Mehran University Research Journal of Engineering and Technology

https://doi.org/10.22581/muet1982.2302.18

2023, 42(2) 169-187

A data mining approach to forecast students' career placement probabilities and recommendations in the programming field

Khalid Mahboob [a, b, *], Raheela Asif [c], Najmi Ghani Haider [d]

[a] *Department of Computer Science and Information Technology, N.E.D University of Engineering and Technology, Karachi Pakistan*

[b] *Department of Software Engineering, Sir Syed University of Engineering and Technology, Karachi Pakistan*

[c] *Department of Software Engineering, N.E.D University of Engineering and Technology, Karachi Pakistan*

[d] *Department of Computer Science and Information Technology, N.E.D University of Engineering and Technology, Karachi Pakistan*

[*] Corresponding author: Khalid Mahboob, Email: nedian07@cloud.neduet.edu.pk

Received: 24 February 2023, Accepted: 24 March 2023, Published: 01 April 2023

K E Y W O R D S

Career

Programming

Classification Tree

Accuracy

Forecast

Data Mining

A B S T R A C T

The career opportunities in computer programming are vast and rapidly increasing. Skilled software engineers, programmers, and developers are vigorously in demand worldwide. The capability to forecast a student's future career can be helpful in a wide variety of pedagogical practices. Data mining is becoming a more robust tool for analysis and forecasting. Therefore, to forecast career placement probabilities in the programming field, data mining classification and forecast techniques are used in this study to facilitate prospective students to make sensible career decisions. To achieve this objective, passed-out graduates' data is utilized, which comprises features like graduates' educational attainments in pre-university grades, i.e. grades of matriculation and intermediate, programming courses taught in early semesters along with the Cumulative Grade Point Average (CGPA) with the internship experience, gender, and family demographic information. Various multi-way Classification Trees are generated, which could help students to choose a branch with high career placement probabilities. From historical data, the Classification Trees have determined whether the branch is 'Good', 'Satisfactory', or 'Poor' based on the given information. The experimental findings indicate that all the features significantly influence the career placement probabilities in the programming field.

1. Introduction

1.1 Contextual and Hypothetical Framework

As the world is changing rapidly, education is rising as an essential means for fostering people, developing society, and building an important mechanism for a quality workforce developing worldwide. Therefore, the growth of education is still crucial and dedicated to cultivating human resources to keep walking back and forth with changes in the economic and social systems to for establish an education system to better academics and careers [1–3].

The biggest challenge to pursuing higher education is choosing an appropriate field for higher studies. Therefore, one of the most vital decisions for students is

to make a better career choice based on their field of study in life. Nevertheless, it often happens that the decision is strongly influenced by the factors around students, such as career doubts, parents, peers, etc. Therefore, in such a case, students probably make the wrong decision regarding their career selection according to their interests [4–6].

The main task is to identify critical factors influencing student career planning which acts as a stepping-stone in a career. Among the careers, one of the most important and worthwhile careers is computer programming. Because computer programming has turned, up as an essential field of innovation, research, and career in the last few decades. The career opportunities in computer programming are endless and constantly evolving. Hence, careers in computer programming are always very demanding for developers, programmers, and software engineers worldwide [7, 8].

Hence, educational data mining is employed in this study to extract relevant information from graduated student data. Educational data mining is the large-scale processing of data to find hidden patterns and relationships in educational data. Educational data mining is an evolution to accumulate and infer the data stored in a database to retrieve information that can reveal hidden knowledge and patterns. In addition, data mining aims to find important information mixed with other data [3, 5, 9].

1.2 Research Objectives and Questions

In this study, data mining techniques are used to find out career placement probabilities in the field of computer programming [10]. The rationale is to anticipate information concerning forthcoming students' career placement probabilities in the programming field that could help them find a better fit in programming careers or not. This study focuses on five different perspectives regarding career placement probabilities in the programming field.

Question 1: Is it possible to forecast career placement probabilities in the field of computer programming using graduates' pre-university grades with sufficient accuracy?

Question 2: Is it possible to forecast career placement probabilities in the field of computer programming using graduates' programming courses and CGPA with sufficient accuracy?

Question 3: In the field of computer programming, is it possible to forecast career placement probabilities using a graduates' family background with sufficient accuracy?

Question 4: To what extent does an internship experience in computer programming influence computer programming career placement probabilities with sufficient accuracy?

Question 5: Does gender influence career aspirations relating to career placement probabilities in the field of computer programming?

In the following sequence, the remainder of the paper is structured. The next section is dedicated to the literature analysis and is accompanied by a summary of approaches to data mining. Then, in section four, we describe the data, pre-processing, and methodology for this research. In the following section, the results and discussion are presented. The final section provides a conclusion and addresses emerging directions for future endorsements.

2. Literature Review

This study involves critical perspectives of educational data mining, i.e., focusing on forecasting graduates' academic performances and linking this with predicting students' career placement probabilities, especially relating to the field of computer programming. This review highlights the present literature's strengths and weaknesses and imparts a unique contribution that studies ensure in this field.

2.1 Related Works on Predicting Graduates' Academic Performance

Different data mining methods have been applied in [11] to study undergraduate students' academic performances in four years. They have concentrated on two aspects of students' performance. First, predicting students' achievement at the end of the four-year degree program; second, reviewing typical progressions and combining them with predicted results. Various classification techniques such as artificial neural networks, decision tree induction, k-nearest neighbours, naive bayes, random forest trees, rule induction, and clustering techniques such as k-means and x-means have been employed in a study. As a result, two main groups of students were identified, i.e., the low and high-attaining students. The study results show that a small number of courses are indicators of good or bad performance. In particular, it can be possible to promptly provide

warnings, support low-achieving students, counsel, and provide high-performing students opportunities.

Three classification algorithms, including J48 decision trees, k-nearest neighbour, and naïve bayes, have been used in [12] to assess students' performances in different engineering technologies. In this study, only one (primary) course using pre-examination marks of three different engineering technologies belonging to different cohorts are analysed to determine the pedagogical progress of students in their related engineering fields and their learning behaviours in the specific courses to prepare for the final examination based on their pre-examination marks. The study further revealed that the J48 decision tree and k-NN classification techniques attained the highest accuracies.

A study in [13] is aimed to validate a tool that measures students' learning behaviours and participation with skills that are defined as essential to student achievement. The academic, behavioural skills with self-efficacy, social skills, self-control, behavioural engagement, cognitive engagement, emotional engagement, correlation of employment and education, behavioural skills, and academic performance are the selected measures used in this validation study. Data of 8,520 students studying in 10th grade from four countries were analysed through item response theory. The research shows a positive and significant correlation among the selected measures involving self-reporting and performance-based academic performance.

An investigation in [14] applied k-means and x-means clustering techniques to find the association between students' academic performance and their personal and social factors. These findings indicate a set of personal and social factors that significantly influence students' performance, such as parental occupation, parental qualifications, and income levels. The study concluded that results from both algorithms show that parental occupation, parental qualifications, income level, and the number of hours spent with friends per week perform an essential role in students' academic performance. Furthermore, percentage of high school, family size, mode of transportation, parental status, and the number of friends were not influential factors.

A study in [15] describes a Spark-based framework for information extraction using raw data. The data was divided among the 14 faculties with 61271 undergraduate students per 2389 courses. This record contains 35 fields with details about students like grades in some courses. Still, the study focused on four primary areas like student identity, faculty name, course identification, and related courses. Different machine learning techniques like Baseline, IBCF, UBCF, ALS, ALS_NN, ALS_NN_IBCF, and ALS_IBCF were integrated for training the prediction model. An evaluation of five experiments shows that the finding for influencing factors or aspects plays a vital role in the accuracy of prediction problems.

2.2 Related Works on Predicting And Guiding Graduates' Careers

Incremental ensemble techniques to predict students' career choice is presented in [16]. The three classifiers, including K-NN, Naïve Bayes, and SVM, have been used with a voting scheme. The dataset was based on a psychometric test with 300 students in 16 to 20. The 300 samples of 10 attributes and seven classes show the type of student interest. The algorithms for ensemble training used in these findings are considered helpful for providing the best career choices for students with better accuracy.

A Multi-way Decision Tree using an information gain is generated in [17], which helps applicants choose a high career placement branch. Data comprise feedback from diploma holders, graduates, and post-graduates in engineering from various engineering institutions and polytechnics between 2000 and 2003. The prediction model is based on the data between 2000-2002 and tested the data from 2003. The dataset used in this exploration contains student information about gender, reservation, sector, and entrance rank. The results returned or predicted the branch that can be excellent, good, average, or poor for the previous records applicants.

