Expert Systems With Applications 176 (2021) 114840

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Big data analytics for default prediction using graph theory

Mustafa Yıldırım [a], Feyza Yıldırım Okay [a,*], Suat Özdemir [b]

[a] Department of Computer Engineering, Gazi University, Maltepe, Ankara, Turkey
[b] Department of Computer Engineering, Hacettepe University, Beytepe, Ankara, Turkey

ARTICLE INFO

Keywords:
Big data analytics
Graph theory
Machine learning
Default prediction
SHAP value

ABSTRACT

With the unprecedented increase in data all over the world, financial sector such as companies and industries try to remain competitive by transforming themselves into data-driven organizations. By analyzing a huge amount of financial data, companies are able to obtain valuable information to determine their strategic plans such as risk control, crisis management, or growth management. However, as the amount of data increase dramatically, traditional data analytic platforms confront with storing, managing, and analyzing difficulties. Emerging Big Data Analytics (BDA) overcome these problems by providing decentralized and distributed processing. In this study, we propose two new models for default prediction. In the first model, called DPModel-1, statistical (logistic regression), and machine learning methods (decision tree, random forest, gradient boosting) are employed to predict company default. Derived from the first model, we propose DPModel-2 based on graph theory. DPModel-2 also comprises new variables obtained from the trading interactions of companies. In both models, grid search optimization and SHapley Additive exPlanations (SHAP) value are utilized in order to determine the best hyperparameters and make the models interpretable, respectively. By leveraging balance sheet, credit, and invoice datasets, default prediction is realized for about one million companies in Turkey between the years 2010–2018. The default rates of companies range between 3%-6% by year. The experimental results are conducted on a BDA platform. According to the DPModel-1 results, the highest AUC score is ensured by random forest with 0.87. In addition, the results are improved for each technique separately by adjusting new variables with graph theory. According to DPModel-2 results, the best AUC score is achieved by random forest with 0.89.

1. Introduction

Default of a company can be defined as a failure of paying of its loan, and generally seen as an early sign of bankruptcy. Therefore, default risk is a matter of great importance that can affect related decision-makers such as institutions, investors, fund managers, and governments for effective credit risk management. For example, an accurate prediction may offer useful information for banks in identifying risky customers in their lending activities or for investors in determining credit datasets for bond pricing and portfolio management (Moscatelli, Parlapiano, Narizzano, & Viggiano, 2020). Especially, after the global financial crisis in 2007–2008, accurate and effective default prediction comes into prominence in the field of credit risk management (Barboza, Kimura, & Altman, 2017).

With the rapid growth of financial data, Big Data Analytics (BDA) has become an indispensable part of data science by allowing companies to transform themselves into data-driven organizations (Lee, 2017). However, traditional data techniques and platforms have failed to acquire the demanded information from a massive amount of financial data due to their scalability issues, limited storage, and performance capacities. Therefore, there is a need for a more powerful platform where it is possible to make deep analytics and to provide more accurate and reliable results for extracting the information. To meet the increasing demands, new big data platforms have emerged to compensate storage and distributing needs inexpensively (Oussous, Benjelloun, Lahcen, & Belfkih, 2018). BDA on economic activities and historical data of companies may extract valuable information about company default probabilities so that companies regulate their loans and assets, become competitive, follow new insights, and adapt their services (Narayanan, 2014).

The new trends in financial systems are shifted to the information technologies as the size of machine-readable data produced by companies increases exponentially. Although balance sheet and credit information of companies are pioneer indicators to point a possible future default (Zhou, Lai, & Yen, 2014; Baek & Cho, 2003), in reality, many more factors such as wrong or bad investments, increasing debts, and

* Corresponding author.
E-mail addresses: mustafa.yildirim2@gazi.edu.tr (M. Yıldırım), feyzaokay@gazi.edu.tr (F.Y. Okay), ozdemir@cs.hacettepe.edu.tr (S. Özdemir).

https://doi.org/10.1016/j.eswa.2021.114840
Received 24 August 2020; Received in revised form 28 February 2021; Accepted 1 March 2021
Available online 13 March 2021

broken supply chain can affect the default rates (Zhu, Zhou, Xie, Wang, & Nguyen, 2019). For this reason, financial systems should consider much more information to improve their prediction capabilities.

The process of extraction valuable knowledge of finance requires fine-grained analyzes on the abundant data by statistical methods or machine learning methods. Econometric methods generally use statistical methods and have explanatory power on the results by leveraging their theoretical background. Even if machine learning methods are limited on that power, it has important advantages such as detecting nonlinear relations on the characteristic of the data, and its potential to deal with a huge volume of data (Kim & Kim, 2019).

In this study, two novel Default Prediction Models (DPModel) are proposed using well-known statistical and machine learning methods on a big data platform in a comparative manner. (*i*) The first model, DPModel-1, utilizes the balance sheet and credit datasets, and it aims to improve the predictability of company default. (*ii*) In order to improve prediction results, the second model (DPModel-2) leverages from invoice dataset to take into account trading interactions of companies via graph theory.

DPModel-1 is a powerful and useful model for binary classification problem. The model can be used in many different areas with small changes. The model can be easily applied to several problems especially in the field of finance such as credit scoring, bankruptcy and default predictions. On the other hand, DPModel-2 benefits from graph theory which is frequently used in different areas such as computer, communication, commerce and transportation (Deo, 2017). In addition, it can be used to solve financial problems such as payment systems, commercial relations of companies, and debt relations between banks or countries (Lautier & Raynaud, 2012). Overall, our proposed models can be generalized to solve different classification problems.

During the experimental analysis, different methodologies and technologies are used to improve the prediction capability of the model. (*i*) Grid search is used to optimize the hyperparameters for each method. (*ii*) Different statistical and machine learning methods are performed to predict the company default. (*iii*) A big data platform is employed for distributed processing and analyzing the huge volume of financial data. (*iv*) Graph theory is used to take advantage of the trading interactions a company with other suppliers or customers in addition to its balance sheet and credit information to achieve more accurate predictions. (*v*) SHapley Additive exPlanations (SHAP) value is used to make results are more interpretable. Thus, the effect of each input variables on the success becomes measurable.

Overall, the main contributions of this paper can be listed as follows:

- To the best of our knowledge, this is the first study using graph theory in default prediction. The results of machine learning algorithms are improved with the new variables obtained by graph theory methods that utilize not only the information of companies but also the trading relations with other companies.
- Real financial data including more than one million companies' information in Turkey are employed in the experiments, which provides real assessments of experiments.
- With the leverage of SHAP value, the contribution level of each input to the output has become measurable in detail, unlike machine learning algorithms.
- We use BDA platform to ensure high performance in terms of decision time and computational cost.

The rest of the paper is organized as follows: Section 2 gives a comprehensive review of the state of the arts in default prediction. Then, Section 3 overviews BDA and its advantages on financial data. Section 4 presents methodologies, technologies, and datasets used in our study. Section 5 presents our proposed default prediction models in detail. The results of our models are examined and discussed in Section 6. Finally, summary, concluding remarks, and future directions are presented in Section 7.

2. Related work

Default prediction of companies is commonly studied both in the real world and the literature. Most studies in the literature are carried out by financial experts using traditional statistical methods such as Logistic Regression (LR), Multiple Discriminant Analysis (MDA), Lasso, etc. (Chijoriga, 2011; Bandyopadhyay, 2006; Pereira, da Basto, & Silva, 2016). In recent years, however, the proposed models have shifted to machine learning methods such as Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), Support Vector Machine (SVM), Multilayer Perceptron Networks (MLP), and Artificial Neural Network (ANN) due to their higher prediction capability (Barboza et al., 2017; Wang, 2017; Son, Hyun, Phan, & Hwang, 2019). Default and bankruptcy prediction of companies are highly popular topics examining in the literature with many different techniques and platforms by considering various datasets. Therefore, we divide the related works into two different groups: (*i*) the studies making the default and bankruptcy prediction with machine learning algorithms, and (*ii*) the studies of BDA on financial data.

2.1. Machine learning algorithms for default prediction

For decades, default and bankruptcy prediction has been conducted with Single Discriminant Analysis (SDA) proposed by Beaver (1966) and MDA proposed by Altman (1968). However, criticisms belonging to the needs of a homogeneous variance assumption for both default and solvent companies force new innovations in predicting analysis. LR is introduced as a solution for this problem by offering less restrictive statistical assumptions and better empirical discrimination (Ohlson, 1980; Aziz, Emanuel, & Lawson, 1988).

Since 1990's, different machine learning algorithms such as DT, RF, SVM, MLP, etc. have gained more attention in default and bankruptcy prediction due to their more accurate prediction results. In Kim, Cho, and Ryu (2020), a comprehensive literature review is presented for default prediction of companies with machine learning algorithms. Kim and Sohn (2010) introduce an SVM based model in the aim of distributing funds effectively to small and medium enterprises by looking at their default estimates. According to the experimental results, SVM shows superiority in comparison with LR and backpropagation neural networks. In Zhou et al. (2014), SVM is combined with a new approach based on direct search and feature ranking to overcome the sensitivity problem in bankruptcy prediction. The authors also compare the results with genetic algorithm. Experimental results show that the proposed hybrid model outperforms the single machine learning model in trade-off high computational time. Wang, Ma, and Yang (2014) improve the traditional boosting method and propose FS-boosting (boosting with Feature Selection) for company bankruptcy prediction. Information gain-based filtering feature selection method is used to avoid noisy data which boosting is highly affected. The results prove the effectiveness of FS-Boosting over traditional boosting method. In addition, profile-based bankruptcy prediction model is offered (du Jardin, 2016) using the information containing the firm's experiences that may lead to bankruptcy in the future. According to the experimental results, the proposed model shows a superiority when compared to both single statistical and machine learning models such as MDA, LR, DT, and ANN, also ensemble models such as boosting and bagging. Similarly, in another work (Tsai & Hsu, 2014), MLP, SVM, and DT based on boosting and bagging methods are presented. The study also considers a different number of classifiers. The results show that DT based boosting achieves the best performance than other classifier ensembles. Danenas and Garsva (2015) suggest particle swarm optimization based SVM for bankruptcy prediction. Even if the proposed model is less stable, it offers more potential when compared to multinomial LR and RBF (normalized Gaussian Radial Basis Function) network. Lastly, Barboza et al. (2017) give comparative results for bankruptcy prediction. In their study, first statistical machine learning methods are compared. Then, the new variables are also

included in the prediction process and the impacts of these variables on the prediction results are analyzed in detail. According to the results, bagging, boosting and RF have better performance results than MDA, LR, and ANN.

2.1.1. Deep learning algorithms for financial data

Deep learning has been highly recommended for default prediction recently. The techniques of deep learning are capable of learning the characteristics of data, and they provide implicit learning (Yeh, Wang, & Tsai, 2015). It should be noted that, although deep learning methods offer innovative and pioneering solutions for financial time series. In order to show the success of deep learning methods for our problem, a sample data derived from original dataset is selected and the results are discussed in Section 6.4.1. In Mai, Tian, Lee, and Ma (2019) deep learning models for bankruptcy prediction are proposed by facilitating textual disclosure. The experimental analysis is conducted on the combination of textual and numeric data. When compared to the average embedded model with Convolutional Neural Network (CNN), the embedded model appears to have better prediction results. Hosaka (2019) also uses CNN through imaged financial ratios to predict bankruptcy, and compares with several techniques such as DT, linear discriminant analysis, SVM, MLP, AdaBoost, or Altman's Z-score. The author obtains higher performance results with CNN in comparison with other machine learning methods. Jing, Yan, and Deng (2020) propose a hybrid model consisting of Zero-Price Probability and Long Short-Term Memory techniques to enhance the probability score of default prediction by analyzing daily stock returns. They compare their proposal with Kealhofer, McQuown, and Vasicek (KMV) model and constant variance ZPP. Another study (Yeh et al., 2015) also adopts Deep Belief Networks (DBN) to improve their prediction results according to the SVM. The experiments are conducted on daily stock returns data belonging to both default and solvent companies.

Credit scoring plays a key role in identifying default prediction of companies. Different machine learning-based solutions are offered for credit scoring in the literature. First of all, Nehrebecka (2018) presents a comparative analysis of accurate credit scoring. It employs logistic regression and SVM under the evaluation metrics of GINI index, Kolmogorov–Smirnov (K-S) statistics, and AuROC curve. Experimental analysis shows that the best accuracy is obtained with LR in comparison with different types of SVM. Luo, Wu, and Wu (2017) present a DBN model for credit scoring which is a key role in default prediction on the credit default swaps data. In this model, DBN is combined with Restricted Boltzmann Machines. When compared to other well-known models LR, MLP, and SVM, the proposed DBN model achieves better prediction results in terms of accuracy and AuROC curve. In addition, the study (Addo, Guegan, & Hassani, 2018) utilizes different machine and deep learning techniques for credit scoring of each company. Four versions of deep learning methods with different metrics are compared with the LR, RF, and GB methods. As a result, it is observed that tree-based models are more stable than multilayer neural networks based models. In Zhao et al. (2015) Average Choosing Random Choosing method is offered to enhance the prediction capability of MLP for credit scoring on a German dataset. The authors prove their proposed method improves the performance results.

