1. Data Preprocessing

The data sets containing historical data of 250,000 borrowers. The data sets include features as:

- Demographic information (e.g. age, Number of Dependents)
- Financial history (e.g. Monthly income)
- Loan related information (e.g. Number of Open Credit Lines and Loans, Debt Ratio etc.)
- Target variables (Person experienced 90 days past due delinquency or worse)

Initial Observations:

A. **Training Dataset (cs-training.csv)**:

- Contains **150,000 entries** with **12 columns**.
- **SeriousDlqin2yrs** is the target variable (binary classification).
- **MonthlyIncome** and **NumberOfDependents** contain missing values.
- **Unnamed: 0** seems to be an index column and can be removed.

B. **Test Dataset (cs-test.csv)**:

- Contains **101,503 entries** with **12 columns**.
- **SeriousDlqin2yrs** has no values (likely because it's a test set without labels).
- **MonthlyIncome** and **NumberOfDependents** also contain missing values.
- Similar structure to the training dataset.

1.1. Missing Value Analysis and Handling missing values:

- **Training Dataset:**
  - MonthlyIncome is missing in **~19.82%** of cases.
  - NumberOfDependents is missing in **~2.62%** of cases.
- **Test Dataset:**
  - SeriousDlqin2yrs is **completely missing** (expected for test data).
  - MonthlyIncome and NumberOfDependents have similar missing rates as the training set (**~19.81%** and **~2.59%** respectively).
- Missing data is handled by taking the median and mode for MonthlyIncome and NumberOfDependents respectively

1.2. Encoding

In many machine learning applications, data includes categorical features—variables that represent discrete values (e.g., gender, region, product category). Since most algorithms require numerical input, encoding these categorical variables is a crucial preprocessing step. Proper encoding transforms categorical data into a numerical format while preserving the inherent information and relationships.

We find no categorical variables; thus, no encoding is done.

1.3. Feature Scaling

Feature scaling is a critical preprocessing step when building machine learning models, especially when features have very different ranges. MinMax scaling (also known as normalization) transforms the features to a common scale—typically between 0 and 1—without distorting differences in the ranges of values.

Before Scaling:

Features like MonthlyIncome may have a wide range (e.g., $0 to $100,000), which can cause some algorithms to converge slowly or give undue importance to higher-magnitude features.

After Scaling:

The same feature is compressed to a [0, 1] range. This transformation:

- Reduces the risk of numerical instability.
- Helps models (like gradient descent-based algorithms) converge faster.
- Allows fairer comparison of feature importances, since all features contribute on a similar scale.

