



BUAN 6340

Spring 2024


Project Report

Professor: Jason Parker

Student: Shiva Kumar Reddy Koppula

NetID: sxx230064

**Project Title: Forecasting Credit Card Defaults Using Machine
Learning Techniques**



CONTENTS

Abstract	3
Introduction.....	3
Exploratory Data Analysis	3
Dataset Description	3
Data Structure and cleaning.....	6
Categorical Features.....	7
Continuous features.....	8
Correlation	8
Dimensionality reduction	9
Class imbalance - Resampling	10
Classification models	10
Logistic Regression:	10
Decision Tree and Random Forest:	11
Support Vector Machine (SVM):	11
Results	11
Logistic regression	11
Decision Tree.....	12
Random forest.....	12
Support Vector Machine (SVM)	13
Overall overview	13
Conclusion	14

ABSTRACT

Financial institutions need to measure risks within their credit portfolios for regulatory requirements and for internal risk management. To meet these requirements financial institutions increasingly rely on models and algorithms to predict losses resulting from customers' defaults. Hence, developing sufficiently accurate and robust models is one of the major efforts of quantitative risk management groups within these institutions.

This project develops robust and efficient models for the credit default risk problem. Specifically, build some Machine Learning classification algorithms and develop them to predict default risk for credit card accounts.

INTRODUCTION

Since 1990, the Taiwanese government has allowed the formation of new banks. In order to increase market share, these banks have issued excess cash and credit cards to unskilled applicants. At the same time, most cardholders, regardless of their repayment ability, have abused their credit card for consumption and piled up heavy credit card debt and cash. Default occurs when a credit card holder is unable to comply with the legal obligation to repay. The crisis has caused a severe blow to confidence in consumer credit and has been a major challenge for both banks and cardholders.

In a well-developed financial system, crisis management is downstream and risk prediction is upstream. The primary purpose of risk forecasting is to use financial information, such as corporate financial statements, customer transaction and refund records, etc., to predict individual customer business performance or credit risk and reduce damage and uncertainty.

In this project, the aim is to reliably predict who is at risk of defaulting. In this case, the bank may be able to prevent the loss by providing the customer with alternative options (such as forbearance or debt consolidation, etc.). Then, we build an automated model based on customer information and historical transactions that can identify key factors and predict credit card default.

EXPLORATORY DATA ANALYSIS

DATASET DESCRIPTION

The Default of Credit Card Clients dataset contains 30 000 instances of credit card status collected in Taiwan from April 2005 to September 2005. The dataset employs the binary variable default payment next month as response variable. It indicates if the credit card holders will be defaulters next month (Yes = 1, No = 0). In particular, for each record (namely, each client) we have demographic information, credit data, history of payments and bill statements. To be more precise, the following is the complete list of all the 23 predictors.

- Client personal information:

1. LIMIT BAL: Amount of given credit (in New Taiwan dollars): it includes both the individual consumer credit and his/her family (supplementary) credit.
2. SEX : 1 = male, 2 = female
3. EDUCATION: 1 = graduate school; 2 = university; 3 = high school; 4 = others.
4. MARRIAGE: Marital status, 1 = married; 2 = single; 3 = others.
5. AGE: Age in years.

- History of past payments from April to September 2005, i.e., the delay of the past payment referred to a specific month:

6. PAY 0: Repayment status in September, 2005.
7. PAY 2: Repayment status in August, 2005.
8. PAY 3: Repayment status in July, 2005.
9. PAY 4: Repayment status in June, 2005.
10. PAY 5: Repayment status in May, 2005.
11. PAY 6: Repayment status in April, 2005.

The measurement scale for the repayment status is: -1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; ...; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

- Amount of bill statement (in New Taiwan dollars), i.e. a monthly report that credit card companies issue to credit card holders in a specific month:

12. BILL AMT1: Amount of bill statement in September, 2005.
13. BILL AMT2: Amount of bill statement in August, 2005.
14. BILL AMT3: Amount of bill statement in July, 2005.
15. BILL AMT4: Amount of bill statement in June, 2005.
16. BILL AMT5: Amount of bill statement in May, 2005.
17. BILL AMT6: Amount of bill statement in April, 2005.