2.1.2. Evolutionary machine learning algorithms for financial data

Evolutionary ML (EML) benefits of evolutionary computation to cope with the growing amount of data size and deficiencies of domain experts in the complex application fields. The use of EML techniques has increased in the field of finance as in other fields such as manufacturing, agriculture, healthcare and energy systems (Al-Sahaf et al., 2019). Different financial problems such as financial distress prediction (Fallahpour, Lakvan, & Zadeh, 2017), stock market prediction (Chung & Shin, 2018), financial crisis prediction (Uthayakumar, Metawa, Shankar, & Lakshmanaprabu, 2020), and portfolio optimization (Ren, Ye, Huang, & Feng, 2018) are addressed by suggesting various EML techniques. However, we limit these studies with the problem of default and bankruptcy prediction. Since financial data consists of time-series data generally, the analyses of these data can become more difficult. EML with its intuitive approach has the ability of global search by evolving candidate solutions to optimal solutions.

Although EML techniques are used especially in optimization problems, with a few enhancements, they achieve very successful results in prediction problems. Genetic Algorithm (GA) is one of these well-known and powerful EML techniques. The study Gordini (2014) shows the suitability of GA for an effective default prediction of SMEs. When it is compared with the SVM and LR, GA has superior results in terms of increasing accuracy rate and decreasing misclassification rate.Another study (Chou, Hsieh, & Qiu, 2017) propose a hybrid structure by integrating genetic algorithm and fuzzy clustering for bankruptcy prediction. While GA with statistical theory and fuzzy theory based fitness functions are used for financial ratio selection, fuzzy clustering algorithm is used for classifier design. The efficiency of the proposed prediction model is proven with the experimental analysis in a comparative manner with the other well-known classifiers and structures. The study of Kim, Jo, and Shin (2016) proposes a hybrid method including clustering technique and GA-based ANN model to solve the imbalance problem by selecting appropriate instances. Then, the model is applied to the bankruptcy prediction problem. The proposed DA-ANN with cluster based evolutionary undersampling model has better results than its counterparts. In Ansari, Ahmad, Bakar, and Yaakub (2020), Optimization Algorithm (MOA) and Particle Swarm Optimization (PSO) are employed to reduce the complexity of ANN training while predicting bankruptcy of companies. By enhancing MOA with PSO, time-complexity of ANN's weight training is reduced efficiently. Therefore, faster and more accurate prediction results are achieved by ANN. The study of Wang et al. (2017), exploits Grey Wolf Optimization (GWO) to construct a novel Kernel Extreme Learning Machine (KELM) model for bankruptcy prediction. When it is compared with three other KELM models, it outperforms in terms of accuracy, Type I and Type II errors, AUC value and computational time.

2.2. Big data analytics in financial predictions

BDA is a related field of Business Intelligence and Analytics which refers to as techniques, technologies, systems, methods, and tools that enable accessing to diverse data, manipulating and transforming this data to offer managers or analysts productivity and competitiveness. It ensures a better understanding of what is their current status and how it can be grown in the future. It helps improving decision making progress to perform accurate actions in the future (Chen, Chiang, & Storey, 2012; Lim, Chen, & Chen, 2013). BDA can be summarized in five sub-phases: (i) acquisition and recording, (ii) extraction, cleaning, and annotation, (iii) integration, aggregation, representation (iv) modeling and analyzing, and (v) interpreting (Gandomi & Haider, 2015). The first three steps are the pre-processing steps which aim to prepare the data to analyze. The data are cleaned from duplicates, incompleteness, noise, or outliers. Besides, if the data are too complex to analyze, it can be reduced. On the other hand, the last two steps are about extracting hidden patterns, rules, or information, and interpreting the findings of what they mean and how can firms benefit from Tsai, Lai, Chao, and Vasilakos (2015). Since BDA is a relatively recent technology, the application areas in the field of default or bankruptcy prediction are highly limited. Therefore, we review the literature in more general so as to cover financial predictions.

There are some comprehensive reviews on BDA on finance by assessing the impacts of BDA to increase the business performance of companies (Wamba et al., 2017; Akter, Wamba, Gunasekaran, Dubey, & Childe, 2016) or banks (Srivastava & Gopalkrishnan, 2015). To this end, some studies employ different techniques on big data platforms and offer valuable financial results. The authors of Óskarsdóttir, Bravo, Sarraute, Vanthienen, and Baesens (2019) use mobile phone data as a

big data source with credit and debit information to estimate credit scores of customers. In addition to this, social network analysis is applied to define the influence scores of customers. The experimental results prove that including call records in traditional data improves the prediction of credit scores. A failure prediction model for construction firms is proposed by Alaka et al. (2018) with different statistical and machine learning techniques consisting of ANN, SVM, MDA, LR. The experiments are conducted on 693.000 data cells from 33.000 sample healthy and failed construction firms in 2008–2017. R software/platform is selected for the experiments. Experimental results indicate that ANN has better accuracy than other methods. Stockinger, Bundi, Heitz, and Breymann (2019) present a scalable architecture for financial analytics by introducing a real-world financial use case to perform in various financial applications. Different parallel implementations are conducted on Apache Spark, with the discussion of pros and cons. Comparative results are given for financial analytics with user-defined functions and SQL functions. In addition, Wang, Gunasekaran, Ngai, and Papadopoulos (2016) introduce big data for an effective Logistic and Supply Chain Management (LSCM) in the aim of increasing business value. Accordingly, a maturity framework is applied to define the impact of Supply Chain Analytics (SCA) on LCSM according to the different maturity criteria identified from the literature. Then, the main roles of SCA on business value are evaluated and emphasized in the success of company achievements.

Different from above studies, we move machine learning algorithms to the big data platform to handle more diversified data with a huge number of companies with more financial features. In our study, we use various statistical and machine learning methods to compare their prediction successes in each other and combine with graph theory to improve the overall success. In addition, we make the default estimation by using many more companies and variables with the help of big data technology unlike the studies in the literature.

3. Methodologies and technologies

This section gives brief information about methodologies and technologies used in this paper.

3.1. Statistical methods

Statistical methods like LR, ridge, lasso are some of the traditional financial methods that are used for default and bankruptcy prediction (Khemais, Nesrine, & Mohamed, 2016), distress management (Ul Hassan, Zainuddin, & Nordin, 2017; Rahim, Rashid, Nayan, & Ahmad, 2019), enterprise risk management (Olson & Wu, 2017), and predicting stock performance (Ali, Mubeen, & Hussain, 2018) for many years. In this paper, LR is selected as a powerful statistical method to compare machine learning algorithms.

3.1.1. Logistic regression

LR is a popular statistical technique used in the default and bankruptcy prediction (Khemais et al., 2016; Agrawal & Maheshwari, 2019). Recently, it been used as a benchmark method in Nehrebecka (2018) and Cowden, Fabozzi, and Nazemi (2019). It is used to define data and explain the relationship between a dependent binary variable (output value) and a set of independent variables (input values). It is very similar to linear regression. Unlike linear regression, LR generally offers a solution for classification problems in addition to regression problems. It also predicts categorical dependent variables rather than continuous dependent variables (Midi, Sarkar, & Rana, 2010).

LR uses a cost function called logistic or Sigmoid function to map the predictions to probability values between 0 and 1. Accordingly, an output value, Y can be measured by combining input values and coefficients as follow:

Table 1
Decision tree algorithm.

The steps of Decision Tree
Step 1. Calculate the cost function (e.g. Entropy or GINI index)
Step 2. Split the dataset according to the cost function.
Step 3. Evaluate all splits.

$$Y = ln(\frac{P}{1-P}) = b_0 + b_1 \times x_1 + b_2 \times x_2 + \ldots. + b_k \times x_k \quad (1)$$

where P is the Sigmoid function, b_0 is the bias, and each b is the constant coefficient of regarding a single input x_k where k is the number of inputs. Here, Sigmoid P is also measured by the following equation:

$$P = \frac{1}{1+e^{-Y}} \quad (2)$$

3.2. Machine learning algorithms

Machine learning is one of the pioneer advances in applied mathematics. It is highly demanded in many application domains such as finance, healthcare, military, industry, etc. to construct a new or solve existing classification or clustering problems (Alpaydin, 2020). It handles a massive amount of data and constructs a more generalized model to enable accurate predictions and extracting hidden patterns. In literature, there are many different machine learning algorithms are introduced. In this paper, we select some of the most preferred and best achievement machine learning algorithms in default prediction and conduct our analyses via these algorithms (Mohri, Rostamizadeh, & Talwalkar, 2018). Below, we briefly explain these machine learning algorithms.

3.2.1. Decision tree

DT is a well-known supervised learning algorithm that classifies instances starting from a root node to leaf nodes. By utilizing different techniques such as ID3, C4.5, CART, it builds a tree-like structure by dividing repeatedly a single node into two sub-nodes according to certain criteria (Safavian & Landgrebe, 1991).

While constructing the tree, different statistical techniques like information gain and GINI index are employed to determine the best attribute in the aim of optimizing each split with maximum purity (Raileanu & Stoffel, 2004). Information gain contains information about which attribute has the maximum information about the class. ID3 is an example of decision tree using information gain. It is an entropy-based approach that calculates the impurity or randomness of dataset. Due to entropy has log function, it is more computationally heavy (Peng, Chen, & Zhou, 2009). Entropy can be calculated as below:

$$Entropy = -\sum_{i=1}^{c} p(x_i)\log_2 p(x_i) \quad (3)$$

where p is the probability of x_i variable being classified to a certain class c.

The steps of DT can be illustrated as seen in Table 1.

3.2.2. Random forest

RF is a popular classification method consisting of a great number of individual decision trees. It draws its strength from the fact that ensemble prediction is better than each prediction of individual trees. It employs bagging and feature randomness to build an uncorrelated forest of trees so as to operate as an ensemble (Svetnik et al., 2003; Oshiro, Perez, & Baranauskas, 2012). To overcome the fragility of DT to the small changes in training data, RF uses bagging to construct different trees by allowing each individual trees to sample randomly from training data with replacement. In addition, it uses feature randomness to split the tree depending on the features. Different from DT, it selects a

random feature subset that ensures a lower correlation and more diversification (Ali, Khan, Ahmad, & Maqsood, 2012).

The steps of RF can be summarized as follows (Table 2) (Yeh, Chi, & Lin, 2014):

3.2.3. Gradient boosting

GB is a machine learning method for both classification, regression, and ranking problems. The basic idea of GB is originated from AdaBoost proposed by Freund and Schapire (1996). GB trains many models gradually, additively, or sequentially in the aim of transforming base learners into strong learners. To do that, it controls loss function. Until loss function is minimized, GB updates its prediction by adding new models. It generally uses DT as base learners, and constructs new trees until it is overfitting or reaching the optimal results (Friedman, 2002; Natekin & Knoll, 2013). It uses loss function to calculate pseudo-residuals. We use log loss in the experiments that are seen in the following equation:Table 3

$$Loss = 2\sum_{i=1}^{N} log(1 + exp(-2y_i F(x_i))) \quad (4)$$

where N is the number of instance, y_i is the feature of instance i, and $F(x_i)$ is the predicted label for instance i.

3.3. Grid search optimization

Most of the machine learning algorithms proposed for default prediction in the literature are parameter-dependent (Zhang, Yang, & Zhou, 2018). Especially in the methods like DT, GB, and RF, certain parameters have a huge impact on prediction score.

Grid search algorithm is a parameter optimization technique that tries to find the best hyperparameter set. It identifies the parameters with the best accuracy by trying all combinations determined at a certain frequency. It generally combines with machine learning algorithms to increase the performance results (Wang et al., 2016; Behera & Nain, 2019). It should be noted, in cases where the experiment frequency is not set very well, it may not find the best parameters. Nevertheless, it can find the near-best parameters.

3.4. Big data technologies

The rapid increase in the amount and speed of data and changes in storage, processing, analysis, and evaluation techniques have created new technological demands. Handling big data encounters many challenges such as collecting data in different formats and from different data sources, managing, storing, distributed/parallel computing, real-time stream data analysis of a massive amount of data volume (Labrinidis & Jagadish, 2012; Chang & Zhang, 2014). In order to cope with those challenges, new techniques and materials are required. However, traditional data management systems have failed to handle the data since it is not possible to analyze the data on a single computer. Instead, a decentralized and distributed computing architecture with more than one computer is required. Big data technologies offer great opportunities in providing distributed and decentralized control on large-volume, heterogeneous, and ubiquitous sources by exploring the complicated relations among the data (Wu, Zhu, Wu, & Ding, 2013).

