Journal of
Risk and Financial Management

MDPI

Article

Accuracy Comparison between Five Machine Learning Algorithms for Financial Risk Evaluation

Haokun Dong, Rui Liu and Allan W. Tham *

Faculty of Science and Technology, University of Canberra, Canberra 2617, Australia; haokun.dong@canberra.edu.au (H.D.); drruiliu@yeah.net (R.L.)
* Correspondence: allan.tham@canberra.edu.au

Abstract: An accurate prediction of loan default is crucial in credit risk evaluation. A slight deviation from true accuracy can often cause financial losses to lending institutes. This study describes the non-parametric approach that compares five different machine learning classifiers combined with a focus on sufficiently large datasets. It presents the findings on various standard performance measures such as accuracy, precision, recall and F1 scores in addition to Receiver Operating Curve-Area Under Curve (ROC-AUC). In this study, various data pre-processing techniques including normalization and standardization, imputation of missing values and the handling of imbalanced data using SMOTE will be discussed and implemented. Also, the study examines the use of hyper-parameters in various classifiers. During the model construction phase, various pipelines feed data to the five machine learning classifiers, and the performance results obtained from the five machine learning classifiers are based on sampling with SMOTE or hyper-parameters versus without SMOTE and hyper-parameters. Each classifier is compared to another in terms of accuracy during training and prediction phase based on out-of-sample data. The 2 data sets used for this experiment contain 1000 and 30,000 observations, respectively, of which the training/testing ratio is 80:20. The comparative results show that random forest outperforms the other four classifiers both in training and actual prediction.

Keywords: financial data analysis; machine learning algorithms; loan default assessment; classification

Citation: Dong, Haokun, Rui Liu, and Allan W. Tham. 2024. Accuracy Comparison between Five Machine Learning Algorithms for Financial Risk Evaluation. *Journal of Risk and Financial Management* 17: 50. https://doi.org/10.3390/jrfm17020050

Academic Editor: Thanasis Stengos

Received: 1 October 2023
Revised: 19 January 2024
Accepted: 22 January 2024
Published: 29 January 2024

1. Introduction

Financial institutions are facing increasing challenges in mitigating various kinds of risks. In his "taxonomy of risks", Christoffersen (2011) defines risks as market volatility, liquidity, operational, credit and business risks. Due to uncertainties, financial risk evaluation (FRE) is increasingly playing a pivotal role in ensuring organizations maximize their profitability by minimizing losses due to a failure to mitigate risks. Noor and Abdalla (2014) argue that there is a direct negative impact on profitability in proportion to unmitigated risks. Hence, the primary approach of FRE is to identify risks in advance to allow for an appropriate course of action before any investments or decisions can be made. As financial risks evolve over time due to factors such as economic fluctuations, market conditions and other factors beyond control, the evaluation process requires constant update to keep up with market conditions.

Credit risk analysis undertaken in recent years mostly involves financial risk prediction. For example, loan default analysis, which often comes in the form of binary classification problems, has become an integral part of FRE. As financial institutions today are dealing with millions of customers, the traditional human approach for loan approval processes are no longer feasible. Moreover, with the advent of computing power today coupled with the advancement of machine learning algorithms and the availability of large volume of information, the world has entered the renaissance of computational modeling with non-parametric classification methods and machine learning for loan default prediction becoming widely adopted. It is worth noting that machine learning classification achieves

an accuracy that directly increases the bottom line whilst providing instantaneous approval decision through real-time decisions.

The use of advanced computing power and machine learning algorithms to predict whether a loan is performing or non-performing (NPL) is increasingly essential for the longevity of any lending institute. As the pool of consumers enlarges with proportional increases in spending, the ability to provide early warnings by accurately predicting the probability of defaulting a loan has become even more crucial. Deploying advanced machine learning algorithms to identify patterns from large features in high-volume data has become a mandatory process by banks to minimize the NPLs, and thus to increase their profitability and consumers' confidence.

In this study, we aim our attention at loan default detection as an element of credit risk analysis through models built in k-nearest neighbour, naïve-bayes, decision tree, logistic regression, and random forest. The intention is to answer the following questions:

- Is there a significant performance difference between the five machine learning algorithms piping through Scikit-learn data transformation steps?
- Can the steps be repeated with the same level of consistency using different data sets but similar analytics pipelines with data transformation?

The study is arranged in the following way:

- In Section 2, we briefly cover the background of the rise of statistical methods used from the 60s to the 80s, primarily in the form of parametric approaches in predicting bankruptcy. This section also covers how modern predictive techniques were born in the 90s and beyond in conjunction with the availability of computing power, resulting in the advancements of this field.
- In Section 3, we conduct a case study to illustrate the use of the five machine learning algorithms to predict the loan default based on University of California at Irvine (UCI) data set. In this section, we present the analytics life-cycle methodology with emphasis on data pre-processing. It highlights the repeatability and validity of the methodology in conducting research.
- In Section 4, we construct various models and measure them using various tools, of which ROC-AUC is the main measurement. Other measurements are accuracy, precision, F1 score, etc. In this section, we draw comparisons between five classifiers and present the results neatly in various tables. We also present the out-of-sample prediction results to validate model accuracy.

2. Related Work

It is imperative for financial institutes to detect NPLs in advance and segregate them for further treatments. Unlike today, however, the ability to predict NPLs in the 60s was not commonplace due to the fact that data mining and predictive capabilities were in their embryonic state. During that era, financial analysis using a quantitative approach was in its nascent form. Mathematical models and statistical methods were basic compared to modern quantitative techniques. Apart from relying on studying a company's financial statements, most financial risks analysis primarily relied on fundamental analysis which involves studying external factors such as market trends and economic indicators.

Beaver (1966) laid a foundation of groundbreaking work in accounting, earning himself management using financial ratios. "Beaver's Model" involved seminal univariate analysis to predict corporate failure. Altman (1968) devised the "Altman Z-Score" to predict the probability of whether a company will undergo bankruptcy. Beaver and Altman's work pioneered approaches to financial risk analysis for the next decade.

Finance-related prediction in the 1970s hinged on Altman's Z-Score, which had garnered popularity since the late 1960s. Although Altman's work primarily involved predicting bankruptcy, academics and researchers adapted the underlying principles to perform prediction of risks to maintain financial health. The 1970s marked the emergence of modern risk management concepts with financial institutes becoming aware of the importance of identifying and managing various risk portfolios. The 1970s laid the groundwork for iden-

tifying and understanding financial risk prediction and management. This development was the beginning of the evolution of risk assessment methodologies and the adoption of risk management practices together. The regulatory frameworks aimed to enhance stability and resilience were set up by regulatory bodies.

Black and Scholes (1973) developed the Black–Scholes–Merton (BSM) model in 1973 which aimed to calculate the theoretical price of European-style options. The model uses complex mathematical formulas and assumes standard normal distribution including logarithms, standard deviations (precursor to Z-Score) and cumulative distribution functions. The Black–Scholes–Merton model remains a foundation of today's market risk assessment and serves as a fundamental tool for pricing options. Although specific research publications in the 1970s may not be common enough to be readily cited, many ideas, concepts and methodologies established during that timeframe set the stage for subsequent developments. Most notable is the gaining of traction of the quantitative approach to credit risk modeling and scoring. The rise of algorithms such as regression, discriminant analysis and logistic regression dominated the 1970s. The duo's empirical results also demonstrated how efficient regulatory policy should be formulated from the regression outcomes. Deakin (1972), standing on the shoulders of Beaver and Altman, brought the analysis one notch higher using a more complex, albeit discriminatory, analysis to improve on the 20% error in misclassification of bankruptcy for the year prior. Deakin's model of an early warning system assumed a random draw of samples and used various financial ratios and indicators including profitability ratios, efficiency ratios and liquidity ratios (amongst others) to distinguish between troubled and healthy firms. Martin (1977) leveraged the logit regression approach to predict the likelihood of banks experiencing financial distress.

The 1980s saw an increased focus on credit risk measurement within banking industries. Managing creditworthiness, credit exposures and the probabilities to default were key research topics by researchers and practitioners. Ohlson took interest of White and Turnbull's unpublished work on systematically developed probabilistic estimates of failures. Ohlson (1980) used the maximum likelihood estimation methodology, which is a form of conditional logit model (logistic regression), to avoid the pitfall of well-known issues associated with multivariate discriminant analysis (MDA) deployed in previous studies. Ohlson's model, primarily a parametric one (as most models were in that era), provided advantages in that no assumptions must be made to account for prior probabilities regarding bankruptcy and the distribution of predictors. Ohlson argued that Moody's manual, as relied on by previous works, could be flawed due to the fact that numerous studies that derived financial ratios from the manual did not account for the timing of data availability and the complexity in reconstructing balance sheet information from the highly condensed report. In his concluding remark, Ohlson stated that the prediction power of any model depends upon when the financial information is assumed to be available. West (1985) combined the traditional parameter approach using a logit algorithm with factor analysis. West's work was promising, as the empirical results show the combination of the two techniques closely matched the CAMEL rating system widely used by bank examiners in that era.

