Aprendizaje automático para la detección de lavado de dinero en transacciones

Sandra María Cavazos Huerta

29 de noviembre de 2023

Resumen

El lavado de dinero (LD) es un problema internacional que afecta la integridad de las instituciones financieras y la estabilidad económica y supone la persistencia del crimen. La detección proactiva de actividades de LD es esencial para mitigar riesgos legales y financieros. En este estudio se explora la aplicación de técnicas de aprendizaje automático (AA) para identificar transacciones sospechosas dentro de un conjunto de datos transaccionales. Dichas transacciones se encuentran etiquetadas y comprenden características tanto numéricas como categóricas, para entrenar y evaluar varios modelos de clasificación, incluyendo regresión logística, árboles de decisión y XGBoost. La eficacia de cada modelo se mide mediante métricas de rendimiento estándar como la precisión, la sensibilidad, el valor F1 y el área bajo la curva ROC. Las implicaciones de estos hallazgos pueden ser significativas para el desarrollo de sistemas automáticos de monitoreo de transacciones en el sector financiero para la pronta y correcta detección de operaciones de LD.

Palabras clave: Prevención, lavado de dinero, aprendizaje automático, supervisado, no supervisado.

1 Introducción

El lavado de dinero es el proceso a través del cual es encubierto el origen de los recursos generados mediante actividades ilícitas, como el tráfico de drogas, contrabando de armas, corrupción, fraude, entre otros, haciendo parecer que son fruto de actividades legítimas para que así circulen sin problema en el sistema financiero[1]. La lucha contra el lavado de dinero es una prioridad para las instituciones financieras y los reguladores globales. Las Naciones Unidas (2021) estiman que el monto del dinero lavado anualmente representa aproximadamente un $2{,}7\,\%$ del PIB mundial [2].

La detección oportuna de operaciones de LD es fundamental no solo para cumplir con la regulación y evitar daños reputacionales, sino también para prevenir el impacto negativo en la economía y el bienestar social. La manera tradicional para detectar operaciones sospechosas considera reglas conservadoras que conllevan una considerable carga administrativa y volumen de falsos positivos. Con la aplicación de técnicas de AA, existe la posibilidad de mejorar significativamente la precisión y eficiencia de los sistemas de detección de LD.

En este artículo, se presenta una investigación exhaustiva sobre la aplicabilidad de varios algoritmos de aprendizaje de máquina para identificar transacciones financieras sospechosas, aprovechando un conjunto de datos de transacciones con etiquetas de lavado de dinero.

2 Metodología

La metodología utilizada en este estudio se estructura en varias etapas clave, cada una desempeñando un papel importante en el análisis y procesamiento de los datos.

2.1 Conjunto de datos

Tras la valoración de diversas bases de datos, se elige ”HI Small Trans” de Kaggle. Este conjunto de datos contiene transacciones sintéticas etiquetadas como LD o legítimas. Fueron generadas por IBM ya que el acceso a datos transaccionales reales es limitado por cuestiones de propiedad y privacidad, además que su correcta clasificación en materia de LD es complicada.

El generador de IBM modela las tres etapas de LD:

1. **Colocación:** introducir los recursos ilícitos en el sistema financiero.
2. **Estratificación:** mezclar los recursos en el sistema mediante varias transacciones que dificulten el rastreo del origen.
3. **Integración:** reintroducir los recursos en la economía para su uso.

La base de datos contiene 5,078,345 registros y once variables:

Variable	**Descripción**
Timestamp	Fecha y hora de la transacción.
From Bank	Código numérico del banco originador.
Account (From)	Código hexadecimal de la cuenta originadora.
To Bank	Código numérico del banco beneficiario.
Account (To)	Código hexadecimal de la cuenta beneficiaria.
Amount Received	Monto recibido (en la moneda de la siguiente columna).
Receiving Currency	Moneda de recepción (dólares, euros, etc).
Amount Paid	Monto pagado (en la moneda de la siguiente columna).
Payment Currency	Moneda de pago (dólares, euros, etc).
Payment Format	Forma de pago: cheque, tarjeta de crédito, etc.
Is Laundering	Etiqueta de la transacción 0 si es legítima, 1 si es de LD.

