

Bagging Supervised Autoencoder Classifier for Credit Scoring

Mahsan Abdoli ^{*1}, Mohammad Akbari ^{†2}, and Jamal Shahrabi ^{‡1}

¹Department of Industrial Engineering, Amirkabir University of Technology (Tehran Polytechnic)

²Department of Mathematics and Computer Science, Amirkabir University of Technology (Tehran Polytechnic)

Abstract

Credit scoring models, which are among the most potent risk management tools that banks and financial institutes rely on, have been a popular subject for research in the past few decades. Accordingly, many approaches have been developed to address the challenges in classifying loan applicants and improve and facilitate decision-making. The imbalanced nature of credit scoring datasets, as well as the heterogeneous nature of features in credit scoring datasets pose difficulties in developing and implementing effective credit scoring models, targeting the generalization power of classification models on unseen data. In this paper, we propose the Bagging Supervised Autoencoder Classifier (BSAC) that mainly leverages the superior performance of the Supervised Autoencoder, which learns low-dimensional embeddings of the input data exclusively with regards to the ultimate classification task of credit scoring, based on the principles of multi-task learning. BSAC also addresses the data imbalance problem by employing a variant of the Bagging process based on the undersampling of the majority class. The obtained results from our experiments on the benchmark and real-life credit scoring datasets illustrate the robustness and effectiveness of the Bagging Supervised Autoencoder Classifier in the classification of loan applicants that can be regarded as a positive development in credit scoring models.

Keywords: Autoencoder, Ensemble Learning, Multi-task Learning, Imbalanced Dataset, Credit Scoring

1 Introduction

Credit scoring is one of the core elements of the financial industry that attempts to assess and measure the creditworthiness of loan applicants and classify them

^{*}m.abdoli@aut.ac.ir (Mahsan Abdoli)

[†]Corresponding author: akbari.ma@aut.ac.ir (Mohammad Akbari)

[‡]jamalshahrabi@aut.ac.ir (Jamal Shahrabi)

into reliable and unreliable ones, aiming to facilitate the decision-making process of banks and financial institutes on granting loans to customers. In general, robust and trustworthy credit scoring models lead to reduction of administrative costs, minimization of human error or bias in accepting or refusing loan applications, and decreasing the probability of rejecting creditworthy applicants or accepting high-risk ones, which in turn, will result in lower financial risks and higher profits. As a result, in credit scoring literature, it is well-established that a minor improvement in the performance of credit scoring models can result in significant savings (Baesens et al., 2003).

Over the years, several events have stressed the benefits and the need to develop more effective credit scoring models. For instance, one of the factors that resulted in the financial crisis of 2008 was the massive default of sub-prime borrowers in mortgage markets, which ultimately triggered the collapse of many financial institutes. To respond to this crisis, financial institutes have put a concerted effort in improving their measures in managing their financial risks by developing more effective credit scoring models (García et al., 2019). In addition, with the emergence of digital technology, the financial sector has dramatically changed. Recently, Peer-to-Peer lending online platforms have emerged as an alternative to traditional lending markets that allow individuals to directly borrow money from lenders without the financial institutes' intermediations. Unlike traditional lending, the offered loans in peer-to-peer platforms are not secured and cannot be backed up by collateral. Hence, it is crucial to determine the default risk associated with granting a loan to each borrower (Emekter et al., 2015; Bastani et al., 2019). Moreover, due to digitalization and new alternative lending markets, such as peer-to-peer lending, loan portfolios have expanded. Therefore, increasing customer data, such as information on their credit history, demographic, and financial data, is being accumulated. This data can be used to propose and implement credit scoring models based on machine learning techniques to reduce financial risk resulting from loan default. Accordingly, numerous credit scoring models based on various binary classification algorithms have been implemented to leverage the massive amounts of customers data throughout the years. For instance, Linear Discriminant Analysis (LDA) (Reichert et al., 1983), Logistic Regression (Serrano-Cinca et al., 2015), Support Vector Machines (SVM) (Yu et al., 2018), and Random Forests (Malekipirbazari & Aksakalli, 2015) are among the utilized classification techniques.

Retrospective studies in credit scoring have proposed several promising approaches to improve the classification performance in classifying loan applicants to help the decision-making process of banks and financial institutes. For example, ensemble methods that combine the decisions of several classification models have resulted in superior classification performance compared to individual models and therefore are widely used in credit scoring (Galar et al., 2011; Lessmann et al., 2015; Dastile et al., 2020). Profit-based credit scoring approaches that incorporate the profitability of loans in the classification process to maximize the obtained profits from the accepted and rejected loans have also been proposed (Bastani et al., 2019; Kozodoi et al., 2019). However, several challenges

remain to be addressed in improving loan default classification performance. First, credit scoring datasets usually suffer from the class imbalance problem since the majority of instances in the dataset are customers who have been able to pay off their loans (negative class) (Brown & Mues, 2012; J. Xiao et al., 2012). The class imbalance problem may adversely affect classification models’ generalization power since most of these models have been formulated to minimize an accuracy-based objective function in the training phase (Haixiang et al., 2017). To alleviate this issue, several approaches have been developed that can be categorized as: Data level, algorithmic level, and cost-sensitive approaches, which are described in detail in Section 2.1. Second, in general, credit scoring data are typically comprised of structured, semi-structured, and non-structured data that illustrate each loan applicant’s demographic, financial, and behavioral information. In order to develop robust credit scoring classification techniques that perform well in practice, it is crucial to embed the underlying information from each aspect of the credit scoring dataset in a lower-dimensional representation of the input data (Lei et al., 2019; Maldonado et al., 2017). Representation learning is an approach that derives lower-dimensional embeddings of initial input features to obtain a more expressive data representation that increases the generalization power (Najafabadi et al., 2015). Among representation learning approaches, autoencoders have recently attracted much attention in extracting the input data’s underlying patterns before classification (Bengio et al., 2012), which are used in this study. Third, despite the superior performance of neural networks and deep learning algorithms in many research fields, these learning models have not yet been extensively applied in credit scoring. Thus, further exploration of neural network-based classifiers in credit scoring is required (Dastile et al., 2020; Bhatore et al., 2020; Shen et al., 2020).

Although several studies in credit scoring literature have focused on alleviating the negative impacts of class imbalance on credit scoring models, and some studies have utilized representation learning as a means to improve the classification performance, to the best of our knowledge, only one paper ((Wong et al., 2020)) has proposed a model that benefits from representation learning and addresses the adverse effect of imbalanced data simultaneously by developing a cost-sensitive classification model based on stacked autoencoders for business problems. Since a further investigation is needed to determine the power of representation learning-based classifiers in imbalanced environments, we propose the Bagging Supervised Autoencoder Classifier in this paper.

