Modelos de aprendizaje supervisado para la clasificación de riesgo crediticio en la entidad financiera Home Credit

Laura Cristina Caro Puerta
Lady Jhoana Rodas Zuluaga

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

Tutor
Efraín Alberto Oviedo Carrascal, Magíster (MSc) en TICs

Universidad de Antioquia
Facultad de Ingeniería
Especialización en Analítica y Ciencia de Datos
Medellín, Colombia
2022

Cita	(Caro Puerta & Rodas Zuluaga, 2022)
Referencia **Estilo APA 7 (2020)**	Caro Puerta, L., & Rodas Zuluaga, L. (2022). *Modelos de aprendizaje supervisado para la clasificación de riesgo crediticio en la entidad financiera Home Credit* [Monografía para Especialización]. Universidad de Antioquia, Medellín, Colombia.

Especialización en Analítica y Ciencia de Datos, Cohorte III.

Tabla de contenido

Lista de Tablas

Lista de Figuras

1. Resumen ejecutivo

Home Credit Default Risk es un reto planteado por Kaggle que busca un modelo de aprendizaje automático que permita hacer predicciones del cumplimiento de pago de sus clientes al ser ésta una entidad financiera dedicada a la entrega de créditos. Se cuenta con 10 datasets y el modelo se selecciona de acuerdo con el mejor resultado de la curva ROC encontrada.

Para el desarrollo del proyecto se decide trabajar con tres de los ocho datasets comenzando un preprocesamiento por separado, buscando nulos, correlaciones y agregaciones, para luego unirlas y obtener un solo dataset con el cual desarrollar el proyecto.

Luego, se realiza un análisis exploratorio de las diferentes variables que representan el dataset y se realiza un preprocesamiento más profundo donde se buscan nulos, correlaciones, *outliers* y se realiza la transformación de las variables categóricas. En este punto, se nota un significativo desbalance de las clases de la variable objetivo, donde más del 80% de los datos se encuentran en la clase 0 (el cliente cumple con los pagos). Debido al desbalance existente en la variable *TARGET*, se realizan dos escenarios aplicando los modelos con la técnica de balanceo *oversampling* y luego con la técnica de balanceo *undersampling*.

Por el tipo de problema trabajado, se decide trabajar con los modelos de Regresión Logística, *Random Forest* y *Gradient Boosting*. Los resultados obtenidos con la técnica oversampling, para el modelo Regresión Logística fue de alrededor del 70% con una curva ROC del 77%, mientras que con los modelos *Random Forest* y *Gradient Boosting*, se obtiene un resultado por encima del 90% para todas las métricas con ambos modelos y ambas curvas ROC con un 98%. Con la técnica de *undersampling*, los tres modelos arrojan resultados muy similares, donde las métricas muestran en promedio un 68% de acierto y las curvas ROC entre 75% y 76%.

Se observa que ambas técnicas generan resultados acordes, incluso aplicando la metodología *Stratified K Fold*, con la única diferencia de que la técnica de *undersampling* baja el resultado obtenido en la curva ROC de los modelos *Random Forest* y *Gradient Boosting*.

Repositorio de GitHub con notebook: https://github.com/LadyRodas/HomeCredit

2. Descripción del problema

Home Credit busca un modelo de aprendizaje automático para hacer predicciones del cumplimiento de pago de sus clientes, así se asegurará de que no se rechace a los clientes capaces de reembolsar y de que los préstamos se otorguen con un capital, vencimiento y calendario de reembolso que facilitará el éxito en sus clientes.

En este sentido, con el modelo se pretende predecir el nivel de cumplimiento de los clientes de la entidad financiera Home Credit, definiendo el *TARGET* en una variable binaria que representa:

1 - cliente con dificultades de pago: tuvo un retraso en el pago de más de *X* días en al menos una de las primeras *Y* cuotas del préstamo en nuestra muestra.

0 - cliente cumple con los pagos.

2.1. Problema del negocio

Home Credit es una entidad financiera enfocada en créditos responsables con un modelo de negocio de préstamos en el punto de venta ofrecidos principalmente a personas con poco o ningún historial crediticio. Esta entidad se caracteriza por ofrecer servicios simples, fáciles y rápidos a esta población no bancarizada que busca esa primera oportunidad de obtener un préstamo que inicie su vida crediticia de manera positiva (*Home Credit Default Risk*, 2018). Es allí, donde surge la necesidad de entrar a un mercado desatendido brindando experiencias que deben ser seguras tanto para el cliente como para la entidad financiera. Así, Home Credit busca la manera de asegurarse de que los clientes capaces de pagar no sean rechazados y que los préstamos se entreguen de manera responsable y oportuna.

2.2. Aproximación desde la analítica de datos

Home Credit cuenta con una variedad de datos centralizados, incluida la información de otras compañías y de diferentes transacciones a través del tiempo, donde usando varias técnicas y métodos estadísticos, al igual que modelos de aprendizaje automático, se puede llegar a predecir la capacidad de pago de sus clientes al explotar todo el potencial que tengan estos datos y que se asegure así la mitigación y gestión del riesgo.

2.3. Origen de los datos

Los datos de Home Credit Default Risk provienen de la plataforma Kaggle, (*Home Credit Default Risk,* 2018), haciendo referencia a los datos de clientes de la entidad Home Credit, los datasets no presentan fechas específicas de los préstamos o de la recolección de los datos pero se asume que es información recolectada por la entidad donde la más reciente es cercana al año 2018. Este desafío cuenta con 10 datasets, de los cuales uno hace referencia a la descripción de todas las variables "*HomeCredit_columns_description*" y otro equivale a una muestra de los resultados "*sample_submission*"; los ocho datasets restantes corresponden a la información de los clientes de Home Credit y las respectivas condiciones de los préstamos adquiridos históricamente.

Los datos traen información tal como saldos de los préstamos anteriores por diferentes metodologías: POS (Puntos de venta), tarjetas de crédito, crédito de vivienda y efectivo; historial de repagos, aplicaciones a créditos y datos básicos de tipo personal como por ejemplo las personas con las que convive, lugar de vivienda, barrio, activos a nombre de quien solicita el crédito, ingresos familiares, entre otros. Dicha información se encuentra como tipo entero, flotante y objeto. Además, algunas variables fueron transformadas por la entidad como las zonas, documentos entregados por el cliente, tipo de organización, entre otros.

2.4. Métricas de desempeño

2.4.1. *Métricas del modelo*

Las métricas de evaluación usadas en los modelos son: *accuracy, precision, recall, f1, matthews, balance accuracy* y curva ROC; las cuales se describen detalladamente en el apartado 4.4.

Se espera un acierto de la curva ROC a partir de un porcentaje del 70% para lograr cumplir satisfactoriamente las métricas de negocio, las cuales se detallan a continuación.

2.4.2. *Métricas del negocio*

Considerando que el desafío Home Credit Default Risk de Kaggle no plantea métricas específicas de negocio, a continuación, se plantean algunas métricas viables para evaluar el desempeño a nivel de negocio con el impacto de los resultados del modelo para entidades del sector financiero.

- Cumplimiento de cartera: porcentaje de recaudo de cartera exitoso, evaluando el cumplimiento de los clientes en el pago de sus obligaciones financieras, (Calderón Bandera, n.d.).
- Riesgo crediticio: porcentaje de riesgo que asume la entidad financiera al sufrir una pérdida como consecuencia del incumplimiento en los pagos de los clientes, (*Riesgo Crediticio*, n.d.).
- Costo de oportunidad: ganancia económica no percibida por la entidad al rechazarle un crédito a un cliente con altas posibilidades de cumplir con sus obligaciones financieras, (*Costo De Oportunidad*, 2022).

Al cumplir con la métrica del modelo, se espera afectar positiva y directamente las tres métricas de negocio propuestas, donde, en primera instancia, una correcta predicción y clasificación de los clientes que desean solicitar un prestamos en Home Credit conlleva a que la métrica del cumplimiento de cartera incremente y se mantenga al garantizar que el cliente paga oportunamente, el riesgo crediticio disminuye al otorgar crédito a clientes confiables donde más adelante la entidad no deberá asumir una pérdida por mora y el costo de oportunidad no se presenta con alta frecuencia al no rechazar clientes que tienen la posibilidad de cumplir con sus obligaciones financieras.

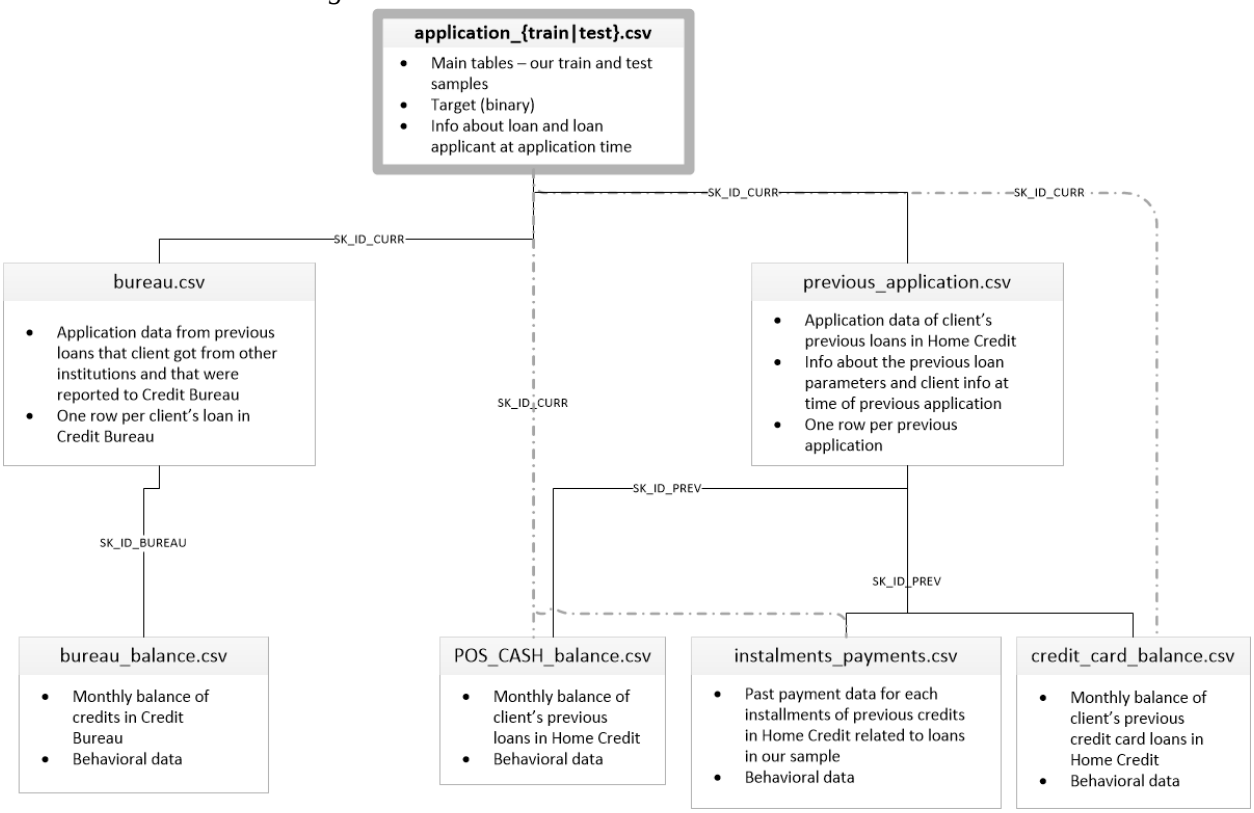
3. Datos

3.1. Datos originales

Los datasets entregados por la entidad financiera Home Credit se encuentran resumidos en la Figura 1, presentando además de una breve descripción, la relación que se tiene entre las tablas por una variable en común. En esta figura se observa cómo las tres tablas en la parte superior logran abarcar en su mayoría con la información contenida en las tablas de la parte inferior, conteniendo estas últimas información más detallada de los créditos obtenidos por los clientes en otras entidades o en la misma entidad Home Credit.

Figura 1

Articulación de datasets originales



Nota: (*Home Credit Default Risk*, 2018)

A continuación, se muestra el detalle de cada dataset con una breve vista de las primeras columnas que contiene cada una y la cantidad de filas y columnas que presenta.

3.1.1. *home_credit_description.csv*

Este archivo contiene descripciones de las columnas en los distintos archivos de datos, permitiendo entender el contexto de todas las variables en el problema.

Tamaño: 37 KB

Forma: (219, 5)

3.1.2. *application_train.csv*

Esta es la tabla principal para train (con *target*). Datos estáticos para todas las aplicaciones. Una fila representa un préstamo en nuestra muestra de datos.

Tamaño: 162.240 KB

Forma: (307511, 122)

Tabla 1

Algunas variables del dataset application_train

Variable	Description	Descripción
SK_ID_CURR	ID of loan in our sample.	ID de préstamo en nuestra muestra.
TARGET	Target variable (1 - client with payment difficulties: he/she had late payment more than X days on at least one of the first Y installments of the loan in our sample, 0 - all other cases).	Variable objetivo (1 - cliente con dificultades de pago: tuvo un retraso en el pago de más de *X* días en al menos una de las primeras *Y* cuotas del préstamo en nuestra muestra, 0 - todos los demás casos).
NAME_CONTRACT_TYPE	Identification if loan is cash or revolving.	Identificación si el préstamo es en efectivo o rotatorio.
CODE_GENDER	Gender of the client.	Género del cliente.

Nota: Elaboración propia.

3.1.3. *application_test.csv*

Esta es la tabla principal para test (sin *TARGET*). Datos estáticos para todas las aplicaciones. Esta tabla tiene las mismas columnas del dataset *application_train* con la diferencia de no traer la columna *TARGET*. Una fila representa un préstamo en nuestra muestra de datos.

Tamaño: 25.945 KB

Forma: (48744, 121)

3.1.4. *bureau.csv*

Todos los créditos anteriores del cliente proporcionados por otras instituciones financieras que se informaron al *Credit Bureau* (para los clientes que tienen un préstamo en nuestra muestra).

Para cada préstamo de nuestra muestra, hay tantas filas como créditos tenía el cliente en el *Credit Bureau* antes de la fecha de solicitud.

Tamaño: 166.032 KB

Forma: (1716428, 17)

Tabla 2

Algunas variables del dataset bureau

Variable	Description	Descripción
SK_ID_CURR	ID of loan in our sample - one loan in our sample can have 0,1,2 or more related previous credits in Credit Bureau.	Identificación del préstamo en nuestra muestra: un préstamo en nuestra muestra puede tener 0, 1, 2 o más créditos anteriores relacionados en *Credit Bureau*.
SK_BUREAU_ID	Recoded ID of previous Credit Bureau credit related to our loan (unique coding for each loan application).	Identificación recodificada del crédito anterior de *Credit Bureau* relacionado con nuestro préstamo (codificación única para cada solicitud de préstamo).
CREDIT_ACTIVE	Status of the Credit Bureau (CB) reported credits.	Estado de los créditos reportados por *Credit Bureau* (CB).
CREDIT_CURRENCY	Recoded currency of the Credit Bureau credit.	Moneda recodificada del crédito de *Credit Bureau*.

