PREDICTION OF MODERNIZED LOAN APPROVAL SYSTEM BASED ON MACHINE LEARNING APPROACH

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

S V V Mani Kumar Garikipati (38110552)
Srikar Meher Nemani (38110362)

DEPARTMENT OF COMPUTER SCIENCE AND   ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI
SALAI, CHENNAI - 600 119

MARCH - 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of S V V Mani Kumar Garikipati (38110552), Srikar Meher Nemani(38110362) who carried out the project entitled "**PREDICTION OF MODERNIZED LOAN APPROVAL SYSTEM BASED ON MACHINE LEARNING APPROACH**" under my supervision from October 2021 to May 2022

InternalGuide

(Ms. C Mana SujaM.E(Ph.D))

Head of the Department

Submitted for Viva voce Examination held on _____

Internal Examiner   **External Examiner**

DECLARATION

I S V V Mani Kumar Garikipati,Srikar Meher Nemani hereby declare that the

Project Report entitled **PREDICTION OF MODERNIZED LOAN APPROVAL**

SYSTEM BASED ON MACHINE LEARNING APPROACH done by me under

the guidance of **Ms. C Mana Suja** .M.E(Ph.D) (Internal) is submitted in partial

fulfillment of the requirements for the award of Bachelor of Engineering /

Technology degree In COMPUTER SCIENCE AND TECHNOLOGY.

DATE:

PLACE: **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D**, **Dean**, School of Computing **Dr. L. Lakshmanan M.E., Ph.D. ,** and **Dr.S.Vigneshwari M.E., Ph.D. Heads** of the

ABSTRACT

Technology has boosted the existence of humankind the quality of life they live. Every day we are planning to create something new and different. We have a solution for every other problem we have machines to support our lives and make us somewhat complete in the banking sector candidate gets proofs/ backup before approval of the loan amount. The application approved or not approved depends upon the historical data of the candidate by the system. Every day lots of people applying for the loan in the banking sector but Bank would have limited funds. In this case, the right prediction would be very beneficial using some classes-function algorithm. An example the logistic regression, random forest classifier, support vector machine classifier, etc. A Bank's profit and loss depend on the amount of the loans that is whether the Client or customer is paying back the loan. Recovery of loans is the most important for the banking sector. The improvement process plays an important role in the banking sector. The historical data of candidates was used to build a machine learning model using different classification algorithms. The main objective of this paper is to predict whether a

new applicant granted the loan or not using machine learning models trained on the historical data set.

1 INTRODUCTION

As the data are increasing daily due to digitization in the banking sector, people want to apply for loans through the internet. Artificial intelligence (AI), as a typical method for information investigation, has gotten more consideration increasingly. Individuals of various businesses are utilizing AI calculations to take care of the issues dependent on their industry information. Banks are facing a significant problem in the approval of the loan. Daily there are so many applications that are challenging to manage by the bank employees, and also the chances of some mistakes are high. Most banks earn profit from the loan, but it is risky to choose deserving customers from the number of applications. One mistake can make a massive loss to a bank. Loan distribution is the primary business of almost every bank. This project aims to provide a loan to a deserving applicant out of all applicants. An efficient and non-biased system that reduces the bank's time employs checking every applicant on a priority basis. The bank authorities complete all other customer's other formalities on time, which positively impacts the customers. The best part is that it is efficient for both banks and applicants. This system allows jumping on particular applications that deserve to be approved on a priority basis. There are some features for the prediction like- 'Gender', 'Married', 'Dependents', 'Education', 'Self_ Employed', 'ApplicantIncome', 'CoapplicantIncome', 'LoanAmount', 'Loan_Amount_Term', 'Credit_History', 'Property_Area', 'Loan_Status'.

2 LITERATURE SURVEY

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, then the next step is to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system. The major part of the project development sector considers and fully survey all the required needs for developing the project. For every project Literature survey is the most important sector in software development process. Before developing the tools and the associated designing it is necessary to determine and survey the time factor, resource requirement, man power, economy, and company strength. Once these things are satisfied and fully surveyed, then the next step is to determine about the software specifications in the respective system such as what type of operating system the project would require, and what are all the necessary software are needed to proceed with the next step such as developing the tools, and the associated operations.

Loan Approval Prediction based on Machine Learning Approach

The main objective of this paper is to predict whether assigning the loan to particular person will be safe or not. This paper is divided into four sections (i)Data Collection (ii) Comparison of machine learning models on collected data (iii) Training of system on most promising model (iv) Testing.

Exploring the Machine Learning Algorithm for Prediction the Loan Sanctioning Process

Extending credits to corporates and individuals for the smooth functioning of growing economies like India is inevitable. As increasing number of customers apply for loans in the banks and non- banking financial companies (NBFC), it is really challenging for banks and NBFCs with limited capital to device a standard resolution and safe procedure to lend money to its borrowers for their financial needs. Inaddition, in recent times NBFC inventories have suffered a significant downfall in terms of the stock price. It has contributed to a contagion that has also spread to other financial stocks, adversely affecting the benchmark in recent times.In this paper, an attempt is made to condense the risk involved in selecting the suitable person who could repay the loan on time thereby keeping the bank's nonperforming assets (NPA) on the hold. This is achieved by feeding the past records of the customer who acquired loans from the bank into a trained machine learning model which could yield an accurate result. The prime focus of the paper is to determine whether or not it will be safe to allocate the loan to a particular person. This paper has the following sections (i) Collection of Data, (ii) Data Cleaning and (iii) Performance Evaluation. Experimental tests found that the Naïve Bayes model has better performance Evaluation. Experimental tests found that the Naïve Bayes model has better performance than other models in terms of loan forecasting.

Loan Prediction using machine learning model

whether or not it will be safe to allocate the loan to a particular person. This paper has the following sections (i) Collection of Data, (ii) Data Cleaning and (iii) Performance Evaluation. Experimental tests found that the Naïve Bayes model has better performance than other models in terms of loan forecasting.With the

enhancement in the banking sector lots of people are applying for bankloans but the bank has its limited assets which it has to grant to limited people only,so finding out to whom the loan can be granted which will be a safer option for the bank is a typical process. So in this project we try to reduce this risk factor behind selecting the safe person so as to save lots of bank efforts and assets. This is done by mining the Big Data of the previous records of the people to whom the loan was granted before and on the basis of these records/experiences the machine was trained using the machine learning model which give the most accurate result The main objective of this project is to predict whether assigning the loan to particular person will be safe or not. This paper is divided into four sections (i)Data Collection (ii) Comparison of machine learning models on collected data (iii) Training of system on most promising model (iv) Testing. In this paper we are predict the loan data by using some machine learning algorithms they are classification, logic regression, Decision Tree and gradient boosting.

Loan Prediction using Decision Tree and Random Forest

In India the number of people or organization applying for loan gets increasd every year. The bank have to put in a lot of work to analyse or predict whether the customer can pay back the loan amount or not (defaulter or non-defaulter) in the given time. The aim of this paper is to find the nature or background or credibility of client that is applying for the loan. We use exploratory data analysis technique to deal with problem of approving or rejecting the loan request or in short loan prediction. The main focus of this paper is to determine whether the loan given to a particular person or an organization shall be approved or not.

Prediction of Loan Risk using NB and Support Vector Machine

In recent years, banks and creditors face lots of challenges associated with the bank loan. In addition, banking transactions are growing rapidly and large data

sets that represent customer behaviour and risks around debt burdens have increased. So, it became mandatory to know the risks related to bank loans. Each consumer knows what they want and what kind of business they would like to choose. Customer can simply find another bank if administrators are not ready to pay attention. Many data analysis technique exists that focus on loan risk. These techniques are oriented toward extraction of quantity and statistical data properties. They facilitate useful data explanations and help to gain better insights in the process behind data. While there can be traditional data analysis techniques lead to intuition knowledge, it is still made by human researchers. Banks and many investment companies are pioneers using data mining. The proposed work in data mining focused on using data from banking sector to predict the status of loans. Two classification algorithms Naïve Bayes and Support Vector Machine are discussed here. The results have been proven that the speed and accuracy has been improved when combined these two techniques.