In this paper, we propose the Bagging Supervised Autoencoder Classifier (BSAC) to tackle the challenges of class imbalance and improve classification generalization by representation learning. BSAC benefits from the representation power of the Supervised Autoencoders, which enhances the power of autoencoders in learning new representations by applying the principles of multi-task learning in order to extract better representations that are guided to learn new embeddings exclusively with regards to the final classification task. In addition, BSAC employs under-sampling and ensemble learning, which are subcategories of data level and algorithmic level approaches in handling imbalanced datasets, respectively, to further improve the classification performance in the imbalanced

setting of credit scoring. The main contributions of this paper are:

- To find the optimal representation of credit scoring input data, we use the supervised autoencoder, which is based on the principles of multi-task learning, in learning a lower-dimensional embedding guided by the final classification task (Section 4.1).
- To decrease the negative effects of class imbalance in credit scoring, we integrate data level and algorithmic level approaches in dealing with imbalanced data, namely a variant of the bagging process and under-sampling of the majority data, respectively (Section 4.3) and evaluate the effectiveness and superiority of obtained results from BSAC against previous papers..

2 Related Works

2.1 Data Imbalance

In many real-life classification problems, the number of observations belonging to the positive class is significantly smaller than the number of observed negative instances. This class imbalance problem may adversely affect the machine learning classification algorithms' performance, as most of them minimize an accuracy-based objective function in the training phase. Therefore, the classification results on an imbalanced dataset may be biased towards the majority class (Haixiang et al., 2017).

So far, much research have been focused on minimizing the effect of class imbalance on the final predictions of learning algorithms in credit scoring (Brown & Mues, 2012; Yu et al., 2018; He et al., 2018). Most of these efforts can be classified into three categories: Data level, Algorithm level, and Cost-sensitive algorithms (Galar et al., 2011; Haixiang et al., 2017). It is worth mentioning that the following categories overlap in most works and cannot be classified as belonging to only one of these categories.

2.1.1 Data level

The imbalance ratio of a dataset is defined as the number of samples with negative class (majority class) divided by the number of samples with positive class (minority class).

Data level approaches in imbalanced learning intend to balance out the class distribution by resampling methods such that the imbalance ratio of the dataset is approximately close to 1. Random Undersampling (RUS), random Oversampling (ROS), Synthetic Minority Oversampling Technique (SMOTE), and its variants are among these methods.

Random undersampling, randomly selects a number of observations in the majority sample, and discards them until the number of majority and minority class samples are roughly the same. One of the major drawbacks of RUS is that valuable information may be lost during the process. Random oversampling,

randomly selects and replicates a sample from the minority class to modify the class distribution in the data set to obtain an approximate imbalance ratio of one.

Since directly oversampling the minority class instances may lead to overfitting and lack of generalization, Chawla et al. (2002) proposed synthetic minority oversampling technique to overcome this problem. In their approach, instead of oversampling minority samples, synthetic samples are generated by computing the distance between a minority class sample and its nearest neighbor, and multiplying this distance by a random number between 0 and 1 and add it to the minority sample’s feature vector. This will generate a synthetic sample that belongs to the minority class, which leads to a more general decision region of the minority class.

Resampling methods have been implemented in several studies in credit scoring literature. For instance, Brown & Mues (2012) conducted a study to examine the robustness of several classification methods in classifying imbalanced credit scoring datasets. They increased the class imbalance severity of their data by undersampling the positive class to observe the changes in classification performance in environments with higher imbalance ratio. He et al. (2018) extend the Balance Cascade technique, which is a supervised undersampling method to create data subsets as inputs to an ensemble learning model. They evaluated their ensemble model by applying it to several credit scoring datasets. Papouskova & Hajek (2019) developed a two-stage ensemble learning model using EUSBoost, which is an evolutionary undersampling method to overcome the problems of class imbalance in credit scoring.

Regardless of implemented classification models and techniques, several studies (Duan, 2019; Shen et al., 2020; Chen et al., 2019) employed SMOTE or a variation of it as a data pre-processing step to enhance the robustness of classification algorithms in classifying highly imbalanced data sets.

2.1.2 Cost-sensitive learning

In practical applications of machine learning, the misclassification costs of different classes are not equal. For instance, it is stated that misclassifying a defaulted loan as a non-default application is more costly for financial institutions (García et al., 2019). Many classification algorithms assume an equal misclassification cost for minority and majority classes in imbalanced data sets. Cost-sensitive approaches try to incorporate misclassification costs into the classifiers’ training phase to improve the classification performance of minority samples. The misclassification cost can be represented as c_{ij} , which is the cost of misclassifying a sample as class i instead of class j .

In credit scoring literature, many researchers have focused on developing models that incorporate misclassification costs into the original cost function of classifiers. J. Xiao et al. (2012) proposed a cost-sensitive dynamic ensemble method that dynamically selects classifiers for each test sample classification. Feng et al. (2018) also developed a dynamic ensemble classification model for credit scoring based on the relative costs of Type I and Type II error which

selects a subset of classifiers for classifying test samples. Xia et al. (2017) propose a cost-sensitive boosted tree loan evaluation model to enhance the classification performance in identifying the minority class instances from majority class ones. In their proposed cost-sensitive model, Wong et al. (2020) assigned a higher misclassification cost to the minority class instances in an ensemble of stacked autoencoders to increase the classification capability. Several other research papers such as (Bahnsen et al., 2015; J. Xiao et al., 2020) have also proposed different classification models based on cost-sensitive learning in classification of defaulted and non-default loans.

2.1.3 Algorithmic level

Algorithmic level approaches attempt to modify algorithms to improve their performance in an imbalanced setting. For instance, modified Support Vector Machine (SVM) (Batuwita & Palade, 2010; Chen et al., 2019), logistic regression (Maalouf & Trafalis, 2011), and several other studies have been proposed in this category. For a detailed review of other modified algorithms, we refer to Haixiang et al. (2017) review of imbalanced learning.

The second approach of the algorithmic level is the ensemble learning method. Ensemble learning models, also known as multiple classifier systems, aggregate the decisions of different accurate and diverse individual (base) classifiers. Ensemble learning methods were developed under the motivation that different, diverse classifiers often make different mistakes because of their innate distinctions in classifying the same data. Therefore, if these classifiers’ decisions are aggregated together, they can compensate for each others’ mistakes when making the final prediction (Kittler et al., 1998).

Empirical research has shown that ensemble models have superior performance in credit scoring (Lessmann et al., 2015; Wang et al., 2011). In addition, out of 527 papers reviewed by Haixiang et al. (2017), 218 papers proposed novel ensemble models in various imbalanced classification tasks. Specifically, in credit scoring, the majority of existing literature revolves around developing ensemble classification models since Lessmann et al. (2015) and Wang et al. (2011) have demonstrated that compared to other classification algorithms in credit scoring, ensemble classifiers show an outstanding classification performance.

As a result of ensemble models’ superior performance in credit scoring, Yu et al. (2018) used support vector machines as the base classifier of their ensemble model. After applying a re-sampling method to generate balanced training subsets, SVM classifiers are trained. To reach the final prediction of their proposed model, they use a deep belief network to fuse the predictions of base classifiers. Xia et al. (2018) also uses a trainable classifier, namely extreme gradient boosting algorithm, to fuse the prediction of base classifiers. Their proposed heterogeneous ensemble model (i.e., an ensemble model with different base classifiers) integrates the bagging algorithm with stacking. Sun et al. (2018) proposed an ensemble of decision tree classifiers to classify the imbalanced data of enterprise credit evaluation. They integrated bagging, SMOTE and differentiated sampling rates to create diverse subsets of data as input

to decision tree classifiers. Since different subsets of data were created, the final trained decision trees will be diverse classifiers that can perform better in an imbalanced environment. So far, much research has been conducted that integrate the data level, cost-sensitive learning, and algorithmic level to explore the improvements that can be made in credit scoring classification (He et al., 2018; García et al., 2019; H. Xiao et al., 2016).

