Annals of Operations Research (2023) 330:609–637
https://doi.org/10.1007/s10479-021-04114-z

ORIGINAL RESEARCH

Credit risk classification: an integrated predictive accuracy algorithm using artificial and deep neural networks

Mohammad Mahbobi[1] · Salman Kimiagari[2] · Marriappan Vasudevan[3]

Accepted: 7 May 2021 / Published online: 1 July 2021

Abstract

This study utilizes classification models to provide a robust algorithm for imbalanced data where the minority class is of the interest, that is, in the context of default payments. In developing an integrated predictive accuracy algorithm, this study proposes machine learning classifiers and applies DNN, SVM, KNN, and ANN. The proposed algorithm utilizes a 30,000 imbalanced dataset to improve the accuracy of the prediction of default payments by implementing oversampling and undersampling strategies, such as synthetic minority oversampling technique (SMOTE), SVM SMOTE, random undersampling, and ALL-KNN. The results indicate that the SVM under the ALL-KNN sampling technique is able to achieve an accuracy of 98.6%, with the lowest cross entropy loss measurement of 0.028. Through the accurate implementation of the neural networks and neurons used in the proposed algorithm, this paper presents better insights into the functioning of the neural networks when used in conjunction with the resampling techniques. Using the methodology and algorithm presented in this study, credit risk assessments can be more accurately predicted in practical applications where most of the clients are categorized as non-default payments.

Keywords Machine learning · Classifications · Finance · Credit risk · Sampling techniques · Deep neural network · Artificial neural network · Support vector machines

✉ Mohammad Mahbobi
mmahbobi@tru.ca

Salman Kimiagari
skimiagari@tru.ca

Marriappan Vasudevan
marriappanvasudev17@mytru.ca

[1] Department of Economics, Thompson Rivers University, Kamloops, BC, Canada

[2] Department of Management, International Business, Information and Supply Chain, Thompson Rivers University, Kamloops, BC, Canada

[3] Thompson Rivers University, Kamloops, BC, Canada

1 Introduction

Credit risk refers to the risk of default on a loan due to the failure of a borrower to make required repayments (Lessmann et al., 2015). Credit risks are categorized either for those that repay their debt to the terms of the agreement known as "good" credit risk, or "bad" risks of those that default on their payments. Classification or regression methods are well known to generate a classifier that produces a score for an individual being at such risks (Finlay, 2015). "Credit risk is the probability of an organization or consumer of financial credit instruments defaulting on the debt payment obligation, i.e., the counterparty failure risk" (Basel I, 1988, p. 8). There are numerous standardized ways through which member central banks and regional banks worldwide can mitigate this risk, e.g., those identified by the Basel Committee and Bank of International Settlements. Also, "These techniques include collateralized transactions" (Basel II, 2004, p. 40), "on-balance sheet netting" (Basel II, 2004, p. 42), "guarantees and credit derivatives" (Basel II, p. 42), "maturity mismatches" (Basel II, 2004, p. 42), and collateral against debt obligations. Basel Accord II recommends "forming credit risk control units" (Basel II, 2004, p. 102), i.e., a team internal to the banking operations that can help maintain the ratings of consumers, and thereby maintain oversight on the overall exposure of the bank to credit risk. These teams are likely to produce internal ratings for a given credit approval request, thereby allowing banking officials to decisively take actions for the approval of debt or any type of financial credit instrument. Although banks have already implemented these techniques in their credit risk management procedures, by predicting these risks during the application process or before the customer request, banks can avert any sort of counterparty failure.

The financial credit instruments investigated in this study are credit cards, which have become a common form of payment in the last decade for a range of financial transactions. As per the report published by Payments Canada (2019) on Canadian Payment Methods and Trends, of the total payment transactions that took place in 2018, 28% of the transactions were conducted with credit cards, an increase of 52% from 2017. Data released by the Canadian Bankers Association (CBA) on credit card statistics (CBA, 2018) indicated that the total net dollar value of transactions conducted by VISA and MasterCard holders exceeded CAD 547 billion in 2018. There were 75.8 million cards in circulation in 2018. However, 0.8% of the cardholders were delinquent in their credit card payments, resulting in more than 600,000 credit card delinquency cases in 2018 alone (CBA, 2018). According to the Global Payment reports (2019) published by JP Morgan Chase in the United States, the US has a credit card penetration of 2.01 per capita, and these cards are enabled for e-commerce transactions. The US Federal Reserve Bank's Economic Research published a delinquency rate of 2.59% for Q1 2019, and this rate has been steadily increasing for the past two years, from 2.42% in Q1 2017.

Given the growing trend in payments through credit cards, it can be assumed that the delinquency rate in terms of credit card payments may increase over the coming years. The major reason for the increase in the delinquency rate as per St. Louis Federal Reserve (2019) has been the increased user base of credit cards, especially between the age group of 18–29 years. The delinquency rate among these users in 2019 alone was 8.05%, as per St. Louis Federal Reserve. To understand delinquency, we must consider the definition of default used by banks worldwide. As per Basel Accord II, the definition of default is as follows (Basel II, 2004, p. 104, 105):

> A default is considered to have occurred with regard to a particular obligor when either or both of the two following events have taken place. The bank considers that

> the obligor is unlikely to pay its credit obligations to the banking group in full, without recourse by the bank to actions such as realizing security (if held). The obligor is past due more than 90 days on any material credit obligation to the banking group.

Following the definition of default, the delinquency rate for credit card payment obligations can be calculated as the percentage of defaulters who fail to pay their obligations for more than 90 days. In this study, owing to the limitations of the dataset, a complete definition of delinquency may not be implemented. However, for conducting this study, as the credit instruments being considered are credit cards, default can be considered as when the clients fail to make any payment in the next month by the due date. By predicting and identifying credit card customers who might be defaulting on payments, banks can avoid major losses owing to credit card defaulters. According to the Canadian Bankers Association data on credit card delinquency, the net annualized loss rate for 2019 alone was 3.45%.

According to the McKinsey Bahillo et al. (2016), by implementing adequate measures with advanced analytics to detect credit risks and avert further losses, portfolios can reduce up to 50% of the costs in the credit risk operations of the business. One of the primary capabilities of a robust risk management system is detecting risks earlier. However, many of the bank systems today lack this key capability, leading to further losses (Bahillo et al., 2016). By implementing a system to monitor and address defaulters, banks can avoid losses, which will help save the bank millions of dollars. In our study, these losses are considered as occurring owing to credit card defaults on payments. This leads us to the rationale behind the study, i.e., developing a model using a deep neural network (DNN) architecture that can efficiently help banks identify defaulters and thereby help them save millions of dollars. Identifying and classifying credit card defaulters using machine learning and advanced analytics can help banks and financial institutions detect their risks early in transactions or in a client's portfolio, based on the data available in the system. This will allow banks and financial institutions to implement appropriate measures for credit risk management.

Our major objective of this study is to develop a robust and efficient DNN model using a combination of specific sampling algorithms based on machine learning techniques. This study then conducts a comparative analysis with already established techniques used in credit risk assessment and discussed in the relevant literature, such as support vector machines (SVMs), K-nearest neighbors (KNNs), and artificial neural networks (ANNs). These models have been developed based on the understanding of the current literature, and on techniques already in place for credit risk identification and classification. To conduct this research, we use datasets including the open-source datasets offered by the University of California, Irvine database, which is available for conducting research and for developing such models.

Our inspiration for this research is based on recent advancements in the use of artificial intelligence and machine learning techniques to solve problems faced by the financial industry. The probability of default and classification of defaulters in credit risk assessments have been widely studied using machine learning techniques, but these studies are limited with regard to deep learning techniques. In this study, we propose a six-layer DNN model for analyzing credit risk assessments. We compare it with techniques such as ANN, SVM, and KNNs, which are some of the widely used models for predicting and studying credit risk assessments. This study also considers sampling techniques that can be used with the imbalanced dataset and models, such as the synthetic minority oversampling technique (SMOTE), random undersampling (RUS), SVM-SMOTE, and All-KNN.

This study aims to fill the gaps in the literature. The methodologies presented in this paper will outlay the sampling techniques used to overcome the imbalanced nature of the dataset. Evaluation techniques such as the F1 score and G-Mean are also presented, along with accuracy, sensitivity, and specificity values for the imbalanced dataset.

Section 2 of this paper starts with a thorough literature review to identify the existing gaps followed by an overview of the classification techniques. In Sect. 4, we described materials and methods, and Sect. 5 introduces the framework of the analysis. Empirical results and analysis are discussed in Sect. 6, and in the last section the potential implications and conclusions are explained.

2 Literature review

One of the earliest risks scoring statistical techniques, was developed based on Fisher's Linear Discriminant, LD, model (1936); his seminal paper discussed quantitative techniques for classifying "good" and "bad" applicants. Post-1980, LD techniques were generally replaced by statistical techniques, such as linear regression, logistic regression, and early stage base classifiers such as nearest neighbors and decision trees; these provided significant results, provided that the data were linearly separable. However, if the data sets were not linearly separable, then these techniques proved to be insufficient for credit risk analysis (Chen et al., 2011).

Yu et al. (2010) studied credit risk evaluation using an SVM with a multiagent ensemble learning system. They used credit card applicants from British financial service companies, and increased the number of bad applicants to match the level of good applicants. This allowed them to perform their study on a balanced dataset. As per Yu et al. (2010), the multiagent system with the SVM outperformed logistic regression, quadratic DA, and feedforward NN, but lagged with a multi-agent feedforward NN model. Chen et al. (2011) studied the bankruptcies of German firms using an SVM with a Gaussian kernel. They identified 28 different financial ratios for firms that went bankrupt between 1996 and 2002, and used these ratios as the features for the algorithm. Chen et al. (2011) identified that an SVM outperforms logit classification in terms of classification problems, especially in cases of linearly non-separable datasets.

Trustorff et al. (2011) conducted a similar study using a least-squares SVM and logistic regression models. They chose five debt ratios for identifying the credit risks of companies and, in total, studied 78,000 companies using these ratios. One of the major outcomes of their study was that the SVM performs well under small training samples with high variance in the input data (Trustorff et al. 2011). Both Trustorff et al. (2011) and Chen et al. (2011) overlooked the imbalanced dataset in their studies. To overcome this problem in our study, we used oversampling and undersampling techniques, which will be explained in detail below.

