modelo conceptual, en la zona de estudio existen 03 texturas de suelo (INETER, 2021), las cuales están asociadas a distintos valores de infiltración y por consiguiente R_p ; por lo tanto, los datos se han dividido en 03 grupos: S01, S02 y S03 los cuales corresponde a la textura de suelo arenoso, franco arenoso, franco arcilloso respectivamente. Se realizó una RNA para cada textura de suelo. Respecto a las variables de precipitación, temperatura y ETP se utilizaron datos mensuales de las estaciones ‘Aeropuerto’, ‘Salvador Allende’, ‘INETER’ y ‘Casa Colorada’. Para la estimación de la ETP y R_p se utilizó los métodos de Thornthwaite (Thornthwaite, 1948) y balance hídrico de suelo (Schosinsky & Losilla, 2006) respectivamente.

2.3 Diseño de la RNA

La Red Neurona Artificial (RNA)

Una red neuronal artificial (RNA) es un sistema de procesamiento de información distribuido, donde el aprendizaje está basado en los procesos neuronales. Una RNA aprende, memoriza y divulga las diversas relaciones encontradas en los datos. La unidad elemental de las RNA es la Neurona Artificial (NA), que es una abstracción matemática simplificada del comportamiento de una Neurona Biológica (NB). (Corea, 2014). Las RNA están compuestas de varias NA agrupadas en capas y conectadas entre si con el fin de resolver un problema.

En la Figura 2 (b) es la NA, que al igual que un árbol de dendritas de la NB, la NA recibe un conjunto de señales mediante las entradas (x_n). Cada entrada es amplificada o reducida por su correspondiente valor de peso (w_{nj}), de tal forma que el producto del binomio "entrada" por "peso" ($x_n * w_{nj}$) son sumados en conjunto con el valor de umbral llamado sesgo (b_j) el cual es multiplicado por el peso del sesgo (w_b). Este producto es transferido a la función de activación "f", que corresponde simbólicamente con el núcleo de la NB. Esta función procesa y genera la salida de la NA (Reed & Marks, 1999) (Cortez y otros, 2002). En la Ecuación 1 se representa el modelo matemático de la RNA:

$$y = f \left(\sum_{j=1}^n w_j x_j + w_b b_j \right)$$

ECUACIÓN 1
Modelo Matemático de una RNA

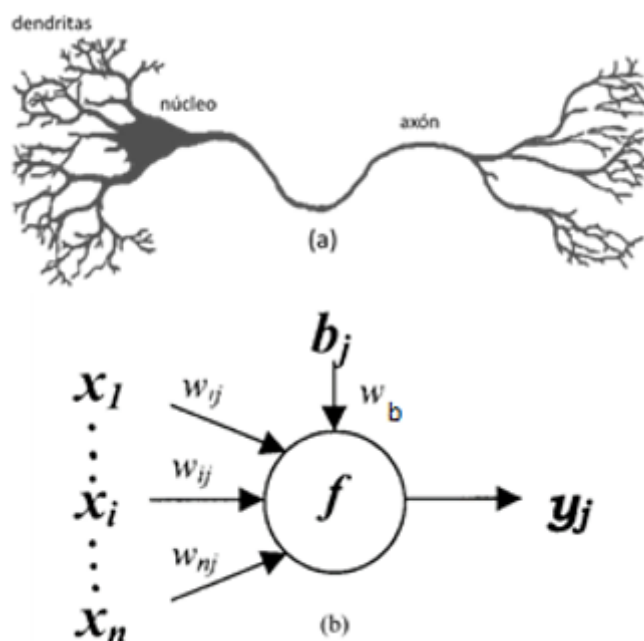


FIGURA 2
 Representación de una NB y una NA.
 Fuente: Adaptada desde Reed & Marks, 1998 y ASCE, 2000.