Nota: Elaboración propia.

3.1.5. *bureau_balance.csv*

Saldos mensuales de créditos anteriores en *Credit Bureau*. Esta tabla tiene una fila para cada mes del historial de cada crédito anterior informado a la Oficina de Crédito, es decir, la tabla tiene (número de préstamos en la muestra de créditos anteriores relativos de meses donde tenemos algo de historial observable para los créditos anteriores) filas.

Tamaño: 366.790 KB

Forma: (27299925, 3)

Tabla 3

Algunas variables del dataset bureau_balance

Variable	Description	Descripción
SK_BUREAU_ID	Recoded ID of Credit Bureau credit (unique coding for each application) - use this to join to credit_bureau table.	ID recodificado del crédito de *Credit Bureau* (codificación única para cada aplicación): utilícelo para unirse a la tabla credit_bureau.
MONTHS_BALANCE	Month of balance relative to application date (-1 means the freshest balance date).	Mes de saldo relativo a la fecha de solicitud (-1 significa la fecha de saldo más reciente).

STATUS	Status of Credit Bureau loan during the month (active, closed, DPD0-30, [C means closed, X means status unknown, 0 means no DPD, 1 means maximal did during month between 1-30, 2 means DPD 31-60, 5 means DPD 120+ or sold or written off]).	Estado del préstamo de *Credit Bureau* durante el mes (activo, cerrado, DPD0-30, [C significa cerrado, X significa estado desconocido, 0 significa que no hay DPD, 1 significa que el máximo lo hizo durante el mes entre 1-30, 2 significa DPD 31-60 , 5 significa DPD 120+ o vendido o cancelado]).

Nota: Elaboración propia.

3.1.6. *previous_application.csv*

Todas las solicitudes anteriores de préstamos de Home Credit de clientes que tienen préstamos en la muestra. Hay una fila para cada solicitud anterior relacionada con préstamos en nuestra muestra de datos.

Tamaño: 395.482 KB

Forma: (1670214, 37)

Tabla 4

Algunas variables del dataset previous_application

Variable	Description	Descripción
SK_ID_PREV	ID of previous credit in Home Credit related to loan in our sample. (One loan in our sample can have 0,1,2 or more previous loan applications in Home Credit, previous application could, but not necessarily have to lead to credit).	Cédula de crédito anterior en *Home Credit* relacionado con préstamo en nuestra muestra. (Un préstamo de nuestra muestra puede tener 0, 1, 2 o más préstamos anteriores en *Home Credit*).
SK_ID_CURR	ID of loan in our sample.	Cédula de préstamo en nuestra muestra.
NAME_CONTRACT_TYPE	Contract product type (Cash loan, consumer loan [POS], …) of the previous application.	Tipo de producto del contrato (préstamo al contado, préstamo al consumo [POS], ...) de la solicitud anterior.
AMT_CREDIT	Final credit amount on the previous application. This differs from amt_application in a way that the amt_application is the amount for which the client initially applied for, but during our approval process he could have received a different amount - amt_credit.	Monto del crédito final en la solicitud anterior. Esto difiere de amt_application que amt_application es la cantidad que el cliente solicitó inicialmente, pero durante nuestro proceso de aprobación podría haber recibido una cantidad diferente: amt_credit.
WEEKDAY_APPR_PROCESS_START	On which day of the week did the client apply for previous application.	¿En qué día de la semana solicitó el cliente la solicitud previa?

Nota: Elaboración propia.

3.1.7. *pos_cash_balance.csv*

Instantáneas de saldos mensuales de POS (Punto de Venta) anteriores y préstamos en efectivo que el solicitante tenía con Home Credit. Esta tabla tiene una fila para cada mes del historial de cada crédito anterior en Home Credit (crédito al consumo y préstamos en efectivo) relacionado con los préstamos de nuestra muestra, es decir, la tabla tiene préstamos en la muestra de créditos anteriores relativos de meses en el que tenemos algo de historial observable para los créditos anteriores) filas.

Tamaño: 383.500 KB

Forma: (10001358, 8)

Tabla 5

Algunas variables del dataset pos_cash_balance

Variable	Description	Descripción
SK_ID_PREV	ID of previous credit in Home Credit related to loan in our sample. (One loan in our sample can have 0,1,2 or more previous loans in Home Credit).	Cédula de crédito anterior en *Home Credit* relacionado con préstamo en nuestra muestra. (Un préstamo de nuestra muestra puede tener 0, 1, 2 o más préstamos anteriores en *Home Credit*).
SK_ID_CURR	ID of loan in our sample.	Cédula de préstamo en nuestra muestra.
MONTHS_BALANCE	Month of balance relative to application date (-1 means the information to the freshest monthly snapshot, 0 means the information at application - often it will be the same as -1 as many banks are not updating the information to Credit Bureau regularly).	Mes del saldo en relación con la fecha de la solicitud (-1 significa la información de la instantánea mensual más reciente, 0 significa la información en la solicitud; a menudo será igual a -1 ya que muchos bancos no actualizan la información a la *Credit Bureau* con regularidad).
CNT_INSTALMENT	Term of previous credit (can change over time).	Plazo del crédito anterior (puede cambiar con el tiempo).

Nota: Elaboración propia.

3.1.8. *installments_payments.csv*

Historial de reembolso de los créditos previamente desembolsados en *Home Credit* relacionados con los préstamos de nuestra muestra. Hay una fila por cada pago que se realizó más una fila por cada pago atrasado. Una fila equivale a un pago de una cuota o una cuota correspondiente a un pago de un crédito hipotecario anterior relacionado con préstamos de nuestra muestra.

Tamaño: 706.171 KB

Forma: (13605401, 8)

Tabla 6

Algunas variables del dataset installments_payments

Variable	Description	Descripción
SK_ID_PREV	ID of previous credit in Home Credit related to loan in our sample. (One loan in our sample can have 0,1,2 or more previous loans in Home Credit).	Cédula de crédito anterior en *Home Credit* relacionado con préstamo en nuestra muestra. (Un préstamo de nuestra muestra puede tener 0, 1, 2 o más préstamos anteriores en *Home Credit*).
SK_ID_CURR	ID of loan in our sample.	Cédula de préstamo en nuestra muestra.
NUM_INSTALMENT_VERSION	Version of installment calendar (0 is for credit card) of previous credit. Change of installment version from month to month signifies that some parameter of payment calendar has changed.	Versión del calendario de cuotas (0 es para tarjeta de crédito) del crédito anterior. El cambio de la versión de pago de un mes a otro significa que algún parámetro del calendario de pagos ha cambiado.
NUM_INSTALMENT_NUMBER	On which installment we observe payment.	En qué plazo observamos el pago.

Nota: Elaboración propia.

3.1.9. *credit_card_balance.csv*

Instantáneas del saldo mensual de las tarjetas de crédito anteriores que el solicitante tiene con *Home Credit*. Esta tabla tiene una fila para cada mes de historial de cada crédito anterior en *Home Credit* (crédito al consumo y préstamos en efectivo) relacionado con préstamos en nuestra muestra, es decir, la tabla tiene préstamos en la muestra de tarjetas de crédito anteriores relativas de meses en los que tenemos algo de historial observable para la tarjeta de crédito anterior) filas.

Tamaño: 414.632 KB

Forma: (3840312, 23)

Tabla 7

Algunas variables del dataset credit_card_balance

Variable	Description	Descripción
SK_ID_PREV	ID of previous credit in Home Credit related to loan in our sample. (One loan in our sample can have 0,1,2 or more previous loans in Home Credit).	Cédula de crédito anterior en *Home Credit* relacionado con préstamo en nuestra muestra. (Un préstamo de nuestra muestra puede tener 0, 1, 2 o más préstamos anteriores en *Home Credit*).
SK_ID_CURR	ID of loan in our sample.	Cédula de préstamo en nuestra

		muestra.
MONTHS_BALANCE	Month of balance relative to application date (-1 means the freshest balance date).	Mes de saldo relativo a la fecha de solicitud (-1 significa la fecha de saldo más reciente).
AMT_BALANCE	Balance during the month of previous credit.	Saldo durante el mes del crédito anterior.

Nota: Elaboración propia.

3.2. Acceso a los datos

Se cargan los datos de Home Credit Default Risk desde Kaggle con la siguiente metodología:

- Descargar el archivo de configuración de Kaggle.
    - Iniciar sesión en la cuenta de Kaggle.
    - Elegir 'Account' en el menú desplegable.
    - Desplazar hacia abajo hasta la sección 'API' y hacer clic en *'Create New API Token'*.
    - Descargar el archivo kaggle.json y guardarlo en la máquina local. Este es el archivo de configuración con sus credenciales que utilizará para acceder a los conjuntos de datos de Kaggle desde Colab.
- Cargar el archivo 'kaggle.json' de configuración en Colab cuando se solicite 'Elegir archivos'.

3.3. Datasets

Inicialmente, los datos entregados para este proyecto están constituidos por ocho tablas que contienen información de los clientes de *Home Credit* (históricos del comportamiento de una persona con un crédito en la misma entidad e históricos en otras entidades crediticias). Se comienza revisando cada variable de cada tabla para un total de 219 columnas, encontrando así, que cinco de las ocho tablas presentan información del comportamiento de los diferentes créditos con una periodicidad mensual, misma información que se encuentra en las tablas restantes por cada crédito (“*application train*”,” *bureau*” y “*previous application*”). Por ende, se decide trabajar con estas tres tablas que presentan la información por cada crédito, debido a que, se considera que las demás tablas, no aportarían información adicional al entrenamiento del modelo.

Ya identificadas las variables con las cuales se desea trabajar durante el desarrollo del proyecto, se realizan los siguientes pasos para cada una de tres tablas:

- ***Imputación de categorías XNA:*** se cambian los datos de las variables que contengan "XNA" por valores nulos.
- ***Agrupación de clases por variable:*** se encuentra que en algunas variables existen clases que pueden ser agrupadas al tener características similares que no afectarían la información que aportan al modelo.
- ***Búsqueda de nulos:*** se eliminan las variables con más del 20% de sus datos con valores nulos justificado por el hecho de que un porcentaje alto de datos que deban ser generados de manera sintética, no aportaría información relevante al modelo.
- ***Variables con el 95% de los datos en una sola clase:*** se realiza el conteo de las clases de las diferentes variables donde se eliminan aquellas con más del 95% de los datos de una misma clase.
- ***Correlación de variables:*** se verifica la correlación presente entre las variables de su misma tabla, tomando la decisión de eliminar aquellas que presenten una correlación superior al 70%.

Para las tablas "*bureau*" y *"previous application*", se realizan dos pasos más descritos a continuación:

- ***Agregación de variables:*** debido a que estas dos tablas muestran la información no solo por cada código de cliente, *"SK_ID_CURR"*, sino también por cada crédito que haya adquirido una persona en una entidad, se hace necesario realizar agregaciones que lleven a mostrar la información por cada cliente y no por cada préstamo (una sola fila por cada cliente y no una fila por cada préstamo repitiendo el cliente).
- ***Transformación de categóricas:*** en este punto se decide realizar la transformación de las variables categóricas por medio de la técnica *One Hot Encoder*, (Yadav, 2019) donde se crean nuevas variables por cada clase.

Una vez realizados los pasos anteriores, se procede a unir las tablas resultantes haciendo uso de la función *merge*, colocando como columna en común la variable *"SK_ID_CURR"*. Como resultado, se

obtiene una tabla compuesta por 247.408 filas y 82 columnas sin valores nulos; siendo esta es la tabla final con la cual se desarrollará el resto del proyecto.

3.4. Descriptiva

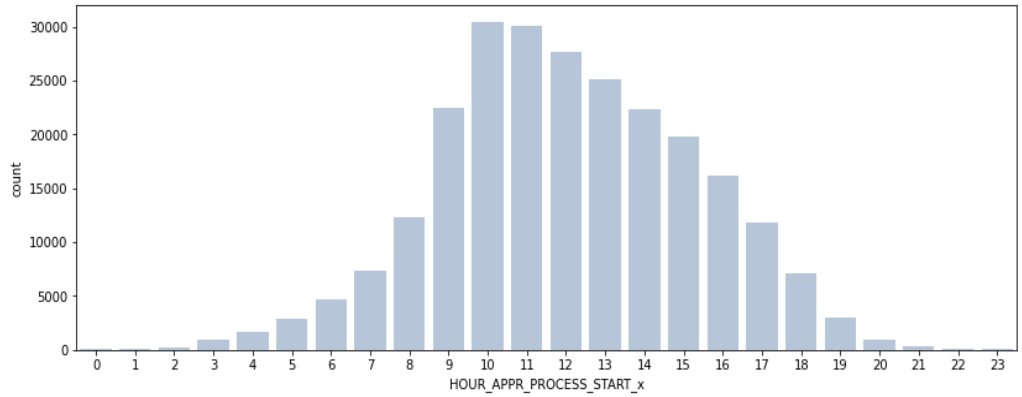
3.4.1. *Análisis univariable*

3.4.1.1. Variables numéricas.

Para realizar el análisis de las variables numéricas del DataFrame se realizan histogramas de frecuencia para las variables continuas y diagramas de barras para las variables discretas.

Figura 2

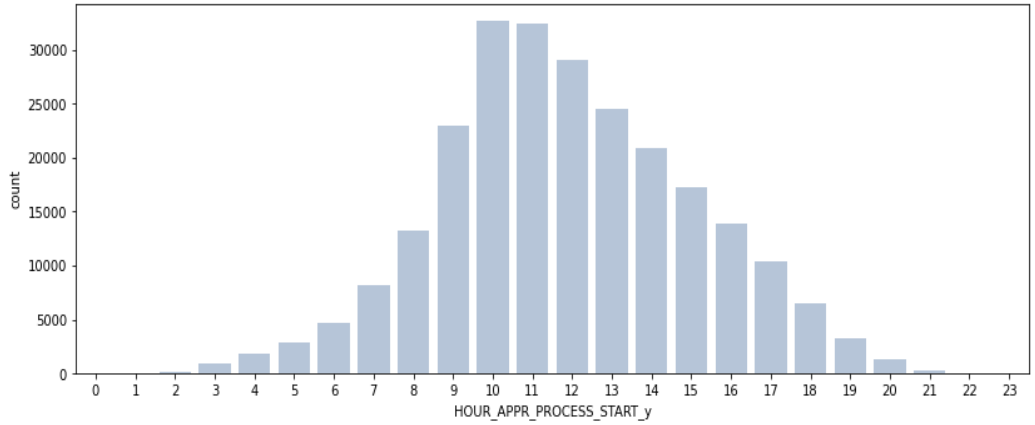
Diagrama de barras variable HOUR_APPR_PROCESS_START_x



Nota: Elaboración propia.

