A

Major Project Report

On

Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

Sai Rakshitha Kulkarni	**(187R1A05B0)**
Puppala Sai Dinesh	**(197R5A0503)**
Gadasu Sai Charan	**(187R1A0579)**

Under the Guidance of

Gadepaka Latha

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)

Recognized Under Section 2(f) & 12(B) of the UGCAct.1956,

Kandlakoya (V), Medchal Road, Hyderabad-501401.

2018-22

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled **“Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning”** being submitted by **Sai Rakshitha Kulkarni (187R1A05B0), Puppala Sai Dinesh (197R5A0503), Gadasu Sai Charan (187R1A0579)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by him/her under our guidance and supervision during the year 2021-22.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Mrs.G.LATHA
(Associate Professor)
INTERNAL GUIDE

Dr. A. RAJI REDDY
DIRECTOR

Dr. K. SRUJAN RAJU
HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on ________________

ACKNOWLEDGMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Mrs. G. Latha,** Associate Professor for her exemplary guidance, monitoring and constant encouragement through out the project work. The blessing, help and guidance given by her shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) **Mr. A. Uday Kiran, Mr. J. Narasimha Rao, Mrs. G. Latha, Dr. T. S. Mastan Rao, Mr. A. Kiran Kumar,** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

Sai Rakshitha Kulkarni	**(187R1A05B0)**
Puppala Sai Dinesh	**(197R5A0503)**
Gadasu Sai Charan	**(187R1A0579)**

ABSTRACT

In the modern era of the banking sector, banks have large datasets containing customers' information and their history of transactions. Hence, banks need to classify these large datasets to be able to analyze these customers' behaviours for using it in the best way to suggest a suitable strategy to attain the highest benefits, customer satisfaction to increase profitability.

To achieve this purpose, we are Building an ensemble learning model that has been proven to be typically more accurate and robust than individual classifiers, it is an important information management task of commercial banks and loan lenders. Profiling produces customer profiles, which provide the banks with a full description of their customers based on a set of attributes.

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

PAGE.NO.

1. INTRODUCTION

1. INTRODUCTION

1.1 PROJECT SCOPE

In the modern era of the banking sector, banks have large datasets containing customers' information and their history of transactions. Hence, banks need to classify these large datasets to be able to analyze these customers' behaviours for using it in the best way to suggest a suitable strategy to attain the highest benefits, customer satisfaction to increase profitability. To achieve this purpose, we are Building an ensemble learning model that has been proven to be typically more accurate and robust than individual classifiers, it is an important information management task of commercial banks and loan lenders. Profiling produces customer profiles, which provide the banks with a full description of their customers based on a set of attributes.

1.2 PROJECT PURPOSE

With the rapid development of the loan lending market, a decision-making strategy based merely on human experience is no longer practicable. The probability of default (PD), which measures the risk that customers will be unable to repay their debts, has drawn most research attention among all risk management tasks. Loan lenders must decide carefully to avoid loss and maximize profit. Customers who are believed to repay their loans are defined as "good" customers, while those who are unwilling or unable to repay are defined as "bad". Classifying customers into the wrong category will incur two different types of loss. If an actually "good" customer is classified as "bad", the loan lenders will bear the corresponding opportunity cost.

It is an important information management task of commercial banks and loan lenders. Profiling produces customer profiles, which provide the banks with a full description of their customers based on a set of attributes. A Banking system includes a large dataset for transactions of customers of their credit cards. Hence, banks would be in need of customer profiling. Profiling bank customers cognize the issuer's decisions about whom to give banking facilities and what a credit limit to provide.

1.3 PROJECT FEATURES

The main features of this project are that the designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

2. SYSTEM ANALYSIS

2.SYSTEM ANALYSIS

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

2.1 PROBLEM DEFINITION

In the modern era of the banking sector, banks have large datasets containing customers’ information and their history of transactions. Hence, banks need to classify these large datasets to be able to analyze these customers’ behaviours for using it in the best way to suggest a suitable strategy to attain the highest benefits, customer satisfaction to increase profitability. To achieve this purpose, we are Building an ensemble learning model that has been proven to be typically more accurate and robust than individual classifiers, it is an important information management task of commercial banks and loan lenders. Profiling produces customer profiles, which provide the banks with a full description of their customers based on a set of attributes. In this project, we have taken a dataset which consists of customer personal details like name, employment, etc. and the credit score details. The main goal is to predict if a particular customer is satisfied with the bank schemes or not. Our proposed model is an ensemble model which fuses the output of five base classifiers

2.2 EXISTING SYSTEM

In the Banking industry, credit card evolution is a noticeable occurrence. Each banking system includes a huge dataset for customer's transactions of their credit cards. Therefore, banks would be in need of customer profiling. Profiling bank customer’s cognize the issuer's decisions about whom to give banking facilities and what a credit limit to provide. It also helps the issuers get a better understanding of their potential and current customers.

2.2.1 LIMITATIONS OF EXISTING SYSTEM

- Requires Computerized infrastructure.
- Requires considerable development background research, planning and data analysis.

2.3 PROPOSED SYSTEM

The main idea of our proposed model is to improve profiling bank customers' behavior using different machine learning techniques. This model starts with the data set. Then data goes through the step of data preprocessing. Then five widely used base classifiers, i.e., extreme gradient boosting, gradient boosting decision tree, support vector machine, random forest, and linear discriminant analysis, are integrated. To amplify the strength and diversity of the base classifiers, a new backflow learning approach is proposed so that the base classifiers will relearn the misclassified data point. A final predictive result is obtained by fusing the prediction of all base classifiers through two-layer ensemble modelling. In this project, we are using credit scoring dataset. As it will be unstructured data at first we are performing data preprocessing on the dataset first then after the splitting the dataset into training and testing data. We are then fitting the training data into the individual classifiers. We are evaluating performance of various machine learning algorithms as base classifiers and later these base classifiers output will be fused and the voting classifier will give the final predictive result in the form of 1 or 0 customer satisfied or not. Voting classifier will choose best performing base classifier. Here credit scoring dataset may contain outlier/class imbalance and to remove outlier we are applying SMOTE algorithm. As base classifiers we are using SVM, Gradient Boosting Decision Tree Random Forest, Gradient Boosting and LDA (linear Deterministic Analysis).

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- Customer Profiling helps banks get to know their customers on a more granular level.
- This approach can give banks better understanding of their clients, allowing them to align tactics and strategies to each customer type for better sales, loyalty, growth, and cost containment.

2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is done. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Behavioural Feasibility

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 BEHAVIOURAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specifies the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

• System	: I5 processor
• Hard Disk	: 20 GB
• Input Devices	: Keyboard, Mouse
• Ram	: 4GB

2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements.

• Operating system	: Windows 7,8,10
• Programming Language	: python 3.6+
• IDE	: Anaconda Spyder, VScode

2.6 Modules Description

In this project we have Two modules

1. Data pre-processing.
2. SMOTE Algorithm
3. Base Classifiers
4. Voting Classifier

- **Data Pre-processing:**

Data comes in all forms, most of it being very messy and unstructured. They rarely come ready to use. Datasets, large and small, come with a variety of issues- invalid fields, missing and additional values, etc. Hence, it’s important to preprocess the data.

- **SMOTE Algorithm:**

One of the problems observed in classification is an imbalanced dataset i.e. the no. of data points in the majority class is very large compared to that of the minority class. If the imbalanced data is not treated beforehand, then this will degrade the performance of the classifier model. SMOTE stands for Synthetic Minority Oversampling Technique. SMOTE is an oversampling technique where the synthetic samples are generated for the minority class.

```
SMOTE ALGORITHM:
Input: the raw training dataset T
Output: the oversampled training dataset T+
•   Calculate the oversampling data size N+ and initialize T+=T
•   Identification of noise, borderline and edge samples and calculate sample importance.
•   Synthetic minority samples generation
        for I from 1 to N+
            select one sample from borderline, edge and noise minority samples as x+
            w.r.to their sample importance;
            select a nearest neighbour of x+ in majority and minority classes as X+
            w.r.to their sample importance;
            calculate α value w.r.to the sample importance of x+ and X+;
            generate the synthetic minority sample and add it to T+;
    end for
•   Export the oversampled training dataset T+ to the classification model.
```

- **Base Classifiers:**

  Five widely used base classifiers, i.e., extreme gradient boosting, gradient boosting decision tree, support vector machine, random forest, and linear discriminant analysis, are integrated. These are considered as the base classifiers. The backflow learning approach is applied to train and optimize the base classifiers. Here the base classifiers will relearn the misclassified data point.

- **Voting Classifier:**

  A final predictive result is obtained by fusing the prediction of all base classifiers through two-layer ensemble modelling. In ensemble learning models, the fusion strategy, i.e., how multiple base classifiers are fused into an ensemble learner, could affect the performance of an ensemble learner significantly. Voting is a popular fusion strategy integrating the prediction of base classifiers. In majority voting, each base classifier will vote for their prediction, and the class label with higher votes is used as the final result.

3.ARCHITECTURE

3. ARCHITECTURE

3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure followed for Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning, starting from input to final prediction.

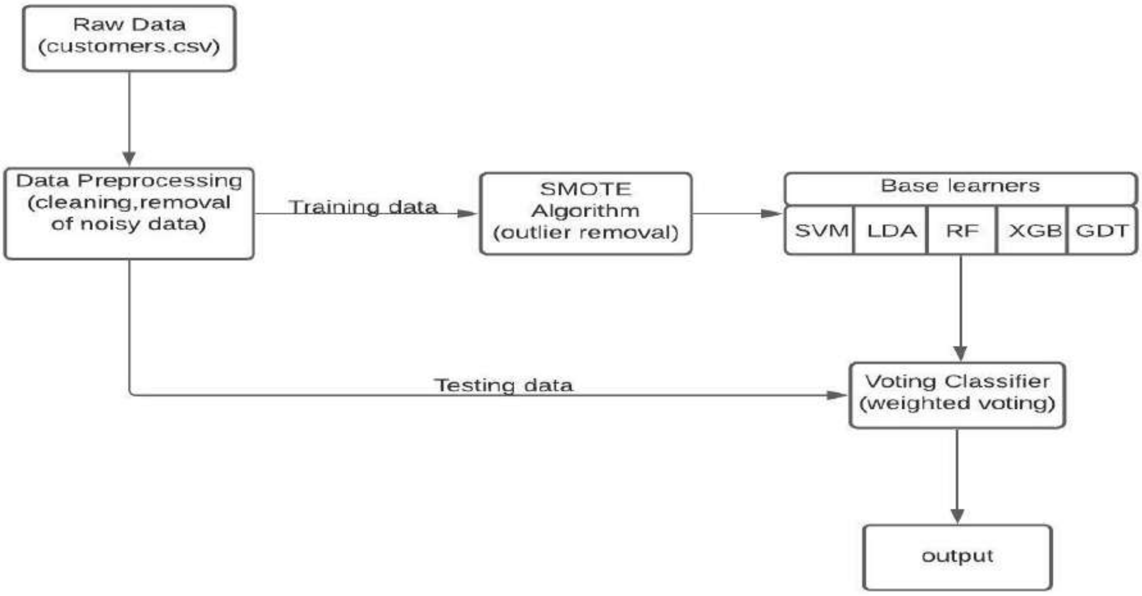

Raw Data (customers.csv)
Data Preprocessing (cleaning,removal of noisy data)
Training data
SMOTE Algorithm (outlier removal)
Base learners
SVM
LDA
RF
XGB
GDT
Testing data
Voting Classifier (weighted voting)
output

Figure 3.1 Project Architecture of Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning

3.2 DESCRIPTION

Input Data: Input data is a dataset consisting of attributes like salary, credit score, loan amount, job type, etc.

Data Processing: We are reading the data using the Pandas library. Then we are going to clean noisy and inconsistent data. Then we are splitting the data into training data and testing data.

SMOTE Algorithm: In this project we are using smote algorithm to generate minority class synthetics. We are making sure that both majority and minority class training examples are equal.

Base learners: To all base classifiers the training data is provided. The training data is fit into the models and it predicts whether the customer is satisfied or not and also gives the accuracy value of individual models.

Voting Classifier: The output of all the models are given to a voting classifier. It fuses all the outputs and with help of weighted voting it gives the final predictive output.

Output: In output it will predict whether the customer is satisfied or not in 0 and 1 form.

3.3 USE CASE DIAGRAM

The use case diagram is a graphical depiction of a user's possible interactions with a system. It shows various use cases and different types of users the system has. In this we have the general user and the customer profiling system. It shows the interactions between users and the system.