Función de Activación (FDA)

La función de activación (FDA) controla en gran medida la información que se propaga desde una capa a la siguiente (forward propagation). El objetivo principal de FDA es agregar componentes no lineales a la red, por lo que una RNA sin FDA está muy limitada para predecir. La FDA convierte el valor neto de entrada a la neurona (sumatoria de pesos y bias), en un nuevo valor (y). Es gracias a combinar FDA con múltiples capas, los modelos de RNA son capaces de aprender relaciones no lineales.

En este estudio, se utilizó la función ReLu (Rectified Linear Unit) o Unidad Lineal Rectificada. El uso de esta función se remonta a varias décadas, pero fue hasta 2010 (Glorot y otros, 2010) donde se demostró era mucho mejor en la mayoría de los casos que la mayoría de la FDA que se utilizaban en el momento. La razón de esto, reside en un costo computacional muy bajo y el comportamiento de su derivada (gradiente) es muy simple: cero (0) para los valores negativos y uno (1) para los valores positivos. ReLu no está acotada para los números positivos (Figura 3), lo cual da un gradiente constante, generando un aprendizaje más rápido.

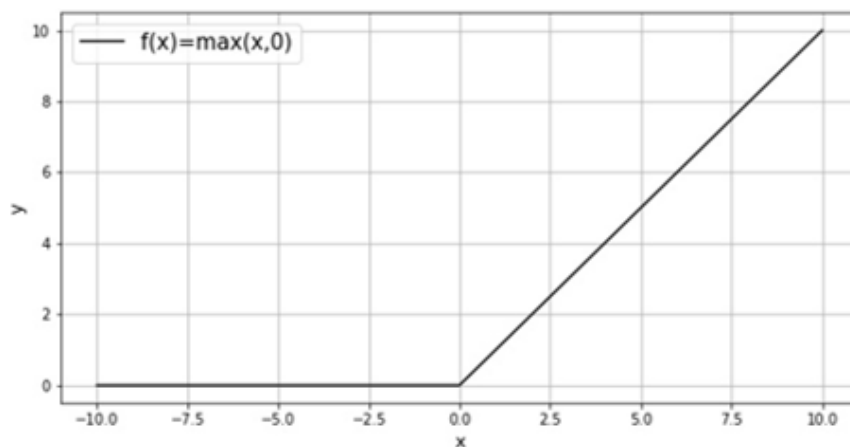


FIGURA 3
Función de activación ReLu.

Fuente: Elaboración propia

Preprocesado

Esta etapa consiste en dar un tratamiento a los datos previo al cálculo. Cabe destacar que la RNA será entrenada (training) con el 80% y la fase de prueba (test) con el 20%, los que representan aproximadamente a 12 y 3 años respectivamente. El preprocesado consiste en normalizar los datos de entrenamiento siendo este proceso necesario para la convergencia de los modelos de RNA y mejorar su rendimiento (Corea, 2014). Los predictores son numéricos y se normalizaron. Todos estos procesos se realizaron con el paquete 'StandardScaler' de la librería 'Sklearn'. Este paquete resta a cada valor de un atributo el valor medio de la columna en la que se encuentra (centrado) y finalmente divide cada predictor entre su desviación típica (normalizado), de esta forma los datos pasan a tener una distribución normal.

Hiperparámetros

Los Hiperparámetros de un modelo, son los valores de las configuraciones utilizadas durante el proceso de entrenamiento. El valor óptimo de un hiperparámetro no se puede conocer a priori para un problema dado, por lo que se tiene que utilizar valores iniciales aleatorios, los cuales son ajustados mediante prueba y error posteriormente. En RNA los hiperparámetros son: número y tamaño de capas ocultas (hidden layer), la tasa de entrenamiento (learning rate) y el parámetro de penalización (alpha). Al entrenar un modelo de Aprendizaje Automático (AA), se fijan los valores de los Hiperparámetros para que con estos los pesos (w) se ajusten, generando predicciones con menor error en contrástate con los datos de prueba. Para este estudio, inicialmente se consideró una (01) capa oculta de 10 neuronas, 10 valores alpha que oscilan entre 10^{-3} y 10^3 y finalmente un learning rate de 10^{-2} y 10^{-1} .

