PREDICTION OF LOAN APPROVAL USING MACHINE LEARNING ALGORITHM

[1]M.Tech Student, Department of Computer Science Engineering, Jogaiah Institutes of Technology and Sciences, National Highway 214, Kalagampudi, Dist, Palakollu, Andhra Pradesh 534268

[2] Professor,Principal, Department of Computer Science Engineering, Jogaiah Institutes of Technology and Sciences, National Highway 214, Kalagampudi, Dist, Palakollu, Andhra Pradesh 534268

[1]KONDETI DURGA BHAVANI, [2]A.VEERABADHRA RAO

ABSTRACT

Now-a-days obtaining loans from banks have become a very common phenomenon. The banks gain profits from the loans lent to their customers in the form of interest. While approving a loan, the banks should consider many factors such as credit history and score, reputation of the person, the location of the property and the relationship with the bank. Many people apply for loans in the name of home loan, car loan and many more. Everyone cannot be approved based on above mentioned conditions. There are so many cases where applicant’s applications for loans are not approved by various finance companies. The right predictions whether to give a loan to a customer or not is very important for the banks to maximize the profits.The idea behind this project is to use NAIVE BAYES algorithm to predict whether a customer can get a loan from a bank or not.

1.1 INTRODUCTION

Finance companies deal with all kinds of loans such as house loans, vehicle loans, educational loans, personal loans etc... And has a presence across areas such as cities, towns and village areas. A Customer- first requests for a loan and after that Finance Company validates the customer eligibility for the loan and of approve. Details like marital status, gender, education, and number of dependents, Income, Loan Amount, credit history, and others are given in the form to fill up by the applicants. Therefore, a robust model is built taking those details as input to verify whether an applicant is eligible to apply for loan or not. The target variable here is Applicants "Loan Status" and the other variables are predictors. After building the Machine Learning model a Web Application is to be developed for a user interface that allows the user to see instantly if he/she is eligible to get a loan by entering the given details.

1.2 MOTIVATION

Loan prediction is a very common real-life problem that every finance company faces in their lending operations. If the loan approval process is automated, it can save a lot of man hours and improve the speed of service to the customers. The increase in customer satisfaction and savings in operational costs are Significant. However, the benefits can only be reaped if the bank has a robust model to accurately predict which customer's loan it should approve and which to reject, in order to minimize the risk of loan default.

1.3 EXISTING SYSTEM:

Existing system uses machine learning algorithms like decision tree and random forest. Although we can get good efficiency using the Random Forest classifier and Decision Tree classifier, we can get great results by using NAIVE BAYES classifier. The lender must manually review each application, supported the main points provided by the applicant like gender, legal status, education, number of dependents, income, loan amount credit.

1.4 PROPOSED SYSTEM:

The proposed system automates the method of determining the applicant’s loan approval status. It concentrates on a single or multiple datasets containing the details of the loan applicants. In the proposed system, NAIVE BAYES Classification Machine Learning model is been used. NAIVE BAYES algorithm is one in all the supervised learning algorithms, which is dependent on BAYES theorem which is applicable for solving classification problems. It is good for foretelling the right result in the current real-world case scenarios and also help the finance agency to offer the finances in the right hands and also help the applicants in obtaining loan in a much faster way. The key advantage of this model is, it gives more accuracy when compared to the existing system.

1.5 ADVANTAGES:

- The result obtained by the NAIVE BAYES Algorithm is best compared to any other Algorithms.

2. LITERATURE SURVEY

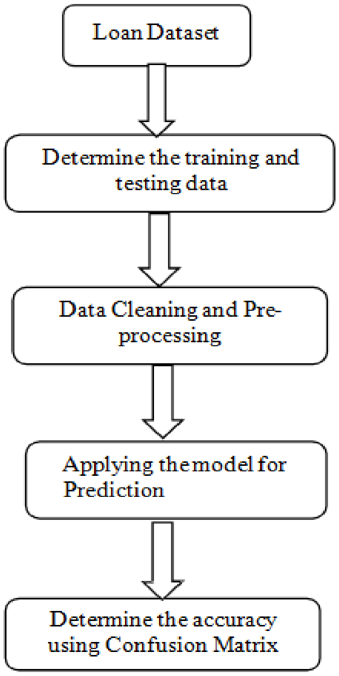
- ***RAJ, J. S., & ANANTHI, J. V***, "Recurrent neural networks and nonlinear prediction in support vector Machine" Journal of Soft Computing Paradigm (JSCP), 1(01), 33-40, 2019.
- ***X.FRENCIS JENSY, V. P. SUMATHIJANANI SHIVA SHRI,*** "An exploratory Data Analysis for Loan Prediction based on nature of clients", International Journal of Recent Technology and Engineering (IJRTE), Volume-7 Issue-4S, November 2018.
- ***SHIYANG LIAO, JUNBO WANG, RUIYUN YU, KOICHI SATO, AND ZIXUE CHENG,*** "CNN for situations understanding based on sentiment analysis of twitter data," PROCEDIA computer science, 111:376-381, 2017.CrossRef.
- ***SHIYANG LIAO, JUNBO WANG, RUIYUN YU, KOICHI SATO, AND ZIXUE CHENG,*** "CNN for situations understanding based on sentiment analysis of twitter.

