Students Academic Performance Prediction System Using Machine Learning

Mini Project Report

Submitted by

RESHANA M M

Submitted in partial fulfillment of the requirements for the award of the degree of

Master of Computer Applications
Of
A P J Abdul Kalam Technological University

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®
ANGAMALY-683577, ERNAKULAM(DIST)
FEBRUARY 2022

DECLARATION

I **R**ESHANA M M, hereby declare that the report of this project work, submitted to the Department of Computer Applications, Federal Institute of Science and Technology (**FISAT**), Angamaly in partial fulfillment of the award of the degree of Master of Computer Application is an authentic record of our original work.

The report has not been submitted for the award of any degree of this university or any other university.

Date : 02/03/2022

Place: Angamaly

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®

ANGAMALY, ERNAKULAM-683577

DEPARTMENT OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project report titled **” Student’s Academic Performance Prediction using Machine Learning”** submitted by **RESHANA M M (FIT20MCA2091),** towards partial fulfillment of the requirements for the award of the degree of Master of Computer Applications is a record of bonafide work carried out by them during the year 2022.

Project Guide **Head of the Department**

Submitted for the viva-voice held on . . . . at . . . .

Examiner1 :

Examiner2 :

ACKNOWLEDGEMENT

Firstly we thank Almighty God for giving us the wisdom, grace and knowledge for making this project, a memorable one.

We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this Mini Project. To complete this project work we needed the direction, assistance and co-operation of various individuals, which is received in abundance with the grace of God.

We hereby express our deep sense of gratitude to **Dr. Manoj George**, Principal, FISAT and **Dr. C Sheela**, Vice principal, FISAT for allowing us to utilize all the facilities of the college.

Our sincere thanks to the scrum master **Dr. Shidha M V**, Assistant professor, Computer Applications, FISAT and our Internal guide for this project **Mrs.Joice T** Assistant professor, Computer Applications, FISAT for giving us valuable guidance, constructive suggestions and comment during our project work. We also express our boundless gratitude to all the lab faculty members for their guidance.

Finally we wish to express a whole hearted thanks to our parents, friends and well-wishers who extended their help in one way or other in preparation of our project.

ABSTRACT

The early indications regarding student's progress help academics to optimise their learning strategies and focus on diverse educational practises to make the learning experiences.Machine learning application can help academics to predict the expected weakness in learning processes and as a result they can proactively engage such student's in better learning experiences. Analysing the student's academic performance periodically is better , so the mentor's can be able make attention to the student's who need it to get improved in their academics

This study proposes a model for predicting the academic performance of Computer Science students using machine learning technique. The data were collected using questionnaires that contain the students' demographics, previous GPA, and family background information.

Researchers have recently proposed several machine learning-based algorithms for predicting academic achievement. In this paper, Support Vector Machine learning algorithm used to predict a student's academic achievement.

Contents

1 INTRODUCTION 8

2 PROOF OF CONCEPT 9

2.1 Objectives 9

3 SCRUM MEETINGS 10

4 IMPLEMENTATION 13

4.1 System Architecture 15

4.2 Dataset 17

4.3 Modules 18

4.3.1 DATA EXPLORATION 18

4.3.2 DATA CLEANING 18

4.3.3 MODELLING 18

4.3.4 PERFORMANCE ANALYSIS 18

5 RESULT ANALYSIS 20

6 CONCLUSION AND FUTURE SCOPE 22

6.1 Conclusion 23

6.2 Future Scope 23

7 APPENDIX 24

7.1 Source Code 25

8 SCREEN SHOTS 32

9 REFERENCES 38

Chapter 1

INTRODUCTION

Analysing the student's academic performance periodically is better , so the mentor's can be able make attention to the student's who need it to get improved in their academics.Every educational system organizational goal is to provide a good and fruitful knowledge to the students. Now a day's most of the educational institutions are spending most of their time and economy on finding out students' performance. Education is one of the most important sector for the application of machine learning.This study proposes a model for predicting the academic performance of Computer Science students using machine learning technique. The data were collected using questionnaires that contain the students' demographics, previous GPA, and family background information.

By analyzing the performance, they identify certain cluster of the students for whom they must give extra bit of care and actions, so that they performance gets enhanced. The model that we have trained will help us to predict that the students who can meet the placement criteria's and those students who don't met the criteria's can find measures how they can be improved to achieve the goal.

Chapter 2

PROOF OF CONCEPT

It is based on an IEEE paper known as Improving Experience of Students by Early prediction academic performance using machine learning techniques.

2.1 Objectives

The objective of this project is to develop a system based on Machine Learning to help the educational institutions to categorize the students and also the students can self analyse their performance .

Nowadays, acquiring a job is the toughest part. We have to competate with a lot of students. This system mainly predict the students who are eligible or not to attend for the placement drives in the colleges. There are many criteria provided by the company who conduct the drives. This system helps the students to realize that they are eligible or not for attending the drive. If they are not eligible, here provide the reason why they are not. The main aim of the project is to categorize the students and prevent the fake applications.

Chapter 3

SCRUM MEETINGS

On 24-11-2021

On this day I started searching the miniproject topic based on new technologies such as deep learning,IOT,Machine learning,classification,predication”.

On 29-11-2021

The topic was selected and did the detail study of the topic,the required dataset was selected.The dataset was searched from the different site such as kaggle,dataset etc.

On 06-12-2021

This day I submitted the synopsis and research paper to guide for the topic approval.

On 15-12-2021

After getting approval from the guide, the algorithm and model for the project were structured.Then the algorithm were choosen.

On 18-12-2021

On this day mam took a detailed class on how to do the project,what IDEs to use,what paper are refered,what steps are follow to do the project and so on

On 06-01-2022

According to the project the required IDE such as Visual Studio Code,Colab are choosen.Even checked whether the system was efficient to train the model.Here colab to code the project,then started to deploying the model using the algorithm.Python language is used to code the project.

On 13-01-2022

After the project first review according to Mam's opinion i change tha attributes name in dataset

On 13-01-2022

Used different algorithm/data model then choose the maximum accuracy one. The algorithm used are:- Support Vector Machine algorithm

On 19-01-2022

Started to do project coding.Firstly study the dataset and download the dataset from kaggle.The dataset is about different airlines in India and their details

On 25-01-2022

Testing the data application

On 28-01-2022

The training done using svm data model then choose the maximum accuracy with regression for predicting the academic performance of student.

On 02-02-2022

Create git repository.

On 07-02-2022

php is used for connection

Chapter 4

IMPLEMENTATION

Academia has seen a significant shift from conventional learning towards a student centric learning approach.As a result,it become critical for the mentors to monitor the progress of each students.

Analysing the performance of each students before moving to the new steps is very important. so they can realise the actual academia of them.Its may help them to work hard to achieve the goal.

Students academic performance prediction system using machine learning

This project aims at developing a system that helps to predict the academic performance of students.In this system,the details about a particular student is eligible or not to attend the placement drive .

This system proceeds in a way that the particular student will not be aware about this matter of checking with their details.This will be really helpful for the educational institutions.The machine learning algorithms like SVM are used here for analysing the input values from the user and thus produce the required result.Support vector machine(SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges.

A number of factors are interrelated with predicting the category of student including previous marks, arrears, absents days, parents satisfaction.using the factors the system predict the class of students. according to the category of student, used to predict the particular student is meeting the placement criteria provided the companys and if they not meet it, mention the reason about why they donot met the criteria.

For this study, we would like to early predict student final grade in one of the postgraduate courses in order to identify weak students. Early grade prediction will not only help instructors to know the students who need academic support, but will also help students to work on their weakness to achieve the goal. Furthermore, the findings will help instructors to revisit their educational strategies to foster better learning experience.

By analysing the class label of student show what category are they depends on. so the prediction of students that they met the placement criteria are done regarding to the above predicted class label

The factors that we used for predicting the classes are: Mark in tenth,Mark in Plustwo,Mark in Degree,Mark in PG,NO Baclogs,No of Absent days, Parent school satisfaction. using these factors, we predict the class of students.

4.1 System Architecture

Existing System

In the existing system, it only predict the academic performance of students by showing the student is good, bad or average for lower school students.

In this system,

* we are not able to analyse the performance of higher students.