Students' career choices depend on their professional skills, the regularity of attitudes, and other relevant behaviours discussed in [18]. The Approach Cluster Centres based on XGBOOST (ACCBOX) model was proposed concerning students' career choice predictions. Evaluation using the smart card data set of 4,634 students in the same grade belonging to 16 colleges. The dataset used in this research entails four types of data: academic performance, students' necessary information, behavioural data, and data on career choice. Several experiments show that the method used in research is better than other methods of predicting career choices.

The Intuitive Career System was created to predict a career that meets the student's aptitude and personality in [19]. The students' personalities are defined through their social media accounts via Facebook Graph API. The students' aptitude, personality, and background

information are the parameters set to predict careers. The classification algorithms: K-NN and Stochastic Gradient Descent have been applied to all three datasets. On the other hand, Logistic Regression was used for background information and personality data. However, the Logistic Regression algorithm cannot be applied to aptitude data due to overfitting. Therefore, a Random Forest is used for background data sets to allocate weights to the features. The models provide an average accuracy of 77.41% in support of aptitude, i.e., 75.4% in support of personality and 60.09% in support of background information. The results revealed that the student's aptitude and personality were the responsible predictors concerning correct career decisions.

The recommendation system comprising decision tree and linear regression algorithms has been employed in [20] for career path selection. The career path recommendation system has five modules: students, admin, recommendation, feedback, and chatbot modules. The study results conclude that a recommendation system was developed for the students, and several tests were performed. They analysed the student performances, and the results originated in a chart form.

3. Data Mining Approaches

To extract hidden knowledge from educational data, various data mining strategies of classification and prediction may be used. This section explains the five classification forecasting techniques used in this study: k-Nearest Neighbour, Classification Tree, Support Vector Machine, Naïve Bayes, and Logistic Regression. When a class, also known as a label or a discrete attribute, is forecast using a classifier, it is called classification. A classifier generates a classification model based on training data, which involves objects defined by the values they have with a set of attributes, with one attribute being identified as the class. The created model should match well with the training data and accurately forecast the class or label of data samples, i.e., the test data, which is a different collection of data not used to create the classifier [11, 20]. The data mining classification and forecasting approaches used in this study are described below:

3.1 K-Nearest Neighbour

The k-Nearest Neighbour (k-NN) algorithm is a method of calculating the distance between two points. This data mining algorithm is simple but efficient. It can be used for both classification and regression. It is, however, most commonly used in classification prediction. The spatial domain looks for the k-nearest training examples and uses the average of those as a prediction. The k-nearest neighbour algorithm classifies new unlabelled data by looking at the groups of its closest neighbours. Unlabelled data is determined by a constant number of nearest neighbours in the k-NN algorithm, where k is a positive integer. The value of k is essential since it specifies the algorithm's accuracy and robustness [21].

3.2 Classification Tree

A classification tree is a simple algorithm that divides data into nodes based on the purity of the class. A classification tree consists of root, branches, and leaf nodes. Each internal node represents attribute testing, each branch represents the outcome, and each leaf node represents a class label. The root node is the tree's topmost node. Each internal node represents a test on an attribute. Each leaf node defines a class. Tree pruning eliminates anomalies in the training data caused by noise or outliers. The trees that have been pruned are smaller and less complicated. It can also be used for classification as well as regression [22].

3.3 Support Vector Machine

Support Vector Machine (SVM) is a data mining algorithm that uses a hyperplane to partition the attribute space, optimizing the margin between grouped into various classes or class values. As a result, the method often produces exceptional predictive efficiency. The optimal hyperplane in an SVM is measured to optimize the model's generalization potential. However, suppose the training data are not conditionally independent. In that case, the classifier generated may not have a strong generalization potential, even if the hyperplanes are optimally defined, i.e., the original input space is converted into a higher-dimensional space called "feature space" to optimize the space between classes. As a result, SVM is one of the most efficient and reliable classifications and regression algorithms [23].

3.4 Naïve Bayes

One of the most well-known data mining algorithms is the Naive Bayes algorithm. Primarily focused on Bayes' theorem and the presumption of feature independence, this is a quick and easy probabilistic classifier. From the data, Naive Bayes learns a Naive Bayesian model. As a result, naive Bayes classifiers perform well in diverse real-world contexts. Furthermore, the classifier can be trained incrementally with naive Bayes because it only takes a limited amount of training data to approximate the parameters used for classification. Thus, the algorithm is beneficial for classification tasks [24].

3.5 Logistic Regression

Logistic regression is a classification algorithm that uses supervised learning to predict the probability of a target variable. The existence of the target or dependent variable is dichotomous, implying that there are only two groups. Thus, a logistic regression model forecasts P(Y=1) as a function of X mathematically. In basic terms, the dependent variable is binary, with data coded as 1 (representing success/yes) or 0 (representing failure/no) [25].

4. Data, Pre-processing and Methodology

4.1 Data

The data have been collected from a four-year degree program in Software Engineering comprised of graduates' educational attainments in pre-university grades i.e., grades of matriculation and intermediate programming courses taught in initial semesters along with the Cumulative Grade Point Average (CGPA) with the internship experience, gender, and family demographic information at a private engineering university in Pakistan. The data used in this research encompasses 450 graduates' who graduated in two consecutive academic cohorts respectively.

Features relating to graduates' pre-university grades i.e., matriculation and intermediate, the marks of two programming courses i.e., SWE-102 Programming Fundamentals and SWE-103 Object-Oriented Programming taught over the first year of the degree program, CGPA, gender, internship experience in the field of programming, and the family demographic and socio-economic information such as parent's qualification, income, and occupation are included in the data collection (see Table 1).

The pre-university grades, two programming courses, CGPA, and gender information are gathered from the university database. The information on the remaining features i.e., family demography and internship experience, are accumulated through an online survey using Google forms from December 2019 to December 2021. The Orange software has been used to implement data mining techniques in this analysis [26].

Table 1

List of features with their description

Features	Description		
Pre-University Grades:	Grade	Scale or Percentage	Grade Description
Matric / Intermediate Grades	A-1	80 % or above mark	Outstanding
	A	70 % to 79 % marks	Excellent
	B	60 % to 69 % marks	Very Good
	C	50 % to 59 % marks	Good
	D	40 % to 49 % marks	Fair
	E	33 % to 39 % marks	Satisfactory
Programming Courses and CGPA:			
SWE-102	0-100 marks		
SWE-103	0-100 marks		
CGPA	Grade Point	% Marks	Remarks
	4.00	90-100	Extra Ordinary
	3.7-3.9	85-89	Excellent
	3.5-3.6	80-84	Very Good
	3.2-3.4	75-79	Good
	3.0-3.1	70-74	Above Average
	2.5-2.9	65-69	Average
	2.0-2.4	60-64	Satisfactory
	1.0-1.9	50-59	Pass
	0.00	0-49	Fail
Gender	Male, Female		
Family Demography:			
Parent's Qualification	Some Education, Less than Matric Pass, Matric Pass, Inter Pass, Diploma / Certificate, Bachelor's Degree, Master's Degree, Doctoral Degree,		
Parent's Income	Very High, High, Medium, Low, Very Low		
Parent's Occupation	Banking, Government, Pharmaceutical, Construction, General Business / Trade, Real Estate, Education / Teaching, Engineering, Sales, Private, Management, Retired, Programming		
Internship Experience	Yes, No		

4.2 Pre-processing

Since irrelevant features have a negative impact on proximity measures and eradicate the tendency to perform forecasts with sufficient accuracy, they can make good forecasts impossible. The goal of using available data is to create informative models that turn relevant features into useful information [27]. However, collecting, cleaning, and converting data is essential in the data mining process. Data that is erroneous is perplexing because there is no solution. Pre-processing is important for delivering higher-quality analysis outcomes for data researchers, data scientists, and enterprise users to avoid this [6, 28, 29].

Data is unidentified, obfuscated, unclassified, and unfixed. As a result, it is critical to ensure that users are not burdened by inaccurate, out-of-range, or missing values, which leads to weak solutions. Data cleaning is the process of removing redundancies from a database, and processing and storing missing, noisy, and inconsistent data [5, 29]. Data pre-processing has been performed based on imputing missing values, selecting relevant features, and normalizing features using an Orange tool.

Remove rows with missing values is applied under impute missing values pre-processor. Only the most useful features are output by selecting relevant features pre-processor. ANOVA (that allows to comparison the means of two or more samples) is used to calculate a score. The number of variables on the output is referred to as strategy. The fixed value is set to 10 for the number of features since it returns a set of top-scoring variables. Values are normalized by using normalized features pre-processor to a general scale i.e., standardize to $\mu=0$, $\sigma 2=1$. Values can be centred using the mean or median, or they can be left alone. The dataset chosen for this research is of high quality, with no missing values. If the datasets have missing values or outliers, they must be detected and processed properly.

