LTU Journal of Advanced Computing & Engineering (LTU-JACE)

ISSN: 3102-0801 (Print)
Volume 1. Issue 1

LTU-JACE 2026 VOL.1 (1):

Performance Analysis of Loan Classification for Commercial Banks in Neural Network

Narendra Bahadur Shahi[1*], Pratik Chand[2], Prakash Gupta[2]

[1]Institute of Engineering and IT, Lumbini Technological University, Nepalgunj, Nepal

[2]Institute of Engineering and IT, Lumbini Technological University, Nepalgunj, Nepal

*Corresponding Email: narendra.shahi@ltu.edu.

Received: 7 December 2025 / Accepted: 1 February 2026 / Published: 1 March 2026

Abstract

With the rapid growth in banking services, there has been a tremendous increase in the number of individuals and businesses applying for loans. It is therefore getting tougher and tougher for banks to make correct and consistent decisions regarding loan approval. In this regard, Neural Networks (NN) can play an important role in financial institutions for such tasks of loan classification and making decisions about loan sanctions. This study develops more accurate Multilayer perceptron (MLP) as an enabling tool to support loan decisions in commercial banks analyzing different features of loan applicant. The dataset consists of different representative cases of loan applications that were considered or rejected based on the guidelines of banks, to train and validate the neural network model. The proposed study shows the effectiveness of the neural networks under balanced datasets which can play an important role to understand the impact of quality of dataset as well. It illustrates the ability of neural network model to predict the creditworthiness of an application accurately and precisely preventing the bank and its officials from making erroneous decisions with regard to loan approvals. The proposal aims to shed light on the exploration of the available datasets, selection of the appropriate neural network and using them for making correct and consistent loan decisions. The main goal is to create an accurate deep neural network that will take into consideration all independent variables and based on that will predict if the applicant is going to get loan approval or not. Also working of the proposed model will also be compared with other classifiers such as KNN (K Nearest Neighbor) Classifier and SVM (Support Vector Machine) classifier in terms of accuracy, sensitivity and F_1-scores.

Keywords: *Loan Classification, Bank dataset, Deep Neural Network, Accuracy, F1-score, balanced and unbalanced data set.*

1. Introduction

The banking sector is one of the most well-established industries globally. However, despite its long history and structured systems, it continues to face significant challenges related to security threats, risk management complexities, and operational inefficiencies, all of which can negatively affect the overall performance and stability of banks. Recently, Artificial Intelligence (AI) has been proposed as a promising solution to address these challenges. Additionally, banks today operate in a highly competitive environment, where survival and growth depend on adopting the latest technologies and ensuring high accuracy. Therefore, banking systems must make precise decisions to minimize cases of loan defaults. Over time, banks have grown increasingly dependent on information systems and Artificial Intelligence to manage banking transactions efficiently and to offer personalized services and products to their customers. AI provides significant benefits to core banking by addressing challenges that threaten the sector's sustainability. It improves credit risk assessment, detects fraud, reduces cyber-attacks, and lowers operational costs. Additionally, AI can predict financial crises, bankruptcies, and exchange rate fluctuations, enhancing overall profitability. One of the key areas of banks is making loan related decisions. Conventionally, loan officers have been involved in making such loan related decisions however things are gradually changing and even decision-making jobs are being done by artificially intelligent technologies. The switch is needed also because studies have shown that the decisions made by loan officers are subjective, inefficient, and inconsistent. This is the reason why this study attempts to come with multilayer Perceptron that can play an important role in making loan decisions precisely.

2. Methodology

2.1 Model Development

A proper model is built, which consists of a series of steps to solve the problem. The general architecture of a framework is shown below. It consists of a data acquisition process for extracting the required dataset, data preprocessing for cleaning the data, finding missing values and outliers, finding and analyzing the appropriate attributes, oversampling the dataset to achieve balanced dataset, and then developing a neural network model for training and a classifier for making final decision of accepting or rejecting the loan.

2.2 Dataset Collection

The dataset to be used here is publicly available dataset, which is then pre-processed and then used for evaluation of the neural network. Data has been collected from open source data (kaggle.com)and these are related to universal bank, it consists of a total of 5000 records and 14 attributes. Here 4520 are rejected loans and 480 are accepted loans. Attribute selection includes numeric and integer attributes along with some factor attributes relevant to the research problem. Dataset consists of a combination of variables as follows:

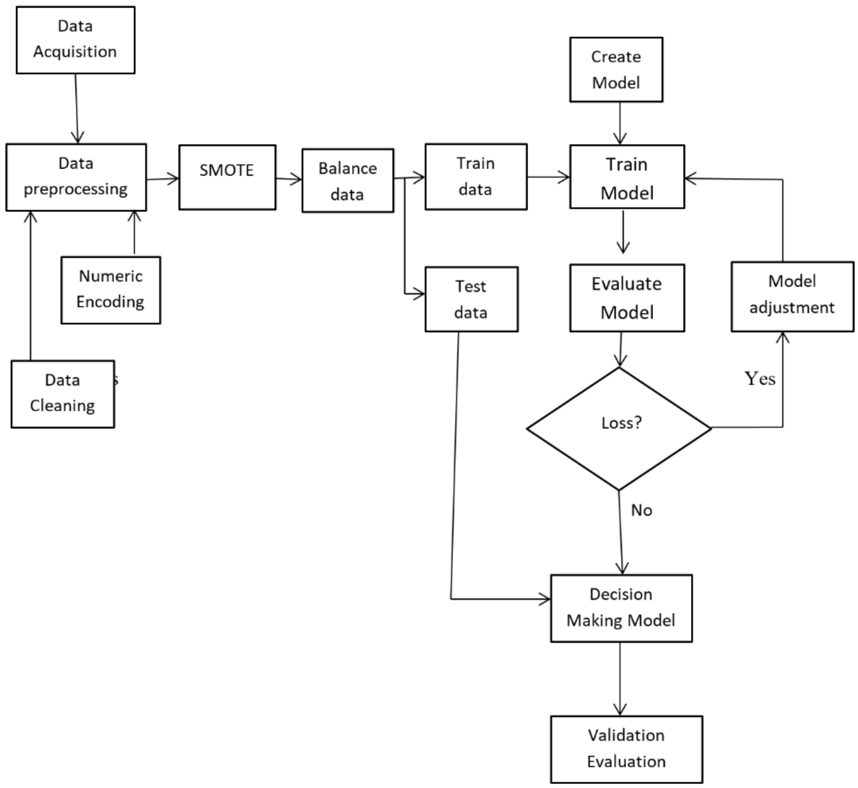

Figure 1: Methodology for loan classification

2.2.1 Dependent Variable

loan decision (0 and 1), this paper aims to predict whether the applicant should be approved for personal loan or not. In this context, if the applicant has good economic status and other supporting attributes, he will be approved of loan and the applicant will be deprived of loan if he is not supported by his attributes. So, to differentiate in a neural network 0 indicates that the personal loan will not be approved and 1 indicates that the applicant will succeed in getting a loan.

