MKT- 869 Machine Learning

Team- Dab

Anjana George

Dhara Dhruv

Kajal Patil

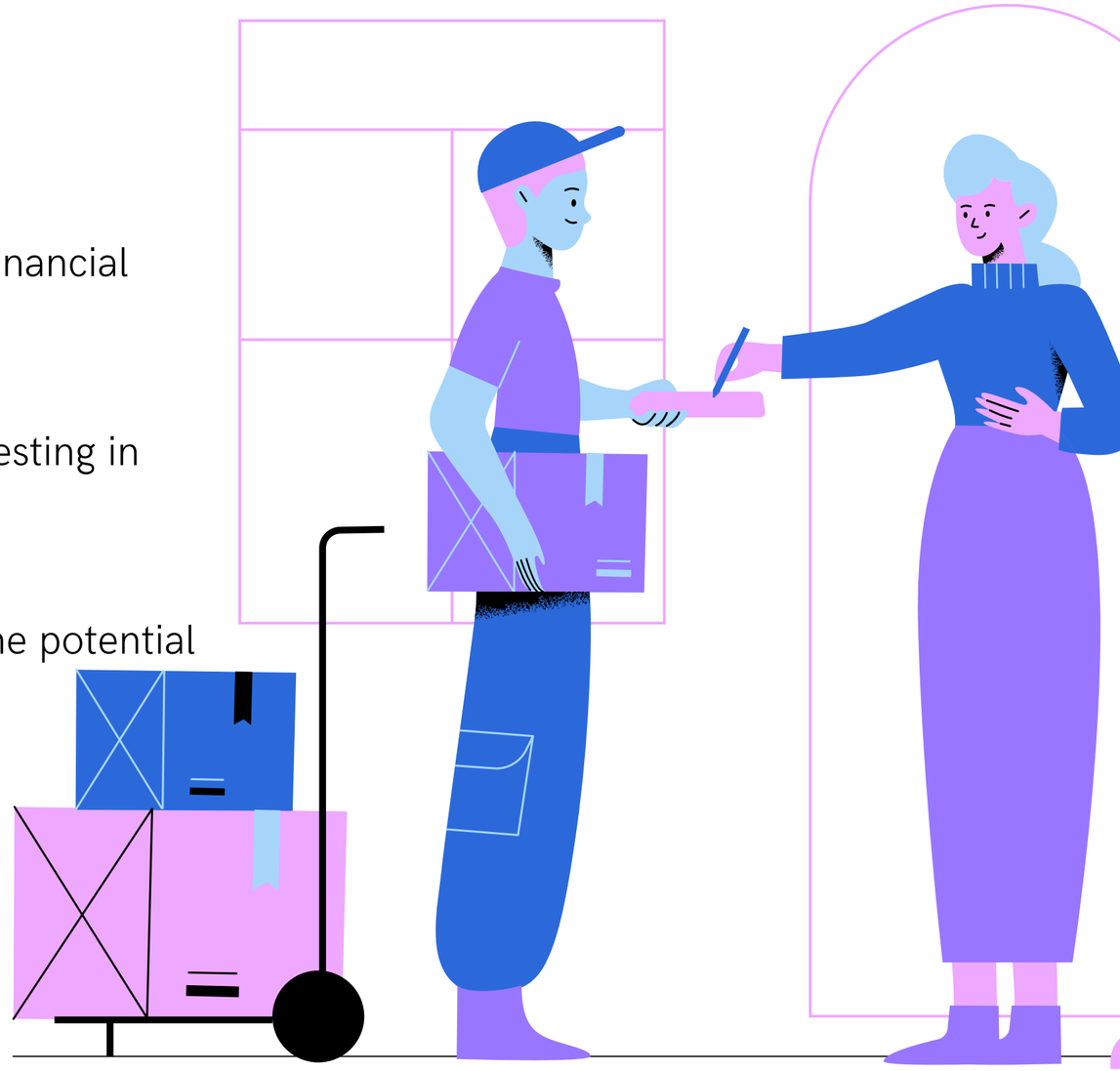
Sarika Nimkar

Agenda

- INTRODUCTION
- RESEARCH OBJECTIVE
- METHODOLOGY
- DATA CLEANING AND FEATURE ENGINEERING
- MODEL DEVELOPMENT PROCESS
- MODEL PERFORMANCE INTERPRETATION
- MARKETING STRATEGIES
- CONCLUSION

Overview

- Banking / Financial Institutes plays a significant role in providing financial service
- To maintain the integrity, bank/institute must be careful when investing in customers to avoid financial loss
- Before giving credit to borrowers, the bank must come to about the potential of customers.
- The term credit scoring, determines the relation between defaulters and loan characteristics



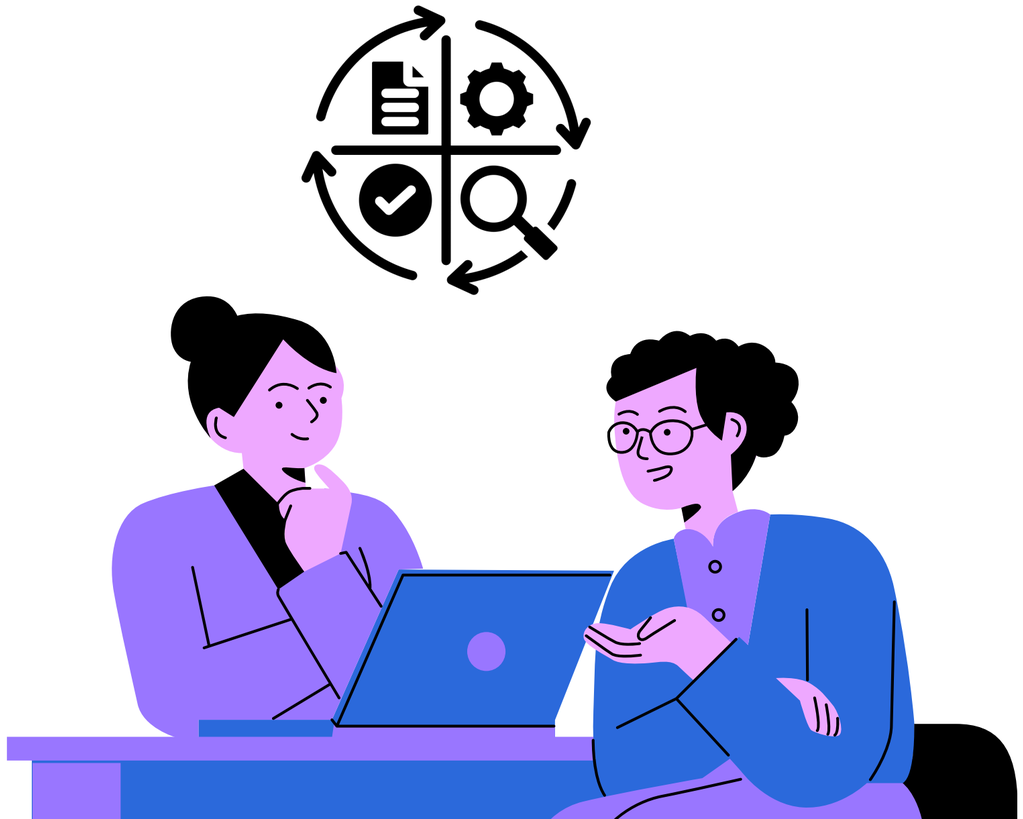
Predicting Credit Card Payment Defaults: A Financial Behavior Analysis

Default is the failure to pay interest or principal on a loan or credit card payment

OBJECTIVE	The model we built here will use all possible factors to predict data on customers to find who are defaulters and non-defaulters next month
	The goal is to find the whether the clients are able to pay their next month credit amount
	Identify some potential customers for the bank who can settle their credit balance
	To determine if their customers could make the credit card payments on-time

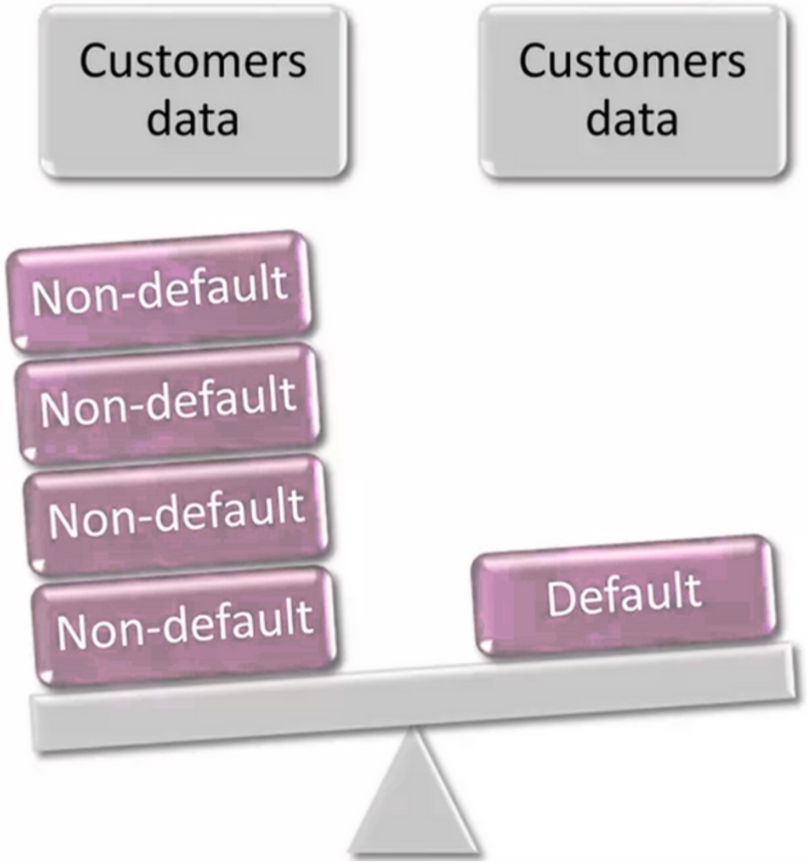
Methodology

The logistic regression model serves as a predictive tool, and various metrics help assess its performance in different aspects such as accuracy, precision, sensitivity, and specificity