- Amount of previous payment (in New Taiwan dollars):

18. PAY AMT1: Amount of previous payment in September, 2005.
19. PAY AMT2: Amount of previous payment in August, 2005.

20. PAY_AMT3: Amount of previous payment in July, 2005.
21. PAY_AMT4: Amount of previous payment in June, 2005.
22. PAY_AMT5: Amount of previous payment in May, 2005.
23. PAY_AMT6: Amount of previous payment in April, 2005.

In Figure 1 we can understand what the data looks like. The target *default.payment.next.month* is renamed DEFAULT to be short, while the PAY_0 column is renamed PAY_1

	ID	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_1	...	PAY_6	BILL_AMT1	...	BILL_AMT6	PAY_AMT1	...	PAY_AMT6	DEFAULT
0	1	20000	2	2	1	24	2	...	-2	3913	...	0	0	...	0	1
1	2	120000	2	2	2	26	-1	...	2	2682	...	3261	0	...	2000	1
2	3	90000	2	2	2	34	0	...	0	29239	...	15549	1518	...	5000	0
3	4	50000	2	2	1	37	0	...	0	46990	...	29547	2000	...	1000	0
4	5	50000	1	2	1	57	-1	...	0	8617	...	19131	2000	...	679	0
5	6	50000	1	1	2	37	0	...	0	64400	...	20024	2500	...	800	0
6	7	500000	1	1	2	29	0	...	0	367965	...	473944	55000	...	13770	0
7	8	100000	2	2	2	23	0	...	-1	11876	...	567	380	...	1542	0
8	9	140000	2	3	1	28	0	...	0	11285	...	3719	3329	...	1000	0
9	10	20000	1	3	2	35	-2	...	-1	0	...	13912	0	...	0	0

Figure 1: Original dataset from UCI machine learning repository through pandas framework

Figure 2 reveals the distribution of the target variable, default payment for the following month. It distinctly illustrates an imbalance favoring the 0 class (indicating no default), which constitutes approximately 78% of the entire dataset. If this imbalance issue is not tackled, it could lead classification models to disproportionately focus on the majority class, thereby neglecting the minority class.



Figure 2: Countplot of default payment next month

DATA STRUCTURE AND CLEANING

So looking at the values present in the attributes some changes have to be done:

- The 'marriage' attribute should only contain one of the following values: 1, 2, 3; however, some records show a value of 0.
- The 'education' attribute should only include one of these values: 1, 2, 3, 4; yet, some records list values of 0, 5, 6.
- The 'PAY N' attributes should only encompass values: -1, 1, 2, 3, 4, 5, 6, 7, 8, 9; nonetheless, some entries contain values of -2 and 0.

In the first two cases, unknown values for 'marriage' and 'education' are mapped to their respective 'Other' categories (3 for marriage, 4 for education). For the 'PAY N' attributes, the values -2 and -1 are reclassified to 0 to numerically indicate the number of months a payment was delayed.

```
<class 'pandas.core.frame.DataFrame'>
Index: 30000 entries, 1 to 30000
Data columns (total 24 columns):
#   Column      Non-Null Count  Dtype
---  -
0   LIMIT_BAL    30000 non-null  int64
1   SEX          30000 non-null  int64
2   EDUCATION    30000 non-null  int64
3   MARRIAGE     30000 non-null  int64
4   AGE          30000 non-null  int64
5   PAY_1        30000 non-null  int64
6   PAY_2        30000 non-null  int64
7   PAY_3        30000 non-null  int64
8   PAY_4        30000 non-null  int64
9   PAY_5        30000 non-null  int64
10  PAY_6        30000 non-null  int64
11  BILL_AMT1    30000 non-null  int64
12  BILL_AMT2    30000 non-null  int64
13  BILL_AMT3    30000 non-null  int64
14  BILL_AMT4    30000 non-null  int64
15  BILL_AMT5    30000 non-null  int64
16  BILL_AMT6    30000 non-null  int64
17  PAY_AMT1     30000 non-null  int64
18  PAY_AMT2     30000 non-null  int64
19  PAY_AMT3     30000 non-null  int64
20  PAY_AMT4     30000 non-null  int64
21  PAY_AMT5     30000 non-null  int64
22  PAY_AMT6     30000 non-null  int64
23  DEFAULT      30000 non-null  int64
dtypes: int64(24)
memory usage: 5.7 MB
```

Table 1: Info returned by pandas on dataframe containing the given dataset

CATEGORICAL FEATURES

Analysis of the categorical features SEX, EDUCATION, and MARRIAGE, as illustrated in Figure 3 and detailed in Table 2, reveals a higher number of females than males in the dataset. Specifically, males exhibit a marginally higher likelihood of defaulting compared to females (0.24% vs 0.21%). Generally, for both genders, the distribution of DEFAULTERS and NON-DEFAULTERS aligns with the patterns observed in the other two categorical features.

