

Student Placement Prediction using Machine Learning

Minor Project II

Submitted by:

Samarth Sajwan (9918103022)
Rudraksh Bhardwaj (9918103020)
Revaan Mishra (9918103016)

Under the supervision of:
Dr. Shruti Jaiswal



Department of CSE/IT
Jaypee Institute of Information Technology University, Noida

MAY 2021

Table of Contents

	Page No.
<i>Abstract</i>	<i>1</i>
<i>List of Figures</i>	<i>2</i>
Chapter 1: INTRODUCTION	3
1.1 Introduction	
Chapter 2: Background Study	4
2.1 GENERAL	4-5
Chapter 3: Requirement Analysis	6-7
3.1 Functional Requirements	6
3.2 Hardware Requirements	7
3.3 Software Requirements	7
Chapter 4: Detailed Design	8-11
4.1 Flowchart	8
4.2 Pre-Processing	9
4.3 Algorithms Used	10-11
Chapter 5: Implementation	12-13
5.1 GENERAL	12-13
Chapter 6: Experimental Results and Analysis	14-17
6.1 Correlation Heatmap	14
6.1 Exploratory Data Analysis	15-17
Chapter 7: Conclusion	18-19
7.1 Algorithms Comparison	18-19
7.1 Future Scope	s19
8.References	20

ACKNOWLEDGEMENT

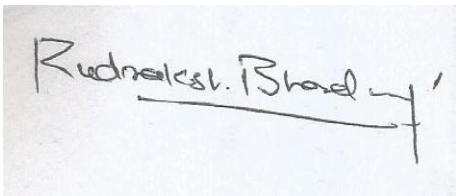
I would like to place on record my deep sense of gratitude to Dr Shruti Jaiswal, Assistant Professor, Jaypee Institute of Information Technology, India for her generous guidance, help and useful suggestions.

I express my sincere gratitude to Dr Mukesh Saraswat, Dept. of Computer Science and Engineering, for his stimulating guidance, continuous encouragement and supervision throughout the course of present work.

I also wish to extend my thanks to Hare Krishna and other classmates for their insightful comments and constructive suggestions to improve the quality of this project work.

A handwritten signature in black ink, appearing to be 'Samarth Sajwan', written on a light-colored background.

Samarth Sajwan (9918103022)

A handwritten signature in black ink, appearing to be 'Rudraksh Bhardwaj', written on a light-colored background.

Rudraksh Bhardwaj (9918103020)

A handwritten signature in purple ink, appearing to be 'Revaan Mishra', written on a light-colored background.

Revaan Mishra (9918103016)

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and beliefs, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma from a university or other institute of higher learning, except where due acknowledgment has been made in the text.

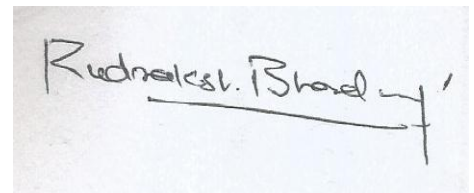
Place: Jaypee Institute of Information Technology, Noida

Date: 5 May, 2021



Name: Samarth Sajwan

Enrolment No.: 9918103022



Name: Rudraksh Bhardwaj

Enrolment No.: 9918103020



Name: Revaan Mishra

Enrolment No.: 9918103016

CERTIFICATE

This is to certify that the work titled “**Student Placement Prediction Using Machine Learning**” submitted by Samarth Sajwan, Rudraksh Bhardwaj and Revaan Mishra of B. Tech of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of any other degree or diploma.



Dr. Shruti Jaiswal
Assistant Professor
5th May, 2021

Abstract

One of the biggest challenges that higher learning institutions face today is to improve the placement performance of students. The placement prediction is more complex when the complexity of educational entities increases. Educational institutes look for more efficient technology that assist better management and support decision making procedures or assist them to set new strategies. One of the effective ways to address the challenges for improving the quality is to provide new knowledge related to the educational processes and entities to the managerial system. With the machine learning techniques, the knowledge can be extracted from operational and historical data that resides within the educational organization's databases using. The dataset for system implementation contains information about past data of students. These data are used for training the model for rule identification and for testing the model for classification. Our model will present a recommendation system that predicts the placement level of a student.

This model helps the placement cell within an organization to identify the prospective students and pay attention to and improve their technical as well as interpersonal skills. Furthermore, the students in pre-final and final years of their **B. Tech** course can also use this system to know their individual placement status that they are most likely to achieve. With this they can put in more hard work for getting placed in to the companies that belong to higher hierarchies.

List of Figures

- Fig. 1. Dataset
- Fig.2. Flowchart according to which work was done
- Fig.3. Using SciKit to Split the Dataset
- Fig.4. Splitting the Dataset to 30% Training Set and 70% Test Set
- Fig.5. Using sklearn. feature_selection to obtain results
- Fig.6. Python Code to plot the graph
- Fig.7. Plotting of feature selection
- Fig.8. Correlation Heatmap
- Fig.9. Number of Students v/s Final Grade
- Fig.10. Final Grade by Weekend Alcohol Consumption
- Fig.11. Final Grade by Frequency of Going Out
- Fig.12. Python Code to plot the graph
- Fig.13. Final Grade by Living Area
- Fig.14. Model Selection

1.Introduction

The main aim of every academia enthusiast is placement in a reputed MNC's and even the reputation and every year admission of Institute depends upon placement that it provides to their students. So, any system that will predict the placements of the students will be a positive impact on an institute and increase strength and decreases some workload of any institute's training and placement office (TPO). With the help of Machine Learning techniques, the knowledge can be extracted from past placed students and placement of upcoming students can be predicted. Data used for training is taken from the same institute for which the placement prediction is done. Suitable data pre-processing methods are applied along with the feature selections. Some Domain expertise is used for pre-processing as well as for outliers that grab in the dataset. We have used various Machine Learning Algorithms like Logistic, SVM, KNN, Decision Tree, Random Forest and advance techniques like Bagging, Boosting and Voting Classifier

Nowadays Placement plays an important role in this world full of unemployment. Even the ranking and rating of institutes depend upon the amount of average package and amount of placement they are providing.