This paper performs prediction analysis by taking advantage of big data technologies which are briefly explained below.

Table 2
Random forest algorithm.

The steps of Random Forest
Step 1. Select random samples from the dataset.
Step 2. Construct a decision tree model from each sample. Then, calculate prediction results for each decision tree.
Step 3. Vote for the results of each model.
Step 4. Select the most voted prediction result as the final prediction result.

Table 3
Gradient boosting algorithm.

The steps of Gradient Boosting
Step 1. Initialize a simple model like decision tree.
Step 2. Calculate the error residuals for every sample by taking the difference of target value and predicted value.
Step 3. Calculate the new residuals with same input variables
Step 4. Add new residuals with the previous residuals.
Step 5. Repeat the steps of 2 to 5 until it starts overfitting or residuals become constant.

3.4.1. Apache spark

It is an open-source big data library used in different domains, especially in the financial sector (Zaharia et al., 2016). Data analysts commonly use this technology to extract, transform, load, and stream data. Hence, meaningless big data can be used to develop the current or future potential of companies by minimizing the risk and maximizing the profit.

It has Resilient Distributed Dataset (RDD) operator that realizes continuous operations with very high performance especially in unstructured data. The implementations can be performed with Scala, Java, R, Phyton languages (Salloum, Dautov, Chen, Peng, & Huang, 2016). The core module called Spark Core contains four modules including Spark SQL, Spark Streaming, MLlib, and GraphX. Each module has a different responsibility during data analytics. Spark SQL is used for structural data processing. Spark Streaming enables scalable, efficient, and fault-tolerant stream processing. MLlib (Meng et al., 2016) is a library for machine learning algorithms in order to perform classification, regression, and filtering. Finally, GraphX (Gonzalez et al., 2014) offers interactive graph-based calculations.

3.4.2. HDFS

HDFS (Hadoop Distributed File System) is a distributed file system, which is capable of handling a huge volume of data by allowing distributed processing and storage. The HDFS is designed by splitting a single server into many commodity servers called nodes. Each node in the system has its processor and RAM. Therefore, rather than gathering all data to a central node, it distributes the SQL-based queries to the nodes and allows to process in those nodes. Thus, it provides concurrent processing and fault-tolerance. In addition, it avoids unnecessary traffic in the system (Borthakur, 2008).

3.4.3. Hive

It is a distributed data warehouse software system running on top of the Apache Hadoop to facilitate summarization, querying, and analyzing large dataset in Hadoop Distributed File System (HDFS) (Thusoo et al., 2009).

3.4.4. Hue

It is an open-source web-based SQL editor to access to a remote warehouse. It enables an interactive graphical representation of SQL results. Thus, operational and configuration changes or errors in HDFS become more apparent (Al Rasyid, Yuwono, Al Muharom, & Alasiry, 2016).

3.5. Graph theory analysis

Graph theory is a branch of mathematics examining the structures to model relationships of objects. Diagrams showing these relationships are called graphs, and they are usually represented by G notation. The basic mathematical representation of the graph is $G(V, E)$ where V represents a set of vertices, and E represents a set of edges here. According to the direction matters, a graph may be directed and undirected (Freeman,

1978; Hage & Harary, 1995). Social friendship network can be given as an example of an undirected graph. In this network, if a person A is a friend with another person B, then B must be a friend with A. On the other hand, tracking mechanisms in social networks is an example of a directed graph. For example, if person A follows person B, however, this does not mean that person B has to follow A. Person B can follow either person A or no one.

Another feature of the graph is whether the edges are weighted or not. Edge weight is a numerical value for the relationship between nodes. Usually, in weighted graphs, the weight is indicated by the thickness of the edge. The mathematical notation of the weighted graph is $G(V, E, W)$ to represent weight W.

The importance of each object or node in the graph is calculated by centrality. More literally, the centrality is a unit that shows how central a node in the graph is. By looking at the positions of the nodes in the graph, it measures how many connections a node has, and which nodes it is connected (Borgattia & Everett, 2006). There are some popular centrality measurements in the literature which are degree centrality, closeness centrality, betweenness centrality, and Eigenvector centrality.

3.5.1. Degree centrality

It is the number of edges that are connected to a node. In directed graphs, two different degrees can be calculated, namely internal degree (indegree) and external degree (outdegree). The indegree indicates the edges entering the node, whereas the outdegree shows the edges leaving from the node (Zhang et al., 2011).

A degree centrality C_D can be calculated by the following equation:

$$C_D(x) = \frac{\sum_{y=1}^{N} a_{xy}}{N-1} \tag{5}$$

where N denotes the number of nodes on the graph, and a denotes a value of 0 or 1 according to existence of an edge between x and y.

3.5.2. Closeness centrality

It shows the degree of proximity of a node to all other nodes in the network. This metric shows the capability of a node according to how quickly can reach other nodes. If a node is located more central, it would be closer to the other nodes. Closeness centrality is calculated by reversing the average of the shortest path length of one node to other nodes (Okamoto et al., 2008).

A closeness centrality C_C can be calculated by the following equation:

$$C_C(x) = \frac{N-1}{\sum_y d(y,x)} \tag{6}$$

where N denotes the number of nodes on the graph, and $d(y,x)$ denotes the distance between the node y and the node x.

3.5.3. Betweenness centrality

It measures the number of the shortest paths where a node is positioned on it. Therefore, more data flow goes through that node. This metric shows the importance of a node in the connection of the graph (Tizghadam & Leon-Garcia, 2010).

A betweenness centrality C_B can be calculated by the following equation:

$$C_B(x) = \sum_{u \neq v \neq x} \frac{\sigma_{uv}(x)}{\sigma_{uv}} \tag{7}$$

where σ_{uv} denotes the number of the shortest path between the node u and the node v, and $\sigma_{uv}(x)$ denotes the number of the shortest path between the node u and the node v which pass through the node x.

3.5.4. Eigenvector centrality

It is calculated not only by the number of nodes it is connected but also by the importance of the nodes to which it is connected. To give an example from social networks, it considers not only how many friends a person has, but also how many friends his/her friends have. This metric is also known as prestige (Taylor, Myers, Clauset, Porter, & Mucha, 2017).

It is one of the most useful criteria in real life simulations is Eigenvector centering. Eigenvector centrality C_E can be calculated by the following equation:

$$C_E(x) = \frac{1}{\lambda} \sum_{y \in M(x)} C_E(y) = \frac{1}{\lambda} \sum_{y \in G} a_{x,y} C_E(y) \tag{8}$$

where G is a graph. In this graph, λ denotes the eigenvalue, $M(x)$ denotes a set of neighbors to the node x, y denotes a neighboring node, and $a_{x,y}$ denotes a value of 0 or 1 depending on whether node x and y are neighbours.

3.6. SHAP value

Although recent models such as deep learning or machine learning have higher predictability than traditional models, their interpretability is low due to their complexity. For this reason, these algorithms are seen as a "black box". One of the most important criticisms especially in machine learning studies on financial data is the difficulty of interpreting these algorithms. To overcome this problem, various models such as LIME (Ribeiro, Singh, & Guestrin, 2016), interpretML (Nori, Jenkins, Koch, & Caruana, 2019), DALEX (Biecek, 2018) have been proposed in the literature. However, in recent years, SHapley Additive exPlanations (SHAP) value (Lundberg & Lee, 2017), derived from Shapley value concept, has been used more frequently in the interpretation of complex models.

The purpose of SHAP value is to calculate the contribution of each attribute to explain the estimation of a sample. Shapley value and LIME as an additive explanation method are combined to construct SHAP value. In order to interpret complex models, LIME proposes a simpler and interpretable new model, such as linear models, by creating a simplified dataset than the model's prediction results (Mokhtari, Higdon, & Başar, 2019). SHAP value combines these two methods with the formula below.

$$g(z') = \Phi_0 + \sum_{j=1}^{M} \Phi_j z_j' \tag{9}$$

where g is an explanatory model, M is the maximum number of attributes, and $z' \in [0,1]^M$ is the simplified new dataset. Lastly, $\Phi_j \in R$ is the contribution of the j attribute, which means that it is the value of shapley.

SHAP value has three main contributions which are listed below (Lundberg & Lee, 2017).

- Global Interpretability: On the whole dataset, the positive or negative contribution of each attribute to the prediction score can be measured.
- Local Interpretability: It can be measured how the attribute values contribute to the prediction score of each observation in the dataset separately. As with linear models, the traditional attribute importance algorithm only gives importance value of a global attribute over the entire dataset, whereas SHAP value can give importance value separately for each observation. Therefore, it can be used for interpretable linear models as well.
- Tree-based Models: It can be used in tree-based models.

Table 4
The changes in company numbers and default rates by years.

Yil	Number of Companies	Default Rates
2010	561.040	0.066325
2011	588.928	0.065607
2012	597.359	0.052610
2013	610.013	0.064018
2014	627.342	0.058156
2015	654.201	0.047626
2016	681.753	0.045043
2017	708.775	0.047369
2018	761.498	0.033091

4. Dataset

The datasets used in this study are obtained from the Central Bank of the Republic of Turkey (CBRT). As our prediction, the default is defined as follows: (*i*) 90 days late credit repayments, (*ii*) Check and Bond payments are not paid, and (*iii*) Legal bankruptcy or concordat.

In case any of the above situations would happen in a year, companies are deemed to be default. In detail, if a company has declared legal bankruptcy or concordat, it is marked as default for the next year. However, in case a company, whose payments are delayed or checks and bonds have not paid, has made their payments, it is marked as healthy for the next year. Otherwise, if the company has not made their payments, it is marked as a default company for the next year.

Three important datasets are used to predict default which are credit, balance sheet, and invoice datasets. Certain features, which are mostly used in the literature, are calculated on these datasets.

The balance sheet table created by calculating the balance sheet items of the companies consists of 61 features. These features include Altman Z score (Altman, 1968) features, and they are frequently used for default prediction in the literature. These features can be divided into four basic categories which are liquidity, financial position, turnover, and profitability ratios.

The credit table created by credit dataset contains 57 features. Since the credit dataset is at a monthly frequency, the annual average of the features are calculated. These features can be divided into seven basic categories which are credit/limit, limit qualification, factoring, payment/nonpayment, restructuring, and banking relationships.

Invoice dataset is another dataset that constitutes the innovative part of the study. It includes the price incurred during the purchase of goods and services that companies make with each other or with real people. According to Turkish commercial laws, it is compulsory to notify the purchases over 5.000 TRY in total, excluding VAT per month. This dataset includes the tax number of the selling company, the sector of the selling company, the tax number of the buying company, the sector of the buying company (if the buyer is the person, the tax number and the sector of that buying company are coded as -1), and the price of the goods or services. Since the dataset has a monthly frequency, the prices between the two companies are summed up so as to be annual.

The experimental analyzes are carried out using data between 2010 and 2018. It contains massive information about 1.016.315 different companies. The distribution of the number of companies and default rates by years is as in Table 4.

5. The proposed default prediction models

This study aims default prediction of companies for next year. For this aim, we utilize many technologies including statistical and machine learning methods to estimate whether a company would go into default or not, grid search to optimize the hyperparameters, graph theory to analyze the relational information among companies, and finally SHAP value to interpret the contribution of each feature to the results.

The current or future status of a company depends not only on its assets, debts, or credits; but also its trading activities with other companies. While the default of a company may put other firms trading with this company in a difficult situation, it may also provide advantages to competitors. For this reason, it is important to consider the relationships with other companies during default prediction. In order to show the effects of trading interactions of companies with each other, we benefit from graph theory.

According to the usage of graph theory, two default prediction models are proposed. In the first model (DPModel-1), only balance sheet and credit data of companies are used. In the second model (DPModel-2), invoice data is also considered to analyze the relational trading information among companies.

In both models, various statistical and machine learning methods are used to predict the default probability of companies. Since the success rates of these algorithms vary according to their hyperparameters, grid search algorithm is applied to these hyperparameters separately to find out the optimal hyperparameters that will give the best results. Here, hyperparameters are determined automatically rather than manually.

To compare the success of each model, the result metrics such as AUC, accuracy, precision, recall, and F1 score are calculated for each model. Since the default rates of companies range between 3%-6% by year, imbalance data classification problem occur. Therefore, when comparing the success of these models, the AUC metric instead of accuracy is selected.

