



**Revista Especializada de Ingeniería y  
Ciencias de la Tierra**

**ISSN: L2805 -1874**

**Vol: 2 N°1 Julio - Diciembre 2022**

**USO DEL APRENDIZAJE AUTOMÁTICO PARA PREDECIR SI UNA FUNDACIÓN  
CUADRADA AISLADA PROPUESTA CUMPLE CON EL ESTÁNDAR ACI 318-11**

**USING MACHINE LEARNING TO PREDICT WHETHER A PROPOSED ISOLATED  
SQUARE FOUNDATION MEETS THE ACI 318-11 STANDARD**

Gabriel Montúfar Chiriboga  
Universidad Tecnológica de Panamá, Facultad de Ingeniería Eléctrica  
gjm31416@gmail.com  
ORCID.ORG/0000-0003-3392-3728

Recibido: 15/4/2022 Aceptado: 18/6/2022 Publicado: 7/2022

Se autoriza la reproducción total o parcial de este artículo, siempre y cuando se cite la fuente completa y su dirección electrónica

**RESUMEN**

Cuando un ingeniero civil diseña un proyecto, se generan y registra múltiple información en documentos físicos o digitales. En el pasado, la mayor parte de esta información se escribía en papel y esta se extraviaba o degradaba con el tiempo, lo que provocaba la pérdida de datos. Actualmente la mayor parte de la información se registra digitalmente en ordenadores, a través de documentos txt, pdf, CSV, bases de datos SQL, imágenes, capturas de sonido, etc. Todos estos datos se acumulan generalmente de forma desordenada y sin un uso específico. ¿Se podría utilizar toda esta información? Mediante el aprendizaje automático y la creación de una base de datos adecuada, la información recopilada por diseños anteriores puede utilizarse para realizar predicciones que nos permitan conocer algún atributo de interés, por ejemplo, si un diseño es

191

adecuado en relación con el estándar ACI 318-11. En este artículo, se utilizó el software WEKA para entrenar y probar modelos con algoritmos como J48, Naive Bayes, Logistic y AdaBoostM1. Se seleccionó el mejor modelo y posteriormente se realizaron predicciones con datos externos al conjunto de datos de entrenamiento.

**PALABRAS CLAVE:** aprendizaje automático; predicción del diseño estructural; árbol de decisión J48; Naive Bayes; AdaBoostM1.

## **ABSTRACT**

When a civil engineer designs a project, multiple information is generated and recorded in physical or digital documents. In the past, most of this information was written on paper and was lost or degraded over time, causing data loss. Currently most of the information is recorded digitally on computers, through txt, pdf, CSV documents, SQL databases, images, sound captures, etc. All this data is generally accumulated in a disorderly way and without a specific use. Could all this information be used? Through machine learning and the creation of a suitable database, the information collected by previous designs can be used to make predictions that allow us to know some attribute of interest, for example, if a design is suitable in relation to the standard ACI 318-eleven. In this article, WEKA software was used to train and test models with algorithms such as J48, Naive Bayes, Logistic, and AdaBoostM1. The best model was selected and then predictions were made with data external to the training data set.

**KEY WORDS:** machine learning; structural design prediction; decision tree J48; Naïve Bayes; AdaBoostM1.

## **INTRODUCCION**

El concepto de computadora digital programable fue inventado por Charles Babbage, un ingeniero mecánico y filósofo, en 1837. El motor analítico fue el término dado a este concepto, que se basaba en una computadora mecánica con funcionalidad genérica. En el motor analítico se incluyeron una unidad lógica aritmética, control de flujo en forma de ramas y bucles condicionales con memoria integrada (Enciclopedia Britanica, 2020).

Luigi Federico Menabrea, matemático y ex primer ministro italiano, basó su artículo “Bosquejo del motor analítico inventado por Charles Babbage” en este concepto, que describe muchas características de la arquitectura y programación de computadoras. Augusta Ada King , matemática y escritora, tradujo el trabajo de Menabrea y contribuyó con una serie de comentarios en 1842. Describió un procedimiento para que el motor analítico calculara los números de Bernoulli en sus notas. Se cree que es el primer algoritmo publicado que fue diseñado específicamente para su implementación en computadora. Su programa nunca se probó porque el motor nunca se terminó (Taylor, 1843).

