JOURNAL OF CRITICAL REVIEWS

ISSN- 2394-5125 Vol 8, Issue 1, 2021

A Machine-Learning Framework for credit risk assessment of margin lending in the capital market of Iran

Reza Tehrani[1], Babak Nadjar Araabi[2,3], Mohsen Mehrara[4], Ehsan Mohebi*[5]

Abstract

In this article we use techniques of machine-learning to build forecasting models of credit risk for margin lending in the Stock Market of Iran which consists of two markets: Tehran Stock Exchange (TSE) and Iran Fara Bourse (IFB). By combining borrower transactions, portfolios and credit scores data from August 2016 to September 2020 for a sample of a major broker's customers, we are able to create out-of-sample forecasts that substantially increase the classification rates of borrower delinquencies and defaults, with ROC-AUC score of prediction being about 0.97 for ensemble model over the period. In this work, we propose a supervised learning model for credit risk assessment which offers an opportunity for brokerage firms to develop an automated credit allocation model for their eligible customers.

Keywords

Credit risk

Margin lending

Portfolio performance

Machine learning

Classification

1. Introduction

Leverage, an investment strategy of leveraging borrowed money, especially the use of various financial instruments or borrowed capital to maximize the future return of an investment[6], is one of the most significant drivers of investor transactions and expected return in the stock market. One approach of this strategy is called margin mending, which involves the provision of borrowing backed by a portfolio of cash, bonds, units of managed securities, securities, derivatives, and other types of market-traded commodity that is lent to individual or corporate borrowers for investment financing purposes.[7] A main characteristic of margin lending is that the ability to repay funds is determined depending on the financial status of the borrower by the properties in the portfolio, their loanable value and the credit limit. Margin lending has been used in the Iranian stock market for more than eight years, with a value of about 35.7 billion Rials, which remained on the brokers' balance sheet until 2019[8]. Over the life of a margin loan, the creditor must retain a negotiated collateral coverage ratio at all times – in most terms, the market value of the portfolio would be a multiple of the loans outstanding (depending on the market performance of the portfolio assets). If the protection coverage ratio falls below the appropriate amount, a "margin call" is triggered and the creditor will be obligated either to pay off the debt or to "top off the portfolio with new funds to restore the coverage ratio" to ensure that it is sustained. Failure by the borrower to fulfill the margin call (by 'topping up' the collateral or paying down the loan) will allow the borrower to sell the assets in the portfolio and to allocate the proceeds of the selling to repayment. In a situation that a borrower defaults on a margin call, the more volatile the portfolio assets that decrease in value, the shorter the timeframes for meeting margin calls, and the sooner the lender may try to liquidate those assets that drop in value. The opportunities and risk exposures to the lender are equally outsized. For example, as a result of midterm recession in the Iranian stock market from 2013 to 2017, many brokerage firms dealt with delinquency of borrowers some of which led to default.

[1] Faculty of Management, University of Tehran, Iran

[2] Control and Intelligent Processing Center of Excellence, School of ECE, College of Engineering, University of Tehran, Tehran, Iran

[3] School of Cognitive Science, Institute for Research in Fundamental Sciences (IPM), Tehran, Iran

[4] Faculty of Economics, University of Tehran, Tehran, Iran

[5] Kish International Campus, University of Tehran, Iran *Corresponding Author: emohebis@ut.ac.ir

[6] See definition at https://www.investopedia.com/terms/l/leverage.asp by Adam Hayes, Jul 2, 2020

[7] See definition at https://www.investopedia.com/terms/m/marginal-lender.asp by Adam Hayes, Jun 25, 2019

[8] According to cumulative balance sheet of all brokerage companies in Iran published by the Association of Stock Brokers and Securities. See more details at https://www.seba.ir

ISSN- 2394-5125 Vol 8, Issue 1, 2021

In order to distribute a certain amount of credit to qualifying customers, brokerage firms contend with a vast number of judgments. It is important to focus on models and algorithms rather than human judgment based on data from borrowers, e.g. the category in credit files of borrower obtained by brokerage firms. Models are usually used to produce numerical "scores" that summarize borrower creditworthiness. In addition, brokerage firms must set up their own customized risk models on the basis of private information about borrower's portfolio. The category of private information typically consists of both "Portfolio Performance" and "Market Environment" data on past borrower behavior. In this article, we suggest a principal indicator of the credit risk of borrower that incorporates conventional credit variables, such as Loan to Value Ratio (LVR)[9] of borrower calculated in brokerage data, which greatly enhances the predictive ability of our model. Using the private dataset of one of the major brokerage firm in Iran (which we shall refer to as the 'Broker' for confidentiality purposes in this paper) from August 2016 to September 2020, we demonstrate that by analyzing patterns in borrower behavior and the amount of used credit in market condition, our model has a significant power to distinguish subtle non-linear relationships that are difficult to find in these large databases using existing brokerage procedures. We extract 97 plus credit score features from approximately 100 Gigabyte data from broker system through evaluating the credit risk of borrower (see Table 3 in appendix). To analyze the performance of borrower's portfolio, we define extra variables which are computed by high-performance computers concurrently. We demonstrate that adding financial variables to the brokerage account of a customer will lead to substantially more reliable predictions of potential credit defaults.

In computer science literature, we use a method known as "machine learning," which refers to a collection of algorithms primarily developed to solve computer-intensive pattern recognition issues in incredibly large datasets. These approaches include Logistic Regression, Linear Discriminant Analysis, k-Nearest Neighbor, Classification and Regression Tree, Naïve Bayes, and Support Vector Machines and are best suited to consumer credit risk analysis due to the broad sampling sizes and complexity of potential interactions and features within borrower transactions.[10]

We show that our ensemble machine-learning model is highly acceptable for the borrower classification problem in the stock market of Iran with ROC-AUC score of prediction being about 0.97 over the investigated period. We explain our dataset in Section 2, address the security problems that concern it, and record some straightforward albeit insightful observational patterns. Section 3 summarizes our approach to creating useful variables or vectors of functions that will act as inputs to the algorithms we use for machine learning. The machine-learning method for combining multiple predictors to produce more efficient forecast models is defined in Section 4. And finally, in Section 5, we will make a conclusion.

2. The data

In this paper, we use a specific data collection consisting of transaction level, credit score and account balance data for individual borrowers of one of Iran 's largest brokerage firms. This data is collected for a subset of the customer base of the brokerage firm for the period from August 2016 to September 2020. Integrating transaction, credit score and account balance data helps us to quantify and change borrower credit risk metrics even more regularly than the inefficient frameworks presently used by a regulator called the SOKNA[11] instruction.

Given the delicate existence of data and borrower privacy rules, before exchanging the data with us, all user identity data elements such as names and addresses were deleted by the Broker and all computations were carried out on computers physically installed at the office of the Broker and inside the automated 'firewalls' of the Broker. In Section 2.2., we study these security protocols.

2.1. Data sources

---

9 Loan to Value Ratio = value of loan/value of investments; According to the instructions for credit transactions provided by the Securities and Exchange Organization of Iran (SEO), this ratio is supposed to be less than 0.6, but it has recently been changed to 0.3 to mitigate credit risk. In this paper we consider the ratio of 0.6 during our investigation.

10 see (Foster and Stine 2004; Huang, Chen, and Wang 2007; Bellotti and Crook 2009; Bhattacharyya et al. 2011; Farquad and Bose 2012; Kruppa et al. 2013; Niklis, Doumpos, and Zopounidis 2014; Koutanaei, Sajedi, and Khanbabaei 2015; Butaru et al. 2016; Abellán and Castellano 2017; Zhang et al. 2018; Bao, Lianju, and Yue 2019; Chen, Guo, and Zhao 2020) for applications of machine learning model to credit risk

11 The word 'SOKNA' is used for a borrower whose Loan to Value Ratio (LVR) has dropped below the permissible rate specified by the Securities and Exchange Organization (SEO). In footnote 3, this ratio is defined.

ISSN- 2394-5125 Vol 8, Issue 1, 2021

In order to extract potential variables or feature vectors, all data was processed and evaluated in the SQL Server database. Fig. 1 illustrates the conceptual model for constructing this database. Market data is provided by three information services firms in the stock market whose identities are not disclosed on the grounds of secrecy. The required data is retrieved and stored in the database using the API web service.

