UNIVERSIDAD CÉSAR VALLEJO

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Machine Learning utilizando el Método Boosting de ensemble para la deserción estudiantil en EBR

TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE:

Ingeniero de Sistemas

AUTORES:

Mantilla Lozano, Fernando Javier (orcid.org/0000-0002-5057-6767)

Vilca Yataco, Pedro Nemecio (orcid.org/0000-0001-7037-1497)

ASESOR:

Mg. Saboya Ríos, Nemias (orcid.org/0000-0002-7166-2197)

LÍNEA DE INVESTIGACIÓN:

Sistemas de Información y Comunicaciones

LÍNEA DE RESPONSABILIDAD SOCIAL UNIVERSITARIA:

Desarrollo económico, empleo y emprendimiento

LIMA – PERÚ

2023

DEDICATORIA

A nuestros padres y familiares, quienes siempre han sido nuestra fuente de inspiración y apoyo inquebrantable. A nuestros amigos y seres queridos, por su aliento constante y su comprensión incondicional a lo largo de este viaje. A nuestros profesores y mentores, por su sabiduría y orientación que han enriquecido nuestro conocimiento. A todas las personas que han contribuido de alguna manera a este logro, ¡gracias por formar parte de este sueño hecho realidad! Esta tesis está dedicada a todos ustedes, con gratitud y cariño.

AGRADECIMIENTO

En este punto culminante de nuestra trayectoria académica, nos gustaría expresar nuestro profundo agradecimiento a todas las personas que han sido parte fundamental de este logro. Sin su apoyo, orientación y contribuciones, este proyecto no habría sido posible. Agradecemos a nuestra familia, amigos, compañeros de estudios, profesores y mentores. Así mismo, quisiéramos extender nuestra gratitud a los participantes y colaboradores que contribuyeron a la recopilación de datos y al desarrollo de este trabajo. Sus aportes fueron esenciales para enriquecer esta investigación.

UNIVERSIDAD CÉSAR VALLEJO

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Declaración de Autenticidad del Asesor

Yo, SABOYA RIOS, NEMIAS, docente de la FACULTAD DE INGENIERÍA Y ARQUITECTURA de la escuela profesional de INGENIERÍA DE SISTEMAS de la UNIVERSIDAD CÉSAR VALLEJO SAC - LIMA NORTE, asesor de la tesis, titulada: “Machine Learning utilizando el Método Boosting de Ensemble para la Deserción Estudiantil en EBR”, cuyos autores son MANTILLA LOZANO FERNANDO JAVIER, VILCA YATACO PEDRO NEMECIO, constato que la investigación tiene un índice de similitud de 13.00%, verificable en el reporte de originalidad del programa Turnitin, el cual ha sido realizado sin filtros, ni exclusiones.

He revisado dicho reporte y concluyo que cada una de las coincidencias detectadas no constituyen plagio. A mi leal saber y entender la Tesis cumple con todas las normas para el uso de citas y referencias establecidas por la Universidad César Vallejo.

En tal sentido, asumo la responsabilidad que corresponda ante cualquier falsedad, ocultamiento u omisión tanto de los documentos como de información aportada, por lo cual me someto a lo dispuesto en las normas académicas vigentes de la Universidad César Vallejo.

LIMA, 18 de Diciembre del 2023

Apellidos y Nombres del Asesor:	**Firma**
SABOYA RIOS NEMIAS **DNI:** 42001721 **ORCID:** 0000-0002-7166-2197	Firmado electrónicamente por: NSABOYARI el 18-12-2023 11:26:36

Código documento Trilce: TRI - 0699771

INVESTIGA UCV

UNIVERSIDAD CÉSAR VALLEJO

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Declaración de Originalidad de Autores

Nosotros, MANTILLA LOZANO FERNANDO JAVIER, VILCA YATACO PEDRO NEMECIO estudiantes de la FACULTAD DE INGENIERÍA Y ARQUITECTURA de la escuela Profesional de INGENIERÍA DE SISTEMAS de la UNIVERSIDAD CÉSAR VALLEJO SAC - LIMA NORTE, declaramos bajo juramento que todos los datos e información que acompañan la Tesis titulada: “Machine Learning utilizando el Método Boosting de Ensemble para la Deserción Estudiantil en EBR”, es de nuestra autoría, por lo tanto, declaramos que la Tesis:

1. No ha sido plagiado ni total, ni parcialmente.
2. Hemos mencionado todas las fuentes empleadas, identificando correctamente toda cita textual o de paráfrasis proveniente de otras fuentes.
3. No ha sido publicado, ni presentado anteriormente para la obtención de otro grado académico o título profesional.
4. Los datos presentados en los resultados no han sido falseados, ni duplicados, ni copiados.

En tal sentido asumimos la responsabilidad que corresponda ante cualquier falsedad, ocultamiento u omisión tanto de los documentos como de información aportada, por lo cual me someto a lo dispuesto en las normas académicas vigentes de la Universidad César Vallejo.

Nombres y Apellidos	**Firma**
MANTILLA LOZANO FERNANDO JAVIER **DNI:** 45151072 **ORCID:** 0000-0002-5057-6767	Firmado electrónicamente por: FMANTILLAL el 10-01-2024 16:13:28
VILCA YATACO PEDRO NEMECIO **DNI:** 21863073 **ORCID:** 0000-0001-7037-1497	Firmado electrónicamente por: PVILCAY el 09-01-2024 17:51:20

Código documento Trilce: INV - 1425493

INVESTIGA UCV

ÍNDICE DE CONTENIDOS

DEDICATORIA…….....ii
AGRADECIMIENTO……..iii
Declaración de Autenticidad del Asesor…….....iv
Declaración de Originalidad de Autores…….....v
Índice de contenidos…….....vi
Índice de tablas……vii
Índice de figuras…….....viii
Resumen……..ix
Abstract…….....x
I. INTRODUCCIÓN……...1
II. MARCO TEÓRICO……....6
III. METODOLOGÍA……..24
3.1. Tipo y diseño de investigación……24
3.2. Variables y operacionalización……....25
3.3. Población, muestra y muestreo……...26
3.4. Técnicas e instrumentos de recolección de datos……27
3.5. Procedimientos……..28
3.6. Métodos de análisis de datos……..46
3.7. Aspectos éticos……..46
IV. RESULTADOS……48
V. DISCUSIÓN……..56
VI. CONCLUSIONES……58
VII. RECOMENDACIONES ……59
REFERENCIAS ……....60
ANEXOS……69

ÍNDICE DE TABLAS

Tabla 1: Matriz de confusión 22
Tabla 2: Descripción general de variables numéricas 31
Tabla 3: Descripción general de variables categóricas 31
Tabla 4: Tabla de contingencia en porcentajes relativos según Situación de matrícula inicial 36
Tabla 5: Tabla de contingencia en porcentajes relativos según promedio final 36
Tabla 6: Resumen de indicadores del resultado de entrenamiento 45
Tabla 7: Precisión de la predicción realizada por XGBOOST 51
Tabla 8: Precisión de la predicción realizada por LIGHTGBM 51
Tabla 9: Precisión de la predicción realizada por CATBOOST 51
Tabla 10: Exactitud de la predicción realizada por XGBOOST 52
Tabla 11: Exactitud de la predicción realizada por LIGHTGBM 52
Tabla 12: Exactitud de la predicción realizara por CATBOOST 53
Tabla 13: Sensibilidad de la predicción realizada por XGBOOST 53
Tabla 14: Sensibilidad de la predicción realizada por LIGHTGBM 54
Tabla 15: Sensibilidad de la predicción realizada por CATBOOST 54
Tabla 16: Comparación entre resultados por indicadores 55

ÍNDICE DE FIGURAS

Figura 1: Fases de la metodología KDD 10
Figura 2: Método Boosting 13
Figura 3: Construcción de hojas por LightGBM 16
Figura 4: Aproximación secuencial con el método Boosting 19
Figura 5: Diagrama de diseño experimental 25
Figura 6: Ajuste inicial en variable de situación final 29
Figura 7: Vista de variables anterior a la Imputación 30
Figura 8: Vista posterior a Imputación 30
Figura 9: Boxplot de edad en el registro según situación final 32
Figura 10: Boxplot de áreas desaprobadas según situación final 33
Figura 11: Boxplot de porcentaje de inasistencias según situación final 33
Figura 12: Barplot de variables categóricas según situación final 35
Figura 13: Diccionario de valores numéricos según variable categórica y proceso de codificación 37
Figura 14: División de data para entrenamiento y validación 38
Figura 15: Flujo de trabajo para modelos Ensembler - Método boosting 38
Figura 16: Configuraciones de entrenamiento del Modelo XGBOOST 39
Figura 17: Configuraciones de entrenamiento del Modelo LIGHTGBM 40
Figura 18: Configuraciones de entrenamiento del Modelo CATBOOST 40
Figura 19: Matriz de confusión de entrenamiento – XGBOOST 42
Figura 20: Feature importance – XGBOOST 42
Figura 21: Matriz de confusión de entrenamiento – LIGHTGBM 43
Figura 22: Feature importance – LIGHTGBM 44
Figura 23: Matriz de confusión de entrenamiento – CATBOOST 44
Figura 24: Feature importance – CATBOOST 45
Figura 25: Matriz de confusión - XGBOOST 48
Figura 26: Matriz de confusión – LIGHTGBM 49
Figura 27: Matriz de confusión – CATBOOST 50

Resumen

La finalidad del presente proyecto fue determinar la mejora del modelo predictivo de machine learning utilizando el método Boosting en la predicción de la deserción estudiantil en EBR (Educación Básica Regular), como metodología se utilizó KDD (Descubrimiento de conocimiento en base de datos) y para la medición se hizo el uso de tres indicadores: Precisión, Sensibilidad y Exactitud. Como resultado final, obtuvimos que el modelo predictivo que hace uso de varios logaritmos de aprendizaje sí mejora la predicción en la deserción estudiantil en la educación mencionada anteriormente. Finalmente, se concluyó que, de los algoritmos empleados, CATBOOST es el que nos brinda unos niveles más altos en lo que respecta a los indicadores seleccionados. Es así como tenemos un 97% en exactitud, 70% en precisión y 74% en sensibilidad.

Palabras clave: machine boosting, kdd, ebr, precisión, exactitud, sensibilidad, catboost.

Abstract

The purpose of this project was to determine the improvement of the predictive machine learning model using the Boosting method in the prediction of student dropout in EBR (Regular Basic Education), KDD (Knowledge Discovery in Database) was used as a methodology and to the measurement was made using three indicators: Precision, Sensitivity and Accuracy. As a final result, we obtained that the predictive model that makes use of several learning logarithms does improve the prediction of student dropout in education mentioned above. Finally, it was concluded that, of the algorithms used, CATBOOST is the one that provides us with higher levels regarding the selected indicators. This is how we have 97% accuracy, 70% precision and 74% sensitivity.

Keywords: machine boosting, kdd, ebr, precision, sensitivity, accuracy, catboost.

I. INTRODUCCIÓN

La deserción estudiantil en la educación básica regular es una problemática que afecta a muchos sistemas educativos en todo el mundo (Coussement et al., 2020). Existen diversas razones que contribuyen a la deserción estudiantil en la educación básica. Una de ellas es la falta de recursos económicos en los hogares, lo que dificulta que las familias puedan cubrir los costos asociados con la educación, como libros, transporte y materiales educativos (Saccaro et al., 2020).

A nivel internacional, según la Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO, 2022), la deserción estudiantil es un obstáculo para lograr los objetivos de desarrollo sostenible que se busca en la educación. En ese sentido, (Chung y Lee, 2019) destacan los 132 millones de jóvenes que no están escolarizados en todo el mundo. Esta cantidad es superior al doble del número de mujeres que no asisten a la escuela (127 millones) y constituye un gran porcentaje de la población joven del mundo.

El uso de la tecnología en la lucha contra la deserción estudiantil abarca diferentes áreas y enfoques. En primer lugar, las plataformas y aplicaciones digitales han proporcionado a los estudiantes acceso a recursos educativos en línea, lo que les permite aprender de manera autónoma y a su propio ritmo (Bognár y Fauszt, 2022). Estas herramientas usan aplicativos inteligentes que apoyan de manera directa e indirecta a los docentes y estudiantes. De igual manera, se dan muchas investigaciones que apoyan a esta necesidad educativa, una de ella es el uso de machine learning en la deserción estudiantil (Mariano et al., 2022) .

En ese sentido, Henríquez y Vargas (2022) señalan que el uso de modelos predictivos de machine learning para abordar la deserción estudiantil en la educación básica regular ha ganado relevancia en los últimos años (Asish et al., 2022). Estos modelos aplican técnicas de análisis de datos y aprendizaje automático para identificar patrones que pueden predecir qué estudiantes corren un mayor riesgo de abandonar sus estudios.

A nivel de Latinoamérica, la Comisión Económica para América Latina y el Caribe (CEPAL, 2020) menciona que la deserción estudiantil tiene consecuencias

negativas tanto a nivel individual como a nivel social, perpetuando la desigualdad y la pobreza. A nivel social, la deserción escolar impide el desarrollo sostenible de los países, ya que reduce la mano de obra calificada y obstaculiza el progreso económico. Orsoni et al. (2023) también señalan que la proporción de personas que viven en la pobreza severa aumentó del 7,8 al 11,3 %, y que la tasa general de pobreza aumentó del 27,5 al 30,5 %.

En ese sentido, Arroyo-Hernández (2020) señala que al procesar grandes cantidades de datos históricos, los modelos predictivos de machine learning pueden identificar patrones sutiles y correlaciones que los humanos podrían pasar por alto. Esto permite a los educadores y responsables de políticas educativas tomar decisiones más informadas y personalizadas en cuanto a la implementación de intervenciones preventivas y programas de apoyo específicos para cada estudiante en riesgo.

A nivel nacional, de acuerdo con la Dirección Regional de Educación de Lima Metropolitana (DRELM, 2020) una problemática relacionada con la deserción estudiantil en la educación básica regular es la desigualdad socioeconómica y la falta de acceso equitativo a la educación. Por otro lado, muestra que en 2020 se redujo a la mitad el número de estudiantes que abandonaron la escuela en Lima Metropolitana, con una tasa de retorno del 52,8% en las escuelas con mayores índices de interrupción estudiantil y el retorno de casi 39.000 alumnos matriculados en Educación Básica Regular.

En ese sentido, Apaza-Tarqui et al. (2022) señalan que al utilizar un modelo predictivo, las instituciones educativas pueden adoptar un enfoque proactivo para la prevención de la deserción escolar. Al identificar a los estudiantes en situación de riesgo, se pueden implementar estrategias de intervención temprana, como tutorías personalizadas, programas de apoyo académico y emocional, orientación vocacional y actividades extracurriculares que fomenten la motivación y el compromiso con la educación.

Igualmente, Contreras-Bravo et al. (2021) indican que es importante tener en cuenta que el uso de modelos predictivos de machine learning para la deserción estudiantil plantea desafíos éticos y de privacidad. Es crucial proteger la privacidad

de los estudiantes al mismo tiempo que se utilizan estos modelos como un recurso para la toma de decisiones que tiene en cuenta el conocimiento de los expertos en el campo.

A nivel local, una problemática de no utilizar un modelo predictivo de machine learning para la deserción estudiantil en una institución educativa es la falta de identificación temprana de los estudiantes en riesgo de abandonar sus estudios. Sin un enfoque predictivo basado en datos, la institución se perdería la oportunidad de intervenir y proporcionar el apoyo necesario para retener a esos estudiantes (Aranciaga y Ccanto, 2021). Esto podría llevar a consecuencias negativas, como un aumento en las tasas de deserción estudiantil y un impacto en la calidad educativa. Al no identificar a los estudiantes en riesgo, la institución podría perder la oportunidad de implementar estrategias de prevención y de apoyo individualizado para abordar los desafíos que puedan enfrentar los estudiantes, ya sean académicos, sociales o personales (Hoyos y Aponte-Novoa, 2019).

Por otro lado, según Rico y Gaytán (2022) la escuela podría tomar mejores decisiones e implementar intervenciones más efectivas si utilizara un modelo predictivo de aprendizaje automático para determinar las causas de la deserción de los estudiantes. Esto podría incluir la asignación de recursos adicionales, la implementación de programas de tutoría, el seguimiento individualizado de los estudiantes en riesgo y la colaboración con otros actores relevantes, como los padres o tutores.

En relación con lo indicado previamente, se formuló como problema general: ¿Cómo el modelo predictivo de machine learning utilizando el método Boosting de ensemble mejora la predicción de la deserción estudiantil en EBR? Asimismo, se plantean los problemas específicos: a). ¿Cuán preciso es el modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR?, b). ¿Cuán exacto es el modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR? y c). ¿ Cuáles son los niveles de sensibilidad del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR?

La investigación se justifica de manera teórica, radica en la necesidad de desarrollar un modelo predictivo de machine learning específicamente diseñado para la deserción estudiantil en la Educación Básica Regular (EBR). Si bien existen investigaciones previas sobre modelos predictivos para la deserción estudiantil, es importante adaptarlos a las características y particularidades propias de la EBR.

Con relación a la justificación práctica, esta investigación tendrá tener un impacto significativo al permitir la identificación temprana de estudiantes en riesgo, una asignación más eficiente de recursos, una mejora en la calidad educativa y una toma de decisiones basada en datos. Además, contribuiría al campo de la educación y el machine learning al aplicar y evaluar la efectividad de este enfoque en un contexto específico. De igual manera, con relación a la justificación social de esta investigación radica en su potencial para reducir la desigualdad educativa, promover la inclusión y la diversidad, mejorar la calidad de vida de los estudiantes, fortalecer el sistema educativo y promover la investigación y la innovación educativa.

Asimismo, se formuló como objetivo general: Determinar la mejora del modelo predictivo de machine learning utilizando el método Boosting en la predicción de la deserción estudiantil en EBR. Asimismo, se tiene los siguientes objetivos específicos: a). Determinar la precisión del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR, b). Determinar la exactitud del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR y c). Determinar cuáles son los niveles de sensibilidad del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR.

Finalmente, se formuló como hipótesis general: El modelo predictivo de machine learning utilizando el método Boosting de ensemble mejora la predicción de la deserción estudiantil en EBR. Asimismo, se tiene las hipótesis específicas: a). El modelo predictivo de machine learning utilizando el método Boosting de ensemble es preciso en la predicción en la deserción estudiantil en EBR, b). El modelo predictivo de machine learning utilizando el método Boosting de ensemble

es exacto en la predicción en la deserción estudiantil en EBR y c). Los niveles de sensibilidad del modelo predictivo de machine learning mejoran utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR.

II. MARCO TEÓRICO

El desarrollo del presente trabajo se basó en estudios previos de fuentes nacionales e internacionales, los cuales serán detallados a continuación:

De acuerdo con Niyogisubizo et al. (2022), en su investigación titulada: " *Using a two-tiered ensemble machine learning technique, we can predict whether or not a student would withdraw from a course at a university. New generalization based on stacking*". El propósito fue emplear el aprendizaje automático para reducir el número de estudiantes que abandonan los estudios. En términos de enfoque, presentamos un nuevo conjunto de apilamiento para predecir la deserción de los estudiantes mediante una combinación de RF, XGBoost, GB y FNN. Los resultados de recuperación del 0,93 logrados para el conjunto de apilamiento en el conjunto de prueba señalan que el modelo tiene la capacidad de prever aproximadamente el 93% de los casos de deserción, confirmando de esta manera que nuestras predicciones son correctas con respecto a los estudiantes universitarios luego de entregar el curso. Por lo cual, llegamos a concluir que, pese a ser un pequeño conjunto de datos, los indicadores seleccionados adecuadamente que no requieren acceso a los registros del sistema pueden ser beneficiosos si se evalúan diferentes métricas de rendimiento.

Conforme a Rodríguez et al. (2023), en su investigación titulada: "*Using the Optimal Probability Threshold Adjustment Method for Unbalanced Data, We Predict Graduate Students Will Drop Out Later*", donde tuvo como propósito contrastar los beneficios de la técnica de ajuste de umbral de probabilidad óptima con otras técnicas de procesamiento de datos desbalanceados en su aplicación a la predicción de la deserción tardía de estudiantes de posgrado en cursos a distancia en dos universidades de la Iberoamérica. Fue una investigación de tipo aplicado, nivel descriptivo; enfoque cuantitativo y diseño no experimental que se aplicó en una muestra de 10.934 estudiantes. La técnica fue el análisis documental y el instrumento la ficha bibliográfica. Los resultados muestran que el clasificador Random Forest demostró ser robusto al manejar datos desequilibrados, obteniendo métricas similares a los tres mejores modelos encontrados. Esto se logró con un umbral óptimo de 0.427, lo que nos indica que seleccionar el umbral de probabilidad

adecuado es una excelente alternativa a las técnicas de remuestreo con umbrales variables. Con todo lo mencionado, se concluye que el clasificador Random Forest demostró su robustez al manejar datos desequilibrados, obteniendo el mejor rendimiento con un umbral predeterminado de 0.5. Logró un valor de recall de 0.47 y un puntaje f1 de 0.53.