Alan Mathison Turing, matemático, filósofo y biólogo teórico publicó en octubre de 1950 en la revista académica MIND de la Universidad de Oxford, un artículo titulado “Equipos de inteligencia y computación”, en el que planteaba la pregunta: ¿pueden pensar las máquinas? Turing alude a la traducción de Augusta Ada King de su afirmación de que el motor analítico no está

diseñado para originar nada y puede realizar cualquier cosa que sepamos cómo indicarle que haga. Turing afirma que las máquinas lo toman con frecuencia desprevenido. Esto se debe en gran parte a que no realiza suficientes cálculos para determinar qué esperar de estas, o más bien, que cuando realiza un cálculo, lo hace con prisa, sin cuidado y arriesgándose (Turing , 1950).

Durante un simposio en el Dartmouth College en 1956, John McCarthy, científico informático y científico cognitivo, acuñó el nombre de Inteligencia Artificial para identificar esta disciplina de la cibernética, y así se estableció la Inteligencia Artificial. McCarthy también creó la familia de lenguajes de programación Lisp, que ahora se ha convertido en el estándar para aplicaciones de inteligencia artificial (McCarthy et al., 1956).

Arthur Lee Samuel, un ingeniero eléctrico, popularizó la frase aprendizaje automático en 1959 cuando desarrolló el software del juego Samuel Checkers, que fue uno de los primeros programas efectivos de autoaprendizaje del mundo y, por lo tanto, una ilustración muy temprana de la noción fundamental de inteligencia artificial (Samuel, 1959).

En el ensayo WEKA: A Machine Learning Workbench, publicado en 1994, el matemático Geoffrey Holmes et al., explicó que WEKA es un banco de trabajo de aprendizaje automático que está destinado a ayudar en la aplicación de técnicas de aprendizaje automático a una variedad de situaciones del mundo real. Para ayudar en este esfuerzo, se ha creado un banco de trabajo para proporcionar un entorno que no solo permite el acceso directo a una serie de enfoques de aprendizaje automático, sino que también incluye las herramientas de procesamiento previo y posterior que se requieren cuando se trabaja con un conjunto de datos mundiales reales (Holmes et al., 1994).

Con la llegada de Internet, ahora hay cantidades masivas de datos disponibles para entrenar modelos y el poder de procesamiento de la computadora se está expandiendo, esto hace que el potencial del aprendizaje automático sea casi ilimitado.

## **ANTECEDENTES**

Se les pide a ingenieros de todo el mundo que diseñen estructuras iguales o muy similares una y otra vez, sin aprender del pasado, según el ingeniero estructural Lorenzo Greco en 2018. Los enfoques basados en el aprendizaje automático pueden aprovechar el conocimiento existente. Cuanto más tiempo se emplea este método, más puntos de datos se recopilan y más preciso se vuelve. Permitiría predicciones estadísticas automáticas sobre grandes cantidades de datos, así como verificaciones, que es un aspecto crucial de la evaluación de proyectos (Greco, 2018).

El aprendizaje automático es una herramienta poderosa para anticipar y evaluar el desempeño estructural, identificar la condición estructural e informar acciones preventivas y de recuperación mediante la identificación de patrones a partir de datos recopilados de diversas fuentes, según el ingeniero de investigación Han Sun et al. en 2020. Se deben superar varios problemas importantes para introducir el aprendizaje automático en la práctica de la construcción. Por ejemplo, actualmente no existen fuentes de datos de alta calidad adecuadas para la construcción de modelos de aprendizaje automático. Como resultado, se requiere un esfuerzo coordinado para desarrollar, recopilar y preservar diversos conjuntos de datos en un repositorio de código abierto al que puedan contribuir académicos y profesionales (Sun et al., 2020).