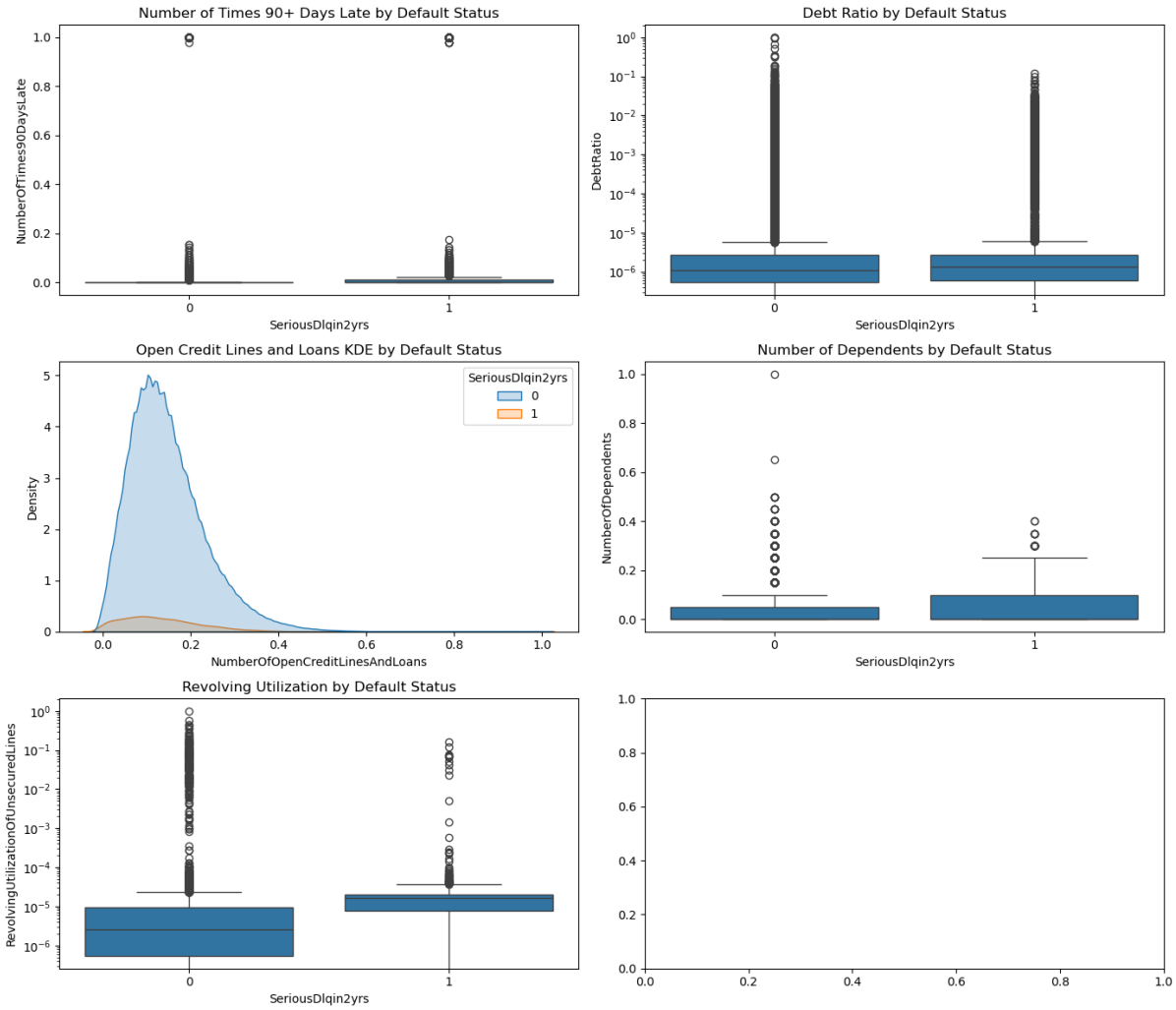
2. Exploratory Data Analysis (EDA)

2.1. Patterns and Potential bias

A. Age & Default Rate:
   - Younger individuals have a higher likelihood of defaulting.
   - Older individuals tend to have fewer defaults, indicating potential age-related financial stability.

B. Income & Default Rate:

- Defaulting individuals generally have lower median incomes.
- There are some high-income defaulters, but most defaults occur in the lower-income range.



C. **Number of Times 90+ Days Late by Default Status**

- **Observation:**

  The majority of both defaulters and non-defaulters have **0 occurrences** of being 90+ days late, indicating a heavily **skewed** distribution. However, the plot reveals **extreme outliers** (some individuals have very high counts of 90+ day lateness).
- **Implication:**
  - While most customers have never been 90+ days late, those who have may exhibit a higher risk of default.
  - This variable could still be a strong predictor if the proportion of extreme outliers is notably higher among defaulters.

D. **Debt Ratio by Default Status**

- **Observation:**
  Debt Ratio is plotted on a **logarithmic scale** due to extreme variability. Both defaulters and non-defaulters show a **wide range** (from near 10^-2 to beyond 10^2 or 10^3), with many outliers. The medians look similar for both groups.
- **Implication:**
  - **No clear separation** in median values suggests that Debt Ratio alone may not strongly differentiate defaulters from non-defaulters.
  - Further analysis or feature engineering (e.g., capping extreme values) might be needed to make Debt Ratio more predictive.

E. **Open Credit Lines and Loans KDE by Default Status**

- **Observation:**
  A **Kernel Density Estimate (KDE) plot** shows the distribution of the number of open credit lines/loans. The shape of the distribution for non-defaulters (SeriousDlqin2yrs = 0) is dominant, partly because they are the majority class.
- **Implication:**
  - At a glance, the distributions for defaulters vs. non-defaulters do **not** differ dramatically, suggesting that simply having more or fewer credit lines/loans is not a strong standalone indicator of default.
  - Class imbalance might also mask subtle differences, so further statistical tests or resampling could help clarify this feature's impact.