As a result of Ensemble learning superior performance in imbalanced data sets, in this study, we employ ensemble learning as well as re-sampling methods to classify a highly imbalanced real-life credit scoring dataset.

2.2 Neural Networks

An Artificial Neural Network (ANN) is a mathematical model inspired by the human brain’s biological neural structure. ANNs attempt to imitate the way the biological neural structures process information by a network of interconnected neurons. Artificial Neural Networks usually consist of three layers: an input layer, a hidden layer, and an output layer. Each layer in ANN contains several neurons that are connected to the neurons of the consequent layer by weighted connections. During the training phase of a classification problem, the neural network’s weights are adjusted by the backpropagation process, such that the error (usually Mean Squared Error (MSE)) between the ANN’s output and the actual label of the training samples is minimized. A more comprehensive introduction to Artificial Neural Networks is presented in Section 3.1.

Variants of Artificial Neural Networks have been used in the credit scoring literature. Bastani et al. (2019) use the wide and deep model proposed by Google to develop a two stage credit scoring model that identifies most profitable loans for investment. In their proposed model, the first stage is responsible to identify non-default loans. In the second stage, the profitability of the identified non-default loans is predicted. In their work, Shen et al. (2020) improved the synthetic minority oversampling technique (SMOTE) along with a deep ensemble learning model to increase the classification performance of credit risk evaluation. Neagoe et al. (2018) compared the performance of deep convolutional neural networks with multi-layer perceptron in the credit scoring classification problem. The result of their study indicates that the deep convolutional neural networks improve the overall accuracy of classification significantly.

Although, studies have been conducted by implementing Artificial Neural Networks and deep learning approaches in the credit scoring literature (Yu et al., 2018; Duan, 2019), they are still among the least frequently used classifiers in this area of research and further exploration of these models in credit scoring is required (Dastile et al., 2020; Shen et al., 2020; Bhatore et al., 2020).

2.2.1 Autoencoders

An Autoencoder is a three-layer neural network with an input, hidden, and an output layer that attempts to reconstruct the input data as its output. In the training phase, the input variables are the same as the target variables

and the parameters of the Autoencoder, specifically its weights, are adjusted to minimize the error between the actual input data and the reconstruction of the training set. The final goal of training Autoencoders is to learn the useful representation of the input data in the hidden layer, which enables the Autoencoder to reconstruct the input data as the output (Goodfellow et al., 2016). To increase the generalization power of other machine learning tasks, such as classification, the hidden representations of Autoencoders can be used as the input of other machine learning models. a comprehensive introduction to Autoencoders is presented in Section 3.2.

In the credit scoring literature, Tran et al. (2016) use genetic programming and stacked Autoencoders to develop a two-step hybrid credit scoring model. In the training phase, if-then classification rules are extracted from the training data by genetic programming and the stacked Autoencoder is trained. In the testing phase, first the test data is classified by the extracted rules. The stacked Autoencoder will classify the remainder of data that could not be classified by the extracted rules. Wong et al. (2020) also used stacked Autoencoders to classify six datasets in the business domain. They proposed the Cost-Sensitive Deep Neural Network (CSDNN) and the Cost-Sensitive Deep Neural Network Ensemble (CSDE) to address the common class imbalance problem in the business domain datasets such as direct marketing, credit scoring, and fraud detection. They modified the Stacked Denoising Autoencoders (SDA) cost function by incorporating the majority and minority class’s misclassification cost vector into the original cost function.

Fan & Yang (2018) used Denoising Autoencoders to reduce noise in the data, to improve the classification accuracy of the Lending Club dataset. A multi-stage ensemble learning model was proposed by Chen et al. (2019) for credit risk prediction. Gradient boosting decision trees and Autoencoders were used to generate new features and dimensionality reduction, respectively. Furthermore, logistic regression was applied to the extracted features from the previous stage to classify borrowers’ credit risk. Mancisidor et al. (2018) create customer clusters to identify segments that have different propensities of default on their loans. First, they transform the input data by the Weight of Evidence transformation, based on the exceeded days past due date to encapsulate the probability of default. Then, they used the Variational Autoencoder latent space, where the data’s complex and non-linear relationships exist for clustering customers in segments with different propensities of default to finalize the classification.

Based on the reviewed credit scoring approaches, numerous studies in this field focus on the class imbalance problem, and few address the use of neural networks and representation learning in improving the classification performance. To the best of our knowledge, only one paper has utilized neural networks, representation learning and approaches in mitigating the effects of class imbalance problem in credit scoring. Therefore, to address this shortcoming in credit scoring literature, in this study, an Autoencoder-based classification system based on multi-task learning is proposed as a credit scoring model. The Bagging Supervised Autoencoder classifier leverages the potentials of ensemble learning, multi-task learning, representation learning to handle imbalanced credit scoring

datasets. The BSAC is described in detail in Section 4.

3 Preliminaries

In this Section, we briefly introduce the related algorithms to this study such as, artificial neural networks, and Autoencoders which are adopted in our study.

3.1 Artificial Neural Networks

An artificial neural network is a biology-inspired mathematical model that tries to emulate how a biological neural network processes information. Artificial neural networks typically consist of three layers: an input, hidden, and an output layer. Each layer is constructed by several units called neurons which are connected to the neurons of the subsequent layer with an associated weight. The output of each neuron is the sum product of its inputs with the corresponding associated weight. Then a non-linear activation function is passed to the output of each neuron to generate an activation output which is then passed to the neurons of the subsequent layer as an input. The activation output layer l is denoted by a^l which is defined in equation 1 and 2:

$$a^l = \alpha(z^l) \quad (1)$$

$$z^l = W^l a^{l-1} + b^l \quad (2)$$

Where α is the activation function, z^l is the linear output of layer l , W^l is the weight matrix associated with the connection of neurons in layer $l-1$ and l , and b^l is the bias vector of neurons in layer l . During each iteration of the training phase, the weight matrix W and the bias vector b are modified and updated to minimize the cost function J based on the comparison of the predicted label (\hat{y}_i) and the ground truth (y_i) by *backpropagation* process. The Mean Squared Error (MSE, equation 3) or in case of a binary classification problem binary cross entropy (equation 4) cost functions are usually used in the training process which for one training sample x_i is calculated as follows.

$$J_{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (3)$$

$$J_{CrossEntropy} = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \quad (4)$$

3.2 Autoencoders

An Autoencoder is a three-layered artificial neural network structure that tries to reconstruct the input data in the output layer (Goodfellow et al., 2016). The Autoencoder has a hidden layer that provides the representation or code that is utilized to reconstruct the input data in the output layer. The Autoencoder’s structure can be described in two steps. First, an encoder function $h = f(x)$, encodes the input data into the hidden layer representation, then the decoder function $r = g(x)$ attempts to reconstruct the input data from the Autoencoder’s hidden layer representation.

$$h(x) = \alpha(Wx + b) \quad (5)$$

$$r(h(x)) = \alpha(W'h(x) + b') \quad (6)$$

In equation 5 and 6 the W and W' are the encoder and decoder weight matrix, respectively. Also b and b' are the bias vectors of encoder and decoder layers of the Autoencoder. Similar to the simple artificial neural networks, the weight matrix and the bias vector are optimized based on a cost function (equation 7) using the backpropagation process. In the Autoencoders cost function, the reconstruction error, which is the difference between the input data (x_i) and its reconstruction (\hat{x}_i) by the Autoencoder, is minimized as follows:

$$L(x, \hat{x}) = \frac{1}{2} \sum_{i=1} N(\hat{x}_i - x_i)^2 \quad (7)$$

The general structure of Autoencoders is depicted in figure 1.