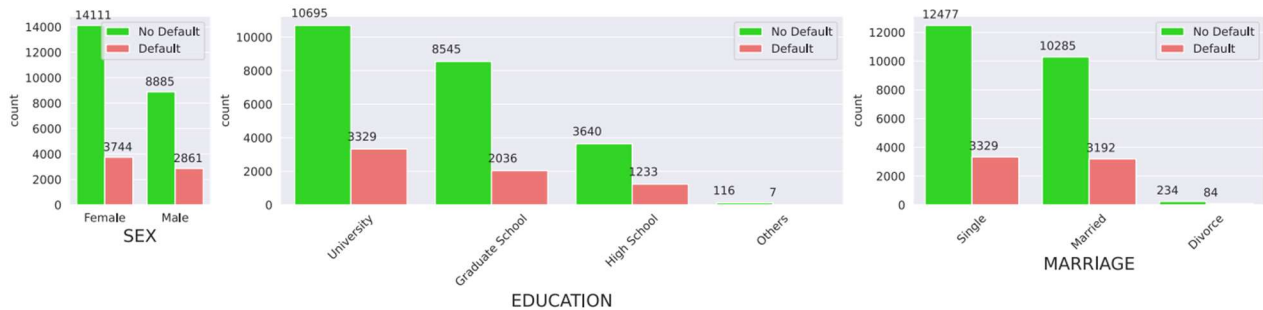
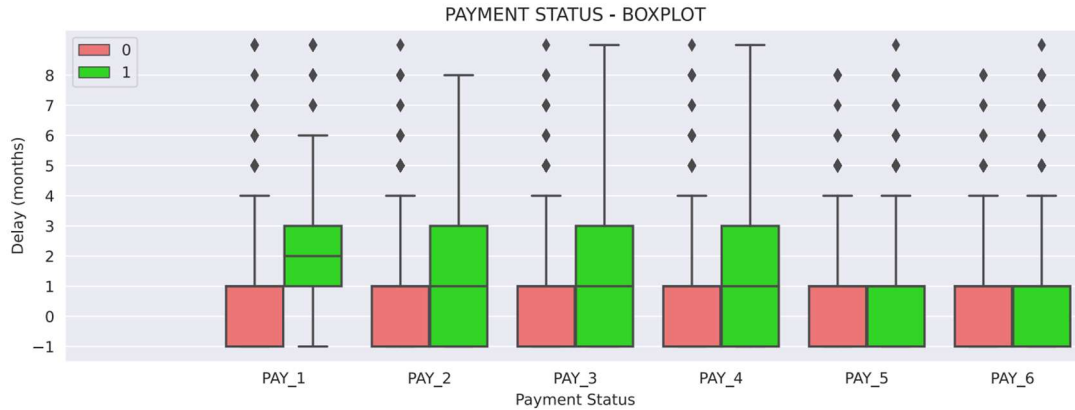


Figure 3: Countplot of SEX, EDUCATION and MARRIAGE grouped by DEFAULT class

I still need to examine the payment status feature, PAY N. The boxplots displayed in Figure 4 are particularly informative. They indicate that clients who delay payments by one month or less tend to have fewer credit card defaults. Specifically, the repayment status for September, PAY 1, demonstrates greater discriminatory power compared to the repayment statuses in other months.

attribute	value	count	defaulters	(%)
SEX	Female	17.855	3.744	20,96%
	Male	11.746	2.861	24,35%
EDUCATION	University	14.024	3.329	23,73%
	Graduate school	10.581	2.036	19,24%
	High school	4.873	1.233	25,30%
	Other	123	7	5,70%
MARRIAGE	Single	15.806	3.329	21,06%
	Married	13.477	3.192	23,68%
	Others	318	84	26,4%

Table 2: Value counts for SEX, EDUCATION and MARRIAGE feature



CONTINUOUS FEATURES

In statistics, Kernel Density Estimation (KDE) is a well-established method for estimating the probability density function in a non-parametric manner, meaning it does not rely on any assumed underlying distribution. We utilized KDE plots to explore this continuous feature. From Figure 5, we observe that the majority of defaults occur at lower LIMIT BAL values (i.e., credit amounts), particularly within a range from a few thousand to about \$140,000 Taiwanese dollars. Customers with credit amounts above this threshold are more likely to repay their debts.