En el trabajo de investigación de Khoushehgir y Sulaimany (2023), el cual se titula: "*Predicting unfavorable links in MOOCs to cut down on student attrition"*. El objetivo principal fue proponer un método novedoso con un algoritmo de predicción de enlace negativo de baja complejidad para la deserción de estudiantes. Fue una investigación de tipo aplicado, enfoque cuantitativo; diseño experimental y nivel explicativo. La muestra fue conformada por 79,186 estudiantes. La técnica fue el análisis documental y el instrumento la ficha bibliográfica. Los resultados muestran que los algoritmos de predicción de enlaces no supervisados tienen el mejor desempeño con un valor ROC de 0.6178 entre los demás. Considerando lo mencionado, se concluye que el método propuesto logra un rendimiento significativo en comparación con los de línea de base.

En el estudio de Panagiotakopoulos et al. (2021) que lleva como título: "*The use of supervised learning and hyperparameter optimization for early dropout prediction in massive open online courses"*. Se planteó como propósito emplear algoritmos de aprendizaje automático supervisado de última generación para predecir la deserción de los estudiantes en un MOOC para profesionales de ciudades inteligentes en una etapa temprana. Fue una investigación de tipo aplicado, enfoque cuantitativo; diseño experimental y nivel explicativo. La muestra fue conformada por 3029 estudiantes. La técnica fue el análisis documental y por otro lado el instrumento la ficha bibliográfica. Los hallazgos revelan que LightGBM es el modelo de mejor rendimiento. La precisión y la puntuación oscilan entre el 91 % y el 95,58 % y entre el 93,16 % y el 96,34 %, respectivamente, lo que demuestra que se puede realizar una predicción muy precisa de los posibles abandonos después de la primera semana del curso.

De conformidad con Pérez y Rojas (2020), en su investigación titulada: "*Diseño de un sistema para predecir la deserción de los alumnos mediante machine*

learning en la Universidad Tecnológica del Perú". Se realizó con el propósito diseñar un sistema de predicción de deserción estudiantil, mediante machine learning. Fue una investigación de tipo aplicado, enfoque cuantitativo; diseño experimental y nivel explicativo. Como muestra se consideró la información histórica de los estudiantes entre los años 2017 y 2019. La técnica fue el análisis documental y como instrumento la ficha bibliográfica. Los resultados muestran que el enfoque de machine learning es viable y puede utilizarse para predecir el abandono escolar con un nivel aceptable de precisión. Esto se basa en los resultados obtenidos durante el entrenamiento y la evaluación del modelo, que demuestran una capacidad predictiva significativa. Con lo dicho, se concluye que al utilizar datos de estudiantes que ya abandonaron la escuela como base de aprendizaje, el algoritmo SVM que se usa aquí puede predecir el comportamiento futuro de los estudiantes y, por extensión, identificar los factores más influyentes en la deserción de los estudiantes.

De forma similar, Luque y Sarazu (2019), en su investigación titulada: "*Modelo predictivo para determinar deserción de estudiantes en la Universidad Tecnológica del Perú*". El estudio se propuso crear un modelo predictivo que se pueda usar con las tecnologías existentes para administrar mejor y controlar la deserción de los estudiantes. Fue una investigación de tipo aplicado, nivel descriptivo; enfoque cuantitativo y diseño no experimental. La técnica fue la encuesta y el instrumento el cuestionario. Como muestra se tomaron a 20 estudiantes. Los resultados permitieron establecer un proceso mejor definido para identificar a los estudiantes en riesgo, brindando a la institución educativa una comprensión más clara de los factores que contribuyen a la deserción y cómo abordarlos. Asimismo, demostraron cómo la tecnología se puede utilizar efectivamente para mejorar el control y seguimiento de la deserción estudiantil. En consecuencia, se concluye que el modelo predictivo desarrollado utilizando técnicas de machine learning es viable y efectivo para predecir la deserción de los estudiantes.

En congruencia a lo indicado anteriormente, García (2020) en su investigación titulada: "*Detección de patrones de deserción estudiantil mediante aplicación de Árboles de Decisión C4.5 en el IESTP "Señor de Chocán" de*

Querecotillo", se planteó como objetivo examinar los patrones de deserción estudiantil utilizando árboles de decisión C4.5. Esta investigación fue de tipo aplicado, nivel explicativo; enfoque cuantitativo y diseño descriptivo. La muestra fueron 12500 registros de los estudiantes en los años 1990 al 2018. Se utilizó la observación como técnica y la ficha de registro de datos como instrumento. En relación con los resultados, el modelo se creó mediante validación cruzada y logró una impresionante tasa de precisión perfecta, es decir, del 100%. El proceso de entrenamiento tomó solo 0.02 segundos, considerando un conjunto de datos de 915 instancias. El rendimiento del árbol de decisión en la validación del modelo alcanzó un nivel de precisión del 81.53%. En conclusión, el modelo validado identificó con precisión las variables que contribuyen a la deserción estudiantil. Tener acceso a estos datos ayudará a la escuela en su futuro análisis y toma de decisiones.

De acuerdo con Shica (2022), en su investigación titulada: "*Modelos de Data Science para mejorar la detección de la deserción académica en la Institución Educativa 88331 en Chimbote - 2021*". Tuvo como propósito ver cómo los modelos de ciencia de datos afectaban la probabilidad de que los estudiantes abandonaran la escuela. Fue una investigación de tipo aplicado con un enfoque cuantitativo que se regía bajo un diseño experimental y nivel explicativo. Como muestra se consideró a 140 estudiantes. La técnica empleada fue el análisis documental y el instrumento correspondiente, la ficha de registro de datos. En relación con los resultados, se realizará una estimación para un modelo predictivo que evalúa el riesgo de abandono de los estudiantes utilizando árboles de decisión C4.5 en el software Weka. El modelo se creó mediante validación cruzada y logró una impresionante tasa de precisión del 100%. El proceso de entrenamiento tomó solo 0.02 segundos, considerando un conjunto de datos de 915 instancias. El rendimiento del árbol de decisión en la validación del modelo de predicción alcanzó un nivel de precisión del 81.53%. Las variables que influyen en la deserción de los estudiantes se predijeron con precisión utilizando el modelo validado, según los autores. Estos datos perspicaces se utilizarán para guiar el futuro análisis institucional y la toma de decisiones.

Prosiguiendo con Gutiérrez (2022), en su investigación titulada: "*Modelo predictivo para la deserción de estudiantes en el primer año de estudio en la*

Universidad Nacional Santiago Antúnez de Mayolo, Huaraz – 2022". El propósito fue determinar la deserción de estudiantes con un modelo predictivo. Fue una investigación de tipo aplicado, enfoque cuantitativo; diseño experimental y nivel explicativo. Como muestra se consideró a 6440 estudiantes. La técnica fue el análisis documental y el instrumento fue la ficha de registro de datos. Los resultados muestran que el modelo Gradient Boosting demostró un rendimiento sobresaliente entre los modelos evaluados. Alcanzó una alta precisión del 94%, sensibilidad del 86% y un sólido puntaje F1 del 90%. Además, obtuvo una impresionante exactitud del 95%. En los datos de entrenamiento, logró un destacado puntaje R-cuadrado del 75.12%, mientras que en los datos de prueba obtuvo un respetable puntaje R-cuadrado del 70.09%. Por ende, se concluye que utilizar algoritmos de aprendizaje automático permite predecir la deserción estudiantil durante su primer año de estudio académico.

Con respecto a la metodología, se empleó KDD (Knowledge discovery in database). Moine (2013) menciona que, "KDD es un proceso iterativo e interactivo". Se hace referencia a "iterativo" para indicar la posibilidad de repetir alguna fase con el objetivo de adquirir conocimiento de alto calibre, y se utiliza "interactivo" debido a que el experto en el campo busca contribuir en la preparación de datos y en la validación del conocimiento. Para un mayor detalle de cada etapa contemplada en la metodología KDD, se tiene la figura 1:

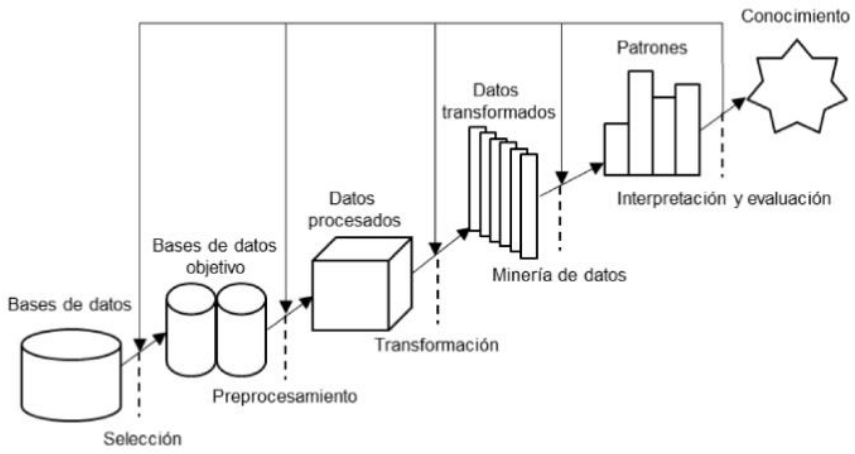


Figura 1: Fases de la metodología KDD

Fuente: Elaboración propia

En la primera fase, se realiza la selección y luego la comprensión de la problemática. Una vez se tengan listos estos pasos, se crearán una matriz de datos que servirán como cimientos para la búsqueda de nuevo conocimiento (Reyes, 2005).

En una segunda fase se realiza el preprocesamiento o limpieza de la data, este proceso se centra en el análisis de la calidad de datos. Se retiran o suprimen aquellos ruidosos, duplicados o nulos (Moine, 2013).

La tercera fase se centra en la transformación/reducción y busca eliminar variables no influyentes según los objetivos del proceso. Se emplean técnicas de reducción para disminuir el número de variables (Moine, 2013).

La cuarta etapa, la minería de datos, utiliza la vista minable generada en la fase anterior y aplica técnicas para descubrir patrones o reglas (Reyes, 2005).

La quinta y última fase, la etapa de interpretación evalúa el conocimiento descubierto con la posibilidad de integrarlo en otro sistema (Reyes, 2005).

Bases teóricas

Machine Learning

El término machine learning se puede relacionar con la teoría del aprendizaje estadístico. De acuerdo con Trujillo et al. (2022), en esta teoría se considera que los datos son una muestra de una distribución más amplia y se busca estimar los parámetros de esta distribución para hacer inferencias sobre nuevos datos. Se utilizan técnicas de inferencia estadística como la estimación de máxima verosimilitud y el enfoque de máxima a posteriori para obtener estimaciones óptimas de los parámetros. Además, la teoría del aprendizaje estadístico se preocupa por medir la incertidumbre en las predicciones y proporcionar estimaciones de la confiabilidad de los resultados.

De igual manera, se puede relacionar este término en la teoría del aprendizaje computacional. De acuerdo con Mantilla y Negre (2021), en esta teoría se investiga cómo los algoritmos de aprendizaje automático pueden aprender a

partir de los datos y adaptarse a diferentes escenarios. Se estudian conceptos relacionados con la complejidad computacional, la teoría de la aproximación y la teoría de la optimización para comprender los recursos computacionales necesarios y los límites de los algoritmos de aprendizaje. Además, la teoría del aprendizaje computacional se preocupa por la selección y diseño de características (feature selection y feature engineering) que permitan representar eficientemente los datos e incrementar el rendimiento de los modelos de aprendizaje.

Asimismo, la teoría del aprendizaje inductivo, según Taborda y López (2020), se inicia desde un conjunto de ejemplos de entrenamiento que contienen características y etiquetas asociadas. El objetivo es aprender una regla general que pueda predecir las etiquetas de nuevos ejemplos no vistos. La teoría del aprendizaje inductivo se apoya en principios de lógica y razonamiento inductivo para extraer conclusiones a partir de los datos. Se basa en la idea de que los ejemplos observados son representativos de una población más amplia y que las reglas aprendidas a partir de esos ejemplos pueden aplicarse a nuevos casos.

También, se considerará la teoría del aprendizaje bayesiano, el cual, de acuerdo con Terreros et al. (2019), se basa en el teorema de Bayes y la probabilidad condicional para realizar inferencias y tomar decisiones. Se centra en el uso de modelos probabilísticos para representar y actualizar el conocimiento a medida que se disponen de nuevos datos. En el aprendizaje bayesiano, se utiliza una distribución de probabilidad inicial, llamada distribución prior, que representa el conocimiento o las creencias iniciales sobre los parámetros del modelo. A medida que se obtienen nuevos datos, se actualiza esta distribución utilizando el teorema de Bayes, para obtener la distribución posterior, que representa el conocimiento actualizado después de haber observado los datos.

En el presente trabajo, se tratarán una serie de modelos basados en árboles de decisión, específicamente del tipo Gradient Boosting. Las predicciones se almacenan en las hojas del árbol a las que se puede acceder siguiendo las ramas del árbol que están representadas por consultas o condiciones relativas a las características de entrada (Bemthuis et al. 2023).

El método de Boosting, de acuerdo con Kaur et al. (2021), es una técnica utilizada en machine learning para mejorar el rendimiento de los modelos predictivos que consiste en combinar varios modelos de aprendizaje débiles (weak learners) para formar un modelo fuerte (strong learner) con mayor capacidad de generalización y precisión en las predicciones. Reforzando lo mencionado, tenemos a Theerthagiri (2022), el cual sostiene que los modelos débiles se construyen secuencialmente, ya que cada modelo se enfoca en corregir los errores cometidos por los modelos anteriores. En otras palabras, en cada iteración se da más importancia a las instancias que fueron clasificadas como erróneas previamente, permitiendo que los modelos posteriores se centren más en ellas y mejoren su rendimiento, tal como se puede visualizar en la figura 2, presentada a continuación:

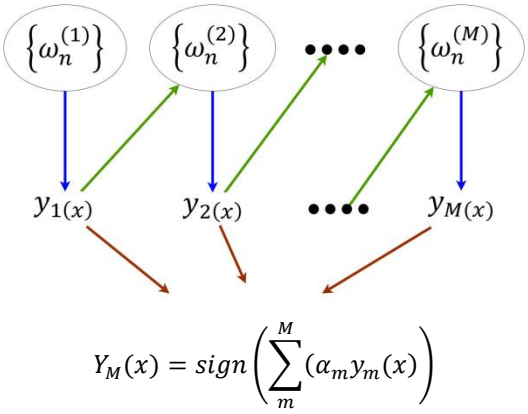


Figura 2: Método Boosting

Fuente: Elaboración propia

Asimismo, Korniichuk y Boryczka (2021) menciona que el algoritmo de Boosting más conocido es AdaBoost (Adaptive Boosting), que ajusta los pesos de las instancias de entrenamiento en cada iteración para darles más importancia o menos importancia según su dificultad de clasificación. Para efectos de este trabajo, se emplearán los algoritmos XGBoost, LightGBM y CatBoost. En ese sentido, Wang et al. (2023) añaden que el método de Boosting ha demostrado ser efectivo en una amplia gama de problemas de aprendizaje automático, incluyendo clasificación, regresión y detección de anomalías. Su enfoque en mejorar continuamente el rendimiento de los modelos débiles lo convierte en una técnica poderosa para mejorar la precisión de los modelos predictivos.

Profundizando en los algoritmos indicados, tenemos que al XGBoost, el cual, de acuerdo con Marchant (2022), surge como una mejora del método denominado Gradient Boosting y ofrece diferentes características destacables, tales como la inclusión de términos de penalización que ayudan a evitar un sobreajuste, reducción que se realiza de forma proporcional en las hojas en cada árbol, método Newtib-Raphson que tiene como función permitir la pérdida e implementación de forma eficiente en el entrenamiento de múltiples procesadores. Por otro lado, el funcionamiento básico de este modelo se realiza así:

Primero, se reciben los inputs que se determinan como un grupo de entrenamiento ($\{(x_i, y_i)\}_{i=1}^{n}$), una cantidad variable de iteraciones (M), una tasa de aprendizaje ($\alpha \in R^{+}$) y una función que represente la pérdida que sea dos veces diferenciable como mínimo.

Luego de ello, el algoritmo inicia con un valor constante como se ve a continuación:

$$f_0(x) = argmin_\theta \sum_{i=1}^{n} L(y_i, \theta)$$

Para el caso de M y n, en estos se calculan los gradientes considerados como de primer y segundo orden.

$$g_m(x_i) = \left[\frac{\partial L(y_i, f(x_i)}{\partial f(x_i)}\right]_{f(x)=f_{m-1}(x)}$$

$$h_m(x_i) = \left[\frac{\partial L(y_i, f(x_i)}{\partial f(x_i)}\right]_{f(x)=f_{m-1}(x)}$$

Seguido a ello, procedemos a calcular el resultado que nos brinda un árbol por medio del conjunto $\left\{x_i - \frac{g_m(x_i)}{h_m(x_i)}\right\}_{i=1}^{n}$, al dar con la solución de esta ecuación:

$$\varphi_m = argmin_\varphi \sum_{i=1}^{n} \frac{1}{2} h_m(x_i) \left[-\frac{g_m(x_i)}{h_m(x_i)} - \varphi(x_i)\right]^2$$

$$f_m(x) = \alpha\varphi_m(x)$$

Finalmente, se procede a actualizar la predicción de esta forma:

$$f_{(m)}(x) = f_{(m-1)}(x) + f_{(m)}(x)$$

y obtenemos la predicción final:

$$f_{(M)}(x) = \sum_{m=0}^{M} f_m(x)$$

Como segundo algoritmo tenemos al CatBoost, que, de acuerdo con Jara (2021), tiene como objetivo el procesamiento eficiente de diferentes variables categóricas, evitando que se realicen procesos previos. Si bien cada modelo cuenta con sus pros y contras, el modelo mencionado cuenta con una gran ventaja al momento de usarlo en datos que posean atributos de tipo categórico. Lo mencionado se respalda en la forma de encoding que tiene incorporada que se divide en One Hot Encoding (eficiente al tener atributos con pocas categorías) y en el caso de contar con categorías con una cardinalidad elevada, se procede a evaluarlas bajo la siguiente fórmula:

$$\bar{x}_k^i = \frac{\sum_{x_{j\in D_k}} 1_{\{x_j^i = x_k^i\}} . y_i + ap}{\sum_{x_{j\in D_k}} 1_{\{x_j^i = x_k^i\}} + a}$$

$$D_k \subset D \backslash \{x_k\}$$

Donde:

$\bar{x}_k^i$: Categoría k-ésima de “x” en el registro “i”.

D_k: Data recorrida en toda la categoría “k”, sin incluir el registro actual.

p: Prioridad a priori.

a: Parámetro.

Como último algoritmo por explicar tenemos a LightGBM, que, tal como nos indica Luna (2021), está diseñado para trabajar ágilmente y ser eficiente en cuanto a términos de precisión, uso de memoria, CPU y los datos a gran escala. A diferencia de otros algoritmos, éste genera un árbol creciente de forma vertical que

comienza a crecer en función a la hoja que le genere la mayor cantidad de pérdida delta para tratar de reducirla. Es así como obtenemos la figura 3:

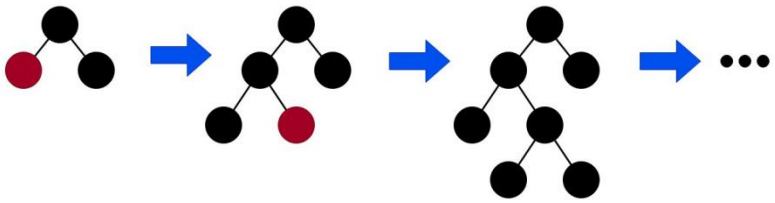


Figura 3: Construcción de hojas por LightGBM

Fuente: Elaboración propia

Vale indicar que este algoritmo es muy sensible al sobreajuste, lo cual puede conllevar a que ejecute ello en pequeños conjuntos.

