Predicción de default financiero usando métodos de aprendizaje automático

Federico Ocampo Ortiz
Andrés Felipe Orrego Quintero

Monografía presentada para optar al título de Especialista en Analítica y Ciencia de Datos

Asesora
María Bernarda Salazar Sánchez, Doctor (PhD)

Universidad de Antioquia
Facultad de Ingeniería
Especialización en Analítica y Ciencia de Datos
Medellín, Antioquia, Colombia
2023

Cita	(Ocampo Ortiz & Orrego Quintero, 2023)
Referencia **Estilo APA 7 (2020)**	Ocampo Ortiz, F., & Orrego Quintero, A. F. (2023). *Predicción de default financiero usando métodos de aprendizaje automático* [Trabajo de grado especialización]. Universidad de Antioquia, Medellín, Colombia.

Especialización en Analítica y Ciencia de Datos, Cohorte V.

Centro de Investigación Ambientales y de Ingeniería (CIA).

Agradecimientos

Agradecemos inmensamente a todos los profesores de la especialización, los cuales fueron pieza fundamental para el desarrollo de este trabajo, agradecemos muy especialmente a la profesora María Bernarda Salazar Sánchez PhD, por su acompañamiento incansable durante todo el proceso de investigación e implementación y por todos los conocimientos compartidos en el proceso académico. Sin ella esto no hubiese sido posible.

Al Departamento de Ingeniería de Sistemas y Facultad de Ingeniería de la Universidad de Antioquia, por desarrollar el programa de Especialización en Analítica y Ciencia de Datos, que nos ha aportado tantos conocimientos desde el ser, saber y hacer para aplicar en nuestro día a día profesional alrededor de este campo de estudio cada vez más relevante.

Agradecemos a nuestros familiares que nos apoyaron durante todo este año de la especialización, en la que en varias ocasiones sacrificamos tiempo con ellos para poder alcanzar el objetivo de terminar nuestros estudios. A nuestros equipos de trabajo, que fueron pacientes y solidarios durante todo este proceso, manifestando su apoyo para poder cumplir este objetivo.

Tabla de contenido

Lista de tablas

Lista de figuras

Siglas, acrónimos y abreviaturas

AUC-ROC	Area Under the Receiver Operating Characteristic curve
CRISP-DM	Cross Industry Standard Process for Data Mining
KNN	K-Nearest Neighbor
ML	Machine Learning
SMOTE	Synthetic Minority Over-sampling Technique
UdeA	Universidad de Antioquia
XGBoost	Extreme Gradient Boosting

Resumen

En el contexto de la investigación realizada en este proyecto sobre el conjunto de datos "Give Me Some Credit" de Kaggle, el trabajo se focalizó en la potenciación de modelos predictivos, particularmente en estrategias de modelado que utilizaban técnicas como Gradient Boosting y XGBoost. Modelos que después de rigurosas evaluaciones y ajustes, lograron con precisión del 89% y 90%, respectivamente, predecir si una persona entrará en mora por 90 días tal que permita a entidades bancarias definir el riesgo financiero asociado a una solicitud de crédito

Los resultados obtenidos en este trabajo revierten de importancia en el ámbito bancario, dado que la aplicación de técnicas avanzadas de aprendizaje automático se ha vuelto crucial y ha permitido aumentar la capacidad de prever riesgos crediticios de manera precisa, lo cual es esencial para la toma de decisiones estratégicas y la mitigación de riesgos. Lo anterior ofrece a las instituciones financieras la posibilidad de optimizar la evaluación de riesgos, reducir pérdidas y mejorar la eficiencia operativa.

Palabras clave: Kaggle, Gradient Boosting, XGBoost, machine learning

Repositorio Github: https://github.com/00Fede/givemesomecredit

Abstract

In the context of the research conducted on the "Give Me Some Credit" dataset from Kaggle, the focus was on enhancing predictive models, particularly Gradient Boosting and XGBoost. Through rigorous evaluations and adjustments, an improvement in the accuracy and predictive robustness of both models was achieved. These improvements are significant in the banking sector, where the application of advanced machine learning techniques has become crucial. In the financial industry, the accurate prediction of credit risks is essential for informed decision-making. The implementation of enhanced models such as Gradient Boosting and XGBoost provides financial institutions with the opportunity to optimize risk assessment, reduce losses, and improve operational efficiency. This approach not only strengthens credit management but also underscores the growing relevance of machine learning in the transformation and innovation of the banking sector, positioning it as a fundamental tool for strategic decision-making and risk mitigation.

*Keywords***:** Kaggle, Gradient Boosting, XGBoost, machine learning

1. Descripción del problema

Los bancos juegan un papel muy importante en las economías de mercado, pueden decidir quién es financiado, bajo qué términos financiar y hacer o declinar decisiones de inversión. Es así como los mercados, las sociedades, los individuos y las compañías en busca de productos/servicios acordes a sus necesidades y metas requieren tener acceso a créditos, cuyo otorgamiento se ha convertido en una parte fundamental del sistema financiero. Estos han impulsado el crecimiento económico y facilitado el acceso a bienes y servicios tanto a individuos como a empresas. Sin embargo, hoy día las instituciones crediticias enfrentan el desafío de determinar la probabilidad de que un prestatario cumpla con sus obligaciones de pago.

Es por ello, que los bancos hacen uso de algoritmos para calcular el puntaje crediticio a través de una estimación de la probabilidad de no pago, esto conlleva a determinar si un préstamo puede otorgarse o no. Una predicción de la probabilidad de estrés financiero en los clientes permitiría adelantarse a situaciones donde la cartera vencida aumente, a causa de la mora en los pagos de los créditos, y tomar decisiones al respecto como ofrecer acuerdos de pago, campañas o contacto con los clientes.

En este contexto, el uso de técnicas de aprendizaje automático y modelos predictivos ha surgido como una herramienta prometedora para evaluar la solvencia crediticia de los solicitantes. Estos modelos aprovechan la disponibilidad de grandes cantidades de datos históricos, que incluyen información sobre el perfil financiero de los prestatarios, antecedentes crediticios, características demográficas y otros factores relevantes. Al utilizar algoritmos sofisticados, estos modelos pueden identificar patrones ocultos y generar predicciones precisas sobre la capacidad de pago de un individuo.

1.1. Problema de negocio

El fenómeno de prestar y otorgar créditos ha tenido una larga historia en el comportamiento humano. Sin embargo, la capacidad de recolectar información para poder tomar decisiones de crédito surgió hace unas décadas. Para que el sistema financiero y bancario cumpla sus propósitos y mantenga su estabilidad a lo largo del tiempo, es necesario establecer un modelo de evaluación crediticia robusto. El cálculo de puntaje crediticio en el contexto de la aprobación de un crédito es uno de los campos más estudiados y críticos en el sector bancario y financiero. Algunos indicadores

importantes como la cartera y la liquidez pueden verse afectados si se realiza una gestión inadecuada del riesgo financiero.

A esto se suma el crecimiento exponencial que ha tenido la demanda crediticia a nivel global. Esto conlleva a un aumento en las solicitudes de crédito, haciendo que este producto sea uno de los pilares que refleja la rentabilidad y estabilidad de una institución financiera.

1.2. Aproximación desde la analítica de datos

La personalización de la evaluación de crédito principalmente se desarrolla en dos métodos: el método estadístico y el método de aprendizaje de máquina (ML) (Liu, Wang, & Han, 2021).

El objetivo de este estudio es desarrollar un modelo predictivo que permita determinar si una persona va a pagar o no un crédito, utilizando un enfoque basado en el ML. Para lograrlo se seguirá el método CRISP-DM, iniciando con un entendimiento del negocio y los datos disponibles, se aplicarán diferentes técnicas de preprocesamiento y modelado, para posteriormente entrenar algoritmos de aprendizaje automático, como árboles de decisión, regresión logística y redes neuronales, con el fin de evaluar su rendimiento en la predicción de la capacidad de pago de los prestatarios (Shearer, 2000).

La literatura sobre el uso de métodos de aprendizaje de máquina muestra que no hay una técnica por excelencia para calcular el riesgo crediticio, en cambio, se ha visto que diferentes técnicas pueden utilizarse dependiendo del contexto, calidad de la información y precisión requerida (Liu, Wang, & Han, 2021).

1.3. Origen de los datos

Los datos provienen del reto Give Me Some Credit publicado en Kaggle en 2011 (Credit Fusion, 2011), el cual contiene datos financieros como el comportamiento histórico, deuda actual e ingreso mensual y datos demográficos como la edad o el número de personas a cargo (dependientes). Esta información histórica de préstamos y comportamiento de usuarios, permiten tener una base a partir de la cual se puede predecir la capacidad de pago de un cliente.

