Evaluación y corrección metodológica de modelos de ensamble para la detección de fraude en tarjetas de crédito

Trabajo Final - Bases de Datos Masivas

Universidad Nacional de Luján

Francisco Guerra (182772) y Kevin Monti (182871)

Mayo 2026

Resumen

Este estudio analiza y corrige las fallas metodológicas de un modelo de ensamble multi-etapa (EIBMC) diseñado para la detección de fraude bajo condiciones de desbalance extremo. Se identifican errores críticos en el diseño original, tales como el Data Leakage derivado de aplicar balanceo y normalización de forma global, y el ajuste de parámetros utilizando datos que deberían permanecer aislados. Asimismo, se cuestiona el uso de la métrica AUC-ROC, la cual resulta engañosamente optimista en este tipo de escenarios. Se propone una reestructuración del flujo experimental que garantiza el aislamiento estricto de los datos de prueba antes de cualquier preprocesamiento. La metodología propuesta introduce una etapa de calibración isotónica para estandarizar las salidas probabilísticas de los modelos, y una optimización del ensamble orientada a maximizar el AUPRC. Los resultados obtenidos mediante la configuración IHT+XGBC muestran un F1-Score de 0,876 y un AUPRC de 0,869. Si bien estas métricas son inferiores a la perfección reportada en estudios previos, representan el rendimiento predictivo real y generalizable del modelo. Se concluye que el rigor metodológico y la correcta elección de métricas son indispensables para desarrollar sistemas de detección de fraude aplicables en entornos financieros reales.

1. Introducción

La detección automática de fraude en transacciones con tarjetas de crédito representa un problema relevante dentro del área de aprendizaje automático aplicado a sistemas financieros. Uno de los principales desafíos de este campo radica en el fuerte desbalance presente en los datos, donde las transacciones fraudulentas representan una fracción realmente pequeña del total de datos. Esta particularidad dificulta el entrenamiento de modelos de clasificación efectivos y vuelve necesaria la elección adecuada de estrategias de evaluación y validación. Si bien diversos trabajos abordan esta problemática mediante modelos de ensamble y técnicas de balanceo, se seleccionó para este análisis un enfoque destacado basado en un esquema multietapa [1]. Sin embargo, luego de un análisis detallado de su flujo experimental, se hallaron limitaciones metodológicas en la evaluación, como el énfasis en la métrica AUC, y la validación cruzada sobre datos previamente balanceados, que pueden producir estimaciones optimistas del rendimiento real del modelo. En este contexto, se propone un nuevo flujo que separa estrictamente los datos antes de aplicar cualquier balanceo, eliminando el riesgo de Data Leakage [3]. La metodología propuesta introduce una etapa de calibración isotónica [6] sobre un nuevo conjunto de validación, y una optimización de pesos del ensamble final orientada a la métrica de F1-Score, garantizando estimaciones de rendimiento robustas y representativas del escenario real de desbalance.

2. Trabajo Relacionado

2.1. Flujo de trabajo del artículo original

La metodología del trabajo original [1] comprende las fases de preprocesamiento de datos, aplicación de estrategias de balanceo, diseño de modelos de ensamble multietapa y la posterior evaluación de su desempeño. El conjunto de datos utilizado [9] está formado por 284.807 transacciones realizadas por titulares europeos durante dos días en septiembre de 2013. Este dataset presenta un desbalance crítico: solo se identificaron 492 fraudes, lo que representa el 0,172 % de la muestra total. Debido a restricciones de confidencialidad, la mayoría de las variables de entrada son numéricas y resultan de una transformación PCA. Las únicas características que mantienen su formato original son:

- Time: Segundos transcurridos desde la primera transacción del conjunto.

- Amount: Importe de la transacción.
- Class: Variable objetivo binaria (1 para fraude, 0 para transacciones legítimas).

A continuación, se describe la interpretación que se tomó del flujo de trabajo original, ilustrado en la Figura 1.

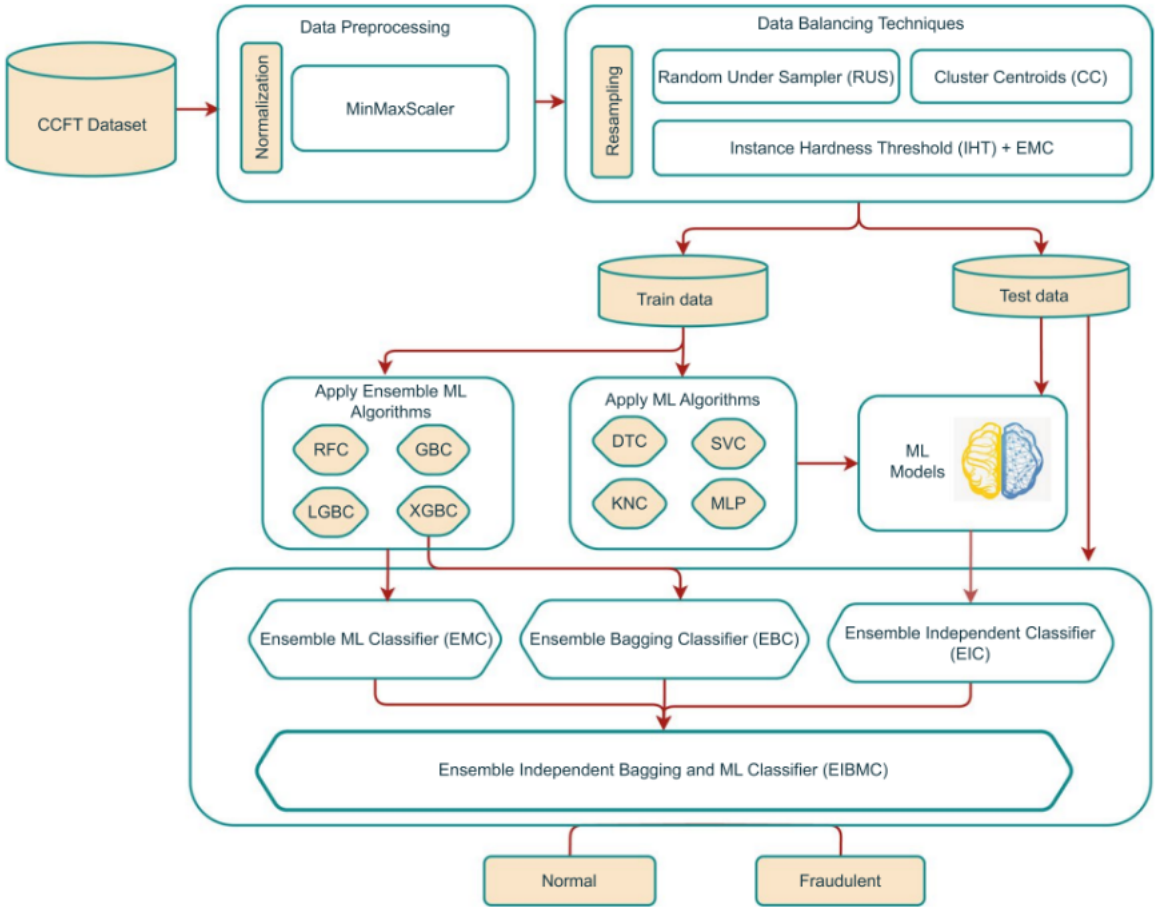

Figura 1: *Flujo del trabajo original.*

- Preprocesamiento: Aplicación de técnicas de escalado Min-Max para normalizar los datos en un rango acotado, asegurando que las características mantengan el mismo peso sin distorsionar sus distancias relativas.
- Estrategias para el balanceo de datos: Uso de diversos métodos de remuestreo para mitigar la tendencia de los algoritmos a favorecer la clase mayoritaria en condiciones de alto desbalance. Los métodos utilizados fueron:
  - Random Under Sampler (RUS)
  - Cluster Centroids (CC)

    - Instance Hardness Threshold (IHT), en combinación con un clasificador base para filtrar las observaciones más difíciles de predecir.