3.3 Multi-task Learning

Multi-task learning is a promising approach in improving the generalization of machine learning models by leveraging the available information that can be shared between other different, related learning tasks so that what is learned for one task can help improve the performance of other learning problems (Caruana, 1997). Multi-task learning can be observed in human behavior when learning different activities related to each other by sharing gained knowledge from one activity to learn another. Therefore, similar to human behavior, learning multiple learning tasks simultaneously in the learning process can be beneficial (Zhang & Yang, 2018).

In general, the typical approach in machine learning problems is to first break up a large problem into different, related small sub-problems and learn them separately. However, it is stated that this approach would result in abandoning and neglecting the valuable information contained in other tasks in the learning process of a specific task. Therefore, it is possible to construct models based on

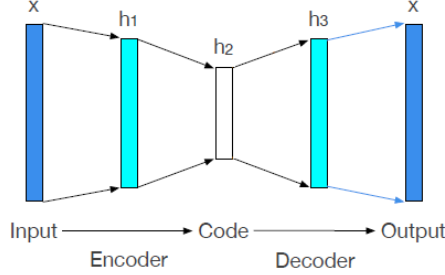


Figure 1: The general structure of an Autoencoder adopted from Le et al. (2018). In the Autoencoder, the input data are first encoded to a new representation. Then, in the decoding phase, the network attempts to reconstruct the input data in the output layer using the new representation.

the available information of other models in parallel (Waibel et al., 1989). As a result, multi-task learning can be defined as:

Given T learning tasks $T = \{t_1, t_2, \dots, t_T\}$, where all or a subset of t_i tasks are related to each other and are not identical, multi-task learning attempts to improve the learning performance of a model for task t_i by using the knowledge learned and contained in all or a subset of $T - 1$ other learning tasks simultaneously.

Based on the principles of multi-task learning, and the importance of an expressive data representation in improving the machine learning tasks, The BSAC is developed for the credit scoring problem at hand, which will be described in Section 4.2.

4 Methodology

4.1 Supervised Autoencoder

In a supervised learning problem, the ultimate goal is to train a model on the training data $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ to learn a mapping function between the input data x_i and its target y_i that performs well in classifying unseen data. Increasing the generalization power of machine learning models and, in turn, increasing their performance in the classification task is one of the essential objectives in this setting. As mentioned in Section 1, constructing a better and more expressive representation of the input data before performing classification is among the approaches that are implemented to increase the classification performance (Najafabadi et al., 2015). Among the various models that have been proposed to improve the representation of the input data in machine learning, much attention has been paid to Autoencoders recently (Bengio et al., 2012).

The Supervised Autoencoder (SA) has been introduced with the idea of utilizing the representation power of Autoencoders in a supervised setting, with regards to the principles of multi-task learning (Le et al., 2018). Essentially, regarding the principles of multi-task learning, the Supervised Autoencoder can be defined as below:

Given two learning tasks of representation learning and classification ($T = \{t_1, t_2\}$) that are related to each other, a Supervised Autoencoder attempts to both improve the generalization power of classification, and enhance the expressiveness of the learned representation by the Autoencoder, using the information available in learning both tasks simultaneously. This goal is achieved by the addition of a supervised loss to the representation layer of the Autoencoder.

Adding a supervised loss to the representation layer of an Autoencoder directs the learning towards representations that are more effective in the classification task. In contrast, if the training process in representation learning takes place solely based on the input data, the Autoencoder may be able to extract underlying information well-fitted to the input data, but the found solutions may not generalize well in the subsequent classification task. In other words, the Supervised Autoencoder combines the reconstruction loss of the Autoencoder with the prediction loss of the classification task to improve the target prediction performance by using representations specific to classification (Le et al., 2018).

In Supervised Autoencoder, the main objective is to create a network that learns a mapping function between the input data $D = \{(x_1, y_1), \dots, (x_m, y_m)\}$ to the output labels y using the new representation that is simultaneously learned by the network in the learning process. In other words, the network consists of two main parts that jointly learn the new input data representation, and the final labels as shown in figure 2. Therefore, the Supervised Autoencoder’s final cost function is defined as shown in equation 8.

$$\frac{1}{m} \sum_{i=1}^m [(\gamma)L_r(\hat{x}_i, x_i) + (1 - \gamma)L_p(\hat{y}_i, y_i)] \quad (8)$$

Here, as shown in equation 9, L_r is the reconstruction loss of the network, which is defined as a Mean Squared Error loss between the input data x and the reconstructed values in the output layer of the Autoencoder \hat{x} .

$$L_r = \frac{1}{M} \sum_{i=1}^N (x - \hat{x}) \quad (9)$$

The L_p (equation 10) is the prediction loss that uses the learned representations to predict the labels of input data, which is defined as a binary cross entropy loss between the true labels y_i of each data sample x_i and the predicted labels \hat{y}_i .

$$L_p = -\frac{1}{M} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \quad (10)$$

In equation 8, the values of γ and $1 - \gamma$ are the weighting parameters of L_r and L_p , respectively, which are incorporated in the cost function in order to assess and balance the contributions each term, namely the reconstruction loss and the prediction loss, in the final structure. The value of γ is optimized by the grid search process using the validation set. It is worth to emphasize that the optimal weights of the connections between neurons of each layer is learned simultaneously by the reconstruction loss and prediction loss.

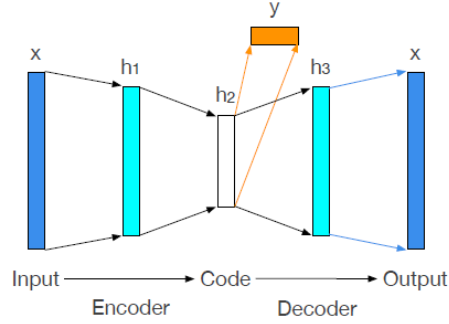


Figure 2: The general structure of a Supervised Autoencoder from Le et al. (2018). The Supervised Autoencoder consists of adding a classification loss to the hidden (representation) layer of an Autoencoder to increase the classification performance by directing the new representations to fit better to the classification task.

