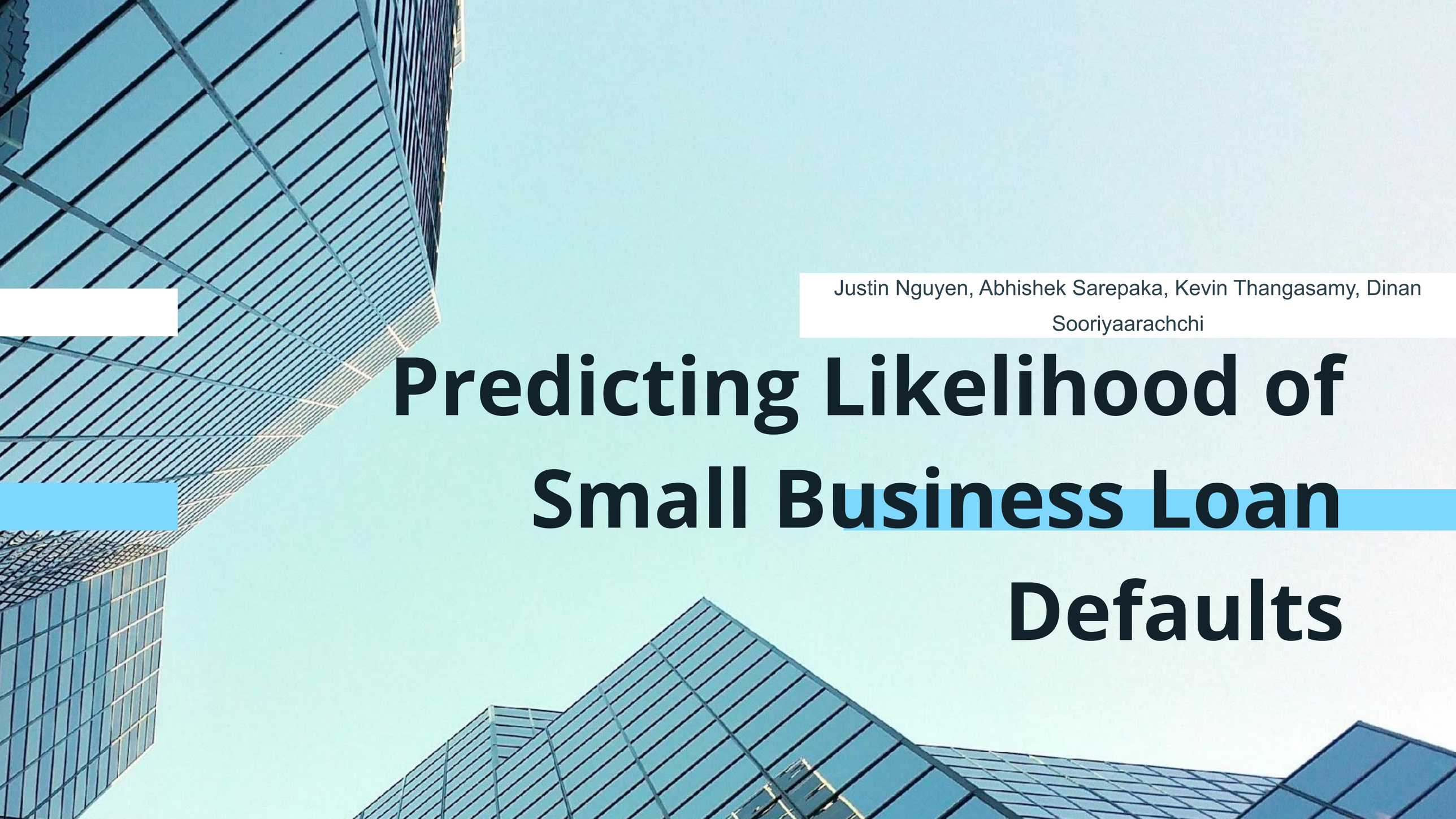


Justin Nguyen, Abhishek Sarepaka, Kevin Thangasamy, Dinan Sooriyaarachchi

Predicting Likelihood of Small Business Loan Defaults

Problem Statement

- Banks are still faced with a difficult choice as to whether they should grant such a loan because of the probability of default.
- One way to inform their decision-making is through predicting this probability of default analyzing relevant historical data such as the SBA National Data with the following Data Dictionary.
- The rate of default on these loans has been a source of controversy for decades. Conservative economists believe that credit markets perform efficiently without government participation.
- Supporters of SBA-guaranteed loans argue that the social benefits of job creation by those small businesses receiving government-guaranteed loans far outweigh the costs incurred from defaulted loans.

Similar Studies/References

- A study by Michele Modina, et al. predicted small and medium-sized enterprises' (SMEs) default likelihood. It found that using the credit line and the amount of credit limit violations of current accounts and long term loans increase a bank's ability to predict the probability of a borrower's inability to repay the loan by at least a year before it occurs [1].
- Another study by Hamid Cheraghali and Peter Molnar finds that addressing sample imbalance, using effective feature selection, and using robust validation techniques are key for accurate SME default predictions [2]
- A third study by Yiannis Dendramis, et al. uses a multilayer artificial neural network (ANN) to predict the probability of default. The research used a large data set of small business loans and found that the most important factors on loan default are the payments and remaining balance of the loan, the outstanding debt amount, and the duration of the loan [3].
- Stefania Albanesi et al., developed a deep learning model. Credit risk assessment to young borrowers as well as tracking variations in systemic default risk provide valuable information about the ways to reduce consumer default [4].
- Alessandro Bitetto et al. sought to understand how we can use machine learning to predict the credit risk of small businesses. They found that imprecise measurement of credit risk can pose a risk to lenders and the financial markets. Using fair, unbiased machine learning techniques can lead to borrowers and small businesses in receiving the appropriate financial support they require [5].

Data Source

LoanNr_Ch	Name	City	State	Zip	Bank	BankState	NAICS	ApprovalDa	ApprovalFY	Term	NoEmp	NewExist	CreateJob	RetainedJo	FranchiseC	UrbanRural	RevLineCr	LowDoc	ChgOffDate	Disb
1E+09	ABC HOBB\	EVANSVILL	IN	47711	FIFTH THIR	OH	451120	28-Feb-97	1997	84	4	2	0	0	1	0	N	Y		28-F
1E+09	LANDMARK	NEW PARIS	IN	46526	1ST SOURC	IN	722410	28-Feb-97	1997	60	2	2	0	0	1	0	N	Y		31-M
1E+09	WHITLOCK	BLOOMING	IN	47401	GRANT COU	IN	621210	28-Feb-97	1997	180	7	1	0	0	1	0	N	N		31-D
1E+09	BIG BUCKS	BROKEN AF	OK	74012	1ST NATL B	OK	0	28-Feb-97	1997	60	2	1	0	0	1	0	N	Y		30-.
1E+09	ANASTASIA	ORLANDO	FL	32801	FLORIDA BU	FL	0	28-Feb-97	1997	240	14	1	7	7	1	0	N	N		14-M
1E+09	B&T SCREV	PLAINVILLE	CT	6062	TD BANK, N	DE	332721	28-Feb-97	1997	120	19	1	0	0	1	0	N	N		30-.
1E+09	MIDDLE ATI	UNION	NJ	7083	WELLS FAR	SD	0	2-Jun-80	1980	45	45	2	0	0	0	0	N	N	24-Jun-91	22-
1E+09	WEAVER PF	SUMMERFII	FL	34491	REGIONS B	AL	811118	28-Feb-97	1997	84	1	2	0	0	1	0	N	Y		30-.

- National SBA
- TYPE: Census
- 899,164 observations, 27 variables
- Source: United States Small Business Administration
- This data set is from the U.S. Small Business Administration (SBA) and provides historical data from 1987 through 2014, containing 27 variables and 899,164 observations. Each observation represents a loan that was guaranteed to some degree by the SBA. Included is a variable [MIS_Status] which indicates if the loan was paid in full or defaulted/charged off.
- The data resides in a comma-separated values (csv) file. A header line contains the name of the variables

Goal

- This case study aims to develop visualizations and a logistic regression model to predict the likelihood of a small business loan default based on historical loan data. The goal is to help financial institutions make informed lending decisions, reducing the risk of defaults and optimizing loan approval strategies.
- We will also use a Random Forest model to compare with the logistic regression model.