For the AGE feature, a similar visual analysis shows that the likelihood of non-default is higher among individuals aged approximately 25 to 42. This suggests that people in this age group are generally more capable of repaying credit card loans, possibly due to more stable work and family situations with less financial pressure.

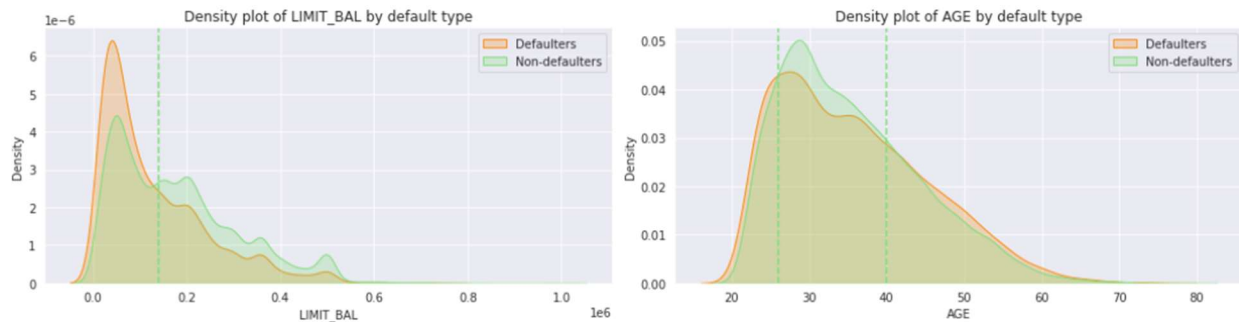


Figure 5: KDE plots of LIMIT BAL and AGE grouped by DEFAULT class

CORRELATION

From Figure 7, as we may think, we can observe an "internal" correlation among the groups of features such as BILL ATM, PAY N. We can also notice that there is no feature with a strong relationship with the target. In fact, there are 15 features with an absolute value of the correlation below than 0.1 and none of the remaining ones have a greater correlation than 0.29.