Pasos para implementar el método de Boosting

Existe una serie de pasos necesarios recomendados para implementar este método en un proyecto Machine Learning. De acuerdo con Kaur et al. (2021) se deben cumplir los siguientes pasos:

- **Paso 1: Preparación de los datos**

  En primer lugar, se deben recopilar y preparar los datos para el modelado. Esto implica tener un conjunto de datos etiquetados que incluya variables de entrada y la variable objetivo que se desea predecir.

- **Paso 2: División del conjunto de datos**

  El siguiente paso es tener 2 subconjuntos de datos, uno para el entrenamiento y otro para las pruebas.

- **Paso 3: Selección de un algoritmo de Boosting**

  En este paso, es necesario elegir un método de refuerzo para el modelo de predicción. Xgboost, Catboost y lightGBM son solo algunas de las opciones conocidas. Investigar y comprender a fondo el algoritmo elegido es esencial debido al hecho de que tiene sus propias propiedades y factores únicos. Para el presente trabajo, se realizará el entrenamiento con los 3 modelos mencionados para comparar los resultados de cada uno de ellos.

- **Paso 4: Configuración del modelo**

  Una vez seleccionado el algoritmo, es necesario configurar los hiperparámetros del modelo. Estos incluyen la tasa de aprendizaje, el número de estimadores (número de modelos débiles) y la profundidad máxima del árbol (en el caso de Gradient Boosting). La elección adecuada de estos hiperparámetros puede afectar el rendimiento y la complejidad del modelo.

- **Paso 5: Entrenamiento del modelo**

  En esta etapa, se procede a entrenar el modelo utilizando el conjunto de entrenamiento. El modelo aplica un aprendizaje de forma iterativo, ajustando los valores de los datos y creando nuevos modelos débiles en cada iteración para mejorar el rendimiento global.

- **Paso 6: Evaluación del modelo**

  Después de entrenar el modelo, se debe evaluar su eficacia en el conjunto de prueba. Se evalúa la precisión, exactitud y especificidad del modelo.

- **Paso 7: Predicciones**

  Una vez que se esté satisfecho con el rendimiento del modelo, se puede utilizar para realizar predicciones en nuevos datos. Simplemente se deben proporcionar las variables de entrada al modelo y se obtendrán las predicciones correspondientes.

- **Paso 8: Entendimiento del modelo**

  Se realizó un análisis descriptivo para comprender que variables son las más importantes en la predicción de cada modelo y cuál es el peso que les asigna en cada uno de estos.

Fórmulas matemáticas para calcular las predicciones y ajustar los pesos de los modelos débiles

De acuerdo con Korniichuk y Boryczka (2021), se usan las siguientes fórmulas:

- **Fórmula de actualización de pesos:**

En cada iteración del Boosting, se actualizan los pesos de los datos de entrenamiento para dar mayor importancia a las muestras clasificadas

incorrectamente por los modelos débiles anteriores. La fórmula general para actualizar los pesos puede variar según el algoritmo de Boosting utilizado, pero en general sigue un enfoque similar al siguiente:

$$Nuevo\ peso = Peso\ anterior\ \times Factor\ de\ actualización$$

Vale indicar que el factor de actualización se calcula en función de la clasificación errónea de la muestra por parte del modelo débil.

- **Fórmula de combinación de modelos débiles:**

En Boosting, los modelos débiles se combinan para formar un modelo fuerte. La forma en que se combinan puede variar según el algoritmo de Boosting. Por ejemplo, en AdaBoost, los modelos débiles se ponderan en función de su rendimiento durante el entrenamiento, mientras que en Gradient Boosting se utiliza una combinación aditiva de los modelos débiles.

$$PMF = SPMD \times PAMD$$

Donde:
PMF: Predicción del modelo fuerte
SPDM: Suma de las predicciones de los modelos débiles.
PAMD: Peso asignado a cada modelo débil

- **Fórmula de cálculo de errores:**

Durante el entrenamiento del modelo Boosting, se calcula el error residual entre las predicciones del modelo fuerte y las etiquetas verdaderas. Este error residual se utiliza para ajustar el modelo en las iteraciones posteriores. La fórmula general para calcular el error residual puede variar según el algoritmo de Boosting, pero suele ser una diferencia o una función de pérdida específica.

$$ER = EV - PMF$$

Donde:
ER: Error Residual
EV: Etiqueta verdadera

PMF: Predicción del modelo fuerte

Gráficamente, la metodología Boosting se puede representar con la siguiente figura 4:

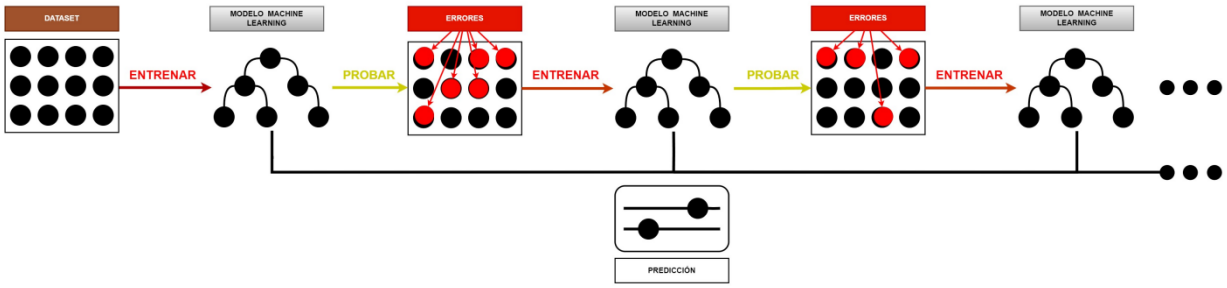


Figura 4: Aproximación secuencial con el método Boosting

Fuente: Elaboración propia

Deserción estudiantil

Con respecto a las bases teóricas de la deserción estudiantil. Se considerarán la teoría del capital humano, que, de acuerdo con Jiménez y Cota-Yañez (2019), se basa en la idea de que la educación es una inversión en capital humano y los individuos toman decisiones racionales sobre su educación en función de los costos y beneficios esperados. En el contexto de la deserción estudiantil, se analizan factores como la rentabilidad económica de la educación, la relevancia de los estudios para el mercado laboral y la influencia de los costos asociados a la educación.

De igual manera, la teoría del desarrollo humano, de acuerdo con Solís-Narváez (2022), se centra en la importancia del desarrollo integral de las personas, considerando aspectos como el bienestar emocional, social y cognitivo. En relación con la deserción estudiantil, se investigan factores como el apoyo socioemocional, la adaptación académica, la satisfacción con el entorno educativo y las oportunidades de participación y pertenencia.

También tenemos la teoría sociocultural, la cual, según Castanelli (2023), se basa en la idea de que el entorno social y cultural influye en el proceso educativo. Se consideran factores como el nivel socioeconómico, la estructura familiar, el

contexto comunitario y las normas culturales y sociales que pueden afectar la permanencia o abandono de los estudios.

Finalmente, la teoría del capital social, que, de acuerdo con Barrios-Hernández et al. (2021), se enfoca en la importancia del apoyo social en el proceso educativo. Se investigan factores como la calidad de las relaciones con profesores y compañeros, el apoyo familiar y comunitario, y la participación en actividades extracurriculares, ya que estas variables pueden influir en la motivación, el compromiso y la persistencia educativa.

Marco conceptual de la deserción estudiantil

De acuerdo con Vega et al. (2022), la deserción estudiantil es el fenómeno en el que los estudiantes abandonan sus estudios antes de completar su formación académica. Asimismo, puede estar influenciada por diversos factores personales o del entorno del estudiante. De igual manera, Viloria y Lezama (2019) señalan que la deserción estudiantil es el abandono prematuro de la educación formal por parte de los estudiantes, lo cual no se limita únicamente a las primeras etapas de la escuela. También señala que la deserción escolar puede tener efectos desfavorables en la vida de los estudiantes involucrados y en la sociedad.

En el mismo sentido, Prenkaj et al. (2020) mencionan que la deserción estudiantil se refiere al proceso en el que los estudiantes interrumpen su participación en el sistema educativo sin completar los programas o cursos en los que están matriculados. Rodríguez Velasco et al. (2023) indican que está relacionada con diversos factores personales o del entorno, tales como la falta de apoyo económico, falta de interés o carencia de recursos educativos adecuados. Indica también que la deserción estudiantil es un desafío significativo en el campo de la educación y requiere intervenciones y políticas efectivas para abordar este problema.

Con relación a las herramientas tecnológicas que se utilizan para abordar y prevenir la deserción estudiantil se encuentran los sistemas de gestión educativa. De acuerdo con Sánchez y Delgado (2020), los sistemas de información estudiantil (SIS) y los sistemas de gestión del aprendizaje (LMS), permiten recopilar y analizar

datos relacionados con el rendimiento académico, la asistencia, el comportamiento y otros indicadores clave. Estos sistemas facilitan el seguimiento individualizado de los estudiantes, identificando posibles señales de riesgo de deserción y brindando intervenciones tempranas.

De igual manera, la analítica de datos, según Apaza-Tarqui et al. (2022), el análisis de datos educativos, utilizando técnicas de minería de datos y aprendizaje automático, ayuda a identificar patrones y tendencias que pueden predecir la deserción estudiantil mediante la recopilación y análisis de datos de múltiples fuentes, como registros académicos, datos demográficos y de comportamiento.

También, las plataformas de seguimiento y apoyo estudiantil, de acuerdo con Perchinunno et al. (2021), existen plataformas específicas diseñadas para brindar apoyo y seguimiento a los estudiantes en riesgo de deserción. Estas herramientas pueden incluir funciones de comunicación y colaboración, recursos educativos personalizados, seguimiento de progreso académico, recordatorios y alertas, y acceso a servicios de apoyo como tutorías y asesoramiento académico. De igual manera, tenemos a los sistemas de alerta temprana, los cuales, según Prenkaj et al. (2020), utilizan algoritmos y modelos predictivos que permiten identificar a los estudiantes en riesgo de deserción y generar alertas para los administradores, tutores y profesores. Estas alertas permiten intervenir de manera oportuna y proporcionar el apoyo necesario para mejorar la retención y el éxito estudiantil.

La información por utilizar para mejorar la predicción de la deserción estudiantil va a provenir de los datos personales. De acuerdo con (Otero, 2021), se refiere al tratamiento de la información individual sobre los estudiantes, como su identidad, características demográficas y socioeconómicas, antecedentes educativos, historial académico, datos familiares y cualquier otro dato personal relevante. Estos datos se utilizan para comprender mejor a los estudiantes, identificar factores de riesgo y establecer perfiles individuales que puedan ayudar a predecir y abordar la deserción estudiantil.

De igual manera, los datos académicos, Pachay-López y Rodríguez-Gámez (2021) señalan que se refiere a la recopilación y análisis sobre la información recolectada acerca del rendimiento académico y el progreso de los estudiantes en

su trayectoria educativa. Estos datos incluyen calificaciones, promedios, asistencia, participación en actividades extracurriculares, resultados de exámenes estandarizados y cualquier otro indicador académico relevante.

Asimismo, los datos institucionales, Rueda et al. (2020) señalan que hacen referencia a la recopilación, así como el análisis de información relacionada con la institución educativa en la que los estudiantes están matriculados. Estos datos incluyen características y políticas institucionales, recursos disponibles, programas de apoyo, servicios estudiantiles y cualquier otro factor relacionado con el entorno institucional.

Finalmente, Sánchez (2021) señala que el término "datos socioeconómicos" implica desde la recopilación hasta la interpretación de la información sobre la situación financiera de los estudiantes y sus familias. Se incluyen los ingresos, la educación de los padres, el empleo de los padres, las condiciones de vivienda, el acceso a los servicios esenciales y otros indicadores socioeconómicos.

Métricas para evaluar la predicción de la deserción estudiantil

Con respecto a las métricas necesarias para evaluar la predicción de la deserción estudiantil, emplearemos una matriz de confusión que muestra los resultados obtenidos en forma matricial, tal como se puede observar en la tabla 1:

Tabla 1: Matriz de confusión

	Valor Predicho		
Valor Actual		Cumple	No cumple
	Cumple	Verdadero Positivo (TP)	Falso Negativo (FN)
	No cumple	Falso Positivo (FP)	Verdadero Negativo (TN)

Fuente: Elaboración propia

Con la herramienta mencionada podremos considerar una serie de métricas, entre estas seleccionamos la precisión. De acuerdo con Gil y Seguro (2022), la métrica seleccionada mide la calidad con la que se realizó el trabajo. Asimismo, se indica que la precisión se determina comparando la proporción de muestras que se etiquetaron correctamente con el número total de muestras utilizadas en el estudio.

$$Precisión = \frac{TP}{(TP + FP)}$$

Continuando con las métricas, también tenemos a la exactitud, la cual, de acuerdo con Kersting (2018), indica que esta medida se utilizar para medir el rendimiento de una herramienta sobre la actividad que realizó. Aparte de ello, la exactitud se determina dividiendo el recuento de predicciones precisas realizadas por el modelo utilizado con el número total de muestras.

$$Exactitud = \frac{(TP + TN)}{Total}$$

Igualmente, la métrica recall o sensibilidad, de acuerdo con Chinguel (2022), es una medida utilizada para evaluar la proporción de casos positivos evaluados correctamente por medio de la siguiente ecuación.

$$Sensibilidad = \frac{TP}{(FN+TP)}$$

III. METODOLOGÍA

3.1. Tipo y diseño de investigación

La investigación por realizar será de tipo aplicada. Ñaupas et al. (2018) resaltan que este tipo de investigación se orienta a responder los problemas que surgen en el contexto donde se aplica, entre los cuales puede ser la educación, tecnología, entre otros. El propósito principal es brindar soluciones consideradas fácilmente aplicables por medio del desarrollo de intervenciones o programas en los procesos existentes.

Asimismo, la investigación se realizará mediante el enfoque cuantitativo. Hernández-Sampieri y Mendoza (2018) señalan que se busca obtener datos con atributos que permitan establecer relaciones causales entre variables. Este enfoque se basa en la premisa de que los fenómenos se describen y miden por medio de números y que tanto los patrones como las relaciones numéricas se pueden evaluar estadísticamente.

Diseño de investigación

Con relación al diseño de investigación, será experimental. Bernal (2016) nos menciona que es un método de estudio científico en el que se manipulan una o más variables independientes para determinar qué impacto tienen sobre una variable dependiente, mientras que las demás variables se mantienen constantes. De igual manera dentro del diseño experimental, se considerará el preexperimental, con una sola medición. En ese sentido, Hernández-Sampieri y Mendoza (2018) menciona que este tipo de diseño es utilizado en ciertos estudios para obtener una idea inicial o exploratoria sobre una variable de interés. Este diseño, se realiza una sola medición antes y después de la intervención o tratamiento.

El nivel será explicativo, Baena (2017) indica que en este nivel se busca relacionar las variables considerando que la actualización en una variable causa un efecto visible en la otra.

Por otro lado, se considera que el corte longitudinal es el apropiado. Según, Castañeda (2022) consiste en seleccionar una muestra de participantes y recopilar

datos en diferentes tiempos que puede establecerse en base a intervalos definidos o irregulares. Permitiendo de esta manera, capturar la forma es que se desarrollan o cambian los aspectos de interés en base al tiempo con ello identificar los factores que influyen en los cambios visualizados. Con respecto al tipo de diseño seleccionado, se tiene la figura 5:

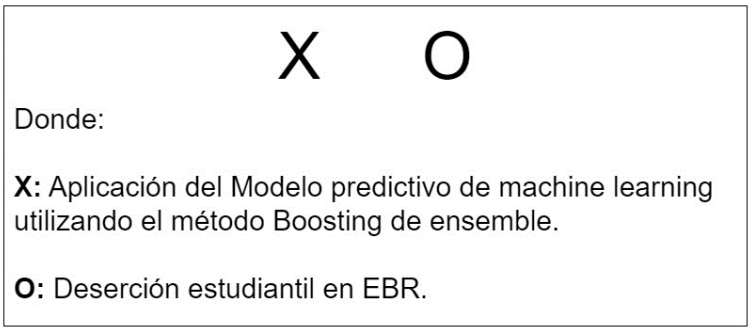


Figura 5: Diagrama de diseño experimental

Fuente: Elaboración propia

3.2. Variables y operacionalización

Variable independiente: Modelo predictivo de machine learning

- **Definición conceptual:** Según Tamada et al. (2022), el modelo predictivo de machine learning es un tipo de modelo que utiliza algoritmos y técnicas de aprendizaje automático para realizar predicciones o estimaciones sobre datos futuros o no vistos.

- **Definición operacional:** Para la construcción de este modelo se utilizarán: datos personales, datos académicos; datos institucionales y datos socioeconómicos. Dicha información es importante porque se necesita entrenar el modelo con datos históricos para hacer predicciones, logrando medir su rendimiento por medio de los indicadores seleccionados.

- **Indicadores:** Para medir el desempeño del modelo creado consideramos la actualización de pesos (Nuevo peso), la combinación de modelos débiles (PMF) y el cálculo de errores (ER).

- **Escala de medición:** Consideramos que la escala adecuada para esta variable es la de razón.

Variable dependiente: Predicción de la deserción estudiantil

- **Definición conceptual:** De acuerdo con, Viloria y Lezama (2019) mencionan que la deserción estudiantil se considera al abandono de la escuela antes de terminarla. Esto no se limita al nivel de la escuela primaria, sino que puede ocurrir en cualquiera.

- **Definición operacional:** Los resultados de la predicción de la deserción estudiantil se evaluarán por medio de una matriz de confusión, la cual nos ayudará a obtener los valores correspondientes para aplicarlos en los indicadores considerados para este proyecto.

- **Indicadores:** Los indicadores seleccionados son los siguientes: Precisión, Exactitud y Sensibilidad.

- **Escala de medición:** La escala seleccionada para esta variable es de razón.

3.3. Población, muestra y muestreo

3.3.1. Población: Arias (2020) menciona que "población" se usa para describir el conjunto de personas examinadas, mientras que "muestra" se usa para describir un subconjunto elegido al azar de la población con el fin de realizar el estudio. En nuestra investigación, definimos como población los registros de los estudiantes que pertenecen a los años 2014 a 2022.

3.3.2. Muestra: Baena (2017) señaló que la muestra en una investigación corresponde a un subconjunto elegido que se obtiene de una población de interés y que tiene como fin proporcionar información sobre la población en general. La selección de la muestra se tiene que realizar con las técnicas apropiadas de muestreo para garantizar la validez y la

generalización de los resultados obtenidos. En la investigación, se tomará el total de la población como muestra. Por lo cual, la muestra estará conformada por los registros de los estudiantes pertenecientes a los años 2014 a 2022.

3.3.3. Muestreo: De acuerdo con, Hernández y Carpio (2019), el muestreo se define como un proceso de selección de una muestra representativa de la población de interés. La muestra es una parte de la población que se usa para recopilar datos y deducir inferencias sobre la población en general. El muestreo por utilizar será probabilístico de tipo documental. En ese sentido, Martínez (2018) señaló que este tipo de muestreo es una técnica utilizada en la investigación para seleccionar y analizar documentos relevantes como fuentes de datos. Consiste en identificar, recopilar y examinar documentos que son pertinentes para la investigación y que proporcionan información valiosa para responder a las preguntas de investigación o alcanzar los objetivos planteados.

3.3.4. Unidad de análisis: Arias (2020) menciona que la unidad de análisis se refiere al objeto o entidad que se está estudiando y sobre el cual se recopilan datos y se realizan análisis. Por ello se considera como parte fundamental de observación en un estudio y puede variar dependiendo del tipo de investigación y del objetivo. Por lo cual, la unidad de análisis serán los registros de los estudiantes.

3.4. Técnicas e instrumentos de recolección de datos

Técnicas: La técnica que se usará será en análisis de registros. Sánchez et al. (2021) indicaron que dicha técnica se refiere a un método de investigación que consiste en examinar y analizar registros o documentos que contienen datos o información relevante para un estudio. Los registros médicos, las transcripciones escolares, los archivos de la empresa, los trabajos de investigación, los extractos bancarios y las declaraciones de impuestos son solo algunos ejemplos de los muchos tipos de registros que existen.

