Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Predictive Analytics of Overdraft Defaults in Large Corporates and SME's Banking: A Jordanian Case Study

Asma Mousa Braizat, Dr. George Sammour

Princess Sumaya University for Technology, Amman, Jordan.
Email: asm20218052@std.psut.edu.jo

A robust and stable banking system is critical for nation development, as it plays a central role in the economy, facilitating the acquisition and allocation of financial resources. Its stability impacts GDP growth and fosters economic growth across sectors. However, banking sector instability can lead to insolvency which means it is unable to meet its financial obligations, such as repaying deposits or honor-ing loan agreements. This paper aims to develop a classification and prediction model for borrowers at risk of falling behind on payments and advancing in overdraft accounts. It utilizes Knowledge Discovery in Datamining (KDD) to un-cover hidden patterns and trends, enabling predictions and decision-making. AdaBoost, Random Forest, and J48 models are Machine learning techniques uti-lized to predict who would default on their payment, their performance is com-pared AdaBoost, especially when using J48 decision trees, consistently exhibits high levels of accuracy, precision, recall, F-measure, and ROC Area values on both training and test datasets. Random Forest also demonstrates strong perfor-mance across a range of criteria.

Keywords: Banking system, Machine Learning, Adaboost, Random forest, J48, ROC Area, F-measure

1. Introduction

Understanding the banking sector's impact on the economy is crucial. Banks are vital for providing credit to individuals, corporations, and governments to foster growth. Their main role is to save deposits and lend money to others. Governments regulate this process to reduce risks. Lending money, especially through overdrafts, is a key revenue stream for banks. An overdraft occurs when a customer's bank account goes into a negative balance, usually because of processing a transaction that exceeds their available balance [1]. The 2007 global financial crisis, also known as the GFC, was a severe economic downturn originating in the US due to the housing market bubble. The crisis was triggered by excessive lending, subprime mortgages, and the use of complex financial instruments. The subsequent credit crisis immobilized the global financial system, causing trust erosion, economic decline, and increased unemployment. Governments and central banks implemented extraordinary policies

to stabilize markets and promote economic expansion [2]. The paper aims to create a classification and prediction model that precisely pinpoints borrowers who are most likely to fall behind on their payments and advance from stage 1 or stage 2 to stage 3 in overdraft accounts.

2. Literature Review

[3] in this paper the author explores the loan default prediction capabilities of Random Forest and XGBoost. During the phase of feature engineering, the author employs the variance threshold and variance inflation factor approaches to remove superfluous features. The selected characteristics are then entered into both the Random Forest and XGBoost models. The findings indicate that both models for predicting loan defaults achieve high levels of accuracy. To determine the most accurate model for predicting loan defaults, future research will involve comparisons of a variety of complex machine learning techniques, such as Neural Networks, KNN, MLP, and hybrid models. [4] talked about Loan lending and how it is a key financial activity for both individuals and financial institutions, with loan repayment determining the profitability of lenders. Despite the fact that lending is advantageous for both parties, it carries the inherent risk of loan default. It is critical for financial organizations to accurately anticipate loan default in order to analyze the possibility of defaults, decrease bad loan difficulties, and ultimately increase profitability. This project combines data mining techniques to extract insights and creates a loan prediction model using machine learning algorithms on the Sparks Big Data platform. Forecasting loan defaults employs six supervised machine learning classification techniques, including Logistic Regression, Decision Tree, Random Forest, Gradient Boosted Tree (GBTs), Factoring Machines (FM), and Linear Support Vector Machine (LSVM). The models are evaluated using measures such as accuracy, precision, recall, ROC curve, and F-measure, with the Decision Tree and Random Forest Models achieving the greatest accuracy of 99.62 percent. [5] provided two machine learning models meant to predict loan eligibility by examining particular borrower traits. These models aim to streamline the decision-making process for banking authorities tasked with picking qualified loan applicants from a pool. The paper provides a complete comparison of two algorithms, Random Forest and Decision Trees, both of which are applied to the same dataset. The results demonstrate that the Random Forest method (80%) accuracy outperforms the Decision Tree algorithm (73%) accuracy. [6] focused on the critical role of commercial banks in the implementation of financial policies and social investments, with an emphasis on credit risk management. Using a Genetic Algorithm Neural Network (GANN) technique, this study proposes a model for assessing the credit risk of commercial banks. This model seeks to address the shortcomings of the Backpropagation Neural Network (BPNN), such as local minima and sluggish convergence. It establishes a credit risk evaluation index system segmented by industry and mostly based on quantitative indicators, complemented by qualitative ones. In addition, the paper presents improved data preparation approaches for credit index data. Simulation findings reveal that this strategy achieves a greater level of accuracy (94.17%) than BPNN (89.46%) and the immune algorithm (90.14 %).