3.1 PREDICTIVE MODELLING

Predictive modeling is used to analyze the data and predict the outcome. Predictive modeling used to predict the unknown event which may occur in the future. In this process, we are going to create, test and validate the model. There are different methods in predictive modeling. They are learning, artificial intelligence and statistics. Once we create a model, we can use many times, to determine the probability of outcomes. So, predict model is reusable. Historical data is used to train an algorithm. The predictive modeling process is an iterative process and often involves training the model, using multiple models on the same dataset.

- **Creating the model:** To create a model to run one or more algorithms on the data set.
- **Testing a model:** The testing is done on past data to see how the best model predicts
- **Validating a model:** Using visualization tools to validate the model.
- **Evaluating model:** Evaluating the best fit model from the models used and choosing the model right fitted for the data.

3.2 PROCESS MODE USED:


Loan Dataset
Determine the training and testing data
Data Cleaning and Pre-processing
Applying the model for Prediction
Determine the accuracy using Confusion Matrix

Architecture of the proposed loan prediction model

Simple NAIVE BAYES classifier algorithm. Banks are in a very competitive environment; therefore, the service quality during credit risk assessment is very important. When a customer demands credit from a bank, the bank should evaluate the credit demand as soon as possible to gain competitive advantage. Additionally, for each credit demand, the same process is repeated and constitutes a cost for the bank.

- Load the data.
- Determine the training and testing data.
- Data cleaning and preprocessing.
- Fill the missing values with mean values regarding numerical values.
- Fill the missing values with mode values regarding categorical variables.
- Outlier treatment.
- Apply the modeling for prediction.
- Removing the load identifier.
- Create the target variable (based on the

requirement). In this approach,

- ➢ Target variable is loan-status.
- ➢ Create a dummy variable for categorical variable (if required) and split the
- ➢ Training and testing data for validation.
- ➢ Apply the model
- ➢ LR method
- ➢ RF method
- ➢ SVM method
- ➢ Determine the accuracy followed by confusion matrix

4.1 FUNCTIONALREQUIREMENTS

- ✓ Data Collection
- ✓ Data Preprocessing
- ✓ Training and Testing
- ✓ Modeling
- ✓ Predicting

4. 2 NON-FUNCTIONAL REQUIREMENTS

- ✓ Usability requirement
- ✓ Serviceability requirement
- ✓ Manageability requirement
- ✓ Recoverability requirement
- ✓ Security requirement
- ✓ Data Integrity requirement
- ✓ Capacity requirement
- ✓ Availability requirement
- ✓ Scalability requirement

4.3 ADVANTAGES OF NON-FUNCTIONAL REQUIREMENT

- The nonfunctional requirements ensure the software system follow legal and compliance rules.
- They ensure the reliability, availability, and performance of the software system
- They ensure good user experience and ease of operating the software.
- They help in formulating security policy of the software system.

4.4 DISADVANTAGES OF NON-FUNCTIONAL REQUIREMENT

- None functional requirement may affect the various high-level software subsystem
- They require special consideration during the software architecture/high-level design phase which increases costs.
- Their implementation does not usually map to the specific software sub- system,
- It is tough to modify non-functional once you pass the architecture phase.

4.5 KEYLEARNING

The character of the time period, the length of road, the weather, the bus speed and the rate of road usage are adopted as input vectors in Support Vector Machine.

4.6 SYSTEM REQUIREMENTS

OS: Windows or Linux

PYTHON IDE: python 2.7.x and above

PYCHARM IDE Required,

JUPYTER NOTEBOOK

Setup tools and pip to be installed for 3.6 and above

LANGUAGE: Python Scripting

4.7 HARDWARE REQUIREMENTS

RAM: 4GB and Higher

PROCESSOR: Intel i3 and above

HARD DISK: 500GB: Minimum

5.1 DESIGN PHASE

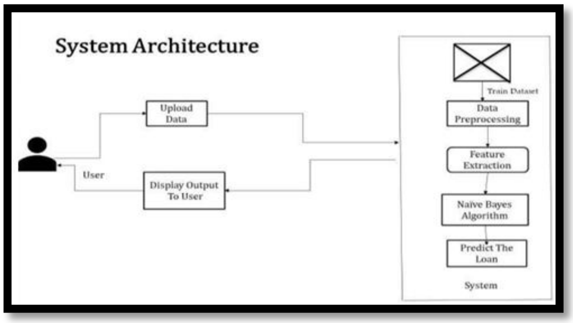
Design is a multi step process that focuses on data structure, Software architecture, procedural details and interface between modules. The design process also translates the requirements into the presentation of software that can be accessed for quality before coding begins. Computer software design changes continuously as new methods; better analysis and broader understanding evolved. Software design at a relatively early stage in its revolution. Therefore, software design methodology lacks the depth, flexibility and quantitative nature that are normally associated with more classical engineering disciplines. However, the techniques for software design do exist, criteria for design qualities are available and design notation can be applied. The purpose of the design phase is to plan a solution of the problem specified by the requirements document. The design of a system is perhaps the most critical factor affecting the quality of the software. It has a major impact on the project during later phases, particularly during testing and maintenance.