Comparing Models

Logistic Regression

Pros

- Can easily extend to multiple classes(multinomial regression)
- very fast at classifying
- Good accuracy for many simple data sets and it performs well when the dataset is linearly separable.
- is easy to implement, interpret, and very efficient to train.

Cons

- Problems of overfitting if # observations < # features
- the assumption of linearity between the dependent variable and the independent variables.
- Non-linear problems can’t be solved with logistic regression
- requires average or no multicollinearity between independent variables.

Random Forest

Pros

- High accuracy since it uses multiple decision trees leading to less variation
- It can handle noise since it combines the predictions of several trees
- Non-parametric in nature, so it can identify intrinsic patterns

Cons

- Can be highly complex and computationally expensive
- Large memory usage
- Longer to predict
- Problems of overfitting

Logistic Regression Model

Overview

01. Data Cleaning

02. Exploratory Analysis

03. Logistic Regression Model

Data Cleaning

Dropped Irrelevant Columns: Removed columns that were not necessary for the analysis, such as LoanNr_ChkDgt, Name, Bank, City, Zip, CreateJob, and RetainedJob.

Dropped Rows with Missing Values: Removed rows with missing values in critical columns like State, BankState, NewExist, RevLineCr, LowDoc, DisbursementDate, and MIS_Status.

Applied Binary Encoding to Categorical Columns: Converted RevLineCr, LowDoc, and UrbanRural into binary (1 or 0) values for machine learning compatibility.
Output: One-hot encoded categorical columns.

Dropped Duplicate Rows: Identified and removed duplicate rows to maintain data uniqueness.

Removed Foreign Values: Removed values in columns that had values that didn't match with what was supposed to be there (e.g. removing values like 'a' from column where valid values are only 0 and 1)

Applied One-Hot Encoding to NAICS Column: Transformed NAICS into one-hot encoded columns to represent industry categories numerically.

Data Cleaning

```
# Dropping irrelevant columns
cols_to_drop = ['LoanNr_ChkDgt', 'Name', 'Bank', 'City', 'Zip', 'CreateJob', 'RetainedJob']

df.drop(cols_to_drop, axis=1, inplace=True)

print('Dropped columns: ', cols_to_drop)
```

```
# Drop duplicate rows
df = df.drop_duplicates()
print('Duplicate rows have been dropped')
```

```
# Drop all rows with missing fields

missing_data = ['State', 'BankState', 'NewExist', 'RevLineCr', 'LowDoc', 'DisbursementDate', 'MIS_Status']

df.dropna(subset=missing_data, inplace=True)

print('Dataframe Shape after dropping: ', df.shape)
```

Data Cleaning

```
# Dropping foreign values and binary encoding
# LowDoc
df.dropna(subset=['LowDoc'], inplace=True)
df = df[df['LowDoc'].isin(['N', 'Y', '1', '0'])].copy()
df['LowDoc'] = df['LowDoc'].map({'Y': '1','N': '0','1': '1','0': '0'})
df['LowDoc'] = df['LowDoc'].astype(int)
```

```
# RevLineCr
df.dropna(subset=['RevLineCr'], inplace=True)
df = df[df['RevLineCr'].isin(['N', 'Y', '1', '0'])].copy()
df['RevLineCr'] = df['RevLineCr'].map({'Y': '1','N': '0','1': '1','0': '0'})
df['RevLineCr'] = df['RevLineCr'].astype(int)
```

```
# UrbanRural
df = df[df['UrbanRural'].isin([1, 2])]

# Convert 'UrbanRural' to integer
df['UrbanRural'] = df['UrbanRural'].astype(int)

# Map 1 -> 0 and 2 -> 1
df['UrbanRural'] = df['UrbanRural'].map({1: 0, 2: 1})
```

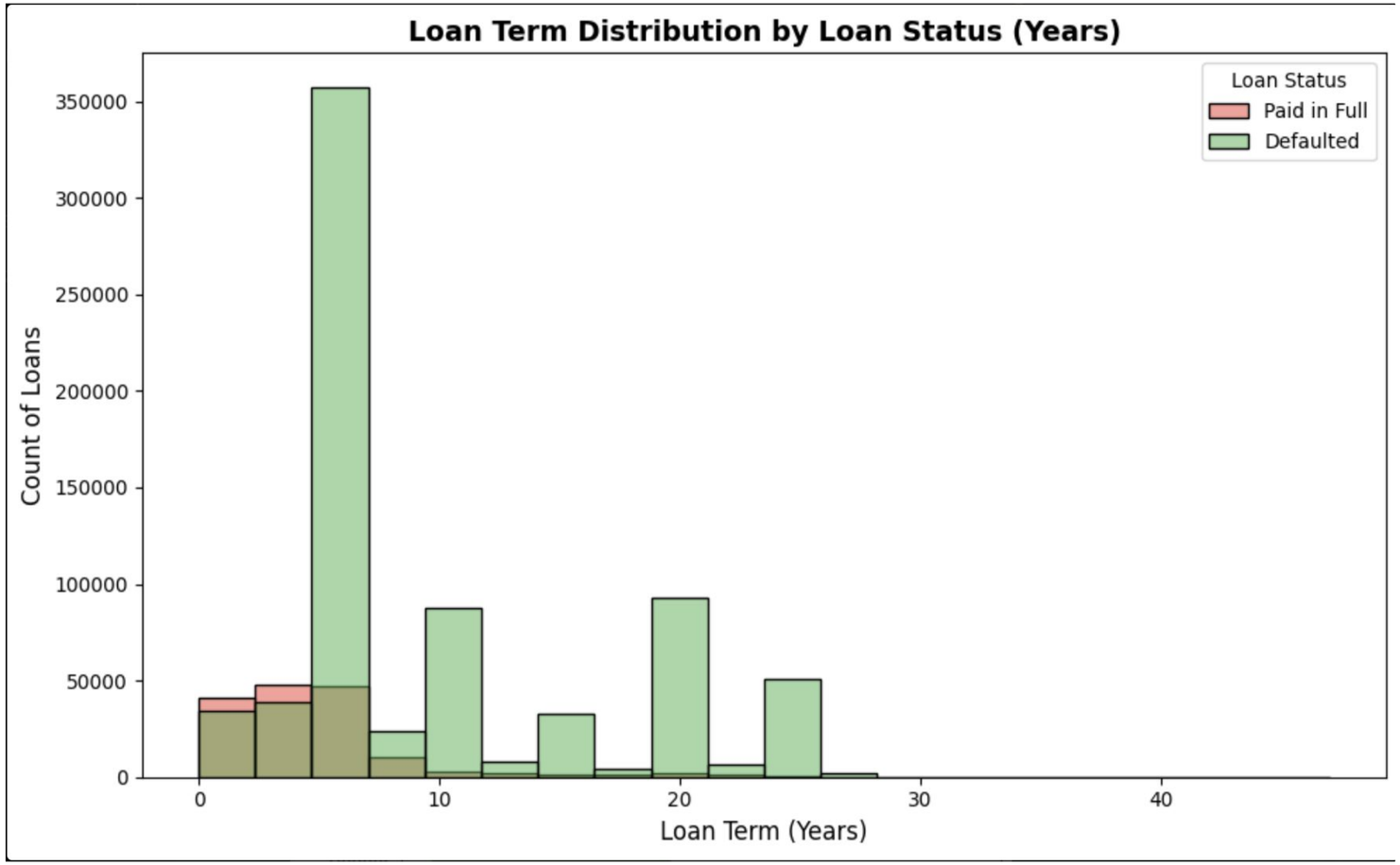
In [9]:
```
# Define NAICS code to category mapping
naics_mapping = {
    "0" : None,
    "11": "Agriculture_Forestry_Fishing_Hunting",
    "21": "Mining_Quarrying_Oil_GasExtraction",
    "22": "Utilities",
    "23": "Construction",
    "31-33": "Manufacturing",
    "42": "WholesaleTrade",
    "44-45": "RetailTrade",
    "48-49": "Transportation_Warehousing",
    "51": "Information",
    "52": "Finance_Insurance",
    "53": "RealEstate_Rental_Leasing",
    "54": "Professional_Scientific_TechnicalServices",
    "55": "ManagementOfCompanies_Enterprises",
    "56": "Administrative_Support_WasteManagement_RemediationServices",
    "61": "EducationalServices",
    "62": "HealthCare_SocialAssistance",
    "71": "Arts_Entertainment_Recreation",
    "72": "Accommodation_FoodServices",
    "81": "OtherServices",
    "92": "PublicAdministration"
}

# Extract the first two digits
df['NAICS'] = df['NAICS'].astype(str).str[:2]

# Map the two-digit code to a category
df['NAICS'] = df['NAICS'].map(naics_mapping)

# Drop the indexes where NAICS is None or NaN
df = df.dropna(subset=['NAICS'])
```

