# Detección de Fraude con Tarjetas de Crédito

## Introducción

Este documento describe el desarrollo y los resultados de un proyecto de detección de fraude con tarjetas de crédito. Incluye los objetivos, preguntas planteadas, aproximación a la solución, desafíos enfrentados y los hallazgos principales obtenidos durante el desarrollo del proyecto.

---

## Descripción del proyecto

El proyecto consiste en desarrollar un sistema para detectar fraudes en transacciones con tarjetas de crédito utilizando técnicas de aprendizaje automático. Incluye la exploración de datos, el manejo del desbalance de clases y la optimización de modelos predictivos, con el objetivo de identificar patrones sospechosos y mejorar la seguridad transaccional.

---

## Objetivos

1. **Detección Temprana de Transacciones Fraudulentas** Identificar transacciones fraudulentas lo antes posible para mitigar pérdidas económicas tanto para las instituciones financieras como para los clientes. Una detección temprana permite bloquear transacciones antes de que se consumen y proteger los fondos de los usuarios.

2. **Analizar Patrones de Fraude** Comprender los patrones comunes en las transacciones fraudulentas, como montos inusuales o tiempos de transacción atípicos. Este análisis puede ser útil para desarrollar reglas adicionales y mejorar los sistemas de prevención en el futuro.

3. **Optimizar los Costos Operativos** Reducir el impacto económico del fraude mediante la detección eficiente de transacciones sospechosas. Minimizar el costo asociado al análisis manual de transacciones sospechosas al automatizar el proceso.

4. **Garantizar la Privacidad y Seguridad de los Datos** Cumplir con normativas de protección de datos, asegurando que el análisis y almacenamiento de datos sensibles sea seguro. Los datos de las transacciones suelen contener información

confidencial, y su manejo inadecuado podría generar riesgos legales y reputacionales.

---

# Preguntas Planteadas

1. **¿Qué características en el dataset están más asociadas al fraude?** Identificar las variables que son indicadores más fuertes de transacciones fraudulentas.

2. **¿Cómo se puede manejar el desbalance de datos entre transacciones legítimas y fraudulentas?** Las transacciones fraudulentas representan una pequeña proporción, lo que puede afectar el entrenamiento del modelo.

3. **¿Qué técnicas y algoritmos son más adecuados para detectar fraudes?** Determinar qué métodos ofrecen el mejor equilibrio entre precisión y rendimiento.

4. **¿Cuáles son los patrones más comunes en las transacciones fraudulentas?** Evaluar si existen montos u horarios recurrentes en las transacciones fraudulentas.

5. **¿Cómo podemos minimizar los falsos positivos y falsos negativos?** Diseñar un modelo que identifique fraudes sin clasificar erróneamente las transacciones legítimas.

---

# Datos utilizados

- Para este proyecto se utilizó un dataset de Transacciones de tarjetas de crédito anonimizadas y etiquetadas como fraudulentas o legítimas. El conjunto de datos contiene transacciones realizadas con tarjetas de crédito en septiembre de 2013 por titulares de tarjetas de crédito europeos.
- La fuente de los datos es:

  https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data

---

# Aproximación a la Solución de los Planteamientos

## 1. Exploración y Preprocesamiento de Datos

- Realizar un análisis exploratorio para comprender la distribución de las variables (por ejemplo, correlaciones entre `Amount`, `Time` y `Class`).
- Identificar y manejar valores nulos o inconsistentes.
- Normalizar o estandarizar las variables relevantes (como `Amount` y `Time`).

### 2. Manejo del Desbalance de Datos

- Aplicar técnicas como SMOTE (Synthetic Minority Oversampling Technique), sobremuestreo o submuestreo.
- Ajustar métricas de evaluación (como F1-score y Precision-Recall) en lugar de solo depender de la precisión general.

### 3. Selección de Algoritmos

- Probar algoritmos supervisados como:
    - Árboles de decisión y Random Forests: Para interpretar las variables más importantes.
    - Gradient Boosting (XGBoost, LightGBM): Para mayor precisión en datasets desbalanceados.
    - Regresión logística: Como modelo base para comparaciones.
- Implementar validación cruzada para evitar sobreajuste.

### 4. Evaluación de Patrones de Fraude

- Analizar transacciones fraudulentas para identificar patrones:
    - Montos más comunes.
    - Distribución horaria.
- Utilizar visualizaciones (mapas de calor, histogramas) para identificar tendencias.