5.2 CONCEPTUAL DESIGN:

Conceptual Design is an early phase of the design process, in which the broad outlines of function and form of something are articulated. It includes the design of interactions, experiences, processes and strategies. It involves an understanding of people's needs - and how to meet them with products, services, & processes. The unified modeling language allows the software engineer to express an analysis model using the modeling notation that is governed by a set of syntactic, semantic and pragmatic rules. A UML system is represented using five different views that describe the system from a distinctly different perspective. Each view can be defined by a set of diagrams.

UML is specifically constructed through two different domains. They are:

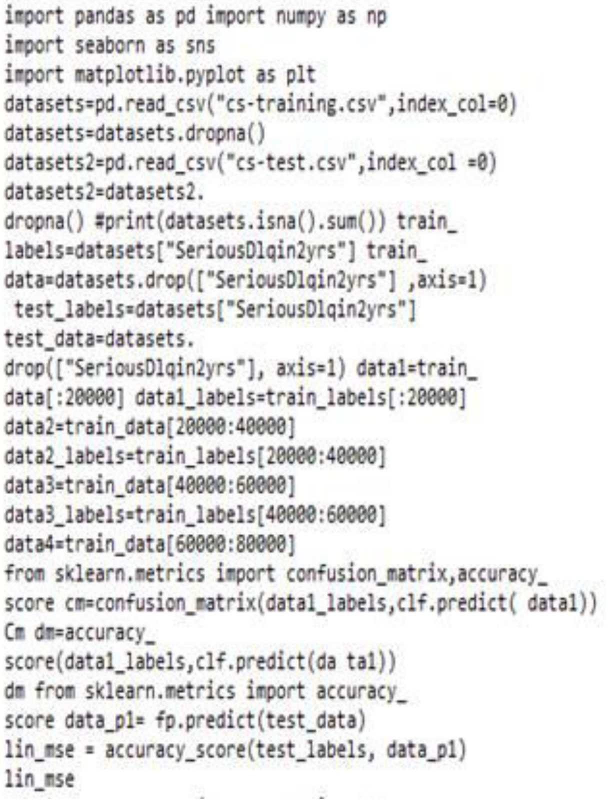
- UML analysis modeling, this focuses on the user model and structural model views of the system.



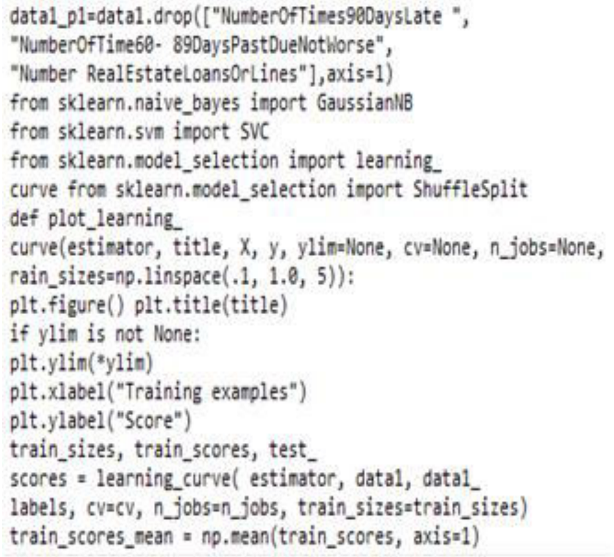
6.1 IMPLEMENTATION

This chapter includes the implementation of the design and source code. In this phase the design is translated into code. Computer programs are written using a conventional programming language or an application generator. Programming tools like Compilers, Interpreters, and Debuggers are used to generate the code. Different high-level programming languages like C, C++, Pascal, Java, .Net are used for coding. With respect to the type of application, the right programming language is chosen.

