UVG UNIVERSIDAD DEL VALLE DE GUATEMALA

Guatemala 02 de junio de 2025

Josue Argueta - 211024

Security Data Science

Proyecto final. Transacciones bancarias

---

1. Resumen

Este proyecto se enfoca en el desafío de detectar fraudes en tarjetas de crédito utilizando un conjunto de datos que está muy desbalanceado. Se diseñan y comparan varias estrategias de modelado y métricas de evaluación con el objetivo de maximizar la detección de fraudes (recall) y reducir al mínimo los falsos positivos, especialmente en clientes con comportamientos complejos. Se implementan técnicas avanzadas de ingeniería de características y métricas personalizadas que se alinean con los objetivos del negocio. Los resultados indican que la elección de la métrica y el umbral de decisión tienen un impacto significativo en el tipo de errores que se cometen y en la experiencia del usuario.

2. Metodología

a) Análisis Exploratorio

Se realizó un análisis visual y estadístico para entender el dataset y las variables más relevantes para el fraude:

- **Desbalance de clases:** Solo el 0.5% de las transacciones son fraudes.
- **Correlación de variables numéricas:** La variable con mayor correlación con fraude es el monto gastado (amt), aunque las correlaciones absolutas son bajas, como es habitual en problemas de fraude.
- **Análisis temporal y por categoría:** Se analizaron los patrones de fraude por hora, día del mes, comercio y categoría de compra.
- **Análisis de complejidad del cliente:** Se exploró la cantidad de comercios únicos por cliente, la varianza de montos y el porcentaje de compras nocturnas. Estas variables ayudan a identificar clientes “no convencionales”.

b) Feature Engineering

- **Variables de complejidad:**
  Se creó un complexity_score combinando el número de comercios únicos, la varianza del monto mensual y el porcentaje de compras nocturnas por cliente.
- **Definición de perfiles complejos:**
  Se considera perfil complejo al 20% de clientes con mayor complexity_score. Esto permite analizar el impacto diferencial de los falsos positivos en clientes legítimos pero atípicos.
- **Features adicionales:**
  Se incluyeron variables temporales (time_since_last_tx, day_of_week),

interacciones (dist_amt_interaction, night_shopping), estadísticas rodantes y codificaciones target de variables categóricas.

c) Modelado y Selección de Métricas

- **Modelo base:**
  Se entrenó un modelo LightGBM optimizando AUC-ROC y utilizando validación tipo TimeSeriesSplit para evitar data leakage temporal.
- **Métricas personalizadas:**
  Dado el desbalance y el objetivo de minimizar FP en perfiles complejos, se diseñaron y probaron métricas como:
  - **Recall at 90%**: Maximiza la detección de fraudes, ajustando el umbral.
  - **FP Ratio**: Minimiza el porcentaje de falsos positivos.
  - **Balanced**: Score que da doble peso a la precisión y uno a recall.
  - **Penalización de FP en perfiles complejos**: Recall penalizado si los FP afectan perfiles complejos.
  - **F-beta 0.5**: Da mayor importancia a la precisión.
- **Optimización de hiperparámetros:**
  Se utilizó RandomizedSearchCV sobre un espacio reducido para acelerar la selección bajo cada métrica.

d) Ajuste de Threshold y Análisis de Errores

Se ajustó el umbral de decisión para alcanzar recalls específicos y se evaluó el impacto de cada estrategia sobre la cantidad y distribución de los FP, diferenciando entre perfiles simples y complejos.

3. Implementación Práctica

1. Análisis Exploratorio de Datos y Visualización

El objetivo fue comprender el comportamiento del dataset, identificar posibles sesgos y detectar variables informativas, así como patrones temporales y de negocio. Este conocimiento fundamenta la posterior ingeniería de variables y selección de modelos.

2. Ingeniería de Variables de Complejidad

Se introdujeron variables para capturar la “complejidad” de los clientes, tales como el número de comercios distintos visitados, la varianza de montos mensuales y el porcentaje de compras nocturnas. La razón principal fue mitigar los falsos positivos en usuarios legítimos pero atípicos, cuyos patrones pueden diferir del grueso de la población sin implicar fraude. Adicionalmente, se creó un *complexity score* agregando estas métricas, permitiendo analizar el desempeño del modelo específicamente en usuarios complejos, segmento crítico en aplicaciones reales de banca y seguridad.

3. Generación de Variables Temporales y Contextuales

El desarrollo de variables como tiempo desde la última transacción, día de la semana, montos relativos a promedios móviles y la interacción distancia-monto respondió a la necesidad de dotar al modelo de contexto temporal y espacial. Estas variables permiten identificar fraudes que se manifiestan como secuencias o anomalías contextuales, incrementando la sensibilidad del modelo sin incrementar los falsos positivos en usuarios legítimos.

4. Preprocesamiento y Evitación de Fugas de Información

La normalización de variables (por ejemplo, z-score para montos) se realizó exclusivamente sobre el conjunto de entrenamiento, asegurando la no contaminación de los datos de validación o prueba (*data leakage*). Esta decisión garantiza la validez de la evaluación del modelo y la transferencia del desempeño a datos futuros.