- División de datos: Partición del 10 % del dataset para el conjunto de evaluación, realizada de forma posterior al preprocesamiento y al balanceo global de los datos.

- Algoritmos de aprendizaje automático utilizados: Ocho modelos base, divididos entre clasificadores independientes y clasificadores de ensamble:

    - Decision Tree (DTC) [10]
    - Support Vector Classifier (SVC) [11]
    - K-Neighbors Classifier (KNC) [12]
    - Multilayer Perceptron (MLP) [13]
    - Random Forest Classifier (RFC) [14]
    - Gradient Boosting Classifier (GBC) [15]
    - Light Gradient Boosting Classifier (LGBC) [13]
    - XGBoost Classifier (XGBC) [16]

- Enfoque de ensamble multietapa: Se implementó un esquema de entrenamiento estructurado en tres etapas para el desarrollo del modelo final. En la primera fase, se entrenan ocho algoritmos de clasificación base (DTC, SVC, KNC, MLP, RFC, GBC, LGBC y XGBC). Posteriormente, estos modelos se integran en tres ensambles intermedios: el EIC (Ensemble Independent Classifier) mediante la votación de los clasificadores independientes (DTC, SVC, KNC y MLP); el EMC (Ensemble ML Classifier) que agrupa por el mismo método a RFC, GBC, LGBC y XGBC; y el EBC (Ensemble Bagging Classifier) basado en la técnica de Bagging sobre XGBC. Como etapa final, el modelo EIBMC (Ensemble Independent Bagging ML Classifier) integra a EIC, EMC y EBC a través de una votación ponderada, cuyos pesos se optimizan mediante búsqueda en cuadrícula para maximizar el rendimiento.

- Evaluación: Aplicación de validación cruzada estratificada de cinco particiones para asegurar que se mantenga la proporción de clases en cada iteración de entrenamiento y prueba.

2.2. Análisis de Resultados

La evaluación del modelo EIBMC consistió en seis experimentos, diferenciados por la técnica de balanceo aplicada. En las primeras cinco pruebas, utilizando los métodos RUS, CC e IHT junto a los algoritmos RFC, GBC y LGBC, el modelo mantuvo una Precision cercana al 94 % y un Accuracy del 99,9 %. El mejor rendimiento fue observado en el sexto experimento, mediante la combinación de IHT y XGBC, alcanzando un AUC del 100 %, un Accuracy del 99,84 % y un F1-Score del 99,52 %. El análisis de la matriz de confusión para este caso registró únicamente un falso positivo y un falso negativo.

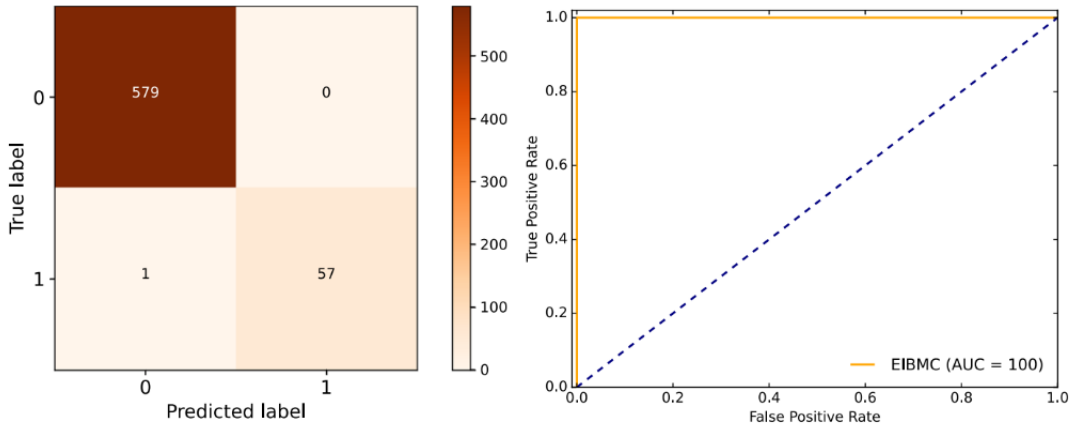

Figura 2: *Matriz de confusión y Curva ROC del modelo IHT+XGBC*

3. Limitaciones y fallas del flujo de trabajo original

3.1. Evaluación basada principalmente en AUC

Si bien el Área Bajo la Curva ROC (AUC-ROC) es una métrica de uso estándar para la clasificación binaria, su uso en contextos de desbalance extremo requiere criterio. La curva ROC cuantifica la capacidad global del modelo trazando la tasa de verdaderos positivos frente a la tasa de falsos positivos (FPR) en distintos umbrales. En este problema de detección de fraude, esta métrica puede ofrecer una perspectiva engañosa debido a la naturaleza del cálculo de la FPR:

$$\left(FPR = \frac{FP}{FP + TN} \right)$$

Dado que la clase mayoritaria (Verdaderos Negativos, TN) es masiva, un aumento sustancial en los Falsos Positivos (FP) apenas altera el denominador, manteniendo el FPR bajo y, en

consecuencia, un AUC-ROC artificialmente elevado [2].

Para obtener una evaluación realista ante un desbalance extremo, el enfoque estándar recomienda utilizar el Área Bajo la Curva de Precisión-Exhaustividad (AUPRC). A diferencia de la curva ROC, la curva PR evalúa la Precisión frente a la Exhaustividad:

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

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

Al excluir los Verdaderos Negativos (TN) de ambas fórmulas, el AUPRC se vuelve sensible a los errores cometidos sobre la clase minoritaria. Exigir una puntuación alta en AUPRC es la forma más robusta de garantizar que el modelo detecta el fraude de manera efectiva sin incurrir en una tasa elevada de falsos positivos.

3.2. Data Leakage

Otro aspecto crítico observado es el diseño de su esquema de evaluación, el cual presenta fallas estructurales que comprometen la validez de los resultados reportados. El procedimiento aplica normalización y balanceo sobre el dataset completo de forma previa a la división en conjuntos de entrenamiento y prueba. En la literatura, este enfoque se documenta como una fuente severa de Data Leakage [17], ya que el modelo se evalúa sobre un conjunto de prueba artificialmente equilibrado y normalizado con información global. Esto destruye la probabilidad a priori del escenario real y contamina el entrenamiento al filtrar parámetros estadísticos del conjunto de prueba, produciendo estimaciones de rendimiento inválidas para un entorno de producción.

A este sesgo se suma la contaminación del conjunto de prueba. El algoritmo de optimización del modelo EIBMC afina sus pesos internos iterando y midiendo la exactitud directamente sobre el conjunto de evaluación. Utilizar el conjunto de prueba para el ajuste de parámetros introduce un sesgo de selección (selection bias), convirtiéndolo en una extensión del conjunto de validación y violando la regla de independencia de los datos. Como consecuencia, el modelo resulta ajustado a un entorno estadísticamente irreal; las métricas reportadas son optimistas y no reflejan la capacidad de generalización frente a datos independientes en producción.