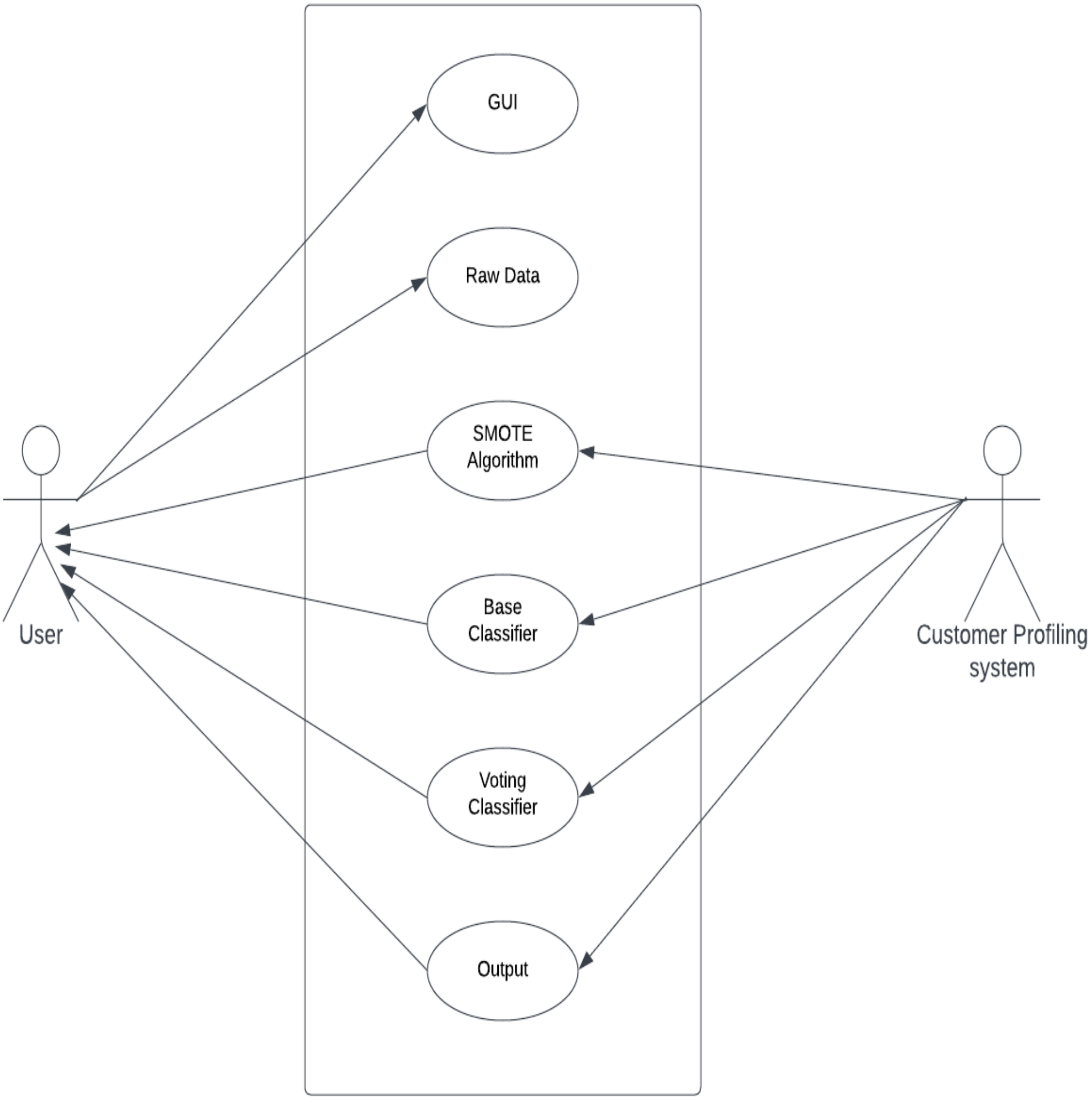


Figure 3.2 Use Case Diagram for user for Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning

3.4 CLASS DIAGRAM

A class Diagram is a collection of classes and objects. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints.

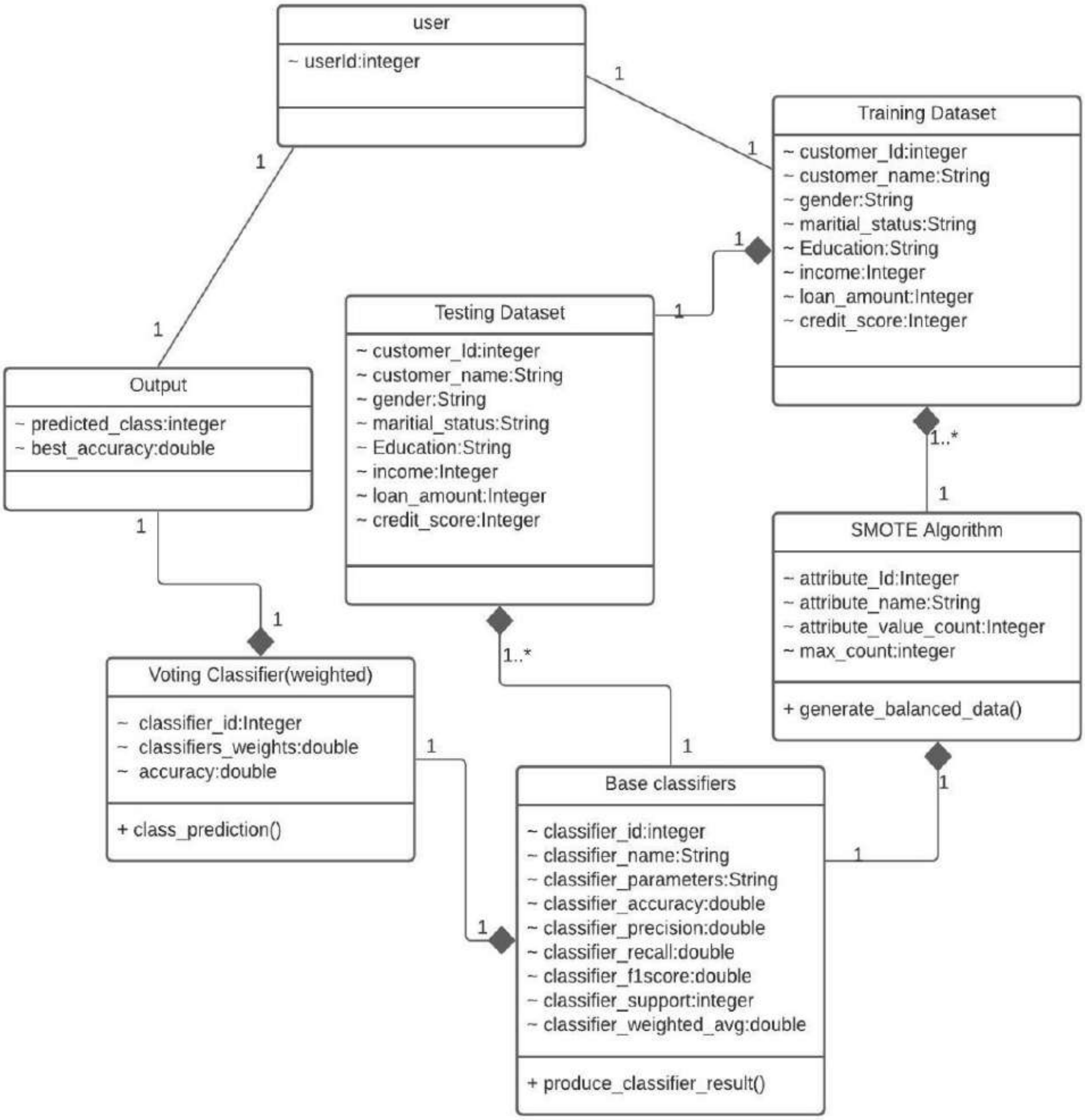


Figure 3.3 Class Diagram for Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning

3.5 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process.

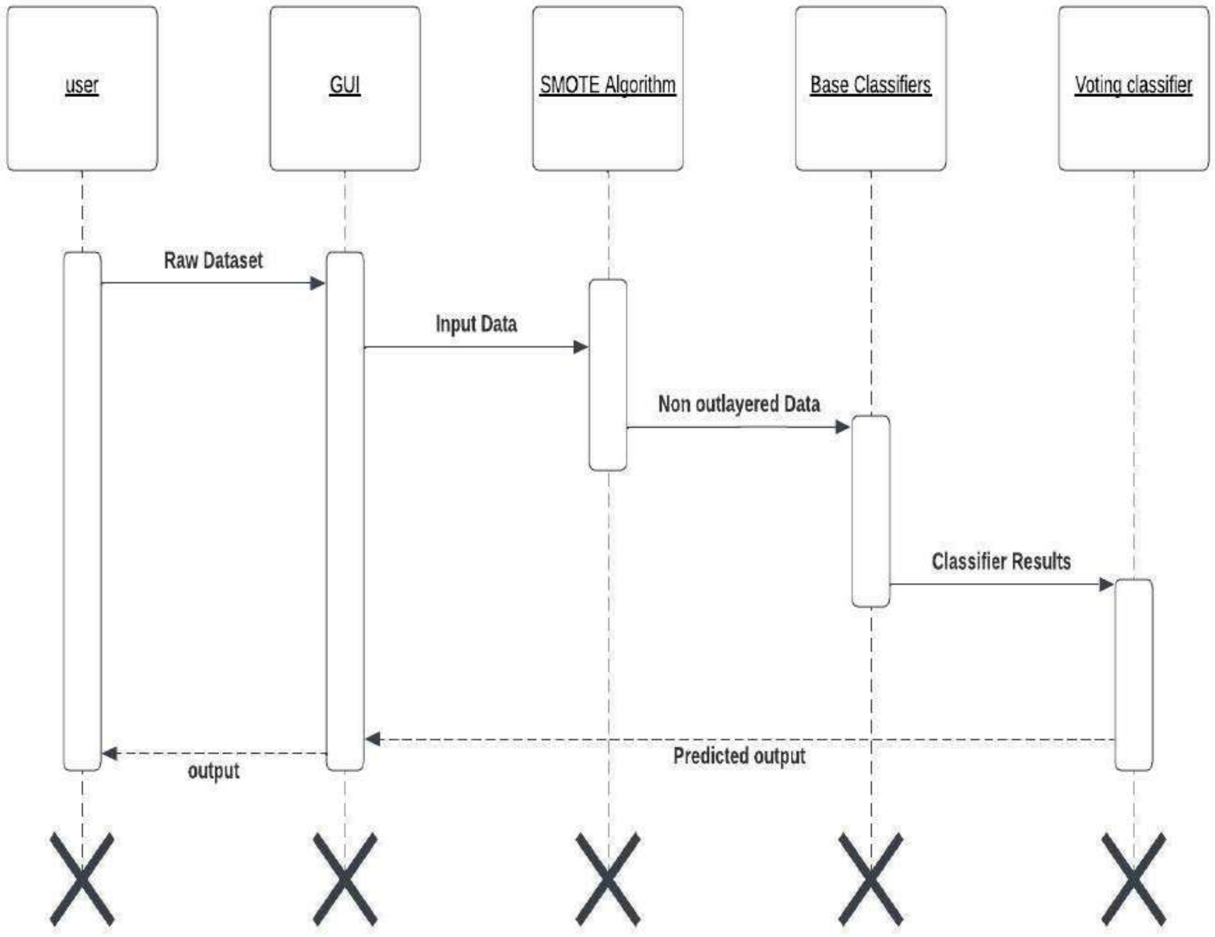


Figure 3.4 Sequence Diagram for Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning

3.6 DATA FLOW DIAGRAM

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow there are no decision rules and no loops.

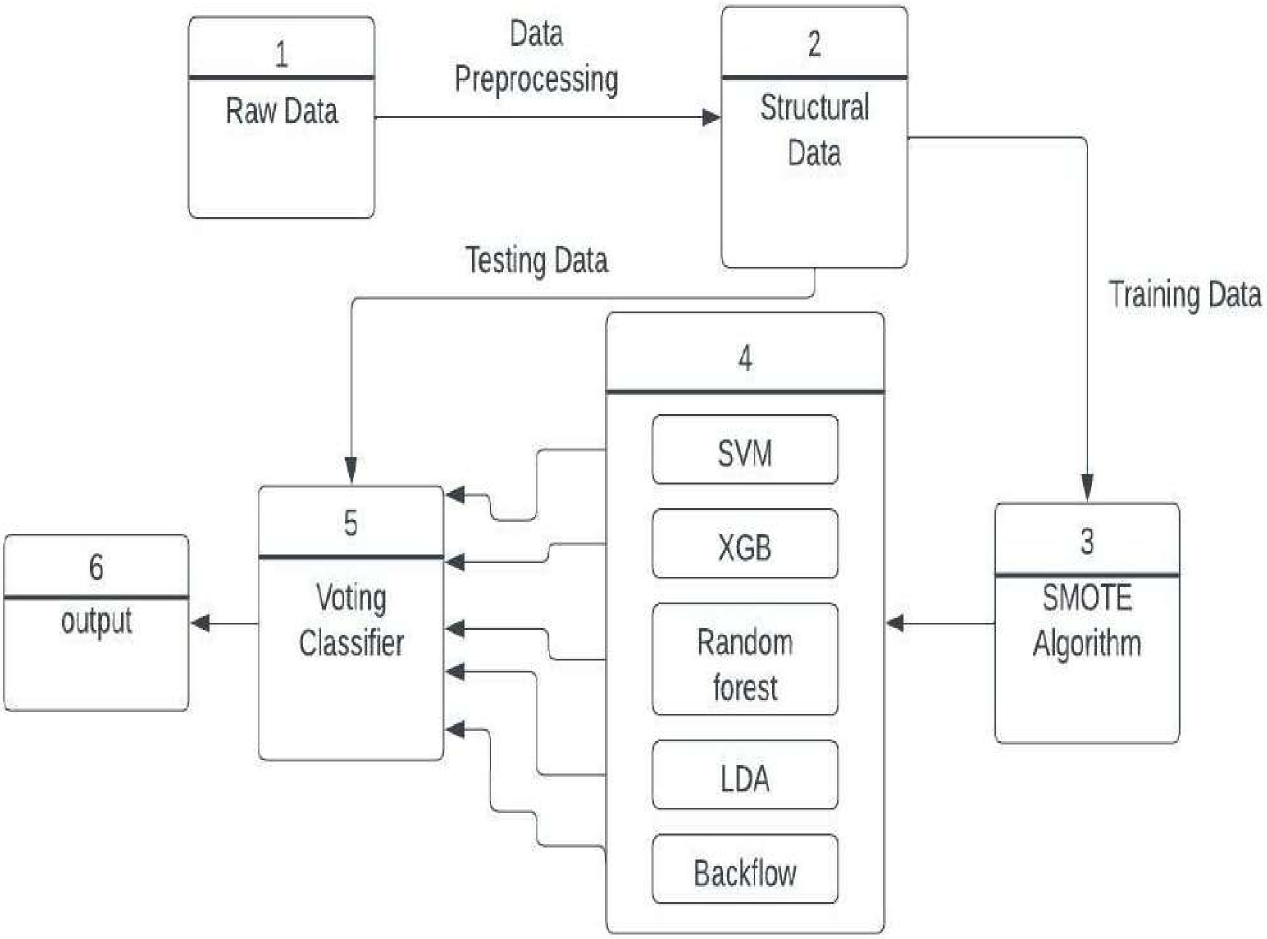


Figure 3.5 Data Flow Diagram for Customer Profiling and
Credit scoring in Banking using Back Flow Ensemble Learning

3.7 ACTIVITY DIAGRAM

It describes about flow of activity states. An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

