Credit default analysis

Nguyen Tuyet Ngan

Table of Contents

Import data

ANALYZE THE CREDIT DEFAULT IN TAIWAN FROM APRIL 2005 TO SEPTEMBER 2005

```
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   LIMIT_BAL  24000 non-null  int64
 1   SEX        24000 non-null  int64
 2   EDUCATION  24000 non-null  int64
 3   MARRIAGE   24000 non-null  int64
 4   AGE        24000 non-null  int64
 5   PAY_1      24000 non-null  int64
 6   PAY_2      24000 non-null  int64
 7   PAY_3      24000 non-null  int64
 8   PAY_4      24000 non-null  int64
 9   PAY_5      24000 non-null  int64
 10  PAY_6      24000 non-null  int64
 11  BILL_AMT1  24000 non-null  float64
 12  BILL_AMT2  24000 non-null  float64
 13  BILL_AMT3  24000 non-null  float64
 14  BILL_AMT4  24000 non-null  float64
 15  BILL_AMT5  24000 non-null  float64
 16  BILL_AMT6  24000 non-null  float64
 17  PAY_AMT1   24000 non-null  float64
 18  PAY_AMT2   24000 non-null  float64
 19  PAY_AMT3   24000 non-null  float64
 20  PAY_AMT4   24000 non-null  float64
 21  PAY_AMT5   24000 non-null  float64
 22  PAY_AMT6   24000 non-null  float64
 23  default_0  24000 non-null  int64
dtypes: float64(12), int64(12)
```

- LIMIT_BAL: Credit limit
- SEX: Gender
- EDUCATION: Education level
- MARRIAGE: Marital status
- AGE: age in years
- PAY_x: Repayment status (x: 0-6 ~ Sep - Apr)
- BILL_AMTx: Bill amount (x: 0-6 ~ Sep - Apr)
- PAY_AMTx: Payment amount (x: 0-6 ~ Sep - Apr)
- default_0: Default next month (1= defalut; 0= not default)

Check null value

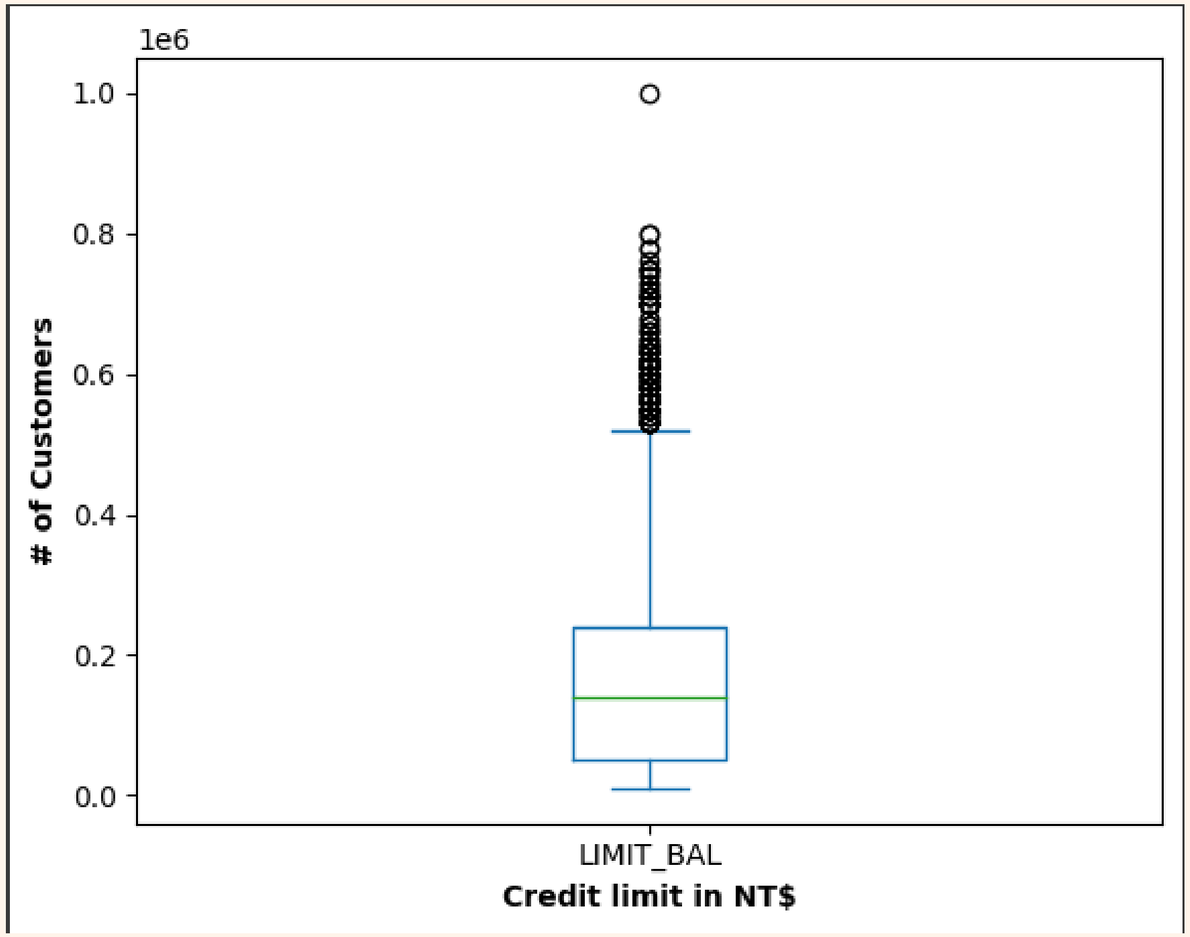
LIMIT_BAL	0
SEX	0
EDUCATION	0
MARRIAGE	0
AGE	0
PAY_1	0
PAY_2	0
PAY_3	0
PAY_4	0
PAY_5	0
PAY_6	0
BILL_AMT1	0
BILL_AMT2	0
BILL_AMT3	0
BILL_AMT4	0
BILL_AMT5	0
BILL_AMT6	0
PAY_AMT1	0
PAY_AMT2	0
PAY_AMT3	0
PAY_AMT4	0
PAY_AMT5	0
PAY_AMT6	0
default_0	0

Data cleaning

ID	
1	False
2	False
3	False
4	False
5	False
	...
23996	False
23997	False
23998	False
23999	False
24000	False