Cuadro 1: Descripción de variables en el conjunto de datos.

2.2 Preprocesamiento de datos

Dado que los datos provienen de una simulación, no se tiene problemas con NAs o datos mal registrados. Se inicia entonces con la codificación de variables categóricas, la conversión de los montos en una moneda común (dólar estadounidense, USD) y el manejo de los datos desequilibrados.

En cuanto a la codificación de las variables categóricas, las variables *Currrency* y *Payment Format* se codificaron de acuerdon con los cuadros 2 y 3:

Como el número de registros es voluminoso y esto puede complicar la ejecución de las pruebas por temas de capacidad de cómputo, se opta por reducir el conjunto de datos de estudio, manteniendo

Código	Moneda
0	*Shekel*
1	*Brazil Real*
2	*Australian Dollar*
3	*Yen*
4	*Canadian Dollar*
5	*Saudi Riyal*
6	*Mexican Peso*
7	*UK Pound*
8	*US Dollar*
9	*Bitcoin*
10	*Euro*
11	*Yuan*
12	*Rupee*
13	*Swiss Franc*
14	*Ruble*

Cuadro 2: Codificación de monedas de transacción.

Código	Forma de pago
0	*ACH*
1	*Bitcoin*
2	*Cash*
3	*Cheque*
4	*Credit Card*
5	*Reinvestment*
6	*Wire*

Cuadro 3: Codificación de forma de pago.

aquellas transacciones donde los bancos involucrados (originadores o beneficiarios) hayan sido objetos para realizar LD. Con dicho tratamiento, ahora el conjunto de datos de estudio se compone de $1,147,470$ transacciones.

2.3 Análisis exploratorio de los datos

Se realiza el análisis descriptivo por variable más relevante, encontrando que, como se observa en la figura 1, las monedas más utilizadas son el dólar estadounidense y el euro, las formas de pago más utilizadas con el cheque y tarjeta de crédito y que el $75\,\%$ de las transacciones son por montos menores a $5,000$ USD.

2.4 Selección de variables

Se seleccionan las variables de acuerdo con los resultados del `Valor-F`, `Valor R de correlación`, `Umbral de varianza`, `Información Mutua` y `XGBoost`, los cuales son métodos estadísticos y algoritmos de selección basados en modelos.

De los análisis realizados y de acuerdo con el último modelo utilizado (`XGBoost`), la variable `PaymentFormat` es aquella que brinda más información que el resto de las variables para determinar la clasificación de

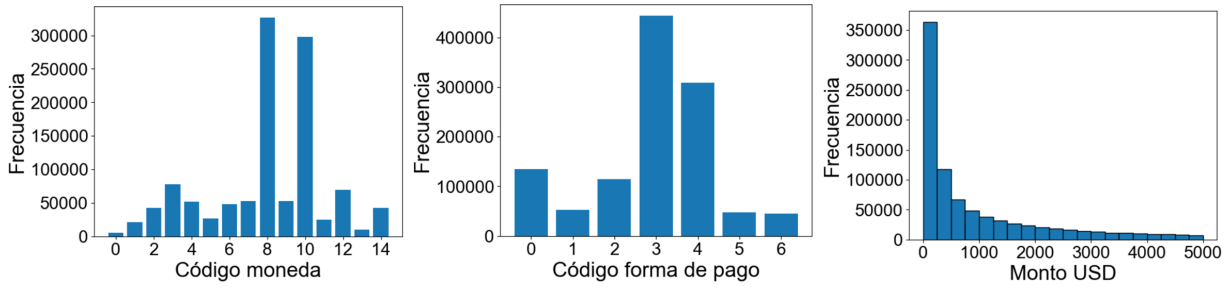

Figura 1: Distribución de variables.

la transacción. De igual manera, aunque en menor medida, las variables `MontoUSD , PCCY, From Bank` y `Timestamp` también aportan un poco más que el resto de las variables.