3 EXISITNG SYSTEM

- Y. Shi and P. Song proposed a method for evaluating project loans using risk analysis. The method evaluate the risk involved in loans of commercial banks.

- R. ZhangandD. Li used machine learning approached in prediction systems. The machine learning approach was used for assessment of water quality. The paper concluded that machine learning is a very unimportant tool in prediction systems.

- C. Franket al. used machine learning in prediction of smoking status. Different machine learning approaches were applied and investigated for finding the smoking status. From the results it was ensured that logistic algorithm performs better.

- R. Lopeset al. applied machine learning approach for the prediction of credit recovery. Credit recovery is very important issue for banking system. The prediction of credit recovery is a challenging task. Different machine learning approach was applied to predict the credit recovery and gradient expansion algorithms (GBM) outperformed the other machine learning approaches.

3.1 DISADVANTAGES OF EXISTING SYSTEM

- Existing system are often relatively inaccurate.

- Calculations can get very complex, particularly if many values are uncertain and/or if many outcomes are linked.

- Existing system often involves higher time to train the model.

- Existing system is relatively expensive as the complexity and time has taken are more.

- Existing system generally leads to the overfitting of data.

- Little bit of noise can make it unstable which leads to wrong predictions.

3.2 PROPOSED SYSTEM

- This proposed model will characterize the behavior of customers on the Basis of their record. These records are taken from the customers, and
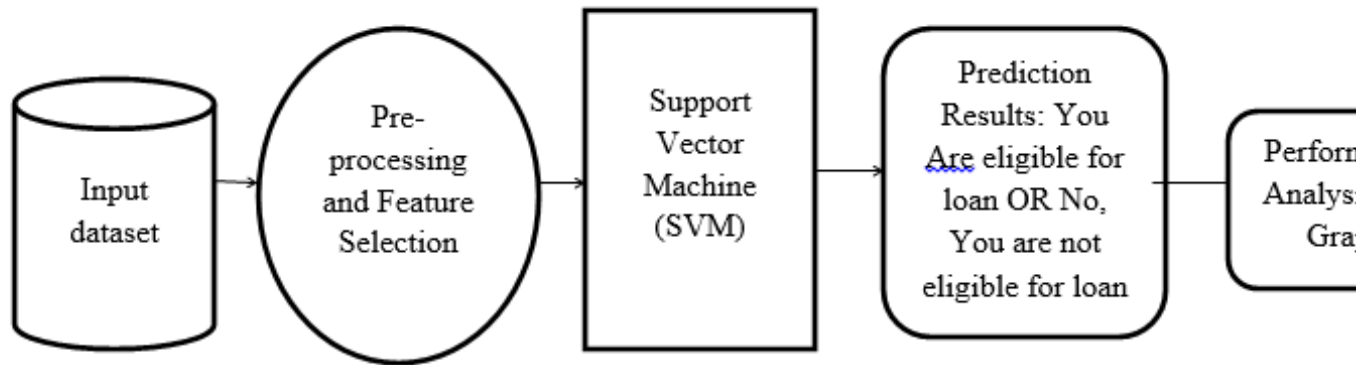
create a data set. With the help of these data sets and training machine learning model, we predict that the customer's loan will pass or not.

- The aim of this Paper is to provide quick, immediate and easy way to choose the deserving applicants. It can provide special advantages to the bank. The Loan Prediction System can automatically calculate the weight of each features taking part in loan processing and on new test data same features are processed with respect to their associated weight. A time limit can be set for the applicant to check whether his/her loan can be sanctioned or not.

- Loan Prediction System allows jumping to specific application so that it can be check on priority basis. This Paper is exclusively for the managing authority of Bank/finance company, whole process of prediction is done privately no stakeholders would be able to alter the processing.

3.3 ADVANTAGES OF PROPOSED SYSTEM

- The advantage of this system is that we provided some conditions by setting the algorithms and just by evaluating the details, we get to know eligibility criteria that client is eligible or not.

- The proposed system also scales relatively well to high dimensional data.

- The proposed system is relatively memory efficient.

- The risk of over-fitting is less in our proposed system.

- A small change to the data does not greatly affect the hyperplane

3.4 SYSTEM ARCHITECTURE

| Input dataset | → | Pre-processing and Feature Selection | → | Support Vector Machine (SVM) | → | Prediction Results: You Are eligible for loan OR No, You are not eligible for loan | → | Perform Analys Gra |

3.5 SYSTEM REQUIREMENTS:

3.6 HARDWARE REQUIREMENTS:

- System : Pentium Dual Core.

- Hard Disk : 120 GB.

- Monitor : 15'' LED

- Input Devices : Keyboard, Mouse

- Ram : 4 GB.

3.7 SOFTWARE REQUIREMENTS:

- Operating system: Windows 7/10.

- Coding Language :Python

4   SOFTWARE ENVIRONMENT

Python:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

- **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

- **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

- **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

4.1 History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

4.2 Python Features

Python's features include −

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.

- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.

- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.

- **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.

- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.

- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.

- **Databases** − Python provides interfaces to all major commercial databases.

- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

- **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below −

- It supports functional and structured programming methods as well as OOP.

- It can be used as a scripting language or can be compiled to byte-code for building large applications.

- It provides very high-level dynamic data types and supports dynamic type checking.

- It supports automatic garbage collection.

- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

Getting Python

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python https://www.python.org.

Windows Installation

Here are the steps to install Python on Windows machine.

- Open a Web browser and go to https://www.python.org/downloads/.

- Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.

- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.

- Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

4.3 First Python Program

Let us execute programs in different modes of programming.

Invoking the interpreter without passing a script file as a parameter brings up the following prompt −

$ python

Python2.4.3(#1,Nov112010,13:34:43)

[GCC 4.1.220080704(RedHat4.1.2-48)] on linux2

Type"help","copyright","credits"or"license"for more information.

>>>

Type the following text at the Python prompt and press the Enter −

>>>print"Hello, Python!"

If you are running new version of Python, then you would need to use print statement with parenthesis as in **print ("Hello, Python!");**. However in Python version 2.4.3, this produces the following result −

Hello, Python!

4.4 Script Mode Programming

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension **.py**. Type the following source code in a test.py file −

```
print"Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows −

```
$ python test.py
```

This produces the following result −

```
Hello, Python!
```

4.5 Flask Framework:

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Http protocol is the foundation of data communication in world wide web. Different methods of data retrieval from specified URL are defined in this protocol.

The following table summarizes different http methods −

Sr.No	Methods & Description

1	**GET** Sends data in unencrypted form to the server. Most common method.
2	**HEAD** Same as GET, but without response body
3	**POST** Used to send HTML form data to server. Data received by POST method is not cached by server.
4	**PUT** Replaces all current representations of the target resource with the uploaded content.
5	**DELETE** Removes all current representations of the target resource given by a URL

By default, the Flask route responds to the **GET** requests. However, this preference can be altered by providing methods argument to **route()** decorator.

In order to demonstrate the use of **POST** method in URL routing, first let us create an HTML form and use the **POST** method to send form data to a URL.

Save the following script as login.html

```
<html>

<body>

<form action="http://localhost:5000/login" method="post">

<p>Enter Name:</p>

<p><input type="text" name="nm"/></p>
```

```html
<p><input type="submit" value="submit"/></p>

</form>

</body>

</html>
```

Now enter the following script in Python shell.

