I.J. Modern Education and Computer Science, **2023, 6, 1-13**
Published Online on December 8, 2023 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2023.06.01

Predicting College Students' Placements Based on Academic Performance Using Machine Learning Approaches

Mukesh Kumar*
School of Computer Application, Lovely Professional University, Phagwara, Punjab, India
Email: mukesh.27406@lpu.co.in
ORCID iD: https://orcid.org/0000-0001-8797-9810
*Corresponding Author

Nidhi Walia
Maharaja Agrasen University, Solan, Himachal Pradesh, India
Email: nidi1990@gmail.com
ORCID iD: https://orcid.org/0000-0001-8797-9811

Sushil Bansal
Maharaja Agrasen University, Solan, Himachal Pradesh, India
Email: sk93recj@gmail.com
ORCID iD: https://orcid.org/0000-0002-3369-5561

Girish Kumar
School of Computer Application, Lovely Professional University-Phagwara, Punjab, 144001, India
E-mail: girish.21706@lpu.co.in
ORCID iD: https://orcid.org/0000-0002-8363-9808

Korhan Cengiz
Department of Information Technologies, Faculty of Informatics and Management, University of Hradec Kralove, Kralove, 50003, Czech Republic
E-mail: korhan.cengiz@uhk.cz
ORCID iD: https://orcid.org/0000-0001-6594-8861

Received: 17 April, 2023; Revised: 15 May, 2023; Accepted: 20 June, 2023; Published: 08 December, 2023

Abstract: Predicting College placements based on academic performance is critical to supporting educational institutions and students in making informed decisions about future career paths. The present research investigates the use of Machine Learning (ML) algorithms to predict college students' placements using academic performance data. The study makes use of a dataset that includes a variety of academic markers, such as grades, test scores, and extracurricular activities, obtained from a varied sample of college students. To create predictive models, the study analyses numerous ML algorithms, including Logistic Regression, Gaussian Naive Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbour. The predictive models are evaluated using performance criteria such as accuracy, precision, recall, and F1-score. The most effective machine learning method for forecasting students' placements based on academic achievement is identified through a comparative study. The findings show that Random Forest approaches have the potential to effectively forecast college student placements. The findings show that academic factors such as grades and test scores have a considerable impact on prediction accuracy. The findings of this study could be beneficial to educational institutions, students, and career counsellors.

Index Terms: Machine Learning Techniques, Logistic Regression, Gaussian Naïve Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbor.

1. Introduction

Data Mining (DM) is a part of the field of knowledge discovery in databases. It is the process of getting useful information from large amounts of data. These techniques are used on huge datasets to find patterns and relationships that were not known before. This helps people make better decisions by giving them more information. The practise of looking for patterns and tendencies within educational databases is referred to as "Educational Data Mining." The modelling of human activities in order to make predictions about the memory process is included in the related study [1]. When tools for data mining are applied to the setting of a school, those responsible for making decisions and those who instruct students can both benefit from the insights gathered.

The process of job placement for college students can be a daunting and challenging task, with the uncertainty of finding the right job that matches their skills and qualifications. However, with the advent of ML techniques, it is now possible to predict the job placement of college students based on their academic records and other relevant factors. This topic has gained significant interest in recent years, as it offers a practical solution for both students and employers [2]. By analysing the data from previous job placements, machine learning algorithms can identify the patterns and trends that lead to successful job placements. This approach enables colleges and universities to offer more targeted career guidance to their students, helping them to make informed decisions about their future careers. Moreover, this predictive approach can also benefit employers, as they can use these predictions to identify potential candidates who are more likely to succeed in their organisation. This saves them time and resources, as they can focus their efforts on candidates who are more likely to fit their requirements. The process of job placement for college students is one of the most crucial steps towards a successful career [3]. However, the placement process can be a daunting task for students, as it involves various factors such as academic performance, skills, personality traits, and other subjective factors. In recent years, ML techniques have gained significant attention in the field of human resource management, especially in predicting job placement outcomes. Predictive analytics models based on ML techniques can help organisations identify the most suitable candidates for a particular job role. These models leverage historical data to identify patterns and predict future outcomes [4]. Similarly, ML techniques can be applied to predict the job placement outcomes of college students.

In this context, the present study aims to develop a predictive model for the job placement outcomes of college students using ML techniques. The study will leverage various factors, such as academic performance, skills, and other subjective factors, to develop a comprehensive predictive model. The study will also explore the potential of various ML algorithms, such as Decision Trees, Random Forests, and Support Vector Machines, in predicting job placement outcomes [5]. The findings of the study will have significant implications for both students and employers. The predictive model can help students identify the areas in which they need to improve to increase their chances of getting a job. On the other hand, employers can use the predictive model to identify the most suitable candidates for a particular job role, thereby improving their recruitment process's efficiency. Overall, the present study aims to bridge the gap between academia and industry by leveraging ML techniques to predict job placement outcomes for college students [6].

Motivation: The primary concern among graduate students is choosing their career path, largely due to insufficient knowledge about their desired field. To assist these students, a system will be provided to offer comprehensive information about various areas, including professional education options such as courses, degrees, institutions, and job prospects. This system is intended to guide students towards informed decision-making.

Objectives: Every year, the Training And Placement Officer (TPO) is faced with the difficult task of finding suitable companies for final year students in their respective fields. The TPO must ensure that the companies that recruit students offer good job profiles, working environments, salary packages, and opportunities for growth. It becomes increasingly challenging to place the maximum number of students when there are no reliable tools available to the TPO to assess student performance during the placement session. The administration's efforts to boost student performance are often ineffective since they are implemented broadly without any specific strategy. This inability to identify the groups of students that require support leads to poor results despite maximum effort. Consequently, a plan was developed to harness the power of DM and ML by building predictive models using past academic and placement data. This study examined the use of ML techniques to predict college students' placements based on their academic performance. The application of various ML algorithms and techniques yielded optimistic results and highlighted the potential of these approaches for enhancing the decision-making process in educational institutions, as demonstrated by the research findings.

The topic, "Predicting College Students' Placements Based on Academic Performance using Machine Learning Approaches" has gained significant interest in academia and industry. The objective of this research is to develop a model that can predict the placement of a college student based on their academic records, skills, and other relevant factors. The model can help students prepare for their careers by identifying the areas they need to improve in and giving them an idea of what to expect in the job market. This paper will provide an overview of the current state of research on predicting college student placement using ML techniques. It will also discuss the challenges and opportunities associated with this research area. Finally, the paper will conclude by discussing the potential applications of this research in the context of higher education and industry.

2. Literature Review

The process of job placement for college students has been a critical area of research in recent years. Several studies have investigated the factors that influence job placement outcomes, such as academic performance, skills, personality traits, and other subjective factors. However, the traditional methods of predicting job placement outcomes based on these factors have limitations, as they often rely on subjective judgements. In recent years, machine learning techniques have gained significant attention in the field of human resource management. Machine learning techniques are capable of processing vast amounts of data and identifying patterns that may not be visible to the human eye. Several studies have explored the potential of machine learning techniques in predicting job placement outcomes for college students.

One of the earliest studies in this area was conducted by Rathore, R. K. et al. [7] who used Neural Network models to predict the job placement outcomes of university graduates in Taiwan. The study used various factors such as academic performance, extracurricular activities, and work experience to predict job placement outcomes. The results showed that the Neural Network model was effective in predicting job placement outcomes, with an accuracy rate of 88%. Similarly, a study by Patel, T. et al. [8] used decision tree models to predict the job placement outcomes of graduates from a Chinese university. The study used various factors such as academic performance, work experience, and personal traits to predict job placement outcomes. The results showed that the Decision Tree model was effective in predicting job placement outcomes, with an accuracy rate of 89%. More recently, a study by Goyal, J. et al. [9] used a hybrid model combining Logistic Regression and Decision Tree algorithms to predict job placement outcomes for graduates from an Indian university. The study used various factors such as academic performance, work experience, and soft skills to predict job placement outcomes. The results showed that the hybrid model was effective in predicting job placement outcomes, with an accuracy rate of 92%.