Check duplicated value

Check outliers

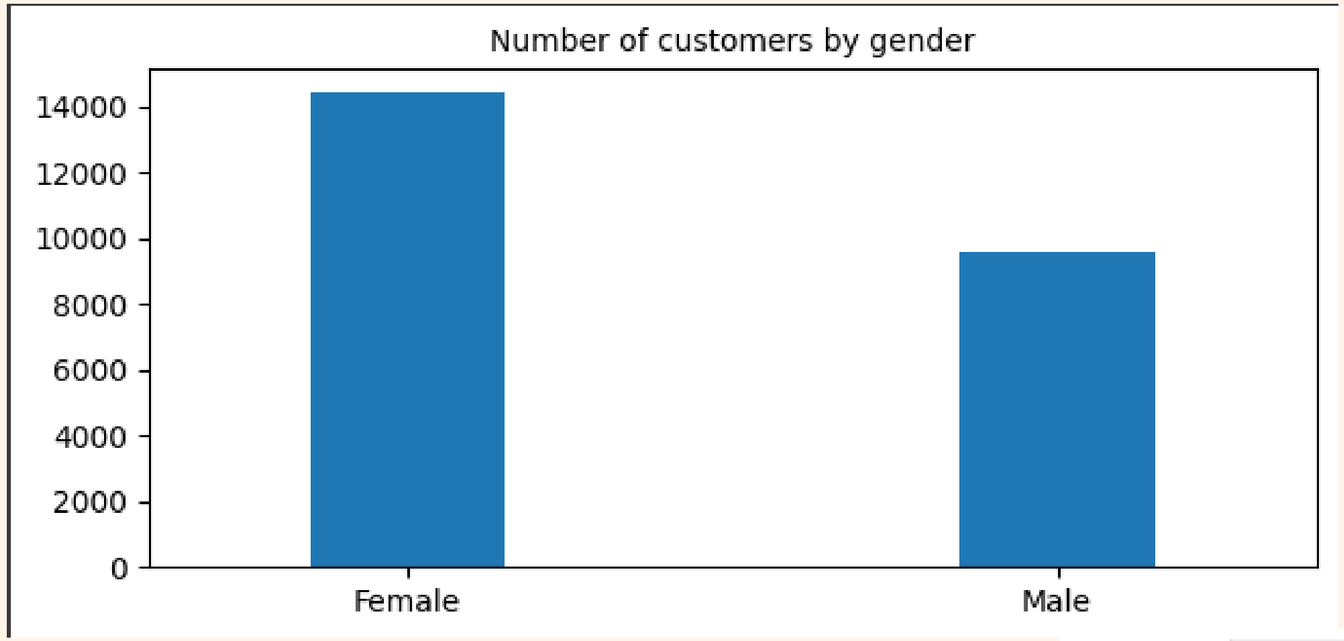


ID	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_1	PAY_2	PAY_3
1769	1000000	2	1	1	47	0	0	0

PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4	PAY_AMT5	PAY_AMT6	default_0
50784.0	50723.0	896040.0	50000.0	50000.0	50256.0	0

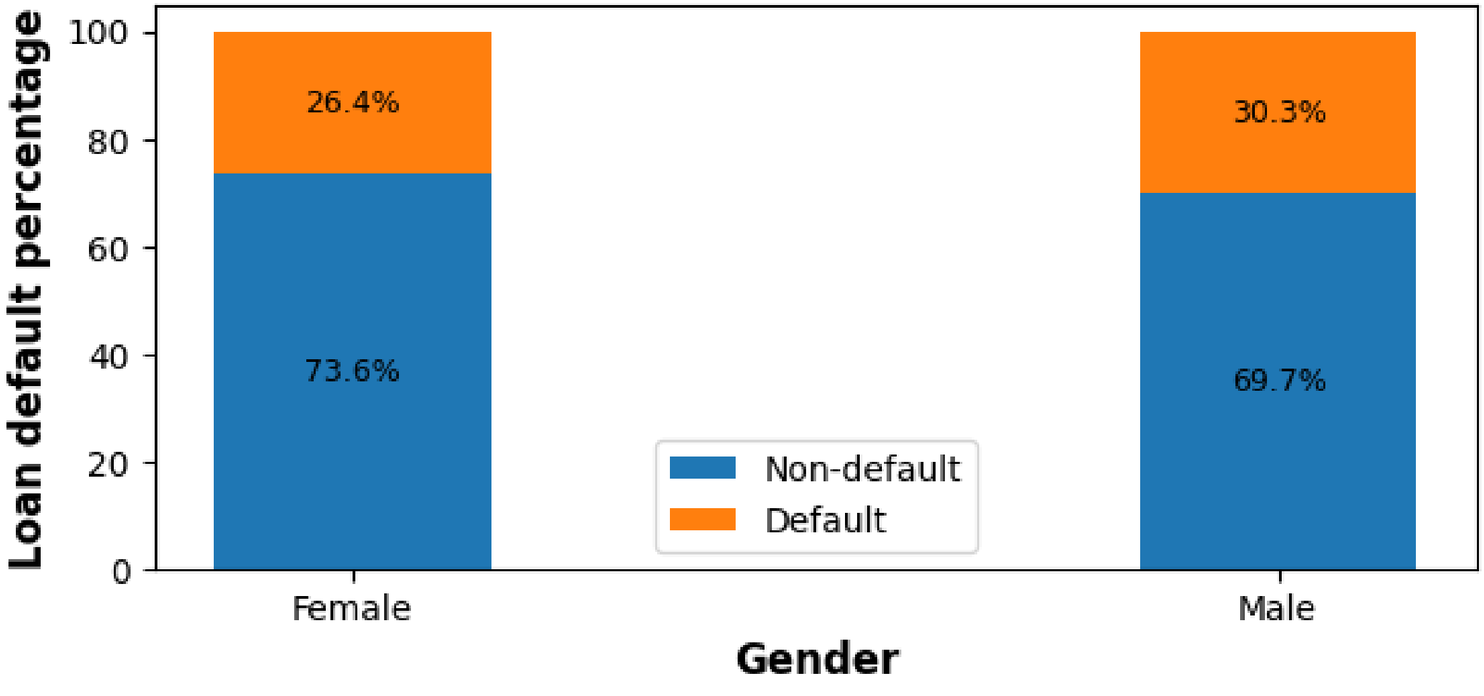
1 outlier: customer who has 100000 limit balance and has an excellent payment history => Valid data