It has been shown, both empirically and theoretically, that Supervised Autoencoders can improve the generalization power of neural networks with the same structure (minus the Autoencoder) (Le et al., 2018). As a result, In this paper a Bagging Supervised Autoencoder Classifier is proposed in loan data classification

4.2 Bagging Supervised Autoencoder Classifier

This Section describes the general structure of the proposed Bagging Supervised Autoencoder Classifier. The BSAC consists of three main steps described in the subsequent Sections:

4.3 Step 1: Creating the subsets of data

The first step in the BSAC is to create subsets of data based on the Bagging procedure. The bagging algorithm is one of the most effective algorithms in

ensemble learning due to its power in decreasing the variance and as a result, preventing overfitting (Breiman, 1996). The BSAC utilizes the principles of the Bagging process to decrease overfitting and increase the model’s robustness in handling credit scoring’s imbalanced data.

In order to handle the class imbalance problem in credit scoring more effectively, here, we use a variant of the Bagging procedure by incorporating subsampling into Bagging to train base classifiers (Supervised Autoencoders) on.

Given a dataset with m samples such that $m = p + n$, and p and n are the number of minority and majority samples respectively, balanced subsets of data are created. Specifically, first, the minority samples are set aside. Then, random subsampling is applied on the majority samples to select approximately p negative samples from the customer data that have successfully paid off their loans. Then, the randomly selected majority samples are combined with the p minority samples to create a balanced subset that consists of $2p$ samples. The number of created subsamples are determined by the rounding down imbalance ratio of the data set. Finally, all subsets are used as training data for Supervised Autoencoders, which are considered our base classifiers, here.

It is worth mentioning that in the proposed model, valuable information will not be lost in the sampling process, since all samples in the majority class will be used to train the base classifiers. In addition, the sampling method can prevent overfitting by training the base classifiers on balanced subsets of data.

4.3.1 Step 2: Training base classifiers

After creating random subsets of the data, each subset is utilized to train a Supervised Autoencoder to both reconstruct data representations, and classify each samples. In the training process, the optimal values of γ and $1 - \gamma$ are determined by using the validation set based on the F1 measure since both recall and precision are an integral part in an effective credit scoring system in distinguishing the differences between minority and majority samples, especially in an imbalanced setting.

4.3.2 Step 3: Base Classifiers Aggregation

After training each Supervised Autoencoder as base classifiers and determining the optimal value for γ , the majority voting process is utilized to aggregate the results of individual classifier to reach to the final prediction.

As a result, the BSAC combines ensemble learning, representation learning, subsampling, and multi-task learning to effectively handle the imbalance data of credit scoring and improve the classification performance. Algorithm 1 illustrates the classification process of the BSAC. In addition, figures 3 and 4 depict the general training and testing process of BSAC, respectively.

Algorithm 1: The BSAC Algorithm

Input: Data set with m samples, $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$
 p : The number of positive samples in D
 n : The number of negative samples in D
 $IR = \frac{n}{p}$: The imbalance ratio
 SAE : The supervised Autoencoder classifier
 $C = \emptyset$: The pool of trained classifiers

- 1 **for** $i \in IR$ **do**
- 2 p_i : Choose p samples randomly without replacement from the positive samples in the data set.
- 3 $D_i = p_i \cup n$ The balanced data set used for training each SAE .
- 4 $c_i = SAE(D_i)$: The trained supervised Autoencoder on the balanced data set.
- 5 $C \cup c_i$: Update the pool of trained classifiers
- 6 **end**
- 7 **return** The pool of classifiers $C = \{c_1, c_2, \dots, c_{IR}\}$

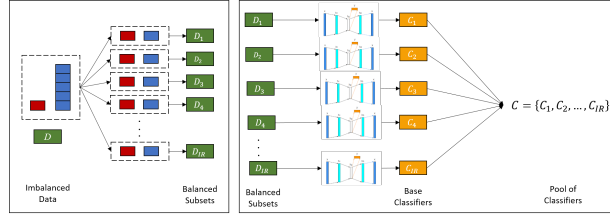


Figure 3: The training phase of the BSAC. In this model, first, balanced subsets of data are randomly created from the imbalanced training data. Then, each subset is used as the training set of the Supervised Autoencoder as the base classifier. Finally, the predictions of each base classifiers are aggregated by the majority voting process.

5 Experimental Setting

5.1 Datasets

5.1.1 Taiwan Credit Data

The Taiwan credit data ¹ is used in this study as a benchmarking dataset. The Taiwan credit dataset contains 30000 data samples with 24 variables to be used in a binary classification task to predict customers' loan default. After using dummy variables for categorical features the final dataset has 33 variables including the target value. The imbalance ratio of the Taiwan dataset is 3.52.

¹retrieved from <https://archive.ics.uci.edu/ml/datasets/Taiwan>

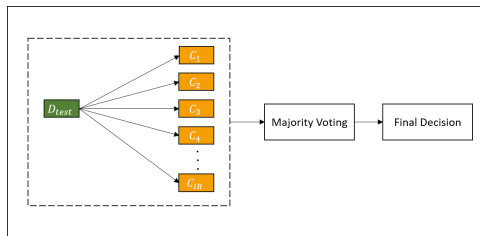


Figure 4: The testing phase of the BSAC. After training the Supervised Autoencoders as base classifiers using the balanced subsets, each test sample is labeled by the base classifiers, and the final output is determined by the majority voting process.

5.1.2 Lending Club Dataset

In this study, we also use the Lending Club dataset², which is one of the most popular online lending platforms in the financial industry. The loan data of the Lending Club dataset consists of loans with the payback duration of either 36 or 60 months. Since the latest payment made by a customer in our dataset was from March 2019, we selected 36 month loans issued before March 2016 to ensure that the loan has had enough time to reach maturity. Out of the 151 variables available in our dataset, we removed variables that included more than half the size of the dataset missing values. The remaining variables were organized such that variables without useful information, such as customers' *id* or *URL*, were dropped.

The *average_fico* was engineered from the average values of original variables *fico_range_high*, and *fico_range_low* for each data sample. We also created the *credit_history* variable by calculating the credit history of each customer from variables *earliest_cr_line* and *issue_d*. Finally, continuous variables were normalized, and as the last step in data preparation, we used dummy variables to represent categorical features of our dataset. The final variables and their descriptions are included in Table 4 in appendix. The final dataset contains 206732 samples with 82 variables, including the target value of loan status. The imbalance ratio of our dataset was calculated as 5.98.

5.2 Evaluation Metrics

To evaluate the performance of the techniques studied in this paper, we consider Accuracy, Precision and Recall, F-measure, and G-mean as the evaluation metrics. The selected evaluation metrics have been chosen as they can cover different aspects regarding the performance of a classification model.

²retrieved from <https://www.kaggle.com/wordsforthewise/lending-club>

5.3 Experiments

To evaluate the performance of our proposed method, BSAC, we use k-fold cross validation as follows. The value of k was determined as 5. Therefore, we split our dataset into 5 equal folds and in each iteration, we hold out one fold as the test set, one fold as the validation set to optimize the hyper-parameter γ , and use the remaining 5-1-1 folds as training data. It is worth mentioning that each fold contains the same ratio of negative and positive samples to increase the training process robustness against the class imbalance in our data set.