Figura 3

Diagrama de barras HOUR_APPR_PROCESS_START_y



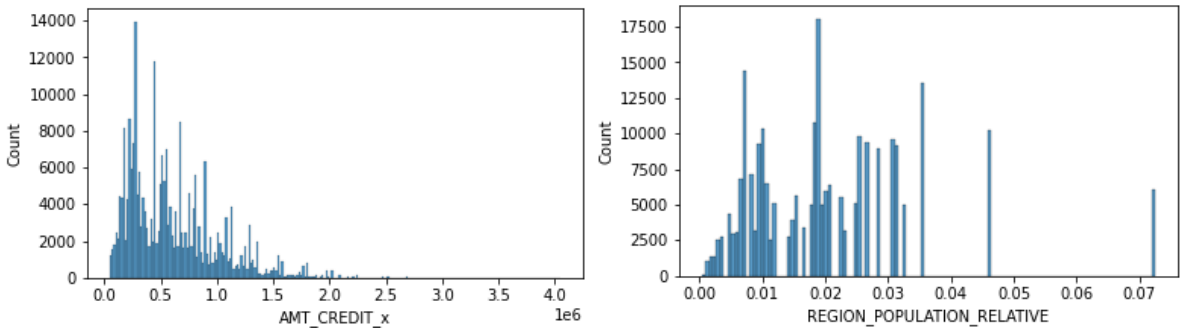
Nota: Elaboración propia.

La mayoría de las variables numéricas presentan un sesgo hacia una de las colas de la respectiva distribución, a excepción de las variables *"HOUR_APPR_PROCESS_START_x"*,

"HOUR_APPR_PROCESS_START_y" y "*EXT_SOURCE_3"*, las cuales muestran un comportamiento semejante a una distribución normal, evidenciando la mayor concentración de datos alrededor de la media.

Figura 4

Histograma variables AMT_CREDIT_x y REGION_POPULATION_RELATIVE

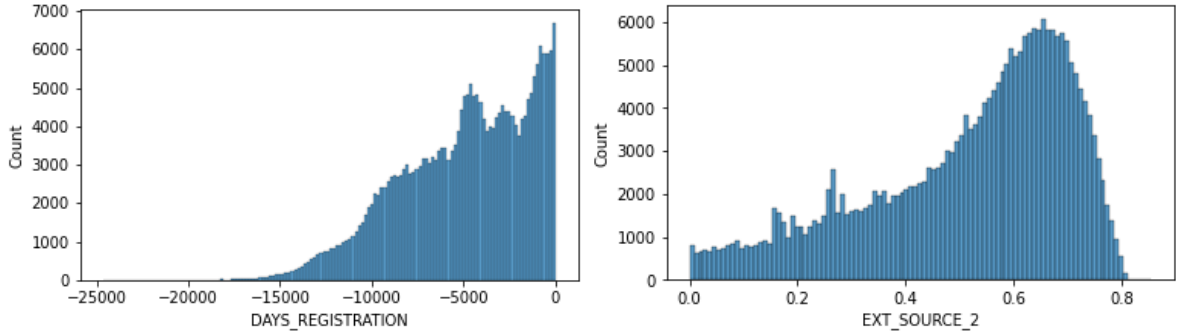


Nota: Elaboración propia.

Específicamente, las variables "*AMT_CREDIT_x"* (Monto del crédito del préstamo) y "*REGION_POPULATION_RELATIVE*" (Región relativa), presentan un sesgo hacia la derecha, donde los datos que presentan la mayor frecuencia son cercanos a valores mínimos de acuerdo con su respectivo rango.

Figura 5

Histograma variables DAYS_REGISTRATION y EXT_SOURCE_2

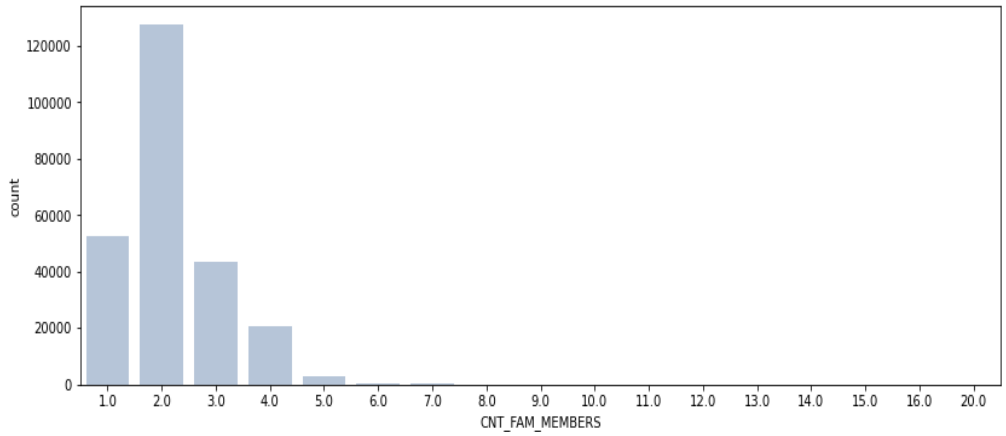


Nota: Elaboración propia.

Por otro lado, las variables *"DAYS_REGISTRATION*" (Días antes de la solicitud en que el cliente cambió su registro) y *"EXT_SOURCE_2"* (Puntuación de una fuente de datos externa), presentan un sesgo hacia la izquierda, atribuyendo la mayor frecuencia de datos a los valores máximos del rango.

Figura 6

Diagrama de barras de variable CNT_FAM_MEMBERS



Nota: Elaboración propia.

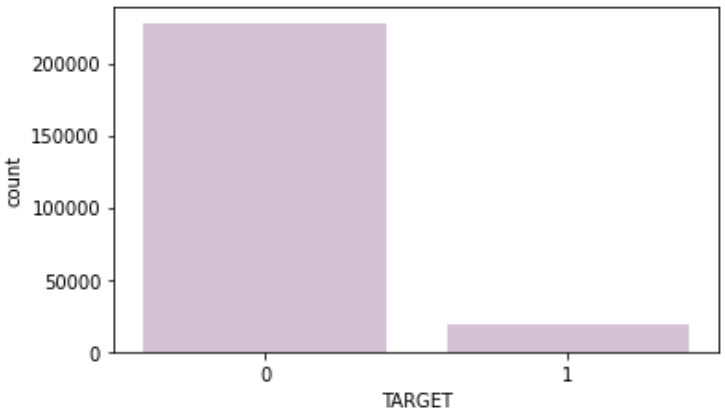
La cantidad de miembros de la familia oscilan entre 1 y 5. Además, se observa que las horas en las que el cliente solicitó el préstamo transcurren entre las 10 y 12 horas aproximadamente, con un comportamiento normal.

3.4.1.2. Variables categóricas.

Para analizar las variables categóricas del DataFrame se realizan diagramas de barras.

Figura 7

Diagrama de barras de variable TARGET

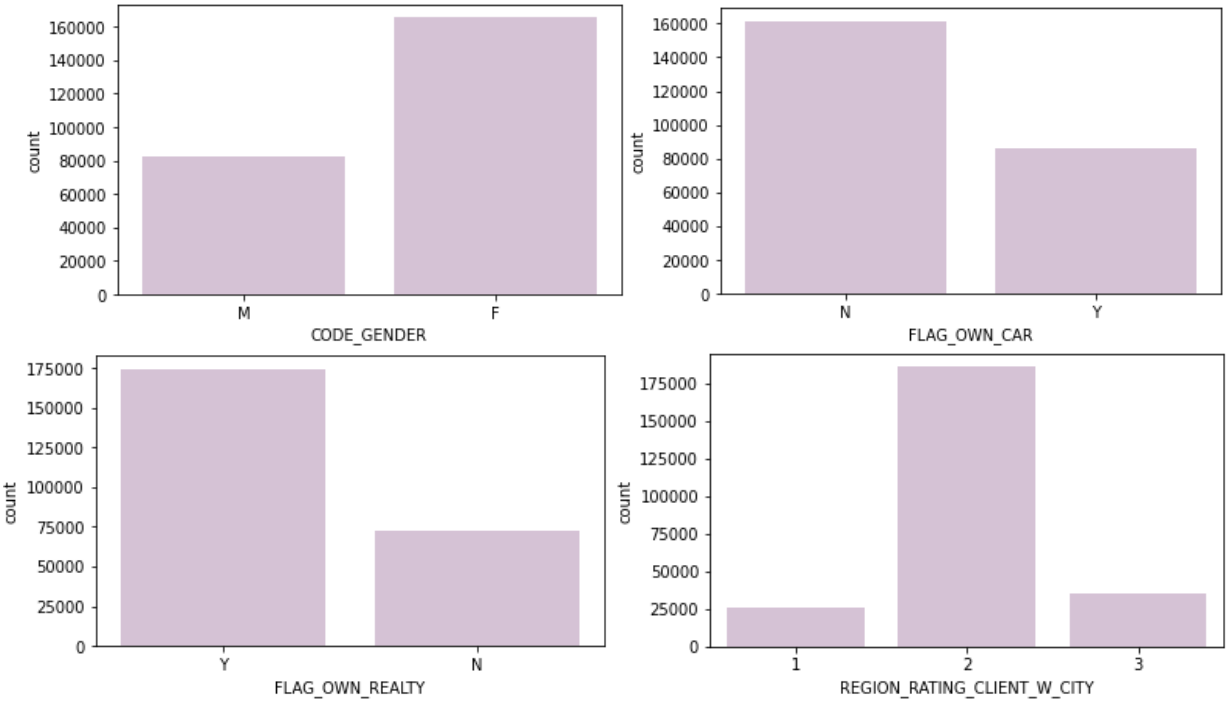


Nota: Elaboración propia.

En la variable *"TARGET"* se evidencia mayor predominancia de la categoría 0, la cual representa a los clientes que no se retrasan en sus pagos; por el contrario, los clientes que no cumplen las condiciones del préstamo (categoría 1) equivalen a menos de la décima parte de la categoría anterior.

Figura 8

Diagrama de barras de las variables CODE_GENDER, FLAG_OWN_CAR, FLAG_OWN_REALTY y REGION_RATING_CLIENT_W_CITY

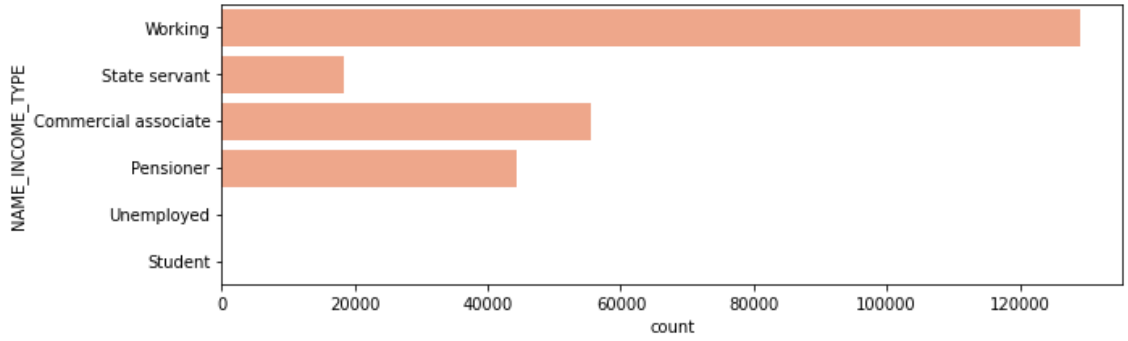


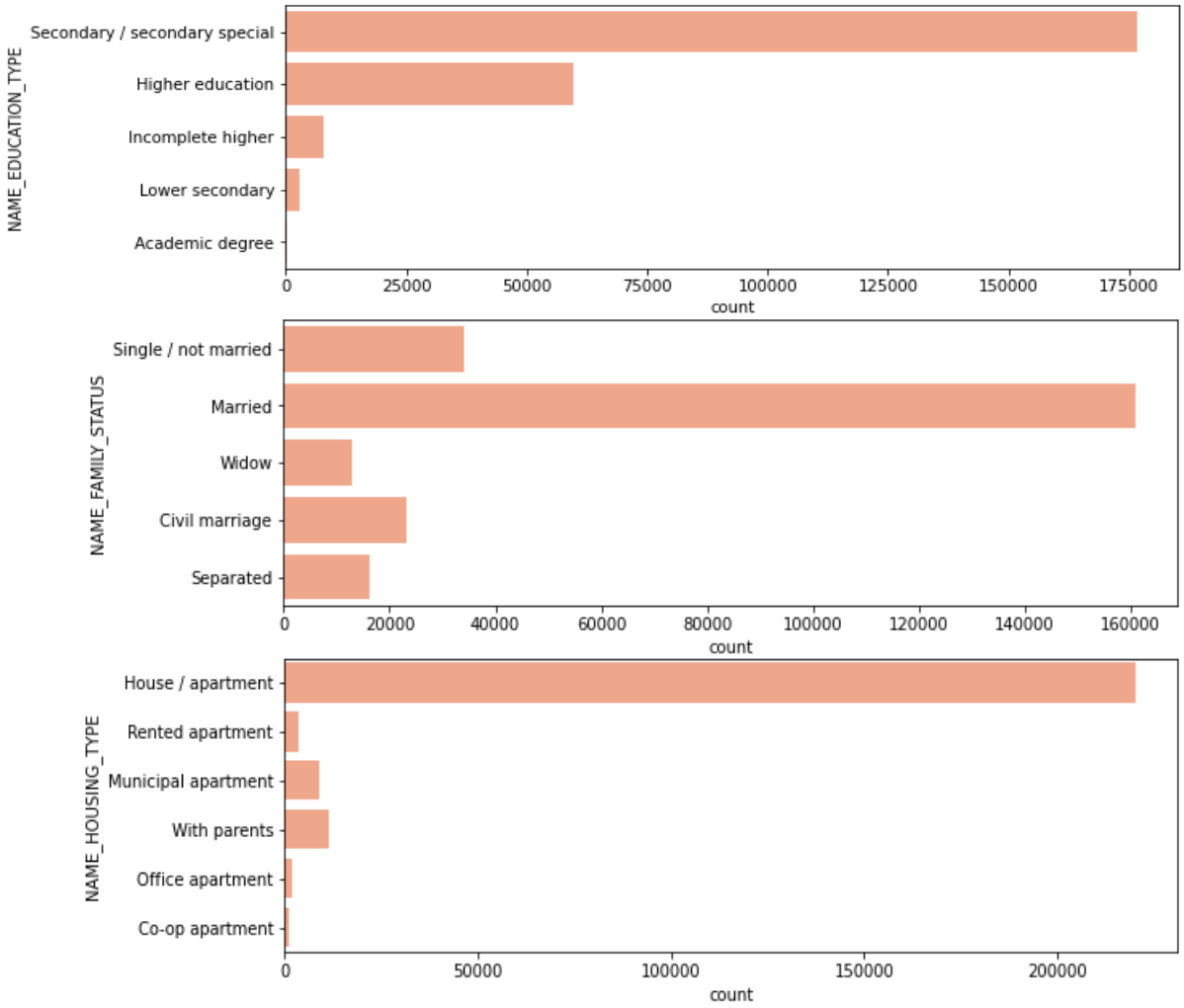
Nota: Elaboración propia.