Respecto al hidden layer, cuantas más neuronas y capas, mayor es la complejidad de las relaciones que puede aprender el modelo, por lo que el número de parámetros a aprender aumenta y con ello el tiempo de entrenamiento. El número de neurona en capas ocultas tiene un significado en el actuar de la RNA; con pocas neuronas, la red se aproximará pobremente; mientras que muchas neuronas sobreajustarán los datos de entrenamiento. (ASCE, 2000).

En el caso del learning rate, si este es muy grande, el proceso de optimización puede ir saltando de una región a otra sin que el modelo sea capaz de aprender; por el contrario, si learning rate es muy pequeño, el proceso de entrenamiento puede tardar demasiado y no llegar a completarse. El objetivo de Alpha es evitar que los pesos tomen valores excesivamente elevados, de esta forma se evita que unas pocas neuronas dominen el comportamiento de la red. Con este hiperparámetro se consigue un modelo más 'suave' en el que las salidas cambian más lentamente cuando cambian las entradas.

Modelo Predictivo

Con el fin de determinar una mejor configuración de RNA, se utilizó el modelo MLP o Multi Layer Perceptron a través del paquete *MLPRegressor* para análisis de regresión. El MLP es considerado un aproximador universal (Hornik y otros, 1989) y se ha utilizado con éxito en el modelado hidrológico (ASCE, 2000) (Maier & G. C., 2000). Este paquete está formado por un solucionador y valor de interacciones. El solver es utilizado para la optimización de los pesos (w). En este caso se utilizó el 'lbfgs', el cual es un optimizador de la familia de métodos Cuasi Newton y es utilizado para pocos datos (observaciones < 1000), permitiendo una conversión más rápida y mejor. Finalmente, se ha instruido a la RNA que trabaje con un máximo de iteraciones de 1000.

El proceso del diseño de la RNA, esta descrito en el Diagrama de Proceso de la Figura 4.

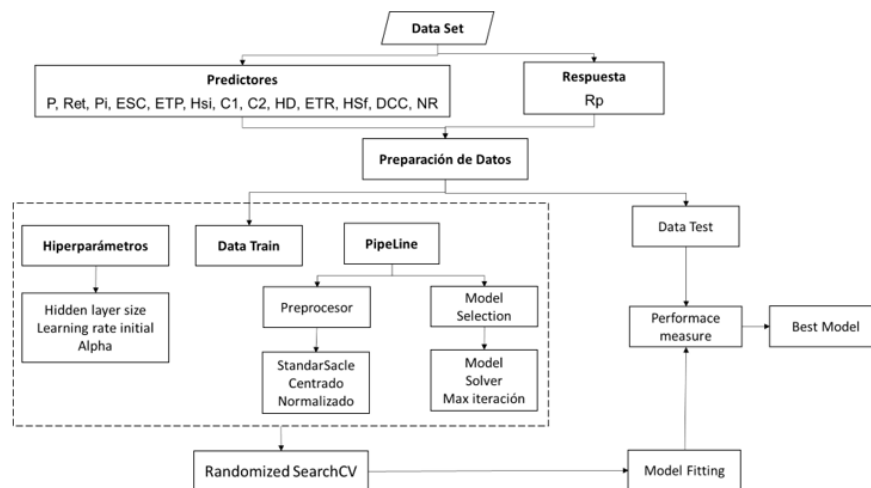


FIGURA 4.
Diagrama de Proceso
Fuente: Elaboración Propia

Entrenamiento de la RNA

Esto es similar a la idea de calibración, el cual es una parte integral de los estudios de modelado hidrológico. El propósito del entrenamiento, es determinar el conjunto de pesos de conexión y umbrales nodales para que la RNA estime salidas lo suficientemente cercanas a los valores observados. Para ello, se utiliza el algoritmo de descenso del gradiente, *'backpropogation'* y la función de coste. Utilizando el algoritmo de Validación cruzada o *"RandomizedSearchCV"*, el set de entrenamiento es dividido en folder (50 en este caso) los cuales son evaluados en la RNA y su resultado final (predicción) es contrastado con el dato de salida (observado) a través función de coste.