Wang et al. (2012) used a hybrid ensemble approach to provide enterprise credit risk assessments. They used the financial records of 239 companies, as provided by the Industrial and Commercial Bank of China. The method involved bagging and boosting techniques, along with linear and polynomial SVM kernels. However, the dataset used in this study was much smaller than that of the other datasets used in most studies. This lack of application of the methodologies to a large dataset is one of the shortcomings of this research.

Harris studied credit risk assessments in 2013 and 2015, and these studies are of particular interest. These two studies involved the use of SVMs in credit risk assessments. Harris (2015) conducted a study on credit risk assessments based on default definitions given by the Bank of International Settlements and the Base Committee. His study argued that by using "narrow" and "broad" definitions of defaults based on the number of days past due in payments, credit risk evaluations could be improved using quantitative credit risk models. His methodologies, however, lacked clear applications of the credit risk models, along with any sensitivity analysis of the models. His study in 2015 involved the application of the clustered SVM proposed by Gu and Han (2013), and compared it with techniques such as logistic regression, decision trees, and combinations of other techniques. In this study, he used the German credit dataset provided by the UCI Machine learning repository and Barbados credit union dataset.

Cao et al. (2013) proposed a novel model-based cost-sensitive SVM enhanced by the particle swarm optimization technique (PSO) for loan default discrimination. Their research improved the SVM model integrating cost sensitivity and the PSO, thereby increasing the accuracy of the output; however, their model was applied as a binary classification technique to a specific bank data, thereby limiting the application of the model to a wider dataset. The limitation of the model's application to a wider dataset led to question regarding the efficiency and scalability of the model used by Cao et al. (2013), and further research on multi-class multi-feature classification clustering models was suggested to address the shortcomings in their research.

Danenas and Garsva studied the application of SVMs in credit risk assessments for different scenarios, and using different combinations of kernel functions. A recent study (Danenas & Garsva, 2015) on credit risk assessment used an SVM with PSO, as used by Cao et al. (2013). They also utilized financial ratios as the input features for credit risk assessments. In their research, they used the Zmijewksi score (Z-score) as a binary output feature, with companies scoring greater than zero (i.e., $Z > 0$) being labeled as bankrupt. They compared the measurements of the model with those from logistic regression and RBF-based network classifiers. However, limitations regarding the stability of the PSO-based SVM were one of the major limitations of their research. The model did not outperform linear SVM models, as used by other researchers in credit risk assessment.

Henley and Hand (1996) studied using the KNN as a classifier for credit risk scoring techniques, based on considering a bad risk rate as part of their research. The authors identified that KNN performed well in identifying the bad risk rate, and was able to perform well relative to decision trees, logistic regression, and linear regression. The dataset used by Henley and Hand (1996) was fairly balanced, with over 54% of the dataset consisting of credit risk, and involved 16 features. Marinakis et al. (2008) studied a nearest neighbor classifier by using metaheuristic algorithms for credit risk assessment based on loan portfolios of 1411 firms from the Greek Commercial Bank. The authors used 16 different financial ratios, including profitability, solvency, and managerial performance ratios. The dataset included 218 firms with default classes, and 1193 firms with non-default classes (Marinakis et al., 2008), making it an imbalanced dataset; nevertheless, their research did not involve any techniques to transform the imbalanced dataset into a balanced one. Using the metaheuristics algorithms, some of the models were able to achieve more than 98% accuracy, with an overall average of between 94 and 97%.

Abdelmoula (2015) studied Tunisian bank credit risk using the KNN algorithm with three nearest-neighbor parameters. The dataset consisted of 924 credit records from 2003 to 2006, held by a Tunisian commercial bank (Abdelmoula, 2015). Abdelmoula (2015) obtained an accuracy of 88.63% with a receiver operating characteristic (ROC) score of over 95%. The author used over 24 financial and non-financial ratios as the features of the study, with cash flow and non-cash flow models. Abdelmoula (2015) also used Type 1 and Type 2 error rates for credit risks and commercial risks to identify whether the models could cover these error rates, which would help banks make efficient risk management decisions. A Type 1 error rate indicates the rate of default customers being categorized as non-default customers, and a Type 2 error indicates the rate of non-default customers being categorized as default customers (Abdelmoula, 2015). Concerning the methodology, although the author used ROC as the main performance metric, there was no discussion regarding the imbalanced nature of the dataset. Nevertheless, to the best of our knowledge, Abdelmoula's (2015) research is one of the highest-quality studies on the use of KNN for credit risk assessments.

Khashman (2010) built a credit risk evaluation system with three different NN models using 24 numerical attributes, and implemented it with nine different learning schemes. From 27 different learning models, he chose the three learning models that provided an error rate of less than 0.008; this indicated that efficient models require iterative regression procedures to deliver accurate risk evaluation techniques. These three models delivered an overall accuracy rate of 83.6%, but the research lacked multiple points, such as feature selection procedures (how the clients were chosen for the training and validation procedures). Cimpoeru (2011) introduced the concept of neural calculus, and studied the concepts of error backpropagation techniques in credit risk assessments. The author of this research focused on multiple models, such as feedforward networks with multiple layers, adaptive networks based on fuzzy algorithms, and SVMs. Cimpoeru (2011) conducted a study on Romanian small-medium enterprises with turnover values between EUR 700,000 and EUR 3,755,000. The research was conducted on 2% of the total population as a sample, and the input variables were financial ratios determined based on the available data.

Karaa and Krichene (2012) conducted a similar study by comparing SVM and NN models, and established the superiority of NN models over SVM models. The researchers focused mainly on the historical datasets of companies and their financial ratios. The authors did not mention whether the dataset was imbalanced, or if sampling techniques were used in the research. They achieved an accuracy of 90.2% with the NN model, and a Type 1 error rate of 18.55%. They also provided comparative results between DA and logistic regression techniques, and proved that logistic regression is a better model for resolving classification problems.

Oreski et al. (2012) investigated the extent of the impact that the total data from a single bank has on the genetic algorithm-based NN (GA-NN) for credit risk assessments. Their primary study was based on feature engineering and feature selection using hybrid models of GAs, which improve feature selection for data processing and evaluation relative to other models. Although the research was conducted with far better accuracy, the GA-NN is a computationally intensive technique, and the feature selection process takes a longer time to complete. Implementing this technique in banks will require optimization of the models and internal parameters, because each bank uses a different set of ratios to determine their credit risk assessments for clients. Moreover, the limited application of this model owing to its technology-intensive requirements necessitates further improvements, along with better models for real-world applications.

Khemakhem and Boujelbènea (2015) studied the differences between DA and ANNs based on Tunisian companies, and established that NN models were more accurate in terms of predictability. However, they criticized NN models as being less robust and less well-founded, terming them "black-box" operating rules, as the NN models are unable to explain the results provided by the models used in the studies of Tunisian companies.

Although in many cases, ANNs have provided better results (Oreski et al., 2012; Khemakhem and Boujelbènea, 2015) as compared to linear models in regards to classification, they have been criticized for being vulnerable to multiple minima problems, such as those concerning ordinary least squares and maximum likelihood estimation (Chen et al., 2011). The major reason for this vulnerability is the minimization of empirical risks, leading to poor classification of the sample datasets (Chen et al., 2011; Haykin, 1998). In recent years, several researchers have performed comparative analyses between different models of ANN and machine learning techniques, aiming to understand the shortcomings and to improve the efficiency of such models. Khashman (2010), Cimpoeru (2011), and Karaa and Krichene (2012) conducted research by comparing different models, aiming to understand their impacts on the data and output.

With advancements in machine learning, developments in software languages, and faster processing capabilities in computers, DNN and deep learning architectures have taken center stage in the study of applications relative to predictions and classifications. Sun and Vasarhelyi (2018) studied the application of DNNs to credit card delinquencies, one of the major influences in conducting this study. Based on credit card applicants from one of the largest banks in Brazil, with over 700,000 credit card applicants, they found that deep learning improves the accuracy of prediction in the case of a large dataset. Although they used a novel approach, they lacked a sensitivity analysis and overlooked the imbalanced dataset; moreover, they did not incorporate any types of sampling techniques that might have helped to overcome the imbalanced dataset.

Hamori et al. (2018) studied credit card delinquency based on the same dataset as that used in this study. Their study involved a comparison of ensemble learning methods along with NNs and DNNs with Tanh and ReLU activation functions. They identified that the dataset used was imbalanced, and used a normalization approach for the dataset, rather than sampling techniques. Zhu et al. (2018) introduced the use of a relief algorithm-based CNN for consumer credit scoring. The researchers used consumer credit data from a Chinese consumer finance company, comprising of 24,387 data points and over 570 numeric attributes. Of these 570 numeric attributes, they used 50 attributes related to consumer credit (Zhu et al., 2018). Their study only included the area under the curve (AUC) and F1 score metrics, indicating that the dataset used was highly imbalanced; moreover, their methodology did not include any data normalization or sampling techniques with the NN.

Kvamme et al. (2018) used a CNN to predict mortgage defaults based on consumer's account balance. They used a dataset from the Norwegian Bank, DNB, consisting of 20,989 data points with a time series from 2012 to 2016. Their NN comprised three hidden layers with ReLU activation functions, and one output layer with a SoftMax activation function. To overcome the imbalanced dataset problem and overfitting of the model, they used data augmentation and regularization on both of the CNN models used in their research. Bayraci and Susuz (2019) studied using DNN-based classification models for credit risk assessments of Tunisian financial institutions with two separate datasets. For the datasets regarding credit card applicants, the researchers used a random selection of the major and minor classes to avoid the imbalanced nature of the dependent variable. They found that the DNN works well with complex datasets. However, their research lacked sufficient evaluations of DNN models in terms of the F1 score and AUC; instead, they chose to use the weighted average accuracy rate. Second, the researchers did not specify the activation functions or number of layers used in the DNN model.

