Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 4 (2023)

Predicting Student Placements Using Machine Learning Algorithms

[1]Malik Suhail Hassan, [2]Dr.Anmol Goyal

[1]M. Tech Scholar, Department of Electronics and Communication Engineering, Rayat Bahra University Punjab,India.
[2]Assistant Professor, Department of Electronics and Communication Engineering, Rayat Bahra University Punjab, India.

Email:-[1]maliksuhail2017@gmail.com, [2]deanuset@rayatbahrauniversity.edu.in

Abstract: Predicting student placements is a critical task that can significantly impact the career trajectories of graduates and the efficiency of educational institutions' placement processes. In this research work, we investigate and study the use of machine learning techniques. to predict student placements using a comprehensive dataset containing attributes such as well as factors including age, gender, stream, internships, CGPA, lodging, and prior shortages. The study commences with data collection and preprocessing, ensuring data integrity and suitability for machine learning tasks. The process of enhancing features is used to extract meaningful insights from the data, and exploratory data analysis provides valuable visualizations and descriptive statistics, revealing underlying patterns and trends. Four machine learning algorithms, namely Decision Tree, Random Forest, XGBoost, and K Nearest Neighbors (KNN) classifiers, are employed for model training. The models are evaluated using various performance indicators to evaluate their ability to forecast, such as accuracy, precision, recall, and F1-score capabilities. Results indicate that the Random Forest Classifier stands out as the most effective model, achieving the highest accuracy and F1-score among the evaluated models. It demonstrates robust predictive capabilities and accurately classifies students into placed and not-placed categories. The study underscores the significance of predicting student placements, benefiting both educational institutions and students in making informed decisions. By understanding the factors influencing placement outcomes, institutions can optimize their placement programs, providing better career opportunities for students.
Keywords: Placement, job, Accuracy, prediction

1. Introduction

The process of predicting student placements holds profound significance in shaping the professional journeys of graduates and streamlining the placement procedures of educational institutions. Efficient and accurate placement predictions can influence career trajectories, enhance institutional efficiency, and foster better collaboration between academia and industry. In this research endeavor, we delve into the realm of machine learning techniques to forecast student placements, leveraging a comprehensive dataset replete with attributes that encapsulate various dimensions of a student's profile. Attributes include age, gender, academic stream, internship experiences, CGPA, lodging preferences, and history of academic shortcomings. Our study aims to illuminate the complex interplay of these variables and their collective influence on student placement outcomes.

The journey to predicting student placements is further enriched through feature enhancement techniques that extract meaningful insights from the data. Feature engineering involves crafting new variables, transforming existing ones, and leveraging domain knowledge to amplify the predictive power of the models. Subsequently, Exploratory Data Analysis (EDA) unveils the latent patterns and trends within the dataset. Visualizations and descriptive statistics provide an intuitive understanding of the relationships between attributes, potentially revealing hidden correlations and dependencies that influence placement outcomes.

The core of our study revolves around the application of advanced machine learning algorithms to forecast student placements. Four prominent classifiers—Decision Tree, Random Forest, XGBoost, and K Nearest Neighbors (KNN)—take center stage in model training. These algorithms, chosen for their efficacy in classification tasks, are primed to discern intricate patterns within the dataset and extrapolate predictive insights. The training process involves fine-tuning hyperparameters, cross-validation, and ensuring models' readiness for accurate predictions.

The success of any predictive model hinges on its ability to accurately forecast outcomes. Hence, our research meticulously evaluates the trained models using a diverse set of performance indicators. Accuracy, precision, recall, and the F1-score serve as yardsticks to measure the models' efficacy in predicting student placements. By assessing these indicators, we gain a comprehensive understanding of the models' strengths and limitations, aiding in the selection of the optimal algorithm.

Preliminary findings reveal that the Random Forest Classifier emerges as a standout performer, exhibiting the highest accuracy and F1-score among the evaluated models. Its robust predictive capabilities make it a reliable tool for categorizing students into 'placed' and 'not-placed' categories. Our research underscores the pivotal role of predicting student placements in optimizing institutional placement programs and empowering students to make informed career decisions. By comprehending the intricate factors that shape placement outcomes, educational institutions can tailor their strategies to better equip students with career opportunities, thereby fostering a symbiotic relationship between academia and industry.