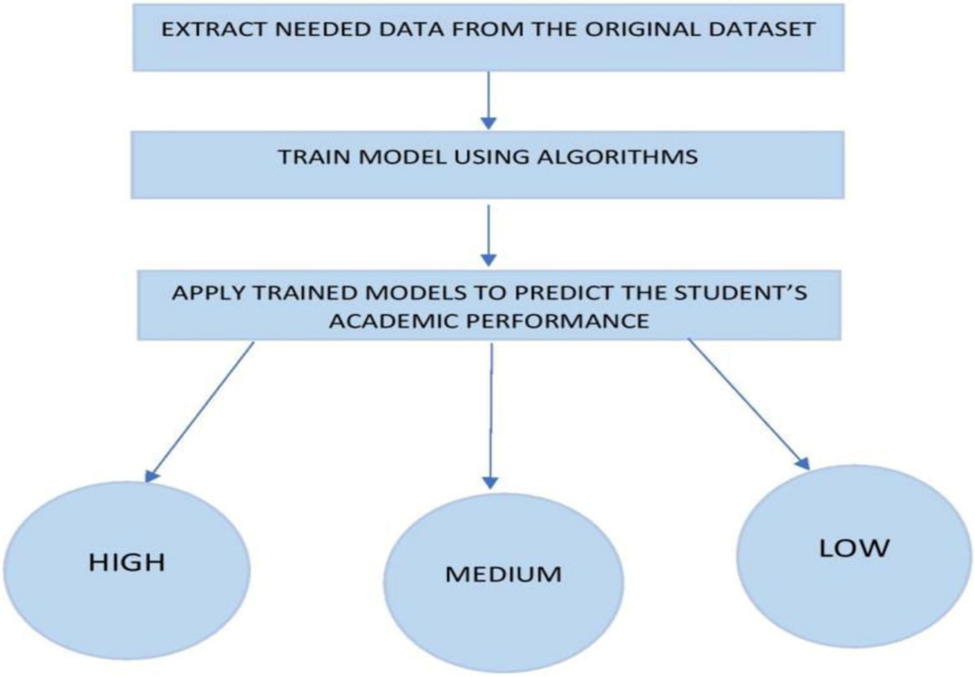
* It doesnot have any future scope.

* It cannot mention the reason if a student is below average.

Proposed System

The student academic performance prediction system is a machine learning project which we are developing for predicting the class of a student also according to the predicted class, specify that the student is eligible or not the placements. Each set of data can tell us a story. Our mission is to extract this story from the data and

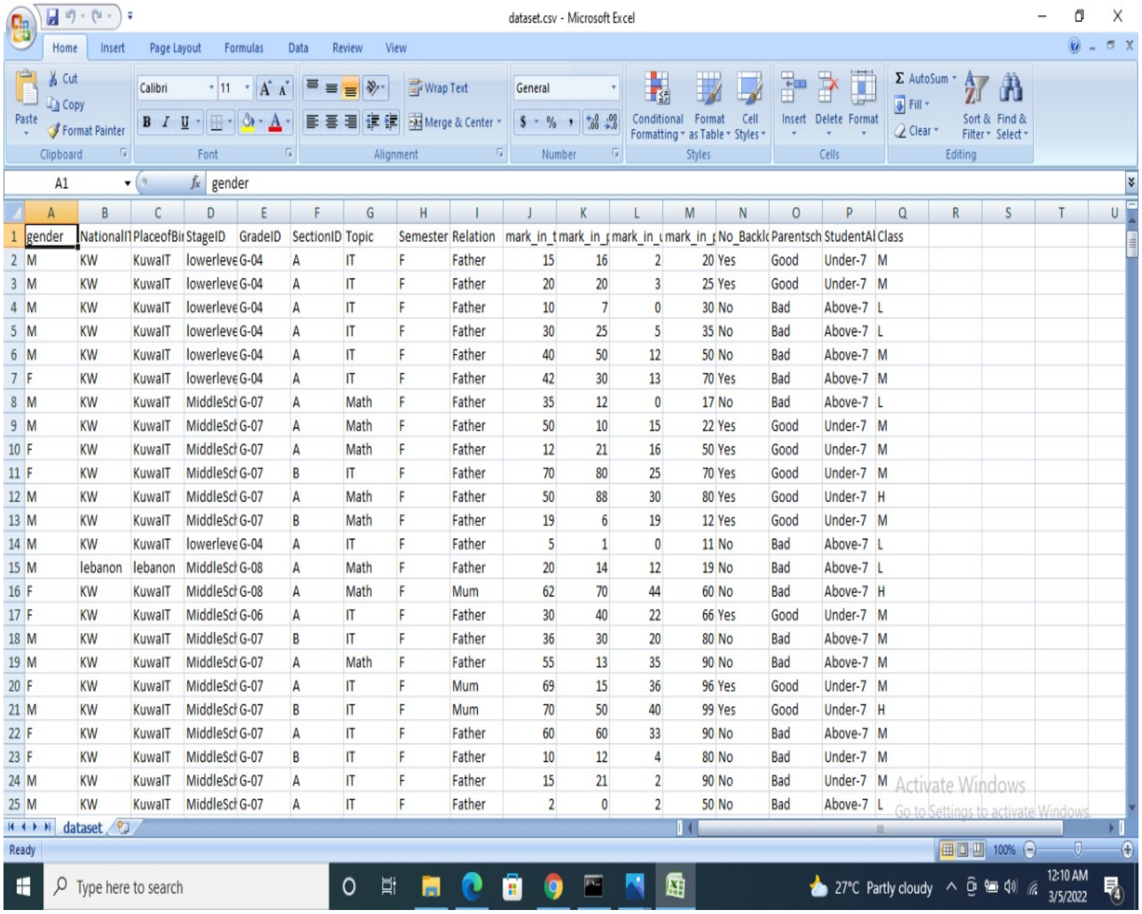
translate it into more readily accessible human language.
Problem which can be solved:
* Drawbacks of existing system can be solved.
* provides self analysing to students about their academics.
* System predict the particular student meet up the criteria for the placement
* provides the reason for their low academic performance .


EXTRACT NEEDED DATA FROM THE ORIGINAL DATASET
TRAIN MODEL USING ALGORITHMS
APPLY TRAINED MODELS TO PREDICT THE STUDENT'S ACADEMIC PERFORMANCE
HIGH
MEDIUM
LOW

The process diagram that describes the working of the system .

4.2 Dataset

The original data were obtained from a questionaire of 8177 respondents in 2020. Personal features and the factors each respondent were recorded according to their answers to a series of questions. Original dataset contained 17 factors, including educational details (previous marks, backlogs, attendance)and demographic information (level of education, age, gender). All categorical data were quantified.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	gender	NationalIT	PlaceofBir	StageID	GradeID	SectionID	Topic	Semester	Relation	mark_in_t	mark_in_	mark_in_	mark_in_	No_Backl	Parentsch	StudentA	Class
2	M	KW	KuwaIT	lowerleve	G-04	A	IT	F	Father	15	16	2	20	Yes	Good	Under-7	M
3	M	KW	KuwaIT	lowerleve	G-04	A	IT	F	Father	20	20	3	25	Yes	Good	Under-7	M
4	M	KW	KuwaIT	lowerleve	G-04	A	IT	F	Father	10	7	0	30	No	Bad	Above-7	L
5	M	KW	KuwaIT	lowerleve	G-04	A	IT	F	Father	30	25	5	35	No	Bad	Above-7	L
6	M	KW	KuwaIT	lowerleve	G-04	A	IT	F	Father	40	50	12	50	No	Bad	Above-7	M
7	F	KW	KuwaIT	lowerleve	G-04	A	IT	F	Father	42	30	13	70	Yes	Bad	Above-7	M
8	M	KW	KuwaIT	MiddleSch	G-07	A	Math	F	Father	35	12	0	17	No	Bad	Above-7	L
9	M	KW	KuwaIT	MiddleSch	G-07	A	Math	F	Father	50	10	15	22	Yes	Good	Under-7	M
10	F	KW	KuwaIT	MiddleSch	G-07	A	Math	F	Father	12	21	16	50	Yes	Good	Under-7	M
11	F	KW	KuwaIT	MiddleSch	G-07	B	IT	F	Father	70	80	25	70	Yes	Good	Under-7	M
12	M	KW	KuwaIT	MiddleSch	G-07	A	Math	F	Father	50	88	30	80	Yes	Good	Under-7	H
13	M	KW	KuwaIT	MiddleSch	G-07	B	Math	F	Father	19	6	19	12	Yes	Good	Under-7	M
14	M	KW	KuwaIT	lowerleve	G-04	A	IT	F	Father	5	1	0	11	No	Bad	Above-7	L
15	M	lebanon	lebanon	MiddleSch	G-08	A	Math	F	Father	20	14	12	19	No	Bad	Above-7	L
16	F	KW	KuwaIT	MiddleSch	G-08	A	Math	F	Mum	62	70	44	60	No	Bad	Above-7	H
17	F	KW	KuwaIT	MiddleSch	G-06	A	IT	F	Father	30	40	22	66	Yes	Good	Under-7	M
18	M	KW	KuwaIT	MiddleSch	G-07	B	IT	F	Father	36	30	20	80	No	Bad	Above-7	M
19	M	KW	KuwaIT	MiddleSch	G-07	A	Math	F	Father	55	13	35	90	No	Bad	Above-7	M
20	F	KW	KuwaIT	MiddleSch	G-07	A	IT	F	Mum	69	15	36	96	Yes	Good	Under-7	M
21	M	KW	KuwaIT	MiddleSch	G-07	B	IT	F	Mum	70	50	40	99	Yes	Good	Under-7	H
22	F	KW	KuwaIT	MiddleSch	G-07	A	IT	F	Father	60	60	33	90	No	Bad	Above-7	M
23	F	KW	KuwaIT	MiddleSch	G-07	B	IT	F	Father	10	12	4	80	No	Bad	Under-7	M
24	M	KW	KuwaIT	MiddleSch	G-07	A	IT	F	Father	15	21	2	90	No	Bad	Under-7	M
25	M	KW	KuwaIT	MiddleSch	G-07	A	IT	F	Father	2	0	2	50	No	Bad	Above-7	L

4.3 Modules

4.3.1 DATA EXPLORATION

DATA EXPLORATION is the first module in this project.Data exploration is the initial step in data analysis, where users explore a large data set in an unstructured way to uncover initial patterns, characteristics, and points of interest. This process isn't meant to reveal every bit of information a dataset holds, but rather to help create a broad picture of important trends and major points to study in greater detail.

