

STUDENT PERFORMANCE PREDICTION USING MACHINE LEARNING



Major project submitted in the partial fulfillment of the requirements for the award of
the degree of

BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING

By

**G.Lavanya
16K91A05D9**

**Under the guidance of
Mr. K. Shiva Shankar Goud
Assistant Professor**

**DEPARTMENTMENT OF COMPUTER SCIENCE & ENGINEERING
TKR COLLEGE OF ENGINEERING & TECHNOLOGY
(Accredited by NBA and NAAC with 'A' Grade)
Medbowli, Meerpet, Saroornagar, Hyderabad-500097**

DECLARATION BY THE CANDIDATE

I, **Ms. G.Lavanya** bearing Hall Ticket Number: **16K91A05D9**, hereby declare that the project report titled **Student Performance Prediction Using Machine Learning** under the guidance of **Mr. K. Shiva Shankar Goud**, Assistant Professor in Department of Computer Science & Engineering is submitted in partial fulfillment of the requirements for the award of the degree of ***Bachelor of Technology in Computer Science & Engineering***.

G.Lavanya,
H.T. No.:16K91A05D9.

CERTIFICATE

This is to certify that the project report entitled “**Student Performance Prediction Using Machine Learning**”, being submitted by **Ms. G.Lavanya**, bearing **Roll.No.:16K91A05D9**, in partial fulfillment of requirements for the award of degree of **Bachelor of Technology in Computer Science & Engineering**, to the Jawaharlal Nehru Technological University is a record of bonafide work carried out by them under my guidance and supervision.

Signature of the Guide
K. Shiva Shankar Goud
Assistant Professor

Signature of the HOD
Dr.A.Suresh Rao
Professor

Signature of the External Examiner

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragement and guidance have crowned my efforts with success.

I am indebted to the **Internal Guide, Mr. K. Shiva Shankar Goud**, Assistant Professor, Dept. of Computer Science & Engineering, TKR College of Engineering & Technology, for his support and guidance throughout my Major Project.

I am also indebted to the **Head of the Department, Dr. A. Suresh Rao**, Professor, Computer Science & Engineering, TKR College of Engineering & Technology, for his support and guidance throughout my Major Project.

I extend my deep sense of gratitude to the **Principal, Dr. D. V. Ravi Shankar**, TKR College of Engineering & Technology, for permitting me to undertake this Major Project.

Finally, I express my thanks to one and all that have helped us in successfully completing this Major Project. Furthermore, I would like to thank my family and friends for their moral support and encouragement.

By,
G.Lavanya,
16K91A05D9.

CONTENTS

Abstract	i
List of Figures	ii
List of Tables	iii
List of Screen	iv
S.No. Topic name	Pg.No.
1. INTRODUCTION	01
1.1 Motivation	01
1.2 Problem definition	01
1.3 Limitations of existing system	02
1.4 Proposed system	02
2. LITERATURE SURVEY	03
2.1 Analysis	05
3. DESIGN	08
3.1 System architecture	08
3.2 Algorithm	15
4. IMPLEMENTATION & RESULTS	18
4.1 Explanation of Key functions	20
4.2 Method of Implementation	25
4.2.1 Forms	28
4.2.2 Output Screens	29
4.2.3 Result Analysis	36
5. TESTING & VALIDATION	37
5.1 Design of test cases and scenarios	39
5.2 Validation	40
5.2 Conclusion	40
6. CONCLUSION	41
7. REFERENCE	42

ABSTRACT

One of the most challenging tasks in the education sector in India is to predict student's academic performance due to a huge volume of student data. In the Indian context, we don't have any existing system by which analyzing and monitoring can be done to check the progress and performance of the student mostly in Higher education system. Every institution has their own criteria for analyzing the performance of the students. The reason for this happening is due to the lack of study on existing prediction techniques and hence to find the best prediction methodology for predicting the student academics progress and performance. Another important reason is the lack in investigating the suitable factors which affect the academic performance and achievement of the student in particular course. So to deeply understand the problem, a detail literature survey on predicting student's performance using data mining techniques is proposed. The main objective of this project is to provide a great knowledge and understanding of different data mining techniques which have been used to predict the student progress and performance and hence how these prediction techniques help to find the most important student attribute for prediction. Actually, we want to improve the performance of the student in academic by using best data mining techniques. At last, it could also provide some benefits for faculties, students, educators and management of the institution..

LIST OF FIGURES

S. No.	Figure Name	Page no.
1.	Figure 3.1.1:Block Diagram	08
2.	Figure 3.1.1:Activity Diagram	10
3.	Figure 3.1.2:Sequence Diagram	11
4.	Figure 3.1.3: Class Diagram	12
5.	Figure 3.1.4:Data Flow Diagram	13
6.	Figure 3.1.5: UseCase Diagram	14
7.	Figure 4.1:Accuracy	22
8.	Figure 4.2.1:Dataset-1	26
9.	Figure 4.2.2:Dataset-2	26

LIST OF TABLES

T. No.	Table Name	Page no.
1.	Table: 2.1.1 Hardware Requirements	05
2.	Table: 2.1.2 Software Requirements	05

LIST OF SCREENS

S. No.	Screen Name	Page no.
1.	Figure 4.2.1:Home Screen	28
2.	Figure 4.2.2:Uploading Dataset	29
3.	Figure 4.2.2:Dataset	29
4.	Figure 4.2.2:Saving Dataset	30
5.	Figure 4.2.2:Output Screen-1	30
6.	Figure 4.2.2:Output Screen-2	31
7.	Figure 4.2.2:Output Screen-3	31
8.	Figure 4.2.2:Output Screen-4	32
9.	Figure 4.2.2:Output Screen-5	32
10.	Figure 4.2.2:Output Screen-6	33
11.	Figure 4.2.2:Output Screen-7	33
12.	Figure 4.2.2:Output Screen-8	34
13.	Figure 4.2.2:Output Screen-9	34
14.	Figure 4.2.2:Output Screen-10	35
15.	Figure 4.2.2:Output Screen-11	35

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION

For higher education institutions whose goal is to contribute to the improvement of quality of higher education. The quality of higher education institutions implies providing the services, which most likely meet the needs of students, academic staff, and other participants in the education system. There are many studies in the learning field that investigated the ways of applying machine learning techniques for various educational purposes. One of the focuses of these studies is to identify high-risk students, as well as to identify features which affect the performance of students. A high prediction accuracy of the student's performance is helpful to identify the low performance students at the beginning of the learning process.

1.2 PROBLEM DEFINITION:

The ability to predict performance of students is very crucial in our present education system. We applied Machine learning concepts for this study. Machine learning techniques are used to discover models or patterns of data, and it is helpful in the decision-making.

1.2.1 OBJECTIVE OF THE PROJECT:

Our main objectives to this proposed work are:

1. To understand, analyse and then find the difference between different prediction techniques of data mining in education.
2. To identify and understand different student attributes which are mainly used for the predicting the student performance.
3. To identify and understand the different prediction techniques which are mainly used for predicting the student performance.

1.3 LIMITATIONS OF EXISTING SYSTEM:

1. One of the most challenging tasks in the education sector in India is to predict student's academic performance due to a huge volume of student data. In the Indian context, we don't have any existing system by which analyzing and monitoring can be done to check the progress and performance of the student mostly in Higher education system.
2. Every institution has their own criteria for analyzing the performance of the students. The reason for this happening is due to the lack of study on existing prediction techniques and hence to find the best prediction methodology for predicting the student academics progress and performance.
3. Another important reason is the lack in investigating the suitable factors which affect the academic performance and achievement of the student in particular course.

1.4 PROPOSED SYSTEM:

Present system takes various inputs and creates a model that will predict the student performance at school or university levels.

Our main goals were as follows:

- 1.To understand, analyse and then find the difference between different prediction techniques of data mining in education.
- 2.To identify and understand different student attributes which are mainly used for the predicting the student performance.
- 3.To identify and understand the different prediction techniques which are mainly used for predicting the student performance.

CHAPTER 2

LITERATURE SURVEY

In Indian education system checking student's performance is a very essential in higher education. But we don't have any fixed criteria to evaluate the student performance. Some institutions student performance can be observed by using internal assessment and co-curriculum.

In the Indian context, an institution with the higher degree of reputation using the good academic record as its basic criteria for their admissions . There are lots of definitions of student academic performance prediction should be given in the literature. Different authors are using different student factors/attributes for analyzing student performance.

Most of the author used CGPA, Internal assessment, External assessment, Examination final score and extra co-circular activities of the student as prediction criteria. Most of the Indian institution and universities using final examination grade of the student as the student academic performance criteria. The final grades of any student depend on different attributes like internal assessment, external assessment, laboratory file work and viva-voice, sessional test. The performance of the student depends upon how many grades a student score in the final examination. Norlida Buniyamin, Pauziah Mohd Arsad et al. (2013) stated that what are the significance of academic analytics for an educational institution and how they work for the improvement of education. They also proposed an intelligent recommendation intervention system to improve the student's performance and achievement in education.

This system uses two different student attribute to measure the achievement and that is student grade and student information . Zaidah Ibrahim and Daliela Rusli et al. (2007) stated that predicting student's performance is very critical for any educational institution because it is important for the formation of new rule and standards for the improvement of the education and reputation. They used CGPA and demographic attributes of the first year student to predict their result in the first year of education in engineering .

Data mining techniques which are used in mostly education are known as Educational data mining. There are lots of data mining techniques are available to predict the student performance.

Education data mining help to find the hidden information from a huge database of education setting, because at present lots of data are generated in educational institution related to student .

Further, this hidden information can be used for performance, dropout and final result prediction of the student. It also helps the educator, management and faculties to work according to the learning standards of the students. Actually data mining help in the different field of education sector . So to properly understand the real meaning of the data mining in education we need to do a systematic literature review on different work done by the different researcher.

2.1 ANALYSIS:

The goal of system analysis is to determine where the problem is in an attempt to fix the system. This step involves breaking down the system in different pieces to analyze the situation, analyzing project goal, breaking down what needs to be created and attempting to engage users so that definite requirements can be defined.

2.1.1 Hardware and Software Requirements:

Hardware Requirements:

The selection of hardware is very important in the existence and proper working of any software. In the selection of hardware, the size and the capacity requirements are also important.

Processor	Any processor speed with 1.1GHz(min)
RAM	1GB(min)
Hard Disk capacity	40GB(Min)

Table: 2.1.1 Hardware Requirements

Software Requirements:

One of the most difficult tasks is the selection of the software, once system requirement is known that is determining whether a particular software package fits the requirements.

Operating System	Windows 10
Programming Language	Python
Environment	Anaconda Spyder 3

Table: 2.1.2 Software Requirements

2.2 Functional Requirements:

Functional requirements are associated with specific functions, tasks or behaviour the system must support. It define what a system is supposed to do.

Functional requirements describe a system's capabilities and services. They are:

1. Data analysis:

Data analysis is a process of inspecting, cleaning, transforming and modeling data with the goal of discovering useful information, informing conclusion and supporting decision making. Here we analyze the datasets to summarize their main characteristics often with visual methods. Data analysis in sports has reached an important level.

2. Data visualization:

It is graphical representation of data by using visual elements like charts, maps and graphs. Here, we have the knowledge on sports domain and attributes.

2.3 Non-Functional Requirements:

Non-functional requirements are requirements that specify criteria that can be used to judge the operation of a system, rather than specific behaviors. It defines how a system is supposed to be.