First, we evaluate the performance of our model using the Taiwan Credit data, which is used as the benchmark to compare our results with previous studies in this field. The structure of our Bagging Supervised Autoencoder was defined as {32, 16, 8, 5, 8, 16, 32} so that each layer contains approximately half the number of neurons from the previous layer. Our experiments were run for 200 epochs.

Second, after verifying the performance of our model on the benchmark data, we apply the BSAC on the Lending Club dataset. Similar to the Taiwan experiments, we use 5-fold cross validation by splitting the data into 5 equal folds and using one fold as the test set, and one fold from the training data as the validation set to optimize hyper-parameter γ . The structure of the BSAC is defined as {81, 60, 30, 15, 30, 60, 81}, which is run for 200 epochs on our dataset.

During the validation process, the hyper-parameter γ is optimized using the validation fold, by searching for the best value of γ in the search space of $\gamma \in \{0.9, 0.8, \dots, 0.1\}$. The experiments were implemented with python, keras and tensorflow 2.0.

5.3.1 Hyper-parameter Selection and Classifier Diversity

As mentioned in Section 2.1.3, ensemble classification models benefit from both the accuracy and the diversity of base classifiers in the pool of classifiers to increase the classification performance. Several approaches can be implemented to increase the diversity of classifiers used in an ensemble learning model (Duin, 2002). The following approaches are ordered based on the level of diversity from lower to higher diversities:

1. **Different initialization:** Such as changing the initial configuration of a neural networks.
2. **Different parameters:** Such as changing the kernels of a Support Vector Machine.
3. **Different architecture:** Such as changing the structure and the number of layers in a neural network.
4. **Different classifiers:** Such as utilizing a heterogeneous pool of classifiers.
5. **Different training sets:** Such as using different training data for each base classifier of the pool.

6. **Different feature sets:** Such as using different features for each base classifier of the pool.

In the Bagging Supervised Autoencoder Classifier proposed in this paper, two approaches have been used in order to increase the diversity of the base classifiers. First, we use different subsets of balanced data, $D_i = p_i \cup n$, which is comprised of the same number of positive and negative samples to train the base classifiers. In addition to using different training sets for each base classifiers, we also tune the parameters of each base classifier separately with regards to its performance on the validation data. Therefore, each base classifier in the final ensemble pool of classifiers uses different hyper-parameters, which guarantees diversity to some extent.

6 Results and Discussion

6.1 BSAC performance

The obtained results from the experiments for the Taiwan credit data is illustrated in Table 1. The mean and standard deviation of each evaluation metric is reported in the last two rows of Table 1 in order to depict the stability of the Bagging Supervised Autoencoder.

Table 1: Obtained results from the BSAC model on Taiwan credit dataset after optimizing hyper-parameter γ

Fold	Recall (TPR)	F1 score	G-mean	Specificity (TNR)
0	0.5783	0.5426	0.6981	0.8427
1	0.4861	0.5200	0.6581	0.8911
2	0.5463	0.5286	0.6823	0.8521
3	0.5735	0.5371	0.6942	0.8404
4	0.5109	0.5297	0.671	0.8812
mean	0.5390	0.5316	0.6807	0.8615
std	0.0357	0.0077	0.0147	0.0207

The Taiwan credit data has been used in this study to verify the performance of our proposed model against other models in the literature. We have selected three papers that address the imbalanced nature of the credit data using either ensemble learning, sampling, or by implementing an Autoencoder-based model. The comparison between the results of our model and the selected papers are illustrated in Table 2. The comparison of the results of BSAC across different evaluation metrics to other studies that have been conducted using the Taiwan dataset, indicates that BSAC is capable of classifying credit scoring data effectively. It is worth mentioning that the obtained competitive results are only based on optimizing hyper-parameter γ .

Table 3 shows the obtained results from the experiments on the Lending Club dataset, which was used as a real-life data set to illustrate the performance of

Table 2: Comparison of BSAC results with similar papers in the literature.

Classification Model	Evaluation Metric			
	Recall	Specificity	F1 score	G-mean
García et al. (2019)	0.396	0.961	not reported	not reported
He et al. (2018)	not reported	not reported	0.50415	0.62006
Wong et al. (2020)	0.62	0.78	not reported	0.7
BSAC (proposed model)	0.5390	0.8615	0.5316	0.6807

our model.

Table 3: Obtained results from the BSAC model on Lending Club dataset after optimizing hyper-parameter γ

Fold	Recall (TPR)	F1 score	G-mean	Specificity (TNR)
0	0.5572	0.3400	0.6292	0.7104
1	0.5425	0.3356	0.6231	0.7156
2	0.5560	0.3369	0.6267	0.7065
3	0.5317	0.3346	0.6200	0.7230
4	0.5465	0.3438	0.6296	0.7252
mean	0.5468	0.3382	0.6257	0.7161
std	0.0094	0.0033	0.0037	0.0071

6.2 Sensitivity Analysis

As mentioned in Section 4, the loss function of BSAC is comprised of two main terms, namely the reconstruction loss L_r and the prediction loss L_p . The hyper-parameter γ controls the contribution of L_r and L_p in the final classification model, which is selected by hyper-parameter optimization during the training phase.

Therefore, after the validation and hyper-parameter selection process, a high value of γ indicates that the final outcome of the classification task, is highly dependent on the new representation of the input data by reconstructing data samples. Similarly, a low γ value decreases the dependency of BSAC to the reconstruction loss and a neural network that has the same structure as the encoder part of the Autoencoder would have been sufficient for obtaining a high classification performance.

Figure 5 illustrates the performance of each base classifier (trained on balanced subsets of data D_i) on the validation set in each fold based on the F1 measure, which was used in the hyper-parameter selection. The decline of the performance on the validation set, as γ is decreasing indicates that the Autoencoder’s representations play a substantial role in the classification performance.

Similarly, Figure 6 illustrates the performance of BSAC with respect to the F1 measure on validation data on the Taiwan dataset. In contrast to the Lending Club dataset, the increasing slope of the plots indicates that the classification

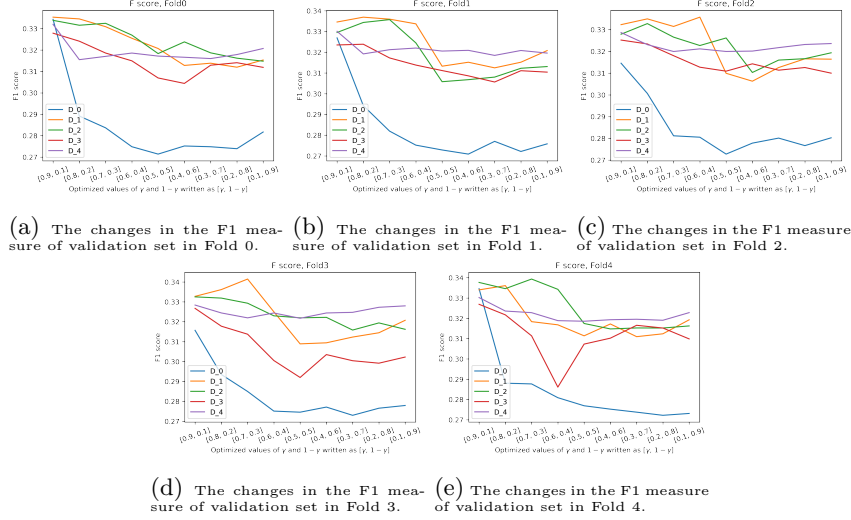


Figure 5: The performance of BSAC in the hyper-parameter selection phase of each Fold on the Lending Club data as the parameter γ decreases. The decline of classification performance based on F1 measure indicates that the extra layer of Supervised Autoencoder plays a substantial role in obtaining a higher classification performance.

performance is not dependent on the high values of γ . This difference can be justified by different complexities of the Taiwan and Lending Club datasets. As mentioned before, the Lending Club dataset suffers from a high class imbalance, which can negatively affect the classification task. Consequently, a higher γ value was selected in the validation process to extract useful patterns for classifying the unknown data by assigning a higher importance to the reconstruction loss L_r .