So basically, main objective of this model is to predict whether the student might get placement or not. Different kinds of classifiers were applied i.e., Logistic Regression, SVM, Decision Tree, Random Forest, KNN, AdaBoost, Gradient Boosting and XGBoost. For this all over academics of students are taken under consideration. As placements activity take place in last year of academics so last year semesters are not taken under consideration

2. Background Study

[1] "Data Mining Approach for Predicting Student and Institution's Placement Percentage", Professor. Ashok M Assistant Professor Apoorva A ,2016 International Conference on Computational Systems and Information Systems for Sustainable Solutions

In this paper author has used the data mining technique for the prediction of the student's placement. For the prediction of student's placement author has divided the data into the two segments, first segment is the training segment which is historic data of passed out students. Another segment consists of current data of students, based on the historic data author has designed the algorithm for calculating the placement chances. Author has used the various data mining algorithms such as decision tree, Naive Bayes, neural network and the proposed algorithm were applied, and decision are made with the help of confusion matrix.

[2] "Student Placement Analyzer: A Recommendation System Using Machine Learning", Senthil Kumar Thangavel , Divya Bharathi P, Abijith Sankar, International Conference on Advanced Computing and Communication Systems (ICACCS -2017), Jan. 06 - 07, 2017, Coimbatore, INDIA

In this paper author is concern about the challenges face by any institute regarding the placement. The placement prediction is very complex when the number of the entities increases in any institute. With the help of machine learning this complex problem of prediction can be easily solved. In this paper all the academic record of student is taken into consideration. Various classification and data making algorithms are used such as Naïve Bayes, Decision Tree, SVM and Regressions. After the prediction of the students can be placed in of the given category that is core company, dream company or support services.

[3] "A Placement Prediction System Using K-Nearest Neighbors Classifier", Animesh Giri, M V ignesh V Bhagavath, Bysani Pruthvi, Naini Dubey, Second International Conference on Cognitive Computing and Information Processing (CCIP), 2016

The placement prediction system predicts the probability of students getting placed in various companies by applying K-Nearest Neighbors classification. The result obtained is also compared with the results obtained from other machine learning models like Logistic Regression and SVM. The academic history of student along with their skill sets like programming skills, communication skills, analytical skills and team work is considered which is tested by companies

ISSN [ONLINE]: 2395-1052 during recruitment process. Data of past two batches are taken for this system.

[4]"Class Result Prediction using Machine Learning", Pushpa S K, Associate Professor, Manjunath T N, Professor and Head, Mrunal T V, Amartya Singh, C Suhas, International Conference On Smart Technology for Smart Nation, 2017

In this paper, the result of a class is predicted using machine learning. Performance of students in past semester along with scores of internal examinations of the current semester is considered to predict whether the student passes or fails in the current semester before attempting the final examination. The author uses SVM, Naive Bayes, Random Forest Classifier and Gradient Boosting to compute the result. Boosting is an ensemble learning algorithm which combines various learning algorithm to obtain better predictive performance.

[5]"Student Placement Analyzer : A Recommendation System Using Machine Learning", Apoorva Rao R, Deeksha K C, Vishal Prajwal R, Vrushak K, Nandini, JARIE-ISSN(O)-2395-4396

Now-a-days institutions are facing many challenges regarding student placements. For educational institutions it is much difficult task to keep record of every single student and predict the placement of student manually. To overcome these challenges, concept of machine learning and various algorithms are explored to predict the result of class students. For this purpose, training data set is historical data of past students and this is used to train the model. This software system predicts placement status in 5 categories viz dream company, core company, mass recruiter, not eligible and not interested in placements. This system is also helpful to weaker students. Institutions can provide extra care towards weaker students so that they can improve their performance. By use Naïve Bayes algorithm all the data will be monitor and appropriate decision will be provided.

3.Requirement Analysis

3.1. Functional Requirements

- **Our goal is to create a proficient and relevant recruitment system for predicting the placement of a student using ML algorithms.**
- We will try to use the following ML Algorithms to obtain a model with the most accurate results for placement prediction:
 - Support Vector Classifier
 - Decision Tree Classifier
 - Random Forest
 - Logistic Regression
 - ADA Boost Classifier
 - Stochastic Gradient Classifier

Dataset Characteristics:

- The dataset we are using was obtained from Kaggle. The dataset for the model consists a small sample of students from a particular institute. The dataset consisted of 32 features with our final variable (placement label) acting as our outcome variable. Sample dataset is shown in Fig. 1.
- Two datasets are used regarding the performance in two distinct streams: Electronics and Computer Science. This data explores student achievement in undergrad education of an institute. The data attributes include student grades, demographic, social and institute related features and it was collected by using institute reports and questionnaires.

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	reason	nursery	internet	guardian.x
1	GP	F	15	R	GT3	T	1	1	at_home	other	home	yes	yes	mother
2	GP	F	15	R	GT3	T	1	1	other	other	reputation	no	yes	mother
3	GP	F	15	R	GT3	T	2	2	at_home	other	reputation	yes	no	mother
4	GP	F	15	R	GT3	T	2	4	services	health	course	yes	yes	mother
5	GP	F	15	R	GT3	T	3	3	services	services	reputation	yes	yes	other
6	GP	F	15	R	GT3	T	3	4	services	health	course	yes	yes	mother
7	GP	F	15	R	GT3	T	3	4	services	teacher	course	yes	yes	father
8	GP	F	15	R	LE3	T	2	2	health	services	reputation	yes	yes	mother
9	GP	F	15	R	LE3	T	3	1	other	other	reputation	no	yes	father
10	GP	F	15	U	GT3	A	3	3	other	health	reputation	yes	no	father
11	GP	F	15	U	GT3	A	4	3	services	services	reputation	yes	yes	mother
12	GP	F	15	U	GT3	T	1	1	at_home	other	course	no	yes	mother
13	GP	F	15	U	GT3	T	1	1	other	other	home	no	yes	father
14	GP	F	15	U	GT3	T	1	1	other	services	course	yes	yes	father
15	GP	F	15	U	GT3	T	1	2	at_home	other	course	no	yes	mother
16	GP	F	15	U	GT3	T	1	2	at_home	services	course	no	yes	mother

Fig.1. dataset samples.