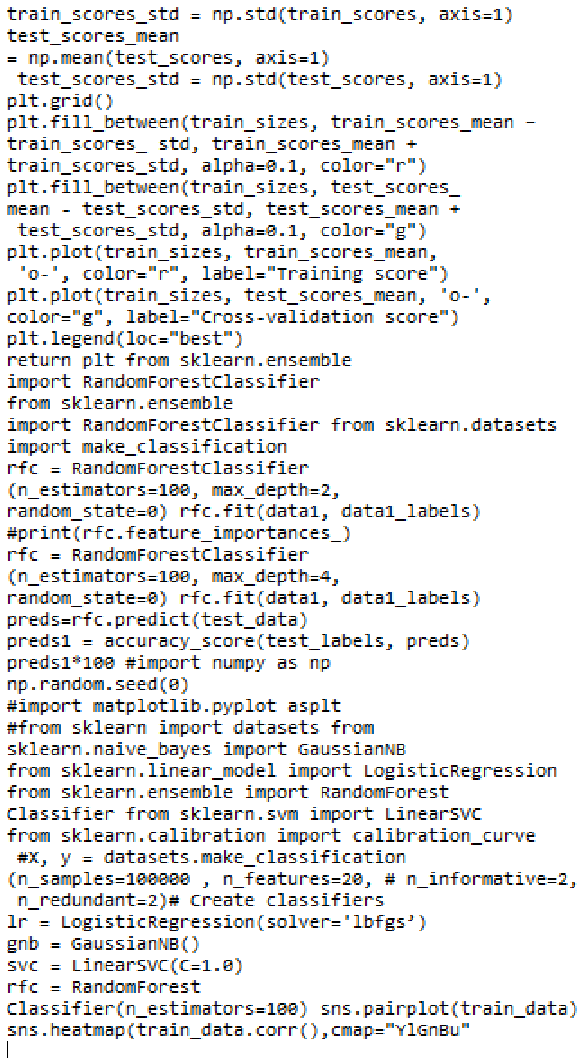
6.2 CODING (SOURCE CODE)



```
import pandas as pd import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
datasets=pd.read_csv("cs-training.csv",index_col=0)
datasets=datasets.dropna()
datasets2=pd.read_csv("cs-test.csv",index_col =0)
datasets2=datasets2.
dropna() #print(datasets.isna().sum()) train_
labels=datasets["SeriousDlqin2yrs"] train_
data=datasets.drop(["SeriousDlqin2yrs"] ,axis=1)
 test_labels=datasets["SeriousDlqin2yrs"]
test_data=datasets.
drop(["SeriousDlqin2yrs"], axis=1) data1=train_
data[:20000] data1_labels=train_labels[:20000]
data2=train_data[20000:40000]
data2_labels=train_labels[20000:40000]
data3=train_data[40000:60000]
data3_labels=train_labels[40000:60000]
data4=train_data[60000:80000]
from sklearn.metrics import confusion_matrix,accuracy_
score cm=confusion_matrix(data1_labels,clf.predict( data1))
Cm dm=accuracy_
score(data1_labels,clf.predict(da ta1))
dm from sklearn.metrics import accuracy_
score data_p1= fp.predict(test_data)
lin_mse = accuracy_score(test_labels, data_p1)
lin_mse
```



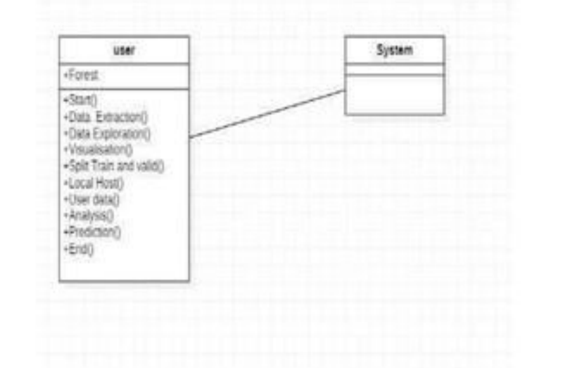
```
data1_p1=data1.drop(["NumberOfTimes90DaysLate ",
"NumberOfTime60- 89DaysPastDueNotWorse",
"Number RealEstateLoansOrLines"],axis=1)
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
from sklearn.model_selection import learning_
curve from sklearn.model_selection import ShuffleSplit
def plot_learning_
curve(estimator, title, X, y, ylim=None, cv=None, n_jobs=None,
rain_sizes=np.linspace(.1, 1.0, 5)):
plt.figure() plt.title(title)
if ylim is not None:
plt.ylim(*ylim)
plt.xlabel("Training examples")
plt.ylabel("Score")
train_sizes, train_scores, test_
scores = learning_curve( estimator, data1, data1_
labels, cv=cv, n_jobs=n_jobs, train_sizes=train_sizes)
train_scores_mean = np.mean(train_scores, axis=1)
```



```
train_scores_std = np.std(train_scores, axis=1)
test_scores_mean
= np.mean(test_scores, axis=1)
 test_scores_std = np.std(test_scores, axis=1)
plt.grid()
plt.fill_between(train_sizes, train_scores_mean -
train_scores_ std, train_scores_mean +
train_scores_std, alpha=0.1, color="r")
plt.fill_between(train_sizes, test_scores_
mean - test_scores_std, test_scores_mean +
 test_scores_std, alpha=0.1, color="g")
plt.plot(train_sizes, train_scores_mean,
 'o-', color="r", label="Training score")
plt.plot(train_sizes, test_scores_mean, 'o-',
color="g", label="Cross-validation score")
plt.legend(loc="best")
return plt from sklearn.ensemble
import RandomForestClassifier
from sklearn.ensemble
import RandomForestClassifier from sklearn.datasets
import make_classification
rfc = RandomForestClassifier
(n_estimators=100, max_depth=2,
random_state=0) rfc.fit(data1, data1_labels)
#print(rfc.feature_importances_)
rfc = RandomForestClassifier
(n_estimators=100, max_depth=4,
random_state=0) rfc.fit(data1, data1_labels)
preds=rfc.predict(test_data)
preds1 = accuracy_score(test_labels, preds)
preds1*100 #import numpy as np
np.random.seed(0)
#import matplotlib.pyplot asplt
#from sklearn import datasets from
sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForest
Classifier from sklearn.svm import LinearSVC
from sklearn.calibration import calibration_curve
 #X, y = datasets.make_classification
(n_samples=100000 , n_features=20, # n_informative=2,
 n_redundant=2)# Create classifiers
lr = LogisticRegression(solver='lbfgs’)
gnb = GaussianNB()
svc = LinearSVC(C=1.0)
rfc = RandomForest
Classifier(n_estimators=100) sns.pairplot(train_data)
sns.heatmap(train_data.corr(),cmap="YlGnBu"
|
```