Surya, M. S. et al. [10] developed a predictive model for job placement outcomes of college students using decision trees and random forests. The study used various factors such as academic performance, skills, and other subjective factors to develop a comprehensive predictive model. The study found that the random forest algorithm outperformed the decision tree algorithm in predicting job placement outcomes. Similarly, Nagamani, S. et al. [11] developed a job placement prediction model for engineering graduates using a support vector machine algorithm. The study used factors such as academic performance, skills, and personality traits to predict job placement outcomes. The study found that the Support Vector Machine algorithm achieved a prediction accuracy of 88%, which was higher than other algorithms such as decision trees and random forests.

In a similar study, Thakar, P. et al. [12] used ML algorithms such as random forests, decision trees, and Support Vector Machine to predict job placement outcomes for MBA graduates. The study used factors such as academic performance, work experience, and personality traits to develop a comprehensive predictive model. The study found that the random forest algorithm outperformed the other two algorithms in predicting job placement outcomes. A study by Casuat, C. D. et al. [13] developed a predictive model for job placement outcomes of college students using a decision tree algorithm. The study used various factors, such as academic performance, extracurricular activities, and work experience, to develop the predictive model. The study found that the decision tree algorithm had a high accuracy rate in predicting job placement outcomes.

Bai, A. et al. [14] used a random forest algorithm to predict job placement outcomes for college students. The study used various factors, such as academic performance, personality traits, and social network analysis, to develop the predictive model. The study found that the random forest algorithm had a higher accuracy rate than other machine learning algorithms in predicting job placement outcomes. A study by Saidani, O. et al. [15] used a support vector machine algorithm to predict job placement outcomes for college students. The study used various factors, such as academic performance, skills, and personality traits, to develop the predictive model. The study found that the support vector machine algorithm had a high accuracy rate in predicting job placement outcomes.

Hariharan, V. J. et al. [16] developed a predictive model for job placement outcomes of college students using a deep learning algorithm. The study used various factors, such as academic performance, skills, and social network analysis, to develop the predictive model. The study found that the deep learning algorithm had a higher accuracy rate than other machine learning algorithms in predicting job placement outcomes.

The literature suggests that machine learning techniques can be effective in predicting job placement outcomes for college students. However, the effectiveness of the models depends on the factors used to predict job placement outcomes and the algorithms used to develop the predictive models. The present study aims to contribute to the existing literature by developing a comprehensive predictive model for job placement outcomes using machine learning techniques. The job placement process for college students has been a challenging task for both students and employers. The traditional recruitment process is time-consuming and involves a considerable number of resources. Moreover, the recruitment process's success largely depends on the recruiter's subjective assessment, which may not always be accurate. The use of machine learning techniques in predicting job placement outcomes for college students has gained significant attention in recent years. Several studies have explored the potential of various machine learning algorithms in predicting job placement outcomes [17]. This section provides a comprehensive review of the existing literature on this topic. Overall, the existing literature suggests that machine learning techniques have a significant potential in predicting job placement outcomes for college students. The accuracy rates of various machine learning algorithms vary, depending on the factors used in developing the predictive model. However, the studies suggest that the use of multiple factors can improve the accuracy rate of the predictive model. The findings of these studies have significant implications for both students and employers in improving the recruitment process's efficiency.

3. Material and Methods

Dataset Description: Since the raw data could have lots of missing values or other issues, we will be taking care to align the data into simpler more understandable and clear data which will be further helpful to be used in our ML models. Since the columns have not been clearly mentioned as to what they exactly mean, I have made the following assumptions to understand the data better. The dataset has a total of 215 entries [18]. The detail description of different attributes of the dataset is shown in Table 1.

Table 1. Different attributes of the dataset along with its domain mapping

S.N.	Attributes	Description of Attributes	Attribute Domain Mapping
1	gender	Gender of the student	M-0, F-1
2	ssc_p	10th Grade %age	0-60% : 3, 61-80% : 2, 81-100% :1
3	ssc_b	10th Grade Board	Central-116 : 1, Others-99 : 0
4	hsc_p	12th Grade %age	0-60% : 3, 61-80% : 2, 81-100% :1
5	hsc_b	12th Grade Board	Central-116 : 1, Others-99 : 0
6	hsc_s	Higher Secondary Stream	Commerce-113, Science-91, Arts-11 : one hot encodes
7	degree_p	Undergraduate %age	0-60% : 3, 61-80% : 2, 81-100% :1
8	degree_t	Undergraduate Degree Type	Comm&Mgmt-145, Sci&Tech-59, Others-11 : one hot encodes
9	workex	Work Experience	Yes : 1, No : 0
10	etest_p	Placement Test %age	0-60% : 3, 61-80% : 2, 81-100% : 1
11	specialisation	MBA Specialisation	Mkt&Fin-120 : 1, Mkt&HR-95 : 0
12	mba_p	MBA %age	0-60% : 3, 61-80% : 2, 81-100% : 1
13	salary	Salary Offered	Variable value
14	status	Hiring Status	Placed : 1, Not Placed : 0

This is roughly the size of an MBA batch for a particular year of a college. Hence, the data provided could be that for the batch of a particular year. Let us check the various data types available to us and see if there are any missing values. As we can clearly see, the only missing values present are in the salary column. This is because the missing values are corresponding to students to did not get placed in the placement programme. So, we have replaced all the missing values with the median value of Rs. 2,65,000. Gender column is straightforward with either Male (M) or Female (F) indices. Since we cannot leave the data in object form, let us replace M with 0 and F with 1. As we can observe, the unique entries are either Central or others. By Central, the indication must be that the board is CBSE. While others could mean variety of boards such as state board, international board or ICSC board. The candidates with central board are slightly higher as compared to other boards. We shall perform the exact same treatment as we did for gender (Central : 1 Others : 0)

Data Visualisation: In this section, we shall be visualising the various features to understand if they will have some correlation with the target feature. As it can be seen, a high percentage of male candidates got placed. On the other hand, a comparatively lower number of female candidates got placed. This is expected, as male candidates are generally higher in most branches. Hence, their chances of getting placed are normally higher as well. Let us check the male-to-female ratio of this branch. The male-to-female ratio is approximately 2. This can be interpreted as meaning that for every 1 female candidate, there are 2 male candidates sitting for placements, as shown in Figure 1.

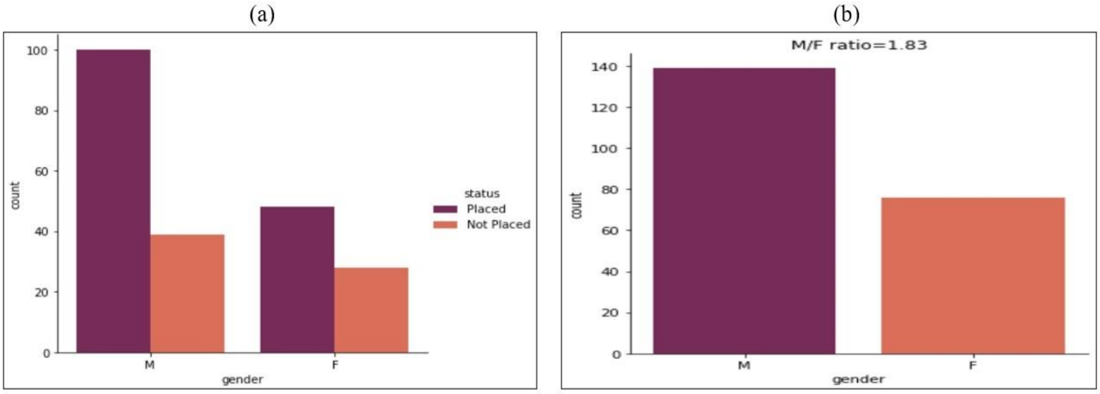


Fig. 1. (a) Graphical representation of student's gender counts w.r.t Placed and Not placed student status (b) Graphical representation of student's gender ratio w.r.t Placed and Not placed student status