HARDWARE & SOFTWARE REQUIREMENTS

3.2. Hardware Requirements

- Processors: Intel® Core™ i5 processor 4300M at 2.60 GHz or 2.59 GHz (1 socket, 2 cores, 2 threads per core), 8 GB of DRAM Intel® Xeon® processor E5-2698 v3 at 2.30 GHz (2 sockets, 16 cores each, 1 thread per core), 64 GB of DRAM Intel® Xeon Phi™ processor 7210 at 1.30 GHz (1 socket, 64 cores, 4 threads per core), 32 GB of DRAM, 16 GB of MCDRAM (flat mode enabled)
- Disk space: 2 to 3 GB

3.3. Software Requirements

- Operating systems: Windows* 7 or later, macOS, and Linux
- Python* versions: 2.7.X, 3.6.
- Jupyter Notebook
- Python Modules: NumPy, Pandas, Matplotlib, Seaborn

4. DETAILED DESIGN

4.1. FLOWCHART

The flowchart shown in Fig. 2 represents the flow of our work, which will be followed for implementation.

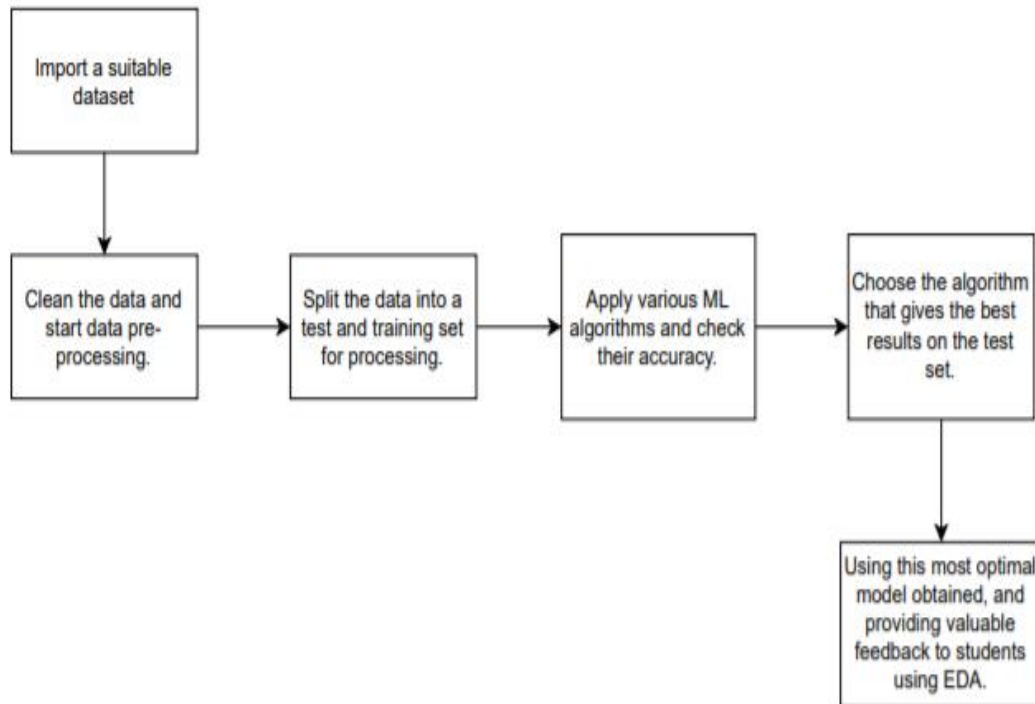


Fig .2. Flowchart according to which work was done

4.2. Pre- Processing the Dataset

First, we have to import a suitable dataset to work on, cleaning the data, and pre-process it so that it can be worked upon by the machine learning algorithms.

We can pre-process the data by using the following steps:

- The two datasets that are used regarding the performance in two distinct streams Electronics and Computer Science were merged using the **pd.concat()** function of the **Pandas** library in Python 3.9.

```
df = pd.concat([mat,por])
```

- The variable final score (target variable) is used to make a new categorical variable called Final Grade so as to easily perform different classification techniques on the data.

```
df['final_grade'] = 'na'  
df.loc[(df.final_score >= 15) & (df.final_score <= 20), 'final_grade'] = 'good'  
df.loc[(df.final_score >= 10) & (df.final_score <= 14), 'final_grade'] = 'fair'  
df.loc[(df.final_score >= 0) & (df.final_score <= 9), 'final_grade'] = 'poor'  
df.head(5)
```

- The final grade variable is label encoded and the dataset is split into training set and testing set respectively using the **train_test_split** using **sklearn.cross_validation** library in Python 3.9.

```
X = dfd.drop('final_grade',axis=1)  
y = dfd.final_grade  
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3)
```