6.3 CLASS DIAGRAM

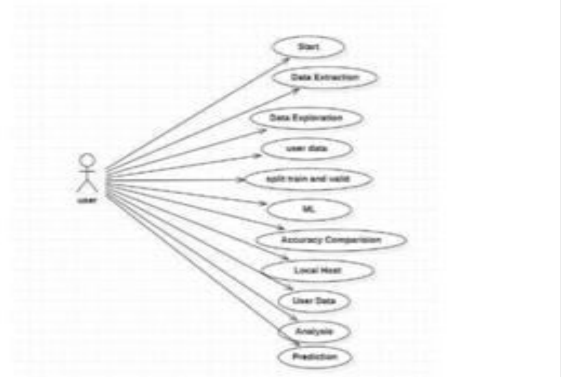
Class diagrams give an overview of a system by showing its classes and the relationships among them. Class diagrams are static - displays what interacts but not what happens when they do interact. In general a class diagram consists of some set of attributes and operations. Operations will be performed on the data values of attributes.



CLASS DIAGRAM OF LOAN PREDICTION SYSTEM

6.4 USE CASE DIAGRAM

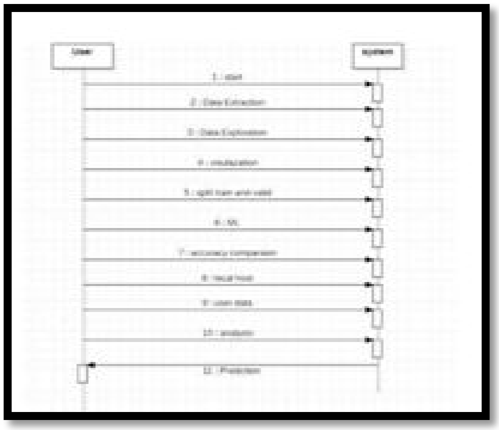
Use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. Actors are the external entities that interact with the system. The use cases are represented by either circles or ellipses.



Use case diagram of loan prediction system

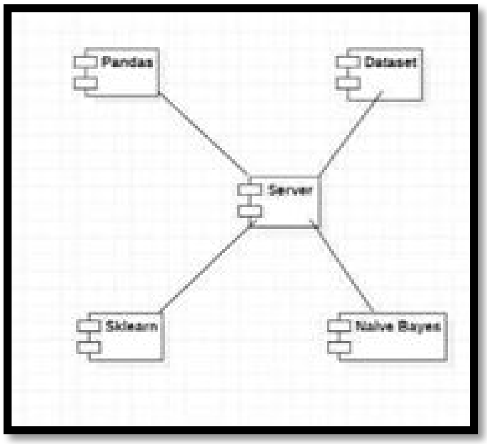
6.5 SEQUENCE DIAGRAM

A Sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development.



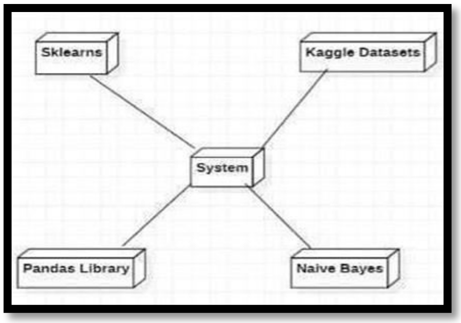
6.6 COMPONENT DIAGRAM

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities.