Lo que se desea es reducir el error; por lo tanto, se utiliza el descenso de gradiente para estimar el vector gradiente asociado a los parámetros de ajuste (pesos y bias). Este vector es multiplicado por el radio de aprendizaje (*learning rate*) y luego restado a cada peso (w). Para actualizar o ajustar los pesos (w) se utiliza el algoritmo de *'Backpropogation'*, el cual propaga hacia atrás el error, de tal forma que los pesos se actualicen durante cada iteración.

Descenso del Gradiente

El *"gradient descens"*, es un algoritmo clave dentro del *Machine Learning* (ML) y que se localiza en la gran mayoría de los sistemas de Inteligencia Artificial (IA) que se desarrollan. La derivada de una función nos brinda información de la pendiente de dicha función. Al iniciar el entrenamiento, los parámetros se inicializan con un valor aleatorio, es decir iniciar en un punto cualquiera del gráfico. (Figura 5) Lo que se desea es descender (reducir el error) evaluando la pendiente en la posición inicial, por lo tanto, se calcula la derivada en dicho punto, pero en una función multidimensional se estiman las derivadas parciales para cada uno de los parámetros (θ_1 y θ_2) y cada uno de estos valores nos indica la pendiente en el eje de dicho parámetro.

Todas las derivadas parciales conforman de un vector que nos indica la dirección hacia que la pendiente tiende, este vector se denomina el gradiente (Δf). (Ecuación 2). El vector gradiente es utilizado en sentido opuesto para descender, es decir que si el gradiente nos indica como tendríamos que actualizar nuestros parámetros para subir, lo que se hace es restarlo ($\theta = \theta - \Delta f$). Esto nos llevaría a un nuevo lugar de la función donde repetimos el proceso múltiple veces hasta llegar a la zona donde movernos ya no suponga una variación notable del 'coste' o error, es decir que la pendiente es próxima nula y que estemos en el mínimo de la función.

$$\begin{bmatrix} \frac{\partial error}{\partial \theta 1} \\ \dots \\ \frac{\partial error}{\partial \theta 2} \end{bmatrix} = \Delta f$$

ECUACIÓN 2
Vector gradiente

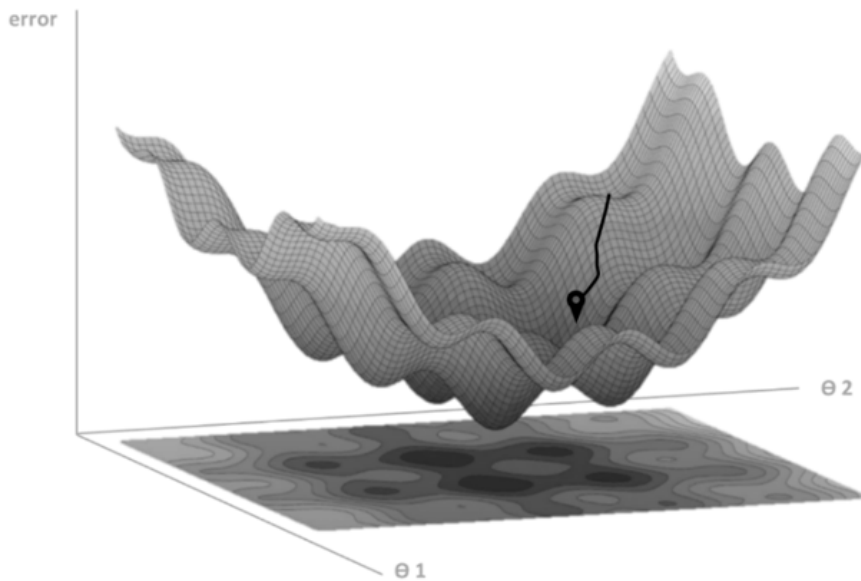


FIGURA 5
Descenso del Gradiente.

