

PROFESSIONAL TRAINING REPORT
at
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology Degree in Information Technology

By
ANIRUDH SRINIVAS G.N
(REG.NO: 37120009)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119. TAMILNADU
SEPTEMBER 2020



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" Grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119.

Phone: 044 - 2450 3150 / 51 / 52 / 54 / 55 Fax: 044 - 2450 2344

www.sathyabama.ac.in



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of ANIRUDH SRINIVAS G N (Reg.no.37120009) who carried out the project entitled "HOME LOAN SANCTION SYSTEM USING MACHINE LEARNING" under our supervision from MAY 2020 to SEPTEMBER 2020.

Internal Guide

Dr. G. Mathivanan B.Sc., B.Tech., M.E., Ph.D.,

Head of the Department

Dr. R. Subhashini M.E.,PhD.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I ANIRUDH SRINIVASG N (Reg. No. 37120009) hereby declare that the Project Report entitled “ HOME LOAN SANCTION SYSTEM USING MACHINE LEARNING ” done by me under the guidance of Dr. G.MATHIVANAN at Sathyabama Institute of Science and Technology (Deemed to be University), Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119 is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge our 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 our thanks to **Dr. T. SASIKALA M.E., PhD., Dean, School of Computing** and **Dr. R. SUBHASHINI M.E., PhD., Head of the Department, Department of Information Technology** for providing us the necessary support and details at the right time during the progressive reviews.

I would like to express our sincere and deep sense of gratitude to our Project **Dr. G. MATHIVANAN B.Sc., B.Tech., M.E., Ph.D.,** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express our thanks to all Teaching and Non-teaching staff members of the Department of **INFORMATION TECHNOLOGY** who were helpful in many ways for the completion of the project.

ONLINE COURSE CERTIFICATES





INTERNSHIP TRAINING CERTIFICATE



ABSTRACT

HOME LOAN SANCTION SYSTEM USING MACHINE LEARNING

Machine learning is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence. We use this in the field of Banking to predict whether or not to sanction loan to a person based on certain criteria.

Customer first applies for home loan and after that the application validates the customer eligibility for loan. The application automates the loan eligibility process (real time) based on customer detail provided while filling online application form. These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others. To automate this process, we use a dataset to identify the customers segments that are eligible for loan amount so that they can specifically target these customers.

By using machine learning algorithms like random forest, decision trees, k nearest neighbours, we train these datasets to create a ML model to predict the eligibility of a person for sanctioning loan. By comparing the accuracy of the three models, we can choose the best model for the home loan sanction system. By automating this process we can eliminate man power, human errors, physical records and save time. The application also uses a responsive user interface that can be understood by all kinds of users.

Name: G.N.Anirudh Srinivas

Guide Name: Dr. G. Mathivanan

CHAPTER NO	TITLE NAME	PAGE NO
	ABSTRACT	VII
	LIST OF FIGURES	IX
	LIST OF ABBREVIATION	X
1	INTRODUCTION	11
	1.1 GENERAL	11
	1.2 OUTLINE OF THE PROJECT	11
2	AIM AND SCOPE OF THE PRESENT INVESTIGATION	12
	2.1 AIM	12
	2.2 PROBLEM STATEMENT	12
	2.3 LITRATURE REVIEW	13
	2.4 SCOPE	14
3	EXPERIMENTAL OR MATERIALS AND METHODS;ALGORITHMS USED	15
	3.1 TOOLS USED	15
	3.2 LIBRARIES AND FRAMEWORKS USED	15
	3.3 ALGORITHMS USED	16
4	RESULT AND DISCUSSION,PERFORMANCE ANALYSIS	18
	4.1 WORKING MODULE	18
	4.2 RESULTS	21
	4.3 PERFORMANCE ANALYSIS	21
	4.4 MINIMUM REQUIREMENTS	26
5	SUMMARY AND CONCLUSION	27
	5.1 CONCLUSION	27
	5.2 FUTURE SCOPE	27
	REFERENCES	28
	APPENDIX	28

LIST OF ABBREVIATIONS

ABBREVIATIONS	EXPANSION
IDE	INTEGRATED DEVELOPMENT ENVIRONMENT
ML	MACHINE LEARNING
UI	USER INTERFACE
PY	PYTHON
CSV	COMMA-SEPERATED VALUES
HTML	HYPERTEXT MARKUP LANGUAGE

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

1.2 OUTLINE OF THE PROJECT

Applying Random forest algorithm in Machine learning model to train a dataset containing details like Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others. Finally with a user interface the system can check whether a person is eligible for a loan by getting the required input from the user.

