

DEPÓSITO LEGAL ZU2020000153

ISSN 0041-8811

E-ISSN 2665-0428

# Revista de la Universidad del Zulia

Fundada en 1947  
por el Dr. Jesús Enrique Lossada



**Ciencias del**  

---

**Agro**  

---

**Ingeniería**  

---

**y Tecnología**  

---

**Año 12 N° 32**

**Enero - Abril 2021**

**Tercera Época**

**Maracaibo-Venezuela**

## Evaluación de la estabilidad de taludes en presas de tierra empleando Redes Neuronales Artificiales

Isaida Flores Berenguer \*

Jenny García Tristán \*\*

Yoermes Glez Haramboure \*\*\*

### RESUMEN

Se propone el análisis de la estabilidad de taludes en presas de tierra en Cuba, empleando Redes Neuronales Artificiales. Actualmente, no hay precedentes en el país de este tipo de estudios. Por tanto, se evalúan los modelos de la caja de herramientas de redes neuronales de MATLAB® fijando como punto de partida una red perceptrón multicapa con algoritmo de retropropagación, con dos capas ocultas, combinando las funciones de entrenamiento y de activación disponibles. Se analiza una presa de tierra conformada por cuatro suelos parcialmente saturados en la cortina, en estado de operación y final de la construcción. Se obtuvo un coeficiente  $R^2$  de 0,99998 para la función de Regularización Bayesiana considerando la función tangente hiperbólica en la primera capa y lineal pura en la segunda capa. Se propone a futuro ampliar el uso del método evaluando diversas variables que afectan la estabilidad de taludes en presas bajo múltiples condiciones de carga.

**PALABRAS CLAVE:** estabilidad de taludes, funciones de activación, funciones de entrenamiento, inteligencia artificial, Redes Neuronales Artificiales.

\*Ingeniera Civil. Máster en Ingeniería Civil. Centro de Estudio de Construcciones y Arquitectura Tropical (CECAT), Universidad Tecnológica de La Habana “José Antonio Echeverría” (CUJAE). E-mail: [isaidafb@civil.cujae.edu.cu](mailto:isaidafb@civil.cujae.edu.cu)

\*\*Ingeniera Civil. Doctora en Ciencias Técnicas. Centro de Estudio de Construcciones y Arquitectura Tropical (CECAT), Universidad Tecnológica de La Habana “José Antonio Echeverría” (CUJAE). E-mail: [jenny@civil.cujae.edu.cu](mailto:jenny@civil.cujae.edu.cu)

\*\*\* Ingeniero Hidráulico. Doctor en Ciencias Técnicas. Centro de Investigaciones Hidráulicas (CIH), Universidad Tecnológica de La Habana “José Antonio Echeverría” (CUJAE). E-mail: [yoermes@civil.cujae.edu.cu](mailto:yoermes@civil.cujae.edu.cu)

Recibido: 14/10/2020

Aceptado: 09/12/2020

# Evaluation of slope stability in earth dams using Artificial Neural Networks

## ABSTRACT

Analysis of slope stability in earth dams in Cuba is proposed, using Artificial Neural Networks. Currently, there are no precedents in the country for this type of study. Therefore, neural networks toolbox of MATLAB® models are evaluated, setting as a starting point a multilayer perceptron network with backpropagation algorithm, with two hidden layers, combining the available training and activation functions. Earth dam with four partially saturated soils in the embankment is analyzed, in states of operation and at the end of construction. A coefficient R2 of 0.99998 was obtained for the Bayesian Regularization function, considering the hyperbolic tangent function in the first layer and pure linear in the second layer. In the future, it is proposed to expand the use of the method by evaluating variables that affect slope stability in dams under multiple loading conditions.

KEYWORDS: activation functions, artificial intelligence, Artificial Neural Networks, slope stability, training functions.

## Introducción

En los últimos años, con el desarrollo de las tecnologías de cómputo, las Redes Neuronales Artificiales (RNA) han sido ampliamente utilizadas debido a sus buenos resultados, su capacidad de generalización y la posibilidad de analizar gran cantidad de datos (Chafra, 2019; Montoya, 2018; Pano, 2015). Una RNA está conformada por un conjunto de unidades elementales de procesamiento llamadas neuronas, elemento procesador a partir del cual un vector de entrada procedente del exterior o de otras neuronas proporciona una única respuesta o salida. Las neuronas se agrupan en capas, de manera que una red neuronal está compuesta por varias capas de neuronas, las cuales suelen ser del mismo tipo. Estas neuronas almacenan la información a partir de los pesos sinápticos, resultando así, un proceso iterativo de adaptación que dependerá del proceso de aprendizaje (Chafra, 2019; Montoya, 2013, 2018).

El proceso de aprendizaje consiste en el ajuste de los parámetros libres de la RNA, a partir de la estimulación del entorno que la rodea, y es conocido también como entrenamiento. Llegados a este punto es importante verificar los errores que comete la red ante patrones no utilizados en el aprendizaje, para medir su capacidad de generalización. El desempeño del

entrenamiento de una red neuronal depende del algoritmo de aprendizaje utilizado, del número de capas ocultas, del número de neuronas en cada capa oculta, de la conectividad o arquitectura de la red y también del tipo de función de activación que se elija para cada neurona (Llano et al., 2007).

En el análisis de la estabilidad de taludes intervienen diversidad de variables, y la relación entre ellas puede tornarse compleja. La implementación de la Mecánica de Suelos Parcialmente Saturados, adiciona nuevos parámetros que modifican y dificultan en cálculo (Flores et al., 2019; Flores et al., 2020). Por ello, esta investigación propone un primer acercamiento al análisis de la estabilidad de taludes usando RNA, para proyectos ubicados en la República de Cuba.

Se evalúan los resultados obtenidos mediante el software GeoStudio (2012) empleando el Método de Elementos Finitos (MEF) en una presa de tierra homogénea, definiendo como variables de entrada: la altura de la cortina, el ángulo de inclinación del talud aguas abajo, la cohesión, el ángulo de fricción interna, el peso específico, el ángulo de succión del suelo parcialmente saturado de la cortina, el nivel del agua en la presa, el suelo en la base, la cohesión, el ángulo de fricción interna y el peso específico de la cimentación. Siendo el factor de seguridad la variable dependiente o de salida. La modelación empleando el GeoStudio se dificulta debido a la disponibilidad del software y a que es necesario establecer correctamente la geometría de la presa, los materiales que la conforman, generarlos en el modelo y fijar condiciones de borde, todo ello unido al tiempo de corrida para generar un resultado en cada caso. Las facilidades que ofrecen las RNA es diversa, pues una vez que la red esté adecuadamente entrenada continúa aprendiendo y puede facilitar un buen resultado para combinaciones de entrada que no recibió durante el entrenamiento. La aproximación consistió en usar las mismas variables para entrenar una red neuronal artificial, que eliminaría la necesidad de emplear diversos modelos numéricos empleando el GeoStudio (2012).