Default by gender

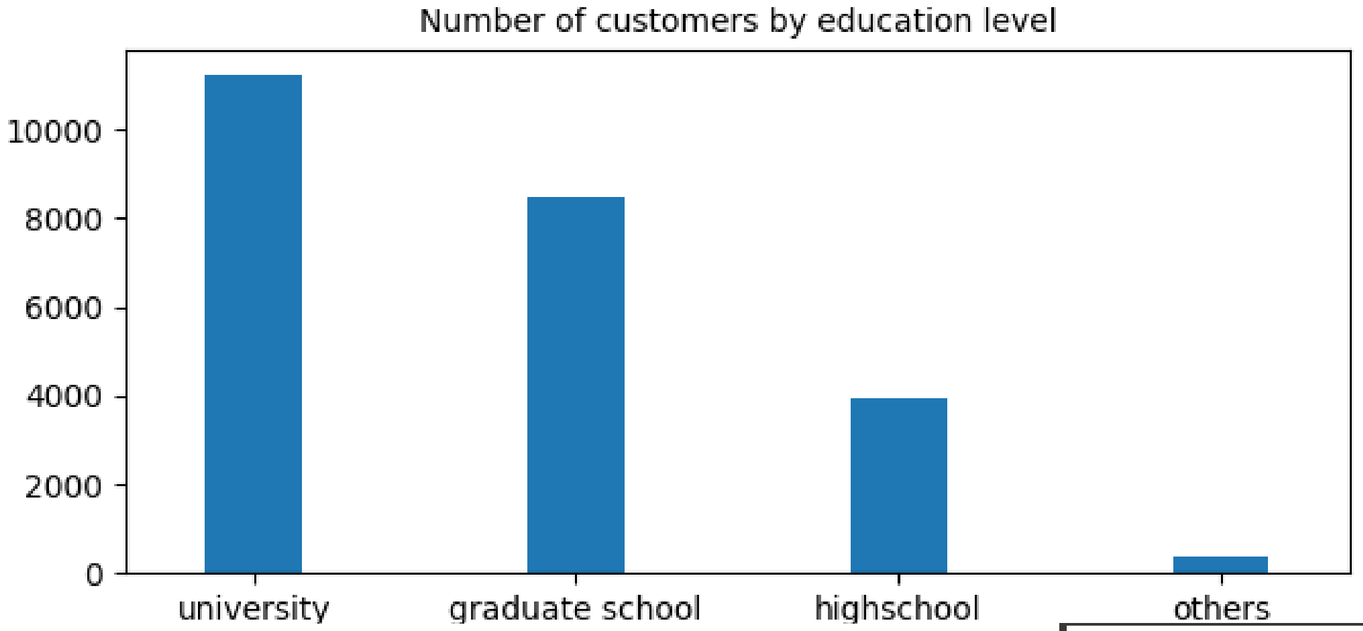


Number of customers by gender

Loan default by gender

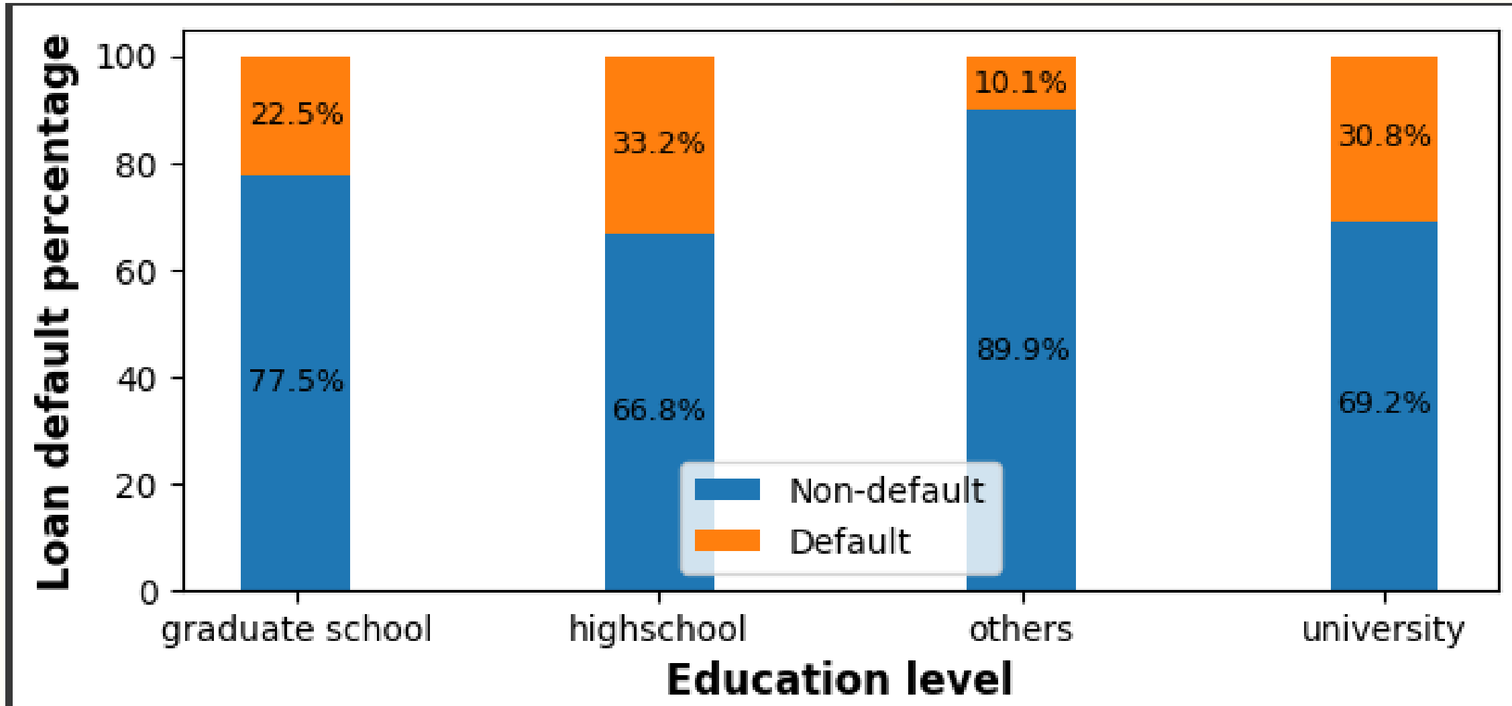


Default by education level

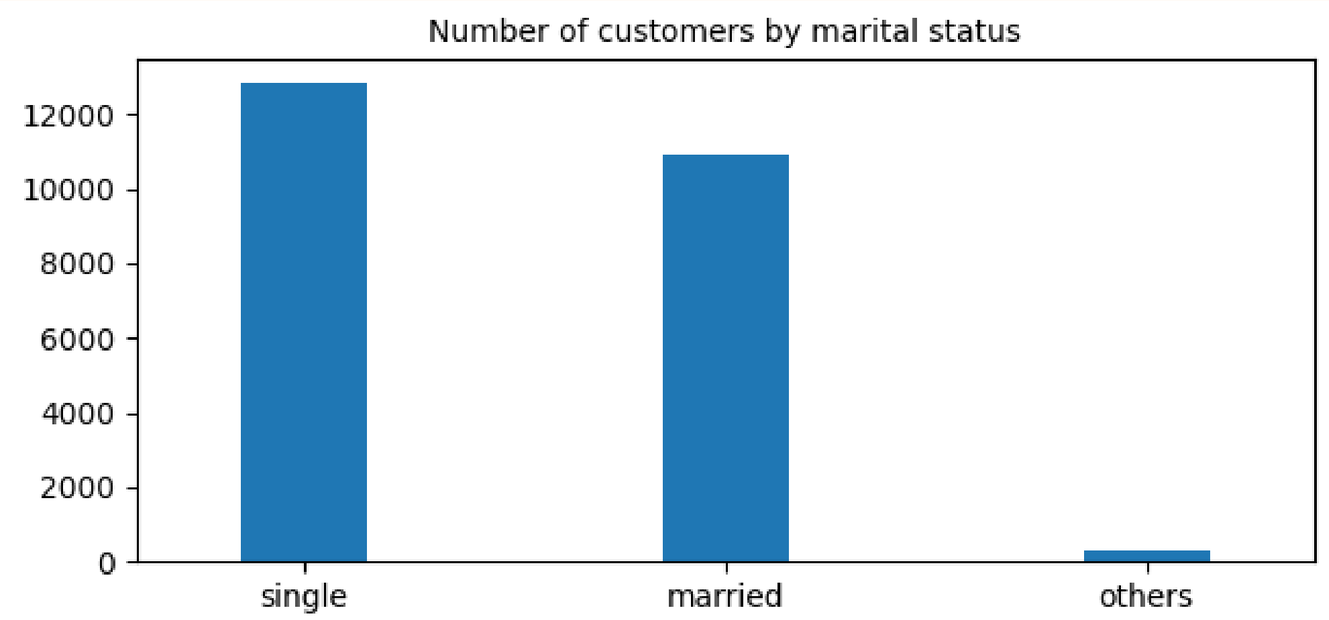


Number of customers by education level