- Data Preprocessing
- Exploratory Data Analysis
- Feature Scaling
- Model Testing and Optimization

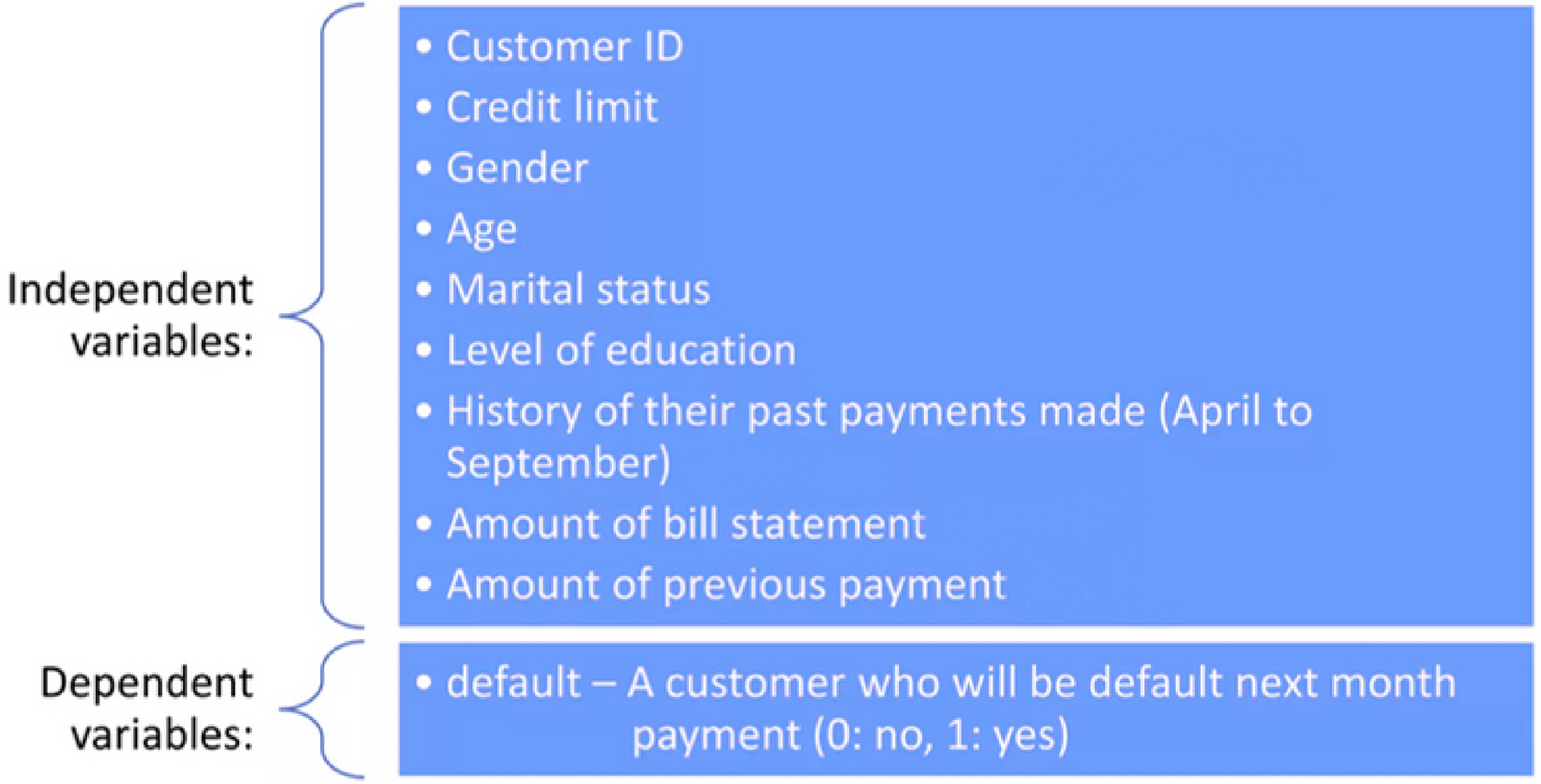
Dataset Overview



Attributes: 24

Tuples: 30000

Knowing Features/ Variables



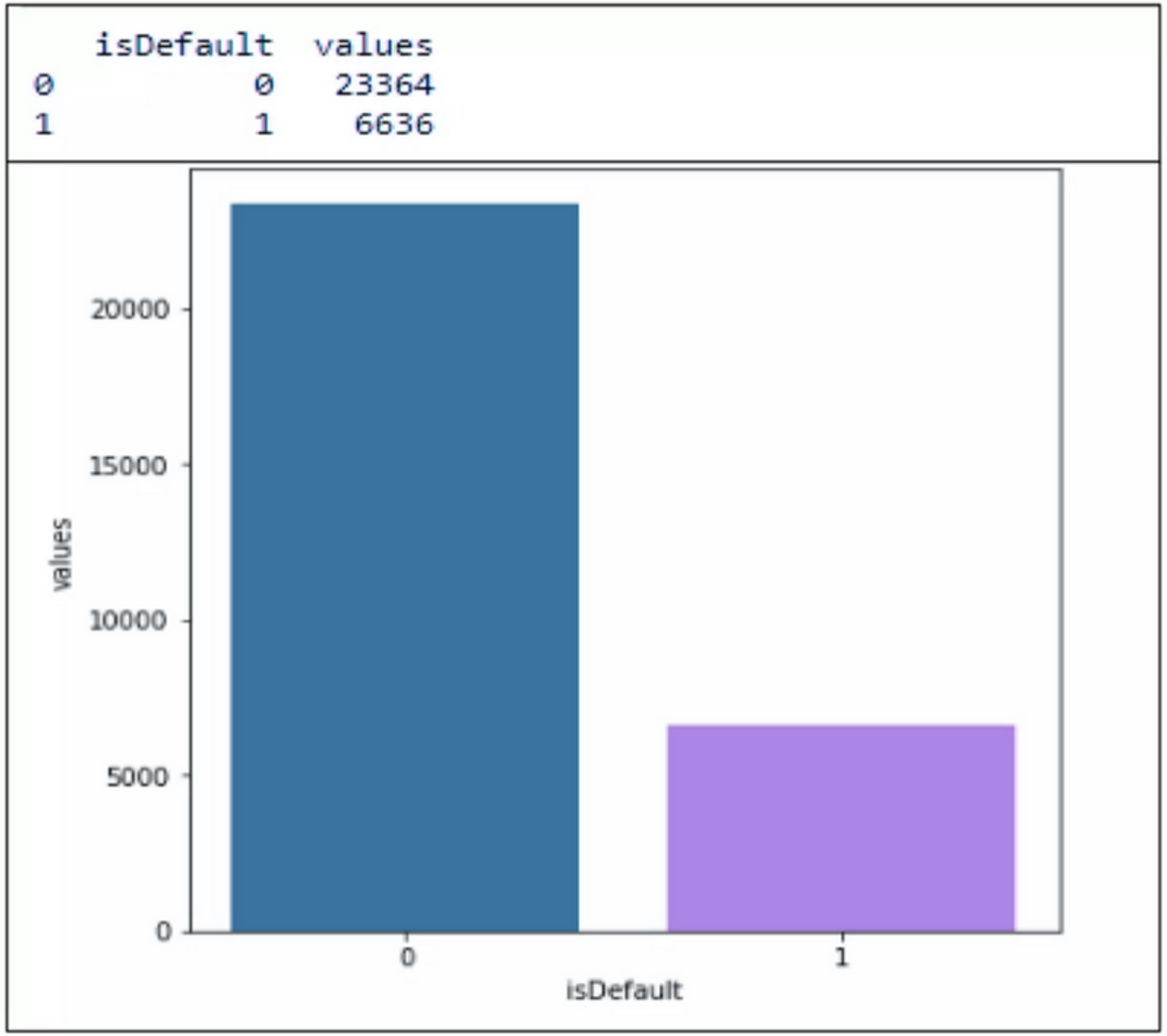
Data Exploration

- Graph shows total number of records for defaulters and non-defaulters.