Fuente: Adaptada desde Portal de YouTube: DotCSV

Para que el sistema esté completo, solo se añade el radio de aprendizaje (a), el cual define cuánto afecta el gradiente en la actualización de nuestros parámetros en cada iteración o lo que es lo mismo cuánto se avanza en cada paso. Ecuación 3

$$\theta = \theta - a * \Delta f$$

ECUACIÓN 3
Actualización del peso

Backpropagation

El “*backpropagation*” (Werbos, 1994), es quizás el algoritmo más utilizado para el entrenamiento de RNA (Wasserman, 1989). Básicamente, el algoritmo de *Backpropagation* consta de 2 pasos: primeramente, cada patrón de entrada del conjunto de datos de entrenamiento se pasa a través de la RNA desde la capa de entrada hasta la capa de salida. La salida de la red se compara con la salida de objetivo deseada (dato observado) y se calcula un error basado en la función de coste. El segundo paso es cuando el error es propagado hacia atrás a través de la RNA en cada nodo (neurona) y en consecuencia los pesos de conexión se ajustan.

Función de Coste

La función de coste o ‘*cost function*’, es la encargada de cuantificar la distancia entre el valor real y el valor predicho por la red. En otras palabras, mide cuánto se equivoca la red al realizar predicciones. En la mayoría de casos, la función de coste retorna valores positivos. Cuanto más próximo a cero (0) es el valor de coste, mejor son las predicciones de la red (menor error). En este caso, al ser un problema de regresión, se utiliza la función del error cuadrático medio o RMSE (*root mean square error*). Ecuación 4

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(Predicted_i - Actual_i)^2}{n}}$$

ECUACIÓN 4

Función de coste: RMSE

Modelo de Validación

Semejante al modelado hidrológico e hidrogeológico, el rendimiento de una RNA entrenada puede evaluarse sometiendo a nuevos patrones que no ha visto durante el entrenamiento. Este rendimiento es estimado a partir de una función de coste que relaciona los datos previstos y observados. Durante el preprocesado, el conjunto de datos se dividió en datos para el entrenamiento y de prueba, estos últimos son vitales para la validación. Con el modelo RNA ajustado, se mide el rendimiento de este ingresando los datos de prueba (Figura 4) y evaluando el dato generando versus el de prueba mediante la ecuación 4 y finalmente obteniendo el mejor modelo.

2.4 Algoritmos y Paquetes

Para el tratamiento de los datos, gráficos y cálculos se han utilizados distintas librerías del lenguaje de programación *Python*. Las bondades de utilizar este lenguaje son: es de código abierto, lenguaje sencillo y de gran potencial, comprende una comunidad muy amplia de colaboradores, crecimiento en la proyección de nuevas librerías para ser aplicadas, entre otras. La librerías usadas en este estudio fueron:

NumPy, da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas. (Oliphant, 2005). *Pandas* es una biblioteca de software, el cual ofrece estructuras de datos y operaciones para manipular tablas numéricas y series temporales. (McKinney, 2008)

Matplotlib es una biblioteca para la generación de gráficos a partir de datos contenidos en listas o array. (D. Hunter, 2003). *Scikit-learn* o mejor conocida como *Sklearn* es una biblioteca de Aprendizaje automático (AA). Incluye varios algoritmos de clasificación, regresión y análisis de grupos entre los cuales están máquinas de vectores de soporte, bosques aleatorios, K-means, entre otros (Pedregosa y otros, 2010)

3. RESULTADOS Y RECOMENDACIONES

3.1 Características del modelo RNA

La organización y estructuración de los datos de entrada y la identificación de los parámetros más adecuados, son las actividades más complejas en el desarrollo de los modelos de RNA. En esta sección se presentan los resultados de los parámetros y errores de las validaciones cruzadas.