2.2.2 Independent Variable

Following variables are considered as an independent variable:

a. ID: Customer ID

b. Age: Age of the Customer
c. Experience: The amount of work experience in years
d. Income: the amount of annual income (in thousands)
e. Zip Code: Zip Code where customer lives
f. Family: Number of family members
g. CCA vg: Average monthly credit card spending
h. Education: Education level (1: Bachelor, 2: Master, 3: Advanced Degree)
i. Mortgage: Mortgage of house (in thousands)
j. Securities Account: Boolean of whether customer has a securities account
k. CD Account: Boolean of whether customer has Certificate of Deposit account
l. Online: Boolean of whether customer uses online banking
m. Credit Card: Does the customer use a credit card issued by the bank?

2.3 Dataset Collection

Once the required data is collected, it is tailor made for neural network with the help of data preprocessing. It is only after this step that the data is fed to the neural network. This phase mainly involves exploratory data analyzing, data cleaning and augmentation, finding and replacing missing values, and data normalization. It also involves visualization of major attributes to observe their distribution pattern. The process also includes observing the co-relation coefficient through heat map to learn about the relation and impact of different attributes. Outliners should be removed from the dataset and Missing value should be either delete or place with values with the help of mean, median etc. dataset can be split into training and testing set (70 and 30). Following steps are done in data preprocessing:

a. Importing libraries.
b. Importing the data set.
c. Finding the missing values.
d. Encoding categorical data.
e. Splitting data into training and testing set.

2.4 SMOTE

In this for synthetic minority oversampling technique and used to oversample the minority class. In this process, first selection a minority class instance at a random and find its K nearest minority class neighbors. Then synthetic instance is created by choosing and synthetic instances are generated as a convex combination of the two chosen instances a and b. It can be represented in an example as below shown in figure 2

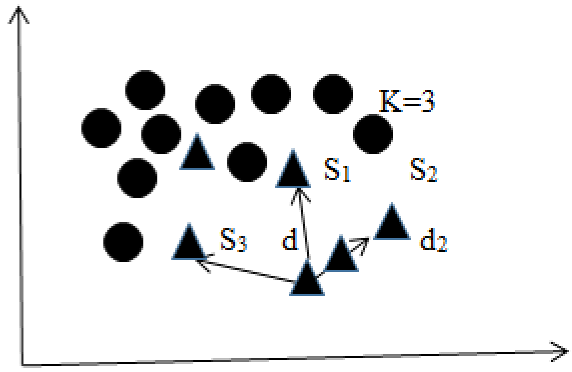

Figure 2: SMOTE

Here, SMOTE(sampling_strategy is auto, random_State is None, K_neighbours is 5, n_jobs is none) is library function. n_jobs represents no of CPU cores to do cross validation. We generally set none, it means single processor. k_neighbours represents no of instances for neighborhood. Default value is 5. Random_state is like a seed used by the random number generator. Default value is None. Sampling_strategy represents sampling information to resample the data. Default value is auto. The main steps are as below:

a. Identify feature vectors and it's the nearest neighbor.
b. Takes a differences between the two.
c. Multiple differences with random number between 0 and 1.
d. Identify the new point on the line segment by adding the random number to feature vector.
e. Repeat this process for identified features vector.

2.5 Back Propagation Algorithm

The back propagation algorithm has two passes, which are forward pass and backward pass. In forward pass, the calculation and storage of the intermediate variables (including outputs) is done for neural network in order from input layer to output layer. In the backward pass, it traverses in the reverse order. It is computed the output error and then go backward into the network and update the weights using gradient descent. This algorithm can be summarized in following steps:

Step 1: Firstly initialize all the weights, parameters by some random values.

Step 2: The network takes a training data as input, goes through the forward propagation step and finds the output probabilities for each class.

Step 3: Now we calculate binary cross entropy for loss calculation for batch as

$$L = -\frac{1}{m}\sum_{i=1}^{m} yi.\log(\overline{yi}) + (1-y_i).\log(1-\overline{yi}) \quad \text{Eq 1}$$

Where y is actual label and $\overline{yi}$ is classifier's output.

Step 4: After obtaining the predicted output from the forward pass, the next objective is to minimize the error by adjusting the network parameters. This is achieved through backpropagation followed by stochastic gradient descent.

Step 5: Repeats step two to four with all dataset in the training set.

2.6 Back Propagation Algorithm

Gradient descent algorithm is used to minimize a cost function J(W) parameterized by model parameters W.

Algorithm:

a. Randomly initialize the weights W.

b. Determine the gradient G of the cost function in relation to the W parameters.

c. Update the weights.

   $W_{new} = W_{old} – n * G$

   W_{new} = new weights

   W_{old} = old weights

   n=Learning Rate

   G=Gradient

d. Continue until the cost J(W) stops decreasing or unless other predetermined termination requirements are satisfied.

2.7 Back Propagation Algorithm

Now the model is trained using the training data. The batch size is set to 32 and a total of 100 iterations are involved on each epoch. During model compilation, we set the loss function to Binary Cross Entropy because this is a binary classification task. This loss function is:

$$\text{Loss}= -\frac{1}{N}\sum_{i=1}^{N} yi * log\big(p(yi)\big) + (1 - yi) * \log(1 - p(yi)) \quad \text{Eq 2}$$

where y is the label and p(y) is the probability of y

In order to minimize the above loss function, Adam optimizer was used and the learning rate was assigned to 0.1.During the model training, the validation set was also fed to the model.

2.7 Model Evaluation and Testing

The performance matrix is used to evaluate the accuracy of our model. There should be minimum loss in order to get the accurate model. The minimum loss of a classification task is typically measured by accuracy and F1 score.

A true positive (TP) is an outcome where the model correctly predicts the positive class. Similarly, a true negative (TN) is an outcome where the model correctly predicts the negative class.

A false positive (FP) is an outcome where the model incorrectly predicts the positive class. And a false negative (FP) is an outcome where the model incorrectly predicts the negative class.

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad \text{Eq 3}$$

If accuracy comes out to 0.91, or 91% means 91 correct predictions out of 100 total examples.

This F-score is way of the combining the precision and the recall of this model, and this is defined as harmonic mean of model's precision and the recall.

Precision is the fraction of true positive examples among the examples that the model classified as positive. In other words the number of true positives divided by the number of false positives plus true positives.