Instrumentos: No existe un instrumento como tal, para la recolección de la data se empleó una base y/o matriz de datos de los estudiantes en el año 2014 a 2022 importada directamente desde el sistema SIAGIE. De acuerdo con Sánchez et al. (2021) indican que la matriz de datos es utilizada en el análisis cuantitativo, ya que permite organizar los datos de manera estructurada y facilita su manipulación y cálculos estadísticos. A través de la matriz de datos, se pueden realizar diversas operaciones, como la suma, el promedio, la desviación estándar, la correlación, entre otros análisis estadísticos.

La plataforma SIAGIE, Sistema de Información de Apoyo a la Gestión de la Institución Educativa, es la fuente oficial de consultas que se utilizó para este proyecto debido a que todo registro expuesto está garantizado por el Ministerio de Educación. Este sistema, que viene siendo ejecutado por todos los centros educativos a nivel nacional, contiene restricciones con respecto al libre acceso de algunos datos sensibles que por motivos de fuerza mayor no se nos pudo brindar.

3.5. Procedimientos

En esta sección se detallan las fases que se realizaron para el desarrollo del Modelo Machine Learning para la predicción de deserción estudiantil en EBR, empleando el método KDD, la cual está compuesta de las siguientes etapas:

Etapa 1: Etapa de selección de datos

En primer lugar, se solicitaron los permisos y las autorizaciones necesarias de las instituciones educativas y autoridades correspondientes para acceder a los datos estudiantiles y poder así realizar el presente estudio. La data se extrajo desde la plataforma SIAGIE mediante la exportación de notas, nóminas y otros documentos similares.

En un segundo momento, se recopiló toda la información recolectada en una sola matriz de datos. Dicha tabla cuenta con 21 datos clasificados de la siguiente forma: de tipo personal (10 datos), de tipo parental (3 datos) y de tipo académico (8 datos). La estructura de estos registros se puede observar en el ANEXO 3. Cabe resaltar que el total de alumnos en la matriz fueron 1206.

Etapa 2: Etapa de preprocesamiento / limpieza de datos

Luego de la recolección de información se verificó la calidad, donde se halló en primera instancia que existían algunos campos con valores que no aportan información relevante al estudio como se aprecia en el ANEXO 5.

Debido a ello se procedió a eliminar estas variables por los siguientes motivos:

- 'tipo_discapacidad' , 'horas_semanales_labora' y 'segunda_lengua': alto porcentaje de valores nulos.
- ‘trabaja_estudiante’: Poca variabilidad, ya que solo tenía el valor ‘No’.
- ‘Nombre’ y ‘fec_retiro’: No son variables relevantes para la creación de los modelos.

Posteriormente, se realizó un cambio en la variable ‘situación_final’ en donde a todos aquellos registros que no tenían como valor ‘R’: retirado , se les consideró como ‘no_retirado’, tal como se ve en la figura 6:

```
# se considera a todo aquel diferente de 'R: retirado' como "no retirado"
situacion_final_ajust = {'RR':'no_retirado', 'A':'no_retirado', 'D':'no_retirado', 'T':'no_retirado','R':'retirado'}
ds_alumnos.replace({"situacion_final": situacion_final_ajust},inplace=True)
ds_alumnos.situacion_final.value_counts()

no_retirado    1130
retirado         76
Name: situacion_final, dtype: int64
```

Figura 6: Ajuste inicial en variable de situación final

Fuente: Elaboración propia

Finalmente se procedió a imputar los valores de las variables ‘lengua_materna’ y ‘escolaridad_madre’ ya que tenían algunos registros nulos. Se utilizó el método de imputación con la moda de las variables (‘S’: secundaria y ‘C’: Castellano respectivamente). En el siguiente par de imágenes (figura 7 y 8) tenemos la vista de las variables antes y luego de la imputación:

```
ds_alumnos.isnull().sum()

sexo                          0
situacion_matricula_inicial   0
pais                          0
padre_vive                    0
madre_vive                    0
lengua_materna                3
escolaridad_madre            17
nacimiento_registrado         0
edad_en_registro              0
prom_final                    0
areas_desaprobadas            0
porcentaje_inasitencias       0
grado_ult_registro            0
seccion_ult_registro          0
situacion_final               0
dtype: int64
```

Figura 7: Vista de variables anterior a la Imputación

Fuente: Elaboración propia

```
ds_alumnos['escolaridad_madre'] = ds_alumnos['escolaridad_madre'].fillna(value="S")
ds_alumnos['lengua_materna'] = ds_alumnos['lengua_materna'].fillna(value="C")
ds_alumnos.isnull().sum()

sexo                          0
situacion_matricula_inicial   0
pais                          0
padre_vive                    0
madre_vive                    0
lengua_materna                0
escolaridad_madre             0
nacimiento_registrado         0
edad_en_registro              0
prom_final                    0
areas_desaprobadas            0
porcentaje_inasitencias       0
grado_ult_registro            0
seccion_ult_registro          0
situacion_final               0
dtype: int64
```

Figura 8: Vista posterior a Imputación

Fuente: Elaboración propia

Etapa 3: Etapa de transformación y reducción

En esta etapa se realizó un Análisis exploratorio de las variables para entender el comportamiento que tienen respecto a la clase desertora y no desertora (‘retirado’ y ‘no_retirado’).

Descripción de variables:

En la tabla 2 podemos visualizar la descripción inicial para cada una de las variables**:**

Tabla 2: Descripción general de variables numéricas

Variable	count	mean	std	min	25%	50%	75%	max
edad_en_registro	1206.0	8.810116	1.905118	6.0	7.0	9.0	10.0	17.0
areas_desaprobadas	1206.0	1.234660	1.901864	0.0	0.0	0.0	2.0	7.0
porcentaje_inasistencias	1206.0	15.306799	4.586511	3.0	12.0	15.0	18.0	30.0

Fuente: Elaboración propia

- La edad promedio de los alumnos es de 8-9 años, y la edad máxima es 17 años.
- El número promedio de áreas desaprobadas es de 1-2 áreas, y la cantidad máxima es de 7.
- El porcentaje promedio de inasistencias es de 15% aproximadamente y el máximo de 30%.

En el caso de las variables categóricas, el detalle de cada una puede ser observado en la tabla 3, presentada a continuación:

Tabla 3: Descripción general de variables categóricas

Variable	**count**	**unique**	**top**	**freq**
sexo	1206	2	M	609
situacion_matricula_inicial	1206	4	P	860
pais	1206	2	P	1202
padre_vive	1206	2	SI	1138
madre_vive	1206	2	SI	1189
lengua_materna	1203	2	C	1195
escolaridad_madre	1189	4	S	828
nacimiento_registrado	1206	2	SI	1149
prom_final	1206	3	A	968
grado_ult_registro	1206	6	1-PRI	311
seccion_ult_registro	1206	8	H	266
situacion_final	1206	5	A	908

Fuente: Elaboración propia

- Se observa que se tiene una mayor cantidad de alumnos de género masculino.
- La situación inicial con mayor frecuencia es la de P: promovido.
- Tal y como se espera el país de origen de cada alumno es Perú en su mayoría.
- La mayoría tienen ambos padres vivos, tienen como lengua materna el castellano, no trabajan y tienen promedio final = 'A'.

Exploración de variables numéricas:

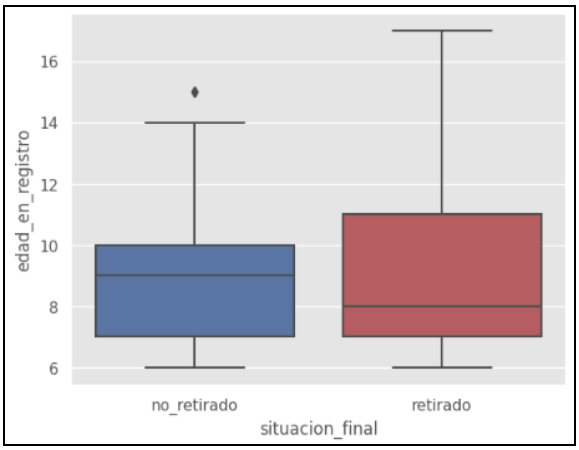


Figura 9: Boxplot de edad en el registro según situación final

Fuente: Elaboración propia

- Se observa que la edad promedio de alumnos retirados es 8, la de no retirados es de 9 aproximadamente, y la distribución de valores de retirados va desde los 6 a 17 años.

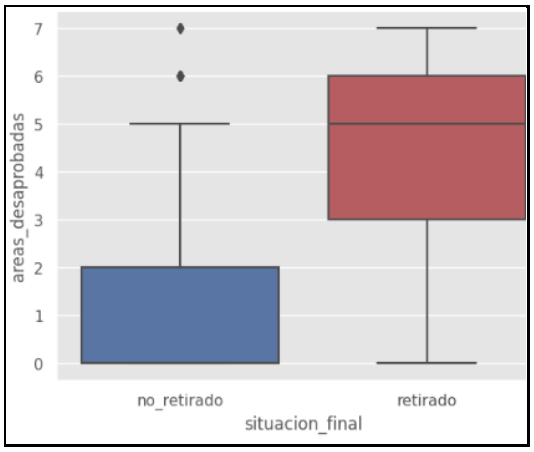


Figura 10: Boxplot de áreas desaprobadas según situación final

Fuente: Elaboración propia

- Se observa que la media de áreas desaprobadas de alumnos retirados es superior a la de no retirados, siendo de 5 áreas y 7 como valor máximo.

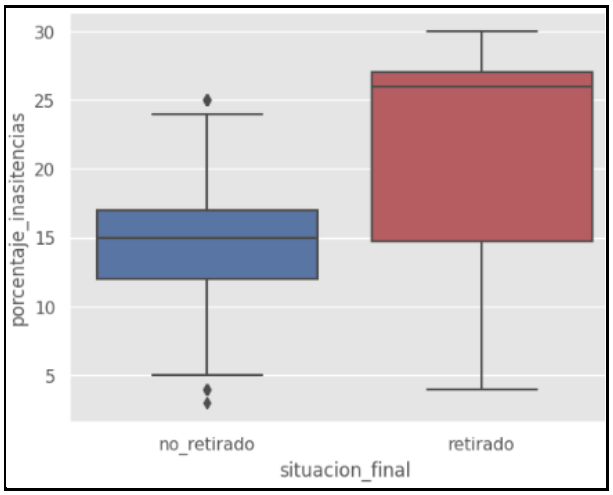
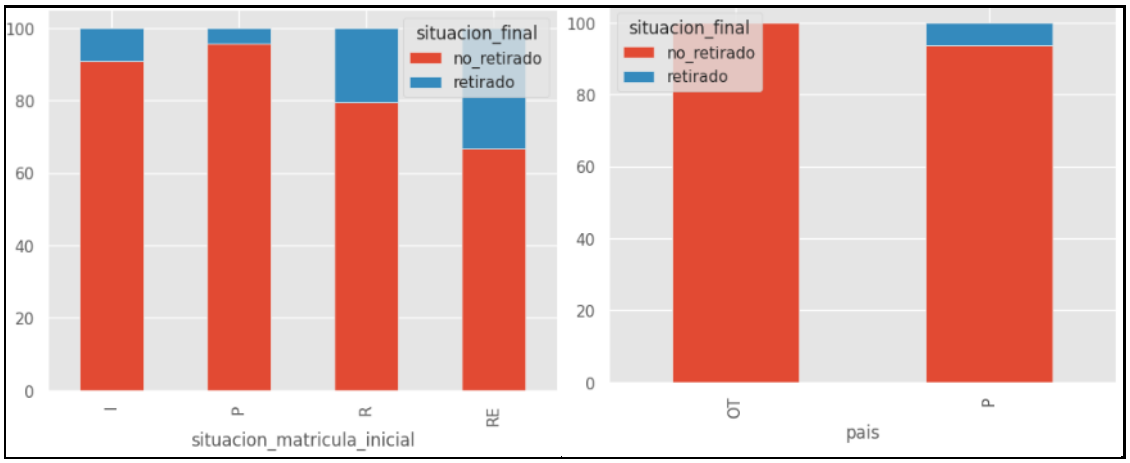


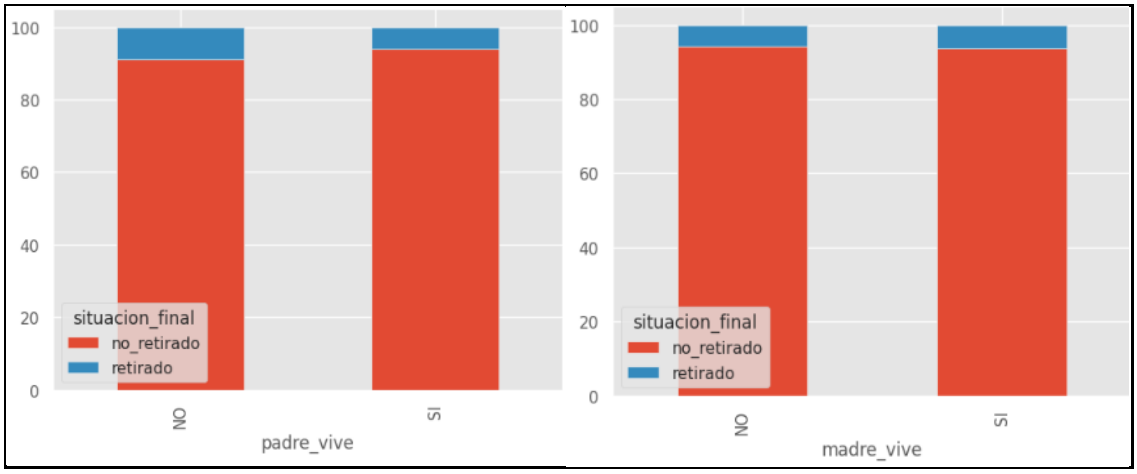
Figura 11: Boxplot de porcentaje de inasistencias según situación final

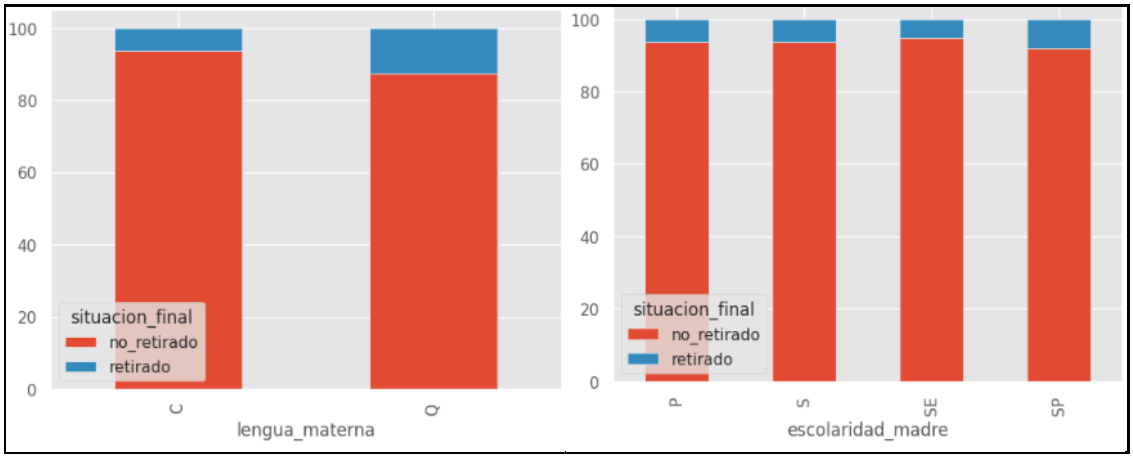
Fuente: Elaboración propia

- Por último, el porcentaje promedio de inasistencias de alumnos retirados es bastante superior a la de no retirados, siendo este de 26%.

Exploración de variables categóricas:







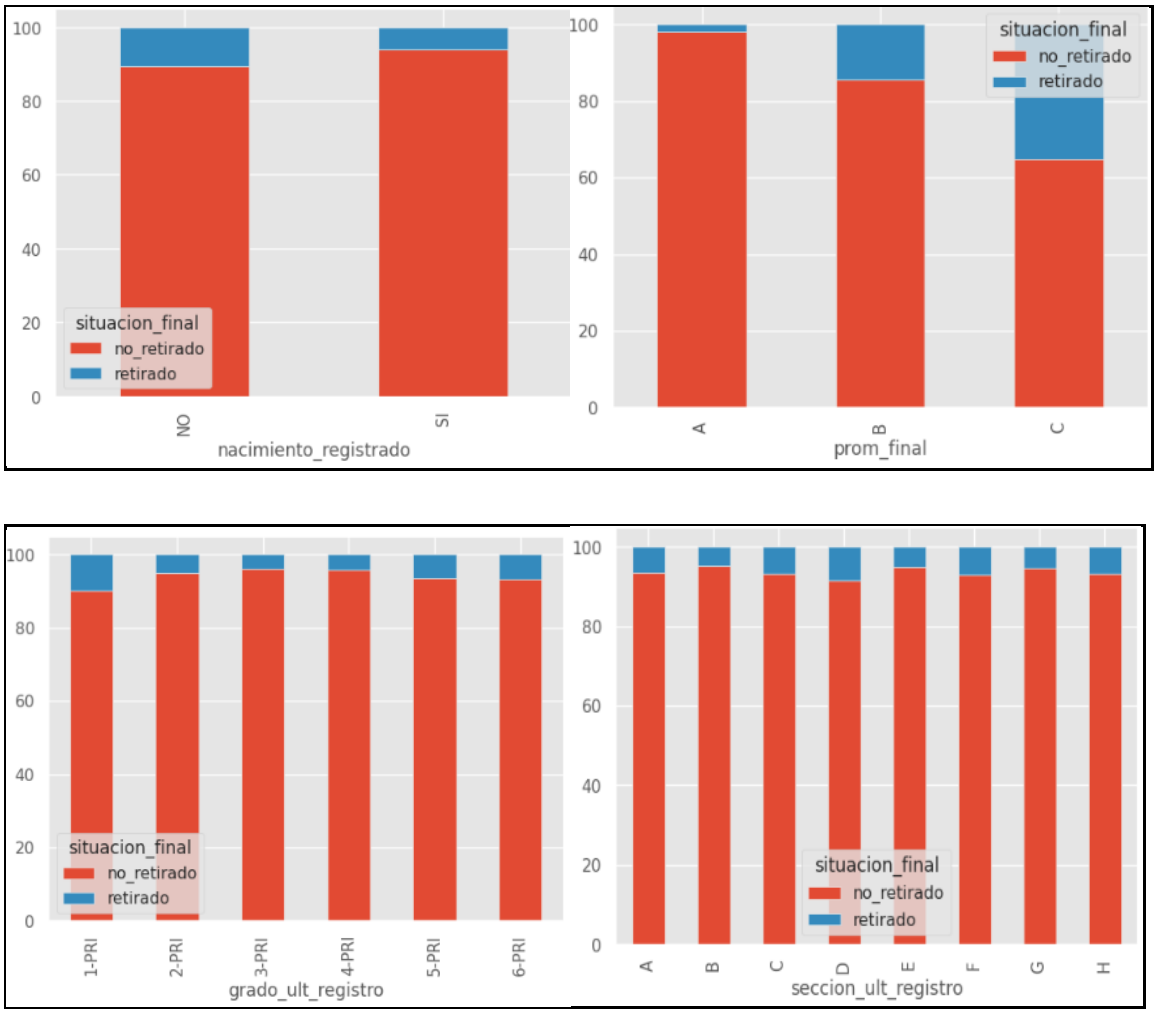


Figura 12: Barplot de variables categóricas según situación final

Fuente: Elaboración propia

Entre los Insights más relevantes se encontró que:

- Existe una mayor proporción de alumnos retirados en el grupo de alumnos que tienen una situación de matrícula inicial "R: repitente" (20%) y "RE: re-ingresante" (28%).
- De la misma forma con aquellos alumnos que tienen como lengua materna Q: Quechua (15%) y un nacimiento "No registrado" (10 %).
- Asimismo, se observa una mayor proporción de alumnos retirados en los registros con un promedio ponderado previo al retiro igual a "C" (28%) o "B" (18%).

- Por otro lado, otras variables donde se observa una diferencia proporcional de retirados notoria según los valores son: ‘nacimiento_registrado’, ‘lengua_materna’, ‘padre_vive’ y ‘pais’.

Tablas de Contingencia en variables relevantes:

Tabla 4: Tabla de contingencia en porcentajes relativos según Situación de matrícula inicial

	situacion_matricula_inicial			
situacion_inicial	I	P	R	RE
no_retirado	90.909001	95.697674	79.411765	66.666667
retirado	9.090900	4.302326	20.588235	33.333333

Fuente: Elaboración propia

La probabilidad de que el alumno se retire cuanto es re-ingresante o repitente es de 33.3% y 20.59% de probabilidad respectivamente.