4.3.2 DATA CLEANING

Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset. When combining multiple data sources, there are many opportunities for data to be duplicated or mislabeled. If data is incorrect, outcomes and algorithms are unreliable, even though they may look correct.

4.3.3 MODELLING

The process of modeling means training a machine learning algorithm to predict the labels from the features, tuning it for the business need, and validating it on holdout data.The output from modeling is a trained model that can be used for inference, making predictions on new data points.

4.3.4 PERFORMANCE ANALYSIS

Machine learning model performance is relative and ideas of what score a good model can achieve only make sense and can only be interpreted in the context of the skill scores of other models also trained on the same data.

Algorithm

1. 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.

There are two types of SVM model. Linear SVM is used for linearly separable data. When Single straight line classify two classes of a dataset, such data is called as linearly separable data and the classifier used for this type of data is Linear SVM.

Chapter 5

RESULT ANALYSIS

Since we have 3 categories of class in the data-set . After inputting users personal features and educational factors the system predict in which class he/she is belongs to ie, ”HIGH”, ”MEDIUM”, ”LOW”.

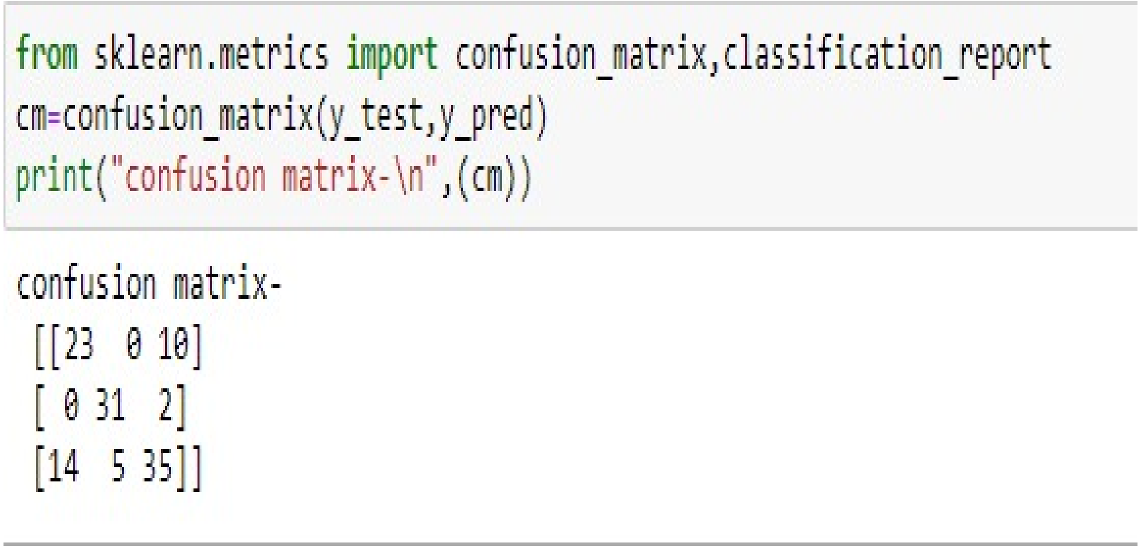
From the predicted output, we analyse the class of student and then we can able to say that the particular student is eligible or not for the placement criteria. if a student in HIGH class category is surely eligible for the drive. if a student in MEDIUM class may or may not be its belongs to the factors of the students.if a student is in LOW category, then the student is not eligible.

The educational factors like the marks, arrears, attendance are used mainly for the prediction. if the marks obtained is low but have attendance cannot be eligible but if a student have good marks but no proper may be eligible for the drive

Found a student with low marks in the academics surely say that the student is not eligible for the placement.that is for the prediction of class label the educational factors is much more concentrated than the personal factors.

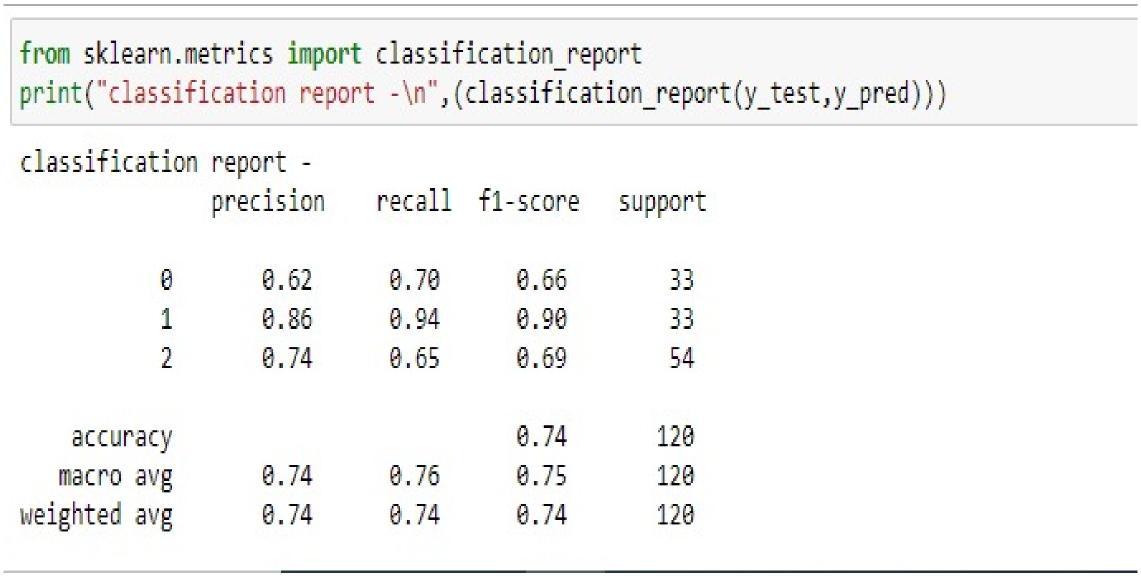
Also highlight the links between the student and educational institutions. A positive correlation between student institution, while, increasing scores for exams and A decreases risk of attendance shortage. Previous studies demonstrated that participants who have good academics and attendance have a strong point that will acquire the goal easily.

Also we compare three alogirthms namely svm, decision tree and knn using confusion matrix, ROC curve, accuracy, precision , recall etc. and founded that our proposed algorithm SVM is performed well when compared to other two's.



```
from sklearn.metrics import confusion_matrix,classification_report
cm=confusion_matrix(y_test,y_pred)
print("confusion matrix-\n",(cm))
```

```
confusion matrix-
 [[23  0 10]
 [ 0 31  2]
 [14  5 35]]
```



```
from sklearn.metrics import classification_report
print("classification report -\n",(classification_report(y_test,y_pred)))
```

```
classification report -
               precision    recall  f1-score   support

           0       0.62      0.70      0.66        33
           1       0.86      0.94      0.90        33
           2       0.74      0.65      0.69        54

    accuracy                           0.74       120
   macro avg       0.74      0.76      0.75       120
weighted avg       0.74      0.74      0.74       120
```

Chapter 6

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

In this work, machine learning algorithms are applied to predict the category of students which they belongs to according to their academic performance . SVM is a powerful tool to predict the output when compared to Decision tree and knn algorithm.
According to results , educational factors is the most important in predicting the academic performance. also predict the eligibility of the students to future placement process.

6.2 Future Scope

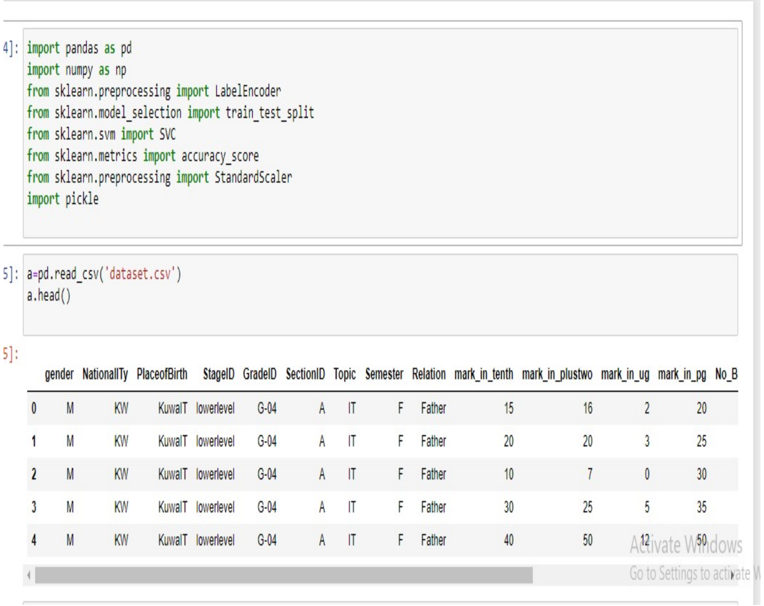
We can modify or update the project. In this project we find out the students who are eligible to attend the placement drive. If they are not then, provide reason why they cannot. In Future, it can be directly implemented by the companies as an initial process of applying for the placement drive. That means the students who are eligible only get the link for applying the current job position. Also it can be implemented in institutions for categorizing the students so it is easy to find out the not eligible students and can restrict them from fake applications.