Recall is the fraction of examples classified as positive, among the total number of positive examples. In other words the number of true positives divided by the number of true positives plus false negatives. These metrics are calculated as:

$$Precision = \frac{TP}{TP+FP} \quad \text{Eq 4}$$

$$Recall = \frac{TP}{TP+FN} \quad \text{Eq 5}$$

$$F1 = 2 * \frac{Precision * Recall}{Precision+Recall} \quad \text{Eq 6}$$

AUC-ROC Curve: It is an evaluation metric which can be used to evaluate the performance evaluation of binary classifiers. Here TPR is plotted along the y-axis and FPR is plotted along the x-axis. Through this curve, the performance of a classifier can be analyzed.

2.8 Comparative Study

The methodology above describes a binary classifier using MLP. This classifier is finely tuned and hence its performance is also compared with other models like SVM and KNN on the given dataset and hence conclusion will be derived accordingly. The comparison is to be done on metrics such as AUC-ROC curve, precision, recall and f1-score.

3. Results and Discussion

3.1 Model Training Analysis

The model was trained on the training data sets that consist of a total of 5000 records and 14 attributes. Here 4520 are rejected loans and 480 are accepted loans. Attribute selection includes numeric and integer attributes along with some factor attributes relevant to the research problem. Some of the major attributes are depicted as follows:

3.2 Data attributes

The data has 14 attributes, thirteen are independent variables and one is dependent variables. Value of personal loan depends on the remaining thirteen attributes. Some attributes have high contribution on loan decision and some attributes have less contribution. Education, mortgage, income plays an important role regarding loan decision so these are explained detail below. Following attributes are present on data:

Table 1: Distribution of data in Imbalanced dataset

	ID	Age	Experience	Income	ZIP Code	Family	CCAvg	Education	Mortgage	Personal Loan	Securities Account	CD Account	Online	CreditCard
0	1	25	1	49	91107	4	1.6	1	0	0	1	0	0	0
1	2	45	19	34	90089	3	1.5	1	0	0	1	0	0	0
2	3	39	15	11	94720	1	1.0	1	0	0	0	0	0	0
3	4	35	9	100	94112	1	2.7	2	0	0	0	0	0	0
4	5	35	8	45	91330	4	1.0	2	0	0	0	0	0	1

Firstly, the data set is highly imbalanced only 10 percentage of loan is accepted and remaining 90 percentage of loan is rejected. Some of the attributes have high role on loan decision. These attributes are shown in figure. The imbalanced dataset are visualized in following Figures.

Figure 3 below shows the count of personal loan approvals (target class) and education of applicants respectively. On the left, 0 represents the number of applicants whose loan has not been approved and 1 represents the count of applicants whose loan is approved. The figure in the rights shows the education level of applicants 1 for bachelors, 2 for masters and 3 for more advanced degrees.

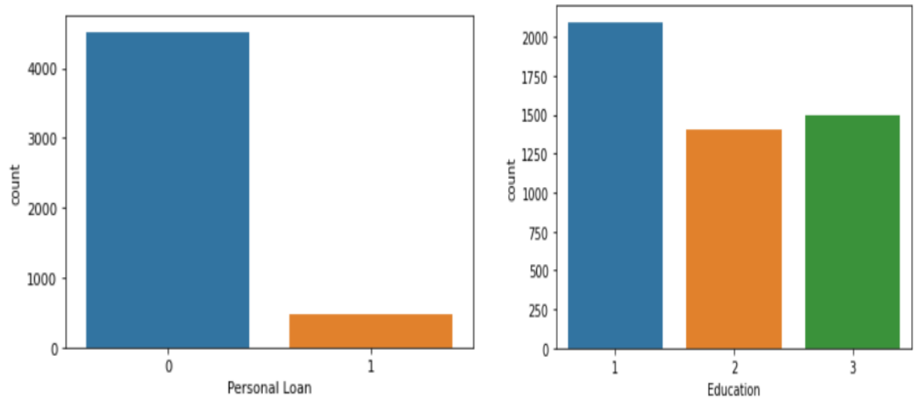

Figure 3: Personal loan and Education Count (imbalanced dataset)

In above figure, most of loans are rejected. Accepted loans are below 500 and remaining loan decisions are rejected. About half of applicants are done bachelor degree and total applicants who are done master are about thirteen hundred and fifteen hundred applicants are done advanced degree so these data are highly imbalanced regarding education.

Fig 4 below shows the distribution of income (in thousands) of the applicants. The average income of the applicants comes out to be 73 thousand hence the density is high on the left. Most of applicants have income less than hundred and very less applicants have income more than two hundred and fifty. It is highly imbalanced data as more applicants are under hundred. Bell curve is used to represent the distribution of income. Y axis represents in density instead of count as it is continuous data.

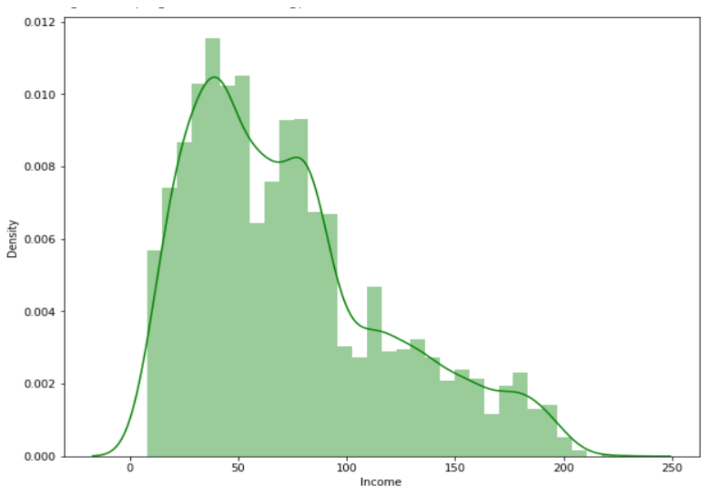

Figure 4: Income distribution (imbalanced dataset)

Figure 5 below shows the distribution of mortgages of the houses (in thousands) of applicants. As seen from the figure a significant number of the applicants have not taken out a mortgage facility. From the figure it represents that most of applicants do not have mortgage and some have mortgage and very less applicants have good mortgage facility.

