Una herramienta de clasificación para artículos no citados aplicada a la Revista de la Facultad de Ingeniería de la UdeA –Redin

Santiago Gallego Zapata

Trabajo de grado presentado para optar al título de Ingeniero Industrial

Asesor
Juan G.Villegas, PhD

Universidad de Antioquia
Facultad de Ingeniería
Ingeniería Industrial
Medellín, Antioquia, Colombia
2023

Cita	(Gallego Zapata, 2023)
Referencia **Estilo APA 7 (2020)**	Gallego Zapata, S. (2023). *Una herramienta de clasificación para artículos no citados aplicada a la Revista de la Facultad de Ingeniería de la UdeA –Redin* [Trabajo de grado profesional]. Universidad de Antioquia, Medellín, Colombia.

Grupo de Investigación Analítica e Investigación para la Toma de Decisiones (ALIADO). .

Dedicatoria

A mi familia por el apoyo que me ha brindado en todos sus sentidos a lo largo de toda mi carrera y a los compañeros de estudio que me han acompañado y dejado su grano de arena.

Agradecimientos

Agradecimientos especiales para Ana Cristina Salazar Zapata (Estudiante de ingeniería Industrial) por apoyo y colaboración en la formulación y realización del proyecto. Al equipo editorial de la revista Redin, en particular Sandra Hernández por apoyarnos constantemente con su retroalimentación y la información necesaria.

Tabla de contenido

Lista de tablas

Lista de figuras

Siglas, acrónimos y abreviaturas

API	Application Programming Interfaces
Csv	Comma-Separated Values
CV	k-fold Cross-validation
Dataset	Conjunto de datos
DOI	Digital Object Identifier
Issue	Edición de la revista a la que pertenece el artículo
KEY	Variable llave (para realizar fusión de *Datasets*)
KNC	K-Neighbors Classifier
Keywords	Palabras clave
LR	Logistic Regression
ML	Machine Learning
P	Precision
R	Recall
RFC	Random Forest Classifier
Redin	Revista la Facultad de Ingeniería
SKB	sklearn.feature_selection.SelectKBest
SFM	sklearn.feature_selection.SelectFromModel
WoS	Web of Science

Resumen

En este trabajo se aborda la problemática del bajo índice de citación de la Revista Facultad de Ingeniería (Redin) de la Universidad de Antioquia. La baja citación afecta la visibilidad y el prestigio de la revista y puede disminuir el interés de los investigadores en publicar allí. Por esta razón, se construyó una herramienta basada en modelos de *Machine Learning* (ML) que permitiera predecir si un nuevo artículo será o no citado. Se utilizó la metodología de investigación en diseño en conjunto con la metodología para la minería de datos *Cross-Industry Standard Process for Data Mining* (CRISP-DM), las cuales permitieron el diseño y desarrollo de la herramienta a través de la recopilación, limpieza, entendimiento y preparación de los datos, además de la modelación, evaluación y despliegue de modelos de ML. Se encontró que el modelo *Random Forest Classifier* (RFC) ofrece un mejor desempeño en identificar los artículos que no serán citados, al tiempo que permite identificar los factores que influyen en la citación de los artículos. Se propone a Redin utilizar la herramienta para tomar decisiones encaminadas a aumentar la probabilidad de citación de los artículos que presentan un mayor riesgo.

Palabras clave: análisis de citación. cienciometría. aprendizaje automático. inteligencia artificial

Abstract

This paper addresses the problem of the low citation rate of the *Revista Facultad de Ingeniería de la Universidad de Antioquia* (Redin). Low citation affects the visibility and prestige of the journal and may decrease the interest of researchers in publishing there. For this reason, a tool based on Machine Learning (ML) models was built to predict whether a new article will be cited. The design research methodology was used in conjunction with the Cross-Industry Standard Process for Data Mining (CRISP-DM) data mining methodology, which allowed the design and development of the tool through the collection, cleaning, understanding and preparation of data, as well as modeling, evaluation and deployment of ML models. It was found that the Random Forest Classifier (RFC) model offers a better performance in identifying the articles that will not be cited, while allowing the identification of the factors that influence the citation of the articles. It is proposed that the journal use the tool to make decisions aimed at increasing the probability of citation of the articles that present a higher risk.

*Keywords***:** citation analysis. scientometrics. machine learning. artificial intelligence

Introducción

La Revista Facultad de Ingeniería (Redin) de la Universidad de Antioquia busca consolidarse como una publicación académica de renombre en el ámbito científico y académico, promoviendo la difusión de los avances en investigación nacionales e internacionales. Sin embargo, uno de los principales desafíos que enfrenta la revista es su bajo índice de citación, lo que indica la necesidad de mejorar su impacto y prestigio en la comunidad científica. En el año 2021, Redin obtuvo un índice de citación bajo que lo ubica en el cuartil 3 (Q3) según el indicador SJR de Scopus. A pesar de haber adoptado prácticas como el acceso abierto y la migración al inglés, y de estar indexada en múltiples bases de datos de renombre, Redin aún no ha logrado clasificarse en un cuartil superior en su historia, lo que limita su capacidad de atraer investigadores y obtener citas en sus publicaciones.

En este contexto, el presente estudio tiene como objetivo crear una herramienta de clasificación y alerta temprana para clasificar artículos no citables utilizando indicadores cienciométricos y algoritmos de aprendizaje automático. Este modelo permitirá identificar los factores más relevantes que influyen en la cantidad de citas y clasificar las publicaciones de Redin como citables o no citables, lo que proporcionará un instrumento valioso para el equipo editorial de la revista en su búsqueda de mejorar su índice de citación y prestigio en la comunidad científica.

Lo que resta del documento se organiza de la siguiente forma: primero se definen los objetivos del trabajo, seguido de una revisión de literatura, donde se exploran los conceptos, metodologías y teorías consideradas, así como el estado del arte que muestra la forma como se ha dado solución a la problemática en contextos similares. Luego se expone la metodología empleada para finalmente mostrar y analizar los resultados obtenidos y concluir sobre el trabajo.

Objetivos

Objetivo General

Crear una herramienta de alerta temprana para clasificar artículos no citados para la Revista de la Facultad de Ingeniería de la UdeA – Redin.

Objetivos Específicos

- Revisar cómo se ha abordado la problemática de baja citación en la literatura, estudiando el desarrollo en la investigación de herramientas de alerta temprana e identificando las causas que originan un bajo índice de citación para las revistas científicas.
- Diseñar una base de datos de prueba para la construcción de la herramienta y aplicar modelos de aprendizaje automático para analizar la citación.
- Construir y validar la herramienta de alerta temprana utilizando modelos de aprendizaje de máquina.
- Validar en el caso de estudio, analizando los resultados de cada modelo y evaluando los modelos de ML empleados para elegir el de mejor desempeño.
- Crear una herramienta en línea que pueda ser utilizada por el equipo de la revista en la detección de los artículos no citados.

Marco Teórico

El índice de citación es un indicador importante de la calidad de las revistas científicas y se basa en el número de veces que se citan los artículos publicados en la revista en relación con el número total de artículos publicados. Este índice es un buen indicador de la relevancia de la revista a nivel mundial. Sin embargo, existen otras métricas utilizadas por las indexadoras, como Web of Science (WoS) y Scopus, que clasifican las revistas en cuartiles según su ubicación en relación a otras revistas de la misma área de conocimiento. Los cuartiles se dividen en cuatro grupos, siendo el cuartil 1 (Q1) las revistas de mayor percentil que superan el 75% y el cuartil 4 (Q4) aquellas revistas que su percentil está por debajo del 25%.(Marín Velásquez & Arriojas Tocuyo, 2021)

En el año 2021, el indicador de Scopus, SCImago Journal Rank (SJR) otorgó a la revista un SJR de 0.206 debido a su escaso índice de citación, ya que solo 78 de los 207 documentos que publicó fueron citados. Esta baja puntuación la ubica en el cuartil 3 (Q3), lo que indica que su percentil oscila entre 25% y 50%. (Revista Facultad de Ingenieria, s. f.) Aunque Redin ha adoptado la práctica del acceso abierto y la migración al idioma inglés, y ha conseguido el logro de ser indexada en múltiples bases de datos de renombre, tales como SCOPUS, LATINDEX, SciFinder, Web of Science ESCI, Redalyc, CAplus, EBSCO host, J-Gate Plus, Scielo Colombia, PROQUEST CSA, Elektronische Zeitschriftenbibliothek EZB, Cengage Learning Inc., Copernicus, y Red Colombiana de Revistas de Ingeniería (RCRI) (Vargas, 2021) todavía no ha podido clasificarse en un cuartil superior en su historia. Lo cual tiene un impacto directo en Redin, ya que cuanto menor sea la clasificación de la revista, menor será el interés de los investigadores por publicar en ella. Además, se limita la influencia de las citas en sus publicaciones y su productividad, lo que afecta negativamente su divulgación y prestigio.