The 1990s and 2000s saw the birth of some exciting machine learning algorithms. Up until this point, most statistical methods used for credit assessment were related to the parametric approach. The parametric algorithms mandate that the assumptions of linearity, independence, or constant variance are met before meaningful analysis can be derived. The birth of Adaptive Boosting can be indebted to the work of Freund and Schapire (1997). The duo proposed that a strong classifier can be obtained by combining multiple weak classifiers iteratively. Friedman (2001) devised a method to improve the predictive accuracy by optimizing a loss function through iterative processes. Friedman's gradient boosting machine (GBM) builds the trees sequentially, with each tree correcting by fitting the residuals of the previous trees. Friedman's work was influential and subsequently gave rise to other boosting variations, including XGBoost by Chen and Guestrin (2016) and LightGBM by Ke et al. (2017). Breiman and Cutler (1993), however, proposed a way

to construct multiple independent decision trees during training, with each tree deriving from a subset of training data and available features. Breiman's (2001) random forest model ensures that each tree is trained on a bootstrap sample of data (random sample with replacement). The final prediction is made from aggregating the prediction from an ensemble of diverse decision trees. Vapnik and Chervonenkis' early work dated as far back as the early 1960s in theory of pattern recognition, and statistical learning laid the groundwork for their support vector machine (SVM). Vapnik's (1999) algorithm is known for the ability to classify both linear and non-linear data by finding the optimal hyperplane that best separates various classes whilst maximizing the margin between them.

Contemporary literature works in predicting financial risk has mushroomed over the past decade. Peng et al. (2011) suggest that a unique classification algorithm that could achieve the best accuracy given different measures under various circumstances does not exist. In their early attempts, Desai et al. (1996), and later West (2000), both proposed that the performance of generic models such as linear discriminant were not a better performer than customized models, except for a customized neural network. However, further studies by Yobas et al. (2000) using linear discriminant, neural network, genetic algorithms and decision tree concluded that the best performer was linear discriminant analysis. Due to the inconsistencies of previous studies, Peng et al. (2011) suggested multiple criteria decision making (MCDM), whereby a process to allow systematic ranking and selecting of an appropriate classifier or cluster of classifiers should be at the forefront of classification research. In the first ever academic study of Israeli mortgage, Feldman and Gross (2005) applied the simple yet powerful classification and regression tree (CART) to 3035 mortgage borrowers in Israel, including 33 features such as asset value, asset age, mortgage size, number of applicants, income, etc. The goal was to classify between potential defaulters and those unlikely to default. The distinct feature of CART that resulted in it being chosen over its primary competitors is its ability to manage missing data. Khandani et al. (2010) predicted the binary outcome that indicates whether an account is delinquent by 90 days by including the time dimension of 3-, 6- or 12-month windows. Using a proprietary dataset from a major bank, Khandani and others combined customer banking transactions (expenditures, savings and debt repayments), debt-to-income ratios and credit bureau data to improve the classification rates of credit card holders' delinquencies and defaults. CART was chosen as the non-parametric approach due to its ability to manage the non-linearity nature of data and inherent explainability of the algorithm. Their work proved that the time series properties of the machine learning prediction commensurate with realized delinquency rates, with R^2 of 85%. He suggested assigning weight in training data as adaptive boosting to manage imbalanced class.

The rise of data gathering exercises made available hundreds or thousands of features compounded with imbalanced data, posing an issue for traditional approaches. The non-parametric approach burst onto the scene to manage the ever-increasing dimension, imbalanced data and the non-linear nature of models. The 2000s saw a rise of applying multi-layer neural networks and support vector machines (SVM) to financial prediction. Atiya (2001) proposed a non-parametric approach using a novel neural network model and was able to achieve accuracy of 3-year-ahead out-of-sample predictions between 81–85% accuracy. Zhang et al. (1999) suggested that artificial neural networks outperformed logistic regression. Huang et al. (2004) deployed backpropagation neural networks (BNN) and SVMs to achieve an accuracy of 80%.

Although the majority of datasets used for the studies are propriety in nature, there was little mention regarding the engagement of various data preparation techniques except from the recent study of the importance of data pre-processing effects on machine learning by Zelaya (2019) using the contemporary machine learning package such as Scikit-learn popularized by Pedregosa et al. (2011). The modern machine learning packages with full pipeline feature as shown by Varoquaux et al. (2015) are worth exploring. Equally omitted is the implementation of techniques such as SMOTE to manage imbalanced class, as proposed by Fernández et al. (2018), which is also worth further study.

In this study, we aim our attention at loan default detection as an element of credit risk analysis.

3. Case Study—Advanced Machine Learnings for Financial Risk Mitigation

3.1. Methodology—Computational Approach

In this study, the machine learning analytics cycle use Scikit-learn packages to implement an analytics pipeline that includes data collection, data pre-processing, model constructions and model performance comparisons. Matplotlib supplies graphing capability to allow for the visual analysis of data.

Scikit-learn allows for the full analytics pipeline to specifically unravel the underlying pattern in data sets, therefore resulting in the best fitting for various classifiers. The pipeline contains end-to-end processes that performs these tasks: (i) ingest the data sets and perform preliminary data analysis to identify missing values, outliers and imbalanced class—any missing values will be imputed and imbalanced data is identified; (ii) standardize data which includes scaling and normalization to ensure consistent model performance; (iii) encode categorical (nominal and ordinal) and one-hot-encode for predictors and label for target variable; (iv) select top N most influential predictors and reduce total dimension to the influential ones; (v) cross validate using k-fold stratified to ensure the ratio of imbalance remains intact and subsequently treated by SMOTE as suggested by Chawla et al. (2002); (vi) train and fit data using various distance- and tree-based classifiers; (vii) compare the final performance measurements and report the most effective hyper-parameters.

Figure 1 illustrates the machine learning analytics life cycle implemented as an end-to-end analytics pipeline using Scikit-learn's pipeline capability.

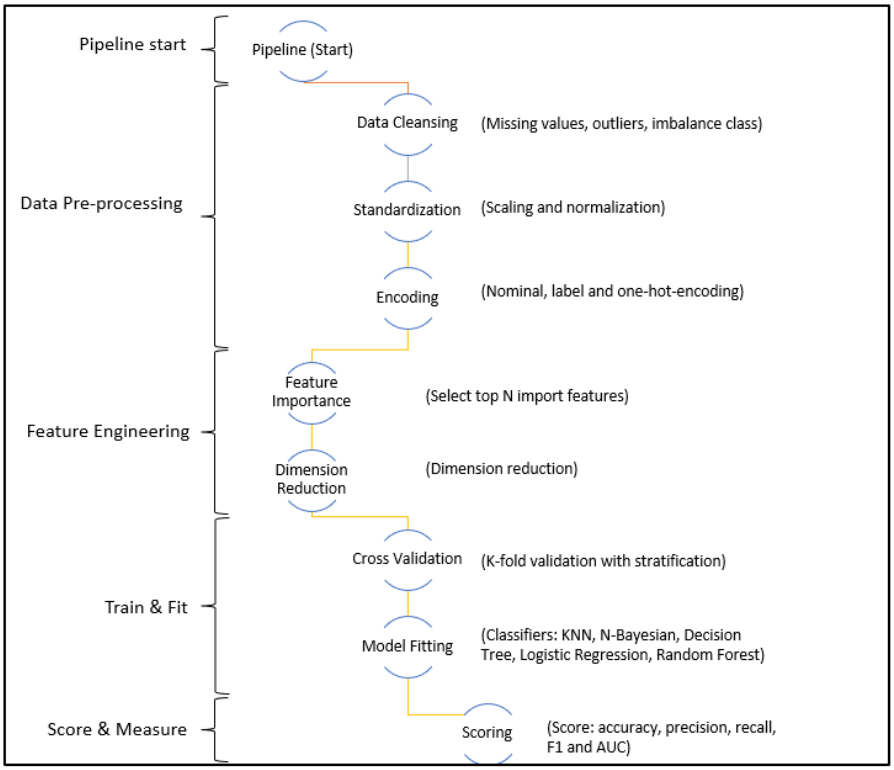

Figure 1. End-to-end analytics pipeline for machine learning classification.

The machine learning lifecycle is implemented as Scikit-learn's pipeline, easing the foremost data pre-processing in missing value imputation with either the most frequent value (categorical features) or standard mean/median (numeric variable). The analytics pipeline detects outliers and imbalanced class, as well as manages the treatment of detection further down the pipeline right before the actual model fitting. Next is to apply data

standardization, which includes transforming the data into a common scale using Z-Score, and normalize the data to a range between zero and one. The goal is to ensure data consistency across various classifiers which will result in comparisons at similar scales, thus improving model performance.

Subsequently, the analytics pipeline automatically detects the champion model (winner of the best classifier) and reports the top N predictors that are most influential to the model. The analytics pipeline finds the least influential predictors which subsequently truncated to reduce the dimension whilst not affecting the performance of the models. The analytics pipelines split the data into two sections with training and testing data segregated by a ratio of 80:20. The analytics pipeline implements k-fold with stratification to ensure that the imbalanced class stays intact. It also ensures a full data split throughout with little-to-no possibility of a data leak. Finally, it trains and fits the data through the five classifiers. At the end, it obtains the performance scores for final comparisons.