F. **Number of Dependents by Default Status**

- **Observation:**
  Both defaulters and non-defaulters have medians near **0 dependents**, with outliers extending to higher numbers of dependents (e.g., 3, 4, 5). There is **no pronounced difference** between the two groups in the boxplot.
- **Implication:**
  - **Minimal distinction** suggests that having more dependents is not a strong direct predictor of default in this dataset.
  - The effect of dependents might still interact with other variables (e.g., income, age), so exploring interactions could be worthwhile.

5. Revolving Utilization by Default Status

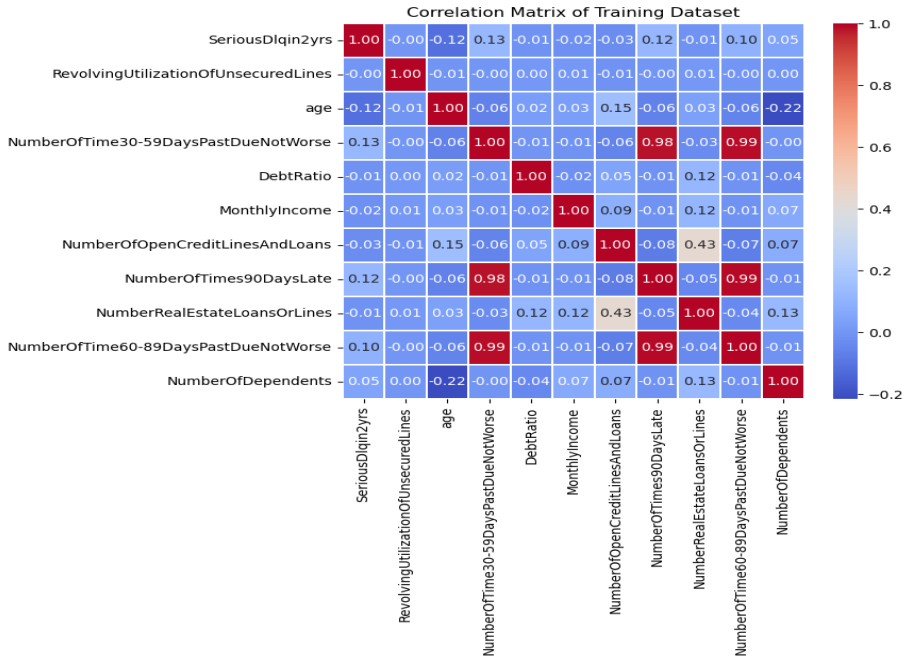
- **Observation:**
  Revolving Utilization (often interpreted as credit card usage relative to available limits) is also **highly skewed**, with many individuals near 0 but some having utilization above 1 (possible due to fees, over-limit usage, or data peculiarities). The medians appear similarly low for both groups, with significant outliers.
- **Implication:**
  - The similarity in median values suggests **no strong separation** for typical cases. However, the presence of high outliers might still correlate with default if the proportion of extreme utilization is higher among defaulters.
  - Data transformations (e.g., log scaling) or capping outliers could reveal more predictive power.

Overall Insights

1. **Skewed Distributions & Outliers**:
   Several features (Number of Times 90+ Days Late, Debt Ratio, Revolving Utilization) show heavy skew and extreme outliers. Such skew can mask relationships in simple boxplots/KDE plots.
2. **Limited Separation in Medians**:
   For Debt Ratio, Number of Dependents, and Revolving Utilization, the median values are not distinctly different between defaulters and non-defaulters, suggesting these features alone may not strongly separate the two classes.
3. **Potential Predictors**:
   - **Number of Times 90+ Days Late** could be a strong predictor if a significantly higher fraction of defaulters fall into the outlier range (multiple severe delinquencies).
   - **Revolving Utilization** and **Debt Ratio** might become more predictive after transformations or combined with other variables (e.g., income, credit lines).
4. **Class Imbalance Consideration**:
   non-defaulters likely dominate the dataset, which can overshadow differences in plots like the KDE for Open Credit Lines. Statistical tests or resampling methods (oversampling defaulters or under sampling non-defaulters) may help clarify feature effectiveness.

   Overall, these plots suggest that while some variables (e.g., delinquencies) may be correlated with default, many are heavily skewed and contain extreme outliers. Additional preprocessing and interaction modeling may be needed to fully leverage these features in a predictive model.