También se observa que en el conjunto de datos, prevalecen los clientes con género femenino, sin vehículo propio y con al menos una propiedad raíz. Además, dentro de la calificación de la región donde vive el cliente (rango entre 1 y 3), prevalece la calificación 2, de modo general y también considerando la ciudad.

Figura 9

Diagrama de barras de las variables NAME_INCOME_TYPE, NAME_EDUCATION_TYPE, NAME_FAMILY_STATUS y NAME_HOUSING_TYPE



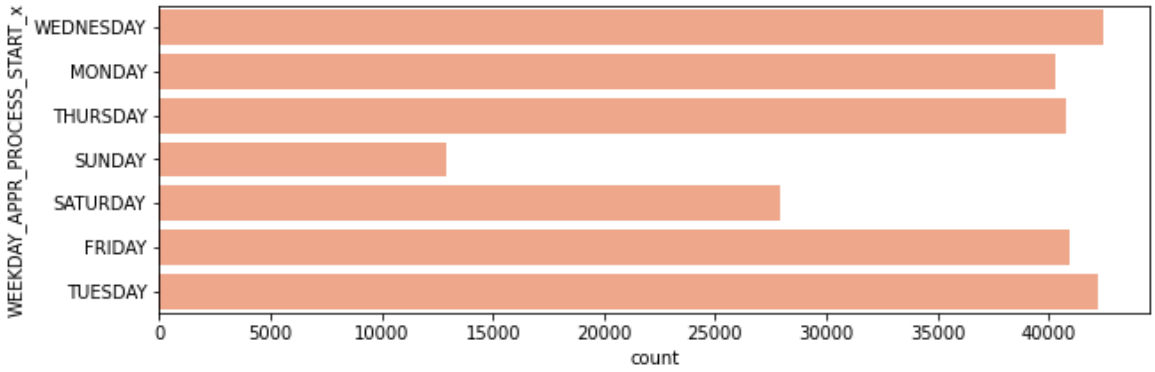


Nota: Elaboración propia.

Adicionalmente, se registra que la mayoría de los clientes se encuentran trabajando y su educación formal llega hasta nivel de secundaria. Por otro lado, la mayoría de los clientes están casados y viven en una casa o apartamento.

Figura 10

Diagrama de barras de la variable WEEKDAY_APPR_PROCESS_START_x

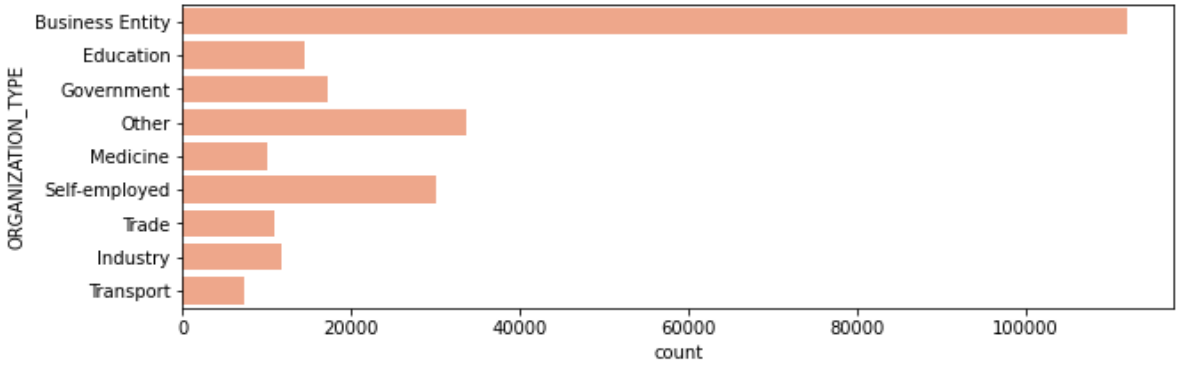


Nota: Elaboración propia.

En cuanto a los días de la semana donde se atienden las solicitudes de préstamo, los días hábiles presentan frecuencias muy similares, mientras que hay una reducción del flujo en la atención los sábados y domingos.

Figura 11

Diagrama de barras de la variable ORGANIZATION_TYPE



Nota: Elaboración propia.

También, se encuentra que la mayoría de las personas que acuden a los préstamos trabajan como personal de *Business Entity.*

3.4.2. *Análisis bivariable*

En el análisis bivariable se comparan todas las variables con respecto al *"TARGET"* o variable objetivo, para validar si existe un comportamiento notorio o una correlación entre cada par de variables.

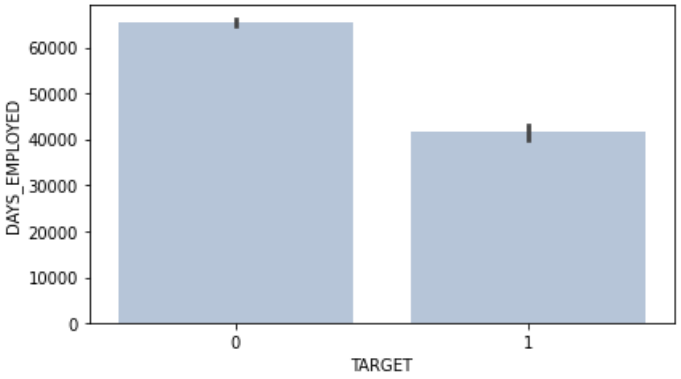
3.4.2.1. Variables numéricas.

Para analizar las variables numéricas con la variable *"TARGET"* se utilizan diagramas de barras.

La mayoría de las variables numéricas presentan una frecuencia muy semejante entre los clientes con clase 1 (cliente con dificultades de pago) y clase 0 (cliente cumplido).

Figura 12

Diagrama de barras de la variable DAYS_EMPLOYED

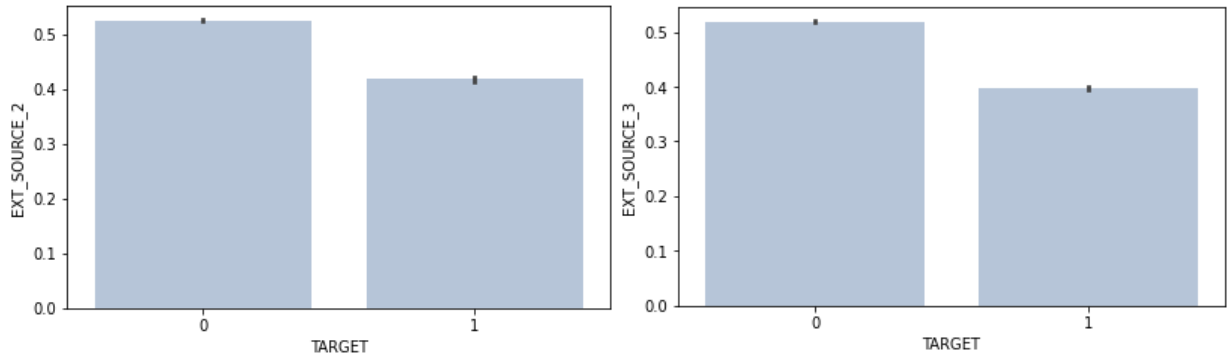


Nota: Elaboración propia.

Sin embargo, se observan diferencias significativas en algunas de las variables, como es el caso de *"DAYS_EMPLOYED"* que presenta una frecuencia máxima de 40.000 para clientes con dificultades de pago, mientras que para los clientes cumplidos asciende a más de 60.000; es decir que cuentan con mayor cantidad de días trabajando antes de presentar la solicitud, lo cual puede aludir a una mayor estabilidad económica para cumplir con las responsabilidades financieras.

Figura 13

Diagrama de barras de las variables EXT_SOURCE_2 con TARGET y EXT_SOURCE_3 con TARGET



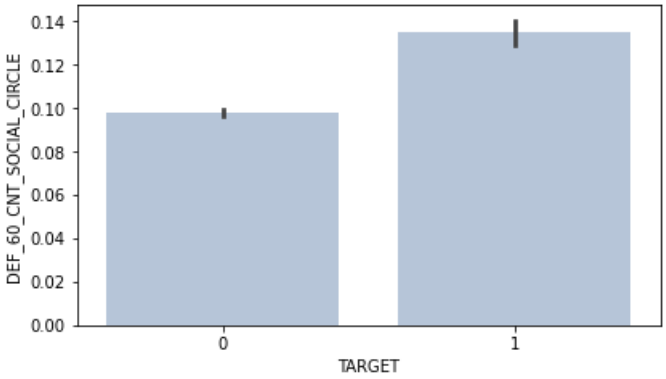
Nota: Elaboración propia.

Igualmente, se observa que los clientes cumplidos en sus pagos (clase 0) presentan una mayor puntuación en entidades crediticias externas, aproximadamente de 0.5 para las variables

"EXT_SOURCE_2" y *"EXT_SOURCE_3"*; mientras que los clientes con dificultades de pago presentan una puntuación de 0.4 para las mismas variables.

Figura 14

Diagrama de barras de la variable DEF_60_CNT_SOCIAL_CIRCLE con TARGET

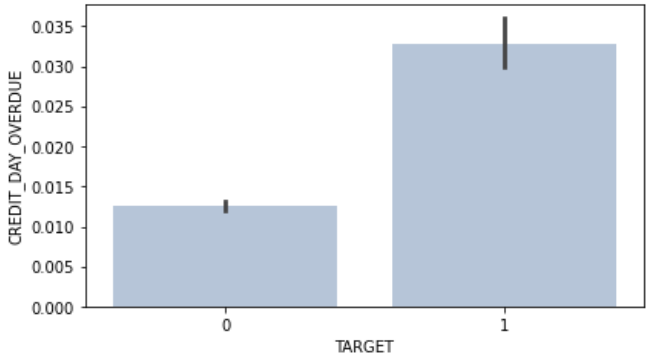


Nota: Elaboración propia.

Ahora, la variable *"DEF_60_CNT_SOCIAL_CIRCLE"* que muestra la cantidad de observaciones del entorno social del cliente con un incumplimiento de 60 días de atraso, evidencia que los clientes con cumplimiento en sus obligaciones presentan una menor cantidad de observaciones de incumplimiento en su círculo social, con respecto a los clientes con dificultades de pago.

Figura 15

Diagrama de barras de la variable CREDIT_DAY_OVERDUE con TARGET

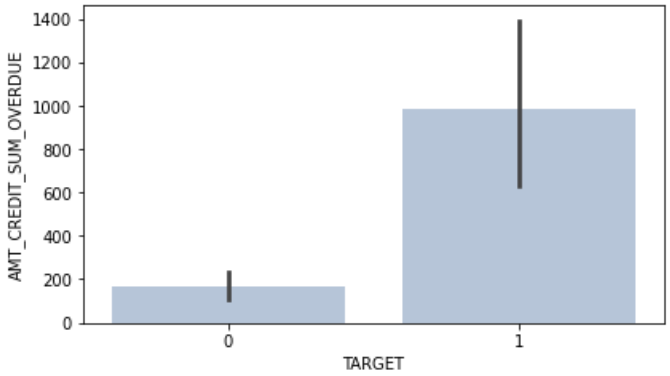


Nota: Elaboración propia.

También se observa que los clientes cumplidos (clase 0) presentan menos del 50% de días de atraso del crédito en *Bureau* en el momento de la solicitud, con relación a los clientes con dificultades de pago que presentan más días de atraso en *Bureau*.

Figura 16

Diagrama de barras de la variable AMT_CREDIT_SUM_OVERDUE con TARGET



Nota: Elaboración propia.

Y similarmente, se presenta el mismo comportamiento con la variable que representa el monto actual atrasado de los créditos *Bureau* para cada cliente; en este caso el monto atrasado en *Bureau* es cinco veces mayor para los clientes que tienen dificultades de pago en el crédito de Home Credit.

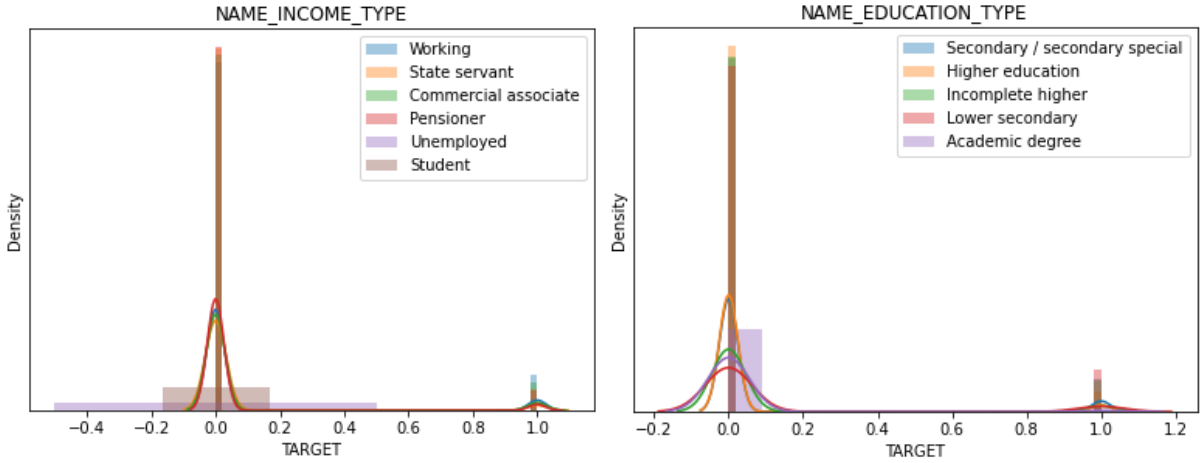
3.4.2.1. Variables categóricas.

Para el análisis de las variables categóricas en relación con el "*TARGET*", se utilizan diagramas de distribución de densidad, acompañados de diagramas de barras con las diferentes categorías asociadas a cada variable.

