Fraccionamiento Transaccional

Presentado por: William Amaya Patiño

Contenido

1 Alcance del proyecto

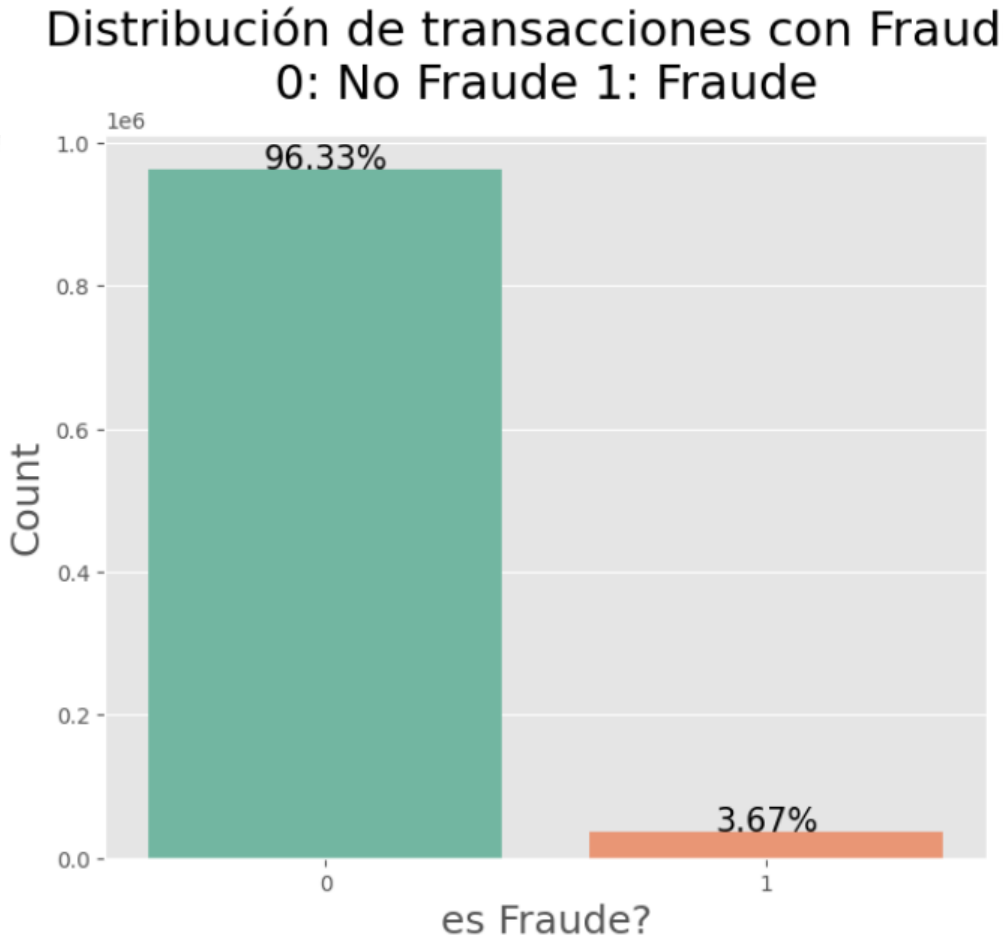
Fraccionamiento Transaccional:

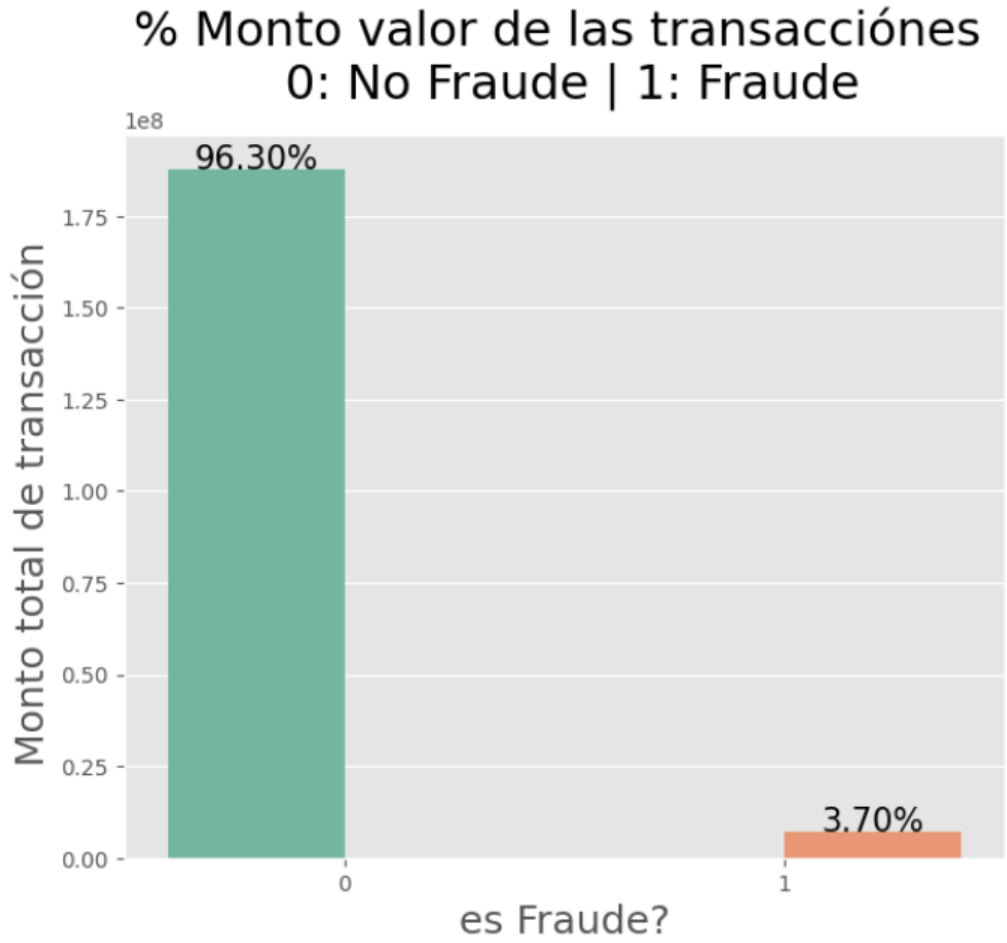
Esta mala práctica consiste en dividir una transacción en varias transacciones de menor monto, cuyo total agrupado equivale al valor de la transacción original. Estas transacciones suelen realizarse dentro de un mismo período de tiempo, generalmente 24 horas, y tienen como origen o destino la misma cuenta o cliente.