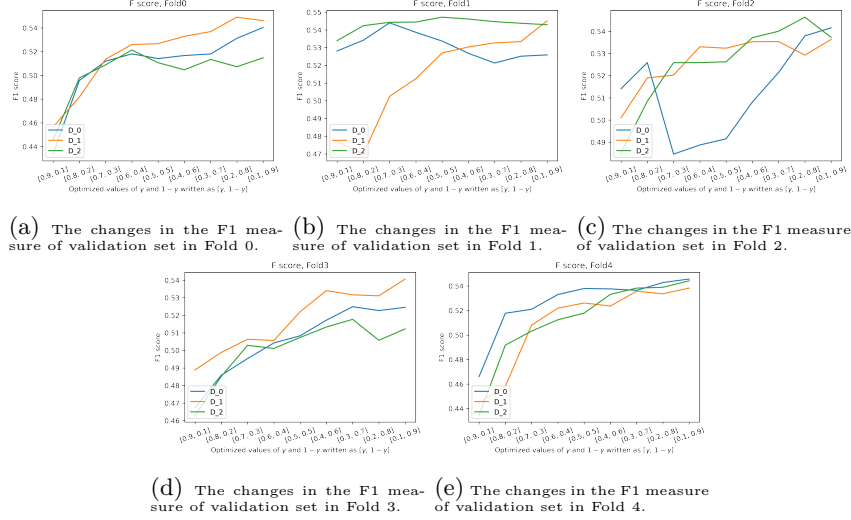


Figure 6: The performance of BSAC in the hyper-parameter selection phase of each Fold in cross validation on the benchmark data as the parameter γ decreases. The general decrease of classification performance based on F1 evaluation metric indicates that the input data reconstruction does not affect the classification performance substantially due to the lower complexity of the Taiwan dataset.

7 Conclusion

Credit scoring can be considered as one of the most important measures that banks and financial institutes take to evaluate and decrease financial risk. In the credit scoring literature, it is well established that a minor improvement in the credit scoring models' performance can lead to significant savings in the financial sector. The class imbalance problem in the credit scoring datasets, as well as the heterogeneous nature of credit scoring data are among the challenges that limit the generalization power of credit scoring models to achieve higher performance. Therefore, in this study, we set out to develop a novel method in credit scoring classification by proposing a supervised Autoencoder network to embed the underlying information of credit scoring heterogeneous data into a lower-dimensional space and decrease the negative effects of class imbalance. Our proposed method, Bagging Supervised Autoencoder Classifier, combines the advantages of representation learning, ensemble learning, multi-task learning and under-sampling of negative samples, in order to develop a robust classification model for credit scoring, which inherently suffers from the class imbalance problem. The results of our experiments on benchmark data and a real-life credit scoring dataset proves the promising classification performance of BSAC.

Due to the our computational limitations in running the experiments, we

have only focused on optimizing the hyper-parameter γ in our model. It is clear that a more detailed hyper-parameter optimization would further improve the classification performance.

Since the decision-making process of accepting or rejecting loan applications is highly prone to error and bias, in the future we intend to investigate the explainability and fairness of our proposed model in credit scoring to decrease bias and disparity towards customers, and increase transparency of the decision-making process of financial institutions.

A Lending Club Dataset variables

Table 4: Variables used in this study.

Variables	Description
loan_amnt	Amount of the loan applied for by the borrower.
acc_now_delinq	The number of accounts on which the borrower is now delinquent.
int_rate	Interest Rate on the loan
installment	The monthly payment owed by the borrower if the loan originates.
annual_inc	The self-reported annual income provided by the borrower during registration.
emp_length	employment length in years.
verification_status	Indicates if income was verified by LC ³ , not verified, or if the income source was verified
dti	The borrower’s total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower’s self-reported monthly income.
delinq_2yrs	The number of 30+ days past-due incidences of delinquency in the borrower’s credit file for the past 2 years
average_fico	The average value of the upper and lower boundary range of the borrower’s FICO at loan origination.
inq_last_6mths	The number of inquiries in past 6 months (excluding auto and mortgage inquiries).
open_acc	The number of open credit lines in the borrower’s credit file.
pub_rec	Number of derogatory public records.
revol_util	Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
total_acc	The total number of credit lines currently in the borrower’s credit file.
chargeoff_within_12_mths	Number of charge-offs within 12 months.
delinq_amnt	The past-due amount owed for the accounts on which the borrower is now delinquent
mort_acc	Number of mortgage accounts
pub_rec_bankruptcies	Number of public record bankruptcies
tax_liens	Number of tax liens
grade	LC assigned loan grade. The values are: [A,B,C,D,E,F,G]

³Lending Club

Table 4 continued from previous page

Variables	Description
subgrade	LC assigned loan subgrade. The values are: [A1,...,A5,...,G1,...,G5]
home_ownership	The home ownership status provided by the borrower during registration or obtained from the credit report. The values are: RENT, OWN, MORTGAGE
purpose	A category provided by the borrower for the loan request such as: Debt consolidation, small business, vacation, etc.
initial_list_status	The initial listing status of the loan. The values are: Whole, Fractional
loan_status	Current status of the loan (Target variable)

References

- Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the operational research society*, 54(6), 627–635.
- Bahnsen, A. C., Aouada, D., & Ottersten, B. (2015). Example-dependent cost-sensitive decision trees. *Expert Systems with Applications*, 42(19), 6609–6619.
- Bastani, K., Asgari, E., & Namavari, H. (2019). Wide and deep learning for peer-to-peer lending. *Expert Systems with Applications*, 134, 209–224.
- Batuwita, R., & Palade, V. (2010). Fsvm-cil: fuzzy support vector machines for class imbalance learning. *IEEE Transactions on Fuzzy Systems*, 18(3), 558–571.
- Bengio, Y., Courville, A. C., & Vincent, P. (2012). Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR*, abs/1206.5538, 1, 2012.
- Bhatore, S., Mohan, L., & Reddy, Y. R. (2020). Machine learning techniques for credit risk evaluation: a systematic literature review. *Journal of Banking and Financial Technology*, 1–28.
- Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2), 123–140.
- Brown, I., & Mues, C. (2012). An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications*, 39(3), 3446–3453.