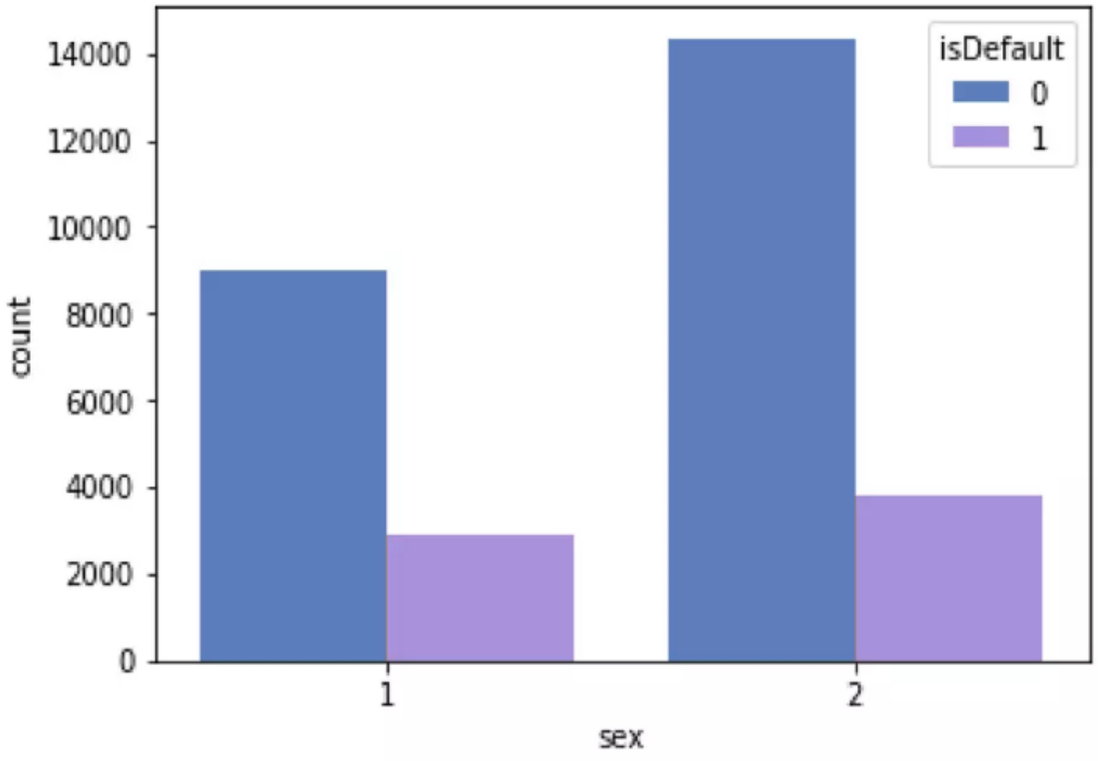
- If they would do payment or not (yes=1 no=0) for next month.

22% - default
78% - non-default

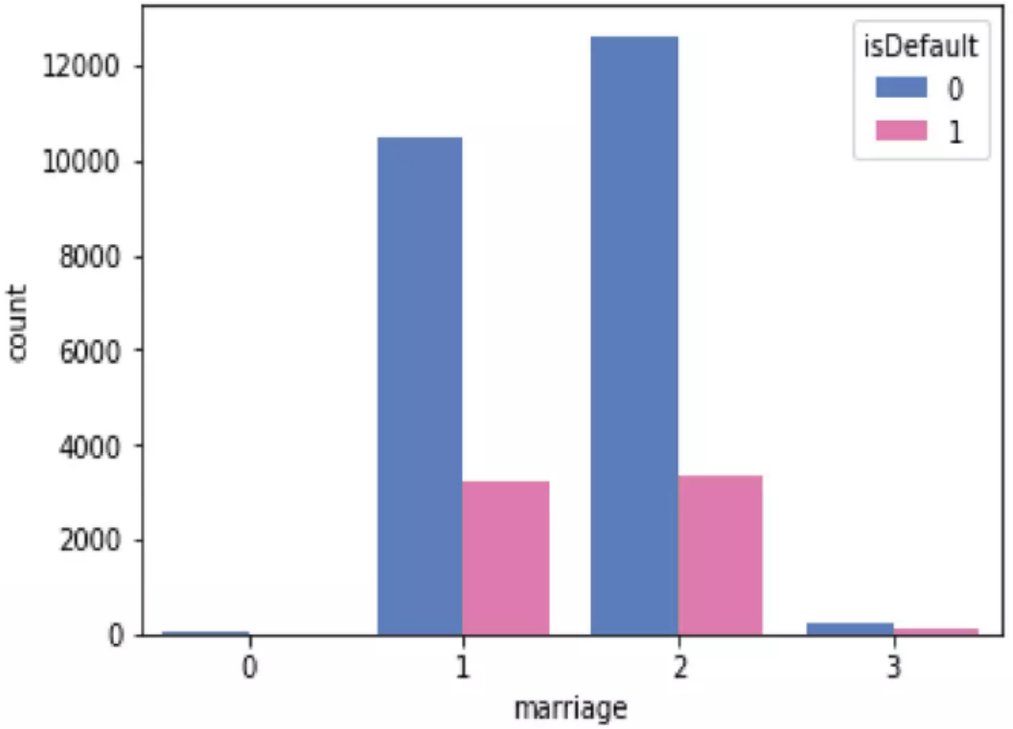
	isDefault	values
0	0	23364
1	1	6636



Continued...