Apart from its stochastic nature, the research method is sound and repeatable, and researchers can refer to it for further studies with various data sets applied to different classifiers.

3.2. Data Collection

This study uses two credit card client data sets obtained from UCI repository.

The first set is the payment data set obtained from one major bank in Taiwan from 2005, donated by Yeh and Lien (2009) and Yeh (2016) to the UCI data repository. The data set holds 30,000 observations, of which 6636 are default payment (showed by variable id, x24 as 1) whilst healthy payment occupies the remaining 23,364 observations. The data set holds no duplicate and missing values. The Taiwan credit card payment data set shows a strong skew (healthy:default ratio) due to imbalanced class of 77.88% to 22.12%. The variable id, x24 is the target whilst it uses the remaining features (x1 to x23) as predictors (Table 1).

Table 1. Dataset 1: Taiwan credit card client data set features and types.

Total Missing Values	Taiwan Credit Data Set Features		
	Feature ID/Name	**Description**	**Numeric/ Nominal/Ordinal**
0	x1 (limit_bal)	Amount of the given credit (NT dollar): includes both the individual consumer credit and his/her family (supplementary) credit	Numeric
0	x2 (sex)	Gender (1 = male, 2 = female).	Numeric
0	x3 (education)	Education (1 = graduate school, 2 = university, 3 = high school, 4 = others)	Numeric
0	x4 (marriage)	Marital status (1 = married, 2 = single, 3 = others)	Numeric
0	x5 (age)	Age (year)	Numeric
0	x6–x11 (pay_1 to pay_6)	History of past payment. We tracked the past monthly payment records (from April to September 2005) as follows: X6 = the repayment status in September 2005, X7 = the repayment status in August 2005,; X11 = the 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.	Numeric

Table 1. *Cont.*

Total Missing Values	Taiwan Credit Data Set Features		
	Feature ID/Name	**Description**	**Numeric/ Nominal/Ordinal**
0	x12–x17 (bill_amt1 to bill_amt6)	Amount of bill statement (NT dollar). X12 = amount of bill statement in September 2005, X13 = amount of bill statement in August 2005,; X17 = amount of bill statement in April, 2005	Numeric
0	x18–x23 (pay_amt1 to pay_amt6)	Amount of previous payment (NT dollar). X18 = amount paid in September 2005, X19 = amount paid in August 2005,; X23 = amount paid in April 2005	Numeric
0	x24 (default_payment_next_ month)	Default or not (default = 1, health = 0)	Numeric

This data set contains only numeric features. It is used as a control data set for the analytics pipeline due to its larger set of observations. It will be used to validate the analytics pipeline that includes data transformations.

The second set is a German credit card client data set obtained from UCI data repository, Hofmann (1994). It contains 1000 observations. The data set contains one target variable with an imbalanced class ratio of 70% to 30% (no:yes ratio). The data set is void of missing values and duplicates. Table 2 shows the data set features, description and data types.

Table 2. Data Set 2: German credit card client data set features and types.

Total Missing Values	German Credit Data Set Features		
	Feature	**Description**	**Numeric/Nominal/Ordinal**
0	checking_balance	Status of existing checking account	Ordinal
0	months_loan_duration	Duration in months	Numeric
0	credit_history	Credit history	Ordinal
0	purpose	Purpose of loan	Nominal
0	amount	Credit amount	Numeric
0	savings_balance	Saving accounts/bonds	Ordinal
0	employment_duration	Present employment since	Ordinal
0	percent_of_income	Install rate (% of disposable income)	Numeric
0	years_at_residence	Present residence since	Numeric
0	age	Age in years	Numeric
0	other_credit	Other installment plans	Nominal
0	housing	Housing Situation	Nominal
0	existing_loans_count	Number of existing credits	Numeric
0	job	Job skill level	Ordinal
0	dependents	Number of dependents	Numeric
0	phone	Holding Telephone or not	Nominal
0	default	Default or not	Nominal

This data set contains both numerical and categorical data and is used to train and test various classifiers initially.

3.3. Visual Data Exploration

Either a classifier is parametric or non-parametric. Visual data exploration aids in understanding data structure and nature, which includes the data distributions, correlations, multi-collinearity and other patterns. Visual data exploration helps to identify anomalies and outliers in the data set that can skew analysis and model accuracy. In particular, the

involvement of logistic regression and naïve-bayes necessitate a thorough analysis of data structure and patterns as these classifiers assume independence and linearity, amongst other things. Table 3 reveals the correlation between numerical features.

Table 3. Correlation between numerical features—German credit data set.

Correction Matrix							
	C1	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**
C1	1.0000	0.6250	0.0747	0.0341	−0.0361	−0.0113	−0.0238
C2	0.6250	1.0000	−0.2713	0.0289	0.0327	0.0208	0.0171
C3	0.0747	−0.2713	1.0000	0.0493	0.0583	0.0217	−0.0712
C4	0.0341	0.0289	0.0493	1.0000	0.2664	0.0896	0.0426
C5	−0.0361	0.0327	0.0583	0.2664	1.0000	0.1493	0.1182
C6	−0.0113	0.0208	0.0217	0.0896	0.1493	1.0000	0.1097
C7	−0.0238	0.0171	−0.0712	0.0426	0.1182	0.1097	1.0000

Note: C1—months_loan_duration, C2—amount, C3—percent_of_income, C4—years_at_residence, C5—age, C6—existing_loans_count, C7—dependents.

As indicated in Table 3, the German credit data set has a low correlation between features. The highest correlation is 0.6250 between "months loan duration" and "amount". It can be said that the correlation amongst other features is non-existent as indicated by scatterplots in Figure 2. The only correlation of 0.6250 shows a positively trending linear relationship. However, "age" and "percent of income" do not show a visual pattern and therefore do not indicate a relationship with "months loan duration."

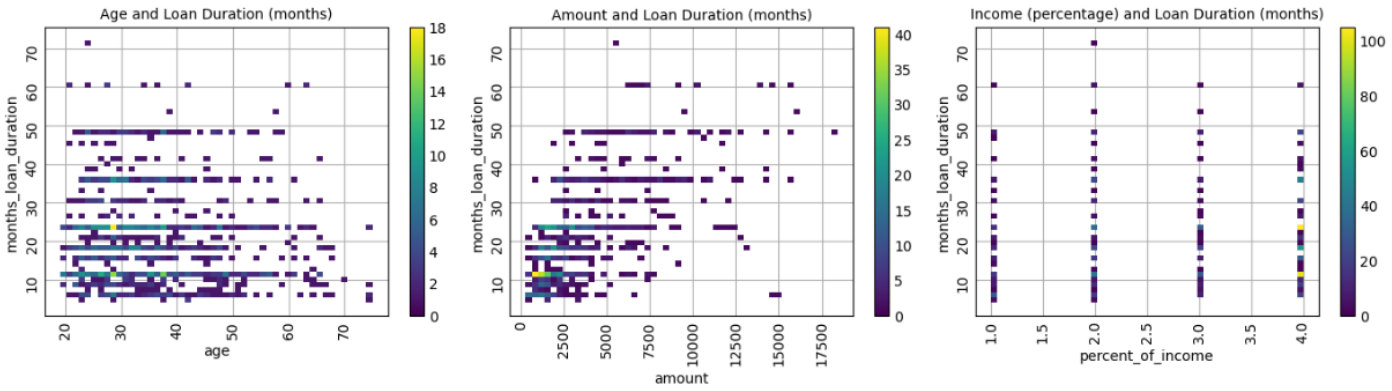

Figure 2. Scatterplots for selected numeric features.

Further investigation into multicollinearity of the German credit data set (Table 4) shows that the variance inflation factor (VIF) values between predictors are reasonable and do not cause alarm.

Table 4. VIF—German credit data set.

Feature	**VIF**
months_loan_duration	7.3588
amount	4.5758
percent_of_income	7.9587
years_at_residence	7.7354
age	10.9257
existing_loans_count	6.6793
dependents	8.7906

The distribution of data can be considered partial-normal or normal (right skewed) for only three predictors, as seen in Figure 3. The remaining numerical predictors are dichotomous in nature.