En la mayoría de las gráficas se evidencia que el comportamiento de las categorías con mayor frecuencia en las variables categóricas se mantiene tanto para clientes con clase 0 como con clase 1, siendo coherente con la categoría de mayor frecuencia en el *"TARGET"*, es decir, 0 para representar el cumplimiento del cliente.

Figura 17

Diagrama de distribución de densidad con diagrama de barras de las variables NAME_INCOME_TYPE con TARGET y NAME_EDUCATION_TYPE con TARGET

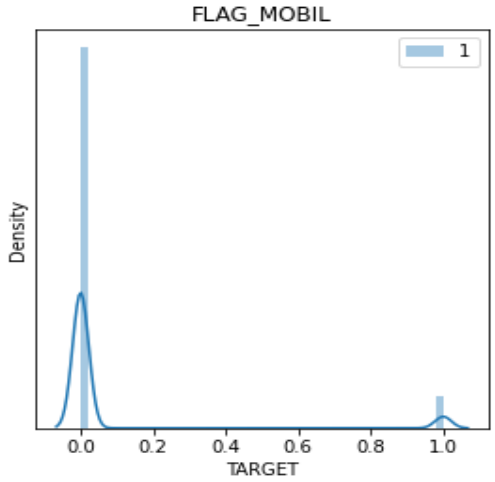


Nota: Elaboración propia.

Particularmente, se observa que las personas pensionadas presentan un mayor comportamiento de pago oportuno del préstamo, frente a las diferentes ocupaciones de los clientes. También se observa que en cuanto a la educación, los clientes con cumplimiento tienen educación superior, mientras que los clientes con dificultades de pago en su mayoría solo llegan a nivel de secundaria.

Figura 18

Diagrama de distribución de densidad con diagrama de barras de la variable FLAG_MOBIL con TARGET



Nota: Elaboración propia.

Para la variable *"FLAG_MOBIL"* que indica si el cliente proporcionó un teléfono móvil, se evidencia que todos los clientes proporcionaron esta información, independientemente de su nivel de cumplimiento.

4. Proceso de analítica

4.1. Pipeline principal

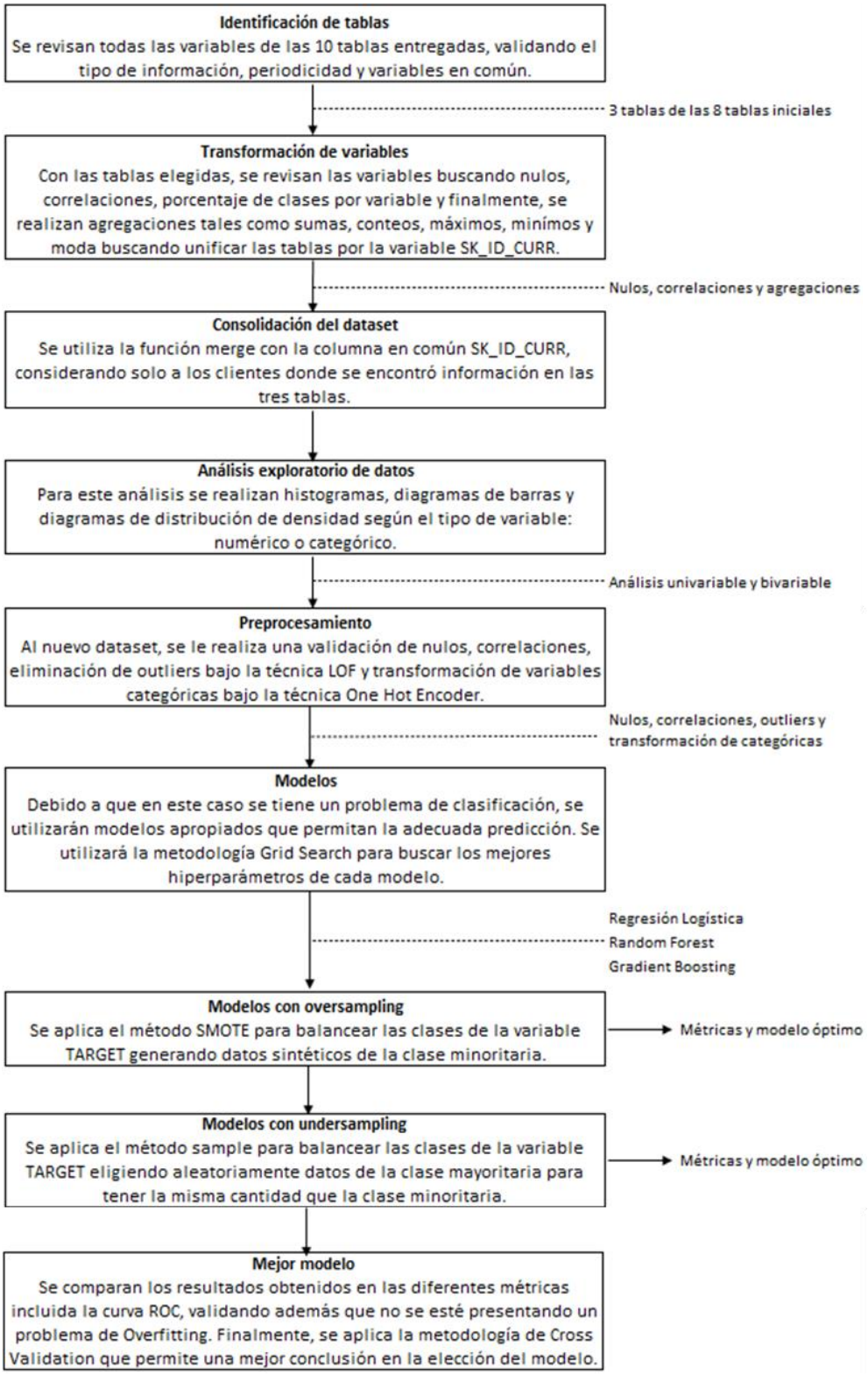
A nivel general, se llevó un proceso de nueve etapas como se muestra en la Figura 19, donde se comienzan identificando las tres tablas con las cuales se desarrolla el proyecto. Se realizan las transformaciones necesarias para llevar las tres tablas a una misma variable en común. Luego, se unen dichas tablas asegurando que el nuevo dataset solo tenga los clientes donde se cuenta con información en las tres tablas.

Ya con el dataset final, se comienza a realizar una exploración de los datos identificando patrones y demás, para con ello, pasar a una etapa de preprocesamiento donde se busca tratar el ruido que puede ocasionar el tener nulos, *outliers* y variables categóricas sin tratar.

Finalmente, se presentan los modelos que se llevarán a cabo bajo las dos metodologías de balanceo que lleva a escoger el mejor modelo por el resultado obtenido tanto en las métricas evaluadas como en la curva ROC.

Figura 19

Pipeline del proyecto



Nota: Elaboración propia.

4.2. Preprocesamiento

Después de consolidar el dataset con las tablas de interés y realizar el análisis exploratorio, se desarrolla un preprocesamiento de los datos para mejorar la calidad del dataset, con las fases que se describen a continuación:

4.2.1. *Tratamiento de duplicados*

Para el tratamiento de duplicados, se utiliza inicialmente el método "*duplicated* ()" para identificar si hay presencia de registros duplicados. En este caso se observa que la suma de duplicados es cero, es decir que no se presentan registros duplicados en la tabla.

4.2.2. *Tratamiento de datos nulos*

Para realizar el tratamiento de datos nulos, se realiza una suma de los valores nulos, y se detecta que no hay datos nulos porque éstos fueron tratados previamente a la unión de tablas.

Adicionalmente, se encuentran variables con su mayoría de registros equivalentes a una misma clase, lo cual sugiere que no aportará información relevante al modelo. Por lo cual, a continuación se eliminan las variables que contienen el 95% de sus registros asociados a una misma clase. Las variables eliminadas fueron: *"FLAG_CONT_MOBILE", "REG_REGION_NOT_LIVE_REGION", "REG_REGION_NOT_WORK_REGION", "AMT_REQ_CREDIT_BUREAU_HOUR", "AMT_REQ_CREDIT_BUREAU_DAY", "AMT_REQ_CREDIT_BUREAU_WEEK", "CREDIT_DAY_OVERDUE", "CNT_CREDIT_PROLONG", "CREDIT_TYPE_Another type of loan", "CREDIT_TYPE_Microloan".*

4.2.3. *Correlación de variables*

Se realizó la matriz de correlación para validar cuáles variables aportan la misma información al problema, considerando así, eliminar aquellas con una correlación mayor al 70%. Las variables que son eliminadas por presentar una correlación mayor al 70% con alguna otra variable son: *"PRODUCT_COMBINATION_POS", "CHANNEL_TYPE_Credit and cash offices", "NAME_CONTRACT_TYPE_Consumer loans", "NAME_CLIENT_TYPE_Repeater", "CREDIT_TYPE_Credit card", "NAME_CONTRACT_TYPE_Cash loans" y*

"PRODUCT_COMBINATION_Cash". Particularmente, en la Tabla 8 se observa que la variable *"SK_ID_PREV"* presenta una correlación del 86% con *"NAME_CONTRACT_TYPE_Cash loans"*, y *"AMT_CREDIT_y"* muestra una correlación del 72% con la misma variable. En este caso, se elimina la variable *"NAME_CONTRACT_TYPE_Cash loans"*, la cual indica si el préstamo en cuestión es en efectivo; lo cual no tiene mayor relevancia que la variable *"AMT_CREDIT_y"*, la cual representa el monto del crédito otorgado para la solicitud previa en Home Credit. Y en cuanto a la variable *"SK_ID_PREV"*, hace referencia a la identificación del cliente para créditos previos en Home Credit y permite la intersección con la información del mismo cliente en la solicitud del crédito actual.

Por otro lado, la variable *"FLAG_MOBIL"* cuenta con una sola clase equivalente a 1, por lo cual no aporta información al problema.

Tabla 8

Resultados de correlación entre variables

	CREDIT_TYPE_Mortgage	SK_ID_PREV	AMT_CREDIT_y
NAME_CONTRACT_TYPE_Cash loans	-0.028665	0.860667	0.721406

Nota: Elaboración propia.

4.2.4. *Tratamiento de outliers*

Para el tratamiento de *outliers*, inicialmente se grafican los boxplot de todas las variables numéricas, con el fin de identificar visualmente los valores atípicos en las distribuciones de los datos. Posteriormente se utiliza la técnica LOF, (*Outlier Detection With Local Outlier Factor (LOF)*, n.d.) para eliminar los *outliers*, variando el número de vecinos para identificar aquel con mayor capacidad para eliminar los *outliers* (se varía entre 3, 5, 7 y 9 número de vecinos). Finalmente, se vuelven a graficar las distribuciones de las variables a través de diagramas de cajas para visualizar la disminución de dichos datos atípicos.

De acuerdo con los resultados obtenidos en la búsqueda de los n-vecinos que eliminarán el mayor número de *outliers*, se encuentran resultados de la Tabla 9. Por lo tanto, se selecciona un n-vecinos igual a 3 para la eliminación de *outliers*.

Tabla 9

Resultados de iteraciones en búsqueda de los n-vecinos

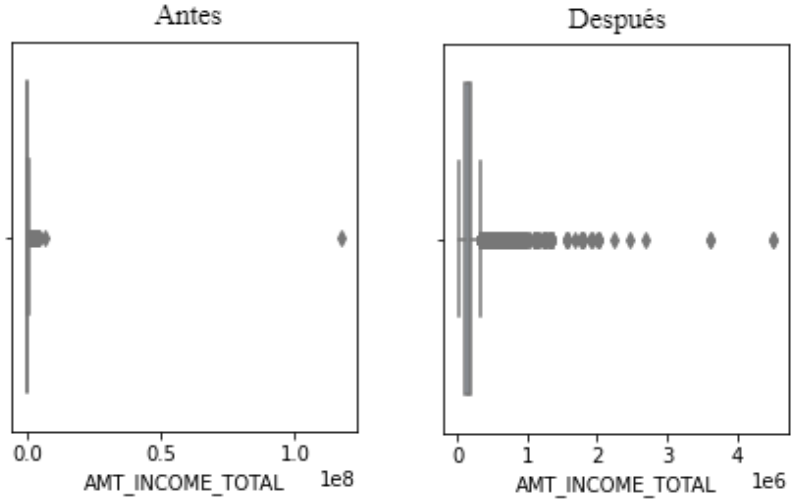
N° de vecinos	N° de outliers eliminados
3	2702
5	1454
7	1175
9	1071

Nota: Elaboración propia.

En la Figura 20 se observa el boxplot que representa la distribución de la variable *"AMT_INCOME_TOTAL"* antes y después del tratamiento de *outliers.* Allí, es preciso destacar que antes del tratamiento de *outliers* se presentaron muestras atípicas con valores totales del préstamo superiores a $1{,}0 \times 10^8$ ($100.000.000), mientras que la mayoría de los valores se ubican alrededor de $0{,}1 \times 10^8$ ($10.000.000). Ahora bien, después del tratamiento se eliminan la mayoría de los datos atípicos de la variable y se visualizan montos de préstamos oscilando entre 0 y 5×10^6 ($5.000.000).

Figura 20

Tratamiento de datos outliers para el monto del préstamo



Nota: Elaboración propia.

4.2.5. *Transformación de variables categóricas*

En la transformación de variables categóricas se determinan todas las categorías a transformar de cada variable, donde se encuentra que todas corresponden a variables categóricas de tipo nominal. Por lo

tanto, se decide hacer uso de la metodología *One Hot Encoder* para asignarle un valor binario a cada categoría representada en nuevas columnas.

Una de las variables categóricas transformada fue "*NAME_EDUCATION_TYPE"* donde se crean cinco columnas con valores binarios (0 y 1) para representar el tipo de nivel educativo alcanzado por cada cliente (*Secondary / secondary special, Higher education, Incomplete higher, Lower secondary, Academic degree*).

4.2.6. *Correlación con variables transformadas*

Se determina la matriz de correlación entre las variables para identificar posibles correlaciones altas con la variable *"TARGET"*. Después de realizar una búsqueda, se identifica que ninguna variable presenta una correlación mayor al 20% con el *"TARGET"*. Sin embargo, se incluirán todas las variables en la generación del modelo de clasificación.

4.2.7. *Balanceo de la variable objetivo*

Teniendo en cuenta que la variable objetivo *"TARGET"* presenta un desbalance muy significativo en sus clases, donde hay 225.586 registros de la clase 0 y solamente 19.120 de la clase 1; se implementan dos métodos de balanceo, con el fin de comparar cómo los resultados de los modelos se ven afectados por ambos métodos de balanceo, (Pykes, 2020).