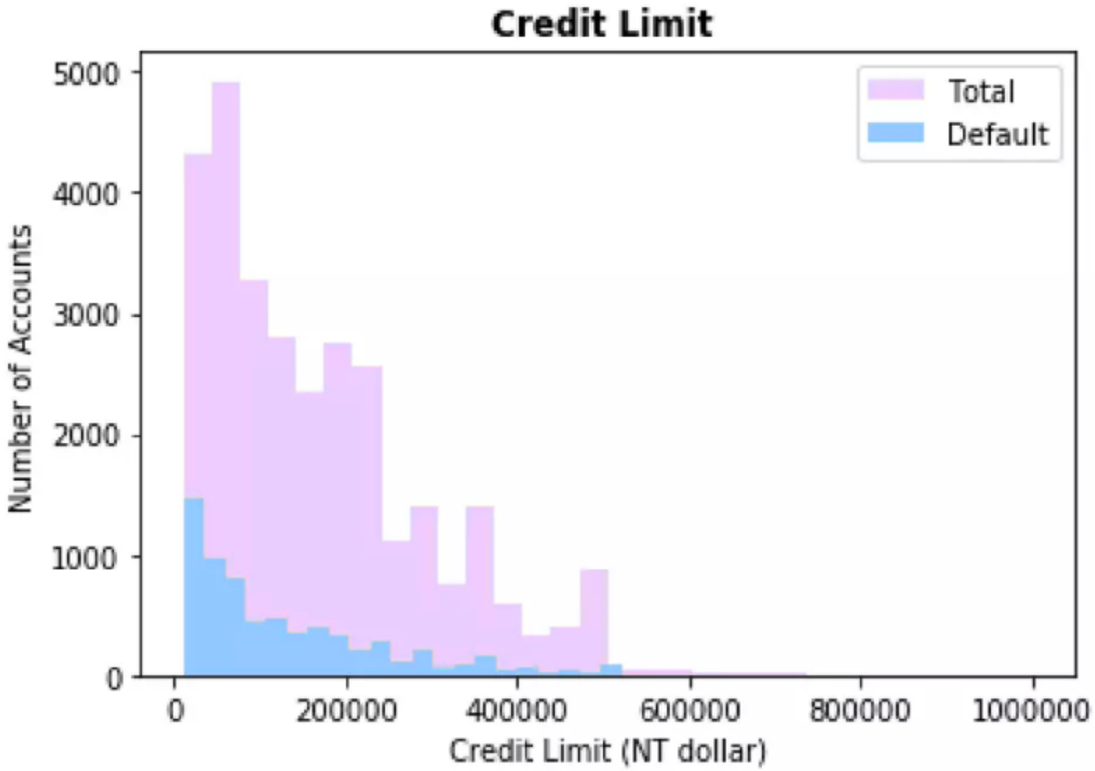


1. It shows count for 'sex' attribute
1 - Male and 2 - Female

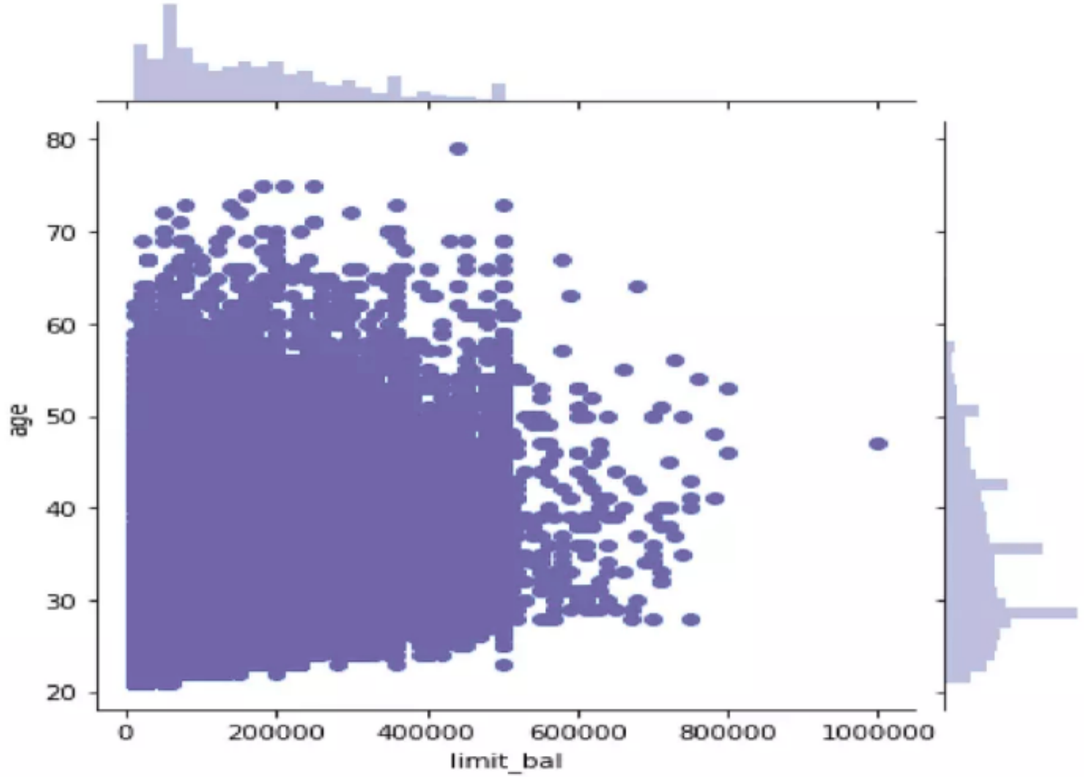


2. It shows default count for 'marriage' attribute
0,3 - Others, 1 - Married, 2 - Single

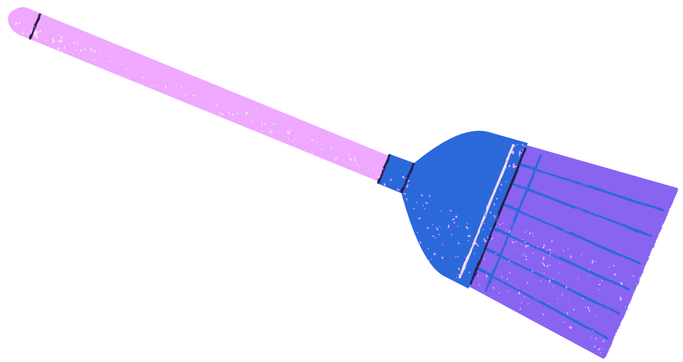
Continued...



1. It shows count for Total 'Credit limit' attribute values with respect to Number of Account



2. It shows Limit of balance with respect to age.



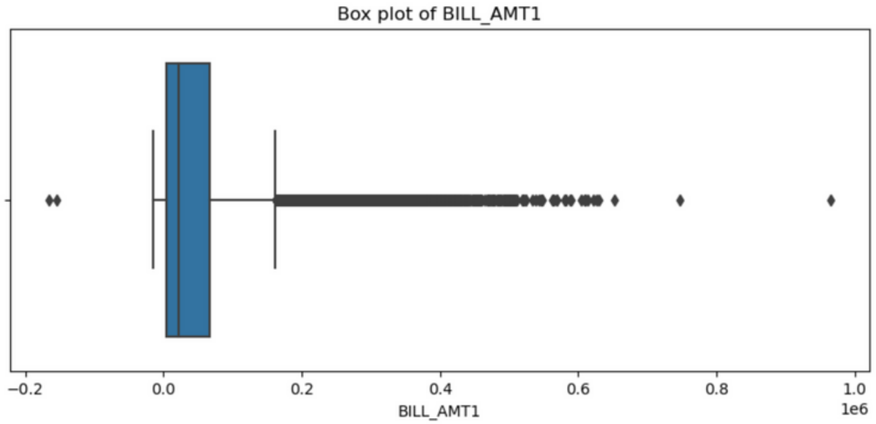
Data Cleaning

```
In [19]: # Count your missing cases:
         print(df.isnull().sum())

SEX          0
EDUCATION    0
MARRIAGE     0
AGE          0
PAY_1        0
PAY_2        0
PAY_3        0
PAY_4        0
PAY_5        0
PAY_6        0
BILL_AMT1    0
BILL_AMT2    0
BILL_AMT3    0
BILL_AMT4    0
BILL_AMT5    0
BILL_AMT6    0
PAY_AMT1     0
PAY_AMT2     0
PAY_AMT3     0
PAY_AMT4     0
PAY_AMT5     0
PAY_AMT6     0
dpnm         0
dtype: int64
```

- The dataset, containing data on 30,000 credit card clients, showed no missing values across all 25 columns.
- This absence of null values streamlined the data cleaning process, as there was no immediate need for handling missing data.
- The dataset's completeness is an excellent starting point for analysis, ensuring reliability and integrity in the data.

Outlier Detection & Removal



OUTLIER DETECTION METHOD

→ Used box plots for each numeric variable to visually identify outliers, highlighting data spread and extremes

→ The interquartile range (IQR) method was employed to define outliers, where any data point lying beyond 3 times the IQR above the third quartile and below the first quartile was considered an outlier.

```
# Count your outliers:
outlier_counts = {}

for col in df_num:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 3 * IQR
    upper_bound = Q3 + 3 * IQR

    outliers_count = df[(df[col] < lower_bound) | (df[col] > upper_bound)].shape[0]
    outlier_counts[col] = outliers_count

print("Outlier Counts per Column:", outlier_counts)
```

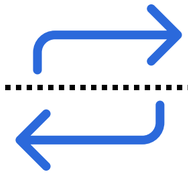
```
Outlier Counts per Column: {'SEX': 0, 'EDUCATION': 51, 'MARRIAGE': 0, 'AGE': 0, 'PAY_1': 141, 'PAY_2': 157, 'PAY_
3': 150, 'PAY_4': 169, 'PAY_5': 164, 'PAY_6': 129, 'BILL_AMT1': 786, 'BILL_AMT2': 791, 'BILL_AMT3': 858, 'BILL_AMT
4': 900, 'BILL_AMT5': 949, 'BILL_AMT6': 930, 'PAY_AMT1': 1629, 'PAY_AMT2': 1639, 'PAY_AMT3': 1560, 'PAY_AMT4': 158
3, 'PAY_AMT5': 1544, 'PAY_AMT6': 1639}
```

Outlier Count and Variables Affected

Each numeric variable was analyzed for outliers. For instance, 'PAY_1', 'PAY_2', 'BILL_AMT1', 'PAY_AMT1', and 'PAY_AMT2' showed several outliers, with payment-related variables exhibiting the highest counts. This detailed analysis is crucial for understanding which aspects of the dataset are most affected by extreme values.

```
df.shape
```

```
Out[2]: (30000, 25)
```



Impact on Dataset

The removal of outliers led to a significant reduction in the dataset size, underscoring the substantial impact of these extreme values.