4.3 Methodology

Classification techniques (or classifiers) forecast the class or label of a data object under data mining techniques. A set of attributes describes a data object. Data items with a known label or class make up a training dataset. A classifier uses a learning algorithm to build a model that best describes the connection between the training dataset's attributes and class labels. The learning algorithm's model should accurately forecast the class label of the testing data, which is unrelated to the training data and hence not needed to create the classifier. Typically, the performance of classification models is assessed by counting the number of test records that the model correctly and wrongly forecasts [30].

There are many different types of classifiers, and none of them is known to perform better in all cases. This is true for educational and career-related information as well. As a result, one must evaluate if one classifier outperforms the others in a specific field. In this research, the five data mining predictive classification techniques that have been employed that have given the best results are k-Nearest Neighbour, Classification Tree, Support Vector Machine, Naïve Bayes, and Logistic Regression [16, 31].

The data, together with the selected features, is inputted into Orange for pre-processing and analysis, as mentioned previously. The data is separated into training and testing segments since it is necessary to evaluate models once they have been trained. Following that, the data of each batch or cohort (a total of 450 occurrences) is trained with a 70 percent training and 30 percent testing split using a fixed proportion of data [32]. The models have been trained or labelled into three classes as ‘Good’, ‘Satisfactory’, and ‘Poor’ according to the graduates' attainments in CGPA and in both programming courses as high, average, and low attainments respectively.

5. Results and Discussion

This section summarizes the results from the methodology mentioned above, which included assessing and comparing the performance of the classifiers, concentrating on research objectives, and evaluating the performance metrics of the classifiers.

5.1 Performance Metrics Evaluation

The data must be correctly processed before executing the classifiers for the model to learn more about the patterns successfully. The preceding section stated the dataset description of academic and demographic features used in this investigation. Assessing a model's quality might be difficult without looking at its training and testing results. The use of a performance metric generally accomplishes this, whether it be measuring the kind of error, the model fit's accuracy, or some other method [32]. To forecast the students' career placement probabilities in the programming field, AUC (Area under the ROC Curve), CA (Classification accuracy), F1-measure, Precision, and Recall are chosen as performance metrics and evaluated to estimate the

classifier's performance on each cohort dataset (see Table 2).

Table 2

Summary of Classifiers Performance Metrics

Classifiers	Cohort 1					Cohort 2				
	AUC	CA	F1	Precision	Recall	AUC	CA	F1	Precision	Recall
k-Nearest Neighbor	0.912	86%	0.846	0.848	0.855	0.936	90%	0.894	0.896	0.902
Classification Tree	0.955	95%	0.947	0.948	0.947	0.964	97%	0.973	0.973	0.973
Support Vector Machine	0.982	90%	0.892	0.902	0.900	0.953	84%	0.797	0.820	0.835
Naïve Bayes	0.988	90%	0.910	0.936	0.902	0.983	91%	0.913	0.921	0.908
Logistic Regression	0.993	93%	0.927	0.934	0.926	0.987	96%	0.955	0.956	0.955

The area under the ROC curve is used to compare the efficacy of tests since the area under a ROC curve is a degree of the expediency of a test overall, with a greater area indicating a more helpful test. Receiver Operating Characteristic (ROC) is a term that refers to how well a technique works [33]. The number of accurate forecasts divided by the total number of forecasts is the classification accuracy metric, which describes the performance of a classification model. It is the most often used metric for comparing classifier models since it is simple to compute and interpret (see Eq. 1).

The F-score, also known as the F1-score, is a metric for how accurately a model has performed on a given dataset. It is used to assess binary classification algorithms that classify instances as either "positive" or "negative." The F1-score, described as the harmonic mean of the model's accuracy and recall, implies combining the model's precision and recall (see Eq. 2). The fraction of true positives with the instances classified as positive is known as precision (see Eq. 3). The fraction of true positives with all positive instances in the data is known as recall (see Eq. 4) [3].

$$Accuracy = \frac{(Correct\ graduates\ career\ placement\ probabilities)}{(Number\ of\ graduates\ instances)} = \frac{(TP+TN)}{(TP+FP+FN+TN)} \quad (1)$$

$$F1 = 2.\frac{(precision.recall)}{(precision+recall)} = \frac{(TP)}{\left(TP+\frac{1}{2}(FP+FN)\right)} \quad (2)$$

$$Precision = \frac{(True\ positive\ graduates\ career\ placement\ probabilites)}{(Total\ positive\ graduates\ career\ placement\ probabilites)} = \frac{(TP)}{(TP+FP)} \quad (3)$$

$$Recall = \frac{(True\ positive\ graduates\ career\ placement\ probabilities)}{(Number\ of\ positive\ graduates\ instances)} = \frac{(TP)}{(TP+FN)} \quad (4)$$

Where,

TP stands for the measure of data that is correctly predicted to be of current relevance to a class.

TN is the measure of data that is correctly predicted to be indifferent to a class.

FP stands for the measure of data that is incorrectly predicted to be of current relevance to a class.

FN stands for the measure of data that is incorrectly predicted as indifferent by a class.

5.2 Prediction Results and Recommendations

The data of the graduates have been collected from two consecutive cohorts comprising academic features i.e. pre-university grades i.e., matriculation and intermediate, the marks of two programming courses i.e., SWE-102 Programming Fundamentals and SWE-103 Object-Oriented Programming taught over the first year of the degree program, and CGPA and demographic and socio-economic features i.e. gender, internship experience in the field of programming,

parent's qualification, income, and occupation to forecast the students' career placement probabilities into three classes classified as good, satisfactory, or poor.

In this research, the five classifiers were analysed. Their performance was evaluated on two separate datasets with 70% training data and 30% testing data split with stratified sampling. Stratified sampling divides the population into smaller groups or strata to complete the training process. The strata are created using standard features found in the population data [34]. The forecast results are based on the training process using the five classifiers: k-Nearest Neighbour, Classification Tree, Support Vector Machine, Naïve Bayes, and Logistic Regression, as mentioned earlier. The performance of the classifiers is then compared in terms of accuracy [35]. Among the classifiers, the Classification Tree gave the highest accuracy for both datasets. The tree visualizations about analysis for both cohort datasets are presented in Appendices A (see Fig.s A1-A5) and B (see Fig.s B1-B5).

A. Classification Tree Forecasts and Recommendations Using Pre-University Grades

The Classification Tree in Fig. A1 shows that the tree is constructed with 23 nodes and 12 leaves for cohort 1 around the feature Intermediate Grade with the class 'Good' as the root-splitting criterion. One notices six instances out of 9 instances are classified as 'Poor' when their Intermediate Grade is A-1, and Matric Grades are A-1, A, B, or C. Similarly, 126 instances out of 216 instances are classified as 'Good' when their Intermediate Grades are A, B, or C and Matric Grades are A-1, A, B, or C.

The Classification Tree in Fig. B1 shows that the tree is constructed with 23 nodes and 12 leaves for cohort 2 around the feature Intermediate Grade with the class 'Good' as the root-splitting criterion. One notices five instances are classified as 'Poor' when their Intermediate Grade is A-1. Similarly, 136 out of 220 instances are classified as 'Good' when their Intermediate Grades are A, B, or C and Matric Grades are A-1, A, B, C, or D.

This recommends that graduates with an A-1 grade in an intermediate will surprisingly have a poor career in the programming field. Therefore, the chances of career placement probability in a programming field are low for A-1 grade holders. However, graduates who got A, B, or C grades in an intermediate will likely have a promising career in a programming field, so the chances of career placement probability in a programming field are high. On the other hand, the graduates who acquired A-1, A, B, or C grades for both cohorts and precisely a D grade for the only second cohort in matriculation will likely have a promising career, and thus, the chances of career placement probability in a programming field are high. Besides, one might wonder if the Classification Tree classifier did not classify the instances with matriculation and intermediate grades in a 'Satisfactory' class. Therefore, Research Question 1 is answered positively in this study, as it is possible to forecast career placement probabilities in computer programming using pre-university grades with sufficient accuracy.

B. Classification Tree Forecasts and Recommendations Using Programming Courses and CGPA

The Classification Tree for cohort 1 is created with nine nodes and five leaves around the feature CGPA, with the class 'Good' as the root-splitting criterion, as shown in Fig. A2. One can observe that graduates get a degree with a CGPA of more than 2.06. They are classified as 'Good' in 127 out of 225 instances. The remaining instances with a CGPA less than 2.06 are classified as 'Poor.' Similarly, when the graduates attained more than 48 marks in both programming courses, i.e., SWE-102 with 36 out of 44 occurrences and SWE-103 with 62 out of 98 occurrences, are classified as 'Satisfactory', and the rest of the instances with marks less than equal to 48 in both courses are classified as 'Poor.'