2 EDA

Distribución y análisis del comportamiento de transacciones

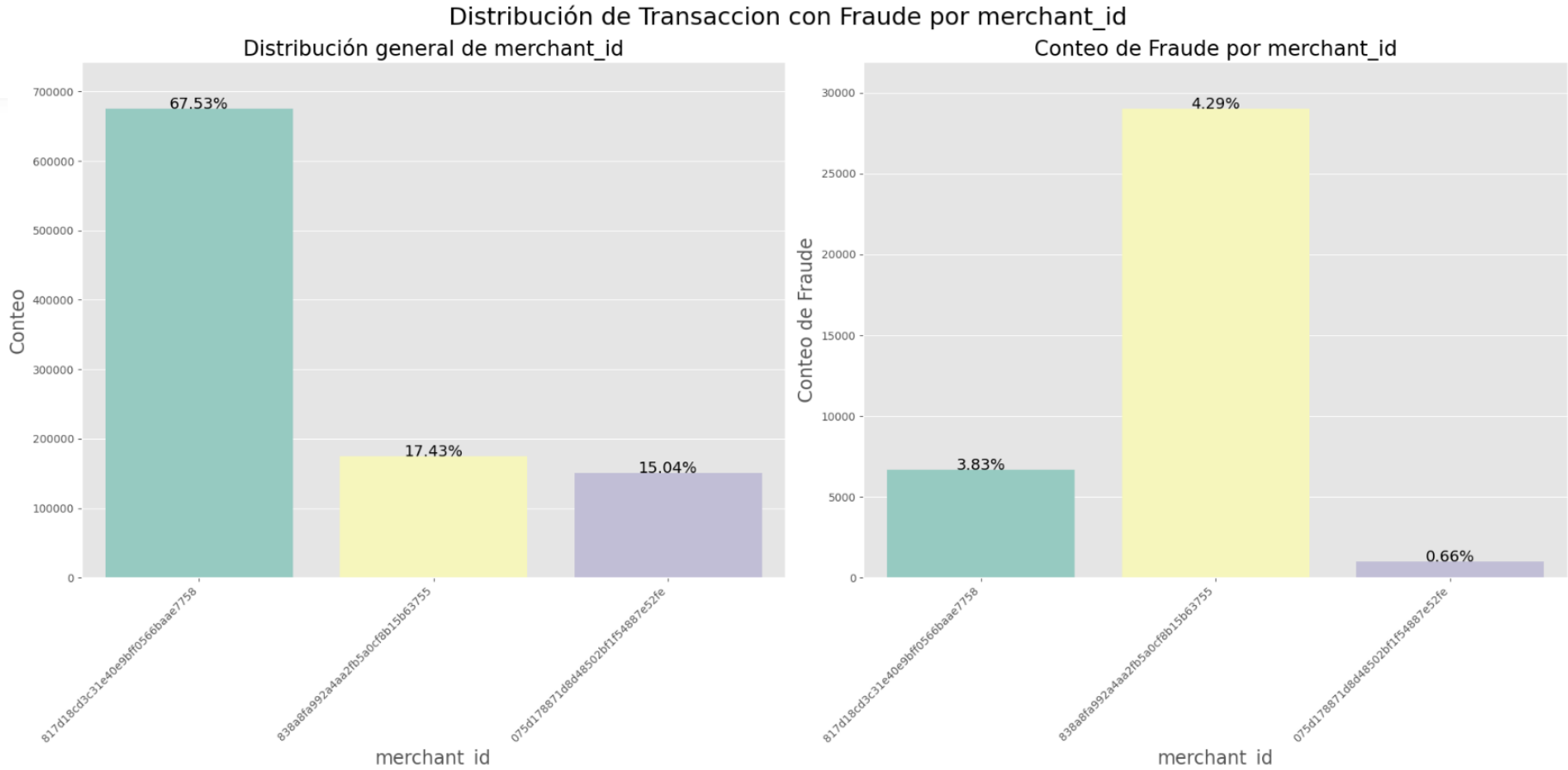
- Las transacciones fraudulentas representan solo el **3.67% del total**, pero concentran una proporción significativa del monto transaccional (**3.70% del valor total**). Esto resalta su tendencia a involucrar montos altos, subrayando la importancia de detectar y prevenir estos eventos de manera efectiva.