Diversos autores han utilizado las RNA en los últimos años para el análisis de la estabilidad de taludes y otros fenómenos geotécnicos. (Costa, 2016) plantea que no existe ningún método que determine la arquitectura adecuada al resolver un problema específico por lo que esta tarea se basa en la experiencia del analista que diseñe la red; en su investigación utiliza la función de Regularización Bayesiana obteniendo un coeficiente de correlación  $R^2$  igual a 95,5%

mediante la función tangente hiperbólica con 10 neuronas en la primera capa oculta y evalúa además sólo la función sigmoidea. Por otro lado, (Montoya, 2018) emplea la función de Levenberg-Marquardt obteniendo para suelos arcillosos un coeficiente de correlación  $R^2$  de 83,2%. (Beiranvand, Mohammadzadeh, & Komasi, 2019) emplean la función de Regularización Bayesiana obteniendo un coeficiente de correlación  $R^2$  de 99%, sin embargo, utilizan 25 neuronas en la primera capa oculta. (Gomes, 2016) utiliza seis modelos de redes neuronales con 20 y 25 neuronas en la primera capa oculta en todos los casos, obteniendo coeficientes de correlación  $R^2$  entre 83% y 90%. En ninguno de estos casos se plantea la evaluación de la diversidad de combinaciones disponibles para obtener la óptima respecto a las catorce funciones de entrenamiento y las tres funciones de activación implementadas en MATLAB®, con un número reducido de neuronas en la primera capa oculta.

A partir de lo anterior, se define un modelo con once variables de entrada, dos capas ocultas y una capa de salida. En la primera capa oculta se fijan cinco neuronas, y en la segunda una neurona. Se mantienen constantes la función de aprendizaje adaptativo de tipo gradiente descendente con momento y la función de rendimiento siendo la de error cuadrático medio. Con todo ello se evaluarán las combinaciones entre las funciones de entrenamiento y de activación disponibles en MATLAB®, a partir del coeficiente de correlación  $R^2$  y del error obtenido en cada caso para definir el modelo más adecuado para evaluar la estabilidad en presas homogéneas con suelos parcialmente saturados en la cortina.

## 1. Bases teóricas

Las RNA son un conjunto de unidades elementales, basadas en modelar la forma de procesamiento de la información en sistemas nerviosos biológicos, fundamentalmente, a partir del funcionamiento del cerebro humano, el cual, corresponde a un sistema altamente complejo, no-lineal y paralelo; lo que significa que es capaz de realizar simultáneas actividades al mismo tiempo. Para establecer una similitud entre la actividad sináptica y la analogía con las RNA, se considera que las señales que llegan a la sinapsis son las entradas a la neurona las cuales son atenuadas o simplificadas a través de un parámetro denominado peso, el cual está asociado a la sinapsis correspondiente. Las variables de entrada y salida del sistema pueden ser binarias,

también llamadas digitales o analógicas, dependiendo del modelo de activación del sistema. En la Figura 1 se muestra el modelo de una neurona artificial.

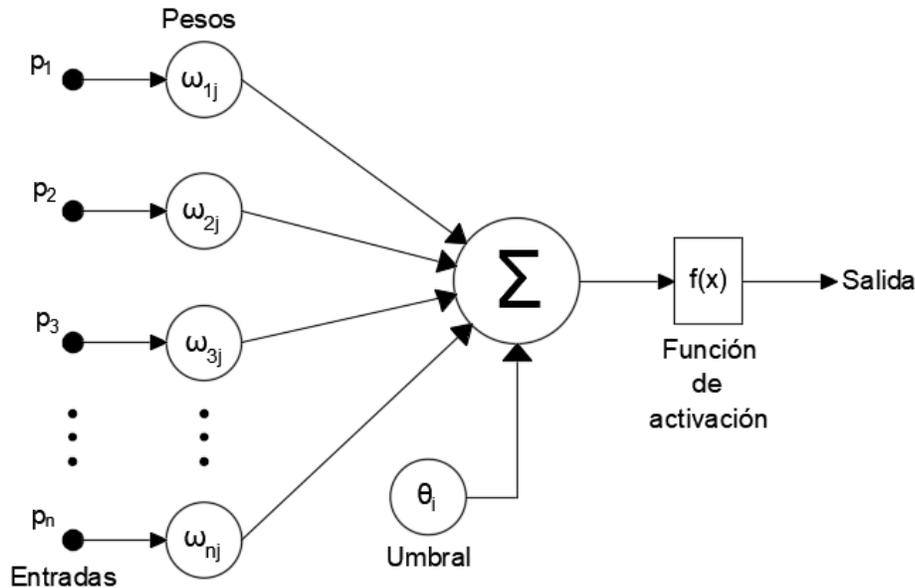
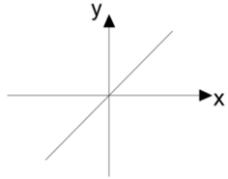
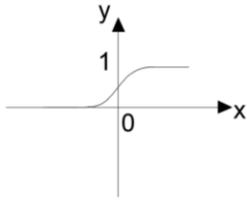
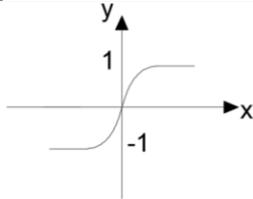


Figura 1. Modelo de una neurona artificial. (Montoya, 2018)

En el modelo de una neurona artificial las variables de entrada ( $p_i$ ) generan los valores de los pesos sinápticos de las dendritas ( $w_{ij}$ ) a partir de una regla de propagación, asociada a la sumatoria de todas las entradas correspondientes a los pesos sinápticos generados ( $\Sigma$ ). Debe considerarse la variable que representa el umbral ( $\theta_i$ ) que debe sobrepasar la señal para ser enviada hacia la neurona post-sináptica. En este punto se requiere una función de activación ( $f(x)$ ) que definirá la salida de la neurona.

Teóricamente, cualquier función derivable se puede utilizar como una función de activación. Sin embargo, una función de activación que tiene un carácter no lineal es muy importante para que sea capaz de discriminar las complejas relaciones que existen en el conjunto de características. El software MATLAB® tiene implementado en la caja de herramientas para redes neuronales, neural network tool box, tres funciones de activación que se muestran en la Tabla 1, aunque el código de entrenamiento de una red neuronal permite manipular y colocar otras funciones de activación.

Tabla 1. Funciones de activación incorporadas en MATLAB®.

Función de activación	Ecuación matemática	Representación gráfica
Lineal pura (PURELIN)	$y = x$	
Sigmoidea o logística (LOGSIG)	$y = \frac{1}{1 + e^{-x}}$	
Tangente hiperbólica (TANSIG)	$y = \frac{1 - e^{-x}}{1 + e^x}$	

La función lineal pura, permite que el resultado obtenido pueda tomar cualquier valor. La función sigmoidea o logística produce la señal de salida en el rango cerrado de 0 a 1 cuyos valores se obtienen únicamente para valores infinitos negativo y positivo, respectivamente. La función tangente hiperbólica es una variante bipolar de la función sigmoidea, la cual produce una salida dentro del rango cerrado de -1 a +1, que se obtienen en el menos y más infinito, respectivamente.

(Montoya, 2018) plantea que no existe un criterio uniforme para definir el número de capas intermedias ni el número de neuronas por capa que compondrá la RNA. Por lo que este proceso depende del investigador y del tipo de aplicación que se le dará a la red. Sin embargo, (González Salcedo, Gotay Sardiñas, Roodschild, Will, & Rodríguez, 2017) plantean que para definir el número de neuronas ocultas la metodología que usa el algoritmo de poda es una herramienta útil para los usuarios sin experiencia en el uso de redes neuronales artificiales.