As per Figure 2, studying the MBA percentage data, we can see that no students crossed the 80% threshold. Hence, the data for class 1 is not available for MBA percentage. Upon studying the MBA percentage data, we can see that more students from percentage class 2 got placed as opposed to class 3. However, the difference is not as high as for board and UG percentages. Hence, we can say that the MBA percentage is not as big a factor. This could be because MBA is a branch that places much more importance on speaking skills, internships, case studies, etc. than academic scores. Hence, the play is much more level fielded in this case. The graphs tell us about how the students have performed in the E-tests and what value those have added to their offers. Most students in the class 1 category have successfully received offers from the company. It is also encouraging to see that more students in Class 3 categories have also successfully converted offers. The reason could be that E-tests are just preliminary screening tests. These tests usually do not hold much value once the student passes them and moves ahead with a further screening process that could include group discussions and interviews.

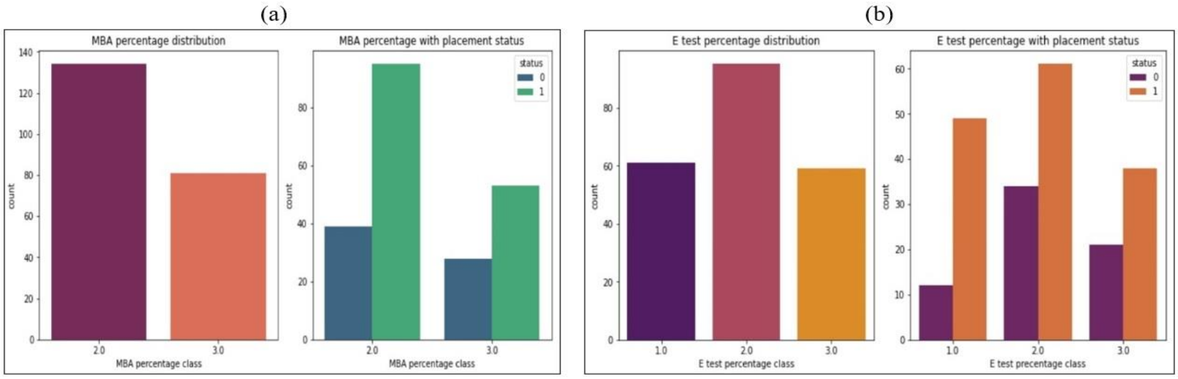


Fig. 2. (a) MBA percentage distribution and percentage w.r.t Placed and Not Placed students’ status (b) E test percentage distribution and E test percentage w.r.t Placed and Not Placed students’ status

We are now interested in checking if the high school stream, UG degree specialisation, or MBA degree specialisation has any particular importance with respect to placement status. As we can see, many students enrolled in the programme were either from commerce or science backgrounds. Very few students were from an arts background. In terms of placements, most students who got placed were from commerce backgrounds, followed by science and then the arts. This could be because both commerce and the arts make great use of mathematical and analytical skills, which are quite important in finance operations. Since commerce students have economics and finance as a part of their school and UG studies, they are in high demand by companies. Arts students will be a bigger asset for the human resources departments of companies, as their skillsets are ideal for HR-related work. Hence, students getting placed with an arts background are demanded mostly for HR-related roles. From the MBA specialisation data, most students who were placed were in the marketing and finance division. In the Marketing and HR section, students could not convert as many placement opportunities. The number of placed and unplaced students is about the same in this case, as shown in Figure 3.

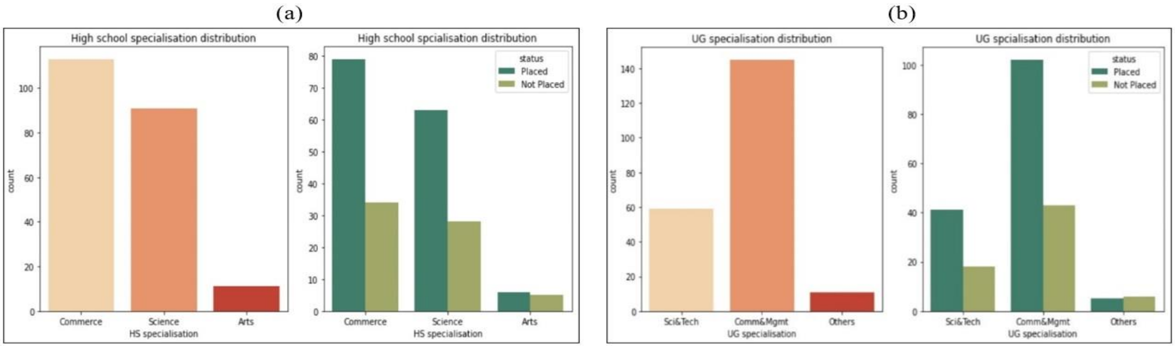


Fig. 3. (a) HS specialisation comparison graph with total count of types of specialisations and placement distribution with respect to that specialisation (b) UG specialisation comparison graph with total count of types of specialisations and placement distribution with respect to that specialisation

Let us visualise the data in Figure 4 for the boards of candidates who managed to get placed. This will give us an indication if the board of examination has any role to play in placements. The placement ratio for each of the boards is similar. Hence, we could say that the 10th board of examination does not hold much value for placement. Unlike the case for the 10th board, more students opted for state boards in their 12th examinations. The performance of state board students was better as more students from state board 12th grade were placed. However, the placed/unplaced ratio for both is nearly identical once again. Hence, the 12th board is not playing a significant role once again.

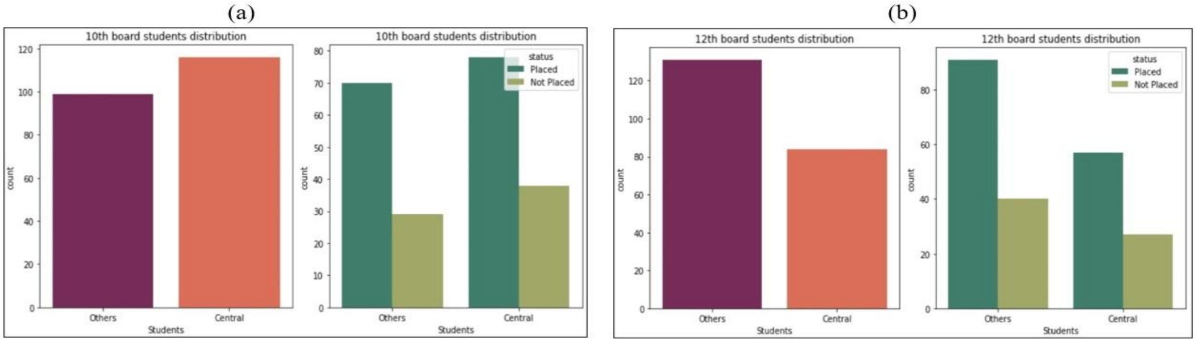


Fig. 4. (a) 10th board student count distribution w.r.t central and others board and 10th board student count distribution w.r.t Placed and Not Placed students (b) 12th board student count distribution w.r.t central and others board and 12th board student count distribution w.r.t Placed and Not Placed students

Let us check how the companies are paying their freshmen, as shown in Figure 5. We shall find the required median and mean salaries for the college placements. As we can see, the salary curve is right-skewed. This is because there will always be some dream jobs that offer high packages. However, these packages are very few. Most of the packages will be in the region of 2-4 LPA. Let us mark the mean and median salaries on the distribution curve to understand what the maximum number of students are earning. The green line shows the median salary of the candidates, while the red line shows the mean salary.