Por otro lado, el uso del aprendizaje automático en el análisis de citas ha cobrado importancia en la cienciometría. La cienciometría implica el estudio cuantitativo de las publicaciones, mientras que el aprendizaje automático se enfoca en la creación y análisis de algoritmos capaces de aprender de los datos.(Ibáñez Martín, 2015) La aplicación de ambas disciplinas permitirá crear una herramienta valiosa para el equipo editorial de Redin, ya que se podrán identificar los factores que influyen en la cantidad de citas de las publicaciones y clasificarlas como citables o no citables. Proporcionando así un instrumento útil para mejorar la medición del impacto de la revista Redin y por tanto mejorar el índice de citación.

En la literatura se han utilizado diferentes técnicas de aprendizaje automático, como la regresión logística regularizada, la clasificación con algoritmos de aprendizaje automático como perceptrón multicapa (MLP), Regresión Logística (LR), máquinas de vectores de soporte (SVM) y Naive Bayes (NB). Estas técnicas han permitido predecir el número de citas que recibirá un artículo, lo que es un indicador de relevancia en la comunidad científica.

Algunos estudios identificaron características que influyen en la cantidad de citas de un trabajo utilizando la base de datos Web of Science (WoS).(BinMakhashen & Al-Jamimi, 2022) agruparon las características en previas y posteriores a la difusión del trabajo, y encontraron que el número de autores, la información de financiación de la investigación, la colaboración internacional y el cuartil de la revista son las características más relevantes para predecir el número de citas. Además, (Abrishami & Aliakbary, 2019) propusieron un método novedoso para predecir el número de citas a largo plazo, donde utilizaron la cantidad de citas en los primeros años después de la publicación y encontraron que su método resultó ser más efectivo que los métodos de predicción más avanzados.

Por otro lado,(Himani et al., 2022), y (Qayyum et al., 2022) realizaron análisis comparativos sobre la predicción de citas para artículos y propusieron diversos modelos automáticos y semiautomáticos para clasificar las citas de los artículos científicos según su propósito utilizando la regresión lineal múltiple, la clasificación con algoritmos de aprendizaje automático y las técnicas de procesamiento del lenguaje natural (NLP) para predecir el número de citas. Estos autores señalan que su resultado puede tener implicaciones importantes para la toma de decisiones en la investigación científica.

Los modelos de aprendizaje automático también se han utilizado en otros ámbitos de la investigación científica, como la predicción de la renovación de becas y la identificación de características que predicen un alto volumen de citas en la literatura radiológica. (Tohalino & Amancio, 2022) utilizaron modelos de aprendizaje automático para predecir si una beca de investigación otorgada a un investigador es renovable o no, utilizando un modelo de LR y SVM. Por su parte, (Rosenkrantz et al., 2016) utilizaron un modelo de regresión logística regularizada para identificar las características de los artículos que predicen un alto volumen de citas en la literatura radiológica.

Existen estudios que identifican factores medibles que influyen en la citación de artículos. En particular, (Ha, 2022) identificó 14 factores medibles que influyen en la citación de artículos y

señaló que el año y la fuente de publicación son los más importantes. Sin embargo, otros autores han propuesto la identificación de un mayor número de factores medibles, como la relevancia de sus contenidos, su accesibilidad, su difusión y su autoridad científica, la cantidad de caracteres, la cantidad de autores y la cantidad de vistas. Además, se destaca la importancia de contar con datos etiquetados para entrenar modelos de aprendizaje automático que identifiquen citas importantes en los artículos científicos, así como de los datos no etiquetados en la estrategia de auto-entrenamiento de los modelos.

Por otra parte, se hace referencia a la necesidad de considerar el área bajo la curva de precisión-recall (AUC-PR) al trabajar con conjuntos de datos con distribuciones extremadamente sesgadas y se sugiere explorar el potencial de modelos de aprendizaje profundo con estrategias semi-supervisadas para la identificación de citas importantes. (An et al., 2022); (Repiso et al., s. f.); (Elgendi, 2019);(Dias et al., 2023).

Es así como esta breve revisión literaria permite distinguir múltiples investigaciones que han identificado características importantes para la predicción del número de citas de un artículo en la comunidad científica, lo que resalta la necesidad de utilizar técnicas de aprendizaje automático para obtener predicciones precisas. En este sentido, la exploración realizada conduce al desarrollo de una herramienta más precisa y particularmente adaptada a la revista Redin considerando como posibles opciones los métodos de predicción y las variables que se han usado previamente en la literatura, tal como se resumen en la **Tabla 1**.

Tabla 1
Revisión de literatura

Autores	Características relevantes	Variable objetivo	Algoritmos
BinMakhashen y Al-Jamimi (2022)	- Número de autores - Colaboración internacional - Financiación de la investigación - La revista de publicación	Artículo altamente citado o no (Binaria)	SVM DT RF AC NB
Ali Abrishami, Sadegh Aliakbary (2019)	- Identificador del trabajo - La revista de publicación - Año de publicación	Cantidad de citas	RNN
Himani et al (2022)	- Número de palabras en el título	Cantidad de citas	MLR

	- Año de publicación - Tipo de publicación		GBR SVM
Qayyum et al (2022)	- Número de referencias - Número de palabras en el título - Resumen - Keywords	Clasificación de citas importantes/incidental	MLP SVM LR MNB
Tohalino y Amancio (2022)	- Tema de investigación - Número de autores - Afiliación del investigador	Beca exitosa (Binaria)	SVM MLP KNN NB DT
Rosenkrantz et al (2016)	- La revista de publicación - Tipo de publicación	Artículo altamente citado o no (Binaria)	NB SVM BBR
Taehyun Ha (2022)	- Tipo de publicación - Año de publicación - Número de referencias - Acceso abierto - Número de autores	Cantidad de citas	XGB LGBM CB
Xin An, Xin Sun, Shuo Xu (2022)	- Año de publicación - Número de autores - Resumen - Número de referencias	Clasificación de citas importante/incidental	SVM RF
Elgendi (2019)	- Número de caracteres - Número de autores - Número de vistas - Número de palabras en el título	Artículo altamente citado o no (Binaria)	PCA
Dias et al (2023)	- Año de publicación - Número de páginas - Keywords		

Los resultados encontrados en la tabla indican que varios estudios de investigación han utilizado diversos algoritmos de aprendizaje automático. Sin embargo, se observa que los algoritmos más comúnmente utilizados son SVM, LR, NB y MLP para analizar distintas características relevantes en sus investigaciones, basándose en las variables objetivo establecidas, ya sean binarias o cuantitativas-regresoras. Además, estos estudios comparten algunas

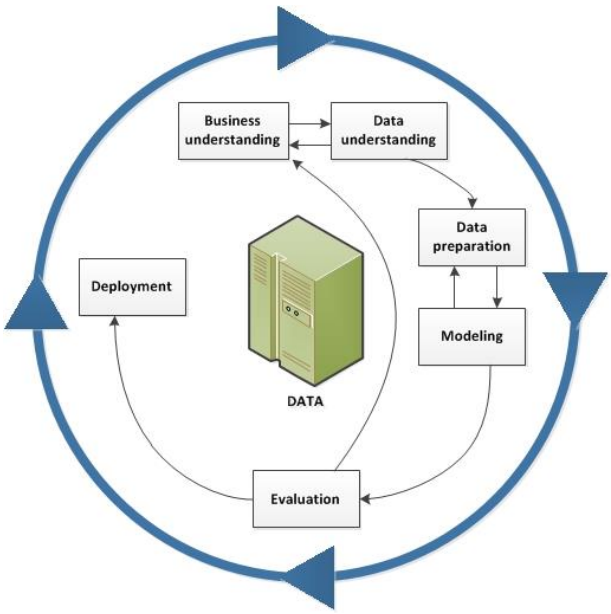
características similares que resultaron relevantes en sus análisis, como el año de publicación, el número de autores, el número de referencias y los caracteres en el título. Por lo tanto, para el desarrollo de este estudio y la creación de una herramienta de clasificación precisa, es importante basarse en los hallazgos obtenidos de investigaciones anteriores, que incluyen las características de mayor influencia y los algoritmos con mayor precisión.

