B5W5: Credit Risk Probability Model for Alternative Data - Interim Submission - Report

Gashaye Adugna

Introduction

This report provides a comprehensive overview of the project aimed at developing a credit risk probability model for ati Bank's new buy-now-pay-later (BNPL) partnership with an eCommerce platform. The document outlines the business context, technical progress, and implementation details, ensuring clarity and professionalism throughout. Visual aids, such as tables and charts, will be utilized to enhance comprehension, and the writing adheres to high grammatical standards for a polished final document.

This report outlines the foundational work undertaken for developing a Credit Risk Probability Model. The core business problem addresses the inherent risk associated with offering a "buy-now-pay-later" service, a key offering in our partnership with a leading eCommerce company. As this service expands, accurately assessing the creditworthiness of customers becomes paramount to mitigate potential financial losses from defaults.

The model's development is directly relevant to **Basel II regulatory requirements**. Basel II mandates that financial institutions hold capital reserves proportional to their risk exposure. By developing a robust credit risk probability model, we aim to:

Quantify Risk: Accurately estimate the Probability of Default (PD) for each customer, a critical component for calculating Expected Loss (EL) under Basel II.

Optimize Capital Allocation: Ensure that capital is allocated efficiently, neither excessively tying up resources nor exposing the company to undue risk.

Enhance Decision-Making: Provide data-driven insights for approving or denying credit applications, setting credit limits, and tailoring service offerings.

The partnership with the eCommerce company for the buy-now-pay-later service represents a significant growth opportunity. Our technical work is directly connected to this real-world use case by enabling responsible and sustainable growth of this service. A reliable credit risk model allows the eCommerce platform to extend credit confidently, expand its customer base, and minimize financial exposure, thereby ensuring the long-term viability and profitability of the partnership.

Business Context

ati Bank is embarking on the development of a robust credit scoring system to assess customer creditworthiness using transactional data. This initiative is critical for supporting the BNPL service while complying with the Basel II regulatory requirements. The Basel II Accord emphasizes risk measurement and necessitates the creation of an interpretable model that provides accurate probability of default

estimates. Given the absence of traditional credit history data, the model will rely on proxy risk indicators derived from behavioral patterns. This innovative approach presents challenges related to validation and the management of business risks associated with imperfect proxies.

2. Reporting on Technical Progress

Significant technical progress has been made in establishing a robust and reproducible data processing pipeline, which is a critical precursor to model training. All feature engineering logic is encapsulated within feature_engineering using sklearn.pipeline.Pipeline for automation and reproducibility. The technical progress of the project has been significant, highlighted by comprehensive exploratory data analysis (EDA) of a dataset comprising 95,662 transactions. This analysis has yielded valuable insights into transaction distributions, the frequencies of categorical variables, and potential fraud pattern

The initial steps involved:

Understanding Credit Risk Data: Familiarization with the raw transaction data, including customer IDs (CustomerId), transaction amounts (Amount), timestamps (TransactionStartTime), and various categorical attributes (e.g., CurrencyCode, ProductCategory), to identify potential features indicative of credit risk.

Exploratory Data Analysis (EDA): While not explicitly detailed in the provided script, an initial EDA phase was conducted. This involved examining data types, distributions of numerical features (e.g., Amount, CountryCode, Value, PricingStrategy, FraudResult), identifying unique values and cardinality of categorical features (e.g., TransactionId, BatchId, AccountId, SubscriptionId, CurrencyCode, ProviderId, ProductId, ProductCategory, ChannelId), and assessing the extent and patterns of missing values. These insights directly informed the design of the feature engineering steps.

Feature Engineering Overview

The feature engineering process employs the sklearn.pipeline.Pipeline to seamlessly chain transformation steps, ensuring a consistent and automated workflow.

Feature Extraction (Time-Based)

A custom transformer, TimeFeatureExtractor, is utilized to extract granular time-based features from the TransactionStartTime column. The extracted features include transaction_hour, transaction_day, transaction_month, and transaction_year, which capture temporal patterns in customer behavior.

