IOSR Journal of Environmental Science, Toxicology and Food Technology (IOSR-JESTFT)
e-ISSN: 2319-2402, p- ISSN: 2319-2399.Volume 19, Issue 9 Ser. 1 (September 2025), PP 01-05
www.iosrjournals.org

Credit Card Approval System

Mallesha P
Department Of Computer Science, Bengaluru North University, Kolar

Abstract:
The correct assessment for credit card approval is very important for banks and organisations who lend a credit card to the people. The recent years have seen a huge growth in credit cards and loans. The exact judgement of person to be approved for credit cards allows the organisations to minimize losses and the same time make suitable credit arrangements as per requirement. Due to the huge growth in the number of applicants, there is a need for a more sophisticated method to automate the process and speed it up.
Credit card approval can be beneficial for organisations that lend credit cards, and due to increase in a huge number of the applicant, there is need to automate the task and classify the applicants into if they are eligible for a credit card or not. This helps to avoid organisation losses by avoiding potential defaulters. Here we are not just looking into bank balance but into there personal attributes like gender, married, age, Occupation etc. We account for these personal attributes to evaluate if the given applicant is a good customer. This can also help cut down the weeks-long process into a few days. This gives benefit by cutting down costs on credit analysis and faster credit decisions.
Keywords: *Logistic Regression, Hyper Parameters and Machine Learning*

Date of Submission: 22-08-2025 Date of Acceptance: 02-09-2025

I. Introduction

Approving for credit, e.g. payroll services and credit cards is an essential part of a developed economy. In the present interconnected world, even in developing countries like India, the use of, credit cards are no more a dream. However, for moneylenders, credit approval is still a problem as it is difficult to predict which customers represent an acceptable credit risk and should be granted credit. This is specifically valid in developing countries, as the established guidelines and models from developed countries may not be applicable. There is thus a need to research productive ways for automatic credit approval that can assist bankers in assessing consumer credit. This paper inspects the credit application data taken from the UCI machine learning repository. Various pre-processing techniques like exploratory data analysis, data mining, and transformations like handling missing values, continuous values, and categorical values are computed. Data visualization techniques are also adopted to understand the data. A few Machine Learning models including Logistic Regression, Sequential Neural Network, Random Forest are generated and implemented on the data along with hyper parameter tuning using Grid Search CV.

The background of this study involves data collection, data cleaning, data analysis, data visualization, and implementing some classifiers in Python. The objective of this paper is to find the appropriate classifier to automatically predict the approval of Credit Cards based on the attributes of the Credit Card application. The study also shows that each classifier outperforms in one or the other metric. The main contribution of this paper is an intelligent approach to predict Credit Card approval using efficient Machine Learning models in which Grid Search CV based hyper parameters optimization is implemented to optimize certain parameters in order to increase the performance of each model. The performance of each model is evaluated based on several metrics.

The organization of the thesis is structured as follows. Section II consists of related works on prediction of Credit Card approval. In Section III, information related to the experimental setup, data and its processing is provided. A brief explanation of the models used for the comparison can be seen in Section IV. Experimental analysis of Confusion Matrix is presented in Section V. Section VI shows the comparison of the classifiers on the basis of certain parameters. Section VII gives a brief information on ROC-AUC

II. Methodology

Machine learning model can predict an individual's application for a credit card will be accepted or not. To manipulate data, if there are any missing entries in the dataset.

In this Credit Card Approval Dataset from UCI Machine Learning Repository. The structure of our project will be as follows —

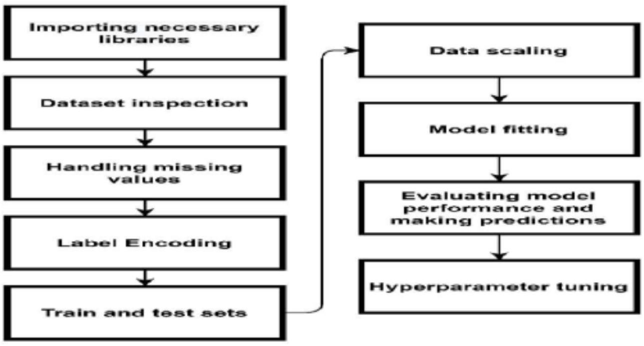

Figure 1: Block diagram

Step 1: Importing the pandas packages and Loading the dataset
Step 2: Importing NumPy package and manipulating the dataset
Step 3: Imputing the missing values with mean
Step 4: Imputing the missing values with the most frequent value in that column
Step 5: Converting the non-numeric values into numeric values
Step 6: Splitting the data into training set (70%) and test set (30%)
Step 7: Scaling the feature values to a given range
Step 8: Importing Logistic Regression classification model from sklearn package
Step 9: Predicting the accuracy of model on the test set
Step 10: Applying Hyper-parameters to make the model perform better
Step 11: Best score after applying hyper-parameters

Classifier
Logistic Regression:

Logistic Regression is derived from the field of Statistics and is a technique under Machine Learning which mainly focuses on the problems with two class values.
To predict an output value(y), the input values(x) are combined linearly using coefficient values or weights.

$$y = \frac{e^{(\beta_0+\beta_1 x)}}{1 + e^{(\beta_0+\beta_1 x)}}$$

Where, y is the predicted output, β_0 is the intercept term, β_1 is the coefficient for the single input value x. Logistic Regression uses the logistic function to transform the predictions to either 0 or 1.