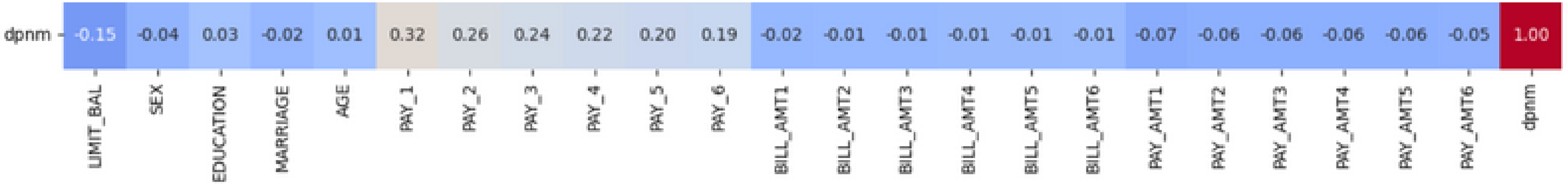
```
df_cleaned
```

```
(21591, 23)
```

Feature Engineering

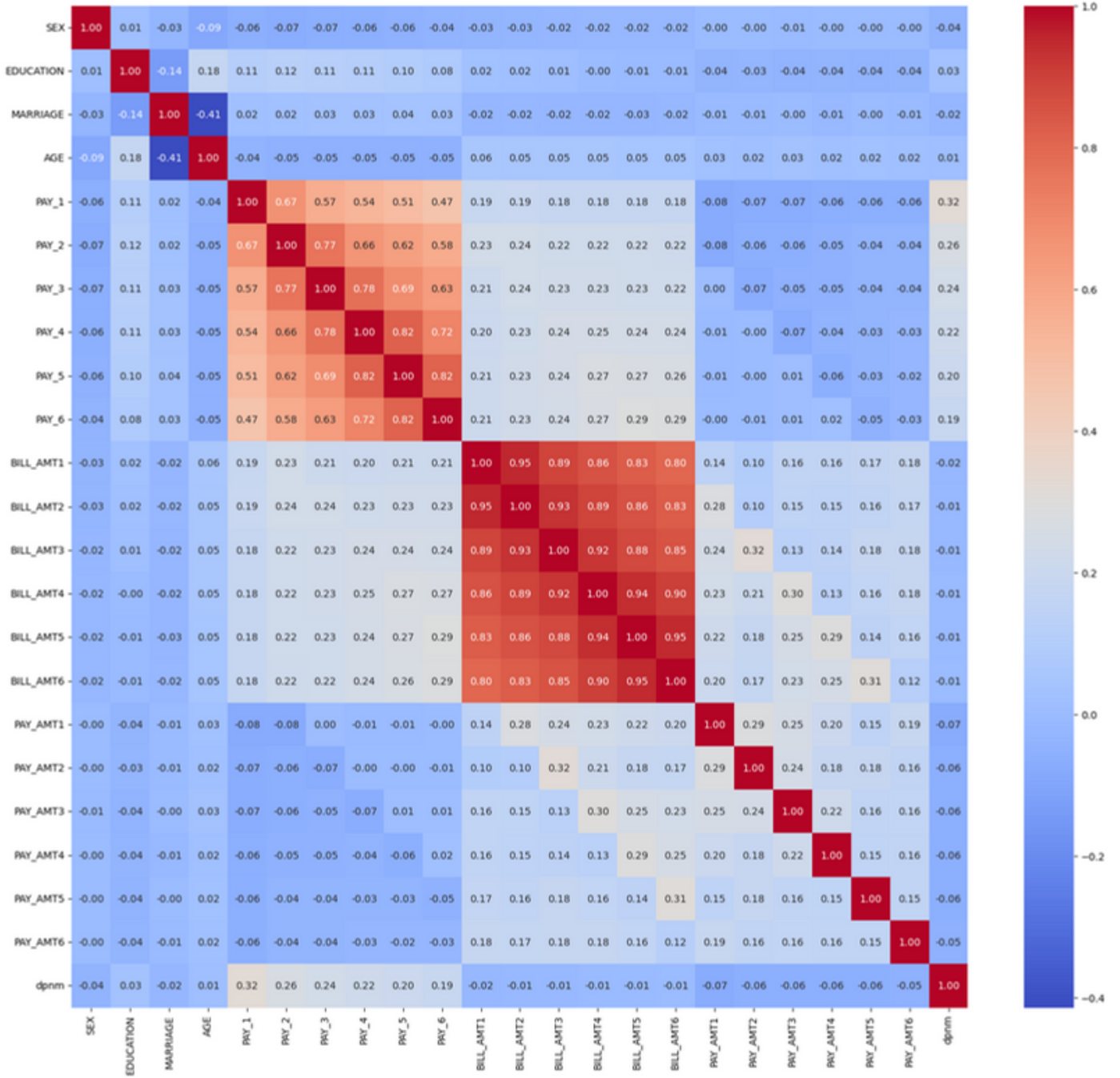
DROPPING 'LIMIT_BAL'

The decision to drop 'LIMIT_BAL' was based on its weaker direct association with default risk. Despite being a potential indicator of a client's financial health, it was not directly contributing to the likelihood of default. The focus was shifted to more directly relevant variables like payment behavior and outstanding balances.



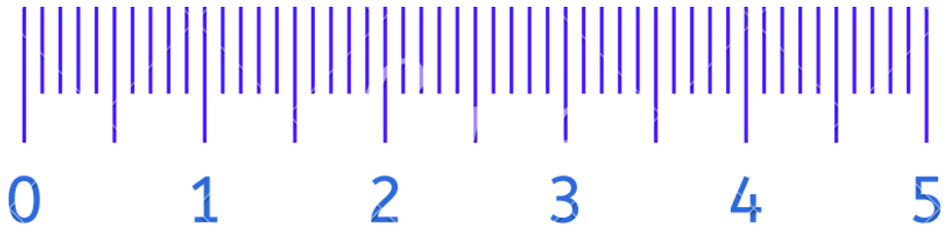
Correlation Analysis

A correlation heatmap was used to detect high multicollinearity among repayment status variables ('PAY_1' to 'PAY_6') and bill amount variables ('BILL_AMT1' to 'BILL_AMT6'). Features like 'PAY_3' to 'PAY_6' and 'BILL_AMT2' to 'BILL_AMT6' were removed to reduce redundancy and simplify the model.



Feature Transformation

Standardization is a feature scaling process that transforms the distribution of data so that the mean of observed values is 0 and the standard deviation is 1. This ensures that each feature contributes equally to the analysis and helps with the convergence of algorithms that are sensitive to the scale of input data, like many machine learning algorithms.



Standardization of Numerical Variables

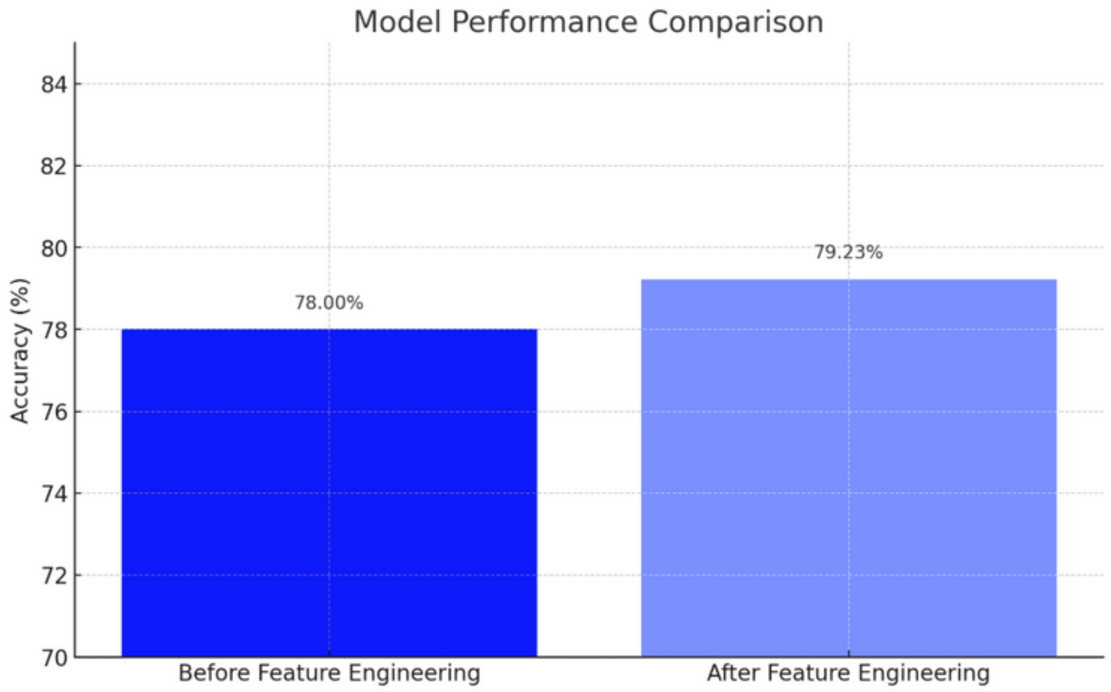
Numerical variables such as 'AGE', 'BILL_AMT', and 'PAY_AMT' were standardized to ensure a uniform scale across the dataset, enhancing model accuracy and consistency.

ID	SEX	EDUCATION	MARRIAGE	AGE	PAY_1	PAY_2	PAY_3	PAY_4	PAY_5	PAY_6	...	BILL_AMT4	BILL_AMT5	BILL_AMT6	PAY_AMT1	PAY_AMT
1	2	2	1	-1.201502	2	2	-1	-1	-2	-2	...	-0.802298	-0.779221	-0.758819	-0.984928	-0.69342
2	2	2	2	-0.988735	-1	2	0	0	0	2	...	-0.714815	-0.681760	-0.665331	-0.984928	-0.56439
3	2	2	2	-0.137670	0	0	0	0	0	0	...	-0.419131	-0.357559	-0.313053	-0.391735	-0.35695
4	2	2	1	0.181480	0	0	0	0	0	0	...	-0.045269	0.037671	0.088249	-0.203383	-0.14163
6	1	1	2	0.181480	0	0	0	0	0	0	...	-0.283762	-0.225797	-0.184761	-0.007996	-0.22626
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	.
29992	1	2	1	-0.137670	3	2	2	2	2	2	...	-0.735455	-0.708700	-0.687148	-0.984928	-0.97928
29993	1	3	1	0.819779	0	0	0	-2	-2	-2	...	-0.802298	-0.779221	-0.758819	-0.203383	-0.97928
29995	1	2	2	-0.137670	2	2	2	2	2	2	...	1.270320	1.551005	1.567860	1.750482	0.47281
29997	1	3	2	0.819779	-1	-1	-1	-1	0	0	...	-0.562227	-0.632819	-0.758819	-0.267079	0.48360
30000	1	2	1	1.138929	0	0	0	0	0	0	...	0.174535	0.135527	-0.319818	-0.172902	-0.23249