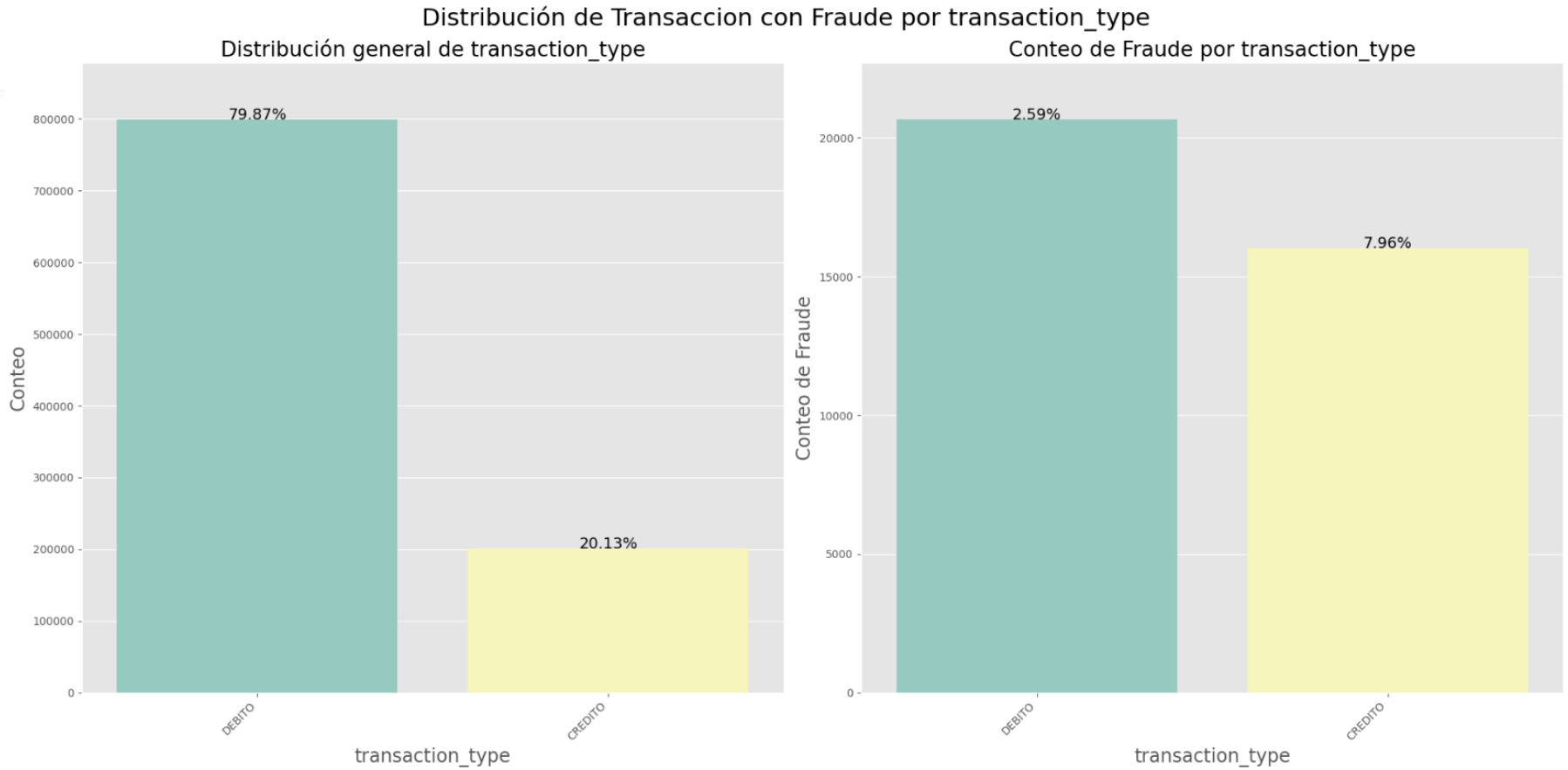
Distribución de Transacciones Fraudes por Tipo de Marchant_id

- El merchant ID con mayor actividad representa el **67.53%** de las transacciones totales, mientras que los otros dos concentran el **17.43%** y **15.04%**, respectivamente.
- A pesar de la alta actividad del merchant ID principal, el mayor porcentaje de fraudes se registra en el segundo merchant ID (**4.29%**), lo que indica un comportamiento atípico que requiere atención prioritaria para análisis y mitigación.