The Classification Tree for cohort 2 is created with seven nodes and four leaves around the feature CGPA, with the class 'Good' as the root-splitting criterion, as shown in Fig. B2. One can observe that when the graduates got a degree with a CGPA of more than 0.00, they are classified as 'Good' in 137 out of 225 instances, and the rest with a CGPA less than and equal to 0.00 are classified as 'Poor.' Similarly, when the graduates attained more than 47 marks in a programming course, SWE-102 with 57 out of 88 occurrences and more than 37 marks in a programming course, SWE-103 with 31 out of 49 occurrences are classified as 'Satisfactory.' The rest of the instances are classified as 'Poor' with marks less than and equal to 47 in SWE-102 and less than and equal to 37 in SWE-103, respectively.

This recommends that the chances of career placement probability in a programming field will likely be high for those graduates who acquired more than 2.06 CGPA belonging to cohort 1 and more than 0.00 CGPA belonging to cohort 2. In contrast, the graduates who scored more than 48 marks in both programming courses in cohort 1 and more than 47 marks in a course SWE-102 with more than 37 marks in course SWE-103 in cohort 2 will likely have good chances of career placement probability in a programming field.

A 2-dimensional scatter plot visualizations for both cohorts are provided below (see Fig.s 1 and 2). The data is represented as a series of points, with the x-axis attribute defining the horizontal axis position w.r.t the programming course SWE-102 and the y-axis attribute determining the vertical axis position w.r.t the programming course SWE-103. The Fig.s below show that the graduates' marks in SWE-102 and SWE-103 have a positive linear relationship. This implies that as graduates' marks in SWE-103 increase, the graduates' marks in SWE-102 increase as well. It does not imply that an increase in the graduates' marks in SWE-103 triggers the graduates' marks in SWE-102 to increase. It is noteworthy from the scatter plot visualizations that some data instances are overlapped and mixed for all three classes' which means they have misclassified instances for both features.

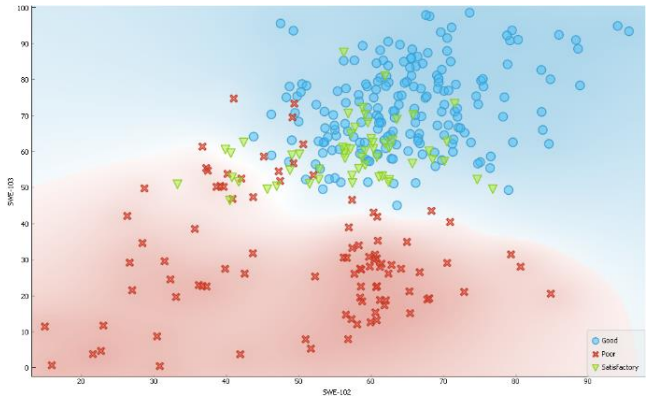

Fig. 1. Scatter plot demonstrating the relationship between SWE-102 and SWE-103 of Cohort 1

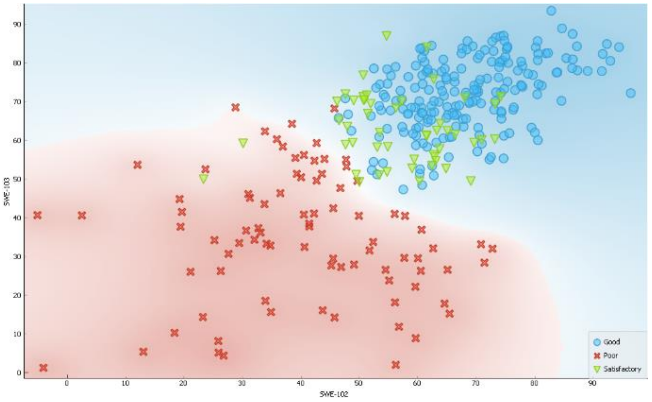

Fig. 2. Scatter plot demonstrating the relationship between SWE-102 and SWE-103 of Cohort 2

A heat map is created by grouping the graduates' belonging to both cohorts into three classes' w.r.t two programming courses to understand better, how they have evolved (see Fig. 3 and Fig. 4). The values of the class or label feature are shown at the left based on their tuples' columns, i.e., SWE-102 and SWE-103. Colours in a heat map visualization are used to indicate the values: the greater the value, the darker the colour shown. The change in colour of the columns in both cohorts' heat map visualizations tends to be redder, which corresponds to the graduates' who attained high marks in both courses. These courses progress from low to high attainments (from blue to red) as shown at the top of the visualizations, and the colour shift closely corresponds to the classes; this is specifically true for the graduates who are likely to have a promising, poor, or satisfactory career in the programming field based on their attainments in both programming courses. As a result, Research Question 2 is answered positively in this study because the programming courses and CGPA may be used to forecast career placement possibilities in computer programming with sufficient accuracy.

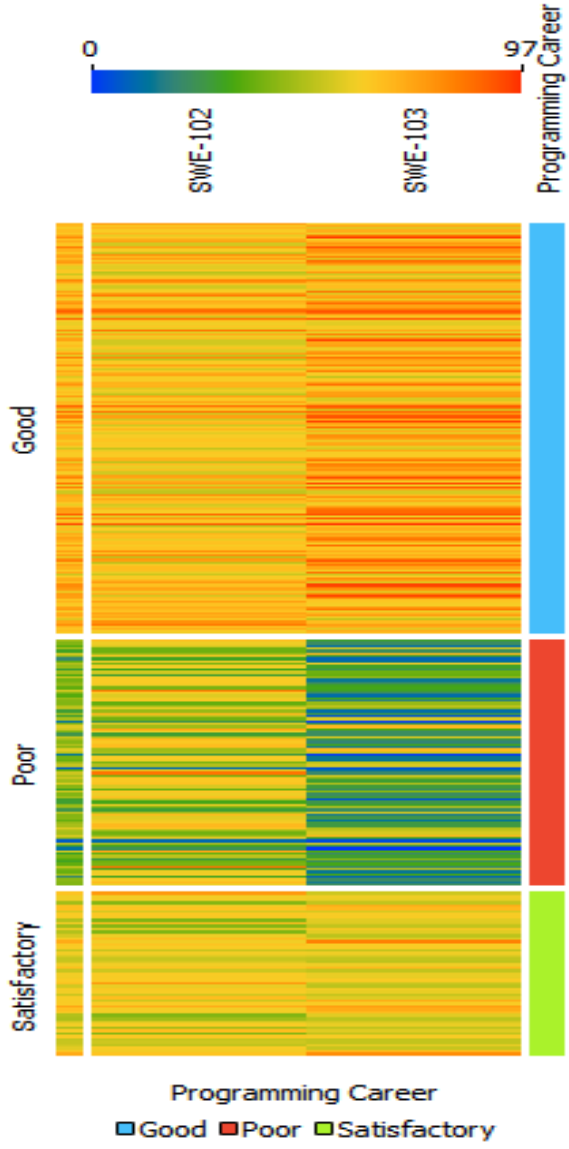

Fig. 3. Heat map of Cohort 1

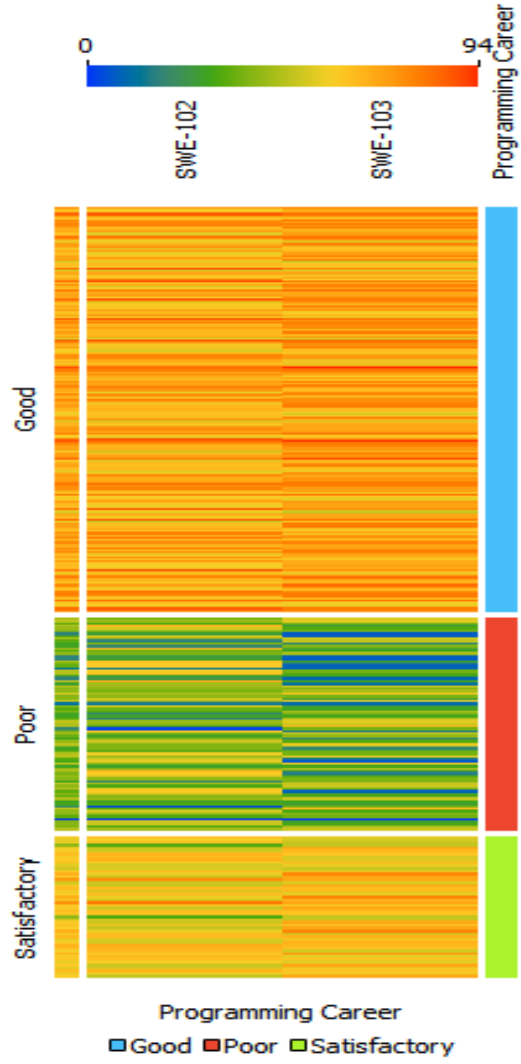

Fig. 4. Heat map of Cohort 2

C. Classification Tree Forecasts and Recommendations Using Family Demography

As illustrated in Fig. A3, the Classification Tree for cohort 1 is constructed with 77 nodes and 39 leaves around the feature Parent's Occupation, with the class 'Good' as the root-splitting criterion. The top professions of the graduate's parents are input to find out the influence of parents' occupation on graduates when selecting programming as a career. It is noteworthy when the parent's occupations are Management (3 out of 5 instances), Sales (20 out of 22 instances), Government (28 out of 42 instances), Engineering (13 out of 16 instances), and Retired (27 out of 34instances) are purely classified as 'Good.' At the same time, when the occupations are Construction (22 out of 24 instances), Pharmaceutical (6 out of 14 instances), Private (31 out of 54 instances), and Education / Teaching (5 out of 7 instances) are partially classified as 'Good' or ‘Poor’. Similarly, when the parent's occupations are, Banking

and Programming (5 out of 9 instances) are classified as 'Satisfactory,' and when the parent's occupations are General Business / Trade, (9 out of 16 instances) are classified as 'Poor'.