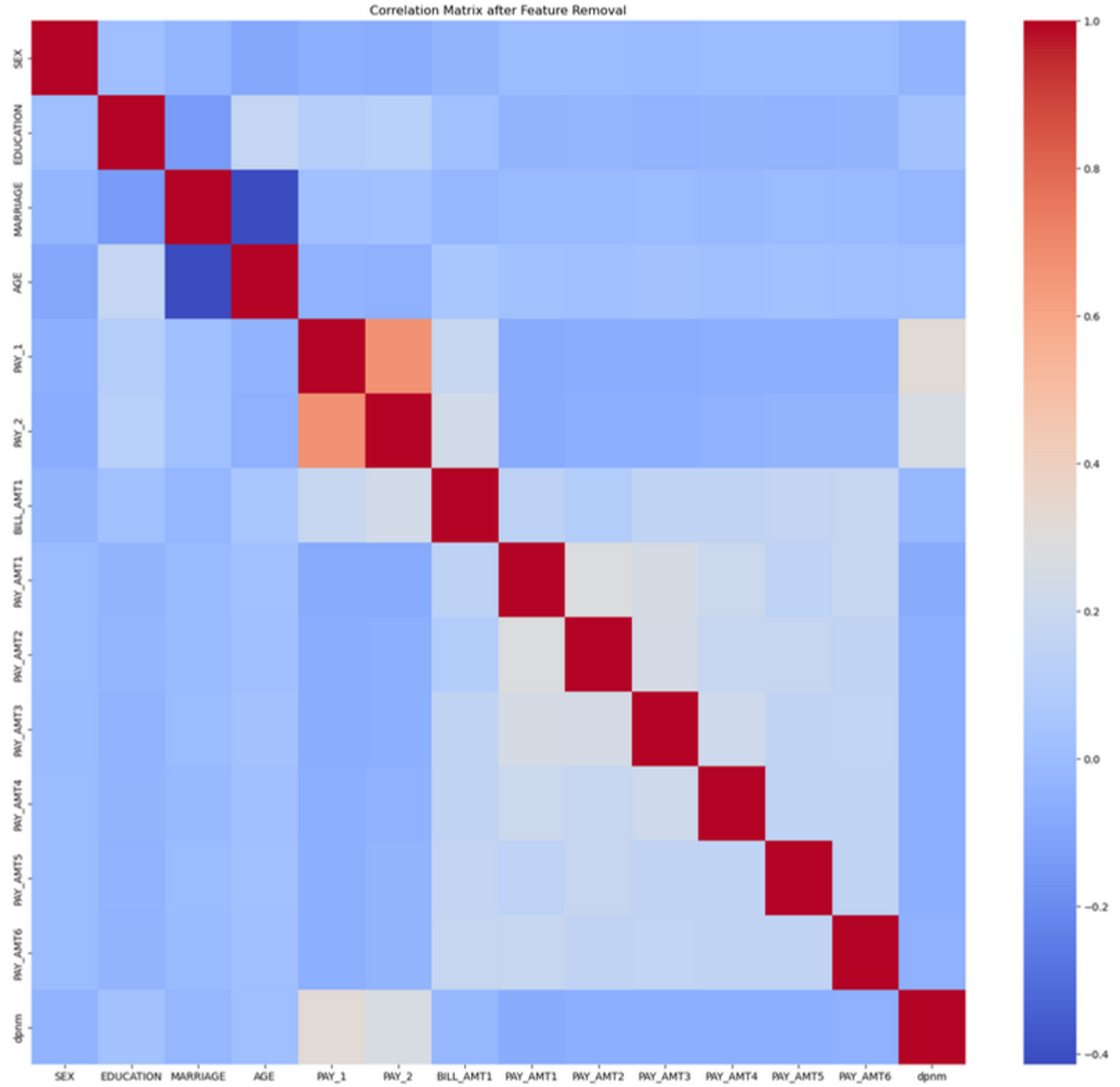
21592 rows × 23 columns

Results of Feature Engineering

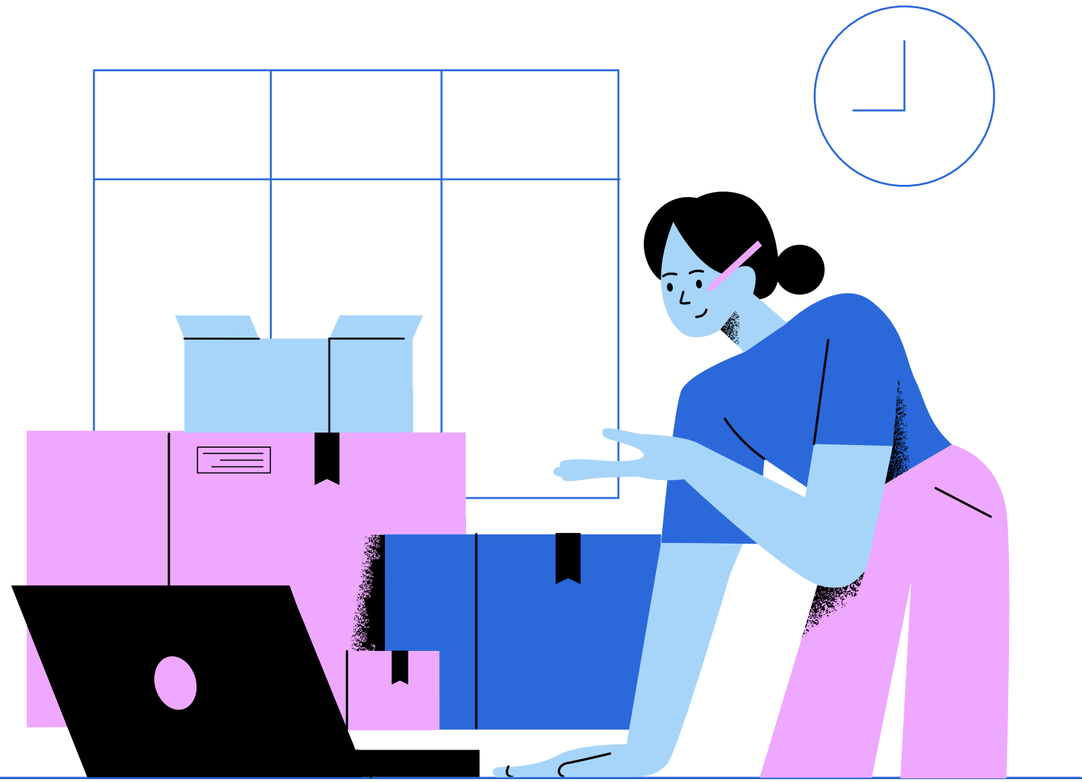
Model accuracy after feature engineering and a correlation matrix with reduced features.

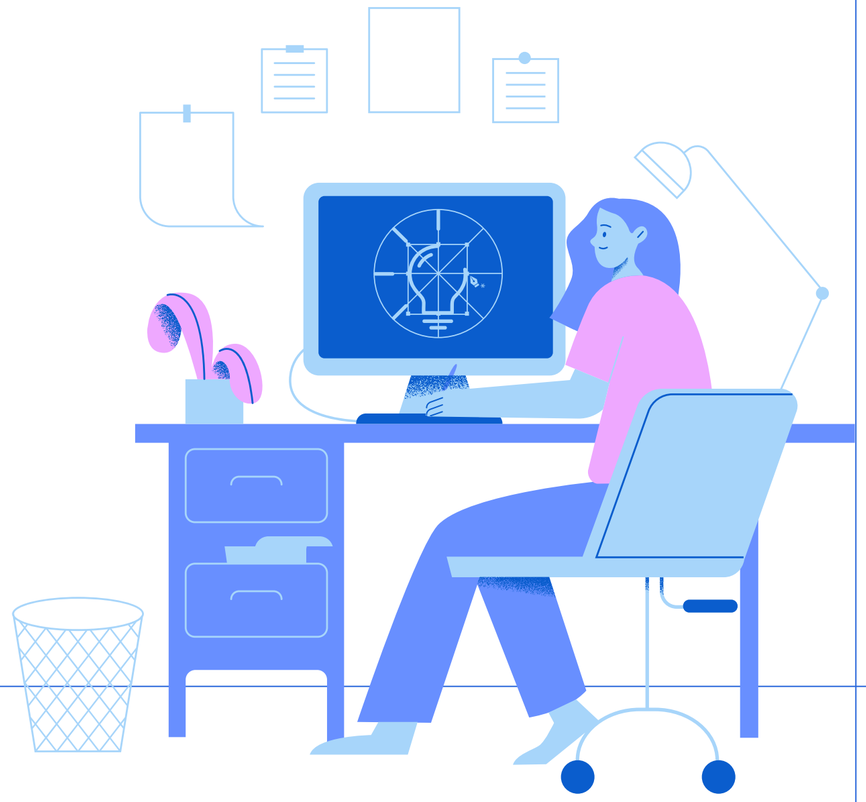


Reduced Features



Model Development Process





Objective

Our primary goal is to develop a predictive model for credit risk management, assisting in the identification of customers at a higher risk of default. This model aims to provide actionable insights for marketing strategies and risk mitigation.

Target Audience

Our target audience includes stakeholders in the financial industry, particularly those involved in risk management, marketing, and customer relations.