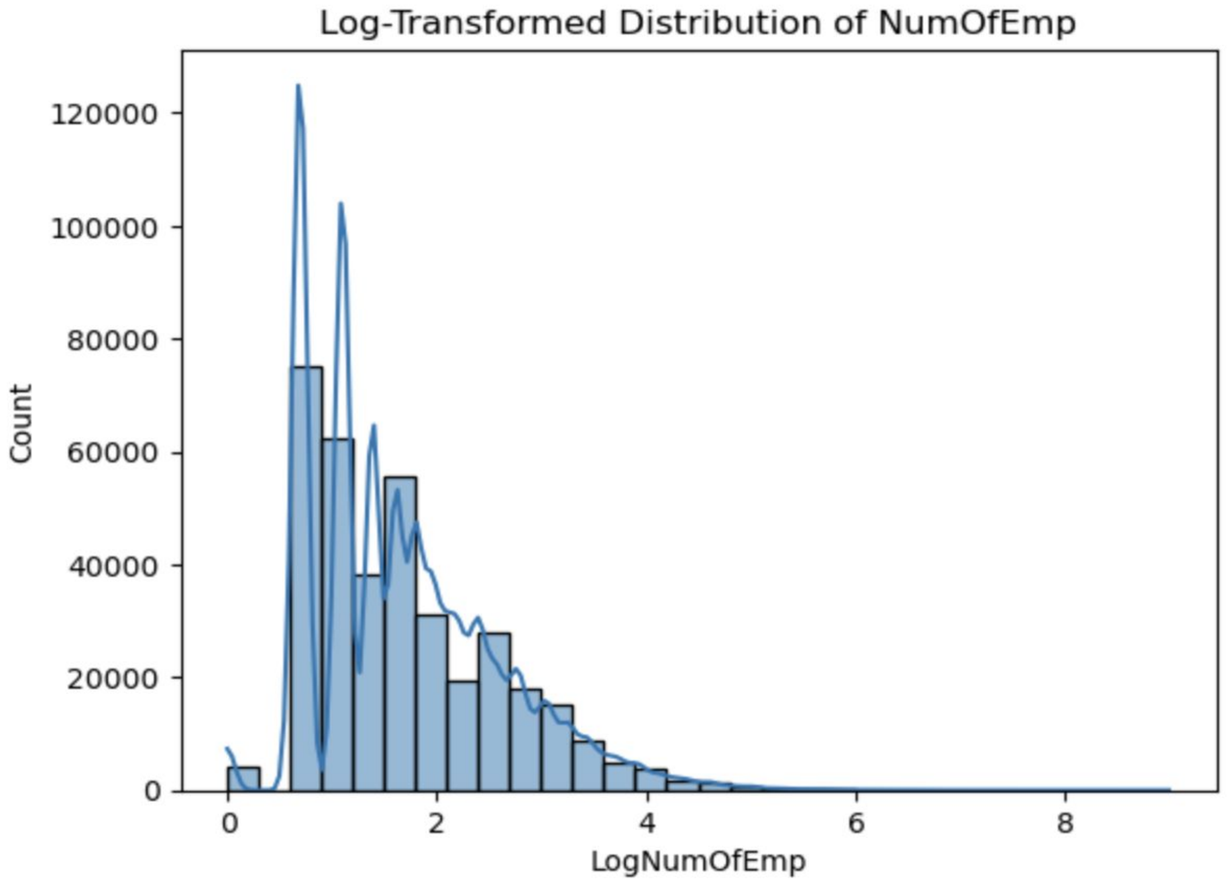
Exploratory Analysis



Key Takeaways:

- Created the “Term_Years” column by converting loan terms from months to years for easier interpretation.
- Plotted a histogram to show the relationship between loan term lengths and loan status (Paid in Full vs. Defaulted).
- Observed that shorter-term loans are more likely to be repaid, while longer-term loans show higher default rates.

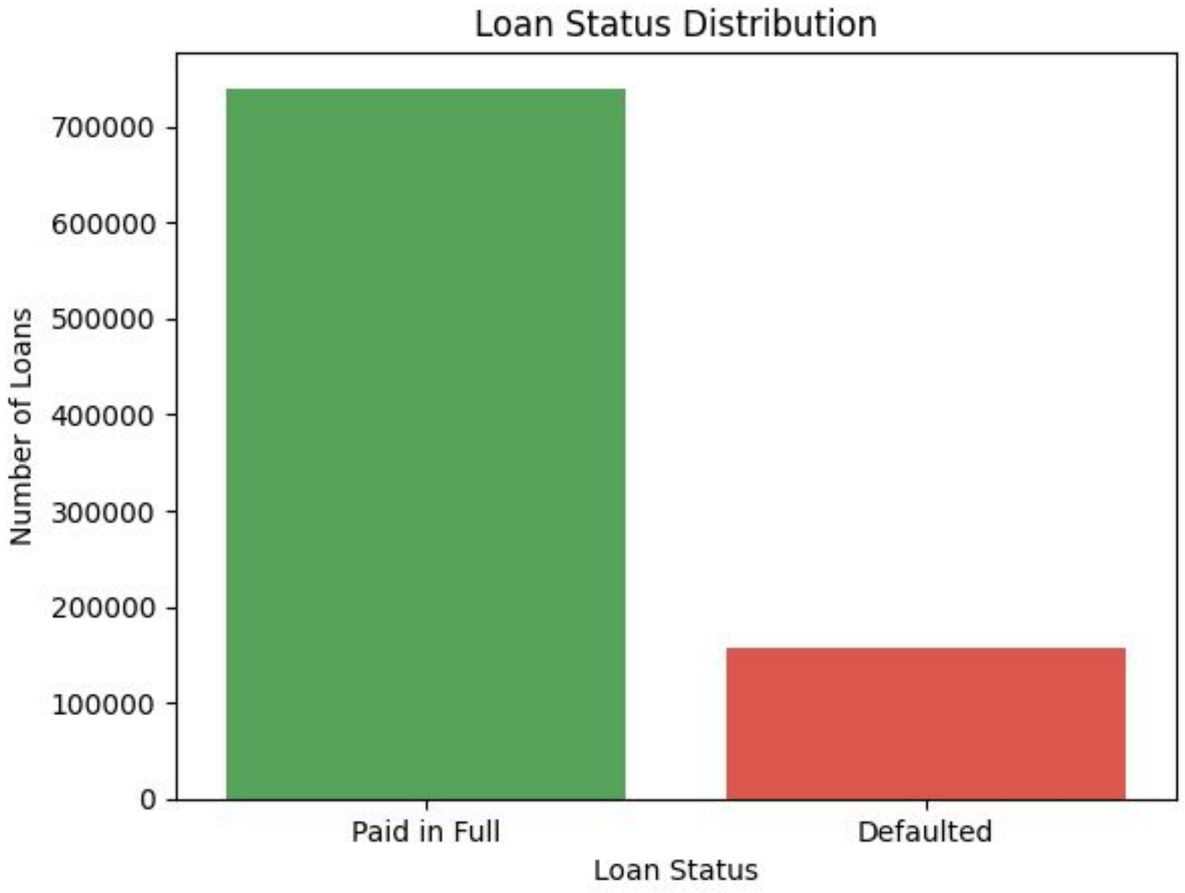
Exploratory Analysis



Key Takeaways:

- Applied a log transformation to the “NumOfEmp” column to handle skewed data.
- Revealed that most businesses have a small number of employees, shown by high frequencies at lower log values.
- Fewer businesses have large workforces, indicated by the lower frequencies at higher log values.
- The transformation smoothed the distribution, making patterns easier to interpret and compare.