4.3. The methods and algorithms of Machine Learning we have decided to apply on our dataset are:

4.3.1. Support vector Machine

One of the most popular and spectacular supervised learning techniques with related learning algorithms for treatment classification and regression tasks in patterns is SVM. SVM is a classification machine learning algorithm based on hinge function as shown in the figure., where z is a label from 0 to 1, $w \cdot I - b$ is the output, w and b are coefficients of linear classification, and I is an input vector. The loss function to be minimized can be implemented below:

$$h_j = \max(0, 1 - z_j(w \cdot I_j - b))$$

$$loss = \frac{1}{n} \sum_{i=1}^n \max(0, h_i)$$

4.3.2. Decision tree

The decision tree is the classification model of computation based on entropy function and information gain. Entropy computes the amount of uncertainty in data as shown in Fig.3. Where D is current data, and q is a binary label from 0 to 1, and $p(x)$ is the proportion of q label. To measure the difference of entropy from data, we calculate information gain (I) as illustrated below:

$$E(D) = \sum_{i=1}^m -p(q_i) \cdot \log(p(q_i))$$

$$I = E(D) - \sum_{v \in D} p(v) E(v)$$

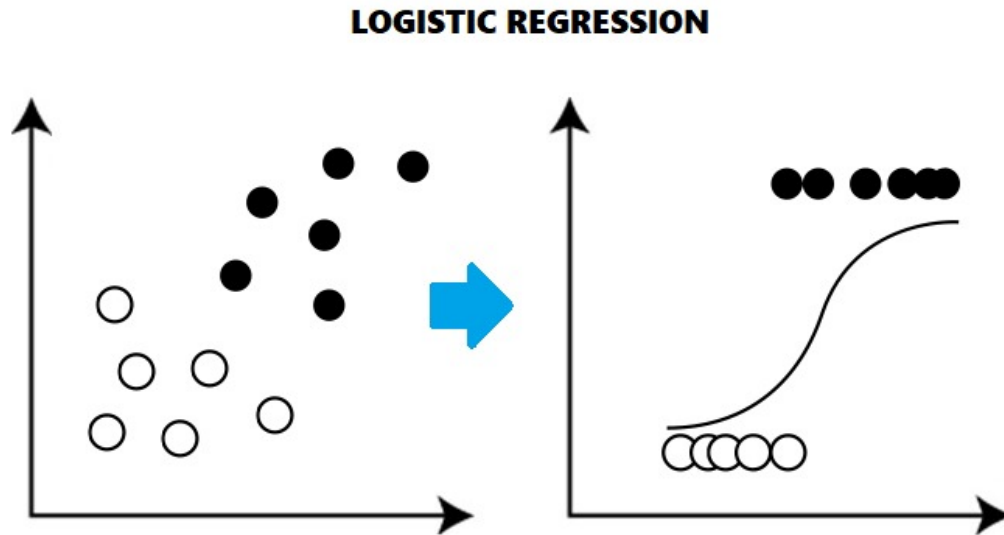
4.3.3. Random Forest

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

$$\bar{z} = \sum_{i=1}^M \alpha_i z_i$$

4.3.4. Logistic Regression

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).



4.3.5. Ada Boost Classification

An AdaBoost classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases. The mathematical formula behind Adaboost classification is :

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right)$$

5.IMPLEMENTATION

Step by step implementation of our work is done in Python which is explained below with code snippet.

1)We use the **train_test_split** Function of the **SciKit Library** to split our data set into two halves the **Training Set and the Test Set** as shown in Fig. 3.

```
from sklearn.cross_validation import train_test_split
X = dfd.drop('final_grade', axis=1)
y = dfd.final_grade
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
```

Fig .3. Using SciKit to Split the Dataset

2)We have kept 30% of our data for the training of our model and the remaining 70% is used for the testing. Test data size setting is shown in Fig. 4.

```
test_size=0.3
```

Fig.4. Splitting the Dataset to 30% Training Set and 70% Test Set

3) Finding optimal number of features to be used in the model using the **sklearn library** of Python is shown in Fig. 5.

```
from sklearn.feature_selection import SelectKBest, chi2

ks=[]
for i in range(1,58):
    sk = SelectKBest(chi2, k=i)
    x_new = sk.fit_transform(X_train,y_train)
    x_new_test=sk.fit_transform(X_test,y_test)
    l = lr.fit(x_new, y_train)
    ll = l.score(x_new_test, y_test)
    ks.append(ll)

ks = pd.Series(ks)
ks = ks.reindex(list(range(1,58)))
ks
```

Fig.5. Using sklearn.feature_selection for obtaining results

4) Code for plotting the result obtained from the above Python code is shown in Fig. 6. And corresponding plot is shown in Fig. 7

```
plt.figure(figsize=(10,5))
ks.plot.line()
plt.title('Feature Selction', fontsize=20)
plt.xlabel('Number of Feature Used', fontsize=16)
plt.ylabel('Prediction Accuracy', fontsize=16)
```