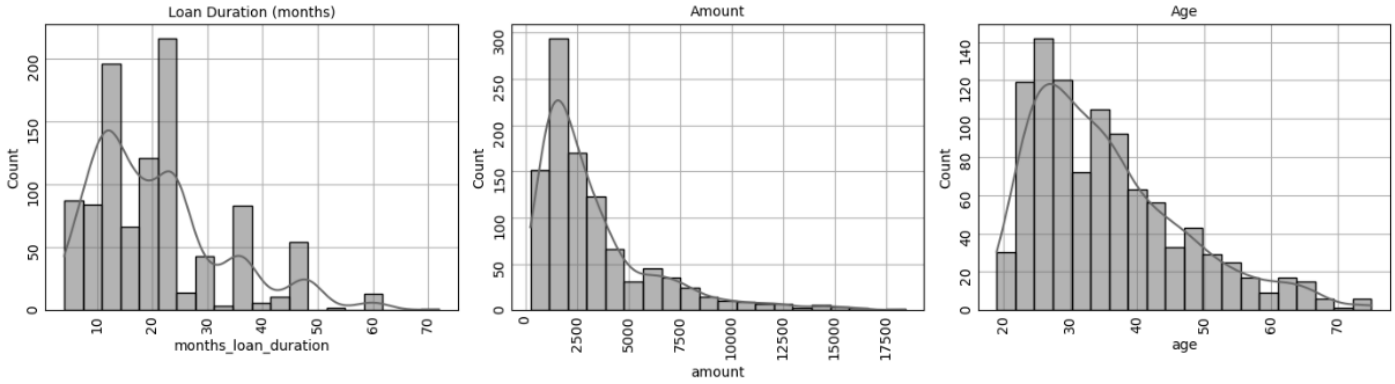

Figure 3. Data distributions—German credit data set.

The final examination confirms that the German credit data set does not contain outliers or missing values.

3.4. Data Pre-Processing

The analytics pipelines correspond in a 1:1 ratio with the permutations of the test scenarios. The first analytics pipeline generates Permutation-1, the second generates Permutation-2, etc. The goal of having various permutations is to achieve the best model performance for the classifiers.

The analytics pipelines apply standardization, including scaling and normalization to all analytics pipelines in this study. For example, the k-nearest neighbour being a distance-based classifier requires that the features contribute more equally to distance calculation, therefore enhancing model performance. The pipelines ensure that there is no unintentional data leakage between the training and testing data sets.

After standardization, the analytics pipelines perform encoding (ordinal, one-hot and label encodings) for categorical features and class feature followed by imbalanced data treatment using SMOTE as investigated by Alam et al. (2020). Subsequently, the analytics pipelines reduce the dimensions of features to the system default and a preset number, respectively.

Prior to model training and fitting, the analytics pipelines implement a manual split of training/testing (80:20) data sets with stratification to ensure the imbalanced data ratio is intact. In search for the optimal hyper-parameters, the grid search function performs the 10-fold cross validations where data is split and internally evaluated for each fold.

3.5. Model Construction and Evaluation

During the model construction phase, the analytics pipeline includes the five classifiers (k-nearest neighbour, naïve-bayes, decision tree, logistic regression and random forest). The final prepared and split data, after being fully cleansed, standardized and encoded, has become a training source to fit the models. The only distance-based classifier used in this study is k-nearest neighbour. K-nearest neighbour is a simple, non-parametric classifier that is not subservient to the Gaussian distributions and is robust to outliers. The curse of dimension takes effect with k-nearest neighbour in that it poses two challenges: (i) it increases computational challenges when high dimensions and large data sets are involved and (ii) it degrades model accuracy when including irrelevant features.

The two tree-based methods are decision tree and random forest. Similar to k-nearest neighbour, decision tree is a tree-based classifier and can manage non-linearity and outliers well. Decision tree has an inherent ability to be unaffected by non-related features. However,

the downside is that it tends to overfit. In this study, the analytics pipeline considers the tree-depth hyper-parameter to ensure that the decision tree classifier does not overfit. Random forest as an ensemble method inherits the strength from decision tree. Additionally, unlike decision tree, it aggregates the prediction of multiple decision trees and offsets the tendency to overfit.

Logistic regression and naïve-bayes are the only two parametric approaches used in this study. That said, they are susceptible to independence assumptions, non-linearity and outliers. In addition, imbalanced data affects naïve-bayes predictions. The analytics pipeline includes the data pre-processing to ensure the training is conducive to fit using the five classifiers, in particular, the parametric ones.

Table 5 illustrates the characteristics of the classifiers implemented in this study.

Table 5. High level characteristics of classifiers.

Classifier	Type	Dependence (H/L) and Tolerance (H/L) for Various Characteristics								
		Dependence					Tolerance			
		C1	C2	C3	C4	C5	C6	C7	C8	C9
k-nearest neighbour (knn)	NP—DB	L	L	L	L	L	L	L	L	B
naïve-bayes (nb)	P—PB	L	H	L	L	H	H	L	H	S
decision tree (dt)	NP—TB	L	L	L	L	H	H	H	L	S
logistic reg. (lr)	P—PB	L	H	L	H	L	L	L	L	B
random forest (rf)	NP—TB	L	L	L	L	H	H	H	L	S

Note: L—low, H—high. P—parametric, NP—non-parametric. DB—distance-based, TB—tree-based, PB—probability-based. B—bigger size data, S—smaller size data. C1—normality, C2—independence, C3—homoscedasticity, C4—linearity. C5—outliers, C6—multicollinearity, C7—irrelevant features. C8—imbalanced class, C9—minimum sample size required for stable estimates.

Whilst these dependency and tolerance level are common in the statistical and machine learning techniques, not all analyses require these assumptions to be met. Under certain conditions, there are methods to relax these dependencies and increase tolerance for the classifiers. Blatant ignorance of the requirements based on each classifier's characteristic will result in poor and unreliable models.

As far as model measurement is concerned, Han et al. (2022) outlined the limitations of relying only on the rate of error as the default measurement as suggested by Jain et al. (2000) and Nelson et al. (2003). Since most of dataset one (Taiwan credit card client) is made up of non-risky value (77.88%), the error rate measurement is not appropriate as it is insensitive to the classification accuracy. The main measurement in this study is AUC despite the fact that Lobo et al. (2008) asserted that area under the receiver operating characteristic (ROC) curve, known as AUC, has its own limitations. Furthermore, the study includes error rate measurement which includes accuracy, precision, recall (sensitivity) and F1 score for the sake of completeness.

Table 2 shows the four error rate-related measurements in model evaluations:

- Accuracy provides the proportion of correctly classified instances from the total instances.
- Precision provides the ratio of true positive predictions versus the total number of positive predictions made.
- Recall provides the proportion of actual positives correctly predicted by the model.
- F1 provides a balance mean of precision and recall that deals with imbalanced class.

Table 6 depicts the performance metrics used throughout the model comparisons.

Table 6. Performance measurements.

Evaluation Metrics	Formula
Accuracy Score	$\frac{TP+TN}{TP+FN+TN+FP}$
Precision Score	$\frac{TP}{TP+FP}$
Recall Score (Sensitivity)	$\frac{TP}{TP+FN}$
F1 Score	$\frac{2*(Precision*Recall)}{(Precision+Recall)}$

Note: *N*—sample size, *TP*—true positive, *FN*—false negative, *TN*—true negative, *FP*—false positive.

All the scores used in this study are based on prediction results.

4. Results

The results of the experiments are made up of performance metrics in tabular and graph formats as well as variables of importance in graph format. The results compare the performances based on the two distinct data sets with three permutations of analytics pipelines. Due to the stochastic nature of classifiers, the results differ slightly for each run. The red dotted line for ROC-AUC graphs represents a random guess for random guess.

The analytics pipelines produce three permutations in search of the best performing classifiers with their respective hyper-parameters. Tables 7–9 and Figures 4–6 list the performance metrics for various permutations. All permutations include data cleansing and standardization:

- Permutation-1—with or without SMOTE using the default hyper-parameters and scoring using full features (all predictors).
- Permutation-2—with or without SMOTE using the best hyper-parameters and scoring using full features (all predictors).
- Permutation-3—with or without SMOTE using the best hyper-parameters and scoring using reduced features (best performing predictors).

Table 7. German credit card dataset—Permutation-1.

	Default Hyper-Parameters									
	Full Features (without SMOTE)					Full Features (with SMOTE)				
	M1	**M2**	**M3**	**M4**	**M5**	**M1**	**M2**	**M3**	**M4**	**M5**
knn	0.7000	0.5000	0.3333	0.4000	0.6746	0.6550	0.4536	0.7333	0.5605	0.7073
nb	0.6950	0.4912	0.4667	0.4786	0.7305	0.7100	0.5111	0.7667	0.6133	0.7408
dt	0.6550	0.4211	0.4000	0.4103	0.5821	0.6450	0.4068	0.4000	0.4034	0.5750
lr	0.7500	0.6250	0.4167	0.5000	0.7679	0.7300	0.5366	0.7333	0.6197	0.7630
rf	0.7800	0.7222	0.4333	0.5417	0.7748	0.7500	0.6087	0.4667	0.5283	0.7618

Note: M1—accuracy score, M2—precision score, M3—recall score, M4—F1 score, M5—AUC.

Table 8. German credit card dataset—Permutation-2.