Rao et al. (2020) applied a random forest model to manage the borrowing risk of borrowers in the rural areas by applying a two-stage Syncretic Cost-sensitive Random Forest (SCSRF) model of "three rural" borrowers. In their study, no over and undersampling strategies were used. Rtayli and Enneya (2020) used an SVM for identifying credit card risks from a highly imbalanced dataset from European credit card transaction data held by Libre Brussels University. In this study, the authors identified that the SVM had a good accuracy rate of 95% and sensitivity of 87%, but did not incorporate any balancing techniques for analysis. Kalid et al. (2020) used an ensemble learning approach with the same credit card dataset as that used by Rtayli and Enneya (2020), and was able to improve the true positive rate for credit card fraud detection. Their approach also did not incorporate any balancing techniques along with the ensemble learning approach and several machine learning techniques. Based on the literature presented above, SVM has been one of the most studied models in credit risk assessments. This makes it ideal for study, and for comparative research with the DNN model presented in this study.

Sariannidis et al. (2020) compared prediction accuracy of seven classification methods such as KNN, Logistic Regression, Naïve Bayes, Decision Trees, Random Forest, SVC, and Linear SVC. They found that only some of the features can adequately be used to analyze the characteristics of the lending decisions. No use of resampling techniques are applied into the data set.

Several gaps can be outlined from the literature review. Previous research on DNN models has largely overlooked the sampling techniques that can be implemented along with these models. The evaluation of the models has been limited in regards to accuracy; in the case of an imbalanced dataset, it is recommended to use other measures, such as F1 score, G-Mean, and AUC–ROC Curve. Limited research has been completed on comparing the established scoring techniques in the context of SVM and/or DNN models, which could help us to understand whether DNN models have an advantage. Previous research has been limited to presenting the outcomes of the models in terms of their performances; however, limited discussion has been presented on the policy implications from using such models in financial institutions. Table 1 presents literature gaps in applying SVM, KNN models and Table 2 shows the gap related to the use of ANN and DNN models in credit score predictions. The major gap identified in the current reviewed literature is the lack of applying over and undersampling methods in classification of the defaults.

3 Classification techniques and approaches

Post-Great Recession (2008–2009), credit risk identification and prevention have received significant attention from managers of financial institutions, e.g., for issuing debts and lines of credit (Harris, 2015). Regulatory developments following global financial crisis have mandated the performance of complete due diligence on the credit histories of the companies and candidates requesting credit lines. These regulations have initiated the development of a variety of techniques under the credit risk scoring model (e.g., Basel III, 2011). Financial firms and investment banks heavily rely on these scoring techniques to identify defaulters, so that credit lines can be offered to the most legitimate candidates. One of the earliest risk-scoring statistical techniques, discriminant analysis (DA), was developed based on Fisher's linear discriminant model (1936); his seminal paper discussed quantitative techniques for classifying "good" and "bad" applicants.

Table 1 Literature review and relative gaps using SVM and KNN models

Author/authors	Models used	Dataset	Sampling techniques	Gap(s) in the literature	Literature gap(s) filled by this study
Yu et al. (2010)	SVM with Ensemble learning, LogitR, FeedForward Neural Network	Balanced by increasing bad applicants	No	Sampling techniques	Sampling techniques
Chen et al. (2011)	SVM with Gaussian Kernel	Imbalanced	No	Sampling techniques	Sampling techniques
Trustorff et al. (2011)	SVM with Least Squares	Imbalanced	No	Sampling techniques	Sampling techniques
Wang et al. (2012)	SVM with the hybrid ensemble	Smallerc239 instances	No	Smaller dataset	30,000 instances used in this study
Harris (2015)	SVM	Smaller—1000 instances	No	Smaller dataset, Sampling techniques	30,000 instances used in this study, Sampling techniques
Danenas and Garsva (2015)	Particle swarm optimization (PSO)-SVM, SVM	Imbalanced, 24,000 instances	No	Measurements for the imbalanced dataset, receiver operating characteristic (ROC), Sampling techniques	Sampling techniques and better performance measurement techniques
Henley and Hand (1996)	KNN	Balanced	No	Performance measurements	Performance measurements under imbalanced dataset
Marinakis et al. (2008)	KNN	Imbalanced, 1411 instances	No	Sampling techniques, Smaller dataset	Sampling techniques
Abdelmoula (2015)	KNN	N/A, 924 instances	No	No discussion on imbalanced dataset	Sampling techniques
Rtayli and Enneya (2020)	SVM	Imbalanced	No	Sampling techniques	Sampling techniques
Kalid et al. (2020)	SVM	Imbalanced	No	Sampling techniques	Sampling techniques
Sariannidis et al. (2020)	KNN, NB, DT, RF, SVC, and Linear SVC	Balanced	No	Sampling techniques	Sampling techniques

Table 2 Literature review and relative gaps using ANN and DNN models

Author/authors	Models used	Dataset	Sampling techniques	Gap(s) in the literature	Literature gap(s) filled by this study
Yeh and Lien (2009)	ANN	Credit Card	No	Sampling techniques	Performance measurements under imbalanced dataset
Khashman (2010)	ANN	24 Attributes of Financial ratios	No	Performance measurements like ROC, F-Measure	Performance measurements under imbalanced dataset
Oreski et al. (2012)	Genetic algorithm-neural network (GA-NN)	Financial Ratios, No discussion on dataset's nature	No	Technology intensive	DNN model used in this study (Able to run on any laptop with 8 GB ram)
Khemakhem and Boujelbènea (2015)	ANN	Financial Ratios of Tunisian Companies	No	Less robust model	Consistent results obtained by ANN and DNN model used in this study
Sun and Vasarhelyi (2018)	DNN (Layers not mentioned)	Credit Card delinquencies—700,000 instances	No	Overlooked imbalanced dataset	Sampling techniques along with DNN Model
Hamori et al. (2018)	DNN—2 Layers	Same Dataset as used in this study	No	Sampling techniques	Sampling techniques, DNN model used in this study has four hidden layers
Zhu et al. (2018)	Convolutional NN (CNN)	Chinese Consumer Finance Company, 24,387 instances, Imbalanced Dataset	No	Sampling techniques, No model comparison	Sampling techniques, four different models used in this study
Kvamme et al. (2018)	CNN	DNB Bank, 20,989 instances, Augmentation and Regularization for imbalanced nature of the dataset	No	Sampling techniques	Sampling techniques
Bayraci and Susuz (2019)	DNN (Layers not mentioned)	Tunisian Financial Institutions, Random selection of major and minor classes	No	Sampling techniques	Sampling techniques

3.1 Support vector machines (SVMs) with sigmoid and radial basis function (RBF) kernel

SVMs are some of the prominent binary classification machine learning models utilized to resolve classification problems, especially if the dataset consists of binary features (Harris, 2015). The SVM was first developed by Vapnik (2000). It attempts to find the optimal separating hyperplane between binary classes by maximizing the difference between the class margins (Vapnik, 2000; Harris, 2015). The points lying on the boundaries of the hyperplane are called support vectors. The optimal hyperplane is determined by maximizing the width of the margin.

The optimization function in the SVMs for finding the optimal hyperplane is conducted using functions called kernel functions. These functions play a role in finding an optimized solution that is similar to an optimization problem. In this study, the radial basis function (RBF) was used as a kernel function with an SVM model. The RBF can reflect SVMs with exponential functions, whereas sigmoid functions can be taken as functions of the tangents to the input parameters. Table 3 indicates the functional form of the SVM involved in the study, along with its parameters and default values.

The SVM works based on the optimization of the margin between hyperplanes. In this study, for a set of training instances $\left\{\left(\mathrm{x}_1, \mathrm{y}_1\right), \ldots\ldots\ldots\ldots..\left(\mathrm{x}_\mathrm{n}, \mathrm{y}_\mathrm{n}\right)\right\}$, $\mathrm{x} \in \mathrm{R}^\mathrm{n}$, $\mathrm{y} \in \{-1, 1\}$, and y is the class label for the dependent feature in a binary classification problem. In a binary classification problem, the SVM attempts to find a classifier f(x), which minimizes the misclassification rate. f(x) is the hyperplane, and can be represented as $\mathrm{f(x)} = \mathrm{sgn}\left(\mathrm{w}^\mathrm{T}\mathrm{x} + \mathrm{b}\right)$. Using this function in the training results in a convex quadratic optimization problem. The convex optimization problem can be rewritten using Lagrangian functions, as follows Eqs. (1) and (1a):

$$minimize\ W(\alpha) = \frac{1}{2\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j K\left(x_i, x_j\right)} \tag{1}$$

subject to:

$$\sum_{i=1}^{n} y_j \alpha_i = 0;\quad \forall i\ :\ 0 \le \alpha i \le C \tag{1a}$$

Here, α is the Lagrange multiplier, and C is the tradeoff between the maximum margin and misclassification error. The term $K\left(x_i, x_j\right)$ represents the kernel functions used to map linearly non-separable instances into a higher-dimensional space. The kernels used in the study are presented in Table 3.

Table 3 Functions and parameters of the support vector machine (SVM) (Khemakhem and Boujelbènea, 2015)

Kernel function	Functional form	Parameters	Default values
Radial basis function	$K\,(x_i, x_j) = \exp\,(-\,\gamma x_i - x_j\text{^}2)$	$\gamma \in R$	$\gamma = 1$

3.2 K-nearest neighbor (KNN)

The nearest neighbor algorithm is one of the most widely studied algorithms for classification problems. The algorithm was first introduced by Fix and Hodges (1951), in their seminal paper on discriminatory analysis and nonparametric discrimination. They were the first to establish the rules for the nearest neighbors, and how the algorithm identifies them using a Euclidean distance. Cover and Hart introduced the nearest-neighbor algorithm for pattern classification in (1967), and identified how it could fit into broader applications of classification problems. The KNN was introduced by Altman (1992) as a nonparametric method for pattern recognition and classification. This algorithm also belongs to the class of supervised learning techniques, as the algorithm requires training before the actual application of the algorithm on a given set of independent features. It is also a machine learning method that can be extended and applied to large-scale data mining problems (Nadkarni & Nadkarni, 2016). The algorithm uses a common principle, i.e., that similar objects or features exist within the proximity of one another in a given dataset.