Chapter 7

APPENDIX

7.1 Source Code

TRAIN

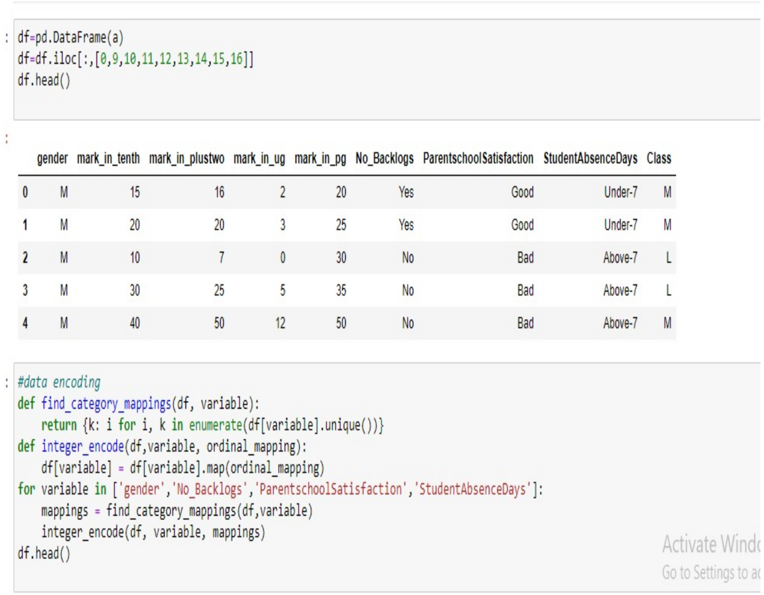


```
4]: import pandas as pd
    import numpy as np
    from sklearn.preprocessing import LabelEncoder
    from sklearn.model_selection import train_test_split
    from sklearn.svm import SVC
    from sklearn.metrics import accuracy_score
    from sklearn.preprocessing import StandardScaler
    import pickle
```

```
5]: a=pd.read_csv('dataset.csv')
    a.head()
```

5]:

	gender	NationalITy	PlaceofBirth	StageID	GradeID	SectionID	Topic	Semester	Relation	mark_in_tenth	mark_in_plustwo	mark_in_ug	mark_in_pg	No_B
0	M	KW	KuwaIT	lowerlevel	G-04	A	IT	F	Father	15	16	2	20	
1	M	KW	KuwaIT	lowerlevel	G-04	A	IT	F	Father	20	20	3	25	
2	M	KW	KuwaIT	lowerlevel	G-04	A	IT	F	Father	10	7	0	30	
3	M	KW	KuwaIT	lowerlevel	G-04	A	IT	F	Father	30	25	5	35	
4	M	KW	KuwaIT	lowerlevel	G-04	A	IT	F	Father	40	50	12	50	

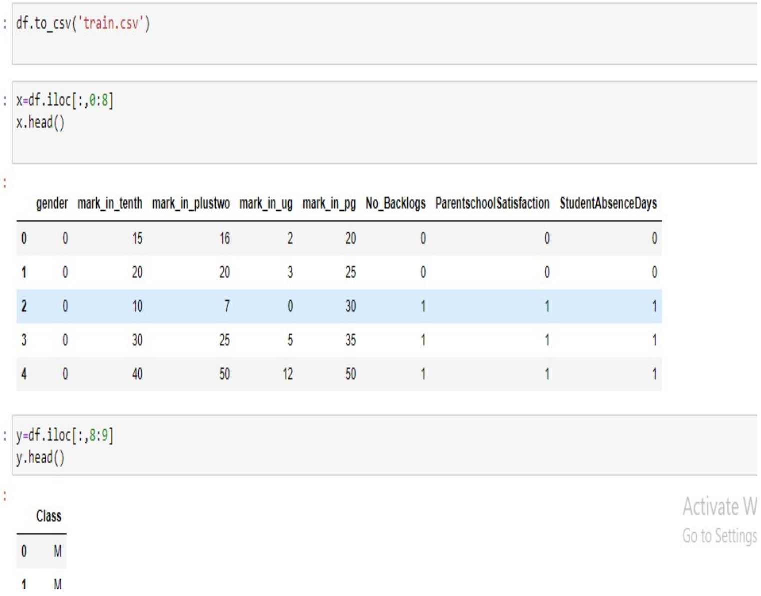


```
: df=pd.DataFrame(a)
  df=df.iloc[:,[0,9,10,11,12,13,14,15,16]]
  df.head()
```

:

	gender	mark_in_tenth	mark_in_plustwo	mark_in_ug	mark_in_pg	No_Backlogs	ParentschoolSatisfaction	StudentAbsenceDays	Class
0	M	15	16	2	20	Yes	Good	Under-7	M
1	M	20	20	3	25	Yes	Good	Under-7	M
2	M	10	7	0	30	No	Bad	Above-7	L
3	M	30	25	5	35	No	Bad	Above-7	L
4	M	40	50	12	50	No	Bad	Above-7	M

```
: #data encoding
  def find_category_mappings(df, variable):
      return {k: i for i, k in enumerate(df[variable].unique())}
  def integer_encode(df,variable, ordinal_mapping):
      df[variable] = df[variable].map(ordinal_mapping)
  for variable in ['gender','No_Backlogs','ParentschoolSatisfaction','StudentAbsenceDays']:
      mappings = find_category_mappings(df,variable)
      integer_encode(df, variable, mappings)
  df.head()
```



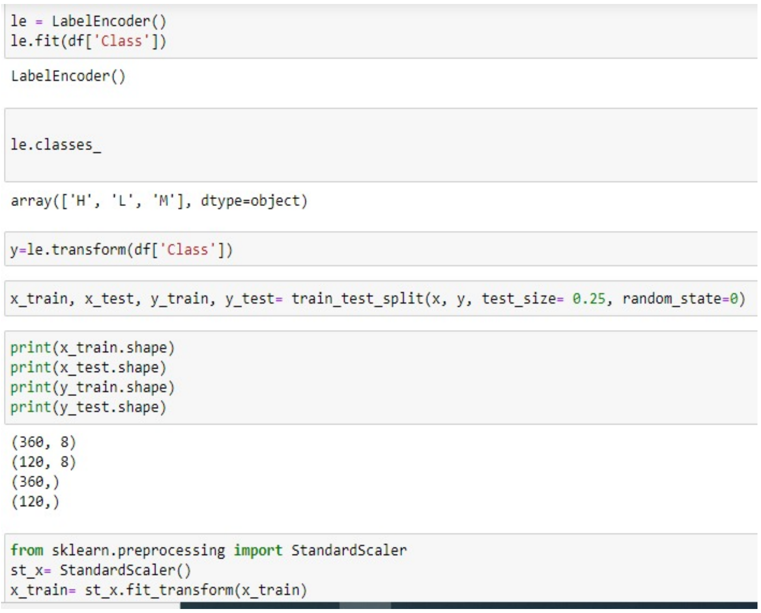
```
df.to_csv('train.csv')
```

```
x=df.iloc[:,0:8]
x.head()
```

	gender	mark_in_tenth	mark_in_plustwo	mark_in_ug	mark_in_pg	No_Backlogs	ParentschoolSatisfaction	StudentAbsenceDays
0	0	15	16	2	20	0	0	0
1	0	20	20	3	25	0	0	0
2	0	10	7	0	30	1	1	1
3	0	30	25	5	35	1	1	1
4	0	40	50	12	50	1	1	1

```
y=df.iloc[:,8:9]
y.head()
```

	Class
0	M
1	M



```
le = LabelEncoder()
le.fit(df['Class'])
```

```
LabelEncoder()
```

```
le.classes_
```

```
array(['H', 'L', 'M'], dtype=object)
```

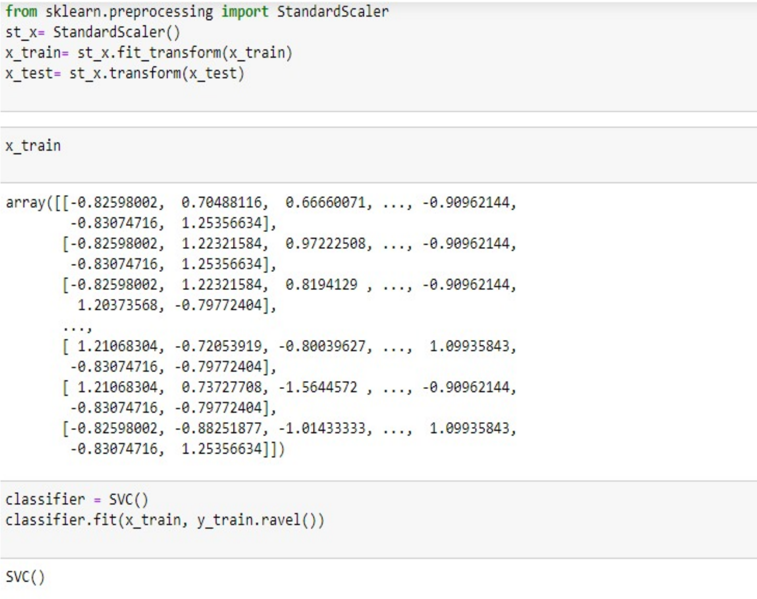
```
y=le.transform(df['Class'])
```

```
x_train, x_test, y_train, y_test= train_test_split(x, y, test_size= 0.25, random_state=0)
```

```
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(360, 8)
(120, 8)
(360,)
(120,)
```

```
from sklearn.preprocessing import StandardScaler
st_x= StandardScaler()
x_train= st_x.fit_transform(x_train)
```