Fig.6. Python Code to plot the graph

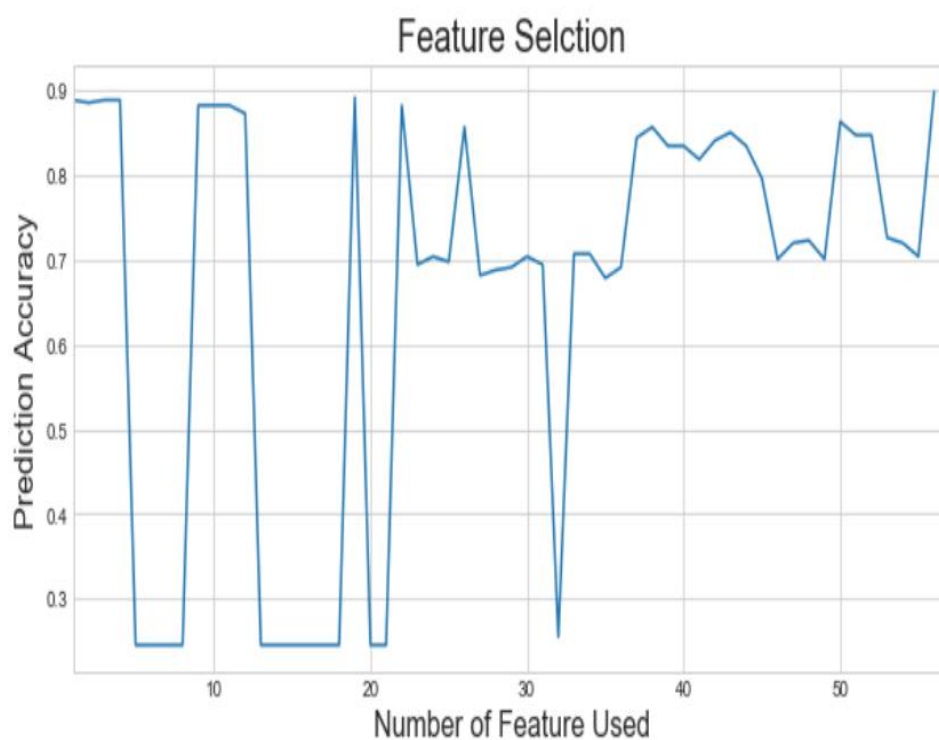


Fig.7. Plotting of Feature Selection

6. Experimental Result and Analysis

6.1 Correlation Matrix

A correlation heatmap was plotted to check the multicollinearity and correlation between different variables, which is shown in Fig. 8.

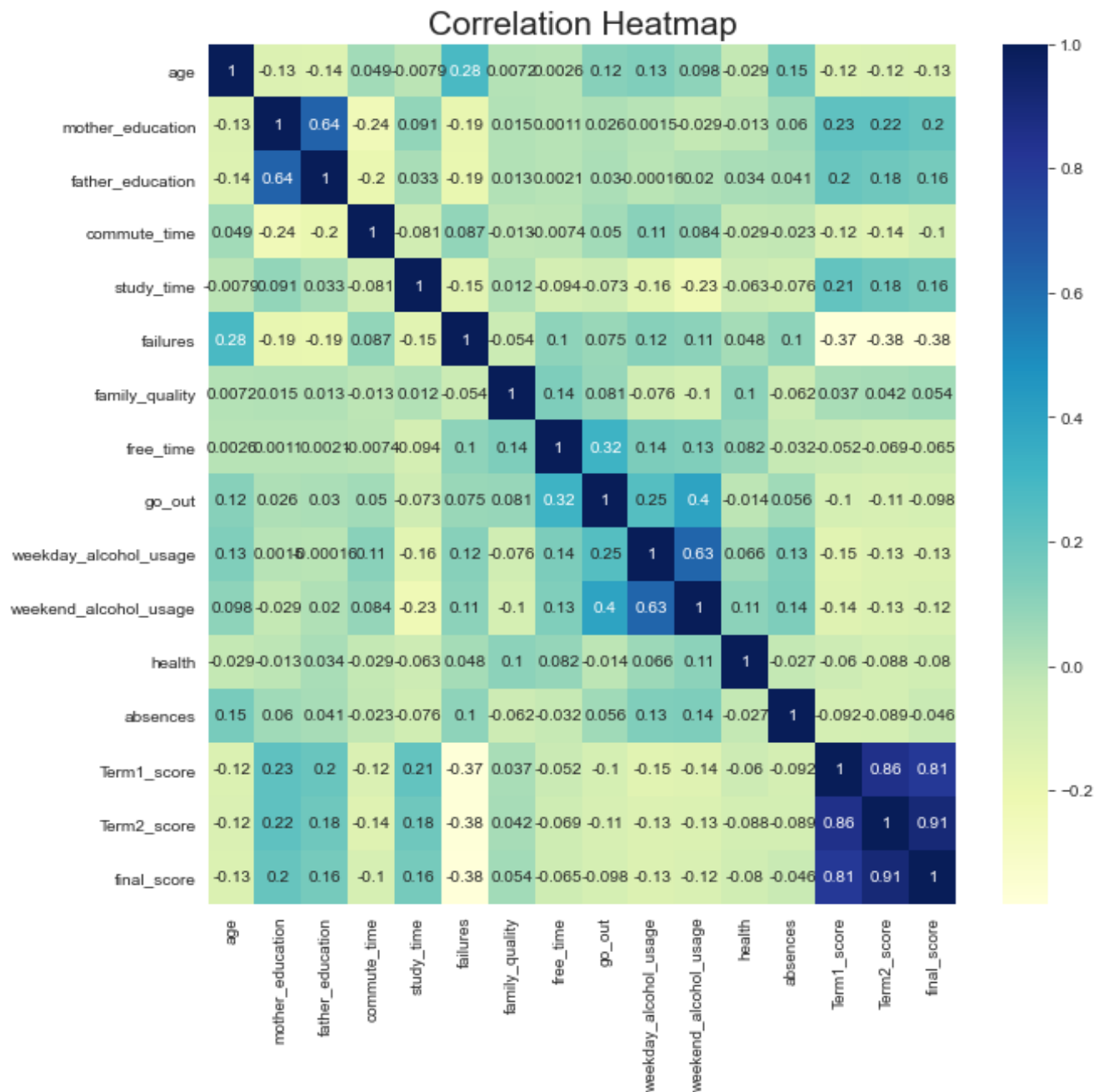


Fig.8. Correlation Heatmap

6.2 Exploratory Data Analysis (EDA)

In this part of our report, exploratory data analysis of our work is explained.