### 5. Reducción de Falsos Positivos y Negativos

- Ajustar umbrales de clasificación para maximizar Recall (detección de fraudes) y minimizar errores.
- Utilizar técnicas de ajuste de hiperparámetros (Grid Search o Random Search).

---

# Desafíos

1. **Lidiar con el desbalance de datos**
   La alta disparidad entre transacciones legítimas y fraudulentas dificultó el análisis de los datos y el entrenamiento de modelos. Este problema requirió el uso de técnicas como SMOTE. Sin embargo, el balanceo pudo influir en los resultados finales, y en otro escenario, un dataset más grande habría permitido aplicar submuestreo en lugar de sobremuestreo.

2. **Sobrecarga de documentación de librerías**
   La cantidad de información disponible en las librerías utilizadas, como `scikit-learn` y `imbalanced-learn`, complicó encontrar soluciones específicas para problemas concretos. Esto demandó tiempo adicional en investigación y pruebas.

3. **Evaluación del desempeño de modelos**
   Seleccionar las métricas adecuadas para evaluar los modelos fue un punto crítico. Métricas como precisión y recall no siempre reflejan completamente la efectividad de los modelos.

4. **Evaluar apropiadamente las capacidades del modelo**
   En ocasiones, se sobreestimaron las capacidades de los modelos utilizados. Las evaluaciones tradicionales empleadas en este proyecto no siempre reflejan resultados completamente precisos. Esto generó cuestionamientos sobre la conveniencia de incorporar métodos más robustos, como la validación cruzada (Cross-Validation), para obtener una evaluación más confiable y generalizable del rendimiento del modelo.

---

# Hallazgos y Conclusiones

## Baja Frecuencia de las Transacciones Fraudulentas

- Los fraudes son eventos extremadamente poco frecuentes (menos del 0.2% del total), lo que complica su detección. Este desbalance requiere estrategias específicas para entrenar modelos efectivos, como el uso de técnicas de sobremuestreo (SMOTE) o ajustes en las métricas de evaluación.

## Patrones Inconsistentes

- Las transacciones fraudulentas no siempre presentan patrones obvios, como montos extremadamente altos. Esto indica que los fraudes pueden ocurrir en cualquier rango de valores, lo que demanda modelos capaces de identificar relaciones complejas.

## Distribución Temporal

- Las transacciones fraudulentas no siguen una tendencia horaria clara, lo que indica que los fraudes pueden ocurrir en cualquier momento.

## Frecuencia por Monto

- Algunas transacciones fraudulentas se concentran en montos específicos (bajos o moderados), probablemente para evitar alertas automatizadas.

## Seguridad de Datos

- Dado que el dataset contiene información transformada, se garantiza la privacidad. Sin embargo, en entornos reales, el manejo ético de datos sensibles es esencial.

### Implicaciones para la Industria

- Impacto de los Falsos Positivos: Aunque los falsos positivos no generan pérdidas directas, pueden deteriorar la experiencia del cliente al bloquear transacciones legítimas.
- Prevención Proactiva: La implementación de sistemas predictivos permite a las instituciones financieras actuar antes de que se complete una transacción fraudulenta.
- Optimización Operativa: La automatización reduce el tiempo y costo dedicados a revisar manualmente transacciones sospechosas.
- Reputación y Confianza: Proteger a los usuarios de fraudes fortalece la relación cliente-banco.

# Conclusiones técnicas

### Variables Relevantes

- Las características derivadas de los datos transaccionales (`Time`, `Amount`, etc.) tienen una correlación importante con la clase de fraude.
- Algunas variables pueden ser irrelevantes debido a su naturaleza transformada o cifrada.

### Impacto del Desbalance de Datos

- Sin un manejo adecuado del desbalance, los modelos tienden a sesgarse hacia la clase mayoritaria (transacciones legítimas), reduciendo la efectividad.

### Eficiencia de los Algoritmos

- Métodos basados en árboles (como Random Forests o Gradient Boosting) suelen ser más efectivos debido a su capacidad de manejar relaciones no lineales.