En el caso de las redes tipo perceptrón multicapa, se realiza un entrenamiento supervisado, con retropropagación hacia adelante. En la pasada hacia adelante, se aplica en la capa de entrada un patrón o vector de entrada, este propaga su efecto a través de las diferentes capas y como consecuencia produce un vector de salida. Durante este proceso, los pesos

sinápticos de la red son fijos y no se modifican. Durante la pasada hacia atrás en cambio, los pesos si se modifican de acuerdo con la regla de corrección del error. La señal de salida real se compara con la señal deseada y como resultado se obtiene una señal de error, que se propaga en dirección contraria a través de la red modificando los pesos, de forma que, al volver a pasar el vector de entrada hacia adelante, la respuesta obtenida se asemeje más a la salida deseada (Aldabas, 2002).

La topología de red neuronal artificial retropropagación hacia adelante multicapa es la más popular, debido a la flexibilidad en los buenos resultados que entrega y a su robustez ante la presencia de ruido. En esta red hay disponibilidad de muchos algoritmos de entrenamiento y son aproximadores universales; más aún, son capaces de proporcionar buenas respuestas en presencia de datos complejos y ruidosos (González Salcedo et al., 2017).

Las funciones de entrenamiento implementadas en MATLAB® son ejecutadas mediante el algoritmo multicapa de retropropagación hacia adelante, aunque cada una lo implementa de una manera diferente. A continuación, se muestra un breve resumen de estas funciones con el código identificativo de MATLAB®:

-Función de Broyden-Fletcher-Goldfarb-Sanno Quasi-Newton (*trainbfg*):

Permite obtener el mínimo de una función a partir de la propia función y su gradiente, buscando en cada iteración la inversa de la matriz Hessiana, aplicando técnicas de optimización de escalada para encontrar el punto estacionario de la función, aunque no se garantiza la convergencia a menos que exista una expresión cuadrática de Taylor cerca de un valor óptimo. Puede tener un rendimiento aceptable para instancias de optimización no uniformes.

-Función de Regularización Bayesiana (*trainbr*):

Se basa en la optimización de la función de densidad de probabilidad, expresada en la varianza del error entre la salida esperada y la salida estimada de la red neuronal, a partir de una aproximación gaussiana del mismo. Se minimiza la función del error por medio de un proceso de optimización no lineal.

-Función de Gradiente Conjugado con reinicio de Powell/Beale (*traincgb*):

Se basa en el método del gradiente conjugado, incluyendo un procedimiento especial que interrumpe la generación normal de direcciones de búsqueda, evitando así la generación de una

dirección dependiente. Reduce el esfuerzo computacional total para las funciones que requieren muchas reinicializaciones.

*-Función de Gradiente Conjugado de Fletcher-Powell (**traíncgf**):*

Se basa en el método del gradiente conjugado, requiriendo solamente las derivadas de primer orden de la función. Se asume que la dirección inicial de búsqueda es la dirección de descenso empinado en el punto inicial.

*-Función de Gradiente Conjugado de Polak-Ribière (**traíncgp**):*

Se basa en el método del gradiente conjugado, demandando búsquedas lineales exactas, pero asumiendo un modelo general de la función objetivo, por lo que converge más rápido. Presenta una tasa lineal de convergencia si no se emplea la reinicialización.

*-Función de Gradiente Descendente (**traíngd**):*

Permite estimar cada nuevo parámetro a partir del anterior, teniendo en cuenta la derivada de la función de coste, definida como el error cuadrático entre los datos conocidos y los encontrados con el modelo. Se parte en algún punto de la función de error definida sobre los pesos y se mueve el mínimo global de dicha función realizando cualquier paso en una dirección descendente.

*-Función de Gradiente Descendente con Momento (**traíngdm**):*

Variante del método del gradiente descendente donde se incorpora un factor de suavidad llamado momento. Este suaviza los cambios en los pesos filtrando las variaciones de alta frecuencia y amplifica la tasa de aprendizaje efectivo lo que conduce a una convergencia más rápida. El momento permite al sistema evitar mínimos locales, proporcionándole al estado del vector la inercia suficiente en la superficie del error.

*-Función de Gradiente Descendente con Velocidad de Aprendizaje Adaptativa (**traíngda**):*

Variante del método del gradiente descendente, donde se intenta mantener el tamaño del paso de aprendizaje lo más grande posible, mientras se mantiene el aprendizaje estable. La tasa del mismo responde a la complejidad de la superficie del error local.

*-Función de Gradiente Descendente con Razón de Aprendizaje Adaptativa (**traíngdx**):*

Variante del método del gradiente descendente, en el cual la tasa de aprendizaje se calcula en línea, para hacer un mejor ajuste y disminuir las oscilaciones.

-*Función de Lavenberg-Marquardt (trainlm)*:

Técnica estándar para problemas de mínimos cuadrados no lineales. Cuando la solución se encuentra lejos del mínimo local, se comporta lento, pero garantizando convergencia. Por el contrario, cuando la solución es cercana al mínimo local, presenta una rápida convergencia. Elimina el cálculo de la matriz Hessiana.

-*Función Secante de Un Paso (trainoss)*:

Método para encontrar los ceros de una función de forma iterativa, utilizando una serie de raíces de las líneas secantes para aproximar la raíz de la función. Puede considerarse una aproximación en diferencias finitas del método de Newton-Raphson.

-*Función de Entrenamiento Incremental de Orden Aleatorio con Función de Aprendizaje (trainr)*:

Con este método se entrena a la red con reglas de aprendizaje con actualizaciones incrementales después de la presentación de cada entrada, lo cual se realiza en orden aleatorio.

-*Función de Retropropagación Elástica (trainrp)*:

Se basa en el método del gradiente descendente, donde el patrón de entrada genera la salida, luego se calcula el error, el cual se propaga hacia las neuronas anteriores, por lo que cada neurona tiene una única contribución al error total. Las funciones de transferencias deben ser continuas.

-*Función de Gradiente Conjugado Escalado (trainscg)*:

Resuelve numéricamente los sistemas de ecuaciones lineales cuyas matrices son simétricas y definidas positivas, minimizando los funcionales convexos garantizando una única solución. Es un método iterativo aplicable a sistemas dispersos que son demasiado grandes para ser tratados por métodos directos, es el único algoritmo del gradiente conjugado que no requiere buscar linealidad.

En las RNA se emplea el criterio de superficie de error, en donde cada uno de los pesos y umbrales de la red son tomados como una dimensión del espacio. Para cualquier configuración

posible de pesos, el error puede ser dibujado en una dimensión correspondiente formando una superficie de error. El objetivo del entrenamiento de la red consiste en encontrar el punto más bajo en esta superficie multidimensional.

Dentro de las tres funciones de rendimiento que tiene implementada MATLAB®, se utiliza el promedio de los errores al cuadrado (MSE). La cual es igual a la suma de los errores al cuadrado sobre el número de registros, por lo que resulta independiente del número de casos empleados para el cálculo del error, siendo una medida del error comparable para conjuntos de datos con diferente tamaño.

Igualmente, la caja de herramientas para redes neuronales, neural network tool box de MATLAB® tiene implementadas dos funciones de aprendizaje adaptativo, de las cuales se selecciona la de gradiente descendente con momento (LEARNNGDM). Esta función calcula el peso de una neurona dada a partir de la entrada de dicha neurona y del error correspondiente.

Hoy en día resulta difícil concebir la gestión del agua sin la existencia de las presas, tanto en sistemas de aprovechamiento de agua como en los de control avenidas. De la variedad de presas, las de materiales sueltos son muy utilizadas por la adaptabilidad de sus proyectos a las condiciones topográficas, geológicas y a los materiales disponibles en la zona de construcción. Por esta razón, los ingenieros geotécnicos se encuentran constantemente desarrollando técnicas y modelos que permitan estimar de manera adecuada el comportamiento de las presas bajo diferentes solicitaciones (Costa, 2016).