As a non-parametric classification technique, KNNs can be used for non-linear datasets, such as in credit risk assessment. In this study, the KNN algorithm was used as a classification technique for identifying default payments in ta dataset. Parameter tuning is a key aspect of the KNN model. One of the most important parameters to be identified for the KNN is the number of nearest neighbors. Using the square root of the total number of samples, we tuned our nearest neighbors to 173, based on our understanding of overfitting and underfitting in regards to the model. Overfitting the model means using excessive data points to fit the data into the model, resulting in plain memorization of the data points by the model (Massaron & Boschetti, 2016), and ultimately in the provision of incorrect measurements for the model prediction. In contrast, underfitting indicates the use of too few data points or too little information to fit the model, thereby not utilizing the complete information for accurately training the model.

3.3 Artificial neural network (ANN)

ANNs consist of neurons that are similar to human neurons. These neurons form a single functional unit in the network layer. An ANN can consist of one-to-many layers, making them easily programmable algorithms in the field of computer science. The mathematical model of a neuron was proposed by McCulloch and Pitts (1943). The neuron proposed by McCulloch and Pitts (1943) consisted of a binary input, binary output, and single activation function. Stacking multiple neurons with a given set of input variables and connecting them with different weights and activation functions provides us with ANNs, or more simply, neural networks (NNs). The most common form of NN is known as the feed-forward network, where the information from the input variables is carried forward linearly through cross-connected neurons as the middle layers, and finally towards the desired output layer. These networks are termed as "feed-forward" because the information flows in only one direction, i.e., without any feedback loops or back into the hidden layers.

Over the past few years, with the help of advanced programming languages, NN research has led to several other architectures, such as error back-propagation NNs, recurrent NNs, and convolutional NNs (CNNs). These have been widely implemented in image processing and image recognition technologies. The ANN in this study was influenced by the work of Khemakhem and Boujelbènea (2015), who used an ANN to conduct a credit risk assessment. The ANN used in this study comprises four layers, as follows:

Layer 1: An input Layer consisting of 10 neurons representing the 10 input variables;
Layer 2: A hidden layer consisting of 109 neurons;
Layer 3: A hidden layer consisting of 109 neurons; and
Layer 4: An output layer consisting of a single neuron.

This study used a rectified linear unit (ReLU) as the activation function for neurons with a feed-forward neural architecture, as explained above. The hidden layer neurons were optimized throughout this study for better accuracy and classification results, based on a trial-and-error method. The choice of neurons in the hidden layer was decided based on a common assumption, to form a tunnel architecture in the network topology of the NNs, and thereby to reduce the error rates in the NNs. Combined with this assumption and by using multiple trials to avoid overfitting of the models, the neurons were appropriated at 16 and 10 for the hidden layers in the ANN architecture. A similar method was used to finalize the architecture of the DNN model. We used binary cross-entropy as the loss function for the NN model, and stochastic gradient descent as the optimizer.

3.4 Deep neural networks (DNNs)

DNNs consist of multiple layers of NNs, and work based on a principle similar to that of ANNs. They form a part of the larger family of deep learning architectures, including deep recurrent NNs, deep belief networks, and deep CNNs. DNN architectures for broader applications can include N-different hidden layers, depending on the optimization of the model, and the problem being solved using the DNN. The DNN used in this study comprised six layers, as follows:

Layer 1: An input Layer consisting of 10 neurons representing the 10 input variables;
Layer 2: A hidden layer consisting of 60 neurons;
Layer 3: A hidden layer consisting of 55 neurons;
Layer 4: A hidden layer consisting of 60 neurons;
Layer 5: A hidden layer consisting of 55 neurons; and
Layer 6: An output layer consisting of a single neuron.

This study used a ReLU as the activation function for the neurons with a feed-forward neural architecture, as explained above. The hidden layer neurons were optimized throughout this study for better accuracy and classification results using the trial-and-error method. To reduce the loss function, we used binary cross entropy, and we used stochastic gradient descent as the optimizer for the DNN model.

4 Materials and methods

4.1 Dataset

The data utilized for the research were obtained from the University of California, Irvine machine learning repository, which is one of the leading databases for research datasets in artificial intelligence and machine learning. The dataset contains over 30,000 rows of individual client credit cards, with 23 explanatory features. These 23 explanatory features are provided in the “Appendix 1”. The explanatory features are based on 30,000 client credit card transactions occurring from April to September 2005. The response variable (or dependent variable) is the “default payment next month,” which indicates that the client will fail to pay any amount to the financial institution in the next month, thereby defaulting on the credit card payment. Figure 1 shows the imbalanced distinction between the next-month default and non-default customers. The descriptive statistics for all of the other variables are shown in the “Appendix”.

For training and testing the models, this study used the Python machine learning package Scikit-Learn. A ratio of 80:20 for splitting the entire dataset and sampling techniques were used to randomly split the data into 80% for training and 20% for testing the models. A preliminary analysis of the dataset is explained in the next section.

4.2 Sampling techniques

4.2.1 Oversampling techniques

SMOTE was first proposed by Chawla et al. (2011) in their seminal paper on this technique. Based on Google Scholar’s estimation, over 9000 papers have cited this research, indicating extensive review of this technique over the past two decades. SMOTE is implemented by oversampling the minority class and undersampling the majority class (Chawla et al., 2011). In this study, the minority class would be the segment of data representing credit card clients defaulting in their payment, and the majority class would be the opposite.

SVM-SMOTE is a variant of the SMOTE algorithm that uses an SVM kernel algorithm to detect samples and generate new synthetic samples (Karaa and Krichene, 2012). Based on our literature review, SVM-SMOTE has not been used in the literature for credit risk assessment, as researchers prefer to use SMOTE as a form of oversampling, and then to conduct a further comparison. In this study, by using one more method, a comparison between these two oversampling methods can be established.

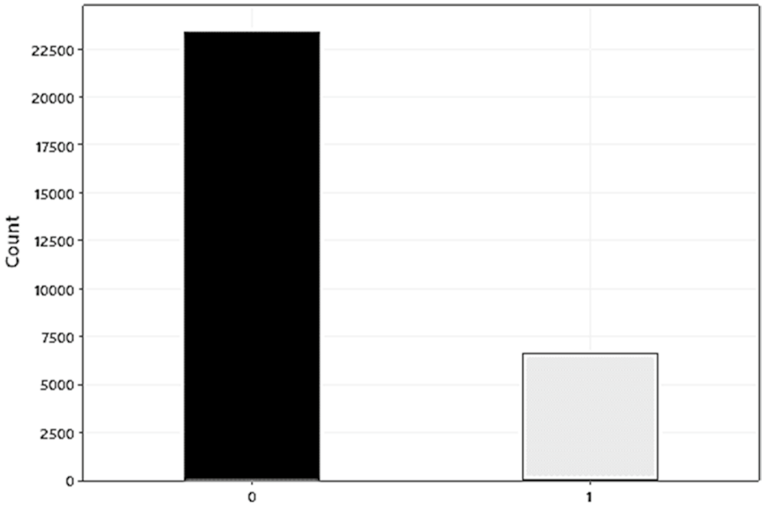

Fig. 1 Imbalanced default payments

4.2.2 Undersampling techniques

RUS has been one of the widely used undersampling techniques in the reviewed literature. This technique undersamples the majority class by randomly selecting samples, with or without replacement.

All-KNN uses a KNN algorithm to conduct undersampling. This technique was developed based on a paper published by Tomek (1976). Based on our literature review, the All-KNN undersampling technique has not been previously employed to study the effect of this technique on the respective models used in this study. Using this technique in this study will allow us to establish a comparison between the RUS and All-KNN techniques for further analysis.

4.3 Performance criteria

The classification models were evaluated based on the accuracy of the models for correctly predicting the target variables. The major performance criteria used in this study were the accuracy, sensitivity, and specificity. The computation of these criteria can be summarized as follows Eqs. (2)—(4):

$$Accuracy = \frac{TP + TN}{TN + TP + FP + FN} \tag{2}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{3}$$

$$Specificity = \frac{TN}{TN + FP} \tag{4}$$

In the above, TP, TN, FP, and FN indicate true positives, true negatives, false positives, and false negatives, respectively. They are defined as follows:

True positive: Model predicts the value as true when the actual value is true;
True negative: Model predicts the value as false when the actual value is true;
False positive: Model predicts the value as true when the actual value is false; and
False negative: Model predicts the value as false when the actual value is false.

These values can be summarized in a confusion matrix providing all of the values for the models, as shown in Table 4.

The accuracy indicates a model's precision in predicting the correct trend, whereas the sensitivity and specificity show the model's precision in predicting bullish and bearish

Table 4 Confusion matrix

		Predicted Y	
		Default payment (Y = 1)	Payment on time (Y = 0)
Actual Y	Default payment (Y = 1)	True positive (TP)	False negative (FN)
	Payment on time (Y = 0)	False positive (FP)	True negative (TN)

trends, respectively. Sensitivity also helps in the model's precision when predicting a positive change, whereas specificity helps when predicting a negative change.

Because the dataset used in this study was imbalanced, in the process of balancing the dataset, sampling techniques were used. Hence, several other performance criteria were measured for the models, so as to make an accurate decision on which model performs well relative to the others. The second set of performance criteria consisted of the balanced accuracy, G-mean, F1 score, and AUC or ROC characteristics for the models. The computation of these criteria can be summarized as follows Eqs. (5)–(7):

$$Balanced\,Accuracy = \frac{Sensitivity + Specificity}{2} \tag{5}$$

$$G{-}Mean = \sqrt{Sensitivity \times Specificity} \tag{6}$$

$$F1{-}Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{7}$$

4.3.1 ROC–AUC

The AUC is the measurement of the ROC of the model, which is calculated based on the prediction scores. Any classifier that follows the 45-degree line is considered as a useless classifier. A perfect classifier classifies a default payment as "default" 100% of the time, whereas a real-life classifier's performance lies somewhere between useless and perfect classifiers.