	Best Hyper-Parameters									
	Full Features (without SMOTE)					Full Features (with SMOTE)				
	M1	**M2**	**M3**	**M4**	**M5**	**M1**	**M2**	**M3**	**M4**	**M5**
knn	0.7700	0.6667	0.4667	0.5490	0.8227	0.8550	0.6824	0.9667	0.8000	0.9670
nb	0.7200	0.5303	0.5833	0.5556	0.7420	0.7300	0.5333	0.8000	0.6400	0.7614
dt	0.7650	0.6857	0.4000	0.5053	0.7726	0.8250	0.6667	0.8333	0.7407	0.9115
lr	0.7750	0.6596	0.5167	0.5794	0.7979	0.7350	0.5412	0.7667	0.6345	0.7944
rf	0.8900	0.9318	0.6833	0.7885	0.9868	0.8900	0.8519	0.7667	0.8070	0.9457

Note: M1—accuracy score, M2—precision score, M3—recall score, M4—F1 score, M5—AUC.

Table 9. German credit card dataset—Permutation-3.

	Best Hyper-Parameters									
	Reduced Features (without SMOTE)					Reduced Features (with SMOTE)				
	M1	**M2**	**M3**	**M4**	**M5**	**M1**	**M2**	**M3**	**M4**	**M5**
knn	0.765	0.6857	0.4000	0.5053	0.8221	0.770	0.5761	0.8833	0.6974	0.8801
nb	0.700	0.5000	0.1833	0.2683	0.6912	0.635	0.4253	0.6167	0.5034	0.6792
dt	0.720	0.5909	0.2167	0.3171	0.7274	0.770	0.6094	0.6500	0.6290	0.8206
lr	0.695	0.4762	0.1667	0.2469	0.6855	0.635	0.4316	0.6833	0.5290	0.6879
rf	0.875	0.9730	0.6000	0.7423	0.9677	0.835	0.6957	0.8000	0.7442	0.9141

Note: M1—accuracy score, M2—precision score, M3—recall score, M4—F1 score, M5—AUC.

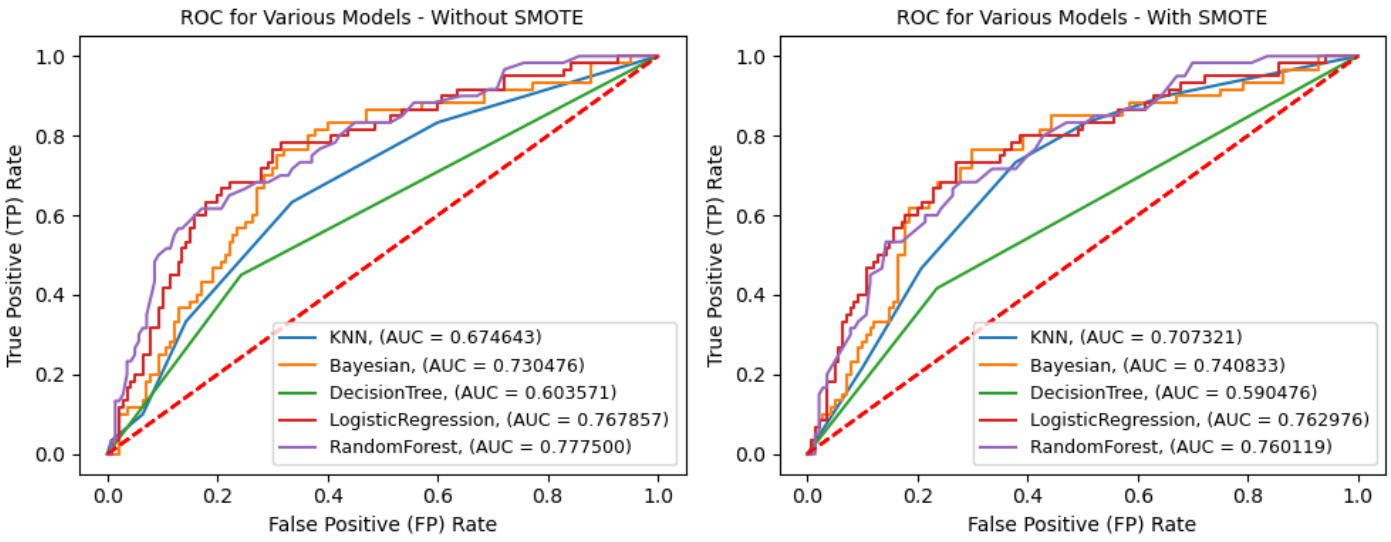

Figure 4. ROC for default hyper-parameters (full features)—German credit data set (Permutation-1).

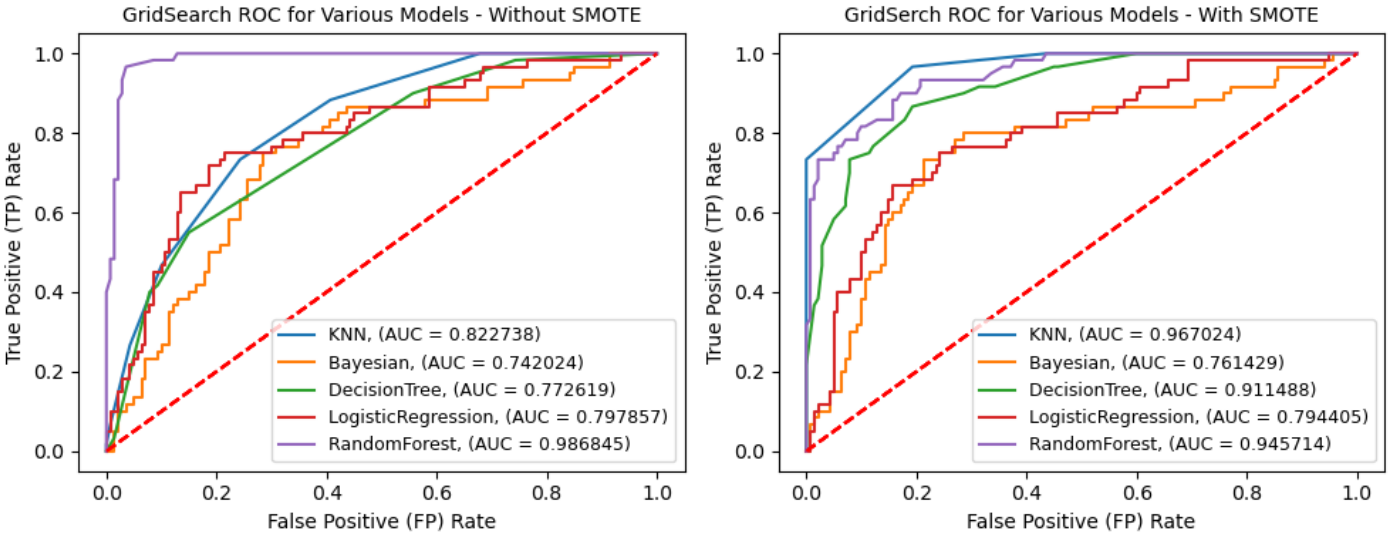

Figure 5. ROC for best hyper-parameters (full features)—German credit data set (Permutation-2).

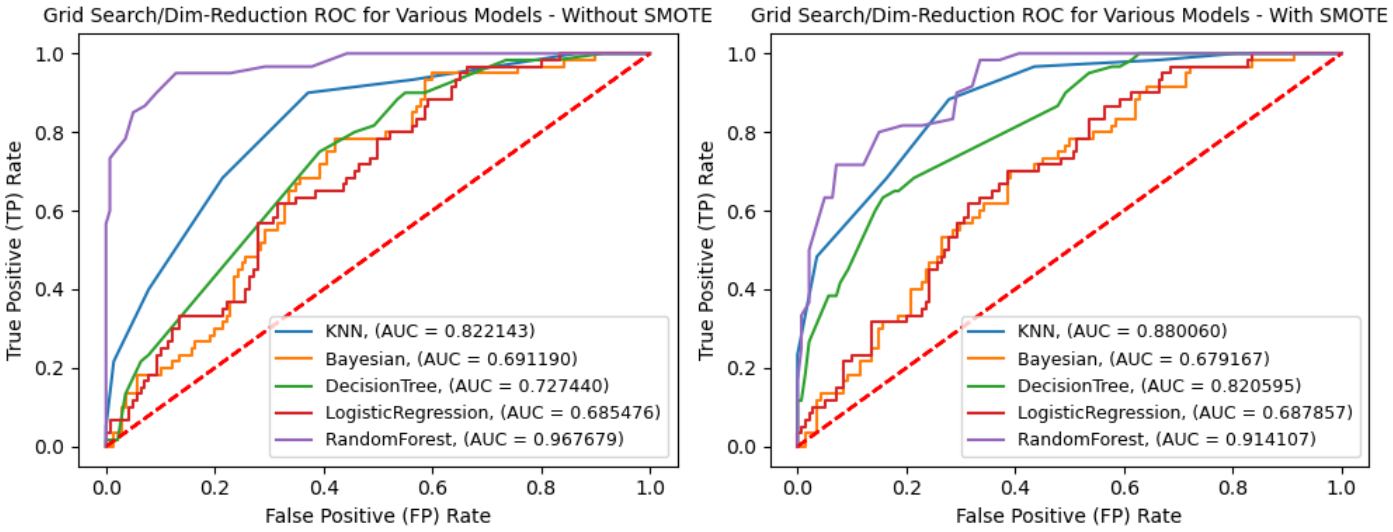

Figure 6. ROC for best hyper-parameters (reduced features)—German credit data set (Permutation-3).