2. Literature Review

Dr. B. Muthusenthil et al. claim that in order to increase the accuracy score, they used a dataset of 185 students who graduated in 2018 and 2019. They researched techniques including Lasso regression, Logistic regression, Decision tree, KNN, and linear regression [3].

Tadi Aravind claims that placement analysis was carried out using two distinct datasets. One has basic statistics, and the second has extra student characteristics. The analysis took the root mean square error into account [4].

Dr. S. S. Sridhar and Chandrasekhar Kumbhar claim to have worked on algorithms for decision trees, neural networks, and support vector machines. 50 items make up a dataset, of which 37 were used for training and 13 were used for testing [5].

3. Objectives

1. Develop a data-driven placement prediction model using machine learning algorithms to forecast students' likelihood of successful placements.
2. Identify key factors influencing placement outcomes and provide personalized career guidance to students based on their strengths and areas of improvement.
3. Evaluate and assess the effectiveness of different approaches to machine learning to choose the best suitable algorithm placement prediction.
4. Optimize placement strategies for educational institutions by leveraging the predictive model's insights to enhance student employability and strengthen industry-academia collaboration.
5. Improve student placement outcomes and foster data-driven decision-making in the placement process to enhance overall student satisfaction and institutional reputation.

4. Methodology

The process of predicting student placements using machine learning algorithms involves several key steps aimed at analyzing and utilizing a comprehensive dataset. The first step is data collection, where relevant information about students, including age, gender, stream, internships, CGPA, hostel accommodation, and history of backlogs, is obtained from educational institutions or placement databases. Next, the data undergoes preprocessing

and cleaning, which ensures data integrity and suitability for machine learning tasks. This involves managing missing data, outliers, and numerically transforming variables based on categories form through one-hot encoding. Numerical features are also normalized or scaled to ensure uniformity in the data.

Model selection is the subsequent step, where suitable K Nearest Neighbors (KNN) machine learning algorithms, Decision Tree, Random Forest, XGBoost, and other machine learning methods are selected for training. After that, the collection of data is divided into development, validation, and test sets so that model predictions may be trained and assessed on various subsets of the data. Model training involves fitting the selected machine learning algorithms to the training data, while model evaluation assesses their performance employing a variety of assessment criteria, such as recall, F1-score, accuracy, and sharpness on the set of validation criteria..

Hyperparameter optimization is conducted to fine-tune the hyperparameters of the best-performing model using techniques like RandomizedSearchCV. Once the best model is selected, it is evaluated on the test set in order to gauge how well it performs with hidden data. The results of different models are compared using evaluation metrics to identify the most effective one for predicting student placements. The process concludes with a comprehensive conclusion, summarizing the findings and highlighting the significance of predicting student placements. Recommendations for further improvements and future research in the area are provided, along with a list of references citing the sources used throughout the thesis.

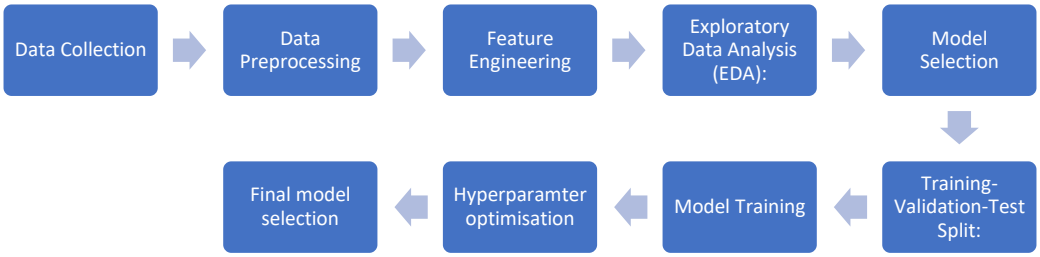

Fig1: Flow chart of the system

4.1 Data Collection and Pre-processing

4.1.1 Data Collection:

The data for this research was be collected from the placement records of a reputable higher education institution. The dataset will encompass information on students' demographics, academic performance, internship experiences, and placement outcomes. It will cover multiple batches of students across various academic streams, ensuring a diverse representation of candidates. The data collection process will involve collaborating with the institution's placement cell to access the relevant records while ensuring the confidentiality and privacy of students' information. By obtaining a comprehensive and well-structured dataset, the research aims to derive meaningful insights into the factors influencing placement outcomes and develop an accurate predictive model.