### Optimización del Modelo

- Los modelos ajustados mediante técnicas de optimización como Grid Search y Random Search mostraron mejoras significativas en las métricas de evaluación. Estas técnicas permitieron explorar una variedad de combinaciones de hiperparámetros para maximizar el rendimiento.
- Los resultados indicaron que, si bien Gradient Boosting fue efectivo, Random Forests demostró ser más eficiente en términos de tiempo de entrenamiento y evaluación.
- Los umbrales ajustados para la clasificación ayudaron a reducir los falsos positivos sin comprometer significativamente la detección de fraudes.

---

## Implicaciones Prácticas

1. **Automatización y Reducción de Costos**
    - Un modelo bien entrenado puede integrarse en sistemas de detección en tiempo real, minimizando la necesidad de revisiones manuales. Esto no solo reduce costos operativos, sino que también mejora la experiencia del usuario al disminuir interrupciones en transacciones legítimas.
2. **Mejoras en la Seguridad**
    - Implementar un modelo basado en aprendizaje automático refuerza los sistemas de seguridad actuales al identificar patrones complejos que no son evidentes mediante reglas tradicionales.
3. **Cumplimiento Regulatorio**
    - Al garantizar la privacidad y seguridad de los datos, este enfoque ayuda a las instituciones financieras a cumplir con normativas internacionales, como el RGPD, protegiendo tanto a los clientes como a la organización.

---

## Recomendaciones Futuras

1. **Incorporar Datos Adicionales**
    - Explorar la integración de nuevas fuentes de datos, como la ubicación geográfica o la categoría del comerciante, para enriquecer el contexto y mejorar la precisión del modelo.
2. **Validación con Datos Reales**
    - Probar el modelo en datos de producción para evaluar su desempeño en escenarios reales. Esto permitirá ajustar los hiperparámetros según los resultados observados.
3. **Desarrollo de un Dashboard Interactivo**
    - Diseñar un dashboard que permita monitorear el desempeño del modelo y visualizar las transacciones detectadas como sospechosas, facilitando la interpretación por parte de los analistas de fraude.
4. **Experimentar con Deep Learning**
    - Investigar el uso de redes neuronales profundas, como LSTM, que podrían capturar patrones temporales en las transacciones y mejorar la detección de fraudes.

---

## Conclusión final

El proyecto de detección de fraude con tarjetas de crédito demostró ser un ejercicio efectivo para aplicar técnicas de ciencia de datos en un contexto crítico. Los resultados mostraron que es posible diseñar modelos con un alto grado de precisión y recall, siempre que se aborden los desafíos inherentes al problema, como el desbalance de datos y la selección de hiperparámetros.

Este proyecto subraya la importancia de la optimización continua y la validación en entornos reales para garantizar un impacto positivo y sostenible en la detección de fraudes financieros. El uso de modelos avanzados y un enfoque basado en datos promete revolucionar la forma en que las instituciones abordan la seguridad transaccional.

---

## Repositorio en GitHub

- https://github.com/JHiguerosG/ProyectoDS

## Referencias


- https://claude.ai/new
- https://docs.profiling.ydata.ai/latest/
- https://seaborn.pydata.org/tutorial.html
- https://numpy.org/devdocs/user/index.html
- https://matplotlib.org/stable/users/index.html
- https://scikit-learn.org/stable/user_guide.html
- https://scikit-learn.org/1.5/modules/neighbors.html
- https://pandas.pydata.org/docs/user_guide/index.html
- https://www.youtube.com/watch?v=Qnth2VXopLg&t=192s
- https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data
- https://docs.profiling.ydata.ai/latest/getting-started/installation/
- https://scikit-learn.org/1.5/modules/generated/sklearn.linear_model.LogisticRegression.html
- https://imbalanced-learn.org/dev/references/generated/imblearn.over_sampling.SMOTE.html#
- https://scikit-learn.org/1.5/modules/generated/sklearn.ensemble.RandomForestClassifier.html
- https://scikit-learn.org/1.5/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html


## Libros consultados


- Data Science Handbook: A practical approach - Kolla Bhanu Prakash
- Ciencia de datos desde cero: Principios básicos con Python (Segunda Edición) - Joel Grus

- Estadística - Serie Schaum (Sexta Edición) - Murray R. Spiegel, PhD & Larry J. Stephens, PhD
- Python para análisis de datos: Manipulación de datos con pandas, NumPy y Jupyter (Tercera Edición) - Wes McKinney
- Inteligencia de negocios y analítica de datos: Una visión global de Business Intelligence & Analytics - Luis Joyanes Aguilar