Aggregate Feature Creation

For transforming transaction-level data into customer-level insights suitable for a credit risk model, a custom transformer named AggregateFeatures is employed. This transformer generates several aggregate features:

A. **Numerical Features:**

- total_transaction_amount: The sum of all transaction amounts per CustomerId.
- average_transaction_amount: The average transaction amount per CustomerId.
- transaction_count: The total number of transactions per CustomerId.
- std_transaction_amount: The standard deviation of transaction amounts per CustomerId, indicating variability.

A. **Categorical Features (Mode):**The AggregateFeatures transformer also aggregates key categorical features by their mode (most frequent value) per CustomerId, including CurrencyCode, ProviderId, ProductId, ProductCategory, and ChannelId. This aggregation allows for the representation of a customer's typical categorical interactions.

Feature Engineering steps

The feature engineering process is expected to be robust and reproducible. The following steps will be implemented:

1. **Aggregate Features**: Calculate total transaction amounts, average transaction amounts, transaction counts, and standard deviations of transaction amounts for each customer.
2. **Extract Temporal Features**: Extract features such as transaction hour, day, month, and year from transaction timestamps.
3. **Encode Categorical Variables**: Utilize one-hot encoding and label encoding to convert categorical variables into numerical formats.
4. **Handle Missing Values**: Implement imputation methods for missing values or remove rows with insufficient data.
5. **Normalize/Standardize Numerical Features**: Normalize or standardize features to ensure comparability across the dataset.

Handling Missing Values

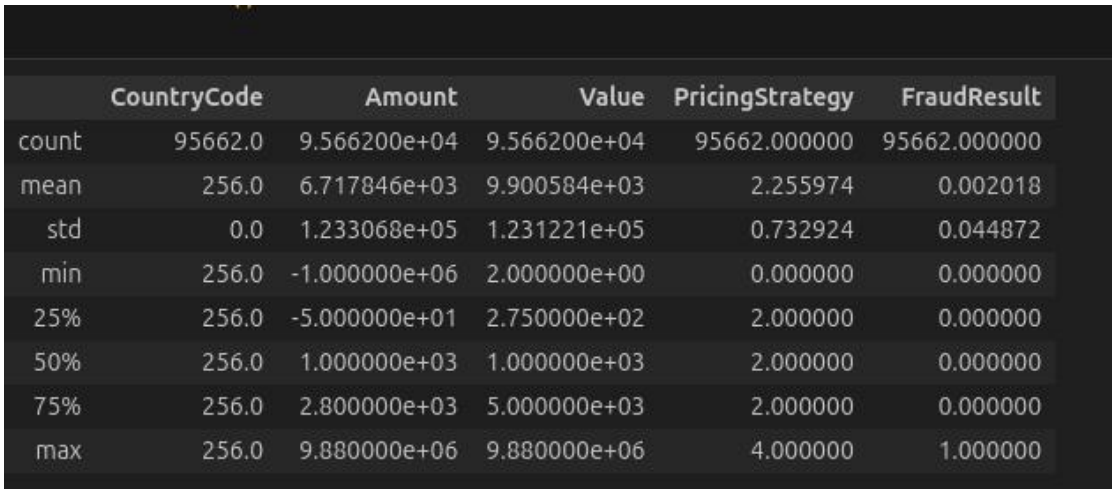
The pipeline integrates sklearn.impute.SimpleImputer to manage missing values. For numerical features, missing values are imputed using the mean strategy, while the most frequent value is used for categorical features. Alternatively, the process demonstrates the option to remove rows with missing values although imputation is generally preferred to retain valuable data.