CHAPTER 2

AIM AND SCOPE OF THE PRESENT INVESTIGATION

2.1 AIM

We use Machine learning in the field of Banking to predict whether or not to sanction loan to a person based on certain criteria.

2.2 PROBLEM STATEMENT

Customer first applies for home loan and after that the application validates the customer eligibility for loan. The application automates the loan eligibility process (real time) based on customer detail provided while filling online application form. These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others. To automate this process, we use a dataset to identify the customers segments that are eligible for

loan amount so that they can specifically target these customers.

2.3 LITERATURE STUDY

2.3.1 MACHINE LEARNING

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

2.3.2 PYTHON

Python is an interpreted, high-level and general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help

programmers write clear, logical code for small and large-scale projects.

2.3.3 IDE (INTEGRATED DEVELOPMENT ENVIRONMENT)

IDE stands for Integrated Development Environment. It's a coding tool which allows you to write, test, and debug your code in an easier way, as they typically offer code completion or code insight by highlighting, resource management, debugging tools ,... And even though the IDE is a strictly defined concept, it's starting to be redefined as other tools such as notebooks start gaining more and more features that traditionally belong to IDEs. The IDE used in this project is Anaconda Navigator.

2.3.4 USER INTERFACE

The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, a mouse and the appearance of a

desktop. It is also the way through which a user interacts with an application or a website.

2.3.5 PYTHON WEB FRAMEWORK

A Web framework is a collection of packages or modules which allow developers to write Web applications or services without having to handle such low-level details as protocols, sockets or process/thread management. The Python web framework used in this project is Flask.

2.4 SCOPE

This is a python web app that runs based on a machine learning model that checks whether a person is eligible for a home loan or not based on certain criteria. The model is trained using a dataset consisting of 614 rows and implemented using random forest algorithm. It can be used to automate the process of checking the eligibility manually in banks.

CHAPTER 3

METHODS AND ALGORITHMS USED

3.1 TOOLS USED

- Python 3
- Anaconda
- Flask
- Bootstrap

3.2 LIBRARIES AND FRAMEWORKS USED

3.2.1 FLASK

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular

tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

3.2.2 SCIKIT-LEARN

Scikit-learn (formerly scikits.learn and also known as sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

3.2.3 PANDAS

In computer programming, pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.

3.2.4 PICKLE

Python pickle module is used for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk. What pickle does is that it “serializes” the object first before writing it to file. Pickling is a way to convert a python object (list, dict, etc.)

3.2.5 BOOTSTRAP

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

3.3 IDE USED

3.3.1 ANACONDA

Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing, that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. It is an integrated development

environment developed by data scientists and is best suitable for Machine learning and Data science.

3.3 ALGORITHMS USED

3.3.1 RANDOM FOREST CLASSIFIER

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean prediction of the individual trees.

3.3.2 MIN-MAX SCALER

The Min-MaxScaler is the probably the most famous scaling algorithm, and follows the following formula for each feature: $\frac{x_i - \min(x)}{\max(x) - \min(x)}$ It essentially shrinks the range such that the range is now between 0 and 1 (or -1 to 1 if there are negative values). It is a preprocessing and feature scaling method inbuilt in Scikit-learn library.

3.4 ALGORITHM

Initially the Dataset is read using Pandas library. Then the necessary features are extracted.

```
features=['Married','Education','Self_Employed','ApplicantIncome','CoapplicantIncome','LoanAmount','Loan_Amount_Term','Credit_History','Property_Area']
```

The American currency in the dataset is converted to Indian rupee by multiplying it with 74.81.

To convert the yes or no to 0 or 1 in some features we use this:

```
X=pd.get_dummies(X,drop_first=True)
```

To fill the null values in the data we use:

```
fillna(value=0)
```

```
value_counts().idxmax()
```

Min-Max Scaler is then used to scale the values between the range 0 and 1:

```
sc=MinMaxScaler(feature_range = (0,1)).fit(X.values)
X_train=sc.transform(X.values)
```

Parameters used in random forest, random forest is implemented.