- **Oversampling:** se emplea la técnica SMOTE para crear valores sintéticos en la clase 1, la cual representa la menor proporción de registros de la variable *"TARGET"*.
- **Undersampling:** se escoge aleatoriamente una muestra para la clase 0 con el mismo tamaño de la cantidad minoritaria de las clases (clase 1).

4.3. Modelos

Los modelos seleccionados para la búsqueda de la mejor predicción son: Regresión Logística, *Random Forest* y *Gradient Boosting*. Estos tres modelos son utilizados para problemas de clasificación y serán puestos a prueba con el fin de compararlos entre sí y encontrar el modelo capaz de realizar la mejor predicción.

- **Regresión Logística:** "una de las principales aplicaciones de la regresión logística es la de clasificación binaria, en el que las observaciones se clasifican en un grupo u otro dependiendo del valor que tome la variable empleada como predictor", (Amat, 2016).
- ***Random Forest:*** "son una combinación de predictores de árboles de modo que cada árbol depende de los valores de un vector aleatorio muestreado de forma independiente y con la misma distribución para todos los árboles del bosque. El error de generalización para los bosques converge en un límite a medida que aumenta el número de árboles en el bosque", (Breiman, 2001).
- ***Gradient Boosting*:** "el procedimiento de aprendizaje ajusta consecutivamente nuevos modelos para proporcionar una estimación más precisa de la variable de respuesta. La idea principal detrás de este algoritmo es construir los nuevos aprendices base para que se correlacionen al máximo con el gradiente negativo de la función de pérdida, asociado con todo el conjunto", (Natekin & Knoll, 2013).

Para cada uno de los tres modelos definidos anteriormente, se emplea el método *Grid Search*, (*Sklearn.model_selection.GridSearchCV*, n.d.) con el objetivo de encontrar los mejores hiperparámetros que se ajustan al modelo, como se muestra a continuación:

4.3.1. *Regresión Logística*

Para este modelo se hace variación de tres hiperparámetros:

- ***Solver:*** se entrena el modelo con newton-cg, sag y saga, considerando que éstos son más apropiados para el tipo de problema abordado, debido que el *solver* "liblinear" se utiliza para problemas multiclase y "lbfgs" se emplea principalmente para pequeños conjuntos de datos.
- ***Penalty:*** se entrena el modelo con "none" y "l2", los cuales aplican para los tipos de solver seleccionados.
- **C:** se entrena el modelo con valores de 1.0, 0.1, 0.01 y 10, valores aleatorios escogidos por las analistas.

4.3.2. *Random Forest*

Para este modelo se hace variación de dos hiperparámetros, en ambos los valores son escogidos aleatoriamente por las analistas:

- **n_estimators:** se entrena el modelo con valores de 10, 20, 30, 40, 50, 60 y 100.
- **max_depth:** se entrena el modelo con valores de 4, 6, 8, 10, 12, 14.

4.3.3. *Gradient Boosting*

Para este modelo se hace variación de dos hiperparámetros, en ambos los valores son escogidos aleatoriamente por las analistas:

- **n_estimators:** se entrena el modelo con valores de 20, 40 y 60.
- **max_depth:** se entrena el modelo con valores de 6 y 12.

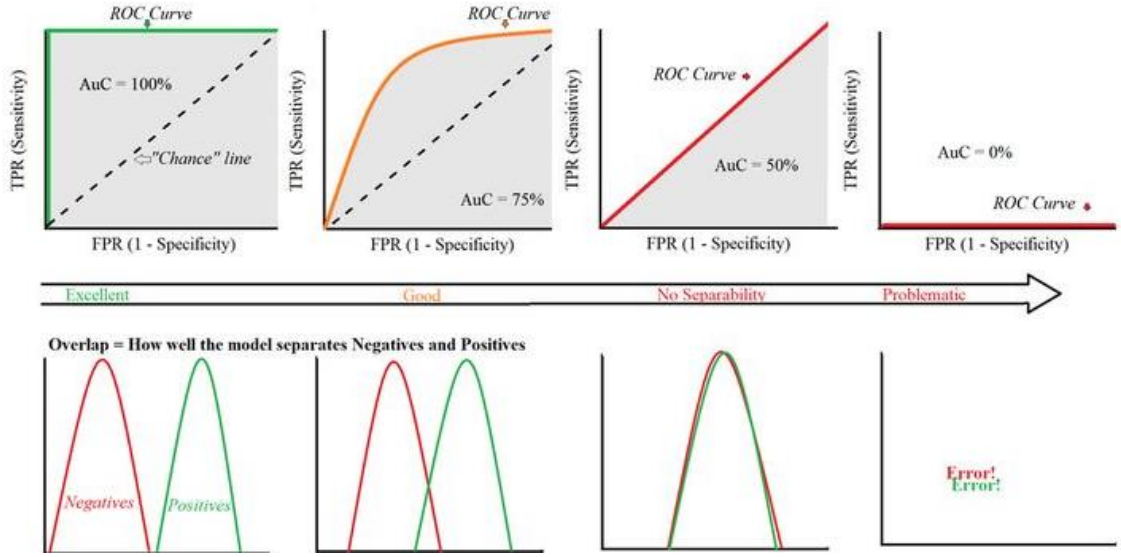
4.4. Métricas

Las métricas de desempeño de los modelos entrenados fueron calculadas con el módulo *metrics* de la librería sklearn (sklearn.metrics), el cual implementa varias funciones de pérdida, puntuación y utilidad para medir el rendimiento de la clasificación. A continuación, se presentan las métricas evaluadas, (Recuero de los Santos, 2021) y la curva ROC, (Torres, 2010) evaluadas en el caso de estudio:

Tabla 10
Métricas evaluadas en el caso de estudio

Métrica	Función	Definición	Fórmula
accuracy (acc)	metrics.accuracy_score	"La exactitud (accuracy) representa el porcentaje de predicciones correctas frente al total".	$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$
precision (ppv)	metrics.precision_score	"La precisión (precision) se refiere a lo cerca que está el resultado de una predicción del valor verdadero".	$precision = \frac{TP}{TP + FP}$
recall	metrics.recall_score	"La sensibilidad (recall) representa la tasa de verdaderos positivos".	$recall = \frac{TP}{TP + FN}$

f1	metrics. f1_score	"El valor f1 se utiliza para combinar las medidas de precision y recall en un sólo valor".	$f1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$
matthews (mcc)	metrics.matthews_corrcoef	"El valor matthews (mcc) ayuda a resumir la matriz de confusión o una matriz de error".	$mcc = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP) \cdot (FN+TN) \cdot (TP+FI}}$
balanced accuracy (bacc)	metrics.balanced_accuracy_score	"El valor balanced accuracy (bacc) es la media aritmética de sensibilidad y especificidad, su caso de uso es cuando se trata de datos desequilibrados".	$bacc = \frac{1}{2}(\frac{TP}{P} + \frac{TN}{N})$
curva ROC	metrics.roc_auc_score	"La curva ROC es un gráfico en el que se observan todos los pares de Sensibilidad y complemento de la Especificidad, resultantes de la variación continua de todos los puntos de corte en todo el rango de resultados observados".	



Nota: (Glen, 2019)

Nota: Elaboración propia con información de otras fuentes: (Recuero de los Santos, 2021), (Torres, 2010) y (Glen, 2019).

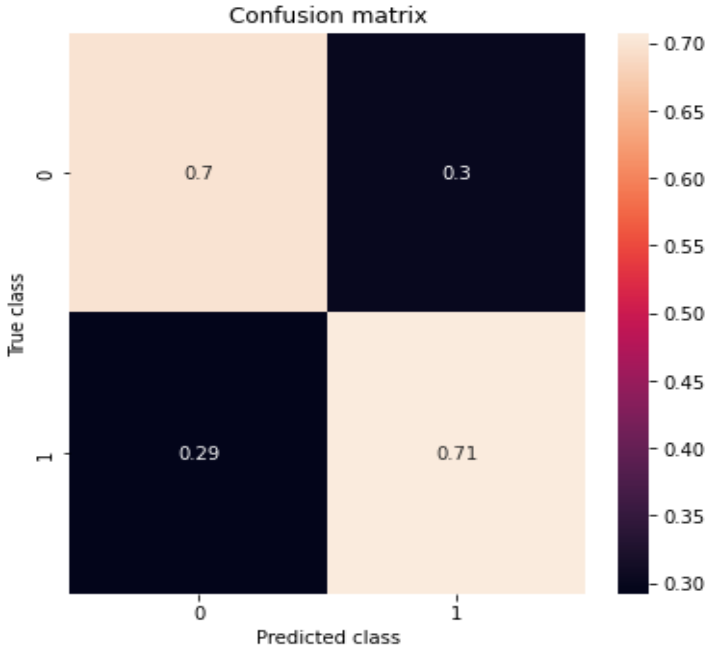
5. Metodología

5.1. Baseline

La **primera iteración** realizada fue con el modelo de clasificación Regresión Logística, donde se aplicó la metodología de balanceo *oversampling* con la técnica SMOTE, equilibrando la distribución de clases al aumentar aleatoriamente los ejemplos de clases minoritarias. En esta primera iteración se encuentra que el mejor score se da con los hiperparámetros C=1.0, *penalty*='l2' y con un *solver*="newton-cg", obteniendo así, una matriz de confusión con las siguientes probabilidades de las predicciones correctas e incorrectas.

Figura 21

Matriz de confusión del modelo Regresión Logística con oversampling



Nota: Elaboración propia.

La matriz de confusión evidencia las probabilidades de las predicciones correctas y los errores de clasificación del modelo de Regresión Logística.

- **Verdaderos Positivos (TP):** 0.70
- **Verdaderos Negativos (TN):** 0.71
- **Falsos Positivos (FP):** 0.29

- **Falsos Negativos (FN):** 0.30

En cuanto a las métricas evaluadas, se obtienen resultados alrededor del 70% en la mayoría de éstas. Además, se obtiene una curva ROC del 77%.

Tabla 11

Métricas evaluadas con el modelo Regresión Logística con oversampling

	Regresión Logística
accuracy (acc)	0.703100
precision (ppv)	0.700922
recall	0.707845
f1	0.704367
matthews (mcc)	0.406223
balanced accuracy (bacc)	0.703103

Nota: Elaboración propia.

La métrica mcc, es muy útil al momento de evaluar problemas biclase (como lo es en este caso), y el hecho de que esta métrica sea la más baja, evidencia que el clasificador tiene dificultad para determinar correctamente tanto las predicciones negativas como las positivas, lo que lleva a pensar, que es conveniente revisar otros modelos de clasificación para abordar el problema.

5.2. Validación

Para la partición de los datos, se utiliza la función “train_test_split”, la cual divide el dataset en dos bloques, en este caso, uno de *train* con el 70% de los datos y otro de *test* con el 30% restante.

Además de las métricas elegidas para el proceso de evaluación, se implementó la metodología de validación cruzada o *Cross Validation*, (Amat Rodrigo, 2020) para soportar el proceso de validación del mejor modelo al realizar 10 iteraciones con diferentes grupos aleatorios del dataset. Esto permitirá validar que el modelo no depende de la aleatoriedad de los datos, no presenta sesgo o problemas de *overfitting* y que el resultado obtenido es aceptable para el caso de estudio.

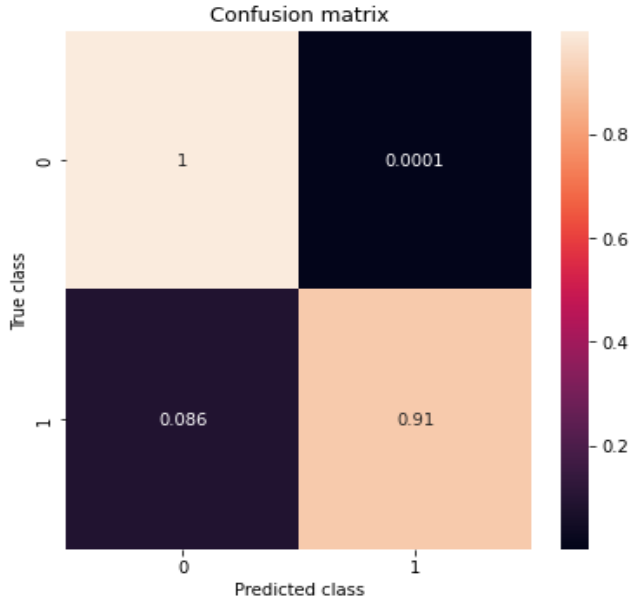
5.3. Iteraciones y evolución

La **segunda iteración** realizada fue con el modelo de clasificación *Random Forest*, donde se aplica la metodología de balanceo *oversampling* con la técnica SMOTE. En esta segunda iteración se encuentra

que el mejor score se da con los hiperparámetros n_estimators=100 y max_depth=14, obteniendo así, una matriz de confusión con las siguientes probabilidades de las predicciones correctas e incorrectas.

Figura 22

Matriz de confusión del modelo Random Forest con oversampling



Nota: Elaboración propia.

La matriz de confusión evidencia las probabilidades de las predicciones correctas y los errores de clasificación del modelo de *Random Forest*.

- **Verdaderos Positivos (TP):** 1
- **Verdaderos Negativos (TN):** 0.91
- **Falsos Positivos (FP):** 0.086
- **Falsos Negativos (FN):** 0.0001

En cuanto a las métricas evaluadas, se obtienen resultados entre 91% y 99%. Además, se obtiene una curva ROC del 98%.

Tabla 12

Métricas evaluadas con el modelo Random Forest

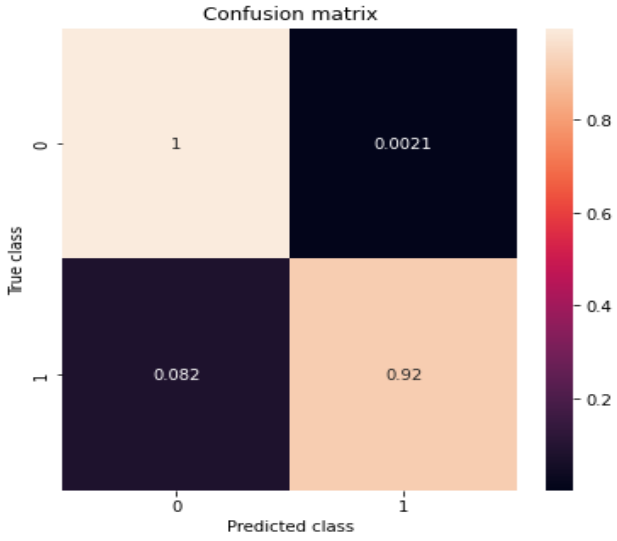
	Random Forest
accuracy (acc)	0.956898
precision (ppv)	0.999887
recall	0.913843
f1	0.954930
matthews (mcc)	0.917189
balanced accuracy (bacc)	0.956870

Nota: Elaboración propia.

