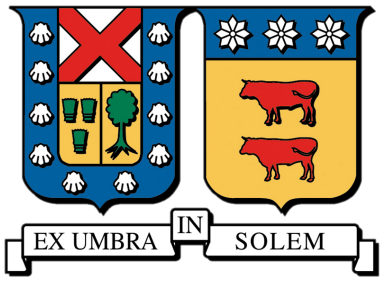

Universidad Técnica Federico Santa María

MAT-306

Detección de Transacciones Fraudulentas en Tarjetas de Crédito

---

Proyectos Estadísticos

Autores:
Bastián Aceitón
Vicente Frías

2024-2

Índice

Objetivo del trabajo

Crear un modelo capaz de identificar potenciales fraudes en transacciones bancarias realizadas con tarjetas de crédito, dicho modelo, eventualmente debe ser rápido (en cuanto a velocidad predicción) y preciso (en cuanto a tener un buen manejo de falsos positivos y falsos negativos).

Análisis del Problema

En esta sección, se examina el problema de la detección de transacciones fraudulentas en tarjetas de crédito, las características necesarias para un sistema de detección eficiente y los criterios que determinarán su éxito.

Contexto del Problema

El fraude en tarjetas de crédito es un problema grave que afecta tanto a las instituciones financieras como a los usuarios. Las compañías necesitan identificar de manera rápida y eficiente las transacciones fraudulentas, de manera que se minimicen las pérdidas y se proteja al cliente. Este informe se enfoca en la creación de un modelo capaz de detectar fraudes, utilizando una base de datos que contiene transacciones realizadas con tarjetas de crédito en septiembre de 2013 por titulares de tarjetas europeos. Este conjunto de datos presenta transacciones que ocurrieron en dos días, donde tenemos 492 fraudes de un total de 284,807 transacciones. El conjunto de datos está altamente desbalanceado, la clase positiva (fraudes) representa el 0.172 % de todas las transacciones.
El dataset, Contiene solo variables de entrada numéricas, que son el resultado de una transformación PCA. Desafortunadamente, debido a problemas de confidencialidad, en el dataset no están las características originales ni más información de fondo sobre los datos. Las características V1, V2, ... V28 son los componentes principales obtenidos con PCA, y las únicas características que no han sido transformadas con PCA son 'Time' y 'Amount'. La característica 'Time' contiene los segundos transcurridos entre cada transacción y la primera transacción en el conjunto de datos. La característica 'Amount' es el monto de la transacción, y esta característica puede usarse, por ejemplo, para aprendizaje sensible al costo dependiendo del ejemplo. La característica 'Class' es la variable de respuesta y toma el valor 1 en caso de fraude y 0 en caso contrario.

Características deseadas

Para que el sistema sea efectivo en la detección de fraudes, se deben cumplir las siguientes características:

1. **Detección de fraudes en tiempo real con alta precisión:**

Es ideal que el sistema logre identificar las transacciones fraudulentas a medida que ocurren, sin demoras que puedan afectar al cliente o que permitan que se materialice el fraude. Además, la precisión del modelo es clave, ya que es necesario evitar la no detección de fraudes (falsos negativos) y la identificación incorrecta de transacciones legítimas como fraudulentas (falsos positivos).

2. **Minimización de falsos positivos y falsos negativos:**
   Debe haber un equilibrio entre la detección de fraudes y las interrupciones innecesarias de las transacciones legítimas. Minimizar los falsos positivos es esencial para no generar molestias a los clientes y evitar bloqueos injustificados a sus cuentas. También, minimizar falsos negativos es vital para asegurar que la mayor cantidad posible de transacciones fraudulentas sea detectada.

3. **Capacidad para procesar grandes volúmenes de datos de manera eficiente:**
   Debido a que las instituciones financieras manejan cientos de miles de transacciones diariamente, el sistema debe estar optimizado de tal forma que procese grandes volúmenes de datos en forma eficiente. Esto implica poder realizar análisis precisos y rápidos, todo esto sin influir en la capacidad de identificar las actividades fraudulentas.

Criterios de Éxito

Para asegurar el éxito del modelo de detección de fraudes, se deben cumplir los siguientes criterios:

1. **Área bajo la curva de Precision-Recall (AUPRC):**
   Por el desbalance de las clases, el rendimiento del modelo se evaluará mediante la AUPRC, la cual es más adecuada para medir en contextos en donde las clases están desbalanceadas. Un alto AUPRC asegura que el modelo equilibre bien la precisión y la recuperación, minimizando tanto los falsos positivos como los falsos negativos.

2. **Tasa de detección de fraudes (Recall):**
   El modelo debe maximizar la detección de fraudes, reduciendo al mínimo los casos no detectados. Un Recall elevado es crucial para proteger tanto a la empresa como a los clientes de pérdidas financieras debido a fraudes no detectados.