Non-functional requirements describe the properties of capabilities and the desired level of services. They are:

1. Performance:

Performance defines how fast the system or its particular piece responds to certain users' actions under certain workload. In most cases, this metric explains how much a user must wait before the target operation happens.

2. Scalability:

Scalability assesses the highest workloads under which the system will still meet the performance requirements.

3. Reliability:

Reliability is a quality attribute that specifies how likely the system or its element would run without a failure for a given period of time under predefined conditions. Traditionally, it's expressed as a probability percentage.

4. Security:

Security assures that all data inside the system or its part will be protected against malware attacks or unauthorized access.

5. Robustness:

Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance.

CHAPTER 3

DESIGN

INTRODUCTION:

Software design is the process by which an agent creates a specification of a software artifact, intended to accomplish goals, using a set of primitive components and subject to constraints. Software design may refer to either "all the activity involved in conceptualizing, framing, implementing, commissioning, and ultimately modifying complex systems". Software design usually involves problem solving and planning a software solution. This includes both a low-level component and algorithm design and a highlevel, architecture design.

3.1SYSTEM ARCHITECTURE

The block diagram is typically used for a higher level, less detailed description aimed more at understanding the overall concepts and less at understanding the details of implementation.

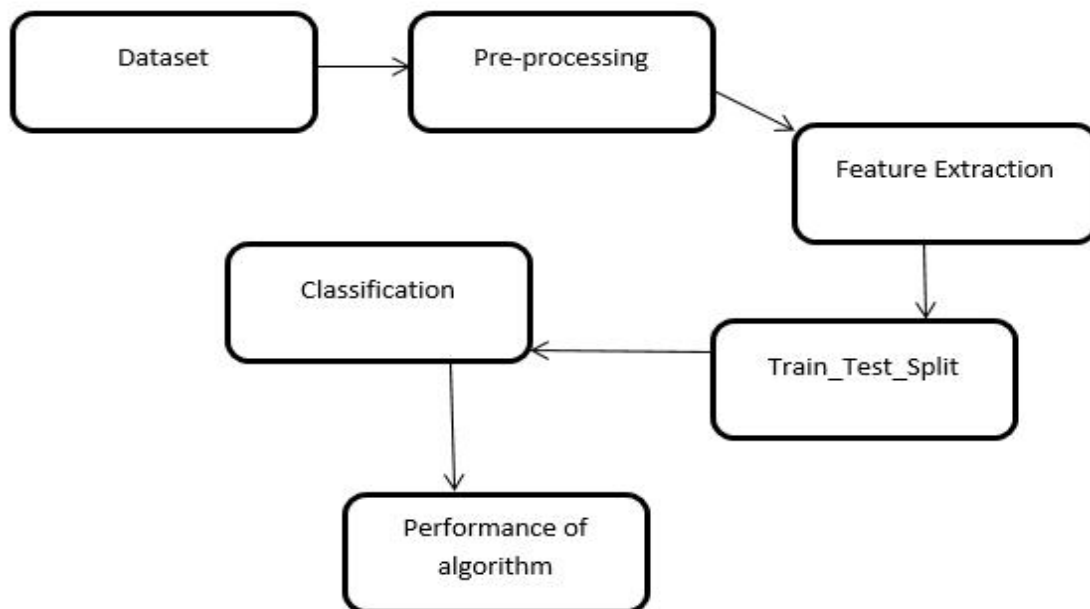


Figure 3.1.1:Block Diagram

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 UML 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 very important part of developing object-oriented software and software development process. UML mostly uses graphical notations to express the design of software objects.

GOALS:

The primary goals in the design of design of UML are: -

1. Be independent of programming languages and development process.
2. Provide a formal basis for understanding the modelling language.
3. Provide extensibility and specialization mechanisms to extend the core concepts
4. Integrate best practices.

3.1.1 ACTIVITY DIAGRAM:

These are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the UML, activity diagrams are used to describe the business and operational step-by-step workflows of components in a system. Activity diagrams show overall flow of control.

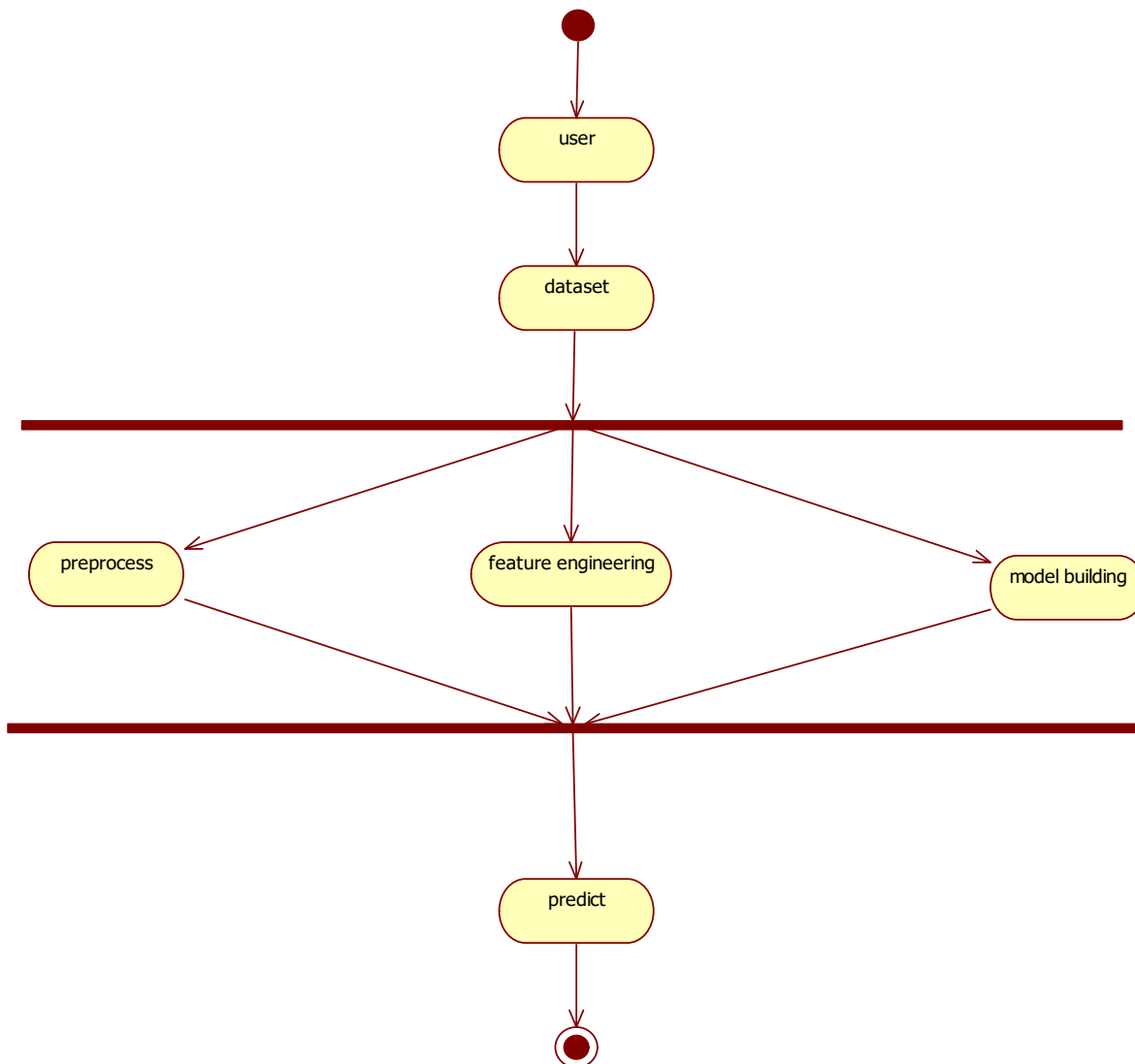


Figure 3.1.1:Activity Diagram

3.1.2 SEQUENCE DIAGRAM:

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart.

A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

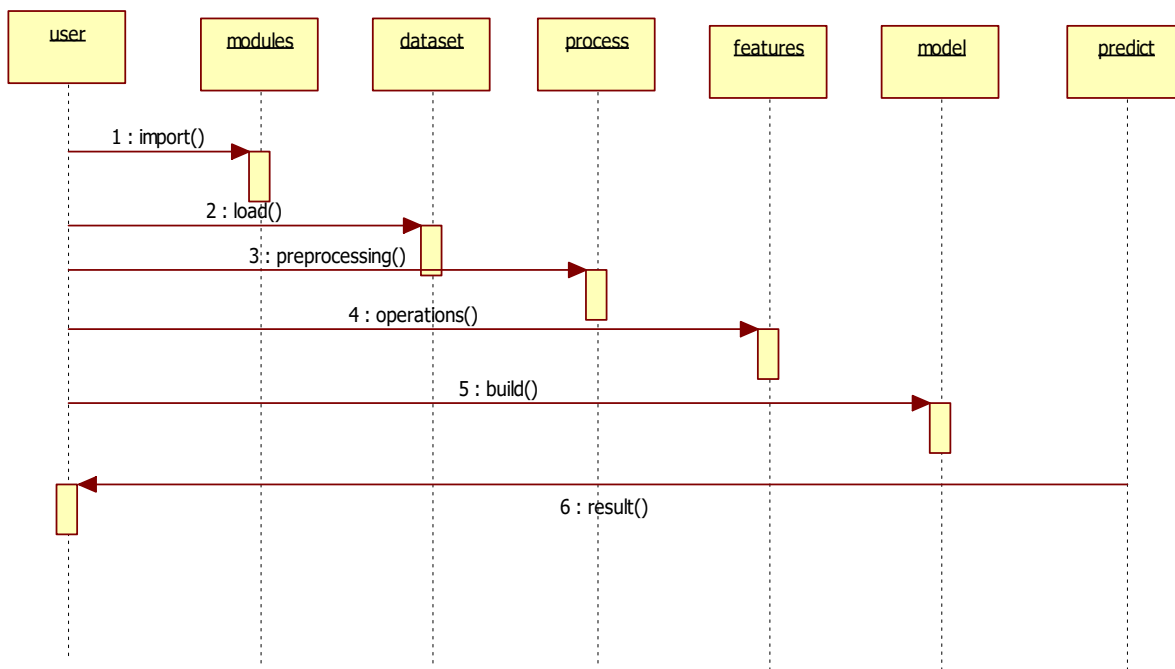


Figure 3.1.2:Sequence Diagram

3.1.3 CLASS DIAGRAM:

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the systematic of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

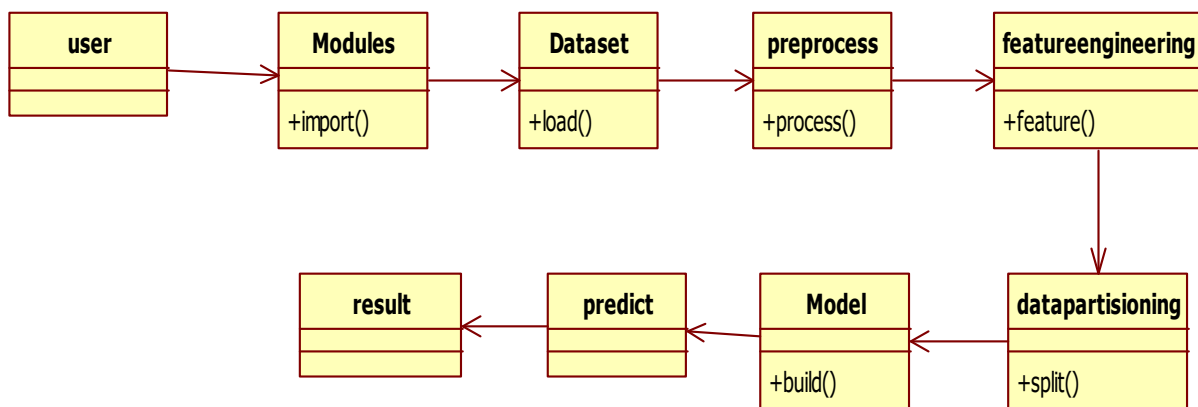


Figure 3.1.3: Class Diagram

3.1.4 Data flow Diagram:

Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation.

Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow.

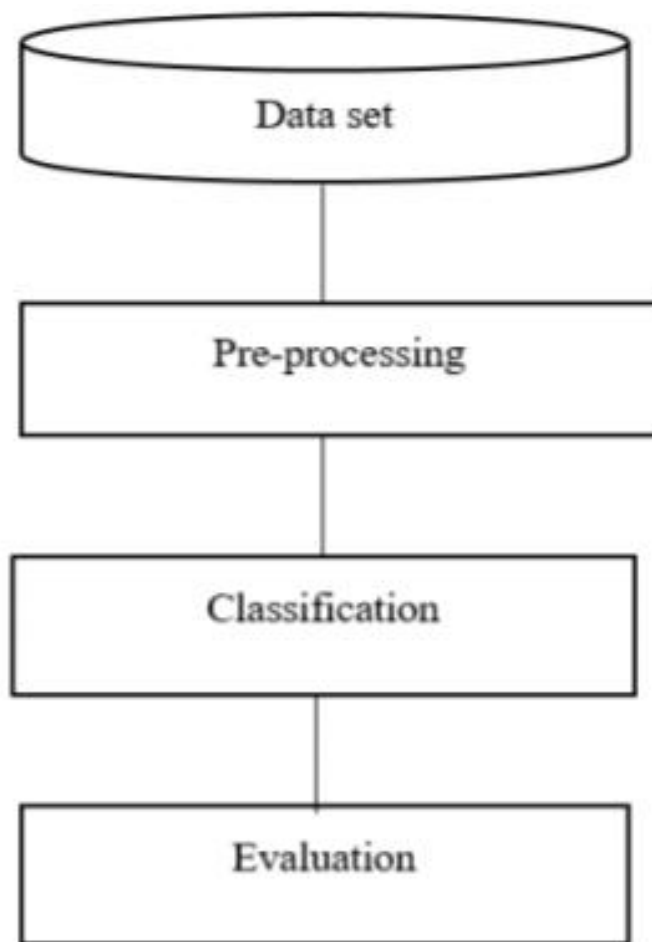


Figure 3.1.4:Data Flow Diagram

3.1.5 USE CASE DIAGRAM:

Use case diagrams are one of the five diagrams in the UML for modelling the dynamic aspects of the systems (activity diagrams, sequence diagram, state chart diagram, collaboration diagram are the four other kinds of diagrams in the UML for modelling the dynamic aspects of systems). Use case diagrams are central to modelling the behaviour of the system, a sub-system, or a class. Each one shows a set of use cases and actors and relations.

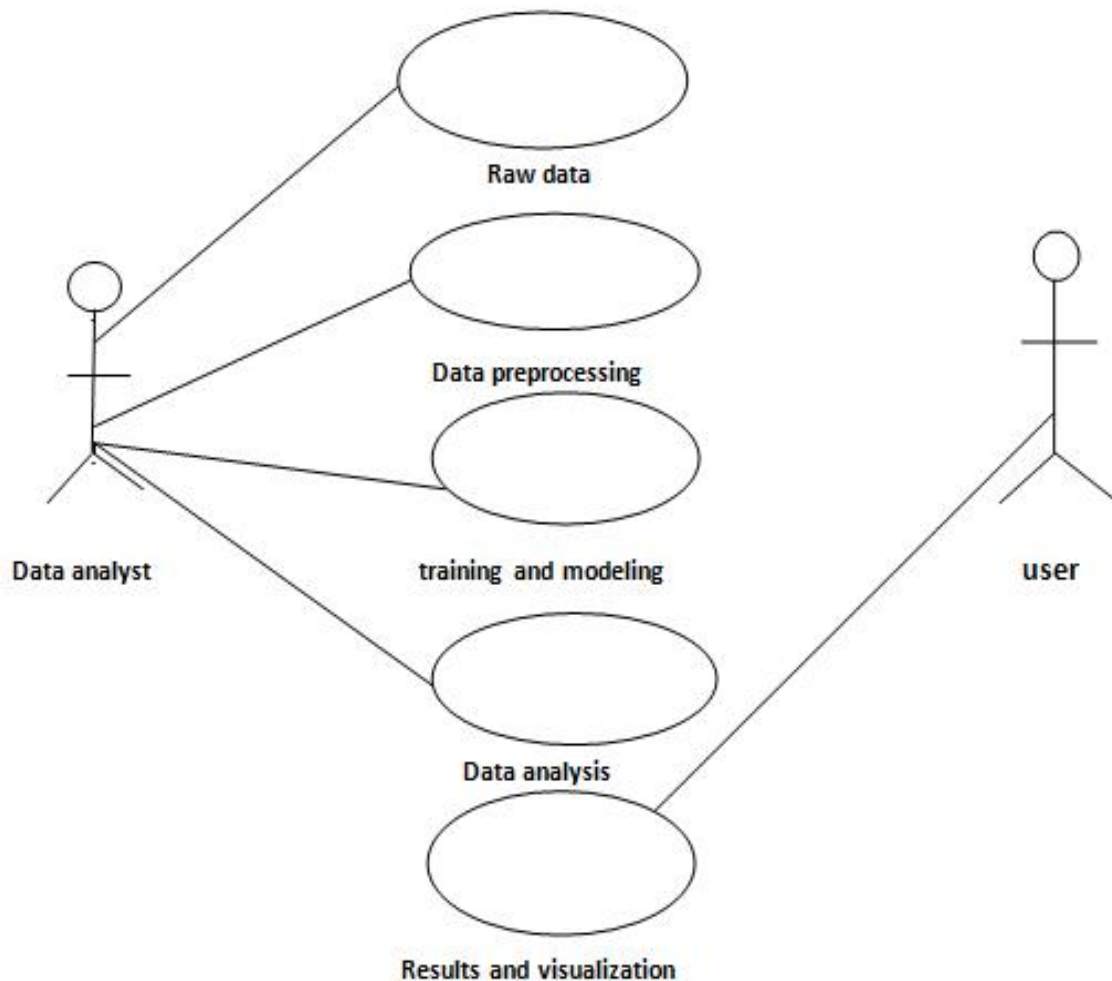


Figure 3.1.5: UseCase Diagram

3.2 ALGORITHM

1. Stochastic Gradient Descent (SGD):

The word ‘stochastic’ means a system or a process that is linked with a random probability. Hence, in Stochastic Gradient Descent, a few samples are selected randomly instead of the whole data set for each iteration. In Gradient Descent, there is a term called “batch” which denotes the total number of samples from a dataset that is used for calculating the gradient for each iteration. In typical Gradient Descent optimization, like Batch Gradient Descent, the batch is taken to be the whole dataset. Although, using the whole dataset is really useful for getting to the minima in a less noisy or less random manner, but the problem arises when our datasets get really huge.

Suppose, you have a million samples in your dataset, so if you use a typical Gradient Descent optimization technique, you will have to use all of the one million samples for completing one iteration while performing the Gradient Descent, and it has to be done for every iteration until the minima is reached. Hence, it becomes computationally very expensive to perform.

This problem is solved by Stochastic Gradient Descent. In SGD, it uses only a single sample, i.e., a batch size of one, to perform each iteration. The sample is randomly shuffled and selected for performing the iteration.

for i in range(m):

$$\theta_j = \theta_j - \alpha (\hat{y}^i - y^i) x_j^i$$

So, in SGD, we find out the gradient of the cost function of a single example at each iteration instead of the sum of the gradient of the cost function of all the examples.

In SGD, since only one sample from the dataset is chosen at random for each iteration, the path taken by the algorithm to reach the minima is usually noisier than your typical Gradient Descent algorithm. But that doesn’t matter all that much because the path taken by the algorithm does not matter, as long as we reach the minima and with significantly shorter training time.

This cycle of taking the values and adjusting them based on different parameters in order to reduce the loss function is called **back-propagation**.

2.SUPPORT VECTOR MACHINE

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

Types of SVM

SVM can be of two types:

1. **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
2. **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

3. RANDOM FOREST

There are two stages in Random Forest algorithm, one is random forest creation, the other is to make a prediction from the random forest classifier created in the first stage. The whole process is shown below, and it's easy to understand using the figure.

Random Forest creation pseudocode:

1. Randomly select "**K**" features from total "**m**" features where $k \ll m$
2. Among the "**K**" features, calculate the node "**d**" using the best split point
3. Split the node into **daughter nodes** using the **best split**
4. Repeat the **a to c** steps until "l" number of nodes has been reached
5. Build forest by repeating steps **a to d** for "n" number times to create "**n**" **number of trees**

In the next stage, with the random forest classifier created, we will make the prediction. The random forest prediction pseudocode is shown below:

1. Takes the **test features** and use the rules of each randomly created decision tree to predict the outcome and stores the predicted outcome (target)
2. Calculate the **votes** for each predicted target
3. Consider the **high voted** predicted target as the **final prediction** from the random forest algorithm

CHAPTER 4

IMPLEMENTATION

INTRODUCTION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

The project is implemented by accessing simultaneously from more than one system and more than one window in one system. The application is implemented in the Internet Information Services 5.0 web server under the Windows XP and accessed from various clients.

4.1Technologies Used

What is Python?

Python is an interpreter, high-level programming language for general-purpose programming by “Guido van Rossum” and first released in 1991, Python has a design philosophy that emphasizes code readability, and a syntax that allows programmers to express concepts in fewer lines of code, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional, procedural, and has a large and comprehensive standard library.

Python interpreters are available for many operating systems. Python, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. Python is managed by the non-profit Python Software Foundation.

Python is a general purpose, dynamic, high level and interpreted programming language. It supports object-oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high level data structures.

1. Windows XP
2. Python Programming
3. Open source libraries: Pandas, NumPy, SciPy, matplotlib, OpenCV

WHY PYTHON?

- Python is a scripting language like PHP, Perl, and Ruby.
- No licensing, distribution, or development fees
- It is a Desktop application.
- Linux, windows
- Excellent documentation
- Thriving developer community
- For us job opportunity

Libraries Of python:

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom numbers, arithmetic with arbitrary precision decimals, manipulating regular expressions, and unit testing.

Some parts of the standard library are covered by specifications (for example, the Web Server Gateway Interface (WSGI) implementation `wsgiref` follows PEP 33), but most modules are not.

They are specified by their code, internal documentation, and test suites (if supplied). However, because most of the standard library is cross-platform Python code, only a few modules need altering or rewriting for variant implementations.