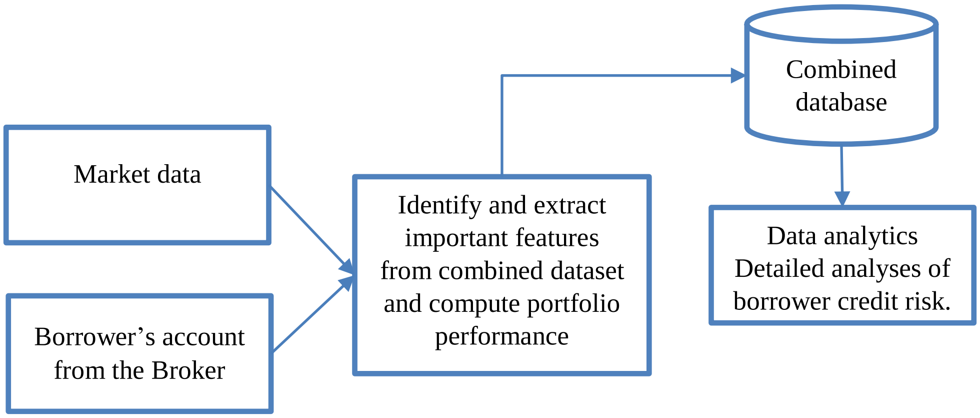

Fig. 1 Building a unified transaction and credit score database used in a borrower credit risk machine-learning platform

2.1.1. Borrower's data

The Broker presented us with data of its clients in two fields: transaction and portfolio. The transaction refers to any change of account through the market. A complete list of possible transaction includes Buy and Sell Securities, Cash Inflow and Outflow, as well as Securities Inflow and Outflow. The portfolio classification is mostly meant to capture the nature of the transaction. Examples of this category include Stock, ETF, Company bond, Treasury bond, etc. listed in TSE[12] and IFB[13]. In the raw data, we observed 7 types of instruments (Fixed Income Fund Unit, Mixed Fund Unit, OTC Options, Stock Fund Unit, Common Share, Certificate of Deposit, Bonds), which were categorized by the TSE and IFB. (See Table. 1)

Table 1 Types of instruments in stock market of Iran

	Instrument Type	**Number of Instruments**
1	Fixed Income Fund Unit	30
2	Mixed Fund Unit	264
3	IFB Options	2
4	Stock Fund Unit	31
5	Common Share	6,962
6	Certificate of Deposit	7
7	Bond	486
	Total Instrument	**7,782**

2.1.2. Market data

Using data comprising TSE stock market index (Overall Index) and IFB stock market index (IFX), we analyze the market impact on borrower credit risk during the investigated period.

In order to compute some variables like Value at Risk (VaR), Beta, Sharp Ratio, etc. we collect historical price data of all stocks traded at TSE and IFB from March 1991 to September 2020.

2.1.3. Credit Score Data

The Broker complements the portfolio of borrowers and transactions data with proprietary credit file information. In terms of statistical properties and predictive ability, the same credit score given in these results, which we will refer

12 Tehran Stock Exchange

13 Iran Fara Bourse Securities Exchange (IFB), one of the four Iranian Exchanges, operates under the supervision of the Securities and Exchange Organization (SEO), a member of IOSCO. IFB was established on November 12, 2008, to be a gate for the majority of companies to enter the capital market and enhance their corporate governance and their businesses by providing easier listing requirements.

to as the Score, is equivalent to scores which are the standard technique for assessing the rating of consumer credit. We will use this credit score as a benchmark for improving the efficiency of credit risk machine learning models.

2.2. Data security

In consideration of the importance of customer data, a range of steps have been taken to protect the integrity of those data. First the Broker de-identified all records — deleting all identifying details, such as names, addresses, phone numbers, and the age of the customers — before they were moved to the databases to which we were allowed entry. Second all computations were done on dedicated machines located at the Broker's office and all external network links for these machines were removed. Therefore, no raw customer data was moved during this project beyond the electronic firewalls of the Broker.

3. Constructing feature vectors

The objective of any machine-learning model is the identification of statistically reliable relationships between certain features of the input data and the target variable or outcome (Khandani, Kim, and Lo 2010). Classification is one of the common tasks in machine learning, which involves predicting a target variable in previously unknown data (Abdelhamid et al. 2012). The goal of classification is to predict a target variable (class) by creating a classification model based on a dataset of training and then using that model to predict the value of the test data class (Witten and Frank 2002). To construct feature vectors, we use data items such as cash inflows and outflows, daily and monthly returns, portfolio risk, account balances, etc. as seen in Table 3 in appendix. To run the machine-learning algorithm, we use the target variable as a binary outcome that indicates one when the account is counted for SOKNA after 30 days and zero if not. Computed variables such as beta, sharp ratio, etc. are calculated over a 30-day timeframe.

4. Modeling methodology

We will explain machine-learning algorithms in this chapter.[14] We use credit and transaction data of stock market borrowers, collected by the Broker, from August 2016 to September 2020 to construct delinquency and default model parameters. This challenge is well suited to being conceived as a problem in supervised learning, which is one of the machine-learning literature's most commonly applied techniques. Supervised learning is based on the training of a data sample from a data source already assigned to the appropriate classification (Sathya and Abraham 2013). Such data sample is defined typically as a vector which can consist of continuous or discrete values that lead to the "regression problem" when there is continuous output, and the "classification problem" when there is discrete output.

Using this machine-learning approach, we are developing a credit-risk prediction model that forecasts delinquency and default for individual borrower. Therefore, in this paper we are answering two questions:

1. Is Score substantially appropriate to represent borrower credit risk?
2. Which of the machine-learning algorithms offers us a great outcome in borrower credit risk delinquency classification?

In addition, model consistency must be taken into consideration when discussing these two issues, and a more thorough discussion of these and other modeling issues is presented in Section 5. The method mentioned in Section 3 can be used to define certain variables, and Fig. 2 is a description of our supervised learning system.

---

14 For an outstanding review of Machine Learning algorithms, see literature in (Vapnik 2013; Duda, Hart, and Stork 2012; Bishop 2006)

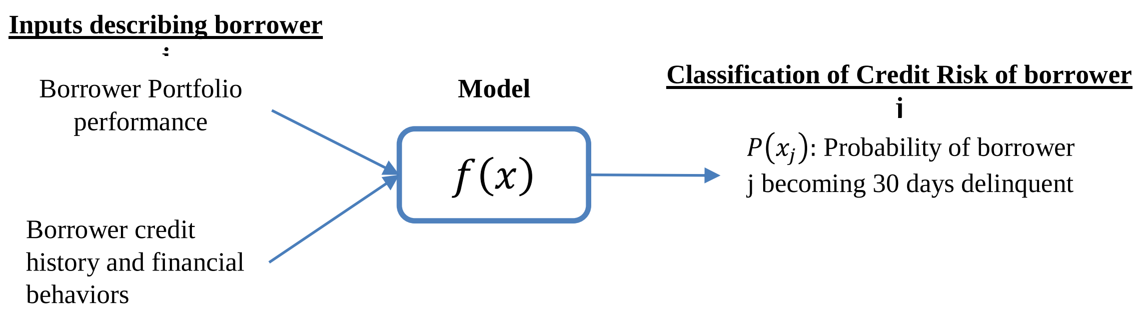

Fig. 2 Schema of a machine-learning algorithm to run a credit-risk model for borrowers in Iran's stock market

4.1. Credit Score Test

We run a back test on the Broker's estimated score in order to answer question 1 to analyze the likelihood of delinquency according to the score given to each borrower during the time. The outcome in Fig. 3 gives us an insight to create a robust classifier based on the features in the next step. The figure shows that the higher the score, the smaller the risk of delinquency. In comparison, the distribution of scores gives us an excellent insight into the borrower's credit file.

In measuring the efficiency of credit score models[15], it is standard practice to use ordinal metrics such as the Receiver Operating Characteristic (ROC) curve and its related discriminatory power statistics (Blöchlinger and Leippold 2006). In Fig. 4 the results of K-S and ROC demonstrate the accuracy and goodness of test.[16] As can be shown, the KS statistic is equal to 0.6187 and the AUC statistic is equal to 0.8853, which amount to scores that are substantially appropriate in our investigation. Fig. 5 explains the values of scores distributed over a scale from 300 to 850.