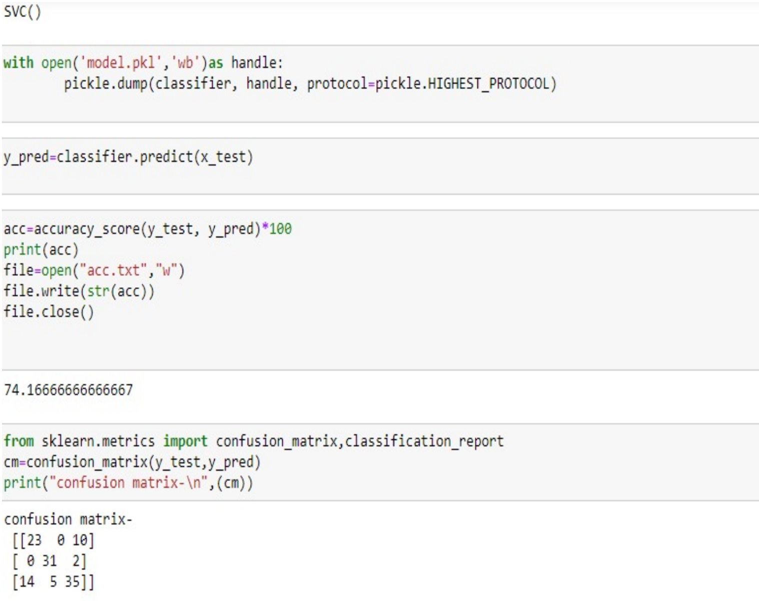
```
from sklearn.preprocessing import StandardScaler
st_x= StandardScaler()
x_train= st_x.fit_transform(x_train)
x_test= st_x.transform(x_test)
```

```
x_train
```

```
array([[-0.82598002,  0.70488116,  0.66660071, ..., -0.90962144,
        -0.83074716,  1.25356634],
       [-0.82598002,  1.22321584,  0.97222508, ..., -0.90962144,
        -0.83074716,  1.25356634],
       [-0.82598002,  1.22321584,  0.8194129 , ..., -0.90962144,
         1.20373568, -0.79772404],
       ...,
       [ 1.21068304, -0.72053919, -0.80039627, ...,  1.09935843,
        -0.83074716, -0.79772404],
       [ 1.21068304,  0.73727708, -1.5644572 , ..., -0.90962144,
        -0.83074716, -0.79772404],
       [-0.82598002, -0.88251877, -1.01433333, ...,  1.09935843,
        -0.83074716,  1.25356634]])
```

```
classifier = SVC()
classifier.fit(x_train, y_train.ravel())
```

```
SVC()
```



```
SVC()
```

```
with open('model.pkl','wb')as handle:
        pickle.dump(classifier, handle, protocol=pickle.HIGHEST_PROTOCOL)
```

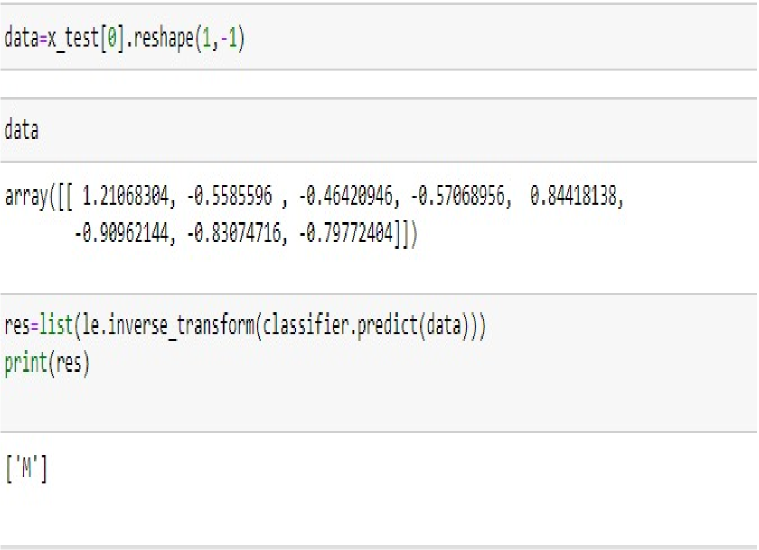
```
y_pred=classifier.predict(x_test)
```

```
acc=accuracy_score(y_test, y_pred)*100
print(acc)
file=open("acc.txt","w")
file.write(str(acc))
file.close()
```

```
74.16666666666667
```

```
from sklearn.metrics import confusion_matrix,classification_report
cm=confusion_matrix(y_test,y_pred)
print("confusion matrix-\n",(cm))
```

```
confusion matrix-
 [[23  0 10]
 [ 0 31  2]
 [14  5 35]]
```



```
data=x_test[0].reshape(1,-1)
```

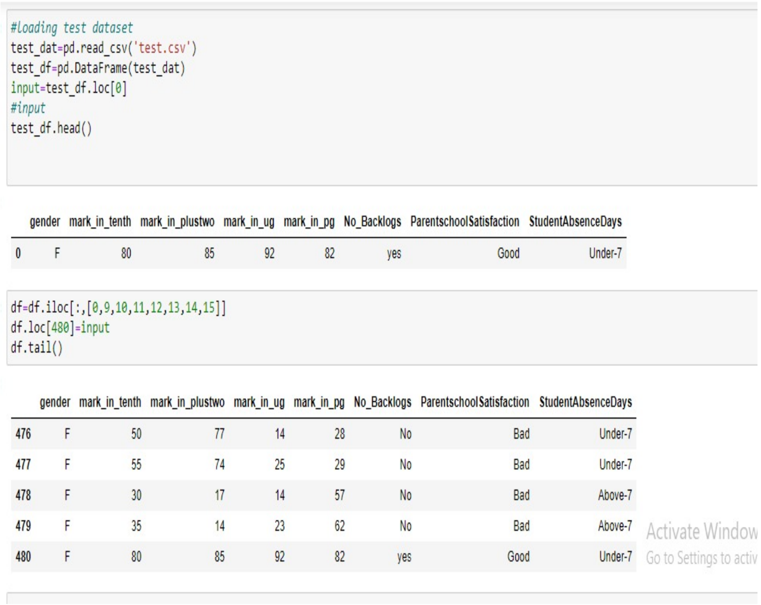
```
data
```

```
array([[ 1.21068304, -0.5585596 , -0.46420946, -0.57068956,  0.84418138,
        -0.90962144, -0.83074716, -0.79772404]])
```

```
res=list(le.inverse_transform(classifier.predict(data)))
print(res)
```

```
['M']
```

TEST

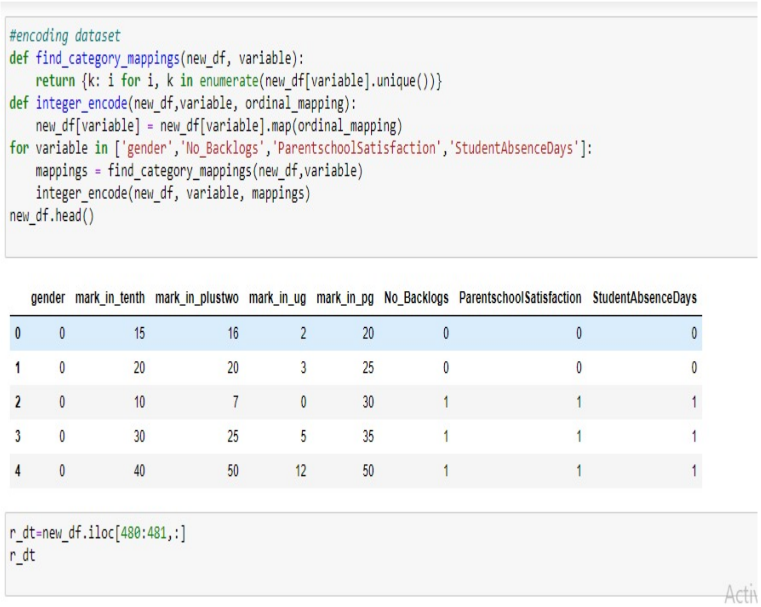


```
#loading test dataset
test_dat=pd.read_csv('test.csv')
test_df=pd.DataFrame(test_dat)
input=test_df.loc[0]
#input
test_df.head()
```

	gender	mark_in_tenth	mark_in_plustwo	mark_in_ug	mark_in_pg	No_Backlogs	ParentschoolSatisfaction	StudentAbsenceDays
0	F	80	85	92	82	yes	Good	Under-7

```
df=df.iloc[:,[0,9,10,11,12,13,14,15]]
df.loc[480]=input
df.tail()
```