Loan default by education level

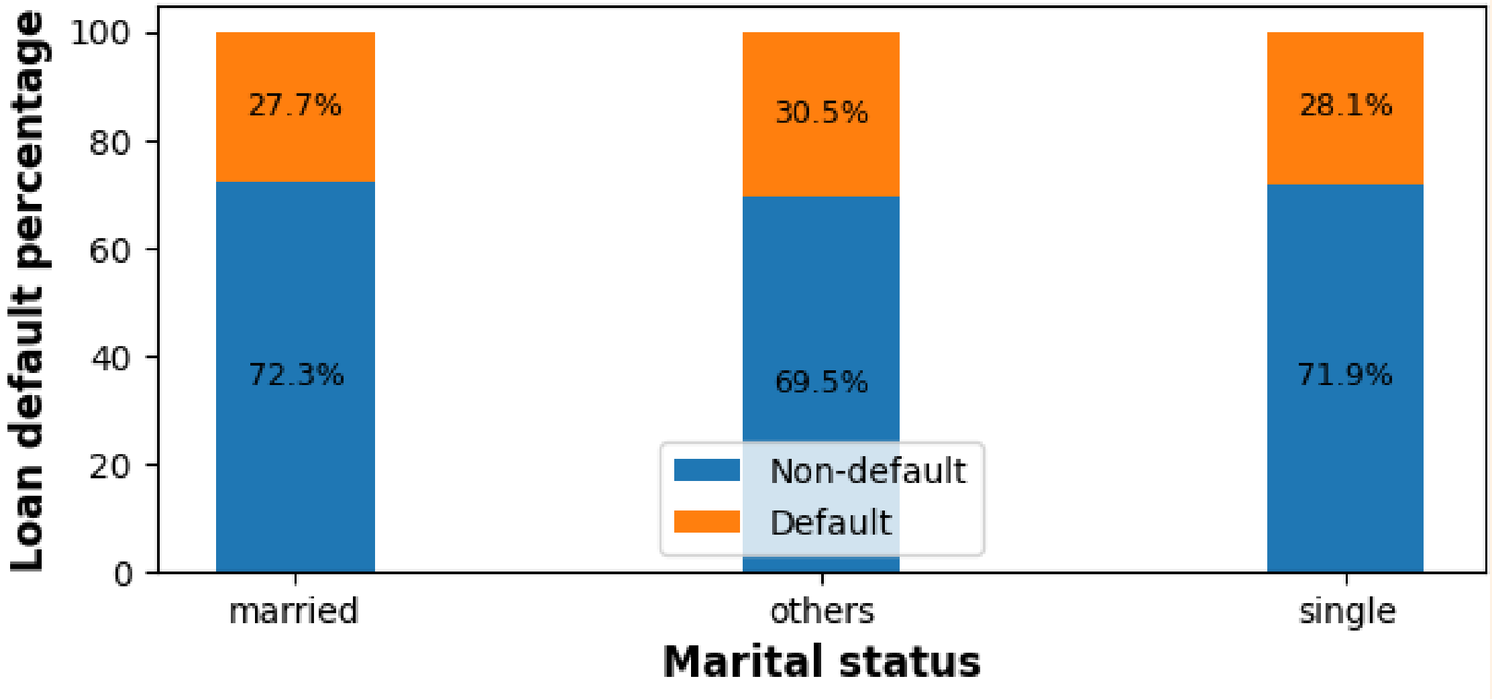


Default by marital status

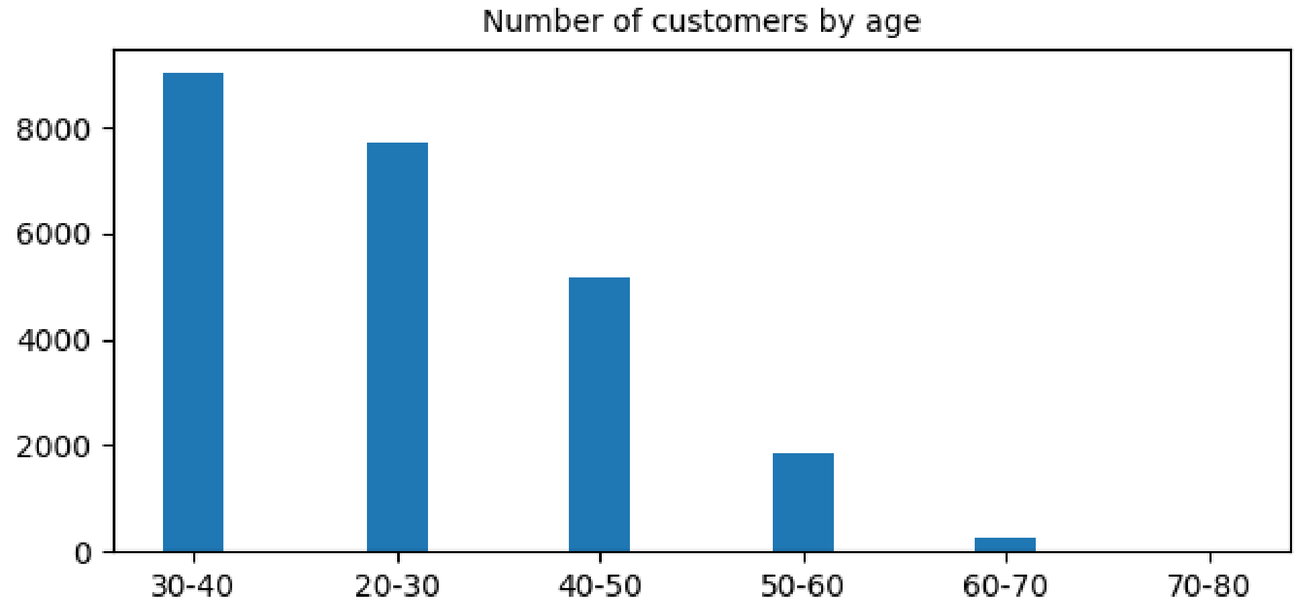


Number of customers by marital status

Loan default by marital status

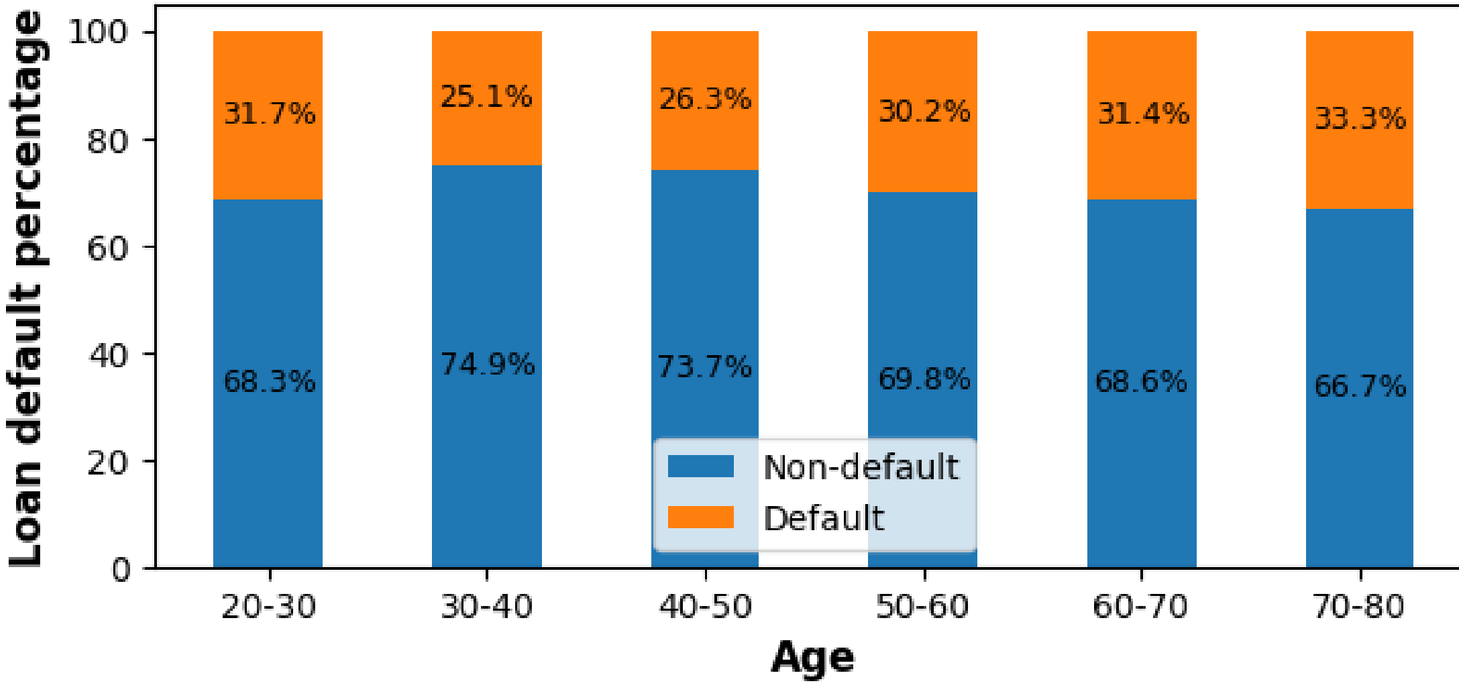