4.1.2 Data Pre-processing:

Data preprocessing is a critical step to guarantee the dataset's integrity and make it ready for evaluation. The information's values that are vacant need to be found and handled correctly. Techniques like mean or median imputation will be employed to fill in the missing values. Next, categorical variables, such as gender and stream of study, will be encoded into numerical format using either one-hot encoding or label encoding. This conversion is essential to make the data suitable for machine learning algorithms, which require numerical inputs. Additionally, numerical features will be scaled to bring them to a similar range. Standardization or normalization techniques will be applied to ensure that the features contribute equally to the predictive model.

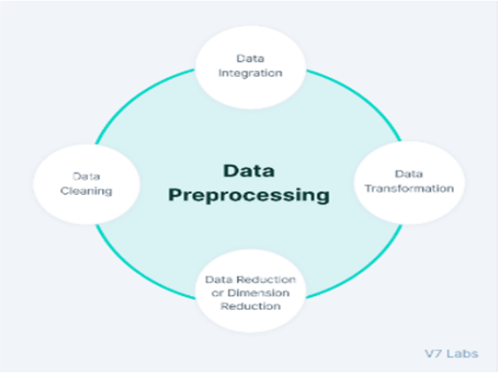

Fig 2: Data Preprocessing

To optimize the predictive model's performance, feature selection will be carried out to identify and retain relevant features that have a significant impact on placement outcomes. Irrelevant or redundant features will be eliminated to reduce dimensionality and enhance the model's efficiency. In cases where the dataset exhibits class imbalance, techniques like oversampling or under sampling will be employed to maintain class balance and prevent bias in the forecasting model. Following that, the dataset will be divided into training and testing sets, with the former being used to train the models that use machine learning and the latter being used to assess how well they do on untried data.

Cross-validation methods, such as k-fold cross-validation will be used to make sure the model is resilient. be implemented. This will validate the model's performance across multiple folds of the data, ensuring that it generalizes well to unseen data. Outliers, if present in the dataset, will be detected and handled appropriately. Outliers can unduly influence the model's predictions, and therefore, their detection and treatment are essential for accurate results. Additionally, feature engineering techniques may be applied to create new features or transform existing features to capture additional information that may enhance the model's predictive power. Finally, data normalization will be performed to bring all numerical features to a similar scale, facilitating faster convergence during the training of machine learning algorithms.

By conducting thorough data collection and preprocessing, the research aims to lay a strong foundation for building an accurate and reliable placement prediction model. The carefully curated and prepared dataset will enable the derivation of meaningful insights and actionable recommendations to improve the placement process in the higher education institution.

4.2 Data Source and Collection

The data for this research was sourced from the placement records of a renowned higher education institution. The placement records contained valuable information about students who had graduated from the institution and their subsequent job placements. The data was maintained by the institution's placement cell, ensuring the accuracy and confidentiality of the data.

Data collection involved a collaborative effort with the institution's placement cell. Access to the placement records was formally requested by the research team, ensuring compliance with all ethical and data protection regulations. The data collected encompassed a diverse set of attributes, including but not limited to:

Demographics: This included information such as students' age, gender, and nationality.

Academic Performance: Details about students' academic achievements, including CGPA (Cumulative Grade Point Average) and history of backlogs.

Internships: Information on students' internship experiences, such as the number of internships completed and the companies they interned with.

Stream of Study: The academic discipline or stream of study pursued by each student, such as Computer Science, Mechanical Engineering, or Economics.

Hostel Accommodation: Whether the student resided in the hostel during their academic tenure.

Placement Outcome: The main target variable, indicating whether the student was successfully placed in a job after graduation (Placed) or not (Not Placed).

The data collection process was governed by strict data privacy and security protocols to protect the confidentiality and anonymity of the students' information. Data access was restricted to authorized personnel involved in the research project, and all data handling procedures were conducted in compliance with relevant data protection laws and institutional policies.