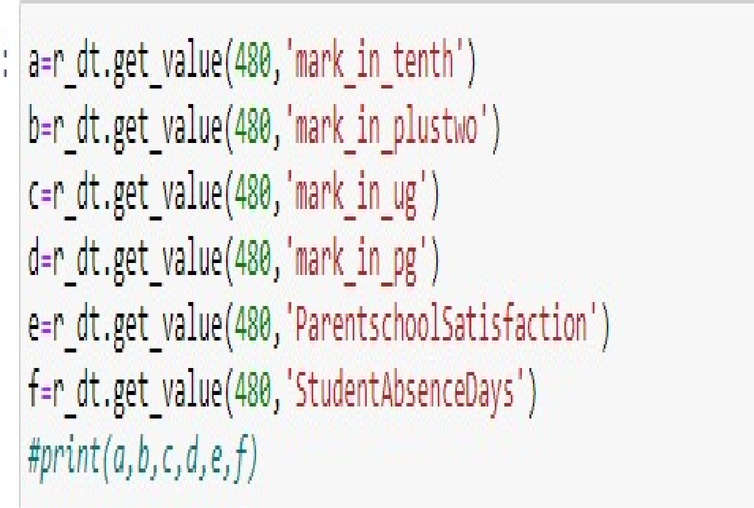
	gender	mark_in_tenth	mark_in_plustwo	mark_in_ug	mark_in_pg	No_Backlogs	ParentschoolSatisfaction	StudentAbsenceDays
476	F	50	77	14	28	No	Bad	Under-7
477	F	55	74	25	29	No	Bad	Under-7
478	F	30	17	14	57	No	Bad	Above-7
479	F	35	14	23	62	No	Bad	Above-7
480	F	80	85	92	82	yes	Good	Under-7



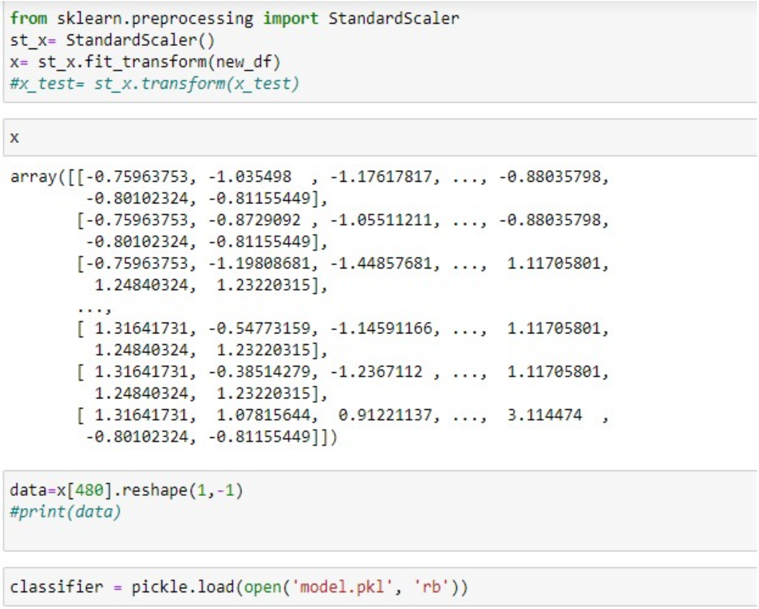
```
#encoding dataset
def find_category_mappings(new_df, variable):
    return {k: i for i, k in enumerate(new_df[variable].unique())}
def integer_encode(new_df,variable, ordinal_mapping):
    new_df[variable] = new_df[variable].map(ordinal_mapping)
for variable in ['gender','No_Backlogs','ParentschoolSatisfaction','StudentAbsenceDays']:
    mappings = find_category_mappings(new_df,variable)
    integer_encode(new_df, variable, mappings)
new_df.head()
```

	gender	mark_in_tenth	mark_in_plustwo	mark_in_ug	mark_in_pg	No_Backlogs	ParentschoolSatisfaction	StudentAbsenceDays
0	0	15	16	2	20	0	0	0
1	0	20	20	3	25	0	0	0
2	0	10	7	0	30	1	1	1
3	0	30	25	5	35	1	1	1
4	0	40	50	12	50	1	1	1

```
r_dt=new_df.iloc[480:481,:]
r_dt
```



```
a=r_dt.get_value(480,'mark_in_tenth')
b=r_dt.get_value(480,'mark_in_plustwo')
c=r_dt.get_value(480,'mark_in_ug')
d=r_dt.get_value(480,'mark_in_pg')
e=r_dt.get_value(480,'ParentschoolSatisfaction')
f=r_dt.get_value(480,'StudentAbsenceDays')
#print(a,b,c,d,e,f)
```



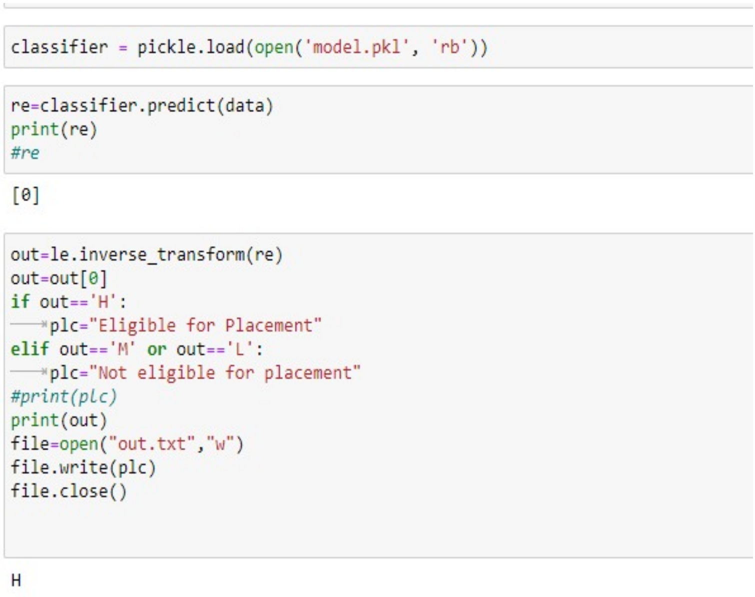
```
from sklearn.preprocessing import StandardScaler
st_x= StandardScaler()
x= st_x.fit_transform(new_df)
#x_test= st_x.transform(x_test)
```

```
x
```

```
array([[-0.75963753, -1.035498  , -1.17617817, ..., -0.88035798,
        -0.80102324, -0.81155449],
       [-0.75963753, -0.8729092 , -1.05511211, ..., -0.88035798,
        -0.80102324, -0.81155449],
       [-0.75963753, -1.19808681, -1.44857681, ...,  1.11705801,
         1.24840324,  1.23220315],
       ...,
       [ 1.31641731, -0.54773159, -1.14591166, ...,  1.11705801,
         1.24840324,  1.23220315],
       [ 1.31641731, -0.38514279, -1.2367112 , ...,  1.11705801,
         1.24840324,  1.23220315],
       [ 1.31641731,  1.07815644,  0.91221137, ...,  3.114474  ,
        -0.80102324, -0.81155449]])
```

```
data=x[480].reshape(1,-1)
#print(data)
```

```
classifier = pickle.load(open('model.pkl', 'rb'))
```



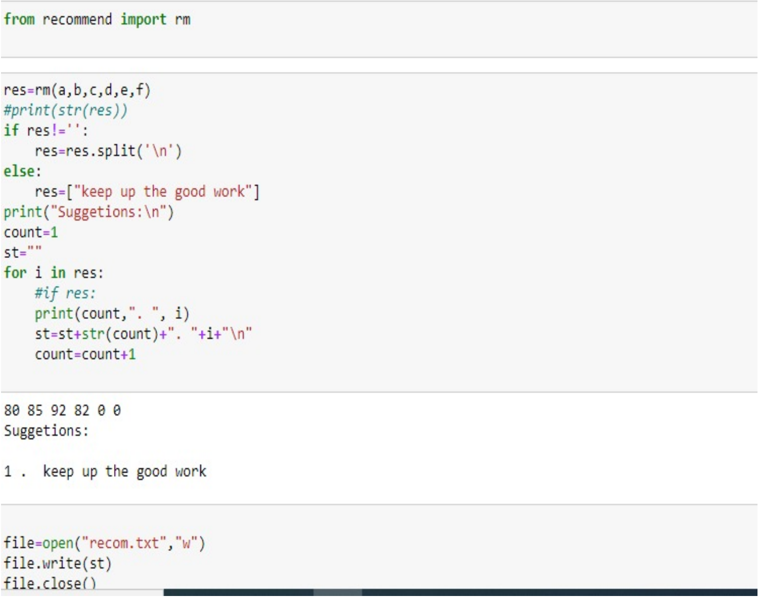
```
classifier = pickle.load(open('model.pkl', 'rb'))
```

```
re=classifier.predict(data)
print(re)
#re
```

```
[0]
```

```
out=le.inverse_transform(re)
out=out[0]
if out=='H':
    plc="Eligible for Placement"
elif out=='M' or out=='L':
    plc="Not eligible for placement"
#print(plc)
print(out)
file=open("out.txt","w")
file.write(plc)
file.close()
```

```
H
```



```
from recommend import rm
```