5. Codificación de Variables Categóricas

Se optó por codificación basada en el target (target encoding) para variables de alta cardinalidad, como comercios y categorías, en lugar de técnicas más simples como one-hot encoding. Esto permite capturar relaciones directas entre cada entidad y la probabilidad de fraude.

6. División Temporal y Validación Cruzada

La separación entre entrenamiento y prueba se realizó respetando el eje temporal, simulando el escenario real de despliegue y asegurando una evaluación fidedigna. Se empleó *TimeSeriesSplit* para la validación cruzada, preservando la secuencia temporal y evitando sesgos por “mirar hacia el futuro”.

7. Ajuste de Hiperparámetros y Estrategias para Desbalance

El ajuste de hiper parámetros se realizó mediante búsqueda aleatoria y validación temporal. Se utilizó la opción *scale_pos_weight* en LightGBM para balancear la contribución de la clase minoritaria (fraude), en línea con la literatura sobre detección en datasets desbalanceados.

8. Desarrollo y Comparación de Métricas Personalizadas

Se diseñaron y compararon varias métricas alternativas al tradicional AUC-ROC, incluyendo penalización explícita de falsos positivos (*fp_ratio*, *fp_complex_penalty*), métricas balanceadas (*balanced*), recall objetivo (*recall90*), y F-beta enfocada en precisión. Esta pluralidad de métricas se justifica en la necesidad de alinear la optimización con los objetivos de negocio: reducir el costo operativo y la afectación a clientes legítimos, sin sacrificar la sensibilidad a fraudes.

9. Análisis Diferenciado de Falsos Positivos

Se realizó un análisis específico de la distribución de falsos positivos en perfiles simples y complejos, validando así que el modelo no solo maximiza el desempeño global, sino que cumple la meta de evitar alertas injustificadas en clientes con comportamientos inusuales pero legítimos.

10. Selección de la Estrategia “Balanced”

De acuerdo con los resultados obtenidos, la métrica “balanced” se seleccionó como la mejor estrategia, ya que ofrece un compromiso óptimo entre recall y precisión, manteniendo bajos los falsos positivos tanto en el total como en el subgrupo de perfiles complejos. Esto permite cumplir de forma robusta el objetivo central del proyecto.

La lógica responde de manera proactiva a la problemática real del negocio: la detección efectiva de fraudes con el mínimo impacto adverso sobre clientes legítimos. El resultado es un sistema de modelado que, mediante una ingeniería de variables exhaustiva y un enfoque de métricas alineadas con el negocio, logra un desempeño equilibrado, robusto y transferible a escenarios reales.

4. Análisis de Resultados

4.1. Métricas Globales y Matrices de Confusión

- **Modelos optimizados a recall alto (recall90)** logran detectar más del 90% de los fraudes, pero generan un número significativo de falsos positivos, con una fracción considerable cayendo en clientes de perfil complejo.
- **Modelos optimizados para precisión o FP Ratio** reducen drásticamente la cantidad de FP, especialmente sobre perfiles complejos, pero sacrifican recall (pueden dejar escapar algunos fraudes).
- **Balanced y F-beta 0.5** ofrecen puntos intermedios, permitiendo cierto control sobre ambos tipos de error.

Ejemplo concreto:
En el modelo optimizado para recall90, el 38.3% de los FP caen sobre perfiles complejos, mientras que modelos orientados a precisión logran que menos del 10% de los FP afecten estos perfiles.

4.2. Distribución de Falsos Positivos

El análisis muestra que los perfiles complejos, si bien son minoría, concentran una fracción desproporcionada de los FP en modelos agresivos en recall. Por tanto, la métrica personalizada que penaliza FP en perfiles complejos resulta más alineada a la realidad operativa y a la experiencia del cliente.

5. Conclusiones

- La extrema desproporción entre clases requiere un enfoque específico tanto en modelado como en evaluación.
- La incorporación de variables de complejidad y la diferenciación de perfiles es fundamental para evitar penalizar a usuarios legítimos con comportamientos inusuales.
- La selección de métrica de optimización determina el tipo de trade-off aceptado. Métricas orientadas a recall garantizan máxima cobertura, pero incrementan el costo operativo y el potencial daño reputacional por FP.
- La métrica de penalización de FP en perfiles complejos permite adaptar el modelo a necesidades reales del sector financiero, privilegiando la experiencia del cliente y la eficiencia operativa.
- Se recomienda una revisión periódica de la definición de “perfil complejo” y la evaluación de impacto real de los FP sobre la operación, ajustando thresholds y estrategias según el contexto y la evolución del fraude.
- Entre los modelos utilizados, el que mejor se adaptó, fue el balanced, pues tiene, como su nombre lo indica, un balance entre precisión, recall y falsos positivos. Siendo de utilidad para ello

Nota:
Todas las visualizaciones y resultados cuantitativos (tablas, barras, matrices de confusión) se encuentran documentadas en el notebook pf_with_GPU.ipynb y en pf_with_CPU.ipynb. Asi mismo, se adjuntan los modelos en el siguiente enlace.
https://drive.google.com/drive/folders/1r1qjdMdeNOGtNogCMdToHnOUkS7WUmQg?usp=drive_link

CONSIDERACIONES:
Para CPU versión, solo es necesario tener lightGBM. Para GPU, seguir instrucciones del README.md