As of March 2018, the Python Package Index (PyPI), the official repository for third party Python software, contains over 130,000 packages with a wide range of functionality, including:

1. Graphical user interfaces
2. Multimedia
3. Databases
4. Documentation
5. System administration

4.1 EXPLANATION OF KEY FUNCTIONS

Modules

1. DATASET

Dataset selection for experimentation is a significant task, because the performance of the system is based on the correctness of a dataset. The more accurate the data, the greater the effectiveness of the system. The dataset can be collected by numerous means, such as 1) sanitized dataset, 2) simulated dataset, 3) testbed dataset, and 4) standard dataset. However, complications occur in the application of the first three methodologies. A real traffic method is expensive, whereas the sanitized method is unsafe. The development of a simulation system is also complex and challenging. Additionally, different types of traffic are required to model various network attacks, which is complex and costly. To overcome these difficulties, the NSL-KDD dataset is used to validate the proposed system for intrusion detection.

2. PRE-PROCESSING

The classifier is unable to process the raw dataset because of some of its symbolic features. Thus, pre-processing is essential, in which non-numeric or symbolic features are eliminated or replaced, because they do not indicate vital participation in intrusion detection. However, this process generates overhead including more training time; the classifier's architecture becomes complex and wastes memory and computing resources.

Therefore, the non-numeric features are excluded from the raw dataset for improved performance of intrusion detection systems.

3. CLASSIFICATION

Placing an activity into normal and intrusive categories is the core function of an intrusion detection system, which is known as an intrusive analysis engine. Thus, different classifiers have been applied as intrusive analysis engines in intrusion detection in the literature, such as multilayer perceptron, SVM, naive Bayes, self-organizing map, and DT. However, in this study, the three different classifiers of SVM, RF, and ELM are applied based on their proven ability in classification problems. Details of each classification approach are provided

4.SUPPORT VECTOR MACHINE

Implementation of the SVM model in the proposed system. The kernel function uses squared Euclidean distance between two numeric vectors and maps input data to a high dimensional space to optimally separate the given data into their respective attack classes. Therefore, kernel RBF is particularly effective in separating sets of data that share complex boundaries. In our study, all the simulations have been conducted using the freely available Lib SVM package.

5.RANDOM FOREST

First, Random Forest algorithm is a supervised classification algorithm. We can see it from its name, which is to create a forest by some way and make it random.

There is a direct relationship between the number of trees in the forest and the results it can get: the larger the number of trees, the more accurate the result. But one thing to note is that creating the forest is not the same as constructing the decision with information gain or gain index approach.

The decision tree is a decision support tool. It uses a tree-like graph to show the possible consequences. If you input a training dataset with targets and features into the decision tree, it will formulate some set of rules. These rules can be used to perform predictions.

6. EXTREME LEARNING MACHINE

ELM is another name for single or multiple hidden layer feed forward neural networks. ELM can be used to solve various classification, clustering, regression, and feature engineering problems. This learning algorithm involves input layer, one or multiple hidden layers and the output layer. In the traditional neural networks, the tasks of adjustment of the input and hidden layer weights are very computationally expensive and time-consuming because it requires multiple rounds to converge. To overcome this problem, Huang et al. proposed an SLFN by arbitrarily selecting input weights and hidden layer biases to minimize the training time.

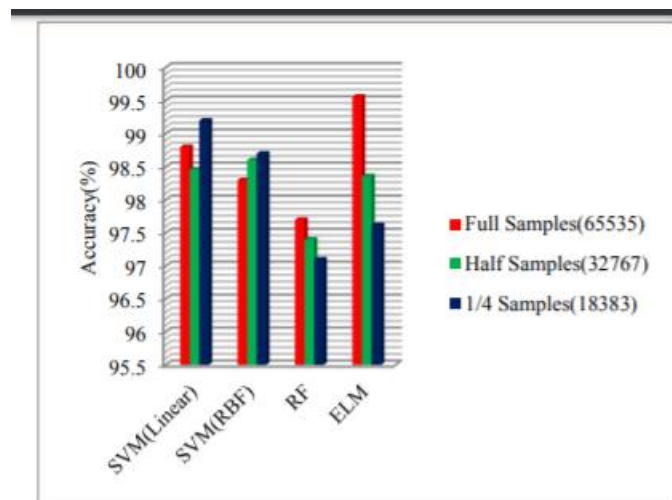


FIGURE 4.1:Accuracy

LIBRARIES

Numpy:

Numpy is a general-purpose array-processing package. It provides a high-Performance multidimensional array object and tools for working with these Arrays. It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

1. A powerful N-dimensional array object
2. Sophisticated (broadcasting) functions
3. Tools for integrating C/C++ and FORTRAN code
4. Useful linear algebra, Fourier transforms, and random number capabilities

Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety of databases.

Pandas:

Pandas are an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. The name Pandas is derived from the word Panel Data – an Econometrics from Multidimensional data. In 2008, developer Wes McKinney started developing pandas when in need of high performance, flexible tool for analysis of data. Prior to Pandas, Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data — load,prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc. Standard Python distribution doesn't come bundled with Pandas module. A lightweight alternative is to install NumPy using popular Python package installer, pip. pip install pandas

matplotlib:

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object- oriented API for embedding plots into applications using general- purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+.

There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. [3] SciPy makes use of Matplotlib. Matplotlib was originally written by John D. Hunter, has an active development community, [4] and is distributed under a BSD-style license. Michael Droettboom was nominated as matplotlib's lead developer shortly before John Hunter's death in August 2012, [5] and further joined by Thomas Caswell. As of 23 June 2017, matplotlib 2.0.x supports Python versions 2.7 through 3.6. Python3 support started with Matplotlib 1.2. Matplotlib 1.4 is the last version to support Python 2.6. Matplotlib has pledged to not support Python 2 past 2020 by signing the Python 3 Statement.

Sklearn

Scikit-learn is a library in Python that provides many unsupervised and supervised learning algorithms. It's built upon some of the technology you might already be familiar with, like NumPy, pandas, and Matplotlib!

The functionality that scikit-learn provides include:

1. Regression, including Linear and Logistic Regression
2. Classification, including K-Nearest Neighbors
3. Clustering, including K-Means and K-Means++
4. Model selection
5. Preprocessing, including Min-Max Normalization

As you move through Codecademy's Machine Learning content, you will become familiar with many of these terms. You will also see scikit- learn (in Python, sklearn) modules being used.

For example: `sklearn.linear_model.LinearRegression()` is a Linear Regression model inside the `linear_model` module of `sklearn`. The power of `scikit-learn` will greatly aid your creation of robust Machine Learning programs.

4.2 METHOD OF IMPLEMENTATION

Step 1: Define the objective of the Problem Statement

At this step, we must understand what exactly needs to be predicted. In our case, the objective is to predict the performance of student by studying features available. At this stage, it is also essential to take mental notes on what kind of data can be used to solve this problem or the type of approach you must follow to get to the solution.

Step 2: Data Gathering

At this stage, you must be asking questions such as,

1. What kind of data is needed to solve this problem?
2. Is the data available?
3. How can I get the data?

Once you know the types of data that is required, you must understand how you can derive this data. Data collection can be done manually or by web scraping. However, if you're a beginner and you're just looking to learn Machine Learning you don't have to worry about getting the data. There are 1000s of data resources on the web, you can just download the data set and get going.

Coming back to the problem at hand, the data needed for predicting the performance includes measures such as study time, grades, absences, etc. Such data must be collected and stored for analysis.

Step 3: Data Preparation

The data you collected is almost never in the right format. You will encounter a lot of inconsistencies in the data set such as missing values, redundant variables, duplicate values, etc.

Removing such inconsistencies is very essential because they might lead to wrongful computations and predictions. Therefore, at this stage, you scan the data set for any inconsistencies and you fix them then and there.

school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	reason	guardian	traveltime	studytime	failures	schoolsup	famsup	paid	act1
GP	F	18	U	GT3	A	4	4	at_home	teacher	course	mother	2	2	0	yes	no	no	no
GP	F	17	U	GT3	T	1	1	at_home	other	course	father	1	2	0	no	yes	no	no
GP	F	15	U	LE3	T	1	1	at_home	other	other	mother	1	2	3	yes	no	yes	no
GP	F	15	U	GT3	T	4	2	health	services	home	mother	1	3	0	no	yes	yes	yes
GP	F	16	U	GT3	T	3	3	other	other	home	father	1	2	0	no	yes	yes	no
GP	M	16	U	LE3	T	4	3	services	other	reputation	mother	1	2	0	no	yes	yes	yes
GP	M	16	U	LE3	T	2	2	other	other	home	mother	1	2	0	no	no	no	no
GP	F	17	U	GT3	A	4	4	other	teacher	home	mother	2	2	0	yes	yes	no	no
GP	M	15	U	LE3	A	3	2	services	other	home	mother	1	2	0	no	yes	yes	no
GP	M	15	U	GT3	T	3	4	other	other	home	mother	1	2	0	no	yes	yes	yes
GP	F	15	U	GT3	T	4	4	teacher	health	reputation	mother	1	2	0	no	yes	yes	no
GP	F	15	U	GT3	T	2	1	services	other	reputation	father	3	3	0	no	yes	no	yes
GP	M	15	U	LE3	T	4	4	health	services	course	father	1	1	0	no	yes	yes	yes
GP	M	15	U	GT3	T	4	3	teacher	other	course	mother	2	2	0	no	yes	yes	no
GP	M	15	U	GT3	A	2	2	other	other	home	other	1	3	0	no	yes	no	no
GP	F	16	U	GT3	T	4	4	health	other	home	mother	1	1	0	no	yes	no	no
GP	F	16	U	GT3	T	4	4	services	services	reputation	mother	1	3	0	no	yes	yes	yes
GP	F	16	U	GT3	T	3	3	other	other	reputation	mother	3	2	0	yes	yes	no	yes
GP	M	17	U	GT3	T	3	2	services	services	course	mother	1	1	3	no	yes	no	yes
GP	M	16	U	LE3	T	4	3	health	other	home	father	1	1	0	no	no	yes	yes
GP	M	15	U	GT3	T	4	3	teacher	other	reputation	mother	1	2	0	no	no	no	no
GP	M	15	U	GT3	T	4	4	health	health	other	father	1	1	0	no	yes	yes	no
GP	M	16	U	LE3	T	4	2	teacher	other	course	mother	1	2	0	no	yes	no	yes
GP	M	16	U	LE3	T	2	2	other	other	reputation	mother	2	2	0	no	yes	no	yes

