Post Graduate Admission Prediction System

Dr. Bindiya M K[1], Abhijna S[2], Abhishek Rawat[3], Anushri N R[4], Indudhar L Gowda[5]

[1]*Associate Professor, Department of Computer Science and Engineering, SJB Institute of Technology*
[2,3,4,5]*Student, Department of Computer Science and Engineering, SJB Institute of Technology*

---***---

Abstract - *Each year, a huge number of students apply to post-graduate programs all over the world in order to advance their careers. In this project, we focus on the students applying for Masters programs in universities abroad, particularly the universities in the United States of America. Here, we present a new university admission prediction system using machine learning algorithms by recognizing the factors that affect the likelihood of admission. We use several machine learning algorithms to statistically analyze the admission predictability of such factors. The objective is to predict the list of universities that a student might get into rather than the chance of admit.*

Key Words: Post-graduate admission prediction, Machine Learning, Masters programs, PCA, MLR, SVM

1. INTRODUCTION

Applying to universities abroad is often difficult. Students find it hard to estimate exactly where they stand. It involves many steps and procedures to follow. Researching the universities and programs is itself an arduous and lengthy task. Choosing the right universities to apply to is definitely a hurdle students have to face. Many students apply for the universities in which they have a slim chance of acceptance. University applications alone can cost hundreds of dollars. This can take a toll on the finances of students coming from a poor economic background. Students have to throw away lots of hard-earned money for nothing if they get rejected by these universities.

In this project, we will be using the admission_predict dataset in CSV format to predict the universities that a student might get into based on several academic performance measurements.

Parameters that help predict the universities:

1. GRE Scores (out of 340)
2. TOEFL Scores (out of 120)
3. University Rating (out of 5)
4. Statement of Purpose and Letter of Recommendation Strength (out of 5)
5. Undergraduate GPA (out of 10)
6. Research Experience (either 0 or 1)

This project will be implemented using several machine learning algorithms, Python programming language and Flask web framework. To yield the most accurate result, we will be going through several steps such as data pre-processing, exploratory data analysis, feature selection, cross validation, model selection, training the model and so on.

2. LITERATURE SURVEY

A substantial number of research programs have been carried out on subjects related to university admissions. Each one has used datasets from various sources and is specific to a certain course or university. Hence, the prediction may not be accurate for every student.

Previous studies done in this area evaluated the chance of student admission into respective universities primarily based on a single parameter – the GRE score. The downside is that they only considered the GRE score and left out all the other factors that might contribute to admission prediction.

Our base research paper has considered all these parameters, but the machine learning model has low accuracy. Another con is that the dataset used is specific to Indian students and is not generalized.

The main drawback of all the previous research carried out is that it only predicts the chance of admission of a student. A better system would predict the universities that a student might get into, rather than the chance of admission. In this way, a student would be able to apply to universities where they have a better chance at getting admission.

3. METHODOLOGY

Problem Understanding: We first need to understand the problems faced by students during the university application process and figure out ways to overcome them. Precise goals must be set. A clear path to achieve the set goals must be apparent.

Data Understanding: We need to understand the trends in each dataset and consider the most diverse one in order to get a generalized model that can be useful for every student. Exploratory data analysis is performed for the same. During this phase, we will also get a deeper understanding of the data and will be able to estimate which machine-learning model might give good results.

Data Preparation: Data should be cleaned for it to be suitable for the machine learning algorithm. In order to do this, we need to perform data pre-processing steps. Here, we will be splitting the dataset into X and Y columns, checking for any missing values, categorical values, outliers and so on and dealing with them in an effective manner.[4]

Building Models: We need to experiment with various ML models in order to identify the best fit for our dataset.

Evaluation: In this step, we analyze how well the model fits our data. We check for any chances of overfitting or underfitting of the data. We also strive for a good accuracy and make sure the error rate is minimal.

Creation of an interface: After modeling the right ML algorithm we move on to creation of a website. We will be using Flask as a web framework with HTML for structure, CSS for styling, JavaScript for interactivity, and Bootstrap for a responsive website and SQLite for backend. [5]