```python
from flask import Flask, redirect,url_for, request

app=Flask(__name__)

@app.route('/success/<name>')

def success(name):

return 'welcome %s'% name

@app.route('/login',methods=['POST','GET'])

def login():

if request.method=='POST':

user=request.form['nm']

return redirect(url_for('success',name= user))

else:

user=request.args.get('nm')

return redirect(url_for('success',name= user))

if __name__ =='__main__':

app.run(debug =True)
```
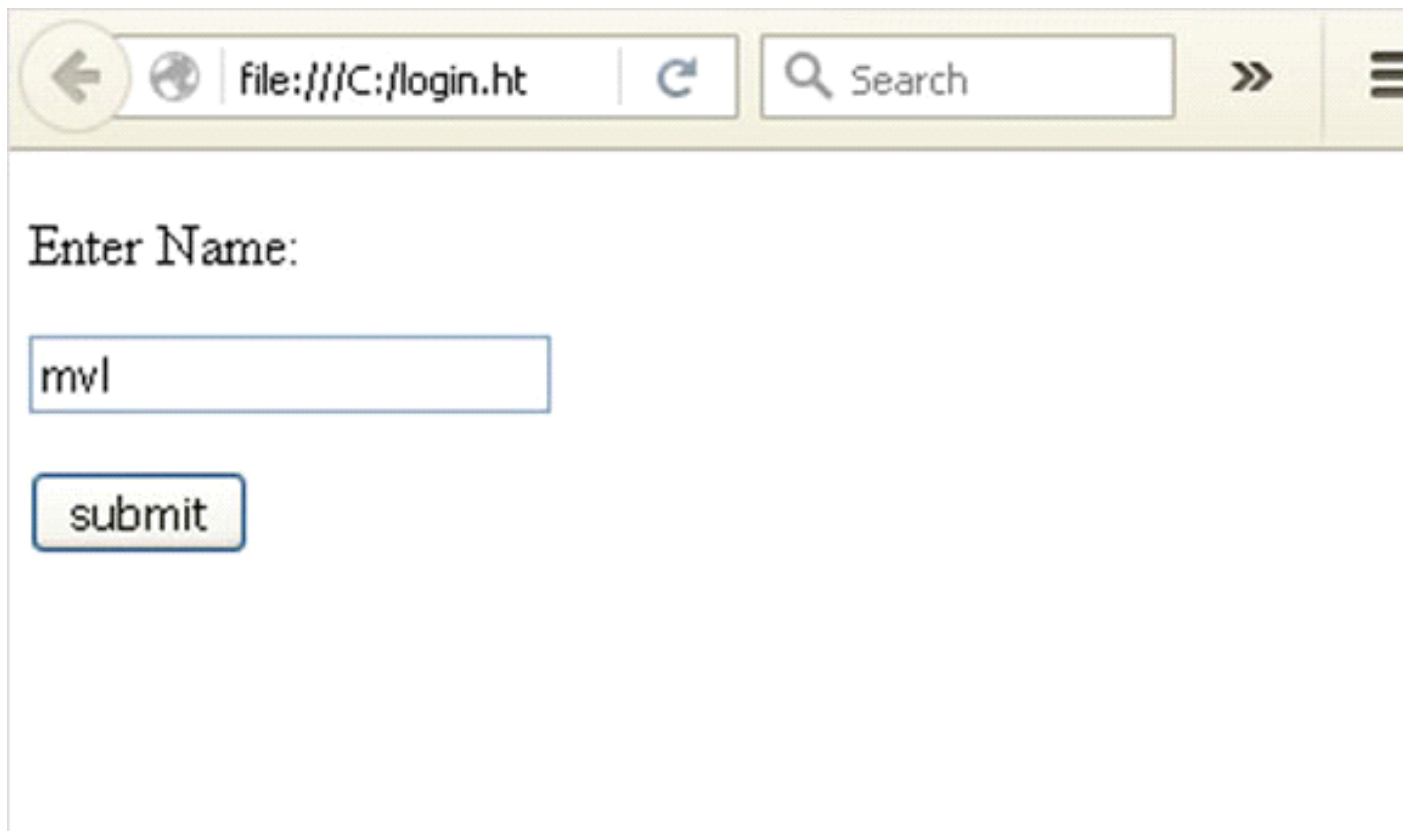
After the development server starts running, open **login.html** in the browser, enter name in the text field and click **Submit**.

Form data is POSTed to the URL in action clause of form tag.

http://localhost/login is mapped to the **login()** function. Since the server has received data by **POST** method, value of 'nm' parameter obtained from the form data is obtained by —