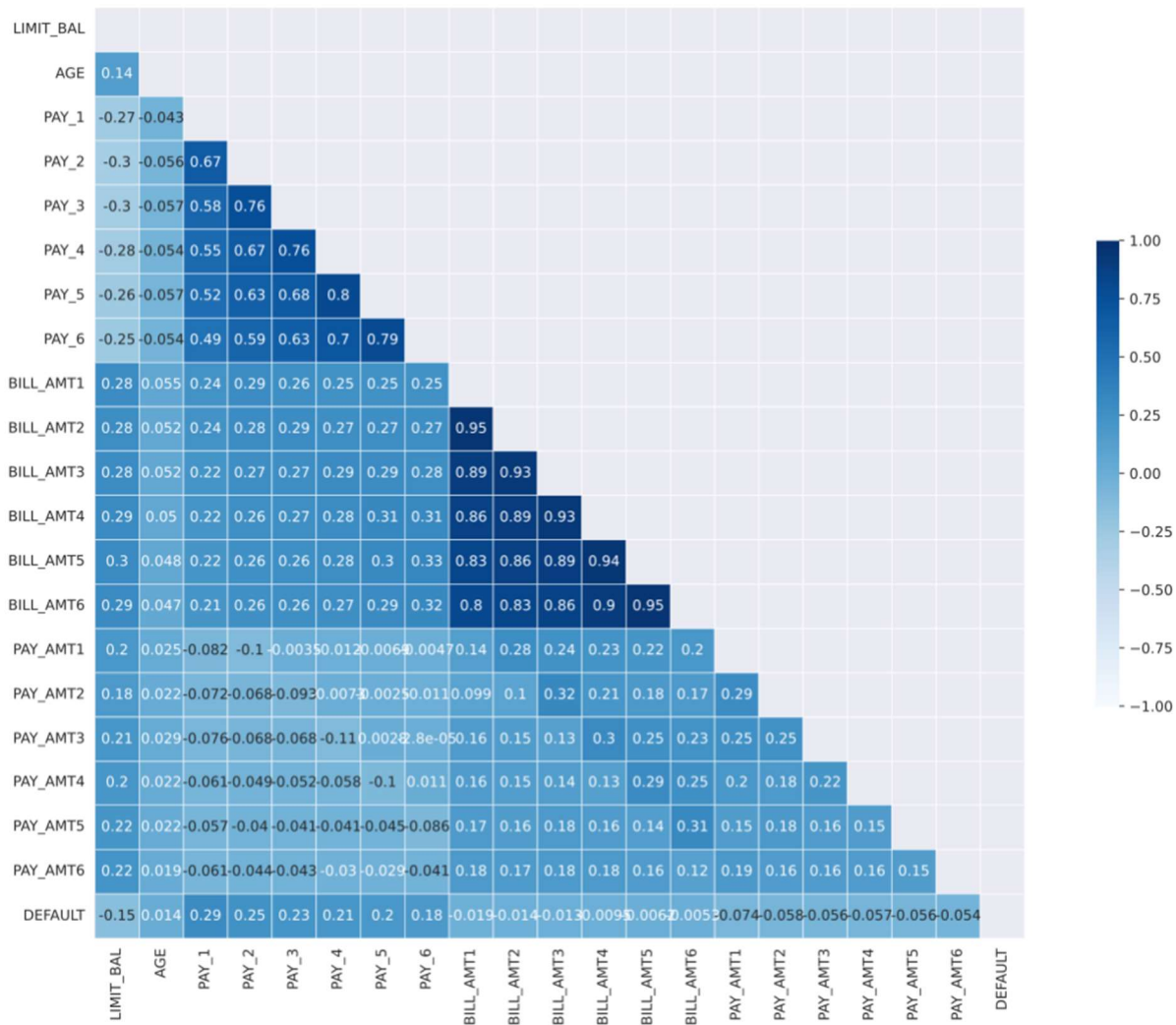


Figure 7: Heatmap correlation

The categorical features EDUCATION, SEX, and MARRIAGE are encoded as integers, suitable for machine learning models. However, these are nominal categories without a meaningful order, making one-hot encoding necessary to avoid implied ordinality. This technique creates a new binary feature for each category, removing any ordinal relationship. While Scikit-Learn offers automatic one-hot encoding, we manually mapped these features to control multicollinearity—a common issue with highly correlated features. Consequently, we replaced the original EDUCATION, SEX, and MARRIAGE features with new boolean columns.

DIMENSIONALITY REDUCTION

As previously noted, reducing the number of strongly correlated features and lowering data dimensionality can benefit many algorithms. Too many features result in overly complex models that fit too closely to the training data but fail to generalize well to new data. This situation, known as overfitting, leads to models with high variance.

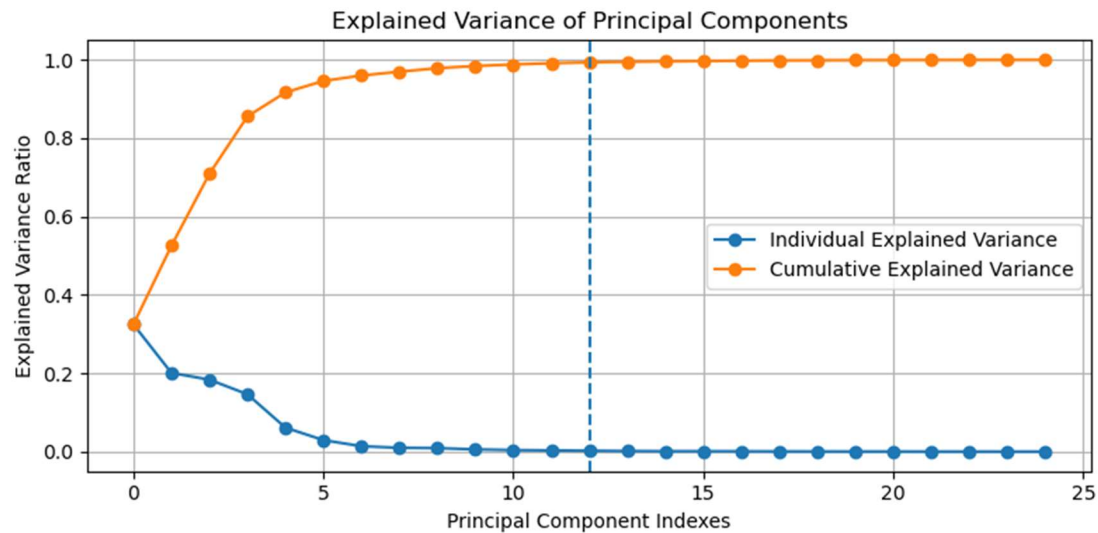


Figure 7: Cumulative and individual explained variance plotted against principal components