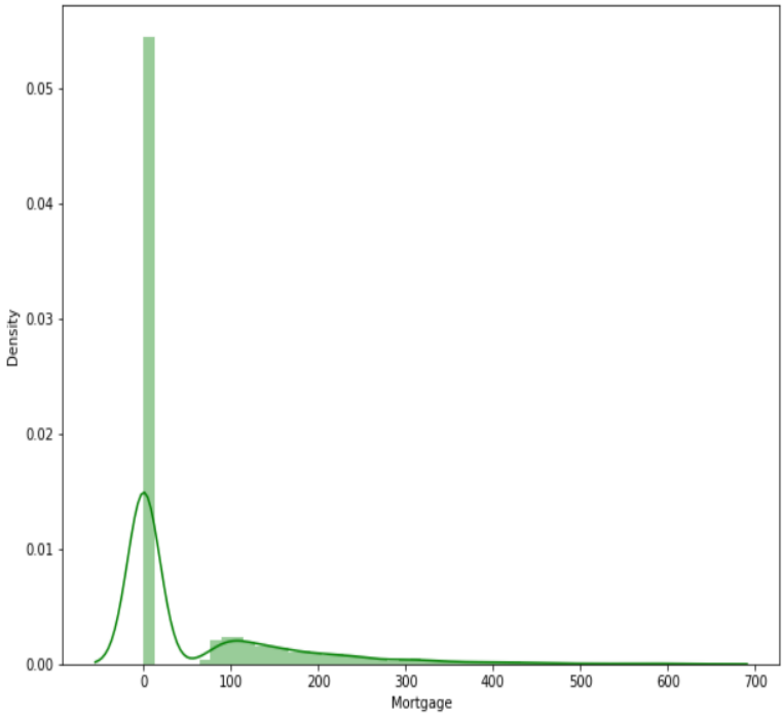

Figure 5: Mortgage distribution (imbalanced dataset)

Bell curve is used to represent the distribution of mortgage. Y axis represents in density instead of count as it is continuous data and average mortgage facility is 56.6 thousand.

3.3 Distributed of data in balanced dataset

The imbalanced dataset is not a healthy one to train the neural network as it causes over-fitting and makes the network biased. Therefore, an attempt has been made to balance the dataset using smote algorithm. Some of the attributes under the balanced dataset are visualized in following figures. Fig 6 below shows the count of personal loan approvals (target class) and education of applicants respectively. On the left, 0 represents the number of applicants whose loan has not been approved and 1 represents the count of applicants whose loan is approved. In this case, the number of applicants with and without personal loan is equal hence the balanced dataset. After oversampling the rejected applicants are increased and became equal to the accepted applicants and now data is quite balanced.

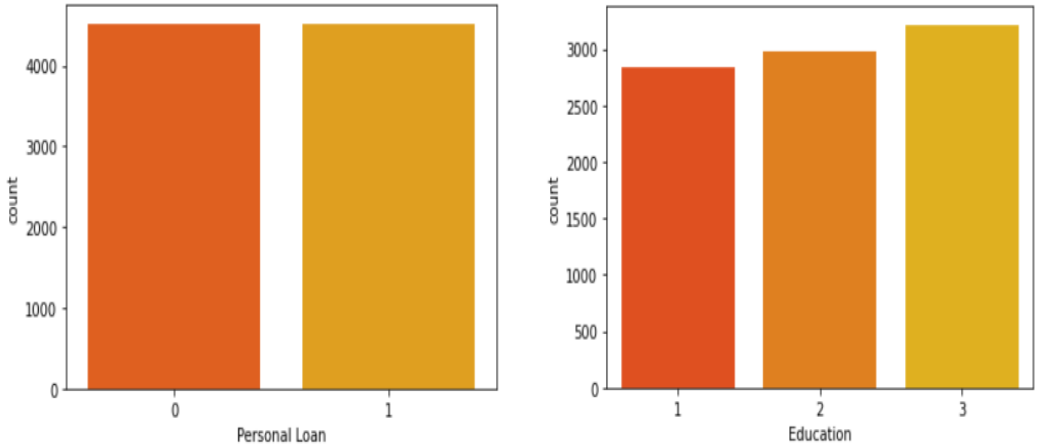

Figure 6: Personal loan and Education Count (balanced data)

The figure in the rights shows the education level of applicants 1 for bachelors, 2 for masters and 3 for more advanced degrees. Quite interestingly, the counts here are different from the former case. The number of applicants with masters and higher studies has increased as they are more likely to get personal loan approvals.

Figure 7 below shows the distribution of income (in thousands) of the applicants. As seen from the distribution plot, the income of the applicants in this case is more uniformly distributed with average around the middle and comes out to be 105 thousand.

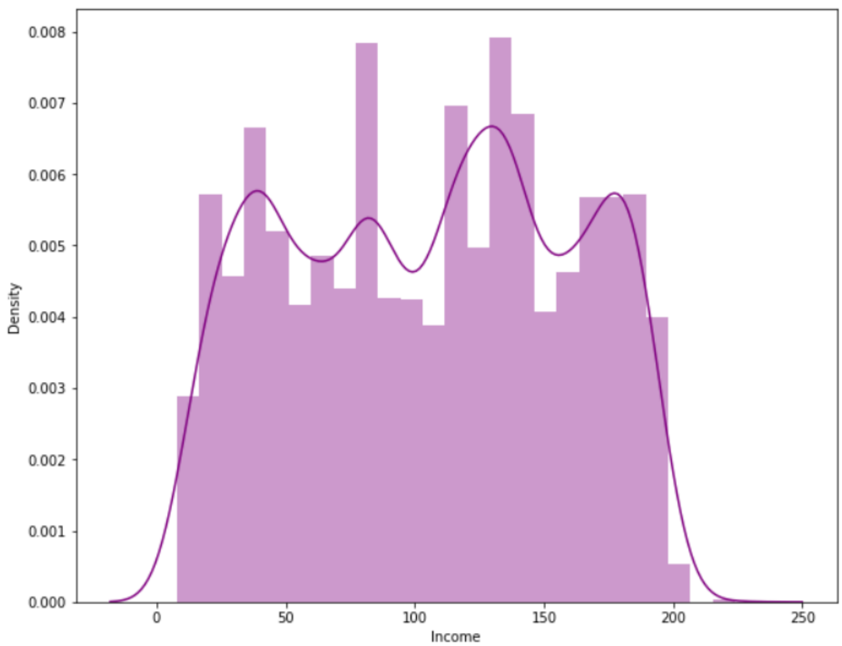

Figure 7: Income distribution (balanced dataset)