```
user = request.form['nm']
```

It is passed to **'/success'** URL as variable part. The browser displays a **welcome** message in the window.

welcome mvl
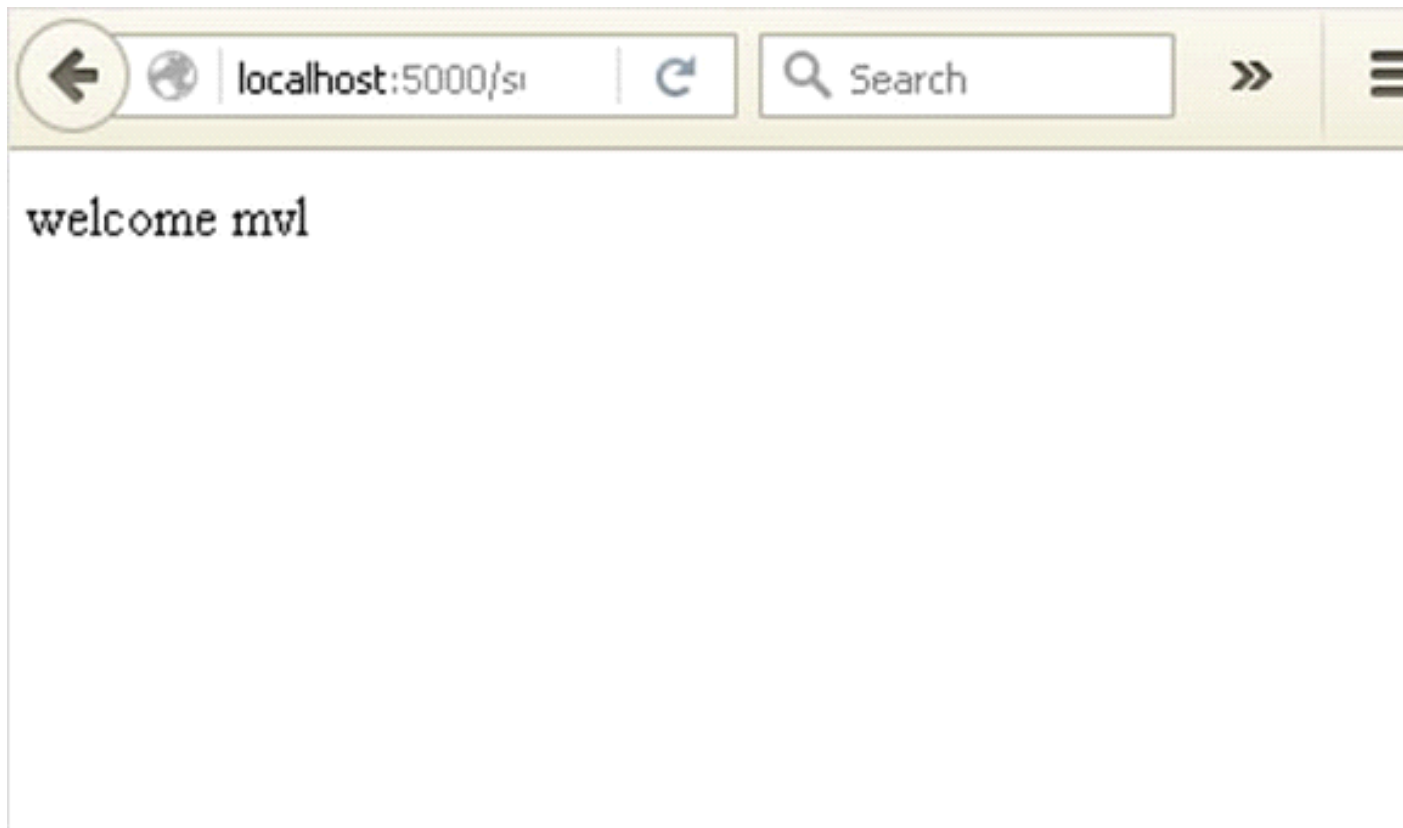
Change the method parameter to **'GET'** in **login.html** and open it again in the browser. The data received on server is by the **GET** method. The value of 'nm' parameter is now obtained by −

User = request.args.get('nm')

Here, **args** is dictionary object containing a list of pairs of form parameter and its corresponding value. The value corresponding to 'nm' parameter is passed on to '/success' URL as before.

5 MODULES

- Data Collection

- Dataset

- Data Preparation

- Model Selection

- Analyze and Prediction

- Accuracy on test set

- Saving the Trained Model

5.1 MODULE DESCRIPTION

Data Collection:

This is the first real step towards the real development of a machine learning model, collecting data. This is a critical step that will cascade in how good the model will be, the more and better data that we get, the better our model will perform.

There are several techniques to collect the data, like web scraping, manual interventions and etc.

Prediction of Modernized Loan Approval System Based on Machine Learning Approach

We given the data set in the project folder

5.2 Dataset:

The dataset consists of 511 individual data. There are 10 columns in the dataset, which are described below.

- **Application_ID**   : Unique Loan ID

- **Gender** : Male/ Female

- **Married**   : Applicant married (Y/N)

- **Dependents**   : Number of dependents

- **Education**   : Applicant Education (Graduate/ Under Graduate)

- **Self_Employed** : Self employed (Y/N)
- **Credit_History :** credit history meets guidelines
- **Property_Area** : Urban/ Semi Urban/ Rural
- **Income** : Applicant income
- **Application_Status** : (Target) Loan approved (Y/N)

5.3 Data Preparation:

Wrangle data and prepare it for training. Clean that which may require it (remove duplicates, correct errors, deal with missing values, normalization, data type conversions, etc.)

Randomize data, which erases the effects of the particular order in which we collected and/or otherwise prepared our data

Visualize data to help detect relevant relationships between variables or class imbalances (bias alert!), or perform other exploratory analysis

Split into training and evaluation sets

6 Model Selection:

We used support vector machine algorithm, We got a accuracy of 0.82 test set so we implemented this algorithum.

Support Vector Machines (SVM) are learning systems that use a hypothesis space of linear functions in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical learning theory

Goal of the SVM is to find the optimal hyperplane that divides the two classes. There can be different planes that can divide the two classes, but the main focus is on to finding out such plane that we can achieve maximum margin between the classes. It means pick the hyperplane so that the distance from the hyperplane to the nearest data point is maximized

How does it work?

Above, we got accustomed to the process of segregating the two classes with a hyper-plane. Now the burning question is "How can we identify the right hyper-plane?". Don't worry, it's not as hard as you think!