El ingeniero civil M.Z. Naser et al. declaró en 2020 que el análisis y diseño de columnas tubulares de acero rellenas de hormigón es sofisticado y requiere mucho tiempo debido a las dificultades que surgen de la interacción entre el tubo de acero y el relleno de hormigón. Esto condujo al desarrollo de expresiones predictivas breves de un solo paso que pronostican con precisión la reacción estructural de las columnas utilizando técnicas de aprendizaje automático inspiradas en la naturaleza. Los resultados de 3103 experimentos disponibles en columnas tubulares de acero rellenas de hormigón durante los últimos años se utilizaron para construir y validar estas expresiones (Naser et al., 2020).

El modo de falla y la capacidad de carga de las columnas de concreto reforzado son consideraciones importantes en el diseño estructural y los métodos de evaluación del desempeño, según el ingeniero civil De-Cheng Feng et al. en 2020. Por lo tanto, al utilizar técnicas de aprendizaje automático en conjunto, proporcionan una solución inteligente para la categorización del modo de falla y la predicción de la capacidad de carga de las columnas de hormigón armado. La capacidad de carga proyectada también se comparó con la prevista por las ecuaciones empíricas de los códigos de diseño. En esta era de grandes volúmenes de datos, las técnicas de aprendizaje automático, en particular el aprendizaje conjunto, pueden ofrecer una alternativa a los modelos tradicionales impulsados por la mecánica en el diseño estructural (Feng et al., 2020).

Los criterios del código para el diseño y construcción de hormigón estructural que son necesarios para garantizar la seguridad pública se presentan en ACI 318 (Instituto Americano del Concreto). El proceso de determinar la reacción de una estructura bajo condiciones de carga impuestas o combinaciones de varias cargas se conoce como análisis estructural. La resistencia, rigidez, estabilidad y respuesta de todos los elementos se estudian durante el análisis de una estructura.

Los métodos analíticos y numéricos se pueden utilizar para resolver problemas. Una solución analítica implica presentar el problema de manera clara y calcular la solución precisa. Una solución numérica implica hacer suposiciones informadas sobre la solución y determinar si el problema está lo suficientemente resuelto como para permitir detenerse. En nuestro caso, se utilizó un enfoque analítico para el diseño de la base cuadrada aislada para recopilar rápidamente una cantidad significativa de datos y luego utilizar el aprendizaje automático.

## **MATERIALES Y METODOS**

### **RECOLECCION DE DATOS**

WEKA se utilizó como un software de minería de datos para clasificar la información, evaluarla y determinar la precisión de las predicciones de varios algoritmos de minería de datos. En este experimento se investigó la exactitud de muchos algoritmos y se descubrió el algoritmo más apropiado con la mayor exactitud de clasificación.

El experimento se llevó a cabo utilizando una hoja de cálculo de Microsoft Excel para generar un conjunto de datos de diseño de zapata aislada cuadrada. La herramienta WEKA, creada por el Grupo de Aprendizaje Automático de la Universidad de Waikato en Hamilton, Nueva Zelanda, se utilizó para aplicar varios algoritmos de clasificación en este conjunto de datos. Para la categorización de zapatas cuadradas aisladas en diseños estructurales aceptables e insatisfactorios según el estándar ACI 318-11, el conjunto de datos utilizado para el análisis tiene un total de 518400 casos, un total de 11 atributos, de los cuales 10 atributos son numéricos y 1 atributo es nominal. La herramienta WEKA realiza la preparación de datos de forma automática.

La Tabla 1 describe brevemente los atributos del conjunto de datos.

Tabla 1. Base de datos de pies cuadrados aislados.



No.	Atributo utilizado	Tipo de atributo	Descripción del atributo
1	Pu	Numérico	Carga última
2	qa	Numérico	Capacidad de carga permitida
3	Gammasite	Numérico	peso específico del suelo
4	Bpedestal	Numérico	Dimensión del pedestal
5	Apedestal	Numérico	Dimensión del pedestal
6	fc	Numérico	Fuerza compresiva
7	Df	Numérico	Profundidad
8	H	Numérico	Espesor
9	B	Numérico	Base
10	As	Numérico	Acero de refuerzo
11	Conclusión	Nominal	Diseño adecuado o inadecuado

La Tabla 2 muestra los atributos y un extracto de los datos seleccionados para crear los modelos.