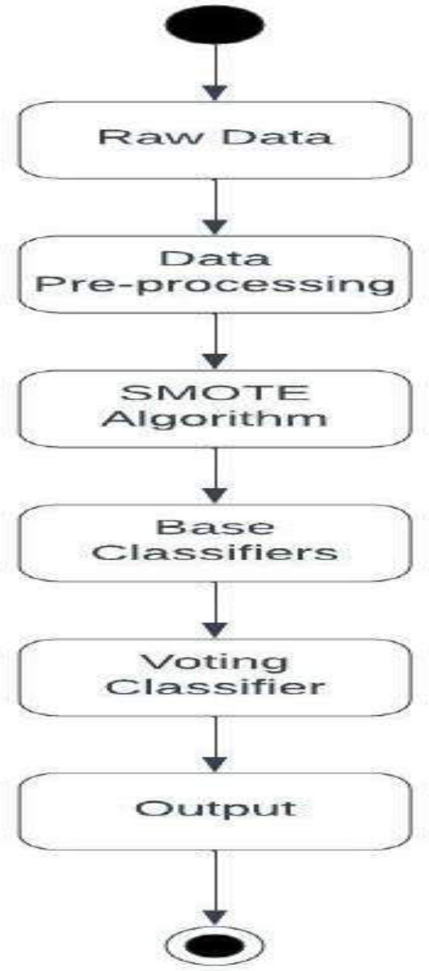


Figure 3.6 Activity Diagram for Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning

4.IMPLEMENTATION

4. IMPLEMENTATION

4.1 SAMPLE CODE

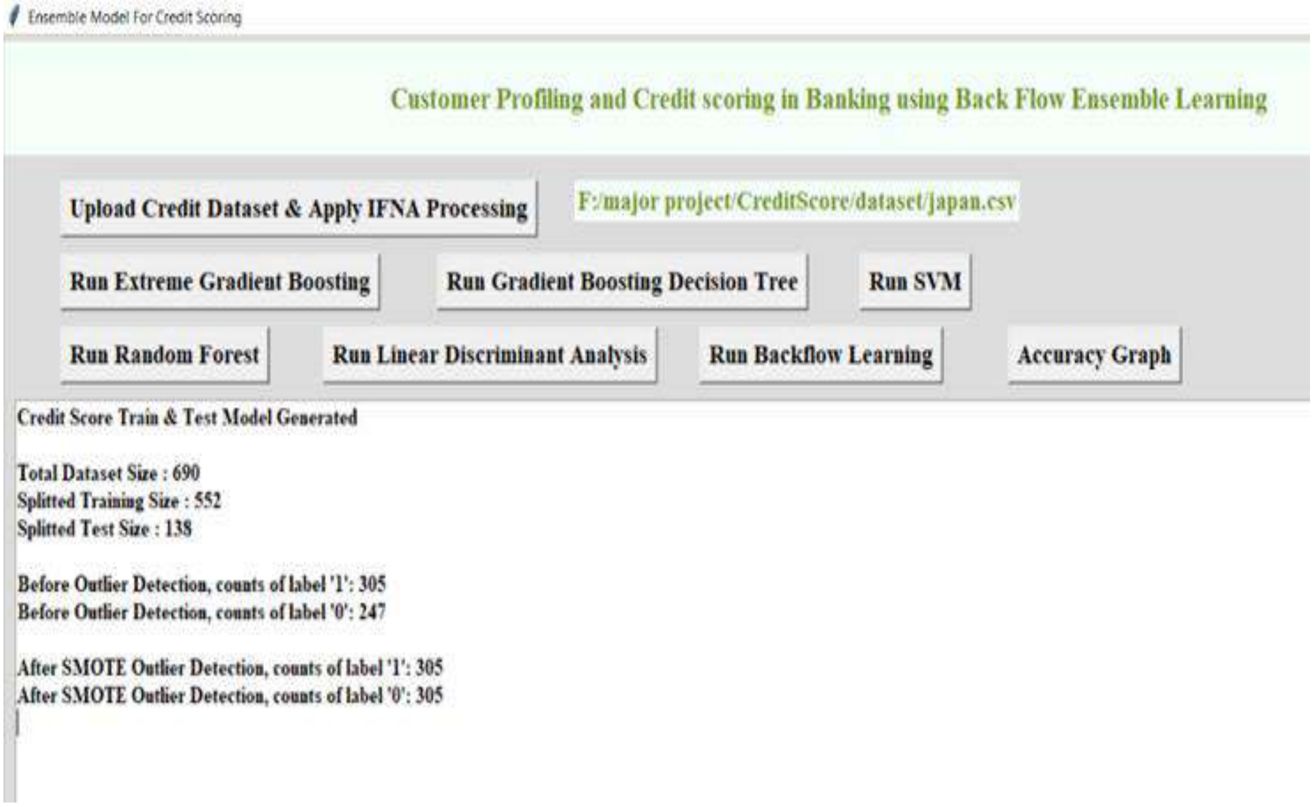
```
from tkinter import messagebox
from tkinter import *
from tkinter import simpledialog
import tkinter
from tkinter import filedialog
import matplotlib.pyplot as plt
from tkinter.filedialog import askopenfilename
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.ensemble import RandomForestClassifier, VotingClassifier
from sklearn import preprocessing
from imblearn.over_sampling import SMOTE
import xgboost as xgb
from sklearn.metrics import classification_report
from sklearn.ensemble import GradientBoostingClassifier
from sklearn import svm
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis

main = tkinter.Tk()
main.title("Ensemble Model For Credit Scoring")
main.geometry("1300x1200")

global filename
global cols
global x,y
global X_train, X_test, y_train, y_test
global boosting_acc,tree_acc,svm_acc,random_acc,linear_acc,backflow_acc

def prediction(X_test, cls):
    y_pred = cls.predict(X_test)
    for i in range(len(X_test)):
        print("X=%s, Predicted=%s" % (X_test[i], y_pred[i]))
    return y_pred

# Function to calculate accuracy
def cal_accuracy(y_test, y_pred, details):
    accuracy = accuracy_score(y_test,y_pred)*100
    textarea.insert(END,details+"\n\n")
    textarea.insert(END,"Accuracy : "+str(accuracy)+"\n\n")
    textarea.insert(END,"Report : "+str(classification_report(y_test, y_pred))+"\n")
    return accuracy
def upload():
    textarea.delete('1.0', END)
    global filename
    global cols
    global X,Y
    global X_train, X_test, y_train, y_test
    filename = filedialog.askopenfilename(initialdir="dataset")
    pathlabel.config(text=filename)

    train = pd.read_csv(filename)
    le = preprocessing.LabelEncoder()
    train['Gender'] = le.fit_transform(train['Gender'])
    train['Married'] = le.fit_transform(train['Married'])
    train['Bank_Customer'] = le.fit_transform(train['Bank_Customer'])
    train['Education'] = le.fit_transform(train['Education'])
    train['Ethnicity'] = le.fit_transform(train['Ethnicity'])
    train['Years_Employed'] = le.fit_transform(train['Years_Employed'])
    train['Prior_Default'] = le.fit_transform(train['Prior_Default'])
    train['Credit_Score'] = le.fit_transform(train['Credit_Score'])
    train['Drivers_License'] = le.fit_transform(train['Drivers_License'])
    train['Approved'] = le.fit_transform(train['Approved'])

    cols = train.shape[1]
    X = train.values[:, 0:cols-1]
    Y = train.values[:, cols-1]
    Y = Y.astype('int')
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.2, random_state = 0)
    textarea.insert(END,"Credit Score Train & Test Model Generated\n\n")
    textarea.insert(END,"Total Dataset Size : "+str(len(X))+"\n")
    textarea.insert(END,"Splitted Training Size : "+str(len(X_train))+"\n")
    textarea.insert(END,"Splitted Test Size : "+str(len(X_test))+"\n\n")

    textarea.insert(END,"Before Outlier Detection, counts of label '1': {}".format(sum(y_train ==
1))+"\n")
    textarea.insert(END,"Before Outlier Detection, counts of label '0': {} \n".format(sum(y_train ==
0))+"\n")

    sm = SMOTE(random_state = 2)
    X_train, y_train = sm.fit_resample(X_train, y_train)
    textarea.insert(END,"After SMOTE Outlier Detection, counts of label '1': {}".format(sum(y_train
== 1))+"\n")
    textarea.insert(END,"After SMOTE Outlier Detection, counts of label '0': {}".format(sum(y_train
== 0))+"\n")

def boosting():
    textarea.delete('1.0', END)
    global boosting_acc
    cls = xgb.XGBClassifier(n_estimators=1, max_depth=1)
    cls.fit(X_train, y_train)
    prediction_data = prediction(X_test, cls)
    boosting_acc = cal_accuracy(y_test, prediction_data,'Extreme Gradient Boosting Accuracy &
Classification Details')
    ''' X1=np.arange(0,len(X_train),1)
    y1=np.arange(0,len(y_train),1)
    plt.scatter(X1,y1,color='red')
    X2=np.arange(0,len(X_test),1)
    y2=np.arange(0,len(y_test),1)
    plt.scatter(X2,y2,color='green',s=10)
    plt.xlabel("X-axis")
    plt.ylabel("Y-axis")
    plt.show() '''

def tree():
    global tree_acc
    textarea.delete('1.0', END)
    cls = GradientBoostingClassifier(n_estimators=1, max_depth=1, random_state=0)
    cls.fit(X_train, y_train)
    prediction_data = prediction(X_test, cls)
    tree_acc = cal_accuracy(y_test, prediction_data,'Gradient Boosting Decision Tree Accuracy &
Classification Details')
    """X1=np.arange(0,len(X_train),1)
    y1=np.arange(0,len(y_train),1)
    plt.scatter(X1,y1,color='red')
    X2=np.arange(0,len(X_test),1)
    y2=np.arange(0,len(y_test),1)
    plt.scatter(X2,y2,color='green',s=10)
    plt.xlabel("X-axis")
    plt.ylabel("Y-axis")
    plt.show() """

def SVM():
    global svm_acc
    textarea.delete('1.0', END)
    cls = svm.SVC(C=2.0,gamma='scale',kernel = 'rbf', random_state = 0)
    cls.fit(X_train, y_train)
    prediction_data = prediction(X_test, cls)
    svm_acc = cal_accuracy(y_test, prediction_data,'SVM Accuracy & Classification Details')
    ''' X1=np.arange(0,len(X_train),1)
    y1=np.arange(0,len(y_train),1)
    plt.scatter(X1,y1,color='red')
    X2=np.arange(0,len(X_test),1)
    y2=np.arange(0,len(y_test),1)
    plt.scatter(X2,y2,color='green',s=10)
    plt.xlabel("X-axis")
    plt.ylabel("Y-axis")
    plt.show()'''

def randomForest():
    global random_acc
    textarea.delete('1.0', END)
    cls = RandomForestClassifier(n_estimators=1,max_depth=1,random_state=0)
    cls.fit(X_train, y_train)
    prediction_data = prediction(X_test, cls)
    random_acc = cal_accuracy(y_test, prediction_data,'Random Forest Accuracy & Classification
Details')
    ''' X1=np.arange(0,len(X_train),1)
    y1=np.arange(0,len(y_train),1)
    plt.scatter(X1,y1,color='red')
    X2=np.arange(0,len(X_test),1)
    y2=np.arange(0,len(y_test),1)
    plt.scatter(X2,y2,color='green',s=10)
    plt.xlabel("X-axis")
    plt.ylabel("Y-axis")
    plt.show() '''

def linear():
    global linear_acc
    textarea.delete('1.0', END)
    cls = LinearDiscriminantAnalysis()
    cls.fit(X_train, y_train)
    prediction_data = prediction(X_test, cls)
    linear_acc = cal_accuracy(y_test, prediction_data,'Linear Discriminant Analysis Accuracy &
Classification Details')
    ''' X1=np.arange(0,len(X_train),1)
    y1=np.arange(0,len(y_train),1)
    plt.scatter(X1,y1,color='red')
    X2=np.arange(0,len(X_test),1)
    y2=np.arange(0,len(y_test),1)
    plt.scatter(X2,y2,color='green',s=10)
    plt.xlabel("X-axis")
    plt.ylabel("Y-axis")
    plt.show() '''

def backFlow():
    global backflow_acc
    textarea.delete('1.0', END)
    cls1 = xgb.XGBClassifier(n_estimators=100, max_depth=100, learning_rate=0.01,
subsample=0.01)
    cls2 = GradientBoostingClassifier(n_estimators=100, learning_rate=1.0, max_depth=50,
random_state=0)
    cls3 = svm.SVC(C=2.0,gamma='scale',kernel = 'rbf', random_state = 0)
    cls4 = RandomForestClassifier(n_estimators=100,max_depth=50,random_state=0)
    cls5 = LinearDiscriminantAnalysis()
    cls6 = VotingClassifier(estimators=[
         ('xgb', cls1), ('dt', cls2), ('svm', cls3), ('rf', cls4), ('lda',cls5)], voting='hard')
    cls6.fit(X_train, y_train)
    prediction_data = prediction(X_test, cls6)
    backflow_acc = cal_accuracy(y_test, prediction_data,'Backflow Learning Accuracy &
Classification Details')
    """ p=[1,2]
    q=[(0.86,0.83,0.85),(0.88,0.90,0.89)]
    for pe,qe in zip(p,q):
        plt.scatter([pe]*len(qe),qe,color='green')
    plt.xticks([1,2])
    plt.axes().set_xticklabels(['0','1'])
    plt.show()
    X1=np.arange(0,len(X_train),1)
    y1=np.arange(0,len(y_train),1)
    plt.scatter(X1,y1,color='red')
    X2=np.arange(0,len(X_test),1)
    y2=np.arange(0,len(y_test),1)
    plt.scatter(X2,y2,color='green',s=10)
    plt.title("Training and Testing Data Scatter Graph")
    plt.xlabel("Independent Variables")
    plt.ylabel("Dependent Variable (Approved)")
    plt.show() """

def graph():
    plt.title("Accuracy Graph")
    height = [boosting_acc,tree_acc,svm_acc,random_acc,linear_acc,backflow_acc]
    bars = ('XGB Accuracy','GB Decision Tree Acc','SVM Acc','Random Forest Acc','Linear
Acc','Backflow Acc')
    y_pos = np.arange(len(bars))
    width=0.35
    pps=plt.bar(y_pos, height,width,align='center')
    plt.xticks(y_pos, bars)
    for p in pps:
        h=p.get_height()
        plt.text(x=p.get_x()+p.get_width()/2,y=h+.10,s="{:.2f}%".format(h),ha='center')
    plt.show()


font = ('times', 16, 'bold')
title = Label(main, text='Customer Profiling and Credit scoring in Banking using Back Flow
Ensemble Learning')
title.config(bg='mint cream', fg='olive drab')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=0,y=5)

font1 = ('times', 14, 'bold')
uploadButton = Button(main, text="Upload Credit Dataset & Apply SMOTE Algorithm",
command=upload)
uploadButton.place(x=50,y=100)
uploadButton.config(font=font1)

pathlabel = Label(main)
pathlabel.config(bg='mint cream', fg='olive drab')
pathlabel.config(font=font1)
pathlabel.place(x=500,y=100)

boostingButton = Button(main, text="Run Extreme Gradient Boosting", command=boosting)
boostingButton.place(x=50,y=150)
boostingButton.config(font=font1)

treeButton = Button(main, text="Run Gradient Boosting Decision Tree", command=tree)
treeButton.place(x=380,y=150)
treeButton.config(font=font1)

svmButton = Button(main, text="Run SVM", command=SVM)
svmButton.place(x=750,y=150)
svmButton.config(font=font1)

randomButton = Button(main, text="Run Random Forest", command=randomForest)
randomButton.place(x=50,y=200)
randomButton.config(font=font1)

linearButton = Button(main, text="Run Linear Discriminant Analysis", command=linear)
linearButton.place(x=280,y=200)
linearButton.config(font=font1)

backButton = Button(main, text="Run Backflow Learning", command=backFlow)
backButton.place(x=610,y=200)
backButton.config(font=font1)

graphButton = Button(main, text="Accuracy Graph", command=graph)
graphButton.place(x=880,y=200)
graphButton.config(font=font1)

font1 = ('times', 12, 'bold')
textarea=Text(main,height=20,width=150)
scroll=Scrollbar(textarea)
textarea.configure(yscrollcommand=scroll.set)
textarea.place(x=10,y=250)
textarea.config(font=font1)

main.config(bg='gainsboro')
main.mainloop()
```