The next node of the Classification Tree is constructed around the feature Parent's Qualification. It is notable when the parent's qualifications are Doctoral Degree (21 out of 43 instances), Inter Pass (20 out of 33 instances), Some Education (22 out of 44 instances), Less than X class (26 out of 50 instances), Master's Degree (25 out of 48 instances), and Diploma / Certificate (10 out of 14 instances) are partially classified as 'Good' or 'Poor .'However, when the parent's qualification is Matric Pass (5 out of 8 instances) are only classified as 'Good', and when the parent's qualification is Bachelor's Degree (38 out of 60 instances) are partially classified as 'Good', 'Satisfactory', or 'Poor'.

The next node of the Classification Tree is constructed around the feature Parent's Income. It is notable when parents' incomes are High (24 out of 30 instances), Very Low (63 out of 88 instances), and Very High (59 out of 84 instances) and are partially classified as 'Good', 'Satisfactory', or 'Poor '. Moreover, when the parent's income is Medium (82 out of 131 instances) is partially classified as 'Good' or 'Poor', and when the parent's income is Low (23 out of 32 instances) is partially classified as 'Good' or 'Satisfactory.'

As illustrated in Fig. B3, the Classification Tree for cohort 2 is constructed with 89 nodes and 45 leaves around the feature Parent's Occupation, with the class 'Good' serving as the root-splitting criterion. It is noteworthy when the parent's occupations are Private (38 out of 49 instances), Banking (11 out of 13 instances), Sales (8 out of 8 instances), Construction (3 out of 4 instances), General Business / Trade (16 out of 25 instances), Education / Teaching (9 out of 15 instances), Retired (15 out of 25 instances), and Pharmaceutical (16 out of 30 instances) are completely classified as 'Good', while when the occupations are Government (23 out of 50 instances), Management (13 out of 23 instances), and Engineering (6 out of 8 instances) are partially classified as 'Good' or 'Poor'.

Here also, the next node of the Classification Tree is constructed around the feature Parent's Qualification. It is notable when parent's qualifications are Less than X class (22 out of 25 instances), Master's Degree (38 out of 59 instances), and Matric Pass (16 out of 25 instances) are partially classified as 'Good', 'Satisfactory', or 'Poor'. However, when the parent's qualification is a Bachelor's Degree (30 out of 51 instances) is simply classified as 'Good'. When the parent's qualification is Some Education (22 out of 31 instances) and Diploma / Certificate (27 out of 38 instances) are partially classified as 'Good' or 'Poor', and when the parent's qualification is Inter Pass (31 out of 43 instances) and Doctoral Degree (7 out of 8 instances) are partially classified as 'Good' or 'Satisfactory'.

The feature Parent's Income is the focus of the Classification Tree's next node. It is notable when parent's incomes are High (20 out of 35 instances), Very Low (36 out of 61 instances), Medium (77 out of 131 instances), Low (27 out of 43 instances), and Very High (36 out of 61 instances) are partially classified as 'Good 'or 'Poor'.

This recommends that when the parent occupations of the graduates belonging to both cohorts are Sales and Retired, they will likely have high chances of career placement probability in a programming field. Whereas, as far as the other occupations are concerned, the chances of career placement probability will likely be diversified. Interestingly, the chances of career placement probability for the graduates will likely be high when their parent's qualifications are Matric Pass for cohort1 and Bachelor's Degree for cohort 2, and for the rest of the parent's qualifications input, the chances of career placement probability will likely to be diversified. Similarly, the distinguished results are generated on the parent incomes of the graduates belonging to both cohorts. So the chances of career placement probability will likely be completely diversified here. Therefore, Research Question 3 is answered a trivial positively with digressions.

D. Classification Tree Forecasts and Recommendations Using Internship Experience

The Classification Tree for cohort 1 is formed with three nodes and two leaves around the feature Internship Experience, with the class 'Good' acting as the root-splitting criterion, as shown in Fig. A4. Interestingly, in both cases, when the graduates either have or have no internship experience, i.e. (45 out of 67 instances) and (82 out of 158 instances) respectively, are all classified as 'Good'.

Almost the same Classification Tree as shown in Fig. B4 is generated for cohort 2 only with the difference of the number of instances, i.e. (117 out of 194 instances) with no internship experience and (18 out of 31 instances) with having an internship experience are all classified as 'Good'.

This recommends that regardless the graduates have internship experience in computer programming or not, they will likely have a high chance of career placement probability in the programming field. Even though it is a matter of fact that an internship experience assists learners in making links between their traditional courses and the job. In addition, surprisingly the Classification Tree classifier did not depict the instances with the internship experience in a 'Satisfactory' and 'Poor' class. Hence, Research Question 4 is answered positively, since an internship experience, to some extent, may influence the career placement probabilities in the field of computer programming for graduates with sufficient accuracy.

E. Classification Tree Forecasts and Recommendations Using Gender

As shown in Fig. A5, the Classification Tree for cohort 1 has three nodes and two leaves centred on the feature Gender, with the class 'Good' serving as the root-splitting criterion. The output clearly shows that both males (94 out of 188 instances) and females (33 out of 37 instances) are classified as 'Good'.

For cohort 2, a nearly identical Classification Tree is constructed as shown in Fig. B5, except for the number of instances, which are all classed as 'Good', i.e. (89 out of 172 instances) of males and (47 out of 53 instances) of females.

This recommends that both male and female graduates will likely have high chances of career placement probability in the programming field. Here also surprisingly, the Classification Tree classifier did not depict the instances with the gender in a 'Satisfactory' and 'Poor' class. Therefore, Research Question 5 is answered positively.

5.3 Comparing Classifiers

Comparing classifiers' accuracies with all five techniques on both cohorts is depicted below (see Fig. 5). It shows the accuracy results of five classifiers that outperformed the benchmark in terms of accuracy. These findings indicate that using the values of pre-university grades, marks in programming courses, CGPA, gender, internship experience, and family demography of each cohort data may constitute to forecast the career placement probabilities in the programming field with sufficient accuracy. When compared to the other four classifiers, the Classification Tree classifier provides the best accuracy on the cohort 1 dataset, at 95%. K-Nearest Neighbour classifier, in contrast, has the lowest accuracy, i.e., 86%. Similarly, while evaluating the performance of five classifiers on the cohort 2 dataset, here also the Classification Tree classifier performed the best with 97% accuracy, and the Support Vector Machine classifier performed the lowest with 84% accuracy.

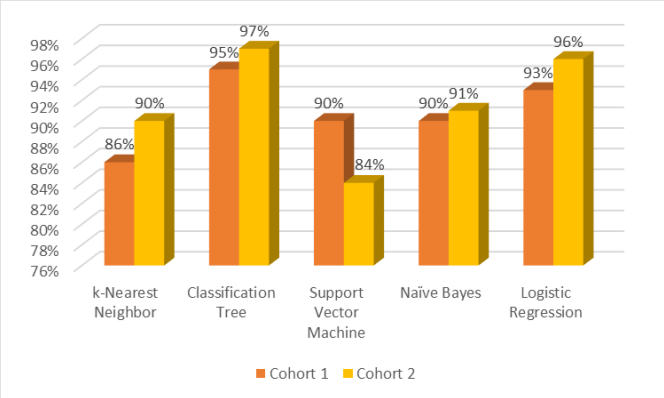

Fig. 5. Classifiers comparison with accuracies

Now the ROC (Receiver Operating Characteristics) analysis on each target class for both cohorts is presented in Appendices C (see Fig.s C1-C3) and D (see Fig.s D1-D3). The ROC curve is one of the most significant assessment indicators for evaluating the performance of any classification model. ROC curves are commonly used to depict the relationship/trade-off among both clinical sensitivity and specificity for each potential cut-off for a test or a set of tests in a graphical format. Furthermore, the area under the ROC curve provides insight into the value of applying the test(s) in question.