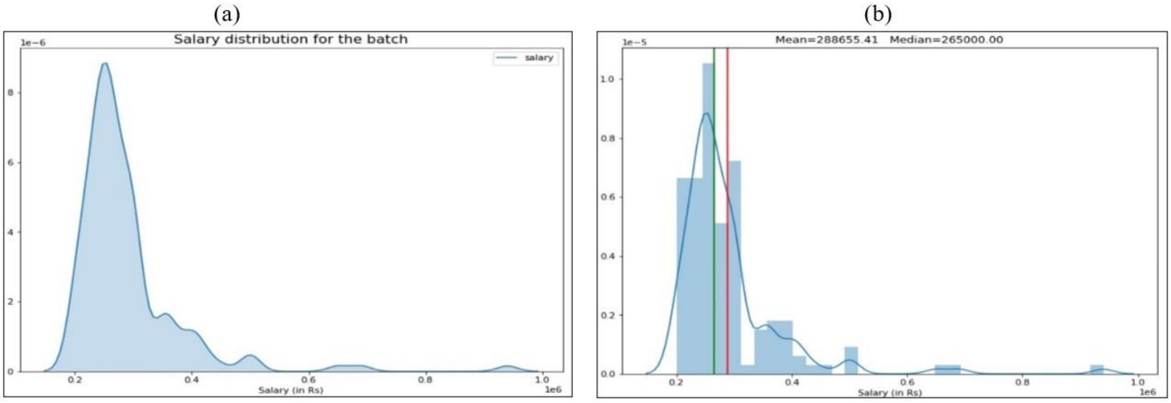
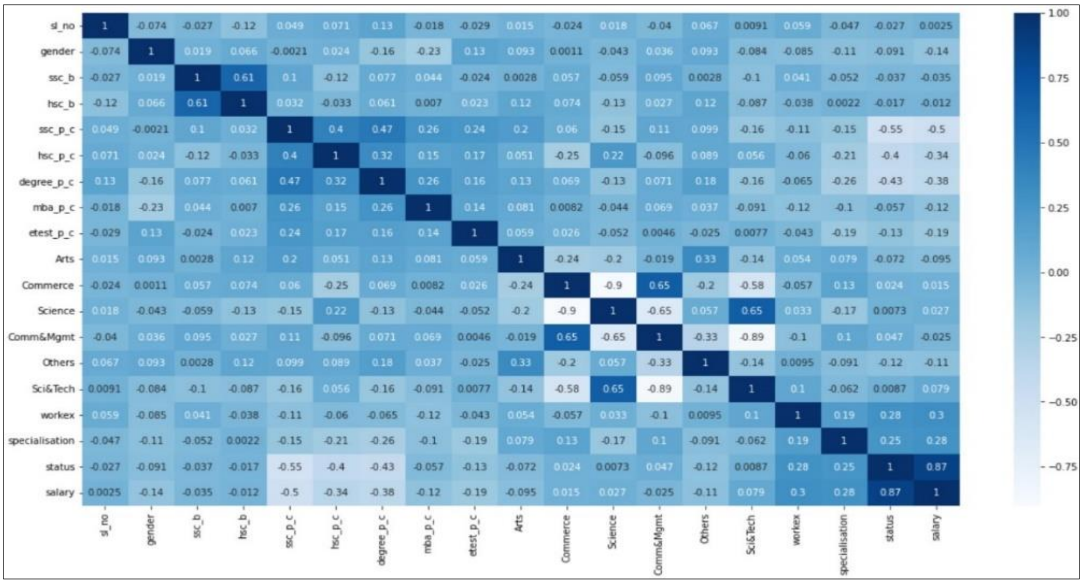


Fig. 5. (a) Salary distribution for the batch, (b) mean and median of the salary attributes for the student placement dataset



	sl_no	gender	ssc_b	hsc_b	ssc_p_c	hsc_p_c	degree_p_c	mba_p_c	etest_p_c	Arts	Commerce	Science	Comm&Mgmt	Others	Sci&Tech	workex	specialisation	status	salary
sl_no	1	-0.074	-0.027	-0.12	0.049	0.071	0.13	-0.018	-0.029	0.015	-0.024	0.018	-0.04	0.067	0.0091	0.059	-0.047	-0.027	0.0025
gender	-0.074	1	0.019	0.066	-0.0021	0.024	-0.16	-0.23	0.13	0.093	0.0011	-0.043	0.036	0.093	-0.084	-0.085	-0.11	-0.091	-0.14
ssc_b	-0.027	0.019	1	0.61	0.1	-0.12	0.077	0.044	-0.024	0.0028	0.057	-0.059	0.095	0.0028	-0.1	0.041	-0.052	-0.037	-0.035
hsc_b	-0.12	0.066	0.61	1	0.032	-0.033	0.061	0.007	0.023	0.12	0.074	-0.13	0.027	0.12	-0.087	-0.038	0.0022	-0.017	-0.012
ssc_p_c	0.049	-0.0021	0.1	0.032	1	0.4	0.47	0.26	0.24	0.2	0.06	-0.15	0.11	0.099	-0.16	-0.11	-0.15	-0.55	-0.5
hsc_p_c	0.071	0.024	-0.12	-0.033	0.4	1	0.32	0.15	0.17	0.051	-0.25	0.22	-0.096	0.089	0.056	-0.06	-0.21	-0.4	-0.34
degree_p_c	0.13	-0.16	0.077	0.061	0.47	0.32	1	0.26	0.16	0.13	0.069	-0.13	0.071	0.18	-0.16	-0.065	-0.26	-0.43	-0.38
mba_p_c	-0.018	-0.23	0.044	0.007	0.26	0.15	0.26	1	0.14	0.081	0.0082	-0.044	0.069	0.037	-0.091	-0.12	-0.1	-0.057	-0.12
etest_p_c	-0.029	0.13	-0.024	0.023	0.24	0.17	0.16	0.14	1	0.059	0.026	-0.052	0.0046	-0.025	0.0077	-0.043	-0.19	-0.13	-0.19
Arts	0.015	0.093	0.0028	0.12	0.2	0.051	0.13	0.081	0.059	1	-0.24	-0.2	-0.019	0.33	-0.14	0.054	0.079	-0.072	-0.095
Commerce	-0.024	0.0011	0.057	0.074	0.06	-0.25	0.069	0.0082	0.026	-0.24	1	-0.9	0.65	-0.2	-0.58	-0.057	0.13	0.024	0.015
Science	0.018	-0.043	-0.059	-0.13	-0.15	0.22	-0.13	-0.044	-0.052	-0.2	-0.9	1	-0.65	0.057	0.65	0.033	-0.17	0.0073	0.027
Comm&Mgmt	-0.04	0.036	0.095	0.027	0.11	-0.096	0.071	0.069	0.0046	-0.019	0.65	-0.65	1	-0.33	-0.89	-0.1	0.1	0.047	-0.025
Others	0.067	0.093	0.0028	0.12	0.099	0.089	0.18	0.037	-0.025	0.33	-0.2	0.057	-0.33	1	-0.14	0.0095	-0.091	-0.12	-0.11
Sci&Tech	0.0091	-0.084	-0.1	-0.087	-0.16	0.056	-0.16	-0.091	0.0077	-0.14	-0.58	0.65	-0.89	-0.14	1	0.1	-0.062	0.0087	0.079
workex	0.059	-0.085	0.041	-0.038	-0.11	-0.06	-0.065	-0.12	-0.043	0.054	-0.057	0.033	-0.1	0.0095	0.1	1	0.19	0.28	0.3
specialisation	-0.047	-0.11	-0.052	0.0022	-0.15	-0.21	-0.26	-0.1	-0.19	0.079	0.13	-0.17	0.1	-0.091	-0.062	0.19	1	0.25	0.28
status	-0.027	-0.091	-0.037	-0.017	-0.55	-0.4	-0.43	-0.057	-0.13	-0.072	0.024	0.0073	0.047	-0.12	0.0087	0.28	0.25	1	0.87
salary	0.0025	-0.14	-0.035	-0.012	-0.5	-0.34	-0.38	-0.12	-0.19	-0.095	0.015	0.027	-0.025	-0.11	0.079	0.3	0.28	0.87	1

Fig. 6. Heatmap of the correlations plot

Now that we have done all the required visualisations and data wrangling and cleaning operations, we shall focus on applying numerous ML algorithms to see if we can accurately predict the placement status of a student. In order to understand how strongly the features are correlated to each other, let us create a heatmap of the correlation plot that will let us know where the correlations are strong. From the heatmap, we can make strong arguments to remove some of the unimportant features that can be removed for our ML training and testing, as shown in Figure 6.