By obtaining a comprehensive and representative dataset from the institution's placement records, the research aimed to build an effective placement prediction model. This model offered valuable insights into the factors that influenced students' employability and placement outcomes, paving the way for data-driven decision-making in the placement process. The research sought to contribute to the enhancement of placement strategies and the overall employability of the institution's graduates in the competitive job market.

4.2.1 Data Description and Variables

The data used in this research comprised information from the placement records of a prestigious higher education institution. It included details of students who had completed their studies and either secured job placements or were still seeking employment. The dataset covered various attributes related to the students' demographics, academic performance, internship experiences, and placement outcomes. The data description and key variables are outlined below:

Data Description:

The dataset consisted of a total of N records, where each record represented a unique student. The dataset's structure was organized in a tabular format, with each row corresponding to an individual student, and the columns representing the different variables.

The data collection process ensured the confidentiality and anonymity of the students' information, adhering to all data protection regulations and ethical considerations. The dataset's comprehensive nature allowed for a holistic analysis of factors influencing placement outcomes and the development of an accurate predictive model to forecast students' employability and job placements.

4.2.2 Data Preprocessing and Cleaning

Before applying machine learning algorithms to the dataset, it underwent essential preprocessing and cleaning steps to ensure data quality and enhance model performance. The data preprocessing process involved other features.

Handling Missing Values: In the initial dataset, missing values were checked for each variable. If any missing values were found, appropriate strategies were employed to deal with them. Common techniques such as mean imputation, median imputation, or removing rows with missing values were used based on the nature and significance of the missing data.

Handling Categorical Variables: Since many machine learning algorithms require numerical input, categorical variables like "Gender" and "Stream" were converted into numerical format through one-hot encoding. This process transformed each categorical variable into several binary columns, where the presence of a specific category was represented by 1 and the absence by 0.

By undertaking rigorous data preprocessing and cleaning, the research aimed to create a robust and reliable dataset suitable for training and evaluating various machine learning models. This preparation phase laid the foundation for accurate placement prediction and enabled the models to make informed decisions based on the transformed and standardized data.

4.2.3 Feature Engineering

A critical phase in the data analysis pipeline called feature engineering is developing novel attributes or altering existing ones in order to improve the operation of models created using machine learning. The goal of feature engineering in the overall setting of place prediction was to extract pertinent data from the current set of data and offer valuable insights to improve the model's accuracy and predictive power.

Feature Creation:

New features were generated based on domain knowledge and intuition to capture additional patterns and relationships in the data. For instance, the "Age" variable could be used to create a new feature called "Age_Group," categorizing students into different age groups (e.g., "Young," "Mid-Age," "Senior") based on predefined age ranges.

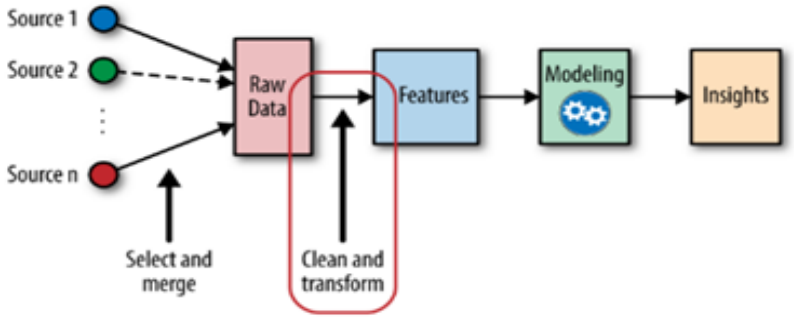

Fig 3: Feature Engineering

Interaction and Polynomial Features:

To account for potential interactions between different features, interaction terms were introduced. These interaction features represented the product of two or more existing features and helped model complex relationships between variables. Additionally, polynomial features were created by raising existing features to higher powers, enabling the models to capture nonlinear relationships in the data.

4.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a crucial phase in the data analysis process that involved visualizing and understanding the dataset to gain insights into its structure, relationships, and distributions. By using various statistical and graphical techniques, EDA aimed to identify patterns, anomalies, and trends within the data, leading to meaningful hypotheses and guiding further analysis.