Reducing the dataset's dimensionality involves compressing it into a new subspace using principal component analysis (PCA). The number of principal components selected should retain most of the original data's information. This is assessed using the explained variance ratio, which measures how much variance each component captures. Results shown in Figure 11 indicate that the first six principal components account for over 90% of the total variance. By using the first 12 components, 99% of the variance is explained, despite halving the number of features.

CLASS IMBALANCE - RESAMPLING

Class imbalance is prevalent in our dataset, as shown in Section 2.2 and Figure 2, where non-defaulters significantly outnumber defaulters. Addressing this, it's important to focus on metrics beyond accuracy when evaluating models, due to the majority class bias in model training. We opt to explore advanced techniques like the Cluster Centroid method and Synthetic Minority Oversampling Technique (SMOTE) to effectively manage this imbalance.

CLASSIFICATION MODELS

In the "Classification Models" section, I explore several supervised learning algorithms applied to our dataset to predict credit card defaults for the next month. These include Support Vector Machines (SVM), Logistic Regression, and tree-based methods like Decision Trees and Random Forest, adhering to the Empirical Risk Minimization framework.

LOGISTIC REGRESSION:

Logistic Regression is a type of Generalized Linear Model (GLM) which predicts the probability of class membership, such as defaulting on a credit card. The logistic function models this probability, ensuring

outputs between 0 and 1. Coefficients are estimated using Maximum Likelihood, and regularization (L2 by default in Scikit-learn) is applied to minimize overfitting.

DECISION TREE AND RANDOM FOREST:

Decision Trees create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. It is highly interpretable but less robust and more prone to overfitting, which is mitigated by pruning the tree. Random Forest, an ensemble method using multiple decision trees to reduce variance, improves prediction accuracy and robustness, although it decreases model interpretability.

SUPPORT VECTOR MACHINE (SVM):

SVM is a powerful classification technique that constructs a hyperplane or set of hyperplanes in a high-dimensional space to separate classes with a maximum margin. The approach can be extended using the kernel trick to handle non-linear boundaries, using functions that map inputs into higher-dimensional spaces where a linear separator is constructed. Soft Margin SVM allows for some misclassifications, increasing the model's flexibility, particularly when data isn't linearly separable.

These models will be evaluated on their ability to accurately predict credit card defaults, with a focus on handling the inherent class imbalance in our dataset, using methods like adjusting class weights or resampling techniques. The model's performance will be measured beyond mere accuracy, incorporating metrics such as precision, recall, and the AUC-ROC curve, to ensure robustness against the skewed class distribution.

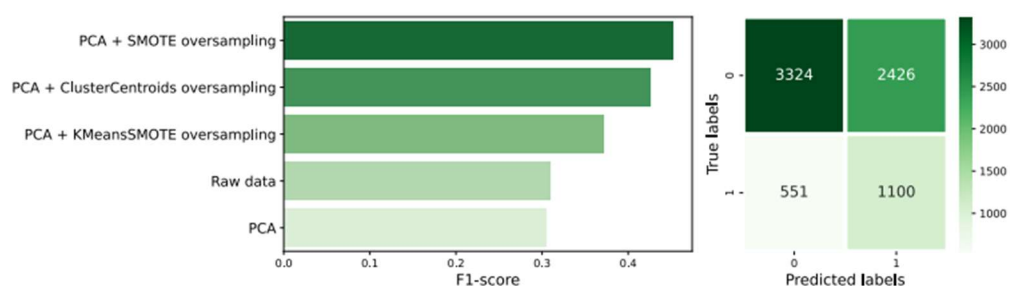
RESULTS

In the following pages, an overview on the results is given for each classifier. Different preprocessing combinations were tested: applying dimensionality reduction techniques (PCA) or not, using or not different resampling techniques. The metric we choose to adopt is F1-score. Precision-recall curve and Confusion Matrix of the best model have been provided for each model.