Asimismo, se considera la correlación (figura 2) que existe entre las variables, observando que la moneda y montos de recepción y de pago aportan información similar, por lo que los montos y la moneda de recepción se excluyen de las primeras pruebas del modelo, y en caso de que su desempeño sea muy malo se valora su integración en búsqueda de mejorarlo.

2.5 Análisis de agrupamiento

Se prueba el algoritmo `Mini Batch K-Means` al ser una variación del método de `K-Means`, pero adaptado para grandes conjuntos de datos, ya que utiliza ”batches” aleatorios de información para que puedan ser almacenados en la memoria reduciendo el tiempo de cómputo. Este algoritmo mantiene gran parte de las propiedades de `K-Means`, pero se ejecuta más rápido[3].

Para determinar el número óptimo de grupos, se utiliza el Índice de Davies-Bouldin (figura 3), una métrica que evalúa la calidad del agrupamiento al ser un indicador de cuán bien se separan los diferentes grupos y cuán compactos son internamente. Este índice mide la similitud promedio de cada grupo con su grupo más similar, donde la similitud es la relación entre las distancias dentro del grupo y las distancias entre grupos.

El objetivo es minimizar el índice de Davies-Bouldin. Un valor bajo implica que los grupos están bien separados (es decir, la distancia entre los centroides es grande) y que son internamente compactos (es decir, los puntos están cerca de su centroide), lo cual es deseable en un buen agrupamiento.

Para visualizar los grupos, es necesario reducir la dimensionalidad. Para esto se utiliza la técnica de Análisis de Componentes Principales (PCA por sus siglas en inglés), el cual tiene como objetivo conservar la mayor cantidad de información en menos variables, las cuales consideran una combinación lineal de las variables involucradas (figura 4).

2.6 Análisis de regresión y `boosting`

La regresión logística es utilizada para predecir el resultado de una variable categórica en función de las variables predictoras[4]. También se conoce como regresión logit, clasificación de máxima entropía (MaxEnt) o clasificador log-lineal y se enmarca dentro de los modelos denominados de predicción lineal generalizdos (GLM). En este modelo, las probabilidades que describen los posibles resultados de un único ensayo se modelan mediante una función logística.

Esta implementación puede adaptarse a regresión logística binaria, uno contra resto o multinomial con