3. Research Methodology

In this paper, KDD is used to achieve the paper objective. KDD stands for Knowledge Discovery in Datamining. The sequential stages of data selection, preprocessing, transformation, data mining, and interpretation ensure a thorough and systematic examination, resulting in reliable and significant results.

1 Data Selection and Integration

This paper was built using financial data graciously shared by an important bank in Jordan. The dataset, spanning from the years 2019 to 2023, comprises of a total of 6046 records and 21 attributes shared for each quarter on a separate spreadsheet pulled from the banks database, the data is related to large, medium and small size companies with overdraft accounts in the bank. By examining payment patterns and financial dynamics, the paper goal is to predict who would default on their payments and also to unearth trends and potential risks of payment defaults influencing these companies

Table 1. IFRS9 – Mastersheet Metadata

Metadata	Description	
Title	IFRS 9 Master sheet	
Source	A Bank in Jordan- Database	
Data period	2019-2023	
Description	Quarterly data made available on 17 Excel spreadsheets – 6063 instances	
Attributes	1. CURR_JCB_SUB_CLASS_CODE 2. CURR_SECTOR_CODE 3. CURR_TOTAL_COLLATERAL 4. CURR_JCB_ECO_SECTOR_CLASS_CODE 5. CURR_STAGE 6. PREV_STAGE_REASON 7. CURR_STAGE_REASON 8. CURR_ACCT 9. CURR_DPD 10. CURR_ECL 11. PREV_OUTST 12. CURR_OUTST 13. CURR_EAD	14. CURR_AMOUNT_FINANCED 15. CURR_IIS 16. PREV_RATING 17. CURR_INT_RATING 18. CURR_EFFECTIVE_PD 19. CURR_LGD 20. PREV_INTEREST_RATE 21. CURR_INTEREST_RATE 22. CURR_NUM_RESTRUCTURED 23. CURR_NUM_RESTRUCTURED_CURR_PERD 24. CURR_NUM_RESCHEDULED 25. PREV_NUM_RESCHEDULED_CURR_PERD

Table 1 gives brief information about our primary dataset. Also, it lists the 25 attributes that were originally in the spreadsheets.

2 Attribute description

Among the 25 attributes within the dataset, a subset of 16 features were selected for focused analysis. Notably, 5 attributes associated with the preceding years results were deliberately excluded to mitigate potential influence on the outcome. Also 1 Attribute was related to customer account details that were masked for confidentiality reasons also removed, 1 attribute that had manual adjustment also removed, finally 2 attributes had Null values also removed. A novel attribute named 'class' was introduced later, this attribute shows if the customer defaulted or not. Further explanation of the preprocessing steps follows in the next section. Table 2 shows the attributes description.

Table 2. List of attributes

Attribute	Description
CURR_JCB_SUB_CLASS_CODE	Type of Account - Corporate / SME
CURR_SECTOR_CODE	Industry Sub - Telecom / Food / Tourism
CURR_TOTAL_COLLATERAL	Collaterals for Loans
CURR_JCB_ECO_SECTOR_CLASS_CODE	Industry main - Trade/Services/Individual
CURR_STAGE	Stage of account 1/2/3 current quarter
CLASS	the change in current stage and previous stage
CURR_DPD	number of days late for payment
CURR_ECL	expected credit loss (amount of money lost in case of default)
CURR_OUTST	outstanding balance in the account- current quarter
CURR_AMOUNT_FINANCED	total amount allowed to be financed
CURR_INT_RATING	Generated from system (+1 to 7-) - current quarter
CURR_EFFECTIVE_PD	Probability of default - calculated by the risk department
CURR_LGD	Loss given default - amount of loss after accounting for collaterals
CURR_INTEREST_RATE	Interest rate- current quarter
CURR_IIS	Interest in suspense
CURR_EAD	exposure at default
CURR_NUM_RESTRUCTURED	number of times the loan was restructured-current

3 Feature Selection

Feature selection has been proven to be effective and efficient in preparing data for various data mining and machine learning problems [7]. It decreases the number of dimensions by prioritizing the most informative qualities, hence enhancing computing efficiency. This strategy enhances the performance of the model by mitigating overfitting and prioritizing the discriminative parts of the data.