Metodología

Para la creación de la herramienta se utilizó la metodología de Investigación en Diseño propuesta por Peffers et al. (2007) en conjunto con la metodología para la minería de datos *Cross-Industry Standard Process for Data Mining* (CRISP-DM) (Chapman et al, 1999). La primera está orientada a la creación de herramientas o artefactos que ayuden a solucionar problemas organizacionales, que, en caso de la revista, se trata de una herramienta con interfaz web que clasifica los artículos que serán publicados según su riesgo de no ser citados, de manera que sirva como recurso para la toma de decisiones internas. Esta metodología se compone de seis etapas: (1) identificación del problema y motivación, (2) definición de objetivos, (3) diseño y desarrollo, (4) demostración, (5) evaluación y (6) comunicación.

Transversal a esta metodología, se utilizó CRISP-DM que es una metodología para la minería de datos muy utilizada en la industria y adecuada para la manipulación de información, creación y despliegue de modelos de ML. Contiene seis etapas: (1) comprensión del negocio, (2) comprensión de los datos, (3) preparación de los datos, (4) modelado, (5) evaluación y (6) despliegue. Estas fases se pueden observar en la **Figura 1**.

Figura 1
CRISP-DM Cross-Industry Standard Process for Data Mining



Nota. Fuente https://www.ibm.com/docs/es/spss-modeler/saas?topic=dm-crisp-help-overview (Conceptos básicos de ayuda de CRISP-DM).

A partir de la adaptación de estas dos metodologías, se siguieron 3 fases, que son descritas a continuación. Cabe resaltar que para todas las fases del proyecto se utilizó el lenguaje de programación Python en Google Colab, que es un entorno de ejecución ofrecido por Google, que permite ejecutar Python en el navegador, evitando el uso de recursos computacionales propios.

Fase 1: Identificación del Problema de Negocio y Definición de Objetivos

Para esta primera fase se identificó el bajo índice de citación como el problema en la revista, así mismo se definieron los objetivos de la solución con apoyo de los avances en investigación identificados en la literatura para contextos y problemas similares, así como también varias conversaciones con el equipo editorial orientadas a recoger sus expectativas o intereses como usuarios de la herramienta. Comprender este problema llevó a la necesidad de crear la herramienta de alerta temprana para identificar los artículos que no serán citados antes de su publicación.

Fue necesaria la traducción del problema del negocio a un problema analítico, es decir, en términos de minería de datos. En este sentido, se tradujo en la creación de un modelo de clasificación de ML que, a partir de diferentes variables que caracterizan los artículos que han sido publicados por la revista, predijera si un nuevo artículo será citado o no.

Fase 2: Diseño y Desarrollo de la Herramienta

El diseño y desarrollo de la herramienta es la segunda etapa, que comprende la construcción de un modelo de predicción basado en ML. Para llevar a cabo este desarrollo, teniendo en cuenta que en la primera fase se comprendió el problema del negocio, se procedió con la recopilación, entendimiento, limpieza y preparación de los datos, aplicación de técnicas de modelado, evaluación y despliegue de la herramienta.

Recopilación de los Datos

Se realizó la recopilación de diferentes conjuntos de datos (Datasets) de múltiples fuentes y con diferentes métodos con el fin de tener la mayor cantidad de características de los artículos.

Esto permitió un mejor entendimiento de la revista y de las características de sus artículos, así como un mejor ajuste de los modelos de ML; de igual forma, la recopilación se hizo teniendo en cuenta las variables identificadas en la revisión de literatura, ya que han mostrado un buen rendimiento en contextos similares. A continuación, la **Tabla 2** presenta cada uno de los Datasets recopilados, así como la fuente y el método de adquisición.

Tabla 2
Datasets recopilados, fuentes y métodos de recolección

Dataset	Fuente	Método de recolección	Descripción
Scopus_1.csv (df_1)	Base de Datos Scopus	Descarga manual del archivo *.csv (comma-separated values)* desde el perfil de la revista	Este *Dataset* contiene información de cada uno de los artículos de la revista desde 2008, año en el que fue indexada en la Base de Datos. Dentro de las variables más destacables, se encuentran: • Cantidad de citas recibidas • Año de publicación • Autores del artículo • Número de Issue (edición de la revista a la que pertenece el artículo) • Página de inicio y de fin del artículo dentro del Issue • *Digital Object Identifier* (DOI) del artículo • Afiliaciones de los autores, • Resumen • Palabras clave (Keywords) • Idioma • Tipo de documento • Tipo de acceso • Referencias
Scopus_2.csv (df_2)	Base de Datos Scopus	Descarga manual del archivo *.csv* desde el perfil de la revista	Este *Dataset*, al igual que el anterior, contiene información de cada uno de los artículos indexados en la Base de Datos. En este caso, contiene la cantidad de citas que recibió el artículo por cada año.

savedrecs.xls (df_3)	Base de Datos WoS	Descarga manual del archivo *.csv* desde el perfil de la revista	Este *Dataset* contiene información de los artículos, pero esta vez, los indexados en la Base de Datos de WoS. **Nota**: el Dataset fue descartado para el análisis actual ya que contiene información redundante con los Datasets *df_1* y *df_2*.
articles-redin-20221123.csv (df_4)	Personal de la revista	Compartido por el personal de la revista	El *Dataset* contiene información de los artículos que se han publicado: título, nombre de primer autor, disciplinas del artículo, keywords, fecha de publicación. **Nota**: el Dataset fue descartado para el análisis actual puesto que el objetivo inicial era obtener una variable que indicara la disciplina del artículo, sin embargo, se encontró que había demasiados niveles de disciplinas, lo que dificultaba su análisis al tratarse de una variable categórica. Adicionalmente, había cierto nivel de dificultad para crear una variable *KEY* que permitiera su fusión con los demás *Datasets*.
df_authors.csv (df_5)	Base de Datos Scopus – API	Web Scraping – *Application Programming Interfaces* (API)	*Dataset* extraído de la Base de Datos de Scopus, haciendo uso de la API disponible en el sitio, el cual incluye información de todos los autores que han publicado al menos una vez en la revista. Esta información incluye, entre otras variables: el índice h, cantidad de co-autores con los que ha trabajado, cantidad de publicaciones, cantidad de citas que ha recibido, afiliaciones a las que ha pertenecido, afiliación actual, dirección de las afiliaciones, áreas o temas de publicación, fecha de creación del perfil. **Nota**: el proceso realizado para la recolección de este Dataset se describe en el **Anexo 1**.
df_redin_scapy.csv (df_6)	Página web de la revista	Web Scraping	*Dataset* extraído de la página web de la revista, la cual contiene la cantidad de vistas del *abstract*, la cantidad de vistas del *pdf*, la cantidad de páginas , los autores y títulos de los artículos de la revista. **Nota 1**: El proceso de extracción se detalla en el **Anexo 2**. **Nota 2**: el Dataset fue descartado para el análisis actual ya que las vistas del *abstract* y *pdf* son datos que se generan luego de la publicación, por lo que no son variables predictoras útiles para el caso actual.
df_special_issue.csv (df_7)	Página web de la revista	Lectura de cada editorial	*Dataset* generado manualmente a partir de la lectura de cada editorial de la revista (64 editoriales), con el fin de identificar cuáles de los *Issues* publicados corresponden a un *Special Issue*.

Luego de realizar una exploración inicial, se dejan los *Datasets* df_3, df_4, y df_6 para futuros análisis, puesto que contienen variables que son redundantes con los demás *Datasets* o innecesarias para el objetivo del trabajo.

Entendimiento y Exploración de los Datos

Luego de recopilar todos los *Datasets* necesarios, fue necesario describir y explorar los datos con el fin de mejorar su comprensión y realizar una limpieza general. Algunas de las actividades consideradas dentro de esta exploración fueron:

- Revisar los tamaños de los Datasets
- Analizar y convertir los tipos de variables al más adecuado
- Identificar los rangos y niveles de cada variable
- Identificar datos faltantes y duplicados con el fin de definir la forma de imputarlos
- Estadísticas básicas (media, máximo, mínimo, desviación estándar)
- Visualizar la distribución de las variables
- Analizar la relación de las variables con la variable respuesta
- Decidir si la variable es relevante para el estudio

Preparación de los Datos

La preparación de los datos, unido con la exploración, son las etapas que normalmente consumen más tiempo en un proyecto de minería de datos puesto que es muy importante tener los datos correctos y en el formato adecuado para aumentar el rendimiento de los modelos. Esto no fue la excepción para el proyecto actual, puesto que aproximadamente el 50% del tiempo fue empleado en estas etapas. A continuación, se presentan algunas de las actividades realizadas en esta etapa.