3.3. Conjunto de evaluación relativamente pequeño

El diseño original destina un 10 % de los datos al conjunto de evaluación. Más allá del porcentaje relativo, el factor crítico en este contexto es la cantidad absoluta de instancias de la clase minoritaria evaluadas [4]. Según la matriz de confusión reportada para el modelo EIBMC final, el conjunto de prueba cuenta con apenas 58 instancias fraudulentas. Esta escasez de casos positivos resta relevancia estadística a la evaluación, impidiendo una estimación fiable del rendimiento real del modelo. La literatura especializada señala que las pruebas con menos de 75 a 100 observaciones minoritarias producen resultados dominados por la incertidumbre aleatoria, donde variaciones fortuitas en los aciertos o errores alteran drásticamente las métricas finales [5].

3.4. Evaluación Empírica del Impacto Metodológico

Con el objetivo de verificar la validez de las observaciones anteriores, se decidió recrear el flujo de trabajo original, conservando los errores metodológicos discutidos.

3.4.1. Recreación del flujo original

Se recreó el flujo de trabajo del artículo original para el experimento que utiliza la técnica de balanceo IHT+XGBC, conservando el preprocesamiento global (escalado) y el balanceo previo a la partición. La única alteración introducida fue ampliar el conjunto de evaluación de un 10 % a un 20 % (garantizando así una muestra cercana a los 100 casos fraudulentos) para incluir en la evaluación mayor cantidad de fraudes, y poder comparar con otros experimentos.

Las métricas obtenidas de la evaluación fueron:

AUC	F1-Score	Recall	Precision	Accuracy
0,991	0,901	0,836	0,976	0,998

El modelo obtuvo métricas excesivamente optimistas, pero aún así, replicando los errores metodológicos, es incapaz de alcanzar las métricas tan altas como las publicadas en el artículo original (F1-Score = 0,99). Esto evidencia que los resultados reportados por los autores fueron producto de un sobreajuste extremo (overfitting), probablemente por ajustar los pesos del modelo final sobre un conjunto de prueba tan pequeño.

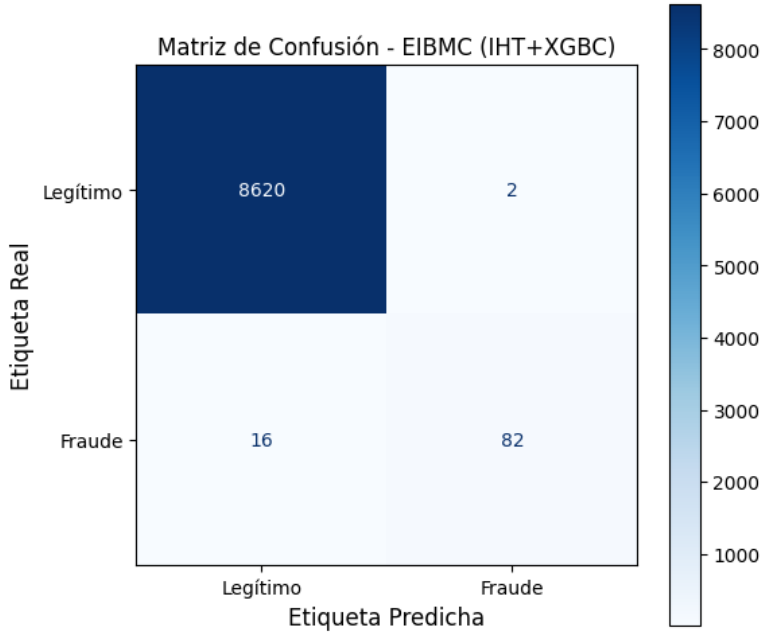

Figura 3: *Matriz de confusión del modelo EIBMC (IHT+XGBC) replicando los sesgos metodológicos.*

3.4.2. Corrección Metodológica

Posteriormente, se realizó un experimento manteniendo la misma estructura algorítmica, pero aplicando un esquema de partición de datos más riguroso (80 % entrenamiento y 20 % test). En este diseño, el conjunto de prueba se mantuvo aislado. Sobre el conjunto de entrenamiento se aplicó validación cruzada estratificada de 5 particiones (5-fold cross-validation), y dentro de cada ciclo el escalado y balanceo se ajustaron únicamente sobre la partición de entrenamiento. Por último, la optimización de pesos del modelo final se realizó sobre la partición de validación interna.

Con estos cambios, las métricas obtenidas fueron las siguientes:

AUC	F1-Score	Recall	Precision	Accuracy
0,978	0,866	0,796	0,951	0,998

Los resultados de este nuevo experimento con las correcciones metodológicas, muestran un descenso en las métricas, tal como se había previsto. El F1-Score es ahora 0,866; y el Recall disminuyó a 0,796. En consecuencia, si bien el modelo ensamble final posee capacidad predictiva, este experimento evidencia empíricamente que la perfección reportada en el estudio original

consistía en una ilusión estadística derivada de una metodología inválida.

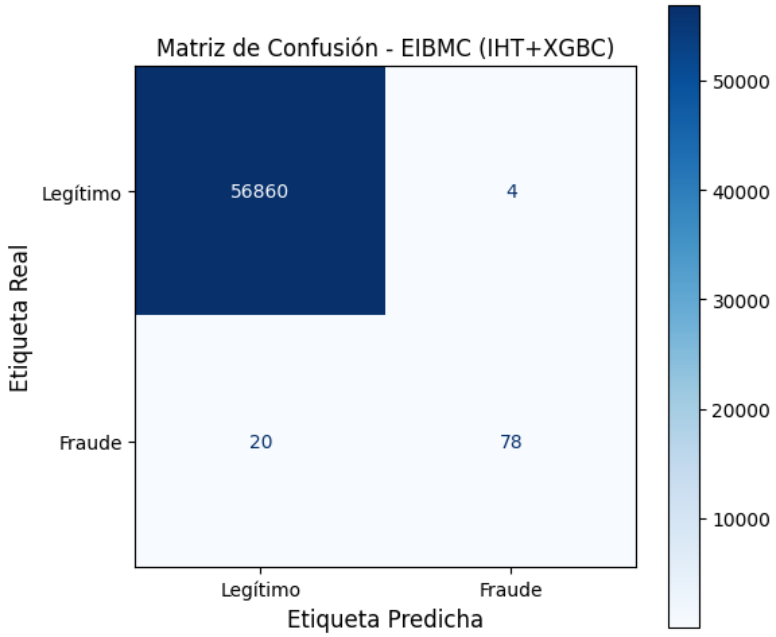

Figura 4: *Matriz de confusión del modelo EIBMC (IHT+XGBC) bajo experimento con conjunto de prueba aislado correctamente.*

4. Objetivos y metodología propuesta

Para superar las severas limitaciones metodológicas detalladas en la sección anterior, se propone una reestructuración completa del flujo experimental. El propósito central es construir un modelo de detección de transacciones fraudulentas riguroso, libre de errores metodológicos y del riesgo de filtración de información. Mediante el correcto aislamiento del conjunto de prueba, se busca obtener una estimación imparcial, sólida y realista del desempeño predictivo del modelo frente a datos no adulterados.

