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 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, ya que el fraude está a la orden del día.

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

Puesto que Pontia Bank SL no llevó a cabo ningún tipo de procesamiento y/o análisis sobre los datos de los que disponen, nuestro objetivo fue 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 este paso, nos centramos en el proceso ETL (extracción, transformación y carga de datos) con los archivos facilitados por Pontia Bank SL en formato json para detectar, a posteriori, incidencias o errores, tanto de formato como de contenido. Este proceso consta de importar dichos archivos, modificar su estructura y extracción a varios archivos en formato csv.

Los archivos facilitados, venían con un formato desestructurado, lo que dificulta el proceso de análisis. Para solucionarlo, creamos varias columnas nuevas derivadas de las ya dadas.
De la columna "**balances**" creamos cuatro columnas nuevas para los balances previos y posteriores de los clientes, tanto remitentes como destinatarios.
Se realizó un proceso muy parecido con la columna "**clientes**". Esta columna fue desglosada en dos: cliente remitente y cliente destinatario.
También fue modificado el valor de "**es_fraude**". El formato original del dato de esta columna era de tipo booleano y fue modificado a dummy, siendo 0=false y 1=true. De esta manera obtenemos un resultado más representativo.
De la columna "**tiempo**" fue modificado el formato, consiguiendo representar la fecha completa de una transacción en formato fecha y hora. El resultado queda en formato aaaa-mm-dd hh:mm:ss, mucho más representativo.
El resto de columnas no fueron modificadas, a excepción del título de la columna "**cuantía**", que fue modificado a "**monto**", pero su contenido no fue alterado.

Este proceso nos sirve para, en futuros pasos, poder relacionar unas columnas con otras de forma más eficiente y llegar a conclusiones más específicas.

Para finalizar, realizamos un proceso EDA (análisis exploratorio de los datos). Este paso consta de analizar los archivos para comprobar nulos, patrones o incidencias en los datos.

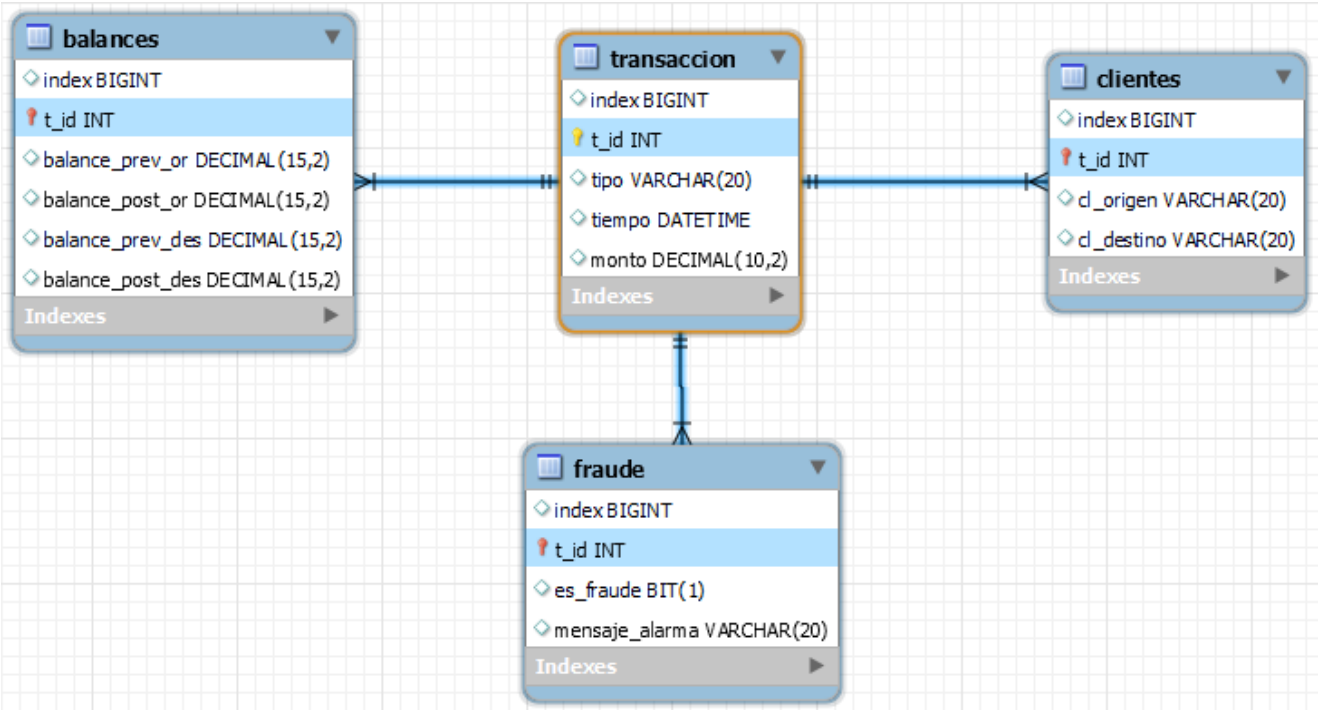
Esta es una fase muy importante, puesto que podemos conocer la estructura de los datos y la calidad de su contenido para tomar las mejores decisiones con datos fiables.