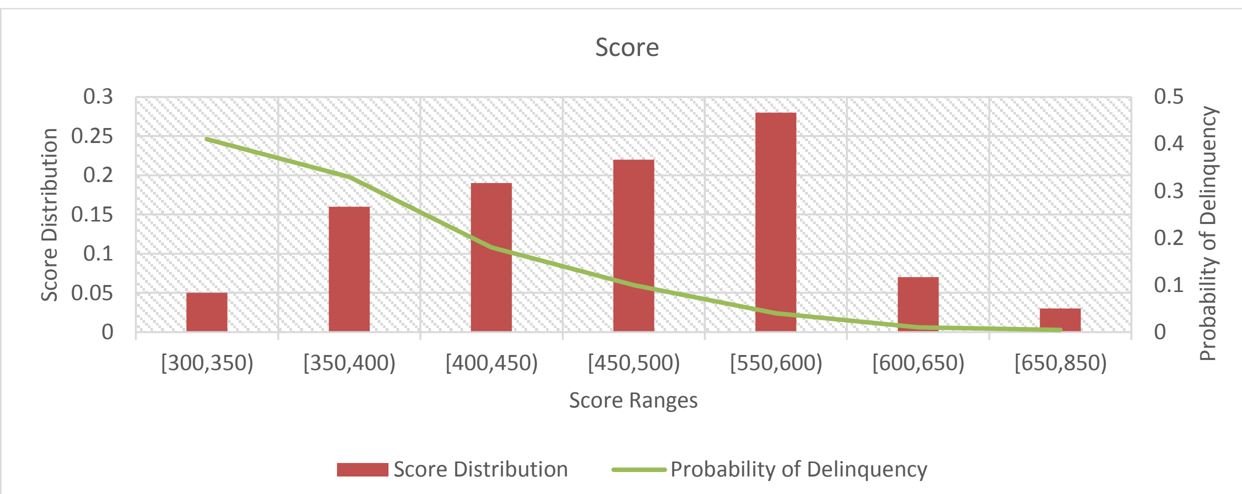

Fig. 3 Distribution chart of Scores and probability of delinquency of borrowers during the investigated period according to back test result of computed Scores given by the Broker.

[15] For a review of credit scoring models, refer to (Thomas, Crook, and Edelman 2017)

[16] See literature in (Hanley and McNeil 1982; Massey 1951)

ISSN- 2394-5125 Vol 8, Issue 1, 2021

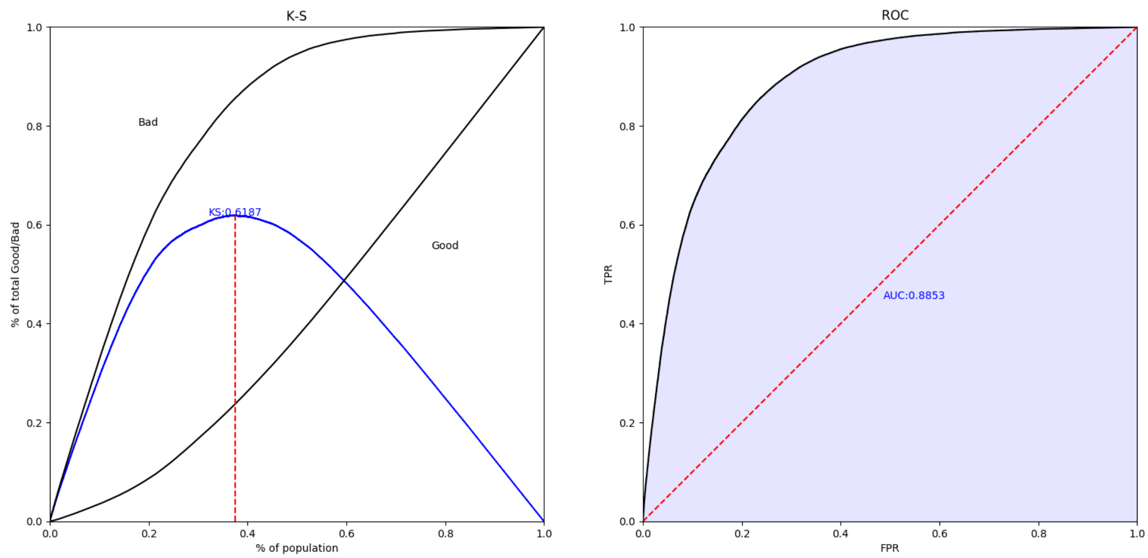

Fig. 4 Result of K-S and ROC tests on Score

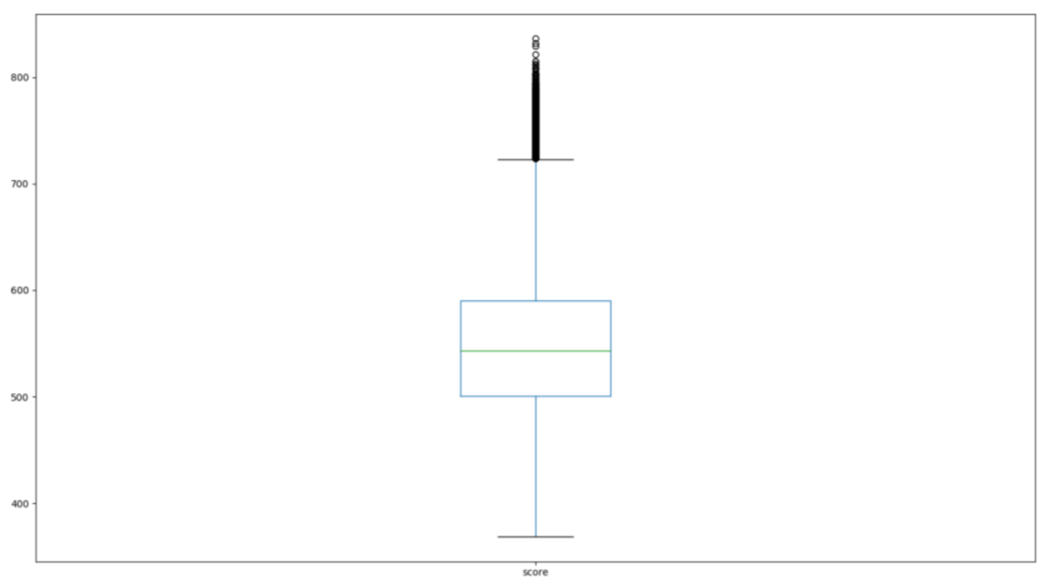

Fig. 5 Box Chart of Score

4.2. Machine-learning framework

One of the important applications for intelligent retail environment of pattern recognition (PR)[17] in financial institutions is the analysis of credit risk, which can be deemed as a method of recognition. The classification of good and bad loans is based on historical experience and the purpose of the study is to draw patterns from the dataset and classify them into different categories. Machine learning has developed from a joint analysis of pattern recognition (PR) and computational learning theory, and is especially concerned with the development of software that can learn and make decisions about new data in an autonomous manner. In fact, in supervised learning, a target function which is learned, is used to predict the values of a discrete class as accepted or reject, for example, when there are

[17] See a good review of pattern recognition application at (Paolanti and Frontoni 2020)

ISSN- 2394-5125 Vol 8, Issue 1, 2021

labeled samples of two or more classes. In this paper we classify borrowers in the stock market of Iran into good and bad customers of the Broker.

To conduct our model we use six methods in machine learning, and then in an ensemble model we show the improvement of the outcome. The training process continues until the model achieves a level of accuracy on the training dataset. These six methods of learning algorithms are:

Logistic Regression (LR) (Steenackers and Goovaerts 1989; Baesens et al. 2003; Bensic, Sarlija, and Zekic-Susac 2005), Linear Discriminant Analysis (LDA) (Eisenbeis 1978) (Hand and Henley 1997) (Ince and Aktan 2009), k-Nearest Neighbors (kNN) (Devroye et al. 1994), Classification and Decision Tree (CART) (Breiman et al. 1984) (Quinlan 2014) , Naïve Bayes (NB) (Rish 2001) and Support Vector Machine (SVM) (Cortes and Vapnik 1995) (Vapnik 2013) (Boser, Guyon, and Vapnik 1992).

4.2.1. Logistic regression

This approach is partly parametric, since the probability density functions for the groups are not modeled, rather the ratios between them are modeled. The dependent variable will only take values of 0 and 1 for two classes.