From the heatmap, it is not entirely certain which of the values are highly correlated. However, there are some bad correlations that we may omit. But the bigger issue is that there is no clear value that can tell us which features are non-important. Hence, we will leave out these features. However, we can remove some features, such as examination boards, since they do not seem to give us any extra information. Let us proceed with dropping the unnecessary features like sl_no, ssc_b, hsc_b, and salary from the dataset.

4. Machine Learning Algorithms Implementation

Figure 7, give an idea about the flow diagram of all the implementation work of this research. Below elaborate different implemented steps:

Algorithm Selection: After figuring out which characteristics are the most important, the next step is to choose an algorithm for ML approaches. Several different algorithms, including Logistic Regression, Gaussian Naive Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbour frequently utilised when attempting to predict the college students' placements based on academic performance. It will depend on the specifics of the situation and the available data to determine which approach to apply.

Model Training: After that, the ML approaches is trained with the assistance of the pre-processed data. The class variable is used as the output variable, and the algorithm is trained to detect the link that exists in the dataset between the variables that are used as input and the class variable.

Model Tuning: Altering the model in a variety of ways can lead to improvements in its performance. One method for achieving this objective is to adjust the hyperparameters used by the different ML approaches.

Model Evaluation: When the training phase of the model is complete, it must then be validated to determine how accurately it can predict where students will be placed based on their academic achievement. A portion of this stage consists of evaluating the model's performance on a data set that is distinct from the one it was trained on.

Prediction: The trained model is used to make predictions about college students' placements based on academic achievement for the next academic year. These predictions are based on the input variables that determine the prediction's accuracy.

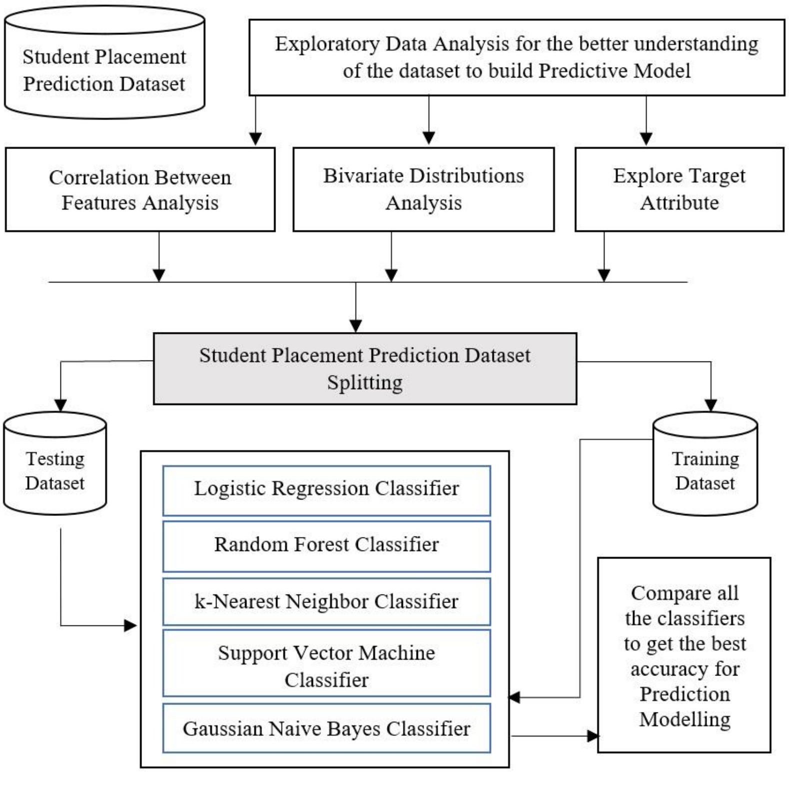

Student Placement Prediction Dataset
Exploratory Data Analysis for the better understanding of the dataset to build Predictive Model
Correlation Between Features Analysis
Bivariate Distributions Analysis
Explore Target Attribute
Student Placement Prediction Dataset Splitting
Testing Dataset
Training Dataset
Logistic Regression Classifier
Random Forest Classifier
k-Nearest Neighbor Classifier
Support Vector Machine Classifier
Gaussian Naive Bayes Classifier
Compare all the classifiers to get the best accuracy for Prediction Modelling

Fig. 7. Model diagram for Data Processing, Model Training and Testing, Prediction and Accuracy check.

4.1 Classification Algorithms: Classification algorithms were used to create predictive models and understandable patterns. To provide the most accurate diagnoses, seven widely used Classification algorithms were identified, like Logistic Regression, Gaussian Naïve Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbours. There are two phases, when building predictive models: the training phase, where the model is built using a set of training data that contains the expected outputs, and the validation phase, where the quality of the trained models is estimated using the validation dataset that does not contain the expected outputs.

4.1.1 Logistic Regression: It is one of the ML approaches that is used the most and does not fall under the purview of the supervised learning method. It is feasible to make a prediction about a categorical dependent variable by making use of a predetermined collection of independent factors. Logical Regression allows for the prediction of the value of a categorical dependent variable. A statistical model that determines the relationship between variables and delivers an answer that is either yes or no [19]. It first determines the difference between the result with no predictors and the baseline outcome, and then compares the outcome to the baseline outcome.

4.1.2 Gaussian Naïve Bayes: Naïve Bayes can be applied to features with real-world values, with the most popular implementation assuming the features follow a Gaussian distribution. In this context, we use the term "Gaussian Naïve Bayes" to describe this modification of the original Naïve Bayes method. The Gaussian distribution is the simplest method for determining the shape of a data distribution because it simply requires you to determine the mean and standard deviation of your training set. The distribution of the data can be estimated using several different supplementary functions [20]. The frequency determined the likelihood of each group of input values. The mean and standard deviation of the values in each group provide useful summary statistics if the input values (x) are real numbers. Here is the formula for figuring out the average and standard deviation. That is why it is important to monitor the means and standard deviations of all input variables, not just the probabilities attached to each category. All that must be done to complete (equation 1) is to determine the mean and standard deviation of each input variable (x) for each class value.

$$\text{mean(x)} = \frac{1}{\text{n}} \times \text{sum(x)} \tag{1}$$

For any given input variable in your training data, the range of potential values is denoted by x, and the total number of occurrences is denoted by n. The data standard deviation can be calculated using (equation 2) as follows:

$$\text{standard deviation(x)} = \sqrt{1/\text{n} \times \text{sum}(x_i - \text{mean(x)\^{}2}} \tag{2}$$

where n is the number of instances, sum() is the sum function, x_i is a specific value of the x variable for the i^{th} instance.