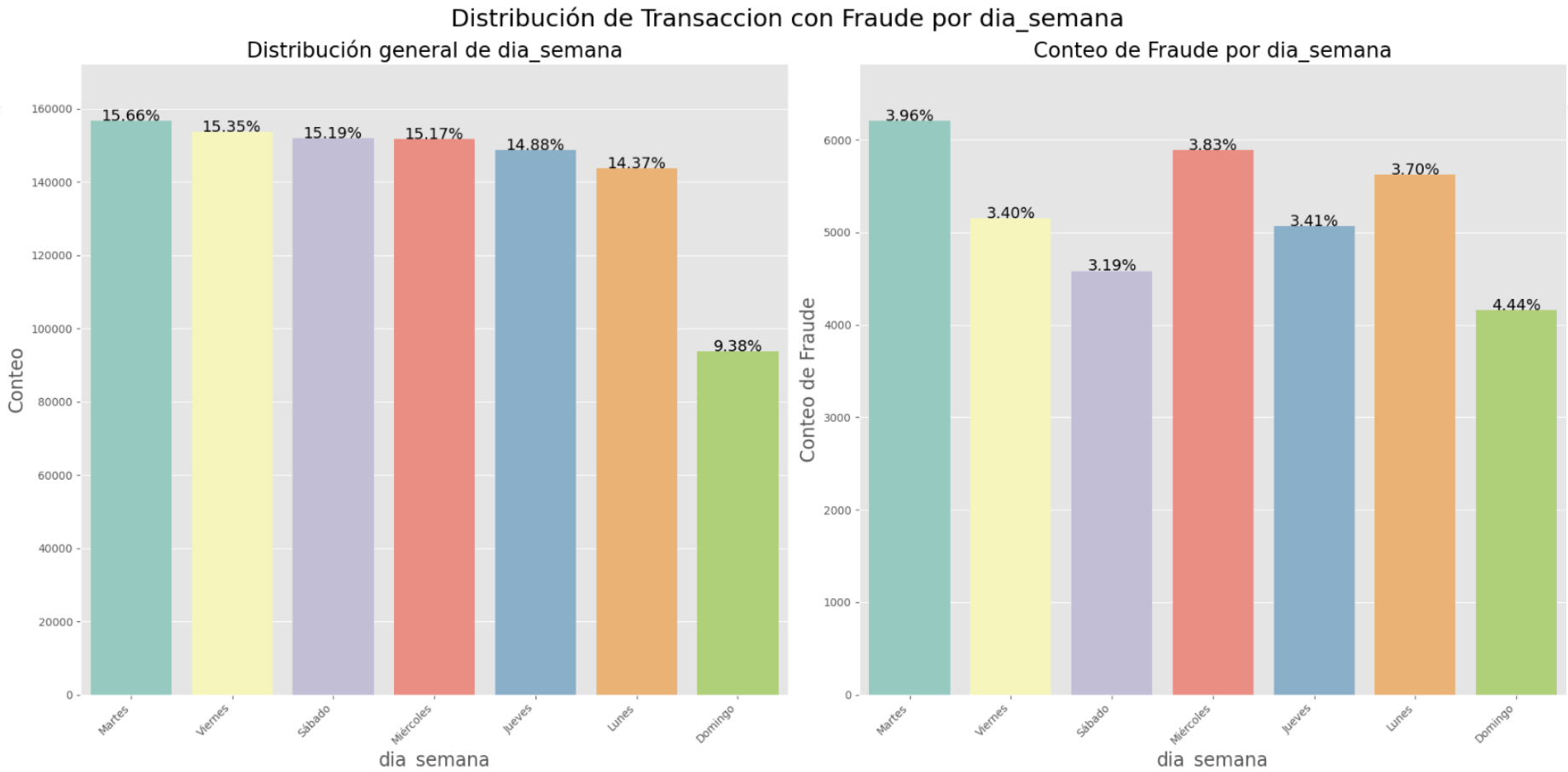
Distribución de Transacciones con Fraude por transaction_type

- Las transacciones de débito representan la mayor parte del total (**79.87%**), mientras que las de crédito solo el **20.13%**. Sin embargo, las transacciones de crédito tienen un mayor porcentaje de fraude (**7.96%**) en comparación con las de débito (**2.59%**), lo que evidencia un mayor riesgo asociado a las transacciones de crédito.



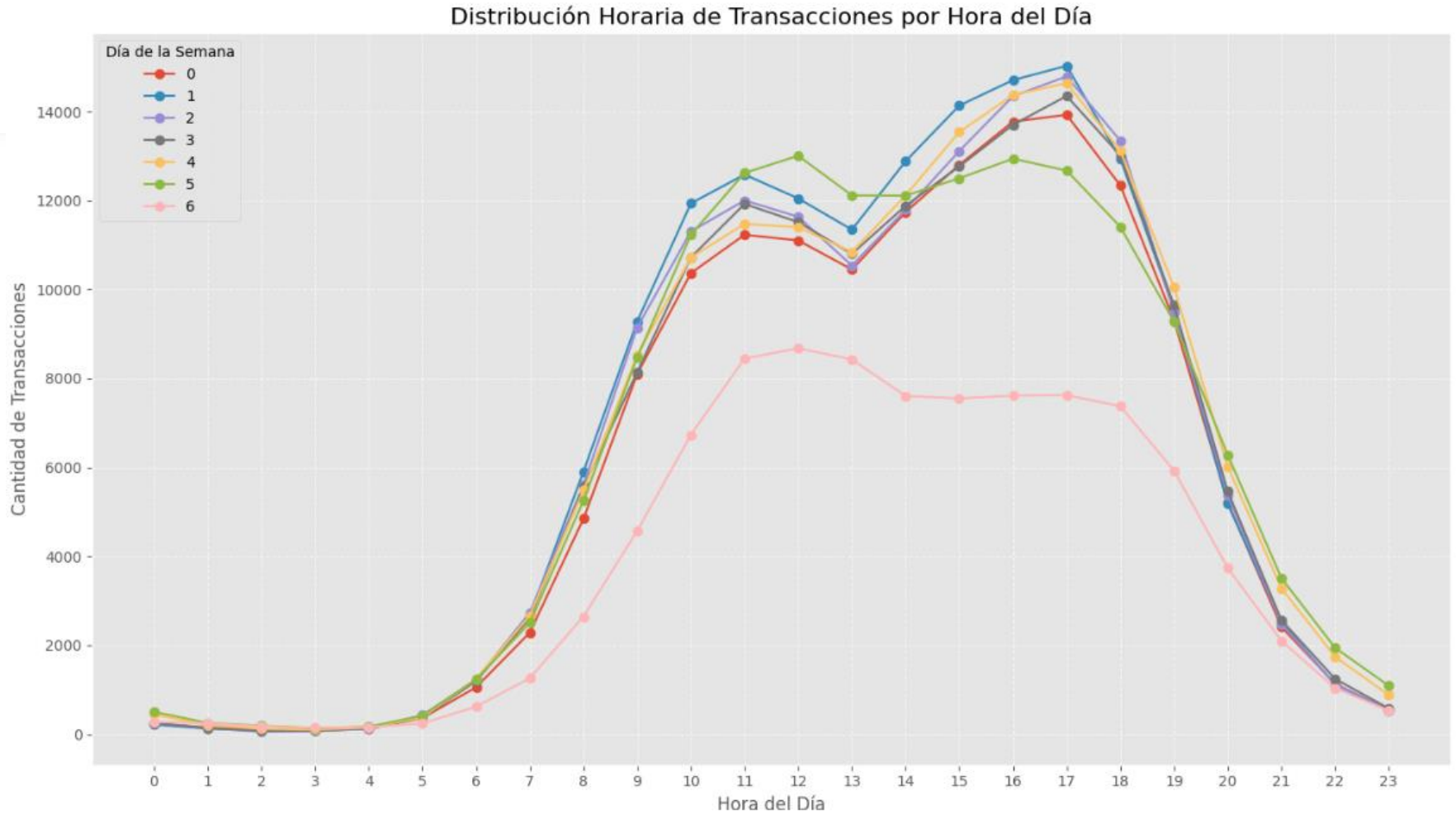
Distribución de Transacciones con Fraude por Dia de la semana