Let's understand:

Identify the right hyper-plane (Scenario-1): Here, we have three hyper-planes (A, B, and C). Now, identify the right hyper-plane to classify stars and circles.
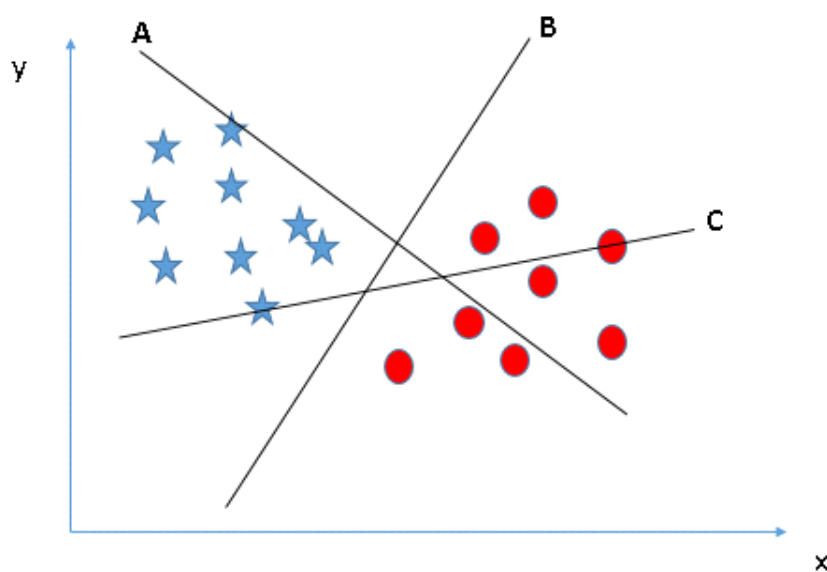
You need to remember a thumb rule to identify the right hyper-plane: "Select the hyper-plane which segregates the two classes better". In this scenario, hyper-plane "B" has excellently performed this job.

Identify the right hyper-plane (Scenario-2): Here, we have three hyper-planes (A, B, and C) and all are segregating the classes well. Now, How can we identify the right hyper-plane?
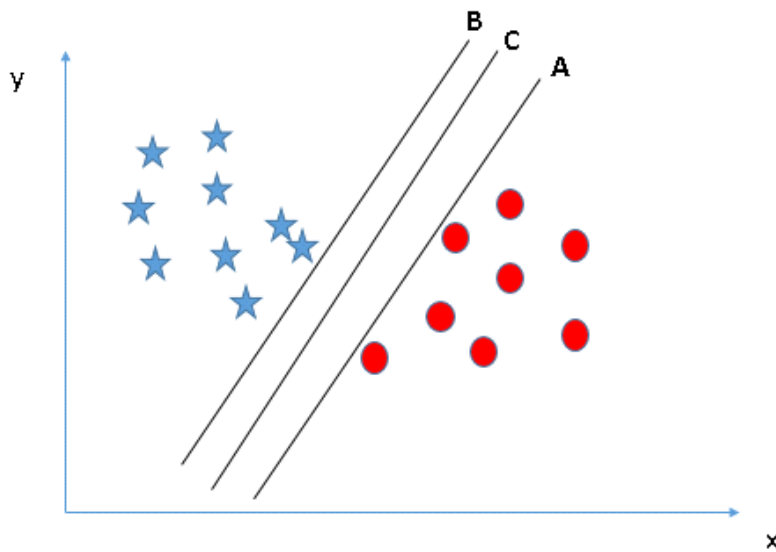
Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**.   Let's   look   at   the   below   snapshot:
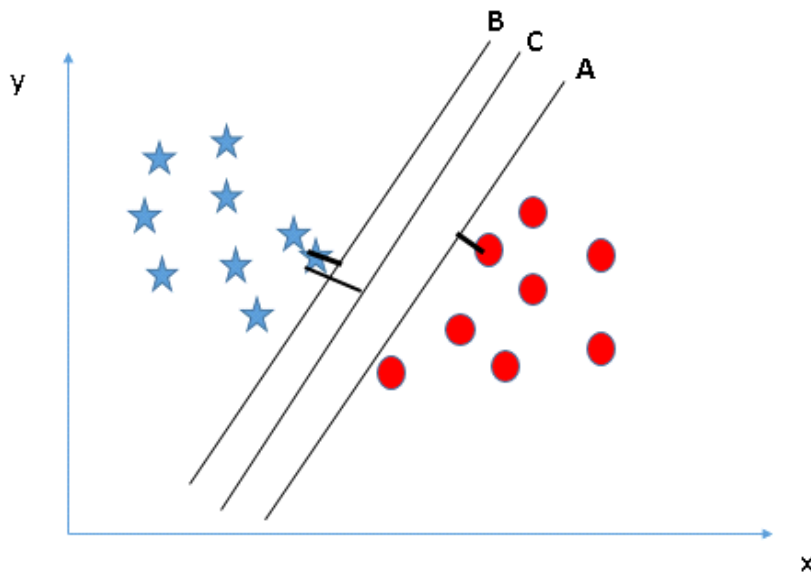
Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyper-plane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

Identify the right hyper-plane (Scenario-3): Hint: Use the rules as discussed in previous section to identify the right hyper-plane
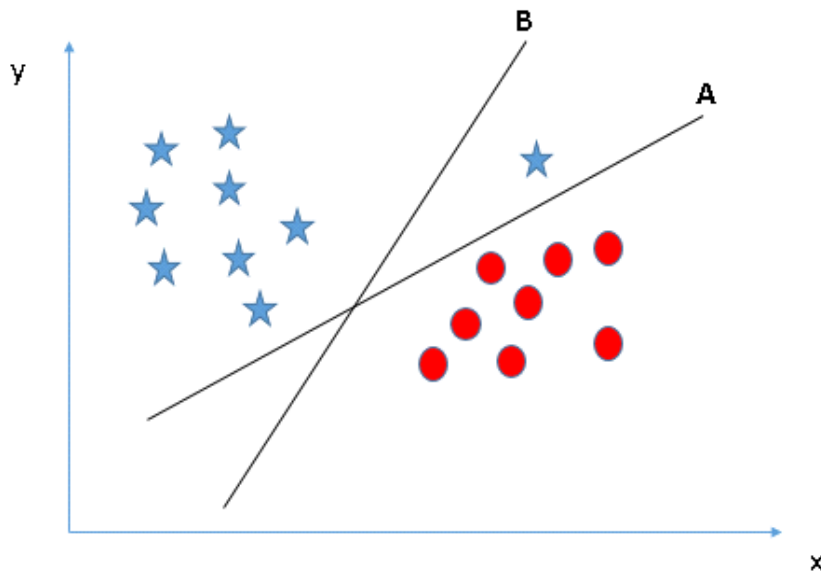
Some of you may have selected the hyper-plane **B** as it has higher margin compared to **A.** But, here is the catch, SVM selects the hyper-plane which classifies the classes accurately prior to maximizing margin. Here, hyper-plane B has a classification error and A has classified all correctly. Therefore, the right hyper-plane is **A.**

Can we classify two classes (Scenario-4)?: Below, I am unable to segregate the two classes using a straight line, as one of the stars lies in the territory of

other(circle) class as an outlier.

As I have already mentioned, one star at other end is like an outlier for star class. The SVM algorithm has a feature to ignore outliers and find the hyper-plane that has the maximum margin. Hence, we can say, SVM classification is robust to outliers.
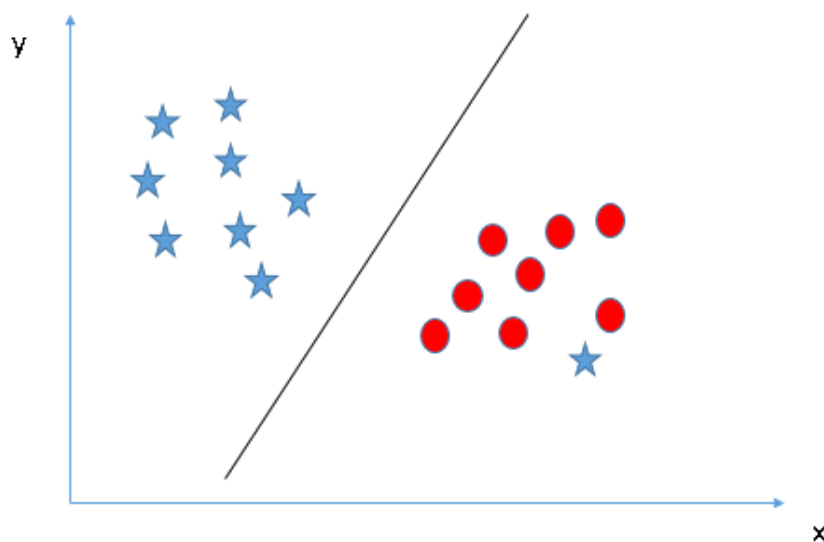
Find the hyper-plane to segregate to classes (Scenario-5): In the scenario below, we can't have linear hyper-plane between the two classes, so how does SVM classify these two classes? Till now, we have only looked at the linear hyper-plane.
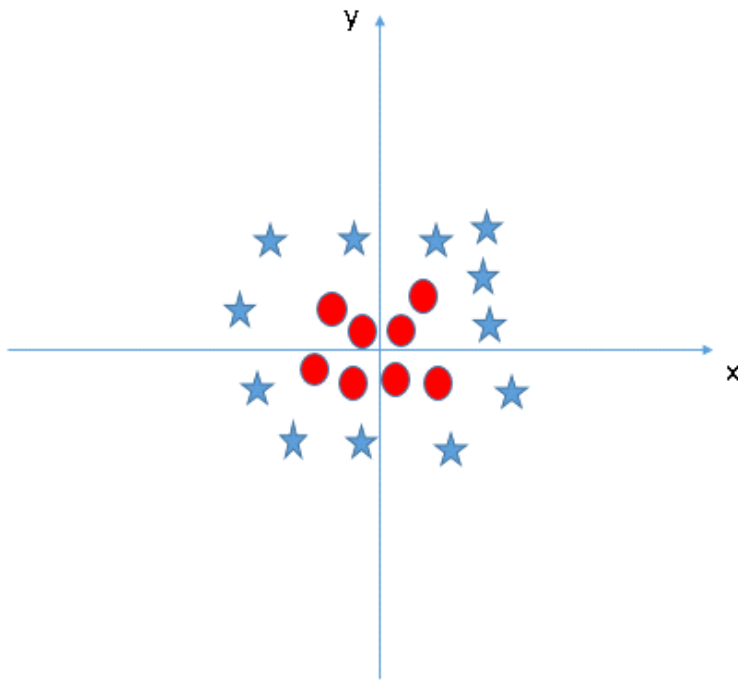
SVM can solve this problem. Easily! It solves this problem by introducing additional feature. Here, we will add a new feature z=x^2+y^2. Now, let's   plot   the   data   points   on   axis   x   and   z:
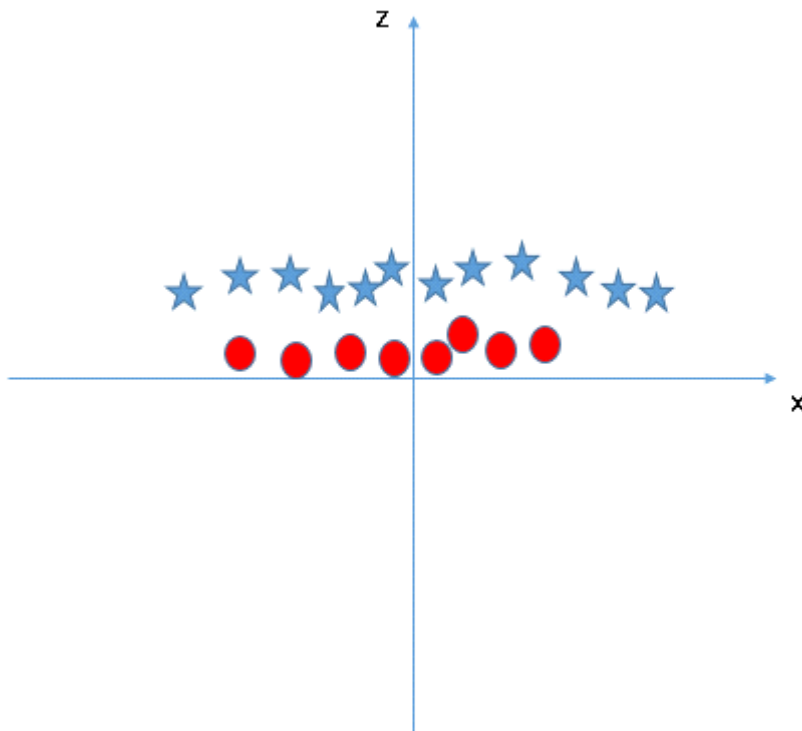
In above plot, points to consider are:

All values for z would be positive always because z is the squared sum of both x and y

In the original plot, red circles appear close to the origin of x and y axes, leading to lower value of z and star relatively away from the origin result to higher value of z.

In the SVM classifier, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, the SVM algorithm has a technique called the **kernal trick**. The SVM kernel is a function that takes low dimensional input space and transforms it to a higher dimensional space i.e. it converts not separable problem to separable problem. It is mostly useful in non-linear separation problem. Simply put, it does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs you've defined.