Feature selection additionally improves interpretability and transparency, rendering models more easily understandable to domain experts and stakeholders.

```
Attribute Evaluator (supervised, Class (nominal): 6 Class):
        ReliefF Ranking Filter
        Instances sampled: all
        Number of nearest neighbours (k): 10
        Equal influence nearest neighbours

Ranked attributes:
 0.37421   5 CURR_STAGE
 0.23447   2 CURR_SECTOR_CODE
 0.09057  11 CURR_AMOUNT_FINANCED
 0.0838    4 CURR_JCB_ECO_SECTOR_CLASS_CODE
 0.06474  14 CURR_EFFECTIVE_PD
 0.05693  13 CURR_INT_RATING
 0.05335  15 CURR_LGD
 0.05137   1 CURR_JCB_SUB_CLASS_CODE
 0.04314  16 CURR_INTEREST_RATE
 0.03448   3 CURR_TOTAL_COLLATERAL
 0.02623   9 CURR_OUTST
 0.01187   7 CURR_DPD
 0.00944  17 CURR_NUM_RESTRUCTURED
 0.00526  12 CURR_IIS
 0.00231   8 CURR_ECL
 0.0012   10 CURR_EAD

Selected attributes: 5,2,11,4,14,13,15,1,16,3,9,7,17,12,8,10 : 16
```

Fig. 1. Feature selection result

Figure 2 shows results after using ReliefF attribute evaluation model.

4 Data Preprocessing and Transformation

4.1 Data Preprocessing

Feature Engineering.

The data preprocessing stage started using Microsoft Excel. We started by creating a “Previous Stage” attribute as an additional column in every sheet. The VLOOKUP function was utilized to retrieve the stage that corresponded to the previous quarter for each client entry. This careful data augmentation made it possible to evaluate the client’s progress between quarters.Additionally, a new attribute called "Movement" was added, which represents the direction of change from the current to the previous stages. To accomplish this change, the algebraic difference between the current and previous quarter stages—represented as "Previous Q - Current Q"—was computed. As such, the resulting "Movement" variable included important information about clients whose stage transitions, providing a more nuanced view of changing trends in the dataset. For new clients who didn’t have previous stage, their “previous stage” was filled with their “current stage” number so to keep their movement equal zero.

In addition, a new attribute called "Class" was added to encode the default state of the clients according to their stage transitions. Using Excel's logical IF statement, the "Class" attribute was programmatically given the label "Not Default" if and only if the associated "Movement" cell had a positive result or equaled zero, meaning there was no default on a payment from quarter to quarter. On the other hand, the "Class" attribute was marked as "Default" in cases when the "Movement" field had a negative value, indicating a default on payment. The new generated attribute (Class) can be utilized for various purposes such as pattern analysis, predictive modeling, segmentation, performance metrics, visualizations, feature importance, correlation analysis, and validation and testing. It is based on the change in client stage between quarters. It offers valuable understanding of the usual progression of clients, aids in forecasting future phases, and enables the categorization of clients into distinct groups according to their patterns of stage transitions.

Data Integration.

The 17 sheets were later put all together in one excel sheet, previous years columns, masked account numbers, Null columns and manually adjusted column were removed from the final sheet. Finally, the newly added attributes (Movement, Previous Quarter) that were used to create our “Class” Attribute were removed so not to influence the data model.The resulting dataset contained 6046 records with 17 attributes all in one sheet, ready for preprocessing.

Data Cleaning.

The Dataset was generated from the bank’s system and there wasn’t any inconsistent data. The preprocessing phase was done using Knime . Missing data was only in the pre-stage attribute new clients who didn’t have previous stage, previous stage was filled with their current stage this approach ensures that the absence of a historical stage for new clients doesn't mislead the analysis by falsely indicating a change in their class.