Tabla 2. Muestra de conjunto de datos.

Pu (kN)	qa (kPa)	Gammasite (kN/m <sup>3</sup> )	Bpedestal (m)	Apedestal (m)	fc (MPa)	Df (m)	H (m)	B (m)	As (squarecm)	Conclusión
1	100	10	0.2	0.2	20	1	0.2	0.3	1.199	OK
1	102	10	0.2	0.2	20	1	0.2	0.3	1.199	OK
1	105	10	0.2	0.2	20	1	0.2	0.3	1.199	OK
1	120	10	0.2	0.2	20	1	0.2	0.3	1.199	OK
100	185	30	0.25	0.25	20	3	0.4	1.1	11.724	OK
100	190	30	0.25	0.25	20	3	0.4	1.1	11.724	OK
100	245	30	0.25	0.25	20	3	0.4	0.9	9.593	OK
100	250	30	0.25	0.25	20	3	0.4	0.9	9.593	OK
100	340	30	0.25	0.25	20	3	0.4	0.7	7.461	OK
180	100	15	0.25	0.25	25	4	0.5	2.5	34.973	OK
180	102	15	0.25	0.25	25	4	0.5	2.4	33.574	OK
180	105	15	0.25	0.25	25	4	0.5	2.3	32.175	OK
180	120	15	0.25	0.25	25	4	0.5	1.9	26.58	OK
180	125	15	0.25	0.25	25	4	0.5	1.9	26.58	OK
210	100	30	0.25	0.25	20	3	0.2	5.5	37.965	REDISEÑO
210	102	30	0.25	0.25	20	3	0.2	4.8	32.637	REDISEÑO
210	105	30	0.25	0.25	20	3	0.2	4.2	28.076	REDISEÑO
210	120	30	0.25	0.25	20	3	0.2	2.8	17.467	REDISEÑO
210	125	30	0.25	0.25	20	3	0.2	2.6	15.958	REDISEÑO
320	150	15	0.3	0.3	32	3	0.2	1.8	14.122	REDISEÑO

320	185	15	0.3	0.3	32	3	0.2	1.6	11.901	REDISEÑO
320	190	15	0.3	0.3	32	3	0.2	1.6	11.901	REDISEÑO
320	245	15	0.3	0.3	32	3	0.2	1.3	8.621	REDISEÑO
320	250	15	0.3	0.3	32	3	0.2	1.3	8.621	REDISEÑO

## SELECCIÓN DEL MODELO

Para seleccionar el modelo ideal se inicia el proceso en la opción de exploración de la interfaz gráfica de usuario de WEKA, en la pestaña de preproceso se cargan los datos mediante un archivo con extensión ARFF. Al cargar el archivo se muestra el número de atributos, casos y datos estadísticos, máximos, mínimos, media y desviación estándar para cada uno de los atributos, además de la visualización de cada atributo en forma gráfica. En esta pestaña el usuario puede eliminar atributos si los considera redundantes. Luego, el siguiente paso fue seleccionar la pestaña de clasificación para elegir los algoritmos que crearán los modelos, en nuestro caso: J48, AdaBoostM1, logístico y Naïve Bayes. Para probar los modelos, el usuario puede seleccionar todos los datos preprocesados, cargar otros datos además de los preprocesados, realizar una validación cruzada o dividir los datos preprocesados en un porcentaje de entrenamiento y un porcentaje de prueba. Luego se selecciona el atributo para el cual se probará el modelo, se inicia su creación y luego se usa el modelo con el mejor desempeño para hacer predicciones como se muestra en la Fig. 1.