When we look at the hyper-plane in original input space it looks like a circle:
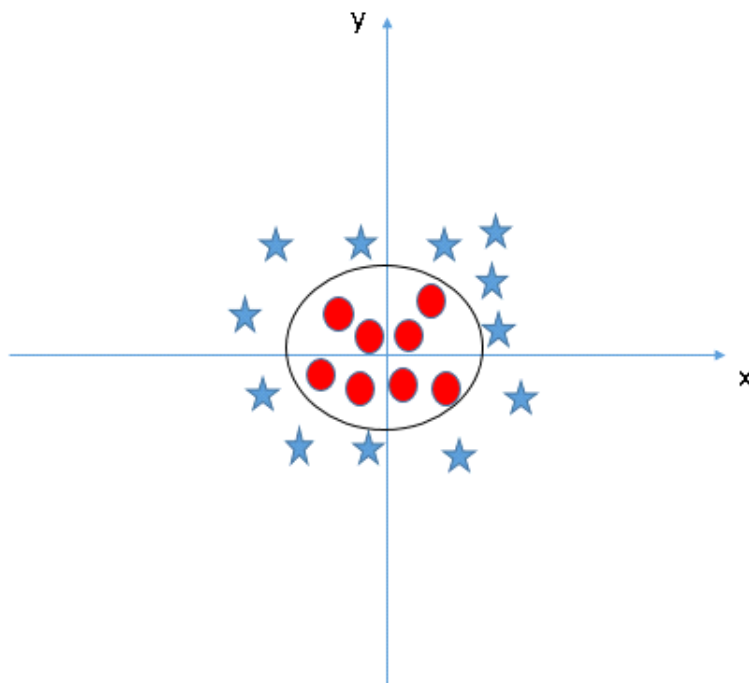
6.1 Analyze and Prediction:

In the actual dataset, we chose only 7 features:

- **Gender** : Male/ Female

- **Married** : Applicant married (Y/N)

- **Education** : Applicant Education (Graduate/ Under Graduate)

- **Self_Employed** : Self employed (Y/N)

- **Credit_History :** credit history meets guidelines

- **Property_Area** : Urban/ Semi Urban/ Rural

- **Income** : Applicant income

- **Application_Status** : (Target) Loan approved (Y/N)

Accuracy on test set:

We got a accuracy of 0.82% on test set.

Saving the Trained Model:

Once you're confident enough to take your trained and tested model into the production-ready environment, the first step is to save it into a .h5 or .pkl file using a library like `pickle`.

Make sure you have `pickle` installed in your environment.

Next, let's import the module and dump the model into `.pkl` file

6.2 REQUIREMENT ANALYSIS

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analysing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

6.3 FUNCTIONAL REQUIREMENTS

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

Usability

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

Robustness

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

Security

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

Reliability

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and survive and even capable of working around any failure.

Compatibility

It is supported by version above all web browsers. Using any web servers like localhost makes the system real-time experience.

Flexibility

The flexibility of the project is provided in such a way that is has the ability to run on different environments being executed by different users.

Safety

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one's personal information.

6.4 NON- FUNCTIONAL REQUIREMENTS

Portability

It is the usability of the same software in different environments. The project can be run in any operating system.

Performance

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

Accuracy

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

Maintainability

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyse, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

7 SYSTEM DESIGN AND TESTING PLAN

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

7.1 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

8 SYSTEM STUDY

8.1 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 to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

8.2 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

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. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

8.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

8.4 DATA FLOW DIAGRAM:

- The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

- The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

- DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

- DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