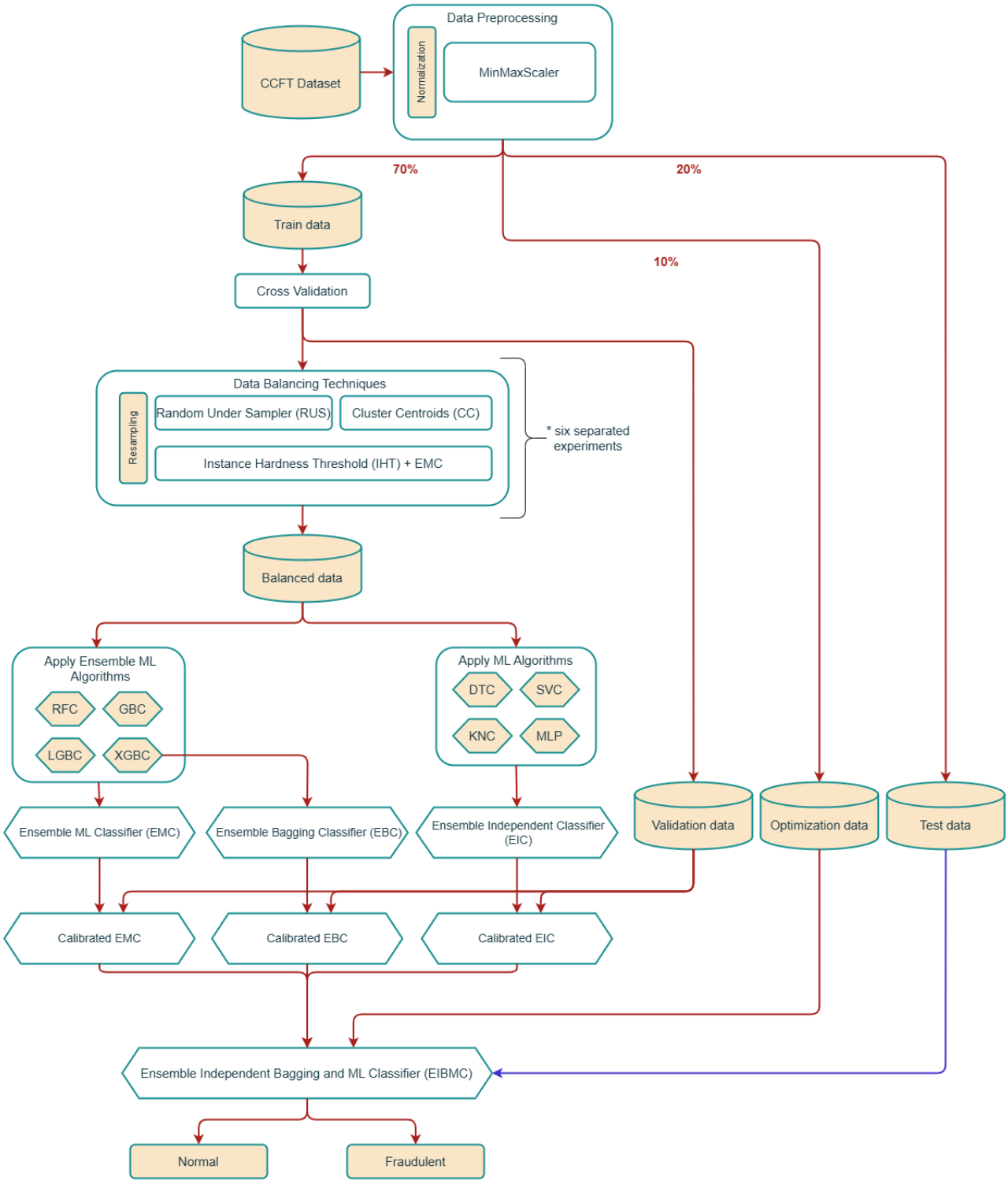

Figura 5: *Diagrama del flujo de trabajo propuesto.*

4.1. División de datos

Con el objetivo de evitar el data leakage y sobreajuste discutido en la sección 3.2, se realiza la división de datos antes de aplicar cualquier técnica de preprocesamiento. De esta manera,

el dataset original, altamente desbalanceado, se divide estratificadamente en tres subconjuntos para mantener la proporción de las clases en cada uno:

- Conjunto de Entrenamiento (70 %): Utilizado exclusivamentepara aprender los parámetros de escalado, aplicar el balanceo de clases y entrenar los algoritmos de clasificación.
- Conjunto de Optimización (10 %): Conjunto intermedio que mantiene el desbalance original, y es crucial en la optimización de los pesos de los modelos de ensamble.
- Conjunto de Prueba (20 %): Conjunto que se mantiene aislado e inalterado hasta la etapa final. Asegura que la evaluación del modelo refleje el verdadero rendimiento en su entorno de producción.

4.2. Estrategia de entrenamiento

Para la fase de entrenamiento, se emplearon los ocho algoritmos base y los hiperparámetros óptimos del estudio original para aislar el impacto de las correcciones metodológicas, garantizando que los cambios en el rendimiento no se deban a una parametrización distinta. La arquitectura utiliza tres ensambles intermedios: EIC y EMC, construidos mediante votación suave con pesos uniformes, y EBC, implementado con un esquema de Bagging sobre estimadores XGBC. El sistema se entrenó mediante validación cruzada estratificada de 5 particiones (o 5-fold cross-validation). Para evitar la fuga de información, los parámetros de normalización y el algoritmo de submuestreo se calcularon y aplicaron exclusivamente sobre la partición de entrenamiento de cada iteración. La partición de validación permaneció sin balancear para conservar la distribución natural de las clases . La obtención de predicciones sobre el conjunto de prueba se realizó bajo un enfoque de Ensamblado de Validación Cruzada (Cross-Validation Ensembling). Las predicciones finales resultan del promedio probabilístico de los cinco modelos independientes generados, lo que maximiza la robustez estadística y asegura una estimación de rendimiento estable para la puesta en producción.

4.3. Calibración Isotónica Probabilística

Varios algoritmos de machine learning, como las Máquinas de Vectores de Soporte (SVM) y los modelos de ensamble basados en árboles (Random Forest o XGBoost), no generan estimaciones probabilísticas precisas de forma nativa. En su lugar, producen puntuaciones de confianza o márgenes de decisión que se desvían de las probabilidades estadísticas reales [6].

Dado que la arquitectura propuesta integra tres ensambles heterogéneos (EIC, EMC, EBC) mediante una votación ponderada final, es matemáticamente indispensable que las predicciones de todos los componentes operen bajo una escala probabilística estandarizada y coherente. Combinar puntuaciones de modelos no calibrados resultaría en un comportamiento impredecible del modelo integrador. Para resolver esto, se introdujo una etapa de Calibración Isotónica. La curva de calibración se ajusta exclusivamente sobre la partición de validación en cada iteración de la validación cruzada. El uso de este conjunto permite que la función isotónica mapee las puntuaciones brutas de los modelos hacia probabilidades reales, basándose en la frecuencia observada de fraudes y respetando el desbalance natural de los datos.

4.4. Construcción del Modelo Final (EIBMC) y Búsqueda de Pesos Óptimos

El modelo integrador final (EIBMC) calcula sus predicciones a partir de una votación ponderada de los tres ensambles calibrados (EIC, EMC, EBC). En escenarios de desbalance extremo, métricas como el ROC-AUC y la Exactitud pueden resultar engañosas, por lo que se diseñó una búsqueda de pesos iterativa sobre el conjunto de optimización para maximizar la métrica AUPRC. Optimizar esta métrica garantiza el mejor equilibrio posible entre Precision y Recall, maximizando la detección de fraudes reales sin incurrir en una alta tasa de falsas alarmas.