5.RESULTS

5. RESULTS

5.1 SCREENSHOTS

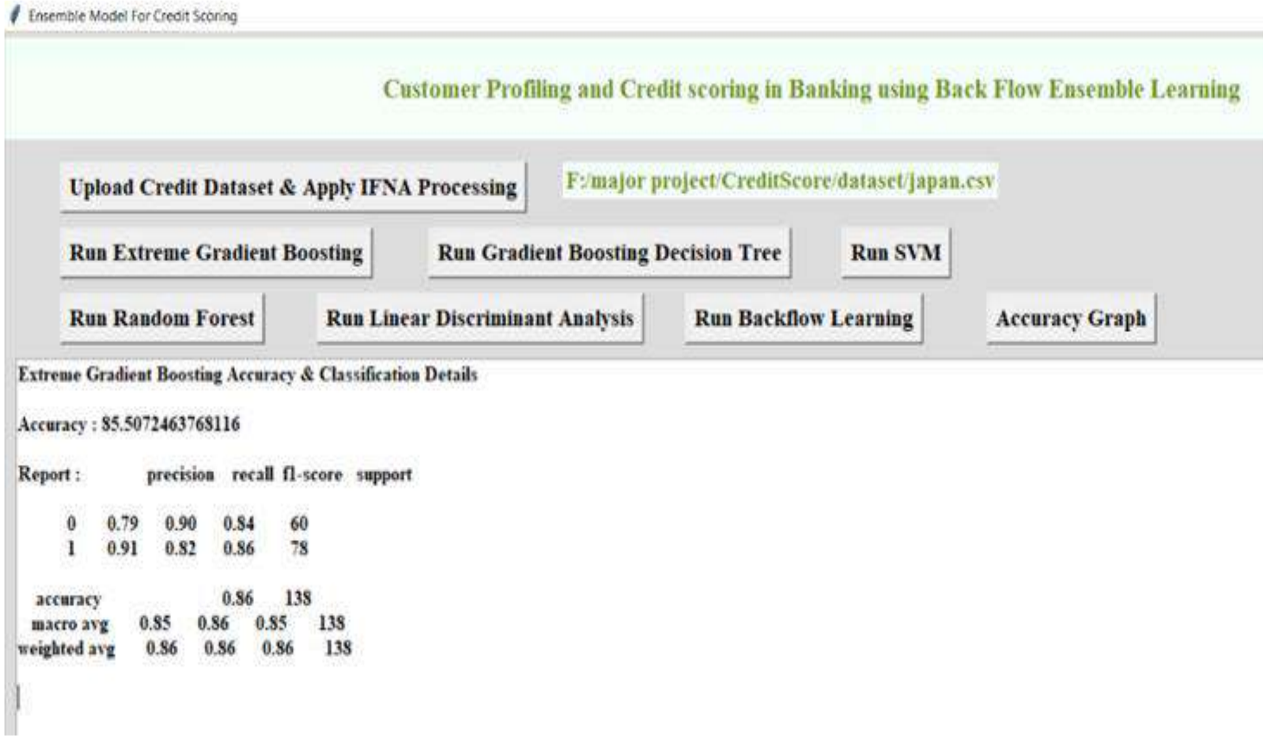
5.1.1 Before and After applying SMOTE Algorithm


Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
F:/major project/CreditScore/dataset/japan.csv
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

```
Credit Score Train & Test Model Generated

Total Dataset Size : 690
Splitted Training Size : 552
Splitted Test Size : 138

Before Outlier Detection, counts of label '1': 305
Before Outlier Detection, counts of label '0': 247

After SMOTE Outlier Detection, counts of label '1': 305
After SMOTE Outlier Detection, counts of label '0': 305
```

Screenshot 5.1.1 Before and After Applying SMOTE Algorithm

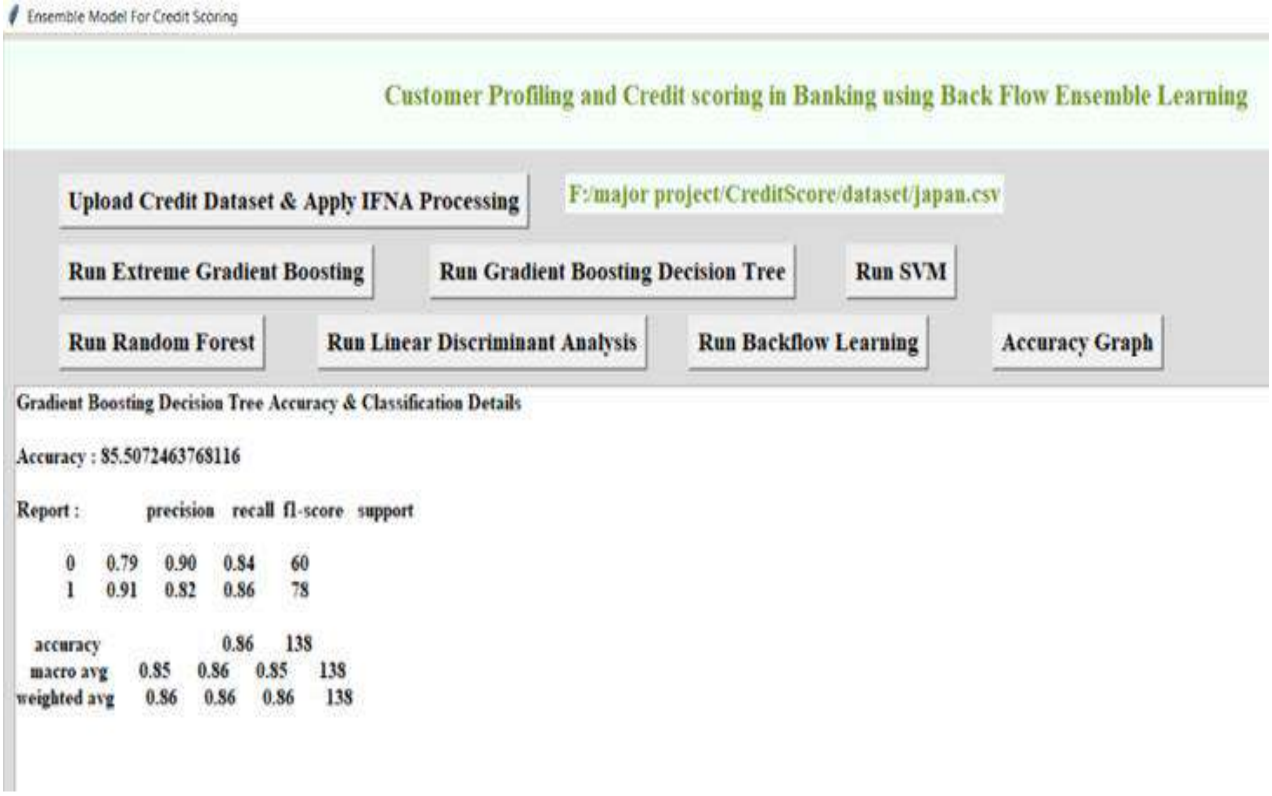
5.1.2 Extreme Gradient Boosting Prediction


Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
F:/major project/CreditScore/dataset/japan.csv
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

```
Extreme Gradient Boosting Accuracy & Classification Details

Accuracy : 85.5072463768116

Report :          precision    recall  f1-score   support

       0     0.79     0.90     0.84      60
       1     0.91     0.82     0.86      78

   accuracy                     0.86     138
  macro avg     0.85     0.86     0.85     138
weighted avg     0.86     0.86     0.86     138
```

Screenshot 5.1.2 Extreme Gradient Boosting Prediction

5.1.3 Gradient Boosting Decision Tree Prediction

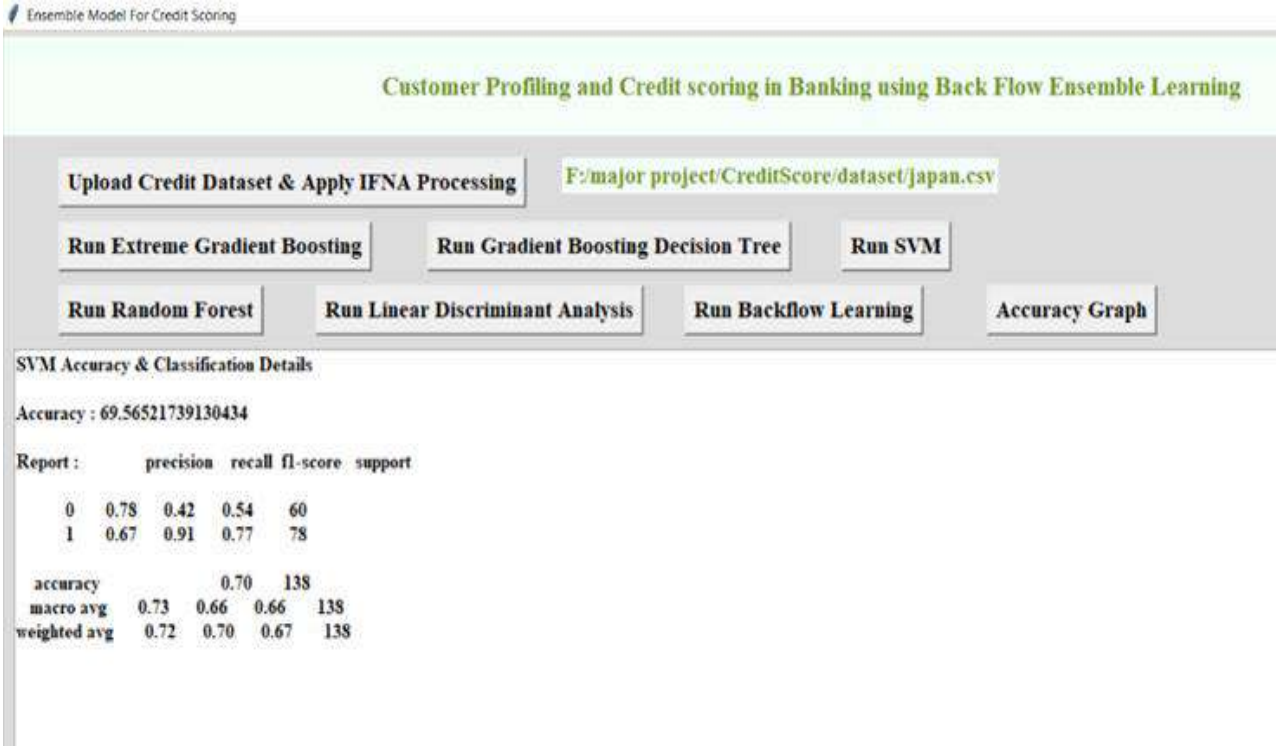

Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

F:/major project/CreditScore/dataset/japan.csv

```
Gradient Boosting Decision Tree Accuracy & Classification Details

Accuracy : 85.5072463768116

Report :          precision   recall  f1-score   support

       0      0.79     0.90     0.84      60
       1      0.91     0.82     0.86      78

    accuracy                    0.86     138
   macro avg      0.85     0.86     0.85     138
weighted avg      0.86     0.86     0.86     138
```