The tables and figures summarize the results:

- Table 7 and Figure 4—Permutation-1 by using German credit data set with the default hyper-parameters and full features.
- Table 8, Figures 5 and 7—Permutation-2 by using German credit data set with the best hyper-parameters and full features.
- Table 9, Figures 6 and 8—Permutation-3 by using German credit data set with the best hyper-parameters and reduced features.
- Table 10, Figures 9 and 10—Permutation-2 using Taiwan credit data set with the best hyper-parameters and full features.

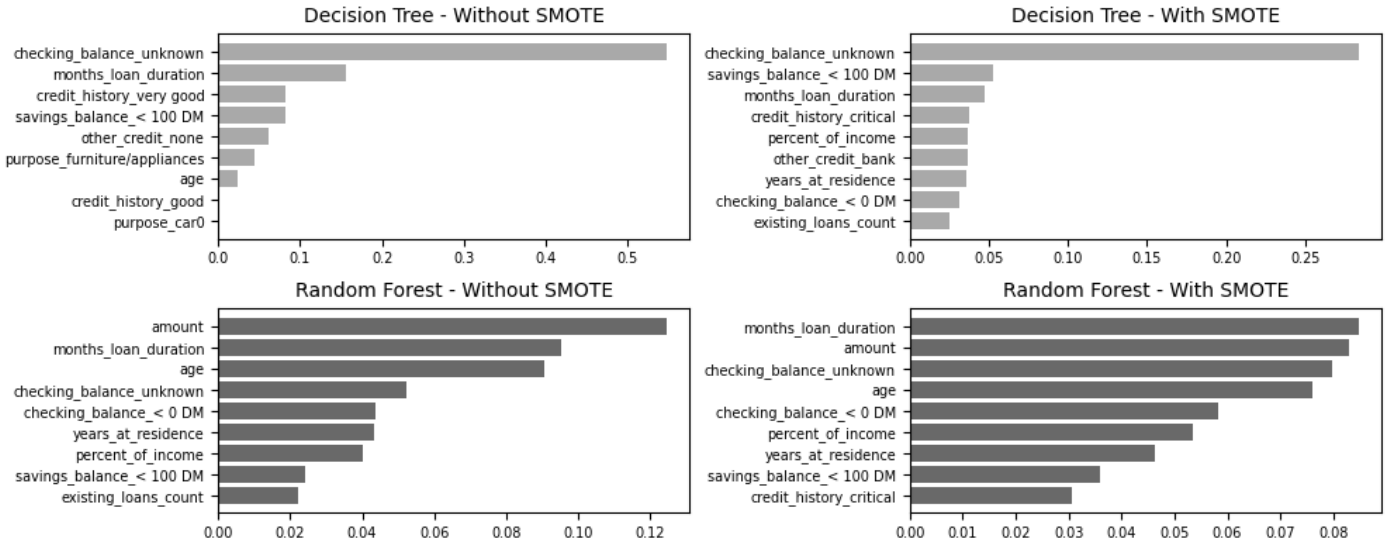

Figure 7. Variable of importance for best hyper-parameters (full features)—German credit data set (Permutation-2).

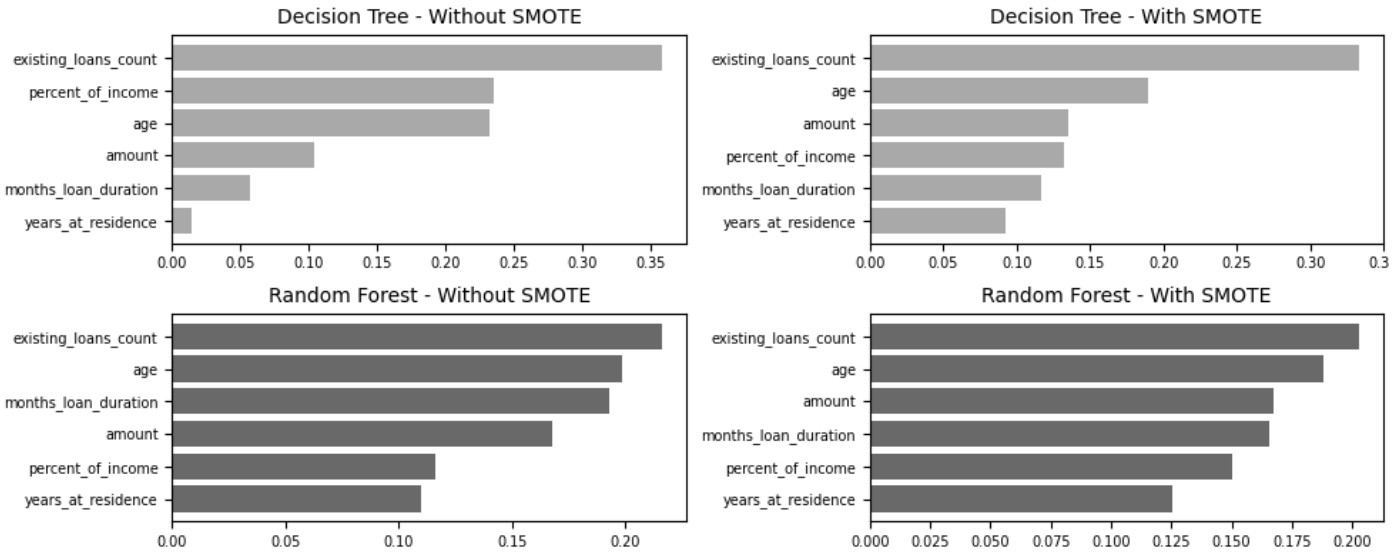

Figure 8. Variable of importance for best hyper-parameters (reduced features)—German credit data set (Permutation-3).

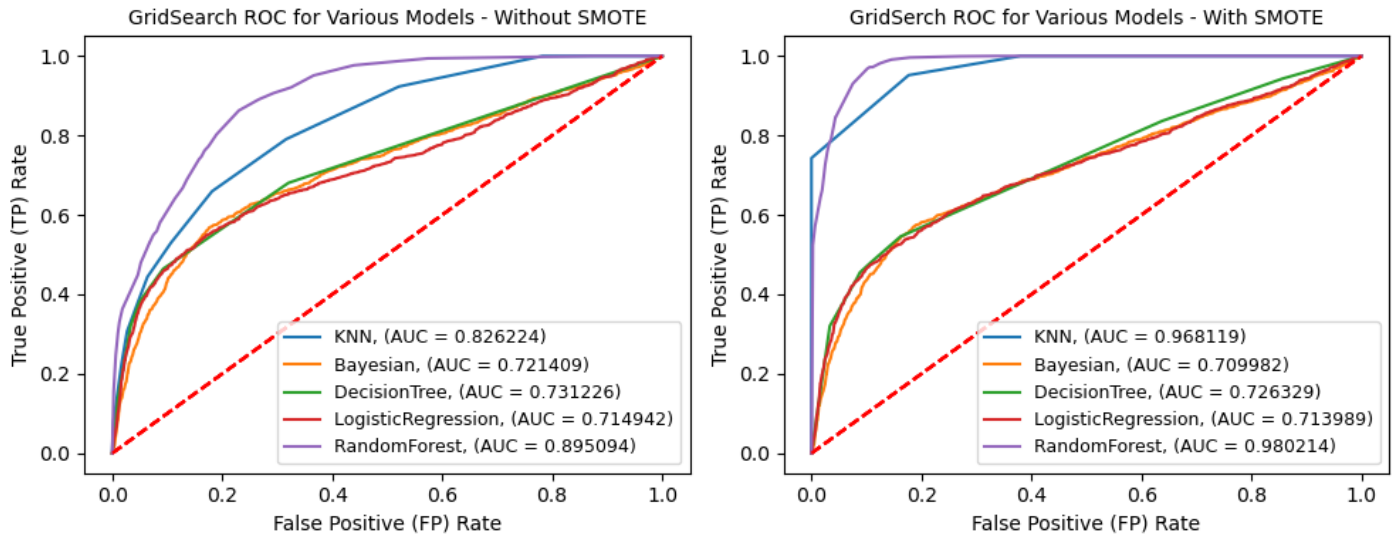

Figure 9. ROC for best hyper-parameters (full features)—Taiwan credit data set.

Table 10. Taiwan credit card dataset—performance measurement.

	Best Hyper-Parameters									
	Full Features (without SMOTE)					**Full Features (with SMOTE)**				
	M1	**M2**	**M3**	**M4**	**M5**	**M1**	**M2**	**M3**	**M4**	**M5**
knn	0.8265	0.7061	0.3693	0.4849	0.8262	0.8517	0.6046	0.9518	0.7395	0.9681
nb	0.7485	0.4479	0.5893	0.5089	0.7214	0.3915	0.2484	0.8644	0.3859	0.7100
dt	0.8243	0.6842	0.3821	0.4903	0.7312	0.7733	0.4889	0.5456	0.5157	0.7263
lr	0.8135	0.7441	0.2389	0.3617	0.7149	0.6860	0.3759	0.6360	0.4726	0.7140
rf	0.8433	0.7706	0.4152	0.5397	0.8951	0.9292	0.8292	0.8561	0.8424	0.9802

Note: M1—accuracy score, M2—precision score, M3—recall score, M4—F1 score, M5—AUC.