Let $y \epsilon \{0, 1\}$ be the dependent variable or the response variable, and let $x = x_{i1}, x_{i2}, \dots, x_{ip}$ be the vector variable predictor. The linear predictor η_i is given by $\beta_0 + \acute{\beta}x$ where β_0 is the constant and $\acute{\beta}$ is the vector of regression coefficients $(\beta_0, \dots, \beta_p)$ to be derived from the results. They can be explicitly translated as log-odd ratios or as $xp(\acute{\beta})$ ratios. The probabilities of a posteriori class are determined by logistic distribution:

$$P(y = 1 | x_{i1}, x_{i2}, \dots, x_{ip}) = \pi_i = \frac{exp\{\eta_i\}}{1 + exp\{\eta_i\}} \qquad \text{Eq. 1}$$

Which $\acute{\beta}$ are estimated by maximum likelihood estimation method

$$L(\beta_0, \dots, \beta_p) = \prod_{i=1}^{n} \pi_i^{y_i} (1 - \pi_i)^{1-y_i} \qquad \text{Eq. 2}$$

The estimated predicted value $\hat{\eta}_j$ and the estimated probability $\hat{\pi}_j$ for a new observation $x_{j1}, x_{j2}, \dots, x_{jp}$ are given by $\hat{\eta}_j = \hat{\beta}_0 + \hat{\acute{\beta}}x$ and $\hat{\pi}_j = \pi(x, \hat{\acute{\beta}}) = \frac{exp\{\hat{\eta}_i\}}{1+exp\{\hat{\eta}_i\}}$

These terms are often referred to as "predictions" for given characteristic vector x. Therefore, a new element is classified as 0 if $\pi_0 \leq c$ and as 1 if $\pi_0 > c$, where c is the cut-off point score. Typically, the cut-off point used could be 0.5. In fact, it has been argued that the slope of the cumulative logistic probability function is steepest in the region where $\pi_0 = 0.5$. For a prediction problem with more than two classes, multinomial logit models are used.

4.2.2. Linear Discriminant Analysis

The aim of the Linear Discriminant Analysis (LDA) is to classify the heterogeneous population as homogeneous subsets and to further the decision-making process for these subsets. We should presume that a certain number of explanatory variables are required for each applicant. The idea is to search for a linear combination of explanatory variables such as this, which separates most subsets from each other. Simplifying the method, the goal of seeking a linear function of explanatory variables in the case of two subcategories is to take the greatest margin between two means of both subsets.

$p(x|G)$ and $p(x|B)$ distributions, which are regular multivariate distributions with common variances, are widely considered. Then the sum of the Eq.3 is reduced to

$$A_G = x | \sum w_i x_i > c \qquad \text{Eq. 3}$$

Explanatory variables are x_i, and coefficients (weights) are correlated with w_i in the linear combination of explanatory variables. If one takes $s(x) = \sum w_i x_i$, then according to this ranking, it is possible to differentiate and thus reduce the problem to just one dimension.

A prevalent fallacy is the need for multivariate normality. The linear discriminant rule is ideal if the variables obey a multivariate ellipsoidal distribution (of which the normal distribution is a special case) (ignoring sampling variation). However, if discriminant analysis is assumed to yield the linear combination of the variables that maximizes a certain criterion of separation, then it is obviously widely applicable. Only if significance tests are to be conducted will the normality assumption become relevant.

The benefits of the LDA approach are that it is simple, that it can be calculated very easily and that it actually works very well. The downside is that normally distributed data is needed by LDA, but credit data is often non-normal (and categorized).

4.2.3. k-Nearest Neighbor Classification

An example of a non-parametric mathematical method serves as the k-nearest neighbor classifier. This approach assesses the parallels between the pattern found in the training set and the pattern of input. One selects the space metric of the applicants and takes the nearest k-NN of the input pattern that is nearest in any metric sense. The new candidate will be categorized in the class to which the majority of neighbors belong (in the situation where the cost of misclassification is equal). This means that this procedure calculates the likelihood of $p(G|x)$ or $p(B|x)$ by the proportion of points in the G or B class among the k-nearest neighbors in the x-point to be classified.

The choice of the metric used is a very critical move when carrying out the k-NN technique. The standard Euclidean norm given by the Eq. 4 is a widely used metric.

$$\rho_1(x,y) = \left[(x \acute{-} y)(x-y)\right]^{1/2} \qquad \text{Eq. 4}$$

Where x and y are vectors of measurements.

However, it is important to use some sufficient standardization of variables where the variables are in different units or classified as well as to choose some data-dependent variant of the Euclidean metric such as:

$$\rho_2(x,y) = \left[(x \acute{-} y)A(x-y)\right]^{1/2} \qquad \text{Eq. 5}$$

Where **A** is a matrix of $n \times n$ with n vector numbers.

Since matrix **A** can rely on *x*, the selection of **A** can depend on two types of metrics: local metrics are those where **A** relies on *x*; global metrics are those where **A** is independent of *x*. The bias/variance trade-off in the estimator is determined by the choice of the number of closest neighbors to be selected (*k*). The *k* must be less than the smallest class. The fact that *k* is finite (and hence does not carry asymptotic properties) results in a non-monotonic relationship between *k* and the proportion of each properly classified class in problems where there are two unbalanced groups. In general, that means that a greater *k* may not yield better output than a smaller *k*.

4.2.4. Classification and Regression Tree

It is easy to apply CART models to problems with high-dimensional feature spaces. Let *N* dependent variable observations as $\{y_1, \ldots, y_N\}$ and its corresponding vectors for D-dimensional features as $\{x_1, \ldots, x_N\}$. The estimated CART model parameters on the training dataset are based on the recursive collection of features from $x \in \{x_1, \ldots, x_D\}$ and $\{L_j\}$ parameters reducing the residual sum-of-squared errors. We therefore need to set a "cut-off threshold' to stop a tree expansion in order to avoid overfitting the training data. One of the most commonly used pruning measures is the Gini measure:

$$G(\tau) \equiv \sum_{k=1}^{K} P_\tau(k)\big(1 - P_\tau(k)\big) \qquad \text{Eq. 6}$$

Where τ refers to a tree's leaf node and $P_\tau(k)$ refers to the proportion of training data applied to leaf node s in class *k*. The pruning criterion for the model τ of the CART is then defined as:

$$C(T) \equiv \sum_{\tau=1}^{|T|} G(\tau) + \lambda\,|T| \qquad \text{Eq. 7}$$

Where $|T|$ represents the number of leaf nodes in model *T*, and λ refers to the regularization parameter selected by cross validation. If the pruning criterion has reached the minimum, the CART algorithm will cease to extend the tree.

4.2.5. Naïve Bayes

Enable D to be the total number of applicant characteristic categories (quantitative characteristics are categorized) and w_k to be the risk weighted coefficients associated with the categories. Finally, let $j = 0, 1$ be the markers for the poor risk class and the good risk class. NB shall express the logistic transformation $ln(p/(1-p))$ of p for the applicant as the sum of the risk-weighted coefficients of the categories in which the applicant's characteristics take the values, i.e. as follows:

$$\ln\left(\frac{p}{1-p}\right) = \sum_{k=1}^{D} x_k w_k \qquad \text{Eq. 8}$$

The dummy variables referring to the classes are x_k. The distinction between LR and NB is that while the former simply divides the characteristics into dummy variables in order to estimate the maximum probability of the weights in Eq. 8, the latter assumes class conditional independence of Z_i characteristics and the weights are described as:

$$w_i = \ln\left(\frac{\widehat{f_\imath}(z_i|1)}{\widehat{f_\imath}(z_i|0)}\right) \quad \text{Eq. 9}$$

And a formal Bayesian argument estimates the class conditional probabilities $\widehat{f_\imath}(z_i|j)$ in Eq. 9.

4.2.6. Support Vector Machine

We consider the SVM approach for a two-class classification problem. In the range of a labeled training dataset of pairs $(x_i, y_i),\ i = 1, 2 \dots, m$ when $x_i \in R^n$ and $y_i \in \{+1, -1\}$, by solving the following optimization problem, SVM considers an optimal separation hyperplane with the highest margin:

$$\min_{w,b} \frac{1}{2} w^T w \quad \text{Eq. 10}$$

Subject to: $y_i(\langle w.x_i\rangle + b) - 1 \geq 0$

It is understood that in order to solve this quadratic optimization problem, one must find the saddle point of the Lagrange function:

$$L_p(w, b, \alpha) = \frac{1}{2} w^T.w - \sum_{i=1}^{m} (\alpha_i y_i(\langle w.x_i\rangle + b) - 1) \quad \text{Eq. 11}$$

Where the α_i represents the Lagrange multipliers, hence $\alpha_i \geq 0$. Since this L_p must be minimized with respect to the primary variables *w* and *b* and maximized with respect to the non-negative dual variable α_i; the search for an optimum saddle point is necessary.