4.5. Evaluación

La evaluación cuantitativa del modelo se fundamentó en un conjunto de métricas estándar de clasificación binaria. Todas las mediciones discretas se derivaron analíticamente a partir de las frecuencias observadas en la matriz de confusión, definida por los Verdaderos Positivos (TP), Verdaderos Negativos (TN), Falsos Positivos (FP) y Falsos Negativos (FN). Las métricas implementadas se calculan de la siguiente manera:

- Precisión (*Precision*): Define la proporción de predicciones positivas correctas respecto al total de clasificaciones positivas emitidas por el modelo.

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

- Exhaustividad (*Recall*): Cuantifica la proporción de instancias de la clase positiva real

que fueron correctamente identificadas.

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

- Exactitud (*Accuracy*): Representa la proporción global de clasificaciones correctas respecto al tamaño total del conjunto de datos evaluado.

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

- F1-Score: Se define como la media armónica entre la Precisión y la Exhaustividad, penalizando los valores extremos al otorgar mayor peso al valor más bajo de ambas proporciones.

$$F1\text{-}Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

- Área Bajo la Curva de Precisión-Exhaustividad (*AUPRC*): Seleccionada como la métrica principal para la optimización de pesos del ensamble y la evaluación del rendimiento. A diferencia del ROC-AUC, el AUPRC es altamente sensible a las mejoras en la clase minoritaria. La literatura especializada establece que aproximar esta integral utilizando la regla del trapecio convencional introduce un sesgo geométrico sistemáticamente optimista, dado que asume una interpolación lineal incorrecta entre los puntos operativos en el espacio PR [7]. Para mitigar esta sobreestimación, la métrica se calculó mediante la Precisión Media (*Average Precision*, AP), la cual realiza una integración discreta escalonada, definida matemáticamente como:

$$AP = \sum_{n} (R_n - R_{n-1}) P_n$$

donde P_n y R_n representan la precisión y la exhaustividad evaluadas en el n-ésimo umbral operativo.

5. Experimentos

Se ejecutaron 6 experimentos principales utilizando cada una de las técnicas de balanceo propuestas. Los resultados obtenidos en el conjunto de prueba (20 % del total de datos) demuestran la superioridad del modelo integrado EIBMC frente a sus componentes individuales.

5.1. Resultados

La evaluación de los seis experimentos determinó que la combinación IHT+XGBC es la estrategia de balanceo más sólida para el modelo EIBMC. En el escenario de desbalance extremo analizado, esta configuración alcanzó el mejor desempeño con un F1-Score de 0,876. El resto de las métricas obtenidas fueron: AUPRC de 0,869; Recall de 0,827; Precision de 0,931 y Accuracy de 1,000.

Este modelo destaca por su alta Precisión, manteniendo un comportamiento conservador ante las falsas alarmas. El análisis de la matriz de confusión confirma esta eficacia: se identificaron 81 casos de fraude real (Verdaderos Positivos) con solo 6 Falsos Positivos sobre un volumen total de 56.858 transacciones legítimas.

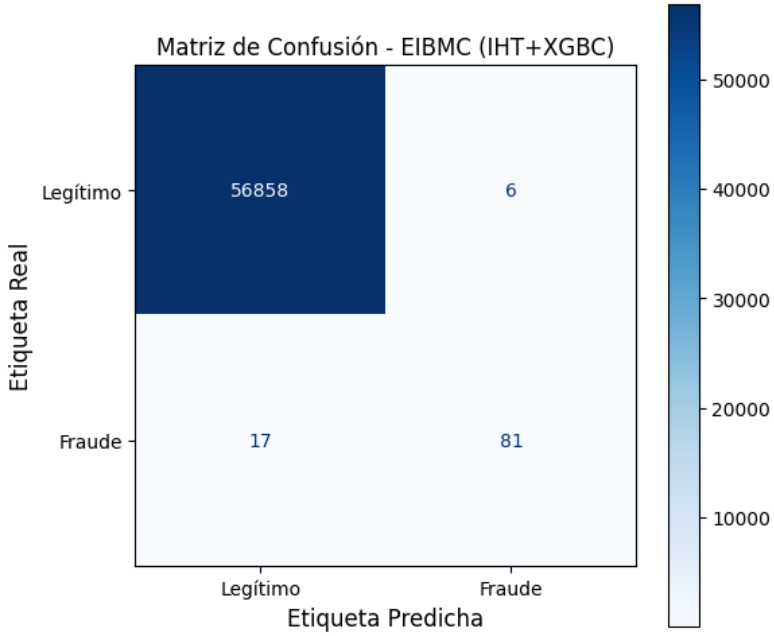

Figura 6: *Matriz de confusión del modelo EIBMC (IHT+XGBC) del flujo de trabajo propuesto.*

Model	AUPRC	F1-Score	Recall	Precision	Accuracy
DTC	0,816	0,851	0,816	0,889	0,100
SVC	0,766	0,796	0,816	0,777	0,999
KNC	0,829	0,851	0,786	0,928	1,000
MLP	0,718	0,682	0,592	0,806	0,999
RFC	0,867	0,863	0,806	0,929	1,000
GBC	0,800	0,754	0,704	0,812	0,999
LGBC	0,648	0,597	0,469	0,821	0,999
XGBC	0,755	0,823	0,735	0,935	0,999
EBC	0,862	0,865	0,816	0,920	1,000
EIC	0,831	0,813	0,755	0,881	0,999
EMC	0,863	0,843	0,796	0,897	0,999
EIBMC	**0,869**	**0,876**	**0,827**	**0,931**	**1,000**

Tabla 1: *Métricas de evaluación del experimento IHT+XGBC.*

Los resultados de la totalidad de los experimentos pueden encontrarse en la Tabla 5, en el apartado de anexo.

5.2. Curvas de calibración

Se comparó el desempeño de los modelos de ensamble (EIC, EMC y EBC) antes y después de la calibración isotónica. Las curvas resultantes no presentan una mejora morfológica drástica, conservando un comportamiento oscilatorio y una estructura escalonada en los rangos de probabilidad media y alta. Este fenómeno se atribuye al desbalance extremo de las clases, que provoca que el número de muestras en los deciles de probabilidad superior sea estadísticamente bajo en comparación con el primer decil. La persistencia de estos escalones es una característica intrínseca de la regresión isotónica ante la escasez de instancias de la clase minoritaria. A pesar de que la mejora visual es marginal, la calibración cumple la función técnica de normalizar los rangos de salida para el procesamiento posterior en el ensamble EIBMC.