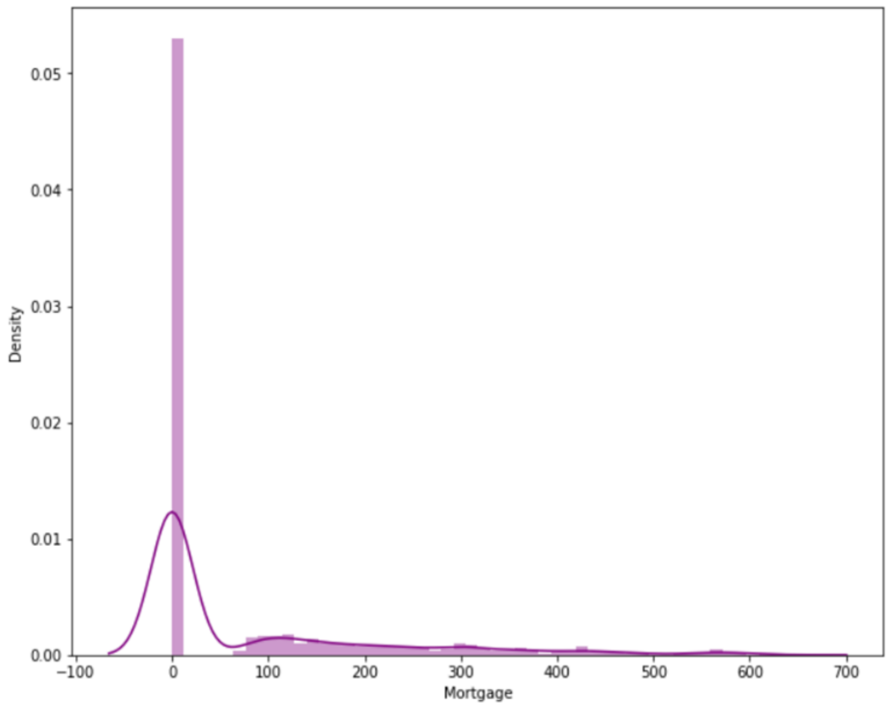

Figure 8: Mortgage distribution (balanced dataset)

Also, Figure 9 below shows the distribution of mortgages of the houses (in thousands) of the applicants. As seen from the figure a significant number of the applicants have not taken out a mortgage facility. The average amount of mortgage here comes out to be 77.7 thousand compared to 56.5 thousand in the previous case.

The visual representation of applicants getting loan approval versus the applicants not getting the loan approvals against their income can be depicted in the figure below. It shows the range of income of applicants that are likely to be approved and rejected for loans. In other words, it shows the distribution of two income groups. One with the people who got loan approval versus the second group that did not get loan approval. At the same time, it also depicts the cases in which the decisions may fall in either of the categories.

The visual representation of applicants getting loan approval versus the applicants not getting the loan approvals against their CCAvg can be depicted in the figure below. It shows the range of CCAvg of applicants that are likely to be approved and rejected for loans. In other words, it shows the distribution of two income groups. One with the people who got loan approval versus the second group that did not get loan approval. At the same time, it also depicts the cases in which the decisions may fall in either of the categories.

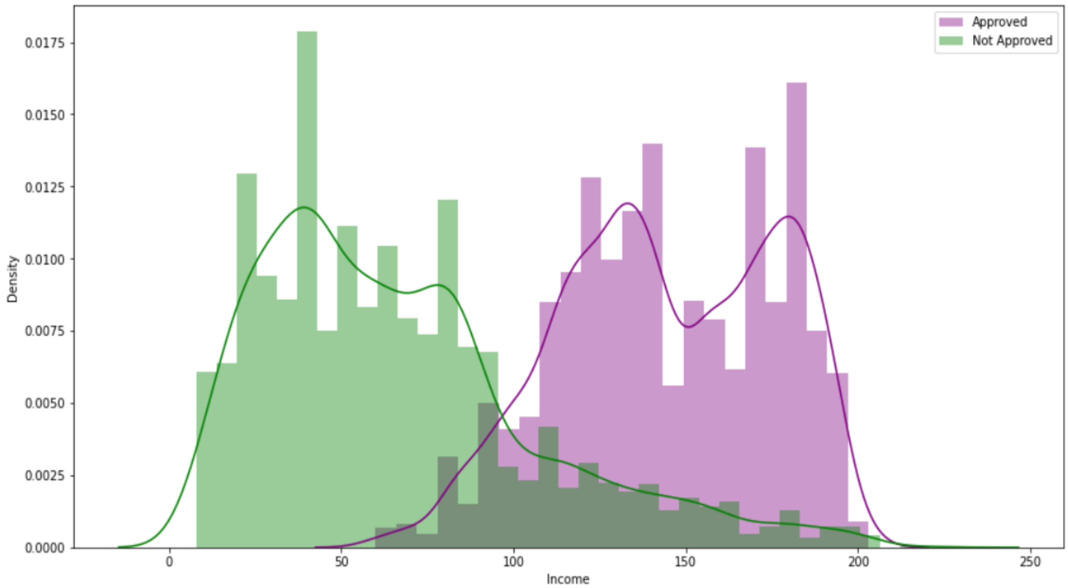

Figure 9: Loan approvals vs income (balanced dataset)

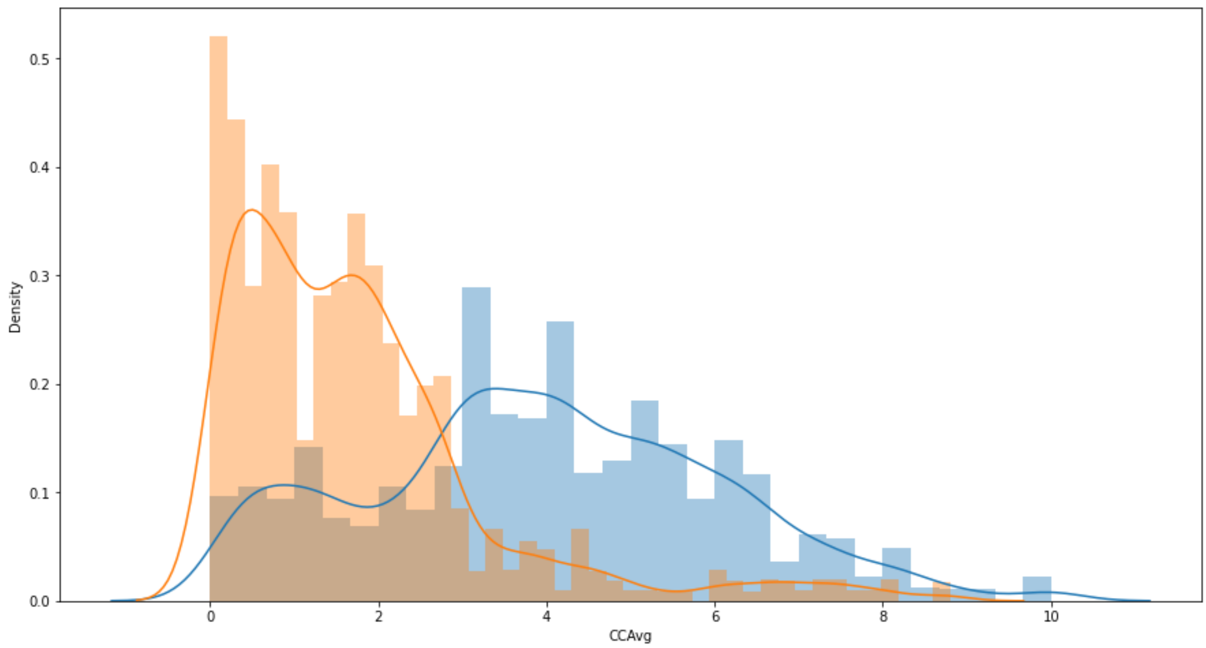

Figure 10: Loan approvals vs CCAvg (balanced dataset)

Figure 11 below shows correlation between various features. For example, it can be seen how experience and age have high correlation. As people grow older, they gain more work experience and vice versa. It is found how different features are related to personal loan feature. It shows the degree or amount of influence each attribute can have on decision making.

