**Prevención de fraude**

**INTRODUCCION**

Uno de los principales riesgos a los que están sometidas las entidades financieras son los ataques de fraudes electrónicos. Billones de dólares en pérdidas son absorbidas cada año por las entidades financieras debido a transacciones fraudulentas.

La necesidad de las empresas y entidades bancarias cuyos usuarios gestionen operaciones y transacciones por internet eleva el potencial riesgo que dichos procesos sean manipulados por hackers poniendo en peligro la integridad de los datos de los usuarios y la información reservada de las entidades bancarias

El fraude contra el sistema financiero constituye un gran problema para los bancos ya que ocasionan pérdida económica, pérdida de imagen y desconfianza de los clientes.

Con este análisis se propone un modelo de Machine Learning (ML) que pueda identificar transacciones fraudulentas, para luego alimentar al mismo modelo con datos nuevos, y poder tomar acción sobre compras peligrosas.

**MODELO**

Primero, se buscó una base de datos en donde haya información relevante como para crear un modelo aplicado a prevenir fraudes con tarjetas de crédito, se encontró en Kaggle.com ([link](https://www.kaggle.com/kerneler/starter-credit-card-transactions-fraud-cd58bfd8-a/notebook)) un dataset con más de un millón doscientas mil filas y 24 columnas.

Por el volumen de datos, se crearon dos bases del mismo archivo, uno tomando 600.000 filas y otro de test con 200.000. Para ello se utilizó Pandas:

df\_train = df1.iloc[:600000,:]df\_test = df1.iloc[600001:800000,:]

Una vez definida la base, comenzamos con la construcción de nuestra notebook. Primero importamos las librerías que utilizaremos, y se importa el archivo .csv



Comenzamos a revisar la data, y nos encontramos que quedaron 600000 filas, y 24 columnas. Con el método .info revisamos el tipo de dato y confirmamos que no existen nulls.

Para evitar sobrecargar el análisis con datos que no serán de utilidad, eliminamos algunas columnas y luego revisamos con qué nos quedamos

Imagen que contiene Texto

Descripción generada automáticamente

Y, mediante el .describe, revisamos los Principales Estadísticos

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Por medio de Pandas Profilling revisamos los datos con más detalle, y habiendo conseguido verificar la data importante, comenzamos con los primeros análisis.

**ANÁLISIS UNIVARIADO**

Nos parece interesando conocer información sobre dos variables ‘category’ y ‘job, para saber qué tipo de datos son (top 5):

Gráfico, Gráfico de barras

Descripción generada automáticamente

Gráfico, Gráfico de barras

Descripción generada automáticamente

También revisamos otras variables como ‘state’ y ‘amt’, para conocer su densidad, utilizando otro tipo de gráficos: líneas e histograma.

**DISTRIBUCION DE FRECUENCIAS**

Nos encargamos de conocer la frecuencia absoluta y acumulada de la variable ‘category’, para lo cual creamos un DataFrame con la información. Primero conseguimos los valores de las frecuencias absolutas, para luego crear una función que en una lista vacia guarde las frecuencias acumuladas y vaya iterando la lista para ir sumándolas.

Texto

Descripción generada automáticamente

Y, por último, calculamos la frecuencia relativa (%) y la acumulando. Quedándonos la siguiente información:

Tabla

Descripción generada automáticamente

**ANÁLISIS BIVARIADO**

Al contar con tanta información y al ser transacciones de diferentes montos (‘amt’), la variable cuenta con valores concentrados en por de bajo de los 30USD pero tiene otro muy altos. En el gráfico del análisis univariado de ‘amt’ de puede ver claramente:

Gráfico, Histograma

Descripción generada automáticamente

Como primer análisis bivariado se seleccionaron las variables ‘gender’ y ‘amt’, para revisar si existe alguna correlación, y luego se agregó la variable que determina si la transacción fue fraudulenta o no, habiendo quedado un resultado como este:

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Y, además, se incluyó un mapa de calor para poder ver la correlación de datos entre las variables en análisis:

Gráfico, Gráfico de rectángulos

Descripción generada automáticamente

**ANÁLISIS MULTIVARIADO**

Dado que nuestro dataset cuanta con pocas variables numéricas, sólo realizamos un Pairplot para dimensionar la correlación de las variables, diferenciando los casos en el que la transacción fue fraude y cuáles no, habiendo quedado algo así:

Imagen que contiene Gráfico de dispersión

Descripción generada automáticamente

Mediante este análisis logramos tener una perspectiva más clara de en qué variables hay que hacer hincapié a la hora de generar el modelo.

**FEATURE ENGINEERING**

Con .get\_dummies generamos nuevas features, y así logramos hacernos de variables discretas que nos permitirán profundizar el análisis al relacionar las nuevas features.

El método .get\_dummies crea columnas en nuestra base con valores numéricos, en referencia a ciertas variables categóricas como ‘gender’ y ‘category’.

Tomamos esas dos variables porque creímos las más relevantes para nuestro objetivo. Por ejemplo, conociendo la categoría del comercio podemos detectar si es existe cierta frecuencia de transacciones fraudulentas en algún tipo de actividad.

Además, se quiso normalizar la columna ‘amt’ (amount), pero resultó en un error en el modelo lineal. Da mejor resultado dejando el dato tal como se obtuvo.

**DATA FLOW**

Se podría decir que el flujo del proceso de ciencia de datos se compone por:

1. Ingesta de datos
2. Data cleaning
3. Data featuring
4. Selección de modelo
5. Evaluación del modelo
6. Resultados

Habiendo trabajado en los primeros tres puntos, es que podemos seleccionar el modelo que sea acorde a nuestro objetivo y se pueda utilizar según nuestra data.

**OBJETIVO Y SELECCIÓN DE MODELOS**

Nuestro objetivo es detectar transacciones fraudulentas, es por ello que creemos conveniente probar con dos modelos: Linear Regression y RandomForest.

Es decir, que los dos modelos seleccionados son supervisados. Pero de ambos, es conveniente trabajar con el Linear Regression, por el tipo de resultado que buscamos (0 y 1).

Igualmente incorporamos en nuestra notebook al RandomForest para poder experimentar con otro modelo.

Se construyó la Matriz de Confusión, arrojando estos resultados:

Gráfico, Gráfico de rectángulos

Descripción generada automáticamente

Y, también, se hizo un benchmark para comparar los resultados de los modelos mediante una línea ROC para validar los resultados del modelo Linear Regression:

Gráfico, Gráfico de líneas

Descripción generada automáticamente