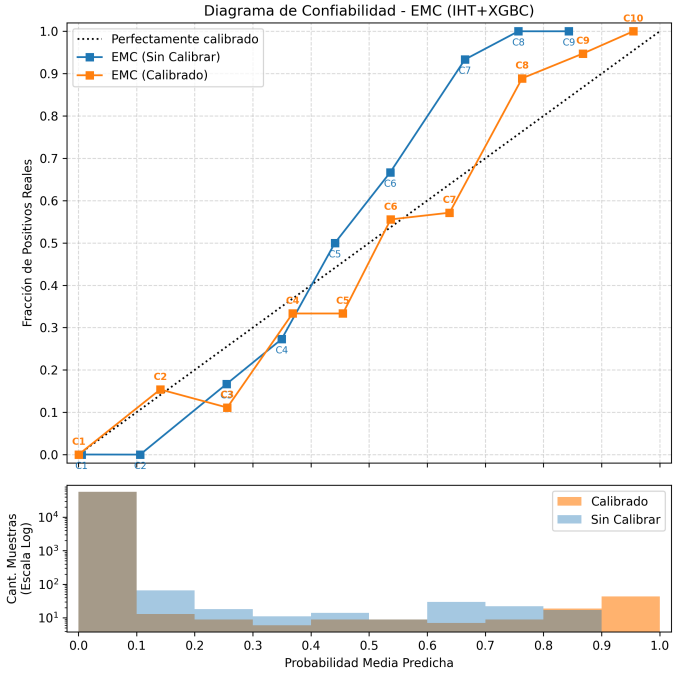

Figura 7: *Curva de calibración del modelo ensemble EMC (IHT+XGBC) comparando sus probabilidades antes y después de la calibración.*

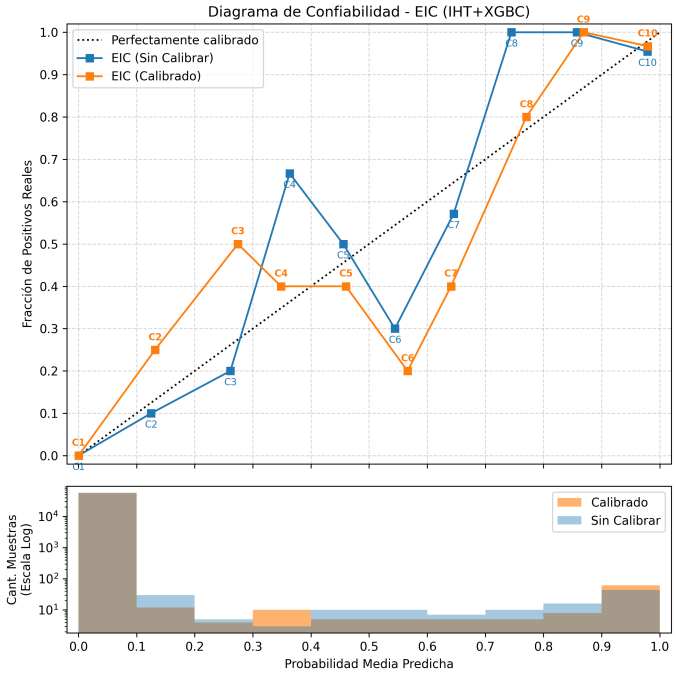

Figura 8: *Curva de calibración del modelo ensemble EIC (IHT+XGBC) comparando sus probabilidades antes y después de la calibración.*

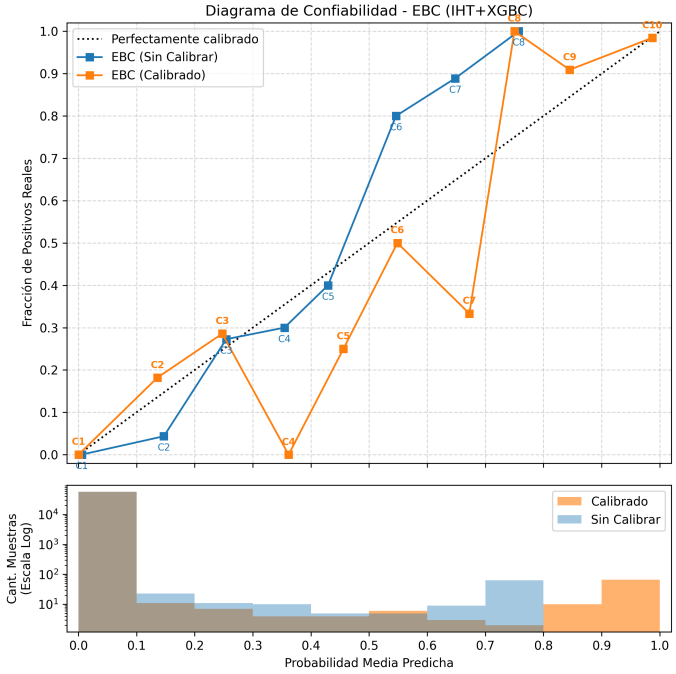

Figura 9: *Curva de calibración del modelo ensemble EBC (IHT+XGBC) comparando sus probabilidades antes y después de la calibración.*

5.3. Análisis de las técnicas de balanceo

Con el fin de analizar las distintas técnicas de balanceo, se decidió medir las instancias resultantes luego de aplicar cada una de ellas, y calcular el porcentaje de reducción sobre la cantidad original de instancias legítimas del conjunto de datos.

Experimento	**Instancias totales**	**Legítimas**	**Fraudes**	**Porcentaje de reducción**
ORIGINAL (Sin Balanceo)	199364	199019	345	0,00 %
RUS	690	345	345	99,83 %
CC	690	345	345	99,83 %
IHT+RFC	194647	194302	345	2,37 %
IHT+GBC	40144	39799	345	80,00 %
IHT+LGBC	124861	124516	345	37,44 %
IHT+XGBC	58093	57748	345	70,98 %

Tabla 2: *Comparativa de reducción, según la estrategia de balanceo aplicada.*

Las métricas de reducción de instancias y distribución de clases presentadas en esta tabla ilustran el impacto global de cada algoritmo sobre la totalidad del conjunto de entrenamiento (previamente escalado). En el flujo real, para garantizar la ausencia de filtración de datos), el submuestreo no se aplicó de forma global, sino de manera dinámica e independiente exclusivamente dentro de cada partición (fold) de entrenamiento durante la validación cruzada.

6. Aplicación del flujo de trabajo propuesto sobre otro conjunto de datos con desbalance de clases

Con el objetivo de comprobar si el flujo de trabajo propuesto es robusto frente a diversos conjuntos de datos que también presentan un desbalance de clases binarias, se replicaron los experimentos sobre el conjunto de datos *Intrusion detection evaluation dataset (CIC-IDS2017)* [8].

Es importante destacar que, para este segundo caso de estudio, no se partió de los archivos de tráfico de red en crudo, sino que se utilizó una versión unificada y procesada. Para garantizar la reproducibilidad y centrar el análisis en la arquitectura de ensamble, se optó por una variante consolidada y binarizada (tráfico legítimo vs. ataques).

Adicionalmente, dado el volumen masivo del conjunto de datos original, se aplicó una técnica de de submuestreo aleatorio estratificado para extraer una muestra estadísticamente representativa, que se asemeja al volumen total de instancias del experimento de fraude financiero (alrededor de 280.000 registros).

6.1. Resultados y evaluación del modelo EIBMC

El experimento que mejores resultados obtuvo en el modelo final EIBMC sobre este nuevo dominio fue el que utilizó la estrategia de balanceo RUS. Como se observa en las métricas de evaluación de la Tabla 3, se obtuvo un F1-Score de 0,956 y un AUPRC de 0,995. Esto demuestra empíricamente que el flujo de trabajo calibrado se puede aplicar en otros conjuntos de datos sin perder desempeño predictivo.