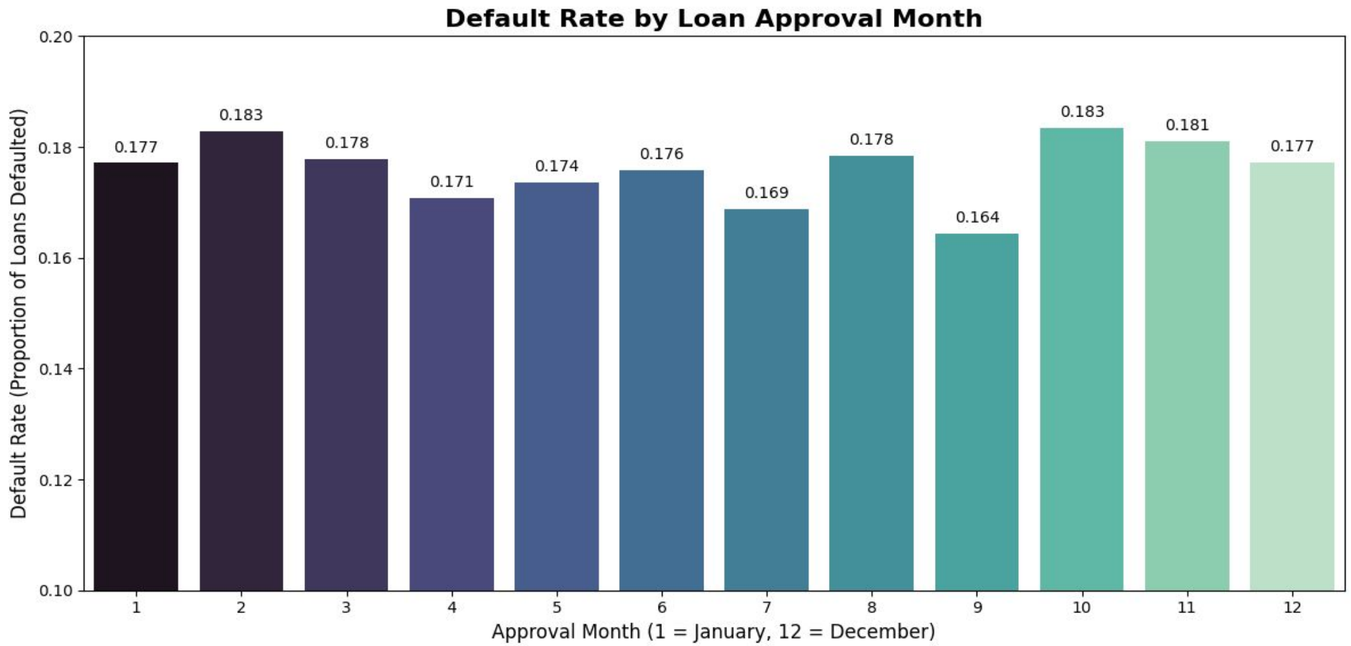
Exploratory Analysis



Key Takeaways:

- Visualized the distribution of loan statuses using a bar chart.
- The green bar indicates loans that were "Paid in Full", which occur significantly more often than defaults.
- The red bar represents "Defaulted" loans, showing a much smaller proportion.
- This clear class imbalance highlights a potential issue for model training.
- Addressing this imbalance is important to ensure accurate and fair predictive performance.

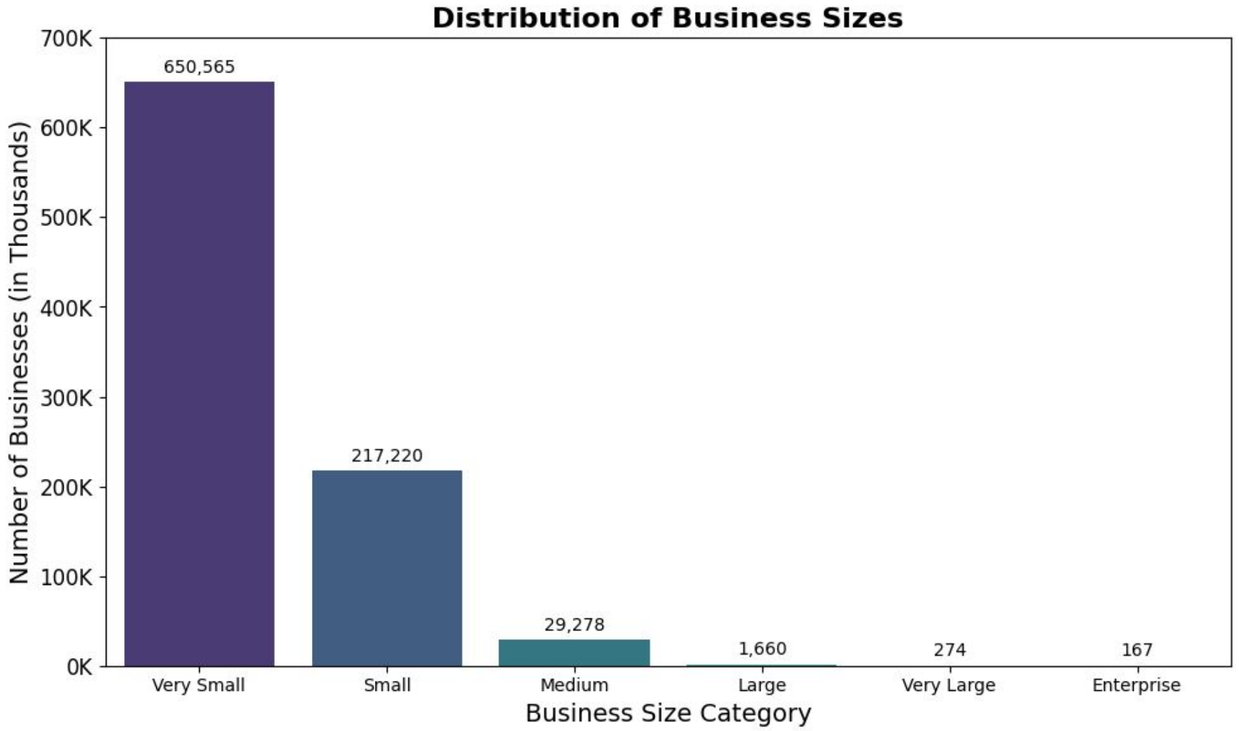
Exploratory Analysis



Key Takeaways:

- The graph displays default rates by loan approval month to identify temporal trends.
- September has the lowest default rate at 0.164, indicating better borrower performance.
- February and October have the highest default rates at 0.183.
- Most months show default rates between 0.176 and 0.183, suggesting overall consistency.
- Seasonal variations may influence borrower behavior and risk.
- Lenders should analyze high-default months to refine approval strategies.
- Borrowers tend to perform best with loans approved in September.