Con los hiperparámetros, la RNA tuvo un buen desempeño a nivel arquitectónico (1 capa oculta con 10 neuronas). Adicional, el *learning rate* se configuró de tal forma que adoptaba valores de 10^{-2} y 10^{-1} , lo que al final el modelo optó por 10^{-2} puesto que un valor menor implica que el proceso de optimización no vaya saltando regiones a otra y logrando la convergencia para el número de iteración establecida. Finalmente, con respecto al parámetro Alpha, este inicialmente presentó 10 valores que oscilan entre 10^{-3} y 10^3 , pero el modelo regresor después de las iteraciones, aprendió con 0.0215 es decir $10^{-1.666}$.

Con los parámetros ajustados, el modelo regresor brindó un RMSE de 0.05, el cual es aceptable debido a que es menor a 0.8, bajo la configuración solver = 'lbfgs', este permite optimizar los valores, converger más rápido y funcionar mejor. En la Figura 6 se presenta la configuración arquitectónica de la RNA. Como se observa a cada input es multiplicado por un peso (flecha sináptica) y luego dentro de la neurona se realiza la suma pondera de $x_i * w_i + b * w$, la cual es evaluada en la función de activación (ReLU) y finalmente se obtiene un valor resultante. Este proceso se repite hasta que se obtenga el resultado en la neurona de salida.

3.2 Resultados

Posterior a la evaluación de la eficiencia del modelo RNA para estimar R_p , se presentan los resultados de la RNA para las distintas texturas de suelo en la zona de estudio. En este caso, cada textura (S01, S02, S03) tiene la misma cantidad de predictores y observaciones en la misma escala de tiempo (2005-2020), variando las propiedades físicas del suelo, es decir, distintos valores de observación.

Los resultados indican que, bajo las mismas configuraciones iniciales en cada RNA, se obtiene un ajuste de acuerdo a los parámetros, siendo evaluados con $RMSE < 0.8$, concluyendo así que los modelos son aceptables. Dentro del proceso de ajuste o aprendizaje, se observa que para la misma arquitectura y número de iteración, las RNA optaron por el valor menor de Learning rate, siendo este el mejor optimizador para la pronta convergencia y solución del modelo; no obstante, cada RNA optó por un valor de alpha distinto, pero dentro de los rangos de 10^{-3} y 10^3 . Tabla 1