Model	AUPRC	F1-Score	Recall	Precision	Accuracy
DTC	0,966	0,955	0,990	0,923	0,984
SVC	0,946	0,796	0,984	0,668	0,915
KNC	0,987	0,939	0,993	0,891	0,978
MLP	0,973	0,869	0,987	0,776	0,950
RFC	0,996	0,955	0,994	0,919	0,984
GBC	0,988	0,926	0,989	0,870	0,973
LGBC	0,994	0,946	0,995	0,901	0,981
XGBC	0,994	0,947	0,996	0,902	0,981
EBC	0,994	0,955	0,972	0,939	0,985
EIC	0,992	0,954	0,979	0,929	0,984
EMC	0,995	0,956	0,972	0,940	0,985
EIBMC	**0,995**	**0,956**	**0,973**	**0,940**	**0,985**

Tabla 3: *Métricas de evaluación del experimento RUS*

6.2. Análisis de las técnicas de balanceo y discusión general

Para comprender a fondo por qué una técnica de submuestreo aleatorio puro (RUS) superó a los métodos inteligentes en este escenario, resulta fundamental analizar el impacto de cada técnica de balanceo sobre la reducción del conjunto de entrenamiento.

Experimento	Instancias totales	Legítimas	Fraudes	Porcentaje de reducción
ORIGINAL (Sin Balanceo)	176452	146654	29798	0,00 %
RUS	59596	29798	29798	79,68 %
CC	59596	29798	29798	79,68 %
IHT+RFC	166543	136745	29798	6,76 %
IHT+GBC	70943	41145	29798	71,94 %
IHT+LGBC	59604	29806	29798	79,68 %
IHT+XGBC	59994	30196	29798	79,41 %

Tabla 4: *Comparativa de reducción, según la estrategia de balanceo aplicada.*

El comportamiento observado en la Tabla 4 revela que la eficacia del submuestreo no depende solo de la proporción del desbalance, sino del volumen absoluto de la clase minoritaria.

A diferencia del primer escenario de fraude financiero, donde la aplicación de RUS redujo el conjunto a 690 instancias, en el dataset CIC-IDS2017 la clase minoritaria cuenta con 29.798 muestras. Esto permite que una técnica de reducción extrema y aleatoria como RUS conserve un total de 59.596 instancias totales. Este volumen de datos resulta lo suficientemente robusto para que el modelo EIBMC logre converger y capturar los casos de decisión complejos.

Además, al analizar la técnica IHT en combinación con GBC, LGBC, y XBGC en ambos escenarios, se observa que el nivel de reducción es consistente: el algoritmo elimina entre el 70 % y el 80 % de la clase mayoritaria tanto en el dataset de fraude como en el de intrusión. Si bien en este segundo dominio las clases convergen a una proporción cercana a 1:1, no se debe a que el filtro sea más agresivo, sino a que el volumen absoluto de la clase minoritaria es lo suficientemente grande como para equipararlo con el de las instancias legítimas.

En conclusión, el flujo de trabajo propuesto demostró una gran capacidad de adaptación y generalización. Los resultados evidencian que la elección de la técnica de balanceo depende del volumen de los datos: RUS resulta eficiente cuando contamos con una cantidad abundante de casos minoritarios. Pero en contextos de escasez extrema, es mejor recurrir a métodos como IHT. En estos escenarios más críticos, preservar la forma o estructura original de la clase mayoritaria es importante para que los modelos de ensamble funcionen correctamente.

7. Conclusión

En el presente trabajo se abordó el complejo desafío de la detección de fraudes en tarjetas de crédito bajo el escenario de desbalance extremo de clases, tomando como caso de estudio un artículo publicado por *Talukder et al.* [1], donde plantean un modelo de ensamble multietapa. A través de un análisis empírico y metodológico, este estudio demostró cómo la obtención de métricas de rendimiento aparentemente perfectas (como un AUC-ROC del 100 %, o un F1-Score del 99,52 %) puede ser resultado de fallas metodológicas en el diseño del flujo de trabajo.

La replicación y análisis del flujo de trabajo original realizado en esta investigación revelaron errores críticos que comprometen la validez de sus resultados. En primer lugar, la aplicación de técnicas de preprocesamiento y submuestreo de manera global, antes de la división de los datos, introdujo una fuga de información que afectó el entrenamiento, introduciendo parámetros

estadísticos del conjunto de prueba.

Por otro lado, la optimización de los pesos del modelo final sobre el conjunto de evaluación generó un sesgo de selección y un sobreajuste extremo, violando el principio de independencia de los datos. A esto se sumó el uso de un conjunto de prueba estadísticamente insignificante (con apenas 58 fraudes) y la dependencia del Área Bajo la Curva ROC, una métrica que resulta engañosamente optimista en contextos de alto desbalance.

Para subsanar estas deficiencias, se propuso e implementó una reestructuración metodológica completa. El nuevo flujo de trabajo aisló de manera estricta el conjunto de prueba (20 %) desde el inicio, garantizando una estimación imparcial del rendimiento. Además, se reemplazó la dependencia de la métrica AUC-ROC por el Área Bajo la Curva de Precisión-Exhaustividad (AUPRC); y se introdujo una etapa de Calibración Isotónica, corrigiendo las distorsiones probabilísticas inherentes a los modelos de ensamble.

Al someter el esquema de ensamble multietapa a la evaluación (específicamente utilizando la estrategia IHT+XGBC, que demostró ser la más sólida), el modelo alcanzó un F1-Score de 0,876 y un AUPRC de 0,869.

Si bien estas métricas evidencian una caída sustancial frente a la perfección reportada en el estudio original, representan el rendimiento predictivo real y generalizable del modelo. Los resultados demuestran que el modelo integrado (EIBMC) sigue teniendo una alta capacidad operativa: logró detectar 81 transacciones fraudulentas cometiendo tan solo 6 falsos positivos sobre un volumen de más de 56.000 transacciones legítimas.

Adicionalmente, la metodología y el flujo de trabajo propuestos demostraron una alta capacidad de generalización al ser replicados con éxito en un dominio distinto, como el de detección de intrusiones de red. Este segundo caso de estudio validó la robustez de la arquitectura al mantener un desempeño predictivo y permitió establecer empíricamente que la elección óptima de la técnica de balanceo depende fundamentalmente de la densidad de la clase minoritaria.

En resumen, este estudio evidencia que reportar métricas casi perfectas en la detección de fraude carece de validez si la evaluación presenta fallas metodológicas. Para desarrollar sistemas que sean aplicables en entornos reales, es un requisito técnico fundamental implementar esquemas de partición estrictos que eviten la fuga de información y utilizar métricas verdaderamente representativas frente al desbalance de clases.

8. Anexo

Experimento	Model	AUPRC	F1-Score	Recall	Precision	Accuracy
RUS	DTC	0,146	0,050	0,929	0,026	0,940
	SVC	0,700	0,416	0,888	0,272	0,996
	KNC	0,598	0,186	0,888	0,104	0,987
	MLP	0,664	0,213	0,898	0,121	0,989
	RFC	0,686	0,108	0,918	0,057	0,974
	GBC	0,697	0,085	0,918	0,045	0,966
	LGBC	0,740	0,095	0,939	0,050	0,969
	XGBC	0,683	0,087	0,929	0,046	0,967
	EBC	0,721	0,789	0,765	0,815	0,999
	EIC	0,716	0,751	0,755	0,747	0,999
	EMC	0,732	0,808	0,796	0,821	0,999
	EIBMC	0,721	0,789	0,765	0,815	0,999
CC	DTC	0,005	0,007	0,980	0,003	0,500
	SVC	0,714	0,378	0,888	0,240	0,995
	KNC	0,742	0,362	0,878	0,228	0,995
	MLP	0,688	0,309	0,888	0,187	0,993
	RFC	0,804	0,008	0,959	0,004	0,567
	GBC	0,243	0,007	0,980	0,004	0,546
	LGBC	0,515	0,008	0,980	0,004	0,602
	XGBC	0,548	0,009	0,980	0,005	0,634
	EBC	0,686	0,625	0,510	0,806	0,999
	EIC	0,800	0,772	0,776	0,768	0,999
	EMC	0,810	0,783	0,755	0,813	0,999
	EIBMC	0,825	0,770	0,735	0,809	0,999