- Las transacciones se distribuyen de manera uniforme durante la semana, siendo el martes el día con más actividad. Sin embargo, el domingo registra el mayor porcentaje de fraudes (4.44%), lo que destaca la necesidad de mayor vigilancia en este día.



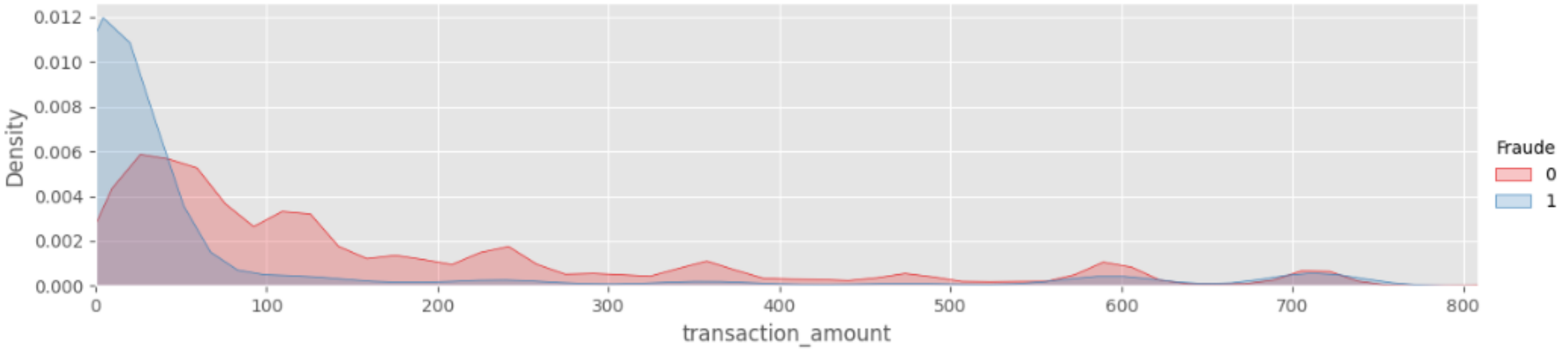
Distribución de Transacciones con Fraude por Hora del Día

- Todos los días presentan un incremento en la actividad a partir de las 7:00 horas, alcanzando picos entre las 9:00 y las 13:00 horas, con una disminución progresiva después de las 19:00 horas. Sin embargo, el día 6 (domingo) registra consistentemente menos transacciones, especialmente durante la tarde, lo que destaca un comportamiento diferente respecto a los demás días.



Distribución de Transacciones con Fraude por transaction_amount

- Se observa que la mayoría de las transacciones fraudulentas tienden a concentrarse en montos bajos, similares a las no fraudulentas. Sin embargo, en rangos más altos de montos, las transacciones no fraudulentas son más frecuentes, destacando una posible diferenciación en el comportamiento transaccional según el monto. Esto podría indicar que los montos más altos son menos comunes en transacciones fraudulentas.