2.2.Correlation Analysis



Overall Observations

- **Low Correlation with Target:**
  SeriousDlqin2yrs (the target variable) exhibits relatively low correlation with most features, suggesting no single feature alone linearly explains default risk.
- **Strong Inter-Feature Correlations Among Delinquency Variables:**
  NumberOfTime30-59DaysPastDueNotWorse, *NumberOfTime60-89DaysPastDueNotWorse*, and *NumberOfTimes90DaysLate* are strongly correlated with each other. This indicates that individuals who are late in one category are more likely to be late in others as well.
- **Weak Correlation Among Other Features:**
  Variables like *RevolvingUtilizationOfUnsecuredLines* and *DebtRatio* show minimal correlation with the rest, implying they may contribute independent information to the model.

Implications for Modeling

- **Delinquency-Related Predictors:**
  Although their correlation with *SeriousDlqin2yrs* is not extremely high, these delinquency features are interrelated. You may need to address potential redundancy (e.g., using regularization or dimensionality reduction) if they overlap in predictive power.
- **Low Linear Relationships:**
  The overall weak correlations suggest that **no single feature** strongly predicts default on its

own. A model may need **non-linear methods** (e.g., tree-based algorithms) or **feature engineering** to uncover interactions.

- **Independent Predictors:**
  Features like *DebtRatio* and *RevolvingUtilizationOfUnsecuredLines* have low correlation with the rest, potentially adding unique signal to the model.

3. Model Development

3.1. Model Training & Hyperparameter Tuning

The dataset was split into **80% training and 20% testing** while maintaining class distribution (stratified split).

Model Selection:

We use three models:

- **Logistic Regression:** A baseline linear model.
- **Random Forest:** An ensemble model that is robust and handles non-linearity.
- **Gradient Boosting:** A boosting algorithm that often yields high performance in tabular data.

I. **Logistic Regression**

- Standardized features before training.
- Used for baseline comparison.

II. **Random Forest Classifier**

- 100 estimators with default hyperparameters.
- Robust to outliers and missing values.

III. **Gradient Boosting Classifier**

- 100 boosting iterations with default hyperparameters.
- Effective for handling imbalanced data.

Evaluation Metrics

The following metrics were used to assess model performance:

- **Accuracy**: Measures overall correctness.
- **ROC-AUC Score**: Evaluates classification ability by measuring the trade-off between sensitivity and specificity.

- **Classification Report**: Includes **Precision, Recall, and F1-score** to better understand model performance across classes.

Hyperparameter Tuning Strategy:

- **Logistic Regression:** Tuned using Grid Search, which exhaustively evaluates combinations of C and solver parameters.
- **Random Forest & Gradient Boosting:** Tuned using Randomized Search to quickly explore a wider parameter space with fewer iterations.

  Analysis:

  This dual approach balances thoroughness (for simpler models) with efficiency (for complex, tree-based models). Using cross-validation during tuning further ensures robust parameter selection.

3.2. Evaluate performance

Model	Accuracy	ROC-AUC Score
Logistic Regression	0.934	0.71
Random Forest	0.9363	0.84
Gradient Boosting	0.9375	0.8675