In the literature, there are many different statistical and machine learning based classification methods for default prediction. In this study, LR from statistical methods, DT, RF and GB algorithms from machine learning methods are selected for the experiments. The following information are the justifications of choosing these algorithms:

- While constructing models, we consider the frequency of use and success rates of each ML method in the literature. LR algorithm is the benchmark algorithm commonly used for default prediction (Ciampi & Gordini, 2013; Moula, Guotai, & Abedin, 2017; Figini, Savona, & Vezzoli, 2016). Because its success rate is higher than other statistical methods and its results are interpretable. Also, our selected machine learning methods are also frequently used in the literature and show high success rates too. Especially RF and GB algorithms outperforms in many studies (Boughaci & Alkhawaldeh, 2020; Chang, Chang, & Wu, 2018).
- Feature selection is not applied in this study, although a total of 115 variables are used. Feature selection is applied generally before modeling in many studies with a large number of features. However, it is inevitable to lose some information with feature selection. Also, it adds extra computational cost. For these reasons, feature selection is not preferred for this study. Instead tree-based machine learning methods are chosen. Due to the nature of tree-based algorithms, it can work without the need for any feature selection mechanism. Furthermore, variables that decrease the success rate are penalized by changing the elasticNet parameter in the LR algorithm with grid search (ElasticNet $= 1$ is Lasso, ElasticNet $= 0$ is Ridge). Thus, there is no need for feature selection for the LR algorithm.
- The study is conducted on a BDA platform. Since the BDA platform is a distributed architecture, many machine learning algorithms have not been implemented yet to perform on this architecture. For example, XGBoost algorithm is a successful algorithm and highly applied for prediction problems; however, it is not yet available in the Spark ML library. That limits us to select such ML methods for our model.

5.1. DPModel-1: the default prediction model

Balance sheet and credit data are often used in the literature for default prediction. The balance sheet shows the assets of the companies and the source from which these assets are obtained annually, whereas

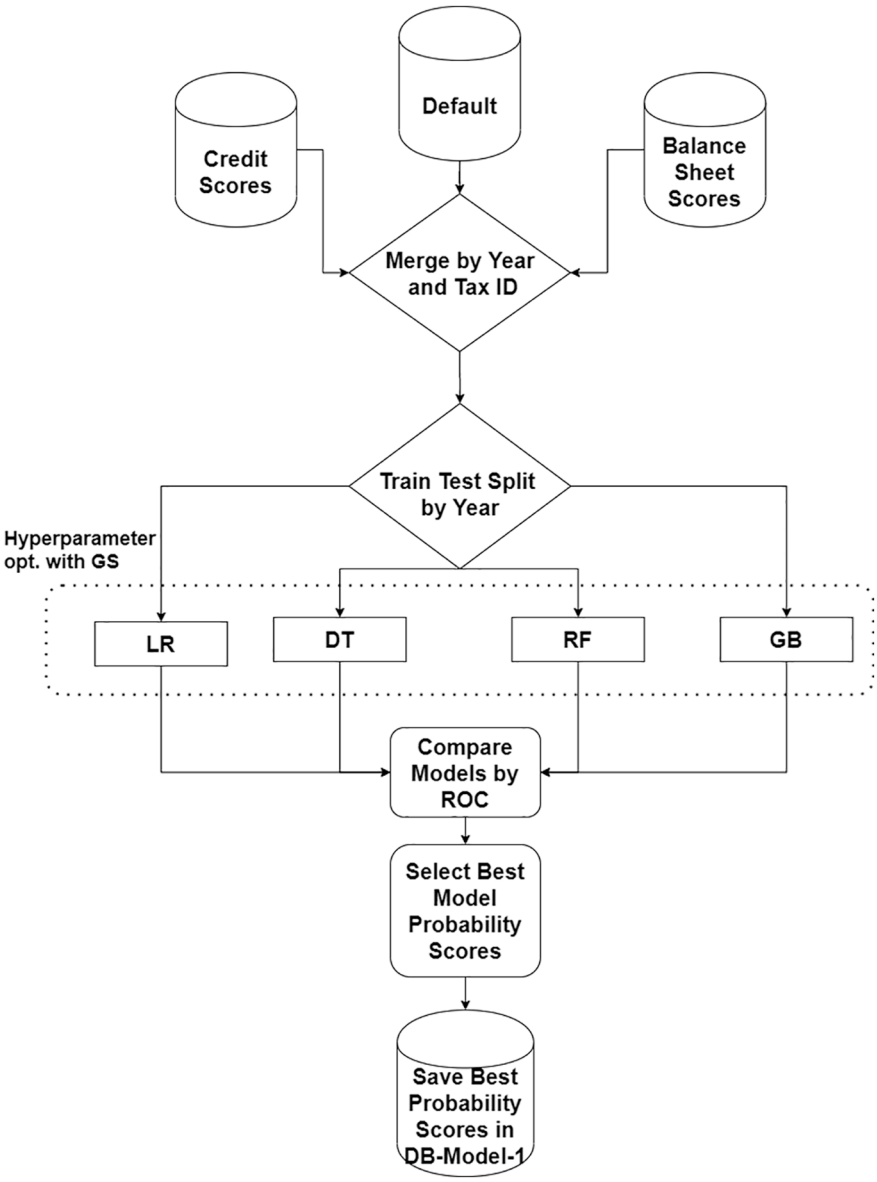

Fig. 1. The flowchart of DPModel-1.

credit data shows the financial status of companies.

In this model, the estimation is performed on balance sheet and credit data. First of all, various features/variables are calculated from the companies' balance sheet and credit data at the annual frequency, which is frequently used in the literature. 61 balance sheet variables are obtained from balance sheet dataset between 2010–2018. These variables are combined with the annual default data by using the year and tax numbers. This dataset containing time information are not randomly divided into training and test data like standard machine learning studies. In order to simulate real-life, next year's data is estimated with the previous year's data. For this reason, although our data started in 2010, our estimates are from 2011. In addition, by following the same steps with balance sheet dataset, 57 credit variables are obtained from credit dataset between 2010–2018. Lastly, empty values are filled with zero value for all variables.

Instead of combining balance sheet and credit variables, they are evaluated separately over the sub-models derived from DP-Model-1. This is because not all companies use credit data. Therefore, when the variables obtained from both datasets are combined, a huge amount of missing values occur, which are difficult to handle. In both sub-models, LR as a statistical method, and DT, RF, and GB algorithms as machine learning methods are used. Grid search is used to assigned optimal hyperparameters to reach the best results. Then, the obtained results of each sub-model are combined and analyzed by the model again as shown in Fig. 1. The final DPModel-1 results for each method are evaluated in terms of AUC, F1-score, accuracy, precision and recall.

5.2. *DPModel-2: the improved default prediction model with graph theory*

For companies, in addition to balance sheet and credit data, suppliers and customers that are in trading relationships are other important factors for an effective default prediction. These relationships of companies can be determined by invoices. Therefore, in this paper, invoice dataset of Revenue Administration is used at monthly frequency. This data is then recalculated annually. Due to the supply chain, invoice data creates a very complex network structure. However, it contains very valuable information for companies. A company in default would damage first the other companies in the supply chain. We model this complex structure using graph theory. Our model is simulated with the help of Spark's GraphX library. In the generated network, nodes

represent companies, and edges represent the total amount of invoices between two companies. As seen in Fig. 2, a directed network is created by selling companies to buying companies. A weighted score is calculated on the network according to customers and new variables obtained from centrality to improve default estimation.

The weighted customer score is the first variable obtained from the network analysis with graph theory. Each company has its customer obtained from the invoice dataset. The default scores of the majority of customers on the DPModel-1 are already known. Also, most of the remaining part consist of small companies, which they operate in the retail sector and do not sell 5.000 TRY or more excluding VAT monthly at once. For weightining such customer scores, the average score of the sector in which the customer scores in a company are ambiguous is used. If the company exporter and the customer's sector are also ambiguous, the weighted customer score is calculated by the sector score in which the company operates. Sector scores are obtained as the average sector score by combining the scores received by companies in the DPModel-1 with the NACE code which is a European industry standard classification system. Sector score averages are calculated annually. Algorithm 1 shows the pseudocode of the customer scores.

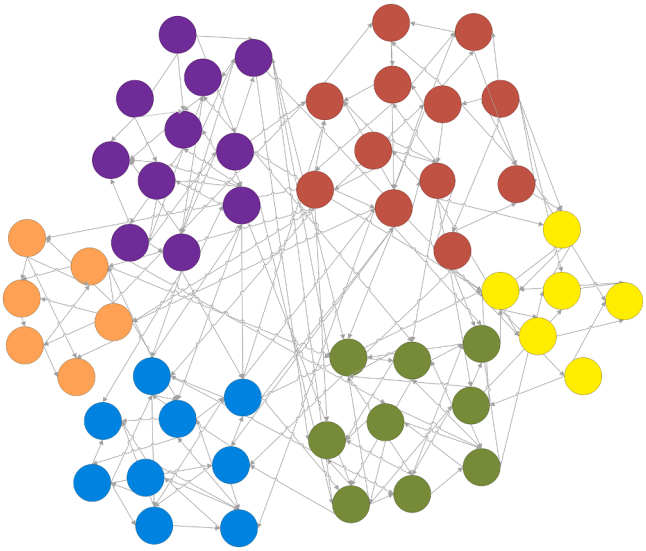

Fig. 2. An example of directed graph generated by a graph theory technique.

Algorithm 1: Pseudocode for the customer score.

```
Input: DPModel-1 scores and sector codes (NACE)
Output: Customer scores by year and sector
for Each year do
    for Each company do
        Customer list: Find the customers over the
          network;
        for Each customer in the customer list do
            if Customer has a score then
                Customer score = DPModel-1 score;
                break;
            end
            if Customer has no score but has a sector
              name (like micro retailers) then
                Customer score = Average sector score;
                break;
            end
            if Customer has no score and has no sector
              name (like exporters) then
                Customer score = Average score of the
                  company's sector;
                break;
            end
        end
        for Each customer do
            Calculate customer score;
        end
    end
    for Each company do
        Calculate customer score;
    end
end
for Each year do
    Calculate customer score;
end
```

Each customer scores can be obtained separately according to the pseudocode. The weighted customer scores (*WCS*) for each company on the graph are calculated. As shown in the equation below, the weighted customer score, *WCS*, for each company is obtained by the ratio of the sales of the company's customers to the total sales.

$$WCS = \frac{\sum_{i=1}^{n} CS_i \times \frac{IA_i}{\sum IA}}{n} \tag{10}$$

where n indicates the number of customers of the company, CS_i indicates the i_{th} customer score, and IA_i indicates the i_{th} customer's total invoice amount in a year.

In addition to weighted customer score, this study also calculates new variables from centrality measurements. Graph theory with centrality indicators has a great influence in determining the important node in the network. On the network, we generated through the invoice information of companies, in case a company in default, it allows us to see how other companies will be affected.

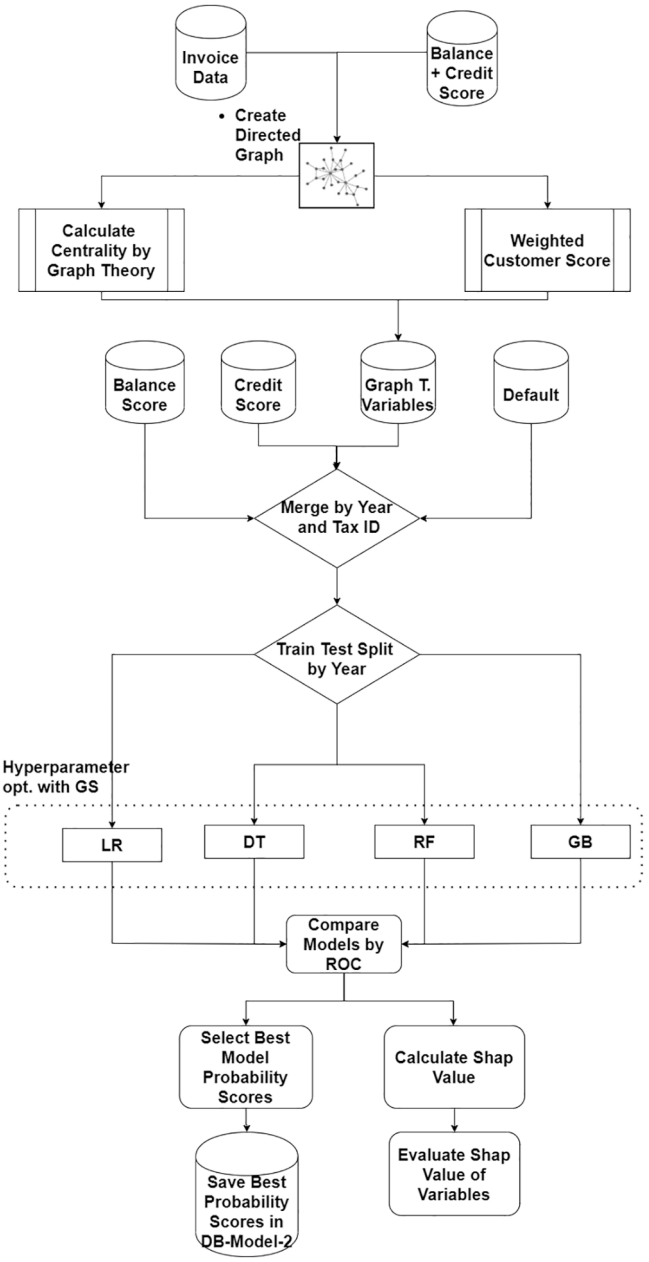

Fig. 3. The flowchart of DPModel-2.