6.2.1 Analysis of Final Grade of student with the number of students attached to each: Analysis of student final grade is shown in Fig. 9

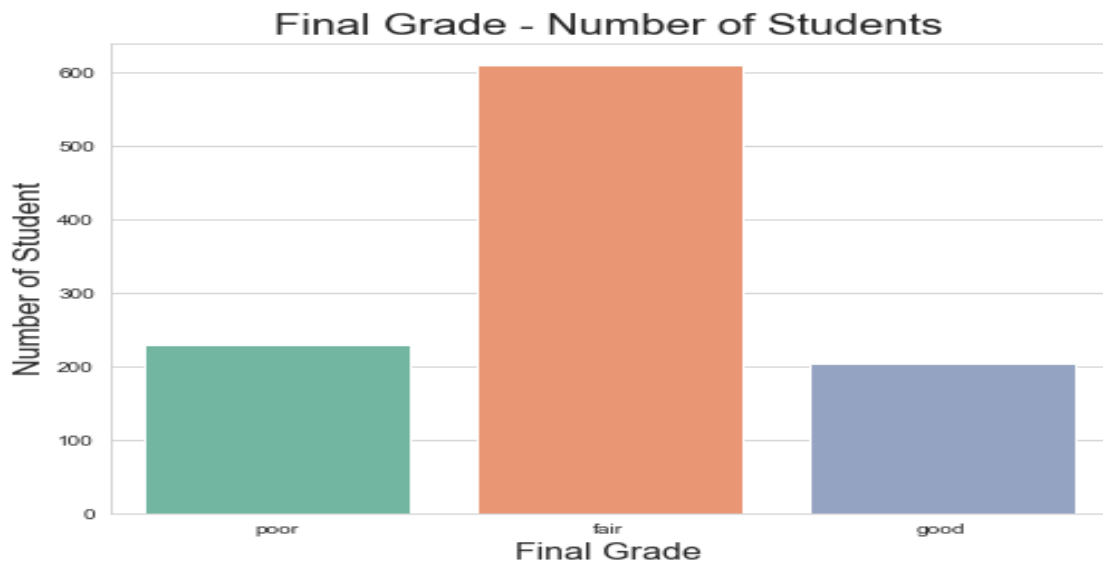


Fig.9. Number of Students v/s Final Grade

6.2.2 Analysis of final grade on the basis of alcohol consumption by a student on the weekend: Fig. 10 shows the impact of Alcohol consumption on final grade.

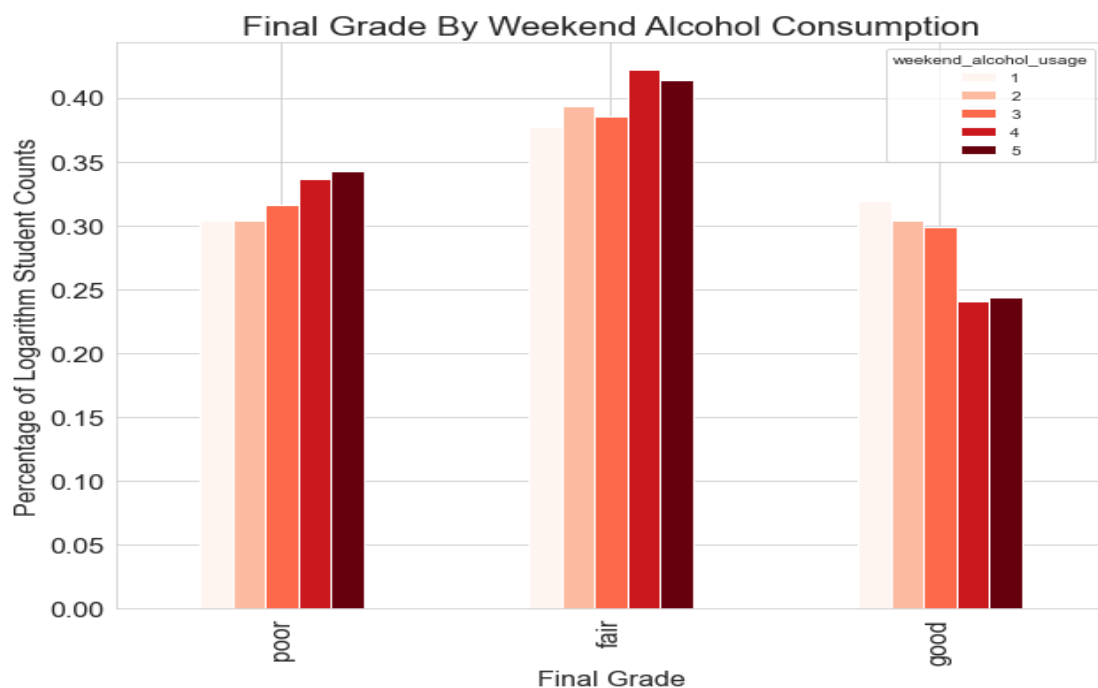


Fig.10. Final Grade by Weekend Alcohol Consumption

6.2.3 Analysis of final grade on the basis of frequency of going out: Fig. 11 shows the impact of students going out frequency on their final grades.