1.4. Métricas de desempeño

Como métrica de desempeño se utilizará principalmente la métrica AUC-ROC, aprovechando que no es sensible al desequilibrio entre las clases (Fawcett, 2006). Para este caso

se espera una AUC de al menos 80% para considerar el modelo como aceptable. Con este valor se busca aumentar la sensibilidad del modelo para evitar falsos negativos en las predicciones, es decir, clientes que van a tener estrés financiero y son clasificados sin estrés financiero en dos años. La ocurrencia de falsos negativos puede generar pérdidas financieras al no lograr predecir de manera efectiva a aquellas personas que van a presentar incumplimiento.

En el contexto de predicción del default financiero, la clase positiva es la que más interesa, para poder identificar aquellas personas que van a generar mora en sus obligaciones. Teniendo en cuenta que el conjunto de datos esta desbalanceado, una métrica como la exactitud puede ser engañosa. Por esta razón, se selecciona F1-score y Sensibilidad como métricas secundarias para medir el desempeño del modelo.

Como métrica de negocio se puede determinar la disminución de la cartera vencida al utilizar el modelo como criterio para la aprobación de créditos financieros. También es importante analizar qué pasaría si el modelo califica de manera negativa a un cliente, y en el caso contrario, qué ocurre si el modelo califica de manera positiva. Dar una calificación crediticia negativa, supone el tomar medidas frente a un cliente que pueda entrar en mora, por ejemplo, rechazar la solicitud de crédito u ofrecer una línea que se ajuste a su capacidad. Por otro lado, una calificación crediticia positiva, permite ofrecer productos financieros a un cliente, aumentar su cupo o generar productos preaprobados de fácil acceso.

Finalmente, se espera que una entidad bancaria al utilizar el modelo planteado en esta monografía disminuya la cartera vencida en un porcentaje que debe ser definido por la misma entidad. Esta disminución debe reflejarse en los siguientes años según el modelo financiero de la entidad, teniendo en cuenta que la cartera se calcula a cortes trimestrales, semestrales o anuales.

2. Objetivos

2.1. Objetivo general

Desarrollar un modelo para predecir si una persona entrará en mora por noventa (90) días como estrategia de calificación utilizada por entidades bancarias para definir el riesgo financiero asociado a una solicitud de crédito.

2.2. Objetivos específicos

1. Identificar el conjunto de características que más aporten a la clasificación de la capacidad de pago de un cliente.
2. Proponer y evaluar una estrategia de clasificación de la capacidad de pago de un cliente.
3. Validar los resultados del modelo de aprendizaje automático en términos de las implicaciones, beneficios e impacto que puede tener el negocio, a través de la métrica de desempeño y de negocio establecida.

3. Datos

3.1. Datos originales

La base de datos utilizada se denomina Give Me Some Credit y fue obtenida del reto de Kaggle propuesto en el año 2011 (Credit Fusion, 2011). Los datos proporcionados vienen subdivididos así: conjunto de datos de entrenamiento conformado por 150.000 muestras, 11 variables de entrada y una variable de salida (SeriousDlqin2yrs), mientras que los datos de validación cuentan con aproximadamente 100.000 muestras. En total se tienen los siguientes cuatro archivos (ver Tabla 1):

A. 'Data Dictionary.xlsx': Describe cada una de las variables junto con especificar el tipo de dato.
B. 'cs-test.csv': Este archivo tiene un total de 101.503 instancias con todas las características y sin información de la variable de salida SeriousDlqin2yrs, este archivo es utilizado para la validación del modelo implementado.
C. 'cs-training.csv': Comprende 150.000 instancias con las características y la correspondiente variable predictora, este archivo es utilizado en la fase de entrenamiento del modelo.
D. 'sampleEntry.csv': Este archivo contiene la referencia para computar las predicciones del modelo implementado y poder participar en la competición en la plataforma Kaggle.

Tabla 1 *Dimensiones de las fuentes de datos utilizadas*

Nombre del archivo fuente	Número de instancias	Peso (MB)
cs-test.csv	101.503	4.867
cs-training.csv	150.000	7.388
sampleEntry.csv	101.503	1.863
Data Dictionary.xls	N/A	0.015

Las instancias cuentan con once (11) características numéricas. Dos (2) son de tipo porcentaje, con valores entre 0 y 1, una (1) es de tipo real y las siete (7) restantes son de tipo entero. Esta base de

datos no cuenta con variables categorías, para facilitar su comprensión se han realizado modificaciones en los nombres de las características (ver Tabla 2).

Tabla 2 *Descripción de las variables de entrada y salida del modelo de predicción*

Nombre original	Nuevo nombre	Descripción	Tipo
RevolvingUtilizationOfUnsecuredLines	TasaUtilizacionLineasRotativas	Saldo total en tarjetas de crédito y líneas de crédito personales, excepto bienes raíces y sin deuda a plazos, como préstamos para automóviles, dividido por la suma de los límites de crédito	Porcentaje
Age	Edad	Edad en años	Entero
NumberOfTime30-59DaysPastDueNotWorse	Mora30a59dias	Número de veces que el prestatario ha estado atrasado entre 30 y 59 días en los últimos 2 años.	Entero
DebtRatio	RazonDeudaMensual	Pagos mensuales de deuda, pensión alimenticia, costos de vida divididos por el ingreso bruto mensual	Porcentaje
MonthlyIncome	IngresoMensual	Ingresos mensuales	Real
NumberOfOpenCreditLinesAndLoans	CantidadCreditosActivos	Número de créditos del prestatario, incluyendo tarjetas de crédito y créditos rotativos	Entero
NumberOfTimes90DaysLate	MoraMayorA90	Número de veces que el prestatario ha estado en mora por 90 días o más.	Entero
NumberRealEstateLoansOrLines	CantidadCreditosDeVivienda	Número de préstamos hipotecarios y de bienes raíces, incluidas las líneas de crédito con garantía hipotecaria	Entero
NumberOfTime60-89DaysPastDueNotWorse	Mora60a89dias	Número de veces que el prestatario ha estado atrasado entre 60 y 89 días los últimos 2 años.	Entero
NumberOfDependents	CantidadDependientes	Número de dependientes en la familia (cónyuge, hijos, etc.)	Entero
SeriousDlqin2yrs	Incumplimiento	Probabilidad de que una persona experimente 90 días o más de morosidad.	Porcentaje

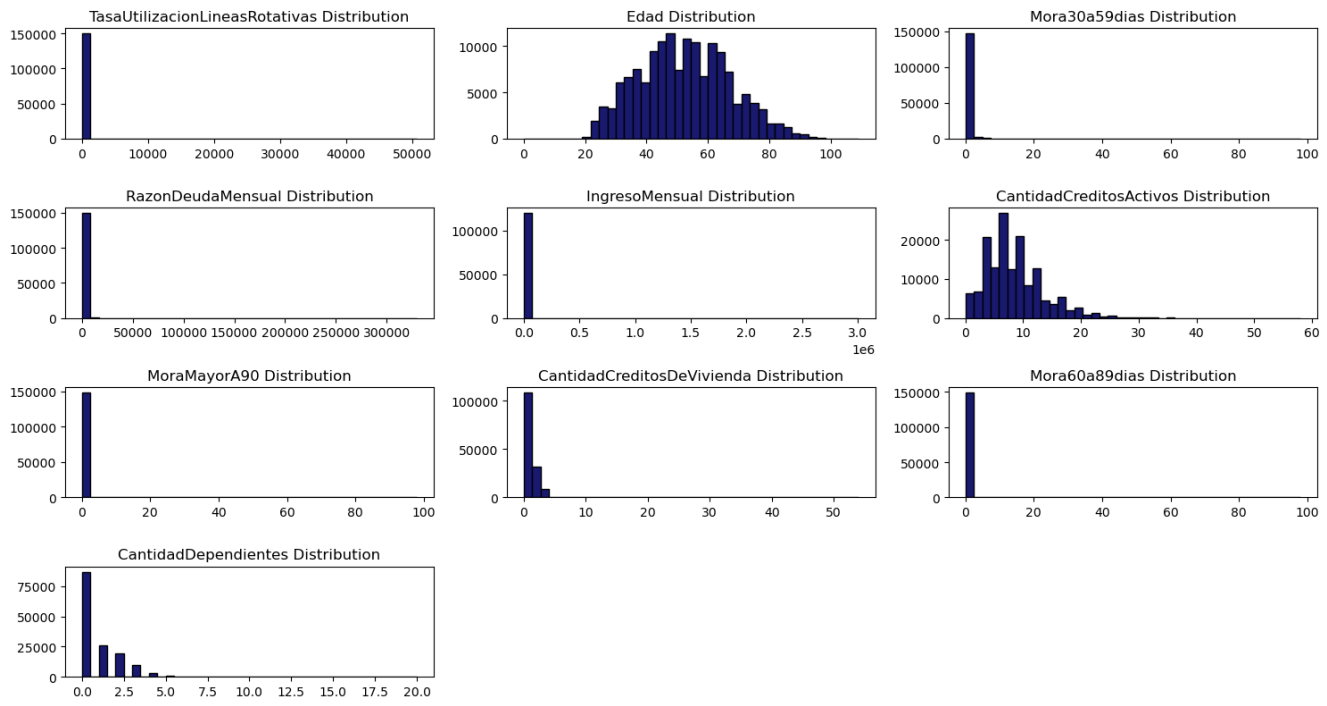
3.2. Datasets

Para este problema se encuentran preestablecidos los conjuntos de datos de entrenamiento y validación (Credit Fusion, 2011). El conjunto de entrenamiento cuenta con 150.000 y el de validación con 101.503 instancias. Sin embargo, se deben realizar transformaciones en algunas de las características antes del entrenamiento del modelo (ver el subcapítulo de Preprocesamiento).