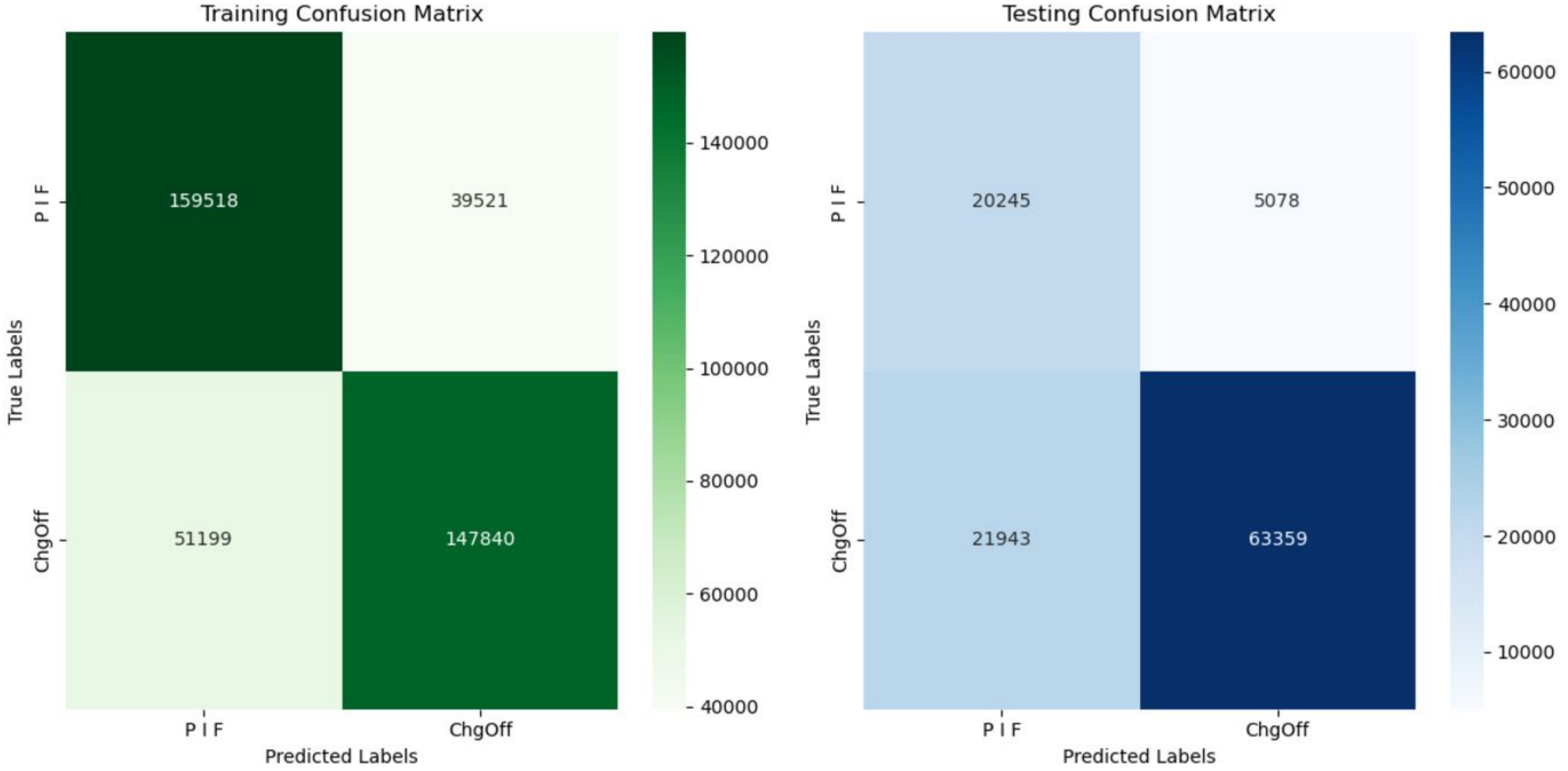
Exploratory Analysis



Key Takeaways:

- The graph shows the distribution of businesses by size category.
- "Very Small" businesses dominate the dataset with 650,565 entities.
- "Small" businesses follow with 217,220 entities.
- Larger categories like "Medium" (29,278), "Large" (1,660), "Very Large" (274), and "Enterprise" (167) have significantly fewer businesses.
- This highlights the predominance of smaller enterprises in the dataset.

Confusion Matrix



Classification Report

Training Classification Report:

	precision	recall	f1-score	support
0	0.76	0.80	0.78	199039
1	0.79	0.74	0.77	199039
accuracy			0.77	398078
macro avg	0.77	0.77	0.77	398078
weighted avg	0.77	0.77	0.77	398078

Testing Classification Report:

	precision	recall	f1-score	support
0	0.48	0.80	0.60	25323
1	0.93	0.74	0.82	85302
accuracy			0.76	110625
macro avg	0.70	0.77	0.71	110625
weighted avg	0.82	0.76	0.77	110625

Specificity

```python
y_pred = logit_reg.predict(X_test)

from sklearn.metrics import confusion_matrix

# Assuming y_test (true labels) and y_pred (predicted labels) are available
cm = confusion_matrix(y_test, y_pred)

# Extract values from confusion matrix
TN = cm[0, 0]  # Top-left value: True  Negatives
FP = cm[0, 1]  # Top-right value: False Positives

# Calculate Specificity
specificity = TN / (TN + FP)

print(f"Specificity: {specificity:.2f}")
```

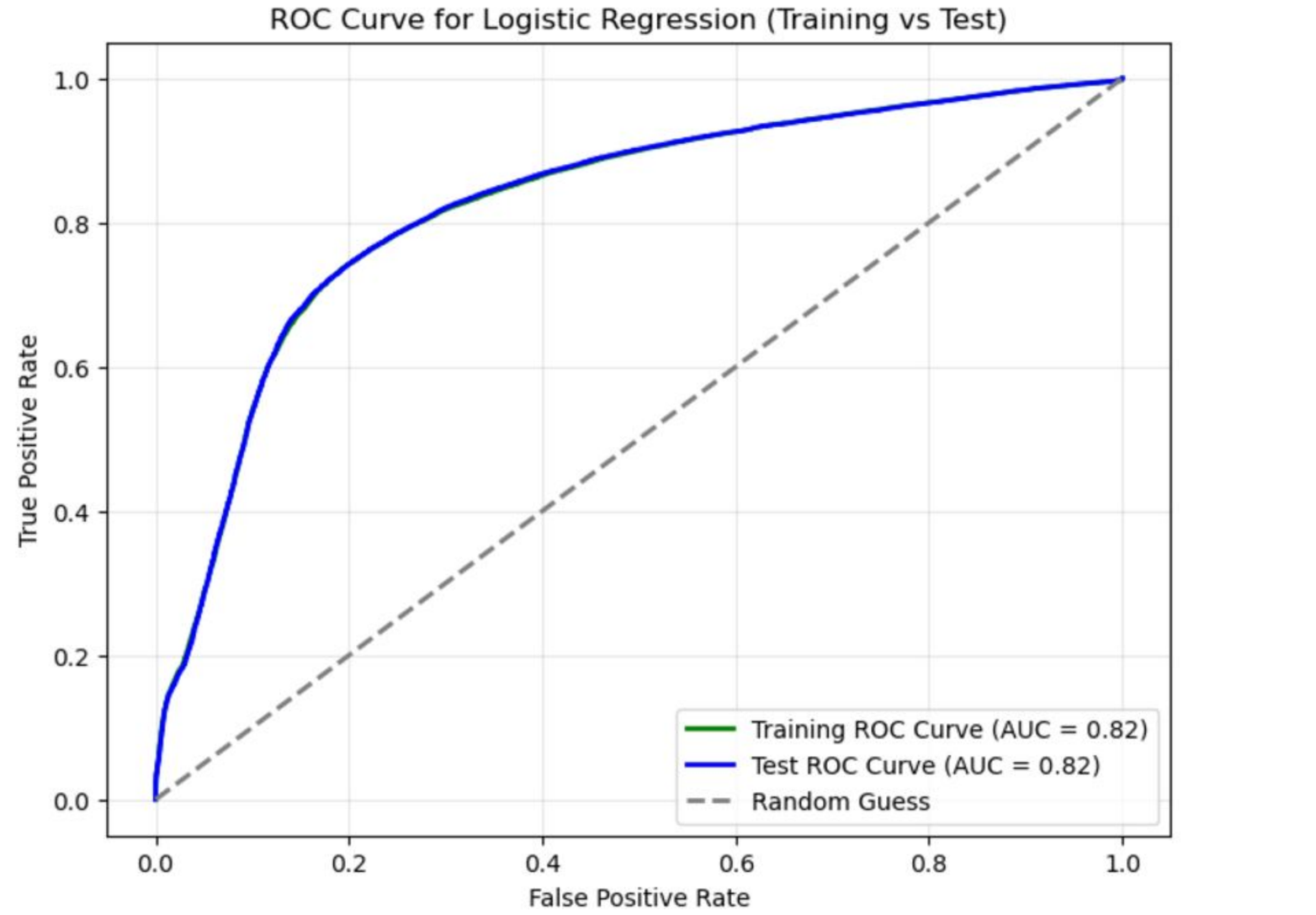
```
Specificity: 0.80
```

$$\text{Specificity} = \frac{\text{True Negatives (TN)}}{\text{True Negatives (TN)} + \text{False Positives (FP)}}$$

True Negatives (TN): Loans predicted as "No Default/PIF" that truly didn't default.