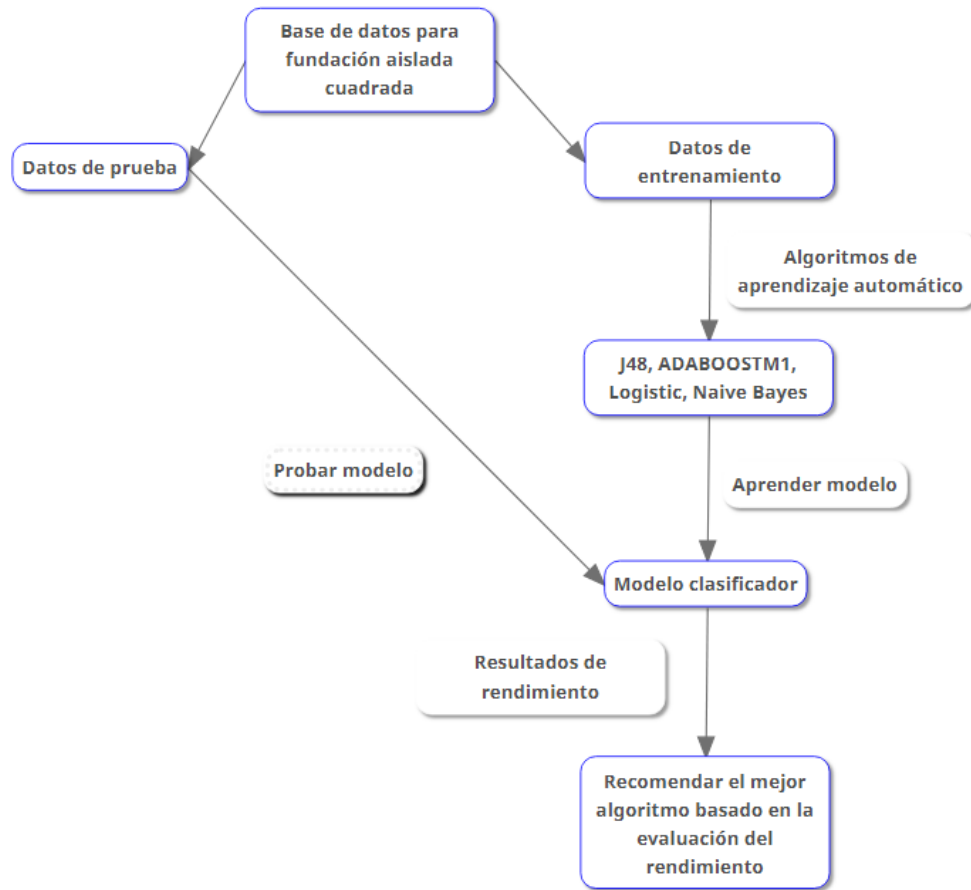


Fig. 1 Diagrama de flujo de la metodología propuesta.

Se utilizó validación cruzada con 10 pliegues para entrenar y probar los modelos en nuestro caso. Los datos de muestra se separan en K subgrupos en la validación cruzada de K iteraciones. Una fracción se utiliza como datos de prueba, mientras que las otras se utilizan como datos de entrenamiento. Con cada uno de los diversos subconjuntos de datos de prueba, el proceso de validación cruzada se repite para k ensayos. En última instancia, la media aritmética de los resultados de cada iteración se calcula para obtener un único resultado.

Para evaluar el rendimiento de los algoritmos, se utilizaron las siguientes medidas:

Como se indica en la ecuación 1, la precisión es la relación entre la cantidad de casos positivos esperados y la cantidad total de casos positivos anticipados.

Como se indica en la ecuación 2, la recuperación es la relación entre la cantidad de casos positivos proyectados y la cantidad total real de casos positivos.

Como se indica en la ecuación 3, la exactitud es la relación de los números de casos reales esperados positivos y negativos con respecto a la cantidad total de casos.

Como se indica en la ecuación 4, la tasa de TP es la relación entre la cantidad de casos positivos proyectados y la cantidad total real de casos positivos.

Como se indica en la ecuación 5, la tasa de FP es la relación entre la cantidad de casos negativos proyectados y la cantidad total real de casos negativos.