Then L_p is transformed to the dual Lagrangian $L_D(\alpha)$ by differentiating with respect to *w* and *b* and applying the Karush Kuhn-Tucker (KKT) condition for the optimum constrained function.

$$\max_{\alpha} L_D(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle x_i.x_j\rangle \quad \text{Eq. 12}$$

Subject to: $\alpha_i \geq 0\ i = 1, \dots, m\ and\ \sum_{i=1}^{m} \alpha_i\, y_i = 0$

In order to find the optimum hyperplane, the dual Lagrangian $L_D(\alpha)$ must be maximized with respect to non-negative α_i. The solution α_i for the dual optimization problem specifies the optimum hyperplane parameters $w^*$and b^*. As a consequence, the optimum hyperplane decision function $f(x) = sgn\ (\langle w^*.x\rangle + b^*)$ can be written as:

$$f(x) = sgn\ \left(\sum_{i=1}^{m} y_i \alpha_i^* \langle x_i.x\rangle + b^*\right) \quad \text{Eq. 13}$$

Normally, only a small subset of the Lagrange α_i multipliers appear to be greater than zero in a standard classification task. These vectors are, geometrically, the nearest to the optimum hyperplane. The related training vectors that have support vectors are called nonzero α_i, since they exclusively rely on the optimal decision hyperplane $f(x, \alpha^*, b^*)$.

The definitions described above can also be generalized to the non-separable case (SVM, linear generalized). In the light of these added slack variables, the problem of finding a hyperplane that offers the least number of training errors is formally expressed as follows:

$$\min_{w,\alpha,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{m} \xi_i \quad \text{Eq. 14}$$

Subject to: $y_i(\langle w.x_i\rangle + b) + \xi_i - 1 \geq 0$

$\xi_i \geq 0$

If *C* is a penalty parameter for a training error, and ξ_i is a non-negative slack vector.

SVM considers a hyperplane that sets a minimum number of training errors. This optimization model can be solved using the Lagrangian method, which is almost similar to the method used to solve the problem of optimization in the separable case. One must optimize the dual variables for Lagrangian:

$$\max_{\alpha} L_D(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle x_i.x_j\rangle \quad \text{Eq. 15}$$

Subject to: $0 \leq \alpha_i \leq C\ i = 1, \dots, m\ and\ \sum_{i=1}^{m} \alpha_i\, y_i = 0$

In order to find the best hyperplane, a dual Lagrangian $L_D(\alpha)$ must be maximized under the constraints of non-negative α_i , $\sum_{i=1}^{m} \alpha_i \, y_i = 0$ and $0 \leq \alpha_i \leq C$. The user defines the penalty parameter C as upper bond on α_i. Finally, the form of the optimum decision function of the hyperplane is the same as the Eq.13

4.2.7. Ensemble Method

Ensemble approaches are known to be the state-of-the-art approach to many machine-learning problems. The principle of an ensemble is to train multiple learners for a single task and then aggregate their results. Different methods for creating assemblies may be used in order to distinguish new instances by integrating outputs. The traditional approaches to the development of ensembles used some techniques for training such as bagging (Breiman 1996; Bauer and Kohavi 1999), boosting (Schapire 1990; Drucker et al. 1994), stacking (Wolpert 1992) and its improvement application in credit risk assessment (Papouskova and Hajek 2019; Shen et al. 2019). Ultimately, a simple majority vote is taken for forecasting. In this study we apply 4 techniques which are explained in the next sections.

4.2.7.1 Adaptive Boosting

By resampling the original data set, the boosting algorithm generates an ensemble of classifiers, which are then combined by a majority vote. However, resampling is intended to provide the most insightful training data for each consecutive classifier in the boosting process. The algorithm of Adaptive Boosting, proposed by (Freund and Schapire 1996), is the most well-known member of the boosting family. A sequence of base classifiers is generated by using successive $T_1, T_2, \ldots, T_M$ bootstrap samples obtained by weighing the training instances in M iterations, $C_1, C_2, \ldots, C_M$. Initially, Adaptive Boosting assigns equivalent weights to all training instances and changes these weights in each iteration on the basis of the misclassifications created by the resulting base classifier. Thus, in the next bootstrap sample T_i, instances misclassified by model C_{i-1} are more probable to occur. A weighted vote of the base classifiers would then obtain the final decision.

4.2.7.2 Gradient Boosting

Gradient Boosting is a machine-learning methodology that constructs a model in a step-by-step manner, as do other boosting techniques, and generalizes them by enabling the optimization of arbitrary differentiable loss feature. The principle of Gradient Boosting emerged in (Breiman 1997) finding that boosting can be represented as an optimization algorithm for an effective cost function. (Friedman 2001, 2002) subsequently developed explicit regression Gradient Boosting algorithms concurrently with (Mason et al. 1999)'s more general practical Gradient Boosting viewpoint.

4.2.7.3 Random Forest

Random Forests were suggested by (Breiman 2001), and bring an extra layer of randomness to bagging as a technique of ensemble method. In addition to using a separate bootstrap sample of the data to build each tree, Random Forests modify how the classification or regression trees are designed. In regular trees, each node is separated by the best partition between all variables. Each node is divided using the best of a subset of predictors randomly selected at that node in a Random Forest. Unlike many other classifiers, such as support vector machines, discriminatory analysis, and neural networks, this very intuitive approach is quite well performed. It can even solve the problem of over-fitting. In addition, thanks to the use of only two parameters, it is very simple to implement and is generally not very sensitive to their values. One of the parameters is the number of trees in the forest and the other is the number of variables in the random subset of each node.(Liaw and Wiener 2002)

4.2.7.4 Extra Trees

The Extra-Trees algorithm generates an ensemble of unpainted decision or regression trees. There are two main differences from other tree-based collection approaches. One of them is that it splits the nodes completely at random by gathering the cut-points. The other is that the entire learning sample is used to enlarge the trees. There are two parameters in the algorithm: the first one is K, which is the number of randomly chosen attributes for each node, and the second is $n_{\min}$, which is the minimum sampling size for separating a node. To create an ensemble model, it is used many times with the (full) initial learning sample. In order to yield the final prediction, the forecasts of the trees are aggregated by majority vote in classification problems and arithmetic average in regression problems. (Geurts, Ernst, and Wehenkel 2006)

4.3. Model inputs

The data from the three sources discussed in Section 2.1 were used as inputs to our delinquency prediction model. As discussed in Section 2.2 not all data elements can be used because of legal restrictions. We conducted extensive exploratory data analysis, similar to illustrative examples in Section 3, and finalized the set of input variables to be used in constructing our feature vectors listed in Table 3 in the appendix. For all computed items, like Beta, Sharp Ratio, Information Ratio, etc. as performance variables of borrower’s portfolio we use a 30-day rolling-window as we call it the time horizon.

ISSN- 2394-5125 Vol 8, Issue 1, 2021

The empirical results of Section 4.1 show that the Score is indeed useful in rank-ordering borrowers by their delinquency and default rates, hence it should serve as a reasonable benchmark for the machine-learning forecasts. Fig. 3 compares the result of 30-day delinquencies of the Broker's credit accounts to Score Ranges. In particular, there are a handful of accounts with relatively high scores (note that higher scores indicate higher credit quality or lower credit risk) that have high forecast delinquency and default risk according to the machine-learning model. We provide a more quantitative assessment in the next section.

4.4. Model evaluation

We now turn to a more rigorous evaluation of our models. We will focus on the model used to produce forecasts of 30-day delinquencies over subsequent 30-day windows for the period from March 1991 to September 2020. We choose this period because it includes the most severe deterioration in borrower credit quality in recent history, providing a natural laboratory for gauging the performance of machine-learning models during periods of financial dislocation. To minimize the effects of look-ahead bias, we only train the model based on delinquencies over 30-days windows that were observable at the time of the forecast. One measure of the model's success is ROC curve (receiver operating characteristic curve) which is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:

- True Positive Rate
- False Positive Rate

True Positive Rate (TPR) is a synonym for recall and is therefore defined as follows:

$$TPR = \frac{TP}{TP + FN} \qquad \text{Eq. 16}$$

False Positive Rate (FPR) is defined as follows:

$$FPR = \frac{FP}{FP + TN} \qquad \text{Eq. 17}$$

An ROC curve plots TPR vs. FPR at different classification thresholds. Lowering the classification threshold classifies more items as positive, thus increasing both False Positives and True Positives. The following figure shows a typical ROC curve.