Tabla 5: Tabla de contingencia en porcentajes relativos según promedio final

	prom_final		
situacion_final	A	B	C
no_retirado	98.140496	85.483871	64.912281
retirado	1.859504	14.516129	35.087719

Fuente: Elaboración propia

La probabilidad de que el alumno se retire es de 35.1% cuando tiene como promedio ponderado "C" y de 14.5% cuando su promedio es "B".

En base a este análisis se procedió a quitar algunas variables consideradas poco relevantes por su comportamiento y naturaleza: 'seccion_ult_registro', 'grado_ult_registro'.

Posteriormente se realizó la codificación numérica de variables categóricas, esto es necesario para el entrenamiento de los modelos, pues algunos no permiten textos como entrada de valores. Para ello se crearon diccionarios por cada variable en donde las ‘llaves’ son los datos categóricos actuales y los ‘valores’ son los datos categóricos que los reemplazarán, tal como se observa en la figura 13:

```
sexo_ajust = {'H':'0', 'M':'1'}
situacion_matricula_inicial_ajust =  {'P':'0', 'I':'1', 'R':'2', 'RE':'3'}
pais_ajust = {'P':'0', 'OT':'1'}
padre_vive_ajust = {'SI':'0','NO':'1'}
madre_vive_ajust = {'SI':'0','NO':'1'}
lengua_materna_ajust = {'C':'0','Q':'1'}
escolaridad_madre_ajust = {'S':'0', 'P':'1', 'SE':'2', 'SP':'3'}
nacimiento_registrado_ajust = {'SI':'0','NO':'1'}
promedio_final_ajust  = {'A':'0', 'B':'1', 'C':'2'}
situacion_final_ajust  = {'no_retirado':'0','retirado':'1'}

data_p2.replace({"sexo": sexo_ajust},inplace=True)
data_p2.replace({"situacion_matricula_inicial": situacion_matricula_inicial_ajust},inplace=True)
data_p2.replace({"pais": pais_ajust},inplace=True)
data_p2.replace({"padre_vive": padre_vive_ajust},inplace=True)
data_p2.replace({"madre_vive": madre_vive_ajust},inplace=True)
data_p2.replace({"lengua_materna": lengua_materna_ajust},inplace=True)
data_p2.replace({"escolaridad_madre": escolaridad_madre_ajust},inplace=True)
data_p2.replace({"nacimiento_registrado": nacimiento_registrado_ajust},inplace=True)
data_p2.replace({"prom_final": promedio_final_ajust},inplace=True)
data_p2.replace({"situacion_final": situacion_final_ajust},inplace=True)
```

Figura 13: Diccionario de valores numéricos según variable categórica y proceso de codificación

Fuente: Elaboración propia

Finalmente se tuvo como datos finales transformados y limpios para el entrenamiento los mostrados en el Anexo 4**.**

Etapa 4: Minería de datos

En esta penúltima etapa, se entrenaron cada uno de los modelos propuestos en este trabajo con la data limpia y preprocesada.

El total de datos se dividió en 2 bloques: 70% destinada a entrenamiento (844 registros) y 30% a validación (362 registros). Se utilizó el hiperparámetro “stratify= y” para mantener la proporción de las clases “retirados” y “no retirados” en cada subconjunto, siendo “y” la variable ‘situación_final’ del dataset. Con esto se tuvo un total de 53 “retirados” y 791 “no retirados” en el subconjunto de entrenamiento, y 23 “retirados” y 339 “no retirados” en el subconjunto de validación (en ambos casos la clase “retirados” representa un 6.7% del total de registros en cada subconjunto aproximadamente). Además, se estableció un “valor de estado aleatorio” o “random state” para tener la misma división de registros en cada corrida de código. Resumiendo lo indicado, tenemos la siguiente figura 14:

```
# SEPARANDO EL DATASET EN VARIABLES DEPENDIENTES E INDEPENDIENTES
y= ds_alumnos_fin[["situacion_final"]].copy()
x= ds_alumnos_fin.drop(columns=['situacion_final']).copy()

# SEPARANDO EN DATA DE ENTRENAMIENTO Y DE VALIDACION
from sklearn.model_selection import train_test_split
x_train_01 , x_validation_01 , y_train_01 , y_validation_01 = train_test_split( x ,
                                                                                y ,
                                                                                stratify     = y ,
                                                                                train_size   = 0.70   ,
                                                                                random_state = 1234
                                                                               )
```

Figura 14: División de data para entrenamiento y validación

Posteriormente, se aplicaron los algoritmos o modelos que utilizan la **"Técnica de Ensamble"**, específicamente aquellos basados en el **"Método Boosting"** que consiste en el ajuste iterativo de "pequeños modelos", en donde por cada iteración se van ajustando los pesos de las malas clasificaciones del modelo anterior, para que el nuevo modelo secuencial pueda aprender y enfocarse en los parámetros adecuados para tener un mejor performance, de esta forma las predicciones individuales de los modelos generan un peso o "poder predictivo", y da como resultado la asignación de más importancia aquellos que tienen mayor poder predictivo al momento de realizarse las votaciones. El flujo de trabajo se realiza de la forma presentada en la figura 15:

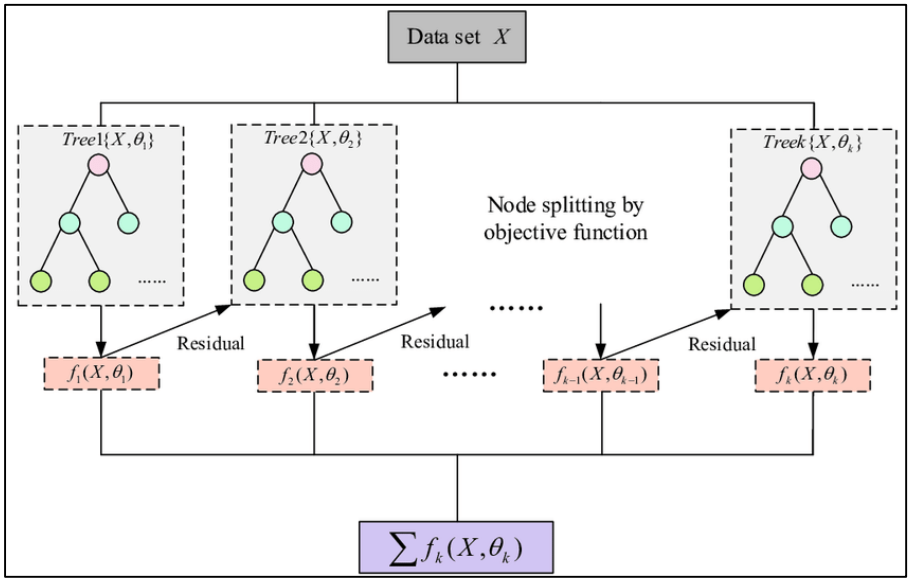


Figura 15: Flujo de trabajo para modelos Ensembler - Método boosting

Fuente: Elaboración propia

Debido a la efectividad y popularidad en la resolución de problemas de clasificación, en el presente trabajo se aplicarán los siguientes modelos:

- XGBOOST: Es un algoritmo predictivo supervisado que se basa en el principio del Boosting y los árboles de decisión. Implementa el algoritmo de "Gradient Descent" en el proceso de optimizar la métrica de interés en la ejecución iterativa de los árboles. Además, tiene incorporada una "regularización" para prevenir el sobreajuste, es de fácil implementación y funciona bien cuando se tienen variables numéricas y categóricas. La siguiente figura 16 resume la configuración del modelo indicado:

```
def objective_xgb_01 (trial):
    print("Started Trial...")
    #hiperparametros para entrenar el modelo
    param_grid = {
        "n_estimators"      : trial.suggest_categorical("n_estimators", [ 50 , 80 , 100 ]),
        "learning_rate"     : trial.suggest_float("learning_rate", 0.05, 0.10),
        "max_depth"         : trial.suggest_int("max_depth", 5, 15),
        "scale_pos_weight"  : trial.suggest_int('scale_pos_weight', 1, 5) ,
        "subsample"         : trial.suggest_categorical("subsample", [0.6, 0.8, 1.0]),
        "colsample_bytree"  : trial.suggest_categorical("colsample_bytree", [0.6, 0.8, 1.0]),
        "gamma"             : trial.suggest_categorical("gamma", [0.5, 0.8, 1.0]),
        'random_state'      : 42          ,
        'tree_method'       : "gpu_hist"
    }
    xgb_optuna_model = xgb.XGBClassifier( objective = "binary:logistic", **param_grid )
    return cross_val_score( xgb_optuna_model        ,
                            x_train_01, y_train_01  ,
                            scoring = 'recall' ,
                            cv      =  cv  ).mean()

xgb_study_01 = optuna.create_study(direction = 'maximize')

[I 2023-11-25 16:02:35,520] A new study created in memory with name: no-name-2c6ff19f-d432-4371-bb76-c3f26b3251ed

xgb_study_01.optimize(objective_xgb_01, n_trials = 100)
```

Figura 16: Configuraciones de entrenamiento del Modelo XGBOOST

Fuente: Elaboración propia

- LIGHTGBM : También está basado en el Boosting y los árboles de decisión, incorpora métodos de muestreo en el cálculo de la ganancia de información, lo que implica que ya no se tenga que evaluar cada variable sino una parte de ellas, esto se traduce en un tiempo de ejecución más breve. Los árboles de decisión generados en el algoritmo tienen un crecimiento vertical, lo cual ayuda a tener mayor precisión, pero es posible que resulte en un sobreajuste de los datos de entrenamiento, esto se puede controlar regulando la profundidad de los árboles y la tasa de entrenamiento. De forma resumida, tenemos el código plasmado en la siguiente figura 17:

```
def objective_lgbm_01 (trial):
    print("Started Trial...")
    param_grid = {
        "n_estimators"     : trial.suggest_categorical("n_estimators", [ 50 , 80 , 100 ]),
        "learning_rate"    : trial.suggest_float("learning_rate", 0.01, 0.1),
        "num_leaves"       : trial.suggest_int("num_leaves", 100, 200, step=10),
        "max_depth"        : trial.suggest_int("max_depth", 5, 15) ,
        "scale_pos_weight" : trial.suggest_int('scale_pos_weight', 1, 5 ) ,
        "random_state"     : 42     ,
        "n_jobs"           : -1,

    }
    lgbm_optuna_model = lgb.LGBMClassifier( objective = 'binary', error_score = 'raise', **param_grid)
    return cross_val_score( lgbm_optuna_model ,
                            x_train_01, y_train_01  ,
                            scoring = 'recall' ,
                            cv      =  cv  ,
                            n_jobs  = -1).mean()
```

```
lgbm_study_01 = optuna.create_study(direction = 'maximize')
```

```
[I 2023-11-25 16:10:25,078] A new study created in memory with name: no-name-fd81b957-3e59-4e1b-8404-b7540047ede0
```

```
lgbm_study_01.optimize(objective_lgbm_01, n_trials = 100 )
```

Figura 17: Configuraciones de entrenamiento del Modelo LIGHTGBM

Fuente: Elaboración propia

- CATBOOST: Este algoritmo incorpora un procesamiento interno de las variables categóricas y también tiene métodos de muestro para no evaluar todas las variables y así reducir tiempos de ejecución. Además, sus árboles de decisión generados internamente tienen un crecimiento simétrico, donde se utiliza el mismo criterio de división en todo el nivel del árbol lo cual sirve como una “regularización” para prevenir el sobreajuste de los datos de entrenamiento. En la figura 18 presentada a continuación tenemos la configuración correspondiente:

```
from catboost import CatBoostClassifier
v_sup = {'verbose': False}
```

```
def objective_cat_01 (trial):
    print("Started Trial...")
    #the hyperparameters to tune
    param_grid = {
        "n_estimators"      : trial.suggest_categorical("n_estimators", [ 80 , 100 , 150 ]),
        "learning_rate"     : trial.suggest_float("learning_rate", 0.05, 0.15),
        "max_depth"         : trial.suggest_int("max_depth", 7, 10),
        "scale_pos_weight"  : trial.suggest_int('scale_pos_weight', 1, 5 )  ,
        "eval_metric"       : "Recall"      ,
        "task_type"         : "GPU",
        'random_state'      : 42
    }

    cat_optuna_model = CatBoostClassifier(  **param_grid )
    return cross_val_score( cat_optuna_model        ,
                            x_train_01, y_train_01  ,
                            scoring    = 'recall' ,
                            fit_params =  v_sup ,
                            cv         =  cv  ).mean()
```

```
cat_study_01 = optuna.create_study(direction = 'maximize')
```

```
[I 2023-11-25 16:14:48,188] A new study created in memory with name: no-name-1fe64ade-ba7b-4ab1-b6bf-df48e20173e0
```

```
cat_study_01.optimize(objective_cat_01, n_trials = 100 )
```

Figura 18: Configuraciones de entrenamiento del Modelo CATBOOST

Fuente: Elaboración propia

Tal y como se aprecia en las figuras 16, 17 y 18, para la configuración de cada modelo se estableció una serie de hiperparámetros para controlar el sobreajuste de los modelos, optimizar tiempos de entrenamiento y mejorar el performance de cada uno de estos, los cuales se detallan a mayor profundidad en el Anexo 6, 7 y 8 respectivamente.

Debido al notorio desbalance de clases en la data (6.7% “retirados” vs 93.3% “no retirados” aproximadamente) se optó por utilizar el hiperparámetro “scale_post_weight” en todos los modelos, el cual tiene el efecto de escalar los errores cometidos por el modelo durante el entrenamiento en la clase positiva (valor igual a ‘1’ correspondiente a la clase “retirado”) y alienta al modelo a corregirlos en exceso. Esto ayuda a lograr un mejor rendimiento al realizar predicciones sobre la clase ‘retirado’, la cual corresponde a los alumnos desertores.

Para elección de los mejores valores de hiperparámetros se utilizó la librería “optuna”, la cual nos permite ejecutar un número definido de veces el entrenamiento de modelos utilizando combinaciones diferentes de hiperparámetros y valores de estos con la finalidad de maximizar la métrica deseada, que en este caso fue el “recall” ya que se busca predecir con la mayor precisión posible a la clase “retirado”. Se le asignó un valor de 100 ejecuciones o intentos a cada modelo.

Etapa 5: Interpretación

En esta etapa se observaron los resultados obtenidos al realizar predicciones con el subconjunto de validación.

XGBOOST: Para un total de 362 registros, se observa una precisión de 74%, una exactitud del 96% y una sensibilidad del 61% al predecir la clase “retirado”, en donde se tuvieron 14 predicciones acertadas y 14 incorrectas (entre Falsos Negativos y Falsos Positivos). La matriz de confusión con los datos de validación se muestra a continuación.

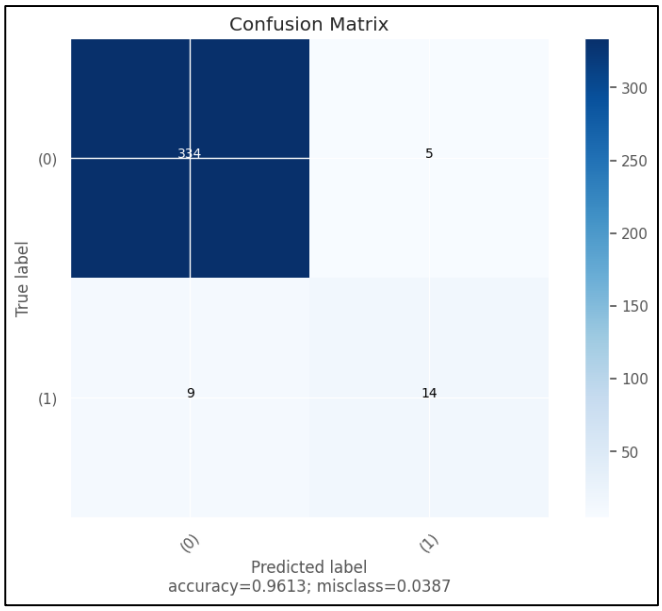


Figura 19: Matriz de confusión de entrenamiento – XGBOOST

Fuente: Elaboración propia

Para este modelo los valores de hiperparámetros elegidos fueron:

‘n_estimators’: 50, learning_rate: 0.095, 'max_depth': 13, 'scale_pos_weight': 5, 'subsample': 0.6, 'colsample_bytree': 0.8, 'gamma': 0.8. Como resultado de la búsqueda con optuna.

Las variables más importantes y sus pesos respectivos se muestran a continuación:

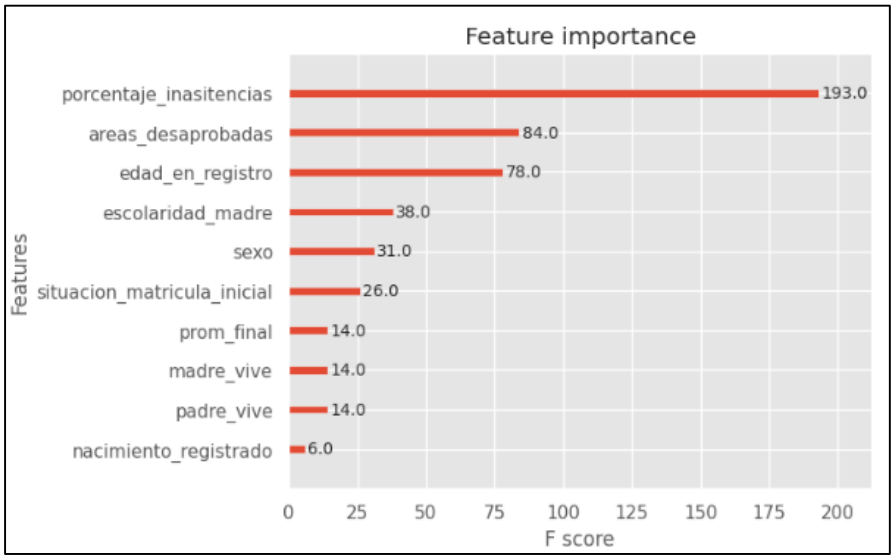


Figura 20: Feature importance – XGBOOST

Fuente: Elaboración propia

LIGHTGBM: Para un total de 362 registros, se observa una precisión de 82%, una exactitud del 97% y una sensibilidad del 61% al predecir la clase “retirado”, en donde se tuvieron 14 predicciones acertadas y 12 incorrectas (entre Falsos Negativos y Falsos Positivos). La matriz de confusión con los datos de validación se muestra a continuación.

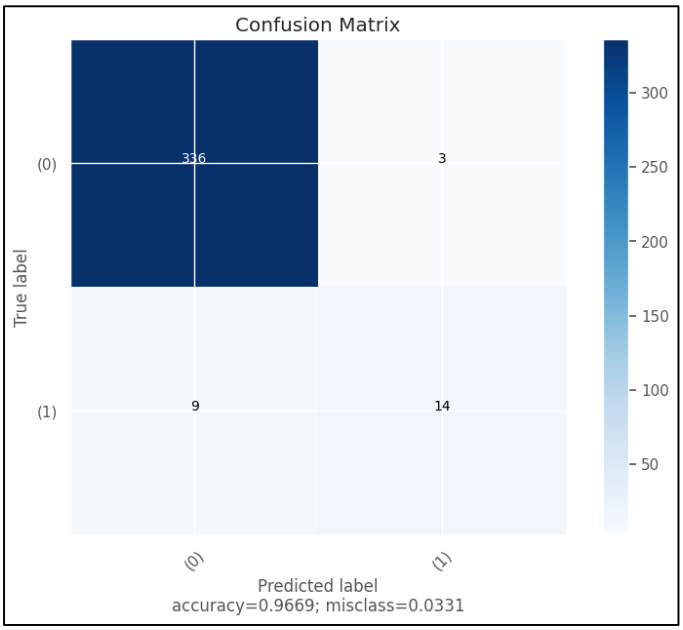


Figura 21: Matriz de confusión de entrenamiento – LIGHTGBM

Fuente: Elaboración propia

Para este modelo los valores de hiperparámetros elegidos fueron:

'n_estimators': 100, 'learning_rate': 0.040, 'num_leaves': 170, 'max_depth': 6, 'scale_pos_weight': 4. Como resultado de la búsqueda con optuna.