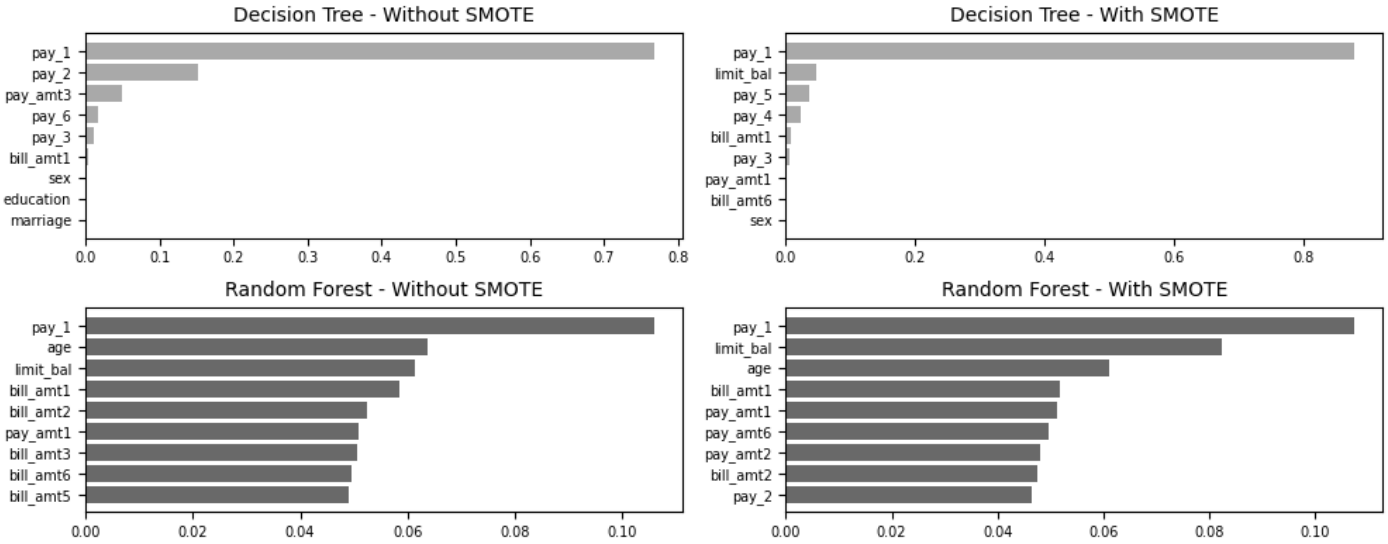

Figure 10. Variable of importance for best hyper-parameters (full features)—Taiwan credit data set.

The best hyper-parameters (with or without SMOTE) obtained for various classifiers can be seen below:

- K-nearest neighbour—{'kn__n_neighbours': 7}
- Naïve-bayes—{'nb__priors': None, 'nb__var_smoothing': 1×10^{-9}}
- Decision tree—{'dt__max_depth': 3, 'dt__splitter': 'best'}
- Logistic regression—{'lr__C': 100, 'lr__max_iter': 1000}
- Random forest—{'rf__max_features': 'sqrt', 'rf__max_samples': 0.3, 'rf__n_estimators': 100}

The three permutations are graphical depictions of the performance metrics for various classifiers.

The last two permutations identify most significant predictors (features of importance) for decision tree and random forest classifiers. The other classifiers produce comparable results.

This study also involves a control data set (Taiwan credit card client) with larger data (30,000 observations). The same analytics pipelines containing data transformations are applied to the data with an identical split ratio, namely 80:20. The results of searching for the best hyper-parameters with full features, ROC graph and most influential predictors can be seen in Table 7, Figures 9 and 10 respectively.

First, it is observed that the default hyper-parameters perform poorly in both smaller (Table 7) and bigger data (Table 10) sets. For example, the five metrics (accuracy, precision, recall, F1 and AUC) hover below 0.8000 for the German credit data set. This shows that the default hyper-parameters are not sufficiently tuned to uncover the hidden pattern in both data sets. Using AUC as a more robust measurement, it is shown that k-nearest neighbour and decision tree are the two worst performing classifiers with 0.6746 and 0.5821, respectively, with untreated and imbalanced data. Interestingly, none of the five classifiers perform better when imbalanced data is treated with SMOTE. Only k-nearest neighbour improves marginally.

Much can be said regarding the full features from the German credit data set being subjected to the best hyper-parameters search. Table 8 shows vast improvements for all five classifiers. Without SMOTE, naïve-bayes is the worst performing classifier with modest improvement alongside logistic regression. However, k-nearest neighbour, decision tree and random forest improve greatly. With imbalanced treatment, k-nearest neighbour improves further. However, the greatest improvement is decision tree which jumps from 0.7726 to 0.9115 followed by k-nearest neighbour which leaps from 0.8227 to 0.9670. What is worth noting, however, is that random forest degrades slightly from 0.9868 to 0.9457. In Permutation-2, both naïve-bayes and logistic regression are indifferent regardless of the inclusion imbalanced data treatment in data pre-processing. The relevant features check

(Figure 7) using the built-in features for decision tree and random forest show the few key features are primarily between "checking balance", "amount", "months loan duration", "age", "percent of income" and "years of residence."

Permutation-3 (Table 9, Figures 6 and 8) differs with Permutation-2 in that it further reduces most relevant features from nine to five, where Permutation-2's nine features are selected by system whilst Permutation-3 is configured to take the best five features. The results are consistent as naïve-bayes and logistic regression are indifferent to imbalanced data treatment whilst k-nearest neighbour and decision tree show big improvements from 0.8221 to 0.8801 and 0.7274 to 0.8206, respectively. The performance for random forest, however, degrades slightly from 0.9677 to 0.9141. In general, the performance of models using the five most relevant features are less optimal than the nine selected by the system.

Finally, comparing with a larger Taiwan credit data set and Permutation-2 (winner), Table 10 and Figures 9 and 10 show that k-nearest neighbour and random forest are the two best performing classifiers across the two data sets. Decision tree performs well in the smaller German credit data set but worse when data is on a larger scale, as in the Taiwan credit data set. Naïve-bayes and logistic regression are indifferent to either smaller or larger data sets, with or without imbalanced data treatment.

5. Discussion

The three analytics pipeline permutations used to construct the five models based on five classifiers contain data cleansing and standardization. It is worth noting that the German credit data set contains categorical features and class labels that require encoding, whilst the Taiwan credit data set contains only numeric features.

The main observations and possible explanations of model performance can be summarized as follows:

- Using the default hyper-parameters for the five classifiers does not necessarily produce the best performing metrics. As data sets have distinctive characteristics such as total observations, complexity and patterns, it is rarely the best practice to use the default hyper-parameters settings until certain tuning is implemented based on each data set.
- K-nearest neighbour and random forest perform consistently well across both data sets either with or without imbalanced data treatment. However, k-nearest neighbour's execution time is a great magnitude faster than random forest. It is likely that random forest requires more processing power and time due to the fact that it is a form of ensemble.
- Naïve-bayes and logistic regression are indifferent to the volume of data sets and imbalanced data treatment, and their performances are mediocre. It can also be attributed to the sub-optimal hyper-parameters selected.
- Decision tree performs really well, which is at par with decision tree and random forest in a smaller data set. However, with imbalanced data, it performs as the worst classifier when a bigger Taiwan credit data set is used. It is highly likely that decision tree overfits with smaller training and testing sets. Moreover, it can be seen that with a smaller data set, decision tree achieves a high AUC score after imbalanced data treatment. This is consistent with its characteristic of being sensitive to imbalanced data.
- The data processing step detects multicollinearity in Taiwan credit data set with all features, "bill_amtX"'s VIF above 20 (between 20.8453 and 38.2155). The presence of multicollinearity in this data set affects only k-nearest neighbour and logistics regression.
- All analytics pipelines with various permutations include standardization which includes scaling and normalization. Whilst k-nearest neighbour requires and benefits from standardization, naïve-bayes, logistic regression, decision tree and random forest are robust to standardization.
- Forcing the classifiers to pick a smaller set of relevant features will degrade the model performance. This results in insufficient data, which will often affect model accuracy.
- In various scenarios, hyper-parameter selection will determine model performance degradation or remain similar after imbalanced data treatment. It is important to

note that apart from naïve-bayes classifier, which is based on probability, all other classifiers are susceptible to imbalanced data. Due to the limitations of computing resources, obtaining the best hyper-parameters for the larger Taiwan credit data set is not achievable.