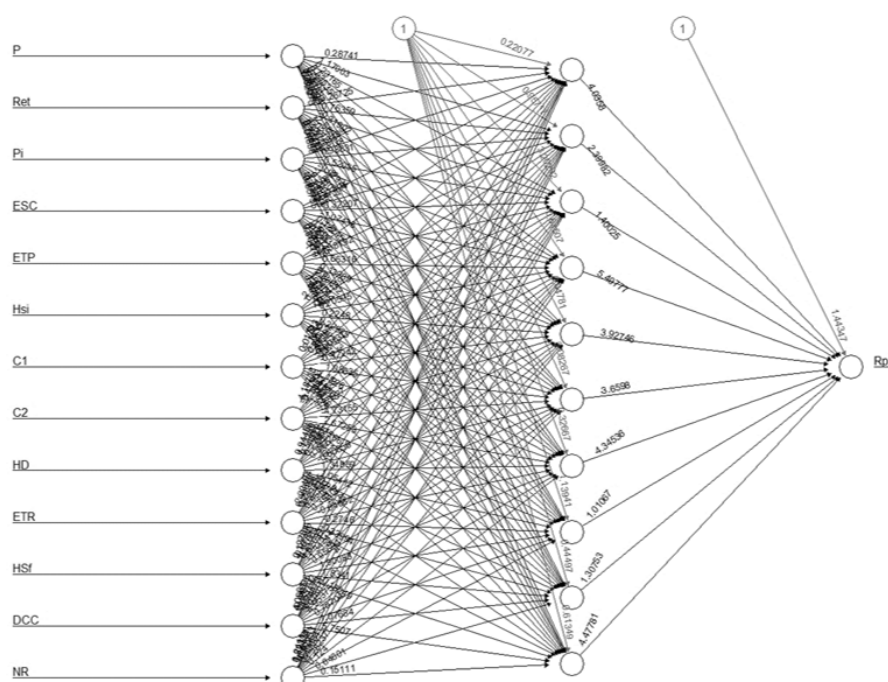


FIGURA 6
Configuración de arquitectónica RNA para S01.
Fuente: Elaboración propia

TABLA 1
Configuración de las RNA para las texturas de suelo.

Textura	Alpha	Hidden layer	Learning rate	Max iteración	RMSE
S01	0.0215	10	0.001	1000	0.05
S02	0.1000	10	0.001	1000	0.02
S03	0.0046	10	0.001	1000	0.16

Fuente: Elaboración propia.

Los resultados de este estudio han permitido identificar la relación entre cada una de las variables y parámetros con los que se diseñó, configuró y ajustó el modelo regresor. Basado en la estimación de errores, el modelo de predicción es aceptable respecto al modelo conceptual. Esto implica que un modelo regresor no lineal de tipo RNA, es capaz de realizar estimaciones y predicciones de variables hidrogeológicas.

3.3 Recomendaciones

Para trabajos futuros, se recomienda el diseño y desarrollo de un conjunto de programas para recolectar e integrar observaciones de más variables en el tiempo y espacio, de tal forma que se genere una base de datos que haga posible utilizar estos de forma simple e interoperable. Los modelos de RNA funcionan mejor cuando la información de distintas variables relacionadas con el fenómeno en cuestión son más largas y completas.

Diseñar una base de datos hidrogeológicas capaz de almacenar la información en un silo y que su diseño sea de forma ‘relacional’ o ‘no relacional’ en dependencia de la variabilidad, volumen y velocidad con que se producen estos datos.

Se recomienda el desarrollo de nuevas investigaciones usando RNA con datos de precipitación, temperatura y nivel de agua subterránea en acuíferos libres. Esto, debido a que los cambios en los niveles de dichos acuíferos (descenso y ascenso) están relacionados con el periodo seco y húmedo.

4. AGRADECIMIENTOS

A INETER, UNAN- Managua y FAO porque través del Proyecto: "Mejoramiento de los sistemas de información hidrometeorológica y climática para favorecer las inversiones en cambio climático en Nicaragua", brindó la oportunidad del desarrollo de esta investigación en el ámbito de la Geoinformática e Hidroinformática, con el objetivo de contribuir al desarrollo profesional de especialistas en hidrología en Nicaragua y Centroamérica.