4.1.3 Random Forest: During the training process, Random Forest will construct many individual decision trees. The mean prediction is the sum of the predictions from each tree. Decision-making using an ensemble approach involves basing choices on several previously gathered results. The reduction in node impurity is balanced against the probability of accessing that node in order to compute feature significance [21]. The probability of a node is calculated by taking the total number of samples and dividing that by the number of samples that reach the node. The more significant the feature, the higher the value as shown in (equation 3).

We essentially need to know the dataset's impurity, and we will use that feature as the root node, or the feature with the lowest impurity, or the feature with the lowest Gini Index (GI). It can be expressed mathematically as:

$$\text{GI} = 1 - \sum_{i=1}^{n} (P_i)^2 = 1 - [(P_+)^2 + (P_-)^2] \tag{3}$$

Where (P_+) is the probability of a positive class and (P_) is the probability of a negative class. The formula for calculating the Feature Importance (FI) is shown in (equation 4).

$$(\text{FI})_\text{i} = \frac{\sum_{j:node\ j\ splits\ on\ feature\ i} n_{i_j}}{\sum_{k\ \varepsilon\ all\ nodes} n_{i_k}} \tag{4}$$

To calculate the importance of a node, we will use this formula for n_i is

$$n_i = \frac{N_t}{N} [impurity - \left(\frac{N_{r(right)}}{N_t} * right\ impurity \right) - \left(\frac{N_{l(left)}}{N_t} * left\ impurity \right)] \tag{5}$$

Where N_t is number of rows that node has, N is the total number of rows present in the data, impurity is our GI value, N_r (right) is the number of nodes in the right side and N_l (left) is the number of nodes in the left side as mentioned in (equation 5).

4.1.4 Support Vector Machine: In the realm of supervised ML, the Support Vector Machine (SVM) is useful for both classification and regression. Despite our exploration of regression difficulties, this method is more naturally applicable to classification tasks than it is to regression ones. The SVM method is useful for classifying data points in an N-dimensional space. The support vector machine technique finds a hyperplane in N-dimensional space along which data points can be separated into discrete classes. To differentiate between the two classes of data points, several available hyperplanes can be selected [22]. We have made it our mission to identify the aircraft with the largest margin, also known as the largest gap between data points in the two categories. As the margin distance grows, more reinforcement is given so that the following set of data may be classified with more ease. If we want to use the SVM method, we need to maximise the separation between the hyperplane and the data points. One approach to accomplishing this objective is by using a loss function called the hinge loss.

4.1.5 K-Nearest Neighbors: The conventional methods of machine learning have been improved in order to make them compatible with massive data sets. One example of such a collection is one that has been gathered through data mining. When it comes down to it, identifying each data point demands a significant amount of training data to be run through the system. In theory, points are represented in a space that has many dimensions, and in this space, each axis of the space stands for a different independent variable. The term "hyperspace" is used to describe this area when referring to it. In order to locate the missing data points, we look first for the K data points that have the most in common with the new one. When assessing how far apart two points are, most people agree that the Euclidean distance is the one that should be used as the benchmark. As demonstrated in (equation 6), the Euclidean distance can also be referred to simply as "distance." In this equation, x_i and y_i are the two points that are closest to each other.

$$\mathrm{d(x, y) = d(x, y)} = \sqrt{(x_1 - y)^2 + (x_1 - y_1)^2 + \cdots + (x_n - y)^2)} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{6}$$

When working with a large amount of continuous or dense data, it is strongly advised that you use the Euclidean distance metric. The Euclidean distance is the most reliable indicator of how close two points are. The Euclidean distance between two places is equal to the length of the section of line that runs between them.

5. Results and Discussion

In this study, we aimed to predict college students’ placement using various ML techniques. The dataset comprises academic records, aptitude test scores, and other relevant parameters for students’ academic performance . The primary objective of the study is to develop an accurate prediction model that can help identify students who have a high probability of getting placed in a company. The results of the study will provide insights into the effectiveness of various machine learning techniques for predicting student placement. Additionally, it can be useful to educational institutions, students, and recruiters by streamlining the placement process. The discussion of the study results will cover the strengths and limitations of the prediction models used and their implications for future research.

Table 2. Train and Test accuracy of different machine learning algorithms on student placement dataset

Classifier	Train Accuracy	Test Accuracy
Logistic Regression	87.33 %	76.92 %
Gaussian Naive Bayes	81.33 %	78.46 %
Random Forest	96.00 %	64.62 %
Support Vector Machine	85.33 %	75.38 %
K-Nearest Neighbor	87.33 %	73.85 %

Table 2 displays the train accuracy and test accuracy of different classifiers on a dataset. The classifiers included are Logistic Regression (train accuracy: 87.33 %, test accuracy: 76.92 %), Gaussian Naive Bayes (train accuracy: 81.33 %, test accuracy: 78.46 %), Random Forest (train accuracy: 96.00 %, test accuracy: 64.62 %), Support Vector Machine (train accuracy: 85.33 %, test accuracy: 75.38 %), and K-Nearest Neighbor (train accuracy: 87.33 %, test accuracy: 73.85 %). Among these, the highest train accuracy is achieved by Random Forest with a value of 96.00 %, indicating it performs the best on the training set compared to the other classifiers in this table. The lowest train accuracy is observed for Gaussian Naive Bayes with a value of 81.33 %. The graphical representation of Table 2 is shown in Figure 8.

Similarly, the highest test accuracy is achieved by Gaussian Naive Bayes with a value of 0.7846, indicating it performs the best on the test set compared to the other classifiers. The lowest test accuracy is observed for Random Forest with a value of 0.6462, indicating it may be overfitting to the training data.

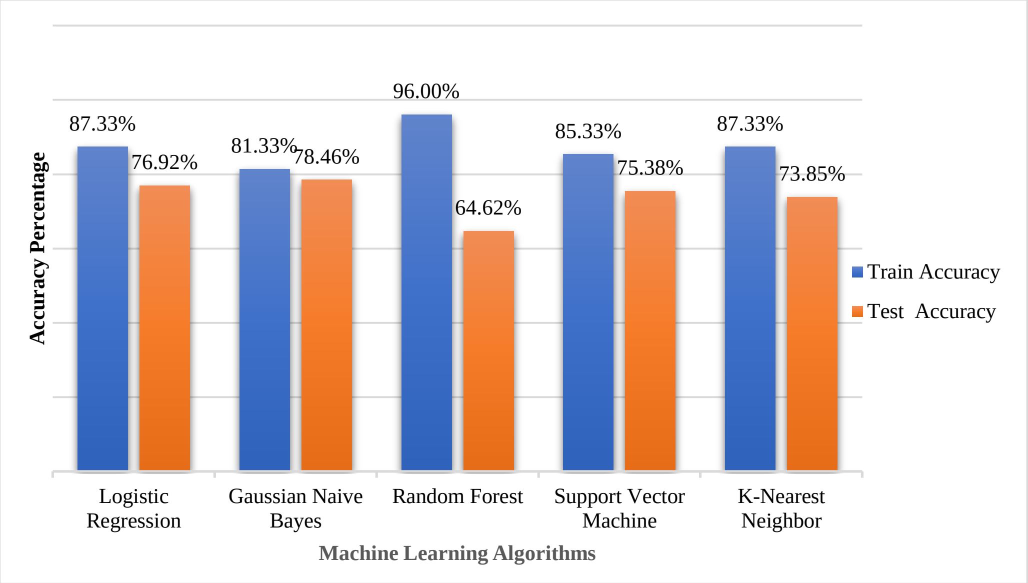


Fig. 8. Graphical representation of Train and Test accuracy of different machine learning algorithms on student placement dataset

Table 3. Different performance metric of different machine learning algorithms

Performance Metric	Logistic Regression		Gaussian Naive Bayes		Random Forest		Support Vector Machine		K-Nearest Neighbor	
	0	1	0	1	0	1	0	1	0	1
precision	0.64	0.80	0.65	0.83	0.39	0.74	0.60	0.80	0.60	0.76
recall	0.47	0.55	0.58	0.87	0.37	0.76	0.47	0.87	0.32	0.91
f1-score	0.89	0.85	0.61	0.85	0.38	0.75	0.53	0.83	0.41	0.83

Table 3 shows the performance metrics (precision, recall, and F1-score) for different classifiers, namely Logistic Regression, Gaussian Naive Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbor. The values in the table correspond to two class labels, 0 and 1, and represent the performance of each classifier in terms of these metrics. The highest precision value is 0.80, which is achieved by both Gaussian Naive Bayes and Support Vector Machine. The lowest precision value is 0.39, which is achieved by Logistic Regression. The highest recall value is 0.91, which is achieved by K-Nearest Neighbor. The lowest recall value is 0.32, which is achieved by Support Vector Machine. The highest F1-score value is 0.89, which is achieved by Logistic Regression. The lowest F1-score value is 0.38, which is achieved by Random Forest.