6. Further Studies

It is desirable to further study the effect to the classifiers using more refined analytics pipelines such as the inclusion of non-standardized (non-scaling and non-normalizing) approach that includes outliers and missing values in the data sets. The effect of multi-collinearity to various classifiers can be explored further. The concept of volatility introduced by Zelaya (2019), which involves including/excluding specific steps in the analytics pipelines, requires further study.

Apart from SMOTE, the use of other treatments suggested by Alam et al. (2020) such as random oversampling, ADASYN, k-means SMOTE, borderline-SMOTE and SMOTE-Tomek is worth exploring since most classifiers, except naïve-bayes, perform sub-optimally with the presence of imbalanced data. Despite the use of truncated singular-value decomposition (truncatedSVD) in the study, the analytics pipelines will also benefit from exploring other dimension reduction techniques including principal component analysis (PCA), t-distributed stochastic neighbour embedding (t-SNE) and linear discriminant analysis (LDA). Further study will benefit by delving into the decision-making criteria used to determine the most relevant predictors for each classifier. Further explorations should be conducted for classifiers using neural network, support vector machine and other modern ensemble techniques such as gradient boosting, extreme gradient boosting and light gradient boosting.

Finally, since achieving the best hyper-parameters for classifiers is key part of the study, it will be worthwhile to include more computing resources to search for the most optimal hyper-parameters for various classifiers. A failure to obtain sufficient system resources will produce sub-optimal hyper-parameters.

7. Conclusions

This study highlights the distinctive characteristics of the five classifiers and how they perform under different data pre-processing steps. The data pre-processing in this study includes data cleansing, features encoding and selection, reduction of dimensions, treatment of imbalanced data and cross validation of training/testing data sets. The final comparisons of the five classifiers demonstrate that data pre-processing steps in conjunction with the data size, complexity and patterns will determine the accuracy of certain classifiers. For example, decision tree performs superbly (overfits) when data size is minor compared to its poor performance when data volume is large. In contrast, the study also shows that random forest does not tend to overfit even with the presence of imbalanced data. In short, the study demonstrates that data distribution and size, multicollinearity, features relevance and imbalanced class contribute to the final scores of models and each classifier reacts to these factors differently (Table 5).

Equally important is the tuning of the hyper-parameters for respective classifiers, with the study concluding that the default hyper-parameters perform sub-optimally. That being said, investing in computing resources to derive the best hyper-parameters is crucial for striving towards the best performing models and achieving cost savings for lending institutes.

Finally, this study concludes that it is mandatory to apply data domain knowledge prior to selecting a classifier of choice. This is primarily due to fact that a data set may have a pattern that suits one classifier but not the other. Hence, it is imperative to understand by unravelling the complexity and patterns of data sets prior to selecting, training and fitting a model.

Author Contributions: Conceptualization, R.L. and H.D.; methodology, R.L. and H.D.; software, R.L. and H.D.; validation, R.L. and H.D.; formal analysis, R.L., H.D. and A.W.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Acknowledgments: Sincere thanks to Ong Seng Huat, UCSI, Malaysia and Dat Tran, University of Canberra, for their constructive comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

Alam, Talha Mahboob, Kamran Shaukat, Ibrahim A. Hameed, Suhuai Luo, Muhammad Umer Sarwar, Shakir Shabbir, Jiaming Li, and Matloob Khushi. 2020. An investigation of credit card default prediction in the imbalanced datasets. *IEEE Access* 8: 201173–98. [CrossRef]

Altman, Edward I. 1968. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance* 23: 589–609. [CrossRef]

Atiya, Amir F. 2001. Bankruptcy prediction for credit risk using neural networks: A survey and new results. *IEEE Transactions on Neural Networks* 12: 929–35. [CrossRef]

Beaver, William H. 1966. Financial ratios as predictors of failure. *Journal of Accounting Research* 4: 71–111. [CrossRef]

Black, Fischer, and Myron Scholes. 1973. The pricing of options and corporate liabilities. *Journal of Political Economy* 81: 637–54. [CrossRef]

Breiman, Leo. 2001. Random forests. *Machine Learning* 45: 5–32. [CrossRef]

Breiman, Leo, and Adele Cutler. 1993. A deterministic algorithm for global optimization. *Mathematical Programming* 58: 179–99. [CrossRef]

Chawla, Nitesh V., Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research* 16: 321–57. [CrossRef]

Chen, Tianqi, and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. Paper presented at the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13–17; pp. 785–94.

Christoffersen, Peter. 2011. *Elements of Financial Risk Management*. Cambridge, MA: Academic Press.

Deakin, Edward B. 1972. A discriminant analysis of predictors of business failure. *Journal of Accounting Research* 10: 167–79. [CrossRef]

Desai, Vijay S., Jonathan N. Crook, and George A. Overstreet, Jr. 1996. A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research* 95: 24–37. [CrossRef]

Feldman, David, and Shulamith Gross. 2005. Mortgage default: Classification trees analysis. *The Journal of Real Estate Finance and Economics* 30: 369–96. [CrossRef]

Fernández, Alberto, Salvador Garcia, Francisco Herrera, and Nitesh V. Chawla. 2018. SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. *Journal of Artificial Intelligence Research* 61: 863–905. [CrossRef]

Freund, Yoav, and Robert E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* 55: 119–39. [CrossRef]

Friedman, Jerome H. 2001. Greedy function approximation: A gradient boosting machine. *Annals of Statistics* 29: 1189–232. [CrossRef]

Han, Jiawei, Jian Pei, and Hanghang Tong. 2022. *Data Mining: Concepts and Techniques*. Burlington: Morgan Kaufmann.

Hofmann, Hans. 1994. Statlog (German Credit Data). *UCI Machine Learning Repository*. [CrossRef]

Huang, Zan, Hsinchun Chen, Chia-Jung Hsu, Wun-Hwa Chen, and Soushan Wu. 2004. Credit rating analysis with support vector machines and neural networks: A market comparative study. *Decision Support Systems* 37: 543–58. [CrossRef]

Jain, Anil K., Robert P. W. Duin, and Jianchang Mao. 2000. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22: 4–37. [CrossRef]

Ke, Guolin, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30*. Montreal: Curran Associates.

Khandani, Amir E., Adlar J. Kim, and Andrew W. Lo. 2010. Consumer credit-risk models via machine-learning algorithms. *Journal of Banking & Finance* 34: 2767–87.

Lobo, Jorge M., Alberto Jiménez-Valverde, and Raimundo Real. 2008. AUC: A misleading measure of the performance of predictive distribution models. *Global Ecology and Biogeography* 17: 145–51. [CrossRef]

Martin, Daniel. 1977. Early warning of bank failure: A logit regression approach. *Journal of Banking & Finance* 1: 249–76.

Nelson, Benjamin J., George C. Runger, and Jennie Si. 2003. An error rate comparison of classification methods with continuous explanatory variables. *IIE Transactions* 35: 557–66. [CrossRef]

Noor, Jamal A. Mohamed, and Ali I. Abdalla. 2014. The Impact of financial risks on the firms' performance. *European Journal of Business and Management* 6: 97–101.

Ohlson, James A. 1980. Financial ratios and the probabilistic prediction of bankruptcy. *Journal of Accounting Research* 18: 109–31. [CrossRef]

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, and Vincent Dubourg. 2011. Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research* 12: 2825–30.

Peng, Yi, Guoxun Wang, Gang Kou, and Yong Shi. 2011. An empirical study of classification algorithm evaluation for financial risk prediction. *Applied Soft Computing* 11: 2906–15. [CrossRef]

Vapnik, Vladimir. 1999. *The Nature of Statistical Learning Theory*. Berlin and Heidelberg: Springer Science & Business Media.

Varoquaux, Gaël, Lars Buitinck, Gilles Louppe, Olivier Grisel, Fabian Pedregosa, and Andreas Mueller. 2015. Scikit-learn: Machine learning without learning the machinery. *GetMobile: Mobile Computing and Communications* 19: 29–33. [CrossRef]

West, David. 2000. Neural network credit scoring models. *Computers & Operations Research* 27: 1131–52.

West, Robert Craig. 1985. A factor-analytic approach to bank condition. *Journal of Banking & Finance* 9: 253–66.

Yeh, I-Cheng. 2016. Default of credit card clients. *UCI Machine Learning Repository* 10: C55S3H. [CrossRef]

Yeh, I-Cheng, and Che-hui Lien. 2009. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications* 36: 2473–80. [CrossRef]

Yobas, Mumine B., Jonathan N. Crook, and Peter Ross. 2000. Credit scoring using neural and evolutionary techniques. *IMA Journal of Management Mathematics* 11: 111–25. [CrossRef]

Zelaya, Carlos Vladimiro González. 2019. Towards explaining the effects of data preprocessing on machine learning. Paper presented at the 2019 IEEE 35th International Conference on Data Engineering (ICDE), Macao, China, April 8–11; pp. 2086–90.

Zhang, Guoqiang, Michael Y. Hu, B. Eddy Patuwo, and Daniel C. Indro. 1999. Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis. *European Journal of Operational Research* 116: 16–32. [CrossRef]