4.4 Loss functions

4.4.1 Brier score

The Brier score was originally proposed by Brier (1950) in his paper on the verification of weather forecasts outlined with probabilities. It represents the average deviation between predicted probabilities (Brier, 1950); a lower Brier score for a model represents a higher accuracy in the prediction of the outcome.

For a binary classification, the Brier score is given as follows (Martino et al., 2019); Eq. (8):

$$BS = \frac{1}{N}\sum_{i=1}^{N}\left(T\left(y_{i=1}|x_i\right) - P\left(y_{i=1}|x_i\right)\right)^2 \tag{8}$$

In the above, $T\left(y_{i=1}|x_i\right)=1$ if $y_i=1$ and $T\left(y_{i=1}|x_i\right)=0$ otherwise, and $P\left(y_{i=1}|x_i\right)$ is the estimated probability for pattern x_i to belong to Class 1. From the formula, we can see that the Brier score represents the mean squared error, indicating that a lower value of the Brier score indicates better predictions by the models.

4.4.2 Cross entropy loss (log loss)

An alternative measure for the root mean square error in binary classification is known as the cross-entropy loss or log loss. The log loss for a binary classification is determined as follows (Martino et al., 2019); Eq. (9):

$$LL = -\frac{1}{N}\sum_{i=1}^{N}[y_i\log(p_i) + (1 - y_i)\log(1 - p_i)] \tag{9}$$

Here, y_i indicates the true values, and p_i indicates the predicted values. The higher the deviation from the true values, the higher the log loss values, and the lower the log loss values, the better the prediction and accuracy.

5 Framework of the analysis

In this study, we implemented four different models using the two oversampling techniques and two undersampling techniques described in the previous sections. Before applying the models to the dataset, the dataset was preprocessed to perform a preliminary analysis, and the feature selection procedure was conducted. To understand the importance of features and use them for further analysis, we used logistic regression, which is a widely used technique for feature selection in the literature reviewed. Once a set number of features was selected based on the output from the logistic regression, the cleaned dataset was passed through all of the models, along with the sampling techniques. We created 16 models based on combinations of the four models and four different sampling techniques. Therefore, the models were established as follows:

1. Deep NN with SMOTE
2. Deep NN with SVM SMOTE
3. Deep NN with RUS
4. Deep NN with All-KNN
5. Artificial NN with SMOTE
6. Artificial NN with SVM SMOTE
7. Artificial NN with RUS
8. Artificial NN with All-KNN
9. SVM (RBF kernel) with SMOTE
10. SVM (RBF kernel) with SVM SMOTE
11. SVM (RBF kernel) with RUS
12. SVM (RBF kernel) with All-KNN
13. KNN with SMOTE
14. KNN with SVM SMOTE
15. KNN with RUS
16. KNN with All-KNN

The steps involved in the study were as follows.

1. The raw data were downloaded from the repository, and a preliminary analysis was conducted on the features.

2. After the preliminary analysis was completed, the raw data were passed through logistic regression to identify the importance of the features and which features played important roles in detecting default payments.
3. The features were segregated into a separate dataset based on their importance.
4. The segregated dataset was then passed through each of the sampling techniques.
5. Post-sampling, tenfold cross validation was applied to each of the models individually, as outlined above in the combination of the models.
6. Once the segregated dataset was analyzed using the models, the performance criteria were calculated, using the formulas explained in Sect. 4.3.

To compare the performances of the models, we used each model's performance measures without sampling as a benchmark scenario, and then compared them with the performance measures of the models with the sampling techniques. Figure 2 presents a flowchart of the overall framework used in this study.

6 Empirical results and analysis

To better understand the dataset, a preliminary analysis was conducted on the raw dataset, and several descriptive statistics were identified. The descriptive statistics are listed in Table 5, which shows how the dataset is distributed between the defaults and non-default data points. Out of the 30,000 records for clients in the dataset, 6636 indicate that the client defaulted in their payments. The percentage of the default records to total records in the dataset in this study is 22.12%, making the dataset an imbalanced dataset.

Figures 3, 4, 5, and 6 plot the categorical features in the dataset versus the default payments. Middle-aged clients seem to be on the default payment list more than older clients, among which females are on the higher end. Marital status does not have much effect on the default status, as we can see that both married and single clients have the same percentage of default payments.

6.1 Feature selections

To eliminate noise in the dataset and further optimize the importance of the features on the output variable, we implemented logistic regression on the raw dataset, and identified that out of the 23 features in the raw dataset, only 10 features played an important role in the detection of default payments. Out of the 10 variables, six variables were PAY_0 to PAY_6, indicating that the past repayment status plays a major role in identifying whether the client will make any future payments. It can also be stated that these repayment statuses are correlated with the dependent variable.

A logistic regression is applied with the independent variables defined by the characteristics of each client's data as included in this study. These characteristics are outlined in details in the dataset available on "Appendix 1". The choices of independent and dependent variables are made based on these characteristics, and on the definition of default. Based on these definitions, in this study, the dependent variable is the default payment and the independent variables are the remaining features, as outlined in the dataset table in "Appendix 1". Common types of regression analysis use the mean squared error (MSE) as a loss function; it gives a convex shape. Complete optimization can be performed by finding its vertex as a global minimum. However, there is no such option for logistic regression. Because