Las variables más importantes para el modelo y sus pesos respectivos se muestran en la siguiente figura 22:

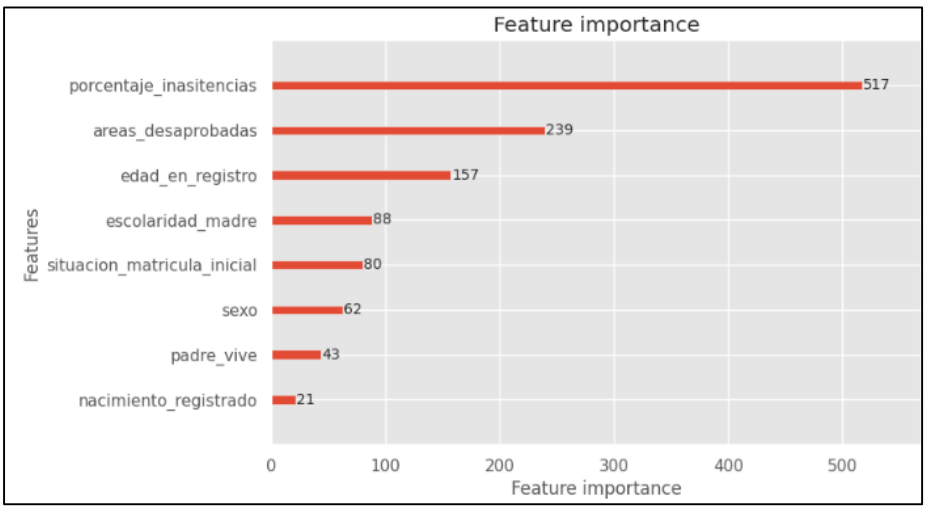


Figura 22: Feature importance – LIGHTGBM

Fuente: Elaboración propia

CATBOOST: Para un total de 362 registros, se observa una precisión de 82%, una exactitud del 97% y una sensibilidad del 61% al predecir la clase "retirado", en donde se tuvieron 14 predicciones acertadas y 12 incorrectas (entre Falsos Negativos y Falsos Positivos). La matriz de confusión con los datos de validación se muestra a continuación en la figura 23:

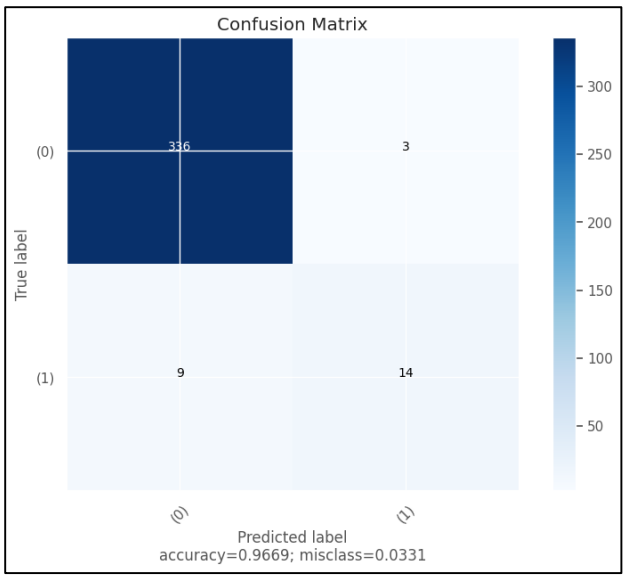


Figura 23: Matriz de confusión de entrenamiento – CATBOOST

Fuente: Elaboración propia

Para este modelo los valores de hiperparámetros elegidos fueron:

'n_estimators': 150, 'learning_rate': 0.057, 'max_depth': 7, 'scale_pos_weight': 4. Como resultado de la búsqueda con optuna.

Las variables más importantes para el modelo y sus pesos respectivos se muestran en la figura 24:

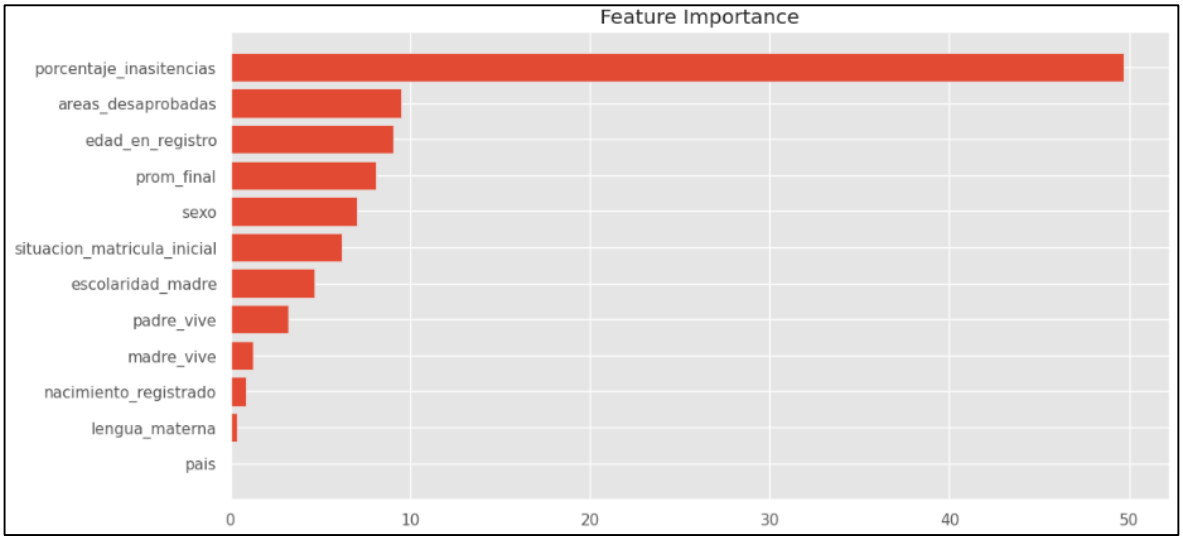


Figura 24: Feature importance – CATBOOST

Fuente: Elaboración propia

Tal como se observa lo modelos XGBOOST Y LIGHTGBM tienen como variables más relevantes: 'porcentaje_inasistencias', 'areas_desaprobadas', 'edad_en_registro' , 'escolaridad_madre', 'sexo' y 'situacion_matricula_inicial'. Mientras que para el modelo CATBOOST la variable 'prom_final' es una de las que tiene más peso.

En resumen, como resultados de la etapa de entrenamiento de modelos se tienen las siguientes métricas en la tabla 6:

Tabla 6: Resumen de indicadores del resultado de entrenamiento

	INDICADORES		
MODELO	PRECISIÓN	EXACTITUD	SENSIBILIDAD
XGBOOST	74%	96%	61%
LIGHTGBM	82%	97%	61%
CATBOOST	82%	97%	61%

Fuente: Elaboración propia

Se observa que los modelos tuvieron un buen performance, siendo CATBOOST y LIGTHGBM los mejores.

3.6. Métodos de análisis de datos

Para el presente trabajo de investigación la data se obtuvo a través de la plataforma SIAGIE, luego se realizó la eliminación de los datos inconsistentes. Se entrenaron los modelos XGBOOST, LIGHTGBM y CATBOOST para determinar el modelo óptimo de acuerdo con la necesidad del trabajo.

A continuación, mediante el uso de las bibliotecas de Python se obtuvieron las métricas requeridas: precisión, exactitud y sensibilidad. En cada proceso del modelo, se hicieron uso de gráficos representativos con el fin de resumir el cálculo durante cada fase.

Finalmente se evaluaron los 3 modelos con data de testeo, es decir, registros de alumnos que no se usaron durante la etapa de entrenamiento y validación, esto con la finalidad de corroborar la efectividad de los algoritmos con casuísticas nuevas y reales. Las métricas finales obtenidas se encuentran en la sección de Resultados.

3.7. Aspectos éticos

La investigación se llevará a cabo de acuerdo con la Resolución Rectoral N° 062-2023-VI-UCV. De igual manera, aplicando la séptima edición de la norma APA. En este sentido, el material suministrado tendrá la debida citación y la bibliografía estará preparada para cumplir con los estándares de esta norma tanto en estructura como en estilo. Igualmente, importantes son los ideales de imparcialidad, secreto, creatividad y anonimato, todos los cuales deben ser defendidos. Rodríguez (2021) enfatizó la necesidad de actuar sin prejuicios, lo que implica considerar solo criterios objetivos relacionados con el elemento que se analiza y no con otros factores, como las personas involucradas o los sentimientos personales del actor. De esta manera, los hallazgos se aceptarán tal cual, sin que se realicen cambios. Asimismo, se tendrá en cuenta la necesidad de mantener la privacidad. La confidencialidad, tal y como la definen Prats et al. (2017), se refiere a un entendimiento entre el investigador y el participante sobre cómo manejar, almacenar y compartir la información personal del participante. Debido a esto, no se presentaron datos que

hubieran permitido una comparación amplia de los antecedentes y características de los sujetos del estudio. Además, tendremos en cuenta el principio de originalidad, que establece que no podemos incorporar ningún concepto o dato de otra investigación a menos que se haga referencia a esos otros estudios. Esto significa que es aceptable tomar prestados pasajes de otros autores siempre que se dé el debido crédito. Por lo tanto, es crucial enfatizar que se trata de una obra de arte completamente nueva e inédita. Finalmente, se tendrá en cuenta el concepto de anonimato, lo que significa que se protegerán los datos a medida que avance la investigación y se mantendrá en secreto el conocimiento institucional del éxito de la evaluación.

IV. RESULTADOS

Luego de aplicar la metodología KDD y evaluar nuestros algoritmos de aprendizaje en base a los indicadores: precisión, exactitud y sensibilidad para obtener el que presenta mejores resultados, se realizó la validación de nuestra hipótesis, indicando si el estudiante desertó o siguió con sus estudios. Prosiguiendo con lo mencionado, se obtuvieron las siguientes matrices de confusión con respecto a cada modelo seleccionado:

- **XGBOOST:**

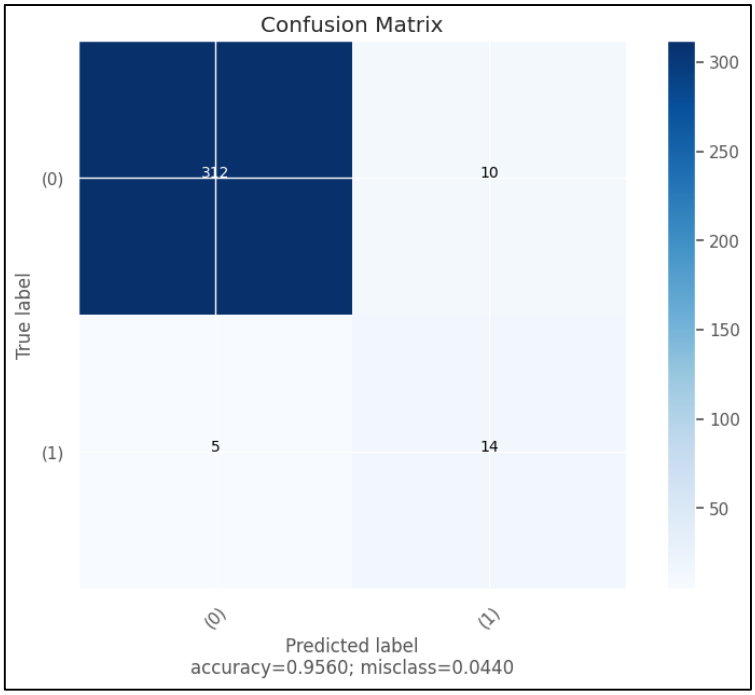


Figura 25: Matriz de confusión – XGBOOST

Fuente: Elaboración propia

Para un total de 341 registros, el modelo logró identificar correctamente 312 casos que no desertaron y 14 estudiantes que sí lo hicieron. Por otro lado, 15 registros se catalogan como predicciones incorrectas.

- **LIGHTGBM:**

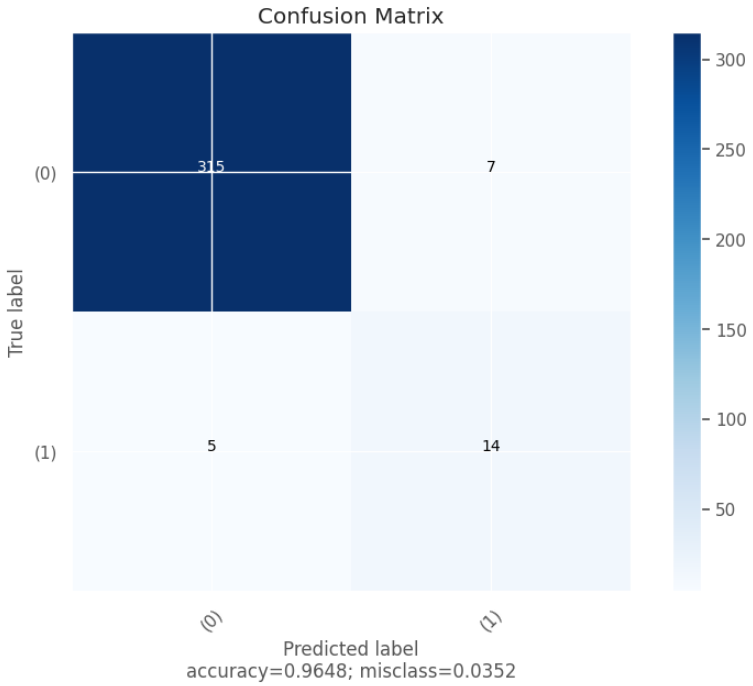


Figura 26: Matriz de confusión – LIGHTGBM

Fuente: Elaboración propia

En este modelo de aprendizaje automático, el modelo logró identificar correctamente 315 casos que no desertaron y 14 estudiantes desertores. Por otro lado, 12 registros se catalogan como predicciones incorrectas.

- **CATBOOST:**

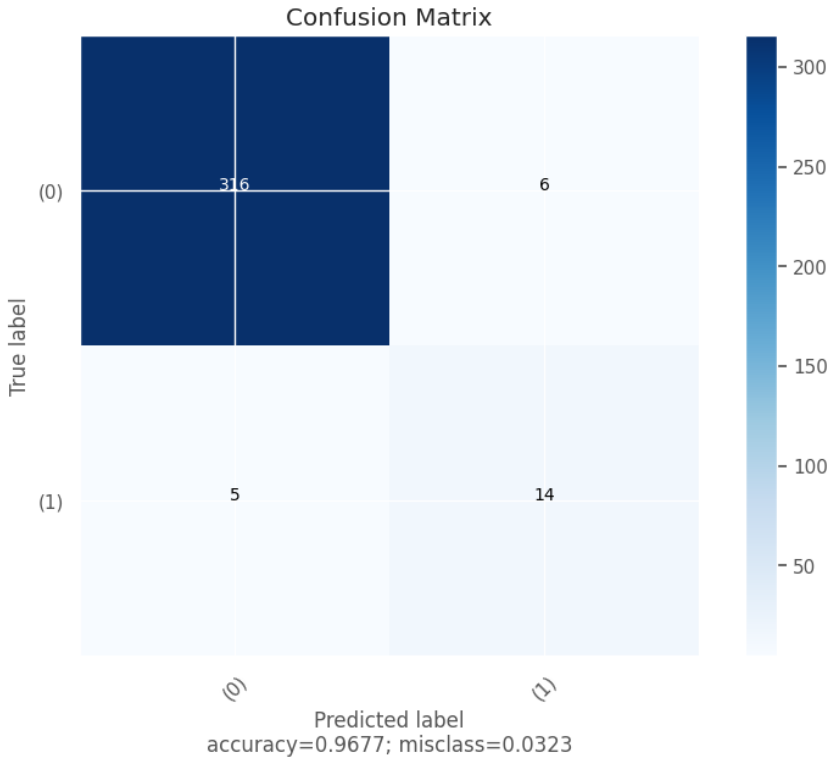


Figura 27: Matriz de confusión – CATBOOST

Fuente: Elaboración propia

Al igual que en los casos anteriores, en este caso también se utilizó la misma muestra. Se observó que 5 predicciones se realizaron incorrectamente para el caso de que los estudiantes siguieron con sus estudios y 6 erróneas para la clase de desertores.

Luego de lo mencionado, procedemos a mostrar los resultados obtenidos con respecto a las hipótesis específicas planteadas en el trabajo de investigación. Vale indicar que estos datos se obtuvieron con ayuda de la herramienta Collab.

HE_1: El modelo predictivo de machine learning utilizando el método Boosting de ensemble es preciso en la predicción en la deserción estudiantil en EBR.

- **XGBOOST:**

Tabla 7: Precisión de la predicción realizada por XGBOOST

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	312	10
	5	14
	FN	TP

Fuente: Elaboración propia

$Precisión = \frac{TP}{(TP + FP)}$	$Precisión = \frac{14}{(14+10)} = 0.58$

Interpretación: El modelo predictivo XGBOOST, utilizando el método Boosting de ensemble, es preciso en un 58% con respecto a la predicción en la deserción estudiantil en EBR.

- **LIGHTGBM:**

Tabla 8: Precisión de la predicción realizada por LIGHTGBM

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	315	7
	5	14
	FN	TP

Fuente: Elaboración propia

$Precisión = \frac{TP}{(TP + FP)}$	$Precisión = \frac{14}{(14+7)} = 0.66$

Interpretación: El modelo predictivo *LIGHTGBM*, utilizando el método Boosting de ensemble, es preciso en un 66% con respecto a la predicción en la deserción estudiantil en EBR.

- **CATBOOST:**

Tabla 9: Precisión de la predicción realizada por CATBOOST

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	316	6
	5	14
	FN	TP

Fuente: Elaboración propia

$Precisión = \frac{TP}{(TP + FP)}$	$Precisión = \frac{14}{(14 + 6)} = 0.70$

Interpretación: El modelo predictivo CATBOOST, utilizando el método Boosting de ensemble, es preciso en un 70% con respecto a la predicción en la deserción estudiantil en EBR.

HE_2: El modelo predictivo de machine learning utilizando el método Boosting de ensemble es exacto en la predicción en la deserción estudiantil en EBR.

- **XGBOOST:**

Tabla 10: Exactitud de la predicción realizada por XGBOOST

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	312	10
	5	14
	FN	TP

Fuente: Elaboración propia

$Exactitud = \frac{(TP + TN)}{Total}$	$Exactitud = \frac{(14 + 312)}{341} = 0.96$

Interpretación: El modelo predictivo XGBOOST, utilizando el método Boosting de ensemble, resulta exacto en un 96% con respecto a la predicción en la deserción estudiantil en EBR.

- **LIGHTGBM:**

Tabla 11: Exactitud de la predicción realizada por LIGHTGBM

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	315	7
	5	14
	FN	TP

Fuente: Elaboración propia

$Exactitud = \frac{(TP + TN)}{Total}$	$Exactitud = \frac{(14 + 315)}{341} = 0.96$

Interpretación: El modelo predictivo LIGHTGBM, utilizando el método Boosting de ensemble, resulta exacto en un 96% con respecto a la predicción en la deserción estudiantil en EBR.

- **CATBOOST:**

Tabla 12: Exactitud de la predicción realizara por CATBOOST

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	316	6
	5	14
	FN	TP

Fuente: Elaboración propia

$Exactitud = \frac{(TP + TN)}{Total}$	$Exactitud = \frac{(14 + 316)}{341} = 0.97$

Interpretación: El modelo predictivo CATBOOST, utilizando el método Boosting de ensemble, resulta exacto en un 97% con respecto a la predicción en la deserción estudiantil en EBR.

HE$_3$: Los niveles de sensibilidad del modelo predictivo de machine learning mejoran utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR.

- **XGBOOST:**

Tabla 13: Sensibilidad de la predicción realizada por XGBOOST

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	312	10
	5	14
	FN	TP

Fuente: Elaboración propia

$Sensibilidad = \frac{TP}{(FN + TP)}$	$Sensibilidad = \frac{14}{(5 + 14)} = 0.74$

Interpretación: El modelo predictivo XGBOOST, utilizando el método Boosting de ensemble, arroja un porcentaje de sensibilidad del 74% con respecto a la predicción en la deserción estudiantil en EBR.

- **LIGHTGBM:**

Tabla 14: Sensibilidad de la predicción realizada por LIGHTGBM

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	315	7
	5	14
	FN	TP

Fuente: Elaboración propia

$Sensibilidad = \frac{TP}{(FN + TP)}$	$Sensibilidad = \frac{14}{(5 + 14)} = 0.74$

Interpretación: El modelo predictivo LIGHTGBM, utilizando el método Boosting de ensemble, arroja un porcentaje de sensibilidad del 74% con respecto a la predicción en la deserción estudiantil en EBR.