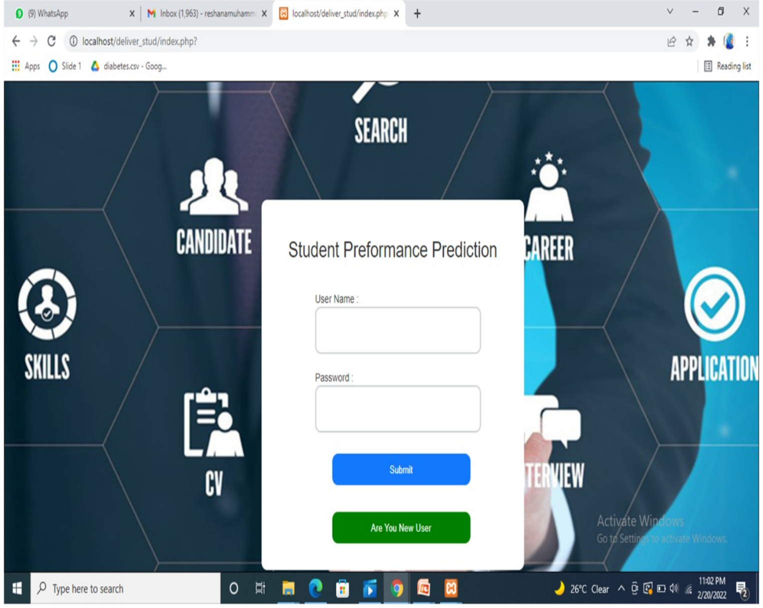
```
res=rm(a,b,c,d,e,f)
#print(str(res))
if res!='':
    res=res.split('\n')
else:
    res=["keep up the good work"]
print("Suggetions:\n")
count=1
st=""
for i in res:
    #if res:
    print(count,". ", i)
    st=st+str(count)+". "+i+"\n"
    count=count+1
```

```
80 85 92 82 0 0
Suggetions:

1 .  keep up the good work
```

```
file=open("recom.txt","w")
file.write(st)
file.close()
```

Chapter 8

SCREEN SHOTS

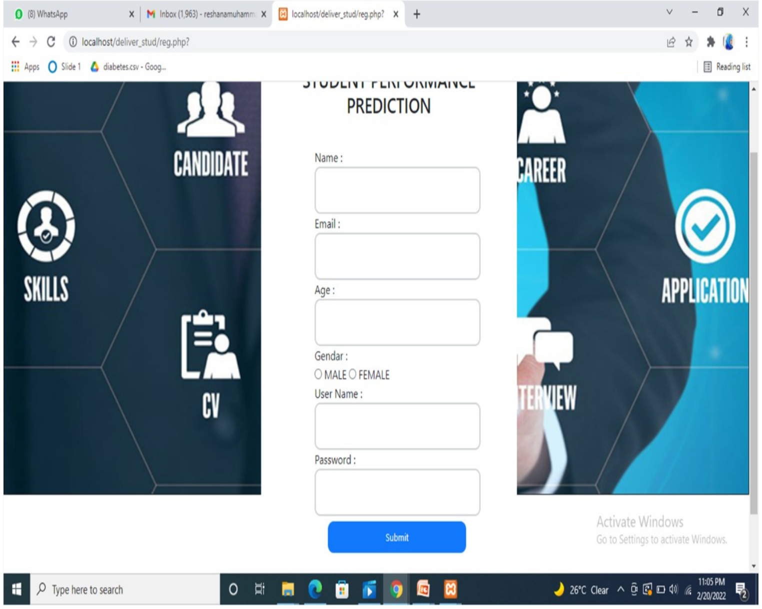


Student Preformance Prediction

User Name :

Password :

Figure 8.1: LOGIN



STUDENT PERFORMANCE PREDICTION

Name :

Email :

Age :

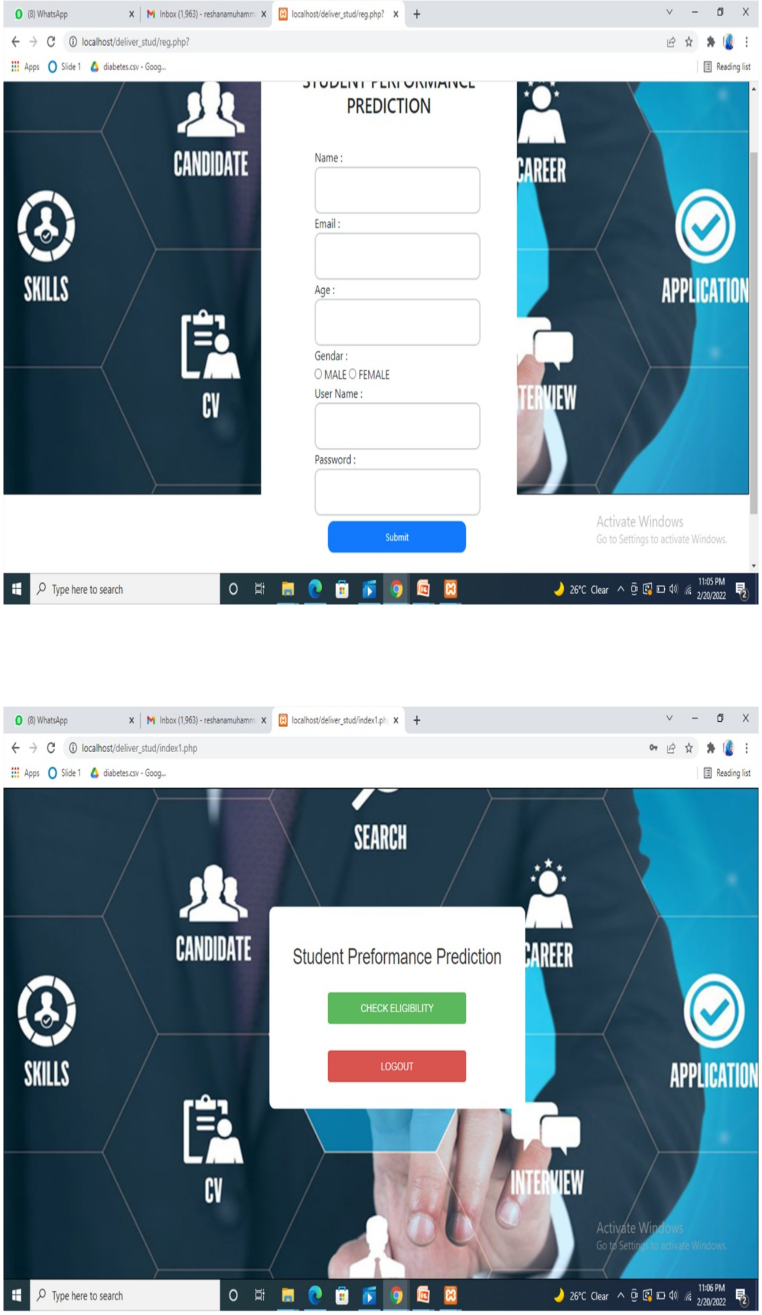
Gendar :

○ MALE ○ FEMALE

User Name :

Password :

Figure 8.2: USER REGISTRAION



STUDENT PERFORMANCE PREDICTION

Name :

Email :

Age :

Gendar :
○ MALE ○ FEMALE

User Name :

Password :

