ISSN No:-2456-2165

Placement Prediction using Various Machine Learning Models and their Efficiency Comparison

Irene Treesa Jose [1], Daibin Raju[2], Jeebu Abraham Aniyankunju[3], Joel James[4], Mereen Thomas Vadakkel [5]
[1, 2, 3, 4] B.Tech , Department of CSE, SJCET Palai, Kerala, India
[5]Assistant Professor, Department of CSE, SJCET Palai, Kerala, India

Abstract:- A placement predictor is to be designed to calculate the possibility of a student being placed in a company, subject to the criterion of the company. The placement predictor takes many parameters which can be used to assess the skill level of the student. While some parameters are taken from the university level, others are obtained from tests conducted in the placement management system itself. Combining these data points, the predictor is to accurately predict if the student will or will not be placed in a company. Data from past students are used for training the predictor.

But the problem was to find a suitable classification algorithm that could do the job with maximum accuracy for our data set. Different algorithms have different accuracy depending on the type of problem it has to solve and the data set it has to work with. So, we decided to select four algorithms, namely KNN, SVM, Logistic Regression and Random Forest and to compare the accuracy levels of each of these algorithms, with respect to our problem and data set. The result of this test would help us in determining which algorithm to use while implementing our predictor in the placement management system.

For this, we trained each of the algorithms with the data set that we acquired and tested it against some test data to find the accuracy of the algorithms. For each algorithm, we can easily obtain the True Positive, True Negative, False Positive and False Negative. With these four values, it was a matter of finding the accuracy using the accuracy equation.

Keywords:- *Classifications, Dataset, Machine learning, Placement.*

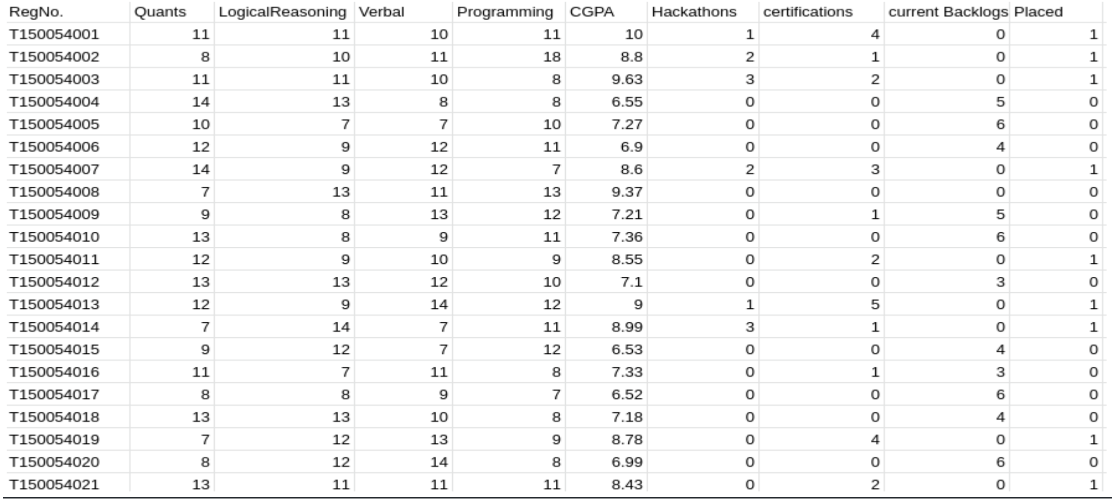
I. INTRODUCTION

We aim to develop a placement predictor as a part of making a placement management system at college level which predicts the probability of students getting placed and helps in uplifting their skills before the recruitment process starts. We are using machine learning for the placement prediction. We consider K-nearest neighbour (KNN), Support Vector Machine(SVM), Logistic Regression, Random Forest to classify students into appropriate clusters and the result would help them in improving their profile.And accuracy of respected algorithms are noted and With the comparison of various machine learning techniques, this would help both recruiters as well as students during placements and related activities.

A. Prediction system

In this paper we use machine learning techniques to predict the placement status of students based on a dataset. The parameters in the dataset which are considered for the prediction are Quantitative scores, LogicalReasoning scores, Verbal scores, Programming scores, CGPA, No. of hackathons attended, No. of certifications and current backlogs number. The placement prediction is done by machine learning using Logical Regression, Random Forest, KNN, SVM.

