Task 2 – Predictive Model Plan

Luis G B A Faria, Sydney, NSW, +61 0403-278-880, lfariabr@gmail.com
linkedin/lfaria / github/lfaria / dev.to/lfaria / luisfaria.dev

1. Model Logic

Chosen Model: Logistic Regression for Delinquency Prediction

Model Description: Logistic Regression is a statistical model that predicts the probability of a customer becoming delinquent by using a logistic function to transform linear combinations of input features into a probability score between 0 and 1.

This model choice is directly informed by Task 1's EDA results, which showed a severe class imbalance (84% non-delinquent vs 16% delinquent) and weak individual correlations across all numerical features ($|r| < 0.3$). These patterns confirm that delinquency is a multi-factor problem where transparency and stability matter more than raw predictive lift.

Key Features Selected (Top 5 based on EDA analysis):

1. Income - Primary indicator of financial stability and repayment capacity
2. Credit_Score - Historical credit behavior and risk assessment
3. Credit_Utilization - Current debt burden relative to credit limit
4. Debt_to_Income_Ratio - Measure of debt capacity and financial stress
5. Missed_Payments - Recent payment behavior and delinquency history

```python
def predict_deliquency_probability(customer_features):
    # Step 1: Preprocess and select features
    processed_features = preprocess_features(customer_features)

    # Step 2: Calculate linear combination (log-odds)
    log_odds = model_intercept
    for feature, coefficient in zip(processed_features, model_coefficients):
        log_odds += coefficient * feature

    # Step 3: Apply logistic function to get probability
    delinquency_probability = 1 / (1 + math.exp(-log_odds))

    # Step 4: Apply classification threshold
    prediction = "Delinquent" if delinquency_probability > 0.5 else "Non-delinquent"

    return {
        "probability": delinquency_probability,
        "prediction": prediction,
        "risk_level": "High" if delinquency_probability > 0.7 else "Medium" if delinquency_probability > 0.3 else "Low"
    }
```

Figure 1: Pseudo code of prediction function

Model Workflow:

1. Data Preprocessing: Handle missing values using median imputation, encode categorical variables, scale numerical features
2. Feature Selection: Choose the most relevant predictors based on correlation analysis and business knowledge
3. Model Training: Fit logistic regression coefficients using maximum likelihood estimation on training data
4. Probability Prediction: Calculate delinquency probability using the logistic function: $P(\text{delinquent}) = 1 / (1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n)})$
5. Threshold Application: Classify customers as delinquent if probability > 0.5 (adjustable threshold based on business needs)

2. Justification

I selected Logistic Regression as the primary model for Geldium's delinquency prediction system because it perfectly balances the critical requirements of financial services: interpretability, reliability, and regulatory compliance.

As a more sophisticated modelling option, a Random Forest or Gradient Boosting Classifier could be explored as a challenger model to improve ROC-AUC and capture non-linear interactions. However, while these models may deliver higher predictive lift, Logistic Regression remains the recommended production model due to its superior transparency, explainability, and regulatory alignment in credit risk settings

Key Advantages for Financial Risk Prediction:

- Transparency & Interpretability: Unlike complex models like neural networks, logistic regression provides clear, explainable coefficients that show exactly how each feature (income, credit score, etc.) influences delinquency risk. This is crucial for regulatory compliance and stakeholder trust.
- Probability-Based Predictions: Outputs calibrated probability scores (0-1) that directly translate to risk levels, enabling flexible decision thresholds based on business risk tolerance.
- Robustness with Financial Data: Works well with structured numerical data typical in credit scoring, handles missing values gracefully, and is less prone to overfitting than more complex models.

Business Suitability:

- Regulatory Compliance: Financial institutions must explain lending decisions. Logistic regression's clear feature importance aligns with fair lending laws and anti-discrimination requirements.
- Operational Feasibility: Easy to implement, monitor, and update. Collections teams can understand and act on probability scores.
- Baseline Performance: Provides a strong starting point before considering more complex models if needed.

Comparison with Alternatives:

- vs. Decision Trees: While trees are also interpretable, they can be unstable and prone to overfitting with financial data.
- vs. Neural Networks: Too complex and "black-box" for regulated financial environments where explainability is mandatory.
- vs. Random Forest: Better performance but sacrifices the clear interpretability needed for compliance.

Given Geldium's focus on responsible AI and the EDA findings of weak individual correlations (suggesting multi-factorial risk), logistic regression offers the best balance of performance, interpretability, and compliance for production deployment.

3. Evaluation Strategy

Primary Evaluation Metrics: Given the severe class imbalance (84% non-delinquent, 16% delinquent) identified in EDA, I will prioritize metrics that account for minority class performance:

- Recall (Sensitivity): Critical metric measuring what percentage of actual delinquent customers are correctly identified. High priority since missing delinquent customers represents significant financial loss.
- Precision: Measures what percentage of predicted delinquent customers are actually delinquent. Important to avoid unnecessary collection actions on good customers.
- F1 Score: Balanced metric combining precision and recall, useful when both false positives and negatives are costly.
- ROC-AUC: Comprehensive measure of model's ability to distinguish between classes across all threshold levels.
- Accuracy: General performance overview, but interpreted cautiously due to imbalance.

Metric Interpretation Guidelines:

- Recall > 0.75: Good at identifying high-risk customers
- Precision > 0.60: Reasonable accuracy in delinquency predictions

- F1 Score > 0.65: Balanced performance acceptable for initial deployment
- ROC-AUC > 0.75: Good discriminatory ability

Bias Detection and Fairness Assessment:

- Demographic Parity Analysis: Check if model predictions are equally distributed across protected groups (age, gender, ethnicity)
- Equalized Odds: Ensure similar true positive and false positive rates across demographic groups
- Disparate Impact Testing: Monitor for disproportionate risk predictions against specific groups
- Feature Importance Analysis: Review coefficient magnitudes to identify potentially biased features

Bias Mitigation Strategies:

- Remove or transform proxy variables that might correlate with protected characteristics
- Apply class weighting to ensure fair representation of minority groups
- Regular fairness audits using statistical tests (e.g., chi-square tests for independence)
- Threshold calibration to balance precision/recall trade-offs

Ethical Considerations:

- Transparency: All predictions include probability scores and feature contributions for customer explainability Right to Explanation: Develop customer-facing explanations of risk factors Regular Monitoring: Implement drift detection to identify when model performance degrades Human Oversight: High-risk predictions flagged for manual review by collections specialists
- Data Privacy: Ensure compliance with data protection regulations (GDPR, CCPA)
- Fair Lending Compliance: Regular audits to prevent discriminatory lending practices

Evaluation Workflow:

1. Train/validation/test split with stratification to maintain class balance
2. Cross-validation (5-fold stratified) to ensure stable performance estimates
3. Threshold tuning based on business cost-benefit analysis
4. Fairness testing across demographic subgroups
5. Stress testing with edge cases and outlier scenarios
6. Documentation of all evaluation results for regulatory compliance

GenAI tools (e.g., ChatGPT) were used to scaffold the modelling pipeline, propose evaluation metrics suitable for imbalanced financial datasets, and refine explanations around model trade-offs.

All AI-generated insights were reviewed, validated, and adjusted based on domain knowledge and the EDA findings.