Figure 4.2.1:Dataset-1

schoolsup	famsup	paid	activities	nursery	higher	internet	romantic	famrel	freetime	goout	Dalc	Walc	health	absences	passed
yes	no	no	no	yes	yes	no	no	4	3	4	1	1	3	6	no
no	yes	no	no	yes	yes	yes	no	5	3	3	1	1	3	4	no
yes	no	yes	no	yes	yes	yes	no	4	3	2	2	3	3	10	yes
no	yes	yes	yes	yes	yes	yes	yes	3	2	2	1	1	5	2	yes
no	yes	yes	no	yes	yes	no	no	4	3	2	1	2	5	4	yes
no	yes	yes	yes	yes	yes	yes	no	5	4	2	1	2	5	10	yes
no	no	no	no	yes	yes	yes	no	4	4	4	1	1	3	0	yes
yes	yes	no	no	yes	yes	no	no	4	1	4	1	1	1	6	no
no	yes	yes	no	yes	yes	yes	no	4	2	2	1	1	1	0	yes
no	yes	yes	yes	yes	yes	yes	no	5	5	1	1	1	5	0	yes
no	yes	yes	no	yes	yes	yes	yes	3	3	3	1	2	2	0	no
yes	no	yes	no	yes	yes	yes	no	5	2	2	1	1	4	4	yes
no	yes	yes	yes	yes	yes	yes	no	4	3	3	1	3	5	2	yes
no	yes	yes	no	yes	yes	yes	yes	5	4	3	1	2	3	2	yes
no	yes	no	no	yes	yes	yes	yes	4	5	2	1	1	3	0	yes
no	yes	no	no	no	yes	yes	yes	4	4	4	1	2	2	4	yes
yes	yes	yes	yes	yes	yes	yes	no	3	2	3	1	2	2	6	yes
yes	no	yes	yes	yes	yes	yes	no	5	3	2	1	1	4	4	yes
no	yes	no	yes	yes	yes	yes	no	5	5	5	2	4	5	16	no
no	no	yes	yes	yes	yes	yes	no	3	1	3	1	3	5	4	yes
no	no	no	no	yes	yes	yes	no	4	4	1	1	1	1	0	yes
no	yes	yes	no	yes	yes	yes	no	5	4	2	1	1	5	0	yes
no	no	no	yes	yes	yes	yes	no	4	5	1	1	3	5	2	yes
no	yes	no	yes	yes	yes	yes	no	5	4	4	2	4	5	0	yes

Figure 4.2.2:Dataset-2

Step 4: Exploratory Data Analysis

Grab your detective glasses because this stage is all about diving deep into data and finding all the hidden data mysteries. EDA or Exploratory Data Analysis is the brainstorming stage of Machine Learning. Data Exploration involves understanding the patterns and trends in the data. At this stage, all the useful insights are drawn and correlations between the variables are understood.

For example, in the case of predicting performance, we know that there is a strong possibility of failure if the grades are low. Such correlations must be understood and mapped at this stage.

Step 5: Building a Machine Learning Model

All the insights and patterns derived during Data Exploration are used to build the Machine Learning Model. This stage always begins by splitting the data set into two parts, training data, and testing data. The training data will be used to build and analyze the model. The logic of the model is based on the Machine Learning Algorithm that is being implemented.

Choosing the right algorithm depends on the type of problem you're trying to solve, the data set and the level of complexity of the problem. In the upcoming sections, we will discuss the different types of problems that can be solved by using Machine Learning.

Step 6: Model Evaluation & Optimization

After building a model by using the training data set, it is finally time to put the model to a test. The testing data set is used to check the efficiency of the model and how accurately it can predict the outcome. Once the accuracy is calculated, any further improvements in the model can be implemented at this stage. Methods like parameter tuning and cross-validation can be used to improve the performance of the model.

Step 7: Predictions

Once the model is evaluated and improved, it is finally used to make predictions. The final output can be a Categorical variable (eg. True or False) or it can be a Continuous Quantity (eg. the predicted value of a stock).

In our case, for predicting the performance of student, the output will be a categorical variable.

4.2.1 FORMS

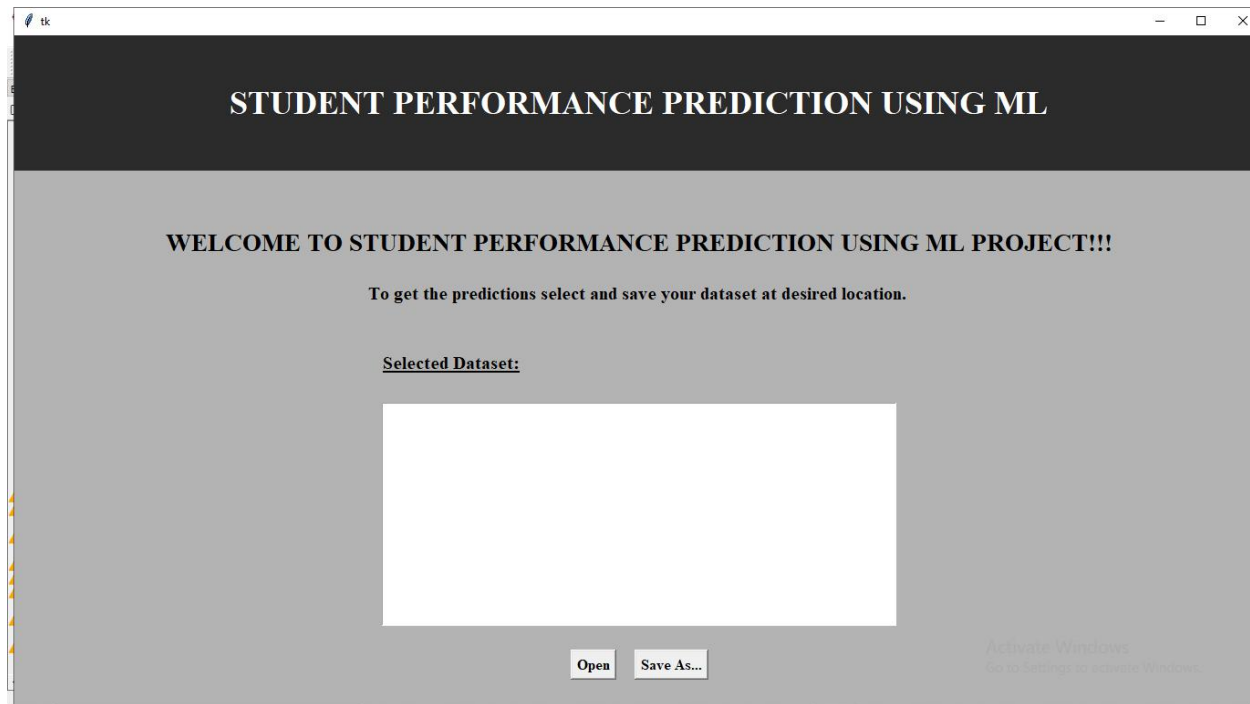


Figure 4.2.1:Home Screen

4.2.2 OUTPUT SCREENS

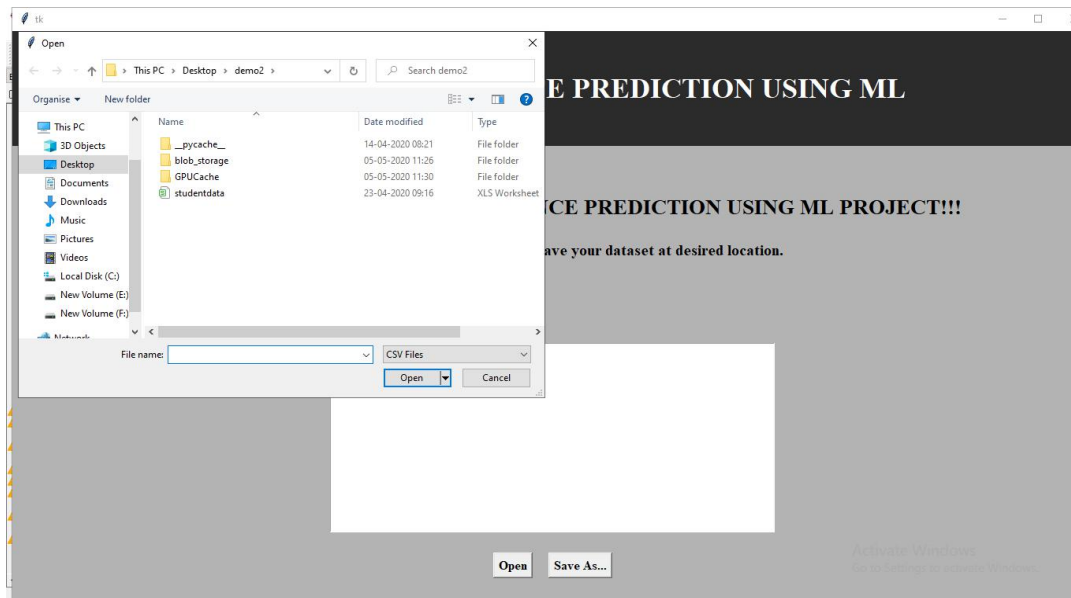


Figure 4.2.2:Uploading Dataset

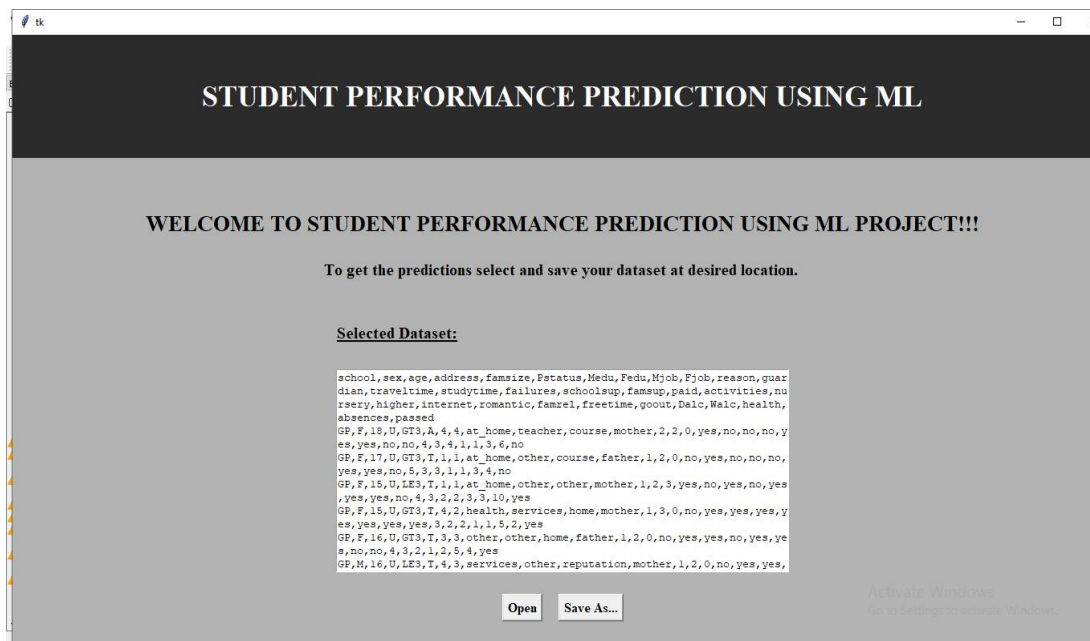
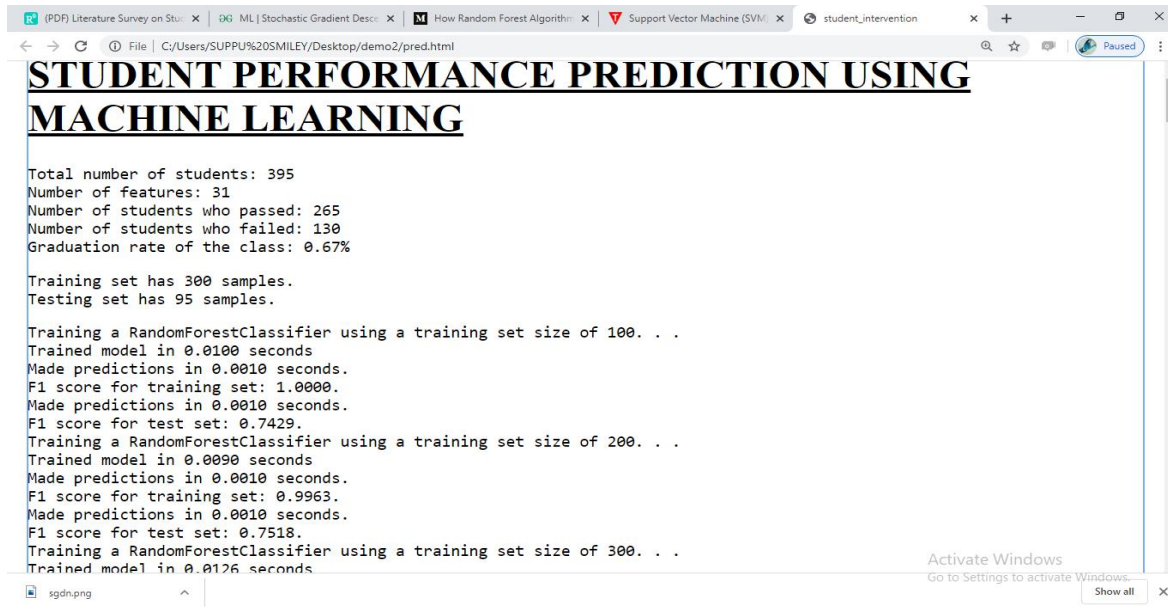