Experimento	Model	AUPRC	F1-Score	Recall	Precision	Accuracy
IHT+RFC	DTC	0,641	0,730	0,867	0,630	0,999
	SVC	0,724	0,796	0,816	0,777	0,999
	KNC	0,755	0,773	0,816	0,734	0,999
	MLP	0,706	0,768	0,827	0,717	0,999
	RFC	0,795	0,762	0,867	0,680	0,999
	GBC	0,664	0,700	0,847	0,597	0,999
	LGBC	0,591	0,716	0,643	0,808	0,999
	XGBC	0,730	0,763	0,837	0,701	0,999
	EBC	0,753	0,806	0,786	0,828	0,999
	EIC	0,794	0,796	0,776	0,817	0,999
	EMC	0,759	0,804	0,796	0,813	0,999
	EIBMC	0,797	0,802	0,786	0,819	0,999
IHT+GBC	DTC	0,666	0,659	0,827	0,547	0,999
	SVC	0,760	0,796	0,816	0,777	0,999
	KNC	0,839	0,806	0,827	0,786	0,999
	MLP	0,746	0,733	0,755	0,712	0,999
	RFC	0,849	0,785	0,878	0,711	0,999
	GBC	0,798	0,711	0,867	0,603	0,999
	LGBC	0,860	0,794	0,867	0,733	0,999
	XGBC	0,851	0,792	0,837	0,752	0,999
	EBC	0,855	0,826	0,776	0,884	0,999
	EIC	0,853	0,822	0,755	0,902	0,999
	EMC	0,846	0,838	0,765	0,926	0,999
	EIBMC	0,858	0,840	0,776	0,916	0,999

Experimento	Model	AUPRC	F1-Score	Recall	Precision	Accuracy
IHT+LGBC	DTC	0,826	0,825	0,816	0,833	0,999
	SVC	0,742	0,796	0,816	0,777	0,999
	KNC	0,829	0,854	0,806	0,908	1,000
	MLP	0,714	0,740	0,684	0,807	0,999
	RFC	0,860	0,857	0,827	0,890	1,000
	GBC	0,780	0,786	0,673	0,943	0,999
	LGBC	0,753	0,794	0,765	0,824	0,999
	XGBC	0,878	0,852	0,796	0,918	1,000
	EBC	0,875	0,859	0,837	0,882	1,000
	EIC	0,845	0,818	0,755	0,892	0,999
	EMC	0,877	0,865	0,816	0,920	1,000
	EIBMC	0,877	0,859	0,837	0,882	1,000
IHT+XGBC	DTC	0,816	0,851	0,816	0,889	1,000
	SVC	0,766	0,796	0,816	0,777	0,999
	KNC	0,829	0,851	0,786	0,928	1,000
	MLP	0,718	0,682	0,592	0,806	0,999
	RFC	0,867	0,863	0,806	0,929	1,000
	GBC	0,800	0,754	0,704	0,812	0,999
	LGBC	0,648	0,597	0,469	0,821	0,999
	XGBC	0,755	0,823	0,735	0,935	0,999
	EBC	0,862	0,865	0,816	0,920	1,000
	EIC	0,831	0,813	0,755	0,881	0,999
	EMC	0,863	0,843	0,796	0,897	0,999
	EIBMC	0,869	0,876	0,827	0,931	1,000

Tabla 5: *Métricas de evaluación agrupadas por experimento, obtenidas en la ejecución del flujo de trabajo propuesto*

Referencias

[1] Md Alamin Talukder, Majdi Khalid, Md Ashraf Uddin, et al. (2024). An integrated multistage ensemble machine learning model for fraudulent transaction detection. https://doi.org/10.1186/s40537-024-00996-5

[2] Takaya Saito, & Marc Rehmsmeier. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. https://doi.org/10.1371/journal.pone.0118432

[3] Shachar Kaufman, Saharon Rosset, Claudia Perlich & Ori Stitelman. (2012). Leakage in data mining: Formulation, detection, and avoidance. https://dl.acm.org/doi/10.1145/2382577.2382579

[4] Andrius Vabalas, Emma Gowen, Ellen Poliakoff & Alexander J. Casson. (2019). Machine learning algorithm validation with a limited sample size. https://doi.org/10.1371/journal.pone.0224365

[5] Claudia Beleites, Ute Neugebauer, Thomas Bocklitz, Christoph Krafft, & Jürgen R. Popp. (2013). Sample size planning for classification models. https://doi.org/10.1016/j.aca.2012.11.007

[6] Alexandru Niculescu-Mizil, & Rich Caruana. (2005). Predicting Good Probabilities With Supervised Learning. https://doi.org/10.1145/1102351.1102430

[7] Jesse Davis, & Mark Goadrich. (2006). The relationship between Precision-Recall and ROC curves. https://doi.org/10.1145/1143844.1143874

[8] Iman Sharafaldin, Arash Habibi Lashkari, & Ali A. Ghorbani. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. https://doi.org/10.5220/0006639801080116

[9] MLG - ULB: Credit Card Fraud Dataset. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud. Accedido en mayo del 2026.

[10] Hammed M & Soyemi J. (2020). An implementation of decision tree algorithm augmented with regression analysis for fraud detection in credit card. Int J Comput Sci Inf Secur. https://www.researchgate.net/publication/341871791_An_implementation

_of_decision_tree_algorithm_augmented_with_regression_analysis_for_fraud_detection_in_credit_card

[11] Talukder, M.A., Sharmin, S., Uddin, M.A. et al. MLSTL-WSN: machine learning-based intrusion detection using SMOTETomek in WSNs. Int. J. Inf. Secur. 23, 2139–2158 (2024). https://doi.org/10.1007/s10207-024-00833-z

[12] Ganji VR. (2012). Mannem SNP. Credit card fraud detection using anti-k nearest neighbor algorithm. Int J Comput Sci Eng. https://www.researchgate.net/publication/236962626_credit_card_fraud_detection_using_anti_k-nearest_algorithm

[13] Talukder MA, Islam MM, Uddin MA, Akhter A, Hasan KF & Moni MA. (2022). Machine learning-based lung and colon cancer detection using deep feature extraction and ensemble learning. https://doi.org/10.1016/j.eswa.2022.117695

[14] Talukder, M.A., Islam, M.M., Uddin, M.A. et al. Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. J Big Data 11, 33 (2024). https://doi.org/10.1186/s40537-024-00886-w

[15] Talukder MA, Hossen R, Uddin MA, Uddin MN, Acharjee UK. Securing transactions: a hybrid dependable ensemble machine learning model using iht-lr and grid search. Cybersecurity 2024. https://doi.org/10.48550/arXiv.2402.14389

[16] Talukder MA, Hasan KF, Islam MM, Uddin MA, Akhter A, Yousuf MA, Alharbi F & Moni MA. (2023). A dependable hybrid machine learning model for network intrusion detection. J Inf Secur Appl. https://doi.org/10.48550/arXiv.2212.04546

[17] Aydin Demircioğlu. (2024). Applying oversampling before cross-validation will lead to high bias in radiomics. https://doi.org/10.1038/s41598-024-62585-z