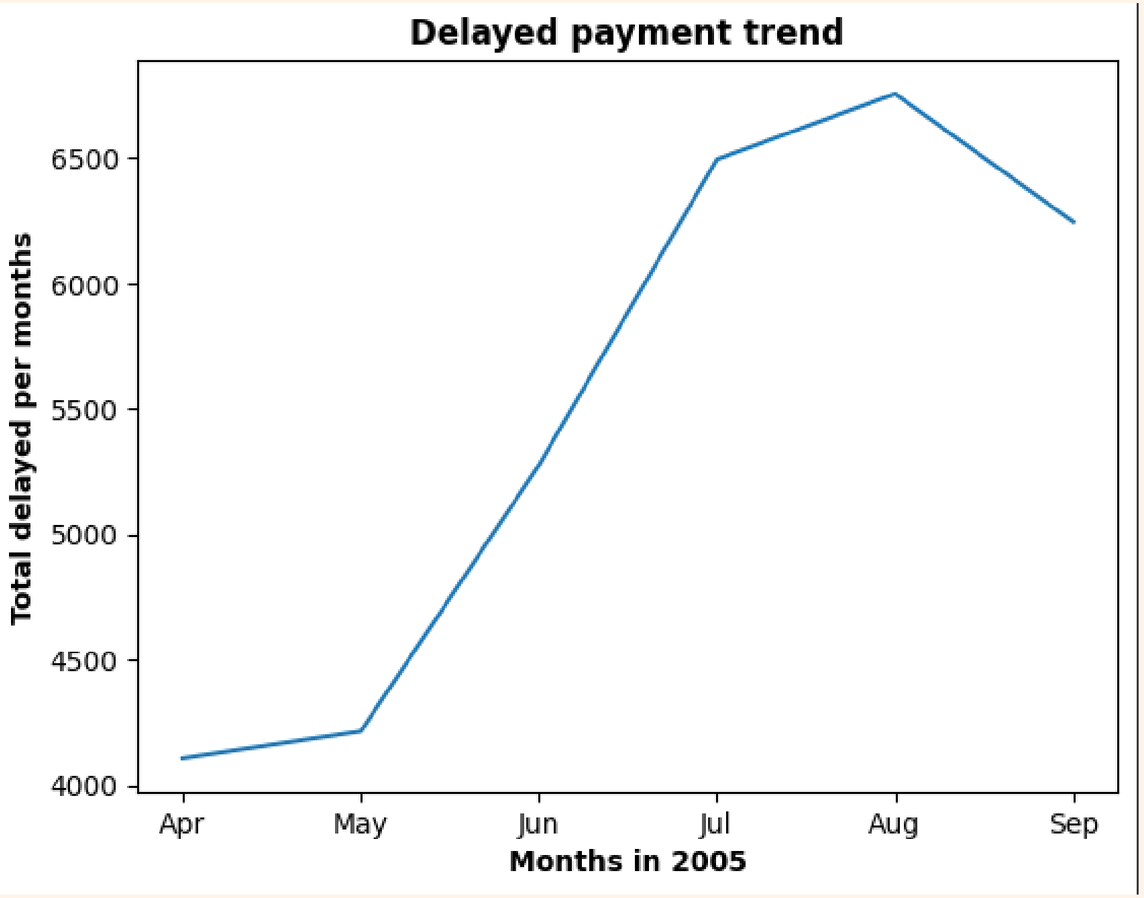
Default by age

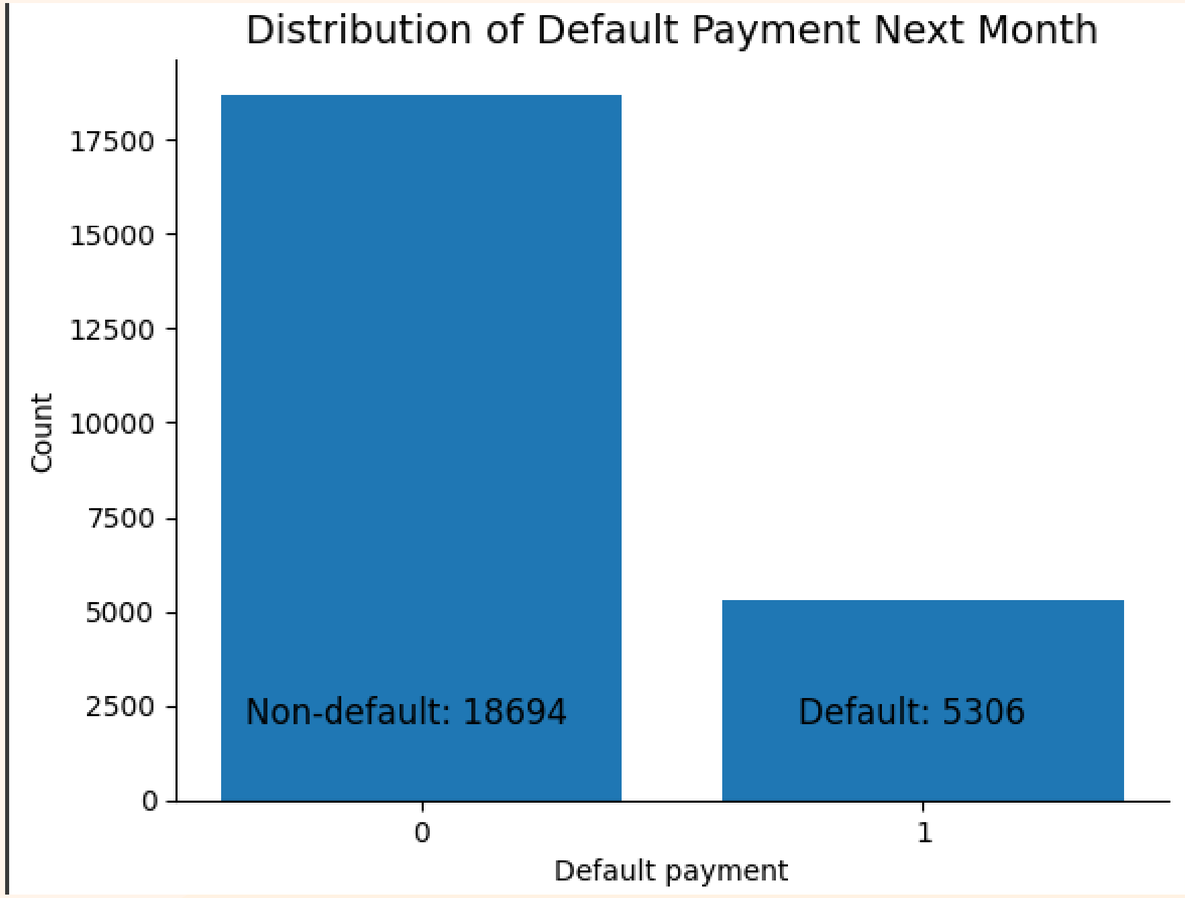


Number of customers by age

Loan default by age







Delay payment trend

Number of default next month

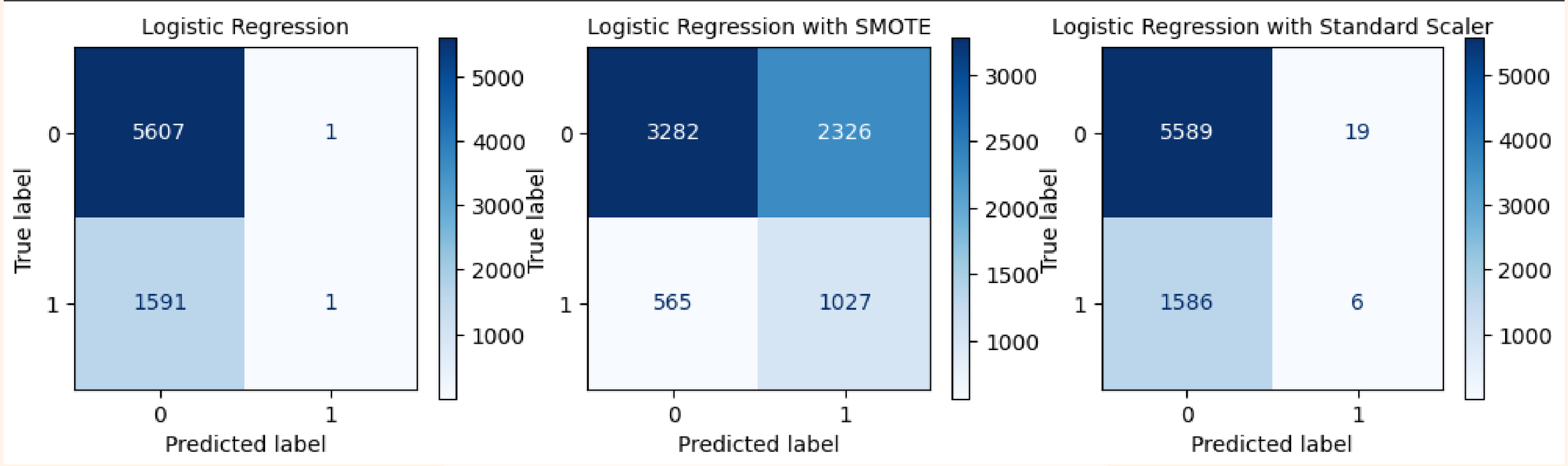
Prepare for modeling

```
# Define predictor variables and target variable
y = data['default_0']
X = data.drop(columns=['default_0'])
```

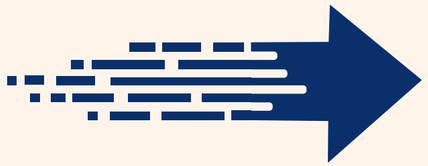
```
#Split data
X_train, X_test, y_train, y_test =
train_test_split(X, y, test_size=0.3,shuffle=True, stratify=y, random_state=42)
```

