**Predictive Model Plan**

# ****1.Model Logic (Generated with GenAI)****

Using ChatGPT, I generated a predictive model using logistic regression to estimate the likelihood of a customer becoming delinquent. To predict a binary outcome: 1 if the customer is likely to become delinquent, and 0 otherwise.

1. Load dataset

- Read data from CSV or database

- Example: data = load("credit\_data.csv")

2. Select features:

- Features = ['Credit\_Utilization', 'Missed\_Payments', 'Income',

'Debt\_to\_Income\_Ratio', 'Account\_Tenure']

- X = data[Features]

3. Define target variable:

- y = data['Delinquent\_Account'] # 1 = Delinquent, 0 = Not delinquent

4. Split data into training and testing sets:

- Use 70% for training, 30% for testing

- X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y)

5. Fit logistic regression model:

- model = LogisticRegression()

- model.fit(X\_train, y\_train)

6. Predict and evaluate:

- y\_pred = model.predict(X\_test)

- Evaluate using metrics:

- Accuracy

- Precision

-Recall

- F1 Score

- Confusion Matrix

# 2. Justification for Model Choice

I selected Logistic Regression as the preferred model for predicting credit delinquency due to the following reasons:  
  
-**Binary Classification**  
Logistic regression is ideal for predicting binary outcomes, such as whether an account is delinquent (1) or not (0).  
  
- **Interpretability**  
The model’s coefficients help explain the influence of each feature (e.g., income, missed payments) on the probability of delinquency, which is important in financial decision-making and regulatory compliance

- **Speed and Efficiency**  
Logistic regression is computationally efficient and performs well even with relatively large datasets and simple features.

- **Compliance-Ready and Auditable**  
Logistic regression aligns with **regulatory requirements** in the financial industry, including RBI guidelines and Basel norms. Its **transparency, explainability, and traceable logic** make it ideal for enterprise use cases like those at Tata, where responsible AI and model governance are critical.

# 3. Evaluation Strategy

These points reflect both **technical robustness** and **business alignment**.

 **Stratified Train-Test Split**  
Ensure class balance (delinquent vs. non-delinquent) is preserved during data splitting to avoid biased evaluation.

 **Use of Relevant Metrics**  
Evaluate model using **Precision, Recall, F1-Score**, and **AUC-ROC** — not just accuracy — to properly assess credit risk classification performance.

 **Confusion Matrix Analysis**  
Analyze false positives and false negatives carefully, as misclassifying a high-risk customer can lead to significant financial loss.

 **Cross-Validation (k-Fold)**  
Use **k-fold cross-validation** to assess model stability and generalizability across different data subsets.

 **Threshold Optimization**  
Adjust decision threshold on predicted probabilities to align with business goals — for example, prioritize high **recall** to catch more risky accounts.