III. Modelo relacional

Con los archivos en formato csv, el siguiente paso es la creación e implementación de un esquema relacional que nos permita almacenar datos de las transacciones e información en tablas.

Con este esquema relacional implementado, el entorno MySQL es capaz de realizar las consultas requeridas por la empresa.

Vemos en la imagen adjunta, que todas nuestras tablas están vinculadas a una misma clave primaria o primary key (PK), lo que nos permitirá realizar consultas con otras tablas.



IV. Metodología de Machine Learning y Deep Learning

Para implementar un buen modelo capaz de automatizar el proceso de detección de fraude, tenemos que asegurarnos primero de que los pasos anteriores están desarrollados de forma correcta para que el dataset a utilizar sea 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 es 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.

- Selección de datos: para ello, por cada transacción realizada, utilizamos las variables "**monto**", "**balance_prev_or**", "**balance_prev_des**", "**hora**" y "**tipo_binario**". Los datos etiquetados nos dan una pista para saber qué algoritmo es más adecuado utilizar, en este caso, uno de aprendizaje supervisado para clasificación, en este caso, un árbol de decisión en el caso de Machine Learning, y una Red Neuronal Artificial (RNA) con 4 capas de neuronas interconectadas, en el caso de Deep Learning.

- Selección del algoritmo: para la elección del algoritmo, es necesario analizar la información que tenemos y el problema que queremos resolver. En base a ello, decidimos utilizar un árbol de decisión para resolver un problema de clasificación basado en aprendizaje supervisado, en el caso de Machine Learning, y una Red Neuronal, en el caso de Deep Learning, para el mismo fin.

- Entrenamiento y parametrización del algoritmo: sabiendo qué algoritmo vamos a utilizar, 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.

  El siguiente paso previo a explotar un modelo, es determinar los datos de prueba y entrenamiento, siendo los de prueba un 30% y los de entrenamiento un 70%.
  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.

- ➢ Evaluación del modelo: una vez tenemos 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 tres tipos de métricas, la precisión positiva (PPV) y la sensibilidad (Recall o TPR). Para el PPV, exigimos que tuviera una precisión mínima del 0,55 y para el Recall, un valor mínimo de 0,65.

- ➢ Explotación del modelo: ya vimos en el anterior paso de selección de datos, que para explotar un modelo íbamos a utilizar variables relevantes e influyentes para determinar si una transacción es o no fraudulenta.

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, con un score del 0.996 utilizando la métrica de precisión (PPV) y un 0.998 utilizando la métrica Recall.

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

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

En el caso (**A**), realizamos 3 epochs y los resultados fueron los siguientes: utilizando las métricas de PPV obtenemos un score de 0.974, con Recall obtenemos un score de 0.981. También decidimos probar con accuracy, y obtuvimos como resultado un score de 0.967.
En el caso (**B**), realizamos 10 epochs y los resultados fueron los siguientes: utilizando las métricas de PPV obtenemos un score de 0.950, con Recall obtenemos un score de 0.970. En este caso, también decidimos probar con accuracy, y obtuvimos como resultado un score de 0.933.

```
[0.06874208897352219,
 0.9743243455886841,
 0.9813252091407776,
 0.9678197503089905]
```

(**A**)

```
[0.13971814513206482,
 0.9504565000534058,
 0.970365583896637,
 0.9332886338233948]
```

(**B**)

Como detalle, el primer valor que aparece nos indica el valor de pérdida (**loss**) en cada iteración epoch. Este valor de pérdida o loss, es una función objetiva que nos indica qué tan bien está funcionando nuestro modelo. Un valor de pérdida bajo nos indica que nuestro modelo se ajusta bien a los datos de entrenamiento.

VI. Conclusiones

- ➢ Además de las consultas exigidas, vimos que el monto mínimo y máximo de los clientes es de 0.1€ y 92.445.516,64€, respectivamente.

- ➢ Se determinó que todas las variables del dataset influyen en si una transacción es fraudulenta o no, a excepción de las transacciones tipo "**payment**, "**cash_in**" y "**debit**", que nunca son fraudulentas.

- ➢ El dataset original tenía una buena cantidad y calidad de datos para poder trabajar con ellos de forma eficiente, pese a haber modificado su estructura.

- ➢ Tras el desarrollo de nuestro modelo de Machine Learning, el árbol de decisión, y nuestra red neuronal, podemos sacar la conclusión de que ambos son aptos y muy recomendables para utilizar frente a la problemática de detección de fraude.

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