En el análisis de estabilidad de taludes, los métodos de equilibrio límite han sido utilizados durante muchas décadas. La idea de dividir una potencial masa deslizante en dovelas verticales fue introducida en los primeros años del siglo veinte, convirtiéndose en la técnica de análisis numérico más antiguo en geotecnia. En los últimos años se han implementado modelos utilizando el Método de Elementos Finitos, obteniéndose con ello resultados relevantes.

Existen programas especializados que posibilitan la realización de los modelos correspondientes para aplicar alguno de estos métodos de análisis. En todos estos programas es necesario definir la geometría total del modelo, asignarle propiedades a cada uno de los materiales que intervienen, definir condiciones de carga y de fronteras; lo que provoca dificultades cuando es necesario considerar gran cantidad de variables o modificaciones en

algunos de los parámetros que intervienen en el problema.

A raíz de obtener respuestas específicas a problemas determinados, como es el caso de la estabilidad de taludes, donde intervienen diversas variables y no todas tienen igual importancia ni afectan de igual manera la estabilidad, se han implementado en los últimos años modelos basados en RNA (Londoño et al., 2007; Montenegro et al., 2019; Montoya, 2018; Neaupane & Piantanakulchai, 2006; Pradhan & Lee, 2009; Yesilnacar & Topal, 2005, Santillán et al., 2014)

La implementación de una RNA posibilita que, luego de entrenado y validado el sistema, puedan modificarse todos los valores de las variables de entrada y obtener una respuesta con un alto grado de confiabilidad. Esto es posible gracias a que la red es capaz de aprender durante el proceso de entrenamiento y puede generalizar la información aprendida, extendiéndola a datos que no formaban parte de las consideraciones iniciales.

## 2. Materiales y métodos

Para el entrenamiento de las redes neuronales artificiales, se conformó una base de datos construida a partir de un modelo geométrico definido en la Figura 2, lo que permitió analizar un problema típico de estabilidad de presas de tierra.

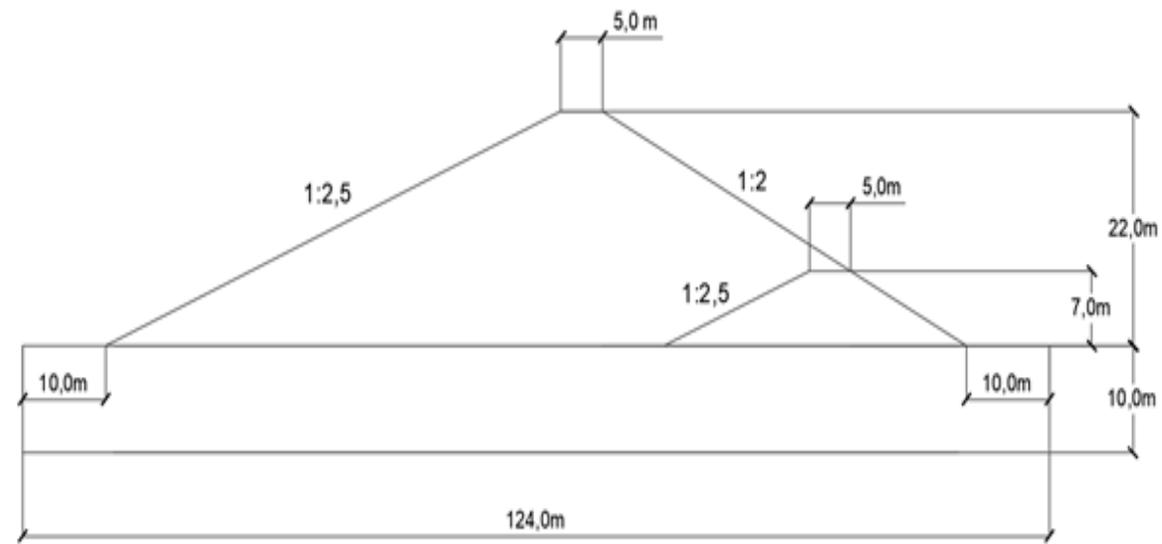


Figura 2. Modelo geométrico.

Las variables relevantes de dicho problema son las que se muestran en la Tabla 2.

Tabla 2. Variables iniciales del modelo de RNA.

Altura de cortina (m)	Inclinación del talud (°)	Cohesión de la cortina (kPa)	Ángulo de fricción de la cortina (°)	Peso específico de la cortina (kN/m <sup>3</sup> )	Nivel de agua (m)	Espesor de cemento (m)	Cohesión del cemento (kPa)	Ángulo de fricción del cemento (°)	Peso específico del cemento (kN/m <sup>3</sup> )
22	27	49,6	14,5	18,53	4,4	10	10	35	17,5
		29	22	16,67			15	37	18,5
		38,7	27,8	14,25			20	39	19,5
		3,4	33,4	17,52					

Para incorporar los suelos parcialmente saturados, se emplea el ángulo de succión para grados de saturación entre 100% y 50% del valor del ángulo de fricción interna para cada caso (Fredlund & Rahardjo, 1993). Esto agregaría una nueva variable al análisis, como se muestra en la Tabla 3.

Tabla 3. Ángulo de succión para el modelo de RNA.

Ángulo de fricción de la cortina (°)	Ángulo de succión correspondiente (°)	Grado de saturación asociado (%)
14,5	14,5	100
	13,1	90
	11,7	80
	10,3	70
	8,8	60
	7,4	50
22	22	100
	19,8	90
	17,7	80
	15,6	70
	13,6	60
	11,4	50
27,8	27,8	100
	25,4	90
	22,9	80
	20,2	70
	17,5	60
	14,8	50
33,4	33,4	100
	30,7	90
	27,8	80
	24,7	70
	21,6	60
	18,2	50

El anterior análisis y definición de variables permitió hacer combinaciones de casos de estudio hipotéticos cuyo resultado consistió en el factor de seguridad del talud aguas abajo. Estas combinaciones conformaron una base de datos con 144 vectores de información, que se utilizó como conjunto de entrenamiento de las redes neuronales. Al estar el conjunto incluyendo las variables de entrada y la variable de salida, correspondieron entonces a un proceso de entrenamiento supervisado. En este proceso, el conjunto se dividió en tres subconjuntos a saber: de entrenamiento (con el 70 % de los datos), de prueba (con el 15% de los datos) y de validación (con el 15% de los datos).

De los 144 modelos, se seleccionaron 101 de ellos, lo que corresponde al 70%, para la etapa de entrenamiento (entrenamiento-validación-prueba) y el 30% restante, es decir, 43 modelos, para comprobar los resultados ofrecidos por la RNA correspondiente a una misma topología de red neuronal con retropropagación hacia adelante, compuesta por cinco neuronas en la primera capa oculta y una neurona en la segunda capa oculta.

La manera de obtener los factores de seguridad para los 144 modelos con los que se evaluó la RNA, fue utilizando los programas SIGMA/W, SEEP/W y SLOPE/W que integran el GeoStudio (2012). Se implementó en todos los casos el MEF para obtener el factor de seguridad del talud aguas abajo.

### 3. Resultados

Los resultados obtenidos con las herramientas SIGMA/W, SEEP/W y SLOPE/W se muestran en las Figuras 3-5.