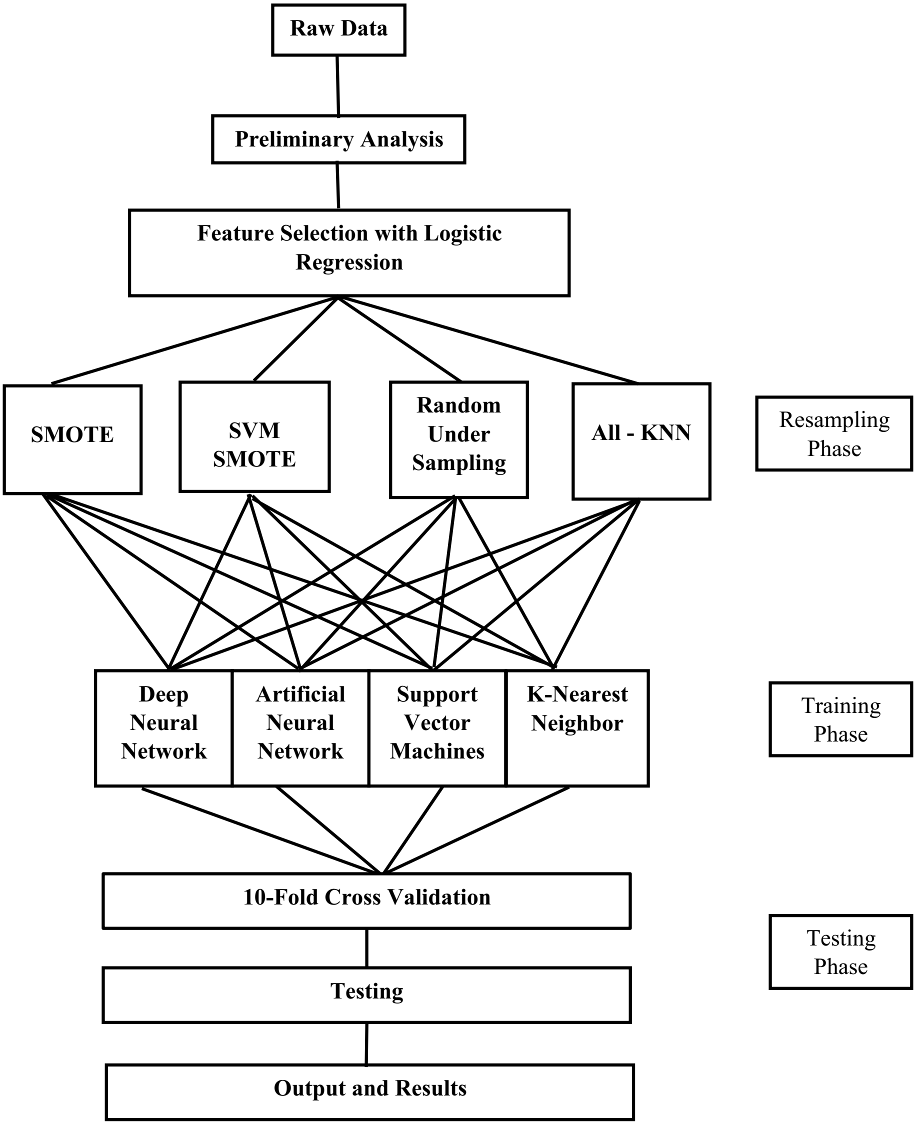

Fig. 2 The architecture of the algorithm used in this study

Table 5 Default payments in the dataset

Total dataset	30,000
Default payments	6636
Percentage of default payments	22.12%

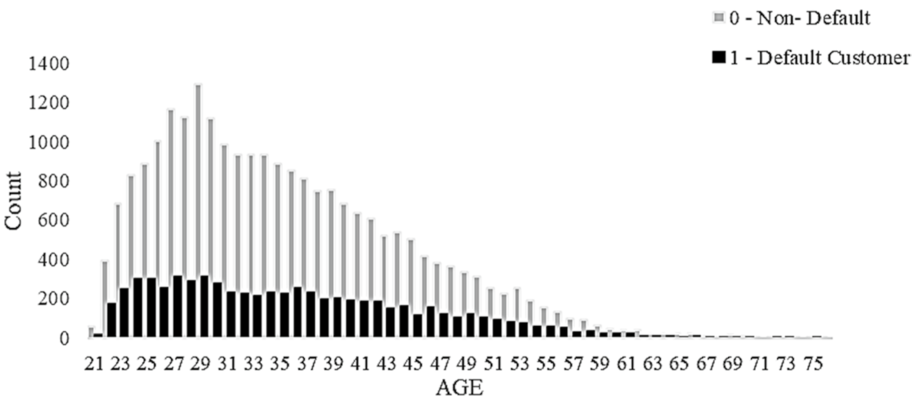

Fig. 3 Plot of age versus default payment

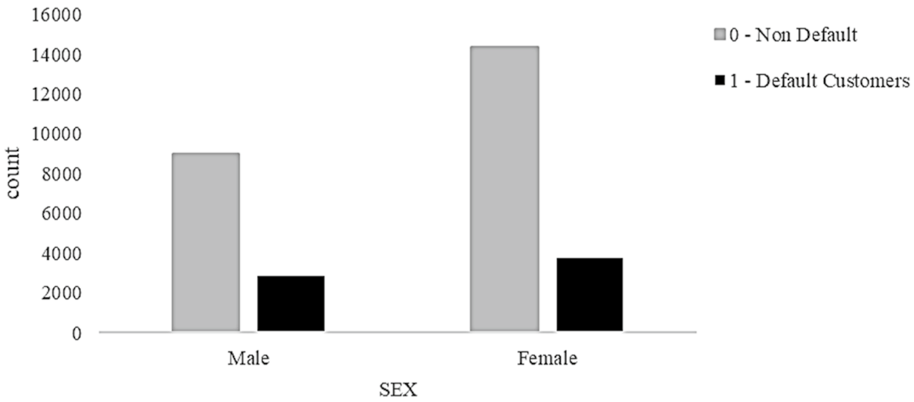

Fig. 4 Plot of sex versus default payment

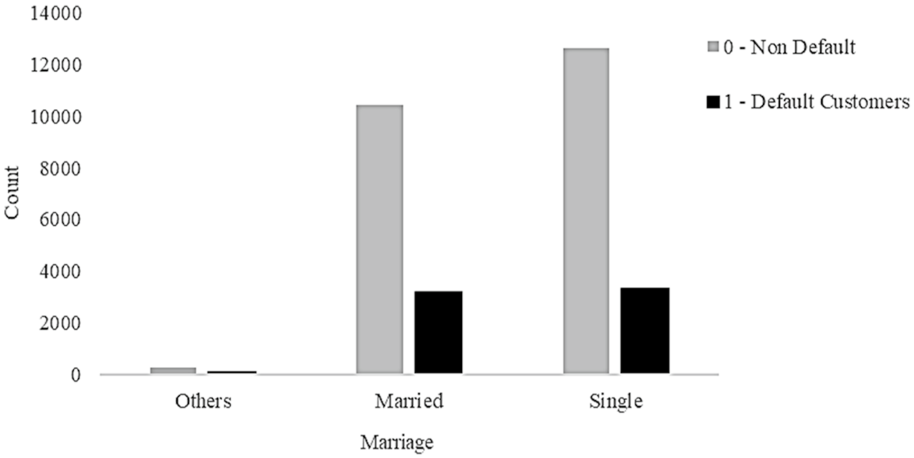

Fig. 5 Plot of marriage versus default payment

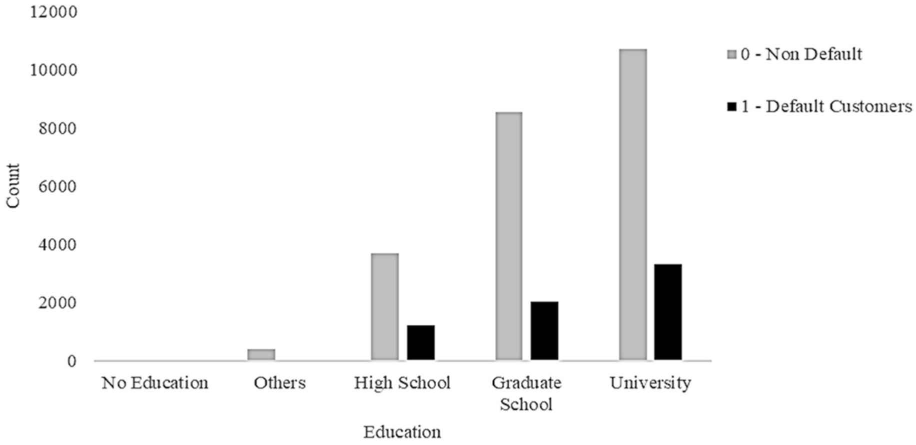

Fig. 6 Plot of education versus default payment

the dependent feature is not continuous, the graph of the MSE will result in a non-convex plot with local minima. The appropriate loss function for logistic regression is known as the cross-entropy loss function for linear classification models, as defined by Murphy (2012). Such a loss function also ensures that as the probability of the correct answer is maximized, the probability of the incorrect answer is minimized; as the two sum to one, any increase in the probability of the correct answer comes at the expense of the incorrect answer (Murphy, 2012). In this study, the optimized cross-entropy loss function is reported by the software packages.

By using the coefficients of the dependent variables obtained from the logistic regression, we can plot a graph of the independent variables against their relative importance. The plot of the relative importance of the features is depicted in Fig. 7, Table 6 displays the logistic regression results as obtained with the variables. The pseudo-R-square value

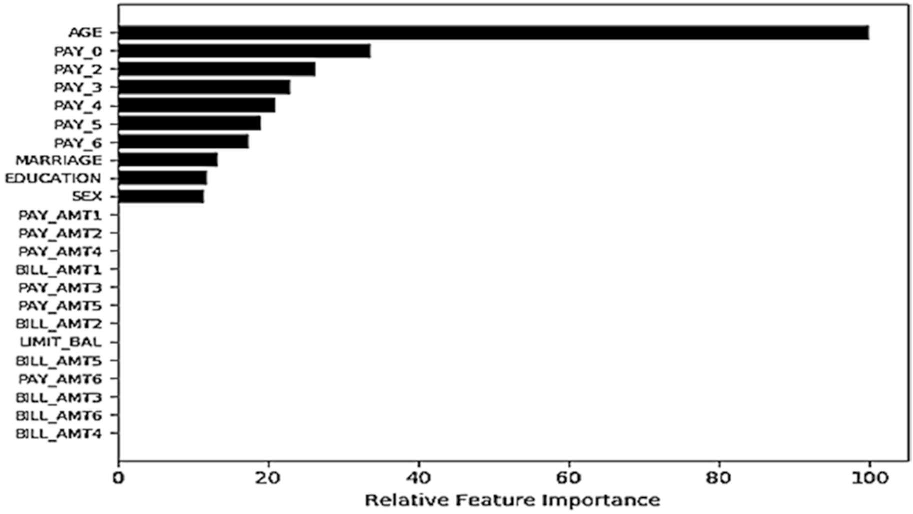

Fig. 7 Plot of features and their relative importance using logistic regression

Table 6 Logistic regression results

Model	Logit
Method	Maximum likelihood estimation (MLE)
Dep. variable	Default payment next month
No. observations	30,000
D.f. residuals	29,976
D.f. model	23
Pseudo R-square	0.1207
Log-likelihood	− 13,939
Log loss (LL)-Null	− 15,853
LLR *p* value	0.00000

of 0.1207 in the table reflects McFadden's R-Square, as per the documentation for the programming used for calculating the values of the logistic regression results. Assuming that L_0 is the value of the likelihood function for a model with no predictors and L_m is the likelihood of the model being estimated (McFadden, 1974), McFadden's R-square is defined as follows Eq. (10):

$$\mathrm{R}^2 = 1 - \left(\frac{\ln\left(L_m\right)}{\ln\left(L_0\right)} \right) \tag{10}$$

According to McFadden (1974), a small ratio for the log-likelihood indicates that the model being estimated is a far better fit than a model with no predictors. Based on the results from this step, the dataset is reduced to only the 10 features playing important roles for further analysis of the models. The *p*-value associated with the log likelihood ratio (LLR) test reported as zero indicating the fitted model is statistically significant.

6.2 Empirical results and advantages of the algorithm

Table 7 presents the complete performance metrics of the models used in this study against each of the sampling techniques, and without sampling (benchmark). The ALL-KNN undersampling technique is the best-performing sampling technique across all of the models and sampling techniques, with an average accuracy of 98% across the four models. The ALL-KNN sampling technique is also able to achieve lower cross-entropy loss or log-loss measures across all models, indicating the efficiency of using this sampling technique in the models. Among the four models under this sampling technique, the SVM outperforms the other models in several performance metrics, and achieves an accuracy of 98.6%, with Brier score loss of 0.006, and log-loss value of 0.028. All of the models used in this study are able to achieve more than 80% specificity, indicating the efficiency of the models in identifying true positives, and the specificity of the models is the highest under the All-KNN sampling technique. The sensitivity of the models is also the highest with ALL-KNN sampling, with a value of 77.4% in the SVM. Among the oversampling techniques of SMOTE and SVMSMOTE, SVMSMOTE outperforms SMOTE in regards

Table 7 Performance metrics for all models using SMOTE, SVM-SMOTE, RUS, and ALL-KNN compared to the benchmark

Accuracy measures	Classifications																			
	SMOTE				SVMSMOTE				RUS				ALL-KNN				NO-SAMPLING-Benchmark			
	KNN	SVM	ANN	DNN	KNN	SVM	ANN	DNN	KNN	SVM	ANN	DNN	KNN	SVM	ANN	DNN	KNN	SVM	ANN	DNN
Accuracy	0.701	0.692	0.691	0.69	0.73	0.721	0.724	0.723	0.669	0.684	0.685	0.685	0.979	**0.986**	0.985	0.983	0.807	0.796	0.802	0.779
Sensitivity	0.569	0.521	0.509	0.51	0.611	0.575	0.563	0.561	0.436	0.518	0.487	0.497	0.640	0.774	0.739	0.719	0.265	0.12	0.891	0.003
Specificity	0.832	0.864	0.873	0.87	0.849	0.867	0.886	0.884	0.903	0.85	0.883	0.874	1.000	0.999	1.000	1.000	0.96	0.987	0.186	1.000
Balanced Accuracy	0.701	0.692	0.691	0.69	0.73	0.721	0.724	0.723	0.669	0.684	0.685	0.685	0.820	0.887	0.869	0.859	0.613	0.554	0.977	0.501
Geometric Mean	0.688	0.671	0.667	0.666	0.72	0.706	0.706	0.705	0.627	0.664	0.656	0.659	0.800	0.880	0.859	0.848	0.504	0.345	0.582	0.054
Precision	0.772	0.793	0.801	0.797	0.802	0.812	0.831	0.829	0.818	0.775	0.806	0.797	0.994	0.977	0.996	0.996	0.655	0.731	0.427	0.655
Recall	0.569	0.521	0.509	0.51	0.611	0.575	0.563	0.561	0.436	0.518	0.487	0.497	0.640	0.774	0.739	0.719	0.265	0.12	0.696	0.003
F1	0.709	0.629	0.622	0.622	0.693	0.674	0.671	0.669	0.569	0.621	0.607	0.613	0.779	0.864	0.848	0.835	0.377	0.207	0.186	0.006
Area Under the ROC Curve	0.701	0.692	0.691	0.69	0.73	0.721	0.724	0.723	0.669	0.684	0.685	0.685	0.820	0.887	0.869	0.859	0.613	0.554	0.294	0.501
Log Loss	0.577	0.602	0.608	0.617	0.55	0.587	0.583	0.591	0.607	0.616	0.613	0.62	0.046	**0.028**	0.065	0.063	0.454	0.474	0.582	0.525
Brier Score Loss	0.196	0.204	0.21	0.214	0.185	0.195	0.198	0.202	0.21	0.212	0.212	0.215	0.015	**0.006**	0.014	0.014	0.143	0.146	0.148	0.171