LOGISTIC REGRESSION

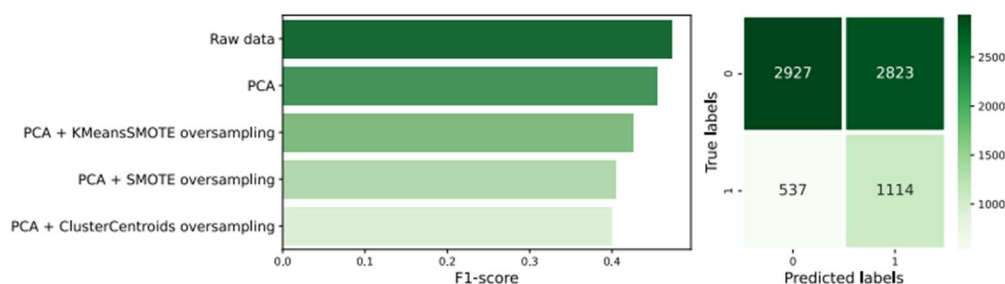
The Logistic Regression model was tested with various settings of the regularization parameter C and different preprocessing strategies. The results varied, with PCA combined with SMOTE (Synthetic Minority Over-sampling Technique) showing the best performance with an F1-score of 0.4504. The confusion matrix for this setting was provided, illustrating the effectiveness of this approach in managing class imbalance.

Preprocessing	Parameters	F1-score
None (Raw data)	C : 50, penalty: l2	0,3090
PCA	C : 50, penalty: l2	0,3028
PCA + SMOTE	C : 35, penalty: l2	0,4504
PCA + KMeans SMOTE	C : 35, penalty: l2	0,3887
PCA + ClusterCentroids	C : 20, penalty: l2	0,4213



DECISION TREE

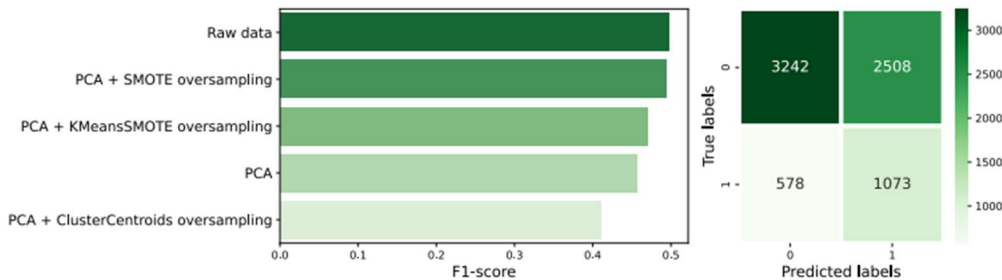
The Decision Tree model performed best with the original dataset, without any preprocessing, achieving higher scores compared to other settings. This suggests that, for Decision Trees, maintaining the natural distribution of the data without manipulation could be beneficial. The associated confusion matrix highlighted the model's performance in classifying default and non-default cases.



RANDOM FOREST

For the Random Forest classifier, various combinations of the number of estimators and the maximum number of features were explored. Like the Logistic Regression model, the best results were achieved using PCA with SMOTE, enhancing the model's ability to generalize by addressing the class imbalance. The optimal settings involved using the square root of the number of features and 200 estimators.

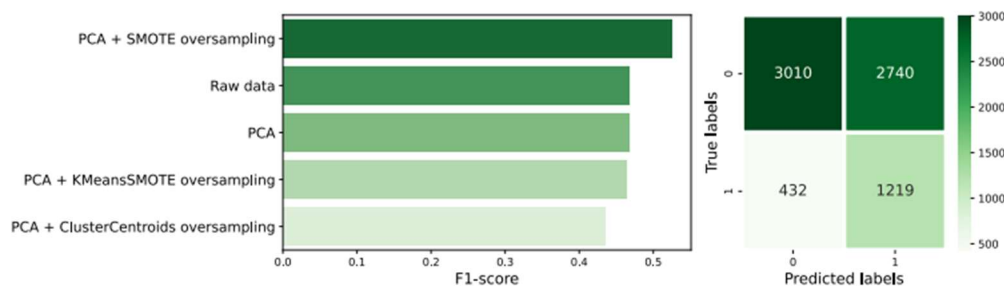
Preprocessing	Parameters	F1-score
None (Raw data)	max_features:None, n_estimators: 200	0,4972
PCA	max_features:None, n_estimators: 200	0,4609
PCA + SMOTE	max_features:sqrt, n_estimators: 200	0,4919
PCA + KMeans SMOTE	max_features:None, n_estimators: 50	0,4607
PCA + ClusterCentroids	max_features:sqrt, n_estimators: 100	0,4153