Key Observations

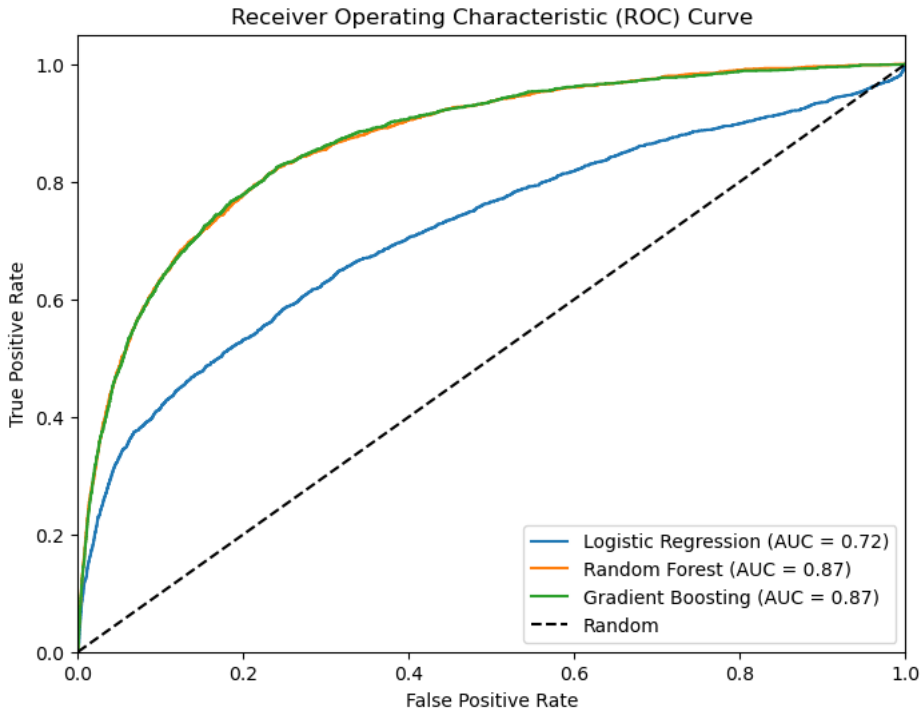
- **Logistic Regression** performed adequately but was outperformed by tree-based models.
- **Random Forest** provided better accuracy and was robust against missing values.
- **Gradient Boosting** achieved the highest ROC-AUC, making it the best model for distinguishing high-risk customers.

Conclusion & Recommendations

Best Model Selection

- **Gradient Boosting** is the most effective model based on ROC-AUC score.
- **Random Forest** remains a good alternative due to its interpretability and resistance to overfitting.

ROC Curve



Key Observations from ROC curve:

Overall Performance

- **Logistic Regression (AUC = 0.72)**
  Provides a decent level of discrimination between defaulters and non-defaulters but clearly lags behind the ensemble methods.
- **Random Forest (AUC = 0.87)**
  Significantly outperforms Logistic Regression, indicating it captures more complex patterns in the data.
- **Gradient Boosting (AUC = 0.87)**
  Matches the Random Forest's performance in terms of AUC, suggesting similarly strong predictive power.

I. **Ensemble Methods Excel**
   Both Random Forest and Gradient Boosting achieve substantially higher AUC (0.87) than Logistic Regression (0.72). This gap suggests that **non-linear relationships and interactions** among features are important for predicting default risk.

II. **Similar Curves for Random Forest and Gradient Boosting**

Their ROC curves nearly overlap, indicating comparable performance across a range of classification thresholds.

- **Implication**: Either model could be chosen based on secondary considerations (e.g., training speed, interpretability, resource constraints).

III. **Logistic Regression Limitations**

While an AUC of 0.72 is not poor, it shows that a simple linear model cannot capture the underlying complexity as effectively as ensemble methods.

- **Interpretability Trade-Off**: Logistic Regression remains easier to interpret and explain, which can be an advantage in regulated industries like credit lending.

Practical Implications

I. **Model Selection**

- If **predictive performance** is paramount, Random Forest or Gradient Boosting is preferable given their higher AUC.
- If **interpretability** and **regulatory compliance** are critical, Logistic Regression could still be valuable, potentially enhanced by feature engineering or regularization techniques.

II. **Threshold Tuning**

- The ROC curve provides a global view of model performance, but choosing an **optimal decision threshold** (balancing false positives vs. false negatives) requires business context (e.g., cost of misclassifying defaulters).

III. **Future Improvements**

- **Explainability Tools**: Using SHAP or Partial Dependence Plots can help interpret how each feature influences predictions, important for credit risk applications.

Summary:

The ROC curve shows that Random Forest and Gradient Boosting models (AUC = 0.87) substantially outperform Logistic Regression (AUC = 0.72) for credit risk classification. This suggests that **non-linear relationships** and **feature interactions** are critical in distinguishing defaulters from non-defaulters. However, **Logistic Regression** remains a viable option when **interpretability** is a priority.