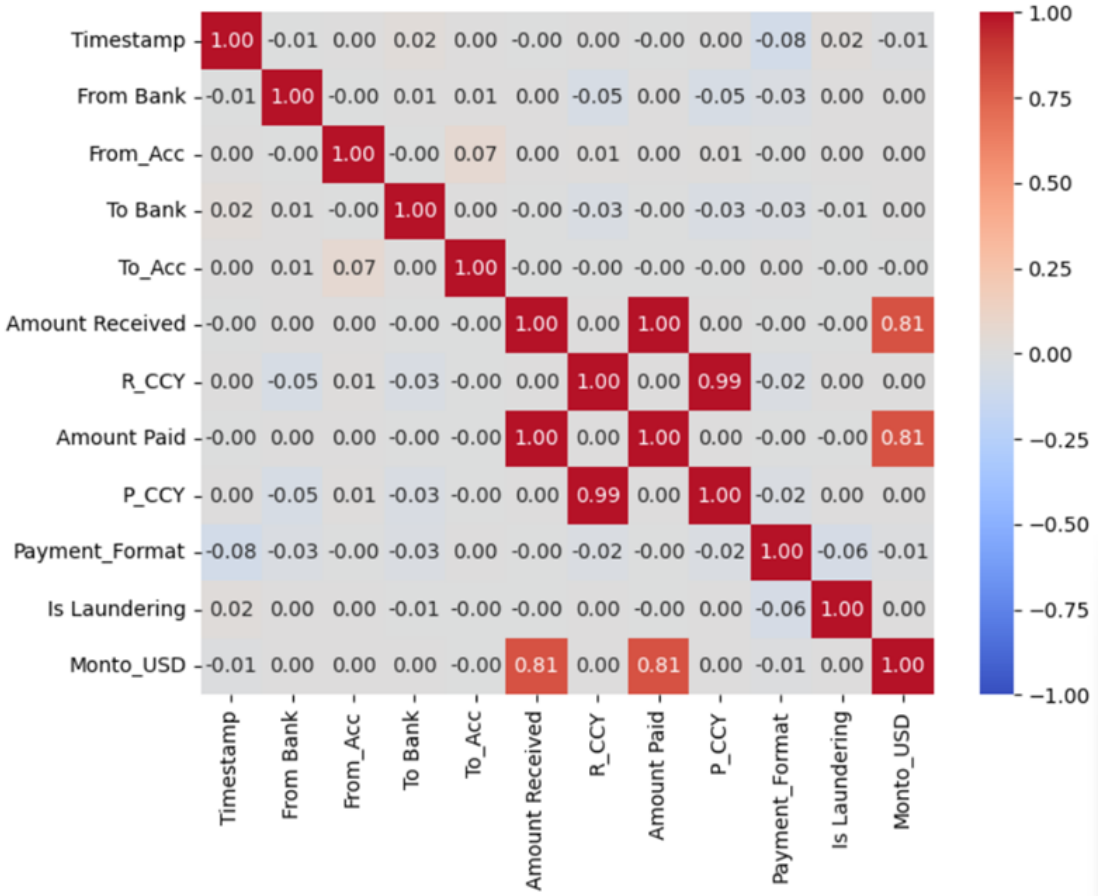

Figura 2: Matriz de correlación.

opción l_1 y l_2.

Para este caso, se requiere de la regresión logística binaria, ya que se maneja un conjunto de datos correspondientes a transacciones que cuentan con la clasificación de tratarse de operacioneos legítimas o no (variable binaria o dicotómica que toma valores 1 o 0).

La binomial es una distribución de probabilidad discreta que cuenta el número de éxitos en una secuencia de n ensayos. Si el evento de éxito tiene una probabilidad de ocurrencia p, la probabilidad del evento contrario -el de fracaso- tendrá una probabilidad de $q = 1 - p$. En la distribución binomial se repite el experimento n veces, de forma independiente, y se trata de calcular la probabilidad de un determinado número de éxitos d, en esas n repeticiones $B(n, p)$.

La denominación de logística se debe a la forma de la propia función de distribución de probabilidad binomial que presenta un crecimiento exponencial y que se parece a una S y que toma el nombre matemático de función logística $\frac{1}{1+e^{-t}}$.

Esta curva, es una aproximación continua a la función discreta binaria, pues el cambio de 0 a 1 se produce en corto espacio y muy pronunciado. Dado que la predicción se da en modo de probabilidad, se debe establecer qué umbral es el que fija el pronostico 0 o 1 (ej. $0{,}5$).

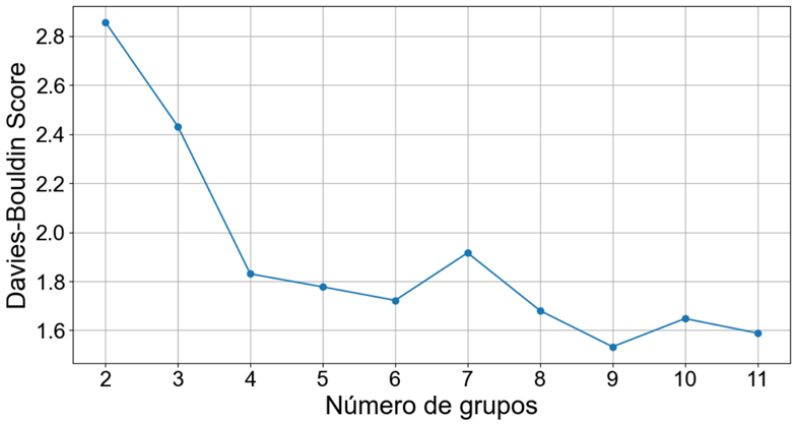

Figura 3: Índice de Davies-Bouldin por grupo.