Figure 4.2.2:Dataset



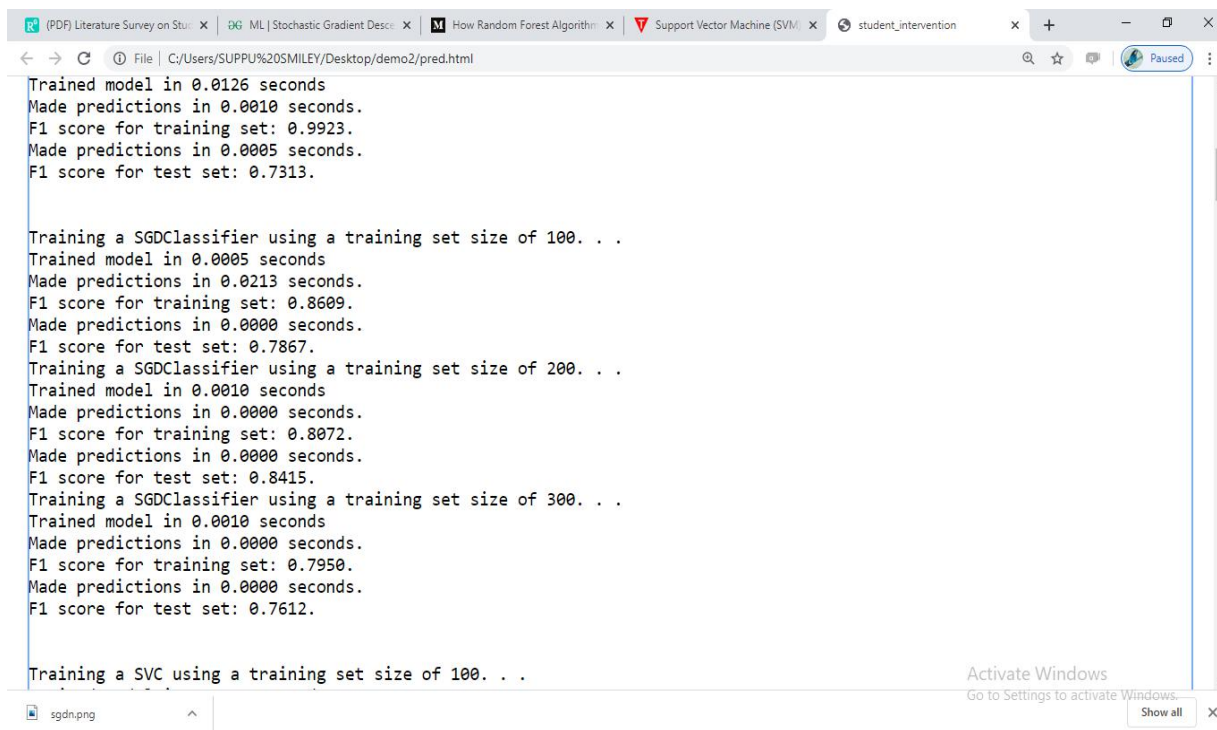
STUDENT PERFORMANCE PREDICTION USING MACHINE LEARNING

Total number of students: 395
Number of features: 31
Number of students who passed: 265
Number of students who failed: 130
Graduation rate of the class: 0.67%

Training set has 300 samples.
Testing set has 95 samples.

Training a RandomForestClassifier using a training set size of 100. . .
Trained model in 0.0100 seconds
Made predictions in 0.0010 seconds.
F1 score for training set: 1.0000.
Made predictions in 0.0010 seconds.
F1 score for test set: 0.7429.
Training a RandomForestClassifier using a training set size of 200. . .
Trained model in 0.0090 seconds
Made predictions in 0.0010 seconds.
F1 score for training set: 0.9963.
Made predictions in 0.0010 seconds.
F1 score for test set: 0.7518.
Training a RandomForestClassifier using a training set size of 300. . .
Trained model in 0.0126 seconds

Figure 4.2.2:Output Screen-2



Trained model in 0.0126 seconds
Made predictions in 0.0010 seconds.
F1 score for training set: 0.9923.
Made predictions in 0.0005 seconds.
F1 score for test set: 0.7313.

Training a SGDClassifier using a training set size of 100. . .
Trained model in 0.0005 seconds
Made predictions in 0.0213 seconds.
F1 score for training set: 0.8609.
Made predictions in 0.0000 seconds.
F1 score for test set: 0.7867.
Training a SGDClassifier using a training set size of 200. . .
Trained model in 0.0010 seconds
Made predictions in 0.0000 seconds.
F1 score for training set: 0.8072.
Made predictions in 0.0000 seconds.
F1 score for test set: 0.8415.
Training a SGDClassifier using a training set size of 300. . .
Trained model in 0.0010 seconds
Made predictions in 0.0000 seconds.
F1 score for training set: 0.7950.
Made predictions in 0.0000 seconds.
F1 score for test set: 0.7612.

Training a SVC using a training set size of 100. . .