El rendimiento general está representado por la variable F-Medida. Como se demuestra en la ecuación 6, es la media armónica ponderada de precisión y recuperación (Mokgonyane et al., 2019).

$$\text{Precisión} = \frac{\text{Verdadero positivo}}{\text{Verdadero positivo} + \text{Falso positivo}} \quad (1)$$

$$\text{Recuperación} = \frac{\text{Verdadero positivo}}{\text{Verdadero positivo} + \text{Falso positivo}} \quad (2)$$

$$\text{Exactitud} = \frac{\text{Verdadero positivo} + \text{Verdadero negativo}}{\text{Número total de casos}} \quad (3)$$

$$\text{TP - tasa} = \frac{\text{Verdadero positivo}}{\text{Verdadero positivo} + \text{Falso negativo}} \quad (4)$$

$$\text{FP - tasa} = \frac{\text{Falso positivo}}{\text{Falso positivo} + \text{Verdadero negativo}} \quad (5)$$

$$\text{F - medida} = \frac{2 * \text{Precisión} * \text{Recuperación}}{\text{Precisión} + \text{Recuperación}} \quad (6)$$

Los resultados obtenidos tras el entrenamiento del modelo con los algoritmos se indican en las Tablas 3, 4 y la Fig. 2.

Tabla 3. Comparación de algoritmos de minería de datos para validación cruzada estratificada de 10 veces (precisión, tasa de recuperación de TP, tasa de FP y medida de F).

Algoritmo	Clase OK					Clase REDISEÑAR				
	Precisión	Recuperación	TP-tasa	FP-tasa	F-medida	Precisión	Recuperación	TP-tasa	FP-tasa	F-medida
J48	1	1	1	0	1	1	1	1	0	1
Naive Bayes	0.899	0.992	0.992	0.32	0.943	0.966	0.68	0.68	0.008	0.798
Logistic	0.937	0.985	0.81	0.19	0.961	0.951	0.81	0.81	0.015	0.875
AdaBoostM1	0.88	0.984	0.984	0.383	0.929	0.93	0.617	0.617	0.016	0.742

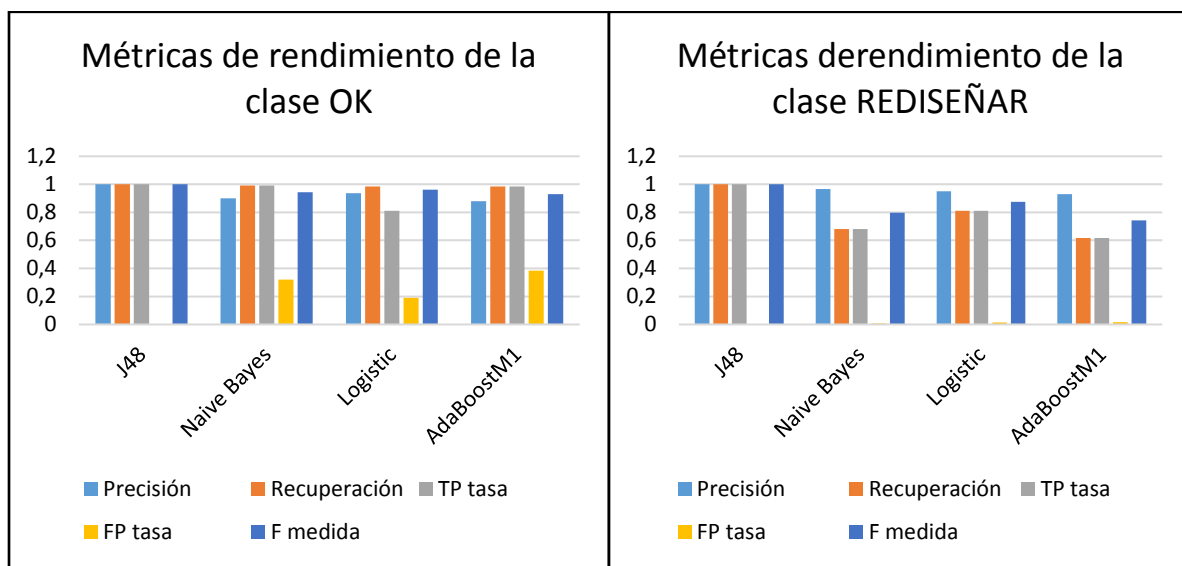


Fig. 2. Valores de medida de rendimiento del clasificador.