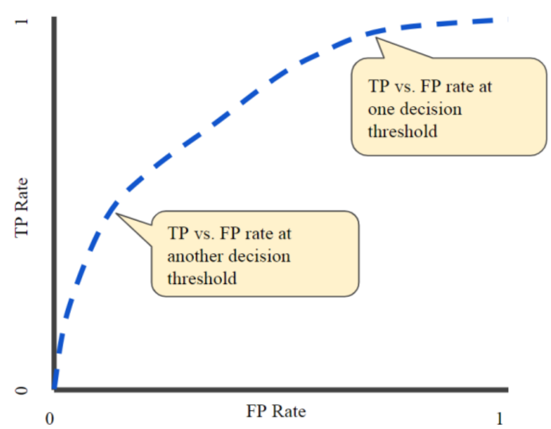

Fig. 6 TP vs. FP rate at different classification thresholds.

The other measure of the model's success is AUC which stands for "Area under the ROC Curve." It measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0, 0) to (1, 1).

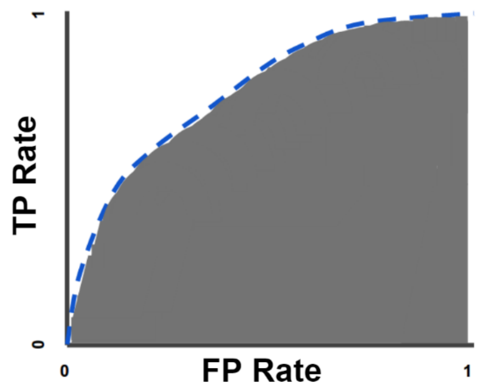

Fig. 7 AUC (Area under the ROC Curve)

AUC provides an aggregate measure of performance across all possible classification thresholds. One way of interpreting AUC is as the probability that the model ranks a random positive example more highly than a random negative example. For example, given the following examples, which are arranged from left to right in ascending order of logistic regression predictions:

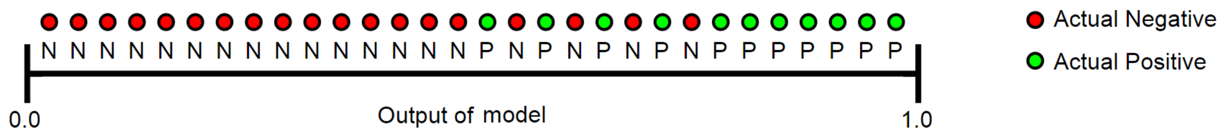

Fig. 8 Predictions ranked in ascending order of logistic regression score

AUC represents the probability that a random positive (green) example is positioned to the right of a random negative (red) example.
AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0.
Therefore we evaluate the models assuming validation size = 0.20, number of folds = 10 and Scoring = ROC-AUC. The initial result is shown in Table 2.

Classifier	ROC-AUC	Standard Deviation
LR	0.759840	0.108033
LDA	0.871612	0.051381
KNN	0.724758	0.098172
CART	0.663867	0.099141
NB	0.867469	0.059223
SVM	0.883970	0.085140

Table 2 Initial result of algorithm comparison

Fig. 9 shows the comparison of the algorithms:

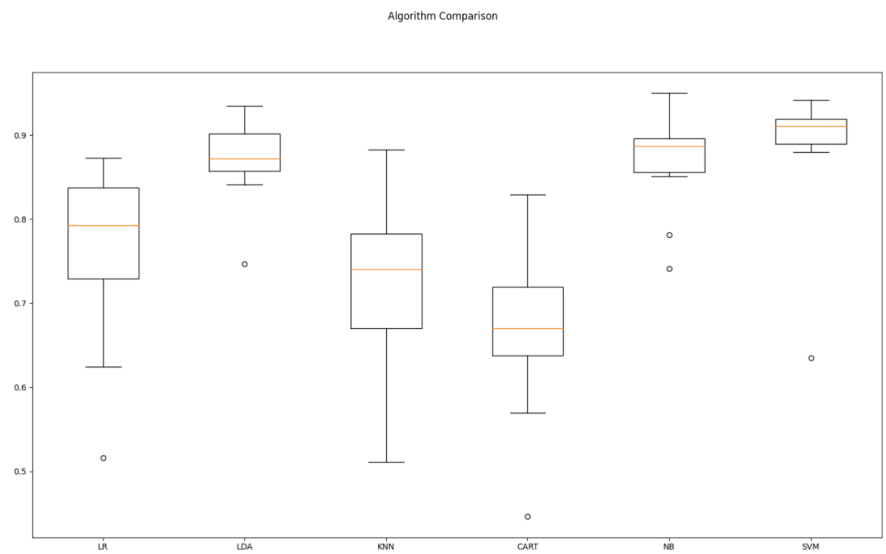

Fig. 9 Algorithm comparison of classification models

To improve classifiers we scaled the variables and ran the models. The result in Table 3 and Fig. 10 illustrates a better performance of all models.

Classifier	ROC-AUC	Standard Deviation
LR	0.878097	0.059663
LDA	0.871612	0.051381
KNN	0.731670	0.080934
CART	0.663796	0.097733
NB	0.663796	0.075974
SVM	0.843738	0.058153

Table 3 Scaled algorithm comparison

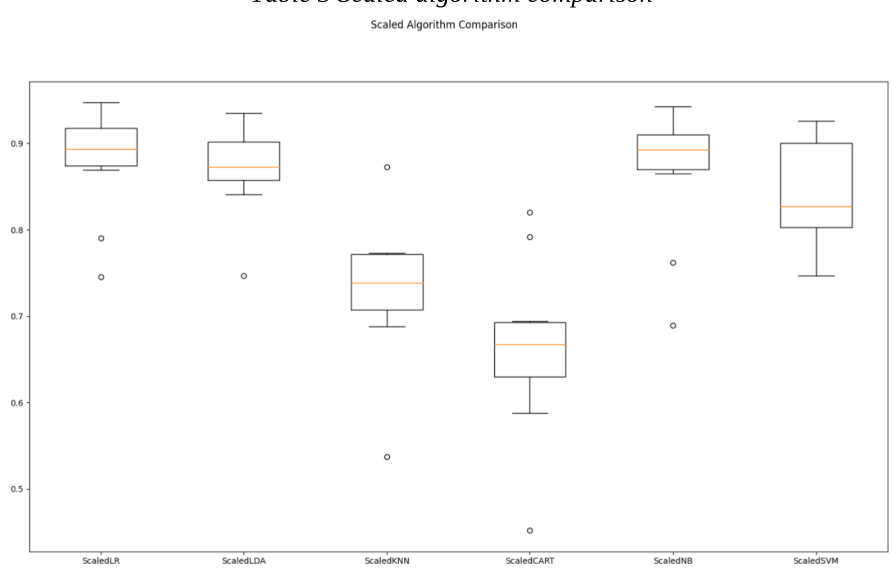

Fig. 10 Scaled algorithm comparison of classification models

In order to increase the accuracy of the KNN model, we evaluate the number of neighborhoods from 1 to 21. The results show that as the number of neighborhoods increases, the accuracy of the model has increased so that the best result in 21 neighborhoods has reached 0.814613.

Number of Neighbors	**ROC-AUC**	**Standard Deviation**
1	0.637321	0.066046
3	0.693201	0.066928
5	0.723334	0.070776
7	0.746696	0.074143
9	0.764677	0.077106
11	0.778799	0.079166
13	0.791387	0.080243
15	0.800065	0.080230
17	0.805871	0.083031
19	0.810384	0.085620
21	0.814613	0.085760

Table 4. Improving k-NN model by changing number of neighborhoods

The tuning results of the model in SVM method are remarkable as the best result is obtained with linear kernel and C = 0.1:

C	**Kernel**	**ROC-AUC**	**Standard Deviation**	**C**	**Kernel**	**ROC-AUC**	**Standard Deviation**
0.1	Linear	0.898026	0.069690	1.0	Linear	0.883995	0.078663
	Poly	0.642204	0.102439		Poly	0.690953	0.065560
	RBF	0.800785	0.069122		RBF	0.839183	0.061630
	Sigmoid	0.800228	0.144444		Sigmoid	0.798667	0.146638
0.3	Linear	0.892585	0.076443	1.3	Linear	0.882295	0.079261
	Poly	0.664381	0.088910		Poly	0.692241	0.073507
	RBF	0.806115	0.067273		RBF	0.847008	0.059002
	Sigmoid	0.799109	0.145782		Sigmoid	0.798628	0.146696
0.5	Linear	0.889621	0.077182	1.5	Linear	0.881578	0.078906
	Poly	0.676239	0.084161		Poly	0.696010	0.076806
	RBF	0.827032	0.063707		RBF	0.843409	0.058761
	Sigmoid	0.798912	0.146102		Sigmoid	0.798616	0.146710
0.7	Linear	0.887126	0.077413	1.7	Linear	0.881162	0.078937
	Poly	0.686151	0.074096		Poly	0.698338	0.077735
	RBF	0.832207	0.063103		RBF	0.843462	0.056305
	Sigmoid	0.798774	0.146559		Sigmoid	0.798577	0.146819