False Positives (FP): Loans predicted as "No Default/PIF" but actually defaulted.

ROC Curve


ROC Curve for Logistic Regression (Training vs Test)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Training ROC Curve (AUC = 0.82)
Test ROC Curve (AUC = 0.82)
Random Guess

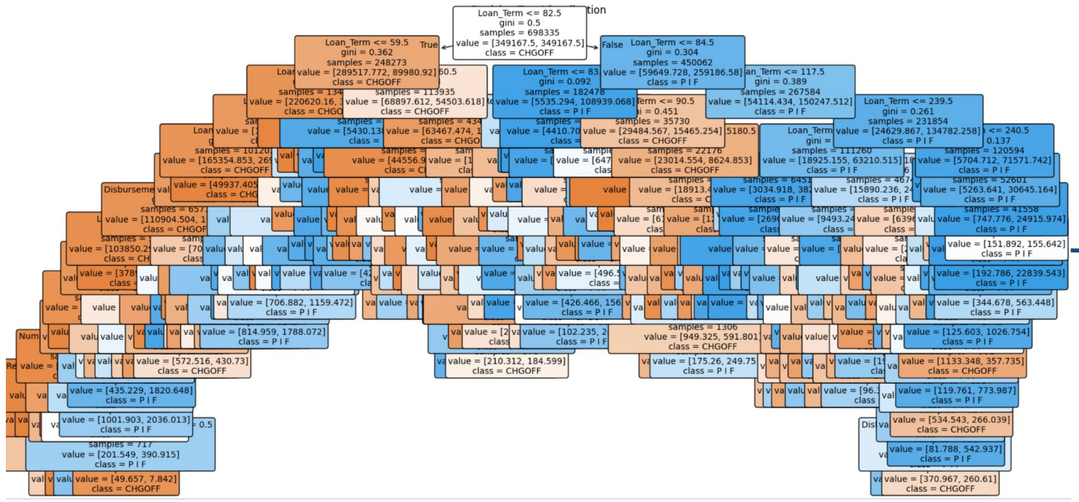
Random Forest Model

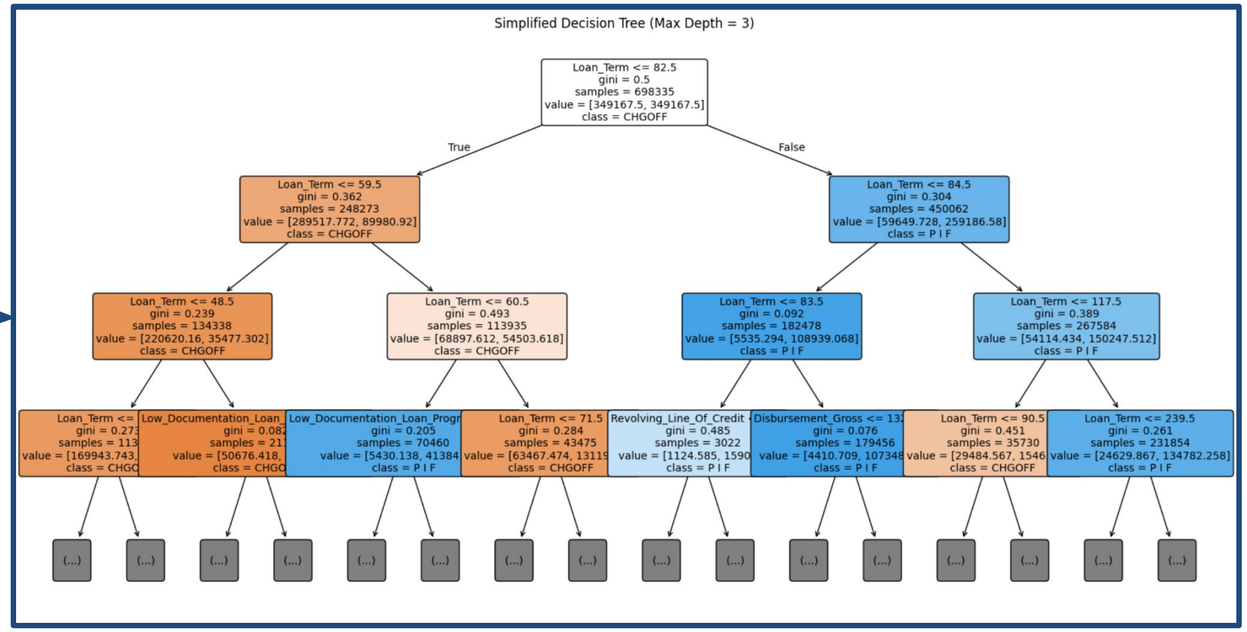
Random Forest

- Random Forest is an ensemble learning method used for:
  - Classification tasks
  - Regression tasks
- It builds multiple decision trees during training.
  - Each tree is trained on:
    - A random subset of the data
    - A random subset of the features
- For classification, it outputs the majority vote of all trees.

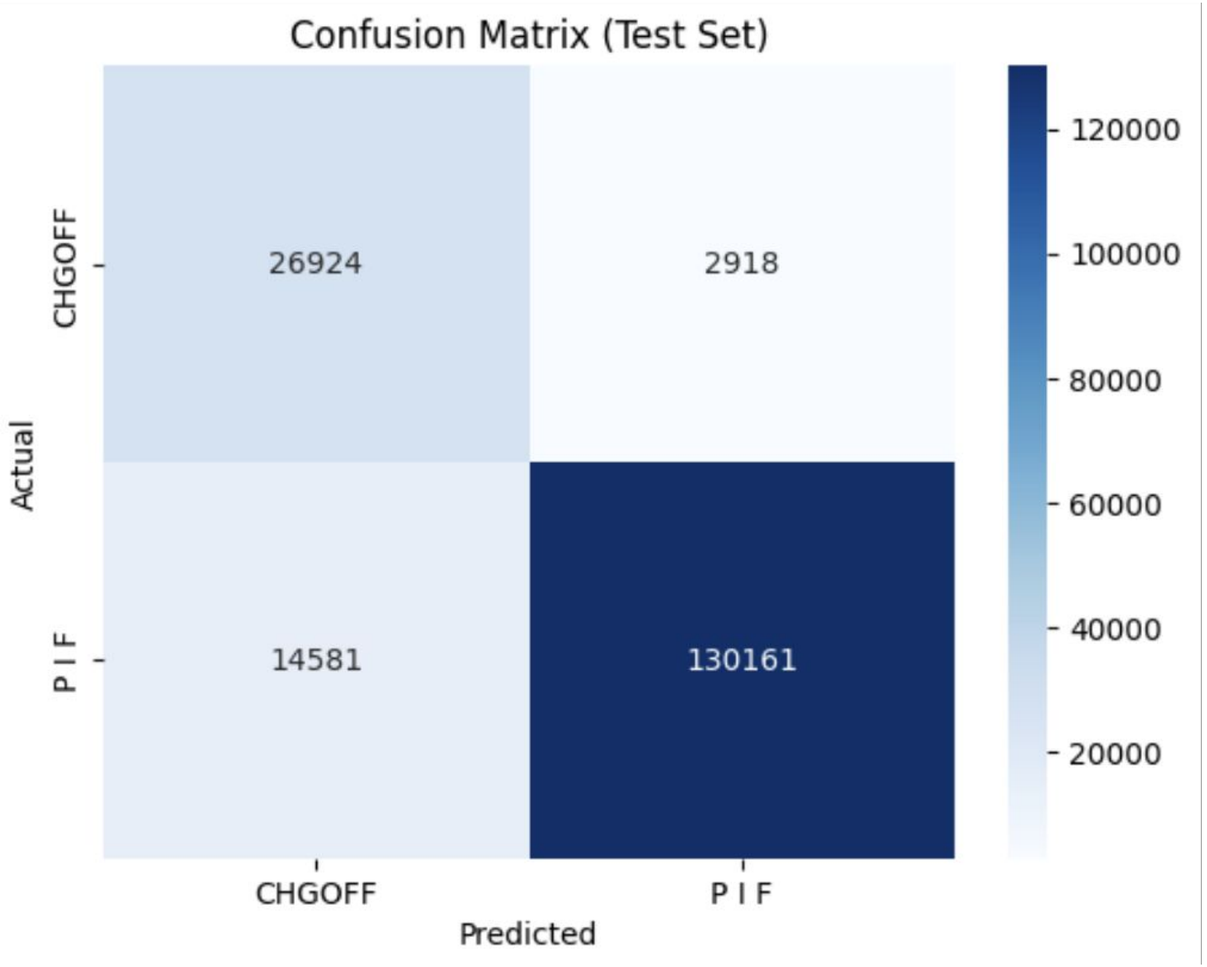
- Why Random Forest?
  - Handles imbalanced data (only 15% defaults)
  - Robust to outliers in loan amounts
  - Provides feature importance