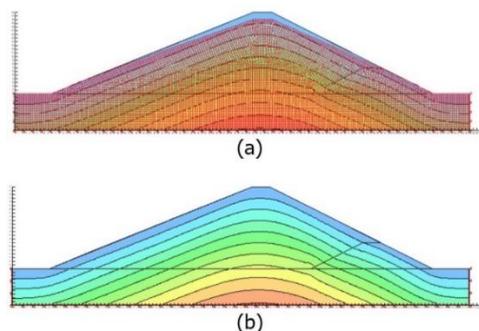


Figura 3. Modelo obtenido en SIGMA/W: (a) Malla deformada; (b) Distribución de tensiones

En la Figura 3 se muestra la malla deformada y la distribución de tensiones asociadas al modelo propuesto.

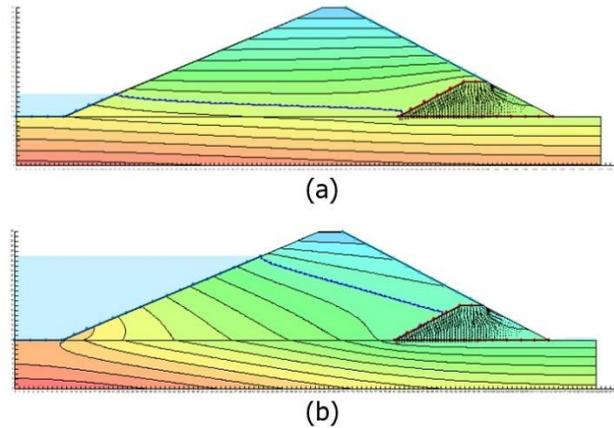


Figura 4. Modelo obtenido en SEEP/W: (a) Final de la construcción; (b) Operación.

En la Figura 4 se muestra la línea de corriente superior generada para los casos de final de la construcción y operación. Se aprecia, además, la variación de la presión de poros en cada caso, producto de la variación del nivel de las aguas en el interior de la cortina de la presa.

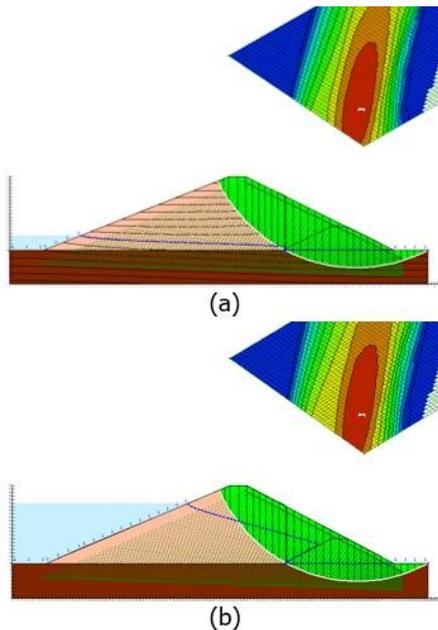


Figura 5. Modelo obtenido en SLOPE/W: (a) Final de la construcción; (b) Operación.

En la Figura 5 se muestra la superficie crítica de deslizamiento para los casos de final de la construcción y operación, notándose que, aunque esta superficie es muy similar para ambos casos, la línea de corriente superior difiere a partir del nivel de agua establecido, lo que implica una variación del factor de seguridad, mostrándose en la diferencia existente en las superficies de respuesta para uno u otro estado de carga.

El modelo general implementado con la RNA se muestra en la Figura 6.

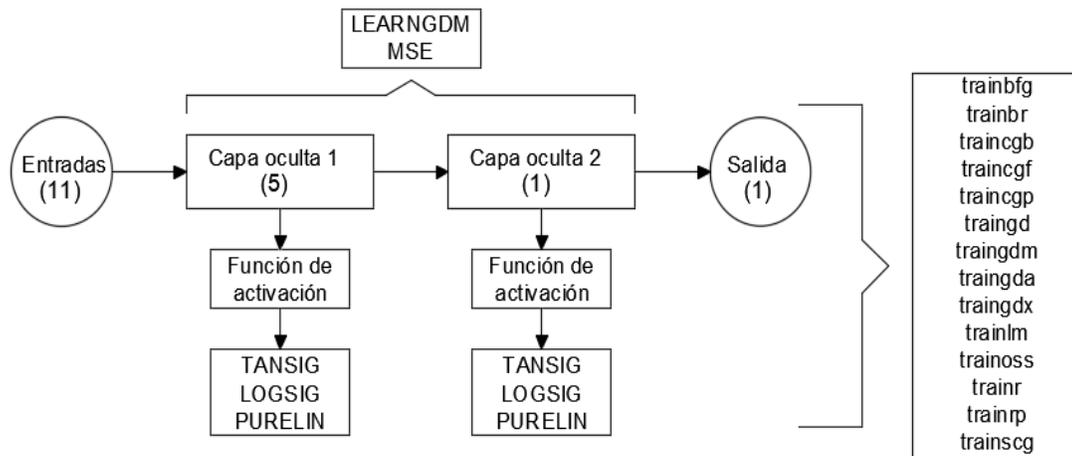


Figura 6. Modelo implementado con la caja de herramientas para redes neuronales en MATLAB®.

A partir de las combinaciones mostradas en la Figura 6 se generan 126 nuevos modelos. Después de analizar cada uno de ellos se obtienen los coeficientes  $R^2$  que se muestran en las Figuras 7 (a), 7 (b) y 7 (c), teniendo en cuenta que la distribución es: función de activación para primera capa oculta - función de activación para segunda capa oculta.

En la Figura 7 (a) se observa que, al comparar los resultados obtenidos con el modelo del MEF respecto a los de RNA para valores superiores al 95% de efectividad, para los casos donde la función de activación de la primera capa oculta es la tangente hiperbólica (TANSIG), los mejores resultados corresponden a la función de activación tangente hiperbólica (TANSIG) en la segunda capa oculta, con un 64,3% de acierto y luego, la función lineal pura (PURELIN) con un 28,6% de aciertos. La función logística (LOGSIG) no tuvo resultados (0% de aciertos) por encima del 95%.