0.9	Linear	0.884915	0.077955	2.0	Linear	0.880557	0.078873
	Poly	0.691624	0.066350		Poly	0.701276	0.082113
	RBF	0.842249	0.058511		RBF	0.844619	0.056011
	Sigmoid	0.798682	0.146638		Sigmoid	0.798559	0.146817

Table 5 Improving SVM by changing kernel and C parameter

The following results are obtained by implementing ensemble models:

Ensemble Method	Classifier	ROC-AUC	Standard Deviation
Adaptive Boost Classifier	AB	0.918971	0.076228
Gradient Boosting Classifier	GBM	0.929975	0.057086
Random Forest Classifier	RF	0.890198	0.109868
Extra Trees Classifier	ET	0.897502	0.093581

Table 6 Result of ensemble method

ROC-AUC score of prediction lead to 0.97 which is a meaningful result for our model. According to the matrix shown in Table 7 one can see the predictive value and actual value of the model.

		Actual Value	
Predicted Value		Good	Bad
	Good	1,475,621	2,442
	Bad	42,485	25,389

Table. 7 Matrix of predictive value vs. actual value

For a given level of threshold, let True Positive (TP) be the number of instances that are actually of type 0 that were correctly classified as type 0 by the classifier, False Negative (FN) be the number of instances that are actually of type 0 but incorrectly classified as type 1, False Positive (FP) be the number of instances that are of type 1 but incorrectly classified as type 0 and, finally, True Negative (TN) be the number of instances that are of type 1 and correctly classified as type 1. Then one can define the following metrics to evaluate the accuracy of the classifier:

True positive (TP) Rate $= \frac{TP}{TP+FN}$

False positive (FP) Rate $= \frac{FP}{FP+TN}$

Precision $= \frac{TN}{TN+FN}$

Recall $= \frac{TN}{TN+FP}$

F-Measure $= \frac{(2*Recall*Precision)}{(Recall+Precision)}$

Therefore we collect the final report in Table 8:

	Precision	Recall	F-Measure	Support
Good	0.97	1.00	0.99	1,478,063
Bad	0.91	0.37	0.53	67,874
Accuracy				1,545,937
Macro average	0.94	0.69	0.76	1,545,937
Weighted average	0.97	0.97	0.97	1,545,937

Table 8 compares factors of model evaluation

5. Conclusion

In the history of Iran's stock market, it has become clear that borrower behavior has played a central role in credit delinquency, causing cascades of turmoil in the market consequences. Therefore, any prospective insights regarding borrower credit that can be gleaned from historical data has become a national priority. In this study, we develop a machine-learning model for borrower credit default and delinquency that is surprisingly accurate in forecasting credit events 30 days in advance. Although our sample from one broker is only a small percentage of the market's total borrower base, the results are promising. Our out-of-sample forecasts are highly correlated with realized delinquencies, with ROC-AUC score of prediction about 0.97 for ensemble model over the period. Moreover, from a macroprudential risk management perspective, the aggregation of machine-learning forecasts of individuals may have much to contribute to the management of enterprise and systemic risk. It is very important for chief risk officers and policymakers in Securities Exchange Organization (SEO). We find that machine-learning forecasts are considerably more adaptive, and are able to pick up the dynamics of changing credit cycles as well as the absolute levels of default rates. We believe that our results are indicative of considerably more powerful models of borrower behavior that can be developed via machine-learning techniques, and are exploring further refinements and broader datasets in ongoing research.

6. Appendix

Table 9 Input Variables

Row	Describe Variables
1	Credit balance based on the guarantee account
2	Commission fee to portfolio value ratio
3	Commission fee to balance ratio without last transactions
4	Value of credit assets
5	Consumable amount of credit assets
6	Daily purchase fee
7	Power of daily credit leverage
8	Daily return of IFB index (IFX)
9	Daily return on net asset value of portfolio assets
10	Daily Return on IFB Portfolio Section
11	Daily portfolio returns
12	Daily Return on Portfolio TSE Section
13	Daily return of stock index
14	Daily sales fee
15	Power of daily balanced credit leverage
16	Weighted return on net worth of portfolio assets
17	Number of industries in the portfolio
18	Average daily portfolio returns
19	Deviation of daily portfolio returns
20	Optimal portfolio returns based on the Markowitz model
21	Optimal portfolio returns based on the mean variance model
22	Optimal portfolio returns based on efficient frontier
23	Degree of cognitive bias in individual decision making (deviation of portfolio return from optimal return)
24	Cognitive bias deviation in individual decision making
25	Optimal portfolio standard deviation based on Markowitz model
26	Optimal portfolio standard deviation based on mean variance model
27	The standard deviation of portfolio optimization based on the efficient frontier
28	Value at risk of daily portfolio
29	Ratio of value exposed on a daily basis to the total portfolio
30	Monthly risk value
31	Ratio of monthly value at risk to portfolio
32	Number of industries with a weight of more than 5% in the portfolio
33	The final balance
34	Account balance to margin account ratio
35	Principle and interest of the credit

ISSN- 2394-5125 Vol 8, Issue 1, 2021

36	Guaranteed account value
37	Net asset value
38	Credit balance based on balance without last transactions
39	The ratio of certificates of deposit to portfolios
40	The ratio of fixed-income ETF units to the portfolio
41	Proportion of fixed income fund to portfolio
42	Ratio of futures to portfolio
43	Number of shares weighing more than 10% in the portfolio
44	The ratio of stock portfolio to portfolio
45	IFB stock to portfolio ratio
46	TSE Stock to portfolio ratio
47	Stock to portfolio ratio
48	The ratio of treasury bills to portfolios
49	Portfolio changes
50	Portfolio value
51	IFB value of portfolio
52	TSE Stock portfolio value
53	Weighted Portfolio Beta
54	IFB portfolio beta
55	TSE portfolio beta
56	Information ratio
57	IFB information ratio
58	TSE information ratio
59	Jensen's Alpha
60	IFB Jensen's Alpha
61	TSE Jensen's Alpha
62	M-Squared (M^2) Ratio
63	IFB M-Squared (M^2) Ratio
64	TSE M-Squared (M^2) Ratio
65	Weighted Sharpe Ratio of portfolio
66	The difference between the Sharp ratio of portfolio to the market
67	IFB Sharpe Ratio
68	Sharpe Ratio of market
69	Sharp ratio of IFB portfolio
70	Sharp ratio of TSE portfolio
71	TSE Sharpe Ratio
72	Treynor Ratio of portfolio
73	IFB Treynor Ratio
74	TSE Treynor Ratio
75	T^2 (Treynor Square) Measure of IFB
76	T^2 (Treynor Square) Measure of TSE
77	Initial credit
78	Total asset value
79	Credit balance based on initial credit
80	Number of days at risk
81	Purchase commission fee on the time horizon
82	Average portfolio value over the time horizon
83	IFB index returns over the time horizon
84	Return on IFB section of portfolio in the time horizon
85	Return on TSE section of portfolio in the time horizon
86	TSE overall index returns over the time horizon
87	Sales commission on the time horizon
88	IFB index standard deviation on the time horizon
89	The standard deviation of return of the IFB section of portfolio over the time horizon
90	The standard deviation of return of the TSE section of portfolio over the time horizon

91	TSE index standard deviation on the time horizon
92	Average balance without last trades on the time horizon
93	Days of inactivity
94	Working days of inactivity
95	Account balance without last trades
96	Account balance without last trades to margin account
97	account balance without loan

7. References

Abdelhamid, Neda, Aladdin Ayesh, Fadi Thabtah, Samad Ahmadi, and Wael Hadi. 2012. 'MAC: A Multiclass Associative Classification Algorithm', *Journal of Information & Knowledge Management*, 11: 1250011.

Abellán, Joaquín, and Javier G. Castellano. 2017. 'A comparative study on base classifiers in ensemble methods for credit scoring', *Expert Systems with Applications*, 73: 1-10.