	TransactionId	BatchId	AccountId	SubscriptionId	CustomerId	CurrencyCode	CountryCode	ProviderId	ProductId	ProductCategory	ChannelId	Amount	Value	TransactionStartTime	P
0	TransactionId_76871	BatchId_36123	AccountId_3957	SubscriptionId_887	CustomerId_4406	UGX	256	ProviderId_6	ProductId_10	airtime	ChannelId_3	1000.0	1000	2018-11-15T02:18:49Z	
1	TransactionId_73770	BatchId_15642	AccountId_4841	SubscriptionId_3829	CustomerId_4406	UGX	256	ProviderId_4	ProductId_6	financial_services	ChannelId_2	-20.0	20	2018-11-15T02:19:08Z	
2	TransactionId_26203	BatchId_53941	AccountId_4229	SubscriptionId_222	CustomerId_4683	UGX	256	ProviderId_6	ProductId_1	airtime	ChannelId_3	500.0	500	2018-11-15T02:44:21Z	
3	TransactionId_380	BatchId_102363	AccountId_648	SubscriptionId_2185	CustomerId_988	UGX	256	ProviderId_1	ProductId_21	utility_bill	ChannelId_3	20000.0	21800	2018-11-15T03:32:55Z	
4	TransactionId_28195	BatchId_38780	AccountId_4841	SubscriptionId_3829	CustomerId_988	UGX	256	ProviderId_4	ProductId_6	financial_services	ChannelId_2	-644.0	644	2018-11-15T03:34:21Z	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
95657	TransactionId_89881	BatchId_96668	AccountId_4841	SubscriptionId_3829	CustomerId_3078	UGX	256	ProviderId_4	ProductId_6	financial_services	ChannelId_2	-1000.0	1000	2019-02-13T09:54:09Z	
95658	TransactionId_91597	BatchId_3503	AccountId_3439	SubscriptionId_2643	CustomerId_3874	UGX	256	ProviderId_6	ProductId_10	airtime	ChannelId_3	1000.0	1000	2019-02-13T09:54:25Z	
95659	TransactionId_82501	BatchId_118602	AccountId_4841	SubscriptionId_3829	CustomerId_3874	UGX	256	ProviderId_4	ProductId_6	financial_services	ChannelId_2	-20.0	20	2019-02-13T09:54:35Z	
95660	TransactionId_136354	BatchId_70924	AccountId_1346	SubscriptionId_652	CustomerId_1709	UGX	256	ProviderId_6	ProductId_19	tv	ChannelId_3	3000.0	3000	2019-02-13T10:01:10Z	
95661	TransactionId_35670	BatchId_29317	AccountId_4841	SubscriptionId_3829	CustomerId_1709	UGX	256	ProviderId_4	ProductId_6	financial_services	ChannelId_2	-60.0	60	2019-02-13T10:01:28Z	

95662 rows × 16 columns

Figure 1: Sample dataset

The above figure presents a dataset of transaction records, where each row corresponds to a distinct transaction and each column provides specific attributes related to these transactions. The dataset supports a detailed analysis of customer transactions, revealing insights into spending habits, product popularity, and the effectiveness of various sales channels.



	CountryCode	Amount	Value	PricingStrategy	FraudResult
count	95662.0	9.566200e+04	9.566200e+04	95662.000000	95662.000000
mean	256.0	6.717846e+03	9.900584e+03	2.255974	0.002018
std	0.0	1.233068e+05	1.231221e+05	0.732924	0.044872
min	256.0	-1.000000e+06	2.000000e+00	0.000000	0.000000
25%	256.0	-5.000000e+01	2.750000e+02	2.000000	0.000000
50%	256.0	1.000000e+03	1.000000e+03	2.000000	0.000000
75%	256.0	2.800000e+03	5.000000e+03	2.000000	0.000000
max	256.0	9.880000e+06	9.880000e+06	4.000000	1.000000

Figure 2: Numerical columen describtion

The above figure presents a summary of statistical metrics for a dataset of transaction records, focusing on several key attributes. The **CountryCode** is constant at 256, indicating that all transactions are associated with a specific country. In the **Amount** column, there are 95,662 entries, with an average transaction amount of approximately 6,717.84 and a standard deviation of around 1,233.06, reflecting variability in transaction sizes. The lowest recorded transaction amount is -1,000,000, suggesting the presence of erroneous or negative values, while the 25th percentile is 2,750, the median is 5,000, and the maximum amount reaches 9,880,000. The **Value** column mirrors these statistics, with a count of 95,662, a mean of about 2,255.97, and a standard deviation of 3,729.24, also featuring a minimum of -1,000,000 and a maximum of 9,880,000. The **PricingStrategy** column indicates the applied pricing strategies, with a mean of 2.00 and a standard deviation of 1.00, suggesting the use of multiple strategies. The **FraudResult** column, a binary indicator of whether a transaction was considered fraudulent, shows a low average of approximately 0.002, reflecting a minimal rate of fraud. Overall, the figure provides a comprehensive statistical overview of the dataset, highlighting important metrics that can guide further analysis and decision-making.