```
                    ┌─────────────────────────┐
                    │       Input data        │
                    └─────────────────────────┘
                                │
                                ▼
                    ┌─────────────────────────┐
                    │     Preprocessing       │
                    └─────────────────────────┘
                                │
                                ▼
                    ┌─────────────────────────┐
                    │    Training dataset     │
                    └─────────────────────────┘
                                │
                                ▼
                    ┌─────────────────────────┐
                    │   Feature Extraction    │
                    └─────────────────────────┘
                                │
                                ▼
          ┌─────────────────────────┐      ┌─────────────────────┐
          │ Prediction/Classification│◄─────│    Testing Data     │
          └─────────────────────────┘      └─────────────────────┘
                                │
                                ▼
                    ┌─────────────────────────┐
                    │   Predicted Results:    │
                    │                         │
                    │  You Are eligible for   │
                    │  loan OR No, You        │
                    │  are not eligible for   │
                    │  loan                   │
                    └─────────────────────────┘
```
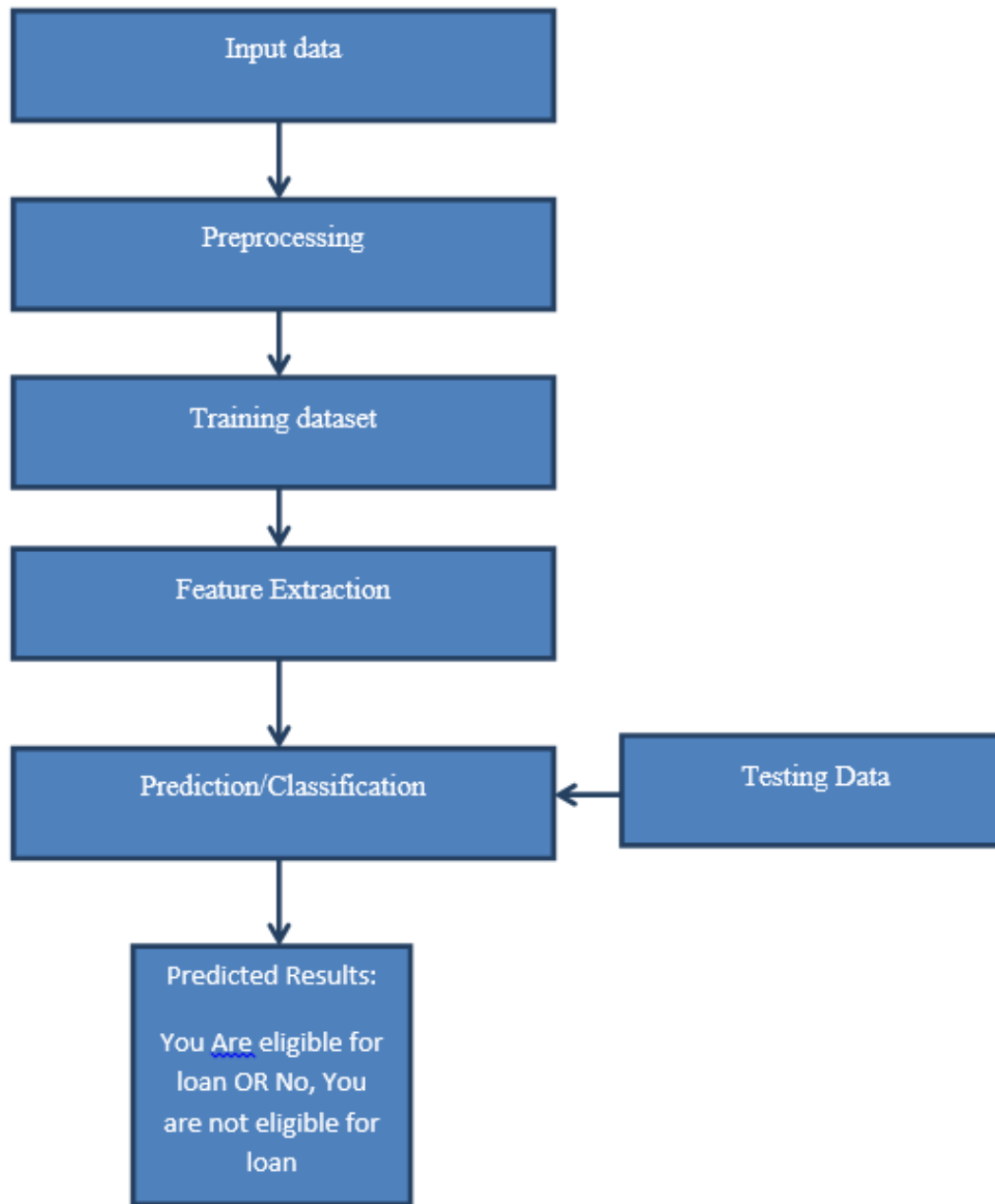
9 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.

- Provide a formal basis for understanding the modeling language.

- Encourage the growth of OO tools market.

- Support higher level development concepts such as collaborations, frameworks, patterns and components.
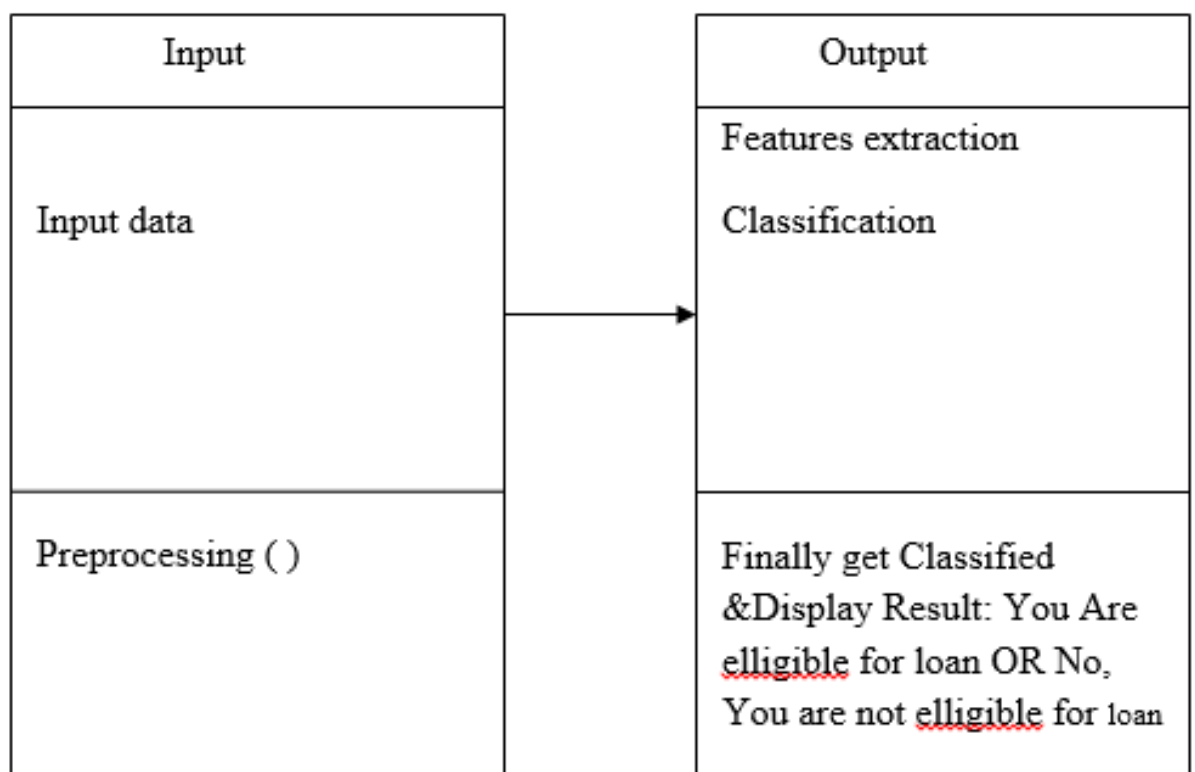
- Integrate best practices.

USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

Input data

Preprocessing

Training

Classification

User
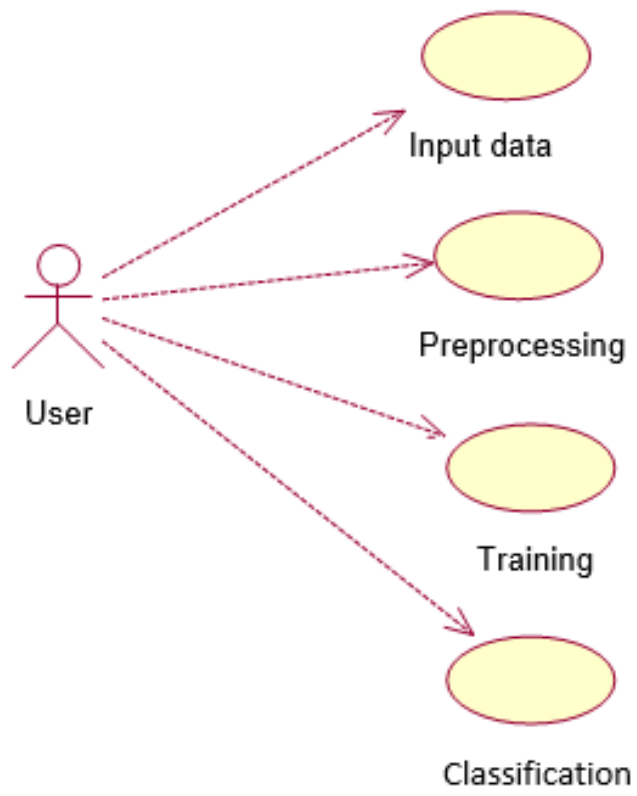
CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

| Input	Output
Input data	Features extraction Classification
Preprocessing ( )	Finally get Classified &Display Result: You Are elligible for loan OR No, You are not elligible for loan

SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.
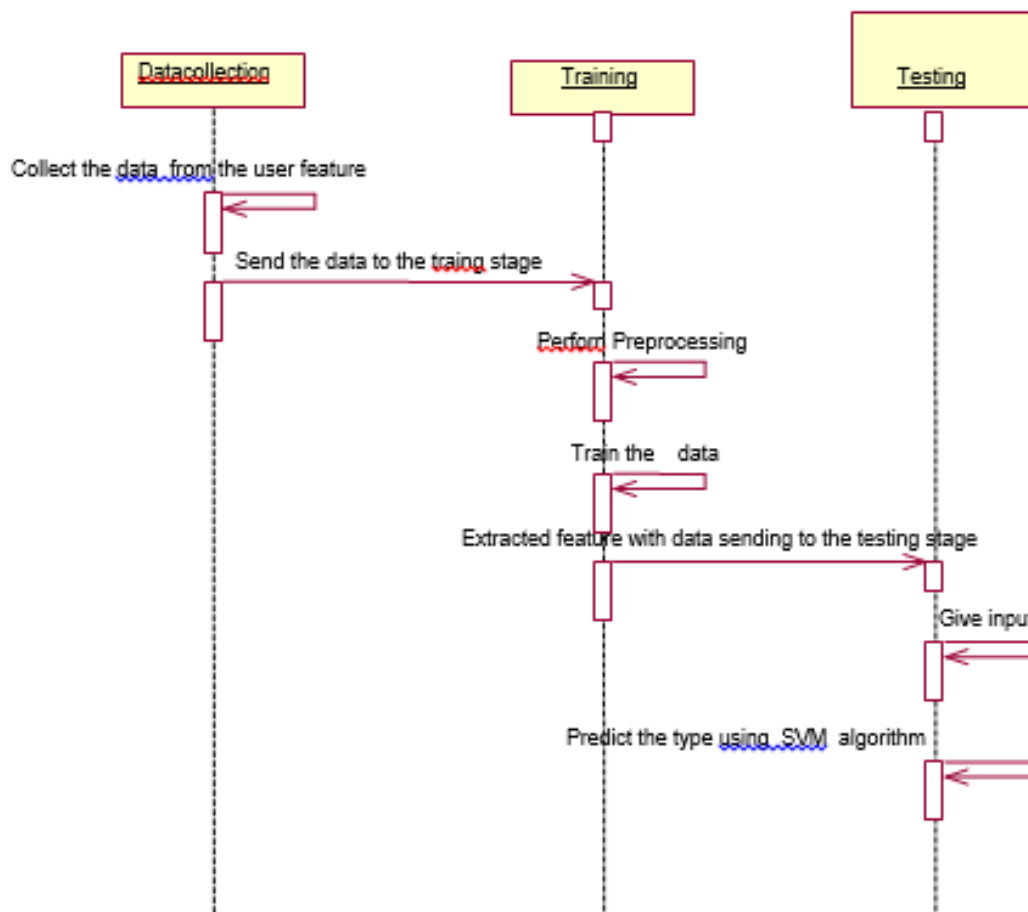
ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.
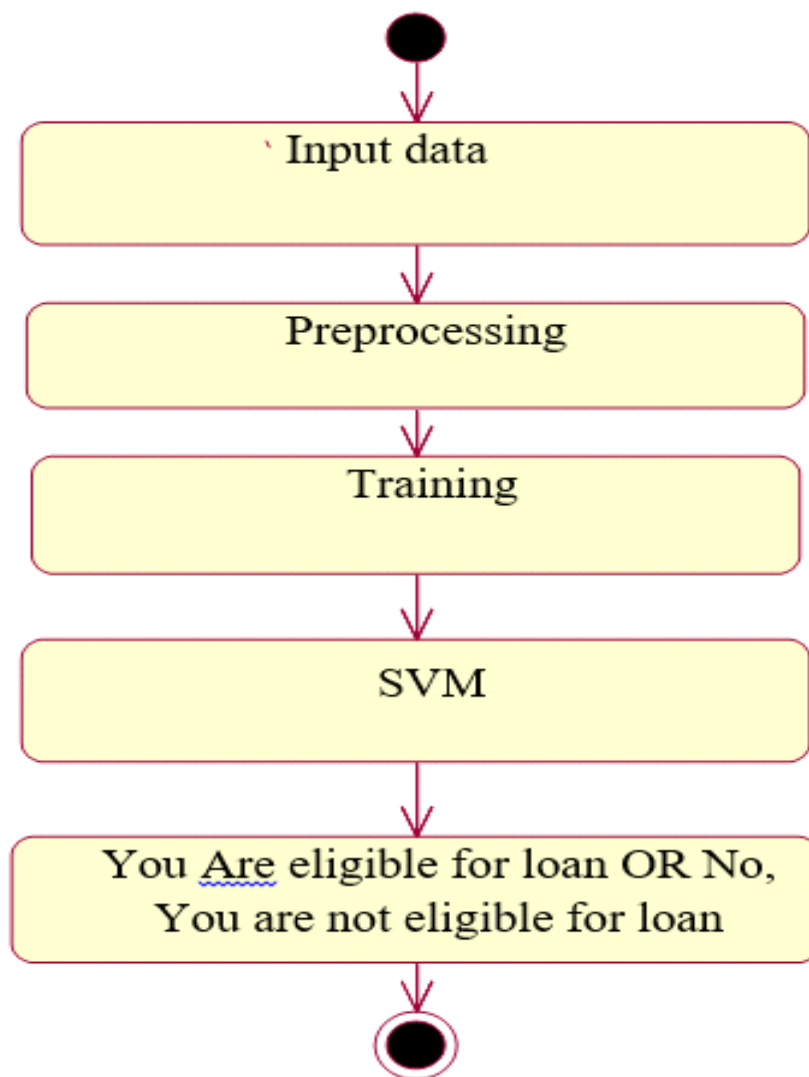
10 SYSTEM TESTING

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 test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