The column “CURR_INT_RATING” consists of ratings related to the client’s companies according to their financial report results, the system extracts these ratings from another application the bank use named “Bluring”. The rating starts from 7+ (highest rank) to 1- (lowest rank), in order for the model to comprehend the ranking a transformation process was implemented on Excel using IF function to map the categorical ratings onto a numerical scale from 1 to 21. This scale inversely correlates with the Bluring INT ratings, with 1 representing the highest rank and 21 representing the lowest.

Binning was used to analyze three key attributes: "current collateral," "current outstanding," and "current amount financed." Binning segmented continuous numerical data into discrete intervals or bins, transforming them into categorical variables. This approach enhances interpretability, addresses outliers, and improves predictive performance of machine learning algorithms. It simplifies the analysis of non-linear patterns and thresholds. This strategic approach aligns with a broader goal of extracting meaningful insights from datasets while mitigating challenges.

4.2 Handling Imbalanced Data

An imbalanced dataset is a dataset that has a skewed or disproportionate distribution of instances across multiple classes, with one or more classes being distinctly underrepresented in comparison to the others. In other words, there are significantly fewer instances of one class (the minority class) than there are of another class or classes (the majority class or classes). In the financial sector, the loan default situation is always faced with the problem of unbalanced data. In this paper, the “default” class represents only 3.8% (227 instances) from the total class balance (6046 instances). This class imbalance will create a huge challenge in the prediction process and will lead to biased model performance. To solve this problem, The Synthetic Minority Oversampling Technique (SMOTE) was used. SMOTE is widely used in different over-sampling techniques methods that generate artificial positive examples along the line segments connecting any of the K-Nearest Neighbors (KNN) of positive instances Huang et al. (2020). After applying the SMOTE resampling technique, the proportion of instances belonging to the "default" class increased to 50%, which corresponds to 5819 instances out of the adjusted total of 11,638 instances.

5 Data Mining

This chapter explores the implementation of three well known data mining techniques: J48, Random Forest, and Adaboost using J48.

5.1 J48 Model

The results of J48 on both training and testing data show a promising performance across the various evaluation metrics. Both figure 3 and4 below discloses the model results using Weka application. TP rate in training data has a rate of 0.973 which means it correctly identifies default instances 97.3% of the time. On the test data the TP rate remains high at 0.971, confirming the model's ability to find defaults in previously unreported data.

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.957    0.012    0.988      0.957    0.972      0.946    0.984     0.985     Not Default
               0.988    0.043    0.959      0.988    0.973      0.946    0.984     0.971     Default
Weighted Avg.  0.973    0.027    0.973      0.973    0.973      0.946    0.984     0.978

=== Confusion Matrix ===

    a    b   <-- classified as
 3899  174 |    a = Not Default
   48 4025 |    b = Default
```

Fig. 2. J48 Training dataset – Confusion matrix results.

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.957    0.014    0.985      0.957    0.971      0.943    0.983     0.981     Not Default
               0.986    0.043    0.958      0.986    0.972      0.943    0.983     0.970     Default
Weighted Avg.  0.971    0.029    0.972      0.971    0.971      0.943    0.983     0.976

=== Confusion Matrix ===

    a    b   <-- classified as
 1671   75 |    a = Not Default
   25 1721 |    b = Default
```

Fig. 3. J48 Test dataset – Confusion matrix results.

5.2 Random Forest Model

Random Forest (RF) model confusion matrix results are in figure 5 and 6. The training data has a TP Rate of 0.98, which means that the model correctly recognizes default instances 98% of the time. Similarly, the TP Rate stays high at 0.98 on the test data, confirming the model's consistency in properly detecting defaults in previously unknown data. FP rate of 0.02 on the training data indicates that the model incorrectly classifies non-default cases as defaults only 2% of the time. Similarly, the FP Rate for test data is low at 0.02, showing a low rate of false alarms on new data.