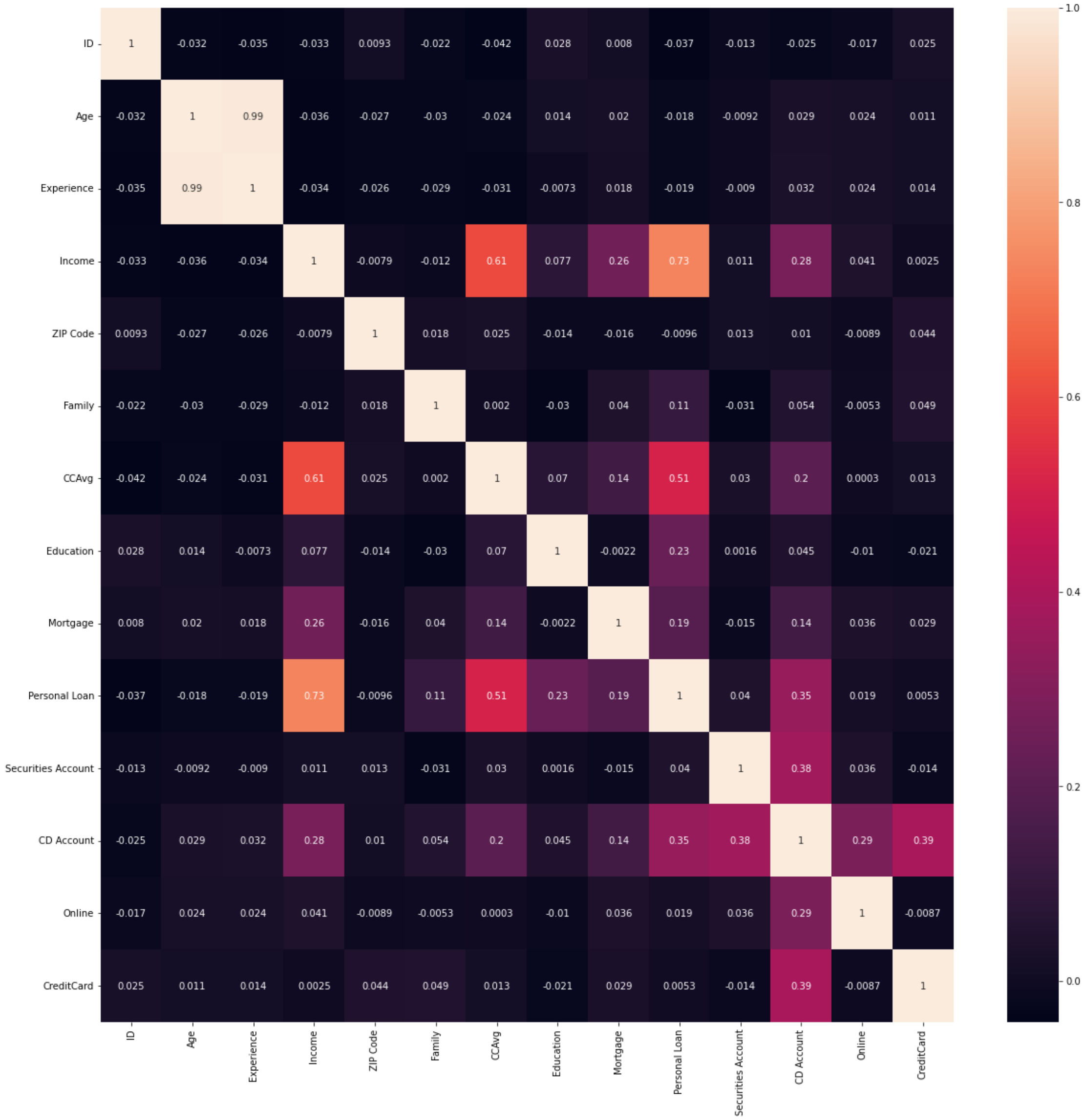

Figure 11: Correlation matrix

The main aim of the project is to apply an MLP on the datasets to find whether loan should be approved or not. The paper uses machine learning libraries including Scikit-learn, pandas, numpy, matplotlib and seaborn from Python 3.9. Furthermore, it uses keras and tensorflow for building MLP with appropriate input size, loss function, optimizer and learning rate. We have used a cross validation technique for splitting the dataset into training and testing data. Different performance metrics have been evaluated to figure out the effectiveness of the neural network.

In preprocessing of that data, the algorithm further divided this dataset into the training and the testing sets. That the process consists of the two phases, phase first is training phase where model is trained with training dataset. Each training data consist of X-train and Y-train data which store independent and the dependent features respectively. In our case, we used 10% of the data for testing and the other 90% for training. The training and testing data were then normalized using standard scales. This is followed by building the model, discussed below, then training the model through appropriate number of epochs. This is followed by testing phase where the model is first tested by using X_test and Y-test datasets to evaluate the accuracy and correctness of the model using other performance metrics described below. The model thus trained and tested is all set for prediction where we can predict the class of a new transaction data such that y_pred = model (X_new). This allows us to find how effective and consistent the model has been in making loan decisions.