3.3.Analítica descriptiva

Cómo se evidenció en la Tabla 2, la base de datos está conformada por diez (10) variables numéricas y una variable de salida binaria. El análisis de distribución de la información original se relaciona en la Figura 1, en la cual se puede evidenciar una población con edades entre 20 y 80 años que siguen una distribución gaussiana. En su gran mayoría no tienen personas a cargo (dependientes) y tienen entre 0 a 20 créditos activos, que pueden ser créditos rotativos (como tarjetas de crédito) o prestamos (crédito de auto, hipoteca, etc.).

Figura 1 *Distribución de los datos originales para las variables de entrada.*



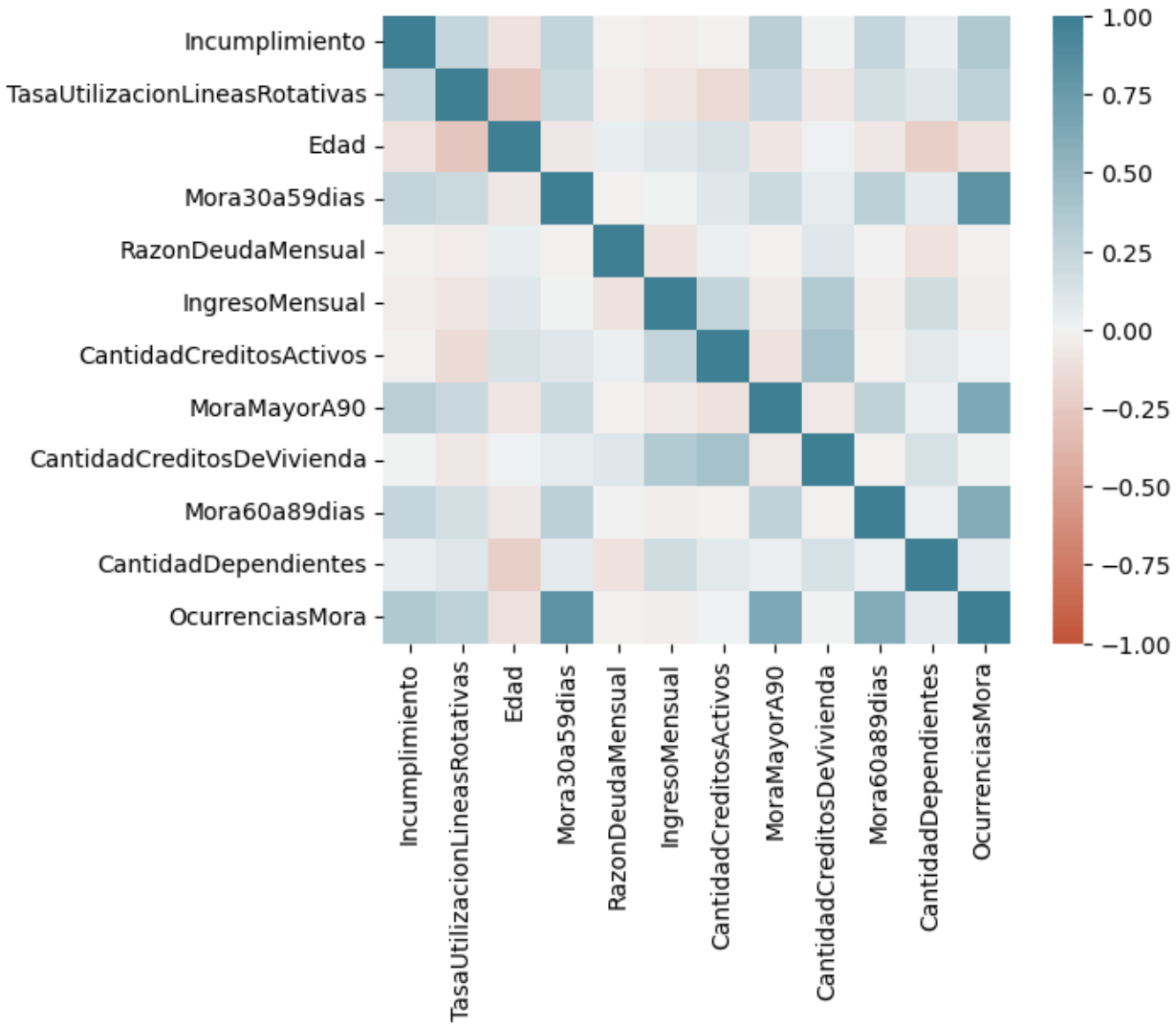
A continuación, a partir del histograma de la Figura 1 se desprenden las siguientes conclusiones específicas para cada variable de la base de datos:

- La variable *TasaUtilizacionLineasRotativas* es una razón dada por el balance total de las tarjetas de crédito y líneas de crédito dividido por el cupo o límite de estos créditos. Un porcentaje mayor al 100% indica que se están usando líneas de crédito por encima del cupo designado, lo cual dista del funcionamiento real de este tipo de productos financieros.
- Para la variable porcentual *RazónDeudaMensual* se consideran valores atípicos todos aquellos mayores a uno (100%), un valor por encima significa que las deudas superan los ingresos percibidos mensualmente, lo cual es posible y podría servir para identificar personas con alto riesgo de incumplimiento en el pago de sus obligaciones. Sin embargo, valores muy alejados de la media, como 300.000, valor máximo de esta variable suponen una situación financiera mensual muy poco probable en la vida real.
- Las variables *Mora30a59dias*, *Mora60a89dias*, *MoraMayorA90* corresponden a las ocurrencias de mora de una persona segmentadas por intervalos de tiempo. Para estas variables se definen unos rangos validos en los que pueden estar los datos teniendo en cuenta la ventana de tiempo de dos años (730 días). Para *Mora30a59dias* el valor máximo permitido es 25 y para *Mora60a89días* el valor máximo permitido es 13. No es posible valores que superen estos umbrales para la ventana de tiempo.
- Para la variable *IngresoMensual* se encuentra que el 75% de los valores está por debajo de los $10.000; esto concuerda con lo esperado para esta característica. Sin embargo, se encuentran algunos datos demasiado elevados, como una persona con ingresos mensuales de $3.008.750, lo cual se hace preguntar si una persona con estos ingresos realmente tendría deudas o necesite créditos.

Para identificar la relación lineal entre las variables se hace uso de la matriz de correlación, esto permite encontrar si una variable aumenta o disminuye en función de otra. En la Figura 2, se observa, por ejemplo, `CantidadCreditosDeVivienda` y `CantidadCreditosActivos` tienen una correlación de 0.43, lo cual indica que a medida que la cantidad de créditos de vivienda aumenta, en un 43% se genera un aumento en el número de créditos activos que tiene un mismo usuario.

En esta matriz también se obtuvo una alta correlación entre las variables temporales *Mora30a59dias*, *Mora60a89dias* y *MoraMayorA90*. Lo anterior se debe a que son variables que representan temporalmente un mismo fenómeno: ocurrencias de mora en diferentes intervalos de tiempo. Otra relación interesante es entre las variables *Edad* y *CantidadDependientes*, que tienen una correlación de -0.21, lo que puede significar que mientras mayor es la edad menor es la cantidad de personas a cargo.