Screenshot 5.1.3 Gradient Boosting Decision Tree Prediction

5.1.4 Support Vector Machine Prediction

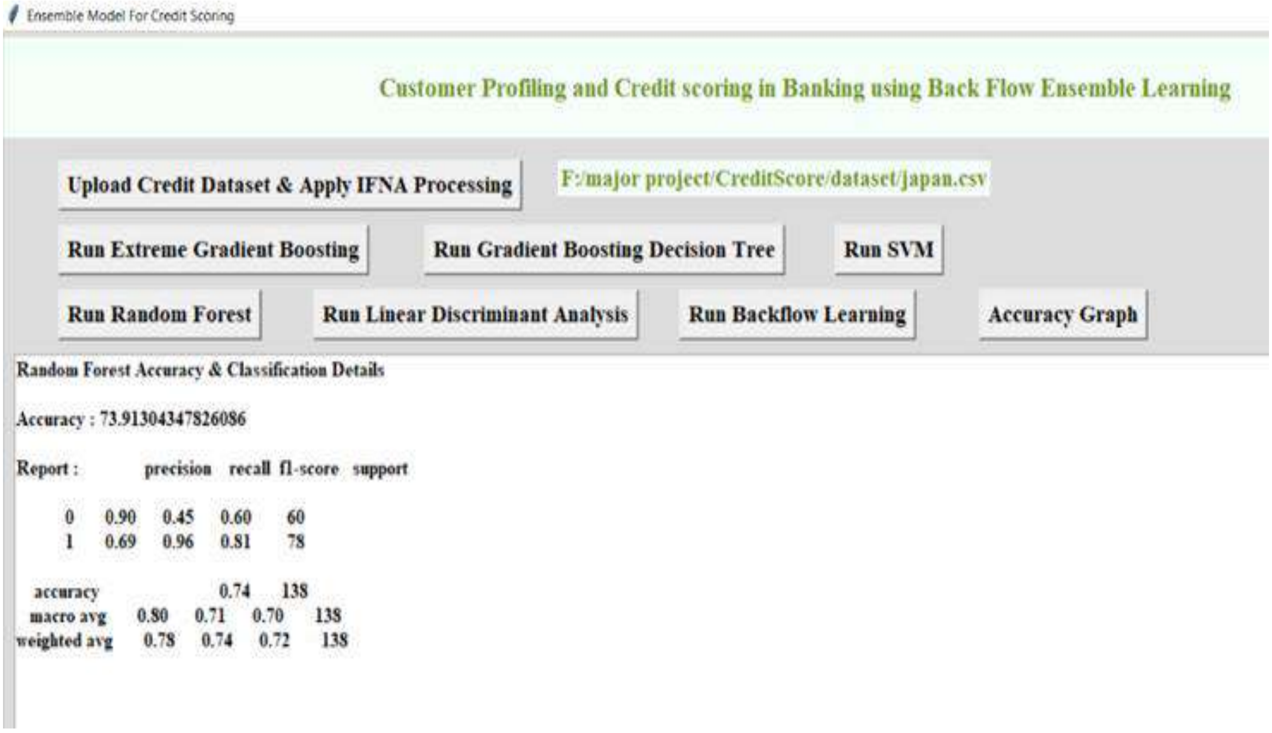

Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

F:/major project/CreditScore/dataset/japan.csv

```
SVM Accuracy & Classification Details

Accuracy : 69.56521739130434

Report :          precision   recall  f1-score   support

       0      0.78     0.42     0.54      60
       1      0.67     0.91     0.77      78

    accuracy                    0.70     138
   macro avg      0.73     0.66     0.66     138
weighted avg      0.72     0.70     0.67     138
```

Screenshot 5.1.4 Support Vector Machine Prediction

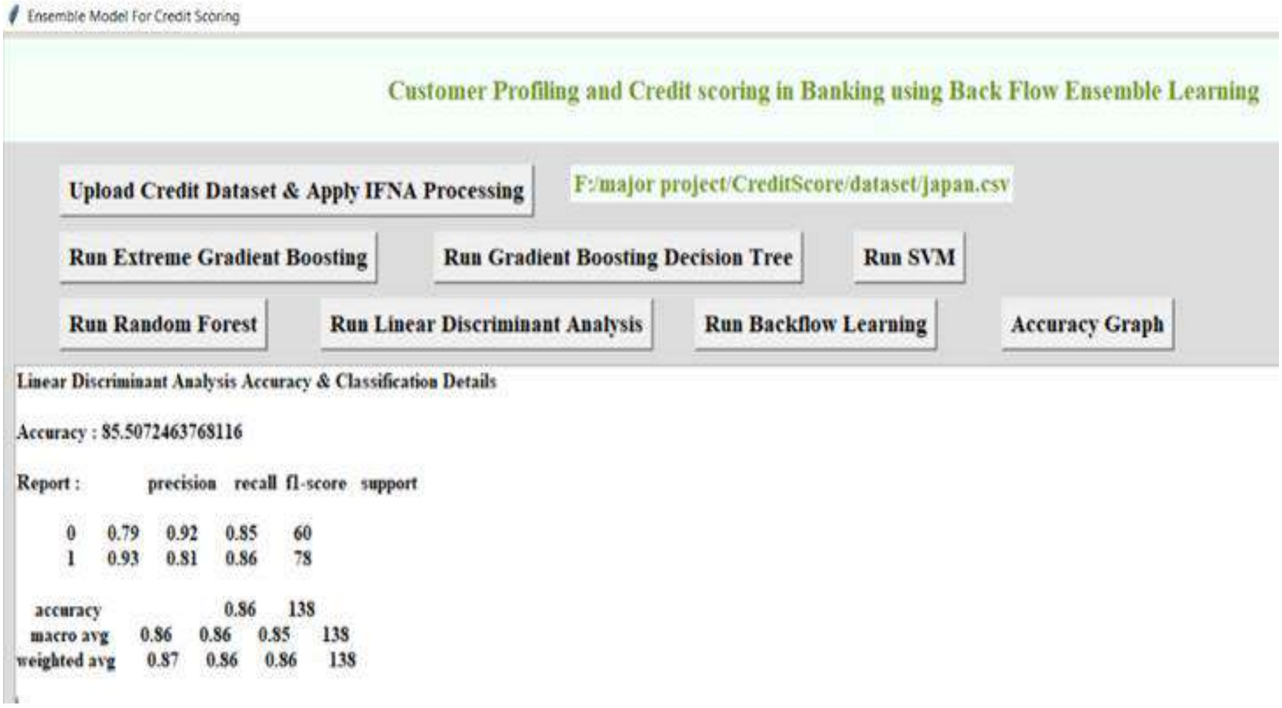
5.1.5 Random Forest Classification


Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
F:/major project/CreditScore/dataset/japan.csv
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

```
Random Forest Accuracy & Classification Details

Accuracy : 73.91304347826086

Report :          precision    recall  f1-score   support

           0       0.90      0.45      0.60        60
           1       0.69      0.96      0.81        78

    accuracy                           0.74       138
   macro avg       0.80      0.71      0.70       138
weighted avg       0.78      0.74      0.72       138
```

Screenshot 5.1.5 Random Forest Classification

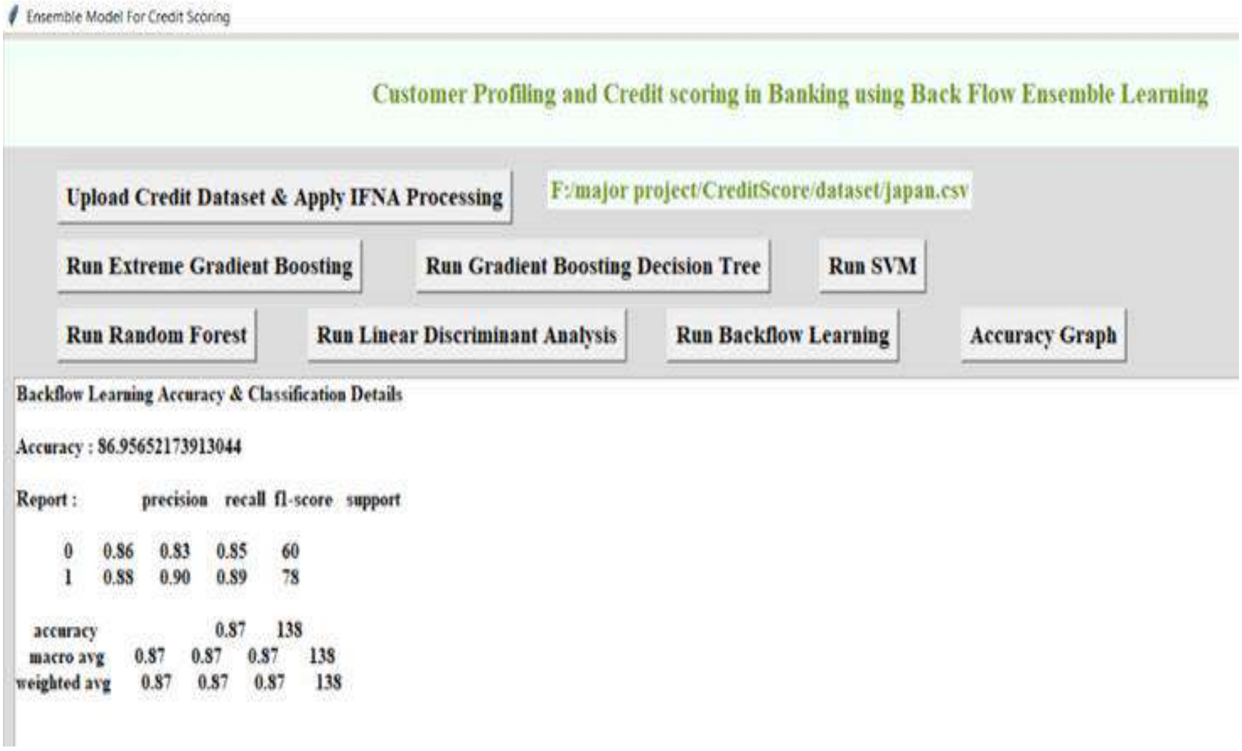
5.1.6 Linear Discriminant Analysis


Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
F:/major project/CreditScore/dataset/japan.csv
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

```
Linear Discriminant Analysis Accuracy & Classification Details

Accuracy : 85.5072463768116

Report :          precision    recall  f1-score   support

           0       0.79      0.92      0.85        60
           1       0.93      0.81      0.86        78

    accuracy                           0.86       138
   macro avg       0.86      0.86      0.85       138
weighted avg       0.87      0.86      0.86       138
```

Screenshot 5.1.6 Linear Discriminant Analysis

5.1.7 Backflow Ensemble Learning Prediction

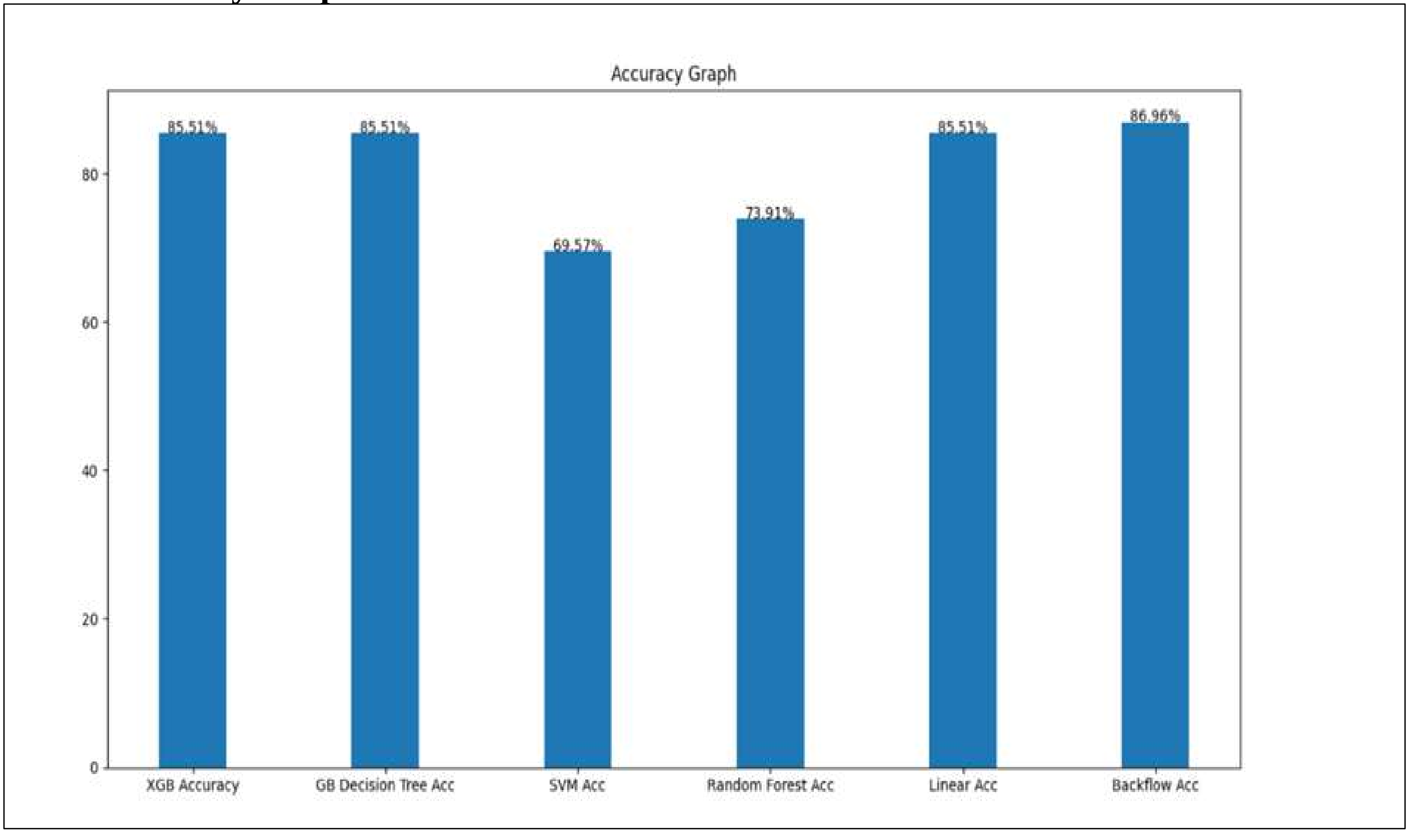

Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
F:/major project/CreditScore/dataset/japan.csv
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

```
Backflow Learning Accuracy & Classification Details

Accuracy : 86.95652173913044

Report :          precision   recall  f1-score   support

       0      0.86     0.83     0.85      60
       1      0.88     0.90     0.89      78

  accuracy                     0.87     138
  macro avg      0.87     0.87     0.87     138
weighted avg     0.87     0.87     0.87     138
```