6.7 DEPLOYMENT DIAGRAM:

The main purpose of the deployment diagram is to represent how software is installed on the hardware component. It depicts in what manner a software interacts with hardware to perform its execution. Both the deployment diagram and the component diagram are closely interrelated to each other as they focus on software and hardware components.



7.1 INTRODUCTION TO TESTING

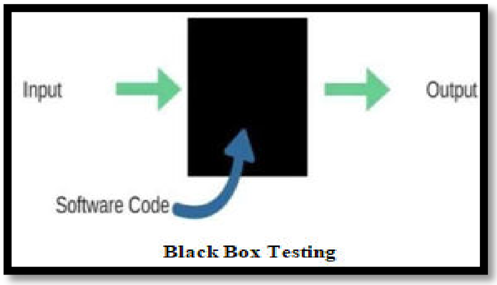
The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software 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 tests. Each test type addresses a specific testing requirement.

7.2 TYPES OF TESTING:

- **Unit Testing**
- **Integration Testing**
- **Functional Testing**
- **White Box Testing**
- **Black Box Testing**

7.3 BLACK BOX TESTING

When applied to machine learning models, black box testing would mean testing machine learning models without knowing the internal details such as features of the machine learning model, the algorithm used to create the model etc. The challenge, however, is to verify the test outcome against the expected values that are known beforehand.



The model gives out the correct output when different inputs are given which are mentioned. Therefore the Program is said to be executed as expected or correct program

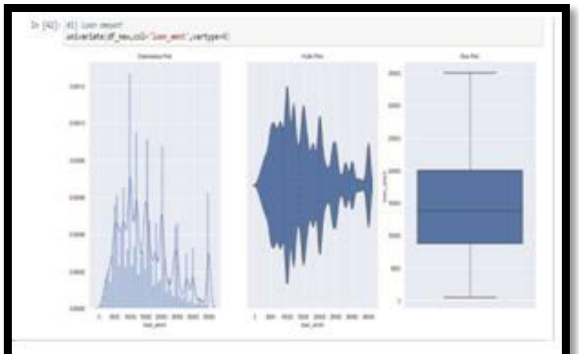
Input	Actual Output	Predicted Output
[16,6,324,0,0,0,22, 0,0,0,0,0,0]	0	0
[16,7,263,7,0,2,700,9,10,1153,832,9, 2]	1	1

Test Case Id	Test Case Name	Test Case Description	Test Steps			Test Case Status	Test PrioritY
			Step	Expected	Actual		
01	Start the	Host the	If it	We	The	High	High
	Applicati o	application	doesn't	cannot	application		
	N	and test if it	Start	run the	hosts		
		starts		applicati	success.		
		making sure		on.			
01		the required software is available					
02	Home Page	Check the Deployment environment for properly loading the Application.	If it doesn't Load.	We Can not Application.	The Application is running successfully	High	High
03	User Mode	Verify the working of the application in freestyle mode	If it doesn't respond	We cannot use the freestyle mode.	The application displays the freestyle page	High	High
a04	Data Input	Verify if the Application takes input and updates	If it fails to take the input or store in the database	We Cannot proceed further	The Application updates the input to the applications	low	low

8. OUTPUT

8.1 UNIVARATE ANALYSIS

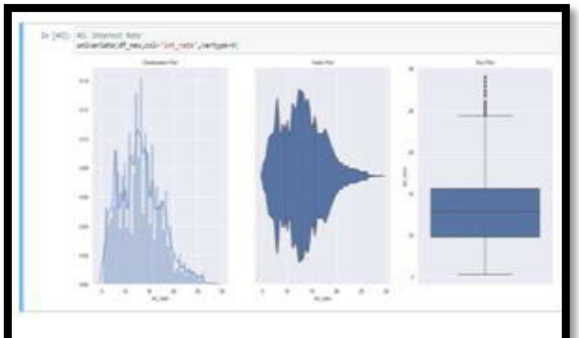
- **LOAN AMOUNT**



```
In [42]: #1. Loan amount
         univariate(df_new,col='loan_amnt',vartype=0)
```

Insights: Most of the loan amounts are distributed between 8000 to 20000 USD.

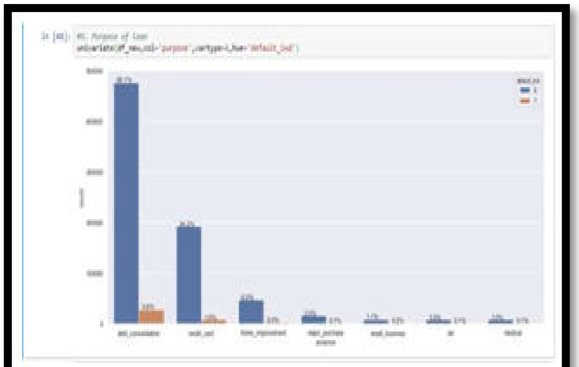
- **INTEREST RATE**



```
In [43]: #2. Interest Rate
         univariate(df_new,col='int_rate',vartype=0)
```

Insights: Most of the loans interest rates are distributed between 10% to 16%.