- **CATBOOST:**

Tabla 15: Sensibilidad de la predicción realizada por CATBOOST

	PREDICCIÓN	
CLASE VERDADERA	TN	FP
	316	6
	5	14

	FN	TP

Fuente: Elaboración propia

$Sensibilidad = \frac{TP}{(FN + TP)}$	$Sensibilidad = \frac{14}{(5 + 14)} = 0.74$

Interpretación: El modelo predictivo CATBOOST, utilizando el método Boosting de ensemble, arroja un porcentaje de sensibilidad del 74% con respecto a la predicción en la deserción estudiantil en EBR.

Hipótesis General:

H_0: El modelo predictivo de machine learning utilizando el método Boosting de ensemble mejora la predicción de la deserción estudiantil en EBR.

H_1: El modelo predictivo de machine learning utilizando el método Boosting de ensemble no mejora la predicción de la deserción estudiantil en EBR.

Tabla 16: Comparación entre resultados por indicadores

	INDICADORES		
MODELO	PRECISIÓN	EXACTITUD	SENSIBILIDAD
XGBOOST	58%	96%	74%
LIGHTGBM	67%	96%	74%
CATBOOST	70%	97%	74%

Fuente: Elaboración propia

De acuerdo con los resultados obtenidos y plasmados en la tabla 16, el modelo elegido es el modelo CATBOOST, ya que para la resolución de la problemática principal se desea predecir con la mayor exactitud y precisión posible cuando el alumno presenta el riesgo de retirarse del periodo académico. En este caso el modelo tuvo una precisión del 70%, una exactitud del 97% y una sensibilidad del 74%. Cabe resaltar que es el modelo que más sobresale en los 3 indicadores mencionados para la clase de deserción estudiantil.

V. DISCUSIÓN

En esta sección del trabajo de investigación se muestran las discusiones presentadas a raíz de los resultados obtenidos durante la investigación.

En este estudio, se empleó la metodología KDD para macar las pautas de cómo llevar a cabo la presente tesis. Esta metodología nos resume en una sería de pasos la implementación recomendada de forma ágil en un tiempo reducido. Autores como Pérez y Rojas (2020), Luque y Sarazu (2019) y García (2020) también utilizaron como referencia la mencionada metodología KDD para obtener el conocimiento requerido. Por otro lado, autores como Gutiérrez (2022) y Shica (2022), utilizaron CRISP-DM ya que buscaron la estandarización del ciclo de vida del proyecto de análisis de datos. Niyogisubizo et al. (2022) empleó la metodología de generalización de apilamiento debido a la naturaleza experimental de su investigación.

Para desarrollar el modelo predictivo se recurrió al entorno colaborativo de Google Colab. Dicha herramienta posee una interfaz completada, interactiva y de fácil uso. El desarrollo se ejecutó con efectividad y rapidez; mientras que autores como García (2020) emplearon la herramienta SPSS Modeler debido al apoyo en la comunidad y en la documentación de esta misma.

Luego de realizar una comparación entre los métodos boosting: XGBoost, LightGBM y CatBoost. Se contrasto que el mejor modelo se creó utilizando CatBoost ya que presento los mejores valores en los indicadores que se plantearon en esta investigación: una precisión del 70%, una exactitud del 97% y una sensibilidad del 74%. Gutiérrez (2022) utiliza el modelo Gradient Boosting para medir la precisión, la sensibilidad, el puntaje F1 y la exactitud. Sus resultados después de la etapa de entrenamiento fueron: 94%, 86%, 90% y 95% respectivamente. Otros autores por el contrario como Rodríguez et al. (2023), emplearon otras métricas como el recall y el puntaje f1; obteniendo como resultado 47% y 53% respectivamente. Por el lado de Panagiotakopoulos et al. (2021), obtuvo como modelo destacado al LightGBM y solo empleo la precisión y la puntación como métricas de aceptación; sus resultados oscilaron entre el 91% y el 95,58 % y entre el 93,16 % y el 96,34 %, respectivamente. Por otro lado, García (2020) y Shica (2022) solo usaron la precisión como valor de medida principal.

La recolección de datos en esta investigación se realizó mediante la extracción de información a través de la plataforma SIAGIE, Sistema de Información de Apoyo a la Gestión de la Institución Educativa. Se empleó el análisis de registros de la data de estudiantes entre los años 2014 y 2022. Dicha inspección, de tipo cuantitativo, se realizó en base a la matriz de datos que se exportaron a través de la mencionada plataforma de gestión de información nacional. De la información recopilada, se visualizaron 21 datos clasificados en: de tipo personal (10), de tipo parental (3 datos) y de tipo académico (8 datos). De igual forma, Niyogisubizo et al. (2022), utilizó un conjunto de datos recopilados directamente desde los registros de la Universidad Constantine the Philosopher en Nitra de 2016 a 2020. Así mismo, el resto de autores como Rodríguez et al. (2023), Khoushehgir y Sulaimany (2023), Panagiotakopoulos et al. (2021), Pérez y Rojas (2020), Luque y Sarazu (2019), García (2020), Shica (2022) y Gutiérrez (2022) decidieron obtener los registros históricos de los estudiantes en sus determinados años; esto permitió obtener una data homologada y con mejores resultados durante el entramiento en cada uno de los modelos.

Los indicadores de la presente investigación arrojaron resultados muy favorables.

En este trabajo. Se implementaron una serie de modelos de predicción basados métodos Boosting. En particular, se usaron 3: XGBoost, LightGBM y CatBoost. Nuestros peores resultados se obtuvieron con el método XGBoost, donde la precisión arrojó un valor de 58%, se consiguió una exactitud del 96% y una sensibilidad del 74%. El método LightGBM, arrojó mejores resultados con respecto al indicador de la precisión, su valor es de 67%, en el resto de indicadores, los porcentajes se mantienen. Por último, el desarrollo de un modelo basado en el método CatBoost, como método principal para el desarrollo de un modelo de deserción estudiantil, fue el que mejores resultados obtuvo en cada uno de los indicadores seleccionados en el presente trabajo: una precisión del 70%, una exactitud del 97% y una sensibilidad del 74%. Sin embargo, para el caso de otros autores como Panagiotakopoulos et al. (2021), obtuvo como modelo destacado al LightGBM, en donde su indicador de precisión fue superior al 90%. Así mismo, para el caso de Gutiérrez (2022), el resultado de su precisión fue de 94%, su exactitud se marcó en un 95% y su sensibilidad en un 86%.

VI. CONCLUSIÓN

Del presente trabajo de investigación realizado, se ha llegado a las siguientes conclusiones:

Se pudo concluir luego de implementar varios modelos boosting como el XGBoost, LightGBM y CatBoost que el método más apropiado para la predicción de deserción estudiantil para alumnos de primaria y secundaria, es el método CatBoost con una precisión del 70%, una exactitud del 97% y una sensibilidad del 74%.

El segundo mejor método, LightGBM, es inferior en 3 puntos respecto del indicador de precisión y respecto a los otros 2 indicadores, obtuvo los mismos resultados.

El tercer método y menos relevante, XGBoost, es inferior por 9 puntos respecto al indicador de precisión comparado con el modelo LightGBM. Y está 12 puntos por debajo respecto al mismo indicador comparado con el modelo CatBoost; en el resto de los otros 2 indicadores, obtuvo los mismos resultados que los modelos previos.

Las variables más importantes para la predicción de la deserción estudiantil, de acuerdo al ranking de pesos asignado por los modelos son el porcentaje de inasistencias, el número de áreas desaprobadas y la edad del estudiante durante el registro.

Por lo tanto, se puede concluir que el modelo CatBoost que utiliza la técnica de ensemble y método boosting permite predecir con alta precisión, exactitud y sensibilidad la deserción estudiantil para alumnos de primaria y secundaria.

VII. RECOMENDACIONES

Las recomendaciones para futuras investigaciones similares con las siguientes:

- Se recomienda recolectar datos de escolares de distintas instituciones educativas estatales con el fin de crear un modelo replicable para todas estas entidades.
- Se recomienda desarrollar un software de uso interno en las instituciones educativas para el seguimiento de cada estudiante, tal como una web app que se utilice en periodos de matrícula y después de cada evaluación periódica que se realice a fin de tomar acciones preventivas ante posibles deserciones estudiantiles.
- Se recomienda crear un modelo predictivo considerando más variables relacionada a la deserción estudiantil: nivel socioeconómico, cercanía al lugar de la escuela, factores psicológicos, tipo y condición de vivienda, condición de salud de los familiares directos, etc.
- Se recomienda realizar estudios sobre el método “stacking” para utilizar la combinación de resultados de estos algoritmos con la finalidad de evaluar mejoras posibles en la predicción de resultados.

REFERENCIAS

Apaza-Tarqui, A., Borda-Navedos, W., Cayo, N. y Huanca-Suaquita, J. (2022). Técnicas de minería de datos para determinar la deserción escolar. *Técnicas de minería de datos para determinar la deserción escolar*, *2*(1), 11–14. https://doi.org/10.35622/INUDI.B.053

Aranciaga, J. y Ccanto, E. (2021). Factores asociados a la deserción de estudiantes en un instituto de educación superior privado de Lima [Universidad Femenina del Sagrado Corazón]. En *Repositorio Institucional - UNIFÉ*. https://repositorio.unife.edu.pe/repositorio/handle/20.500.11955/870

Arroyo-Hernández, J. (2020). Métodos de reducción de dimensionalidad: Análisis comparativo de los métodos APC, ACPP y ACPK. *Uniciencia*, *34*(1), 12–21. https://doi.org/10.15359/RU.30-1.7

Asish, S., Kulshreshth, A. y Borst, C. (2022). Detecting distracted students in educational VR environments using machine learning on eye gaze data. *Computers & Graphics*, *1*(9), 75–87. https://doi.org/10.1016/J.CAG.2022.10.007

Baena, G. (2017). *Metodología de la Investigación* (2a ed., Vol. 1). GRUPO EDITORIAL PATRIA, S.A. DE C.V. https://editorialpatria.com.mx/pdffiles/9786074384093.pdf

Barrios-Hernández, K., García-Villaverde, P. y Ruiz-Ortega, M. (2021). Capital social y los resultados de los grupos de investigación, desarrollo tecnológico e innovación del departamento del Atlántico, Colombia. *Información tecnológica*, *32*(1), 57–68. https://doi.org/10.4067/S0718-07642021000100057

Beckham, N. R., Akeh, L. J., Mitaart, G. N. P. y Moniaga, J. V. (2023). Determining factors that affect student performance using various machine learning methods. *Procedia Computer Science*, *216*, 597–603. https://doi.org/10.1016/J.PROCS.2022.12.174

Bemthuis, R., Wang, W., Iacob, M. y Havinga, P. (2023). Business rule extraction using decision tree machine learning techniques: A case study into smart returnable transport items. *Procedia Computer Science*, *2*(1), 446–455. https://doi.org/10.1016/J.PROCS.2023.03.057

Bernal, C. (2016). *Metodología de la investigación* (3a ed.). Pearson. https://bit.ly/3udSjK8

Bognár, L. y Fauszt, T. (2022). Factors and conditions that affect the goodness of machine learning models for predicting the success of learning. *Computers and Education: Artificial Intelligence*, *3*(1), 101–121. https://doi.org/10.1016/J.CAEAI.2022.100100

Carranza, A. y Jakobsen, J. (2022). Neural network programming: Integrating first principles into machine learning models. *Computers & Chemical Engineering*, *1*(8), 107–111. https://doi.org/10.1016/J.COMPCHEMENG.2022.107858

Castanelli, D. (2023). Sociocultural learning theory and assessment for learning. *Medical Education*, *2*(1), 12–21. https://doi.org/10.1111/MEDU.15028

Castañeda, M. (2022). La cientificidad de metodologías cuantitativa, cualitativa y emergentes. *Revista Digital de Investigación en Docencia Universitaria*, *16*(1), 12–23. https://doi.org/10.19083/RIDU.2022.1555

CEPAL. (2020). *Panorama Social de América Latina* . https://www.cepal.org/sites/default/files/publication/files/46687/S2000966_es.pdf

Chinguel, S. (2022). Evaluación de rendimiento de algoritmos en la identificación de ataques a sitios web utilizando logs de servidor [Universidad Señor de Sipán]. En Universidad Señor de Sipán. https://repositorio.uss.edu.pe/bitstream/handle/20.500.12802/9214/Chinguel%20Tineo%20Segundo%20Florentino.pdf?sequence=1&isAllowed=y

Chung, J. y Lee, S. (2019). Dropout early warning systems for high school students using machine learning. *Children and Youth Services Review*, *9*(6), 346–353. https://doi.org/10.1016/J.CHILDYOUTH.2018.11.030

Contreras-Bravo, L., Tarazona-Bermúdez, G. y Rodríguez-Molano, J. (2021). Tecnología y analítica del aprendizaje: una revisión a la literatura. *Revista científica*, *41*(41), 150–168. https://doi.org/10.14483/23448350.17547

Coussement, K., Phan, M., De Caigny, A., Benoit, D. y Raes, A. (2020). Predicting student dropout in subscription-based online learning environments: The beneficial

impact of the logit leaf model. *Decision Support Systems*, *1*(3), 111–121. https://doi.org/10.1016/J.DSS.2020.113325

DRELM. (2020). *La deserción escolar en Lima Metropolitana se redujo a la mitad en el 2020* . https://www.drelm.gob.pe/drelm/noticias/la-desercion-escolar-en-lima-metropolitana-se-redujo-a-la-mitad-en-el-2020/

García, E. (2020). Detección de patrones de deserción estudiantil mediante aplicación de Árboles de Decisión C4.5 en el IESTP “Señor de Chocán” de Querecotillo [Universidad Cesar Vallejo]. En *Repositorio Institucional - UCV*. https://repositorio.ucv.edu.pe/handle/20.500.12692/55029

Gil, V. y Seguro, C. (2022). Machine learning aplicado al análisis del rendimiento de desarrollos de software. *Revista Politécnica*, *18*(35), 128–139. https://doi.org/10.33571/RPOLITEC.V18N35A9

Gutierrez, H. (2022). Modelo predictivo para la deserción de estudiantes en el primer año de estudio en la Universidad Nacional Santiago Antúnez de Mayolo, Huaraz – 2022 [Universidad Nacional Santiago Antunez de Mayolo]. En *Universidad Nacional Santiago Antúnez de Mayolo*. http://repositorio.unasam.edu.pe/handle/UNASAM/5361

Guzmán-Castillo, S., Körner, F., Pantoja-García, J. I., Nieto-Ramos, L., Gómez-Charris, Y., Castro-Sarmiento, A. y Romero-Conrado, A. R. (2022). Implementation of a Predictive Information System for University Dropout Prevention. *Procedia Computer Science*, *198*, 566–571. https://doi.org/10.1016/J.PROCS.2021.12.287

Henríquez, N. y Vargas, D. (2022). Modelos predictivos de rendimiento y deserción académica en estudiantes de primer año de una universidad pública chilena. *Revista de estudios y experiencias en educación*, *21*(45), 299–316. https://doi.org/10.21703/0718-5162.V21.N45.2022.015

Hernández-Sampieri, R. y Mendoza, C. (2018). *Metodología de la investigación: las rutas: cuantitativa ,cualitativa y mixta* (7a ed.). MCgraw - Hill Interamericana de México, S.A. de C.V. https://virtual.cuautitlan.unam.mx/rudics/?p=2612

Hoyos, J. y Aponte-Novoa, F. (2019). Caracterización de los estudiantes de una institución de educación superior mediante big data. *Ingeniería y Desarrollo*, *37*(2), 159–172. https://doi.org/10.14482/INDE.37.2.1378

Jara, J. (2021). Modelo emprendedor basado en información tributaria. Universidad del Desarrollo. https://repositorio.udd.cl/server/api/core/bitstreams/05970820-51f6-497c-a399-c6e9b0e195fd/content

Jimenez, J. y Cota-Yañez, R. (2019). Relación del grado de escolaridad y el ingreso bajo la perspectiva de la teoría del capital humano. Estudio de caso. *Revista de Comunicación de la SEECI*, *2*(48), 87–108. https://doi.org/10.15198/SEECI.2019.48.87-108

Karunachandra, B., Putera, N., Wijaya, S. R., Suryani, D., Wesley, J. y Purnama, Y. (2023). On the benefits of machine learning classification in cashback fraud detection. *Procedia Computer Science*, *2*(1), 364–369. https://doi.org/10.1016/J.PROCS.2022.12.147

Kaur, P., Kumar, H. y Kaushal, S. (2021). Affective state and learning environment based analysis of students' performance in online assessment. *International Journal of Cognitive Computing in Engineering*, *2*(1), 12–20. https://doi.org/10.1016/J.IJCCE.2020.12.003

Kersting, K. (2018). Machine Learning and Artificial Intelligence: Two Fellow Travelers on the Quest for Intelligent Behavior in Machines. *Frontiers in Big Data*, *0*, 6. https://doi.org/10.3389/FDATA.2018.00006

Khoushehgir, F. y Sulaimany, S. (2023). Negative link prediction to reduce dropout in Massive Open Online Courses. *Education and Information Technologies*. https://doi.org/10.1007/S10639-023-11597-9

Korniichuk, R. y Boryczka, M. (2021). Averaging and Boosting Methods in Ensemble-Based Classifiers for Text Readability. *Procedia Computer Science*, *1*(9), 121–140. https://doi.org/10.1016/J.PROCS.2021.09.141

Luna, H. (2021). Caracterización de arbitraje financiero en el mercado de divisas. Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California.

https://cicese.repositorioinstitucional.mx/jspui/bitstream/1007/3476/1/tesis%20H%C3%A9ctor%20Luna%20Armendariz%2009%20feb%202021.pdf

Luque, R. y Sarazu, C. (2019). Modelo predictivo para determinar deserción de estudiantes en la Universidad Tecnológica del Perú [Universidad Tecnológica del Perú]. En *Universidad Tecnológica del Perú*. http://repositorio.utp.edu.pe/handle/20.500.12867/2924

Maharjan, J., Garikipati, A., Dinenno, F., Ciobanu, M., Barnes, G., Browning, E., DeCurzio, J., Mao, Q. y Das, R. (2023). Machine learning determination of applied behavioral analysis treatment plan type. *Brain Informatics*, *10*(1), 12–21. https://doi.org/10.1186/S40708-023-00186-8

Mantilla, R. y Negre, F. (2021). Pensamiento computacional, una estrategia educativa en épocas de pandemia. *Innoeduca. International Journal of Technology and Educational Innovation*, *7*(1), 89–106. https://doi.org/10.24310/INNOEDUCA.2021.V7I1.10593

Mariano, A., De Magalhães Lelis, Santos, M., Castilho, M. y Bastos, A. (2022). Decision trees for predicting dropout in Engineering Course students in Brazil. *Procedia Computer Science*, *1*(2), 1113–1120. https://doi.org/10.1016/J.PROCS.2022.11.285

Mary, T. A. C. y Rose, P. J. A. L. (2023). Multifaceted Sentiment Detection System (MSDS) to Avoid Dropout in Virtual Learning Environment using Multi-class Classifiers. *International Journal of Advanced Computer Science and Applications*, *14*(4), 357–368. https://doi.org/10.14569/IJACSA.2023.0140440

Marchant, V. (2022). Un modelo predictivo interpretable para la estimación del ingreso monetario de clientes bancarios basado en XGBoost y SHAP. http://repositorio.udec.cl/jspui/bitstream/11594/10173/1/Tesis_Vicente_Marchant.pdf

MINEDU. (2020). *¿Qué significa la competencia "Construye interpretaciones históricas"? – Currículo Nacional*. https://sites.minedu.gob.pe/curriculonacional/2020/11/09/que-significa-la-competencia-construye-interpretaciones-historicas/

MOINE, J.M., 2013. Metodologías para el descubrimiento de conocimiento en bases de datos: un estudio comparativo. XVII Congreso Argentino De Ciencias De La Computación, vol. XVII CACIC, pp. 111.