Figura 2 *Matriz de correlación entre las variables de ingreso y salida del modelo.*

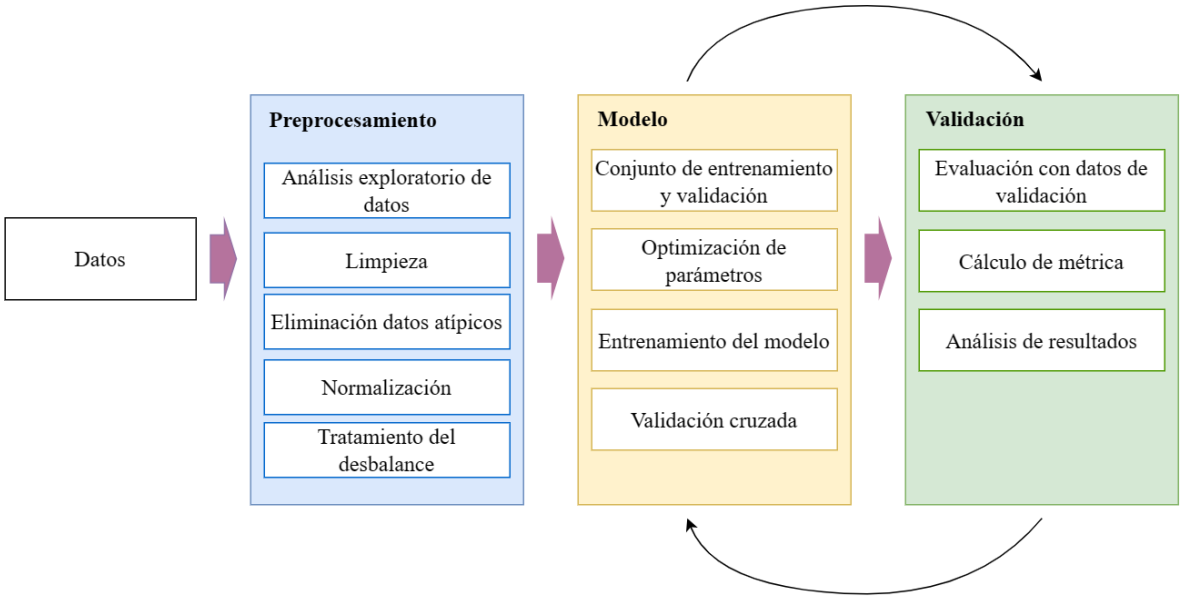


4. Metodología

Para llevar a cabo un buen modelo de aprendizaje de maquina es necesario un flujo de trabajo que abarque desde la etapa de extracción de los datos, hasta la validación de las métricas resultantes de la ejecución del modelo. En cada una de estas etapas se realizan tareas independientes sobre los datos y el modelo creado, con el fin de potenciar los resultados y estos reflejen la realidad del fenómeno que se está intentando predecir.

En la Figura 3 se muestra el pipeline principal con las etapas y tareas específicas asociadas a cada una, así como los procesos iterativos planeados entre las diferentes etapas que permitieron poder optimizar el modelo.

Figura 3 *Pipeline principal de trabajo*



4.1. Preprocesamiento

El preprocesamiento es la primera fase en el proceso de analítica y las decisiones tomadas en está pueden contribuir significativamente en los resultados entregados por el modelo implementado. Consiste en la transformación, limpieza y eliminación de datos de acuerdo con criterios de calidad y el contexto del problema. Realizar un procesamiento adecuado en los datos permite potenciar los resultados entregados por los modelos de aprendizaje automático. Sin

embargo, se debe ser cuidadoso con estas manipulaciones en los datos, dado que se pueden generar sesgos en los fenómenos representados, o perder información relevante para describir el problema y su contexto.

Primero se realiza un análisis exploratorio de los datos con el fin de entender el comportamiento y distribución de cada una de las variables. Para este análisis es necesario entender el contexto del problema, conocer sobre el dominio de los datos y tener claro el fenómeno que buscan representar los datos. Un conocimiento experto puede ser muy útil en esta etapa porque permite identificar sesgos, datos erróneos o desviaciones en los datos que puedan afectar la interpretación del problema o el resultado de los modelos predictivos.

A partir de esto, se aplican diferentes técnicas que puedan preparar mejor el conjunto de datos para la implementación de un modelo. Algunas prácticas aplicadas fueron: detección de valores desconocidos, detección de atípicos, creación de nuevas variables, normalización y tratamiento del desbalance de clases.

4.2.Modelos implementados

Las técnicas utilizadas para la creación de los modelos son las siguientes:

4.2.1. Regresión logística

Modela la probabilidad de que un objeto pertenezca a una clase en función de sus características. Utiliza la función logística (sigmoide) para transformar una combinación lineal de características en una probabilidad en el rango [0, 1]. Este modelo tiene algunos parámetros para tener en cuenta:

- Random_state: Este parámetro se usa para establecer una semilla y así garantizar que el modelo genere el mismo resultado cada vez que se entrene el modelo, siempre y cuando no se cambien los otros parámetros y los datos.
- Max_iter: este parámetro controla el número máximo de iteraciones que hará el algoritmo de regresión logística para converger.

4.2.2. Clasificador K-NN

El k-NN clasifica un punto de datos basado en la mayoría de las clases de sus k vecinos más cercanos en el espacio de características. Para este método hay varios parámetros que se pueden tener en cuenta:

- N_neighbors: Este es el parámetro más importante ya que de él depende cuál es el número de vecinos más cercanos que se consideran para la clasificación.
- Weights: Es la función de peso usada en la predicción, por defecto se usa la opción 'uniform', en ésta todos los puntos en cada vecindad tienen iguales pesos, otra opción es 'distance' en este caso los vecinos más cercanos al punto de consulta tienen mayor peso que los puntos más alejados

4.2.3. Gradient Boosting

Este método combina múltiples árboles de decisión débiles para crear un modelo fuerte. Entrena árboles de forma secuencial, enfocándose en los errores cometidos por los modelos anteriores. Cada árbol se ajusta para corregir los errores del modelo anterior, y la suma de estos árboles proporciona una predicción final. Este método tiene diferentes parámetros para tener en cuenta como:

- N_estimators: determina el número de árboles de decisión del modelo, entre más estimadores tenga, más complejo será el modelo lo cual repercute en el tiempo de entrenamiento requerido. En el caso del modelo se está usando 150 estimadores
- Learning_rate: controla la contribución de cada estimador al modelo, un número menor en el learning rate, hace que los estimadores tengan una contribución más débil, haciendo que el modelo sea más robusto, pero a costa de tener más estimadores. Es importante ajustar el número de estimadores y el learning rate en conjunto. En el caso del modelo se usa un learning rate de 0.05.
- Max_depth: Este parámetro controla la profundidad máxima de cada árbol de decisión, un valor pequeño limita la profundidad de los árboles lo que ayuda a prevenir sobreajuste. En el caso del modelo trabajado se usa un max_depth de 5
- Random_state: al establecer un valor para este parámetro, se garantiza que el modelo genere el mismo resultado cada vez que se ejecute el código, siempre que los otros parámetros y datos no se cambien. En el caso del modelo el número utilizado es 231

4.2.4. Random Forest

Esta técnica construye un conjunto de árboles de decisión que se entrena de forma independiente. Cada árbol vota por una clase y la clase con más votos se elige la predicción final. La aleatoriedad en la construcción de árboles y la combinación de múltiples árboles ayudan a reducir el sobreajuste y mejorar la precisión. Para el árbol de decisión hay diferentes parámetros que se deben tomar en cuenta

- n_estimators: este parámetro determina el número de árboles de decisión que se incluirán en el conjunto del random forest, es importante tener en cuenta que entre más árboles se utilicen, mayor será el tiempo de entrenamiento del modelo. Para el caso del modelo que se está trabajando el número elegido de árboles es 100, para no exceder mucho los tiempos de entrenamiento.
- Max_depth: Este parámetro controla la profundidad máxima de cada árbol de decisión, un valor pequeño limita la profundidad de los árboles lo que ayuda a prevenir sobreajuste. En el caso del modelo trabajado se usa un max_depth de 5
- Max_features: Este parámetro controla la cantidad de características que se consideran al buscar la mejor división en cada nodo de un árbol de decisión, este parámetro podría tomar diferentes valores como: ‘sqrt’ (raíz cuadrada del número de características), ‘log2’ (logaritmo en base 2 del total de características, ‘auto’ (considera todas las características) o un número entero para indicar directamente cuántas características tomar. En el caso del modelo se toman todas las características, debido a que son muy pocas.
- Criterion: este parámetro se utiliza para especificar la función que se utilizará para medir la calidad de una división en un nodo del árbol, las dos opciones más comunes son: gini y entropy. Generalmente la elección entre una u otra función no impacta el rendimiento del modelo ya que trabajan de manera similar, en el caso del modelo se toma la función gini

4.2.5. XGBoost Classifier

Del inglés Extreme Gradient Boosting, el cual es una versión mejorada de Gradient Boosting que utiliza técnicas avanzadas para mejorar el rendimiento y la velocidad. Utiliza una función de pérdida específica y regularización para controlar el ajuste y la complejidad del modelo, lo que lo

hace altamente efectivo en una variedad de problemas de clasificación y regresión. Los parámetros usados en este modelo son similares a los de Gradient Boosting.

