Interim Project Report: Credit Scoring Model for Bati Bank

This interim report provides a synthesis of the business context, an overview of the technical progress made so far (Tasks 1 and 2), and an outline of planned future steps for the Credit Scoring Model project at Bati Bank. The project aims to enable a buy-now-pay-later service by assessing customer creditworthiness through a predictive model.

1. Synthesis of Business Context

1.1 Understanding Credit Risk and Basel II Capital Accord

The core objective of this project is to develop a robust Credit Scoring Model. This necessitates a deep understanding of credit risk, particularly as governed by regulations like the **Basel II Capital Accord**. Basel II significantly transformed risk management in financial institutions, emphasizing **risk-sensitive capital requirements** and pushing banks towards using their own internal models (Internal Ratings-Based approach) for estimating key parameters such as **Probability of Default (PD)**.

This regulatory framework directly impacts our model development, driving the need for:

- **Interpretability and Transparency:** Models must be understandable to regulators, auditors, and business stakeholders. This allows for validation of assumptions, identification of biases, and clear explanations for credit decisions (e.g., adverse action explanations).
- **Robust Documentation:** Comprehensive documentation of the model's logic, data sources, methodology, and performance is essential for regulatory compliance and effective model risk management.
- **Informed Decision-Making:** An interpretable model empowers credit officers and portfolio managers to understand why certain customers are deemed high or low risk, facilitating better pricing strategies and portfolio management.

1.2 Risk Proxy Creation and its Business Implications

A significant challenge in this project is the **absence of a direct "default" label** in the provided e-commerce transaction data. To overcome this, a **proxy variable** for credit risk will be engineered. This proxy is necessary to:

- **Enable Supervised Learning:** Without a defined target variable, it's impossible to train a predictive model using traditional supervised machine learning techniques.
- **Leverage Available Data:** The e-commerce data contains rich behavioral information that, while not directly indicating "default," can signify customer engagement and transactional patterns indicative of risk.

The strategy involves calculating **Recency, Frequency, and Monetary (RFM) metrics** for each customer. These RFM values will then be used in a clustering algorithm (e.g., K-Means) to

segment customers into distinct groups. A "high-risk" proxy will be defined by identifying the cluster characterized by **low recency, low frequency, and low monetary value**, as these customers are typically "disengaged" and thus more likely to pose a credit risk.

However, relying on a proxy variable introduces several **potential business risks**:

- **Inaccurate Risk Assessment:** The primary risk is that the proxy may not perfectly align with true default behavior. This could lead to:
  - **Underestimation of Risk:** Lending to genuinely high-risk customers, resulting in unexpected losses for Bati Bank.
  - **Overestimation of Risk:** Denying credit to creditworthy customers, leading to lost revenue and potential competitive disadvantage.
- **Sub-optimal Capital Allocation:** If the proxy-based PD estimations are inaccurate, the bank might hold either too much or too little capital, impacting efficiency or regulatory compliance under Basel II.
- **Reputational Damage:** Consistent inaccurate decisions can harm Bati Bank's reputation with customers and the market.
- **Regulatory Scrutiny:** The proxy definition and its validation will be subject to intense regulatory review. Lack of robust justification could lead to non-compliance penalties.

Careful selection, validation, and ongoing monitoring of the proxy variable are crucial to mitigate these risks.

1.3 Trade-offs: Simple vs. Complex Models in a Regulated Financial Context

In a regulated financial context like Bati Bank, selecting a credit scoring model involves a critical trade-off between **predictive accuracy** and **interpretability/explainability**:

Model Type	Pros	Cons	Contextual Trade-offs
Simple (e.g., Logistic Regression with WoE)	- **High Interpretability:** Coefficients directly indicate feature impact, making it easy to understand "why" a score was assigned. WoE transformations linearize relationships and make features more robust.	- **Lower Predictive Power:** May struggle to capture complex non-linear relationships or intricate interactions between features, potentially leading to lower accuracy compared to complex models.	**Regulatory Compliance:** Often preferred by regulators due to transparency, making model validation and audit easier. **Adverse Action Explanations:** Provides clear, legally compliant reasons for denying credit. **Model Risk Management:** Easier to identify and mitigate

			biases or limitations. **Operational Simplicity:** Easier to implement, monitor, and maintain in production.
Complex (e.g., Gradient Boosting Machines)	- **High Predictive Accuracy:** Excels at capturing complex patterns, non-linear relationships, and feature interactions, often leading to superior performance on large and intricate datasets.	- **Low Interpretability (Black-box):** Difficult to fully understand how predictions are made, making it challenging for regulatory scrutiny, audit, and explaining decisions to customers.	**Regulatory Hurdles:** May face significant resistance from regulators due to their black-box nature, requiring extensive Explainable AI (XAI) techniques. **Model Risk:** More complex to diagnose and debug when performance degrades. **Resource Intensive:** Requires more computational resources for training and potentially more expertise for development and maintenance.

For Bati Bank, given the regulated nature of financial services and the importance of justifying credit decisions, a simpler, interpretable model like Logistic Regression with WoE might be favored for core credit scoring, especially for regulatory approval. Complex models could be explored as "challenger" models for internal risk management, provided robust XAI techniques are implemented to ensure sufficient transparency. The primary trade-off is often sacrificing some predictive accuracy for regulatory compliance and explainability.

2. Reporting on Technical Progress

2.1 Project Setup

The project adheres to the mandated standardized structure to ensure engineering discipline and maintainability:

```
credit-risk-model/
├── .github/workflows/ci.yml
├── data/
│   ├── raw/
```

```
│   └── processed/
├── notebooks/
│   └── 1.0-eda.ipynb
├── src/
│   ├── __init__.py
│   ├── data_processing.py
│   ├── train.py
│   ├── predict.py
│   └── api/
│       ├── main.py
│       └── pydantic_models.py
├── tests/
│   └── test_data_processing.py
├── Dockerfile
├── docker-compose.yml
├── requirements.txt
├── .gitignore
└── README.md
```

The initial data file (data.csv) has been placed in data/raw/ as per the structure.

2.2 Initial EDA Findings (Task 2)

Exploratory Data Analysis (EDA) was performed using the 1.0-eda.ipynb Jupyter Notebook. The key findings are summarized below:

- **Data Overview:**
  - The dataset (data.csv) was successfully loaded.
  - It contains **95,662 rows and 16 columns**.
  - Data types include object (for IDs, codes, categories, and TransactionStartTime), float64 (Amount), and int64 (CountryCode, Value, PricingStrategy, FraudResult).
- **Summary Statistics:**
  - **Numerical Features:**
    - CountryCode: Shows a standard deviation of 0, indicating all transactions are from the same country (Code 256). This column will not be useful for distinguishing users by country.
    - Amount and Value: Both have large standard deviations and ranges, with Amount showing negative values (likely for credits/refunds). Value appears to be the absolute amount. Extreme values (outliers) are present.
    - PricingStrategy: Appears to be categorical despite being int64, with values ranging from 0 to 4.