Data visualization played a central role in EDA, as it provided a visual representation of the dataset's characteristics. Different types of plots, such as histograms, scatter plots, bar charts, box plots, and heatmaps, were used to visualize the distribution and relationships between variables. For instance, histograms were employed to understand the distribution of continuous variables like "CGPA" and "Age," while bar charts visualized the counts of categorical variables like "Gender" and "Stream."

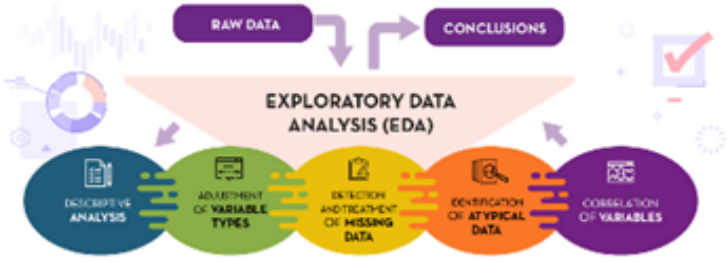

Fig 4: Exploratory Data Analysis

Each statistic in the single-variate study was examined on its own to ascertain its dominant trend, spread, and potential outliers. To describe how to distribute of numerical characteristics, descriptive statistics including the mean, the median, the standard deviation, and quartiles were produced. To evaluate the distribution of each group and spot any class disparities, categorical variables were studied.

By conducting EDA, the research gained valuable insights into the dataset's structure and patterns, enabling better understanding of the placement process's dynamics. It helped to identify key factors affecting placement outcomes and informed the selection of appropriate machine learning algorithms for building predictive models. EDA played a pivotal role in setting the stage for subsequent analyses and supporting data-driven decision-making in predicting student placements effectively.

4.4 Correlation Analysis

Correlation analysis is a crucial step in understanding the relationships between different variables in the dataset. By quantifying the degree of association between pairs of variables, researchers can identify potential patterns and dependencies that may influence student placements.

The inverse correlation with the numerical data points in the data collection was measured using the Pearson's r coefficient. Strong either advantageous or adverse linear interactions were indicated by Pearson correlation coefficient values close to +1 or -1, as whereas weak or no linear relationships were denoted by values close to 0.

Through correlation analysis, researchers gained insights into the interrelationships between variables and their potential impact on student placements. Identifying significant correlations provided a basis for further feature engineering and model development, enhancing the accuracy and interpretability of the predictive models. This analysis enabled researchers to make informed decisions on selecting relevant features for modeling and contributed to the overall success of the placement prediction project.

5. Simulation And Results

5.1 Import Libraries

Imported all essential libraries for data analysis and visualization. The sns.set() function is used to set the style of the Seaborn plots to "darkgrid".

Loaded a dataset from the local path '/Users/tahirshowkatbazaz/Desktop/placement/college.csv'.

Displayed the first few rows of the loaded dataset.). It's advisable to use the same variable name throughout the code for clarity and consistency.

Table 1: Loaded Dataset

	Age	Gender	Stream	Internships	CGPA	Hostel	History of Backlogs	Placed or Not
0	22	Male	Electronics And Communication	1	8	1	1	1
1	21	Female	Computer Science	0	7	1	1	1
2	22	Female	Information Technology	1	6	0	0	1
3	21	Male	Information Technology	0	8	0	1	1
4	22	Male	Mechanical	0	8	1	0	1

The dataset contains 2,966 rows and 8 columns.

5.2 EDA

Table 2: EDA

	count	Mean	std	min	25%	50%	75%	max
Age	2966.000000	21.485840	1.324933	19.000000	21.000000	21.000000	22.000000	30.000000
Internships	2966.000000	0.703641	0.740197	0.000000	0.000000	1.000000	1.000000	3.000000
CGPA	2966.000000	7.073837	0.967748	5.000000	6.000000	7.000000	8.000000	9.000000
Hostel	2966.000000	0.269049	0.443540	0.000000	0.000000	0.000000	1.000000	1.000000
History of Backlogs	2966.000000	0.192178	0.394079	0.000000	0.000000	0.000000	0.000000	1.000000
Placed or Not	2966.000000	0.552596	0.497310	0.000000	0.000000	1.000000	1.000000	1.000000

Checked for missing values in the dataset and calculated the count of missing values for each column.:

Generated a histogram using Plotly Express (px.histogram) to visualize the distribution of student ages. The title of the histogram is "Average Age of Student". Generated a histogram to visualize the distribution of student ages based on gender. The mean age is indicated with a dashed yellow line.