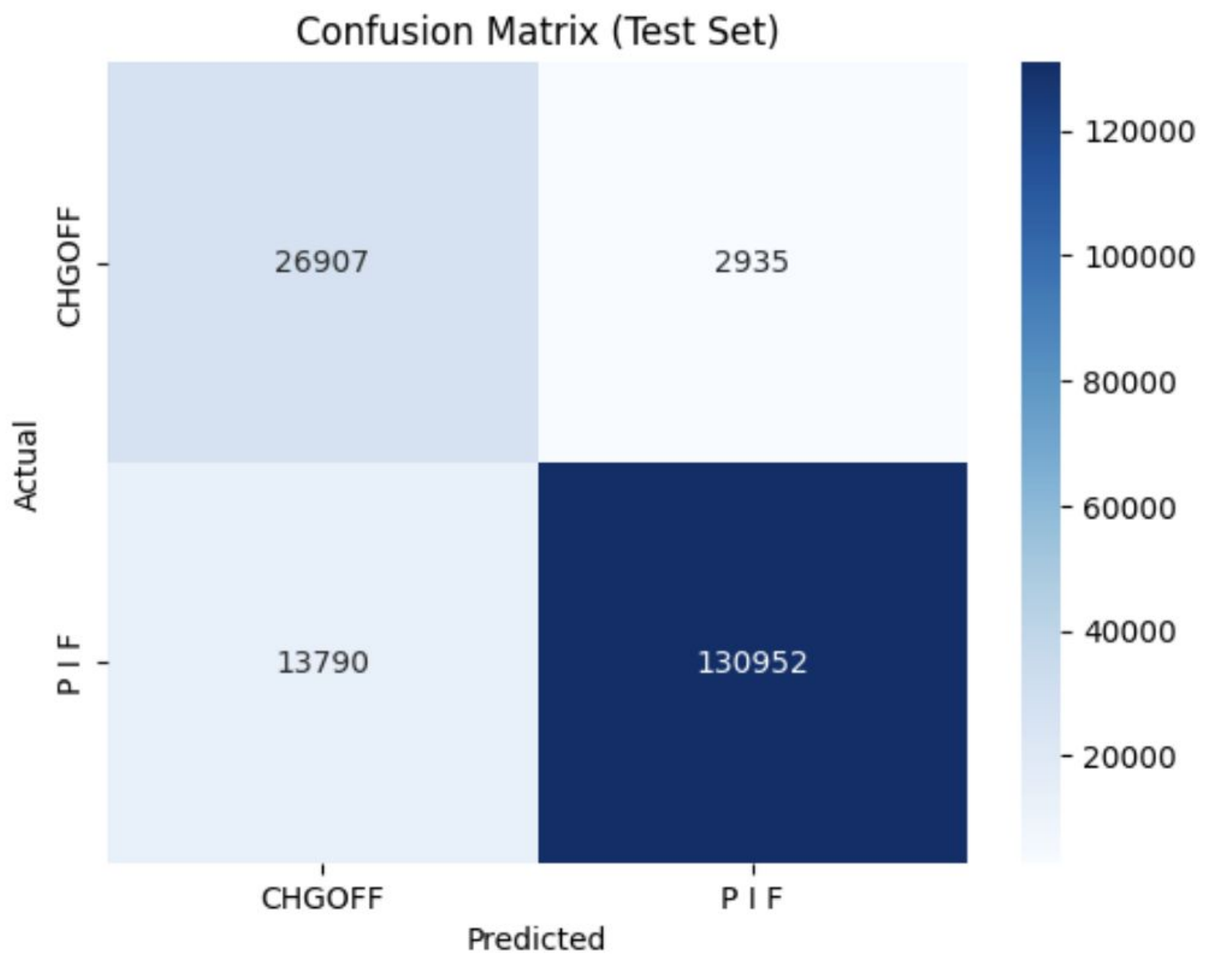
Decision Tree Visualization





Confusion Matrix





Classification Report

```
Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.99      0.98       134
           1       1.00      0.98      0.99       274

    accuracy                           0.99       408
   macro avg       0.98      0.99      0.98       408
weighted avg       0.99      0.99      0.99       408
```

Class-Specific Metrics

Class 0 (Minority Class)

Precision: 96%

When predicted as 0, it's correct 96% of the time

Recall: 99%

Captured 99% of actual 0 instances

F1-score: 98%

Class 1 (Majority Class)