SUPPORT VECTOR MACHINE (SVM)

The SVM was configured with different kernel settings and the regularization parameter C . Again, the combination of PCA and SMOTE provided the best results, particularly using the RBF kernel with $C = 100$ and $\lambda = \text{scale}$, achieving an F1-score of 0.5247. This setup showed the SVM's capability to handle non-linear relationships effectively when combined with appropriate preprocessing.

Preprocessing	Parameters	F1-score
None (Raw data)	kernel: RBF, $C : 100$, $\lambda : 0.1$	0,4681
PCA	kernel: RBF, $C : 100$, $\lambda : 0.1$	0,4665
PCA + SMOTE	kernel: RBF, $C : 100$, $\lambda : \text{scale}$	0,5247
PCA + KMeans SMOTE	kernel: RBF, $C : 100$, $\lambda : \text{scale}$	0,4639
PCA + ClusterCentroids	kernel: RBF, $C : 100$, $\lambda : \text{scale}$	0,4341



OVERALL OVERVIEW

The comparative results of all models and techniques are summarized in a comprehensive table, presenting accuracy, recall, precision, F1-score, and AUC values. It is evident from the results that preprocessing techniques, especially PCA combined with SMOTE, generally enhance model performance

by providing a more balanced dataset. However, the improvement varies by model, with SVM and Logistic Regression showing significant gains in F1-score, while Decision Trees and Random Forests demonstrate moderate improvements.

		Accuracy	Recall	Precision	F1-score	AUC
Logistic Regression	Raw data	0.804891	0.196245	0.734694	0.309751	0.555119
	PCA	0.804216	0.192005	0.733796	0.304369	0.553024
	PCA + SMOTE oversampling	0.631942	0.679588	0.338257	0.451691	0.544661
	PCA + KMeansSMOTE oversampling	0.724767	0.363416	0.378310	0.370714	0.441867
	PCA + ClusterCentroids oversampling	0.597757	0.666263	0.311968	0.424957	0.526340
Support Vector Machine	Raw data	0.822186	0.350697	0.703524	0.468068	0.599533
	PCA	0.821646	0.349485	0.701094	0.466451	0.597847
	PCA + SMOTE oversampling	0.769896	0.569352	0.486542	0.524700	0.575981
	PCA + KMeansSMOTE oversampling	0.747061	0.490612	0.439978	0.463918	0.522112
	PCA + ClusterCentroids oversampling	0.571409	0.738340	0.307906	0.434581	0.552308
Decision Tree	Raw data	0.820565	0.359782	0.686705	0.472178	0.594653
	PCA	0.807864	0.357965	0.620147	0.453917	0.560668
	PCA + SMOTE oversampling	0.684232	0.481526	0.349297	0.404889	0.473242
	PCA + KMeansSMOTE oversampling	0.745980	0.422168	0.429452	0.425779	0.490261
	PCA + ClusterCentroids oversampling	0.546007	0.674743	0.282957	0.398712	0.515128
Random Forest	Raw data	0.820970	0.396729	0.665650	0.497154	0.598478
	PCA	0.807458	0.362810	0.616255	0.456729	0.560604
	PCA + SMOTE oversampling	0.764762	0.516051	0.474916	0.494630	0.549463
	PCA + KMeansSMOTE oversampling	0.784489	0.427014	0.520679	0.469218	0.537757
	PCA + ClusterCentroids oversampling	0.583029	0.649909	0.299637	0.410168	0.513822

CONCLUSION

My project across various supervised learning algorithms demonstrated that data preprocessing could slightly enhance performance, particularly when PCA is used to reduce computational cost without significantly affecting outcomes. Among the resampling strategies, oversampling typically yielded better results than undersampling, likely due to increased data availability for model training. While all models performed comparably in accuracy, further improvements might be achieved through advanced methods such as Gradient Boosting classifiers or anomaly detection techniques like Local Outlier Factor or Isolation Forest. These could potentially address existing shortcomings in model performance and further refine the predictive accuracy for credit card default prediction.