La **tercera iteración** realizada fue con el modelo de clasificación *Gradient Boosting*, donde se aplica la metodología de balanceo *oversampling* con la técnica SMOTE. En esta tercera iteración se encuentra que el mejor score se da con los hiperparámetros n_estimators=60 y max_depth=12, obteniendo así, una matriz de confusión con las siguientes probabilidades de las predicciones correctas e incorrectas.

Figura 23

Matriz de confusión del modelo Gradient Boosting con oversampling



Nota: Elaboración propia.

La matriz de confusión evidencia las probabilidades de las predicciones correctas y los errores de clasificación del modelo de *Gradiente Boosting*.

- **Verdaderos Positivos (TP):** 1
- **Verdaderos Negativos (TN):** 0.92
- **Falsos Positivos (FP):** 0.082

- **Falsos Negativos (FN):** 0.0021

En cuanto a las métricas evaluadas, se obtienen resultados entre 91% y 99%. Además, se obtiene una curva ROC del 98%.

Tabla 13

Métricas evaluadas con el modelo Gradient Boosting con oversampling

	Gradiente Boosting
accuracy (acc)	0.958168
precision (ppv)	0.997719
recall	0.918382
f1	0.956408
matthews (mcc)	0.919240
balanced accuracy (bacc)	0.958142

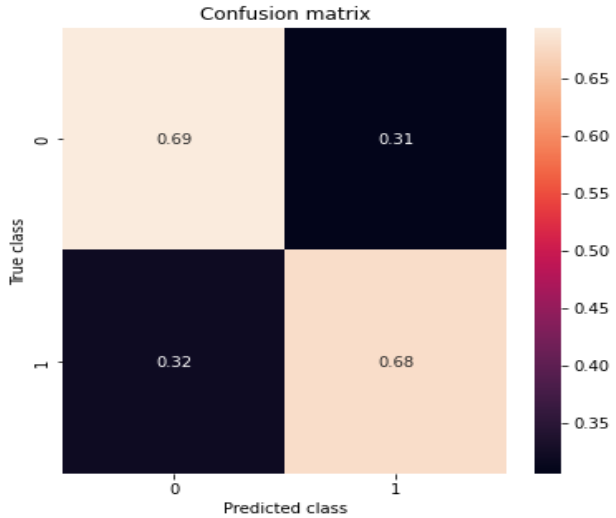
Nota: Elaboración propia.

Para las siguientes tres iteraciones, se aplicó la metodología de balanceo *undersampling* donde se reduce la cantidad de registros previamente al entrenamiento de los modelos.

La **cuarta iteración** realizada fue con el modelo de clasificación Regresión Logística, donde se aplica la metodología de balanceo *undersampling*. En esta cuarta iteración se encuentra que el mejor score se da con los hiperparámetros C=10, *penalty*='l2' y con un *solver*="newton-cg", obteniendo así, una matriz de confusión con las siguientes probabilidades de las predicciones correctas e incorrectas.

Figura 24

Matriz de confusión del modelo Regresión Logística con undersampling



Nota: Elaboración propia.

La matriz de confusión evidencia las probabilidades de las predicciones correctas y los errores de clasificación del modelo de Regresión Logística.

- **Verdaderos Positivos (TP):** 0.69
- **Verdaderos Negativos (TN):** 0.68
- **Falsos Positivos (FP):** 0.32
- **Falsos Negativos (FN):** 0.31

En cuanto a las métricas evaluadas, se obtienen resultados entre 37% y 68%. Además, se obtiene una curva ROC del 75%.

Tabla 14

Métricas evaluadas con el modelo Regresión Logística con undersampling

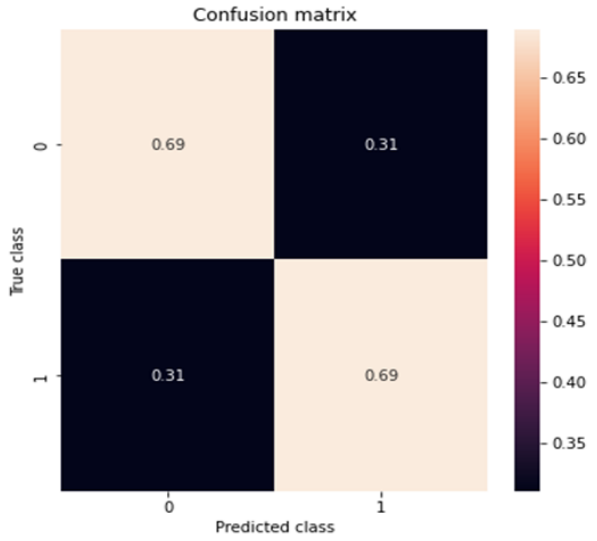
	Regresión Logística
accuracy (acc)	0.686977
precision (ppv)	0.684781
recall	0.679831
f1	0.682297
matthews (mcc)	0.373836
balanced accuracy (bacc)	0.686898

Nota: Elaboración propia.

La **quinta iteración** realizada fue con el modelo de clasificación *Random Forest*, donde se aplica la metodología de balanceo *undersampling*. En esta quinta iteración se encuentra que el mejor score se da con los hiperparámetros n_estimators=180 y max_depth=18, obteniendo así, una matriz de confusión con las siguientes probabilidades de las predicciones correctas e incorrectas.

Figura 25

Matriz de confusión del modelo Random Forest con undersampling



Nota: Elaboración propia.

La matriz de confusión evidencia las probabilidades de las predicciones correctas y los errores de clasificación del modelo de *Random Forest*.

- **Verdaderos Positivos (TP):** 0.69
- **Verdaderos Negativos (TN):** 0.69
- **Falsos Positivos (FP):** 0.31
- **Falsos Negativos (FN):** 0.31

En cuanto a las métricas evaluadas, se obtienen resultados entre 37% y 68%. Además, se obtiene una curva ROC del 75%.

Tabla 15

Métricas evaluadas con el modelo Random Forest con undersampling

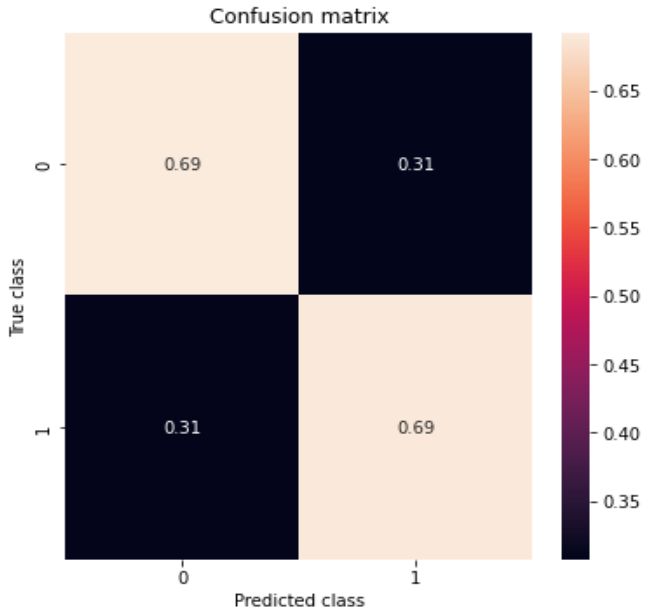
	Random Forest
accuracy (acc)	0.688982
precision (ppv)	0.684303
recall	0.688646
f1	0.686467
matthews (mcc)	0.377937
balanced accuracy (bacc)	0.688978

Nota: Elaboración propia.

La **sexta y última iteración** realizada fue con el modelo de clasificación *Gradient Boosting*, donde se aplica la metodología de balanceo *undersampling*. En esta sexta iteración se encuentra que el mejor score se da con los hiperparámetros n_estimators=120 y max_depth=4, obteniendo así, una matriz de confusión con las siguientes probabilidades de las predicciones correctas e incorrectas.

Figura 26

Matriz de confusión del modelo Gradient Boosting con undersampling



Nota: Elaboración propia.

La matriz de confusión evidencia las probabilidades de las predicciones correctas y los errores de clasificación del modelo de *Gradient Boosting*.

- **Verdaderos Positivos (TP):** 0.69
- **Verdaderos Negativos (TN):** 0.69
- **Falsos Positivos (FP):** 0.31
- **Falsos Negativos (FN):** 0.31

En cuanto a las métricas evaluadas, se obtienen resultados entre 38% y 69%. Además, se obtiene una curva ROC del 76%.

Tabla 16

Métricas evaluadas con el modelo Gradient Boosting con undersampling

	Gradiente Boosting
accuracy (acc)	0.691510
precision (ppv)	0.687072
recall	0.690585
f1	0.688824
matthews (mcc)	0.382982
balanced accuracy (bacc)	0.691500

Nota: Elaboración propia.

En la última iteración se define un modelo de Máquina de Soporte Vectorial; sin embargo, debido a la capacidad de RAM de la herramienta utilizada (Colab), no fue posible finalizar el entrenamiento del modelo.

5.4. Herramientas

Para el desarrollo del proyecto se usa la herramienta Colab, también conocido como "Colaboratory", el cual dispone de un entorno en línea que permite programar y ejecutar Python en el navegador. Los cuadernos de Colab permiten combinar código ejecutable y texto enriquecido en un mismo documento, además de imágenes, HTML, LaTeX, entre otros. Además, los cuadernos de Colab se almacenan en una cuenta de Google Drive, (Google, n.d.).

Del mismo modo, resaltamos las principales librerías utilizadas, las cuales fueron: pandas, seaborn, numpy, matplotlib, sklearn e imblearn.

6. Resultados

6.1. Métricas

6.1.1. *Modelos con oversampling*

Tabla 17

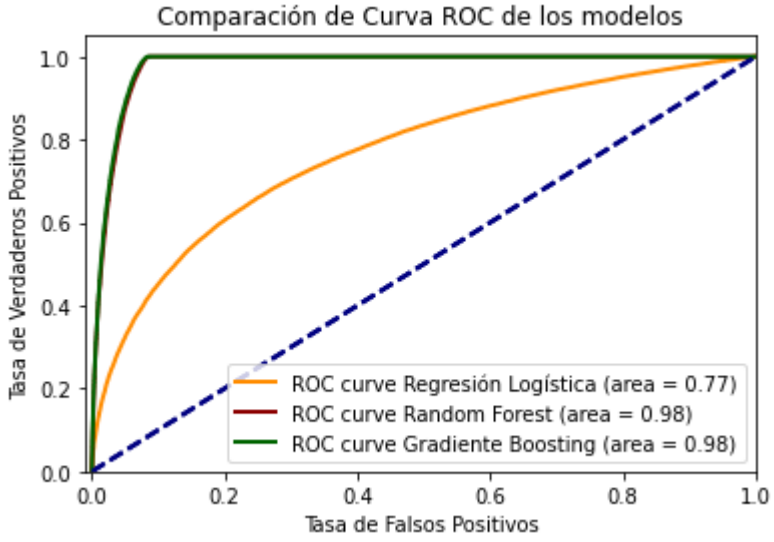
Comparativo de los resultados obtenidos en las métricas evaluadas con los modelos Regresión Logística, Random Forest y Gradient Boosting con oversampling

	Regresión Logística	Random Forest	Gradiente Boosting
accuracy (acc)	0.703100	0.956898	0.958168
precision (ppv)	0.700922	0.999887	0.997719
recall	0.707845	0.913843	0.918382
f1	0.704367	0.954930	0.956408
matthews (mcc)	0.406223	0.917189	0.919240
balanced accuracy (bacc)	0.703103	0.956870	0.958142

Nota: Elaboración propia.

Figura 27

Comparación de las curvas ROC con los modelos Regresión Logística, Random Forest y Gradient Boosting con oversampling



Nota: Elaboración propia.

La curva ROC evidencia que los modelos *Random Forest* y *Gradient Boosting* presentan un desempeño esperado del sistema de clasificación del 98%, mientras que el modelo de Regresión Logística evidencia un desempeño del 77%.

Por tanto, considerando los resultados de las métricas de evaluación y la curva ROC, se determina que el modelo con resultados más acertados es el *Gradient Boosting*.

Con los resultados obtenidos por los últimos dos modelos, se puede intuir que los modelos se han visto afectados por la metodología de balanceo utilizada previamente (SMOTE), posiblemente por la generación de una gran cantidad de datos sintéticos de la clase 1.

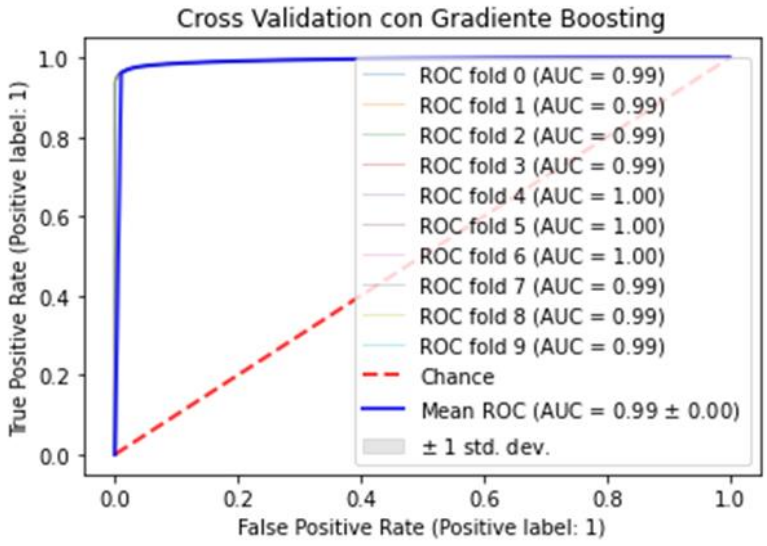
Posteriormente, se emplea la metodología Stratified K Fold, (Amat Rodrigo, 2020) para entrenar el modelo *Gradient Boosting* seleccionado como el modelo de mejor desempeño entrenado con los datos resultantes del balanceo *oversampling*, implementando ahora la validación cruzada.

6.1.2. *Modelo óptimo con oversampling*

Evaluando el modelo con los datos de *train*, el promedio de la curva ROC es 99%.