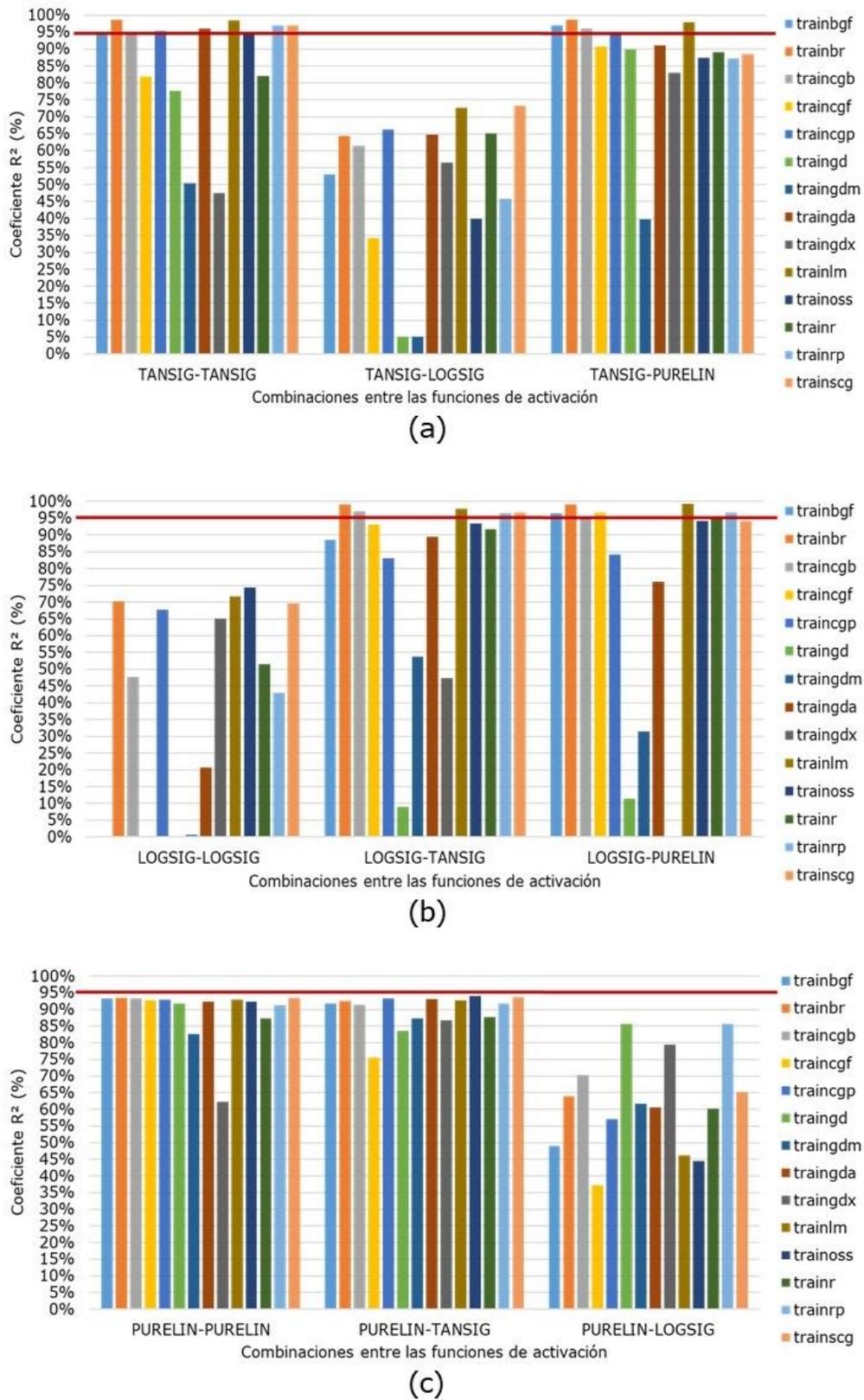


Figura 7. Coeficientes  $R^2$ : (a) TANSIG; (b) LOGSIG; (c) PURELIN.

En la Figura 7 (b) donde la función de la primera capa oculta es la logística (LOGSIG), los mejores resultados corresponden a la función lineal pura (PURELIN) en la segunda capa oculta, con un 50% de aciertos y luego la función tangente hiperbólica (TANSIG) con un 35,7%. Otra vez la función logística no tuvo resultados (0% de aciertos) por encima del 95%.

En la Figura 7 (c) se muestra que para ninguna de las variantes probadas se supera el 95% de semejanza entre los modelos obtenidos con el MEF respecto a los de RNA.

En ninguno de los casos analizados, cuando la función de activación de la segunda capa oculta era la función logística (LOGSIG) se obtuvieron resultados por encima del 95%. Igualmente ocurrió cuando la función de activación de la primera capa oculta era la función lineal pura (PURELIN).

En las Figuras 8 (a), 8 (b) y 8 (c) se muestra el comportamiento del error para cada una de las combinaciones analizadas, al comparar los resultados obtenidos por el modelo con el MEF respecto al basado en RNA.

En la Figura 8 (a) se observa que para los casos donde la función de activación de la primera capa oculta es la tangente hiperbólica (TANSIG), los mejores resultados corresponden a la función de activación tangente hiperbólica (TANSIG) en la segunda capa oculta, con un 28,6% de los valores por debajo de 5%; luego, la función lineal pura (PURELIN) con un 21,4% por debajo de 5% y finalmente la función logística (LOGSIG) con 0% del total de resultados por debajo de 5%.

En la Figura 8 (b) donde la función de la primera capa oculta es la logística (LOGSIG), los mejores resultados corresponden a las funciones tangente hiperbólica (TANSIG) y lineal pura (PURELIN) en la segunda capa oculta, con un 35,7% del total de valores por debajo del 5% en ambos casos; luego, la función logística presenta 0% del total de resultados por debajo del 5%.

En la Figura 8 (c) se muestra que para ninguna de las variantes probadas se obtienen valores del error inferiores al 5% entre los modelos obtenidos con el MEF respecto a los de RNA.

En ninguno de los casos analizados, cuando la función de activación de la segunda capa oculta era la función logística (LOGSIG) se obtuvieron errores por debajo del 5%. Igualmente ocurrió cuando la función de activación de la primera capa oculta era la función lineal pura (PURELIN).

