DETECCIÓN DE FRAUDE

Máster Data Science

21 de julio de 2023

Tutor de proyecto: Carlos Moreno Morera

Raquel López Martínez
Marina Vázquez Vallejo
Mª Gisela Vallejos Velarde

ÍNDICE DE CONTENIDOS

I. Equipo de proyecto y objetivo

Somos Raquel, Marina y Gisela, estudiantes del máster de Data Science de Pontia.tech - promoción febrero 2023.

Decidimos desarrollar el proyecto de detección de fraude, puesto que nuestro objetivo era tener una experiencia lo más similar posible a un caso real de empresa, y el fraude está a la orden del día.

Aunque trabajamos codo con codo en cada uno de los apartados del proyecto, las tareas realizadas por cada una de nosotras fueron asignadas equitativamente, en función del dominio en el área de cada una y de la tarea a realizar.

Entrando en materia del proyecto, dado que Pontia Bank SL no llevó a cabo ningún tipo de procesamiento y/o análisis sobre los datos de los que disponen, nuestro objetivo ha sido realizar una transformación y análisis exhaustivo de los datos facilitados por la empresa para finalmente crear un modelo capaz de automatizar la detección de fraude, pasando por la búsqueda de incidencias en los datos y la respuesta a varias preguntas de negocio.

II. ETL y EDA

En primer lugar, nos centramos en el proceso ETL (extracción, transformación y carga de datos) comenzado con los archivos facilitados por Pontia Bank SL en formato json.. Este proceso se basa en importar dichos documentos, modificar su estructura, crear las tablas necesarias conteniendo información organizada y extraer a varios archivos en formato csv.

Los archivos facilitados contaban con un formato desestructurado, lo que dificulta el proceso de análisis y su manipulación. Para solucionarlo, se crearon varias tablas que contienen la información clara y ordenada.
En la tabla "**balances**" creamos cuatro columnas con los balances previos y posteriores de los clientes participantes en las transacciones, tanto los remitentes como los destinatarios.
Se realizó un proceso similar con la tabla"**clientes**". Ésta contiene la información del cliente origen y destino de cada transacción.
Así mismo, se creó la tabla "**fraude**", que indica por cada transacción si es fraudulenta o no y si ha saltado la señal de alarma del banco por potencial fraude.
Para ello fue modificada la variable "**es_fraude**", siendo su formato original un booleano y pasando a ser un int: 0=false y 1=true. De esta manera, se obtiene un resultado más representativo.
Finalmente, se creó la tabla "**transacciones**", que contiene las siguientes variables:
- "**tiempo**": se modificó su formato, consiguiendo representar la fecha completa de una transacción en formato fecha y hora. El resultado queda en la forma aaaa-mm-dd hh:mm:ss, siendo mucho más representativo.
- "**monto**": inicialmente llamado "**cuantía**", indica el dinero que interviene en la transacción.

- "**tipo**": hace referencia al tipo de la transacción (transferencia, pago de recibo, pago con tarjeta, ingreso y extracción de efectivo).
Cabe destacar que todas las tablas cuentan con la variable "**t_id**", que identifica cada transacción de manera única y que posteriormente sirve para relacionar unas tablas con otras con diversos objetivos.