Table 5
Confusion matrix.

		Actual Values	
		Positive	Negative
Predicted Values	Positive	TP	FP
	Negative	FN	TN

In the network, the degree centrality as indegree and outdegree, closeness centrality, betweenness, and eigenvector centrality are calculated separately. Thus, centrality variables obtained by the network analysis are generated for each company.

As can be seen in Fig. 3, our improved model uses the weighted customer score and centrality variables obtained by graph theory in addition to balance sheet and credit probability scores which are obtained from DPModel-1. Then, these variables are combined with default data obtained from DPModel-1 on year and tax number. The improved model success is evaluated with LR, DT, RF, and GB methods in terms of AUC, F1-score, accuracy, precision, and recall. Also, results of methods are compared with the best results of DPModel-1.

6. Performance evaluation and results

This section gives detailed information about the experimental analysis for default prediction in a comparative manner.

6.1. Experimental setup

All experiments are performed on 3.1.5 version of Hortonworks Data Platform. This platform contains 1 edge node, 3 main nodes, and 4 data nodes, and each of them has 256 GB RAM, 2 × 18 core Intel (R) Xeon (R) Gold 6154 3.00 GHz processor. 3.1.1 version of HDFS, a distributed file system, is used on the platform. The data are kept as hive tables that can store large datasets distributed and query with SQL. Hue SQL assistant is used to easily send SQL queries on Hive tables. Spark 2.3.2 is used for machine learning and graph theory models. To use Spark models, it is benefited from the PySpark library. Besides, the python 3.8 interpreter is used. As an editor, Zepplin notebook 0.8.0 is used.

6.2. Evaluation metrics

Different metrics can be used to evaluate the results of classification models. As in this study, AUC metric, which is frequently used in comparing models created with unbalanced datasets, is determined. The area under the ROC curve is called AUC. ROC is a probability curve that shows the False Positive Rate (FPR) on the x-axis and the True Positive Rate (TPR) on the y-axis. FPR and TPR are calculated as follows:

$$FPR = \frac{FP}{FP + TN} \tag{11}$$

$$TPR = \frac{TP}{TP + FN} \tag{12}$$

where FP, FN, TP, TN represents the sample of a false positive, false negative, true positive, and true negative values, respectively.

It is considered to be the summary of the AUC model performance. The greater the AUC value, the better the model is in distinguishing classes. In an ideal model, AUC is 1.

In addition, precision, recall, accuracy, and F1 score are used to evaluate the success of each proposed model. These evaluation criteria are calculated on the confusion matrix as shown in Table 5.

The confusion matrix is used to compare the actual values and the predicted values by the algorithm. The precision is calculated on the confusion matrix as the following equation (Eq. (13). Precision shows the proportion of how many are actually positive from the algorithm

Table 6
The average success of DPModel-1.

Model	Threshold	AUC	F1-Score	Accuracy	Precision	Recall
LR	0.08	0.762	0.218	0.838	0.145	0.443
DT	0.08	0.801	0.249	0.798	0.153	0.655
RF	0.12	0.818	0.264	0.845	0.174	0.545
GB	0.20	0.815	0.319	0.928	0.306	0.334

Table 11
The critical analysis results with deep learning algorithms.

Model	AUC	F1-Score	Accuracy	Precision	Recall
RNN	0.790	0.140	0.883	0.717	0.077
GRU	0.762	0.164	0.880	0.550	0.096
LSTM	0.803	0.228	0.886	0.676	0.137

predicts positively.

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

Recall is the ratio of how successfully the algorithm predicts positive values. It is calculates as in Eq. (14).

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

Accuracy shows the accuracy rate of the prediction, and it is calculated as in Eq. (15).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

Lastly, F1 score is the harmonic mean of precision and recall.

$$F1_{score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{16}$$

6.3. Experimental results

The experimental results are explained by splitting into three main categories including the results of the DPModel-1, the results of DPModel-2, and finally the examining of SHAP value and the contribution of the variables of these models to the results (Table 6).

6.3.1. Results for DPModel-1

The goal of the DPModel-1 is to show the superiority of machine learning methods over statistical methods in default prediction. Table 11 shows the experimental results for different evaluation metrics without the consideration of any year. Besides, in Fig. 4, ROC curves, and comparison matrices belonging to each model are demonstrated. Accordingly, RF has the best estimation performance among others. On the other hand, LR has the lowest AUC result.

When measuring classification metrics, the threshold value is set to 0.5. However, this reduces the F1 scores and accuracy for unbalanced datasets. For this reason, our study first calculates the optimal threshold

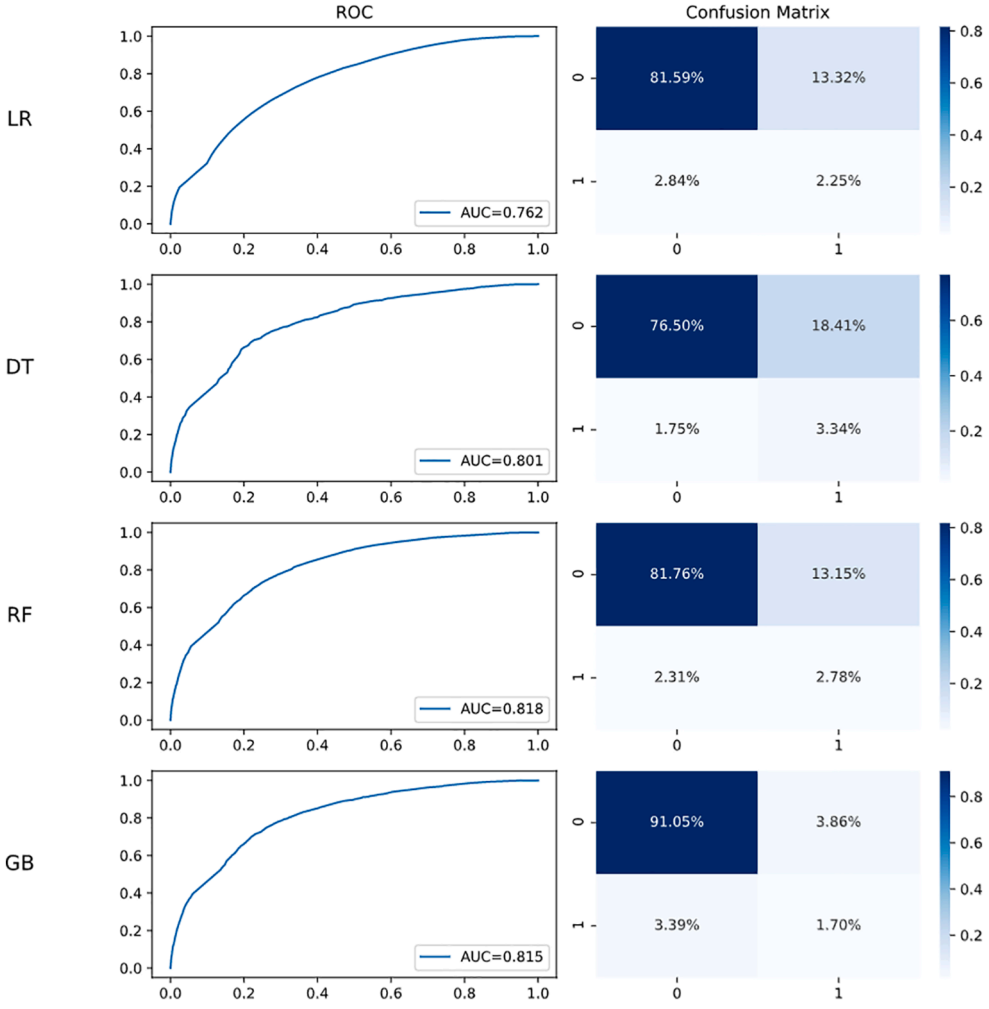

Fig. 4. ROC curves and confusion matrices for each model in DPModel-1.

Table 7
The success of DPModel-1 by years.

Year	Model	Threshold	AUC	F1-Score	Accuracy	Precision	Recall
2011	LR	0.12	0.599	0.152	0.363	0.083	0.87
2011	DT	0.13	0.599	0.152	0.363	0.083	0.87
2011	RF	0.13	0.599	0.152	0.363	0.083	0.87
2011	GB	0.17	0.599	0.152	0.363	0.083	0.87
2012	LR	0.07	0.743	0.227	0.876	0.169	0.348
2012	DT	0.11	0.776	0.296	0.927	0.299	0.292
2012	RF	0.09	0.797	0.302	0.912	0.259	0.362
2012	GB	0.14	0.792	0.301	0.904	0.244	0.394
2013	LR	0.06	0.771	0.278	0.875	0.22	0.376
2013	DT	0.08	0.8	0.358	0.894	0.292	0.461
2013	RF	0.08	0.823	0.375	0.916	0.356	0.395
2013	GB	0.14	0.819	0.371	0.918	0.365	0.377
2014	LR	0.12	0.785	0.279	0.89	0.225	0.366
2014	DT	0.33	0.833	0.357	0.91	0.305	0.43
2014	RF	0.27	0.842	0.362	0.9	0.288	0.487
2014	GB	0.20	0.84	0.347	0.878	0.252	0.557
2015	LR	0.10	0.795	0.268	0.922	0.243	0.297
2015	DT	0.17	0.837	0.341	0.934	0.326	0.356
2015	RF	0.18	0.852	0.363	0.926	0.309	0.439
2015	GB	0.21	0.846	0.36	0.936	0.341	0.381
2016	LR	0.08	0.812	0.294	0.93	0.271	0.322
2016	DT	0.07	0.847	0.32	0.873	0.211	0.661
2016	RF	0.15	0.866	0.397	0.936	0.347	0.464
2016	GB	0.17	0.865	0.382	0.942	0.369	0.396
2017	LR	0.09	0.826	0.32	0.929	0.292	0.354
2017	DT	0.11	0.849	0.381	0.908	0.28	0.596
2017	RF	0.11	0.877	0.411	0.927	0.332	0.54
2017	GB	0.17	0.869	0.419	0.929	0.341	0.544
2018	LR	0.14	0.826	0.289	0.953	0.288	0.29
2018	DT	0.26	0.848	0.334	0.949	0.293	0.387
2018	RF	0.18	0.868	0.34	0.937	0.261	0.488
2018	GB	0.22	0.866	0.338	0.934	0.253	0.51

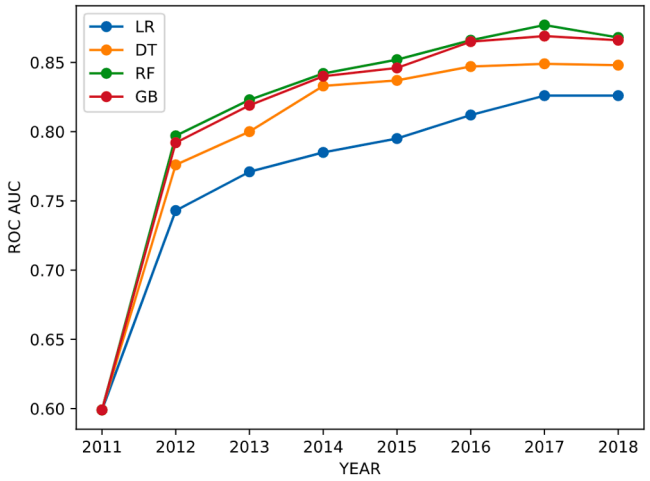

Fig. 5. Comparative ROC curves of each model in DPModel-1.

value that makes the F1 score maximum and then calculates all metrics through this threshold value. According to the results, the RF algorithm predicts better with 0.81 AUC.

During experiments of DPModel-1, training and test data are separated by years. Table 7 shows the success rates of the models by years.

RF algorithm achieves the highest estimation in general by years. Also, the success rate of the models is increased by years. This situation stems for two reasons. The primary reason is data quality. It is seen that the quality of the data collected by years increases. Furthermore, another reason is the increase in the number of companies by years. The increase in the number of companies indicates that the model is better trained. On the other hand, the decline in 2018 is due to the lack of credit data for this year. Fig. 5 shows the trend of the increase for each model.

Table 8
The average success of DPModel-2.

Model	Threshold	AUC	F1-Score	Accuracy	Precision	Recall
LR	0.08	0.763	0.219	0.838	0.145	0.446
DT	0.18	0.825	0.326	0.919	0.282	0.387
RF	0.19	0.828	0.334	0.928	0.316	0.355
GB	0.18	0.823	0.317	0.912	0.263	0.4

6.3.2. *Results for DPModel-2*

The goal of the DPModel-2 is to show that graph theory enhances the results of DPModel-1.