Baesens, B., T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, and J. Vanthienen. 2003. 'Benchmarking state-of-the-art classification algorithms for credit scoring', *Journal of the Operational Research Society*, 54: 627-35.

Bao, Wang, Ning Lianju, and Kong Yue. 2019. 'Integration of unsupervised and supervised machine learning algorithms for credit risk assessment', *Expert Systems with Applications*, 128: 301-15.

Bauer, Eric, and Ron Kohavi. 1999. 'An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants', *Machine Learning*, 36: 105-39.

Bellotti, Tony, and Jonathan Crook. 2009. 'Support vector machines for credit scoring and discovery of significant features', *Expert Systems with Applications*, 36: 3302-08.

Bensic, Mirta, Natasa Sarlija, and Marijana Zekic-Susac. 2005. 'Modelling small-business credit scoring by using logistic regression, neural networks and decision trees', *Intelligent Systems in Accounting, Finance and Management*, 13: 133-50.

Bhattacharyya, Siddhartha, Sanjeev Jha, Kurian Tharakunnel, and J. Christopher Westland. 2011. 'Data mining for credit card fraud: A comparative study', *Decision Support Systems*, 50: 602-13.

Bishop, Christopher M. 2006. *Pattern recognition and machine learning* (springer).

Blöchlinger, Andreas, and Markus Leippold. 2006. 'Economic benefit of powerful credit scoring', *Journal of Banking & Finance*, 30: 851-73.

Boser, Bernhard E., Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. "A training algorithm for optimal margin classifiers." In *Proceedings of the fifth annual workshop on Computational learning theory*, 144–52. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery.

Breiman, Leo. 1996. 'Bagging predictors', *Machine Learning*, 24: 123-40.

———. 1997. "Arcing the edge." In.: Technical Report 486, Statistics Department, University of California at ….

———. 2001. 'Random forests', *Machine Learning*, 45: 5-32.

Breiman, Leo, J Friedman, CJ Stone, and RA Olshen. 1984. 'Classification and regression trees/Leo Breiman...[et al.]', *Belmont, Calif: Wadsworth International Group, Belmont, Calif*.

Butaru, Florentin, Qingqing Chen, Brian Clark, Sanmay Das, Andrew W. Lo, and Akhtar Siddique. 2016. 'Risk and risk management in the credit card industry', *Journal of Banking & Finance*, 72: 218-39.

Chen, Shunqin, Zhengfeng Guo, and Xinlei Zhao. 2020. 'Predicting mortgage early delinquency with machine learning methods', *European Journal of Operational Research*.

Cortes, Corinna, and Vladimir Vapnik. 1995. 'Support-vector networks', *Machine Learning*, 20: 273-97.

Devroye, Luc, Laszlo Gyorfi, Adam Krzyzak, and Gabor Lugosi. 1994. 'On the Strong Universal Consistency of Nearest Neighbor Regression Function Estimates', *The Annals of Statistics*, 22: 1371-85.

Drucker, Harris, Corinna Cortes, L. D. Jackel, Yann LeCun, and Vladimir Vapnik. 1994. 'Boosting and Other Ensemble Methods', *Neural Computation*, 6: 1289-301.

Duda, Richard O, Peter E Hart, and David G Stork. 2012. *Pattern classification* (John Wiley & Sons).

Eisenbeis, Robert A. 1978. 'Problems in applying discriminant analysis in credit scoring models', *Journal of Banking & Finance*, 2: 205-19.

Farquad, M. A. H., and Indranil Bose. 2012. 'Preprocessing unbalanced data using support vector machine', *Decision Support Systems*, 53: 226-33.

Foster, Dean P., and Robert A. Stine. 2004. 'Variable Selection in Data Mining', *Journal of the American Statistical Association*, 99: 303-13.

Freund, Yoav, and Robert E Schapire. 1996. "Experiments with a new boosting algorithm." In *icml*, 148-56. Citeseer.

Friedman, Jerome H. 2001. 'Greedy function approximation: a gradient boosting machine', *Annals of statistics*: 1189-232.

———. 2002. 'Stochastic gradient boosting', *Computational statistics & data analysis*, 38: 367-78.

Geurts, Pierre, Damien Ernst, and Louis Wehenkel. 2006. 'Extremely randomized trees', *Machine Learning*, 63: 3-42.

Hand, D. J., and W. E. Henley. 1997. 'Statistical Classification Methods in Consumer Credit Scoring: a Review', *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 160: 523-41.

Hanley, J. A., and B. J. McNeil. 1982. 'The meaning and use of the area under a receiver operating characteristic (ROC) curve', *Radiology*, 143: 29-36.

Huang, Cheng-Lung, Mu-Chen Chen, and Chieh-Jen Wang. 2007. 'Credit scoring with a data mining approach based on support vector machines', *Expert Systems with Applications*, 33: 847-56.

Ince, Huseyin, and Bora Aktan. 2009. 'A comparison of data mining techniques for credit scoring in banking: A managerial perspective', *Journal of Business Economics and Management*, 10: 233-40.

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

Koutanaei, Fatemeh Nemati, Hedieh Sajedi, and Mohammad Khanbabaei. 2015. 'A hybrid data mining model of feature selection algorithms and ensemble learning classifiers for credit scoring', *Journal of Retailing and Consumer Services*, 27: 11-23.

Kruppa, Jochen, Alexandra Schwarz, Gerhard Arminger, and Andreas Ziegler. 2013. 'Consumer credit risk: Individual probability estimates using machine learning', *Expert Systems with Applications*, 40: 5125-31.

Liaw, Andy, and Matthew Wiener. 2002. 'Classification and regression by randomForest', *R news*, 2: 18-22.

Mason, Llew, Jonathan Baxter, Peter Bartlett, and Marcus Frean. 1999. 'Boosting algorithms as gradient descent', *Advances in neural information processing systems*, 12: 512-18.

Massey, Frank J. 1951. 'The Kolmogorov-Smirnov Test for Goodness of Fit', *Journal of the American Statistical Association*, 46: 68-78.

Niklis, Dimitrios, Michael Doumpos, and Constantin Zopounidis. 2014. 'Combining market and accounting-based models for credit scoring using a classification scheme based on support vector machines', *Applied Mathematics and Computation*, 234: 69-81.

Paolanti, Marina, and Emanuele Frontoni. 2020. 'Multidisciplinary Pattern Recognition applications: A review', *Computer Science Review*, 37: 100276.

Papouskova, Monika, and Petr Hajek. 2019. 'Two-stage consumer credit risk modelling using heterogeneous ensemble learning', *Decision Support Systems*, 118: 33-45.

Quinlan, J Ross. 2014. *C4. 5: programs for machine learning* (Elsevier).

Rish, Irina. 2001. 'An Empirical Study of the Naïve Bayes Classifier', *IJCAI 2001 Work Empir Methods Artif Intell*, 3.

Sathya, R, and Annamma Abraham. 2013. 'Comparison of supervised and unsupervised learning algorithms for pattern classification', *International Journal of Advanced Research in Artificial Intelligence*, 2: 34-38.

Schapire, Robert E. 1990. 'The strength of weak learnability', *Machine Learning*, 5: 197-227.

Shen, Feng, Xingchao Zhao, Zhiyong Li, Ke Li, and Zhiyi Meng. 2019. 'A novel ensemble classification model based on neural networks and a classifier optimisation technique for imbalanced credit risk evaluation', *Physica A: Statistical Mechanics and its Applications*, 526: 121073.

Steenackers, A, and Marc Goovaerts. 1989. 'A credit scoring model for personal loans', *Insurance: Mathematics & Economics*, 8: 31-34.

Thabtah, Fadi, Suhel Hammoud, and Hussein Abdel-Jaber. 2015. 'Parallel Associative Classification Data Mining Frameworks Based MapReduce', *Parallel Processing Letters*, 25: 1550002.

Thomas, Lyn, Jonathan Crook, and David Edelman. 2017. *Credit scoring and its applications* (SIAM).

Vapnik, Vladimir. 2013. *The nature of statistical learning theory* (Springer science & business media).

Witten, Ian H., and Eibe Frank. 2002. 'Data mining: practical machine learning tools and techniques with Java implementations', *SIGMOD Rec.*, 31: 76–77.

Wolpert, David H. 1992. 'Stacked generalization', *Neural networks*, 5: 241-59.

Zhang, Tao, Wei Zhang, Wei Xu, and Haijing Hao. 2018. 'Multiple instance learning for credit risk assessment with transaction data', *Knowledge-Based Systems*, 161: 65-77.