- Caruana, R. (1997). Multitask learning. *Machine learning*, 28(1), 41–75.
- 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.
- Chen, S., Wang, Q., & Liu, S. (2019). Credit risk prediction in peer-to-peer lending with ensemble learning framework. In *2019 chinese control and decision conference (ccdc)* (pp. 4373–4377).
- Dastile, X., Celik, T., & Potsane, M. (2020). Statistical and machine learning models in credit scoring: A systematic literature survey. *Applied Soft Computing*, 106263.
- Duan, J. (2019). Financial system modeling using deep neural networks (dnns) for effective risk assessment and prediction. *Journal of the Franklin Institute*, 356(8), 4716–4731.
- Duin, R. P. (2002). The combining classifier: to train or not to train? In *Object recognition supported by user interaction for service robots* (Vol. 2, pp. 765–770).
- Emekter, R., Tu, Y., Jirasakuldech, B., & Lu, M. (2015). Evaluating credit risk and loan performance in online peer-to-peer (p2p) lending. *Applied Economics*, 47(1), 54–70.
- Fan, Q., & Yang, J. (2018). A denoising autoencoder approach for credit risk analysis. In *Proceedings of the 2018 international conference on computing and artificial intelligence* (pp. 62–65).
- Feng, X., Xiao, Z., Zhong, B., Qiu, J., & Dong, Y. (2018). Dynamic ensemble classification for credit scoring using soft probability. *Applied Soft Computing*, 65, 139–151.
- Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2011). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4), 463–484.
- García, V., Marqués, A. I., & Sánchez, J. S. (2019). Exploring the synergetic effects of sample types on the performance of ensembles for credit risk and corporate bankruptcy prediction. *Information Fusion*, 47, 88–101.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. (<http://www.deeplearningbook.org>)
- Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, 73, 220–239.

- He, H., Zhang, W., & Zhang, S. (2018). A novel ensemble method for credit scoring: Adaption of different imbalance ratios. *Expert Systems with Applications*, 98, 105–117.
- Kittler, J., Hatef, M., Duin, R. P., & Matas, J. (1998). On combining classifiers. *IEEE transactions on pattern analysis and machine intelligence*, 20(3), 226–239.
- Kozodoi, N., Lessmann, S., Papakonstantinou, K., Gatsoulis, Y., & Baesens, B. (2019). A multi-objective approach for profit-driven feature selection in credit scoring. *Decision support systems*, 120, 106–117.
- Le, L., Patterson, A., & White, M. (2018). Supervised autoencoders: Improving generalization performance with unsupervised regularizers. *Advances in neural information processing systems*, 31, 107–117.
- Lei, K., Xie, Y., Zhong, S., Dai, J., Yang, M., & Shen, Y. (2019). Generative adversarial fusion network for class imbalance credit scoring. *Neural Computing and Applications*, 1–12.
- 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.
- Maalouf, M., & Trafalis, T. B. (2011). Robust weighted kernel logistic regression in imbalanced and rare events data. *Computational Statistics & Data Analysis*, 55(1), 168–183.
- Maldonado, S., Pérez, J., & Bravo, C. (2017). Cost-based feature selection for support vector machines: An application in credit scoring. *European Journal of Operational Research*, 261(2), 656–665. Retrieved from <https://www.sciencedirect.com/science/article/pii/S0377221717301595> doi: <https://doi.org/10.1016/j.ejor.2017.02.037>
- Malekipirbazari, M., & Aksakalli, V. (2015). Risk assessment in social lending via random forests. *Expert Systems with Applications*, 42(10), 4621–4631.
- Mancisidor, R. A., Kampffmeyer, M., Aas, K., & Jenssen, R. (2018). Segment-based credit scoring using latent clusters in the variational autoencoder. *arXiv preprint arXiv:1806.02538*.
- Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R., & Muharemagic, E. (2015). Deep learning applications and challenges in big data analytics. *Journal of Big Data*, 2(1), 1.
- Neagoe, V.-E., Ciotec, A.-D., & Cucu, G.-S. (2018). Deep convolutional neural networks versus multilayer perceptron for financial prediction. In *2018 international conference on communications (comm)* (pp. 201–206).

- Papouskova, M., & Hajek, P. (2019). Two-stage consumer credit risk modelling using heterogeneous ensemble learning. *Decision Support Systems*, 118, 33–45.
- Reichert, A. K., Cho, C.-C., & Wagner, G. M. (1983). An examination of the conceptual issues involved in developing credit-scoring models. *Journal of Business & Economic Statistics*, 1(2), 101–114.
- Serrano-Cinca, C., Gutiérrez-Nieto, B., & López-Palacios, L. (2015). Determinants of default in p2p lending. *PloS one*, 10(10), e0139427.
- Shen, F., Zhao, X., Kou, G., & Alsaadi, F. E. (2020). A new deep learning ensemble credit risk evaluation model with an improved synthetic minority oversampling technique. *Applied Soft Computing*, 106852.
- Sun, J., Lang, J., Fujita, H., & Li, H. (2018). Imbalanced enterprise credit evaluation with dte-sbd: Decision tree ensemble based on smote and bagging with differentiated sampling rates. *Information Sciences*, 425, 76–91.
- Tran, K., Duong, T., & Ho, Q. (2016). Credit scoring model: A combination of genetic programming and deep learning. In *2016 future technologies conference (ftc)* (pp. 145–149).
- Waibel, A., Sawai, H., & Shikano, K. (1989). Modularity and scaling in large phonemic neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(12), 1888–1898.
- Wang, G., Hao, J., Ma, J., & Jiang, H. (2011). A comparative assessment of ensemble learning for credit scoring. *Expert systems with applications*, 38(1), 223–230.
- Wong, M. L., Seng, K., & Wong, P. K. (2020). Cost-sensitive ensemble of stacked denoising autoencoders for class imbalance problems in business domain. *Expert Systems with Applications*, 141, 112918.
- Xia, Y., Liu, C., Da, B., & Xie, F. (2018). A novel heterogeneous ensemble credit scoring model based on bstacking approach. *Expert Systems with Applications*, 93, 182–199.
- Xia, Y., Liu, C., & Liu, N. (2017). Cost-sensitive boosted tree for loan evaluation in peer-to-peer lending. *Electronic Commerce Research and Applications*, 24, 30–49.
- Xiao, H., Xiao, Z., & Wang, Y. (2016). Ensemble classification based on supervised clustering for credit scoring. *Applied Soft Computing*, 43, 73–86.
- Xiao, J., Xie, L., He, C., & Jiang, X. (2012). Dynamic classifier ensemble model for customer classification with imbalanced class distribution. *Expert Systems with Applications*, 39(3), 3668–3675.

- Xiao, J., Zhou, X., Zhong, Y., Xie, L., Gu, X., & Liu, D. (2020). Cost-sensitive semi-supervised selective ensemble model for customer credit scoring. *Knowledge-Based Systems*, 189, 105118.
- Yu, L., Zhou, R., Tang, L., & Chen, R. (2018). A dbn-based resampling svm ensemble learning paradigm for credit classification with imbalanced data. *Applied Soft Computing*, 69, 192–202.
- Zhang, Y., & Yang, Q. (2018). An overview of multi-task learning. *National Science Review*, 5(1), 30–43.