The best classification model has been chosen based on the three bold values to the performance metrics across the four models. The results from the models without sampling techniques are completely skewed toward non-default payments, as the dataset is unbalanced (with 78% consisting of non-default payments and 22% consisting of default payments). The accuracy of all models is close to 78%, with the DNN at 77.9%, indicating that the model is much more robust than the other models when used with the sampling techniques, and could serve as the benchmark comparison for the models with sampling techniques. The overfitting and underfitting of the models are eliminated using the tenfold cross-validation. This technique is used in this study as the datasets are stationary, and it has approximately 30,000 client credit cards. The precisions of the models across all of the sampling techniques are higher than the accuracies of the models, indicating that the models are able to identify the positive class (default payments) better in the datasets. The models using the All-KNN sampling technique have the highest precision scores, with a precision score of 99.6% under the DNN model used in this study. The Brier score loss is used as a cost function indicator to predict the probability of default. The Brier score loss is consistent across all models, averaging closer to 0.20, with the lowest values in the All-KNN sampling techniques, indicating that the accuracy of the prediction for the positive class is much better across all of the models (especially with the All-KNN sampling technique). To identify the superiority of the models, we use the F1 score, which includes both precision and recall in its calculations. Based on the values presented in Table 7, the SVM with the All-KNN sampling technique could be said to have superior performance relative to the ANN and DNN with the same sampling technique, and is much better than the benchmark measurements calculated without sampling techniques.

7 Implications and conclusions

7.1 Practical insights

The application of machine learning and DNNs can help financial institutions predict counterparty risk failures, as shown in this study. Assuming that the model is applied in a real-life scenario, the loss owing to credit card delinquency can be reduced considerably. According to McKinsey's Global Institute research on credit risk management (Bahillo et al., 2016), the application of machine learning and advanced analytics can help financial institutions in three different ways. First, there is a potential improvement in revenue, owing to the early detection of credit risk or counterparty risk. Second, potential money is saved in regards to cost reductions, owing to the detection of potential fraud customers in the application process for credit instruments, such as credit cards. Third, the money previously employed in risk mitigation strategies surrounding credit risk management is saved. At each of these stages, financial institutions can save up to 10 to 15% of the potential value in revenue, which in combination reduces losses by up to 30–35%, based on application of advanced analytical tools in credit risk management (Bahillo et al., 2016). Further application of advanced analytical models can help banks improve their return on equity by approximately 4% (Härle et al., 2015).

The Canadian Bankers Association reported that over 600,000 credit cardholders were delinquent in 2018 (CBA, 2018), leading to a net loss of approximately CAD 4.38 billion; the net dollar value for credit card transactions alone was at CAD 547.98 billion. This dataset comprised all credit card issuing institutions in Canada. The delinquency rate for 2018 was 0.8% (CBA, 2018), which indicates the total loss value and total delinquent cardholders. By the application of machine learning models, this can be brought down to approximately CAD 2–3 billion if applying the potential reduction percentages as stated by Bahillo et al. (2016). Thus, the understanding and application of DNN-based models can have profound impacts on the bottom lines of major financial institutions.

Considering a loss of CAD (4.38 billion) with 600,000 cardholders, the average loss per cardholder to financial institutions can be approximated as CAD 7300 annually. Assuming the model applied in this study is applied to identify these 600,000 cardholders in an earlier stage, with an average accuracy of 98% accuracy, 540,000 cardholders will be classified as delinquents. The savings would be approximately CAD 3.9 billion to financial institutions if these delinquent cardholders are detected at the earlier stage.

Financial institutions, such as major banks and credit agencies, can combine applications of models and computing powers to develop algorithms for detecting credit card delinquency with better accuracy. For example, at the expense of personal privacy, clients can provide more accurate information to these institutions, so as to accurately choose the required features to detect default payments. The application of DNN models can only provide the required results if provided with the appropriate features for predicting the dependent feature; in this study, the dependent feature was the default payment for the next month. The choice of features has a profound impact on the application of the DNN models, as features with less significant importance can result in noise and increases in error rates, whereas significant features can increase the accuracy rates, as observed in this study.

7.2 Future work

To identify the importance of the features, logistic regression was used in the pre-processing stage, and several of the features were discarded from further analysis. More robust feature selection procedures can be implemented for the selection of features in conjunction with the DNN model proposed in this study. It is understood that not all discarded features may play an important role, but feature selection can play a vital role in the output variable. The dataset used in this study contained information from 30,000 different clients. To understand the complete operation of the DNN model proposed in this study, a larger dataset (on the order of millions of records) will help to further analyze the model. A larger dataset can also help in understanding how fast the proposed model can help in obtaining the output relative to the different models from the literature for credit risk assessment.

To further realize the importance of DNNs, similar studies will be required using different credit instruments such as home mortgages, lines of credit, and vehicle loans. Comparative studies between two different datasets can also help in further analyzing the model. A separate study on the effectiveness of the All-KNN sampling technique across multiple applications could help identify the potential of this undersampling technique.

7.3 Key contributions

Some of the primary contributions of this study are the use of the DNN Model to achieve 98.6% accuracy with a ROC score of 0.859 compared to the existing research (e.g., Bayraci & Susuz, 2019; Sun & Vasarhelyi, 2018). The application of the four different resampling techniques along with four different classification models for the study in credit risk assessment to be used for the first time to the best of our knowledge. This has been identified as another gap on the existing research works (e.g., Bayraci & Susuz, 2019; Sariannidis et al., 2020). This study showed that the SMOTE and RUS, All-KNN, and SVM SMOTE are equally powerful resampling techniques under alternative classification models and scenarios in case of an imbalanced dataset. ALL-KNN resampling technique proves to be the robust technique based on the performance measures reported in this study. The use of cross entropy loss function in binary classification models as a more appropriate criteria in this research can be considered as another contribution. Finally, SVM-SMOTE can also be applied as an alternate oversampling technique in future studies due to the consistent results presented in this paper. This is another significant step in filling the research gap (e.g., Kalid et al., 2020; Kvamme et al., 2018; Rtayli & Enneya, 2020).

8 Conclusion

One of the primary capabilities of a robust risk management system must be detecting the risks earlier, though many of the financial institutions today lack this key capability which leads to further losses (Bahillo et al., 2016). This paper was able to contribute to this gap by proposing a DNN model to be used along with sampling techniques for imbalanced datasets. The proposed model was able to achieve 98.6% accuracy with the use of the ALL-KNN sampling technique and a ROC score of 0.859. As a direct comparison with the models used by Hamori et al. (2018) since they used the same dataset, our models and techniques have much higher accuracy as they were only able to achieve 69.17% average accuracy. Comparing with other models used in the literature, since many of them lacked the use of the variety of the sampling techniques, this study could not place a direct comparison. Being said that at 98.6% accuracy and 0.859 ROC score, the DNN model proposed in this study under the All-KNN sampling technique can be concluded to be used as a real-life classifier in predicting credit risk assessment. Using the methodology and models presented in this paper, credit risk assessment can be analyzed in practical applications where most of the data points are skewed towards non-default payments. The application of sampling techniques enhances the dependency of the models on the data points for training and thereby providing accurate results as compared to the models which train on the datasets directly. Through the accurate implementation of the neural networks and neurons used in the architecture, this paper presents better insights into the functioning of the neural networks when used in conjunction with the sampling techniques.

Appendix 1: Descriptive statistics of the data set

	Mean	SD	Min	Max	Q1	Median	Q3	Range	IQR	Mode	Skew-ness	Kurtosis
LIMIT_BAL	167,484	129,748	10,000	1,000,000	50,000	140,000	240,000	990,000	190,000	50,000	0.99	0.54
SEX	1.6037	0.4891	1	2	1	2	2	1	1	2	− 0.42	− 1.82
EDUCATION	1.8531	0.7903	0	6	1	2	2	6	1	2	0.97	2.08
MARRIAGE	1.5519	0.522	0	3	1	2	2	3	1	2	− 0.02	− 1.36
AGE	35.486	9.218	21	79	28	34	41	58	13	29	0.73	0.04
PAY_0	− 0.0167	1.1238	− 2	8	− 1	0	0	10	1	0	0.73	2.72
PAY_2	− 0.13377	1.19719	− 2	8	− 1	0	0	10	1	0	0.79	1.57
PAY_3	− 0.1662	1.19687	− 2	8	− 1	0	0	10	1	0	0.84	2.08
PAY_4	− 0.22067	1.16914	− 2	8	− 1	0	0	10	1	0	1	3.5
PAY_5	− 0.2662	1.13319	− 2	8	− 1	0	0	10	1	0	1.01	3.99
PAY_6	− 0.2911	1.14999	− 2	8	− 1	0	0	10	1	0	0.95	3.43
BILL_AMT1	51,223	73,636	− 165,580	964,511	3558	22,382	67,093	1,130,091	63,535	0	2.66	9.81
BILL_AMT2	49,179	71,174	− 69,777	983,931	2984	21,200	64,011	1,053,708	61,027	0	2.71	10.3
BILL_AMT3	47,013	69,349	− 157,264	1,664,089	2665	20,089	60,166	1,821,353	57,502	0	3.09	19.78
BILL_AMT4	43,263	64,333	− 170,000	891,586	2326	19,052	54,512	1,061,586	52,186	0	2.82	11.31
BILL_AMT5	40,311	60,797	− 81,334	927,171	1763	18,105	50,202	1,008,505	48,439	0	2.88	12.31
BILL_AMT6	38,872	59,554	− 339,603	961,664	1256	17,071	49,203	1,301,267	47,947	0	2.85	12.27
PAY_AMT1	5664	16,563	0	873,552	1000	2100	5006	873,552	4006	0	14.67	415.25
PAY_AMT2	5921	23,041	0	1,684,259	833	2009	5000	1,684,259	4167	0	30.45	1641.63
PAY_AMT3	5226	17,607	0	896,040	390	1800	4505	896,040	4115	0	17.22	564.31
PAY_AMT4	4826	15,666	0	621,000	296	1500	4014	621,000	3718	0	12.9	277.33
PAY_AMT5	4799	15,278	0	426,529	252	1500	4033	426,529	3781	0	11.13	180.06
PAY_AMT6	5216	17,777	0	528,666	117	1500	4000	528,666	3883	0	10.64	167.16
Default payment next month	0.2212	0.41506	0	1	0	0	0	1	0	0	1.34	− 0.2