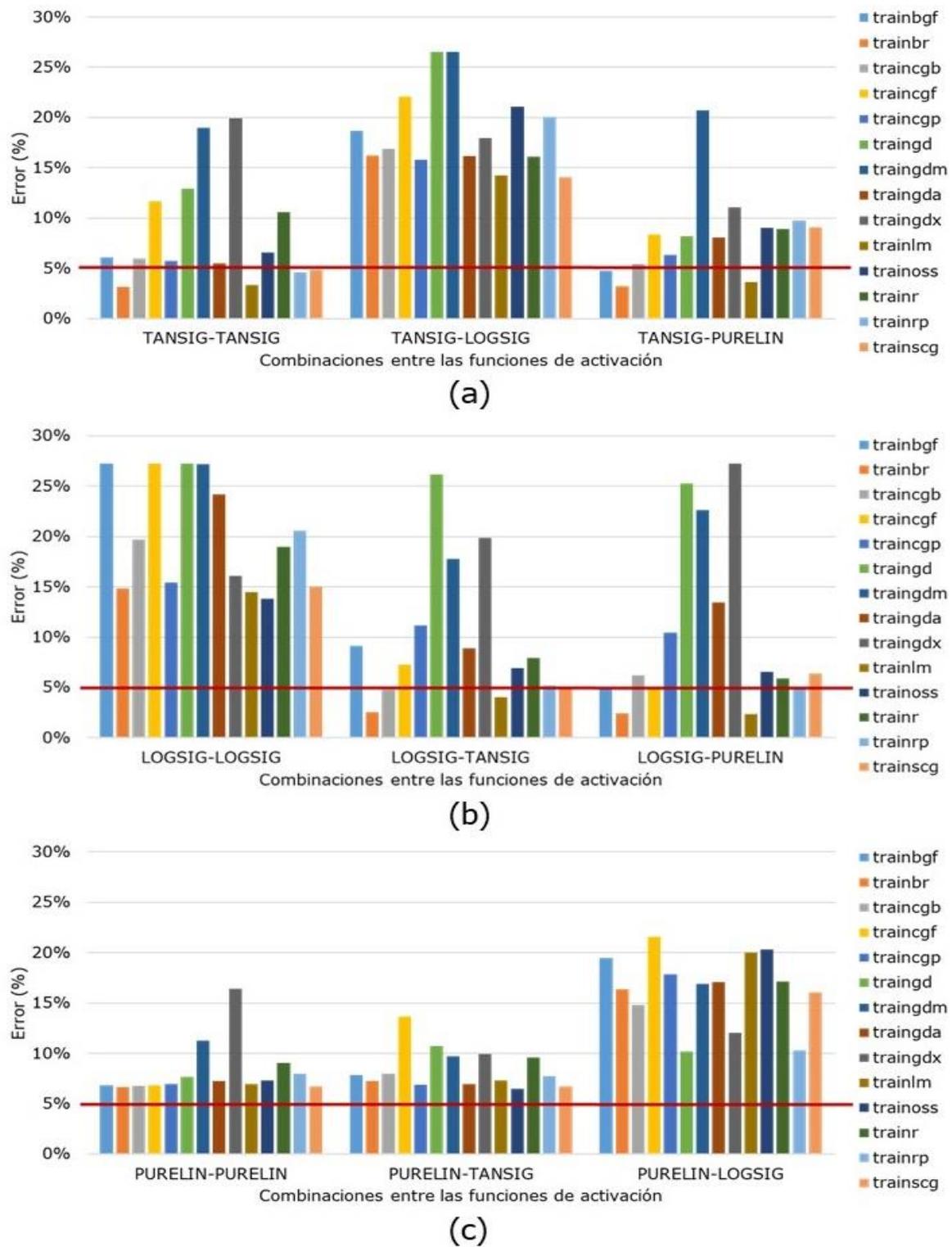


Figura 8. Error: (a) TANSIG; (b) LOGSIG; (c) PURELIN.

A partir de los resultados obtenidos al analizar todas las combinaciones planteadas se obtienen resultados similares utilizando la función de entrenamiento Regularización Bayesiana (*trainbr*), para las combinaciones TANSIG-TANSIG y TANSIG-PURELIN, ambas con un coeficiente  $R^2$  de 99% y un error de 3%. El resultado de los factores de seguridad para ambas combinaciones respecto a los obtenidos por el MEF se muestra en la Figura 9.

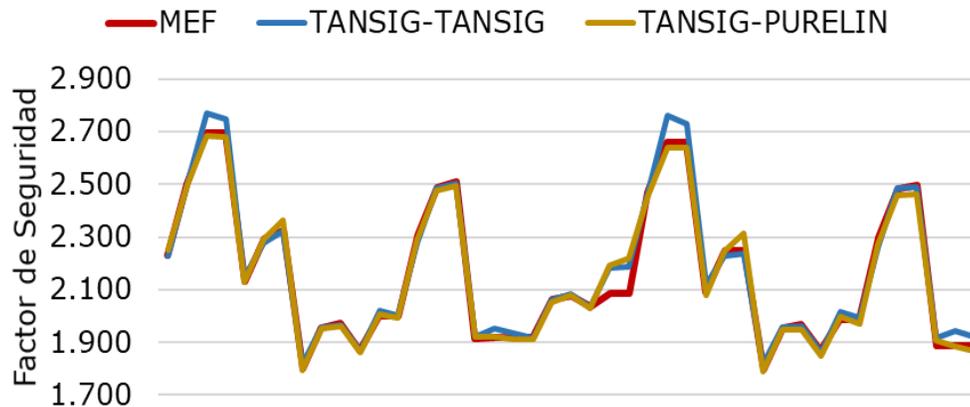


Figura 9. Factores de seguridad obtenidos por el MEF y por las combinaciones de RNA con mejores resultados.

En la Figura 9 se muestran los factores de seguridad obtenidos por el MEF respecto a las combinaciones TANSIG-TANSIG y TANSIG-PURELIN para la función de Regularización Bayesiana (*trainbr*). En la misma, se observa que la combinación TANSIG-PURELIN es la que más se aproxima a los resultados obtenidos empleando el MEF. Esta afirmación se corrobora en los resultados ofrecidos por MATLAB®, al finalizar el proceso de entrenamiento, mostrados en las Figuras 10 (a) y 10 (b)

En las Figuras 10 (a) y 10 (b) se puede apreciar que en el caso de la combinación TANSIG-TANSIG, el resultado general del entrenamiento es de 0,9995; mientras que para la combinación TANSIG-PURELIN se obtuvo un valor de 0,99998. Por lo tanto, se corrobora que el mejor resultado de todos los obtenidos corresponde a la segunda combinación.

Para el caso analizado, se aprecia un elevado coeficiente de correlación obtenido para un número reducido de neuronas ocultas, lo que contrasta con los planteado por diversos autores que han abordado el tema con anterioridad, como (Beiranvand et al., 2019; Costa, 2016; Montoya, 2018).

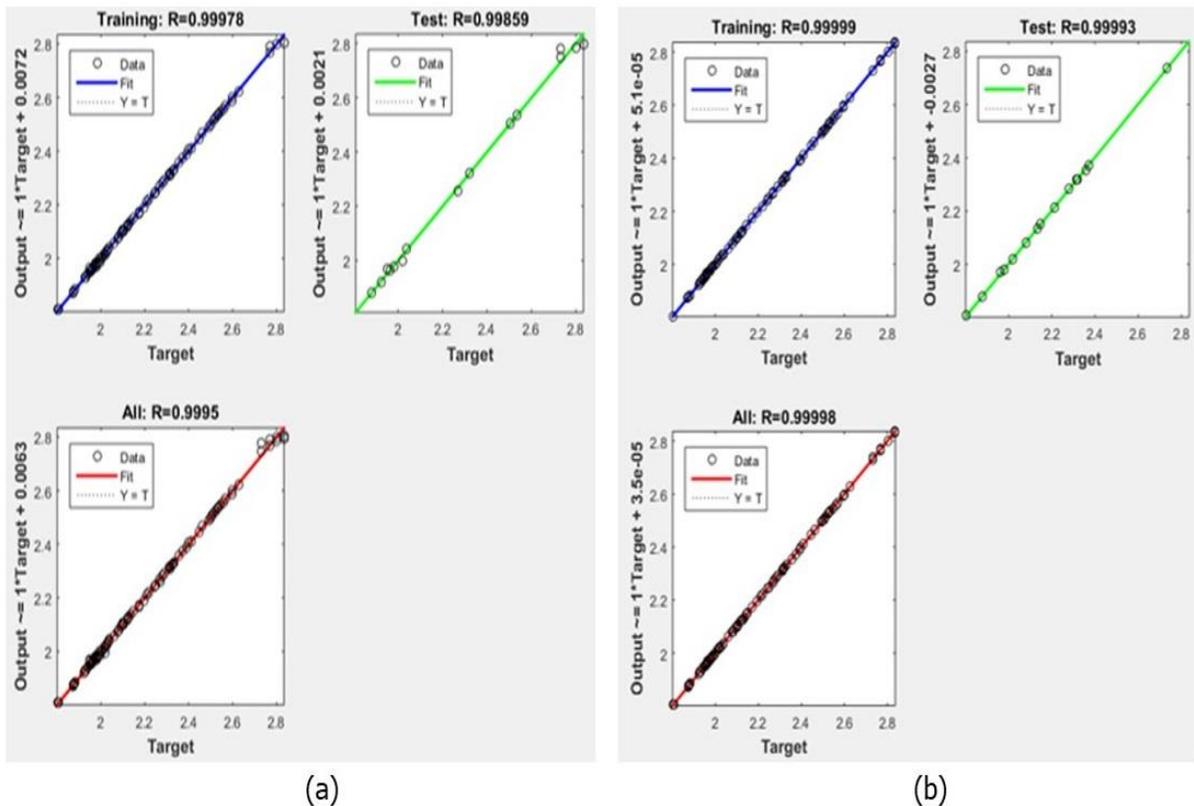


Figura 10. Resultados obtenidos con MATLAB®: (a) TANSIG-TANSIG; (b) TANSIG-PURELIN

## Conclusiones

Se plantea un modelo de RNA, utilizando la caja de herramientas para redes neuronales de MATLAB®, con once variables de entrada, dos capas ocultas, con cinco neuronas en la primera y una neurona en la segunda, cuya función de aprendizaje adaptativo es la de gradiente descendente con momento (LEARNGDM) y la función de rendimiento empleada es la del promedio de los errores al cuadrado (MSE).