Figure 4.2.2:Output Screen-3

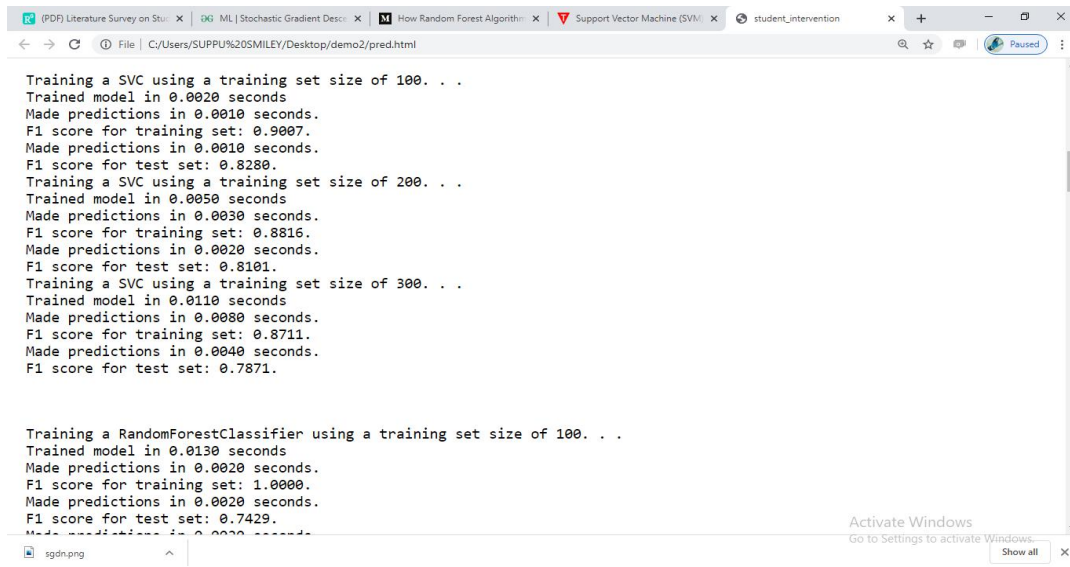


Figure 4.2.2:Output Screen-4

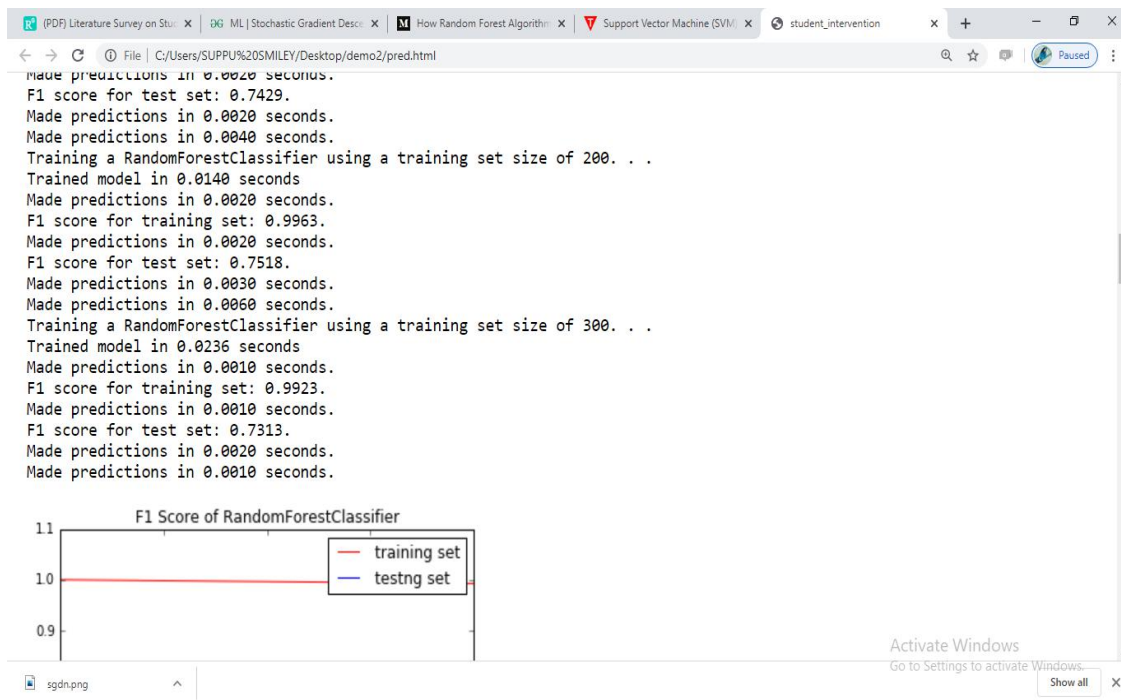


Figure 4.2.2:Output Screen-5

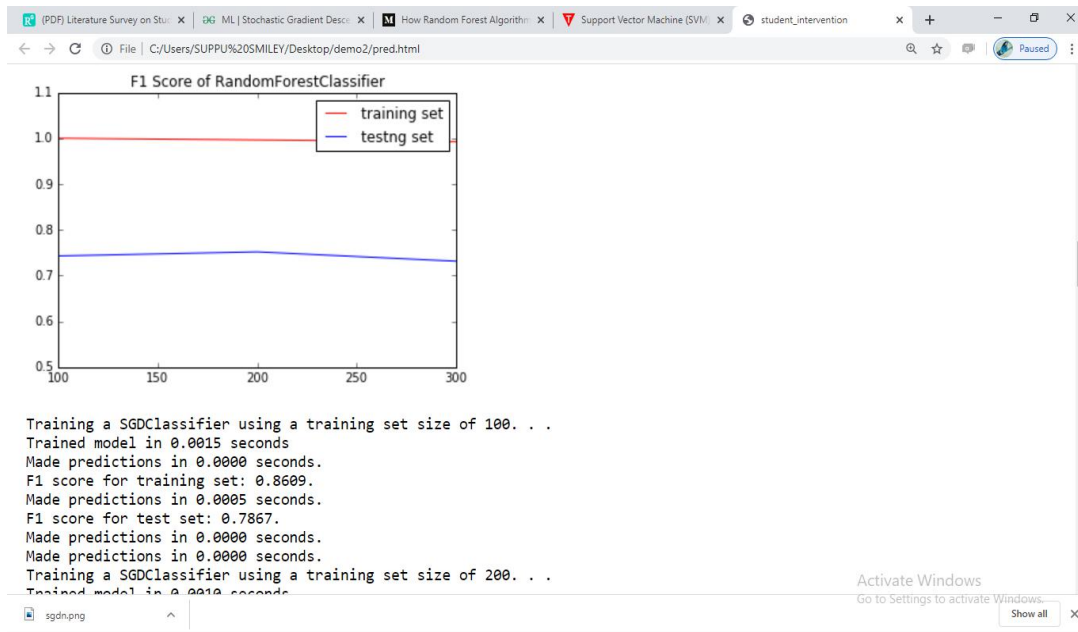


Figure 4.2.2:Output Screen-6

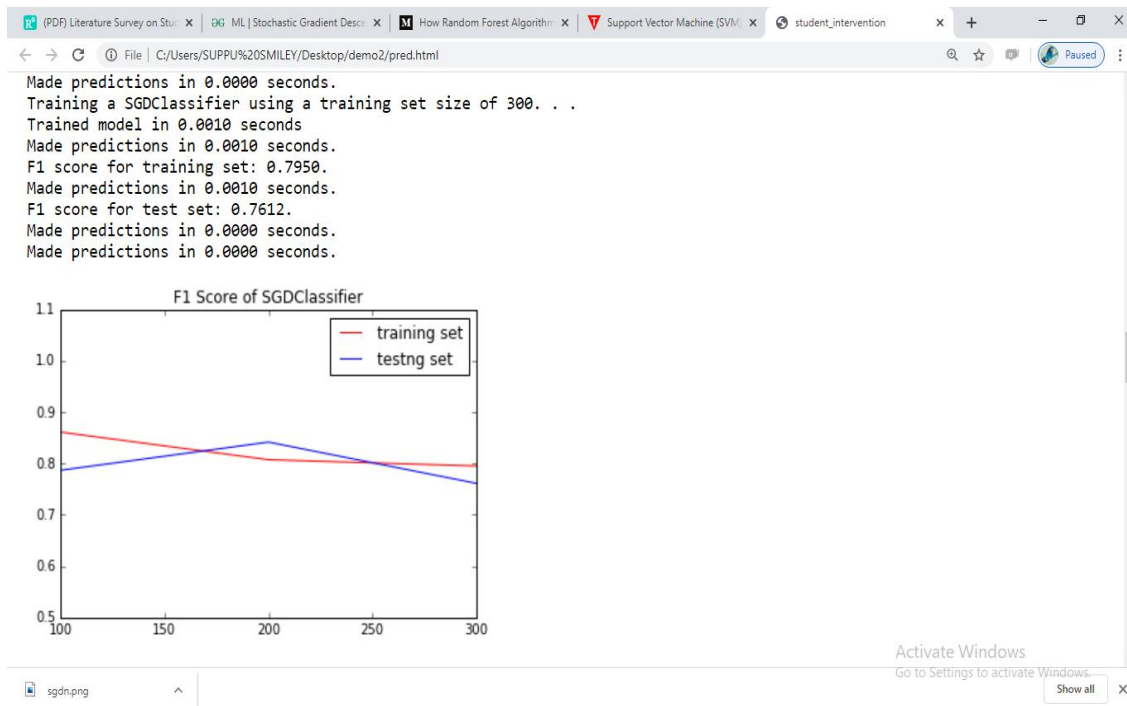


Figure 4.2.2:Output Screen-7

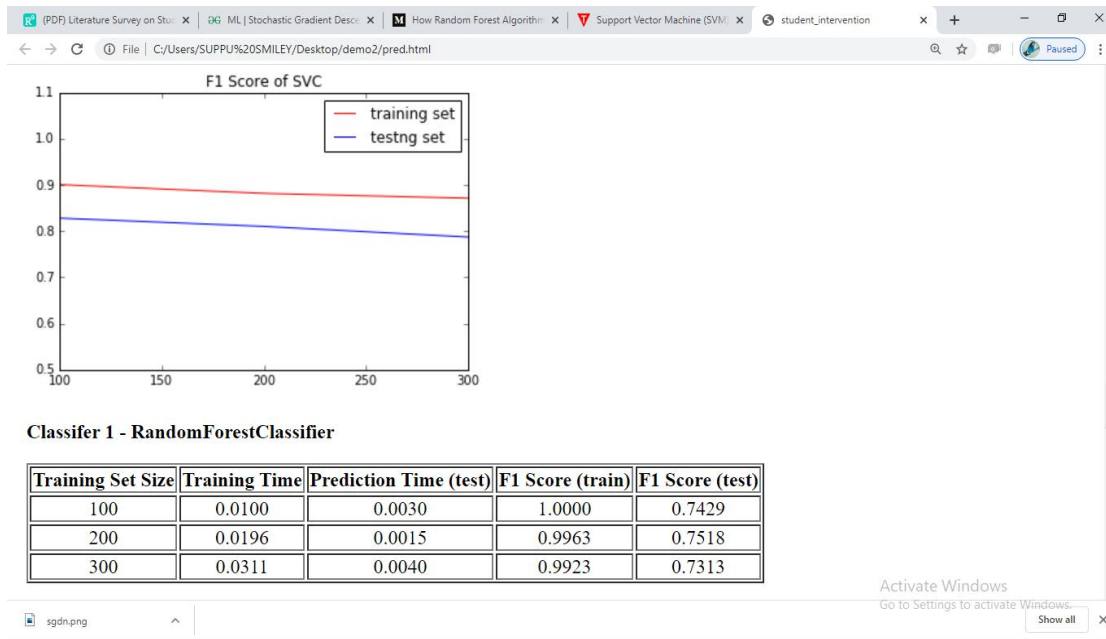


Figure 4.2.2:Output Screen-8

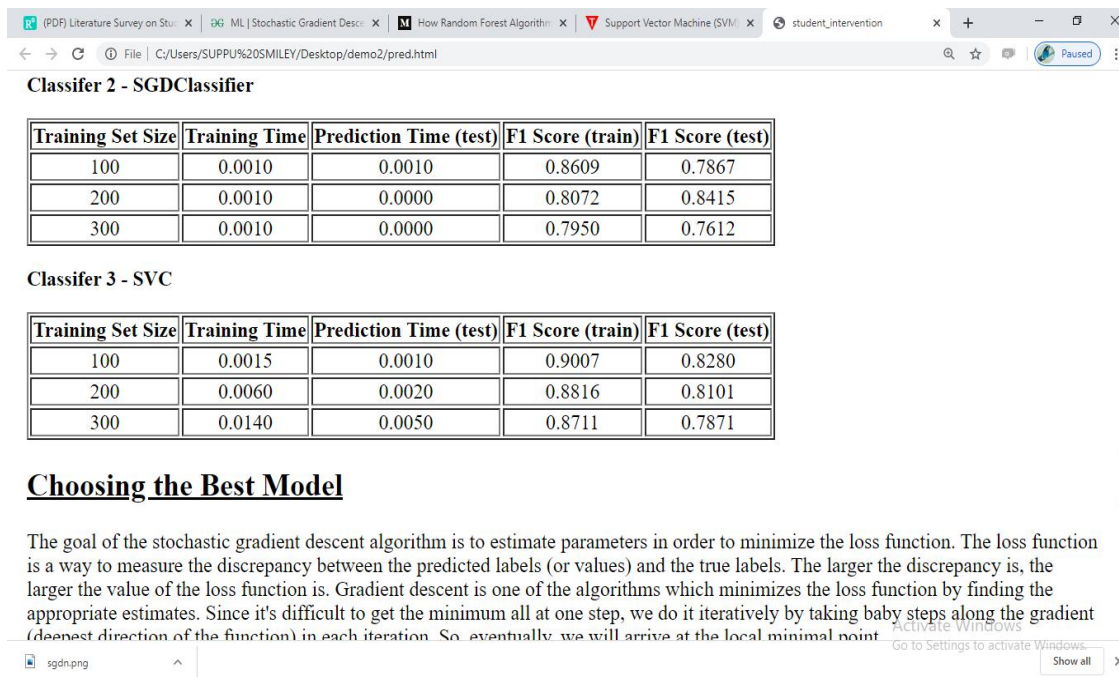


Figure 4.2.2:Output Screen-9

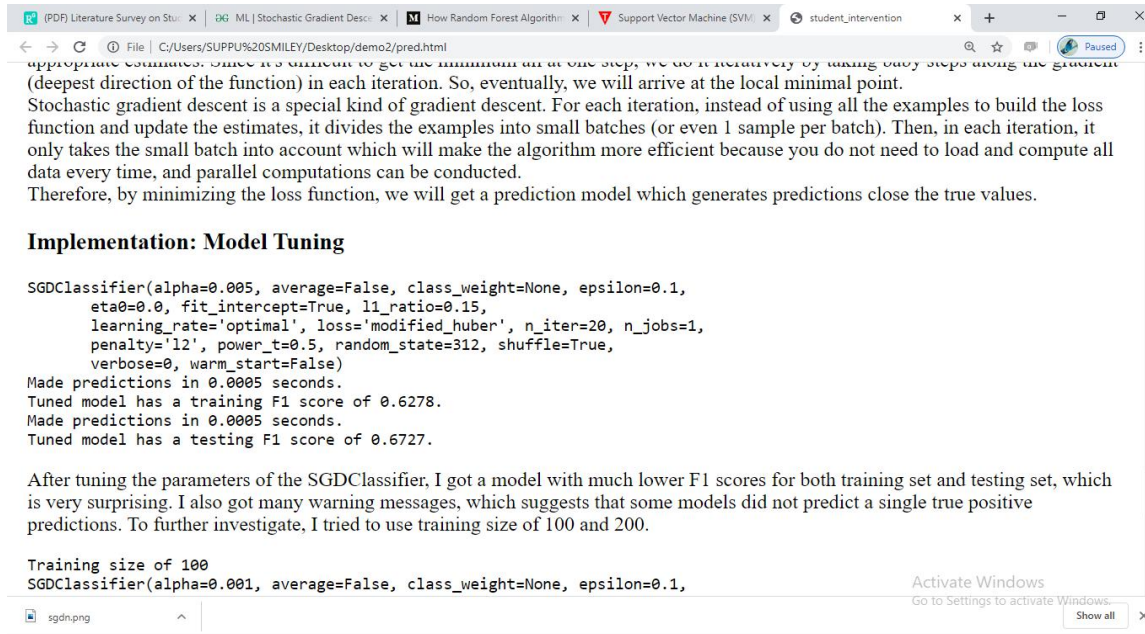


Figure 4.2.2:Output Screen-10

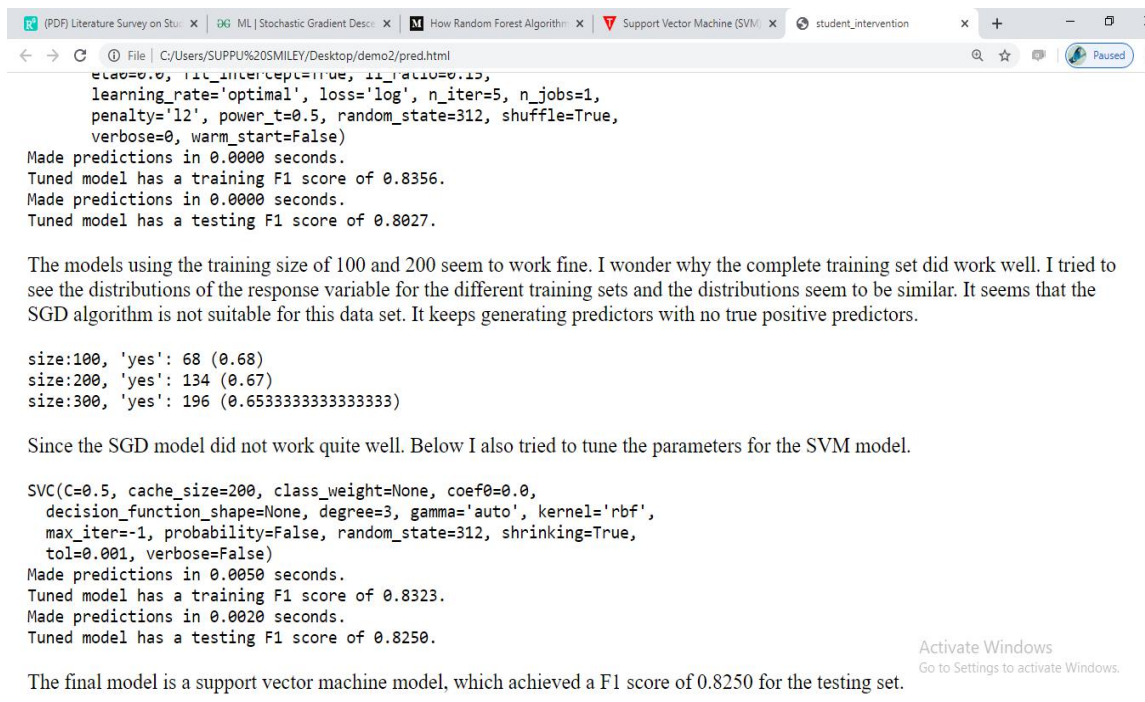


Figure 4.2.2:Output Screen-11

4.2.2 RESULT ANALYSIS

The ability to predict performance of students is very crucial in our present education system. We applied Machine learning concepts for this study. Machine learning techniques are used to discover models or patterns of data, and it is helpful in the decision-making.

Our main objectives to this proposed work are:

- 1.To understand, analyse and then find the difference between different prediction techniques of data mining in education.
- 2.To identify and understand different student attributes which are mainly used for the predicting the student performance.
- 3.To identify and understand the different prediction techniques which are mainly used for predicting the student performance. .

We compared the F1 scores of Random Forest, SGDC and SVC. The result obtained are mentioned below:

The models using the training size of 100 and 200 seem to work fine. We wonder why the complete training set did work well. I tried to see the distributions of the response variable for the different training sets and the distributions seem to be similar. It seems that the SGD algorithm is not suitable for this data set. It keeps generating predictors with no true positive predictors.

The final model is a support vector machine model, which achieved a F1 score of 0.8250 for the testing set.

CHAPTER 5

TESTING AND VALIDATION

It is the process of testing the functionality and it is the process of executing a program with the intent of finding an error. A good test case is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers an undiscovered error. Software testing is usually performed for one of two reasons:

- Defect Detection
- Reliability estimation

5.1 BLACK BOX TESTING:

The base of the black box testing strategy lies in the selection of appropriate data as per functionality and testing it against the functional specifications in order to check for normal and abnormal behavior of the system. Now a days, it is becoming to route the testing work to a third party as the developer of the system knows too much of the internal logic and coding of the system, which makes it unfit to test application by the developer. The following are different types of techniques involved in black box testing. They are:

1. Decision Table Testing
2. All pairs testing
3. State transition tables testing
4. Equivalence Partitioning

Software testing is used in association with Verification and Validation. Verification is the checking of or testing of items, including software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques as reviews, inspections, walk-through. Validation is the process of checking what has been specified is what the user actually wanted.

1. Validation: Are we doing the right job?
2. Verification: Are we doing the job right?

In order to achieve consistency in the Testing style, it is imperative to have and follow a set of testing principles. This enhances the efficiency of testing within SQA team members and thus contributes to increased productivity. The purpose of this document is to provide overview of the testing, plus the techniques. Here, after training is done on the training dataset, testing is done.

5.1.2 WHITE BOX TESTING:

White box testing requires access to source code. Though white box testing can be performed any time in the life cycle after the code is developed, it is a good practice to perform white box testing during the unit testing phase.

In designing a database the flow of specific inputs through the code, expected output and the functionality of conditional loops are tested.

At SDEI, 3 levels of software testing is done at various SDLC phases

1. **UNIT TESTING:** In which each unit (basic component) of the software is tested to verify that the detailed design for the unit has been correctly implemented
2. **INTEGRATION TESTING:** In which progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a whole.
3. **SYSTEM TESTING:** In which the software is integrated to the overall product and tested to show that all requirements are met. A further level of testing is also done, in accordance with requirements:
4. **REGRESSION TESTING:** Is used to refer to the repetition of the earlier successful tests to ensure that changes made in the software have not introduced new bugs/side effects.
5. **ACCEPTANCE TESTING:** Testing to verify a product meets customer specific requirements. The acceptance test suite is run against supplied input data. Then the results obtained are compared with the expected results of the client. A correct match was obtained.

5.1.3 TESTING CASES:

Testers have two main jobs, first one is to create a test case and the second is to use those for test execution. Testers create cases for various types of testing and provide details on what tester should do and what are the expected results of the test. Test cases are documents, may be a spreadsheet or directly maintained in the test management tools maintained by testers, so that even a novice can read and test the application or the product. Please note that, a test case created by one tester may be used by other tester.

1. So it is very essential that the author of the test case writes it in detail so that everyone can understand the test case.