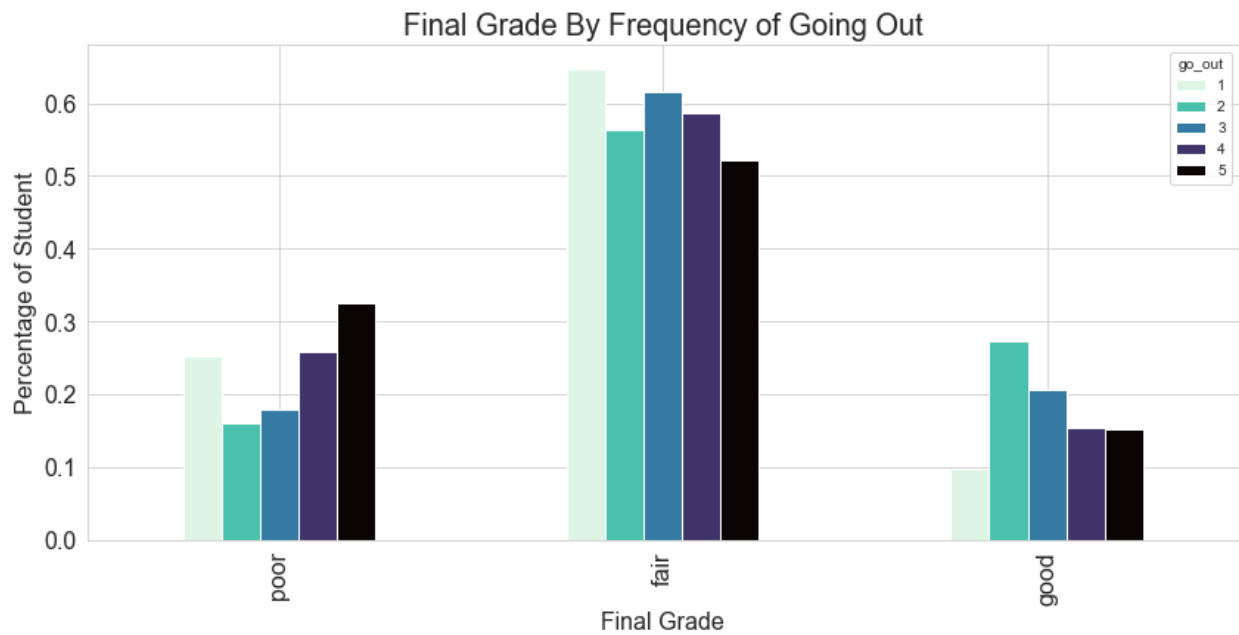


Fig.11. Final Grade by Frequency of Going Out

6.2.4 Analysis of final grade on the basis of the desire of the student to receive higher education: Fig. 12 shows the analysis impact on final grades with respect to the factor “Desire to receive Higher Education”

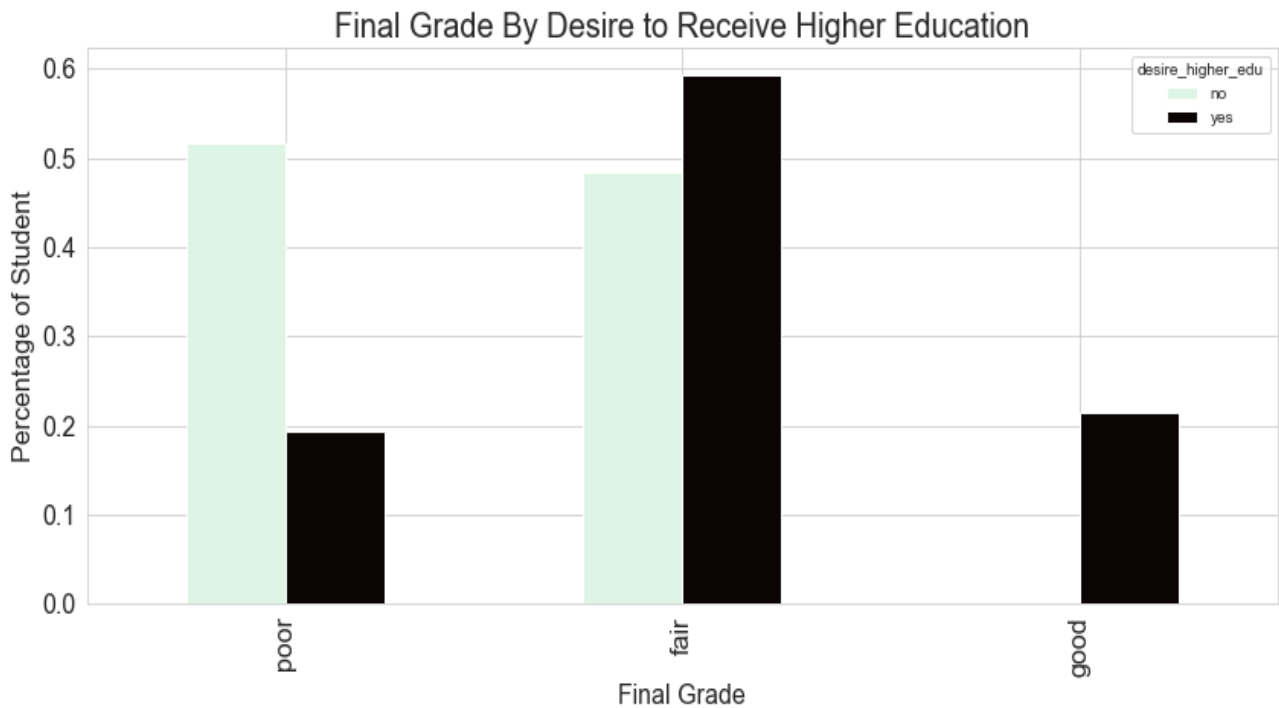


Fig.12. Final Grade by desire to receive higher education

6.2.5 Analysis of final grade on the basis of living area (Urban or Rural): Impact of living area on final grades is shown in Fig. 13