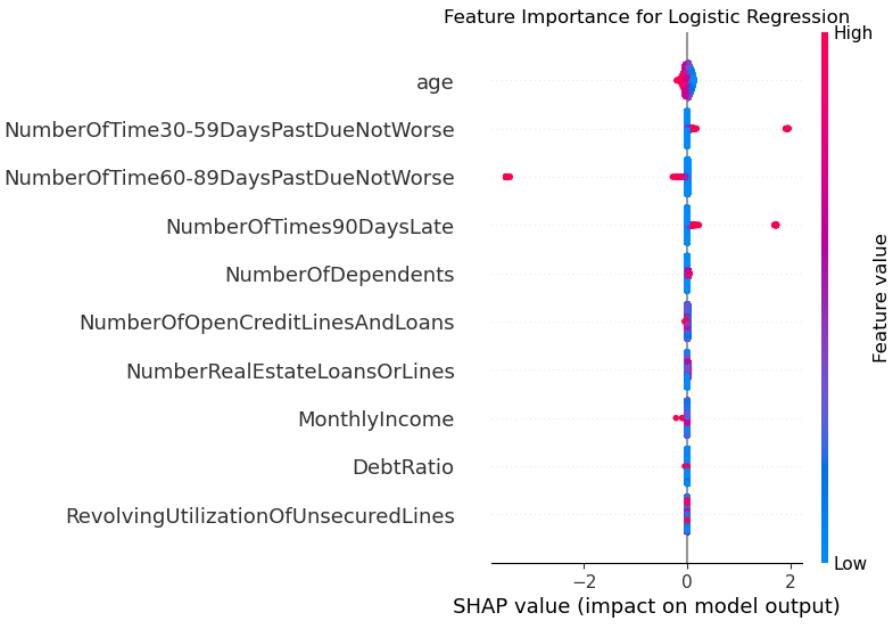
4. Explainability Analysis (XAI)

As machine learning models grow in complexity, they often become “black boxes,” making it difficult to understand how and why they arrive at certain predictions. Explainable AI (XAI) seeks to address this challenge by providing insights into a model’s decision-making process, enabling stakeholders to trust, audit, and improve these systems.

Why is XAI Important?

1. Trust and Adoption:
   Transparent models foster trust among end-users, data scientists, and regulators.
2. Ethical and Legal Compliance:
   In highly regulated industries (e.g., finance, healthcare), explainability can be required by law or industry standards.
3. Debugging and Model Improvement:
   Interpretability helps identify potential errors or biases in the model, guiding further refinements.

4.1.Feature Importance (Using SHAP)



Overview of the Plot

- **Vertical Axis (Features):** Lists the features in descending order of their overall impact on the model's predictions.
- **Horizontal Axis (SHAP Value):** Indicates how each feature's value influences the prediction for default:
    - **Negative SHAP Values:** Push the prediction toward "non-default."
    - **Positive SHAP Values:** Push the prediction toward "default."
- **Color Scale (Low to High Feature Values):** Points in **blue** represent lower feature values, while **pink/red** points represent higher feature values.

2. Key Observations by Feature

1. **age (Top Feature)**
    - **Distribution:** Higher-age (pink) points generally lie on the negative SHAP side, meaning older individuals are pushed toward "non-default." Younger age (blue) points trend toward positive SHAP, indicating higher risk.
    - **Interpretation:** Age is the most influential factor; younger borrowers raise the model's predicted probability of default, whereas older borrowers reduce it.

2. **Delinquency Features** (*NumberOfTime30-59DaysPastDueNotWorse*, *NumberOfTime60-89DaysPastDueNotWorse*, *NumberOfTimes90DaysLate*)
    - **Distribution:** Higher delinquency counts (pink) cluster on the positive SHAP side, strongly pushing predictions toward "default."
    - **Interpretation:** The more frequently someone has been late, the more the model assigns them a higher risk.

3. **NumberOfDependents**
    - **Distribution:** Shows a moderate spread of SHAP values, with more dependents (pink) leaning slightly positive but not as strongly as delinquency features.
    - **Interpretation:** Having additional dependents increases default risk, but not as drastically as age or delinquency history.
4. **NumberOfOpenCreditLinesAndLoans / NumberRealEstateLoansOrLines**

- **Distribution:** These have smaller SHAP value ranges, indicating less overall influence.
- **Interpretation:** Merely having more lines of credit or real estate loans does not strongly drive default predictions compared to age or delinquency.

