Project Journal

Project Title: *Data-Driven Insights for Banking: Financial and Consumer Behaviour Analysis*

Member: Gonzalo Mauricio Bugueño Cuadra

Week 1: Dataset selection and analysis

Three datasets were selected for the group:

- CreditRiskAnalytics' Home Equity Dataset
- Portugal Bank Marketing Dataset
- Taiwanese Bank Default of Credit Clients

For the first dataset, the shape, non-null count, and data types of columns are examined using pandas' detection methods.

Data preprocessing steps, such as data cleaning were performed to ensure consistency in datatypes.

Hypotheses regarding missing values are outlined:

- Zero years at job doesn't mean debt to income ratio is null, meaning 0 years at job doesn't mean unemployed
- A null value as the occupational category `job` doesn't mean the person is unemployed, as they have a populated debit to income ratio`
- Null values may have been added by the data provider to alter the data randomly as to not reflect reality too closely.

The straightforward method of handling missing data avoids complexities of imputation but reduces the dataset size.

Comparing descriptive statistics before and after removing null values ensures consistent patterns and insights.

Time Spent: **6 hours**

Challenges:

Datasets that would align with little to no effort to each other's structure would've been ideal -to join everything and make one big dataset, - but finding such datasets was an impossible task. Most datasets have no reputable source, meaning they may be synthetic, which is not true to the real world. Secondly, their nature and characteristics made them impossible to join, from number-only datasets to wildly different structure.

At first, it was thought that missing values in columns such as YOJ, JOB, and DEBTINC would represent unemployment or particular financial situations. These presumptions weren't always accurate. Utilised inference to comprehend the significance and context of missing values.

Week 2: Datatype manipulation

Based on their observable properties, columns are cast to the appropriate data types to guarantee accuracy and consistency in data processing. To reflect their distinct, non-fractional character, numerical columns like MORTDUE, VALUE, YOJ, NINQ, CLNO, DEROG, and DELINQ—which stand for mortgage dues, property values, years on the job, inquiries, credit lines, derogatory reports, and

delinquencies, respectively—are transformed to integers. In order to better reflect its binary nature and facilitate interpretation and processing in subsequent analyses or model training, the BAD column—which indicates loan default status—is also changed to a boolean type. By carefully adjusting the datatype, each column is guaranteed to correspond with its actual meaning, enabling more concise calculations, visualisations, and insights.

Detecting Outliers:

A function that performs the IQR technique with a panda's DataFrame was implemented for easier detection. In the MORTDUE column, the IQR technique finds 139 outliers dispersed over three clusters. A boxplot of the whole dataset was employed to visualise the outliers detected. Same for the VALUE column, with 503 outliers detected which seem clustered within three or four clusters. Same efforts were performed for the rest of the numeric columns, but no outliers were detected within them.

Time Spent: **6 hours**

Week 3: Visualisation

I experimented with several visualisation and interpretation techniques for the data from my project. I made a number of graphs to help me comprehend the distributions and correlations of features:
Histograms
I examined the distribution of important numerical characteristics, such as loan amount, mortgage due, and debt-to-income ratio, by JOB category using histograms. I was able to see how these factors alter depending on the sort of employment thanks to this method. For example, I observed trends in loan applications and property values that differed by employment.

Pie Diagrams
I then used pie charts to illustrate categorization data, such as loan reasons, employment kinds, and t he top 5 counts for recent queries and credit lines.
By giving me a brief overview of proportions, these visualizations assisted me in determining which ca tegories predominate in the dataset.

Lastly, I employed a pairplot to investigate correlations between numerical parameters that were color-coded according to the BAD target variable. This was a great approach to observe how characteristics like debt-to-income ratio, loan amount, and mortgage due interact and whether any of them would indicate a possible default (as indicated by BAD).

Challenges:

Coming up with visualisations is hard, because one needs to understand the dataset and the context where it is being employed to come up with ideas for visualisations.

Time Spent: **8 hours**

Week 4: Building and Evaluating a Random Forest Classifier

I used a dataset of financial variables to train a Random Forest Classifier to forecast loan defaults. I divided the data into training and testing sets after using LabelEncoder to preprocess the categorical variables (REASON and JOB).With 150 estimators set up, the model's accuracy on the test set was an astounding 94.05%. This achievement demonstrates Random Forest's prowess in categorization challenges and the significance of data preparation. I made a table with SQLite to hold financial

information. Along with a target label label, the table hmeq has a number of attributes, including loan, mortdue, value_, cause, and more.A loop was used to couple each feature row with its matching label when the data from X (features) and Y (target) were entered into the database. I filled in the table, then closed the connection and committed the changes. This will make it simple to retrieve the data for model training or additional research. The trained RandomForestClassifier and the feature encoders (job_encoder and reason_encoder) were saved using Pickle. In order to facilitate simple reloading and future use without retraining, these were saved as .pkl files in the "pickled" folder. The model and encoders are protected for deployment or additional analysis .

Time Spent: **7 hours**

Challenges:

Finding a way to save the model outside memory was tricky: I could've picked using a db and fitting the dataset into a classifier each time a prediction is required, or pickled the file. I chose the former, as the overhead required for the constant training would be too much.

Total Time Spent: 27+ hours

Understanding the dataset, developing insightful visualizations, constructing a prediction model, and putting effective database storage into practice all saw notable advancements over the project.

To make sure the prediction model was user-friendly and accessible, I designed and implemented a dedicated webpage. This interface allows any user to interact with the models. Built with simplicity in mind, it features a clean, straightforward layout, meant for users whose job was related to the datasets picked -bank executive, bank manager, etc.

Also, I made the entire project (excluding the binary files such as databases and pickled objects) available in GitHub to facilitate sharing and visualization.

Regular communication and teamwork also allowed the team to discuss issues and share ideas, ensuring that solutions were implemented collectively. By leveraging one other's skills and working together to solve obstacles, the team was able to move forward efficiently and maintain a high standard of work throughout the project.

Project related meetings :3 hours
Dataset Research: 2hours
Report – 5hours (Reading related research papers)
Presentation