The best acceptable cut-off for a test is determined using ROC curves. The best cut-off has the lowest false positive rate and the highest true positive rate. The area under a ROC curve is used to compare the effectiveness of tests since it is a measure of the usefulness of a test in general, with a larger area indicating a more valuable test [33]. The ROC curve is a graph that includes the following information:

1 – Specificity = false positive fraction = $\frac{FP}{(FP+TN)}$ is shown on the x-axis.

The Sensitivity = true positive fraction = $\frac{TP}{(TP+FN)}$ is shown on the y-axis.

The Fig.s show that the dotted diagonal line in the middle represents the ROC curve of a random classifier and the colored ROC curves there correspond to the input classifiers. The test is more efficient if the ROC curve is closer to the upper left corner. The further away the ROC curve is from the random line, the greater the area under the curve and, therefore, the better the

classifier performs. Fig.s C1 and D1 show that the performance line is positioned in between a range of 0.9 to 1.0. Here the classifiers performed optimally with the target probability of 56% in cohort 1 and 60% in cohort 2 relating to class "Good".

Fig. C2 shows that the performance line is positioned in between a range of 0.4 to 0.5, where the classifiers are also performed optimally with the target probability of 16% for the class "Satisfactory" relating to cohort 1. Similarly, Fig. D2 shows that the performance line is positioned in between a range of 0.5 to 0.6, where the classifiers are also performed optimally with the target probability of 14% for the class "Satisfactory" relating to cohort 2. Fig. C3 shows that the performance line is positioned in between a range of 0.7 to 0.8, where the classifiers are too performed optimal with the target probability of 28% for the class "Poor" relating to cohort 1. Similarly, Fig. D3 shows that the performance line is positioned in between a range of 0.8 to 0.9, where the classifiers are likewise performed optimally with the target probability of 25% for the class "Poor" relating to cohort 2.

Further, Table 3 shows the resulting confusion matrices. A confusion matrix is a description of classification problem forecast results. The number/proportion of instances involving the actual and forecast class is reflected by the Confusion Matrix. The correct classes are represented by each row, whereas the forecast classes are represented by the columns. The first confusion matrix of the classifier "Classification Tree" is described to better comprehend these matrices. For instance, the number of instances of a particular class is mentioned in the rightmost column like 190, and 205 instances of the class "Good," 55 and 50 instances of the class "Satisfactory," and 95 and 85 instances of class "Poor" relating to both cohorts respectively are mentioned in the rightmost column, and the number of instances forecasts into each class is seen in the bottom row in each classifier confusion matrix. The diagonals of each matrix containing all correct forecasts are highlighted in green colour. Here, the four instances of Satisfactory are misclassified as Good, 11 instances of Poor are misclassified as Satisfactory, and three instances of Satisfactory are misclassified as Poor for cohort 1. Similarly, the two instances of Satisfactory are misclassified as Good, three instances of Poor are misclassified as Satisfactory, and four instances of Satisfactory are misclassified as Poor for cohort 2. The misclassified instances are highlighted in red colour. It is noteworthy the datasets used in this study are balanced datasets with an equal number of instances in each dataset. The maximum accuracy is found in well-represented classes like 'Good'.

Table 2

Confusion Matrices

Classification Tree		Cohort 1 Forecast				Cohort 2 Forecast			
		Good	Satisfactory	Poor	Σ	Good	Satisfactory	Poor	Σ
Actual	Good	190	0	0	190	205	0	0	205
	Satisfactory	4	48	3	55	2	44	4	50
	Poor	0	11	84	95	0	3	82	85
Σ		194	59	87	340	207	47	86	340
Naïve Bayes		Forecast				Forecast			
		Good	Satisfactory	Poor	Σ	Good	Satisfactory	Poor	Σ
Actual	Good	176	14	0	190	192	12	1	205
	Satisfactory	0	54	1	55	0	41	9	50
	Poor	0	18	77	95	0	9	76	85
Σ		176	86	78	340	192	62	86	340
Support Vector Machine		Forecast				Forecast			
		Good	Satisfactory	Poor	Σ	Good	Satisfactory	Poor	Σ
Actual	Good	187	0	3	190	202	0	3	205
	Satisfactory	17	31	7	55	20	7	23	50
	Poor	5	2	88	95	7	3	75	85
Σ		209	33	98	340	229	10	101	340
k-Nearest Neighbor		Forecast				Forecast			
		Good	Satisfactory	Poor	Σ	Good	Satisfactory	Poor	Σ
Actual	Good	184	6	0	190	198	7	0	205

	Satisfactory	22	26	7	55	23	25	2	50
	Poor	8	6	81	95	1	0	84	85
Σ		214	38	88	340	222	32	86	340

Logistic Regression		Forecast Good	Satisfactory	Poor	Σ	Forecast Good	Satisfactory	Poor	Σ
Actual	Good	190	0	0	190	205	0	0	205
	Satisfactory	8	47	0	55	2	44	4	50
	Poor	0	17	78	95	0	9	76	85
Σ		198	64	78	340	207	53	80	340

6. Conclusion and Future Directions

For anyone's success, choosing the right career is crucial. To forecast the career placement probabilities specifically in a programming field, there is a need to undertake a lot of historical data analysis and experience-based evaluations of graduates' information. This research aims to bridge the gap among graduates' experimental classifications as 'Good,' 'Satisfactory,' or 'Poor' for a career choice in a programming field using existing data mining classification algorithms. In this regard, the present study is novel by addressing five research questions with the goal of giving prospective students the information that might guide them to seek how likely they will have career placement probabilities in a programming field.

The first question involves forecasting students' career placement probabilities based on their pre-university grades. Incorporating pre-university grades in this study is essential because the initial career direction of any individual begins at the early school or college-level studies. The results reveal that by using pre-university grades, it is possible to forecast career placement probabilities in computer programming with sufficient accuracy. The second question is to identify the attainments in programming courses and CGPA that may be helpful indicators for the successful career placement probabilities in the programming field for graduates. To know better career placement probabilities in the programming field, considering CGPA and the performance in programming courses may be essential to determine what type of student you were — whether you were a hardworking, motivated student or a laggard who was not excelling in their education. With the help of data mining techniques, both programming courses attainments and CGPA have been presented in evidence that can serve as valuable indicators with sufficient accuracy.

The third question deals with forecasting students' career placement probabilities based on their family demography, including their parent's occupation, qualifications, and income. According to previous research, family background has been discovered to fundamentally affect the propagation of values, including appropriate career choice, ambition, career orientation, and independence. The results indicate that it is possible to forecast career placement probabilities using a graduate's family background with sufficient accuracy but with digressions. The fourth question focuses on forecasting the impact of internship experience on the career placement probability in the programming field. The classifier results with sufficient accuracy show that whether or not graduates have had internship experience in computer programming, they are likely to have a good chance of securing a job in the programming field. The fifth question encompasses forecasting gender influence on career aspirations in the programming field. The results indicate that male and female graduates will have a reasonable probability of career advancement in the programming field with sufficient accuracy.

Thus, the present study findings recommend that by employing pre-university grades, programming course grades, CGPA, gender, internship experience, and family demographic data from each cohort, it is possible to forecast career placement possibilities in the programming field with sufficient accuracy using data mining techniques. The Classification Tree outperformed the five classifiers by achieving the highest accuracy on both cohorts' data analysis. However, several limitations remain, which must be addressed in future research. The course information, for example, is so limited that just only two courses have been considered. The limited set of data affects the accuracy of experimental outcomes. Furthermore, some other data mining techniques or heuristic algorithms can be used to increase the efficiency and accuracy of the experimental outcomes.