Se analizaron 144 modelos para evaluar la influencia de las catorce funciones de entrenamiento y las tres funciones de activación disponibles en la caja de herramientas para redes neuronales de MATLAB®, para ambas capas ocultas. En ninguno de los casos analizados, cuando la función de activación de la segunda capa oculta es la función logística (LOGSIG) se obtuvieron resultados del coeficiente  $R^2$  por encima del 95%. Igualmente, cuando la función de

activación de la primera capa oculta es la función lineal pura (PURELIN), todos los valores del coeficiente  $R^2$  es inferior al 95%.

A partir de los resultados obtenidos, en ninguno de los casos analizados, cuando la función de activación de la segunda capa oculta es la función logística (LOGSIG) se obtuvieron errores por debajo del 5%. Igualmente, cuando la función de activación de la primera capa oculta era la función lineal pura (PURELIN), el valor del error en todos los casos es superior al 5%.

Finalmente, según los resultados obtenidos, la función de entrenamiento con los mejores resultados es la de Regularización Bayesiana (*trainbr*) y la combinación TANSIG-PURELIN es la que más se aproxima a los modelos empleando el MEF, con un coeficiente de correlación  $R^2$  igual a 99% y un error de 3%.

A partir de esta calibración puede emplearse el algoritmo planteado para obtener el factor de seguridad en presas de tierra homogéneas con características similares a las estudiadas en la presente investigación.

## Referencias

- Aldabas, E. (2002). Introducción al reconocimiento de patrones mediante redes neuronales. Conferencias de ingeniería electrónica (pp. 1-3).
- Beiranvand, B., Mohammadzadeh, A., & Komasi, M. (2019). Effect of different parameters of heterogeneous dams on safety factor using the neural network . Case study : Marvak dam. Nexo Revista Científica, 32(2), 126-138. doi:10.5377/nexo.v32i02.9263
- Chafra, E. X. (2019). Análisis del rendimiento de algoritmos de entrenamiento de Redes Neuronales Artificiales aplicadas al modelamiento dinámico de represas hidroeléctricas, mediante el error de predicción del nivel de embalse de agua. Tesis de Maestría. Escuela Superior Politécnica de Chimborazo.
- Costa, C. (2016). Predicción de la estabilidad de presas heterogéneas mediante redes neuronales artificiales. Tesis de Maestría. Universidad Politécnica de Madrid.
- Ermini, L., Catani, F., & Casagli, N. (2005). Artificial Neural Networks applied to landslide susceptibility assessment. Geomorphology, 66(3-4), 327-343. doi:http://doi.org/10.1016/j.geomorph.2004.09.025

Flores, I., Castro, I., Tristá, J. G., & Haramboure, Y. G. (2019). Influencia de la permeabilidad del suelo no saturado en los taludes de presas de tierra. *Ingeniería Hidráulica y Ambiental*, XL(3), 86–100.

Flores, I., Tristá, J. G., & Haramboure, Y. G. (2020). Estabilidad de taludes durante un desembalse rápido en presas de tierra con suelos parcialmente saturados. *Ingeniería y Desarrollo*, 38(1), 19.

Fredlund, D. G., & Rahardjo, H. (1993). *Soil Mechanics for Unsaturated Soils* (p. 567). New York: John Wiley & Sons.

Gomes, G. (2016). Modelo de predicción de aportaciones mensuales utilizando Redes neuronales Artificiales. Aplicación a la cuenca de río Amambaí (Brasil). Tesis de Maestría. Universidad Politécnica de Valencia.

González Salcedo, L. O., Gotay Sardiñas, J., Roodschild, M., Will, A. L., & Rodríguez, S. (2017). Optimización en la elaboración de redes neuronales artificiales adaptativas usando una metodología de algoritmo de poda. *Ingenio Magno*, 8(1), 44–56.

Llano, L., Hoyos, A., Arias, F., & Velásquez, J. (2007). Comparación del desempeño de funciones de activación en Redes Feedforward para aproximar funciones de datos con y sin ruido. *Avances en Sistemas e Informática*, 4(2), 10.

Londoño, J. P., Cifuentes, P. A., & De Felipe, J. J. (2007). Modelización de problemas ambientales en entornos urbanos utilizando sistemas de información geográfica y métodos multivariantes. *Revista Internacional de Sostenibilidad, Tecnología y Humanismo*, (2), 21–51.

Montenegro, D. D., Pérez, M. A., & Vargas, V. (2019). Using Artificial Neural Networks to predict monthly precipitation for the Cali river basin, Colombia. *DYNA*, 86(211), 122–130. doi:<http://doi.org/10.15446/dyna.v86n211.76079>

Montoya, E. (2013). Modelo de alerta de escorregamentos deflagrados por chuvas usando redes neurais artificiais. Tesis de Mestría. Universidad de Brasilia.

Montoya, E. (2018). Metodología para la aplicación de Redes Neuronales Artificiales para sistemas de alerta de deslizamientos provocados por lluvias en regiones montañosas. Tesis de Doctorado. Universidad de Brasilia.

Neaupane, K. M., & Piantanakulchai, M. (2006). Analytic network process model for landslide hazard zonation. *Engineering Geology*, 85(3-4), 281–294. doi:<http://doi.org/10.1016/j.enggeo.2006.02.003>

Ni, S. H., Lu, P. C., & Juang, C. H. (1996). A fuzzy neural network approach to evaluation of slope failure potential. *Computer-Aided Civil and Infrastructure Engineering*, 11(1), 59–66. doi:[http://doi.org/10.1016/0148-9062\(96\)81903-6](http://doi.org/10.1016/0148-9062(96)81903-6)

Pano, A. D. (2015). Realización FPGA de una Red Neuronal Artificial para Reproducir Comportamiento Caótico. Tesis de Maestría. Instituto Nacional de Astrofísica, óptica y electrónica.

Pradhan, B., & Lee, S. (2009). Landslide risk analysis using artificial neural network model focussing on different training sites. *International Journal of Physical Sciences*, 4(1), 1-15. doi:<http://www.academicjournals.org/IJPS>

Pradhan, B., & Lee, S. (2010). Regional landslide susceptibility analysis using back-propagation neural network model at Cameron Highland, Malaysia. *Landslides*, 7(1), 13-30. doi:<http://doi.org/10.1007/s10346-009-0183-2>

Santillán, D., Fraile-Ardanuy, J., & Toledo, M. Á. (2014). Predicción de lecturas de aforos de filtraciones de presas bóveda mediante redes neuronales artificiales. *Tecnología y Ciencias del Agua*, V(3), 81-96.

Yesilnacar, E., & Topal, T. (2005). Landslide susceptibility mapping: A comparison of logistic regression and neural networks methods in a medium scale study, Hendek region (Turkey). *Engineering Geology*, 79(3-4), 251-266. doi:<http://doi.org/10.1016/j.enggeo.2005.02.002>