2.6.1 Métricas de desempeño aplicables

Las métricas de desempeño son herramientas fundamentales en la evaluación de modelos de aprendizaje de máquina, ya que permiten medir la calidad y eficacia de un modelo en función de su capacidad para hacer predicciones precisas. Éstas son esenciales para tomar decisiones informadas sobre el modelo ideal para un problema y ajustarlo para obtener mejores resultados. Las métricas aplicables a los modelos utilizados son las siguientes:

Matriz de Confusión Aunque no es una métrica de error per se, proporciona una representación visual de los verdaderos positivos (TP), verdaderos negativos (TN), falsos positivos (FP), y falsos negativos (FN), los cuales a su vez son la base para muchas de las siguientes métricas[5].

Exactitud (Accuracy): Es la proporción de predicciones correctas sobre el total de casos. Aunque es la métrica más intuitiva, puede ser engañosa en conjuntos de datos desbalanceados donde una clase es mucho más frecuente que la otra.

$$\textit{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Error de Clasificación (Classification Error): También conocido como tasa de error, es la proporción de predicciones incorrectas respecto al total de predicciones. Se calcula como $1 - Exactitud$.

Precisión (Precision): Proporción de verdaderos positivas entre el total de positivos. Es una métrica importante cuando el costo de un falso positivo es alto:

$$\textit{Precision} = \frac{TP}{TP+FP}.$$

Sensibilidad (*Recall*) y **Especificidad (*Specificity*)**: De todos los casos positivos reales, cuántos fueron identificados correctamente por el modelo, mientras que la especificidad mide la proporción de negativos reales que se identificaron correctamente.

$$\textit{Recall} = \frac{TP}{TP+FN}$$

$$\textit{Specificity} = \frac{TN}{TN+FP}$$

Valor-F (*F-score*): Es el promedio armónico de precisión y sensibilidad. Es útil cuando se busca un balance entre precisión y sensibilidad, especialmente si la distribución de las clases es desigual.

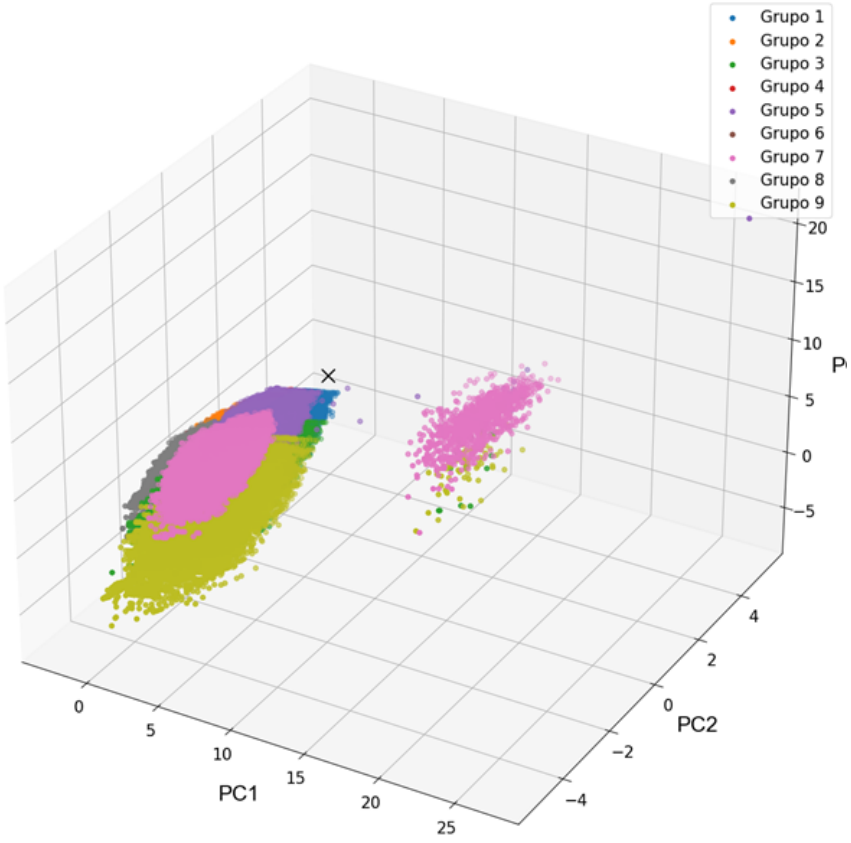

Figura 4: Grupos.