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.965    0.005    0.994      0.965    0.979      0.960    0.996     0.997     Not Default
               0.995    0.035    0.966      0.995    0.980      0.960    0.996     0.995     Default
Weighted Avg.  0.980    0.020    0.980      0.980    0.980      0.960    0.996     0.996

=== Confusion Matrix ===

    a    b   <-- classified as
 3929  144 |    a = Not Default
   22 4051 |    b = Default
```

Fig. 4. RF Training dataset – Confusion matrix results

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.963    0.004    0.996      0.963    0.979      0.960    0.996     0.997     Not Default
               0.996    0.037    0.965      0.996    0.980      0.960    0.996     0.995     Default
Weighted Avg.  0.980    0.020    0.980      0.980    0.980      0.960    0.996     0.996

=== Confusion Matrix ===

    a    b   <-- classified as
 1682   64 |    a = Not Default
    7 1739 |    b = Default
```

Fig. 5. RF Test dataset - Confusion matrix results

The training data has a precision of 0.98, which means that when the model predicts a default, it is 98% accurate. The precision on the test data remained high at 0.98, showing that the model is capable of making accurate predictions with few false positives.The recall, for the training data is 0.98, showing that the model is effective at capturing actual defaults. Similarly, the

recall on the test data is strong, at 0.98, showing that the model can detect a large proportion of true positives. The F-measure of 0.98 on the training data demonstrates a good balance of precision and recall, indicating that the model is generally effective. The F-measure for the test data is consistent at 0.98, indicating strong performance in capturing both precision and recall on previously encountered data.

5.3 AdaBoost using J48 Model

The results for the Adaboost model using J48 are as shown in both figure 7 and 8. TP rate of 0.981, the model correctly recognizes default instances 98.1% of the time. Likewise, the TP Rate stays high at 0.979 on the test data, confirming the model's consistency in finding defaults in previously unknown data.The FP Rate of 0.019 on the training data indicates that the model misclassifies not default instances as defaults 1.9% of the time. Also, the FP Rate for test data is low at 0.021, showing a low rate of false alarms on new data.

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.970    0.009    0.991      0.970    0.981      0.962    0.994     0.995     Not Default
               0.991    0.030    0.971      0.991    0.981      0.962    0.994     0.990     Default
Weighted Avg.  0.981    0.019    0.981      0.981    0.981      0.962    0.994     0.992

=== Confusion Matrix ===

    a    b   <-- classified as
 3952  121 |    a = Not Default
   35 4038 |    b = Default
```

Fig. 6. Adaboost using J48 Training data - Confusion Matrix

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.968    0.011    0.989      0.968    0.978      0.957    0.996     0.997     Not Default
               0.989    0.032    0.969      0.989    0.979      0.957    0.996     0.993     Default
Weighted Avg.  0.979    0.021    0.979      0.979    0.979      0.957    0.996     0.995

=== Confusion Matrix ===

    a    b   <-- classified as
 1690   56 |    a = Not Default
   19 1727 |    b = Default
```

Fig. 7. Adaboost using J48 Testing - Confusion Matrix

With a precision of 0.981, the model correctly predicts a default 98.1% of the time. The precision on the test data remained high at 0.979, showing that the model can make accurate predictions with few false positives.The recall, for the training data is 0.981, showing that the model is effective at capturing actual defaults. Also, the recall on the test data is high (0.979), showing that the model can detect a large proportion of true positives.The F-measure of 0.981 on the training data demonstrates a good balance of precision and recall, indicating that the model is generally effective. The F-measure for the test data is consistent at 0.979, indicating strong performance in capturing both precision and recall on previously unknown data.

4. Knowledge Discovery and Use

4.1 Discussion

Comparing AdaBoost findings using J48, Random Forest, and J48 models reveals information about their different performance in predicting default probability for overdraft accounts

Table 3. J48, RF and Adaboost result - training and test data

	Type of data	TP Rate	FP Rate	Pecision	Recall	F-measure	ROC Area
J48	Training	0.973	0.027	0.973	0.973	0.973	0.984
	Test	0.971	0.029	0.972	0.971	0.971	0.983
RF	Training	0.98	0.02	0.98	0.98	0.98	0.996
	Test	0.98	0.02	0.98	0.98	0.98	0.996
Adaboost	Training	0.981	0.019	0.981	0.981	0.981	0.994
	Test	0.979	0.021	0.979	0.979	0.979	0.996