The balance sheet and credit scores in DPModel-1 are combined with the variables obtained from graph theory as shown in Fig. 3, and predictions are calculated according to four different methods again. Table 8 shows the success rate of the DPModel-2 regardless of the year. Also in Fig. 6 illustrates the ROC curves and comparison matrices for each method. DPModel-2 is run separately for each year. The results obtained by years are given in Table 9. Experimental results reveal that the RF algorithm has the highest success.

Considering results for DPModel-2, there is generally an increase in AUC between 1%-4% compared to results for DPModel-1. This increased rate is similar in accuracy results. This shows that the probability of companies default is also related to the default probability of their customers. This study reveals that graph theory is a powerful and efficient technique to analyze this relationship.

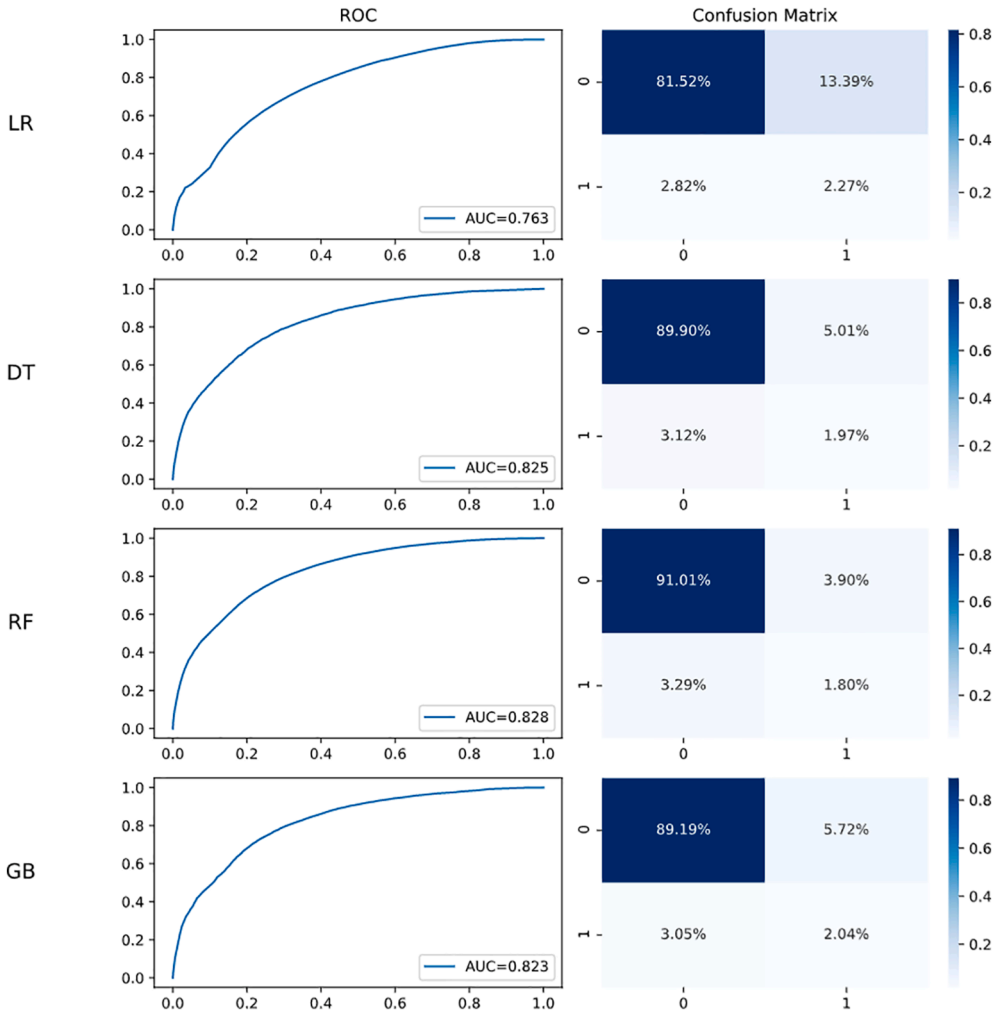

Fig. 6. ROC curves and confusion matrices for each model in DPModel-2.

6.3.3. *Results for SHAP value*

An important issue in the financial datasets is that input variables should be interpreted. However, complex methods such as machine learning on financial data cannot interpret that. SHAP value is an important innovation to interpret the contributions of input variables to the prediction results. In this study, the SHAP values of the variables are calculated for the best methods (RF algorithm has superior success for all models) in sub-models of DPModel-1 and DPModel-2. In addition, since SHAP value implementation details have not been available on the big data platform yet, it is calculated only by random sample selection from 2017 year data.

Fig. 7 shows the SHAP value results for sub-models in DP-Model-1 separately. Here, Fig. 7(a) presents the results for balance sheet data with 61 variables, while Fig. 7(b) presents the results for credit data with 57 variables.

The figures show only 10 variables that make the highest contribution to the estimation. The top line shows the variable that provides the highest contribution. Each point in the graph represents the SHAP value for a sample and an attribute. The color of the dots indicates the value of the features from low to high. The figures are obtained from the results of the RF algorithm since it has the best prediction results. As seen in Fig. 7, "*total loans to total assets*" variable provides the highest contribution among balance sheet variables, whereas "*ttl_accrd_int_amt*" variable which means "Total accrued interest amount" provides the highest contribution among credit variables.

The SHAP value results for the RF algorithm in DPModel-2 is as seen in Fig. 8. According to the figure, "*credit_score*" and "*balance_score*" provide the highest contribution. Besides, "*in_degree_centrality*" which means the number of customers and "*the weighted customer_score*" obtained from the graph theory are the subsequent variables which contribute to the results.

6.4. *Result analysis and discussion*

RF algorithm achieves the highest success according to AUC score in DPModel-1, where machine learning and statistical methods are compared using credit and balance sheet datasets. When the machine learning algorithms are compared with the statistical method LR, it is seen that machine learning methods show significantly higher success. This situation is similar to DPModel-2, where the same algorithms are used. The main reason behind this difference is that the LR algorithm is linear and the machine learning algorithms are nonlinear. When machine learning algorithms are compared in each other, DT algorithm is a weaker predictor than RF and GB algorithms since a single tree is not sufficient for learning due to the amount of data and the large number of variables. RF overcomes this problem by increasing the amount of trees and GB overcomes by turning weak learners into strong learners through iterations.

The prediction results of the models according to years are given in Table 7 and Table 9. According to the results, it is seen that the AUC score increases over the years. This is due to the increase in the quality and amount of data over the years. Also, the decline in success in 2018 stems from to a lack of credit data.

The results of the prediction algorithms used in both models are

Table 9
The success of DPModel-2 by years.

Year	Model	Threshold	AUC	F1	Accuracy	Precision	Recall
2011	LR	0.12	0.603	0.152	0.363	0.083	0.87
2011	DT	0.13	0.653	0.168	0.624	0.098	0.581
2011	RF	0.13	0.653	0.168	0.603	0.098	0.612
2011	GB	0.16	0.651	0.167	0.57	0.096	0.657
2012	LR	0.07	0.748	0.229	0.875	0.169	0.354
2012	DT	0.10	0.804	0.306	0.911	0.259	0.375
2012	RF	0.11	0.811	0.312	0.917	0.277	0.359
2012	GB	0.14	0.804	0.309	0.92	0.282	0.34
2013	LR	0.07	0.77	0.278	0.895	0.247	0.317
2013	DT	0.12	0.818	0.375	0.922	0.386	0.365
2013	RF	0.09	0.822	0.381	0.912	0.346	0.424
2013	GB	0.13	0.82	0.372	0.909	0.333	0.422
2014	LR	0.12	0.785	0.279	0.89	0.225	0.365
2014	DT	0.17	0.841	0.34	0.863	0.236	0.609
2014	RF	0.29	0.848	0.372	0.911	0.316	0.451
2014	GB	0.24	0.845	0.374	0.912	0.32	0.45
2015	LR	0.10	0.795	0.267	0.923	0.244	0.296
2015	DT	0.21	0.854	0.367	0.933	0.334	0.407
2015	RF	0.19	0.859	0.371	0.931	0.326	0.43
2015	GB	0.20	0.853	0.369	0.933	0.337	0.409
2016	LR	0.08	0.812	0.294	0.931	0.271	0.321
2016	DT	0.16	0.87	0.406	0.94	0.368	0.453
2016	RF	0.16	0.877	0.411	0.941	0.374	0.457
2016	GB	0.15	0.87	0.394	0.937	0.346	0.458
2017	LR	0.09	0.826	0.32	0.929	0.292	0.355
2017	DT	0.16	0.883	0.439	0.938	0.386	0.509
2017	RF	0.17	0.89	0.444	0.941	0.399	0.5
2017	GB	0.17	0.878	0.439	0.935	0.372	0.537
2018	LR	0.14	0.826	0.289	0.953	0.288	0.29
2018	DT	0.29	0.872	0.359	0.947	0.301	0.445
2018	RF	0.31	0.874	0.362	0.95	0.314	0.426
2018	GB	0.25	0.871	0.36	0.948	0.302	0.446

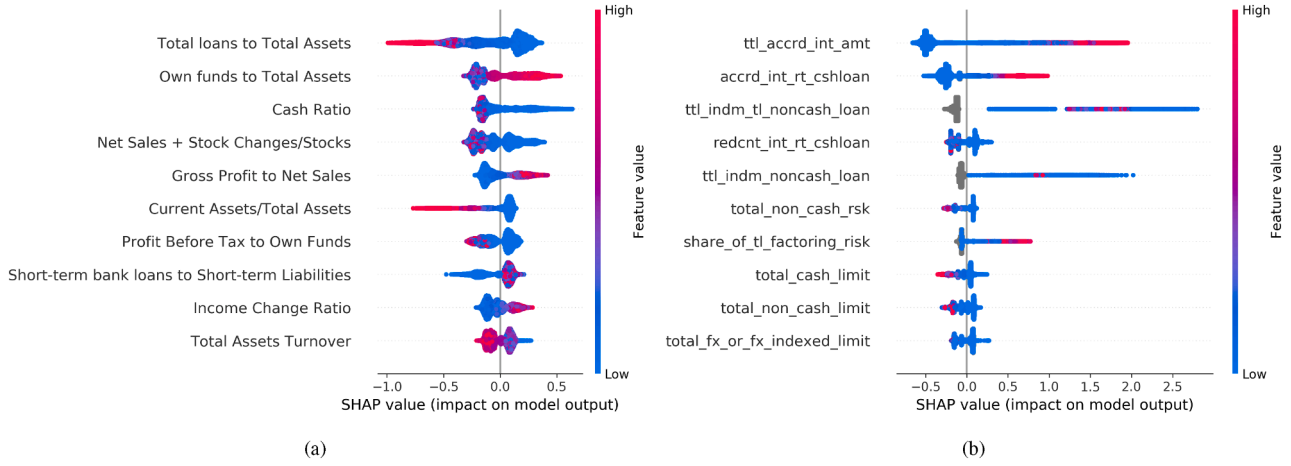

Fig. 7. SHAP value results for (a) balance sheet dataset (b) credit dataset in DPModel-1.

successful at an acceptable level. There are several reasons for this success. First, grid search is used to optimize hyperparameters of each algorithm. Second, the models are run for multiple times using k-fold cross validation and the results are averaged. And third, the threshold is determined as to maximize the $F1_{score}$ instead of using the default value of 0.5 to calculate the classification metrics of the algorithms.

Considering the results of DPModel-2, a significant increase is observed in AUC score compared to DPModel-1. This increase is due to the weighted customer score and centrality variables obtained from the graph theory. The weighted customer score obtained from the graph theory reveals that the probability of the customer in default is related to the size of the trading relationship established with that customer. In other words, the probability of a company default rate increases if it has a large number of customers within default or possible to default, otherwise this rate decreases. Similarly, centrality variables also contribute positively to the default prediction performance with some variables such as the number of companies with which the company has a trading relationship and the importance of these companies in the network.

To the best of our knowledge, this paper considers the most variables and the most data amount for default prediction. Therefore, as can be seen in the Table X, the computation costs of the models are high.

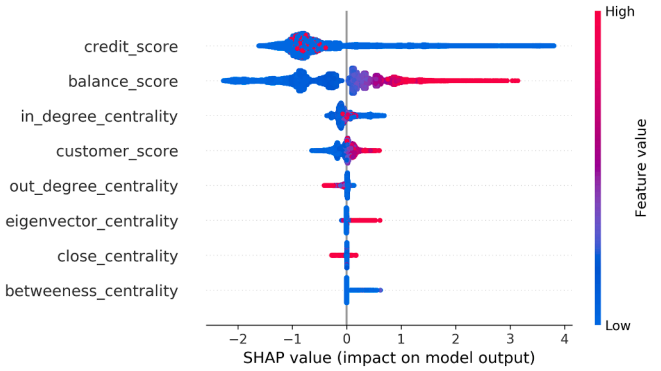

Fig. 8. SHAP value results for DPModel-2.