$$F1 = \frac{2*(precision*recall)}{(precision+recall)}$$

AUC - ROC (*Area Under the Curve and Receiver Operating Characteristic*): Mide la capacidad de un clasificador para distinguir entre clases y es usado como resumen de la curva ROC. Relaciona la tasa de verdaderos positivos con la tasa de falsos positivos para diferentes niveles del umbral[6].

2.6.2 Aplicación regresión logística binaria:

Se entrena y prueba el modelo de regresión logística binaria, resultando en una pobre clasificación de las transacciones, como se puede observar en la matriz de confusión de la figura 5.

2.6.3 Aplicación XGBoost

Xtreme Gradient Boosting (XGBoost) es un algoritmo de árbol de decisión que logra capturar complejidades de los datos. Mientras que los algoritmos de *Bagging* entrenan árboles de decisión en paralelo, los algoritmos de *Boosting* lo hacen en serie, tal que cada modelo se basa en los errores de su predecesor, intentando corregirlos. El modelo final es una colección de aprendizajes débiles entrenados con los residuos de aprendizajes fuertes para formar la predicción final.

Los resultados fueron un poco más favorables con XGBoost, como se observa en la matriz de confusión

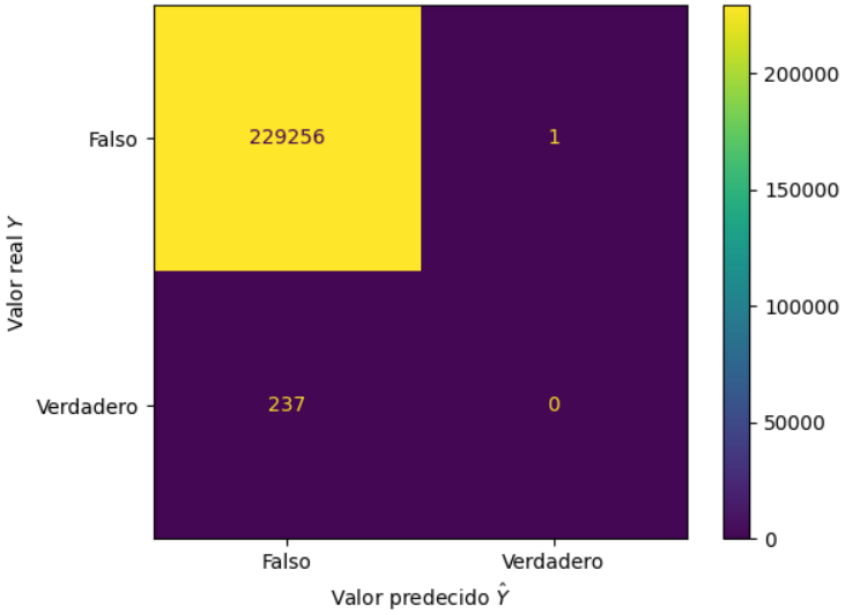

Figura 5: Matriz de confusión | Regresión logística

de la figura 6. Para dimensionar en mejor medida la mejora que representa, se utiliza el AUC visualizado en la figura 7.

2.7 Optimización de parámetros

Se busca la elección del modelo con el mejor rendimiento alcanzable. Para esto se somete a un proceso de optimización de parámetros, cuyo objetivo es encontrar la combinación de los hiperparámetros que le dan mejor ajuste al modelo. Se utiliza la función `GridSearchCV` de la librería `sklearn`[7] y se crea un diccionario de los parámetros que se buscan optimizar.

En este caso los hiperparmetros objetivo son: `max_depth` que hace referencia a la "Profundidad" o número de nodos de bifurcación de los árboles de decisión usados en el entrenamiento (aunque una mayor profundidad puede devolver mejores resultados, puede resultar en un sobre ajuste) [8] ; `n_estimators` que es el número de árboles que lleva a cabo el boosting (iteraciones) y `earning_rate` que reduce el tamaño de los pesos del algoritmo.