3. **Eficiencia e integración en tiempo real:**
   El modelo debe integrarse en el flujo de transacciones en tiempo real, procesando los grandes volúmenes de datos que se manejan de forma rápida y eficiente, sin generar demoras que afecten la experiencia del cliente. Además, este debe ser flexible y saber adaptarse a posibles nuevos patrones de fraude.

Solución Propuesta

En esta sección se presenta la solución planteada para abordar el problema de la detección de fraudes. Se describen los requerimientos funcionales necesarios para que el sistema sea efectivo y las técnicas y modelos específicos que se utilizarán para su implementación.

Requerimientos funcionales

Para la efectividad y eficiencia del modelo se deben cumplir una serie de requerimientos funcionales, los cuales aseguran que este no sólo detecte fraudes con precisión, sino que también se integre de forma adecuada en el entorno de producción y se mantenga un rendimiento óptimo a lo largo del tiempo.

1. **Automatización de la detección:**
   El sistema debe ser capaz de identificar fraudes en tiempo real sin intervención humana.
2. **Acceso y procesamiento de datos:**
   Se deben poder cargar y procesar grandes volúmenes de datos.
3. **Evaluación con métricas adecuadas:**
   Se debe utilizar la métrica AUPRC, ya que esta es más adecuada para conjuntos de datos desbalanceados.
4. **Integración en producción:**
   El modelo se debe integrar sin problemas dentro del pipeline de transacciones, esto para asegurar su funcionamiento en tiempo real.
5. **Monitoreo continuo:**
   El sistema debe ser monitoreado y ajustado dependiendo de las necesidades que vayan surgiendo, adaptándose a nuevos patrones de fraude y a cambios en los datos.

Técnicas y Modelos de la solución

El sistema de detección de fraudes requiere el uso de técnicas y modelos de machine learning que puedan manejar clases desbalanceadas y además asegurar un alto rendimiento.

1. **Modelos:**
   La bibliografía ([1], [2] y [3]) recomienda utilizar modelos robustos, tales como: *Random Forest, Gradient Boosting* y *Redes Neuronales*, ya que estos han demostrado ser eficaces en la detección de fraudes.
2. **Técnicas de balanceo:**
   Para el desbalanceo de los datos, las técnicas utilizadas son el *oversampling* o *undersampling*, las cuales ayudan a equilibrar las clases y servirá para mejorar la capacidad predictiva del modelo.

3. **Métricas:**
   Debido al contexto de desbalance de datos es que se utiliza la métrica AUPRC, ya que esta refleja mejor el balance entre la detección de fraudes y la minimización de falsos positivos.

Planificación

Es esencial, para garantizar una correcta implementación del modelo, establecer un plan de trabajo claro que defina el enfoque, la metodología y los roles del equipo.

Enfoque de Trabajo

El enfoque va a ser iterativo, lo que permite ajustes continuos basados en el rendimiento del modelo y las necesidades que vayan surgiendo. Esto facilita la adaptación a nuevos patrones de fraude y a posibles cambios en el flujo de datos.

Metodología

Se aplicará una metodología ágil, con iteraciones cortas y reuniones regulares de seguimiento para evaluar avances y priorizar tareas. Esto garantiza flexibilidad en la toma de decisiones y permite al equipo responder a imprevistos en el análisis de datos o resultados de los modelos.

Roles

1. **Analista de datos:**
   Encargado del proceso de preparación y exploración del conjunto de datos. Esto incluye la limpieza, el procesamiento y gestión del desbalance de clases.

2. **Desarrollador de modelos:**
   Se encarga del desarrollo, entrenamiento y evaluación de los modelos de machine learning.

3. **Integrador de sistemas:**
   Responsable de integrar correctamente los modelos entrenados en el pipeline de transacciones en tiempo real.

Tareas de la primera iteración

1. **Exploración de datos:**
   Se limpian y estandarizan los datos disponibles, además de identificar patrones claves.

2. **Entrenamiento inicial del modelo:**
   Se desarrolla un modelo de base y se entrena con la base de datos proporcionada.

3. **Evaluación del rendimiento:**
   Se evalúan los primeros resultados obtenidos por el modelo, utilizando la métrica AUPRC. Según los resultados obtenidos, se ajusta y mejora el modelo.

4. **Documentación y seguimiento:**
   Se registran los resultados y avances. Se establecen tareas para la siguiente iteración.

Bibliografía

[1] Gareth James et al. *An introduction to statistical learning*. Vol. 112. Springer, 2013.

[2] Simon Rogers y Mark Girolami. *A first course in machine learning*. Chapman y Hall/CRC, 2016.

[3] Stuart J Russell y Peter Norvig. *Artificial intelligence: a modern approach*. Pearson, 2016.