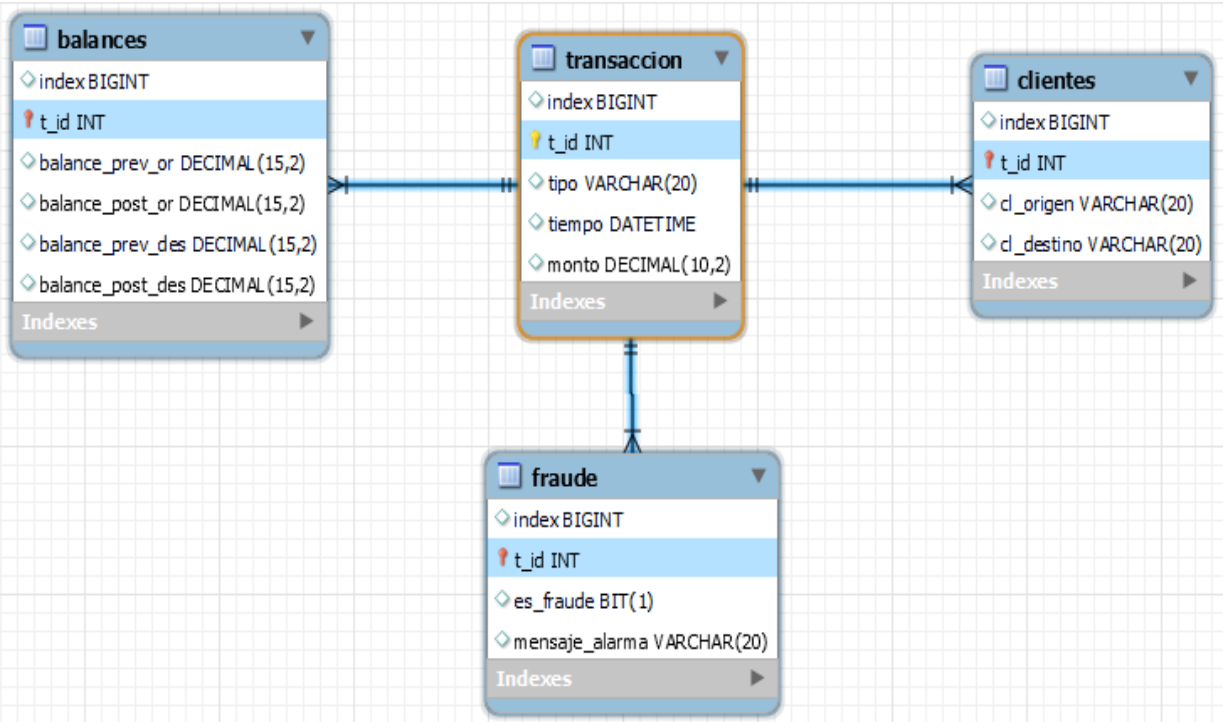
Para finalizar, realizamos un proceso EDA (análisis exploratorio de los datos). En éste se realizan varios procedimientos como la comprobación de datos nulos, la búsqueda de patrones en los datos, su distribución, la aparición de registros duplicados, la existencia de valores anómalos o la presencia de errores en los datos.

Esta es una fase muy importante, puesto que permite conocer la naturaleza y distribución de los datos, la calidad de los mismos o características importantes como la correlación entre variables; lo cual es fundamental para proceder al cálculo de métricas de negocio o a la correcta implementación de modelos de machine learning con el objetivo que corresponda.

III. Modelo relacional

Con las tablas en formato csv, el siguiente paso es la creación e implementación de un esquema relacional de cara a almacenar los datos de manera óptima, así como acceder a ellos y realizar consultas de manera eficiente. Para ello se ha hecho uso de MySQL.

Dicho esquema relacional muestra las cuatro tablas con las que se ha trabajado, así como la información almacenada en cada una de ellas, los tipos de las variables y la manera en que se relacionan entre sí (la variable "t_id", presente en todas ellas).



Haciendo uso de esta herramiento, se procedió a la resolución y cálculo de varias KPIs y métricas de negocio, como la determinación de la media diaria de la cuantía en las transacciones, los clientes con mejor y peor balance a lo largo del mes o la búsqueda de aquellas transacciones fraudulentas.

También se pudieron encontrar errores en los registros, como por ejemplo transacciones en las que el balance de alguno de los clientes participantes no era equivalente al monto de la misma.

IV. Metodología de Machine Learning y Deep Learning

Con el objetivo de implementar un buen modelo capaz de automatizar el proceso de detección de fraude, lo primero fue asegurarse de que los pasos anteriores estaban desarrollados de forma correcta, de modo que el dataset a utilizar fuera exacto, representativo, completo y con datos abundantes y equilibrados.

Las fases a seguir para el desarrollo de un modelo correcto y útil son las siguientes:

- Definición del problema y objetivos: partiendo de la necesidad de automatizar el proceso de detección de fraude en Pontia Bank, el objetivo ha sido la creación e implementación de un algoritmo de Machine Learning y otro de Deep Learning capaces de clasificar como fraudulenta o no cada transacción entrante en la base de datos del banco.

- Selección de datos: por cada transacción registrada escogimos las variables "**monto**", "**balance_prev_or**", "**balance_prev_des**", "**hora**" y "**tipo_binario**".
  "**hora**" contiene la hora a la que se realizó cada transacción (parece, a partir del análisis, influir más en el fraude que el día como tal) mientras que "**tipo_binario**" indica si una transacción es de tipo "transfer" (representada con 1) o "cash_out" (0), pues estas dos son las únicas que generan operaciones fraudulentas.