4.3. Evaluación y validación

Los datasets suministrados por Kaggle tienen una particularidad y es que los datos de entrenamiento y validación ya están divididos, sin embargo, para poder tener las métricas de los modelos creados, se optó por usar solo los datos de entrenamiento, y estos, los dividimos en dos subconjuntos uno para entrenamiento que es el 70% de los datos y otro para la validación con el 30% restante, con este último tomamos las métricas de desempeño de los modelos.

4.3.1. Matriz de Confusión

Para la validación del modelo se utilizó la matriz de confusión. Esta herramienta es muy popular en problemas de clasificación binario porque permite de manera simple identificar la relación y dependencia entre las clases a predecir. A partir de la matriz de confusión, es posible obtener las predicciones correctas en la clase positiva (True Positive), predicciones correctas de la clase negativa (True Negative), predicciones incorrectas de la clase positiva (False Positive) y predicciones incorrectas de la clase negativa (False Negative).

La matriz de confusión permite determinar ciertas métricas que son muy importantes.

- Exactitud: mide la proporción de predicciones correctas sobre el total de predicciones.
- Precisión: Proporción de instancias positivas correctamente clasificadas sobre el total de instancias positivas clasificadas, es útil cuando los falsos positivos son costosos
- Recall: Proporción de instancias positivas correctamente clasificadas entre todas las instancias que son realmente positivas, es útil cuando los falsos negativos son costosos.
- F1score: Media armónica entre precisión y recall. Es una métrica muy usada cuando hay desequilibrio de clases

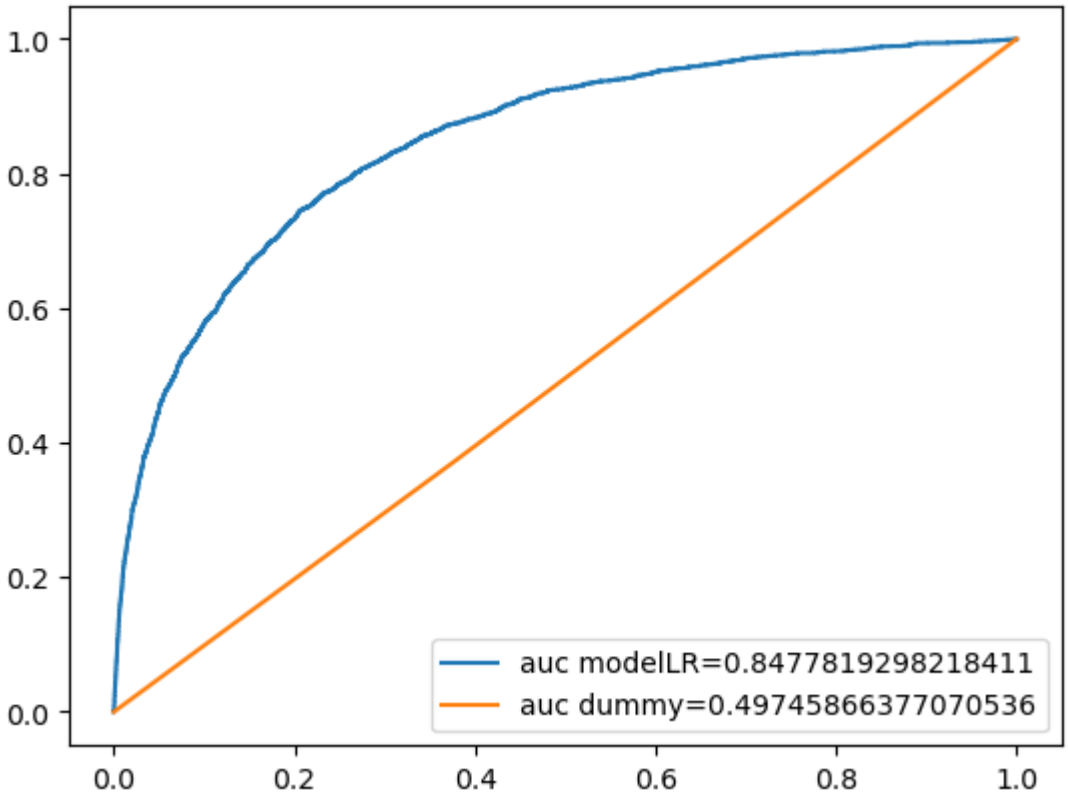
4.3.2. Curva característica operativa del receptor (ROC, acrónimo de Receiver Operating Characteristic)

La curva ROC es una representación gráfica que muestra la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos para diferentes valores de umbral de clasificación. La AUC mide el área bajo esta curva. La forma de interpretar el área bajo la curva es que entre más

cercano esté a 1 la métrica de AUC, mejor será el rendimiento del modelo, y entre más cerca esté de 0.5, más aleatorio es el comportamiento del modelo.

En la Figura 4 Curva ROC de referencia se tiene la curva ROC de un modelo en comparación con una línea dummy, ésta última simula un comportamiento aleatorio al tener un AUC de aproximadamente 0.5, mientras la línea azul muestra una curva con AUC de 0.84, lo cual es más cercano a una buena predicción para la métrica.

Figura 4 Curva ROC de referencia



5. Proceso de analítica y preprocesamiento

5.1.Tratamiento de valores desconocidos

Esta etapa busca detectar aquellos datos que originalmente son vacíos, nulos o en la forma NaN (Not a Number). Para estos casos se aplica la estrategia de imputación, que consiste en rellenar estos valores con estadísticos calculados desde los datos conocidos. Se pueden seleccionar estadísticos como la media o la mediana dependiendo de la distribución de los datos en la variable. La estrategia aplicada se describe en la Tabla 3. Para la variable *IngresoMensual* se seleccionó como estadístico de imputación la mediana, para evitar el impacto que pudiera generar la presencia de atípicos.

Tabla 3 Imputación de valores desconocidos

Variable	Cantidad valores nulos o desconocidos	Estrategia de imputación	Valor imputado
CantidadDependientes	3,678	Media	0
IngresoMensual	26,451	Mediana	5.400

Finalmente, se eliminó la columna *Unnamed: 0* dado que es un índice para cada fila y no aporta valor en el modelo.

5.2.Detección de atípicos

Esta etapa consiste en remover de los datos de entrada aquellos que están por fuera de los valores esperados y puedan generar sesgos en la caracterización de la información.

Se procedió a la detección de atípicos usando el algoritmo LOF (Local Outlier Factor) que mide la desviación local de la densidad de una muestra con una cantidad de vecinos dada (Breunig, 2000). Sin embargo, está técnica tuvo que descartarse dado que eliminaba valores que no debían ser considerados atípicos según el contexto del problema.

Finalmente, se decide definir unas condiciones para cada variable de acuerdo con su distribución y coherencia con el contexto, con el fin de detectar y eliminar aquellos registros que no cumplen con las condiciones esperadas. Los resultados de este procesamiento se detallan en la Tabla 4. Las condiciones definidas fueron las siguientes:

- *Edad* es menor a 21, dado que corresponden a personas menores de edad que están por fuera del alcance del problema.
- *TasaUtilizacionLineasRotativas* es mayor a 1.
- *Mora30a59dias* es mayor a 25, dado que excede el máximo posible para esta variable.
- *Mora60a89dias* es mayor a 13, dado que excede el máximo posible para esta variable.
- *RazónDeudaMensual* se eliminan valores por encima de 4.000. Si bien se espera que esta variable porcentual tenga valores entre 0 y 1, se encontró que el 98.1% de los datos estaban por debajo de 4.000. Se maneja este umbral, dado que puede contener información relevante para el modelo.
- *CantidadDependientes* mayor a 4.
- *IngresoMensual* es mayor a 20.000.
- *CantidadCreditosDeVivienda* es mayor a 20.