In Figure 9(a), the confusion matrix graph for a logistic regression model, considering predicted value with respect to placed (5) and not placed(41), and actual value with respect to not placed (5) and placed (9), using a range of values (5, 10, 15, 20, 25, 30, 35, 40), would show the counts or percentages of true positives (predicted placed and actual placed), false positives (predicted placed but actual not placed), true negatives (predicted not placed and actual not placed), and false negatives (predicted not placed but actual placed). This graph helps assess the performance of the logistic regression model in correctly predicting the placed and not placed categories based on the given range of values for actual and predicted labels. In Figure 9(b), the confusion matrix graph for a Gaussian Naive Bayes model, considering predicted value with respect to placed (6) and not placed (40), and actual value with respect to not placed (6) and placed (11), using a range of values (10, 15, 20, 25, 30, 35, 40), would display the counts or percentages of true positives (predicted placed and actual placed), false positives (predicted placed but actual not placed), true negatives (predicted not placed and actual not placed), and false negatives (predicted not placed but actual placed). This graph provides insights into how well the Gaussian Naive Bayes model is predicting the placed and not placed categories based on the given range of values for actual and predicted labels.

In Figure 9(c) the confusion matrix graph for a Support Vector Machine (SVM) model, considering predicted value with respect to placed(6) and not placed(40), and actual value with respect to not placed(6) and placed(9), using a range of values (10, 15, 20, 25, 30, 35, 40), would display the counts or percentages of true positives (predicted placed and actual placed), false positives (predicted placed but actual not placed), true negatives (predicted not placed and actual not placed), and false negatives (predicted not placed but actual placed). This graph provides insights into how well the SVM model is predicting the placed and not placed categories based on the given range of values for actual and predicted labels.

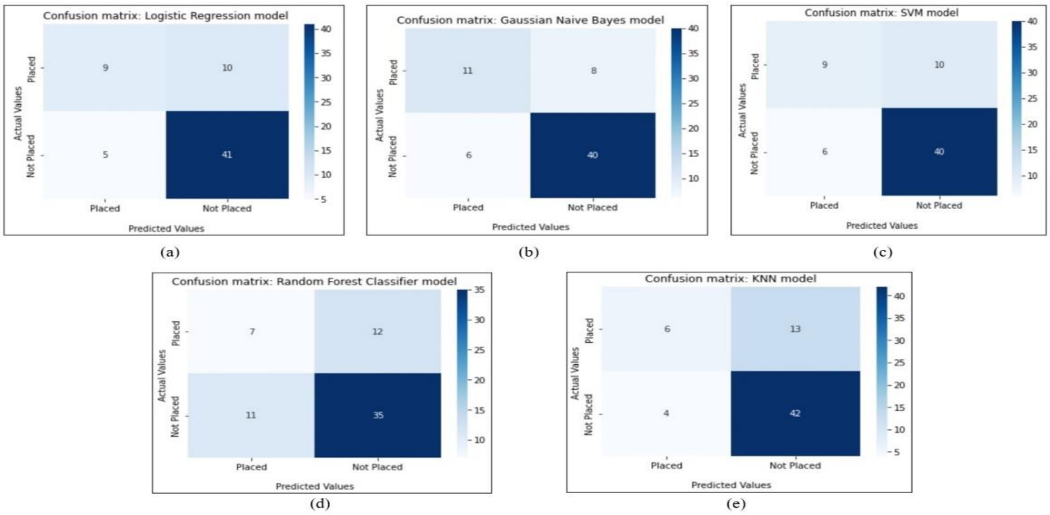


Fig. 9. (a) Confusion matrix by Logistic Regression Model (b) Confusion matrix by Gaussian Na ïve Bayes Model (c) Confusion matrix by Support Vector Machine Model (d) Confusion matrix by Random Forest Model (e) Confusion matrix by K-Nearest Neighbors Model

In Figure 9(d), the confusion matrix graph for a Random Forest Classifier model, considering predicted value with respect to placed(11) and not placed (35), and actual value with respect to not placed (11) and placed (7), using a range of values (10, 15, 20, 25, 30, 35, 40), would display the counts or percentages of true positives (predicted placed and actual placed), false positives (predicted placed but actual not placed), true negatives (predicted not placed and actual not placed), and false negatives (predicted not placed but actual placed). This graph provides insights into how well the Random Forest Classifier model is predicting the placed and not placed categories based on the given range of values for actual and predicted labels. In Figure 9(e), the confusion matrix graph for a K-Nearest Neighbor (KNN) model, considering predicted value with respect to placed(4) and not placed(42), and actual value with respect to not placed (4) and placed (6), using a range of values (5, 10, 15, 20, 25, 30, 35, 40), would display the counts or percentages of true positives (predicted placed and actual placed), false positives (predicted placed but actual not placed), true negatives (predicted not placed and actual not placed), and false negatives (predicted not placed but actual placed). This graph provides insights into how well the KNN model is predicting the placed and not placed categories based on the given range of values for actual and predicted labels.

According to the findings in Table 2 above, the Random Forest algorithm yields a training accuracy of 96.00%, while Gaussian Na ïve Bayes performs well with a test accuracy of 78.46% using the provided dataset. It should be noted that the accuracy of these algorithms can vary depending on the dataset used. Logistic Regression, Gaussian Na ïve Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbour are all effective for binary classification problems, as they yield training accuracies of over 81%.

6. Conclusion

The placement prediction system is designed to forecast the job placement outcomes for MBA students in their final year. Various machine learning algorithms are employed in the Python environment for data analysis and prediction. The accuracy of different algorithms is assessed and presented in the table above. According to the findings, the Random Forest algorithm yields a training accuracy of 96.00%, while Gaussian Na ïve Bayes performs well with a test accuracy of 78.46% using the provided dataset. It should be noted that the accuracy of these algorithms can vary depending on the dataset used. Logistic Regression, Gaussian Na ïve Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbour are all effective for binary classification problems, as they yield training accuracies of over 81%. However, some recruiters may consider other factors, such as CAT scores and the history of backlogs, which were not included in our dataset. Therefore, in rare cases, these results may be subject to change. The model can be further enhanced to keep up with increasing competition and customised to meet the specific criteria of companies. Ultimately, this system can be integrated into the institute's website, enabling students to assess their eligibility for placement preparation. The study relied on a specific dataset from a single institution, which may limit the generalizability of the findings. In addition, the research focused only on academic performance as a predictor, ignoring other significant factors like personal characteristics, career aspirations, and socioeconomic background. Future studies should consider incorporating a wider range of variables to increase the accuracy of the findings. In conclusion, the use of machine learning techniques to predict college placements based on academic performance has immense promise for the educational business. The study's findings emphasise the potential benefits of employing data-driven decision-making processes in higher education. In order to improve the student experience, manage resources more efficiently, and promote overall academic success, educational institutions must integrate new technologies as they evolve.