- Selección de elementos (artículos) y atributos (variables) que serán utilizadas en la modelación
- Imputar los datos faltantes

- Construcción de nuevas variables a partir de los datos existentes
- Tratamiento de las variables *KEY* que permitan la fusión adecuada de los Datasets [1]
- Realizar la fusión de los diferentes Datasets considerados y construcción de Dataset final
- Transformación de los datos (Escalar variables numéricas y codificar variables categóricas), ya que algunos modelos lo requieren

El producto final de esta preparación fue un único *Dataset* con todas las variables necesarias y en el formato adecuado.

Modelado

En esta etapa se aplican diferentes modelos de clasificación, donde a partir de diferentes escenarios (combinando diferentes variables predictoras y variables de respuesta, así como diferentes modelos), se selecciona el que brinda una mejor medida de desempeño.

Se utilizó la librería *Sklearn* de Python, que ofrece una amplia gama de modelos y funcionalidades para la ciencia de datos.

Evaluación

Es importante considerar una evaluación al realizar los modelos de predicción. Una técnica ampliamente utilizada es la validación cruzada, en inglés *k-fold Cross-validation* (CV), la cual divide aleatoriamente el *Dataset* en *K* subconjuntos llamados *folds*, de manera que entrena el modelo *K* veces utilizando *K-1* subconjuntos y lo valida con el subconjunto restante (Géron, 2019). De esta forma se realiza una evaluación cruzada en todo el Dataset, mejorando el rendimiento del modelo.

[1] Para realizar la fusión entre df_1 y df_2 fue necesario:
1. Separar el título, ya que algunos tenían título en inglés y en español.
2. Crear una KEY con las primeras 4 palabras de cada título en inglés.
3. Incluir el Issue dentro del KEY para los títulos que coincidían en las primeras 4 palabras, para diferenciarlos.

Para unir el *Dataset* resultante con df_5 se consideraron dos KEY:
1. El primer autor del artículo (con el Scopus ID del autor)
2. El autor más citado (con el Scopus ID del autor)

Finalmente, para unir el *Dataset* resultante con df_7, se utilizó el Issue.

Fase 3: Despliegue y Comunicación

Se utiliza la librería *Streamlit* de Python para realizar el despliegue del modelo final, el cual permite ingresar las diferentes variables de un nuevo artículo y mostrar si el artículo será o no citado. Además de contener el modelo de predicción, incluye una herramienta de visualización para hacer un análisis descriptivo de algunas de las variables más importantes. A partir de las conversaciones con el equipo editorial de la revista se agregó un motor de búsqueda para que su usuario pueda buscar los artículos que están indexados en Scopus con el fin de generar reportes de los artículos y sus citas de manera más eficiente, ya que se identificó que al buscar algunos dentro de la Base de Datos de Scopus, no era posible encontrarlos, pero si estaban incluidos en la revista.

Resultados

Inicialmente, con el fin de tener una cantidad menor de variables en los modelos, sin afectar las medidas de desempeño, se utilizaron técnicas de selección de variables para tener diferentes conjuntos de datos, donde cada uno considere las variables más importantes. La primera de ellas es *sklearn.feature_selection.SelectKBest* (SKB), que es un método de selección de variables que selecciona las k variables más importantes tras realizar una prueba de *F-test* para estimar si hay dependencia entre cada una de las variables con la variable de respuesta; con este método se seleccionaron 50 variables. La segunda técnica utilizada es el método *sklearn.feature_selection.SelectFromModel* (SFM), el cual selecciona las variables más importantes luego de realizar una estimación, que en este caso, se utilizó un *Random Forest Classifier* (RFC) el cual consideró 36 variables como las más importantes. Para un tercer conjunto de datos, se consideraron todas las variables disponibles, es decir 239.

Luego de tener los diferentes conjuntos de datos se entrenaron diferentes modelos de clasificación con el fin de compararlos y seleccionar el que tenga un mejor rendimiento. Para el entrenamiento y la evaluación de los modelos se utilizó CV, el cual divide aleatoriamente el Dataset en pequeños subconjuntos (en este caso, en 20), de manera que entrena el modelo con 19 y lo valida con el subconjunto restante, repitiendo el proceso 20 veces. De esta forma se realiza una evaluación cruzada del rendimiento del modelo. Los modelos entrenados fueron *Logistic Regression* (LR), *Random Forest Classifier* (RFC) y *K-Neighbors Classifier* (KNC).

La **Tabla 3** muestra la exactitud (*Accuracy*), que es la proporción de predicciones correctas (Géron, 2019), la cual es obtenida tras realizar la evaluación de los modelos. De manera general, todos los escenarios otorgaron un *Accuracy* mayor a 64%, sin embargo, se puede apreciar que en todos los escenarios el RFC tiene el mejor desempeño, con 78% en el peor de los casos. Si bien el objetivo principal es alcanzar el máximo rendimiento posible (el cual se alcanza cuando se utilizan todas las variables), es importante considerar un modelo más parsimonioso, es decir, el que tenga un rendimiento similar con la menor cantidad de variables posibles. Por esta razón, el candidato a mejor modelo en este caso es el RFC con el conjunto de variables que brinda el método SFM. Más abajo en la **Tabla 10** se presentan las variables consideradas en este escenario de modelación.

Tabla 3
Precisión (Accuracy) en % de los modelos entrenados con diferentes conjuntos de variables

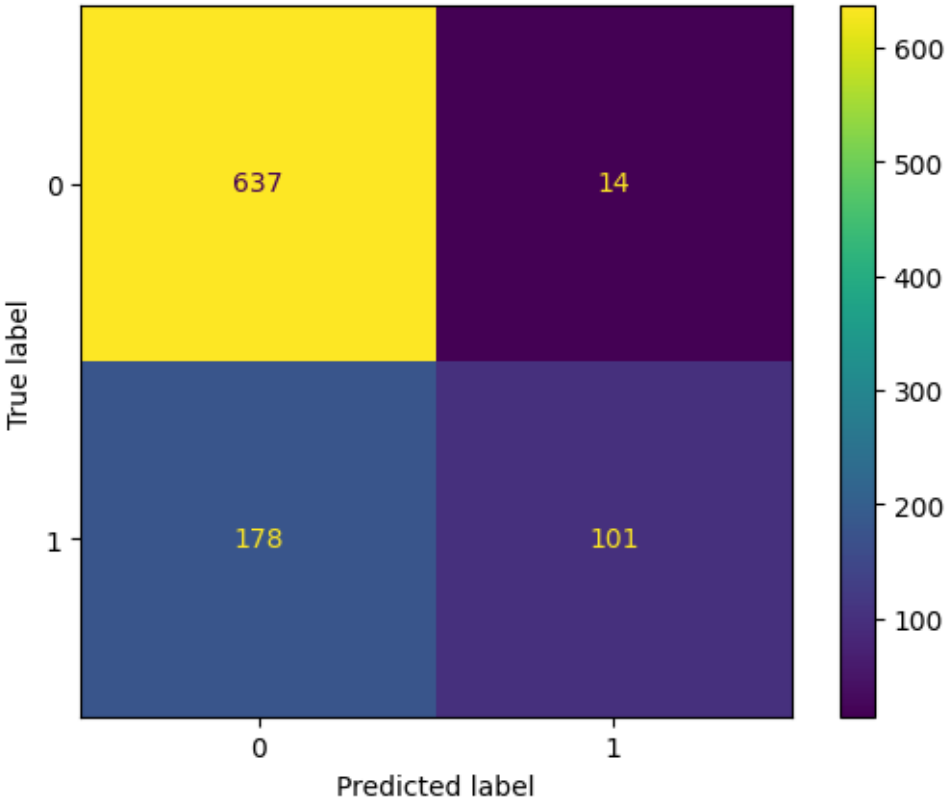
Modelo	Conjunto de variables		
	SKB (50 variables)	**SFM** (36 variables)	**Todas** (239 variables)
LR	72 ± 6.9	71 ± 7.2	68 ± 7.8
RFC	78 ± 5.0	$\mathbf{78 \pm 5.0}$	79 ± 5.1
KNC	67 ± 7.7	65 ± 8.4	64 ± 8.3

Modelos: Logistic Regression (LR), Random Forest Classifier (RFC) y K-Neighbors Classifier (KNC).
Conjunto de variables: Select K-Best (SKB), Select From Model (SFM).