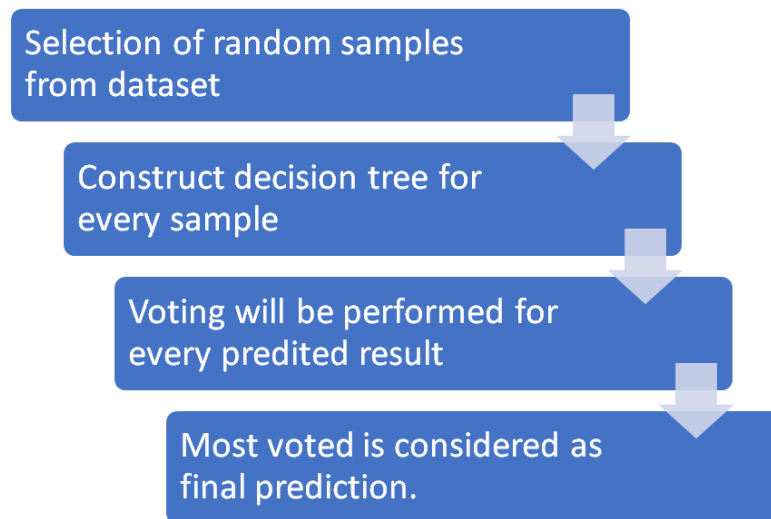
```
n_estimators=[int(x) for x in
np.linspace(start=100,stop=1200,num=12)]
max_features=['auto','sqrt']
max_depth=[int(x) for x in
np.linspace(start=1,stop=50,num=50)]
min_samples_split=[int(x) for x in
np.linspace(start=1,stop=50,num=50)]
min_samples_leaf=[int(x) for x in
np.linspace(start=1,stop=50,num=50)]
oob_score=['True','False']
```

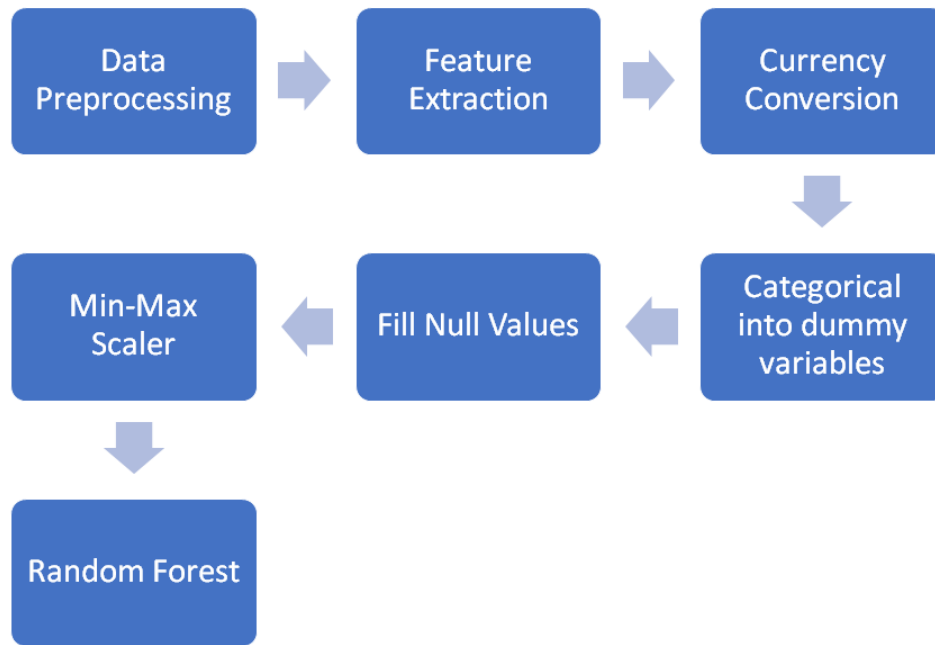
hyperparameter optimization or tuning is the problem of choosing a set of optimal hyperparameters for a learning algorithm.

```
rf_random = RandomizedSearchCV(estimator = rf,  
param_distributions=grid,scoring='accuracy',n_iter=10,cv =  
5,random_state=42,verbose=2,n_jobs = 1)
```

The algorithm is trained with 81% accuracy.

3.5 FLOW OF MODULE





CHAPTER 4

RESULT AND DISCUSSION, PERFORMANCE ANALYSIS

4.1 WORKING MODULE

The command prompt is opened and the virtual environment containing all the packages is activated. The app directory is opened in the CMD. We can open the app by running the command “python app.py”. The user interface developed using bootstrap is then opened.

4.1.1 USER INTERFACE

The user interface consists of the input fields for the following.

- Applicant's Annual Income
- Co-Applicant's Annual Income
- Loan amount
- Loan term
- Credit History
- Marital Status
- Graduate
- Self Employed
- Property Area

By clicking the predict status button after filling the above details, the python model checks the details and print the result whether or not to sanction loan to the person based on the trained model.