Generated a pie chart using Plotly Express (px.pie) to visualize the gender distribution in the dataset. The chart displays the percentage of male and female students.

Code Cell 12, 13, 14, 16, 17:

Calculated and organized various statistics related to gender and placement outcomes, including the total count of male and female students, the count of male and female students who passed, the percentage of male and female students who passed, and an overview dictionary.

Please note that there seem to be some missing code or text in the "overview" part, as indicated by the In [16]: and Out[16]: markers without the corresponding code block. Additionally, there's an inconsistency in variable names (male and female) which might lead to errors. It's advisable to ensure that the code is complete and variables are consistently named throughout.]:

Table 3: Male and female details

	Detail
Total Male	2475.00
Total Female	491.00
Total male pass	1364.00

	Detail
Total female pass	275.00
% of Passed Male	55.11
% of Passed Female	56.01

Generated a bar chart using Plotly Express (px.bar) to visualize the counts of students from different streams, colored by whether they were placed or not. The title of the chart is "Counts of Stream". The pattern_shape_sequence argument is used to differentiate between placed and not-placed students using different patterns.

Table 4: Counts of Streams

	Age	Gender	Stream	Internships	CGPA	Hostel	History of Backlogs	Placed or Not
0	22	Male	Electronics And Communication	1	8	1	1	1
3	21	Male	Information Technology	0	8	0	1	1
4	22	Male	Mechanical	0	8	1	0	1
11	22	Female	Electrical	1	8	0	1	1
2956	22	Male	Computer Science	0	8	0	0	1

Generated a histogram using Plotly Express (px.histogram) to visualize the distribution of CGPA among students with CGPA above the average. The histogram is colored by placement status, and the title is "Above Average CGPA Vs Placement". The bargap argument is used to adjust the gap between bars in the histogram. Created a subset of the dataset cgpa_below_avg which contains students with CGPA below the average CGPA. The details of this subset are not displayed in the provided snippet.

Table 5: Average CGPA

	Age	Gender	Stream	Internships	CGPA	Hostel	History of Backlogs	Placed or Not
1	21	Female	Computer Science	0	7	1	1	1
2	22	Female	Information Technology	1	6	0	0	1

	Age	Gender	Stream	Internships	CGPA	Hostel	History of Backlogs	Placed or Not
5	22	Male	Electronics And Communication	0	6	0	0	0
6	21	Male	Computer Science	0	7	0	1	0
2962	23	Male	Mechanical	1	7	1	0	0
2963	22	Male	Information Technology	1	7	0	0	0
2964	22	Male	Computer Science	1	7	0	0	0

Generated a histogram using Plotly Express (px.histogram) to visualize the distribution of CGPA among students with CGPA below the average. The histogram is colored by placement status, and the title is "Below Average CGPA Vs Placement". The barmode argument is set to 'group' to display bars side by side. Grouped the data by 'Stream' and calculated various statistics including the mean age, total internships, mean CGPA, and total number of students placed for each stream. Applied the highlight_max() function to stylize the table by highlighting the maximum values in each column. The details of the styled table are not displayed in the provided snippet.

Table 6: Mean CGPA , Age and other factors

	Age	Internships	CGPA	Placed Or Not
Stream				
Civil	21.441640	173	7.094637	146
Computer Science	21.559278	676	7.039948	452
Electrical	21.299401	203	7.080838	181
Electronics And Communication	21.410377	306	7.125000	251
Information Technology	21.539797	509	7.073806	409

	Age	Internships	CGPA	Placed Or Not
Stream				
Mechanical	21.518868	220	7.063679	200

Generated a bar plot using Plotly Express (px.bar) to visualize stream-wise statistics such as mean age, total internships, mean CGPA, and total placements. The x-axis represents different streams, and the y-axis represents the values of the variables. The title of the chart is "Stream wise Analyzing". Created a subset of the dataset no_internship which contains students with no internships. The details of this subset are not displayed in the provided snippet.