10.1 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 a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component 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.

10.2 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 satisfaction, as shown by successfully 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.

10.3 Functional test

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 identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

10.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

10.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

10.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.

- Pages must be activated from the identified link.

- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format

- No duplicate entries should be allowed

- All links should take the user to the correct page.

6.2 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

6.3 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

11 CONCLUSION

This paper applied machine learning in prediction of loan approval. Support Vector Machine (SVM) is used to predict the loan approval status of customers for bank loans. The results shown that the prediction accuracy is Train Accuracy: 80% and Test Accuracy: 82%. According to this research paper prediction accuracy is good when compared to other existing systems. In some situations like client going through some disaster so here the algorithm cannot predict the appropriate result. This research paper can find out the client is potential or not.

In future a rigorous analysis of other machine learning algorithms other than these can also be done in future to investigate the power of machine learning algorithms for loan approval prediction

12 REFERENCES

[1] Pidikiti Supriya, Myneedi Pavani, Nagarapu Saisushma, Namburi Vimala Kumari, kVikash,"Loan Prediction by using Machine Learning Models", InternationalJournalofEngineering andTechniques.Volume 5 Issue 2, Mar-Apr 2019.

[2] Nikhil Madane, Siddharth Nanda,"Loan Prediction using Decision tree", Journal of the Gujrat Research History, Volume 21 Issue 14s, December 2019.

[3] S. Vimala, K.C. Sharmili, ―Prediction of Loan Risk using NB and Support Vector Machine‖, International Conference on Advancements in Computing Technologies (ICACT 2018), vol. 4, no. 2, pp. 110-113, 2018.

[4] Aboobyda Jafar Hamid and Tarig Mohammed Ahmed, ―Developing Prediction Model of Loan Risk in Banks using Data Mining‖, Machine Learning and Applications: An International Journal (MLAIJ), Vol.3, No.1, pp. 1-9, March 2016.

[5] Kumar Arun, Garg Ishan, Kaur Sanmeet, ―Loan Approval Prediction based on Machine Learning Approach‖, IOSR Journal of Computer Engineering (IOSR-JCE), Vol. 18, Issue 3, pp. 79-81, Ver. I (May-Jun. 2016).