APPENDIX A Cohort 1 Classification Trees

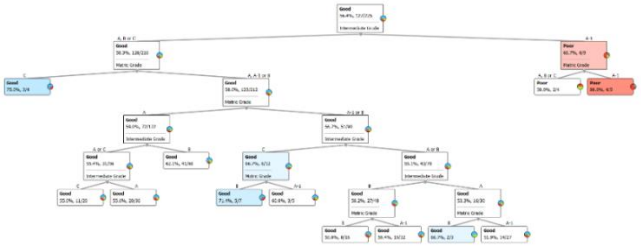

Fig. A1. Classification Tree with pre-university grades

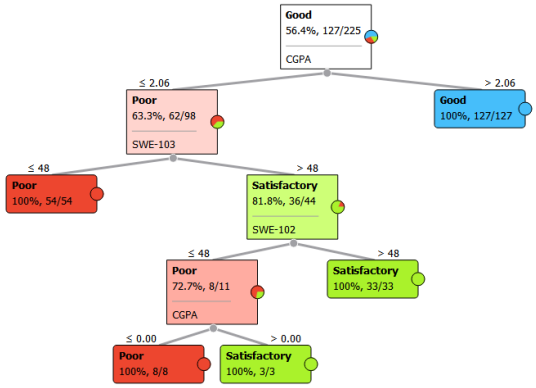

Fig. A2. Classification Tree with programming courses and CGPA

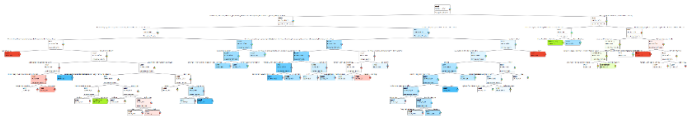

Fig. A3. Classification Tree with family demography

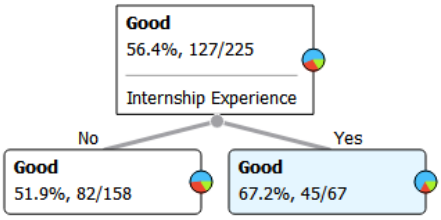

Fig. A4. Classification Tree with internship experience

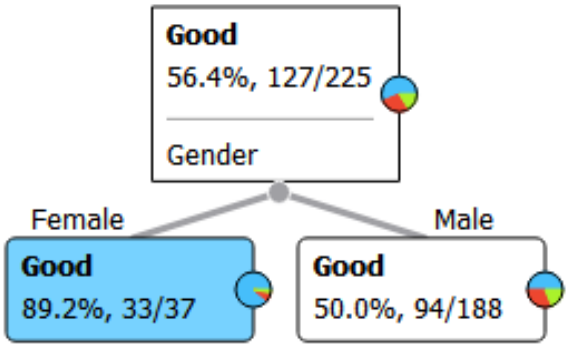

Fig. A5. Classification Tree with gender

APPENDIX B Cohort 2 Classification Trees

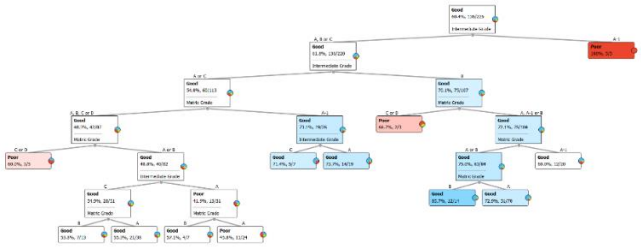

Fig. B1. Classification Tree with pre-university grades

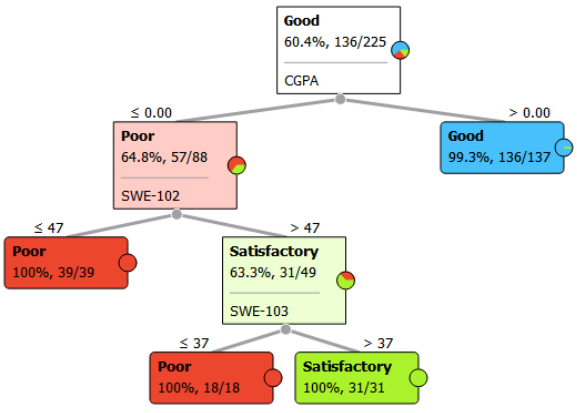

Fig. B2. Classification Tree with programming courses and CGPA

Fig. B3. Classification Tree with family demography

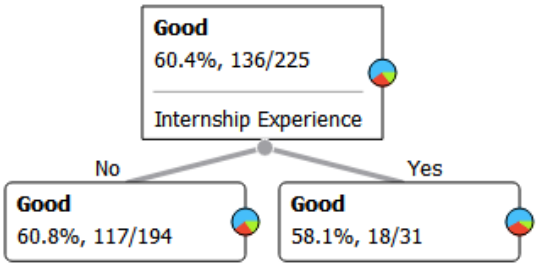

Fig. B4. Classification Tree with internship experience

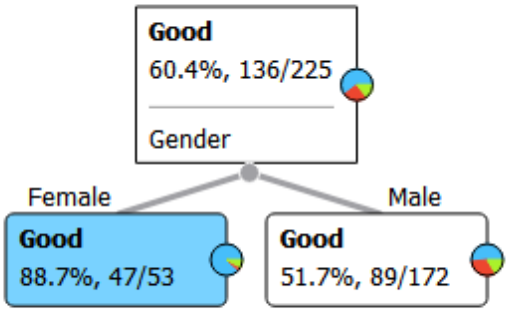

Fig. B5. Classification Tree with gender

APPENDIX C Cohort 1 ROC Analysis

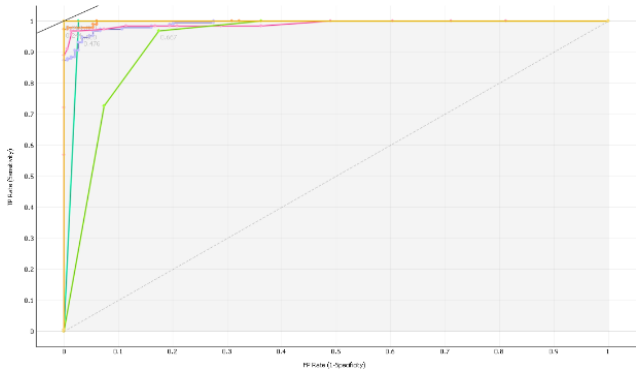

Fig. C1. ROC curve analysis relating to class "Good"

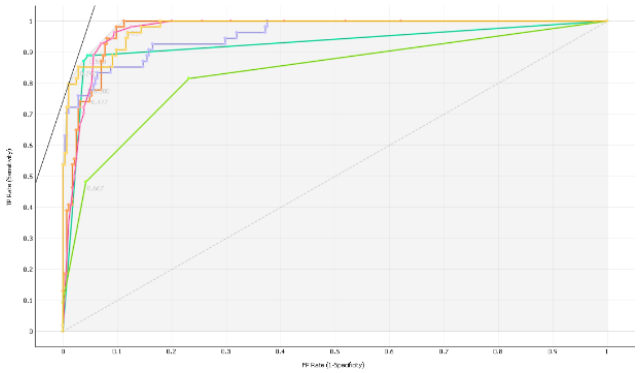

Fig. C2. ROC curve analysis relating to class "Satisfactory"

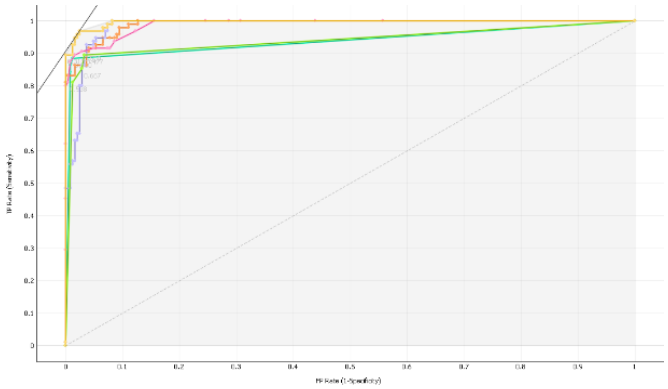

Fig. C3. ROC curve analysis relating to class "Poor"

APPENDIX D Cohort 2 ROC Analysis

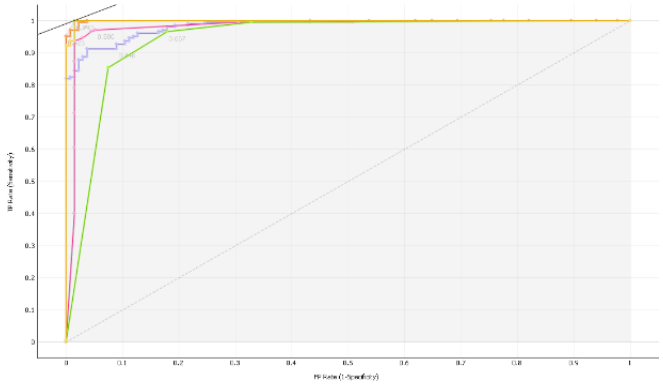

Fig. D1. ROC curve analysis relating to class "Good"

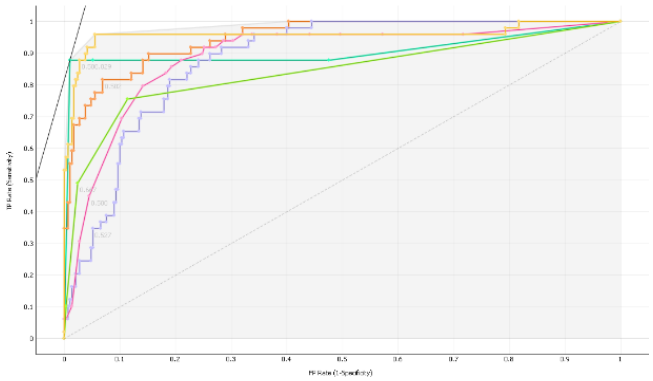

Fig. D2. ROC curve analysis relating to class "Satisfactory"

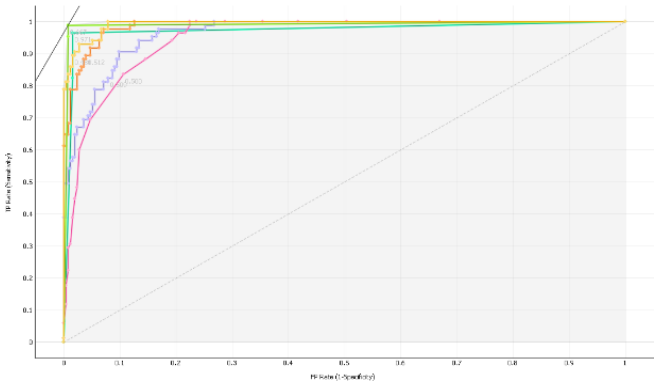

Fig. D3. ROC curve analysis relating to class "Poor"

7. Acknowledgement

The authors wish to acknowledge the support of providing data of Software Engineering graduates for this research by the management of Sir Syed University of Engineering and Technology, Karachi.