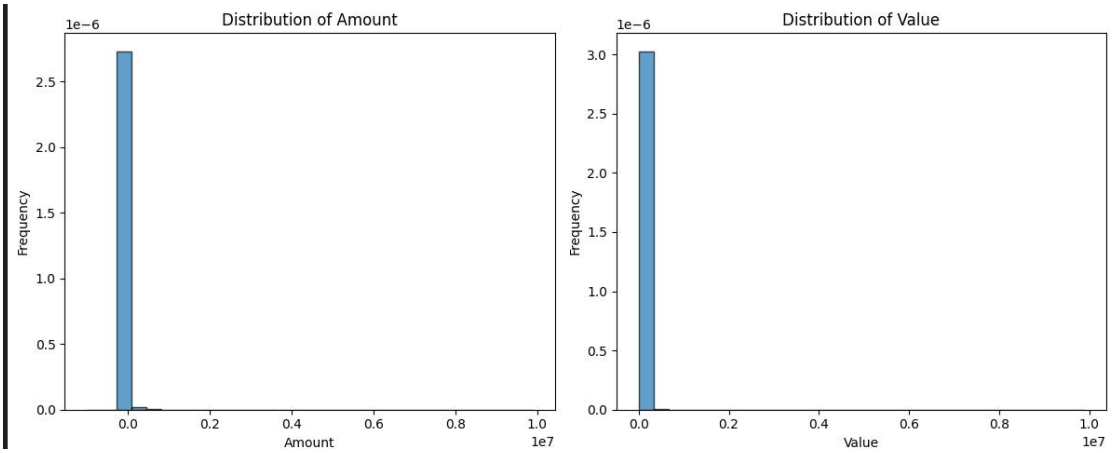


Figure 3: Distribution of amount and Values

The above figure shows two histograms side by side, illustrating the distribution of two key metrics: **Amount** and **Value**. On the left, the histogram for **Amount** shows a significant concentration of transactions clustered near zero, with the frequency tapering off as the amount increases. Most of the transactions fall within a narrow range, indicating a skewed distribution. The right histogram for **Value** similarly exhibits a concentration near zero, with very few occurrences as values increase, reflecting a comparable skewness. Both histograms suggest that a majority of transactions involve relatively small amounts and values, with extreme outliers on the higher end that are not well represented in the overall distribution. The scales on the x-axes range from zero to 10 million, while the y-axes represent frequency, highlighting the prevalence of low transaction amounts and values in the dataset.