Screenshot 5.1.7 Backflow Ensemble Learning Prediction

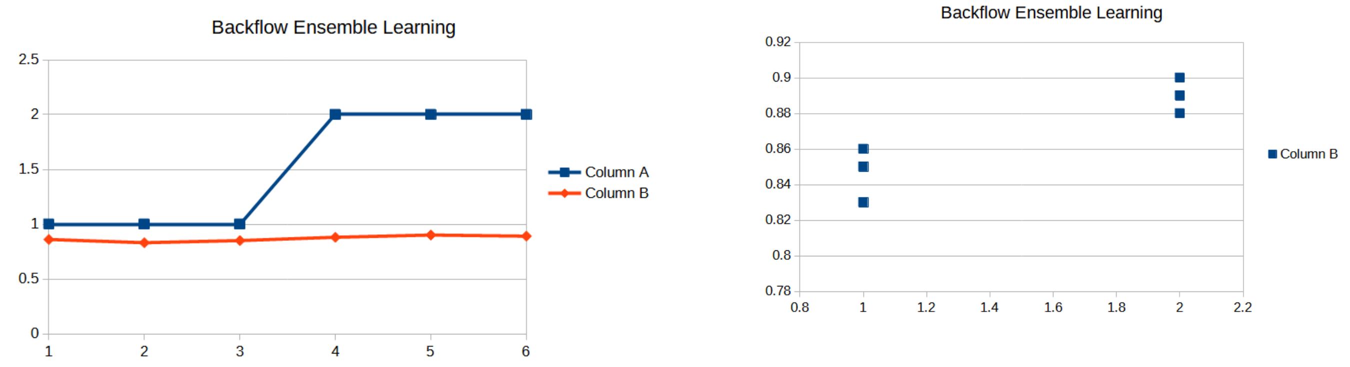
5.2 GRAPHS

5.2.1 Accuracy Graph



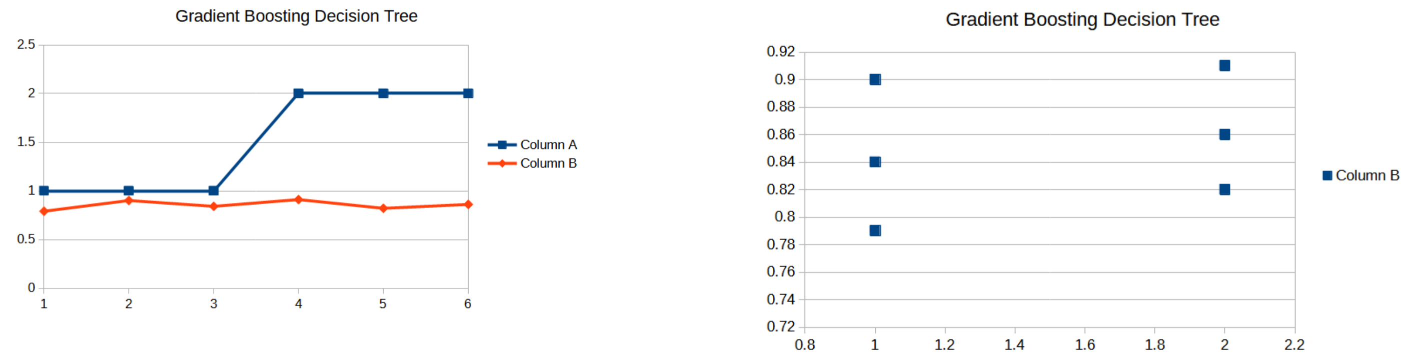
Screenshot 5.2.1 Accuracy Graph

5.2.2 Backflow Ensemble Learning Graph



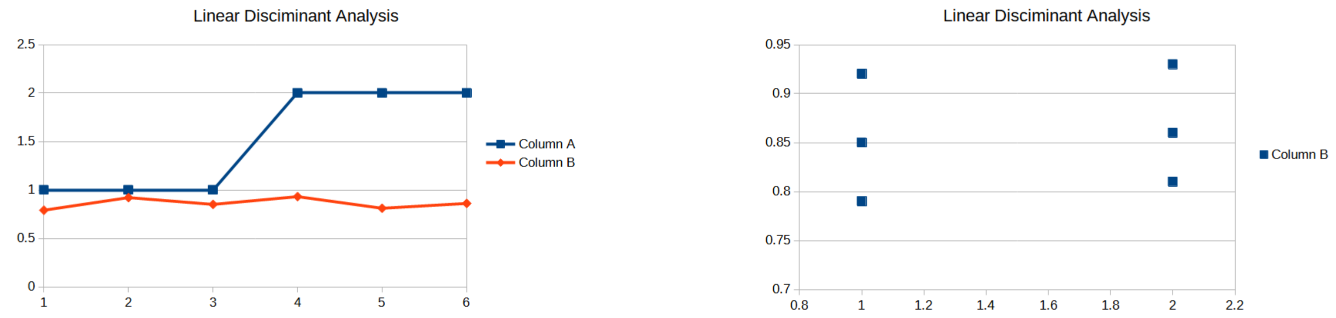
Screenshot 5.2.2 Backflow Ensemble Learning Graph

5.2.3 Gradient Boosting Decision Tree Graph



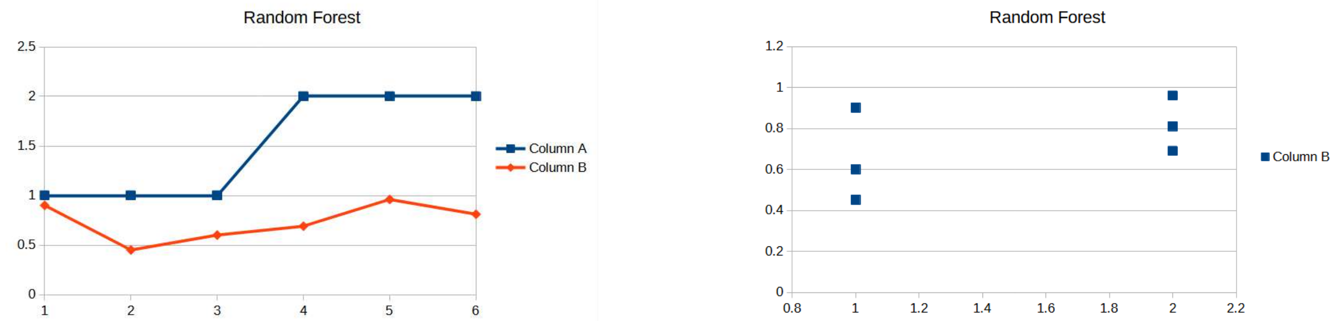
Screenshot 5.2.3 Gradient Boosting Decision Tree Graph

5.2.4 Linear Discriminant Analysis Graph



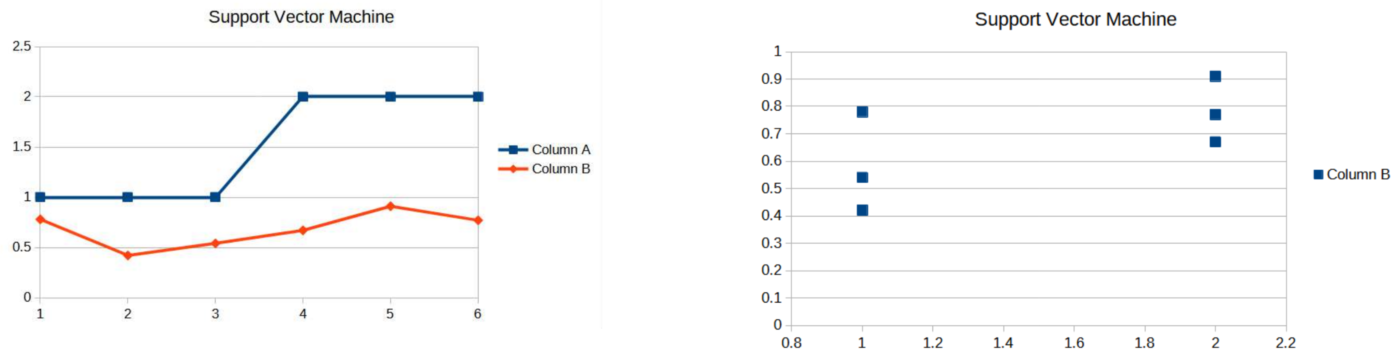
Screenshot 5.2.4 Linear Discriminant Analysis Graph

5.2.5 Random Forest Classification Graph



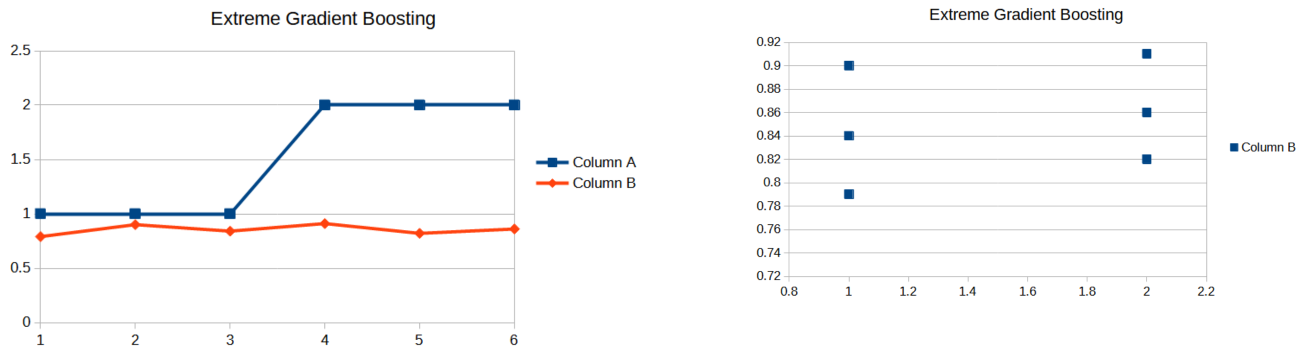
Screenshot 5.2.5 Random Forest Classification Graph

5.2.6 Support Vector Machine Graph



Screenshot 5.2.6 Support Vector Machine Graph

5.2.7 Extreme Gradient Boosting Graph



Screenshot 5.2.7 Extreme Gradient Boosting Graph

6.TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at thecomponent level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event-driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfied, as shown by successful unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted. **Invalid**

Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identifying Business process flows; data fields, predefined processes.

6.3 TEST CASES

6.3.1 PROVIDING INPUT DATA

Test case ID	Test case name	Purpose	Test Case	Output
1	User gives training data	Use it for prediction	The user gives training input data	The test input data provided successfully
2	User gives test data	Use it for prediction	The user gives test input data	The test input data provided successfully

Table 6.1: Providing Input Data

6.3.2 PREDICTION

Test case ID	Test case name	Purpose	Input	Output
1	Prediction test 1	To check if the trained model performs its task	An input data of Customer Details is given	Customer Satisfied.
2	Prediction test 2	To check if the trained model performs its task	An input data of Customer Details is given	Customer Unsatisfied.

Table 6.2: Prediction for the Inputs

7.CONCLUSION

7. CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

The project titled “**Customer Profiling and Credit Scoring in Banking using Back Flow Ensemble Learning**” is a console-based application. Profiling can allow the banks to build an interactive relationship based on humanistic experience and trust. The careful analysis of the profiling environment should be made to ensure effective and efficient segmenting of the bank’s customer pool to help design its service and product offering to win customer loyalty and satisfaction. So that any bank in the future can use this model and technique to improve profiling of its customer, get high profitability, and reduce the risk.

7.2 FUTURE SCOPE

It is an important information management task of commercial banks and loan lenders. Profiling produces customer profiles, which provide the banks with a full description of their customers based on a set of attributes. A Banking system includes a large dataset for transactions of customers of their credit cards. Hence, banks would be in need of customer profiling. Profiling bank customers cognize the issuer's decisions about whom to give banking facilities and what a credit limit to provide. It also helps the issuers get a better understanding of their potential and current customers. In our proposed model we are focusing on customers satisfaction, whether the particular customer is satisfied or not. Further, we can focus on dividing the customers into different groups such as platinum, gold and silver based on their deposits, debt, credit limit.

8.BIBLIOGRAPHY

8. BIBLIOGRAPHY

8.1 REFERENCES

[1] Data Mining: The Textbook 2015 Edition, Kindle Edition by Charu C. Aggarwal.

[2] Data Mining: Concepts and Techniques by Jiawei Han, Jian Pei, Micheline Kamber.

8.2 WEBSITES

[1] Research paper on Credit Scoring in Banking by using Ensemble model at: "https://ieeexplore.ieee.org/document /8768372 ".

[2] Brief about SMOTE algorithm at: 'https://towardsdatascience.com/smote-fdce2f605729".

[3] Understanding Support Vector Machine at: "https://www.analyticsvidhya.com/blog/2017/09/understanding-support-vector-machine-example-code/".

[4] About Linear Discriminant Analysis at: "https://www.geeksforgeeks.org/ml-linear-discriminant-analysis/".

[5] Introduction to Gradient Boosting in Machine Learning at: "https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/".

[6] Decision Tree classifier in machine Learning at: "https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm".

[7] Understanding Random Forest at: "https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/".

[8] Guide to Ensemble Learning at: "https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/ ".

[9] Ensemble Methods in Machine Learning at: "https://towardsdatascience.com/ensemble-methods-in-machine-learning-what-are-they-and-why-use-them-68ec3f9fef5f ".

[10] Overcoming Class Imbalance using SMOTE at: "https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/ ".

8.3 GITHUB LINK

[1] https://github.com/saidinesh7/credit-scoring

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:06/June-2022 Impact Factor- 6.752 www.irjmets.com

CUSTOMER PROFILING AND CREDIT SCORING IN BANKING USING BACK-FLOW ENSEMBLE LEARNING

Latha Gadepaka[*1], **Sai Rakshitha Kulkarni**[*2], **Sai Dinesh Puppala**[*3], **Gadasu Sai Charan**[*4]

[*1]Associate Professor, CMR Technical Campus, Hyderabad, Telangana, India

[*2,3,4]Student, CMR Technical Campus, Hyderabad, Telangana, India.

ABSTRACT

Recently, the machine learning method and artificial intelligence algorithm has become increasingly important in classification problems such as credit scoring [14]. In the modern era of the banking sector, banks have large datasets containing customers' information and their history of transactions. Hence, banks need to classify these large datasets to be able to analyze these customers' behaviours for using it in the best way to suggest a suitable strategy to gain the highest benefits, customer satisfaction to increase profitability. To achieve this purpose, we are Building an ensemble learning model [3][11][12] that has been proven to be typically more accurate and robust than individual classifiers [6][7][8][9][10], It is an important information management task of commercial banks and loan lenders. Profiling produces customer profiles, which provide the banks with a full description of their customers based on a set of attributes. A Banking system includes a large dataset for transactions of customers of their credit cards. Hence, banks would be in need of customer profiling. Profiling bank customers cognize the issuer's decisions about whom to give banking facilities and what a credit limit to provide [14]. It also helps the issuers get a better understanding of their potential and current customers.

Keywords: Customer Profiling, SMOTE, SVM, LDA, Random Forest, Gradient Boosting, Back-Flow Algorithm, Ensemble Learning.