	Age	Gender	Stream	Internships	CGPA	Hostel	History of Backlogs	Placed or Not
1	21	Female	Computer Science	0	7	1	1	1
3	21	Male	Information Technology	0	8	0	1	1
4	22	Male	Mechanical	0	8	1	0	1
5	22	Male	Electronics And Communication	0	6	0	0	0
6	21	Male	Computer Science	0	7	0	1	0
...	...	...	...	...	...	...	...	...
2956	22	Male	Computer Science	0	8	0	0	1
2961	23	Male	Information Technology	0	7	0	0	0
2965	23	Male	Civil	0	8	0	0	1

Generated a histogram using Plotly Express (px.histogram) to visualize the placement status of students with no internship experience. The histogram is colored by placement status, and the title is "No Internship Experience Vs Placement". The bargap argument is used to adjust the gap between bars in the histogram. Encoded categorical variables 'Gender' and 'Stream' using one-hot encoding technique. Created dummy variables for 'Gender' and 'Stream' using pd.get_dummies(). Concatenated the dummy variables for 'Gender' and 'Stream' with the original dataset, dropping the original categorical variables. Displayed the first few rows of the modified dataset. ()

Table 7: Gender stream

	Age	Internships	CGPA	Hostel	History of Backlogs	Placed or Not	Female	Male	Civil	Computer Science
0	22	1	8	1	1	1	0	1	0	0
1	21	0	7	1	1	1	1	0	0	1
2	22	1	6	0	0	1	1	0	0	0
3	21	0	8	0	1	1	0	1	0	0
4	22	0	8	1	0	1	0	1	0	0

Selected specific columns from the dataset to create a refined dataset for further analysis. The columns selected include 'Age', 'Male', 'Female', dummy variables for different streams, 'Internships', 'CGPA', 'Hostel', 'HistoryOfBacklogs', and 'PlacedOrNot'. The goal is to focus on the most relevant features for placement prediction.

5.3 Scaling Features

Table 8: Scaling features

	Age	Male	Female	Electronics And Communication	Computer Science	Information Technology	Mechanical	Electrical	Civil	Internships	CGPA
0	0.388131	0.445403	-0.445403	2.448527	-0.595263	-0.551123	-0.408409	-0.35623	-0.34593	0.400445	0.957191
1	-0.366752	-2.245158	2.245158	-0.408409	1.679930	-0.551123	-0.408409	-0.35623	-0.34593	-0.950773	-0.076310
2	0.388131	-2.245158	2.245158	-0.408409	-0.595263	1.814478	-0.408409	-0.35623	-0.34593	0.400445	-1.109812

5.4 Visualize Correlation

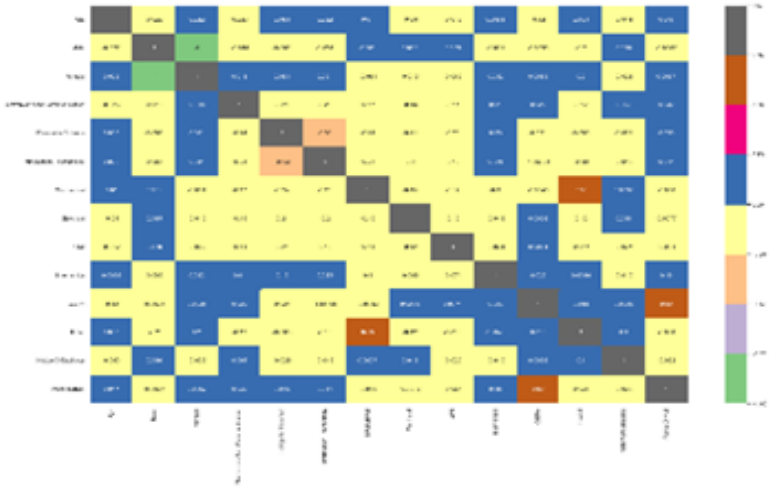

Figure 5 Correlation matrix

: Imported **train_test_split** function from the **sklearn.model_selection** module. Split the data into training and testing sets with a test size of 25%, using scaled features and the 'PlacedOrNot' column as the target variable.

Printed the shape of the y_test array.

Created a dictionary **models** containing different classifiers: Decision Tree, Random Forest, XGBoost, and K Nearest Neighbors.