Mora, J. (2022). Modelos predictivos en salud basados en aprendizaje de maquina (machine learning). *Revista Médica Clínica Las Condes*, *33*(6), 583–590. https://doi.org/10.1016/J.RMCLC.2022.11.002

Niyogisubizo, J., Liao, L., Nziyumva, E., Murwanashyaka, E. y Nshimyumukiza, P. (2022). Predicting student's dropout in university classes using two-layer ensemble machine learning approach: A novel stacked generalization. *Computers and Education: Artificial Intelligence*, *3*(1), 101–111. https://doi.org/10.1016/J.CAEAI.2022.100066

Ñaupas, H., Valdivia, M., Palacios, J. y Romero, H. (2018). *Metodología de la investigación cuantitativa-cualitativa y redacción de la tesis* (5a ed.). Ediciones de la U. https://bit.ly/3upnPFv

Orsoni, M., Giovagnoli, S., Garofalo, S., Magri, S., Benvenuti, M., Mazzoni, E. y Benassi, M. (2023). Preliminary evidence on machine learning approaches for clusterizing students' cognitive profile. *Heliyon*, *9*(3), 111–121. https://doi.org/10.1016/J.HELIYON.2023.E14506

Otero, A. (2021). Deserción escolar en estudiantes universitarios: estudio de caso del área económico-administrativa. *RIDE. Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, *12*(23), 296. https://doi.org/10.23913/RIDE.V12I23.1084

Pachay-López, M. y Rodríguez-Gámez, M. (2021). La deserción escolar: Una perspectiva compleja en tiempos de pandemia. *Polo del Conocimiento*, *6*(1), 130–155. https://doi.org/10.23857/PC.V6I1.2129

Panagiotakopoulos, T., Kotsiantis, S., Kostopoulos, G., Iatrellis, O. y Kameas, A. (2021). Early dropout prediction in moocs through supervised learning and hyperparameter optimization. *Electronics (Switzerland)*, *10*(14). https://doi.org/10.3390/ELECTRONICS10141701

Panigrahi, B., Kathala, K. C. R. y Sujatha, M. (2023). A Machine Learning-Based Comparative Approach to Predict the Crop Yield Using Supervised Learning With Regression Models. *Procedia Computer Science*, *218*, 2684–2693. https://doi.org/10.1016/J.PROCS.2023.01.241

Perchinunno, P., Bilancia, M. y Vitale, D. (2021). A Statistical Analysis of Factors Affecting Higher Education Dropouts. *Social Indicators Research*, *156*(2–3), 341–362. https://doi.org/10.1007/S11205-019-02249-Y/METRICS

Perez, C. y Rojas, L. (2020). Diseño de un sistema para predecir la deserción de los alumnos mediante machine learning en la Universidad Tecnológica del Perú [Universidad Tecnológica del Perú]. En *Repositorio Institucional - UTP*. http://repositorio.utp.edu.pe/handle/20.500.12867/3843

Prenkaj, B., Velardi, P., Stilo, G., Distante, D. y Faralli, S. (2020). A Survey of Machine Learning Approaches for Student Dropout Prediction in Online Courses. *ACM Computing Surveys (CSUR)*, *53*(3). https://doi.org/10.1145/3388792

Reyes Saldaña, J. y García Flores, R., 2005. El proceso de descubrimiento de conocimiento en bases de datos. Ingenierías, vol. 8, no. 26, pp. 37-47. ISSN 1405-0676.

Rico, A. y Gaytán, N. (2022). Modelos predictivos del rendimiento académico a partir de características de estudiantes de ingeniería. *IE Revista de Investigación Educativa de la REDIECH*, *1*(3), 111–121. https://doi.org/10.33010/IE_RIE_REDIECH.V13I0.1426

Rodríguez Velasco, C. L., García Villena, E., Brito Ballester, J., Durántez Prados, F. Á., Silva Alvarado, E. y Crespo Álvarez, J. (2023a). Forecasting of Post-Graduate Students' Late Dropout Based on the Optimal Probability Threshold Adjustment Technique for Imbalanced Data. *International Journal of Emerging Technologies in Learning (iJET)*, *18*(04), 120–155. https://doi.org/10.3991/IJET.V18I04.34825

Rodríguez Velasco, C. L., García Villena, E., Brito Ballester, J., Durántez Prados, F. Á., Silva Alvarado, E. y Crespo Álvarez, J. (2023b). Forecasting of Post-Graduate Students' Late Dropout Based on the Optimal Probability Threshold Adjustment Technique for Imbalanced Data. *International Journal of Emerging Technologies in Learning (iJET)*, *18*(04), 120–155. https://doi.org/10.3991/IJET.V18I04.34825

Rueda, S., Urrego, D., Páez, E., Velásquez, C. y Hernández, E. M. (2020). Perfiles de riesgo de deserción en estudiantes de las sedes de una universidad colombiana. *Revista de Psicología (PUCP)*, *38*(1), 275–297. https://doi.org/10.18800/PSICO.202001.011

Saccaro, A., Aniceto, M. y de Andrade, P. (2020). Dropout in tertiary education in Brazil: An analysis of the effects of the PNAES Bolsa Permanência. *EconomiA*, *21*(3), 407–421. https://doi.org/10.1016/J.ECON.2020.08.001

Sánchez, M. y Delgado, J. (2020). Gestión Educativa en el desarrollo del aprendizaje en las Instituciones Educativas. *Hacedor - AIAPÆC*, *4*(2), 83–96. https://doi.org/10.26495/RCH.V4I2.1492

Sánchez, T. (2021). Educación superior: factores económicos que inciden en la deserción escolar. Caso de las licenciaturas de la UNID Tlalnepantla. *RIDE Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, *12*(23), 12–21. https://doi.org/10.23913/RIDE.V12I23.1061

Santisteban, O. (2022). Aprendizaje basado en proyectos para desarrollar la competencia construye interpretaciones históricas en estudiantes de una institución educativa pública - Chiclayo [Universidad Cesar Vallejo]. En *Repositorio Institucional - UCV*. https://repositorio.ucv.edu.pe/handle/20.500.12692/99067

Shica, Z. (2022). Modelos de Data Science para mejorar la detección de la deserción académica en la Institución Educativa 88331 en Chimbote - 2021 [Universidad Cesar Vallejo]. En *Repositorio Institucional - UCV*. https://repositorio.ucv.edu.pe/handle/20.500.12692/86968

Solís-Narváez, N. (2022). Teorías de la educación y sus implicancias en el desarrollo humano. *Revista Electrónica de Conocimientos, Saberes y Prácticas*, *5*(1), 79–86. https://doi.org/10.5377/RECSP.V5I1.15122

Taborda, Y. y López, L. (2020). Pensamiento crítico: una emergencia en los ambientes virtuales de aprendizaje. *Revista Innova Educación*, *2*(1), 60–77. https://doi.org/10.35622/J.RIE.2020.01.004

Tamada, M. M., Giusti, R. y Netto, J. F. de M. (2022). Predicting Students at Risk of Dropout in Technical Course Using LMS Logs. *Electronics (Switzerland)*, *11*(3). https://doi.org/10.3390/ELECTRONICS11030468

Terreros, A., Vega, A. y Pupo, J. (2019). Aplicación del Teorema de Bayes en la selección de personal para disminuir la deserción laboral. *593 Digital Publisher CEIT*, *4*(6), 27–40. https://doi.org/10.33386/593dp.2019.6.140

Theerthagiri, P. (2022). Predictive analysis of cardiovascular disease using gradient boosting based learning and recursive feature elimination technique. *Intelligent Systems with Applications*, *1*(8), 121–140. https://doi.org/10.1016/J.ISWA.2022.200121

Trujillo, J., Ricardez, A., Valera, M. y Cuevas, L. (2022). Aprendizaje estadístico basado en niveles de investigación. *Revista Educación*, *46*(1), 454–470. https://doi.org/10.15517/REVEDU.V46I1.45425

UNESCO. (2022). *El abandono escolar por parte de los niños*. https://www.unesco.org/es/gender-equality/education/boys#:~:text=A%20escala%20mundial%2C%20132%20millones,un%20juego%20de%20suma%20cero.

Valles-Coral, M. A., Salazar-Ramírez, L., Injante, R., Hernandez-Torres, E. A., Juárez-Díaz, J., Navarro-Cabrera, J. R., Pinedo, L. y Vidaurre-Rojas, P. (2022). Density-Based Unsupervised Learning Algorithm to Categorize College Students into Dropout Risk Levels. *Data*, *7*(11). https://doi.org/10.3390/DATA7110165

Vega, H., Sanez, E., De La Cruz, P., Moquillaza, S. y Pretell, J. (2022). Intelligent System to Predict University Students Dropout. *International journal of online and biomedical engineering*, *18*(7), 27–43. https://doi.org/10.3991/IJOE.V18I07.30195

Viloria, A. y Lezama, O. B. P. (2019). Mixture Structural Equation Models for Classifying University Student Dropout in Latin America. *Procedia Computer Science*, *160*, 629–634. https://doi.org/10.1016/J.PROCS.2019.11.036

Wang, Z., Ritou, M., Da Cunha, C. y Furet, B. (2023). Contextual classification of chatter based on unsupervised machine learning. *Procedia CIRP*, *1*(7), 390–395. https://doi.org/10.1016/J.PROCIR.2023.03.066

ANEXOS

Anexo 1: Matriz de consistencia.

<table>
<tr><td colspan="4">Título: Modelo predictivo de machine learning utilizando el método Boosting de ensemble para la deserción estudiantil en EBR</td></tr>
<tr><td rowspan="2">Problema General</td><td rowspan="2">Objetivo General</td><td rowspan="2">Hipótesis General</td><td colspan="2">Variables e Indicadores</td></tr>
<tr><td colspan="2">Variable independiente: Modelo predictivo de machine learning</td></tr>
<tr><td rowspan="2">¿Cómo el modelo predictivo de machine learning utilizando el método Boosting de ensemble mejora la predicción de la deserción estudiantil en EBR?</td><td rowspan="2">Determinar la mejora del modelo predictivo de machine learning utilizando el método Boosting en la predicción de la deserción estudiantil en EBR.</td><td rowspan="2">El modelo predictivo de machine learning utilizando el método Boosting de ensemble mejora la predicción de la deserción estudiantil en EBR.</td><td>Indicadores</td><td>Escala de medición</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Problemas Específicos</td><td>Objetivos Específicos</td><td>Hipótesis Específicos</td><td colspan="2">Variable dependiente: Deserción estudiantil</td></tr>
<tr><td rowspan="2">a). ¿Cuán preciso es el modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR?

b). ¿Cuán exacto es el modelo predictivo de machine learning utilizando el método Boosting de</td><td rowspan="2">a). Determinar la precisión del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR

b). Determinar la exactitud del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la</td><td rowspan="2">a). El modelo predictivo de machine learning utilizando el método Boosting de ensemble es preciso en la predicción en la deserción estudiantil en EBR.

b). El modelo predictivo de machine learning utilizando el método Boosting de ensemble es exacto en</td><td>Indicadores</td><td>Escala de medición</td></tr>
<tr><td rowspan="2">Métricas de precisión</td><td>$Precisión = \frac{TP}{(TP + FP)}$

$Exactitud = \frac{(TP + TN)}{Total}$</td></tr>
</table>

ensemble en la predicción en la deserción estudiantil en EBR? c). ¿ Cuáles son los niveles de sensibilidad del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR?	predicción en la deserción estudiantil en EBR c). Determinar cuáles son los niveles de sensibilidad del modelo predictivo de machine learning utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR.	la predicción en la deserción estudiantil en EBR c). Los niveles de sensibilidad del modelo predictivo de machine learning mejoran utilizando el método Boosting de ensemble en la predicción en la deserción estudiantil en EBR. .		$Sensibilidad = \frac{TP}{(FN + TP)}$
Nivel - Diseño de investigación	**Población y muestra**	**Técnicas e instrumentos**	**Estadística por utilizar**	
Tipo: Aplicada **Diseño**: Pre-experimental Longitudinal **Nivel**: Explicativo	**Población**: registros de los estudiantes pertenecientes a los años 2018 a 2022. **Tamaño de muestra:** registros de los estudiantes pertenecientes a los años 2018 a 2022. **Muestreo**: Se consideró un muestro no probabilístico intencional o por conveniencia	**Técnicas**: recolección de registros **Instrumento**: ninguno	**Descriptiva:** Se realizará tablas y gráficos de barras, utilizando el software R. **Inferencial:** Se realizará la prueba hipótesis, donde se hallará el nivel de significancia	

Anexo 2: Matriz de operacionalización de las variables

Variables De Estudio	Definición Conceptual	Definición Operacional	Indicadores	Escala de Medición
Modelo predictivo de Machine Learning	De acuerdo con, Tamada et al. (2022) el modelo predictivo de machine learning es un tipo de modelo que utiliza algoritmos y técnicas de aprendizaje automático para realizar predicciones o estimaciones sobre datos futuros o no vistos.	Modelo de aprendizaje automático que se entrena con datos históricos para hacer predicciones basadas datos de entrada, esta herramienta va siendo entrenada y medida por la actualización de pesos (Nuevo peso), la combinación de modelos débiles (PMF) y el cálculo de errores (ER).	$Nuevo\ peso = Peso\ anterior\ \times Factor\ de\ actualización$	De razón
			$PMF = SPMD \times PAMD$ Donde: PMF: Predicción del modelo fuerte SPDM: Suma de las predicciones de los modelos débiles. PAMD: Peso asignado a cada modelo débil	
			$ER = EV - PMF$ Donde: ER: Error Residual EV: Etiqueta verdadera PMF: Predicción del modelo fuerte	

Variables De Estudio	Definición Conceptual	Definición Operacional	Indicadores	Escala de Medición
Predicción de la deserción estudiantil	De acuerdo con, Viloria y Lezama (2019) señalan que la deserción estudiantil es el abandono prematuro de la educación formal por parte de los estudiantes. Esto puede ocurrir en cualquier nivel educativo, desde la educación primaria hasta la educación superior.	Los resultados de la predicción de la deserción estudiantil se evaluarán por medio de una matriz de confusión, la cual nos ayudará a obtener los valores correspondientes para aplicarlos en los indicadores considerados para este proyecto.	$Precisión = \frac{TP}{(TP + FP)}$ Donde: TP: Verdadero Positivo FP: Falso Positivo	De razón
			$Exactitud = \frac{(TP + TN)}{Total}$ Donde: TP: Verdadero Positivo TN: Verdadero Negativo	
			$Sensibilidad = \frac{TP}{(FN + TP)}$ Donde: TP: Verdadero Positivo FN: Falso Negativo	

Anexo 3: Matriz de datos SIAGIE

Datos preliminares		
Datos	**Tipo**	**Datos**
1	Personal	Nombre
2		Fecha de nacimiento
3		Sexo (H/M)
4		País
5		Lengua materna
6		Segunda lengua
7		Trabaja estudiante (si/no)
8		Horas semanales que labora
9		Nacimiento registrado (si/no)
10		Tipo de discapacidad
11	Parental	Padre vive (si/no)
12		Madre vive (si/no)
13		Escolaridad de la madre
14	Académico	Situacion matricula inicial
15		Promedio final del ult bimestre o trimestre (promedio previo al periodo del retiro o promedio previo al periodo del último registro)
16		Nro de áreas desaprobadas ult. trimestre o bimestre (previo al periodo del retiro o previo al periodo del último registro)
17		% de inasistencias
18		Grado ultimo registro
19		Sección último registro
20		Fecha de de retiro
21		Situación final

Anexo 4: Tabla final de registros

Datos obtenidos				
	Tipo	**Datos**	**Nombre de la variable**	**Valores**
1	Personal	Sexo	sexo	{'H':'0', 'M':'1'}
2		País	pais	{'P':'0', 'OT':'1'}
3		Lengua materna	lengua_materna	{'C':'0','Q':'1'}
4		Nacimiento registrado (si/no)	nacimiento_registrado	{'SI':'0','NO':'1'}
5		Edad en el registro	edad_en_registro	*número entero*
6	Parental	Padre vive (si/no)	padre_vive	{'SI':'0','NO':'1'}
7		Madre vive (si/no)	madre_vive	{'SI':'0','NO':'1'}
8		Escolaridad de la madre	escolaridad_madre	{'S':'0', 'P':'1', 'SE':'2', 'SP':'3'}
9	Académico	Situacion matricula inicial	situacion_matricula_inicial	{'P':'0', 'I':'1', 'R':'2', 'RE':'3'}
10		Promedio final del ult bimestre o trimestre (promedio previo al periodo del retiro o promedio previo al periodo del último registro)	prom_final	{'A':'0', 'B':'1', 'C':'2'}
11		Nro de áreas desaprobadas ult. trimestre o bimestre (previo al periodo del retiro o previo al periodo del último registro)	areas_desaprobadas	*número entero*
12		% de inasistencias	porcentaje_inasistencias	*número entero*
13	**Variable target del modelo**	Situación final	situación_final	{'no_retirado':'0','retirado':'1'}

Anexo 5: Exploración inicial de datos

#	Column	Non-Null Count	Dtype
0	nombre	1206 non-null	object
1	sexo	1206 non-null	object
2	situacion_matricula_inicial	1206 non-null	object
3	pais	1206 non-null	object
4	padre_vive	1206 non-null	object
5	madre_vive	1206 non-null	object
6	lengua_materna	1203 non-null	object
7	segunda_lengua	17 non-null	object
8	trabaja_estudiantes	1206 non-null	object
9	horas_semanales_labora	0 non-null	float64
10	escolaridad_madre	1189 non-null	object
11	nacimiento_registrado	1206 non-null	object
12	tipo_discapacidad	8 non-null	int64
13	edad_en_registro	1206 non-null	object
14	prom_final	1206 non-null	object
15	areas_desaprobadas	1206 non-null	int64
16	porcentaje_inasistencias	1206 non-null	int64
17	grado_ult_registro	1206 non-null	object
18	seccion_ult_registro	1206 non-null	object
19	situacion_final	1206 non-null	object
20	fec_retiro	76 non-null	object

Anexo 6: Hiperparámetros del modelo XGBOOST

Modelo XGBOOST		
Hiperparámetros	Definición	Valores/Rangos
"n_estimators"	Número de árboles de decisión posibles	[50, 80, 100]
"learning_rate"	Tasa de entrenamiento (relacionada a la velocidad de aprendizaje y convergencia)	0.05, 0.10
"max_depth"	Máxima profundidad de cada árbol de decisión (relacionado a la complejidad del modelo)	5, 15
"scale_pos_weight"	Se utiliza para escalar el gradiente de la clase positiva	5, 15
"subsample"	Proporción de datos de entrenamiento	[0.6, 0.8, 1.0]
"colsample_bytree"	Proporción de variables que para la construcción de cada árbol	[0.6, 0.8, 1.0]
"gamma"	permite controlar la fuerza de regularización y evitar el sobreajuste	[0.5, 0.8, 1.0]
'random_state'	Valor para controlar la reproductividad de los resultados durante el entrenamiento del modelo	42
'tree_method'	Valor para especificar el método que se utilizará para construir los árboles en el modelo.	gpu_hist (utiliza la GPU para acelerar el proceso de construcción del árbol)

Anexo 7: Hiperparámetros del modelo LightGBM

Modelo Light GBM		
Hiperparámetros	Definición	Valores/Rangos
"n_estimators"	Número de árboles de decisión posibles	[50, 80, 100]
"learning_rate"	Tasa de entrenamiento (relacionada a la velocidad de aprendizaje y convergencia)	0.01, 0.1
"num_leaves"	Número de hojas del árbol (relacionado a la complejidad del modelo y el control del sobre ajuste)	100, 200, step=10
"max_depth"	Máxima profundidad de cada árbol de decisión (relacionado a la complejidad del modelo y el control del sobre ajuste)	5, 15
"scale_pos_weight"	Se utiliza para escalar el gradiente de la clase positiva	1, 5
"random_state"	Valor para controlar la reproductividad de los resultados durante el entrenamiento del modelo	42
"n_jobs"	Controla el número de hilos paralelos que se utilizan para entrenar el modelo	-1 (LightGBM utilizará todos los núcleos del CPU disponibles)

Anexo 8: Hiperparámetros del modelo CatBoost

Modelo CAT Boost		
"n_estimators"	Número de árboles de decisión	[80, 100, 150]
"learning_rate"	Tasa de entrenamiento (relacionada a la velocidad de aprendizaje y convergencia)	0.05, 0.15
"max_depth"	Máxima profundidad de cada árbol de decisión (relacionado a la complejidad del modelo y el control del sobre ajuste)	7, 10
"scale_pos_weight"	Se utiliza para escalar el gradiente de la clase positiva	1, 5
"eval_metric"	Métrica de evaluación	“Recall"
"task_type"	Valor que especifica el tipo de tarea que aborda el modelo.	"GPU"
'random_state'	Valor para controlar la reproductividad de los resultados durante el entrenamiento del modelo	42