Precision: 100%

All predictions of 1 were correct

Recall: 98%

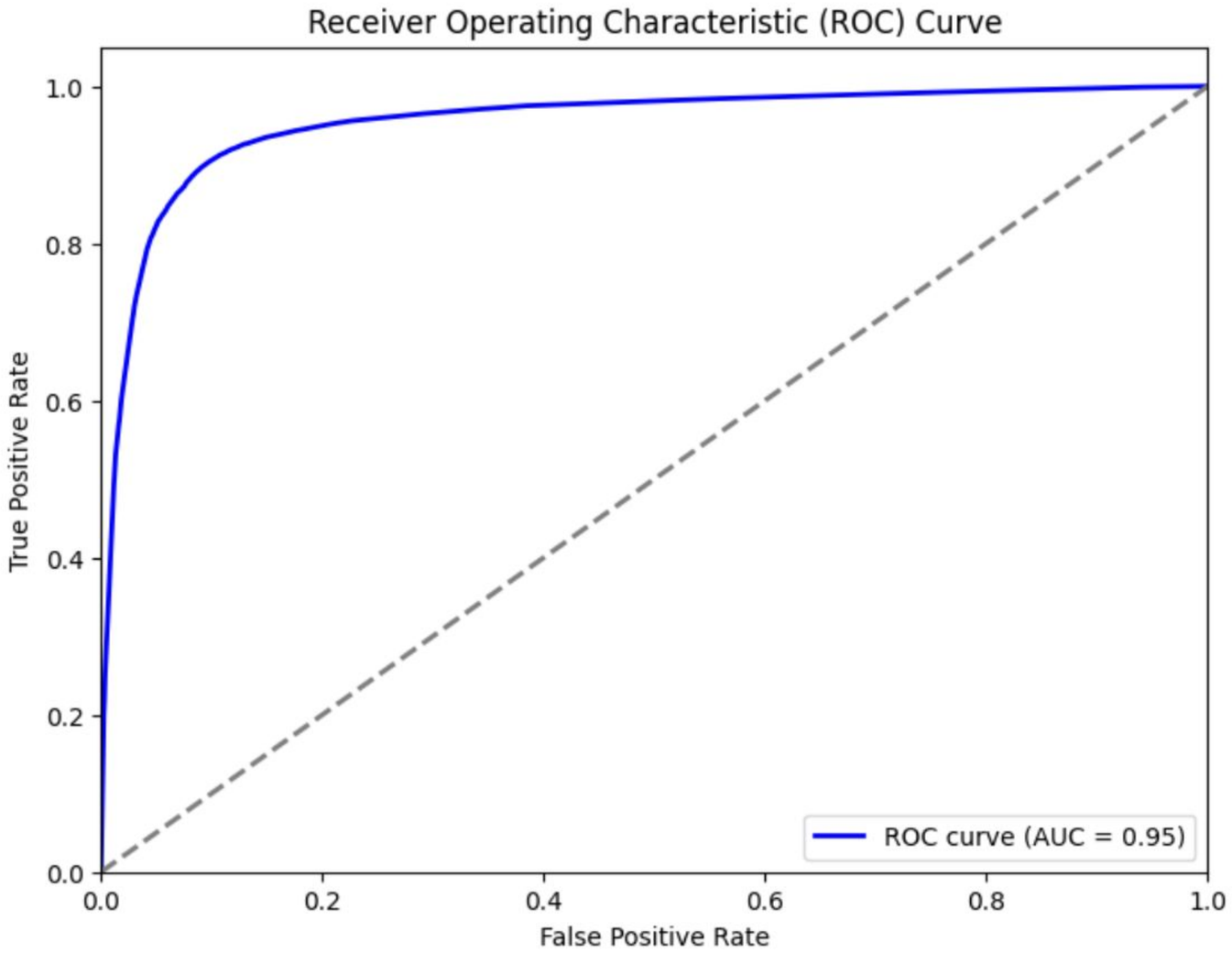
Identified 98% of actual 1 instances

F1-score: 99%

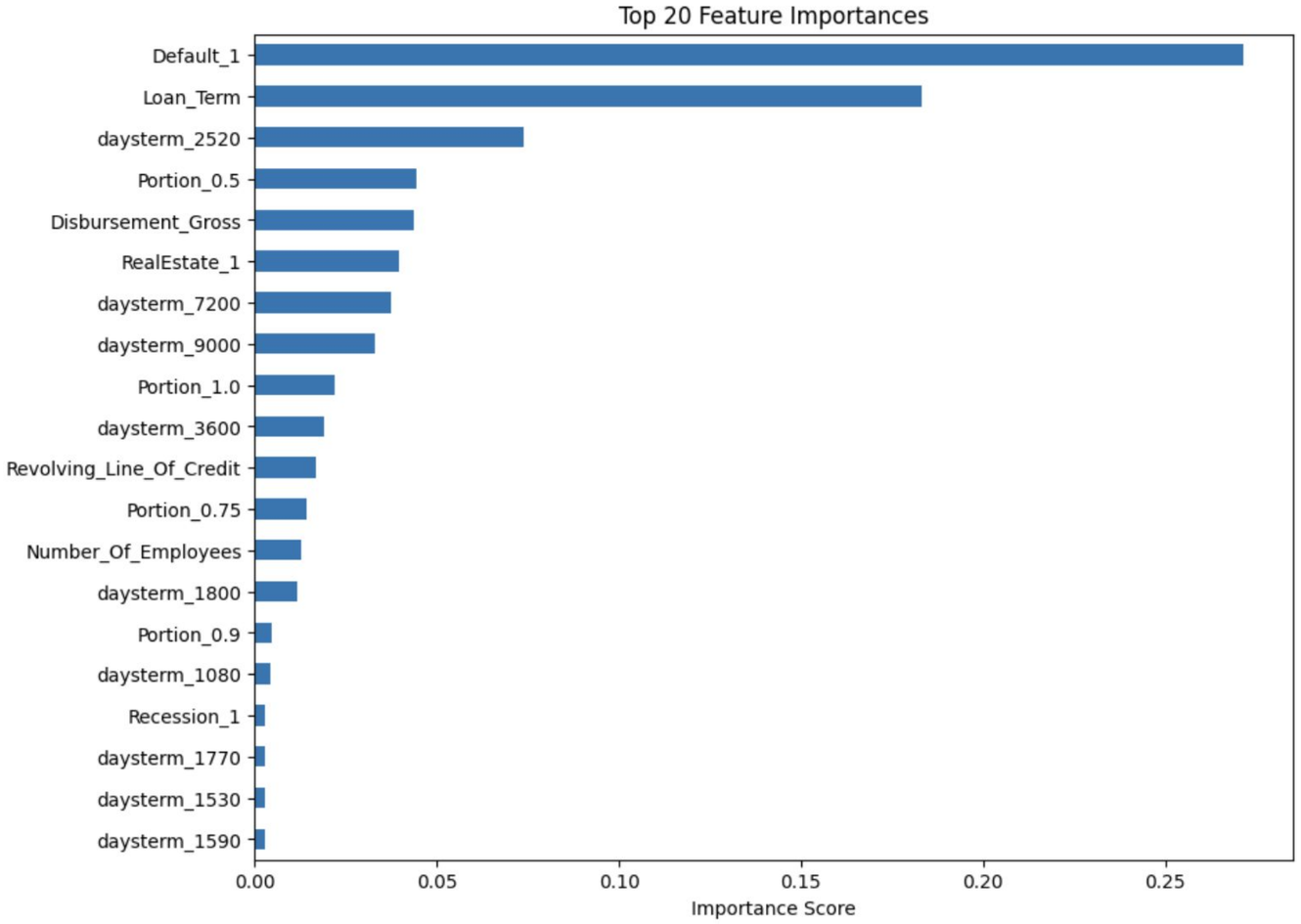
Key Observations

- **Excellent balanced performance (macro avg F1: 98%)**
- **Slightly better performance on majority class (274 samples vs 134)**
- **Near-perfect recall for both classes minimizes false negatives**

ROC Curve


Receiver Operating Characteristic (ROC) Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.95)

Feature Importance



Cross Validation

```
Cross-Validation Accuracy Scores: [0.9877451  0.99509804 0.99509804 0.9877451  0.995086  ]
Mean CV Accuracy: 0.9922 (±0.0036)
```

Conclusion

1. Model Performance
   - Random Forest outperformed Logistic Regression on default **recall (>90% vs. ~75%)** and **ROC-AUC (~0.95 vs. ~0.80)** while maintaining high **precision (~96% vs. ~93%)**.
2. Key Risk Drivers
   - **loan term, disbursement amount, and revolving line of credit status** are the strongest predictors of default.
3. Practical Implications
   - Risk Tiers: Use RF scores to auto-approve low-risk, flag medium-risk for quick review, and escalate high-risk applications.
   - Reason Codes: Leverage logistic coefficients to provide simple “why” explanations (e.g., “Long term +12% risk,” “No RevLineCr +8% risk”).
4. Next Steps
   - Threshold Tuning: Set your cutoff to balance your false-positive vs. false-negative tolerance.
   - Model Refresh: Retrain periodically with new SBA data to adapt to market changes.

References

[1] M. Modina, F. Pietrovito, C. Gallucci, and V. Formisano, “Predicting SMEs’ default risk: Evidence from bank-firm relationship data,” *The Quarterly Review of Economics and Finance*, vol. 89, pp. 254–268, Jun. 2023, doi: https://doi.org/10.1016/j.qref.2023.04.008.

[2] Hamid Cheraghali and P. Molnár, “Practical insights into predicting defaults in small and medium-sized enterprises,” *Journal of the International Council for Small Business*, pp. 1–12, Dec. 2024, doi: https://doi.org/10.1080/26437015.2024.2430573.

[3] Y. Dendramis, E. Tzavalis, and A. Cheimarioti, “Measuring the Default Risk of Small Business Loans: Improved Credit Risk Prediction using Deep Learning,” *SSRN Electronic Journal*, 2020, doi: https://doi.org/10.2139/ssrn.3729918.

[4] S. Albanesi and D. Vamossy, “Predicting Consumer Default: A Deep Learning Approach,” *SSRN Electronic Journal*, 2019, doi: https://doi.org/10.2139/ssrn.3445152.

[5] A. Bitetto, P. Cerchiello, Stefano Filomeni, A. Tanda, and B. Tarantino, “Can we trust machine learning to predict the credit risk of small businesses?,” *Review of Quantitative Finance and Accounting*, Jun. 2024, doi: https://doi.org/10.1007/s11156-024-01278-0.

thanks for watching