3 Resultados

De acuerdo con las medidas estadísticas utilizadas para la selección de variables, así como la correlación que existe entre ellas se opta por conservar 8 variables predictoras. Se experimenta con técnicas de agrupamiento y regresión logística; sin embargo, estos métodos no alcanzaron el nivel de eficacia esperado. En búsqueda de mejores resutlados se implementa un modelo de XGBoost, dada su robustez y capacidad para manejar la complejidad que representan las transacciones de posible LD.

La optimización de hiperparámetros del modelo XGBoost se centra en el ajuste de tres parámetros tal que se maximice la precisión o el área bajo la curva ROC (AUC).

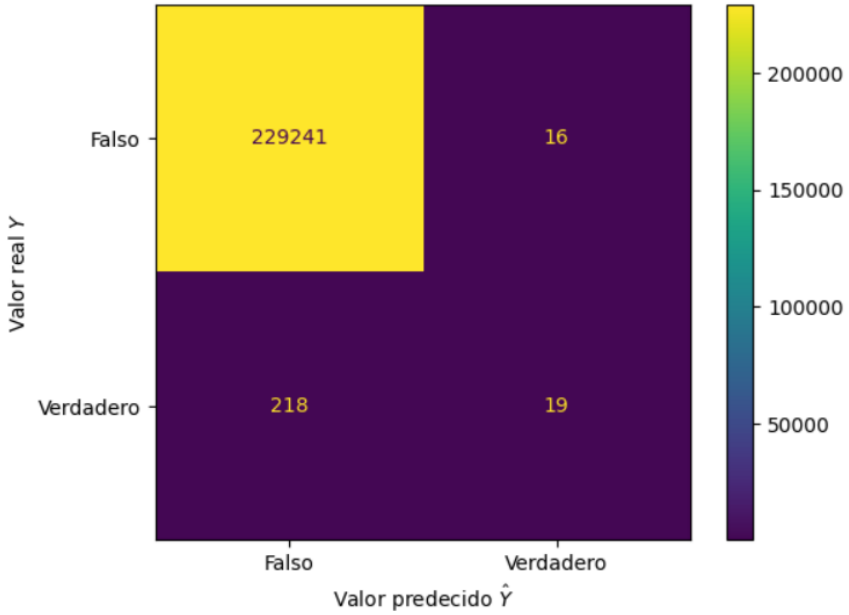

Figura 6: Matriz de confusión | XGBoost.

Maximizando la precisión del modelo, los hiperparámetros resultan de la siguiente manera: `learning_rate`: $0{,}2$, `max_depth`: 10, `n_estimators`: 200, los cuales logran aumentar su precisión del $99{,}90\,\%$ al $99{,}91\,\%$. Si bien esta mejora no representa gran cambio, su clasificación de verdaderos positivos subió 19 a 44, verdaderos negativos bajó de 16 a 12 y sus falsos negativos bajó de 218 a 193.

Maximizando el AUC, los hiperparámetros corresponden conforme lo siguiente: `learning_rate`: $0{,}1$, `max_depth`: 3, `n_estimators`: 200, los cuales logran aumentar el AUC del $0{,}97\,\%$ al $0{,}98\,\%$ en comparación con el de mejor precisión. Esta mejora tampoco representa gran cambio y rebota su desempeño de precisión.

Aunque existe una ligera diferencia en las métricas de los modelos resultantes, ambos reflejaron un desempeño sobresaliente. Este hallazgo resalta la eficacia del XGBoost en el conjunto de datos de análisis y destaca la importancia de elegir hiperparámetros que equilibren adecuadamente la precisión y el AUC, dependiendo de las prioridades específicas del estudio.

4 Conclusión

La aplicación del modelo XGBoost demuestra ser particularmente efectiva en el contexto de clasificar transacciones de aquellas potencialmente vinculadas a LD, destacándose sobre otros modelos como el agrupamiento y la regresión logística clásica.