3

Definir el modelo analítico

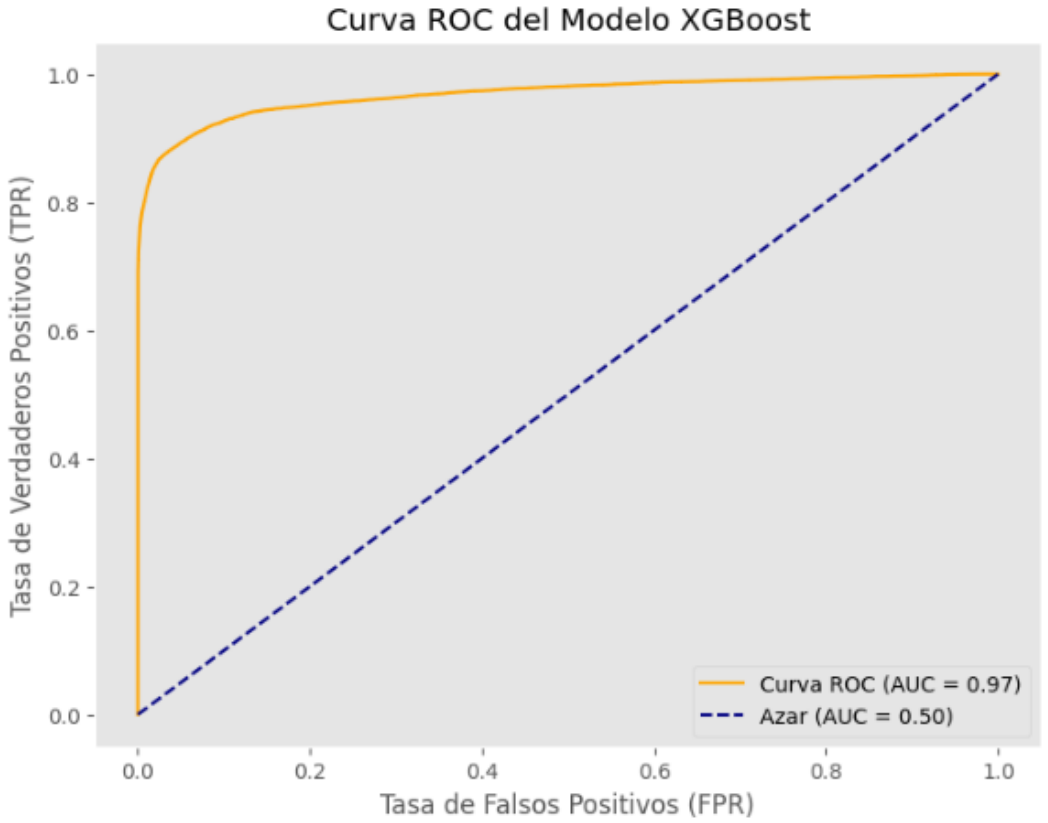
Comparación de los Modelos

El mejor modelo es **XGBoost**, basado en su equilibrio entre las métricas de desempeño clave:
1.ROC AUC: El modelo XGBoost presenta el valor más alto de ROC AUC (0.9688), indicando su excelente capacidad de discriminación entre clases.
2.F1 Score: Su F1 Score es competitivo (0.8341), lo que refuerza su rendimiento en datasets desbalanceados, equilibrando precisión y recall.
3.Precision y Recall: Ofrece una alta precisión (0.9517) y un recall adecuado (0.7425), esencial para minimizar falsos positivos y capturar la mayoría de fraudes.
.

	Accuracy	Recall	Precision	f1_score	ROC_AUC	Description
Logistic Regression	0.9881	0.6763	1.0000	0.8069	0.8381	
Logistic Regression class_weight	0.9027	0.9027	0.9027	0.9027	0.9027	
Random Forest	0.9884	0.7380	0.9306	0.8232	0.9404	
Random Forest class_weight	0.9673	0.8153	0.5356	0.6465	0.9362	
Gradient Boosting	0.9892	0.7437	0.9500	0.8343	0.9654	
XGBoost	0.9892	0.7425	0.9517	0.8341	0.9688	best model
XGBoost parametros	0.9760	0.3562	0.9691	0.5210	0.9144	
AdaBoost	0.9886	0.7125	0.9677	0.8207	0.9640	
Decision Tree	0.9877	0.7331	0.9141	0.8137	0.9132	

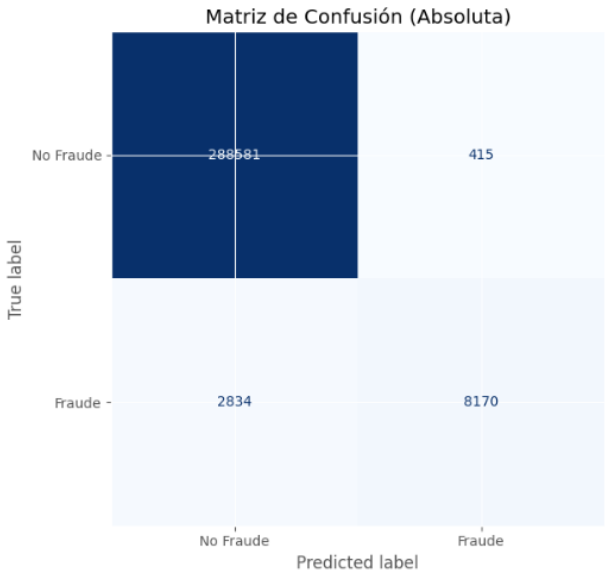
XGBoost

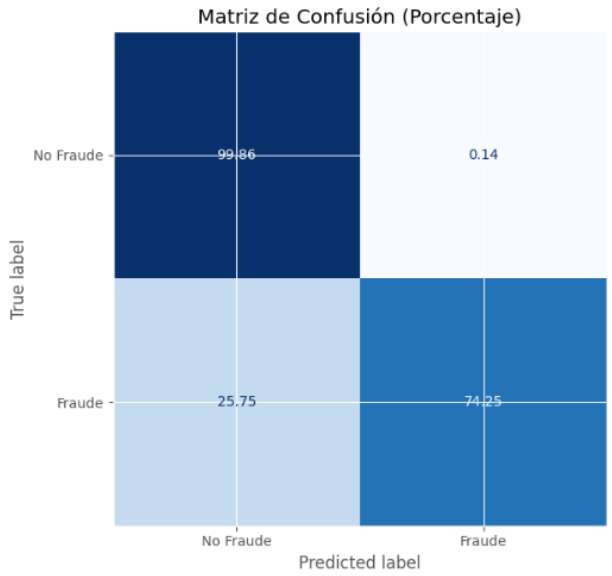
La curva ROC del modelo XGBoost destaca por su excelente desempeño, evidenciado por un AUC (Área Bajo la Curva) de 0.97. Esto significa que el modelo tiene una alta capacidad de distinguir entre clases (fraude y no fraude).
En comparación con la línea de referencia (Azar) con un AUC de 0.50, el modelo XGBoost supera significativamente el rendimiento esperado por casualidad.
La forma de la curva, cercana al punto superior izquierdo, indica una alta tasa de verdaderos positivos (TPR) con una baja tasa de falsos positivos (FPR), haciendo de este modelo una opción robusta para la detección de fraudes