Table 10
Average runtimes of each model depending on the ML algorithm.

Model	DPModel-1	DPModel-2
LR	80 min	32 min
DT	83 min	51 min
RF	107 min	63 min
GB	102 min	58 min

However, we believe that this cost can be acceptable according to the sensitivity of the problem. Also, BDA platform allows us to implement our proposed models with huge amount of data within relatively reasonable runtimes.

SHAP value enables interpretability of models, which is an indispensable part of default prediction in real world applications. Thus, it is no longer necessary to prefer models with poor prediction success, such as LR, simply because it is interpretable. According to the SHAP value results, the expected variables contribute significantly to the prediction. The fact that SHAP value interprets variables locally as well as globally proves that it will provide very important contributions to real-world applications. With local interpretation, it is able to analyze the contribution of each variable to the results for each predicted company.

The experiments are conducted on the CBRT BDA platform. On this platform, different users are fed from the same resource pool. Therefore, the system resources allocated for models change in each run. For this reason, the runtime calculation varies according to the load on the system. In Table 10, average runtimes for each ML algorithm used in DPModel-1 and DPModel-2 are given. These average runtimes are calculated by running 3 times of each model.

As seen in the Table 10, DPModel-1 has a higher runtime compared to DPModel-2. The most important reason is the number of variables used during the experiments. DPModel-1 uses 61 balance sheet and 57 credit variables while DPModel-2 uses balance sheet and credit scores and centrality variables obtained from graph theory. Although runtimes are seemed to be high, they are actually reasonable when considering the amount of data, number of variables, hyperparameter optimization, and cross validation with grid search. Since Apache Spark works as lazy evaluation, these times are not calculated only by running the model. They represent the entire time from reading data to writing results.

Two different statistical tests are used for the significance of the results of machine learning. These are 5-fold cross validation and McNemar tests All machine learning results in our study are the average results obtained with 5-fold cross validation. The McNemar test is used to compare the prediction results of machine learning algorithms used in DPModel-1 and DPModel-2. According to the McNemar test, the results obtained with DPModel-2 and the results obtained from DPModel-1 differ from each other statistically at 0.01 significance level.

Our purposes of making a positive contribution to the default prediction by using graph theory, which is our basic hypothesis in the study, is supported by the increase in the success rate of DPModel-2 results. The analysis results show that in addition to balance sheet and credit datasets the invoice dataset by modeling with graph theory will also contribute to the default prediction of companies.

6.4.1. *Critical analysis*

Deep learning algorithms are shown to be innovative, widely used, and powerful algorithms for many prediction tasks in the literature. They prove their success in various financial application areas with different classification problems (Heaton, Polson, & Witte, 2017; Bao, Yue, & Rao, 2017; Hiransha, Gopalakrishnan, Menon, & Soman, 2018; Fischer & Krauss, 2018; Mai et al., 2019). In this subsection, we focus on the success of deep learning algorithms for default prediction.

Since deep learning algorithms have high computational cost, they require resourceful devices that are computationally enough to deal with high system requirements. However, due to resource limitations, this analysis is performed on an Intel Core i7-7600U CPU 2.9 GHz machine with 15.9 GB RAM. Since our system is highly limited, we select sample data from our credit dataset. More specifically, 67.694 companies are selected with a total credit risk of more than 100 thousand TRY and operating between 2009 and 2017. The default rate in these companies is about 12%. Besides, 57 credit variables of the companies are used at the annual frequency.

Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) deep learning algorithms are selected for the analysis. Python Keras library is used. The network structure and hyperparameters of the algorithms are determined manually and the same for all deep algorithms. Two hidden layers and the number of neurons in these layers are determined as 64 and 128, respectively. The selected activation function is Sigmoid and the loss function is Binary-cross entropy whereas the selected success metric is AUC. Also, the number of iterations (epoch) is set up to 100 for each model.

The data between the years 2009–2016 are used for training and the data in 2017 are used for the critical analysis test. Similar to the main study, the default rate is estimated for the next year. Table I shows the results obtained from the test dataset.

Considering the results of the analysis on the sample dataset, it is seen that deep learning algorithms are successful in predicting the default. It is observed that AUC scores of each algorithm are similar. LSTM algorithm has the highest success with a score of 0.803 AUC. Since the experimental study is carried out with a limited system, it is open to development with larger network structures and hyperparameter optimizations. However, the obtained success rates show that deep learning algorithms are promising in predicting default. We have also extended our future plans with the promising outcomes achieved by deep learning algorithms.

6.5. *Limitations*

We deal with some limitations during the construction of models, selection of algorithms, and moving of models to the big data platform. There are three main limitations in this study. The first one concerns the dataset. In this study, all companies are considered without any restrictions such as sector, scale or active assets. In this case, micro, small, medium and large companies are included to the study. As can be expected, the number of micro and small-scale companies is high. Therefore, it is more difficult to predict the financial status of these companies compared to large companies. Default prediction for micro and small-scale companies becomes a challenging issue for the following reasons: (i) They have less legal obligations, which cause low data quality, (ii) Their status is very dependent on its owner or manager. For these reasons, in many studies in the literature, especially micro-scale companies were excluded from the dataset because they reduce the default prediction success. However, this study includes all companies in the economic system even if it limits the success rate.

The second one is about ML algorithm selection. The BDA platform is a distributed architecture. Some commonly used and successful algorithms in the literature are not suitable for a distributed architecture or have not been implemented in any ML library yet. That limits us in selecting any prediction algorithm.

Lastly, our proposed models have high computational and time complexity due to the several reasons including the size of data, grid search, and k-fold cross validation. Hence, all models are limited to end before the 2-h time-out period of the BDA platform.

7. Conclusion and future work

This paper predicts company default with different statistical and machine learning methods such as LR, DT, RF, and GB by moving the whole analysis to a BDA platform. The aim of this study is to show that companies can improve their estimations by using graph theory, machine learning, and big data technologies. To this aim, two new default prediction models are proposed in a comparative way. DPModel-1 seeks the best result for each company through different methods on balance sheet and credit data, whereas DPModel-2 also considers invoice dataset obtained with graph theory in addition to DPModel-1. For each method, grid search algorithm is utilized for optimized hyperparameter selection. Lastly, SHAP value is used to make the results more interpretable. According to the experimental analysis, the DPModel-2 with the new variables obtained from graph theory ensures demanded rises in success rate of each method according to the best results of the DPModel-1. The best results in both models are obtained by RF algorithm. In addition, BDA enables data processing with higher performance in handling more variables and features.

With the use of graph theory in default prediction, the increase in estimation success is validated in this study. Since enriching the information to be obtained from graph theory allow us to make sharp predictions, it is planned to increase the number of variables in future studies. Although this study only considers the score of the first customers, considering new variables such as customers of the customers or the number of customers will strengthen the prediction capability.

Furthermore, the results obtained with deep learning algorithms in critical analysis clearly show that deep learning algorithms will make significant contributions to default prediction. Therefore, we are planning to expand analyzes on our dataset with deep learning algorithms.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Addo, P. M., Guegan, D., & Hassani, B. (2018). Credit risk analysis using machine and deep learning models. *Risks, 6*(2), 38.

Agrawal, K., & Maheshwari, Y. (2019). Efficacy of industry factors for corporate default prediction. *IIMB Management Review, 31*(1), 71–77.

Akter, S., Wamba, S. F., Gunasekaran, A., Dubey, R., & Childe, S. J. (2016). How to improve firm performance using big data analytics capability and business strategy alignment? *International Journal of Production Economics, 182*, 113–131.

Alaka, H., Oyedele, L., Owolabi, H., Akinade, O., Bilal, M., & Ajayi, S. (2018). A big data analytics approach for construction firms failure prediction models. *IEEE Transactions on Engineering Management, 66*(4), 689–698.

Ali, J., Khan, R., Ahmad, N., & Maqsood, I. (2012). Random forests and decision trees Random forests and decision trees. *International Journal of Computer Science Issues (IJCSI), 9*(5), 272.

Ali, S. S., Mubeen, M., & Hussain, A. (2018). Prediction of stock performance by using logistic regression model: evidence from Pakistan stock exchange (PSX). *Patron of the Conference, 15*.

Alpaydin, E. (2020). *Introduction to machine learning.* MIT Press.

Al Rasyid, M. U. H., Yuwono, W., Al Muharom, S., & Alasiry, A. H. (2016). Building platform application big sensor data for e-health wireless body area network. In *2016 International Electronics Symposium (IES). 2016 international electronics symposium (ies)* (pp. 409–413).

Al-Sahaf, H., Bi, Y., Chen, Q., Lensen, A., Mei, Y., Sun, Y., & Zhang, M. (2019). A survey on evolutionary machine learning. *Journal of the Royal Society of New Zealand, 49*(2), 205–228.

Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance, 23*(4), 589–609.

Ansari, A., Ahmad, I. S., Bakar, A. A., & Yaakub, M. R. (2020). A hybrid metaheuristic method in training artificial neural network for bankruptcy prediction. *IEEE Access, 8*, 176640–176650.

Aziz, A., Emanuel, D. C., & Lawson, G. H. (1988). Bankruptcy prediction-an investigation of cash flow based models [1]. *Journal of Management Studies, 25*(5), 419–437.

Baek, J. & Cho, S. (2003). Bankruptcy prediction for credit risk using an auto-associative neural network in korean firms. 2003 ieee international conference on computational intelligence for financial engineering, 2003. proceedings. (pp. 25–29).

Bandyopadhyay, A. (2006). Predicting probability of default of indian corporate bonds: logistic and z-score model approaches. *The Journal of Risk Finance*.

Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS one, 12*(7), Article e0180944.

Barboza, F., Kimura, H., & Altman, E. (2017). Machine learning models and bankruptcy prediction. *Expert Systems with Applications, 83*, 405–417.

Beaver, W. H. (1966). Financial ratios as predictors of failure. *Journal of Accounting Research*, 71–111.

Behera, G. Nain, N. 2019. Grid search optimization (gso) based future sales prediction for big mart.2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS). (172–178).

Biecek, P. (2018). Dalex: Explainers for complex predictive models in r. *The Journal of Machine Learning Research, 19*(1), 3245–3249.

Borgatti, S. P., & Everett, M. G. (2006). A graph-theoretic perspective on centrality. *Social Networks, 28*(4), 466–484.

Borthakur, D. (2008). HDFS architecture guide. *Hadoop Apache Project, 53*(2), 1–13.

Boughaci, D., & Alkhawaldeh, A. A. (2020). Appropriate machine learning techniques for credit scoring and bankruptcy prediction in banking and finance: A comparative study. *Risk and Decision AnalysisPreprint*, 1–10.

Chang, Y. C., Chang, K. H., & Wu, G. J. (2018). Application of extreme gradient boosting trees in the construction of credit risk assessment models for financial institutions. *Applied Soft Computing, 73*, 914–920.

Chen, C. P., & Zhang, C. Y. (2014). Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences, 275*, 314–347.

Chen, H., Chiang, R. H., & Storey, V. C. (2012). Business intelligence and analytics: From big data to big impact. *MIS Quarterly*, 1165–1188.

Chijoriga, M. M. (2011). Application of multiple discriminant analysis (MDA) as a credit scoring and risk assessment model. *International Journal of Emerging Markets*.

Chou, C. H., Hsieh, S. C., & Qiu, C. J. (2017). Hybrid genetic algorithm and fuzzy clustering for bankruptcy prediction. 56, 298–316.

Chung, H., & Shin, K.s. (2018). Genetic algorithm-optimized long short-term memory network for stock market prediction. *Sustainability, 10*(10), 3765.

Ciampi, F., & Gordini, N. (2013). Small enterprise default prediction modeling through artificial neural networks: An empirical analysis of Italian small enterprises. *Journal of Small Business Management, 51*(1), 23–45.

Cowden, C., Fabozzi, F. J., & Nazemi, A. (2019). Default prediction of commercial real estate properties using machine learning techniques. *The Journal of Portfolio Management, 45*(7), 55–67.

Danenas, P., & Garsva, G. (2015). Selection of support vector machines based classifiers for credit risk domain. 42(6), 3194–3204.

Deo, N. (2017). Graph theory with applications to engineering and computer science. Courier Dover Publications.

du Jardin, P. (2016). A two-stage classification technique for bankruptcy prediction. *European Journal of Operational Research, 254*(1), 236–252.

Fallahpour, S., Lakvan, E. N., & Zadeh, M. H. (2017). Using an ensemble classifier based on sequential floating forward selection for financial distress prediction problem. *Journal of Retailing and Consumer Services, 34*, 159–167.