- **PURPOSE OF LOAN**

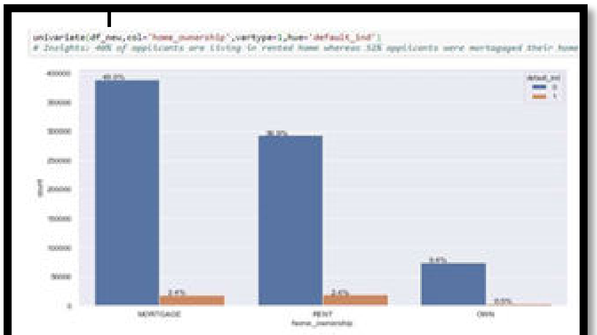


```
In [44]: #3. Purpose of loan
         univariate(df_new,col='purpose',vartype=1,hue='default_ind')
```

Insights: Approx. 60% of the applicants applied loan for paying their other loans (Debt Consolidation).

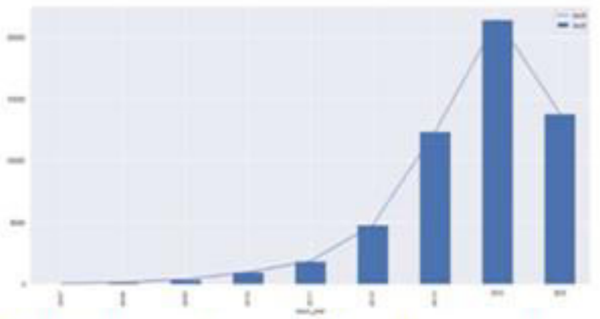
- ***HOME OWNERSHIP WISE LOAN***

```
univariate(df_new,col='home_ownership',vartype=1,hue='default_ind')
# Insights: 40% of applicants are living in rented home whereas 52% applicants were mortgaged their home
```

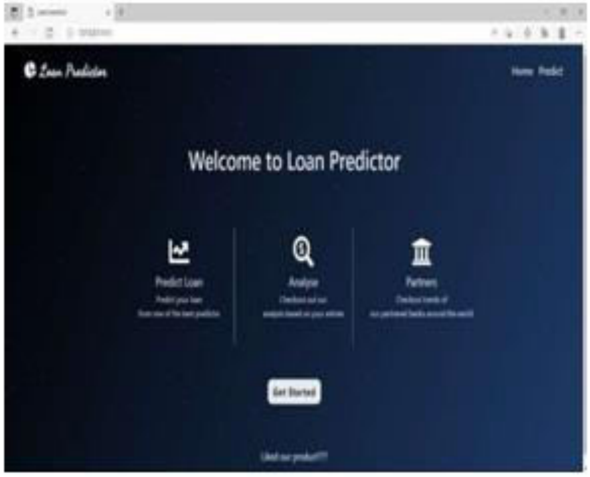


Insights: 40% of applicants are living in rented home where as 52% applicants were mortgaged their home.

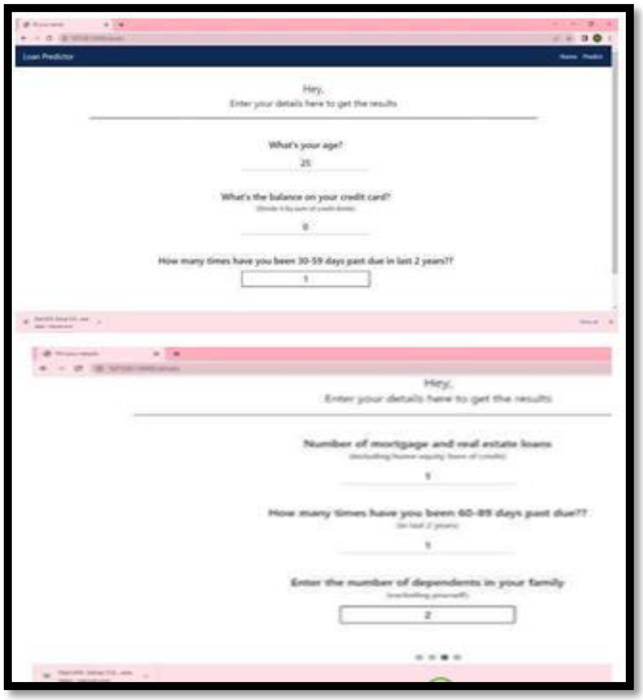
8.2 YEAR WISE LOAN:



nsights: loan applicants are increasing year on ear, approx. 47% of loan applicants received oans in 2011.