4.2 RESULT

Home Loan Sanction System

Applicant's Annual Income

Enter amount in numbers

Co-Applicant's Annual Income

Enter amount in numbers

Loan Amount

Enter amount in numbers

Loan Term (In months)

Enter in numbers

Credit History

Yes



Marital Status

Unmarried



Graduate

Yes



Self Employed

Yes



Property Area

Urban



Predict Status

Home Loan Sanction System

Applicant's Annual Income	60000
Co-Applicant's Annual Income	20000
Loan Amount	100000
Loan Term (in months)	30
Credit History	Yes
Marital Status	Married
Graduate	Yes
Self Employed	Yes
Property Area	Semi-Urban
Predict Status	

your loan can be approved

Home Loan Sanction System

Applicant's Annual Income	20000
Co-Applicant's Annual Income	0
Loan Amount	300000
Loan Term (in months)	20
Credit History	No
Marital Status	Unmarried
Graduate	No
Self Employed	No
Property Area	Urban
Predict Status	

your loan cannot be approved

4.3 PERFORMANCE ANALYSIS

TEST CASE	EXPECTED RESULT
Activate virtual environment	PASS
Open Directory	PASS
Run app.py	PASS
Open UI on local host	PASS
Accept user input and submit	PASS
Loan Approval status	PASS
Loan Denial status	PASS

4.4 MINIMUM REQUIREMENTS

Operating System: Windows/Mac

RAM: 4GB

Processor: 64x 1.0Ghz processor

ROM: 8GB

CHAPTER 5

CONCLUSION AND SUMMARY

5.1 CONCLUSION

The Python app for predicting whether or not to sanction loan to a person using certain criteria is developed using Python, Anaconda, Flask, Machine Learning. It is packaged into web application to be available for all users.

5.2 FUTURE SCOPE

The Banking Sector Industry are sooner to be influenced by Robotic process automation. With careful planning and systematic monitoring , home loan sanction system using machine learning can be a passable alternatively, in case of reducing manual errors and human resource.

REFERENCES:

[www.analyticsvidhya](http://www.analyticsvidhya.com)

www.bootstrap.com

www.anaconda.com

APPENDIX:

```
import numpy as np

import pandas as pd

df=pd.read_csv('loan_train.csv')

print(df.isnull().sum())

print(df)

features=['Married','Education','Self_Employed','ApplicantIncome','CoapplicantIncome','LoanAmount','Loan_Amount_Term','Credit_History','Property_Area']

df['LoanAmount']=df['LoanAmount']*1000*74.81

df['ApplicantIncome']=df['ApplicantIncome']*74.81

df['CoapplicantIncome']=df['CoapplicantIncome']*74.81

X=df[features]

Y=df['Loan_Status']

print(df['LoanAmount'].mean())

X['Married_Yes']=X['Married_Yes'].fillna(value=0)

X['Self_Employed_Yes']=X['Self_Employed_Yes'].fillna(value=0)

X['LoanAmount']=X['LoanAmount'].fillna(value=df['LoanAmount'].value_counts().idxmax())

X['Loan_Amount_Term']=X['Loan_Amount_Term'].fillna(value=X['Loan_Amount_Term'].value_counts().idxmax())

X['Credit_History']=X['Credit_History'].fillna(value=X['Credit_History'].value_counts().idxmax())

print(X.isnull().sum())
```

```

from sklearn.preprocessing import MinMaxScaler
sc=MinMaxScaler(feature_range = (0,1)).fit(X.values)
X_train=sc.transform(X.values)
print(X_train)
values={'Y':0,'N':1}
Y=df.Loan_Status.map(values)
print(Y.values)

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import RandomizedSearchCV
rf=RandomForestClassifier()
n_estimators=[int(x) for x in np.linspace(start=100,stop=1200,num=12)]
max_features=['auto','sqrt']
max_depth=[int(x) for x in np.linspace(start=1,stop=50,num=50)]
min_samples_split=[int(x) for x in np.linspace(start=1,stop=50,num=50)]
min_samples_leaf=[int(x) for x in np.linspace(start=1,stop=50,num=50)]
oob_score=['True','False']
grid={'n_estimators':n_estimators,'max_features':max_features,'max_depth':max_depth,'min_samples_
split':min_samples_split,'min_samples_leaf':min_samples_leaf,'oob_score':oob_score}
print(grid)

rf_random = RandomizedSearchCV(estimator = rf,
param_distributions=grid,scoring='accuracy',n_iter=10,cv = 5,random_state=42,verbose=2,n_jobs = 1)
rf_random.fit(X_train,Y)
rf_random.best_params_
rf_random.best_score

from sklearn.metrics import classification_report,confusion_matrix
preds=rf_random.predict(X)
print(classification_report(preds,Y))
print(confusion_matrix(preds,Y))
import pickle

```

```
with open('loan_classifier.pkl','wb') as f:  
    pickle.dump(rf_random,f)
```