Figura 28

Curvas ROC con la metodología Cross Validation (datos train) aplicada al mejor modelo, Gradient Boosting con oversampling

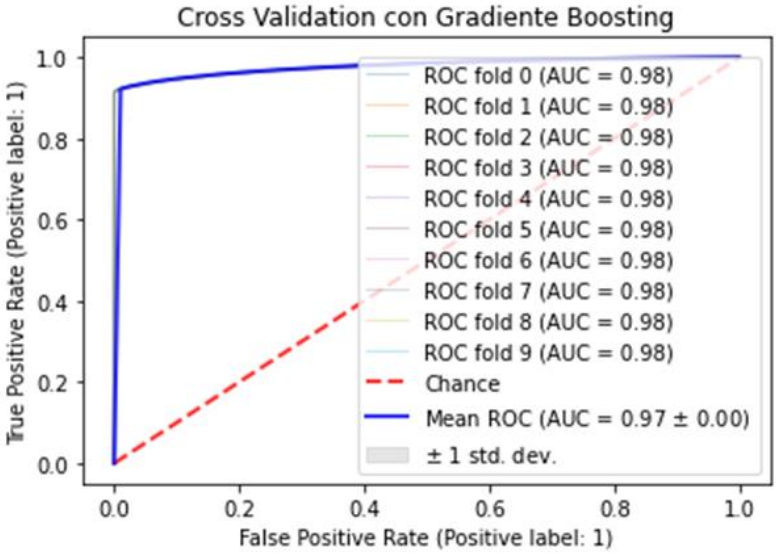


Nota: Elaboración propia.

Evaluando el modelo con los datos de *test*, el promedio de la curva ROC es 97%.

Figura 29

Curvas ROC con la metodología Cross Validation (datos test) aplicada al mejor modelo, Gradient Boosting con oversampling



Nota: Elaboración propia.

6.1.3. *Modelos con undersampling*

Tabla 18

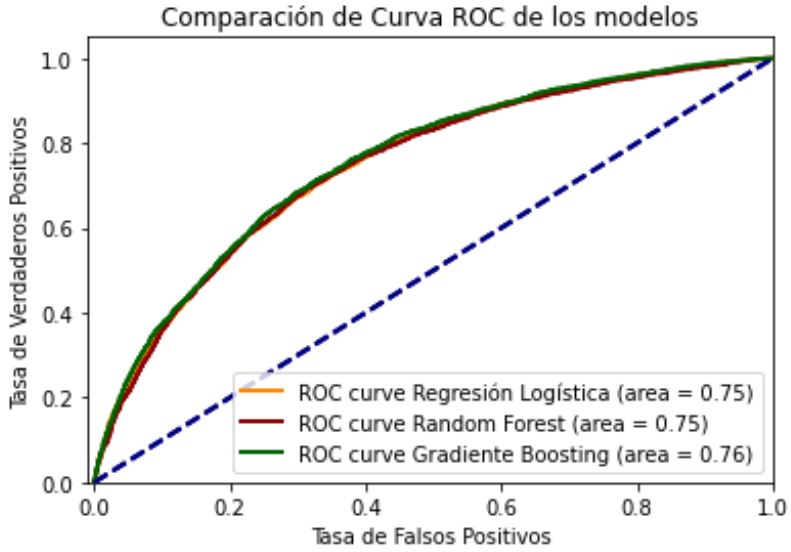
Comparativo de los resultados obtenidos en las métricas evaluadas con los modelos Regresión Logística, Random Forest y Gradient Boosting con undersampling

	Regresión Logística	Random Forest	Gradiente Boosting
accuracy (acc)	0.686977	0.688982	0.691510
precision (ppv)	0.684781	0.684303	0.687072
recall	0.679831	0.688646	0.690585
f1	0.682297	0.686467	0.688824
matthews (mcc)	0.373836	0.377937	0.382982
balanced accuracy (bacc)	0.686898	0.688978	0.691500

Nota: Elaboración propia.

Figura 30

Comparación de las curvas ROC con los modelos Regresión Logística, Random Forest y Gradient Boosting con undersampling



Nota: Elaboración propia.

La curva ROC evidencia que los modelos *Regresión Logística* y *Random Forest* presentan un desempeño esperado del sistema de clasificación del 75%, mientras que el modelo *de Gradient Boosting* evidencia un desempeño del 76%.

Por tanto, considerando los resultados de las métricas de evaluación y la curva ROC, se determina que el modelo con resultados más acertados es el *Gradient Boosting.*

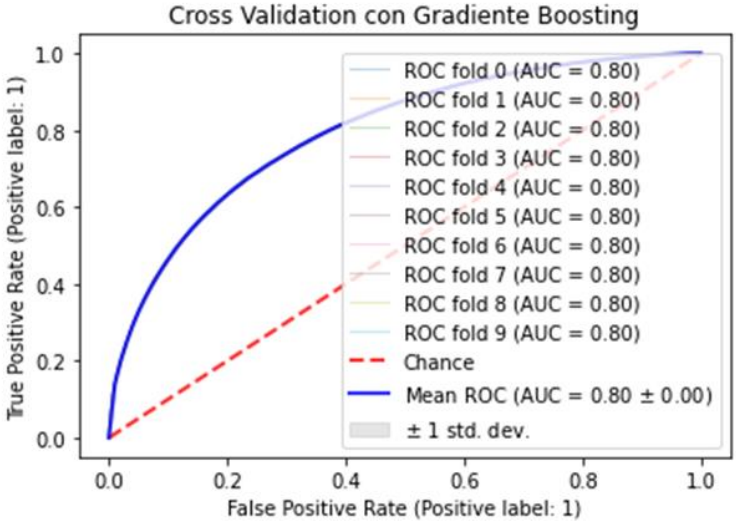
Posteriormente, se emplea la metodología Stratified K Fold, (Amat Rodrigo, 2020) para entrenar el modelo *Gradient Boosting* seleccionado como el modelo de mejor desempeño entrenado con los datos resultantes del balanceo *undersampling*, implementando ahora la validación cruzada.

6.1.4. *Modelo óptimo con undersampling*

Evaluando el modelo con los datos de *train*, el promedio de la curva ROC es 80%.

Figura 31

Curvas ROC con la metodología Cross Validation (datos train) aplicada al mejor modelo, Gradient Boosting con undersampling

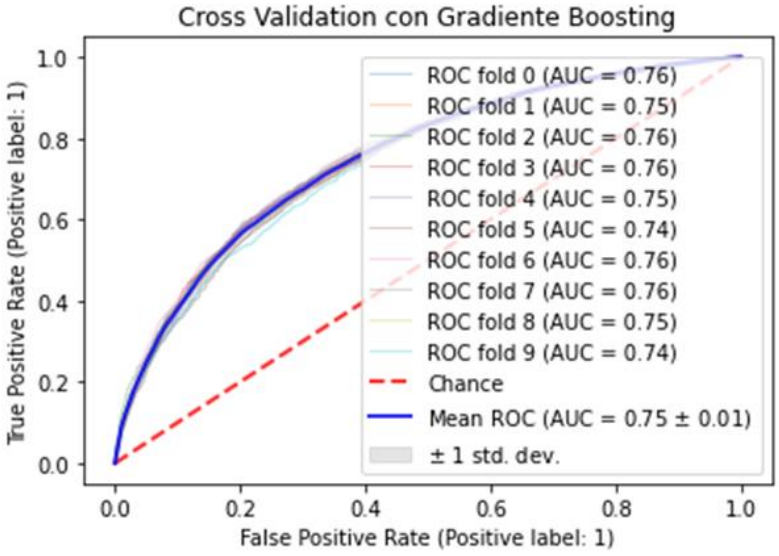


Nota: Elaboración propia.

Evaluando el modelo con los datos de *test*, el promedio de la curva ROC es 75%.

Figura 32

Curvas ROC con la metodología Cross Validation (datos test) aplicada al mejor modelo, Gradient Boosting con undersampling



Nota: Elaboración propia.

6.2. Evaluación cualitativa

Analizando el resultado de las métricas obtenidas al utilizar los tres modelos anteriormente definidos con la técnica de balanceo *oversampling*, se observa un resultado bastante optimista o perfecto, lo que indica la posibilidad de que se esté presentando un problema en el entrenamiento del modelo con una gran cantidad de datos sintéticos sobre la clase inicialmente minoritaria, debido a que al revisar el porcentaje de desbalance de la variable objetivo, se encuentra que casi el 85% de los datos se encuentran en la clase 0 y solo el 15% en la clase 1.

Este resultado tan certero con la técnica de balanceo *oversampling* fue precisamente lo que llevó a aplicar otra técnica diferente para balancear los datos; y en este caso, se emplea la técnica de *undersampling* con los mismos tres tipos de modelos. Con esta segunda alternativa de balanceo se encuentra que las métricas bajaron, pero se mantienen dentro de un rango que podría ser considerado como aceptable.

Bajo este panorama, se evidencia que los modelos *Gradient Boosting* entrenados con los datos resultantes de ambas técnicas de balanceo, cumplen con predicciones acertadas, resaltando que con la técnica *undersampling* se obtiene una curva ROC de 0.75, resultado en un rango bueno, que cumple además con el porcentaje esperado como satisfactorio en las métricas del negocio (0.70).

6.3. Consideraciones de producción

Ante una puesta en producción, el modelo de clasificación requiere un entrenamiento dinámico para reetiquetar los registros de acuerdo con la actualización del estado de cartera de cada cliente, y de este modo validar el cumplimiento actual de cada cliente respecto a su obligación financiera. Cada nuevo entrenamiento del modelo se ejecuta una vez al mes, siendo consecuente con la periodicidad de las obligaciones de pago de cada cliente, es decir que la cartera se recupera mensualmente, incluyendo el capital y los intereses de las cuotas de cada préstamo de vivienda. Por otro lado, las predicciones se pueden evaluar diariamente de acuerdo con las solicitudes recibidas en el día por la entidad Home Credit, y para la visualización de los resultados del modelo es posible emplear un tablero de control en ambiente productivo.

7. Conclusiones

Considerando el caso de clasificación de Home Credit Default Risk donde se busca predecir el cumplimiento o no de las obligaciones financieras por parte de cada cliente, se logra abordar el problema desde tres modelos diferentes de *machine learning*, encontrando que ciertamente el modelo de Regresión Logística no es el óptimo para este caso donde el dataset a trabajar después de realizar la limpieza y el tratamiento previo, presenta una gran cantidad de variables de entrada (112 variables).

Por otro lado, se evidencia que los otros dos modelos, *Random Forest* y *Gradient Boosting*, al poseer configuraciones similares también presentaron resultados semejantes en cada iteración con diferentes hiperparámetros, pero finalmente fue el modelo *Gradient Boosting* el que logra sobresalir con un desempeño ligeramente superior representado en el valor resultante de la curva ROC.

Adicionalmente, para equilibrar el desbalance de las clases de la variable objetivo, se utilizan dos técnicas de balanceo, las cuales representan diferentes comportamientos en el dataset al ejecutar el entrenamiento de cada modelo y en consecuencia, también se evidencian diferentes resultados en las métricas de evaluación en cada caso. Concretamente, al hacer el balanceo por encima aumentando la cantidad de la clase minoritaria, se crea una gran cantidad de datos sintéticos; pero al hacer el balanceo por debajo reduciendo los registros de la clase mayoritaria, se pierde una gran cantidad de datos reales.

A partir de los resultados, es preciso considerar posibles trabajos futuros, los cuales pueden abarcar el entrenamiento del dataset con un modelo de Máquina de Soporte Vectorial, el cual se considera uno de los mejores clasificadores para problemas binarios. Sin embargo, es indispensable considerar la capacidad de RAM necesaria para entrenar este modelo.

Por otro lado, un modelo de Redes Neuronales Artificiales para clasificación también es una opción para considerar en trabajos futuros debido a que este tipo de modelo cada vez se acerca más a la idea de reproducir el funcionamiento del cerebro humano, entrenando un conjunto de neuronas conectadas entre sí que trabajan en conjunto.

8. Referencias

Amat, J. (2016, Ago). *Regresión logística simple y múltiple*. Cienciadedatos.net. Retrieved Nov 20, 2021, from https://www.cienciadedatos.net/documentos/27_regresion_logistica_simple_y_multiple

Amat Rodrigo, J. (2020, Nov). *Validación de modelos predictivos (machine learning): Cross-validation, OneLeaveOut, Bootstraping*. Cienciadedatos.net. Retrieved May 2, 2022, from https://www.cienciadedatos.net/documentos/30_cross-validation_oneleaveout_bootstrap

Breiman, L. (2001, Oct). Bosques aleatorios. Aprendizaje automático. *Springer Link*, (45), 5–32. https://doi.org/10.1023/A:1010933404324

Calderón Bandera, B. (n.d.). *La Cobranza*. CEFA. Retrieved Feb 13, 2022, from https://www.cefa.com.mx/art_art110727.html

Costo de oportunidad. (2022, Mar 26). Gerencie.com. Retrieved Feb 13, 2022, from https://www.gerencie.com/costo-de-oportunidad.html

Glen, S. (2019, Sep 3). *ROC Curve Explained in One Picture*. Data Science Central. Retrieved Nov 6, 2021, from https://www.datasciencecentral.com/roc-curve-explained-in-one-picture/

Google. (n.d.). Te damos la bienvenida a Colaboratory. https://colab.research.google.com/?hl=es

Home Credit Default Risk. (2018, Ago 29). Kaggle. Retrieved Oct 18, 2021, from https://www.kaggle.com/c/home-credit-default-risk/data

Natekin, A., & Knoll, A. (2013, Dic 04). Gradient boosting machines, a tutorial. *Front. Neurorobot*, *7*(21). https://doi.org/10.3389/fnbot.2013.00021

Outlier detection with Local Outlier Factor (LOF). (n.d.). Scikit-learn. Retrieved Nov 16, 2021, from https://scikit-learn.org/0.24/auto_examples/neighbors/plot_lof_outlier_detection.html

Pykes, K. (2020, Sep 10). *Oversampling and Undersampling*. Towards Data Science. Retrieved Abr 28, 2022, from https://towardsdatascience.com/oversampling-and-undersampling-5e2bbaf56dcf

Recuero de los Santos, P. (2021, Dic 13). *Cómo interpretar la matriz de confusión: ejemplo práctico*. Think Big Empresas. Retrieved Mar 13, 2022, from https://empresas.blogthinkbig.com/como-interpretar-la-matriz-de-confusion-ejemplo-practico/

Riesgo crediticio. (n.d.). SAS. Retrieved Feb 13, 2022, from https://www.sas.com/es_co/insights/risk-management/credit-risk-management.html

sklearn.model_selection.GridSearchCV. (n.d.). Scikit-learn. Retrieved May 5, 2022, from https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

Torres, A. (2010, Jul). *Curvas ROC para Datos de Supervivencia. Aplicación a Datos Biomédicos*. Departamento de Estadística, Análisis Matemático y Optimización. Retrieved Nov 29, 2021, from http://eio.usc.es/pub/mte/descargas/ProyectosFinMaster/Proyecto_407.pdf

Yadav, D. (2019, Dic 6). *Categorical encoding using Label-Encoding and One-Hot-Encoder*. Towards Data Science. Retrieved Nov 16, 2021, from https://towardsdatascience.com/categorical-encoding-using-label-encoding-and-one-hot-encoder-911ef77fb5bd