Si bien la precisión observada se puede considerar aceptable para un modelo de predicción, es importante hacer un análisis más profundo de los resultados obtenidos con el fin de mejorar su desempeño, considerando además el objetivo en el contexto actual, que es la detección de artículos que no serán citados.

Una forma de evaluar los modelos de clasificación es a través de una matriz de confusión, la cual indica la cantidad de veces que el modelo clasificó un artículo en la clase correcta y en la clase incorrecta. La **Figura 2** muestra la matriz de confusión para el modelo seleccionado, donde las filas indican la clase o etiqueta real (0 = citado, 1 = no citado), y las columnas indican la clase que predice el modelo. La segunda fila muestra los artículos etiquetados como no citados y la segunda columna muestra los artículos que fueron clasificados por el modelo como no citados.

Figura 2
Matriz de confusión – Modelo RFC, variables SFM. Variable de respuesta citado o no



Nota: 0 = citado, 1 = no citado. Filas = Etiqueta real, Columnas = Predicción.

De esta matriz se pueden calcular dos medidas muy importantes, que son *Precision* (P) y *Recall* (R). La **Ecuación 1** representa P, indicando la exactitud de las predicciones positivas, es decir, cuántos de los artículos que el modelo clasificó como no citados (101 + 14) son realmente no citados (101). Por otra parte, la **Ecuación 2** representa R, que dice cuántos de los artículos que son realmente no citados (178 + 101) son clasificados como no citados por el modelo (101).

Ecuación 1

Ecuación de medida de desempeño Precision

$$\boldsymbol{P = \frac{TP}{TP + FP}}$$

$$\boldsymbol{P} = Precision$$

$$\boldsymbol{TP} = True\ Possitives\ (Verdaderos\ positivos)$$

$$\boldsymbol{FP} = False\ Possitives\ (Falsos\ positivos)$$

Ecuación 2
Ecuación de medida de desempeño Recall

$$\boldsymbol{R = \frac{TP}{TP + FN}}$$

$$\boldsymbol{R} = Recall$$
$$\boldsymbol{TP} = True\ Possitives\ (Verdaderos\ positivos)$$
$$\boldsymbol{FN} = False\ Negatives\ (Falsos\ negativos)$$

Para la revista es importante tener un modelo capaz de clasificar como no citados artículos que realmente no serán citados, es decir, que los artículos que no serán citados sean clasificados realmente como no citados, por lo cual, un alto R es deseable. En la **Tabla 4** se puede observar esta medida para los escenarios planteados, donde nuevamente, se obtiene un mejor rendimiento en el modelo RFC con el conjunto de variables otorgado por SFM. Por esta razón, este escenario es seleccionado como el mejor modelo hallado hasta el momento para continuar realizando análisis sobre él.

Tabla 4
Recall en % de los modelos entrenados con diferentes conjuntos de variables

Modelo	Conjunto de variables		
	SKB (50 variables)	**SFM** (36 variables)	**Todas** (239 variables)
LR	16 ± 11.8	19 ± 9.3	19 ± 12.7
RFC	36 ± 10.5	$\mathbf{36 \pm 11.1}$	35 ± 10
KNC	21 ± 13.6	22 ± 13	21 ± 15.5

En la **Figura 2** se aprecia la matriz de confusión del modelo seleccionado, la cual muestra que para los artículos con etiqueta 1 (artículo no citado), se predicen 178 como citados (falso positivo) y solo 101 como no citados (verdadero positivo). Si bien el modelo en general presenta un buen resultado (78%, visto en **Tabla 3**), el interés está en disminuir esta diferencia para la etiqueta 1, o dicho de otra forma, aumentar el R del modelo. Adicionalmente, la **Tabla 5** muestra

el reporte de clasificación del modelo, con las diferentes medidas de desempeño (*Precision, Recall y F1-score*), donde se confirma el desempeño aceptable del modelo.

Tabla 5
Classification report – Reporte de clasificación del modelo en %

Variable respuesta	Medida de evaluación		
	Precisión	**Recall**	**F1-score**
0 (si citado)	78	98	87
1 (no citado)	87	36	51

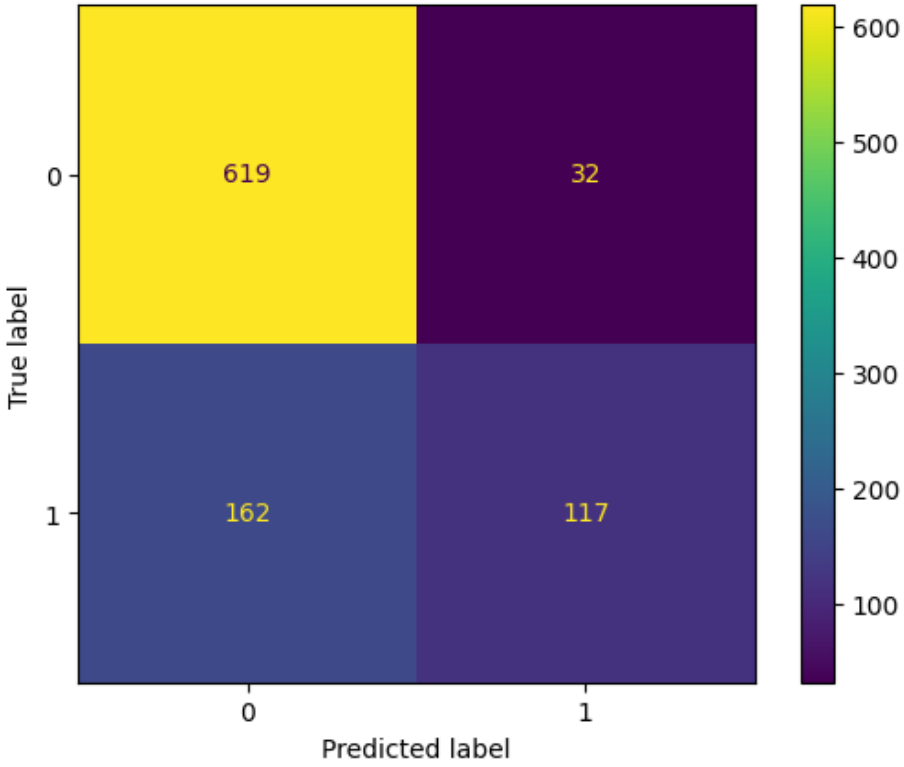
Buscando mejorar el *Recall*, se realiza un ajuste de hiperparámetros del modelo haciendo uso de la función *GridSearchCV* de Sklearn, la cual realiza una combinación de los diferentes hiperparámetros con el fin de encontrar una combinación que mejore su desempeño. La **Tabla 6** muestra los cambios sugeridos.

Tabla 6
Ajuste de hiperparámetros

Hiperparámetro	Valor inicial	Valor luego del ajuste
criterion	"gini"	"gini"
max_features	"sqrt"	None
n_estimators	100	250

Con este ajuste, se logró aumentar el *Recall* a 42%. La **Figura 3** muestra esta mejoría en la matriz de confusión con el cambio en las predicciones, donde se está considerando una cantidad mayor de predicciones en el cuadrante verdadero positivo (117), sin embargo, sigue siendo un valor bajo. Además se aprecia un aumento de FP de 14 a 32, que quiere decir que el modelo clasificará más artículos como no citados, cuando en realidad si serán citados, lo cual no representa un gran problema.

Figura 3
Matriz de confusión luego de ajustar los hiperparámetros



Hasta el momento, se ha considerado como variable respuesta si el artículo será citado o no en algún momento, por lo cual el modelo puede servir para identificar si un nuevo artículo recibirá citas en algún momento; sin embargo, no se está teniendo en cuenta la ventana de tiempo de esta variable, puesto que se comparan artículos que llevan más de 10 años publicados, con artículos publicados recientemente.