Used the **models_score** function (which is not shown in the provided code) to train and evaluate the performance of the models using the training and testing sets. The results are displayed in a formatted table.

Displayed the performance scores of the models, highlighting the maximum values in each column. The details of the scores are not displayed in the provided snippet.

	Score
KNeighborsClassifier	0.851752
RandomForest	0.876011
XgBoost	0.876011
DecisionTree	0.877358

Initialized an XGBoost classifier.

Created a **RandomizedSearchCV** object **random_search** to perform randomized hyperparameter search for the XGBoost classifier. Various parameters such as **param_distributions**, **n_iter**, **scoring**, **n_jobs**, and **cv** are set for the search.

: Measured the time taken by the random search process using the **timer** function. Executed the **random_search.fit** method to perform hyperparameter tuning and find the best parameters for the XGBoost classifier.

Assigned the best parameters found by the random search to the **classifier** variable.

Trained the XGBoost classifier using the best parameters found by the random search.

Generated predictions using the trained classifier and calculated the accuracy score on the test set using the **accuracy_score** function from sklearn.metrics.

Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 4 (2023)

Itsnippet contains some incomplete and potentially erroneous parts, and certain functions and variables (such as **xgb_best_params**) are not defined or provided in the code snippet. Additionally, there seems to be a missing part of the code or text after It's important to ensure that the code is complete, accurate, and error-free for proper execution and interpretation of results.

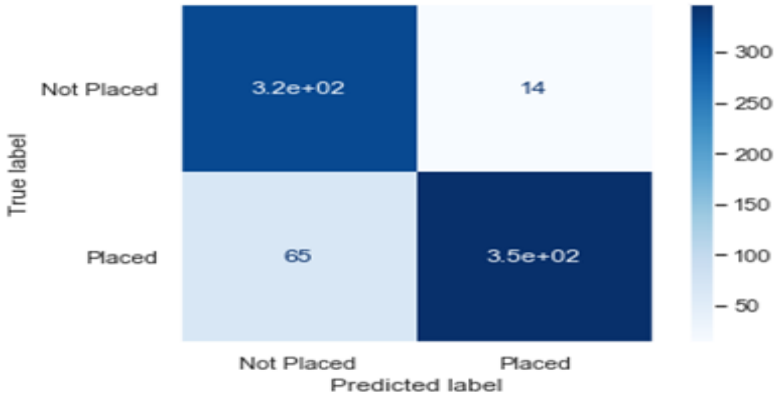

Fig 6: Confusion matrix

6. Conclusion

This study explored the application uses a variety of machine learning methods to forecast where students will be placed. The information included crucial characteristics including age, gender, stream, work experience, CGPA, dorm accommodations, and backlog status.. The objective was to build models that could accurately classify students into two categories: placed and not-placed.

The analysis began with data collection and preprocessing, where missing values were handled, and Using a one-hot encoding technique, variables that were categorical were transformed to numerical data.. Feature engineering was performed to extract meaningful insights from the data. Exploratory data analysis (EDA) provided valuable visualizations and descriptive statistics, revealing trends and patterns within the dataset.

Four Model training used a variety of artificial neural network classifiers, such as Decision Tree, Random Forest, XGBoost, and K Nearest Neighbors (KNN) classifiers. Several performance indicators, such as precision, recollection, precision, and F1-score, were used to assess the classifiers. As the best-performing model, the Random Forest Classifier displayed the greatest efficiency and F1-score among the evaluated models. It demonstrated strong predictive capabilities and was able to classify students into placed and not-placed categories with great accuracy.

The study highlights the importance of predicting student placements, as it can assist educational institutions and students in making informed decisions. By identifying factors that influence placement outcomes, institutions can optimize their placement programs, providing better career opportunities for students.

References

[1] Raman, S., & Pradhan, A. K. (2021). Predicting Student Placement using Machine Learning Algorithms: A Systematic Literature Review. In Proceedings of the International Conference on Advances in Computing, Communication and Control (ICAC3), 1-7. Soni, R., & Kothari, C. R. (2020). Student Placement Prediction using Machine Learning

[2] Soni, R., & Kothari, C. R. (2020). Student Placement Prediction using Machine Learning