3.4 Network Model

In this study, a multilayer perceptron has been used. Here, MLP classifier from sklearn module has been used for creating the model and GridSearch has been used to find the optimal parameter for the model. The optimal parameters for the model are as follows:

{'activation': 'tanh', 'alpha': 0.0001, 'hidden_layer_sizes': (150, 100), 'learning_rate': 'constant', 'max_iter': 100, 'solver': 'sgd'}

The classifier thus developed is trained using the X_train and the Y_train for 100 epochs. The process of the training can be observed from below data.

```
Iteration 1, loss = 0.55019050
Iteration 2, loss = 0.34125560
Iteration 3, loss = 0.27606763
Iteration 4, loss = 0.25993781
Iteration 5, loss = 0.23615281
Iteration 6, loss = 0.21687573
Iteration 7, loss = 0.20543686
Iteration 8, loss = 0.19532038
Iteration 9, loss = 0.18216090
Iteration 10, loss = 0.16983400
Iteration 11, loss = 0.16216931
Iteration 12, loss = 0.14660711
Iteration 13, loss = 0.14017379
Iteration 14, loss = 0.13367704
Iteration 15, loss = 0.12295981
Iteration 16, loss = 0.11442916
Iteration 17, loss = 0.11003679
Iteration 18, loss = 0.10741663
Iteration 19, loss = 0.10330445
Iteration 20, loss = 0.09332315
```

The MLP consists total 13 input variables, 1 hidden layer and the output layer with two neurons that represents that classifier. The hidden layers are using 'tanh' as activation function and learning rate given by 'alpha'. The network is trained by using supervised learning. The algorithm optimizes the neuron weights using the 'sgd' optimizer to minimize the error between actual and desired output. As a result of this the loss which is initially 0.55 goes on decreasing to become 0.003 in the 100th epoch. At the end of 100 epochs, the model has provided with following classification report.

	precision	recall	f1-score	support
0	0.98	0.96	0.97	442
1	0.97	0.98	0.97	462
micro avg	0.97	0.97	0.97	904
macro avg	0.97	0.97	0.97	904
weighted avg	0.97	0.97	0.97	904
samples avg	0.97	0.97	0.97	904

The loss curve for the model is as follows:

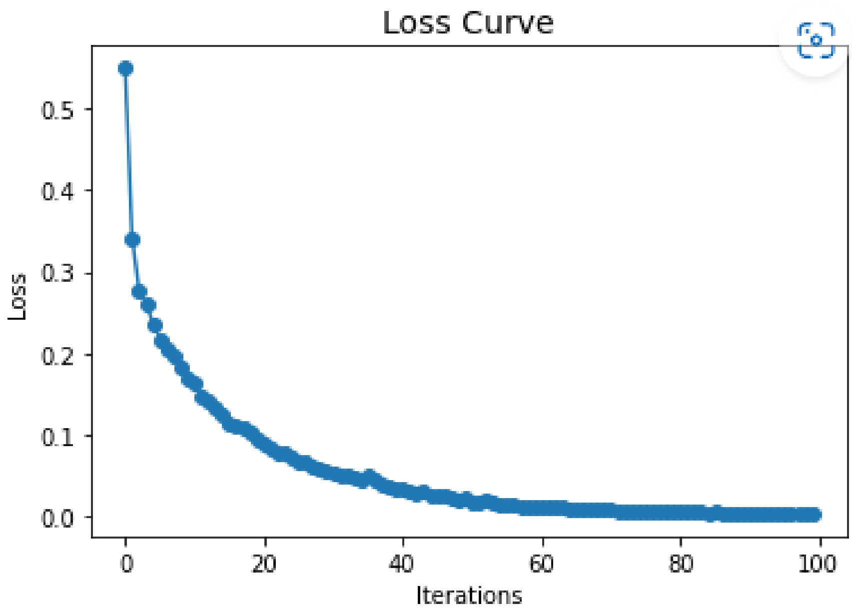

Figure 12: loss curve vs iteration

From the above figure the result of this the loss which is initially 0.55 goes on decreasing to become 0.003 in the 100th epoch. After 80th iterations the loss is almost constant and no need to iteration after 100th epoch.

3.5 Model Performance Analysis

The performance of model is analyzed in terms of the metrics such as AUC-ROC curve, precision, recall and f1-score. Confusion Matrix is generated for the MLP model which is as shown in below

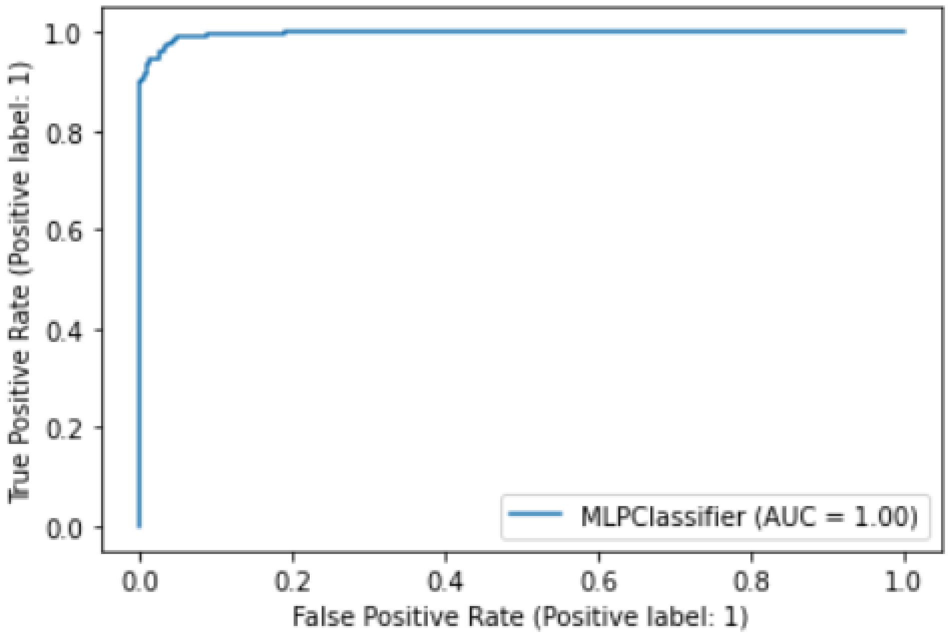

Figure 13: AUC and ROC curve

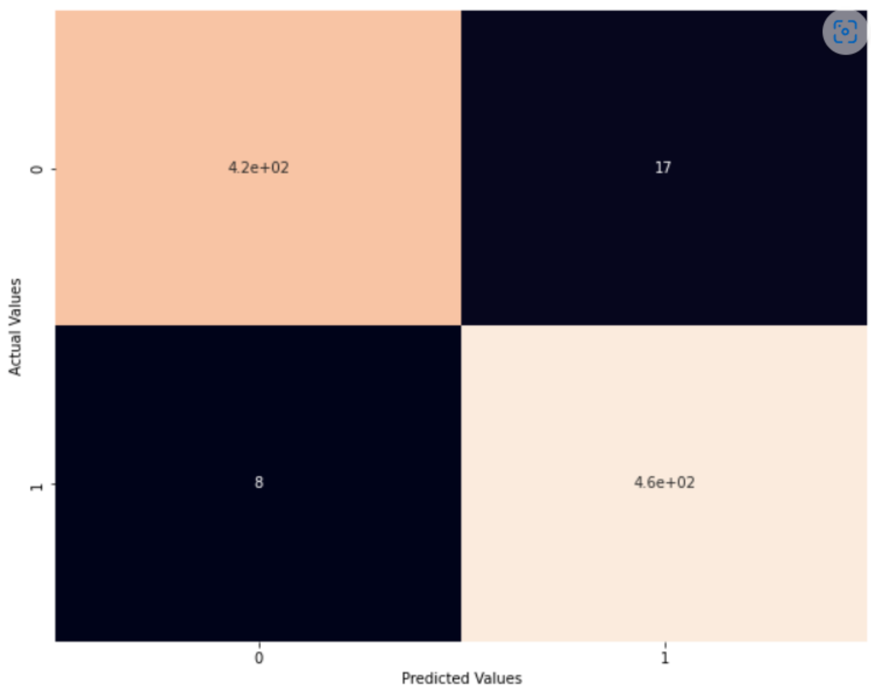

Figure 14: Confusion Matrix for MLP

From the above figure, true positive (TP) is $4.6 x 10^2 = 460$, FN = 8, FP = 17 and TN = $4.2 x 10^2 = 420$.