Figini, S., Savona, R., & Vezzoli, M. (2016). Corporate default prediction model averaging: A normative linear pooling approach. *Intelligent Systems in Accounting, Finance and Management, 23*(1–2), 6–20.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research, 270*(2), 654–669.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks, 1*(3), 215–239.

Freund, Y., & Schapire, R. (1996). Experiments with a new boosting algorithm. In proceeding of the Thirteen International conference on Machine Learning: 1996; San Francisco Edited by: Saitta L. Morgan Kaufmann.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis, 38*(4), 367–378.

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management, 35*(2), 137–144.

Gonzalez, J. E., Xin, R. S., Dave, A., Crankshaw, D., Franklin, M. J., & Stoica, I. (2014). Graphx: Graph processing in a distributed dataflow framework. 11th {USENIX} symposium on operating systems design and implementation ({OSDI}) 14, 599–613.

Gordini, N. (2014). A genetic algorithm approach for SMEs bankruptcy prediction: Empirical evidence from Italy. *Expert Systems with Applications, 41*(14), 6433–6445.

Hage, P., & Harary, F. (1995). Eccentricity and centrality in networks. *Social Networks, 17*(1), 57–63.

Heaton, J. B., Polson, N. G., & Witte, J. H. (2017). Deep learning for finance: Deep portfolios. *Applied Stochastic Models in Business and Industry, 33*(1), 3–12.

Hiransha, M., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. (2018). NSE stock market prediction using deep-learning models. *Procedia Computer Science, 132*, 1351–1362.
Hosaka, T. (2019). Bankruptcy prediction using imaged financial ratios and convolutional neural networks. *Expert Systems with Applications, 117*, 287–299.
Jing, J., Yan, W., & Deng, X. (2020). A hybrid model to estimate corporate default probabilities in china based on zero-price probability model and long short-term memory. *Applied Economics Letters*, 1–8.
Khemais, Z., Nesrine, D., & Mohamed, M. (2016). Credit scoring and default risk prediction: A comparative study between discriminant analysis & logistic regression. *International Journal of Economics and Finance, 8*(4), 39.
Kim, H., Cho, H., & Ryu, D. (2020). Corporate default predictions using machine learning: Literature review. *Sustainability, 12*(16), 6325.
Kim, H. J., Jo, N. O., & Shin, K. S. (2016). Optimization of cluster-based evolutionary undersampling for the artificial neural networks in corporate bankruptcy prediction. *Expert Systems with Applications, 59*, 226–234.
Kim, H. S., & Sohn, S. Y. (2010). Support vector machines for default prediction of SMEs based on technology credit. *European Journal of Operational Research, 201*(3), 838–846.
Kim, T., & Kim, H. Y. (2019). Forecasting stock prices with a feature fusion lstm-cnn model using different representations of the same data. *PloS one, 14*(2).
Labrinidis, A., & Jagadish, H. V. (2012). Challenges and opportunities with big data. *Proceedings of the VLDB Endowment, 5*(12), 2032–2033.
Lautier, D., & Raynaud, F. (2012). *Derivative securities pricing and modelling. Derivative securities pricing and modelling*. Emerald Group Publishing Limited.
Lee, I. (2017). Big data: Dimensions, evolution, impacts, and challenges. *Business Horizons, 60*(3), 293–303.
Lim, E. P., Chen, H., & Chen, G. (2013). Business intelligence and analytics: Research directions. *ACM Transactions on Management Information Systems (TMIS), 3*(4), 1–10.
Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. Advances in neural information processing systems. (4765–4774).
Luo, C., Wu, D., & Wu, D. (2017). A deep learning approach for credit scoring using credit default swaps A deep learning approach for credit scoring using credit default swaps. *Engineering Applications of Artificial*. Intelligence65465–470.
Mai, F., Tian, S., Lee, C., & Ma, L. (2019). Deep learning models for bankruptcy prediction using textual disclosures. *European Journal of Operational Research, 274*(2), 743–758.
Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., & Liu, D. (2016). Mllib: Machine learning in apache spark. *The Journal of Machine Learning Research, 17*(1), 1235–1241.
Midi, H., Sarkar, S. K., & Rana, S. (2010). Collinearity diagnostics of binary logistic regression model. *Journal of Interdisciplinary Mathematics, 13*(3), 253–267.
Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of machine learning*. MIT Press.
Mokhtari, K. E., Higdon, B. P., & Başar, A. (2019). Interpreting financial time series with SHAP values. In *Proceedings of the 29th annual international conference on computer science and software engineering* (pp. 166–172).
Moscatelli, M., Parlapiano, F., Narizzano, S., & Viggiano, G. (2020). Corporate default forecasting with machine learning. *Expert Systems with Applications, 113567*.
Moula, F. E., Guotai, C., & Abedin, M. Z. (2017). Credit default prediction modeling: an application of support vector machine. *Risk Management, 19*(2), 158–187.
Narayanan, V. (2014). Using big-data analytics to manage data deluge and unlock real-time business insights. *The Journal of Equipment Lease Financing (Online), 32*(2), 1.
Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics, 7*(21).
Nehrebecka, N. (2018). Predicting the default risk of companies. comparison of credit scoring models: Logit vs support vector machines. *Econometrics, 22*(2), 54–73.
Nori, H., Jenkins, S., Koch, P., & Caruana, R. (2019). Interpretml: A unified framework for machine learning interpretability. arXiv preprint arXiv:1909.09223.
Ohlson, J. A. (1980). Financial ratios and the probabilistic prediction of bankruptcy. *Journal of accounting research*, 109–131.
Okamoto, K., Chen, W. Li, X.Y. 2008. Ranking of closeness centrality for large-scale social networks.International workshop on frontiers in algorithmics. (186–195).
Olson, D. L., & Wu, D. D. (2017). *Data mining models and enterprise risk management. Enterprise risk management models* (pp. 119–132). Springer.
Oshiro, T. M., Perez, P. S., & Baranauskas, J. A. (2012). How many trees in a random forest?. International workshop on machine learning and data mining in pattern recognition. (154–168).
Óskarsdóttir, M., Bravo, C., Sarraute, C., Vanthienen, J., & Baesens, B. (2019). The value of big data for credit scoring: Enhancing financial inclusion using mobile phone data and social network analytics. *Applied Soft Computing, 74*, 26–39.
Oussous, A., Benjelloun, F. Z., Lahcen, A. A., & Belfkih, S. (2018). Big Data technologies: A survey. *Journal of King Saud University-Computer and Information Sciences, 30*(4), 431–448.
Peng, W., Chen, J., & Zhou, H. (2009). An implementation of ID3-decision tree learning algorithm An implementation of id3-decision tree learning algorithm. From web. arch. usyd. edu. au/wpeng/DecisionTree2. pdf Retrieved date: May13.
Pereira, J. M., da Basto, M., & Silva, A. F. (2016). The logistic lasso and ridge regression in predicting corporate failure. *Procedia Economics and Finance, 39*, 634–641.
Rahim, A. H. A., Rashid, N. A., Nayan, A., & Ahmad, A. R. (2019). SMOTE Approach to Imbalanced Dataset in Logistic Regression Analysis. Proceedings of the Third International Conference on Computing, Mathematics and Statistics (iCMS2017). (pp. 429–433).
Raileanu, L. E., & Stoffel, K. (2004). Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence, 41*(1), 77–93.
Ren, Y., Ye, T., Huang, M., & Feng, S. (2018). Gray wolf optimization algorithm for multi-constraints second-order stochastic dominance portfolio optimization. *Algorithms, 11*(5), 72.
Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). why should i trust you? explaining the predictions of any classifier. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. (pp. 1135–1144).
Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics, 21*(3), 660–674.
Salloum, S., Dautov, R., Chen, X., Peng, P. X., & Huang, J. Z. (2016). Big data analytics on Apache Spark. *International Journal of Data Science and Analytics, 1*(3–4), 145–164.
Son, H., Hyun, C., Phan, D., & Hwang, H. J. (2019). Data analytic approach for bankruptcy prediction. 138, 112816.
Srivastava, U., & Gopalkrishnan, S. (2015). Impact of big data analytics on banking sector: Learning for indian banks. *Procedia Computer Science, 50*, 643–652.
Stockinger, K., Bundi, N., Heitz, J., & Breymann, W. (2019). Scalable architecture for Big Data financial analytics: user-defined functions vs. SQL. *Journal of Big Data, 6*(1), 46.
Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random forest: a classification and regression tool for compound classification and qsar modeling. *Journal of chemical information and computer sciences, 43*(6), 1947–1958.
Taylor, D., Myers, S. A., Clauset, A., Porter, M. A., & Mucha, P. J. (2017). Eigenvector-based centrality measures for temporal networks Eigenvector-based centrality measures for temporal networks. *Multiscale Modeling & Simulation*, 151537–574.
Thusoo, A., Sarma, J. S., Jain, N., Shao, Z., Chakka, P., Anthony, S., & Murthy, R. (2009). Hive: a warehousing solution over a map-reduce framework. *Proceedings of the VLDB Endowment, 2*(2), 1626–1629.
Tizghadam, A., & Leon-Garcia, A. (2010). Betweenness centrality and resistance distance in communication networks. *IEEE Network, 24*(6), 10–16.
Tsai, C. F., Hsu, Y. F., & Yen, D. C. (2014). A comparative study of classifier ensembles for bankruptcy prediction. *Applied Soft Computing, 24*, 977–984.
Tsai, C. W., Lai, C. F., Chao, H. C., & Vasilakos, A. V. (2015). Big data analytics: A survey. *Journal of Big data, 2*(1), 21.
Ul Hassan, E., Zainuddin, Z., & Nordin, S. (2017). A review of financial distress prediction models: Logistic regression and multivariate discriminant analysis. *Indian-Pacific Journal of Accounting and Finance, 1*(3), 13–23.
Uthayakumar, J., Metawa, N., Shankar, K., & Lakshmanaprabu, S. (2020). Financial crisis prediction model using ant colony optimization. *International Journal of Information Management, 50*, 538–556.
Wamba, S. F., Gunasekaran, A., Akter, S., Ren, S. J.f., Dubey, R., & Childe, S. J. (2017). Big data analytics and firm performance: Effects of dynamic capabilities. *Journal of Business Research, 70*, 356–365.
Wang, G., Gunasekaran, A., Ngai, E. W., & Papadopoulos, T. (2016). Big data analytics in logistics and supply chain management: Certain investigations for research and applications. *International Journal of Production Economics, 176*, 98–110.
Wang, G., Ma, J., & Yang, S. (2014). An improved boosting based on feature selection for corporate bankruptcy prediction. *Expert Systems with Applications, 41*(5), 2353–2361.
Wang, M., Chen, H., Li, H., Cai, Z., Zhao, X., Tong, C., & Xu, X. (2017). Grey wolf optimization evolving kernel extreme learning machine: Application to bankruptcy prediction. *Engineering Applications of Artificial Intelligence, 63*, 54–68.
Wang, N. (2017). Bankruptcy prediction using machine learning. *Journal of Mathematical Finance, 7*(04), 908.
Wu, X., Zhu, X., Wu, G. Q., & Ding, W. (2013). Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering, 26*(1), 97–107.
Yeh, C. C., Chi, D. J., & Lin, Y. R. (2014). Going-concern prediction using hybrid random forests and rough set approach. *Information Sciences, 254*, 98–110.
Yeh, S. H., Wang, C. J., & Tsai, M. F. (2015). Deep belief networks for predicting corporate defaults. In *2015 24th wireless and optical communication conference (wocc)* (pp. 159–163).
Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., & Dave, A. (2016). Apache spark: A unified engine for big data processing. *Communications of the ACM, 59*(11), 56–65.
Zhang, H., Fiszman, M., Shin, D., Miller, C. M., Rosemblat, G., & Rindflesch, T. C. (2011). Degree centrality for semantic abstraction summarization of therapeutic studies Degree centrality for semantic abstraction summarization of therapeutic studies. *Journal of biomedical informatics, 44*(5), 830–838.
Zhang, X., Yang, Y., & Zhou, Z. (2018). A novel credit scoring model based on optimized random forest2018 ieee 8th annual computing and communication workshop and conference (ccwc), (pp. 60–65).
Zhao, Z., Xu, S., Kang, B. H., Kabir, M. M. J., Liu, Y., & Wasinger, R. (2015). Investigation and improvement of multi-layer perceptron neural networks for credit scoring. *Expert Systems with Applications, 42*(7), 3508–3516.
Zhou, L., Lai, K. K., & Yen, J. (2014). Bankruptcy prediction using SVM models with a new approach to combine features selection and parameter optimisation. *International Journal of Systems Science, 45*(3), 241–253.
Zhu, Y., Zhou, L., Xie, C., Wang, G. J., & Nguyen, T. V. (2019). Forecasting SMEs' credit risk in supply chain finance with an enhanced hybrid ensemble machine learning approach. *International Journal of Production Economics, 211*, 22–33.