Tabla 4. Comparación de algoritmos de minería de datos para validación cruzada estratificada de 10 veces (exactitud).

Algoritmo	Clasificación correcta		Clasificación incorrecta		Tiempo de entrenamiento (s)
	No. de casos	Exactitud %	No. de casos	Inexactitud %	
J48	518387	99.9975	13	0.0025	41
Naive Bayes	472165	91.0812	46235	8.9188	2
Logistic	487330	94.0066	31070	5.9934	14

J48 y AdaBoostM1, con 41s y 52s respectivamente, son los algoritmos que emplean mayores recursos informáticos y por tanto tardan más en entrenar el modelo. J48 y Logistic, con un 99 % y un 94 % de exactitud, son los algoritmos más exactos. El algoritmo de peor rendimiento, Naive Bayes, también consume la menor cantidad de recursos de la computadora. Debido a que el algoritmo logístico tiene la segunda mejor exactitud y tiempo de entrenamiento, puede ser la opción ideal para un conjunto de datos más grande. En nuestra situación, elegimos el método J48 porque el tiempo de entrenamiento del modelo es rápido y su exactitud es la mejor.

J48 es un árbol de decisiones y se utiliza en el modelado predictivo para identificar métodos que distribuyen un conjunto de datos de acuerdo con ciertas variables. El árbol de decisiones es una organización similar a un diagrama de flujo con nodos y ramas. Cada nodo representa un atributo que será clasificado en el grupo. Teniendo en cuenta que el valor del nodo se expresa como ramas distintas (Chandrasekar et al., 2017).

## RESULTADOS DE PREDICCIÓN

La Tabla 5 a continuación presenta algunos de los datos con los cuales se utilizó el algoritmo J48 para predecir si el diseño es adecuado de acuerdo con el estándar ACI 318-11.

Tabla 5. Datos para la predicción del algoritmo.

Pu (kN)	Qa (kPa)	Gammasite (kN/m <sup>3</sup> )	bpedestal (m)	apedestal (m)	fc (MPa)	df (m)	h (m)	B (m)	As (cm <sup>2</sup> )	Conclusión
45	187	12	0.2	0.2	20	1.5	0.2	0.6	2.398	OK
65	116	18	0.25	0.25	20	2.5	0.2	1	3.997	OK
330	400	14	0.2	0.2	20	1.7	0.25	1	5.662	REDISEÑAR
280	137	22	0.3	0.2	32	1.7	0.2	1.8	14.056	REDISEÑAR
115	370	24	0.3	0.25	32	1.7	0.2	0.6	2.451	OK

270	200	11	0.3	0.25	25	1.7	0.4	1.3	13.856	OK
400	270	23	0.3	0.25	25	3.2	0.45	1.5	18.486	OK
412	103	23	0.3	0.25	20	3.4	0.2	4.4	63.359	REDISEÑAR
324	124	23	0.25	0.2	25	2.7	0.2	2.4	23.843	REDISEÑAR
570	195	23	0.25	0.2	32	1.6	0.24	2	24.683	REDISEÑAR

Los resultados de la predicción se muestran en la tabla 6 y se comparan con los calculados analíticamente.

Tabla 6. Predicciones en el conjunto de pruebas del usuario.

No. de caso	Actual	Predicho
1	OK	OK
2	OK	OK
3	REDISEÑAR	OK
4	REDISEÑAR	REDISEÑAR
5	OK	OK
6	OK	OK
7	OK	OK
8	REDISEÑAR	REDISEÑAR
9	REDISEÑAR	REDISEÑAR
10	REDISEÑAR	REDISEÑAR

Un caso de cada diez no se pronosticó correctamente, esta es la instancia número 3. La conclusión pronosticada fue que el diseño es correcto, cuando en realidad se necesita un rediseño. Por lo tanto, esta metodología debe ser utilizada como apoyo para agilizar la elección de las dimensiones de la cimentación cuadrada aislada y luego proceder a verificar la predicción realizada por el algoritmo.