Tabla 4 Detección y eliminación de atípicos en los datos

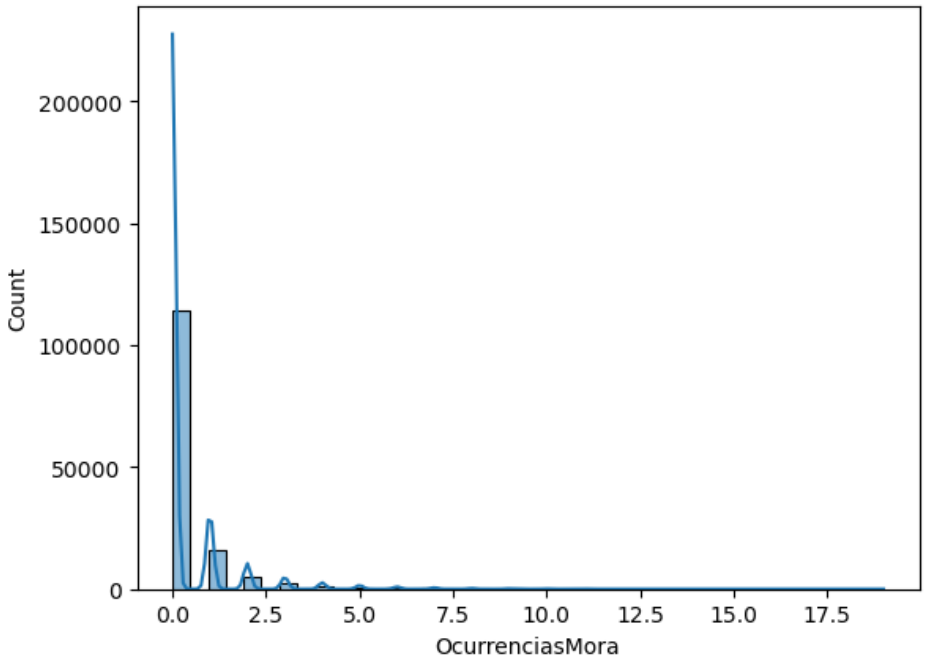
Variable	Umbral	Cantidad atípicos detectados	Porcentaje respecto a datos originales
Edad	Menor a 21	1	~0.00%
TasaUtilizacionLineasRotativas	Mayor a 1	3,321	2.21%
Mora30a59dias	Mayor a 25	269	0.18%
Mora60a89dias	Mayor a 13	269	0.18%
RazonDeudaMensual	Mayor a 4000	2,693	1.80%
CantidadDependientes	Mayor a 4	949	0.63%
IngresoMensual	Mayor a 20.000	2,022	1.35%
CantidadCreditosDeVivienda	Mayor a 20	9	0.01%
TOTAL		9.268	0.94%

5.3.Creación de nuevas variables

En la etapa de preprocesamiento y análisis de los datos resulta importante identificar nuevas variables que puedan potenciar el poder predictivo del modelo. Estas variables normalmente se obtienen a partir de operaciones en las variables originales, o al incluir nuevos conjuntos de datos que puedan relacionarse.

En este caso, se creó la variable *OcurrenciasMora*, la cual resulta de la suma todas las ocurrencias de mora que tuvo una persona independiente. Esta variable puede calcularse sumando los valores de las variables Mora30a59dias, Mora60a89dias y MoraMayorA90, que indican la cantidad de veces que una persona entra en mora entre 30 a 59 días, 60 a 89 días o más de 90 días, respectivamente. El histograma de la Figura 5 muestra la distribución de los datos para esta nueva variable. Aquí se puede evidenciar que la mayoría de las personas cuentan con ocurrencia de mora igual a cero, que es inherente al desbalance de clases propio de este tipo de problemas.

Figura 5 *Distribución de variable OcurrenciasMora en la base de datos*



5.4.Normalización de los datos

Esta es una operación de transformación que se aplica a los datos de entrada para que sus valores queden dentro de un mismo rango definido. Este tipo de operación es muy útil para la optimización del modelo a construir dado que facilita la convergencia de este, da una misma importancia a las mismas variables, y en general ofrece un mejor rendimiento en comparación con un modelo sin normalización de datos (Dodge, 2003).

Para este caso se implementó la estrategia de escalamiento Min-Max, que transforma los datos de manera que queden en un mismo rango definido, normalmente entre [0,1], y se mantenga la distribución (Han, 2011). Específicamente, el escalamiento Min-Max transforma los datos para cada característica, para que se ajusten a un límite mínimo *a* y límite máximo *b*, así:

$$X' = a + \frac{(X - X_{\min})(b - a)}{X_{max} - X_{min}}$$

Dónde,

a, corresponde al límite mínimo de la normalización

b, corresponde al límite máximo de la normalización

X, dato de entrada de la característica imputada

$X_{\min}$, dato mínimo de la característica imputada

$X_{\max}$, dato máximo de la característica imputada

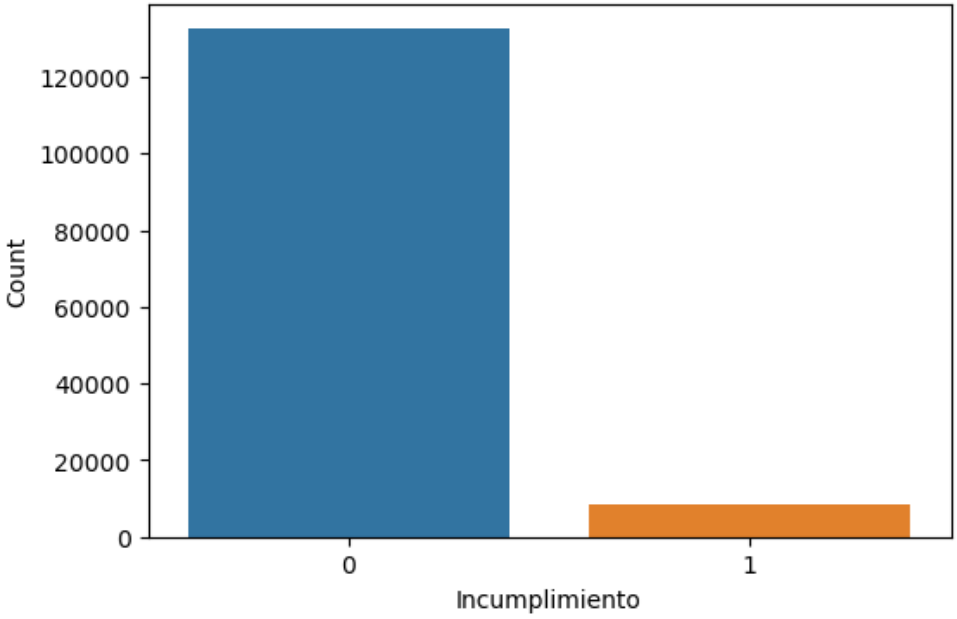
X', dato de salida después de la normalización

5.5.Desbalance de clases

Los problemas de predicción de default financiero suelen tener clases desbalanceadas en el orden de hasta 100 a 1 (Provost & Fawcett, 2001). Tratar el desbalance se vuelve crucial dado que se interesa predecir correctamente la clase minoritaria, que es la que representa el default financiero. Adicional a esto, predecir de manera incorrecta la clase minoritaria podría afectar la efectividad del modelo.

Para el problema planteado la situación normal (mayoritaria) es que la persona cumpla con sus pagos, mientras que la situación anormal (minoritaria) corresponde al incumplimiento del pago. En la Figura 6 se evidencia la relación entre la clase mayoritaria y la clase minoritaria para todo el conjunto de datos

Figura 6 *Distribución de las clases en el conjunto de datos*



Una de las técnicas utilizadas para tratar el desbalance de clases es el remuestreo, que puede darse como el submuestreo de la clase mayoritaria o el sobremuestreo de la clase minoritaria en búsqueda del equilibrio entre las clases (Kubat, Holte, & Matwin, 1998) (Japkowicz, 2000) (Lewis & Catlett, 1994) (Ling & Li, 1998).

SMOTE (Synthetic Minority Over-sampling Technique) es una técnica muy popular de submuestreo propuesta en 2002 (Chawla, Hall, Kegelmeyer, & Philip, 2002). De manera general, consiste en seleccionar muestras cercanas en el espacio de características, trazar una línea entre ellas y generar una nueva muestra en un punto de esta línea. Los autores de SMOTE recomiendan una combinación entre esta técnica y un submuestreo en la clase mayoritaria para obtener mejores resultados.

Se realizó el tratamiento del desbalance de clases en dos etapas, aplicando SMOTE para el sobremuestreo de la clase minoritaria, seguido de un submuestreo aleatorio en la clase mayoritaria. En la Tabla 5 se detallan estas etapas y la evolución de la relación entre las clases del conjunto de entrenamiento.