8.3RESULT



Hey,
Enter your details here to get the results

What's your age?
25

What's the balance on your credit card?
0

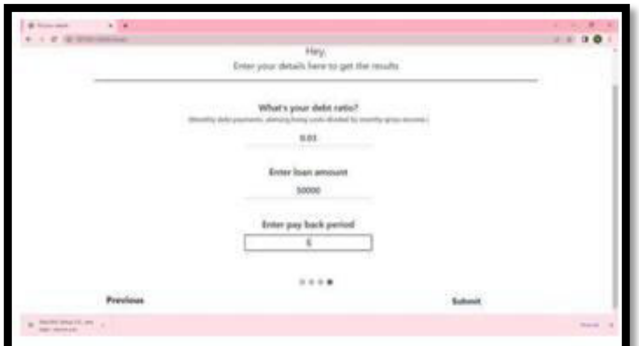
How many times have you been 30-59 days past due in last 2 years??
1

Hey,
Enter your details here to get the results

Number of mortgage and real estate loans
1

How many times have you been 60-89 days past due??
1

Enter the number of dependents in your family
2

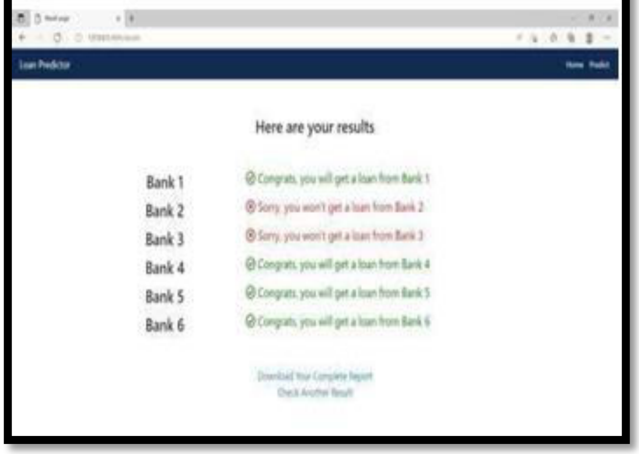


Hey,
Enter your details here to get the results

What's your debt ratio?
0.03

Enter loan amount
50000

Enter pay back period
5



Here are your results

Bank 1	✓ Congrats, you will get a loan from Bank 1
Bank 2	☒ Sorry, you won't get a loan from Bank 2
Bank 3	☒ Sorry, you won't get a loan from Bank 3
Bank 4	✓ Congrats, you will get a loan from Bank 4
Bank 5	✓ Congrats, you will get a loan from Bank 5
Bank 6	✓ Congrats, you will get a loan from Bank 6

9. CONCLUSION

Therefore, the developed model automates the method of determining the applicant's creditworthiness. It focuses on information containing the main points of the loan applicants. In this system NAIVE BAYES Classification model is used. In Machine Learning, NAIVE BAYES classification analysis is one of the supervised learning algorithms, which is dependent on BAYES theorem and used to solve classification problems. Hence, it is good for predicting the right result in the current world scenario and also help the bank to give the money in the right hands and also help the people in getting loan in a much faster way. The main advantage of this system is, it gives more accuracy.

REFERENCES

1 *TOBY SEGARAN,* "Programming Collective Intelligence: Building Smart Web 2.0 Applications." O'REILLY MEDIA.

2 *DREW CONWAY AND JOHN MYLES WHITE*," Machine Learning for Hackers: Case Studies and Algorithms to Get you Started," O'REILLY MEDIA.

3 *TREVOR HASTIE, ROBERT TIBSHIRANI, AND JEROME FRIEDMAN*, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction," Springer, Kindle

4 PHILHYO JIN DO ,HO-JIN CHOI, "Sentiment analysis of real-life situations using location, people and time as contextual features," International Conference on Big Data and Smart Computing (BIGCOMP), pp. 39-42. IEEE, 2015.

5 BING LIU, "Sentiment Analysis and Opinion Mining," Morgan & Claypool Publishers, May 2012.

*6 SHIYANG LIAO, JUNBO WANG, RUIYUN YU, KOICHI SATO, AND ZIXUE CHENG, "*CNN for situations understanding based on sentiment analysis of twitter data," PROCEDIA computer science, 111:376-381, 2017.CrossRef.

7 K I RAHMANI, M.A. ANSARI, AMIT KUMAR GOEL, "An Efficient Indexing Algorithm for CBIR,"IEEE-International Conference on Computational Intelligence & Communication Technology, 13-14 Feb 2015.

8 GURLOVE SINGH, AMIT KUMAR GOEL ,"Face Detection and Recognition System using Digital Image Processing" , 2nd International conference on Innovative Mechanism for Industry Application ICMIA 2020, 5-7 March 2020, IEEE Publisher.

9 *AMIT KUMAR GOEL, KALPANA BATRA, POONAM PHOGAT*," Manage big data using optical networks", Journal of Statistics and Management Systems "Volume 23, 2020, Issue 2, Taylors & Francis.