8. References

[1] J. Britto, S. Prabhu, A. Gawali, and Y. Jadhav, "A machine learning based approach for recommending courses at graduate level", Proceedings of the 2nd International Conference on Smart Systems and Inventive Technology, ICSSIT 2019, Icssit, pp. 117–121, 2019.

[2] A. Broström, G. Buenstorf, and M. McKelvey, "The knowledge economy, innovation and the new challenges to universities: introduction to the special issue", Innovation: Organization and Management, vol. 23, no. 2, pp. 145–162, 2021.

[3] S. Thiennoi, N. Chumuang, and C. Siladech, "An efficiency comparison for predicting of educational achievement Based on LMT", Proceedings - 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing, 2019.

[4] M. K. Omar, M. D. K. Zaman, and M. Hafiz Aziz, "Factors influencing career choice among final semester undergraduate students of a business management faculty in a malaysian public university", International Journal of Academic Research in Progressive Education and Development, vol. 10, no. 2, pp. 361–373, 2021.

[5] H. Yu and Z. Q. Zhang, "The application of data mining technology in employment analysis of university graduates", Proceedings - 17th IEEE/ACIS International Conference on Computer and Information Science, pp. 846–849, 2018.

[6] L. Zhang, X. Tan, S. Zhang, and W. Zhang, "Association rule mining for career choices among fresh graduates", Applied and Computational Mathematics, vol. 8, no. 2, p. 37, 2019.

[7] A. Bradley, M. Quigley, and K. Bailey, "How well are students engaging with the careers services at university?", Studies in Higher Education, vol. 46, no. 4. pp. 663–676, 2021.

[8] D. Smith and A. Ali, "Analyzing computer programming job trend using web data mining", Issues in Informing Science and Information Technology, vol. 11, pp. 203–214, 2014.

[9] M. C. Mihaescu and P. S. Popescu, "Review on publicly available datasets for educational data mining", Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 11, no. 3, pp. 1–16, 2021.

[10] A. Triayudi and W. O. Widyarto, "Educational data mining analysis using classification techniques", Journal of Physics: Conference Series, vol. 1933, no. 1, 2021.

[11] R. Asif, A. Merceron, S. A. Ali, and N. G. Haider, "Analyzing undergraduate students' performance using educational data mining", Computers and Education, vol. 113, pp. 177–194, 2017.

[12] K. Mahboob, S. A. Ali, D. U. R. Khan, and F. Ali, "A comparative study of engineering students pedagogical progress", International Journal of Advanced Computer Science and Applications, vol. 9, no. 6, pp. 323–331, 2018.

[13] F. Siddiq, P. Gochyyev, and O. Valls, "The role of engagement and academic behavioral skills on young students' academic performance—A validation across four countries", Studies in

Educational Evaluation, vol. 66, no. March, p. 100880, 2020.

[14] M. A. Al-Hagery, M. A. Alzaid, T. S. Alharbi, and M. A. Alhanaya, "Data Mining Methods for Detecting the Most Significant Factors Affecting Students' Performance," International Journal of Information Technology and Computer Science, vol. 12, no. 5, pp. 1–13, 2020.

[15] T. Le Mai, M. T. Chung, V. T. Le, and N. Thoai, "From Transcripts to Insights for Recommending the Curriculum to University Students", SN Computer Science, vol. 1, no. 6, pp. 27–29, 2020.

[16] R. Ade and P. R. Deshmukh, "An incremental ensemble of classifiers as a technique for prediction of student's career choice", 1st International Conference on Networks and Soft Computing, ICNSC 2014 - Proceedings, pp. 384–387, 2014.

[17] S. Elayidom, S. M. Idikkula, J. Alexander, and A. Ojha, "Applying data mining techniques for placement chance prediction", ACT 2009 - International Conference on Advances in Computing, Control and Telecommunication Technologies, pp. 669–671, 2009.

[18] M. Nie, Z. Xiong, R. Zhong, W. Deng, and G. Yang, "Career choice prediction based on campus big data-mining the potential behavior of college students", Applied Sciences (Switzerland), vol. 10, no. 8, 2020.

[19] R. H. Rangnekar, K. P. Suratwala, S. Krishna, and S. Dhage, "Career Prediction Model Using Data Mining and Linear Classification", Proceedings - 2018 4th International Conference on Computing, Communication Control and Automation, ICCUBEA 2018, 2018.

[20] P. D. Dusane, N. V. Bhosale, V. A. Avhad, and P. K. Naikwade, "Recommendation System for Career Path using Data Mining Approaches," International Journal of Scientific Research and Engineering Trends, vol. 6, no. 2, pp. 587–589, 2020.

[21] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A brief review of nearest neighbor algorithm for learning and classification", 2019 International Conference on Intelligent Computing and Control Systems, Iciccs, pp. 1255–1260, 2019.

[22] H. H. Patel and P. Prajapati, "Study and analysis of decision tree based classification algorithms", International Journal of Computer Sciences and Engineering, vol. 6, no. 10, pp. 74–78, 2018.

[23] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends", Neurocomputing, 2020.

[24] S. Chen, G. I. Webb, L. Liu, and X. Ma, "A novel selective naïve Bayes algorithm", Knowledge-Based Systems, vol. 192, p. 105361, 2020.

[25] C. Y. J. Peng, K. L. Lee, and G. M. Ingersoll, "An introduction to logistic regression analysis and reporting", Journal of Educational Research, vol. 96, no. 1, pp. 3–14, 2002.

[26] U. of L. Bioinformatics Laboratory, "Data mining", Orange Data Mining - Data Mining. [Online]. Available: https://orangedatamining.com/. [Accessed: 05-June-2022].

[27] L. I. Jimenez-Raygoza, A. S. Medina-Vazquez, and G. Perez-Torres, "Proposal of a computer system for vocational guidance with data mining," IEEE International Conference on Engineering Veracruz, 2019, pp. 11–15, 2019.

[28] K. Mahboob, S. A. Ali, and U. e. Laila, "Investigating learning outcomes in engineering education with data mining", Computer Applications in Engineering Education, vol. 28, no. 6, pp. 1652–1670, Nov. 2020.

[29] A. S. Kuttattu, G. S. Gokul, H. Prasad, J. Murali, and L. S. Nair, "Analysing the learning style of an individual and suggesting field of study using Machine Learning techniques", Proceedings of the 4th International Conference on Communication and Electronics Systems,

ICCES 2019, Icces, pp. 1671–1675, 2019.

[30] D. Kurniadi, E. Abdurachman, H. L. H. S. Warnars, and W. Suparta, “A proposed framework in an intelligent recommender system for the college student”, Journal of Physics: Conference Series, vol. 1402, no. 6, 2019.

[31] L. S. Katore, B. S. Ratnaparkhi, and J. S. Umale, “Novel professional career prediction and recommendation method for individual through analytics on personal traits using C4.5 algorithm”, Global Conference on Communication Technologies, 2015, pp. 503–506, 2015.

[32] R. Katarya, V. Gangwar, and I. Jaisia, “A Study on different data mining classifiers”, 22018 International Conference on Computer Communication and Informatics, 2018, pp. 2–7, 2018.

[33] E. Zamanzade and X. Wang, “Estimating the area under a receiver operating characteristic curve using partially ordered sets”, International Journal of Biostatistics, vol. 17, no. 1, pp. 139–152, 2021.

[34] P. Cash, O. Isaksson, A. Maier, and J. Summers, “Sampling in design research: Eight key considerations”, Sampling in design research: Eight key considerations. Design Studies, vol. 78, p. 101077, 2022.

[35] Q. Zhou, F. Liao, C. Chen, and L. Ge, “Job recommendation algorithm for graduates based on personalized preference”, CCF Transactions on Pervasive Computing and Interaction, vol. 1, no. 4, pp. 260–274, 2019.