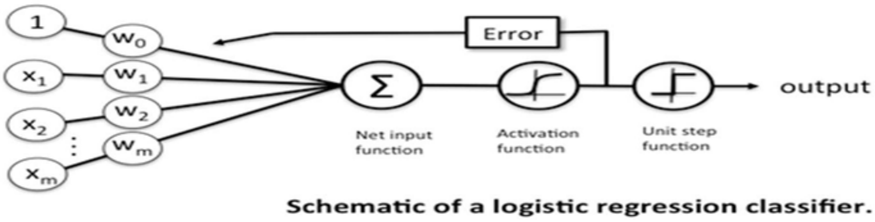

Figure 2 : Schematic of a Logistic Regression Classifier

Implementation

In this project, we'll be using Credit Card Approval Dataset from UCI Machine Learning Repository. The structure of our project will be as follows —

- We'll start by loading and viewing the dataset.
- To manipulate data, if there are any missing entries in the dataset.
- To perform exploratory data analysis (EDA) on our dataset.
- To pre-process data before applying machine learning model to the dataset.
- To apply machine learning model that can predict if an individual's application for a credit card will be accepted or not.

Credit Card Applications and the problems associated with it Nowadays, banks receive a lot of applications for issuance of credit cards. Many of them rejected for many reasons, like high-loan balances, low-income levels, or too many inquiries on an individual's credit report. Manually analyzing these applications is error-prone and a time-consuming process. Luckily, this task can be automated with the power of machine learning and pretty much every bank does so nowadays. In this project, we will be build an automatic credit card approval predictor using machine learning techniques, just like the real banks do.

The first step in any study is to get the dataset and codebook. quick analysis of the codebook gives information about the values in the dataset that have been converted to meaningless symbols to keep the data confidential.

Dataframe: 690 Observations (0-689) of 16 variables		
Gender	chr	"b", "a", "a", "b", ...
Age	chr	"30.83", "58.67", "24.50", "27.83", ...
Debt	num	0.000, 4.460, 0.500, 1.540, ...
Married	chr	"u", "u", "u", "u", ...
BankCustomer	chr	"g", "g", "g", "g", ...
EducationLevel	chr	"w", "q", "q", "w", ...
Ethnicity	chr	"v", "h", "h", "v", ...
YearsEmployed	num	1.25, 3.04, 1.50, 3.75, ...
PriorDefault	chr	"t", "t", "t", "t", ...
Employed	chr	"t", "t", "f", "t", ...
CreditScore	num	1, 6, 0, 5, ...
DriversLicense	chr	"f", "f", "f", "t", ...
Citizen	chr	"g", "g", "g", "g", ...
ZipCode	chr	"00202", "00043", "00280", "00100", ..
Income	num	0, 560, 824, 3, ...
Approved	chr	"+", "+", "+", "+", ...

Table 1: Codeset Databook

The collected data is in raw form which needs to be transformed before it is fed into the machine. We can see that the resulting values approved are '+' or '-' each for credit granted or not respectively. These character symbols are meaningless. Converting the '+' to a '1' and the '-' to a '0' will help with building Machine Learning models later in the analysis. By inspecting the above data, we can see that there are missing values which can be filled in various ways.

Missing values for numeric data is filled using the mean imputation method wherein the mean of the respective column is entered in the place of missing value. For non-numeric data, we fill in the most frequent values present in their respective columns. Once the cleaning of data is performed, label encoding comes into the picture where the data is transformed into numeric values. This is done using Label Encoder() which is imported from sklearn. Preprocessing. It is observed that Driver's license and zip code are not essential features to consider in training the model, hence they are dropped.

After dropping the irrelevant features and converting the data into machine language, the data is split into train and test sets by importing train_test_split from sklearn.model_selection library. As our data consists of feature variables X and y where X consists of feature columns (input variables) and y consists of the target column (output variable) with different ranges and hence, it is important to normalize the data first. The objective of normalizing the data is to change the numeric columns to a common scale without disturbing the range difference. Further, the X_train and X_test is scaled to the feature range from 0 to 1 using the MinMaxScaler [15]. This process is known as Normalization.