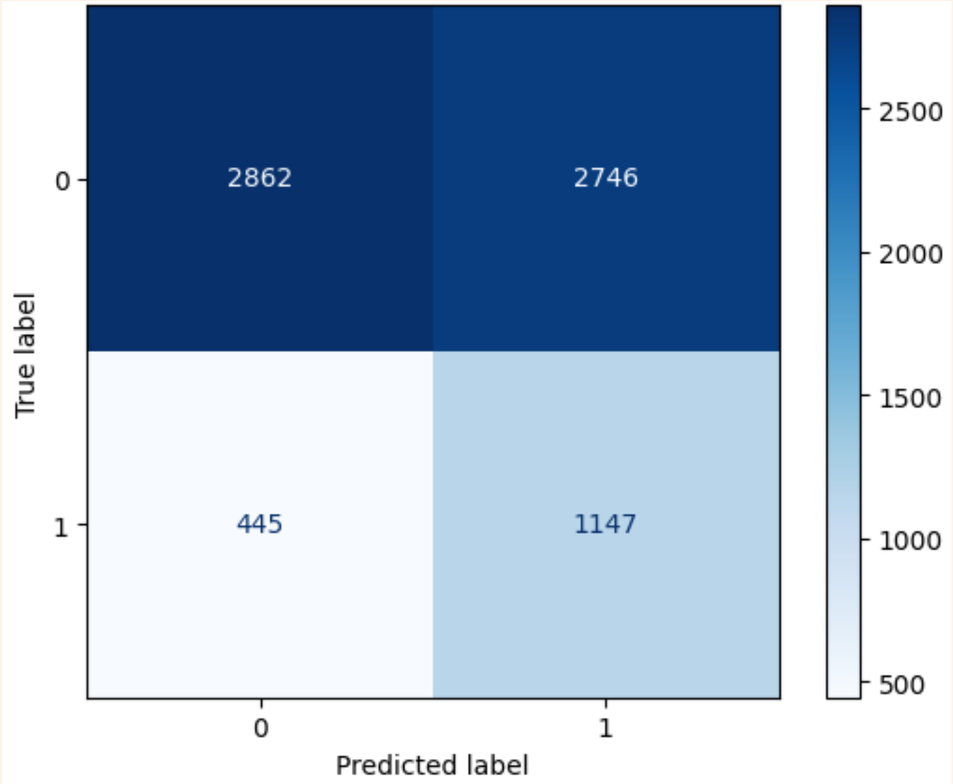
```
# use SMOTE to over sample data
from imblearn.over_sampling import SMOTE
sm = SMOTE(random_state = 1)
X_train_sm, y_train_sm = sm.fit_resample(X_train, y_train)
```

```
#Split data with Standard Scaler
from sklearn.preprocessing import StandardScaler
X_train_std = StandardScaler().fit_transform(X_train)
```

