Fighting Fraud

# Overview

In this project, you work for a major financial institution who is struggling with new account fraud after running a new promotion on certificate of deposits. Fraudsters are opening fake accounts known as “synthetic fraud” in a legitimate customer’s name, they then transfer funds from the legit customer to these new fraudulent accounts, after a period of time, they then close the account and running off with the money!

Your job is to 4-fold:

* First you will need to extract a table of data from a Postgres cloud database.
* Second you will need to enrich the IP Address field by calling the IP address lookup from “freeipapi.com, you will need to create a number of new columns (see source to target mapping).
* Third you will need to train a predictive model to predict the EVENT LABEL outcome – follow the recipe you’ve learned from Dr. Bayan. I don’t care what method you use – BUT you must generate add 3 columns to your data Probability of Fraud, Probability of Legit, Predicted (Fraud / Legit)
* Fourth you will need to use your EXCEL powers to create a visual that answers the following:

1. Is credit score predictive of fraud or not?
2. For your top 5 predictors generate a visual that explains the relationship between fraud and the predictor.
3. A score distribution
4. At a predicted probably of fraud of 0.5 and above what is the accuracy, precision and recall.

Tasks

## Project Setup and Data Extraction

* Connect to a remote PostgreSQL database and extract the fraud data.
* Install necessary libraries and tools
  + !pip install psycopg2-binary
  + !pip install --upgrade SQLAlchemy
* Connect to Postgres database on the cloud.
  + DATABASE\_URL = "postgres://ekolufgpivierh:038a6bd29784f2ca479ebaf9584333c53fb4cd1dc99085a4f7cabaeb7936c55c@ec2-35-169-11-108.compute-1.amazonaws.com:5432/d4tj4h4dvtrjks"
  + conn = psycopg2.connect(DATABASE\_URL, sslmode='require')
  + .
* Write an SQL query to extract the customer\_accounts table from the databse.

## Data Enrichment

* Enrich the data by adding additional information based on IP addresses.
* Tasks:
  + Extract the unique IP addresses from the data.
  + For each unique IP address, call the IP address lookup service `https://freeipapi.com/api/json/{ip\_address}`.
  + Parse the JSON response from the IP address lookup service and add the additional information (e.g., country, city, etc.) to the data.
  + Handle any potential issues with the API requests, such as rate limiting or errors.

## Data Loading

* Load the enriched data into a local MySQL or SQLite database.
* Tasks:
  + Create a table(s) in the database to store the enriched fraud data.
  + Load the enriched data into the database to a table called “CUSTOMER\_PREP”

## Train a Classification Model

Using your new found powers of classification, you will now use this enriched dataset to build a classification model to predict fraud or legit. In python use a train test split, train the model, evaluate it then “score” your full dataset CUSTOMER\_PREP appending three columns to that dataset (probability\_fraud, probability\_legit, prediction)

* You’ll tourn in your notebook showing how you created and evaluated the model.

## Visualize your Data

Using your mastery of Excel and Databases, Connect Excel to your database, create queries and visualizations to answer the question.

1. Is credit score predictive of fraud or not?
2. For your top 5 predictors generate a visual and put them into a dashboard that explains the relationship between fraud and the predictor. Have a short explanation of how to interpret each chart.
3. A score distribution
4. At a predicted probably of fraud of 0.5 and above what is the accuracy, precision, and recall.
5. you must answer is the accuracy of the model better than the default accuracy.

## 6. Project Submission

Submit the completed project.

* Tasks:
  + Prepare an excel file that summarizes the findings, visualizations, and evaluations.
  + Include any code, queries, and visualizations created during the project.

# Evaluation Criteria:

* Code Quality: Is the code well-written, commented, and organized?
* Data Enrichment: Was the data successfully enriched with additional information?
* Visualization: Are the visualizations clear, relevant, and informative?
* Model Evaluation: If applicable, was the machine learning model evaluated properly?
* Presentation: Is the final report/presentation well-structured and informative?

## Source to Target MAPPING

|  |  |  |  |
| --- | --- | --- | --- |
| **Source Column** | **Transformation/Action** | **Target Column** | **Description** |
| email\_age | replace missing values with median | email\_age | Age of the email used in the transaction. Missing values are replaced with the median email age. |
| customer\_tenure | replace missing values with median | customer\_tenure | Duration of the customer's relationship with the service. Missing values are replaced with the median tenure. |
| balance\_inqury\_count | replace missing values with median | balance\_inqury\_count | Number of balance inquiries made by the customer. Missing values are replaced with the median inquiry count. |
| balance\_current\_amt | replace missing values with median | balance\_current\_amt | Current balance amount in the customer’s account. Missing values are replaced with the median amount. |
| current\_customer | default to False | current\_customer | Indicator of whether the customer is currently active. Missing values are defaulted to False. |
| email\_domain | default to "Missing" | email\_domain | Domain of the customer’s email address. Missing values are defaulted to "Missing". |
| business\_state | default to NA | business\_state | State of the customer’s business. Missing values are defaulted to "NA". |
| credit\_score | default to mean | credit\_score | Credit score of the customer. Missing values are replaced with the mean credit score. |
| EVENT\_LABEL | - | EVENT\_LABEL | Label indicating whether the event is fraudulent or legitimate. No transformation required. |
| ip\_address | - | IP Address | IP address associated with the transaction. Used for enrichment; no transformation required. |
| ip\_address | API Call to IP Lookup: **Provider** | Provider | Internet service provider information retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **IP Version** | IP Version | IP version (IPv4 or IPv6) retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Latitude** | Latitude | Geographical latitude information retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Longitude** | Longitude | Geographical longitude information retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Country Name** | Country Name | Name of the country retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Country Code** | Country Code | Country code retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Time Zone** | Time Zone | Time zone information retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **ZIP Code** | ZIP Code | ZIP code information retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **City Name** | City Name | Name of the city retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Region Name** | Region Name | Name of the region or state retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Continent** | Continent | Continent information retrieved based on the IP address. |
| ip\_address | API Call to IP Lookup: **Continent Code** | Continent Code | Continent code retrieved based on the IP address. |