References

[1] Jeevalatha, T., Ananthi, N., & Kumar, D. S. (2014). Performance analysis of undergraduate student's placement selection using decision tree algorithms. International Journal of Computer Applications, 108(15).

[2] Maurya, L. S., Hussain, M. S., & Singh, S. (2021). Developing classifiers through machine learning algorithms for Student placement prediction based on academic performance. Applied Artificial Intelligence, 35(6), 403-420.

[3] Sheetal, M., & Bakare, S. (2016). Prediction of campus placement using data mining algorithm-fuzzy logic and k nearest neighbor. IJARCCE, 5(6), 309-312.

[4] Ahmed, S., Zade, A., Gore, S., Gaikwad, P., & Kolhal, M. (2018). Performance Based Placement Prediction System. IJARIIE-ISSN (O), 4(3), 2395-4396.

[5] Ishizue, R., Sakamoto, K., Washizaki, H., & Fukazawa, Y. (2018). Student placement and skill ranking predictors for programming classes using class attitude, psychological scales, and code metrics. Research and Practice in Technology Enhanced Learning, 13, 1-20.

[6] Manikandan, K., Sivakumar, S., & Ashokvel, M. (2018). A Classification Model for Predicting Campus Placement performance Class using Data Mining Technique. International Journal of Advance Research in Science and Engineering, 7(6).

[7] Rathore, R. K., & Jayanthi, J. (2017). Student prediction system for placement training using fuzzy inference system. ICTACT Journal on Soft Computing, 7(3), 1443-1446.

[8] Patel, T., & Tamrakar, A. (2017). A data mining technique for campus placement prediction in higher education. Indian J. Sci. Res, 14(2).

[9] Goyal, J., & Sharma, S. (2017). Placement Prediction Decision Support System using Data Mining. International Journal of Engineering and Techniques, 4(2).

[10] Surya, M. S., Kumar, M. S., & Gandhi Mathi, D. (2022). Student Placement Prediction Using Supervised Machine Learning. In 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 1352-1355). IEEE.

[11] Nagamani, S., Reddy, K. M., Bhargavi, U., & Kumar, S. R. (2020). Student placement analysis and prediction for improving the education standards by using supervised machine learning algorithms. J. Crit. Rev, 7(14), 854-864.

[12] Thakar, P., & Mehta, A. (2017). A unified model of clustering and classification to improve students' employability prediction. International Journal of Intelligent Systems and Applications, 9(9), 10.

[13] Casuat, C. D., & Festijo, E. D. (2019). Predicting students' employability using machine learning approach. In 2019 IEEE 6th international conference on engineering technologies and applied sciences (ICETAS) (pp. 1-5). IEEE.

[14] Bai, A., & Hira, S. (2021). An intelligent hybrid deep belief network model for predicting students' employability. Soft Computing, 25(14), 9241-9254.

[15] Saidani, O., Menzli, L. J., Ksibi, A., Alturki, N., & Alluhaidan, A. S. (2022). Predicting student employability through the internship context using gradient boosting models. IEEE Access, 10, 46472-46489.

[16] Hariharan, V. J., Abdullah, S., Rithish, R., Prabakar, V., Suguna, M., Ramakrishnan, M., & Selvakumar, S. (2022). Predicting student's placement prospects using Machine learning Techniques. Available at SSRN 4140544.

[17] Manvitha, P., & Swaroopa, N. (2019). Campus placement prediction using supervised machine learning techniques. International Journal of Applied Engineering Research, 14(9), 2188-2191.

[18] Online-Link of Placement predictions Dataset: https://www.kaggle.com/code/arindambaruah/placement-predictions-using-log-reg-knn-rfc-xgb/input, Access on Date: 15/05/2023

[19] Harihar, V. K., & Bhalke, D. G. (2020). Student Placement Prediction System using Machine Learning. SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology, 12(SUP 2), 85-91.

[20] Shejwal, P. N., Patil, N., Bobade, A., Kothawade, A., & Sangale, S. (2019). A Survey on Student Placement Prediction using Supervised Learning Algorithms. International Journal of Research in Engineering, Science and Management, 2(11), 2581-5792.

[21] Shukla, M., & Malviya, A. K. (2019). Modified classification and prediction model for improving accuracy of student placement prediction. In Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE).

[22] Hariharan, V. J., Abdullah, S., Rithish, R., Prabakar, V., Suguna, M., Ramakrishnan, M., & Selvakumar, S. (2022). Predicting student's placement prospects using Machine learning Techniques. Available at SSRN 4140544.

Authors' Profiles

Mukesh Kumar worked as an Assistant Professor in School of Computer Applications, Lovely Professional University, Phagwara, Punjab, India. Prior to his foray into academia, he completed his M. Tech and Ph.D. in Computer Science from HPU Shimla. His research interest includes Educational Data Mining, Machine Learning, Artificial Intelligence. He has 13 years of teaching experience and published 45 research papers in different National and International journals and conferences.

Nidhi Walia is a PhD scholar in Maharaja Agrasen University, Solan, India. Before coming into PhD, she had worked as Senior Research fellow in CSIR-CSIO, India. She has total of 6 years of experience in teaching and Research. Her main interest includes Machine learning, data analysis and Image/Video Processing. She has 4 years of teaching experience and published 10 research papers in different international journals and conferences.

Sushil Kumar Bansal is a Professor within the Department of Computer Science & Engineering at Maharaja Agrasen University, Baddi (Himachal Pradesh). He is having a vast Industrial and Teaching experience of 25 Years. He did his Bachelor of Technology in Computer Science and Engineering from Dr.BR Ambedkar Regional Engineering College , Jalandhar , Master of Technology in Computer Science and Engineering from NITTTR Chandigarh and PhD in Computer Science and Engineering from LPU Jalandhar. His research interests are in Software Security, Machine Learning and IOT. He has published many papers in refereed journals, conference proceedings and book chapters on his research areas.

Girish Kumar holds a B.Sc. (Computer Science) Degree and PGDCA, MIT from GNDU and is a Research Scholar currently working as an Assistant Professor at Lovely Professional University. He has more than 21 years of teaching experience. He has four patents to his credit and has published more than 20 research papers in different national as well as international conferences and journals. He has authored four books published by reputed national and international publishers. He is also a Certified Academic Associate by IBM for DB2. He is an active member of IAENG- International Association of Engineers.

Korhan Cengiz was born in Edirne, Turkey, in 1986. He received the BSc degrees in Electronics and Communication Engineering from Kocaeli University and Business Administration from Anadolu University, Turkey in 2008 and 2009 respectively. He took his MSc degree in Electronics and Communication Engineering from Namik Kemal University, Turkey in 2011, and the PhD degree in Electronics Engineering from Kadir Has University, Turkey in 2016. Since September 2022, He has been an Associate Professor in the department of Computer Engineering, Istinye University, Istanbul, Turkey. Since April 2022, he has been the chair of the research committee of University of Fujairah, United Arab Emirates. Since August 2021, he has been an Assistant Professor at the College of Information Technology in University of Fujairah, UAE. Dr. Cengiz is the author of more than 40 SCI/SCI-E articles including IEEE Internet of Things Journal, IEEE Access, Expert Systems with Applications, Knowledge Based Systems and ACM Transactions on Sensor Networks, 5 international patents, more than 10 book chapters, and 1 book in Turkish. He is editor of more than 20 books.

How to cite this paper: Mukesh Kumar, Nidhi Walia, Sushil Bansal, Girish Kumar, Korhan Cengiz, "Predicting College Students’ Placements Based on Academic Performance Using Machine Learning Approaches", International Journal of Modern Education and Computer Science(IJMECS), Vol.15, No.6, pp. 1-13, 2023. DOI:10.5815/ijmecs.2023.06.01