La optimización de los hiperparámetros del modelo, enfocada en maximizar la precisión y el AUC, no resultó en cambios abismales entre las métricas de cada modelo, mostrando un equilibrio entre la capacidad que tiene para predecir con exactitud y para diferenciar eficientemente las transacciones. Este equilibrio es importante ya que la precisión y la capacidad de distinguir transacciones ilícitas son igualmente importantes.

Con este estudio se proporciona una metodología para la detección de actividades de lavado de dinero,

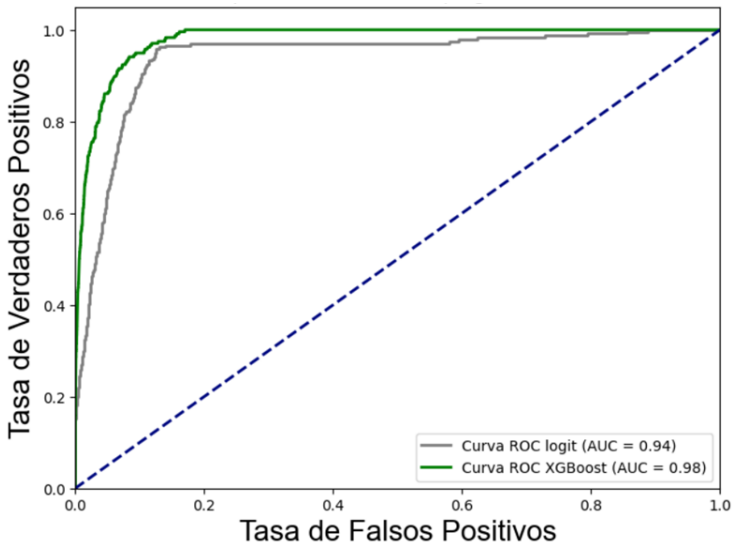

Figura 7: Comparativa de curvas ROC | Logit vs XGBoost.

pero también se establece un marco de referencia para futuras investigaciones en el campo del análisis financiero, particularmente en la lucha contra el crimen financiero. Aplicar técnicas avanzadas de aprendizaje de máquina y ampliar las variables de estudio pudiera robustecer la metodología para modelar de mejor manera las transacciones potencialmente ilícitas.

Referencias

[1] Comisión Nacional Bancaria y de Valores (CNBV), “Lavado de dinero.” [Online]. Available: https://www.gob.mx/cms/uploads/attachment/file/71151/VSPP_Lavado_de_Dinero___130701.pdf

[2] Noticias Organización de las Naciones Unidas (ONU). (2021) Recaudar el dinero lavado, el de la corrupción y el de la evasión de impuestos ayudaría a combatir el covid-19 y la crisis climática. [Online]. Available: https://news.un.org/es/story/2021/02/1488772

[3] GeeksforGeeks. (2023) Ml mini batch k means clustering algorithm. [Online]. Available: https://www.geeksforgeeks.org/ml-mini-batch-k-means-clustering-algorithm/

[4] Scikit-learn. Linear models. [Online]. Available: https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

[5] S. B. Data. (2023) Machine learning: Seleccion métricas de clasificación. [Online]. Available: https://sitiobigdata.com/2019/01/19/machine-learning-metrica-clasificacion-parte-3/

[6] B. for International Settlements. (2014) Evaluación de indicadores adelantados mediante el área auc. [Online]. Available: https://www.bis.org/publ/qtrpdf/r_qt1403z_es.htm#:~:text=El%20%C3%A1rea%20AUC%2C%20es%20decir,bien%20ocurren%20o%20no%20ocurren.

[7] scikit learn.org. sklearn.model_selection.gridsearchcv. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

[8] J. B. M. Vega. (2022) Tutorial: Xgboost en python. [Online]. Available: https://medium.com/@jboscomendoza/tutorial-xgboost-en-python-53e48fc58f73

[9] F. Sanz. (2022) Cómo funciona el algoritmo xgboost en python. [Online]. Available: https://www.themachinelearners.com/xgboost-python/