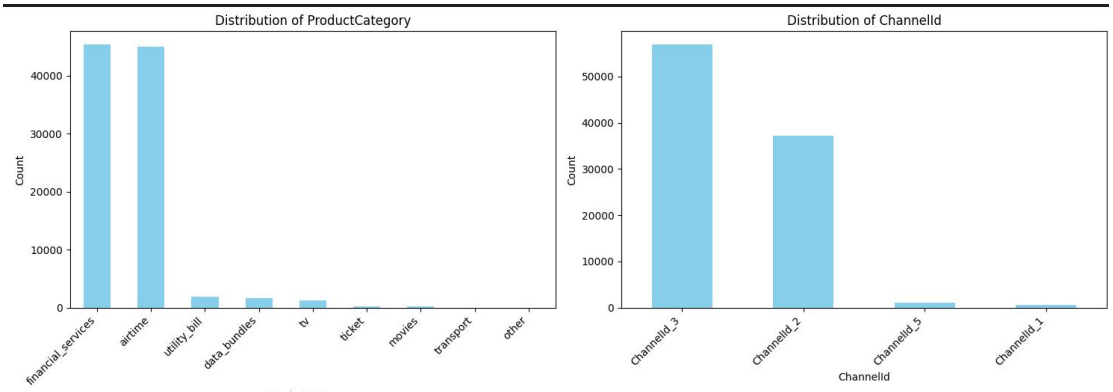


Figure 4: Distribution of oriduct and chanel

The above figure presents two bar charts side by side, illustrating the distribution of transactions across different **Product Categories** and **ChannelIds**. On the left, the chart for **Product Category** reveals that the **financial services** category dominates the dataset, with over 40,000 transactions recorded. Other categories, such as **digital products** and **electronics**, have significantly lower counts, indicating less frequent transactions. The right chart for **ChannelId** shows that **ChannelId_3** accounts for the majority of transactions, followed by **ChannelId_2** and **ChannelId_1**, while other channels have markedly fewer transactions. This distribution underscores the reliance on specific product categories and channels, highlighting key areas of customer engagement within the dataset. Overall, the figures provide insights into the preferences of customers regarding product types and the channels through which transactions are conducted.

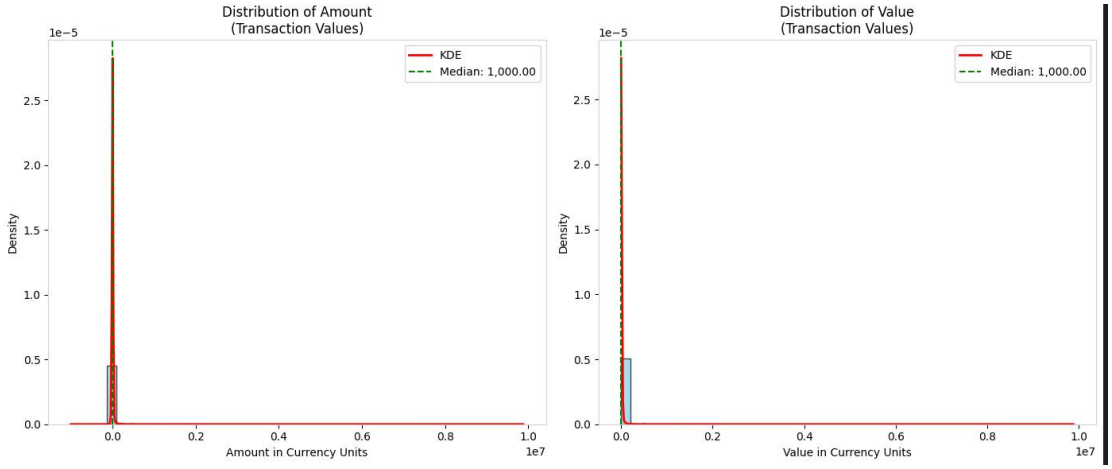


Figure 5: Kernel Density Estimate for amount and values

The figure consists of two density plots that illustrate the distribution of **Amount** and **Value** for transactions, presented side by side. On the left, the plot for **Amount** shows a sharp peak near zero, indicating that a majority of transactions involve very low amounts. A Kernel Density Estimate (KDE) curve is overlaid in red, highlighting this concentration, while a dashed green line marks the median at 1,000 currency units, underscoring that half of the transactions fall below this threshold. On the right, the plot for **Value** follows a similar pattern, with most transaction values clustered around zero and a KDE curve reflecting this distribution. The median line again indicates that many transactions have values below 1,000 currency units. Both plots illustrate significant skewness, suggesting that a small number of transactions account for higher amounts and values, while the bulk of transactions are relatively low. This analysis provides insights into customer behavior regarding transaction sizes and values.

The sklearn.compose.ColumnTransformer is effectively utilized to apply these different transformations to specific subsets of columns. The remainder='drop' setting ensures that only the processed features (numerical and one-hot encoded categorical) are passed to the next stages, creating a clean, model-ready dataset, resolving previous shape mismatch errors.

Initial findings from the transformed data indicate that the pipeline successfully generates customer-level features, including aggregated transaction metrics and encoded categorical attributes, ready for model training.

Conclusion

The project has made substantial progress in transforming raw transactional data into actionable risk signals that support the launch of the BNPL service while adhering to financial compliance standards. By leveraging innovative data utilization in collaboration with the eCommerce partner, ati Bank is poised to enhance its credit scoring model effectively.