Tabla 5 *Tratamiento del desbalance de clases en el conjunto de entrenamiento*

	Cantidad de observaciones por clase		**Razón entre clases**	**Observaciones totales**
	Cumplimiento	**Incumplimiento**		
Original	92.769	5.750	16:1	98.519
Sobremuestreo SMOTE	92.769	9.276	10:1	102.045
Submuestreo Aleatorio	18.552	9.276	2:1	27.828

Para validar que el tratamiento realizado no afecta la capacidad de predicción y generalización del modelo, se implementó un modelo básico de XGBoost. Se evaluó este modelo en el conjunto de prueba y se calcularon las métricas de exactitud, precisión, sensibilidad, F1-score y AUC. Los resultados de las métricas en la Tabla 6 muestran un mejor puntaje F1-score después del remuestreo, esto significa que hay una mejor relación entre la precisión y sensibilidad del modelo. También se encuentra un aumento considerable de la sensibilidad, lo que indica que el modelo tiene más capacidad para predecir la clase de interés. Por otro lado, la métrica AUC no se ve afectada por los cambios en la cardinalidad del conjunto de entrenamiento.

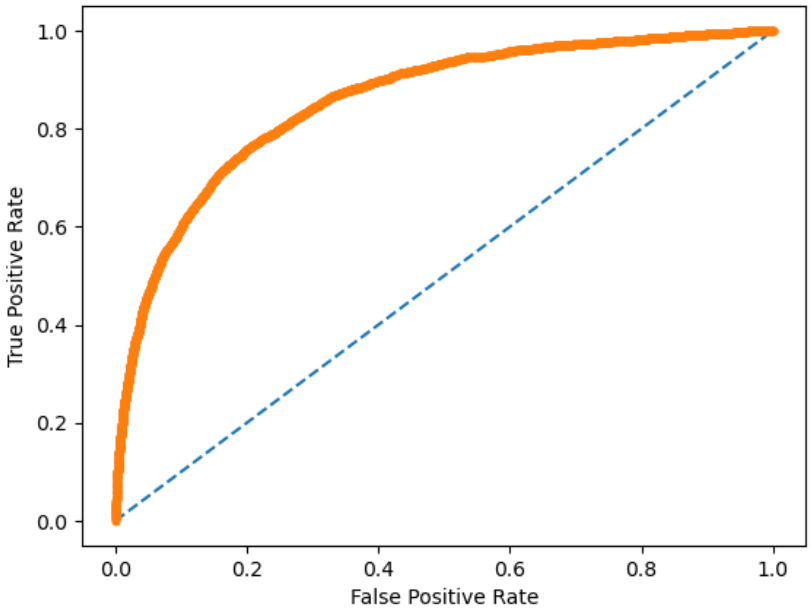
Tabla 6 *Comparación de métricas antes y después del remuestreo*

	Exactitud	**Precisión**	**Sensibilidad**	**F1-score**	**AUC**
Antes del remuestreo	94.37%	63.70%	14.40%	23.49%	85.70%
Después del remuestreo	89.92%	31.02%	55.42%	39.78%	85.70%

6. Resultados y discusión

Inicialmente se implementó un modelo básico de regresión logística para establecer una línea base de comparación frente a otros modelos más robustos. La regresión logística se realiza con los parámetros predeterminados. Al aplicar el modelo al conjunto de prueba se obtienen las siguientes métricas: exactitud 89.24%, precisión 30.90%, sensibilidad 55.38%, F1-SCORE 39.67% y AUC de 85.7%. La curva ROC resultante se evidencia en la Figura 7.

Figura 7 *Curva ROC para modelo de regresión logística*

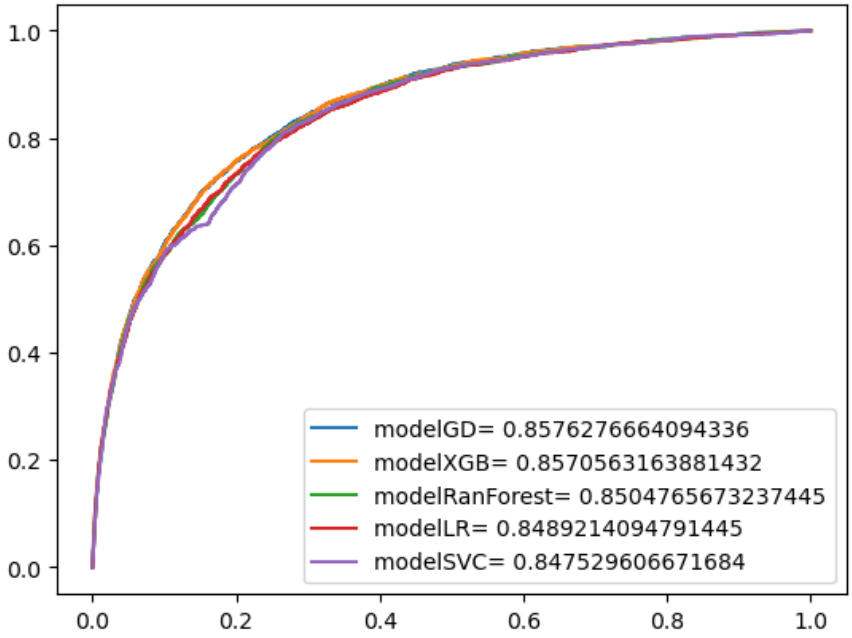


Para la selección del modelo se tuvo en cuenta las métricas F1-score, precisión, exactitud, sensibilidad y AUC. Se buscó un buen balance entre la sensibilidad y la precisión, que son métricas basadas en la relevancia de las predicciones. Por esta razón se tomó como criterio de decisión del modelo la métrica F1-score. Para el cálculo de las métricas se realizó validación cruzada repetida y estratificada de 10 k-folds y 3 repeticiones en el conjunto de entrenamiento. La curva ROC resultante para los modelos se muestra en la Figura 8 y en la Tabla 7 se muestra la media de las métricas para cada uno de los modelos.

Tabla 7 *Métricas de los modelos iniciales*

Modelo	Exactitud	Precisión	Sensibilidad	F1	AUC
Gradient Boosting	83.75%	80.59%	67.53%	73.47%	85.76%
XGBoost	83.63%	80.72%	66.89%	73.15%	85.70%
Random Forest	80.59%	76.09%	60.94%	67.66%	85.04%
Regresión Logística	78.83%	74.81%	55.05%	63.41%	84.89%
Máquina de Soporte Vectorial	78.43%	80.50%	46.60%	59.00%	84.75%

Figura 8 *Curva ROC para los modelos iniciales*



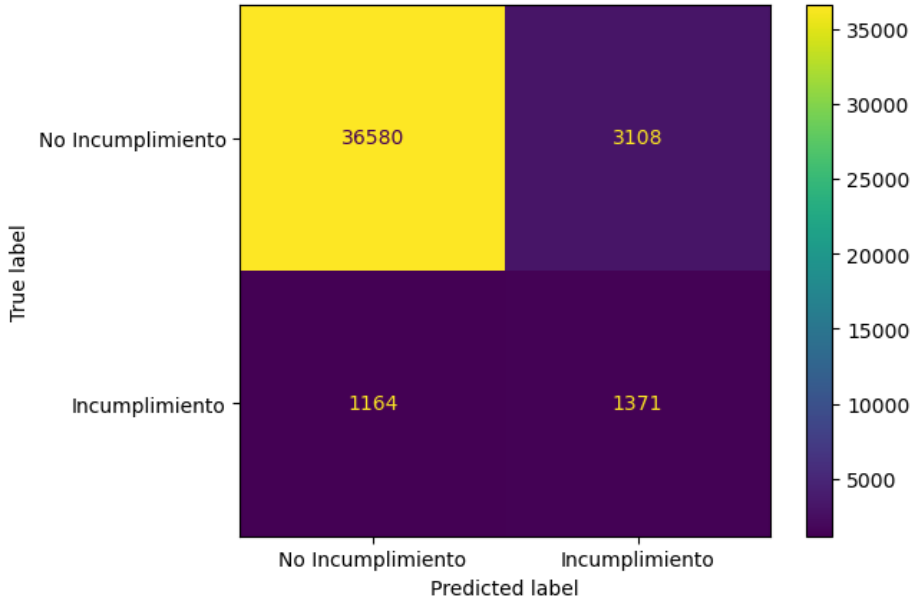
El modelo con el mejor resultado fue Gradient Boosting, y el segundo fue XGBoost. Se selecciona entonces el modelo Gradient Boosting para la primera iteración. Aquí se realiza una optimización de parámetros, dado que inicialmente se había construido con 150 estimadores, tasa de aprendizaje de 5% y una profundidad máxima de cinco niveles para el árbol. Se aplica un método de Rejilla de búsqueda (GridSearch) para seleccionar los parámetros óptimos basado en la métrica F1-score calculada.