Data Preparation

- Feature Selection: We selected relevant features such as demographic information, repayment history, and billing details, excluding the identifier ('ID') from our analysis.
- Data Split: The dataset was divided into training (80%) and test (20%) sets to train and evaluate the model.

Logistic Regression Model

- Model Selection: Logistic regression was chosen due to its interpretability and suitability for binary classification tasks.
- Training: The model was trained on the training set using the logistic regression algorithm.

Model Evaluation

- Confusion Matrix: The model's performance was assessed using a confusion matrix, providing insights into true positives, true negatives, false positives, and false negatives.

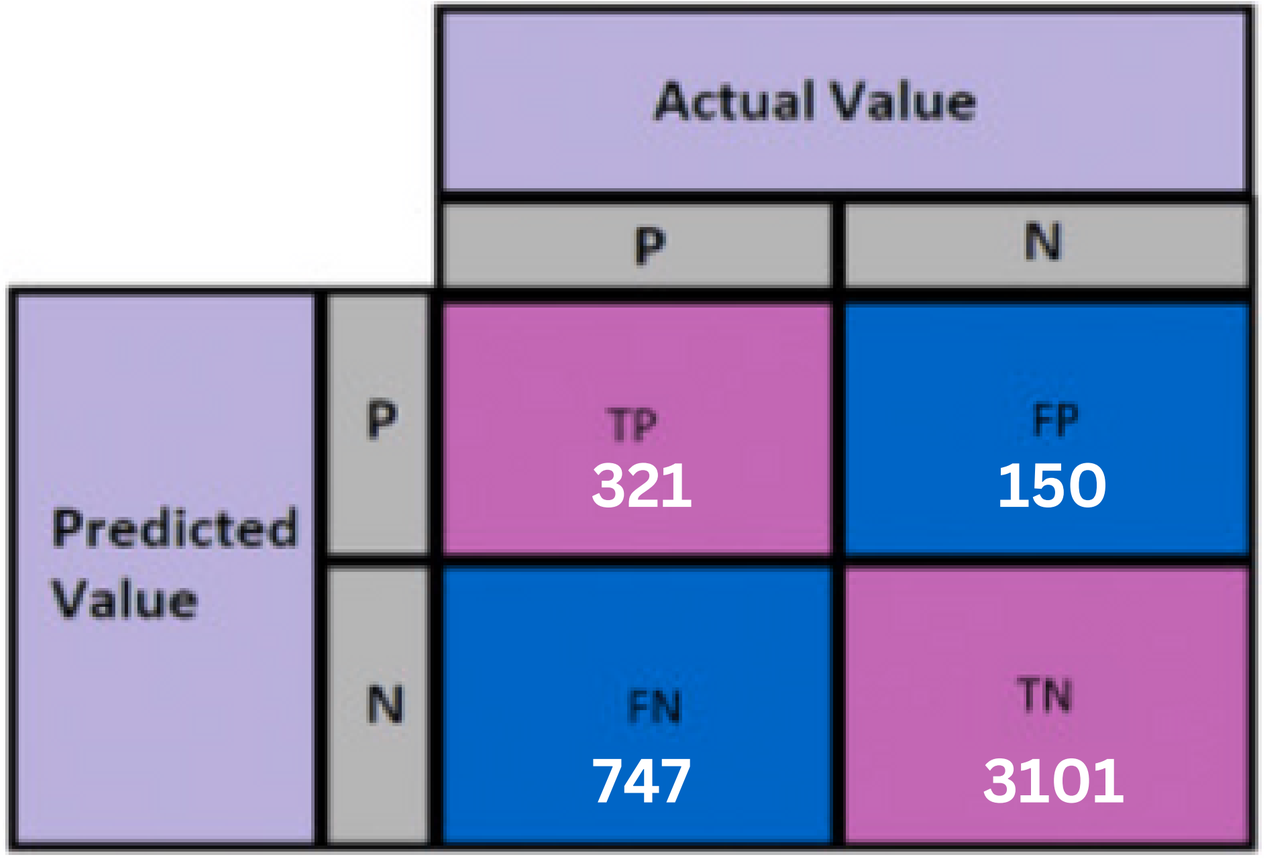
Model Performance Analysis

Confusion Matrix

TRUE POSITIVE	A person who is defaulter and predicted as defaulter.
TRUE NEGATIVE	A person who is non-defaulter and predicted as non-defaulter.
FALSE POSITIVE	A person who is predicted defaulter is non-defaulter.
FALSE NEGATIVE	A person who is predicted non-defaulter is defaulter.

#	Non-defaulter (predicted) - 0	Defaulter (predicted) - 1
Non-defaulter (actual) - 0	TN	FP
Defaulter (actual) - 1	**FN**	TP

Confusion Matrix Overview



Confusion Matrix

TP - True Positive
TN - True Negative
FP - False Positive
FN - False Negative
P - Positive
N - Negative

True Negatives (TN)	3101
False Positives (FP)	150
False Negatives (FN)	747
True Positives (TP)	321

Model Performance Metrics

NULL MODEL 75.7%

THE MODEL OUTPERFORMS THE NULL MODEL, WHICH PREDICTS NO DEFAULTS, BY 3.5%.

PREDICTIVE POWER 79.23%

THE MODEL'S ACCURACY IS BETTER THAN THE NULL MODEL, INDICATING ITS UTILITY IN IDENTIFYING DEFAULT RISK.

Model Performance Metrics

Precision: 68.15%

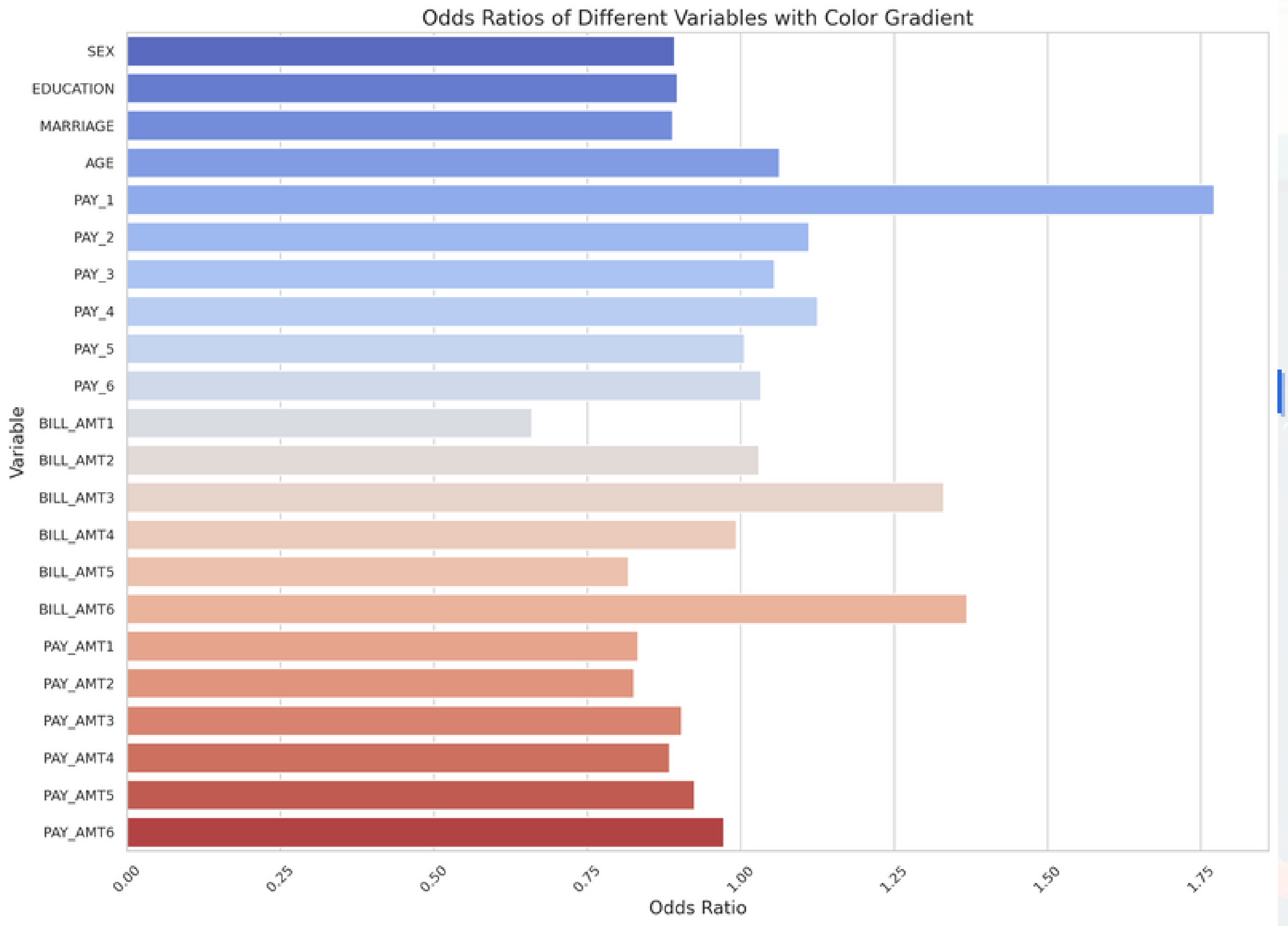
When the model predicts a default, it is correct about two-thirds of the time.