Matriz de confusión en predicciones con XGBoost

El modelo XGBoost muestra un excelente desempeño en la identificación de transacciones "No Fraude" (99.86% de precisión). Sin embargo, aunque el 74.25% de los fraudes fueron detectados correctamente, un 25.75% quedó sin identificar, lo que resalta la necesidad de mejorar la detección de fraudes para reducir los falsos negativos.





Validación cruzada de k-Fold -XGBoost

- Tras realizar la validación cruzada con KFold, nuestro modelo más destacado ha conservado su excelente rendimiento

Resultados de la Validación Cruzada (AUC-ROC):

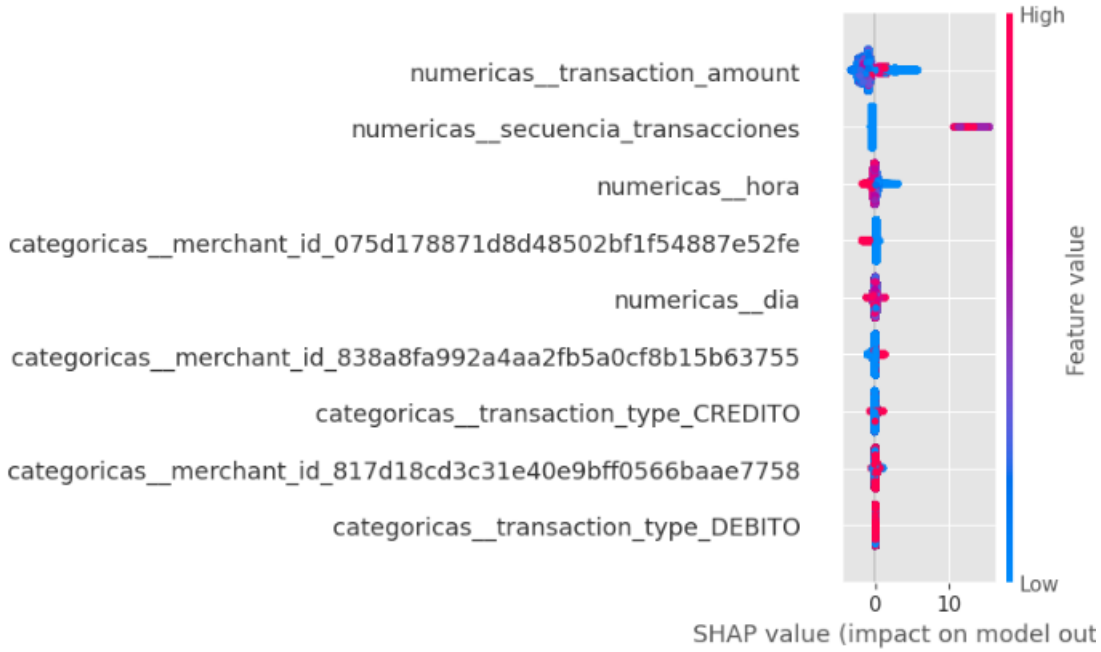
Scores por fold: [0.96998387 0.96945124 0.9707269 0.97148672 0.96840811]

AUC-ROC promedio: 0.9700 AUC-ROC

desviación estándar: 0.0011

Las variables más determinantes en el modelo de **XGBoost.**

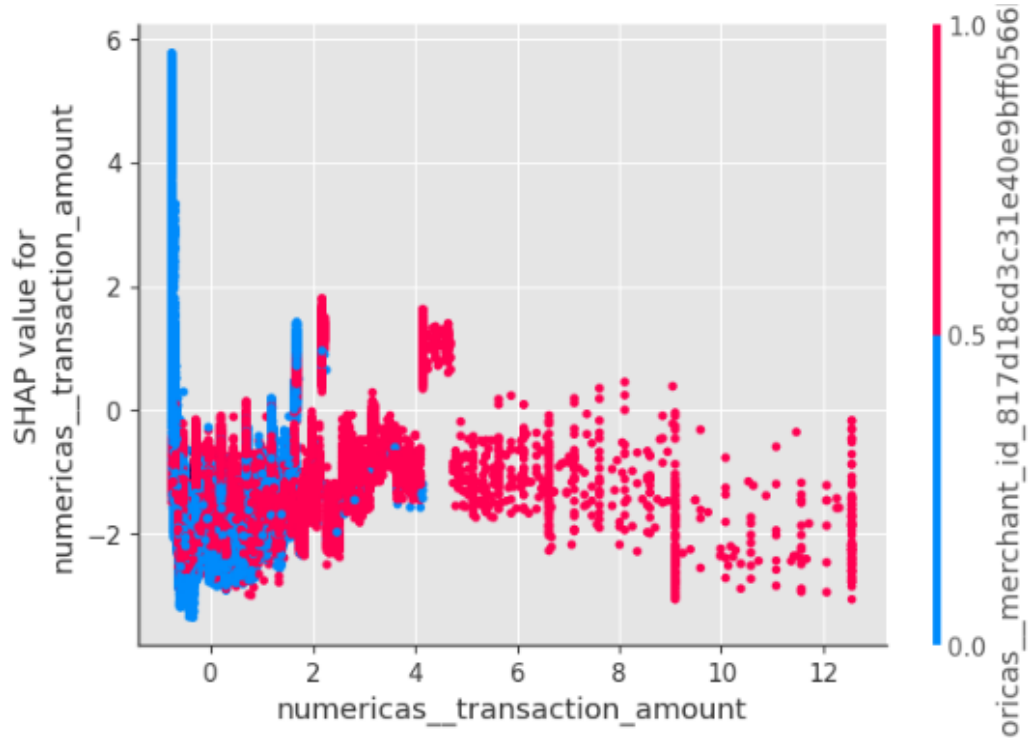
El análisis SHAP destaca las variables con mayor impacto en las predicciones del modelo XGBoost. Entre ellas, **transaction_amount** y **secuencia_transacciones** son las más influyentes, seguidas por variables categóricas como transaction_type_CREDITO. Los valores SHAP permiten interpretar cómo los valores altos (rojo) o bajos (azul) de cada característica afectan la probabilidad de predicción, facilitando la comprensión del comportamiento del modelo y la priorización de variables clave para optimizar su desempeño.