Después de la ejecución de la rejilla se seleccionan los mejores parámetros, los cuales son: tasa de aprendizaje 1%, profundidad máxima de cinco niveles y 200 árboles. Se crea el modelo con los mejores parámetros obteniendo los siguientes resultados:

- AUC = 85.5%
- Exactitud = 89.9%
- Precisión=30.61%
- Recall=54.08%
- F1Score=39.09%

La matriz de confusión Figura 9 Matriz de confusión Gradient Boosting, muestra que de los 2535 registros que presentaban incumplimiento, el modelo calculó que 1164 no presentaban

Figura 9 Matriz de confusión Gradient Boosting



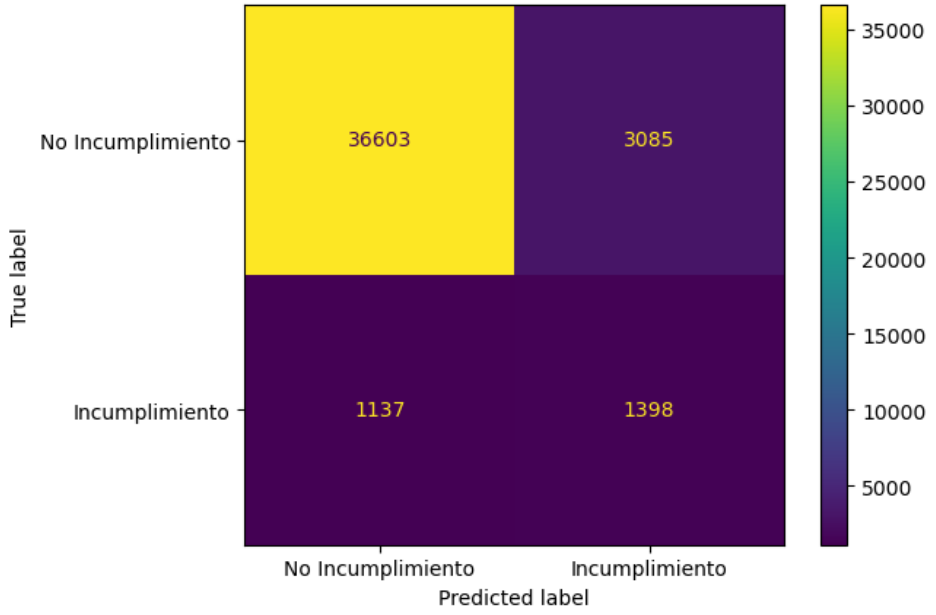
Para la segunda iteración se usa XGBoost, inicialmente se utilizaron los siguientes parámetros: 150 estimadores, profundidad máxima de 5 árboles y una taza de aprendizaje del 5%.

Se realiza búsqueda de los mejores parámetros con el método de rejilla de búsqueda (GridSerch), y arrojó que el único parámetro que se debía cambiar con respecto al usado inicialmente es la taza de aprendizaje, con un valor de 10%. Al entrenar el modelo con estos parámetros, se obtuvieron las siguientes métricas:

- AUC = 85.7%
- Exactitud = 90%
- Precisión=30.61%
- Recall=55.15%
- F1Score=39.84%

En la matriz de confusión Figura 10 Matriz de confusión XGBoost. Se encuentra que, de los 2535 registros de incumplimiento, el modelo calculó que 1137 no presentarían incumplimiento.

Figura 10 Matriz de confusión XGBoost.



7. Conclusiones

Los modelos desarrollados (Gradient Boosting y XGBoost) exhibieron métricas de precisión cercanas al 90%. Esta cifra sugiere que, en la mayoría de los casos, los modelos tienen la capacidad de prever con exactitud si un individuo está en riesgo de incumplimiento financiero. No obstante, la precisión, por sí sola, no proporciona una evaluación completa, especialmente en el contexto bancario, donde los falsos negativos, es decir, la no detección de casos de incumplimiento, resultan costosos y potencialmente problemáticos.

La complejidad añadida a esta investigación provino del desbalance marcado en el conjunto de datos, donde solo el 5% de los casos presentaban incumplimiento financiero. Este desequilibrio planteó desafíos durante el entrenamiento de los modelos, ya que, en su búsqueda de optimizar la precisión global, tendían a pasar por alto la clase minoritaria, que, en nuestro caso, tenía una importancia estratégica considerable. Estrategias específicas para abordar el desbalance, como la ponderación de clases o el remuestreo, se presentan como imperativas para mitigar este problema y garantizar que el modelo pueda captar adecuadamente patrones en ambas clases, incluso cuando una esté significativamente subrepresentada.

El análisis de la métrica de recall revela que, de las personas que eventualmente incumplirían, el modelo logró detectar aproximadamente el 55%. Esto destaca la necesidad de considerar métricas más allá de la precisión, especialmente cuando la detección de la clase minoritaria es de alta relevancia estratégica. Además, la capacidad del modelo para prever con precisión casos negativos, donde los individuos no incurren en incumplimiento financiero, se erige como un punto fuerte. La especificidad del modelo, evidenciada por la alta precisión en la clasificación de casos negativos, proporciona a las instituciones financieras una herramienta valiosa para discernir entre aquellos clientes que presentan bajos riesgos y aquellos que pueden requerir una atención más detenida.

Este trabajo ha destacado la complejidad inherente al desarrollo de modelos predictivos en el contexto de riesgo crediticio, especialmente cuando se enfrentan a conjuntos de datos desbalanceados. La necesidad de considerar métricas más allá de la precisión se vuelve evidente,

y la capacidad del modelo para prever casos negativos ofrece una perspectiva valiosa para la toma de decisiones en la gestión crediticia. Si bien la precisión es un indicador importante, la atención a falsos negativos y estrategias específicas para abordar el desbalance son cruciales para garantizar la efectividad y la aplicabilidad práctica de estos modelos en el sector financiero

8. Recomendaciones

La recomendación principal para futuras investigaciones es el abordaje del desbalance de clases, para hacer que el modelo dé un gran peso a la clase minoritaria, y más aún cuando esta clase es tan importante para el negocio como en el caso de este trabajo, todo esto sin hacer que el modelo pierda la capacidad de clasificar bien la clase mayoritaria.

Se recomienda hacer una búsqueda exhaustiva de los parámetros adecuados para mejorar el rendimiento del modelo, muchas veces las limitaciones tecnológicas no permiten hacer una búsqueda más extensa de parámetros, pero con la utilización de cómputo en la nube podría ser posible hacer búsquedas más largas y pesadas.

9. Repositorio del trabajo

El repositorio donde se desarrollan los artefactos de código necesarios para este trabajo puede encontrarse en la URL https://github.com/00Fede/givemesomecredit.

Referencias

Breunig, M. M.-P. (Junio de 2000). LOF: Identifying Density-Based Local Outliers. *SIGMOD Rec., 29*(2), 93-104. doi:10.1145/335191.335388

Chawla, N. V., Hall, K. W., Kegelmeyer, L. O., & Philip, W. (Enero de 2002). SMOTE: Synthetic Minority over-Sampling Technique. *J. Artif. Int. Res., 16*(1), 321–357.

Credit Fusion, W. C. (2011). *Give Me Some Credit*. Obtenido de Kaggle: https://kaggle.com/competitions/GiveMeSomeCredit

Dodge, Y. (2003). *The Oxford Dictionary of Statistical Terms.* Oxford University Press.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters, 27*(8), 861-874. doi:10.1016/j.patrec.2005.10.010

Han, J. a. (2011). *Data Mining: Concepts and Techniques* (Tercera ed.). Elsevier Science. Obtenido de https://books.google.com.co/books?id=pQws07tdpjoC&pg=PA111&redir_esc=y#v=onepage&q&f=false

Japkowicz, N. (2000). The Class Imbalance Problem: Significance and Strategies. *In Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI'2000): Special Track on Inductive Learning.* Las Vegas, Nevada.

Kubat, M., Holte, R., & Matwin, S. (1998). Machine Learning for the Detection of Oil Spills in Satellite Radar Images. *Machine Learning, 30*, 195-215.

Lewis, D., & Catlett, J. (1994). Heterogeneous Uncertainity Sampling for Supervised Learning. *Proceedings of the Eleventh International Conference of Machine Learning* (págs. 148-156). San Francisco, CA.: Morgan Kaufmann.

Ling, C., & Li, C. (1998). Data Mining for Direct Marketing Problems and Solutions. *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98).* New York, NY: AAAI Press.

Liu, S., Wang, R., & Han, Y. (2021). Research on Personal Credit Evaluation Based on Machine Learning Algorithm. *2021 6th International Symposium on Computer and Information Processing Technology (ISCIPT)* (págs. 48-52). Changsha, China: IEEE. doi:10.1109/ISCIPT53667.2021.00016

Provost, F., & Fawcett, T. (2001). Robust Classification for Imprecise Environments. *Machine Learning, 42*(3), 203-231.

Shearer, C. (2000). The CRISP-DM Model: The New Blueprint for Data Mining. *Journal of Data Warehousing*, *5*, 13-22.