Buscando normalizar esta situación, se considera el rango de tiempo que normalmente se utiliza para calcular el factor de impacto, el cual considera los artículos publicados en los últimos dos años, por esta razón, se construye una variable de respuesta que indica si el artículo será citado o no en una ventana de tiempo de dos años con el fin de construir un segundo escenario. Para esto, es necesario filtrar el Dataset para los artículos publicados hasta 2020, de manera que se consideren artículos que llevan como mínimo dos años publicados. Nuevamente, tras analizar la **Tabla 7**, se selecciona el modelo RFC por ofrecer mejores medidas de desempeño. Es de notar que, a pesar de una disminución en el *Accuracy*, hay un aumento considerable en el Recall (81%), que es la medida de interés.

Tabla 7
Accuracy y Recall en % de los modelos entrenados para las variables seleccionadas con SFM– Variable de respuesta citado o no en 2 años

Modelo	Accuracy	Recall
LR	64 ± 3.8	81 ± 5.3
RFC	$\mathbf{66 \pm 4.0}$	$\mathbf{81 \pm 5.0}$
KNC	61 ± 4.7	78 ± 5.9

En la **Figura 4** y **Tabla 8** se aprecia el cambio del Recall utilizando la nueva variable de respuesta. El Recall obtenido indica que el 82% de los artículos que el modelo clasifica como no citados, efectivamente no serán citados en los próximos dos años, de manera que los esfuerzos que se realicen se hacen sobre los artículos correctos. Dentro de estos esfuerzos se puede sugerir un aumento en la visibilidad del artículo o incluso promocionarlos.

Figura 4
Matriz de confusión - Variable de respuesta citado o no en 2 años

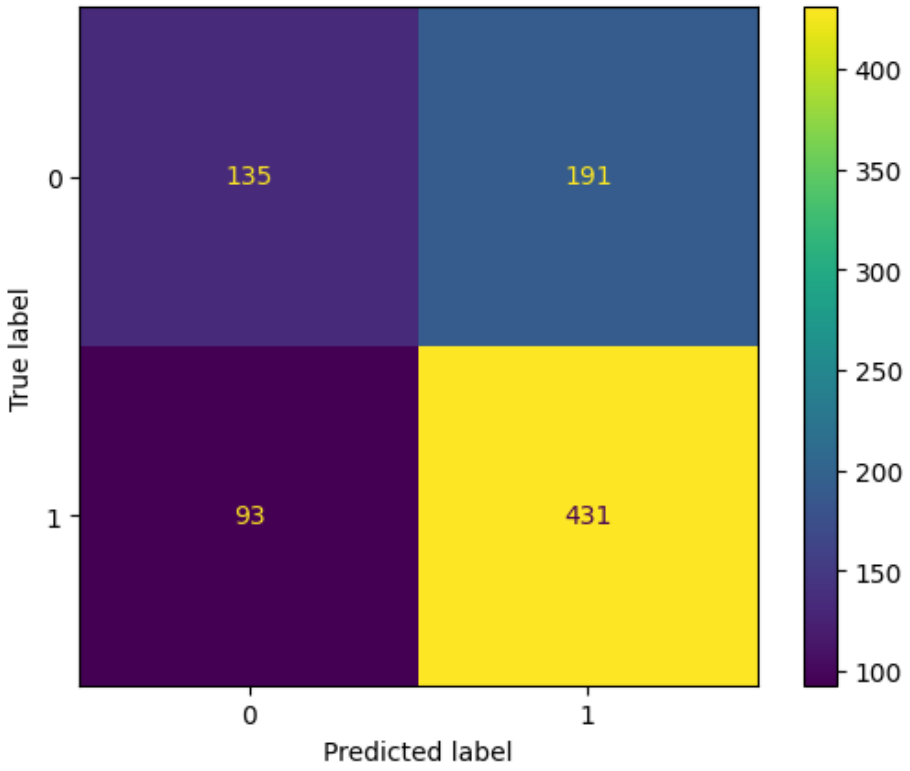


Tabla 8
Classification report – Reporte de clasificación en %, Variable de respuesta citado o no en 2 años

Variable respuesta	Medida de evaluación		
	Precisión	**Recall**	**F1-score**
0 (si citado en dos años)	59	41	49
1 (no citado en dos años)	69	82	75

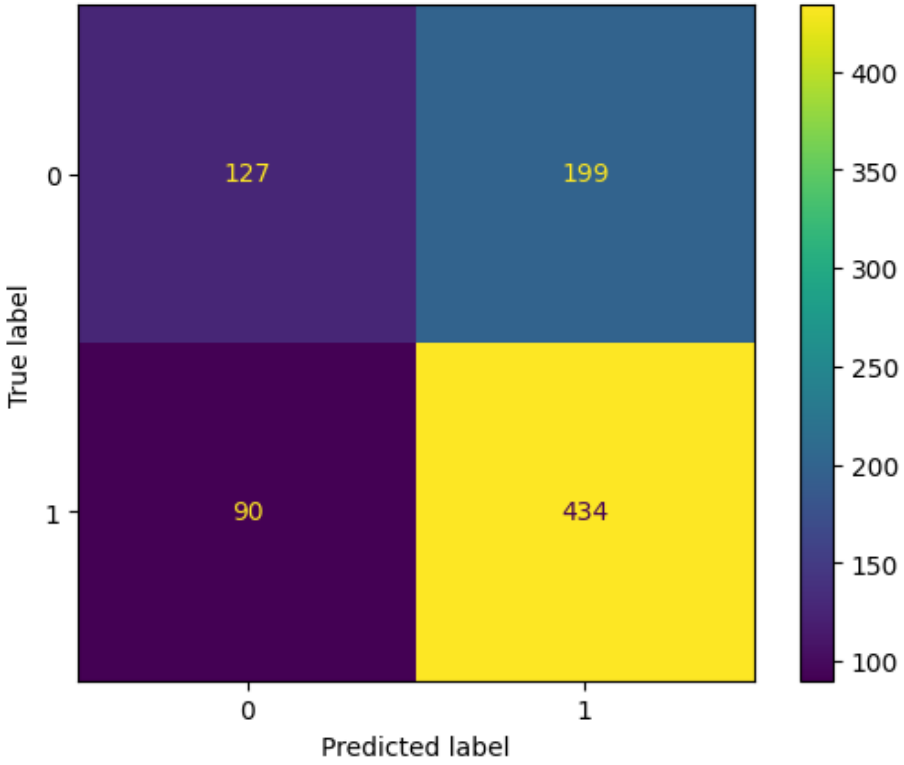
Se realiza el ajuste de hiperparámetros para el segundo escenario, cuyos valores se encuentran en la **Tabla 9**, con este ajuste, se alcanza un Recall de 83% sin afectar las demás medidas. En la **Figura 5** se observa la matriz de confusión, donde se aprecia la mejora (de 431 verdaderos positivos a 433).

Tabla 9
Ajuste de hiperparámetros

Hiperparámetro	Valor inicial	Valor luego del ajuste
criterion	"gini"	"log_loss"
max_features	"sqrt"	"log2"
n_estimators	100	250

Figura 5

Matriz de confusión - Variable de respuesta citado o no en 2 años con ajuste de hiperparámetros



Además de tener un modelo con una medida de evaluación aceptable, es importante interpretar las variables que se están utilizando, así como su importancia, con el fin de conocer cuáles tienen un mayor impacto en la predicción generada.

En la **Tabla 10** se encuentra la importancia de cada variable dentro del modelo de clasificación RFC, organizadas de mayor a menor importancia. Además, es importante conocer la relación de cada variable predictora con la variable de respuesta. En este caso, RFC es un modelo de caja negra, es decir que no ofrece interpretabilidad de las variables, por lo que se realiza un análisis de variables independientes a través de una regresión logística, donde el coeficiente muestra la relación de ambas variables en un contexto de regresión. Interpretando var_1 como ejemplo, al haber una mayor cantidad de documentos que han citado al primer autor (que puede verse como mayor experiencia), menor probabilidad de no ser citado.