Sensitivity 30.06%

The model's ability to capture actual defaults is moderate, suggesting room for improvement in identifying high-risk customers.

Specificity 95.39%

The model is highly effective at identifying customers who will not default, ensuring a low rate of false alarms.



Interpretation Of Odds Ratios

Variables related to payment status (PAY_1 to PAY_6) generally have odds ratios greater than 1, with PAY_1 showing the highest odds ratio, indicating a strong association with the outcome.

Most bill amount variables (BILL_AMT1 to BILL_AMT6) have odds ratios close to 1, except for BILL_AMT1 which is notably lower, and BILL_AMT6 which is higher.

Payment amount variables (PAY_AMT1 to PAY_AMT6) all have odds ratios less than 1, suggesting that higher payments are associated with lower odds of the outcome

SEX, EDUCATION, MARRIAGE: These demographic factors have a slight impact on default likelihood.

AGE: Older customers may have a slightly higher risk of default, requiring attention to product suitability for different age groups.

PAY_X: Repayment statuses strongly influence default risk, highlighting the importance of timely payments.

BILL_AMT_X and PAY_AMT_X: Billing and payment amounts in various months impact default likelihood, informing strategies for financial planning services.

Marketing Strategies

Targeted Campaigns

Tailor messages to specific customer segments identified in the model

Personalized Financial Planning Services

Offer personalized financial planning services to address individual financial needs

Repayment Assistance Program

Implement programs for customers with delayed payments
Offer support and flexible repayment options

Proactive Customer Engagement

Solicit feedback to refine marketing strategies and improve satisfaction

Customer Education Initiatives

Launch campaigns for credit management awareness.

Tailored Marketing for Low-Risk Segments

Direct marketing efforts to low-risk segments

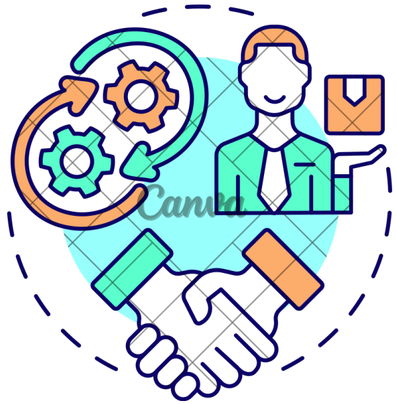
Create targeted promotions, loyalty programs, and exclusive offers.

Incentive Programs

Providing personalized incentives, and loyalty rewards to existing customers who pay their credit card bills on time.

Enhanced Fraud Detection Measures

Implement real-time alerts and additional security measures



Continuous Adaption

Monitor and adjust strategies based on changing trends

Partnerships with Financial Wellness Apps

Explore partnerships for budgeting and financial goal-setting

Conclusion

- Model Improvement: Future Iteration could be explored based on different variable
- Continuous Monitoring: Regularly updating models with new data to maintain its accuracy
- Cross Functional Integration: Collaborate with different departments to implement findings effectively across the business

APPENDIX

#ITERATION 1

ONE HOT CODING

&

LIMIT_BAL AS A VARIABLE

ACCURACY < SELECTED MODEL

Calculate and Print Model Performance Metrics:

In [104]:
```
# Calculate the percentage of the sample that survived
dpnm_perc = df['dpnm'].mean() * 100


# Print the results, rounding to 2 decimal places
if dpnm_perc > 49.9:
    null = round(dpnm_perc, 2)
else:
    null = round(100 - dpnm_perc, 2)

print(f"Null Model: {null}%")
```

```
Null Model: 75.7%
```

Null Model (75.7%): This baseline indicates that without any model, if we predicted that no customers would default, we would be correct 75.7% of the time due to the data's imbalance. Any predictive model must perform better than this baseline to be considered effective.

In [105]:
```
pred_power = round(((TN + TP) / (TN + TP + FN + FP))*100, 2)

print(f"Predictive Power: {pred_power}%")
```

```
Predictive Power: 79.16%
```

Predictive Power (79.23%):

In [106]:
```
precision = round(((TP) / (TP + FP))*100, 2)

print(f"Precision: {precision}%")
```

```
Precision: 64.98%
```

Precision (68.15%): This tells us that when the model predicts a default, it is correct roughly two-thirds of the time. For marketing, this means that interventions based on the model's predictions have a high likelihood of being justified but should still be approached with caution due to the possibility of false positives.

In [107]:
```
sensitivity = round(((TP) / (TP + FN))*100, 2)

print(f"Sensitivity: {sensitivity}%")
```

```
Sensitivity: 28.47%
```

Sensitivity (30.06%): The low sensitivity suggests the model fails to capture many actual defaults. From a marketing perspective, this means there is a risk of missing out on opportunities to engage with customers who might need financial assistance or debt restructuring.

In [108]:
```
specificity = round(((TN) / (FP + TN))*100, 2)

print(f"Specificity: {specificity}%")
```

```
Specificity: 95.16%
```

#ITERATION 2

New Variable ADDED -
'Total_Bill' & 'Total_Payed'

ACCURACY < SELECTED MODEL

Calculate and Print Model Performance Metrics:

In [141]:
```
# Calculate the percentage of the sample that survived
dpnm_perc = df['dpnm'].mean() * 100


# Print the results, rounding to 2 decimal places
if dpnm_perc > 49.9:
    null = round(dpnm_perc, 2)
else:
    null = round(100 - dpnm_perc, 2)

print(f"Null Model: {null}%")
```

```
Null Model: 75.7%
```

Null Model (75.7%): This baseline indicates that without any model, if we predicted that no customers would default, we would be correct 75.7% of the time due to the data's imbalance. Any predictive model must perform better than this baseline to be considered effective.

In [142]:
```
pred_power = round(((TN + TP) / (TN + TP + FN + FP))*100, 2)

print(f"Predictive Power: {pred_power}%")
```

```
Predictive Power: 79.18%
```

Predictive Power (79.23%):

In [106]:
```
precision = round(((TP) / (TP + FP))*100, 2)

print(f"Precision: {precision}%")
```

```
Precision: 64.98%
```

Precision (68.15%): This tells us that when the model predicts a default, it is correct roughly two-thirds of the time. For marketing, this means that interventions based on the model's predictions have a high likelihood of being justified but should still be approached with caution due to the possibility of false positives.

In [107]:
```
sensitivity = round(((TP) / (TP + FN))*100, 2)

print(f"Sensitivity: {sensitivity}%")
```

```
Sensitivity: 28.47%
```

Sensitivity (30.06%): The low sensitivity suggests the model fails to capture many actual defaults. From a marketing perspective, this means there is a risk of missing out on opportunities to engage with customers who might need financial assistance or debt restructuring.

In [108]:
```
specificity = round(((TN) / (FP + TN))*100, 2)

print(f"Specificity: {specificity}%")
```

```
Specificity: 95.16%
```

Specificity (95.39%): The model is very good at identifying customers who will not default. This can help in maintaining customer trust and ensuring that non-risk customers are not subjected to unnecessary credit scrutiny.

#ITERATION 3

New Variable ADDED -
Avg Total bills and
Avg total amount

ACCURACY < SELECTED MODEL

Calculate and Print Model Performance Metrics:

In [104]:
```
# Calculate the percentage of the sample that survived
dpnm_perc = df['dpnm'].mean() * 100


# Print the results, rounding to 2 decimal places
if dpnm_perc > 49.9:
    null = round(dpnm_perc, 2)
else:
    null = round(100 - dpnm_perc, 2)

print(f"Null Model: {null}%")
```

```
Null Model: 75.7%
```

Null Model (75.7%):

In [105]:
```
pred_power = round(((TN + TP) / (TN + TP + FN + FP))*100, 2)

print(f"Predictive Power: {pred_power}%")
```

```
Predictive Power: 79.16%
```

Predictive Power (79.23%): The model's accuracy is about 3.5% better than the null model, which might seem modest. However, in a large customer base, this improvement can translate into significant financial impact, indicating the model's utility in identifying default risk.

In [106]:
```
precision = round(((TP) / (TP + FP))*100, 2)

print(f"Precision: {precision}%")
```

```
Precision: 64.98%
```

Precision (68.15%): This tells us that when the model predicts a default, it is correct roughly two-thirds of the time. For marketing, this means that interventions based on the model's predictions have a high likelihood of being justified but should still be approached with caution due to the possibility of false positives.

In [107]:
```
sensitivity = round(((TP) / (TP + FN))*100, 2)

print(f"Sensitivity: {sensitivity}%")
```

```
Sensitivity: 28.47%
```

Sensitivity (30.06%): The low sensitivity suggests the model fails to capture many actual defaults. From a marketing perspective, this means there is a risk of missing out on opportunities to engage with customers who might need financial assistance or debt restructuring.

In [108]:
```
specificity = round(((TN) / (FP + TN))*100, 2)

print(f"Specificity: {specificity}%")
```

```
Specificity: 95.16%
```

Thank you

Open for Feedback!