B. Sample Dataset



RegNo.	Quants	LogicalReasoning	Verbal	Programming	CGPA	Hackathons	certifications	current Backlogs	Placed
T150054001	11	11	10	11	10	1	4	0	1
T150054002	8	10	11	18	8.8	2	1	0	1
T150054003	11	11	10	8	9.63	3	2	0	1
T150054004	14	13	8	8	6.55	0	0	5	0
T150054005	10	7	7	10	7.27	0	0	6	0
T150054006	12	9	12	11	6.9	0	0	4	0
T150054007	14	9	12	7	8.6	2	3	0	1
T150054008	7	13	11	13	9.37	0	0	0	0
T150054009	9	8	13	12	7.21	0	1	5	0
T150054010	13	8	9	11	7.36	0	0	6	0
T150054011	12	9	10	9	8.55	0	2	0	1
T150054012	13	13	12	10	7.1	0	0	3	0
T150054013	12	9	14	12	9	1	5	0	1
T150054014	7	14	7	11	8.99	3	1	0	1
T150054015	9	12	7	12	6.53	0	0	4	0
T150054016	11	7	11	8	7.33	0	1	3	0
T150054017	8	8	9	7	6.52	0	0	6	0
T150054018	13	13	10	8	7.18	0	0	4	0
T150054019	7	12	13	9	8.78	0	4	0	1
T150054020	8	12	14	8	6.99	0	0	6	0
T150054021	13	11	11	11	8.43	0	2	0	1

Table 1:- Dataset used for Prediction and Analysis

C. Architecture Diagram

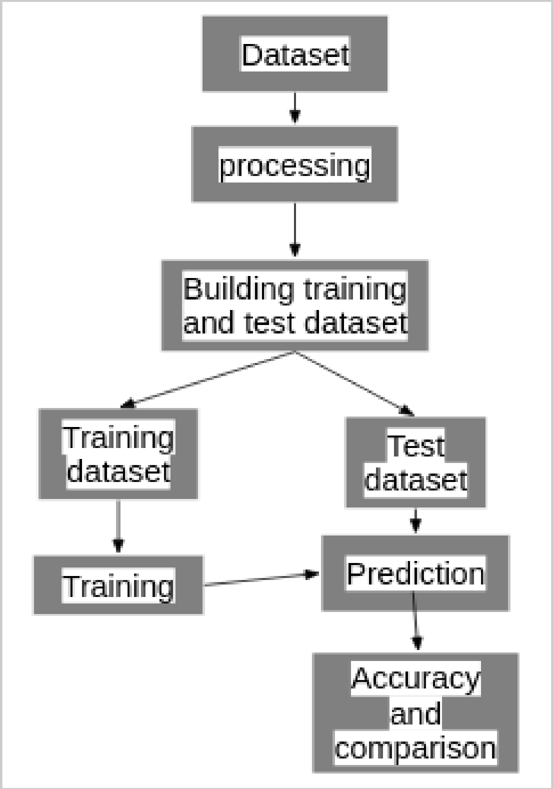


Fig 1:- Architecture for Data Processing , Model Training, Prediction and Accuracy check.

The data frame for the machine learning algorithm is created using pandas library based on the above sample dataset. The handling of null data fields is carried out by dataset.fillna(method='ffill'). We use sklearn which is an efficient tool for predictive analysis. And we import train_test_split in sklearn for creating training and test sets from the dataset. Standardisation is done by standardscaler in sklearn.preprocessing. Based on the respective algorithm it predicts the placement of each student and accuracy can be viewed from the confusion matrix.

II. METHODOLOGY

Machine Learning can be done using many available algorithms. Each algorithm has its own set of merits and demerits, these can change for the type of dataset being used or for the type of problem we have at hand. Given below are some of the algorithms that we used. A detailed description accompanies each algorithm.

A. KNN

KNN stands for k-nearest neighbors. This is a simple algorithm that can be used to solve classification and regression type problems. It is a supervised machine learning algorithm, meaning labels are used.

The basic working of this algorithm revolves around the concept that similar things are always in close proximity within each other. So, for this algorithm to provide any fruitful results, this is an assumption that is taken. Similarity in KNN is expressed using distance, closeness or proximity. A mathematical approach is taken for distance, which is usually the Euclidean Distance as it is the common and familiar choice.

The algorithm:

- Load the data to be used.
- Initialize the value of K to a chosen number of neighbors
- For each entry in the data:
  The distance between the current example and query example is to be calculated from the data. The index of the example and its distance is added to an ordered collection.
- Sort this ordered collection in ascending order by the distances
- Pick the first K entries from this sorted collection
- The labels of the selected K entries are taken.
- Now, if it is regression, calculate and return the mean of the K labels. But if it is classification, calculate and return the mode of the K labels

This is a simple algorithm to implement KNN.

However the error of the algorithm depends on the value selected for K. So, to find the K that is best suited for the data given, it is advised to run the algorithm many times with different values of K, so that the K with the least error can be found.

➢ *Advantages:*

- This is a fairly simple and easy-to-implement algorithm
- Building a model, tuning several parameters or making additional assumptions are not required
- This is a versatile algorithm, being able to be used in regression, classification and even search problems.