- Selección del algoritmo: Los datos etiquetados nos dan una pista para saber qué tipo de algoritmo es más adecuado utilizar: uno de aprendizaje supervisado. Además, ha de ser de clasificación, pues queremos clasificar datos en dos categorías (fraude o no fraude). Se implementó con este fin un modelo de regresión logística, un árbol de decisión y una red neuronal artificial (RNA) con 5 capas de neuronas interconectadas.

- Entrenamiento y parametrización del algoritmo: previo al trabajo en cada algoritmo, implementamos SMOTE para generar muestras adicionales de la clase minoritaria (transacciones fraudulentas) de cara a equilibrar la distribución del conjunto de clases en nuestro dataset.

Después, se determinaron los datos de test y entrenamiento para el modelo, quedando los de prueba en un 30% y los de entrenamiento en un 70% del total.
Tras ello, normalizamos los datos empleando el método RobustScaler, pues este es el más apto para tratar con datos que contienen una gran cantidad de outliers.

Finalmente, entrenamos el modelo para cada algoritmo, haciendo uso de los métodos específicos de cada uno de ellos.

- ➢ Evaluación del modelo: una vez se tiene el modelo entrenado, el siguiente paso es evaluarlo para conocer su rendimiento y considerar si el modelo es aceptable para resolver nuestro problema. Para ello, utilizamos dos tipos de métricas, la precisión positiva o PPV (indica cuánto acierta el modelo cuando predice que una transacción es fraudulenta) y la sensibilidad o Recall (indica cuántas de las transacciones fraudulentas detecta bien).
  Para el PPV, exigimos que tuviera una precisión mínima del 0,75 y para el Recall, un valor mínimo de 0,8. Ambos valores van de 0 (lo peor) a 1 (lo mejor).

- ➢ Explotación del modelo: finalmente, aquellos modelos que superan satisfactoriamente la evaluación, pueden ser puestos en funcionamiento y utilizados por Pontia Bank en el día a día.

V. Comparación de modelos

Tras la realización de varios modelos de aprendizaje supervisado, vimos que lo más útil y eficiente, dada nuestra problemática, es utilizar es un árbol de decisión, que logramos devolviera unos valores de 0.996 para la métrica de precisión (PPV) y un 0.998 para la métrica Recall.

```
precision: 0.9966381050086456
recall: 0.9989065867887685
```

El modelo de Deep Learning, la red neuronal, también es muy recomendable para utilizar frente a la problemática de detección de fraude.

En el primer modelo realizamos 3 epochs y los resultados fueron los siguientes: para la métrica PPV obtuvimos un score de 0.978 y para Recall obtuvimos un score de 0.988. También decidimos probar con Accuracy, obteniendo como resultado un score de 0.983.
En el segundo modelo realizamos 10 epochs para una red neuronal diferente y los resultados fueron los siguientes: para la métrica PPV obtuvimos un score de 0.933 y para Recall obtuvimos un score de 0.97. En este caso, también decidimos probar con accuracy, y obtuvimos como resultado un score de 0.95.

Por otro lado, la regresión logística no devolvió malas métricas, 0.898 de precision o PPV y 0.822 de Recall. Ambas hacen el modelo aceptable, pero menos potente y útil frente a cualquiera de los otros dos.

VI. Conclusiones

- A pesar de la gran cantidad de datos que maneja Pontia Bank SL, estos no se encontraban originalmente en el mejor formato y orden posible, por lo que la implementación de una base de datos relacional supondrá grandes mejoras para el negocio.

- Muchos de los registros en los datos contienen errores en los que hay que poner el foco y solucionar para un mejor desarrollo de la actividad comercial y experiencia del cliente.

- La gran mayoría de KPIs calculadas arrojan información muy positiva sobre el buen funcionamiento del banco.

- Se ha logrado el objetivo principal de automatizar la tarea de detección de fraude de manera muy eficiente y con errores mínimos, gracias a la implementación de varios algoritmos de machine learning. Destacar de entre todos el árbol de decisión por su buen desempeño.

VII. Referencias

Repositorios donde se pueden encontrar todos los archivos ejecutables y procesos realizados durante el proyecto:

- **GitHub**: Repositorio DDF - Data Science Pontia 2023
- **Google Drive:** Detección de Fraude - Data Science Pontia 2023

Documentación de soporte utilizada durante el proyecto:

- https://www.mysql.com
- https://scikit-learn.org/stable/
- https://www.tensorflow.org/?hl=es-419
- https://es.stackoverflow.com
- https://desarrolloweb.com/articulos/1054.php

Sitios web utilizados:

- https://www.canva.com
- www.tome.app