## RESUMEN Y CONCLUSIONES

Se utilizó la metodología de aprendizaje automático para predecir si el diseño de una cimentación de hormigón cuadrada aislada es adecuada según la norma ACI 318-11. Para ello se aplicó el software WEKA, que contiene una amplia selección de algoritmos de aprendizaje automático y

permite tanto entrenar como probar un modelo a partir de los datos de entrada. Se eligieron cuatro algoritmos: Naive Bayes, Logistic, J48 y AdaBoostM1, para comparar su desempeño y seleccionar el más adecuado. En nuestro caso, el algoritmo con mejor precisión y tiempo de entrenamiento fue el J48. Se suministró al programa un conjunto de datos que no estaba originalmente en la base de datos y se utilizó el modelo entrenado para predecir si el diseño es adecuado. Hubo 1 error en la predicción y 9 respuestas correctas. Se concluye que la inteligencia artificial, específicamente el aprendizaje automático, permite, a través de la incorporación de un conjunto de datos, predecir si el diseño de una zapata cuadrada aislada de hormigón armado tiene un diseño adecuado con una exactitud aceptable.

## REFERENCIAS

- Chandrasekar P., Qian K., Shahriar H. and Bhattacharya P. (2017). Mejora de la exactitud de predicción de la minería de árboles de decisión con preprocesamiento de datos. 41.<sup>a</sup> Conferencia anual de software y aplicaciones informáticas de IEEE, 481-482.
- Enciclopedia Británica. (2020). Charles Babbage. Disponible en el sitio Web: <https://www.britannica.com/biography/Charles-Babbage> (Último acceso: 15/07/2021).
- Feng D., Liu Z., Wang X., Jiang Z. and Liang S. (2020). Clasificación del modo de falla y predicción de la capacidad portante para columnas de hormigón armado basadas en algoritmos de aprendizaje automático de conjuntos. Ingeniería Informática Avanzada, (45), 1-3.



- Greco L. (2018). Aprendizaje automático y técnicas de optimización para conexiones de acero. Actas del Simposio IASS 2018 Creatividad en diseño estructural, 1-2.
- Holmes G., Donkin A., and Witten I. (1994). WEKA: un banco de trabajo de aprendizaje automático. Actas de ANZIIS'94-Conferencia de sistemas de información inteligente de Australia y Nueva Zelanda IEEE, 357.
- McCarthy J., Minsky M., Rochester N. and Shannon C. (1956). Una propuesta para el proyecto de investigación de verano de Dartmouth. Revista AI, 27(4), 12.
- Mokgonyane T., Sefara T., Modipa T., Manamela M. and Manamela P. (2019). Sistema de reconocimiento automático de locutores basado en algoritmos de aprendizaje automático. Conferencia SAUPEC/RobMech/PRASA, 144-145.
- Naser M.Z., Thai S. and Thai H. (2020). Evaluación de la respuesta estructural de columnas tubulares de acero rellenas de hormigón a través del aprendizaje automático. Revista de Ingeniería de la Construcción, 1-3.
- Samuel A. (1959). Algunos estudios en aprendizaje automático utilizando el juego de damas. Revista de investigación y desarrollo de IBM, 3(3), 211-212.
- Sun H., Burton H. and Honglan H. (2020). Aplicaciones de aprendizaje automático para el diseño estructural de edificios y la evaluación del rendimiento: 2 Revisión del estado del arte. Revista de Ingeniería de la Construcción, 1-8.

Taylor R. (1843). Memorias científicas seleccionadas de las transacciones de academias extranjeras de ciencia y sociedades científicas y de revistas extranjeras, R. y JE Taylor, (3), 722.

Turing A. (1950). Maquinaria informática e inteligencia, Cambridge, MA: MIT Press, 450-451.