➢ *Disadvantages:*

- The algorithm becomes significantly slower as the number of examples and/or predictors/independent variables increases.

The disadvantage that KNN provides makes it an impractical choice for use where predictions need to be made more rapidly. However, provided that one has enough computational power at their disposal, then KNN can be used in problems where similar items have to be identified.

B. *SVM*

SVM stands for Support Vector Machine. It is also a supervised machine learning algorithm that can be used for both classification and regression problems. However, it is mostly used for classification problems.

A point in the n-dimensional space is a data item where the value of each feature is the value of a particular coordinate. Here, n is the number of features you have. After plotting the data item, we perform classification by finding the hyper-plane that differentiates the two classes very well.

Now the problem lies in finding which hyper-plane to be chosen such that it is the right one.

Scikit-learn is a library in Python which can be used to implement various machine learning algorithms and SVM too can be used using the scikit-learn library.

➢ *Advantages:*

- This algorithm performs best when there is a clear margin of separation
- Effective in high dimensional spaces
- If the number of dimensions is greater than the number of samples, the algorithm would be able to perform better
- It is memory efficient

➢ *Disadvantages:*

- Performance is affected when large data sets are used as the required training time is more.
- Performance is also affected when the data set has too much noise
- SVM doesn’t directly provide probability estimates, rather a computationally intensive five-fold cross-validation is required.

C. *Logistic Regression*

Logistic regression is a classification technique and it is very good for binary classification. It's decision boundary which is generally linear derived based on probability interpretation. The results are in a nonlinear optimization problem for parameter estimation. Parameters can be estimated by maximising the expression using any nonlinear optimization solver.

The goal of this technique is given a new data point, and predict the class from which the data point is likely to have originated. Input features can be quantitative or qualitative.

Instead of a hyperplane or straight line, the logistic regression uses the logistic function to obtain the output of a linear equation between 0 and 1.

The function is defined as logistic(x)=1/(1+exp(-x))

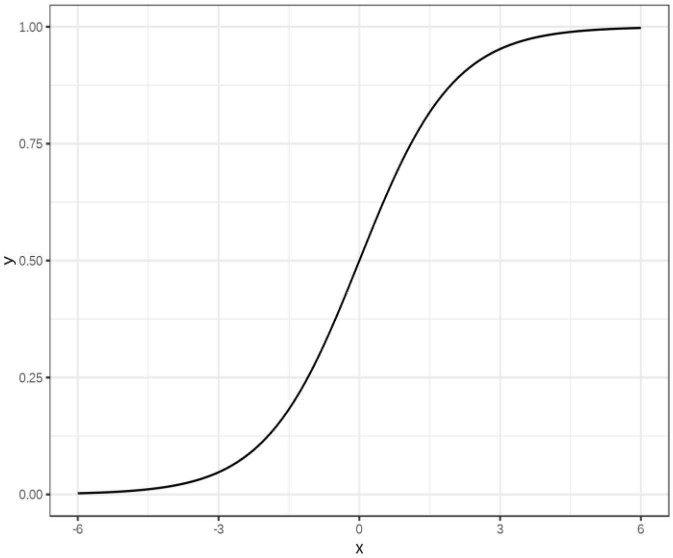


Fig 2:- Logistic regression

➢ *Advantages*

- Logistic Regression is good for linearly separable dataset.
- It is efficient to train and easy to interpret and implement.
- It not only gives a measure of how relevant a predictor is, but also its direction of association.
- Less prone to overfitting.

➢ *Disadvantages*

- It is useful only for predicting discrete functions.
- It should not be used If the No. of observations in the dataset are lesser than the number of features.
- Assumption of linearity between the independent and dependent variables.

D. Random Forest

We have a plethora of classification algorithms at our disposal, including, but not limited to, SVM, Logistic regression, decision trees and Naive Bayes classifier, just to name a few. But, in the hierarchy of classifiers, the Random Forest Classifier sits near the top. The random forest classifier is a group of individual decision trees and so, we shall look into how decision trees work.

It is basically a flowchart-like structure in which each node excluding the leaf node is a test on a feature (i.e, what will be the outcome if some activity, such as flipping a coin, is done), leaf nodes are used to represent the class label (the decision taken after all features are computed) and branches represent the conjunctions of features that lead to those class labels. The classification rules of a decision tree are the paths from the root node to the leaf node.

So then, now let us look into random forest classifiers. As mentioned earlier, it is a collection of decision trees. The basic idea behind random forest is “the wisdom of the crowds”. It is a powerful concept wherein a large number of uncorrelated models, or in this case trees, operating as a group, would provide a much more solid output than any of the constituent models.

So, in a random forest, each individual tree with different properties and classification rules would try to find an appropriate class label for the problem. Each tree would give out its own answer. A voting is done within the random forest to see which class label received the most votes. The class label with the most votes would be considered the final class label for the problem. This provides a more accurate model for class label prediction.