Further, some of the classifiers like Logistic Regression, Random Forest, Gradient Boost, XGBoost, Decision Tree, and Support Vector Machine are imported from sklearn with a random state of 24 and keras is used to import Sequential Neural Network. Once the classifiers are imported rescaledX_train and y_train are fit to the models. After model fitting, predict method with parameter rescaledX_test is used to predict the values. In order to obtain best accuracy, hyperparameter optimization is performed through GridSearchCV which is imported from the sklearn library.

Parameters which are not directly learnt within estimators are nothing but hyper parameters. It works by exhaustive search through a particular subset of hyper parameters. Candidates are exhaustively generated from a grid of parameter values specified within the param_grid argument for the grid search method. Each classifier uses different parameters as mentioned in the below table.

Best score after applying hyper-parameters

Classifiers	Hyperparameters
Logistic Regression	tol, max_iter, solvers, penalty, c_values
Random Forest	max_depth, min_samples_leaf, min_samples_split, n_estimators
Decision Tree	min_samples_split, max_depth
Neural Network	optimizer
Support Vector	C, gamma, kernel
Gradient Boost	min_samples_split, max_depth
XGBoost	max_depth, n_estimators, learning_rate

Table 2: Hyper parameters

III. Conclusion

In this study, the dataset from the UCI Machine Learning Repository is imbalanced and therefore, considering accuracy as a parameter for comparison is not recommended. Though the accuracy of Logistic regression (85.2%) is higher, the classifier cannot be taken as the best classifier since accuracy is not the parameter to be relied on. In place of accuracy, Precision and Recall could have been considered since these two play a vital role for the evaluation of the model. Unfortunately, it is impossible to maximize both these metrics at an equivalent time and this is where F1 Score comes into the picture to balance both Precision and Recall.

Finally, the analysis concludes that logistic regression with hyper parameters classifier is the best suited model for predicting Credit Card approvals based on F1 Score (0.868) with AUC value (0.865). Future scope of this study can include the classifiers to use TensorFlow as the library to increase accuracy, precision, and the other parameters. This study uses only two supervised Machine Learning models which can be increased and some models.

References

[1] K. Chaudhary, J. Yadav, And B. Mallick, “A Review Of Fraud Detection Techniques: Credit Card,” International Journal Of Computer Applications, Vol. 45, No. 1, Pp. 39-44, 2012.

[2] C. Luo And D. Wu, “A Deep Learning Approach For Credit Scoring Using Credit Default Swaps,” Engineering Applications Of Artificial Intelligence, Vol. 65: P. 465-470, 2017.

[3] A. Pumsirirat And L. Yan, “Credit Card Fraud Detection Using Deep Learning Based On Auto-Encoder And Restricted Boltzmann Machine,” International Journal Of Advanced Computer Science And Applications, Vol. 9, No. 1, Pp. 18-25, 2018.

[4] T. B. Trafalis And H. Ince, “Support Vector Machine For Regression And Applications To Financial Forecasting,” In Proc. The Ieee-Inns-Enns International Joint Conference On Neural Networks, 2000.

[5] G. Rushin Et Al., “Horse Race Analysis In Credit Card Fraud—Deep Learning, Logistic Regression, And Gradient Boosted Tree,” In Proc. 2017 Systems And Information Engineering Dessign Symposium (Sieds), 2017.

[6] A. Niimi, “Deep Learning For Credit Card Data Analysis,” In Proc. 2015 World Congress On Internet Security (Worldcis), 2015.

[7] S. Mittal And S. Tyagi, “Performance Evaluation Of Machine Learning Algorithms For Credit Card Fraud Detection,” In Proc. 2019 9th International Conference On Cloud Computing, Data Science & Engineering (Confluence), 2019.

[8] B. K. Wong And Y. Selvi, “Neural Network Applications In Finance: A Review And Analysis Of Literature (1990–1996),” Information & Management, Vol. 34, No. 3, Pp. 129-139, 1998.

[9] A. Vellido, P. J. Lisboa, And J. Vaughan, “Neural Networks In Business: A Survey Of Applications (1992– 1998),” Expert Systems With Applications, 1999, Vol. 17, No. 1, Pp. 51-70.

[10] C.-S. Ong, J.-J. Huang, And G.-H. Tzeng, “Building Credit Scoring Models Using Genetic Programming,” Expert Systems With Applications, Vol. 29, No. 1, Pp. 41-47, 2005.

[11] Z. Huang Et Al., “Credit Rating Analysis With Support Vector Machines And Neural Networks: A Market Comparative Study,” Decision Support Systems, Vol. 37, No. 4, Pp. 543-558, 2004.

[12] M. Saberi Et Al., "A Granular Computing-Based Approach To Credit Scoring Modeling," Neurocomputing, 2013, Vol. 122, Pp. 100-115.
[13] R. Socher Et Al., "Recursive Deep Models For Semantic Compositionality Over A Sentiment Treebank," In Proc. The 2013 Conference On Empirical Methods In Natural Language Processing, 2013.