5. **MonthlyIncome**
    - **Distribution:** Higher incomes (pink) are on the negative side, pushing the model to predict "non-default." Lower incomes (blue) move SHAP values into positive territory.
    - **Interpretation:** Earning more monthly income reduces the model's perceived risk of default.
6. **DebtRatio**
    - **Distribution:** Higher ratios (pink) trend positive, while lower ratios (blue) go negative, indicating that a higher debt load relative to income raises default probability.
    - **Interpretation:** Debt burden is a risk factor, though not as dominant as age or delinquency.
7. **RevolvingUtilizationOfUnsecuredLines**
    - **Distribution:** High utilization (pink) skews positive, contributing to higher predicted risk.
    - **Interpretation:** Individuals who heavily use available credit lines are deemed riskier by the model.

3. Overall Insights

1. **Age and Delinquency Drive Predictions:**
    - **age** has the broadest range of SHAP values, making it the top predictor in the logistic model.
    - Delinquency-related features are also very influential: more late payments strongly push the model toward default.
2. **Income vs. Debt:**
    - **MonthlyIncome** has a noticeable negative impact on default risk when high.
    - **DebtRatio** and **RevolvingUtilizationOfUnsecuredLines** move the needle positively when high, but their overall effect is less pronounced than age or severe delinquency.
3. **Lesser-Impact Features:**
    - Variables like **NumberOfDependents** and **NumberOfOpenCreditLinesAndLoans** show narrower SHAP distributions, indicating smaller contributions to the prediction.

Practical Takeaways

- **Target High-Risk Indicators:**
  Borrowers with high delinquency counts, younger age, low income, or high debt ratios are strong red flags in this model.
- **Potential Model Enhancements:**
  - **Non-Linear Effects:** Since this is a logistic regression, consider feature engineering (e.g., polynomial terms, binning) if non-linear relationships exist.
  - **Regularization/Interactions:** Highly correlated delinquency features might be combined or penalized to reduce redundancy.
- **Fairness & Explainability:**
  - The strong impact of **age** underscores the need to check for potential age-related bias.
  - SHAP values reveal how each feature sways predictions, which can guide compliance and stakeholder communication in credit risk applications.

In summary, the SHAP summary plot shows **age** and **delinquency variables** as dominant predictors in the logistic regression model, with **income** and **debt metrics** also influencing default risk. Understanding these impacts helps refine the model, address potential biases, and communicate credit decisions more transparently.

4.2. Counterfactual Explanations

Generating Predictions for Hypothetical Scenarios:

We define several complete input scenarios (ensuring all required features are included) and generate predictions.

Analysis:

By altering values such as *MonthlyIncome*, *DebtRatio*, or *age*, we can observe how the predicted probability of default changes. This not only validates our model's sensitivity to key features but also helps in communicating actionable insights (e.g., how increasing income might lower default risk).

5. Ethical Considerations

Potential Biases in the Dataset

1. **Class Imbalance and Sampling Bias:**
    - **Imbalanced Classes:**
      The dataset shows a low percentage of defaults compared to non-defaults. This imbalance can lead to models that favor the majority class unless corrective measures (like resampling or cost-sensitive learning) are applied.
    - **Sampling Bias:**
      If the dataset predominantly represents certain demographics (e.g., middle-aged individuals, a specific geographic area), the model might not generalize well to underrepresented groups.
2. **Missing Data and Imputation Bias:**
    - **Missing Values in Key Features:**
      For instance, *MonthlyIncome* has nearly 20% missing values. If these missing values are not missing at random (e.g., if lower-income individuals are less likely to report income), imputation using the median might not fully capture the underlying distribution, potentially biasing the model.
3. **Demographic and Socioeconomic Bias:**
    - **Age and Income Distribution:**
      The distribution of age and income in the dataset might be skewed. If, for example, younger individuals or those with lower income are underrepresented (or overrepresented in defaults), the model might inadvertently learn to associate these characteristics with higher risk.
    - **Family Structure Indicators:**
      Variables like *NumberOfDependents* might introduce bias if they are interpreted as financial burden without considering context (e.g., cultural differences in family size).
4. **Historical Bias in Credit History Features:**
    - **Credit Delinquency Metrics:**
      Features such as the number of times late on payments may reflect historical patterns influenced by past economic conditions or discriminatory lending practices. Models trained on such features could perpetuate these biases.