AdaBoost combines several weak classifiers, in this case J48 decision trees, to form a robust ensemble model. It builds on J48's merits while addressing its flaws. On both the training and test datasets, the AdaBoost model with J48 as the base classifier achieves high accuracy, precision, recall, F-measure, and ROC Area values. It obtains TP rates, FP rates, and properly categorized occurrences that exceed 97%. This model is stable and performs consistently well, demonstrating its ability to capture complicated correlations in the data and properly forecast default probabilities. Random Forest is an ensemble learning method that builds many decision trees during training and returns the mode of the classes (classification) or mean prediction (regression) of the individual trees. On both the training and test datasets, the Random Forest model performs well in terms of TP rates, precision, recall, F-measure, ROC Area, and correctly classified occurrences. Random Forest, like AdaBoost with J48, retains good accuracy and robustness when forecasting default probability.The J48 decision tree model performs well, but it may not reflect the data's intricacies as well as ensemble approaches such as AdaBoost and Random Forest. Overall, all three models perform well in forecasting default probabilities, with AdaBoost possibly slightly outperforming in terms of total accuracy. However, the decision between these models should take into account issues such as computing efficiency, interpretability, and application-specific requirement. According to the findings, both AdaBoost using J48 and Random Forest outperform in forecasting default probability for overdraft accounts, with consistently high accuracy and robustness across both training and test datasets. Between the two ensemble approaches, AdaBoost using J48 as the basis classifier may have a reasonable edge because of its capacity to adaptively alter the weights of misclassified examples, potentially increasing the model's overall performance. However, the decision between AdaBoost and Random Forest may be influenced by a variety of factors, including computational resources, model interpretability, and application specific needs. To make a better informed judgment, it is recommended to conduct additional experimentation and validation, such as evaluating the models' performance on multiple datasets.

4.2 Extra Validation

In order to further the paper's analytical framework, more data was obtained from the banking institution. This extra dataset underwent the same preparation as the previous test and training data, but without the Synthetic Minority Over-sampling Technique (SMOTE) approach. The processed data was utilized using the same three models: J48, Random Forest, and Adaboost using J48, also using the 10-fold cross validation.

The results obtained by using these three models are described below in table 4:

Table 4. Test data Vs Validation data results

	Type of data	TP Rate	FP Rate	Precision	Recall	F-measure	ROC Area
J48	Test	0.971	0.029	0.972	0.971	0.971	0.983
	validation data	0.958	0.001	0.989	0.958	0.97	0.978
RF	Test	0.98	0.02	0.98	0.98	0.98	0.996
	validation data	0.946	0.001	0.988	0.946	0.963	0.987
Adaboost	Test	0.979	0.01	0.979	0.979	0.979	0.996
	validation data	0.976	0.00	0.991	0.976	0.981	0.993

From table 4 we can derive that Adaboost using J48 has the highest precision, recall, and F1-score on the validation data indicating that it performs best in terms of recognizing genuine positives, limiting false positives, and capturing fewer false negatives. As a result, Adaboost appears to be the best fit among the three models for capturing Default instances.

References

1. Borné, R., & Smith, P. High-cost overdraft practices. Center for Responsible Lending (Ed.), The state of lending in America and its impact on US households, 135-158 (2013)
2. Edey, M. The global financial crisis and its effects. Economic Papers: A journal of applied economics and policy, 28(3), 186-195 (2009)
3. Wu, W. Machine Learning Approaches to Predict Loan Default. Intelligent Information Management, 14(05), 157–164 (2022)
4. Uwais, A., & Khaleghzadeh, H. Loan Default Prediction Using Spark Machine Learning Algorithms (2021)
5. Madaan, M., Kumar, A., Keshri, C., Jain, R., & Nagrath, P. Loan default prediction using decision trees and random forest: A comparative study. IOP Conference Series: Materials Science and Engineering, 1022(1) (2021).
6. Bai, Y., & Zha, D. Commercial Bank Credit Grading Model Using Genetic Optimization Neural Network and Cluster Analysis. Computational Intelligence and Neuroscience, (2022)
7. Kumar, V., & Minz, S. Feature selection. SmartCR, 4(3), 211-229 (2014).