References

2019 Global payments trends report—Canada Country Insights. (2019). Retrieved from https://www.jpmorgan.com/merchant-services/insights/reports/Canada

Abdelmoula, A. K. (2015). Bank credit risk analysis with k-nearest neighbor classifier: Case of Tunisian banks. *Accounting and Management Information Systems/Contabilitate Si Informatica de Gestiune, 14*(1), 79–106.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, *46*(3), 175–185.

Bahillo, J. A., Ganguly, S., Kremer, A., & Kristensen, I. (2016). The value in digitally transforming credit risk management. Retrieved from https://www.mckinsey.com/business-functions/risk/our-insights/the-value-in-digitally-transforming-credit-risk-management.

Basel I: International Convergence of Capital Measurement and Capital Standards (1988). Retrieved from https://www.bis.org/publ/bcbs04a.htm

Basel II: International Convergence of Capital Measurement and Capital Standards: A Revised Framework. (2004). Retrieved from https://www.bis.org/publ/bcbs107.htm

Basel III: A global regulatory framework for more resilient banks and banking systems—revised version June 2011. (2011). Retrieved from https://www.bis.org/publ/bcbs189.htm

Bayraci, S., & Susuz, O. (2019). A Deep Neural Network (DNN) based classification model in application to loan default prediction. *Theoretical and Applied Economics, 4*, 75–84.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, *78*(1), 1–3.

Canadian Demands for Speed and Convenience Influencing Payments Innovation. (2018). Retrieved from https://www.payments.ca/industry-info/our-research/canadian-demands-speed-and-convenience-influencing-payments-innovation

CBA—Credit Card Statistics. (2019). Retrieved from https://cba.ca/credit-card-statistics

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 13*(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964

Cao, J., Lu, H., Wang, W., & Wang, J. (2013). A loan default discrimination model using cost-sensitive support vector machine improved by PSO. *Information Technology and Management, 14*(3), 193–204. https://doi.org/10.1007/s10799-013-0161-1

Chen, S., Härdle, W. K., & Moro, R. A. (2011). Modeling default risk with support vector machines. *Quantitative Finance, 11*(1), 135–154. https://doi.org/10.1080/14697680903410015

Cimpoeru, S. S. (2011). Neural networks and their application in credit risk assessment. Evidence from the Romanian Market. *Technological and Economic Development of Economy, 17*(3), 519–534. https://doi.org/10.3846/20294913.2011.606339

Danenas, P., & Garsva, G. (2015). Selection of Support Vector Machines based classifiers for credit risk domain. *Expert Systems with Applications, 42*(6), 3194–3204. https://doi.org/10.1016/j.eswa.2014.12.001

Finlay, S. (2015). Multiple classifier architectures and their application to credit risk assessment. *European Journal of Operational Research, 210*(2), 368–378.

Fix, E., & Hodges, Jr., J. L. (1951). Discriminatory analysis, nonparametric discrimination. Retrieved from https://apps.dtic.mil/dtic/tr/fulltext/u2/a800276.pdf

Gu, Q., & Han, J. (2013 April). Clustered support vector machines. In *Artificial intelligence and statistics* (pp. 307–315). PMLR.

Hamori, S., Kawai, M., Kume, T., Murakami, Y., & Watanabe, C. (2018). Ensemble learning or deep learning? Application to default risk analysis. *Journal of Risk and Financial Management, 11*(1), 12. https://doi.org/10.3390/jrfm11010012

Harris, T. (2015). Credit scoring using the clustered support vector machine. *Expert Systems with Applications, 42*(2), 741–750. https://doi.org/10.1016/j.eswa.2014.08.029.

Härle, P., Havas, A., & Samandari, H. (2015). The future of bank risk management. Retrieved from https://www.mckinsey.com/business-functions/risk/our-insights/the-future-of-bank-risk-management

Haykin, S. S. (1998). *Neural networks:Aa comprehensive foundation*. Prentice-Hall.

Henley, W. E., & Hand, D. J. (1996). A k-nearest-neighbour classifier for assessing consumer credit risk. *Journal of the Royal Statistical Society, Series D, 45*(1), 77. https://doi.org/10.2307/2348414

Kalid, S. N., Ng, K., Tong, G., & Khor, K. (2020). A Multiple classifiers system for anomaly detection in credit card data with unbalanced and overlapped classes. *IEEE Access, 8*, 28210–28221. https://doi.org/10.1109/ACCESS.2020.2972009

Karaa, A., & Krichene, A. (2012). Credit-risk assessment using support vectors machine and multilayer neural network models: A comparative study case of a tunisian bank. *Accounting and Management Information Systems/Contabilitate Si Informatica De Gestiune, 11*(4), 587–620.

Khashman, A. (2010). Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes. *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2010.02.101

Khemakhem, S., & Boujelbènea, Y. (2015). Credit risk prediction: A comparative study between discriminant analysis and the neural network approach. *Accounting and Management Information Systems/ Contabilitate Si Informatica De Gestiune, 14*(1), 60–78.

Kvamme, H., Sellereite, N., Aas, K., & Sjursen, S. (2018). Predicting mortgage default using convolutional neural networks. *Expert Systems with Applications, 102*, 207–217. https://doi.org/10.1016/j.eswa.2018.02.029

Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research, 247*(1), 124–136.

Marinakis, Y., Marinaki, M., Doumpos, M., Matsatsinis, N., & Zopounidis, C. (2008). Optimization of nearest neighbor classifiers via metaheuristic algorithms for credit risk assessment. *Journal of Global Optimization, 42*(2), 279–293.

Martino, A., Rizzi, A., & Frattale Mascioli, F. M. (2019). Efficient approaches for solving the largescale k-medoids problem: Towards structured data. In C. Sabourin, J. J. Merelo, K. Madani, & K. Warwick (Eds.), *Computational Intelligence: 9th International Joint Conference, IJCCI 2017 FunchalMadeira, Portugal, November 1–3, 2017 Revised Selected Papers* (pp. 199–219). Cham: Springer International Publishing.

Massaron, L., & Boschetti, A. (2016). *Regression analysis with Python*. Packt Publishing.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115–133.

McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. In P. Zarembka (Ed.), *Frontiers in econometrics* (pp. 104–142). Academic Press.

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.

Nadkarni, P., & Nadkarni, P. (2016). Core technologies: Data mining and "Big Data". *Clinical Research Computing*, *9*, 187–204.

Oreski, S., Oreski, D., & Oreski, G. (2012). Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. *Expert Systems with Applications, 39*(16), 12605–12617. https://doi.org/10.1016/j.eswa.2012.05.023

Rao, C., Liu, M., Goh, M., & Wen, J. (2020). 2-stage modified random forest model for credit risk assessment of P2P network lending to "Three Rurals" borrowers. *Applied Soft Computing, 95*, 106570.

Rtayli, N., & Enneya, N. (2020). Selection features and support vector machine for credit card risk identification. *Procedia Manufacturing, 46*, 941–948. https://doi.org/10.1016/j.promfg.2020.05.012

Sariannidis, N., Papadakis, S., Garefalakis, A., Lemonakis, C., & Kyriaki-Argyro, T. (2020). Default avoidance on credit card portfolios using accounting, demographical and exploratory factors: Decision making based on machine learning (ML) techniques. *Annals of Operations Research, 294*(1), 715–739.

Sun, T., & Vasarhelyi, M. A. (2018). Predicting credit card delinquencies: An application of deep neural networks. *Intelligent Systems in Accounting, Finance and Management, 25*(4), 174–189. https://doi.org/10.1002/isaf.1437

Tomek, I. (2007). An experiment with the edited nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics SMC-6, 6*, 448–452. https://doi.org/10.1109/TSMC.1976.4309523

Trustorff, J. H., Konrad, P. M., & Leker, J. (2011). Credit risk prediction using support vector machines. *Review of Quantitative Finance and Accounting*, *36*(4), 565–581.

Vapnik, V. N. (2000). *The nature of statistical learning theory* (2nd ed.). Springer.

Wang, J., Hedar, A. R., Wang, S., & Ma, J. (2012). Rough set and scatter search metaheuristic based feature selection for credit scoring. *Expert Systems with Applications*, *39*(6), 6123–6128.

Yeh, I. C., & Lien, C. H. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications, 36*(2), 2473–2480.

Yu, L., Yue, W., Wang, S., & Lai, K. K. (2010). Support vector machine based multiagent ensemble learning for credit risk evaluation. *Expert Systems with Applications*, *37*(2), 1351–1360.

Zhu, B., Yang, W., Wang, H., & Yuan, Y. (2018). A hybrid deep learning model for consumer credit scoring. In *2018 international conference on artificial intelligence and big data (ICAIBD)* (pp. 205–208). https://doi.org/10.1109/ICAIBD.2018.8396195

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.