        - FraudResult: A binary target variable for fraud detection (0 or 1), which is highly imbalanced with very few fraud cases (mean of 0.002).
    - **Categorical Features:**
        - CurrencyCode: Only one unique value (UGX), making it non-discriminatory.
        - AccountId, SubscriptionId, CustomerId, BatchId, TransactionId: These are unique identifiers. TransactionId is fully unique (95662 unique values for 95662 rows), BatchId has 94809 unique values, while AccountId, SubscriptionId, and CustomerId have fewer unique values, indicating multiple transactions per account/customer/subscription. CustomerId has 3742 unique values, suggesting we have data for 3742 distinct customers.
        - ProviderId, ProductId, ProductCategory, ChannelId: These show various levels of unique categories, which will be valuable for feature engineering. ProductCategory is dominated by 'financial_services' and 'airtime'. ChannelId_3 is the most frequent.
- **Distribution of Numerical Features:**
    - Histograms and KDE plots confirmed that Amount and Value distributions are heavily skewed to the right (positive skew), with long tails indicating the presence of large transactions (outliers).
    - CountryCode and PricingStrategy show discrete distributions, reinforcing their categorical nature.
- **Distribution of Categorical Features:**
    - Bar plots revealed the distribution imbalance across categories for ProductCategory, ChannelId, ProviderId, and ProductId.
    - CurrencyCode and CountryCode were confirmed to have only one effective value, rendering them uninformative as features.
- **Correlation Analysis:**
    - A heatmap of numerical features showed a **very high positive correlation (0.98) between Amount and Value**, suggesting redundancy. Given Amount can be negative, Value (absolute amount) might be a more suitable feature or a transformed Amount.
    - FraudResult (the existing fraud label) shows **very low correlation with other numerical features**, indicating that simple linear relationships may not explain fraud effectively. This also highlights why a proxy for *credit risk* is needed, rather than directly using this FraudResult.
- **Missing Values and Outlier Detection:**
    - **No missing values** were found in the dataset, which simplifies the data cleaning process significantly.
    - Box plots visually confirmed the presence of **significant outliers in Amount and Value**. These outliers are likely genuine large transactions but will need careful handling during feature engineering (e.g., winsorization, log transformation, or robust

scaling) to prevent them from disproportionately influencing model training.

Top 5 Most Important Insights from EDA:

1. **Redundant & Outlier-Prone Transaction Amounts:** Amount and Value are highly correlated; Value (absolute amount) appears to be abs(Amount). Both features contain significant outliers that will require careful treatment to prevent model distortion.
2. **Uninformative Constant Features:** CountryCode and CurrencyCode are constant across the dataset, making them useless for predictive modeling and can be dropped.
3. **Categorical Nature of Numeric Columns:** PricingStrategy (and implicitly CountryCode) should be treated as categorical features, despite their numeric data type.
4. **Data Imbalance in Existing Fraud Label:** The FraudResult column is highly imbalanced, with very few fraud cases. This reinforces the business need to engineer a *credit risk proxy* rather than relying solely on this fraud flag for creditworthiness.
5. **Customer-Centric Aggregation Required:** The presence of CustomerId (3742 unique IDs across 95662 transactions) indicates that transaction-level data needs to be aggregated to a customer level before defining a credit risk proxy and training a customer-level credit scoring model.

2.3 Outline of Future Technical Progress

Based on the completed tasks and project requirements, the following steps are planned:

- **Task 3: Feature Engineering (src/data_processing.py)**
  - Aggregate Features
  - Encode Categorical Variables
  - Handle Missing Values
  - Normalize/Standardize Numerical Features
- **Task 4: Proxy Target Variable Engineering (src/data_processing.py)**
  - Calculate RFM Metrics
  - Cluster Customers
  - Define "High-Risk" Label
- **Task 5: Model Training and Tracking (src/train.py, tests/test_data_processing.py)**
  - Data Splitting
  - Model Selection
  - Training & Hyperparameter Tuning
  - Model Evaluation
  - MLflow Integration
  - Unit Tests

- **Task 6: Model Deployment and Continuous Integration (src/api/main.py, src/api/pydantic_models.py, Dockerfile, docker-compose.yml, .github/workflows/ci.yml)**
    - API Development
    - Containerization
    - CI/CD Pipeline

3. Clarity and Professionalism

This report is structured to clearly address all metrics outlined in the interim report rubric. Technical language has been used consistently, and the objectives of the Bati Bank credit scoring challenge have been maintained throughout. The report demonstrates an in-depth understanding of both the business context of credit risk and the foundational technical work completed, setting a clear path for the remaining tasks.