Performance metrics and its value are as below:

Table 2: Performance Metrics and its Values on MPL

SN	Metrics	Values
1	AUC value	0.999
2	Precision Score	0.975
3	Recall Score	0.970
4	F1-Score	0.970

3.6 Overall comparison

The individual performance analysis of each model can be now compared with each other. The comparison shows that MLP shows betterment in evaluation metrics compared to other model KNN and RF. The validation process also requires many comparisons between predicted result and training result. The techniques that are based on decision were looked into. These include precision, recall, accuracy and F1 score.

The validation process also requires many comparisons between predicted result and training result. The techniques that are also based on decisions were also looked into. These include precision, recall, accuracy and F1 Score. which is shown in table below.

Table 3: Overall Comparisons

SN	Metrics	Values for KNN Model	Values for RF Model	Values for MLP Model
1	AUC value	0.9535	0.9259	0.999
2	Precision Score	0.9333	0.9509	0.975
3	Recall Score	0.9809	0.9045	0.970
4	F1-Score	0.9565	0.9272	0.970

There is no threshold for a good precision value so the result of 0.97 can be considered to be good for an loan classification. Similarly perfect recall score is considered to be 1 and the result of this model is 0.97 which is close.

As a F1 score of 1 is generally said to be that of a perfect model, this project has the F1 score of 0.97 which is very close to 1 which is a good result. Although it might have helped to make a comparison of Precision, Accuracy, Recall and F1 score between this model and other ones mentioned above, the fact that the proposed models' metrics are already high means that a direct comparison may not be needed but it has been mentioned in above table

reason to be rejected, it can be refer the attribute that should improve to get loan. It is used medium-size date, data can be increased too.

4. Conclusion

The development of machine learning-based Quantitative Structure-Activity Relationship (QSAR) models for predicting the toxicity of herbal and synthetic organic compounds marks a significant advancement in computational toxicology. Our study demonstrates that Logistic Regression, Random Forest, and Support Vector Machines (SVM) can achieve high accuracy (>90%) in predicting hepatotoxicity, neurotoxicity, and general acute toxicity. Random Forest slightly outperformed the others, with an accuracy of 92.78%, precision of 98.73%, recall of 86.67%, and F1 score of 92.31%. SVM exhibited perfect recall (100%) but lower precision (87%), making it particularly suitable for applications where missing toxic compounds is critical, even at the cost of some false positives.

Feature importance analysis provided valuable insights into the structural determinants of toxicity. For Random Forest, key features included molecular fingerprints corresponding to substructures such as aromatic rings, nitro groups, and tertiary amines, which are well-known for their association with toxic effects. Logistic Regression and SVM highlighted the significance of molecular weight and lipophilicity (logP), consistent with established toxicological principles. These findings not only validate the models' predictive capabilities but also offer actionable insights for designing safer compounds by identifying structural features that contribute to toxicity.

References:

i. Abdou, H. and Pointon, (2018) ' Applications of Artificial Intelligence in commercial banks–A research agenda for behavioral finance' *Journal of behavioral and experimental technology*, 27, p.100-352

ii. Akkoc, S. (2017) 'An empirical comparison of conventional techniques, neural networks and the three stage hybrid adaptive neuro fuzzy inference system (ANFIS) model for credit scoring analysis: the case of Turkish credit card data', *European Journal of Operationa*

iii. Ubarhande, Arti Chadani 'Development of a credit default model using R and Neural Network', *Ieeexplore.ieee.org.*

iv. Bishwajeet Pandey, Naman Bhoj(2021) 'A Comparative analysis of feature selection technique for malicious website detection in SMOTE balanced data, *ResearchGate.*

v. Matoussi, H., &Abdelmoula, A.k (2020). Using a Neural Network-Based Methodology for Credit–Risk Evaluation of a Tunisian Bank. *ResearchGate.*

vi. Jeremy D.Turiel and Tomaso(2019),'P2P Loan acceptance and default prediction with artificial intelligence',*arxiv.org* .

vii. J. Alspector (2020)"Data duplication and inbalanced problem,".SpringerLink.

viii. S. Shetty, A. Sharma, A. Patil and A. Patil, "Review on Methodologies of data balancing," Proceedings 2019*: Conference on Technologies for Future Cities (CTFC), 2019.*

ix. W. Ullah, A. Ullah, I. U. Haq, M. Khan, M. Sajjad and S. W. Baik, "CNN features with bi-directional LSTM for real-time anomaly detection in surveillance networks," *Multimedia Tools and Applications*, pp. 16979-16995, 2021.

x. Shorouq Fathi Eletter(2017)" Loan decision models for the Jordanian commercial banks" ResearchGate.

xi. Mckinsey(2017)"Notes from the AI frontier: Modeling the impact of AI on the world economy ", *International journal of innovative technology and exploring engineering.*

xii. Pradeep Kumar singh(2020),"Digital Technologies for Advancements in Education, Management and IT" *Springer*.

xiii. "Comparative analysis of features selection techniques for malicious website detection in SMOTE balanced data," 2021 *15th International Conference on Signal Processing and Communication Systems (ICSPCS)*, pp. 1-5, 2021.

xiv. A G Pertiwi, N Bachtiar, R Kusumaningrum, I Waspada, A Wibowo. "Comparison of performance of k-nearest neighbor algorithm using smote and k-nearest neighbor algorithm without smote in diagnosis of diabetes disease in balanced data"

xv. "Financial Information Asymmetry: Using Deep Learning Algorithms to Predict Financial Distress*" Chyan-long Jan, 2021*.

xvi. W. Ullah, A. Ullah, I. U. Haq, M. Khan, M. Sajjad and S. W. Baik, "CNN features of with bi-directional LSTM for real time anomaly detection in surveillance network*" Multimedia Tools and Applications, pp*. 16979-16995, 2021.