Transaction_amount

Para transacciones pequeñas (valores bajos de **transaction_amount**), el impacto en el modelo varía ampliamente, mostrando una dispersión significativa en los valores SHAP.

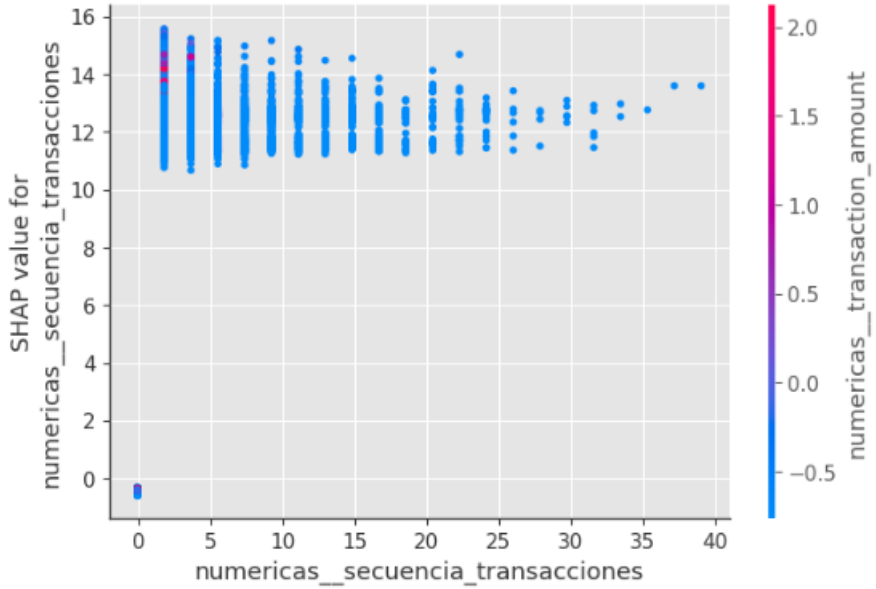
A medida que aumenta el monto de la transacción, los valores SHAP se estabilizan y tienden a ser negativos, lo que sugiere una menor probabilidad de fraude para montos mayores.



secuencia_transacciones

Valores más altos en el eje Y (SHAP) indican que una secuencia alta de transacciones incrementa la probabilidad de fraude.

Valores más bajos en el eje Y sugieren que una menor secuencia de transacciones disminuye dicha probabilidad de fraude.



Proponga con qué frecuencia deben actualizarse los datos y por qué

1. Recolección de Información

Frecuencia: Actualización diaria con datos en tiempo real (día vencido).

Objetivo: Identificar patrones recientes en fraccionamientos transaccionales y mantener los modelos actualizados.

2. Procesamiento y Análisis

Agrupamientos diarios: Detectar fraccionamientos transaccionales para capturar irregularidades con rapidez.

Tratamiento de datos: Depuración y manejo de valores atípicos al final de cada ciclo de predicción y entrenamiento (cada 24 horas).

3. Modelo y Ajustes

Reentrenamiento: Realizado semanalmente o en respuesta a cambios significativos en los datos (nuevos patrones de fraude).

Ajuste de hiperparámetros: Realizado cada 15 días o mensualmente para optimizar el rendimiento del modelo.

Diseñar una arquitectura ideal y los recursos necesarios para desplegar su propuesta (Opcional)

1: Amazon SageMaker

Entrenamiento del modelo: SageMaker se utiliza para entrenar un modelo de ML con los datos disponibles.

Despliegue del modelo: El modelo entrenado se convierte en un archivo binario y se despliega como un endpoint accesible para generar predicciones.

2: AWS Lambda

Configuración de Lambda: Función intermediaria entre el API Gateway y el endpoint de SageMaker.

Invocación del endpoint: Lambda recibe datos desde API Gateway, envía la solicitud al endpoint y devuelve la predicción.

3: Amazon API Gateway

Creación de la API: API REST configurada para recibir solicitudes de los usuarios.

Integración con Lambda: Envía los datos de entrada a Lambda, que invoca el endpoint de SageMaker.

Respuesta al usuario: API Gateway entrega la predicción procesada por Lambda al usuario final.

GRACIAS!!!