2. Test Cases is a document that has the steps to be executed by tester to test a feature and what should be expected output from the application or product under test. The test case is a running document and must be updated and used as per the changing requirements.
3. Test cases should be prepared for constructive as well as destructive purpose. What we usually call as Positive and negative test cases.
4. Ideally, constructive testing is carried out from functionality, system, performance point of view, whereas destructive testing has emphasis on breaking the system by usually putting invalid inputs.
5. Many times, tester performs unplanned testing, which are not based on any test cases known as Random Testing.
6. During this, if any defects are identified then they should be converted to a test case.

The following are some of the tests that were conducted during testing:

5.2 VALIDATION:

Software validation checks that the software product satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements, not as specification artifacts or as needs of those who will operate the software only; but, as the needs of all the stakeholders (such as users, operators, administrators, managers, investors, etc.). There are two ways to perform software validation: internal and external.

During internal software validation it is assumed that the goals of the stakeholders were correctly understood and that they were expressed in the requirement artifacts precise and comprehensively.

If the software meets the requirement specification, it has been internally validated. External validation happens when it is performed by asking the stakeholders if the software meets their needs. Different software development methodologies call for different levels of user and stakeholder involvement and feedback; so, external validation can be a discrete or a continuous event. Successful final external validation occurs when all the stakeholders accept the software product and express that it satisfies their needs. Such final external validation requires the use of an acceptance test which is a dynamic test.

5.3 CONCLUSION:

Machine learning techniques can be useful in the field of students performance prediction considering that they helps to identify from the beginning of academic year. The aim of this application is to apply machine learning algorithms for prediction of student performance. An early analysis of student having poor performance helps the management take timely action to improve their performance through predicting their academic details.

Accurately predicting student performance based on their ongoing academic records is predicted. Also we conclude that proposed system is helping us to make the student performance better. Machine learning can prove to be powerful tool and all algorithms we used increases with increase in dataset size. However students' presence in class and their attendance, marks of bachelor degree in very important for classifiers.

CHAPTER 6

CONCLUSION

Present studies shows that academic performances of the students are primarily dependent on their past performances. Our investigation confirms that past performances have indeed got a significant influence over students' performance. Machine learning has come far from its nascent stages, and can prove to be a powerful tool in academia. In the future, applications similar to the one developed, as well as any improvements thereof may become an integrated part of every academic institution.

FUTURE SCOPE

This research has certain limitations that must be noted. There was not an access to a dedicated student data set, and the study relies on public data sources. In addition, both data sets were small, having less than thousand records. A research that has access to more comprehensive data may offer more conclusive results. Final area that can be improved is the process of feature creation. Since the data is limited, the amount of feature modification that can be made is also limited. Both data sources used in this research consists of a single table, and custom variables were created using variables from the same table. With a more comprehensive data set that spans multiple tables, there will be more potential to create new custom variables, while keeping in mind that the more a custom variable is, the more difficult it is to interpret the relation between it and the dependent variable.

REFERENCES:

- [1]Kaggle web site. <http://www.kaggle.com>
- [2]Roman Kern. 2014. Feature Engineering, Knowledge Discovery and Data Mining, <http://kti.tugraz.at/staff/denis/courses/kddm1/featureengineering.pdf>. Retrieved May 3, 2017.
- [3]Sas. 2017. Predictive Analytics: What it is and why it matters, SAS. https://www.sas.com/en_us/insights/analytics/predictive-analytics.html. Retrieved April 24, 2017.
- [4]A. A. Aburomman, M.B. Reaz, A novel SVM-kNN-PSO ensemble method for intrusion detection system, Applied Soft Computing, Volume 38, 2016, Pages 360-372, ISSN 1568-4946, <https://doi.org/10.1016/j.asoc.2015.10.011>.
- [5] S. Teng, N. Wu, H. Zhu, L. Teng and W. Zhang, "SVM-DT-based adaptive and collaborative intrusion detection," in IEEE/CAA Journal of Automatica Sinica, vol. 5, no. 1, pp. 108-118, Jan. 2018. doi: 10.1109/JAS.2017.7510730.
- [6] Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157-1182.
- [7] G. Dekker, M. Pechenizkiy, and J. Vleeshouwers. 2009. Predicting students drop out: A case study. In: Educational Data Mining 2009, 41-50.
- [8]Tom M. Mitchell. 1997. Machine Learning. McGraw-Hill.
- [9]M. Mohri, A. Rostamizadeh, and A. Talwalkar. 2012. Foundations of Machine Learning (Adaptive Computation and Machine Learning Series). MIT Press
- [10]Surjeet Kumar Yadav, Brijesh Bharadwaj, and Saurabh Pal. 2012. Mining education data to predict student's retention: A comparative study. International Journal of Computer Science and Information Security 10(2), 113-117..