Tabla 10

Importancia de las variables predictoras en el modelo final (RFC)

Nombre variable	Definición	Importancia	coeficiente (LR)
var_1	Cantidad de documentos que han citado al primer autor	0.055	-0.059
var_2	Cantidad de citas recibidas por el primer autor	0.055	0.192
var_3	Cantidad de palabras en el abstract	0.055	-0.17
var_4	Año de inicio de publicación del primer autor	0.05	-0.087
var_5	Cantidad de referencias	0.047	-0.227
var_6	Posición del artículo dentro del Issue	0.041	0.144
var_7	Cantida	0.04	0.153
var_8	Cantidad de citas recibidas por autor más citado	0.039	0.149
var_9	Cantidad de documentos que ha publicado el autor más citado	0.037	-0.109
var_10	Año de inicio de publicación del autor más citado	0.037	-0.273
var_11	Cantidad de co-autores que ha tenido el autor más citado	0.037	-0.191
var_12	Índice H del primer autor	0.036	-0.755
var_13	Cantidad de palabras en el título	0.036	-0.008
var_14	Cantidad de áreas en las que ha publicado el autor más citado	0.035	0.038
var_15	Año de creación del perfil en Scopus del primer autor	0.035	-0.122
var_16	Cantidad de co-autores que ha tenido el primer autor	0.034	0.361
var_17	Cantidad de documentos que ha publicado el primer autor	0.032	-0.017
var_18	Cantidad de páginas del documento	0.032	0.074
var_19	Año de creación del perfil en Scopus del autor más citado	0.028	-0.184
var_20	Cantidad de áreas en las que ha publicado el primer autor	0.028	-0.047
var_21	Cantidad de identificadores (Scopus ID) que ha tenido el autor más citado	0.026	0.062
var_22	Índice H del autor más citado	0.026	-0.339

var_23	Cantidad de identificadores (Scopus ID) que ha tenido el primer autor	0.023	0.144
var_24	Cantidad de afiliaciones a las que ha pertenecido el autor más citado	0.021	0.001
var_25	Cantidad de Keywords del documento	0.02	0.091
var_26	Diferencia entre la cantidad de afiliaciones y la cantidad de autores	0.018	-0.042
var_27	Cantidad de autores que participan en el documento	0.017	0.001
var_28	Cantidad de afiliaciones a las que ha pertenecido el primer autor	0.017	-0.032
var_29	Cantidad de afiliaciones que participan en el documento	0.015	-0.051
var_30	Variable Binaria que indica si el área más frecuente del primer autor es ingeniería	0.006	0.019
var_31	Si el autor más citado es internacional	0.005	-0.089
var_32	Si el autor más citado es nacional	0.005	0.089
var_33	Si el primer autor es internacional	0.005	-0.075
var_34	Si el primer autor es nacional	0.004	0.074
var_35	Variable Binaria que indica si el área más frecuente del autor más citado es ingeniería	0.004	0.331

A pesar de que el coeficiente de regresión ofrece una interpretación general de las variables, se hace necesario una interpretación más específica para cada artículo al momento de hacer una predicción con el fin de saber qué variables están causando el resultado del modelo. En este sentido, se hace uso de la librería *treeinterpreter* de Python, la cual ofrece la oportunidad de interpretar las predicciones de modelos de caja negra como RFC. La **Tabla 11** muestra el *Script* de Python que ejemplifica la forma en que se realiza una predicción con esta librería, para lo cual es necesario instalarla, crear y entrenar el modelo (o tenerlo generado con anterioridad), crear una instancia para la predicción (en este caso, se utilizó un artículo existente a manera de ejemplo) y finalmente generar la predicción, la cual devuelve el porcentaje de predicción para cada clase, el sesgo y la contribución de cada variable.

Tabla 11

Script de Python – Predicción de un artículo, con interpretación de variables

```
# Instalar e importar la librería
!pip install treeinterpreter
from treeinterpreter import treeinterpreter as ti

# Se crea el modelo con los hiperparámetros ajustados
rf_clf = RandomForestClassifier(criterion= 'log_loss',
                                max_features= 'log2', n_estimators= 250)

# Se entrena el modelo
rf_clf.fit(X, y)

# Se crea una instancia (variables del artículo a predecir)
instance = X.iloc[[-1],:]

# Se realiza la predicción haciendo uso de la librería
prediction, bias, contributions = ti.predict(rf_clf, instance)
```

Se realiza la predicción para un artículo que no fue citado (variable respuesta 1), y se obtienen los resultados de la **Tabla 12**. La predicción indica la proporción que se asigna para cada una de las clases, que en este caso se asigna una mayor proporción (0.888) para la clase 1; el sesgo indica el valor inicial de proporción para cada clase (o el intercepto) y las contribuciones indican en qué medida la variable indicada está aportando a mejorar o empeorar el sesgo inicial, por lo que una contribución de mayor magnitud quiere decir que la variable es más importante en la predicción. Analizando las 4 variables presentadas en la tabla, se puede ver que, para el artículo analizado, la variable var_4 (año de inicio de publicación del primer autor) tiene una importancia mayor en la predicción realizada, la cual se puede interpretar como: el año de inicio de publicación del primer autor del artículo está aportando en 0.035 a que el artículo no sea citado, es decir que esta variable tiene una connotación negativa en la predicción. En este sentido, la contribución de la variable se podrá utilizar para saber cuáles variables son más importantes para la predicción realizada y puede ayudar a identificar falencias en los artículos.

Tabla 12

Resultados de predicción

Valor	**Clase 0** (citado)	**Clase 1** (no citado)

Prediction (predicción)		0.112	0.888
Bias (sesgo)		0.381	0.619
Contributions (contribución de cada variable)	var_1	0.003	-0.003
	var_2	0.027	-0.027
	var_3	0.005	-0.005
	var_4	-0.035	0.035
…	…	…	…

Conclusiones

En el presente documento se abordó el problema de baja citación de la Revista Facultad de Ingeniería (Redin) de la Universidad de Antioquia, que es una problemática que preocupa las revistas científicas puesto que la citación es uno de los indicadores más importantes en términos de prestigio e impacto. Actualmente la revista se en el cuartil 3 (Q3) de las revistas que están indexadas en Scopus según el indicador SJR, que considera la cantidad de citas para su cálculo, el cual se considera bajo.

Se construyó una herramienta basada en ML que permita identificar cuáles de los nuevos artículos que serán publicados tienen un mayor riesgo de no recibir citas, de manera que los encargados de tomar decisiones en la revista puedan realizar acciones encaminadas a aumentar la probabilidad de citación a través de acciones como aumentar su visibilidad o promocionarlos.

Para la construcción de la herramienta se utilizó la metodología CRISP-DM, en la que se recopilaron los datos desde diferentes fuentes y haciendo uso de diferentes métodos, posteriormente se exploraron, limpiaron y prepararon para finalmente aplicar modelos de ML. En la modelación y evaluación se consideraron inicialmente 9 escenarios a partir de la combinación de 3 modelos (RFC, LR, KNC) y 3 conjuntos de variables (todas, SKB, SFM), donde el modelo RFC con las variables SFM tuvo el mejor desempeño, además de ser el más parsimonioso. Si bien este escenario brindó el mayor *Accuracy* y *Recall*, se invirtieron esfuerzos en mejorar esta última medida, ya que el interés principal está en identificar los artículos con mayor riesgo de no recibir citas. Se normalizó la variable de respuesta usando una ventana de citación de 2 años, lo cual ayudó a mejorar considerablemente el desempeño.

Se identificaron las variables más importantes para el modelo para finalmente proponer un método de interpretación de estas variables al momento de realizar una predicción con el fin de conocer la contribución individual de cada una, y ganar conocimiento sobre las variables que están aumentando el riesgo de no citación.

Como trabajo futuro, se sugiere explorar otras técnicas de interpretación con el fin de compararlas con la técnica actual y evaluar la que mejor se adapte. Adicionalmente, realizar despliegue web (el cual se encuentra en proceso), el cual sea de uso intuitivo para sus usuarios, así como un instructivo de uso.

La recomendación que se hace a la revista está encaminada a utilizar la herramienta al momento de publicar un nuevo Issue, con el fin de identificar los artículos con mayor riesgo de no recibir citas, de manera que tomen decisiones encaminadas a mejorar esta situación, dentro de las que se pueden sugerir: dar mayor visibilidad a esos artículos particulares o incluso hacer promoción sobre ellos.

Referencias

Abrishami, A., & Aliakbary, S. (2019). Predicting citation counts based on deep neural network learning techniques. Journal of Informetrics, 13(2), 485-499. https://doi.org/10.1016/j.joi.2019.02.011

An, X., Sun, X., & Xu, S. (2022). Important citations identification with semi-supervised classification model. Scientometrics, 127(11), 6533-6555. https://doi.org/10.1007/s11192-021-04212-6

BinMakhashen, G. M., & Al-Jamimi, H. A. (2022). Evaluation of Machine Learning to Early Detection of Highly Cited Papers. 2022 7th International Conference on Data Science and Machine Learning Applications (CDMA), 1-6. https://doi.org/10.1109/CDMA54072.2022.00006

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (1999). CRISP-DM 1.0.