5. REFERENCIAS

- ASCE. (2000). *Artificial neural networks in hydrology*. J. Hydrol. Eng.
- Corea, F. V. (2014). *Predicción espacio temporal de la irradiancia solar global a corto plazo en España mediante geoestadística y redes neurales artificiales*. España.
- Cortez, P., Rocha, M., Allegro, F., & Neves, J. (2002). *Real-time forecasting by bio-inspired models*. In. *Proceeding of the Artificial Intelligence and Applications*. Málaga, Spain.
- Cournapeau, D. (07 de 2010). *Scikit - Learn*. <https://scikit-learn.org/stable/>
- D. Hunter, J. (2003). *Matplotlib*. Matplotlib: <https://matplotlib.org/>
- FAO. (2013). <https://coin.fao.org/>. <https://coin.fao.org/>: https://coin.fao.org/coin-static/cms/media/5/12820625348650/fao_nic_recurshidricos_cepal.pdf
- Galelli, S., G. B., H., H. R. Maier, A., G. C., D., & M. S., G. (62, 33–51). An evaluation framework for input variable selection algorithms for environmental data-driven models. *Environ. Modell. Software*, 2014.
- Glorot, X., Bordes, A., & Bengio, Y. (01 de 01 de 2010). Deep Sparse Rectifier Neural Networks. *Journal of Machine Learning Research*.
- Hornik, K., M., S., & H., W. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5), 359–366.
- INAA/JICA. (1993). *Estudio sobre el Proyecto de Abastecimiento de Agua en Managua*. Managua.
- INETER. (2004). *Mapa de las Provincias Geológicas de Nicaragua*. INETER, Managua.
- INETER. (2020). *Estudio de Potencial de Recarga Hidrica y Deficit de agua Subterránea*. Managua: Nicaragua.
- INETER. (2020). *Informe de Modelo Numerico del Acuifero de las Sierras*. Managua: Nicaragua.
- INETER. (2021). *Atlas Nacional de suelo: Mapa de Textura de suelo*. Managua: Nicaragua.
- INETER. (13 de 06 de 2022). *Atlas Climáticos. Periodo 1971 - 2010*. Instituto Nicaraguense de Estudios Territoriales, Managua. Retrieved 2022, from <https://www.ineter.gob.ni/>
- Kenda K, P., & Klemen, K. (2019). *Groundwater Modeling with Machine Learning Techniques*. Inst Proc.
- Köppen, W. P. (1918). *Klassifikation der Klimate nach Temperatur*. Hamburg.
- Losilla, M., Rodriguez, H., Stimson, J., & Bethune, D. (2001). *Los Acuífero Volcánicos y el Desarrollo sostenible en America Central*. San José: Editorial Universidad de Costa Rica.
- Maier, H., & G. C., D. (2000). Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environ. Modell. Software*, 15(1), 101–124.
- McKinney, W. (2008). *Pandas*. <https://pandas.pydata.org/>
- Oliphant, T. (2005). *NumPy*. NumPy: <https://numpy.org/>

- Pedregosa, F., Varoquaux, G., & Gramfort, A. (2010). Scikit-learn: Machine Learning in Python. *JMLR 12, 2011*, 2825-2830. Scikit-learn: Machine Learning in Python: <https://scikit-learn.org/stable/>
- Pham, Q., Kumar, M., Di Nunno, F., & Elbeltagi, A. (2022). *Groundwater level prediction using machine learning algorithms in a*. Springer.
- Quilty, J., J. Adamowski, B., & M., R. (2016). Bootstrap rank-ordered conditional mutual information (broCMI): A nonlinear. *Water Resour. Res*, 52, 2299–2326.
- Reed, R., & Marks, R. (1999). *Neural smithing supervised learning in feedforward neural networks*. Mit Press.
- Rossum, G. v. (1991). <https://www.python.org/>
- S. Sahoo, T.A, R., J., E., & I., F. (2017). Machine Learning Algorithms for Modeling Groundwater Level changes in agricultural regions of the US. *AGUPUBLICATIONS*, 53,3878-3895.
- Schosinsky, G., & Losilla, M. (2006). Cálculo de la recarga potencial de acuíferos mediante un balance hídrico de suelos. *Revista Geológica de América Central*, 34-35, 13-30. .
- Shortridge J.E, G. (2019). *Machine Learning methos for empirical streamflow simulation*.
- Tao, H., Hameed, M., Marhoon, H., Zounemat-Kermani, M., Heddami, S., Kim, S., . . . Saadi, Z. (2022). Groundwater Level Prediction using Machine Learning Models:A Comprehensive Review. *ELSEVIER*.
- Thorntwaite, C. W. (1948). An Approach Toward a Rational Classification of Climate. American Geographical Society.
- Wasserman, P. (1989). *Neural computing: theory and practice*. New York: Van Nostrand Reinhold.
- Werbos, P. (1994). *The Roots of Backpropagation From Ordered Derivatives to Neural Networks and Political Forecasting*. New York, USA: Wiley Interscience Publication.