I. INTRODUCTION

With the rapid development of the loan lending market, a decision-making strategy based merely on human experience is no longer practicable. The probability of default (PD), which measures the risk that customers will be unable to repay their debts, has drawn most research attention among all risk management tasks. Loan lenders must decide carefully to avoid loss and maximize profit. Customers who are believed to repay their loans are defined as "good" customers, while those who are unwilling or unable to repay are defined as "bad". Classifying customers into the wrong category will incur two different types of loss. If an actually "good" customer is classified as "bad", the loan lenders will bear the corresponding opportunity cost. On the contrary, if they lend funds to an actual "bad" customer, the loan will be non-performing and the loss will be immeasurable. Building an effective PD model is critical for financial institutions to maximize profit and survive in fierce competitions. Machine learning methods have been widely used in credit scoring [14]. Machine learning algorithms such as support vector machine (SVM) [6], ANN, decision tree (DT) have been used to solve such classification problems.

II. METHODOLOGY

In this project, we have taken a dataset which consists of customer personal details like name, employment, etc. and the credit score details [14]. The main goal is to predict if a particular customer is satisfied with the bank schemes or not. Our proposed model is an ensemble model which fuses the output of five base classifiers which we have used [3][11][12]. As base classifiers we are using SVM, Gradient Boosting Decision Tree Random Forest, Gradient Boosting and LDA (linear Deterministic Analysis) [6][7][8][9][10]. The output of these base classifiers is given to a voting classifier which gives us the final predictive result. It will be in the form of 1 or 0 the customer satisfied or not. We have also used backflow algorithm so that any misclassified data can be relearned. We have compared the accuracies of five individual classifiers and the ensemble model. The classifier with least accuracy is SVM and ensemble model had the highest accuracy [6][7][8][9][10][3][11][12]. Customer Profiling helps banks get to know their customers on a more granular level. This approach can give banks better understanding of their clients, allowing them to align tactics and strategies to each customer type for better sales, loyalty, growth, and cost containment.

III. MODELING AND ANALYSIS

In this project, we are using credit scoring dataset [14]. As it will be unstructured data at first we are performing data pre-processing on the dataset first then after the splitting the dataset into training and testing data [1][2][4]. We are then fitting the training data into the individual classifiers. We are evaluating performance of various machine learning algorithms as base classifiers and later these base classifiers output will be fused and the voting classifier will give the final predictive result in the form of 1 or 0 customer satisfied or not. Voting classifier will choose best performing base classifier. Here credit scoring dataset may contain outlier/class imbalance and to remove outlier we are applying SMOTE algorithm [5][13][14]. As base classifiers we are using SVM, Gradient Boosting Decision Tree Random Forest, Gradient Boosting and LDA (linear Deterministic Analysis) [6][7][8][9][10].

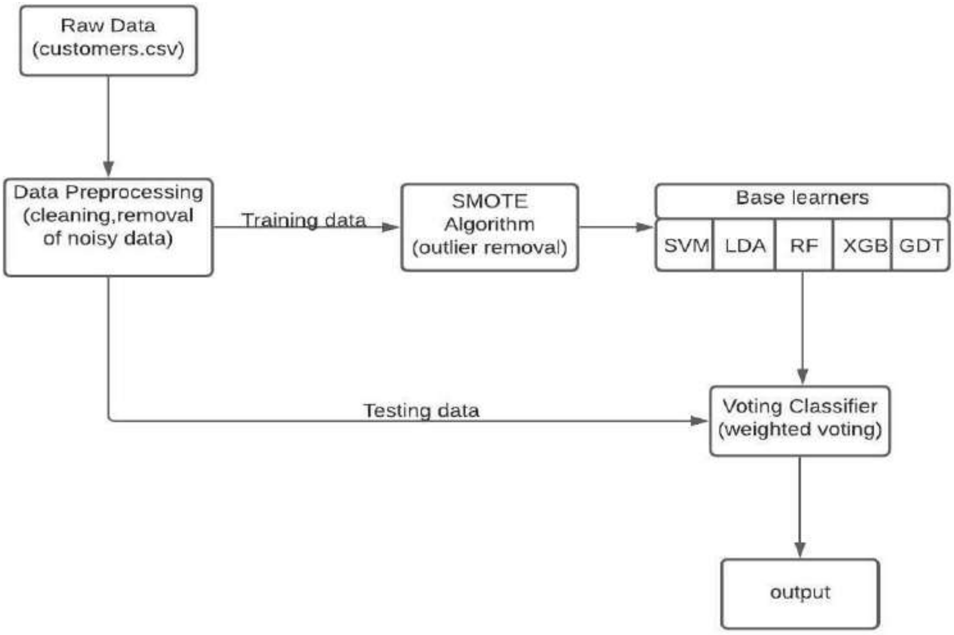
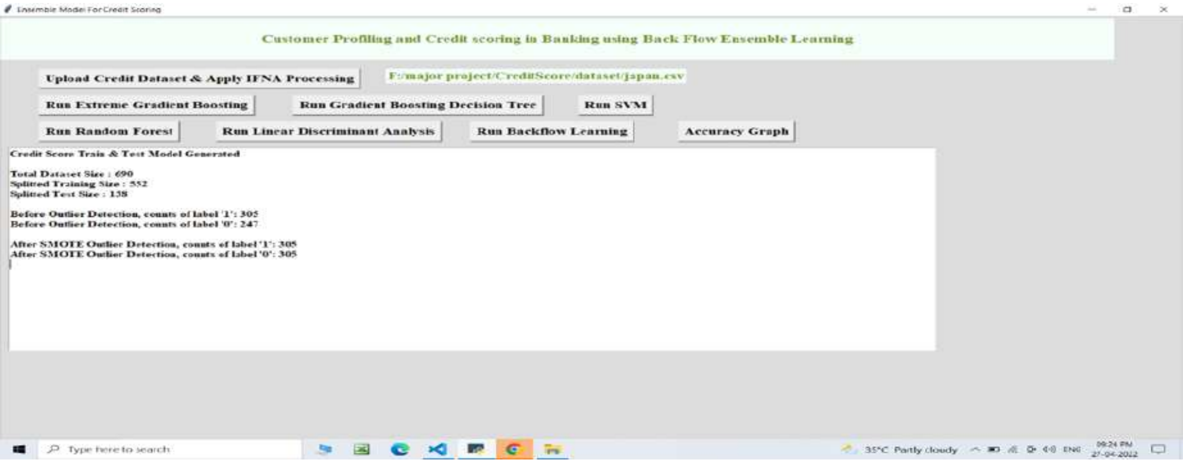


Figure 3.1: Architecture of the Model.

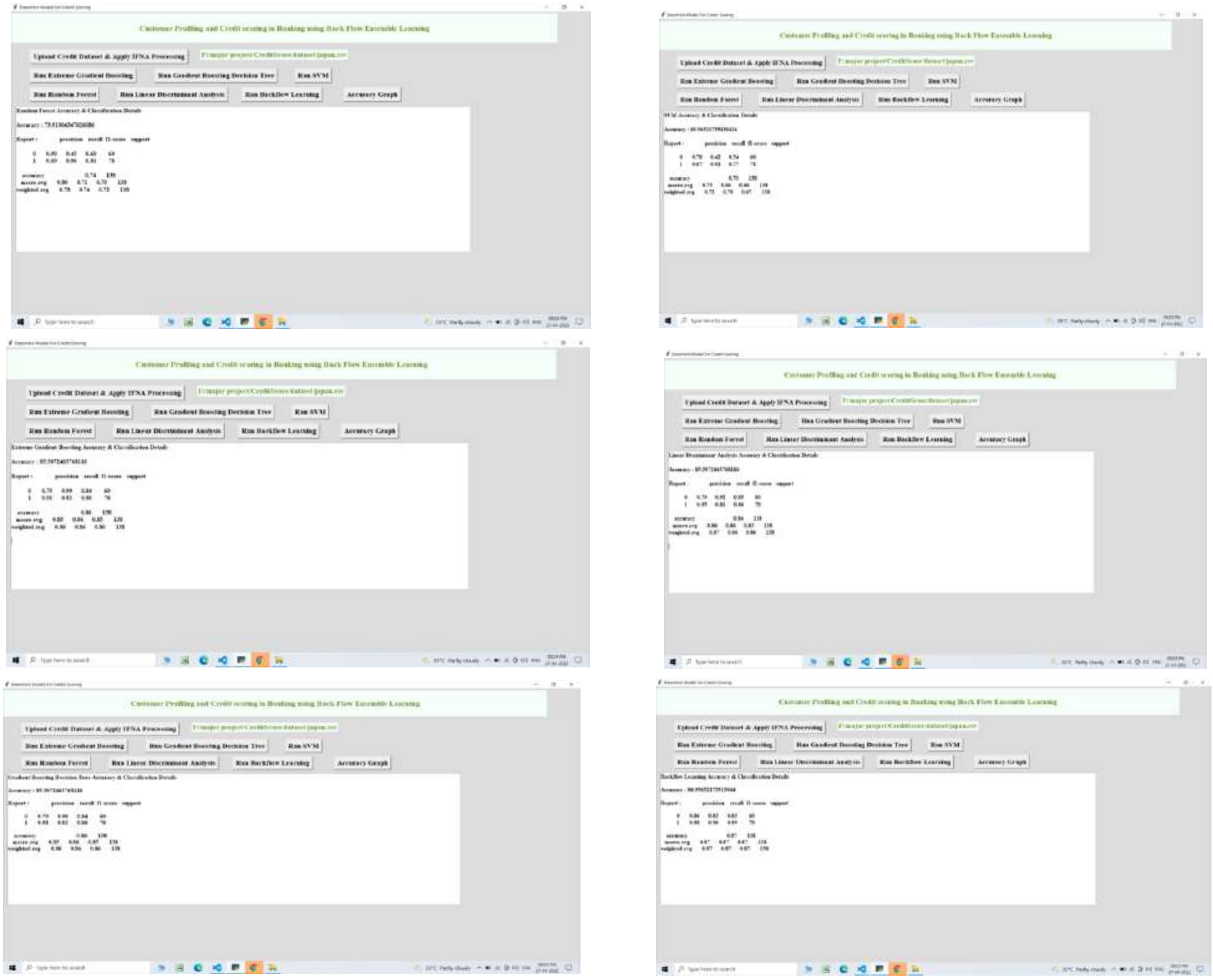
IV. RESULTS AND DISCUSSION

We have uploaded the dataset first and we can see the results before and after applying the smote algorithm [5][13]. Then we have taken output of individual classifiers. Each classifier output can be seen on the screen. Then we have performed the ensemble model which is giving the highest accuracy than any other classifier [6][7][8][9][10]. We have plotted an accuracy graph to compare different accuracies [3][11][12]. We have also plotted a line graph for training and testing data for all individual classifiers and also the ensemble learning model. These results can be observed in the below result screenshots.


Ensemble Model For Credit Scoring
Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
F:/major project/CreditScore/dataset/japan.csv
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

```
Credit Score Train & Test Model Generated

Total Dataset Size : 690
Splitted Training Size : 552
Splitted Test Size : 138

Before Outlier Detection, counts of label '1': 305
Before Outlier Detection, counts of label '0': 247

After SMOTE Outlier Detection, counts of label '1': 305
After SMOTE Outlier Detection, counts of label '0': 305
```

Figure 4.1: Before and After applying SMOTE Algorithm


Customer Profiling and Credit scoring in Banking using Back Flow Ensemble Learning
Upload Credit Dataset & Apply IFNA Processing
Run Extreme Gradient Boosting
Run Gradient Boosting Decision Tree
Run SVM
Run Random Forest
Run Linear Discriminant Analysis
Run Backflow Learning
Accuracy Graph

Random Forest Accuracy & Classification Details

Accuracy : 73.91304347826086

```
Report :        precision    recall  f1-score   support
[illegible]
```

SVM Accuracy & Classification Details

Accuracy : [illegible]

```
Report :        precision    recall  f1-score   support
[illegible]
```

Extreme Gradient Boosting Accuracy & Classification Details

Accuracy : [illegible]

```
Report :        precision    recall  f1-score   support
[illegible]
```

Linear Discriminant Analysis Accuracy & Classification Details

Accuracy : [illegible]

```
Report :        precision    recall  f1-score   support
[illegible]
```

Gradient Boosting Decision Tree Accuracy & Classification Details

Accuracy : [illegible]

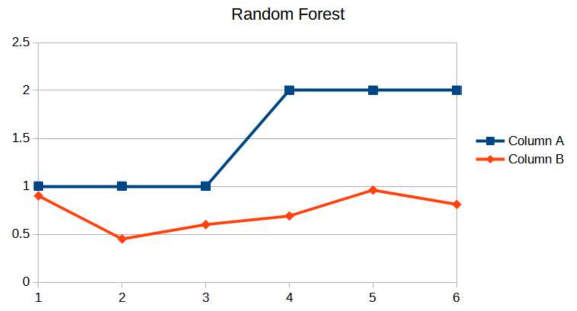
```
Report :        precision    recall  f1-score   support
[illegible]
```

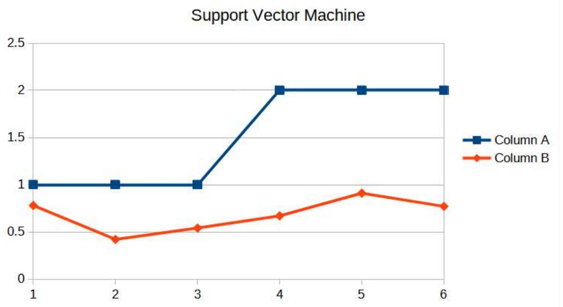
Backflow Learning Accuracy & Classification Details

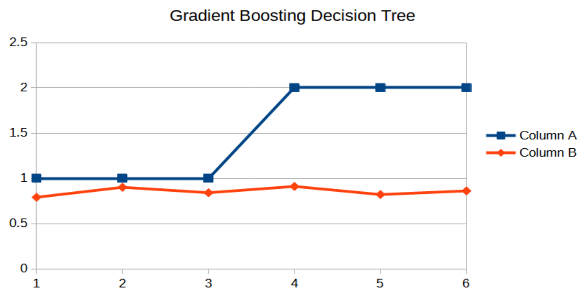
Accuracy : [illegible]