Dias, L. C., Lev, B., & Anderson, J. B. (2023). Low cited articles in operations research / management science. Omega, 115, 102792. https://doi.org/10.1016/j.omega.2022.102792

Elgendi, M. (2019). Characteristics of a Highly Cited Article: A Machine Learning Perspective. IEEE Access, 7, 87977-87986. https://doi.org/10.1109/ACCESS.2019.2925965

Géron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems (2nd ed.). O'Reilly.

Ha, T. (2022). An explainable artificial-intelligence-based approach to investigating factors that influence the citation of papers. Technological Forecasting and Social Change, 184, 121974. https://doi.org/10.1016/j.techfore.2022.121974

Himani, S., Kumar, M. H., Enduri, M. K., Begum, S. S., Rageswari, G., & Anamalamudi, S. (2022). A Comparative Study on Machine Learning based Prediction of Citations of Articles. 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), 1819-1824. https://doi.org/10.1109/ICOEI53556.2022.9777184

Ibáñez Martín, A. (2015). Machine Learning in Scientometrics [PhD Thesis, Universidad Politécnica de Madrid]. https://doi.org/10.20868/UPM.thesis.36488

Marín Velásquez, T., & Arriojas Tocuyo, D. D. J. (2021). Ubicación de revistas científicas en cuartiles según SJR: Predicción a partir de estadística multivariante. Anales de Documentación, 24(1). https://doi.org/10.6018/analesdoc.455951

Qayyum, F., Jamil, H., Iqbal, N., Kim, D., & Afzal, M. T. (2022). Toward potential hybrid features evaluation using MLP-ANN binary classification model to tackle meaningful citations. Scientometrics, 127(11), 6471-6499. https://doi.org/10.1007/s11192-022-04530-3

Repiso, R., Moreno-Delgado, A., & Aguaded, I. (s. f.). Factores que influyen en la frecuencia de citación de un artículo.

Revista Facultad de Ingenieria. (s. f.). Recuperado 9 de abril de 2023, de https://www.scimagojr.com/journalsearch.php?q=12400154740&tip=sid&exact=no

Rosenkrantz, A. B., Doshi, A. M., Ginocchio, L. A., & Aphinyanaphongs, Y. (2016). Use of a Machine-learning Method for Predicting Highly Cited Articles Within General Radiology Journals. Academic Radiology, 23(12), 1573-1581. https://doi.org/10.1016/j.acra.2016.08.011

Tohalino, J. A. V., & Amancio, D. R. (2022). On predicting research grants productivity via machine learning. Journal of Informetrics, 16(2), 101260. https://doi.org/10.1016/j.joi.2022.101260

Vargas, J. (2021). Editorial. Revista Facultad de Ingeniería Universidad de Antioquia. https://doi.org/10.17533/udea.redin.20210529

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Anexos

Anexo 1. Proceso de recopilación de información de autores (API)

Para la recolección de la información de los autores fue necesario realizar Web Scraping haciendo uso de la API disponible en la Base de Datos de Scopus. Este es el sitio web del recurso: *https://dev.elsevier.com/*.

Este recurso está disponible para todos los miembros de la Universidad de Antioquia, puesto que actualmente hace parte de los servicios contratados entre ambas entidades.

Inicialmente, fue necesario crear un *Token* en el sitio web de la API, que es un código único que permite realizar la petición. Posteriormente, se generó el *Script* de Python que se puede apreciar en la **Tabla 13,** que hace uso de la librería *requests* para hacer la petición, la cual se hace con el identificador (ID) de cada autor. Finalmente, se exporta el Dataset obtenido para su posterior exploración y análisis.

Para poder realizar esta petición, fue necesario autenticarse desde una IP de la universidad de manera que se validara la pertenencia a la institución, lo cual implicó la conexión remota a un equipo de la universidad, otorgado por el asesor. Inicialmente, se realizaron varios intentos desde *Google Colab*, sin embargo, persistía un error que fue solucionado al utilizar *Jupyter Notebook* instalado localmente, ya que *Google Colab* utiliza un servidor en la nube e impedía autenticar la IP local.

Cabe resaltar que la información se obtuvo en un formato *Json*, lo cual requirió una transformación posterior para ser utilizada en un formato de tabla.

Tabla 13

Script de python – API Request de la información de los autores disponible en la Base de Datos de Scopus

```
# Importar librerías
import requests
import pandas as pd

# Importar dataset con los ID de cada autor
df = pd.read_csv('AuthorId.csv')

```

```
8  # Código de Request
9  df_final = pd.DataFrame()
10 i = 0
11 while i < df.shape[0]:
12     if i >= 2500:
13         authors = list((df['Author Id'][i:i + 19]).astype('str'))
14     else:
15         authors = list((df['Author Id'][i:i + 25]).astype('str'))
16     URL = "https://api.elsevier.com/content/author"
17 
18     headers = {
19         'X-ELS-APIKey' : 'token', # Aquí va el TOKEN
20         'Accept' : 'application/json'
21     }
22 
23     parameters = {
24         'author_id' : authors,
25         'view' : 'ENHANCED'
26     }
27 
28     response = requests.get(URL, headers=headers, params=parameters)
29     df_temp = pd.DataFrame(response.json()['author-retrieval-response-
   list'])
30     df_final = pd.concat([df_final, df_temp])
31     i += 25
32 
33 # Exportar Dataset con la información completa de los autores
34 df_final.to_csv('df_final.csv')
```

Anexo 2. **Proceso de recopilación de información de los artículos desde la página web de la revista**

Se realizó un proceso de Web Scraping de la información de los artículos desde la página web de la revista https://revistas.udea.edu.co/index.php/ingenieria.

La **Tabla 14** muestra el Script de Python con el proceso realizado haciendo uso de la librería *scrapy*.

Si bien la información adquirida puede ser útil para diferentes análisis, se descartó puesto que son variables que no ayudarán en la predicción (cantidad de vistas de *abstract* y *pdf*), puesto que se generan luego de la publicación.

Tabla 14

Script de python – Web Scraping de la información de los artículos desde la página web de la revista

```
# Instalación de la librería Scrapy
!pip install scrapy

# Creación del proyecto dentro del entorno virtual
!scrapy startproject firstproject

# Importar librería os
import os
os.chdir('/content/firstproject/firstproject/spiders')

#Código del Web Scraping
%%writefile -a quotes_spider.py

from scrapy.item import Field, Item
from scrapy.spiders import CrawlSpider, Rule
from scrapy.linkextractors import LinkExtractor
from scrapy.loader import ItemLoader
from scrapy.loader.processors import MapCompose
from scrapy.selector import Selector

class paper(Item):
    title = Field()
    pages = Field()
    authors = Field()
    abstractView = Field()
    issue = Field()

class paperSpider(CrawlSpider):
    name= 'PaperCrawler'
```

```
30     start_urls =
   ['https://revistas.udea.edu.co/index.php/ingenieria/issue/archive']
31     allowed_domains = ['revistas.udea.edu.co']
32
33     rules = (
34         Rule(LinkExtractor(allow=r'/archive/\d+')),
35         Rule(LinkExtractor(allow=r'/view'), callback='parse_items')
36     )
37
38     def parse_items(self, response):
39         sel = Selector(response)
40         articulos = sel.xpath('//div[@class="sections"]/div/ul/li/div')
41
42         #Iterar sobre artículos
43         for i, elem in enumerate(articulos):
44             item = ItemLoader(paper(), elem)
45             item.add_xpath('title', './/h3/a/text()')
46             item.add_xpath('authors',
   './/div[@class="meta"]/div[@class="authors"]/text()')
47             item.add_xpath('pages',
   './/div[@class="meta"]/div[@class="pages"]/text()')
48             item.add_xpath('abstractView', './/div[@style="padding-top:
   2%;"]/b/text()')
49             yield item.load_item()
50
51 # Realización de la solicitud
52 !scrapy runspider quotes_spider.py -o respuesta1.csv -t csv
53
54 # Exportar el dataset generado
55 df.to_csv('df_redin_scapy.csv')
```