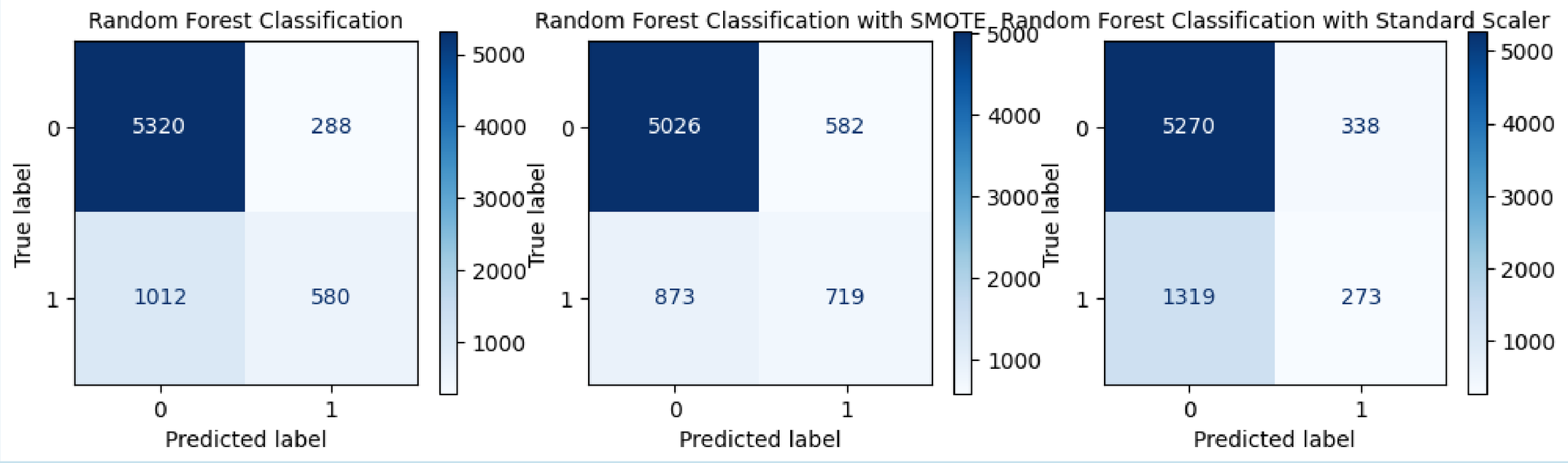


LOGISTIC REGRESSION

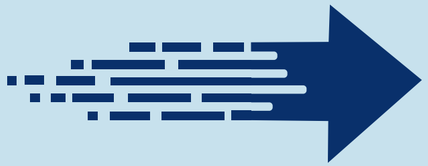


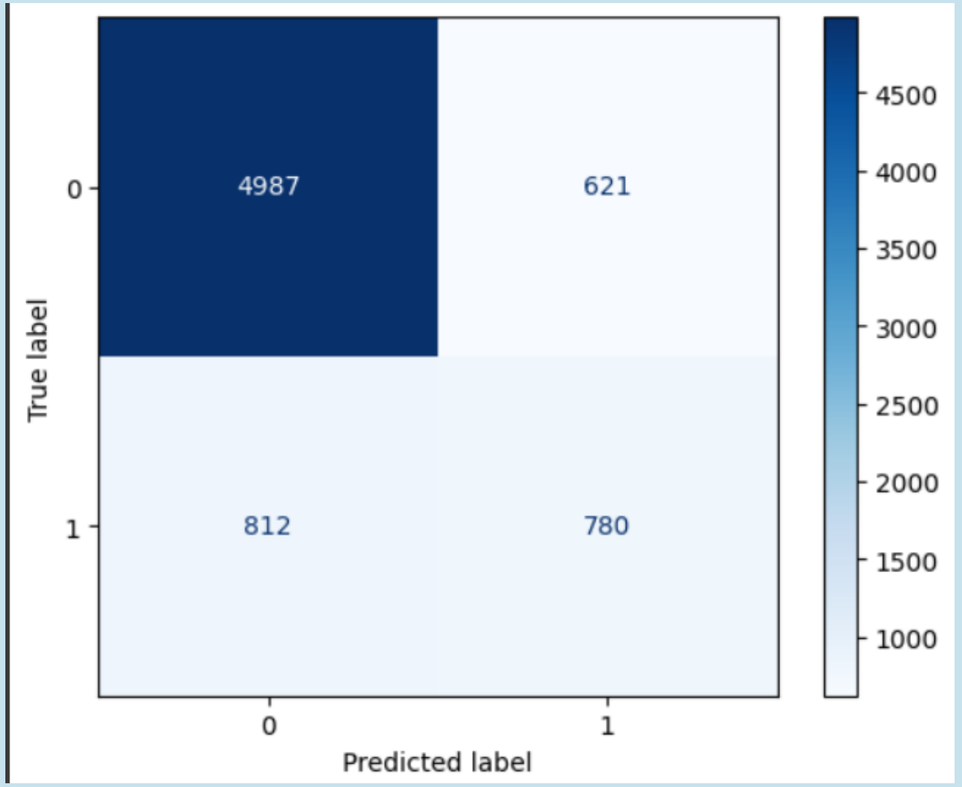


Best model parameter:
{'max_iter': 200,
'penalty': 'l2',
'tol': 0.01}

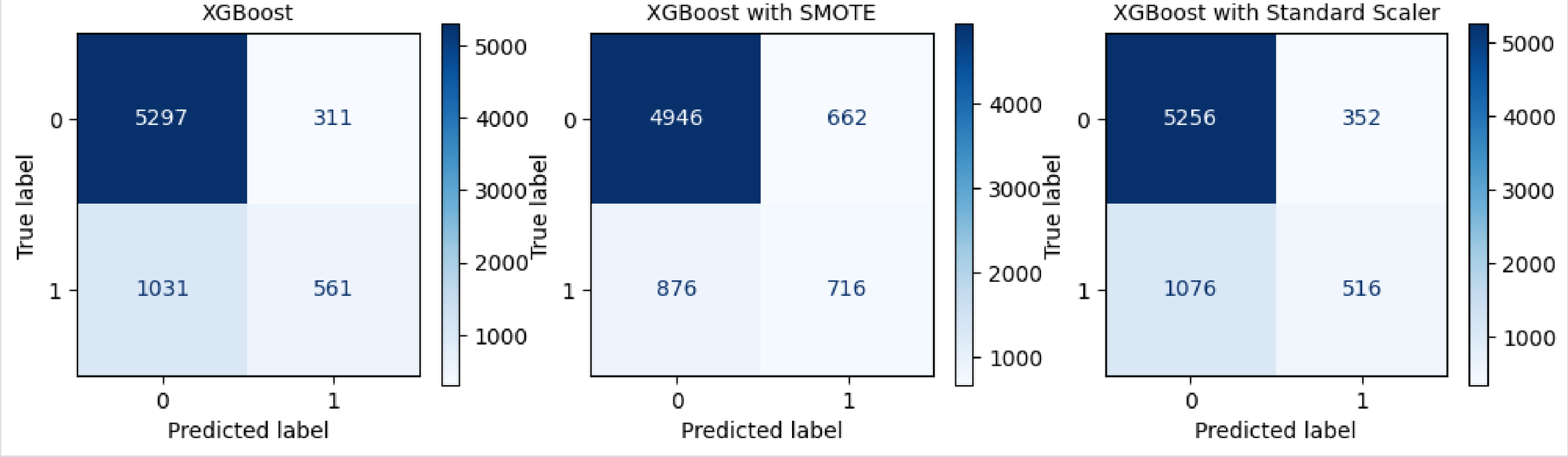


RANDOM FOREST CLASSIFICATION

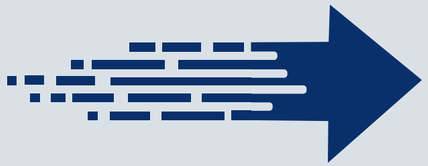


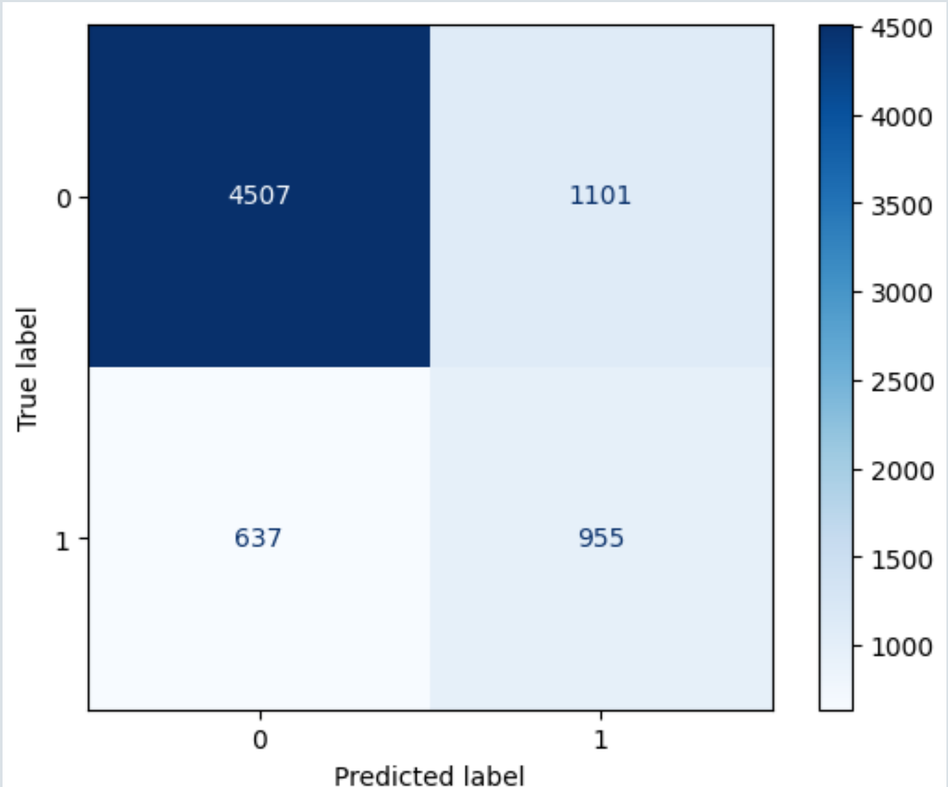


Best model parameter:
{'max_depth': 18,
'min_samples_leaf': 3,
'min_samples_split': 8,
'n_estimators': 200}



XGBOOST





Best model parameter:
{'max_depth': 18,
'max_leaf_nodes': 4,
'n_estimators': 300}

COMPARE 3 BEST MODELS

	precision	recall	f1-score	support
0	0.87	0.51	0.64	5608
1	0.29	0.72	0.42	1592
accuracy			0.56	7200
macro avg	0.58	0.62	0.53	7200
weighted avg	0.74	0.56	0.59	7200

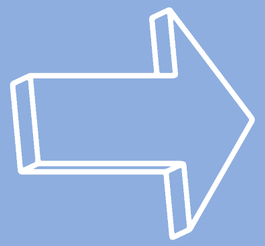
Logistic Regression

	precision	recall	f1-score	support
0	0.86	0.89	0.87	5608
1	0.56	0.49	0.52	1592
accuracy			0.80	7200
macro avg	0.71	0.69	0.70	7200
weighted avg	0.79	0.80	0.80	7200

Random Forest Classification

	precision	recall	f1-score	support
0	0.88	0.80	0.84	5608
1	0.46	0.60	0.52	1592
accuracy			0.76	7200
macro avg	0.67	0.70	0.68	7200
weighted avg	0.79	0.76	0.77	7200

XGBoost



Recommend model: Logistic Regression

{'max_iter': 200,'penalty': 'l2', 'tol': 0.01}

Thank You