Submit

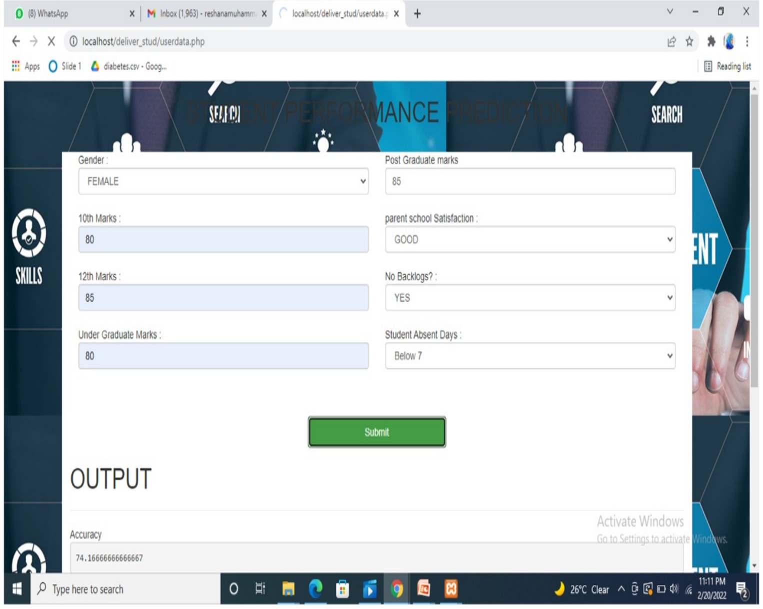
Student Preformance Prediction

CHECK ELIGIBILITY

LOGOUT

Figure 8.3: CHECK ACADEMIC PERFORMANCE

PREDICTING ACADEMIC PEROFRMANCE



Gender : FEMALE

Post Graduate marks 85

10th Marks : 80

parent school Satisfaction : GOOD

12th Marks : 85

No Backlogs? : YES

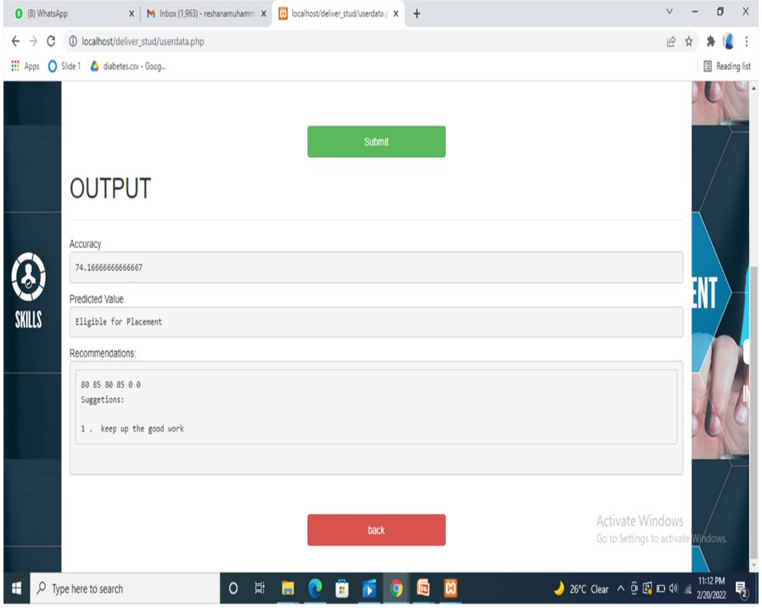
Under Graduate Marks : 80

Student Absent Days : Below 7

OUTPUT

Accuracy

74.16666666666667



OUTPUT

Accuracy

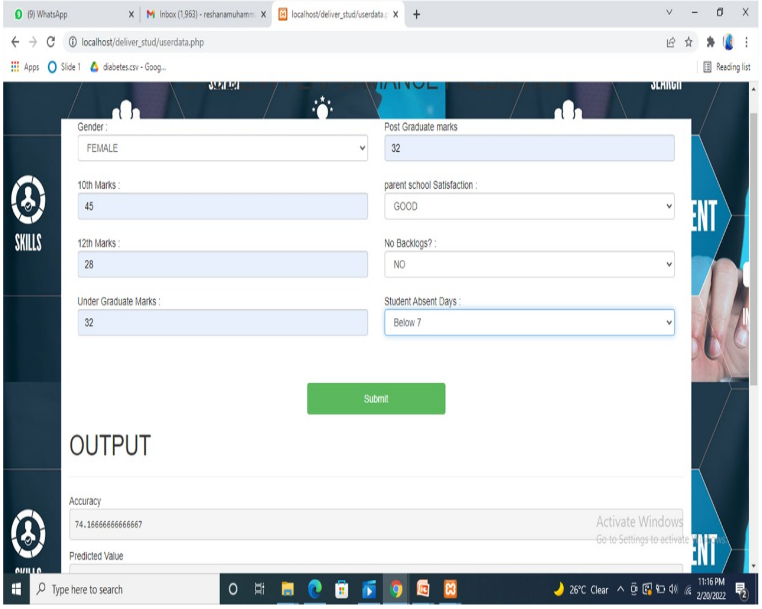
74.16666666666667

Predicted Value

Eligible for Placement

Recommendations:

```
80 85 80 85 0 0
Suggetions:

1 .  keep up the good work
```



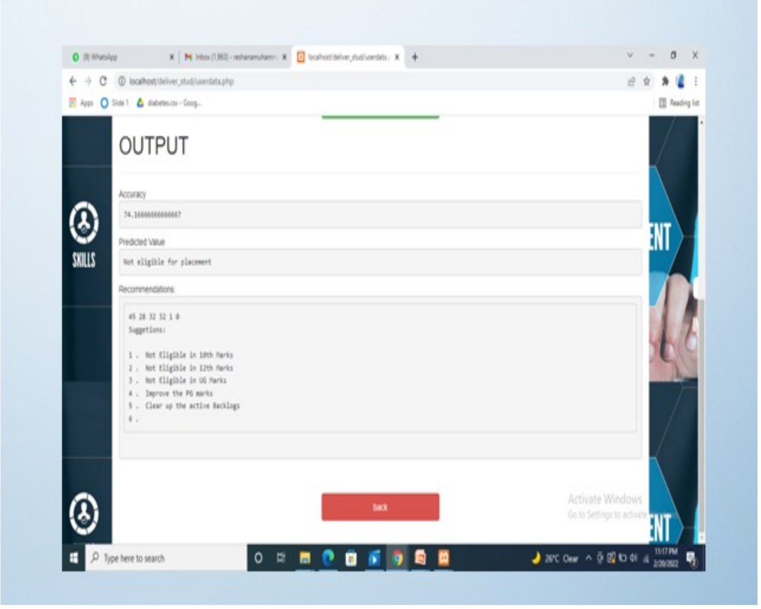
Gender : FEMALE	Post Graduate marks 32
10th Marks : 45	parent school Satisfaction : GOOD
12th Marks : 28	No Backlogs? : NO
Under Graduate Marks : 32	Student Absent Days : Below 7

OUTPUT

Accuracy

```
74.16666666666667
```

Predicted Value



OUTPUT

Accuracy

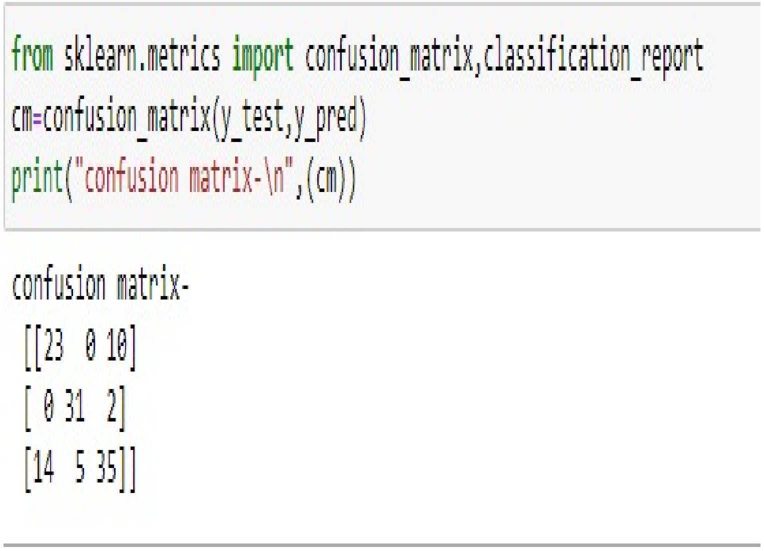
```
74.16666666666667
```

Predicted Value

```
Not eligible for placement
```

Recommendations

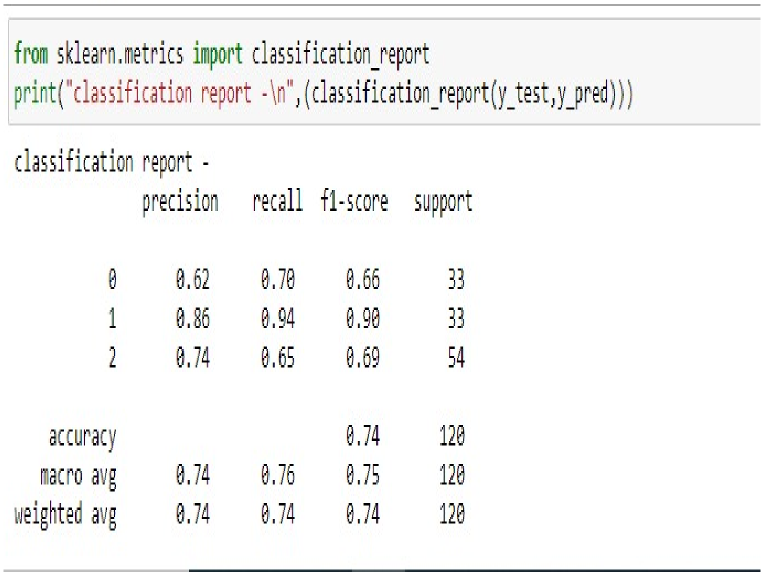
```
45 28 32 32 1 0
Suggetions:

1 .  Not Eligible in 10th Marks
2 .  Not Eligible in 12th Marks
3 .  Not Eligible in UG Marks
4 .  Improve the PG marks
5 .  Clear up the active Backlogs
6 .
```



```
from sklearn.metrics import confusion_matrix,classification_report
cm=confusion_matrix(y_test,y_pred)
print("confusion matrix-\n",(cm))
```

```
confusion matrix-
 [[23  0 10]
 [ 0 31  2]
 [14  5 35]]
```

Figure 8.4: SVM CONFUSION MATRIX



```
from sklearn.metrics import classification_report
print("classification report -\n",(classification_report(y_test,y_pred)))
```

```
classification report -
               precision    recall  f1-score   support

           0       0.62      0.70      0.66        33
           1       0.86      0.94      0.90        33
           2       0.74      0.65      0.69        54

    accuracy                           0.74       120
   macro avg       0.74      0.76      0.75       120
weighted avg       0.74      0.74      0.74       120
```

Figure 8.5: SVM CLASSIFICATION REPORT

Chapter 9

REFERENCES

[1] https://europepmc.org/article/ppr/ppr342884

[2] https://ieeexplore.ieee.org/abstract/document/6826192

[3]https://www.geeksforgeeks.org/

[4] https://www.javatpoint.com

[5]https://ieeexplore.ieee.org/document/9167547

[6] https://ieeexplore.ieee.org/abstract/document/8405502

[7]https://youtu.be/JAyfY21nvpc