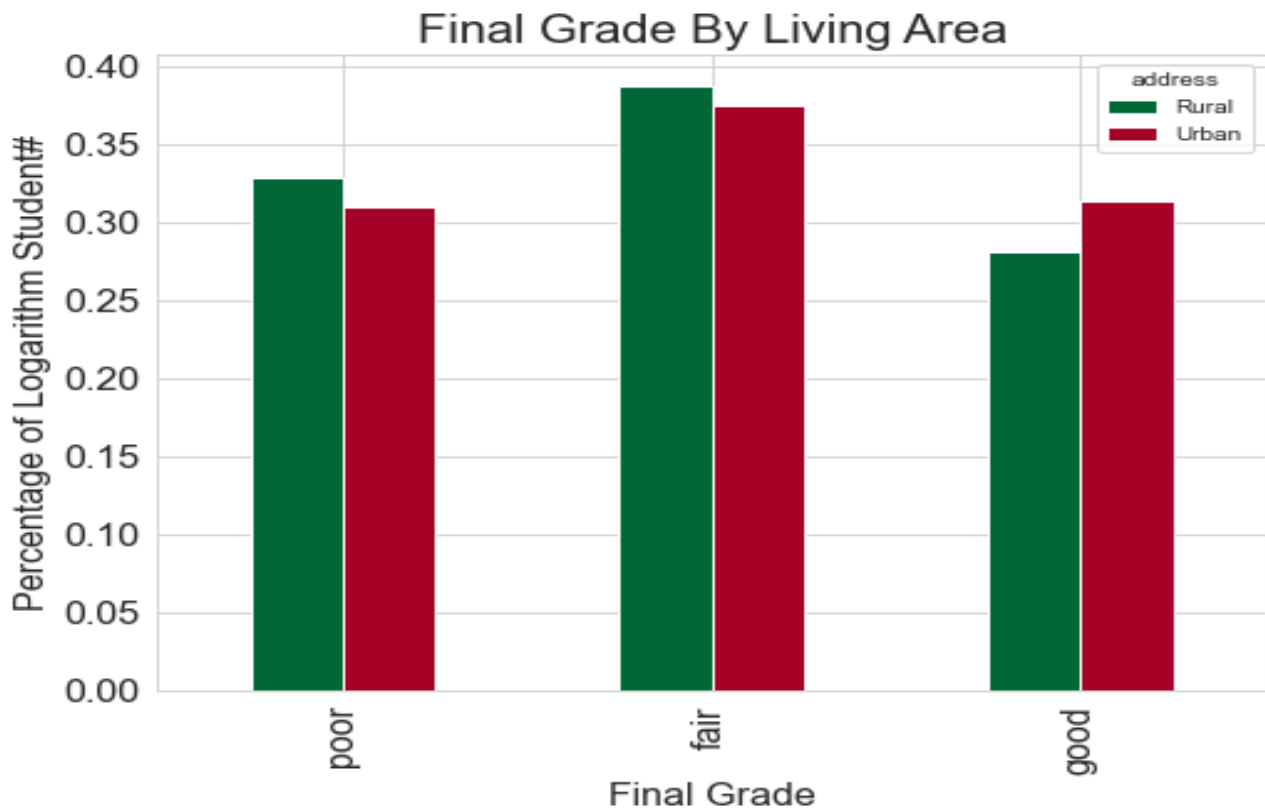


Fig.13. Final Grade by Living Area

7. Conclusion

7.1 The aforementioned Machine Learning Algorithms gave us the following results:

Model score and Cross Validation score of different ML algorithm is shown in Fig. 14. It can be seen from the figure that Support Vector Machine is giving better score value as compared to other models under consideration.

Results And Analysis		
Model Used	Model Score	Cross Validation Score
Logistic Regression	0.887671232877	0.245222929936
Decision Tree	0.902739726027	0.875796178344
Support Vector Classification	0.945205479452	0.891318471338
Random Forest Classification	0.976712328767	0.847133757962
Ada Boost	0.750684931507	0.710191082803
Stochastic Gradient Descent	0.620547945205	0.624203821656

Fig.14. Model Selection

7.2 The reasons why we feel that SVM gave us better results on our dataset as compared to other Machine Learning models that we tried on the same data are:

- SVM models perform better on sparse data than denser data in general. For example, in document classification you may have thousands, even tens of thousands of features and in any given document vector only a small fraction of these features may have a value greater than zero.

Since our data has many binary variables which were encoded into 0s and 1s, our **dataset was transformed became relatively sparsed.**

- Classical machine learning models like SVM are usually better options compared to deep learning models when the amount of data is very less.

This is because **normally deep learning models have a lot of weights (free variables) that need to be tuned with data.** If the number of weights is more than (or around the same as) the number of training examples, the deep models just 'memorize' the data, leading to overfitting.

- Linear **SVM** kernel is used when we have a large number of features, because it is more likely that the data is linearly separable in high dimensional space.

7.3 Future Scope

7.2.1 We will try to improve on these results in the following ways:

1. Increasing the Training Data, so that the model can improve.
2. Use more algorithms to try to improve the accuracy of our model and build a better model which can best predict the placement chances of a student.

7.2.1 In the future, this model can be made into a live web application by creating an API, which can analyze and provide real-time feedback to a student, by taking his information and sending it through the machine learning model and give the students' placement chances.

8.REFERENCES

- [1] "Data Mining Approach for Predicting Student and Institution's Placement Percentage", Professor. Ashok M Assistant Professor Apoorva A ,2016 International Conference on Computational Systems and Information Systems for Sustainable Solutions.
- [2] "Student Placement Analyzer: A Recommendation System Using Machine Learning", Senthil Kumar Thangavel, Divya Bharathi P, Abijith Sankar, International Conference on Advanced Computing and Communication Systems (ICACCS -2017), Jan. 06 - 07, 2017, Coimbatore, INDIA.
- [3] "A Placement Prediction System Using K-Nearest Neighbors Classifier", Animesh Giri, M Vignesh V Bhagavath, Bysani Pruthvi, Naini Dubey, Second International Conference on Cognitive Computing and Information Processing (CCIP), 20164.
- [4] "Class Result Prediction using Machine Learning", Pushpa S K, Associate Professor, Manjunath T N, Professor and Head, Mrunal T V, Amartya Singh, C Suhas, International Conference on Smart Technology for Smart Nation, 2017.
- [5] "Student Placement Analyzer: A Recommendation System Using Machine Learning", Apoorva Rao R, Deeksha K C, Vrushak K, Nandini, JARIIE-ISSN(O)-2395-4396.
- [6] Polikar, R. (2012). Ensemble learning. In Ensemble machine learning (pp. 1-34). Springer, Boston, MA.
https://link.springer.com/chapter/10.1007/978-1-4419-9326-7_1
- [7] Gardner, W. A. (1984). Learning characteristics of stochastic-gradient-descent algorithms: A general study, analysis, and critique. Signal processing, 6(2), 113-133.
https://link.springer.com/chapter/10.1007/978-3-642-35289-8_25
- [8] Schapire, R. E. (2013). Explaining adaboost. In Empirical inference (pp. 37-52). Springer, Berlin, Heidelberg.
https://link.springer.com/chapter/10.1007/978-3-642-41136-6_5