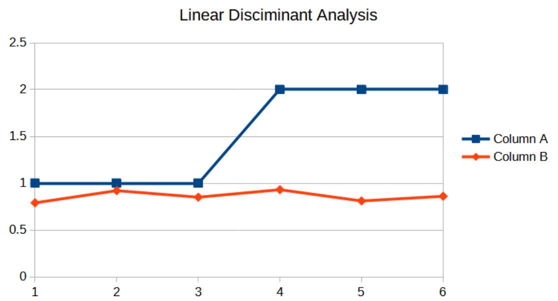
```
Report :        precision    recall  f1-score   support
[illegible]
```

Figure 4.2 to 4.7: GUI and Results of different algorithms after applying to dataset









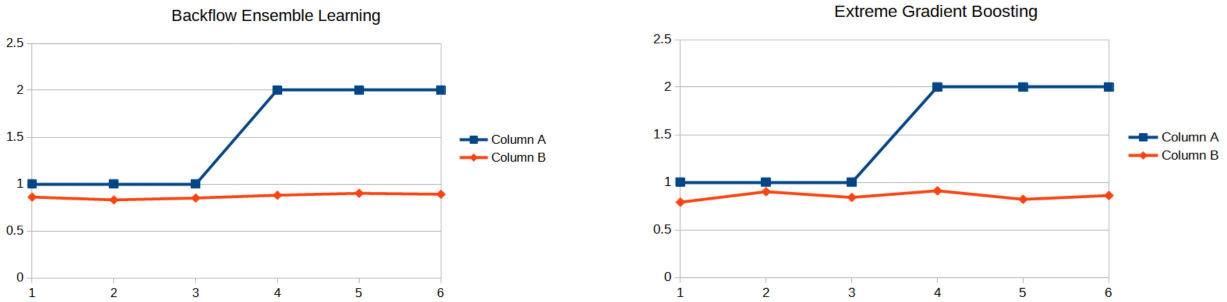
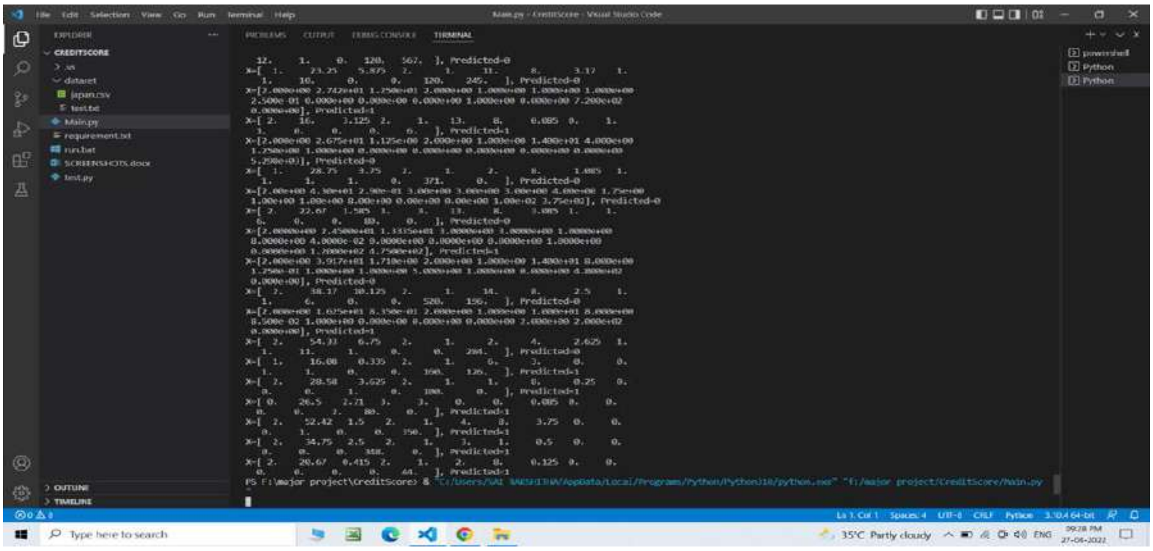


Figure 4.8 to 4.9: Results of individual Graph



```
  12.     1.     0.   120.   567.  ], Predicted=0
X=[  1.     23.25   5.875  2.     1.     11.     8.     3.17   1.
   1.     10.     0.     0.    120.   245.   ], Predicted=0
X=[2.000e+00 2.742e+01 1.250e+01 2.000e+00 1.000e+00 1.000e+00 1.000e+00
 2.500e-01 0.000e+00 0.000e+00 0.000e+00 1.000e+00 0.000e+00 7.200e+02
 0.000e+00], Predicted=1
X=[ 2.    16.     3.125  2.     1.    13.     8.     0.085  0.     1.
  1.     0.     0.     0.     6.   ], Predicted=1
X=[2.000e+00 2.675e+01 1.125e+00 2.000e+00 1.000e+00 1.400e+01 4.000e+00
 1.250e+00 1.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
 5.250e+01], Predicted=0
X=[  1.     28.75    3.75   2.     1.     2.     8.     1.085   1.
   1.      1.     1.     0.    371.     0.   ], Predicted=0
X=[2.00e+00 4.30e+01 2.90e-01 3.00e+00 3.00e+00 3.00e+00 4.00e+00 1.75e+00
 1.00e+00 1.00e+00 8.00e+00 0.00e+00 0.00e+00 1.00e+02 3.75e+02], Predicted=0
X=[ 2.    22.67   1.585  1.     4.    13.     8.     3.085  1.     1.
  6.     0.     0.    80.     0.   ], Predicted=0
X=[2.0000e+00 7.4500e+01 1.3335e+01 3.0000e+00 3.0000e+00 1.0000e+00
 8.0000e+00 4.0000e-02 9.0000e+00 0.0000e+00 0.0000e+00 1.0000e+00
 0.0000e+00 1.2000e+02 4.7500e+02], Predicted=1
X=[2.000e+00 3.917e+01 1.710e+00 2.000e+00 1.000e+00 1.400e+01 8.000e+00
 1.250e-01 1.000e+00 1.000e+00 5.000e+00 1.000e+00 0.000e+00 4.300e+02
 0.000e+00], Predicted=0
X=[  2.     38.17   10.125   2.     1.    14.     8.     2.5    1.
   1.      6.     0.     0.    520.   196.   ], Predicted=0
X=[2.000e+00 1.625e+01 8.350e-01 2.000e+00 1.000e+00 1.000e+01 8.000e+00
 8.500e-02 1.000e+00 0.000e+00 0.000e+00 0.000e+00 2.000e+00 2.000e+02
 0.000e+00], Predicted=1
X=[  2.     54.33    6.75   2.     1.     2.     4.     2.625   1.
   1.     11.     1.     0.     0.    284.   ], Predicted=0
X=[  1.     16.08    0.335   2.     1.     6.     3.     0.     0.
   1.      1.     0.     0.    160.   126.   ], Predicted=1
X=[  2.     28.58    3.625   2.     1.     1.     8.     0.25   0.
   0.      0.     1.     0.    100.     0.   ], Predicted=1
X=[ 0.    26.5    2.71   3.     3.     0.     0.     0.085  0.     0.
  0.     0.     2.    80.     0.   ], Predicted=1
X=[  2.    52.42   1.5    2.     1.     4.     8.     3.75   0.     0.
  0.     1.     0.     0.    350.  ], Predicted=1
X=[  2.    34.75   2.5    2.     1.     3.     1.     0.5    0.     0.
  0.     0.     0.   388.     0.  ], Predicted=1
X=[ 2.    20.67   0.415  2.     1.     2.     8.     0.125  0.     0.
  0.     0.     0.     0.    44.  ], Predicted=1
PS F:\major project\CreditScore> & "C:/Users/SAI [illegible]/AppData/Local/Programs/Python/Python310/python.exe" "f:/major project/CreditScore/Main.py"
```

Figure 4.10: Customer Segmentation Results

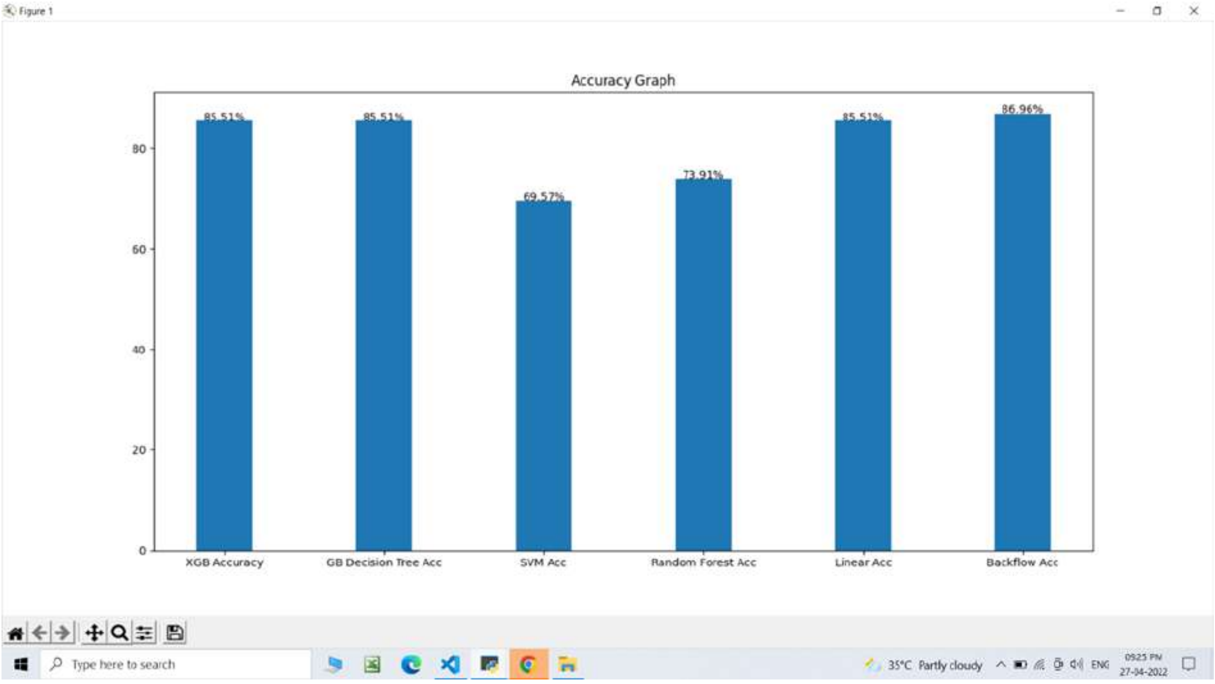


Figure 4.11: Accuracy Graph

V. CONCLUSION

Ensemble learning model is typically more superior to a base classifier. It is more accurate and robust than individual classifiers. This approach can give banks a better understanding of their clients, allowing them to align tactics and strategies to each customer type for better sales, loyalty, growth, and cost containment.

Profiling can allow the banks to build an interactive relationship based on humanistic experience and trust. The careful analysis of the profiling environment should be made to ensure effective and efficient segmenting of the bank's customer pool to help design its service and product offering to win customer loyalty and satisfaction. So that any bank in the future can use this model and technique to improve profiling of its customer, get high profitability, and reduce the risk.

VI. REFERENCES

[1] DATA MINING: THE TEXTBOOK 2015 EDITION, KINDLE EDITION BY CHARU C.AGARAWAL.

[2] Data Mining: Concepts and Techniques by Jiawei Han, Jian Pei, Micheline Kamber.

[3] Research paper on Credit Scoring in Banking by using Ensemble model at: "https://ieeexplore.ieee.org/document /8768372".

[4] Data Preprocessing in Data Mining at: "https://www.analyticsvidhya.com/blog/2021/08/datapreprocessing-in-data-mining-a-hands-on-guide/".

[5] Brief about SMOTE algorithm at: 'https://towardsdatascience.com/smote-fdce2f605729".

[6] Understanding Support Vector Machine at: "https://www.analyticsvidhya.com/blog/2017/09/understanding-support-vector-machine-example-code/".

[7] About Linear Discriminant Analysis at: "https://www.geeksforgeeks.org/ml-linear-discriminantanalysis/".

[8] Introduction to Gradient Boosting in Machine Learning at: "https://machinelearningmastery.com/gentle-introductiongradient-boosting-algorithm-machine-learning/".

[9] Decision Tree classifier in machine Learning at: " https://www.javatpoint.com/machine-learning-decisiontree-classification-algorithm".

[10] Understanding Random Forest at: "https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/".

[11] Guide to Ensemble Learning at: "https://www.analyticsvidhya.com/blog/2021/06/understan ding-random-forest/".

[12] Ensemble Methods in Machine Learning at: "https://towardsdatascience.com/ensemble-methods-in machine-learning-what-are-they-and-why-use-them68ec3f9fef5f".

[13] Overcoming Class Imbalance using SMOTE at: "https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/".

[14] About Credit Scoring at: "https://www.investopedia.com/terms/c/credit_scoring.asp #:~:text=Credit %20scoring%20is%20a%20statistical,to%2 0extend%20or%20deny%20credit. ".

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 4/Issue 06/40600035848 *Date: 08/06/2022*

Certificate of Publication

*This is to certify that author "**Sai Rakshitha Kulkarni**" with paper ID "**IRJMETS40600035848**" has published a paper entitled "CUSTOMER PROFILING AND CREDIT SCORING IN BANKING USING BACK-FLOW ENSEMBLE LEARNING" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 4, Issue 06, June 2022***

Editor in Chief

IRJMETS
Impact Factor
6.752

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future
www.irjmets.com

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 4/Issue 06/40600035848

Date: 08/06/2022

Certificate of Publication

*This is to certify that author "**Sai Dinesh Puppala**" with paper ID "**IRJMETS40600035848**" has published a paper entitled "CUSTOMER PROFILING AND CREDIT SCORING IN BANKING USING BACK-FLOW ENSEMBLE LEARNING"* ***in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 4, Issue 06, June 2022***

Editor in Chief

IRJMETS
Impact Factor
6.752

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future

www.irjmets.com

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 4/Issue 06/40600035848

Date: 08/06/2022

Certificate of Publication

*This is to certify that author "**Gadasu Sai Charan**" with paper ID "**IRJMETS40600035848**" has published a paper entitled "CUSTOMER PROFILING AND CREDIT SCORING IN BANKING USING BACK-FLOW ENSEMBLE LEARNING"* ***in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 4, Issue 06, June 2022***

Editor in Chief

IRJMETS Impact Factor 6.752

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future

www.irjmets.com