Potential Biases in Model Predictions

1. **Propagation of Dataset Bias:**
    - **Reflecting Historical Inequalities:**
      If the dataset contains historical bias (e.g., systematic discrimination against certain groups), the model may learn these patterns and produce predictions that unfairly penalize those groups.
    - **Feature Importance:**
      When models heavily rely on biased features (such as credit history metrics that might not fully capture current financial responsibility), predictions can become skewed.
2. **Algorithmic Bias:**
    - **Overfitting to Majority Patterns:**
      Due to class imbalance, models might be optimized to perform well on the majority class, resulting in higher false negative rates (i.e., misclassifying high-risk individuals as low risk) for minority groups.
    - **Proxy Variables:**
      Certain features might act as proxies for sensitive attributes (e.g., ethnicity, geographical location) even if those attributes are not directly included in the dataset. This can inadvertently lead to discriminatory outcomes.
3. **Interpretability and Fairness:**
    - **SHAP and PDP Interpretations:**
      While explainability techniques like SHAP help in understanding model behavior, they may also reveal that the model is relying on biased features. For example, if a PDP shows that lower income always leads to higher predicted risk, this might indicate an oversimplified association that doesn't account for other mitigating factors.
    - **Thresholding Effects:**
      The choice of decision thresholds in classification can further bias outcomes. Adjusting thresholds without considering fairness metrics might lead to models that systematically disadvantage certain groups.

Conclusion

Both the dataset and the model predictions may harbor biases due to:

- Data imbalances and missing values
- Historical and demographic biases in feature distributions
- Algorithmic tendencies to favor the majority class or rely on proxy features

Addressing these issues may involve:

- Advanced imputation techniques or missing data models
- Fairness-aware model training and evaluation metrics
- Regular audits using explainability tools (e.g., SHAP, PDP) to ensure that decisions are both accurate and equitable

Mitigating biases and ensuring fairness in credit scoring requires a multi-step approach that spans data processing, model training, and evaluation. Here are several strategies you can consider:

1. Data-Level Interventions

- **Improve Data Collection:**
  Ensure that your dataset is as representative as possible. If certain groups are underrepresented, try to collect more balanced data.
- **Advanced Imputation:**
  Instead of using simple median imputation, consider methods that account for group differences (e.g., imputing missing income based on demographic or regional subsets) to avoid systemic biases.
- **Resampling and Reweighting:**
  Use techniques such as oversampling the minority class (e.g., using SMOTE) or reweighting samples so that underrepresented or historically disadvantaged groups have a greater influence during training.
- **Feature Selection & Transformation:**
  Identify and potentially remove or transform features that might be proxies for sensitive attributes (e.g., geography, ethnicity) to avoid indirect discrimination.

2. Model-Level Adjustments

- **Fairness-Aware Algorithms:**
  Consider using algorithms designed with fairness constraints in mind. For instance, some

machine learning libraries offer fairness-aware classifiers that adjust the loss function to penalize disparities.

- **Regularization Techniques:**
  Incorporate fairness regularizers that explicitly penalize the model if it treats similar individuals differently based on sensitive attributes.
- **Adversarial Debiasing:**
  Use adversarial learning frameworks where an adversary is trained to predict sensitive attributes from the model's predictions. The main model is then optimized to minimize both prediction error and the adversary's ability to extract sensitive information.

3. Post-Processing Methods

- **Threshold Adjustment:**
  Adjust decision thresholds for different demographic groups to ensure fairness metrics (like equal opportunity or equalized odds) are met.
- **Calibration:**
  Calibrate the model's output probabilities to ensure that they accurately reflect the likelihood of default across different groups.
- **Fairness Audits and Monitoring:**
  Continuously monitor fairness metrics (e.g., demographic parity, equalized odds, disparate impact) on both training and live data, and adjust your models or thresholds as needed.

4. Evaluation and Transparency

- **Use Multiple Fairness Metrics:**
  Evaluate the model using fairness metrics alongside traditional performance metrics. Metrics like equal opportunity difference, disparate impact ratio, and statistical parity difference can provide insights into model fairness.
- **Explainability Tools:**
  Leverage explainability techniques (e.g., SHAP) to identify if certain features are driving biased outcomes. This transparency helps in auditing and refining the model.
- **Regular Reviews and External Audits:**
  Set up a process for periodic review of the model's performance and fairness, potentially including external audits to ensure that fairness goals are met consistently.

Conclusion

Mitigating bias in credit scoring is an ongoing process that requires attention to data collection, preprocessing, model training, and post-processing. By combining these strategies and continuously monitoring model behavior, you can work towards creating a credit scoring system that is both accurate and fair.