➢ *Advantages:*

- It can balance errors in data sets where classes are imbalanced
- Large data sets with higher dimensionality can be handled
- It can handle thousands of input variables and could identify the most significant variables and as such, it is a good dimensionality reduction method

➢ *Disadvantages:*

- It does more good of a job for classification problems rather than regression problems as it finds it harder to produce continuous values rather than discrete ones

III. RELATED WORKS

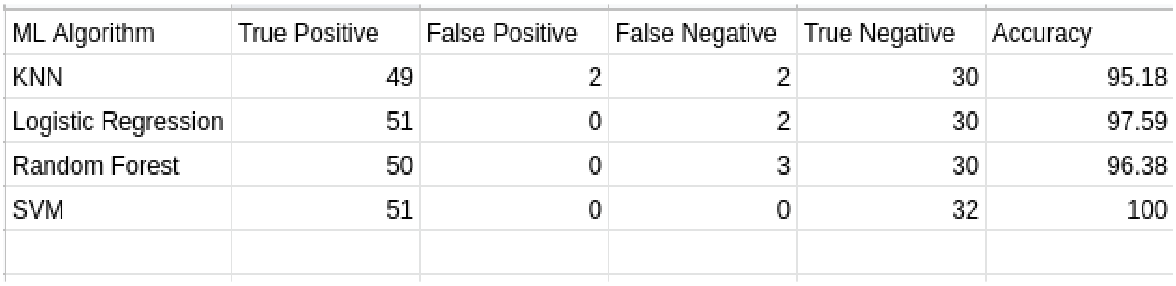
Shreyas Harinath, Aksha Prasad, Suma H and Suraksha A[1] made a study on Student Placement Prediction using Machine learning models which included Naive Bayes Classifier and K-Nearest Neighbors [KNN] algorithm.The study highlighted the efficiency of the algorithms which used the historical data of the previously passed students and aimed to predict the placement probabilities of the current students.

Senthil Kumar Thangavel, Divya Bharathi P and Abhijith Shankar[2] conducted a study to predict the placement chances of the students using Decision Tree Learning, SCI-Kit learning which used two attributes as dataset namely CGPA and arrears which resulted in more time consuming for prediction and being not efficient.

Performing Educational Data Mining and using machine learning algorithms and their efficiencies which vary from dataset to dataset[3] have been very useful to the Institutions which helps them to spot the potential students and provide the necessary support to improve in technical and social skills. This will give a clear idea about the areas needed to be concentrated for placement drives of various campus recruiters and what they actually expect more from the students nowadays.Such a study will help the faculties of the institution to train the students accordingly and thus strengthening the placement department of their institutions.

IV. RESULT

The final result of performing various machine learning algorithms are mentioned in the table below. We considered KNN, Logistic Regression, Random Forest and SVM for the analysis. We trained and predicted the placement status of students based on the same dataset and found the True Positive, False Positive, False Negative, True Negative and accuracy of each algorithm. And it is tabulated in the table below.



ML Algorithm	True Positive	False Positive	False Negative	True Negative	Accuracy
KNN	49	2	2	30	95.18
Logistic Regression	51	0	2	30	97.59
Random Forest	50	0	3	30	96.38
SVM	51	0	0	32	100

Table 2:- Accuracy of algorithms

V. CONCLUSION

Placement prediction system is a system which predicts the placement status of final year B-Tech students. For data analysis and prediction different machine learning algorithms are used in the python environment. We analyse the accuracy of different algorithms and it is shown in the above table. It is clear that SVM gives an accuracy of 100. Logistic Regression is also good which gives an accuracy of 97.59 based on the given dataset. The accuracy of Machine learning algorithms may differ according to the dataset. From the result from our analysis it is clear that SVM, Logistic Regression, Random Forest, KNN are good for binary classification problems since they all give accuracy of above 95. Some recruiters consider GATE scores and history of backlogs which we didn't include in our dataset. In such rare cases these results may change.

REFERENCES

[1]. Shreyas Harinath, Aksha Prasad, Suma H and Suraksha A. **Student Placement Prediction using Machine Learning,** International Research Journal of Engineering and Technology (IRJIET) Volume : 06 Issue: 04 April 2019

[2]. Senthil Kumar Thangavel, Divya Bharathi P and Abhijith Shankar. **Student Placement Analyzer: A recommendation System Using Machine Learning**, International Conference on advanced computing and Communication systems (ICACCS-2017), Jan 06-07,2017, Coimbatore, INDIA.

[3]. K. Sreenivasa Rao, N. Swapna, P. Praveen Kumar **Educational data mining for student placement prediction using machine Learning algorithms** Research Paper, International Research Journal of Engineering and Technology (IRJIET) 2018