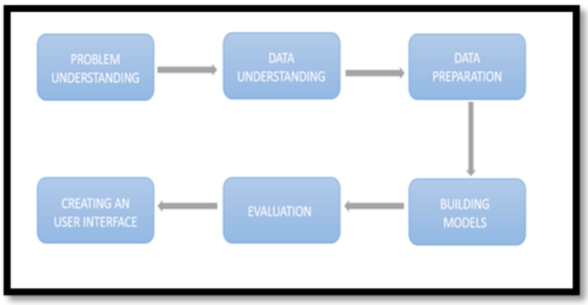


Fig – 1: Methodology

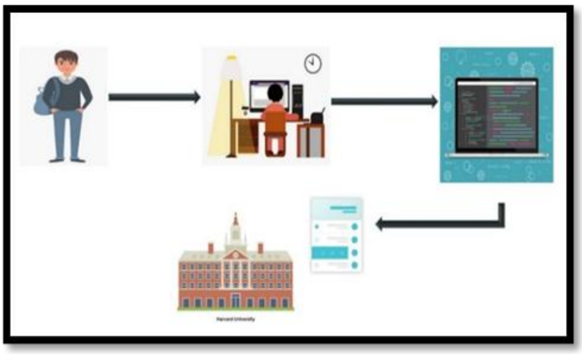


Fig – 2: User's perspective of the workflow

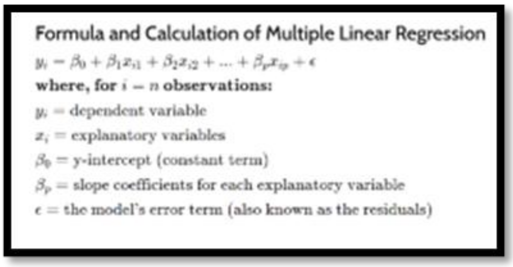
4. IMPLEMENTATION

Multiple Linear Regression with PCA

Large datasets are increasingly common in Machine Learning. While a large dataset is necessary for a good model, this introduces a great deal of complexity and difficulty in interpretation. Principal Component Analysis is a technique that is popularly used for reducing the dimensionality of such datasets. It helps in increasing interpretability and at the same time, minimizes the loss of information. PCA is an unsupervised algorithm and most commonly used as a dimensionality reduction algorithm.

Multiple Linear regression is a statistical technique that is used to model the linear relationship between several explanatory variables and one response variable.

We first import all the required libraries, and the dataset. Data preprocessing in order to make the data suitable for machine learning models. Exploratory Data Analysis (EDA) is done on the dataset in order to understand the data and apply appropriate models. Feature Scaling is performed on the independent variables of the dataset to bring every feature to the same footing, without any upfront importance of one over the other. PCA is then performed on the standardized data. Here, we have opted for three features as that explains over 85% of the variance in the data.



Formula and Calculation of Multiple Linear Regression

$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \epsilon$

where, for $i = n$ observations:

y_i = dependent variable

x_i = explanatory variables

β_0 = y-intercept (constant term)

β_p = slope coefficients for each explanatory variable

ϵ = the model's error term (also known as the residuals)

Fig – 3: MLR Formula

The data is then split into train and test sets. Due to sample variability between the train and test sets, machine learning models may give a better prediction on the training data, but fail to generalize over the test data. This leads to a high test error. In order to avoid such a situation, we have opted for K-fold Cross Validation. This method gave us an accuracy of 0.8949. Using the K-fold Cross Validation method increased the accuracy of the model by over 12%.

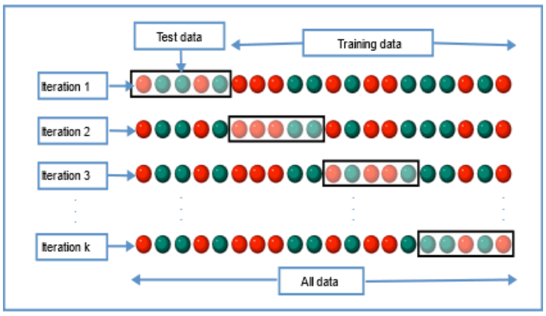


Fig – 4: K-fold Cross Validation

Random Forest Regression

Random Forest Regression (RFR) is another popular machine learning algorithm that uses the concept of ensemble learning. Random Forest is a supervised learning method that can be used for classification as well as regression problems. Random Forest contains a number of decision trees on various subsets of the given dataset and takes the average of the results to improve the predictive accuracy of the dataset. An upside to this model is that it prevents the problem of overfitting.

In this model, we followed a slightly different approach that transforms the regression problem that we have, into a classification one. The approach here is to have a binary column called 'Status' that takes the value True for applicants that have more than 83% of chance of admit. This value was taken as the threshold after a careful study of the data distribution.

This approach gave us an accuracy of 96%. The confusion matrix revealed that 3 out 100 students that were rejected were predicted as admitted and 1 out of 100 of students that were admitted were predicted as rejected.

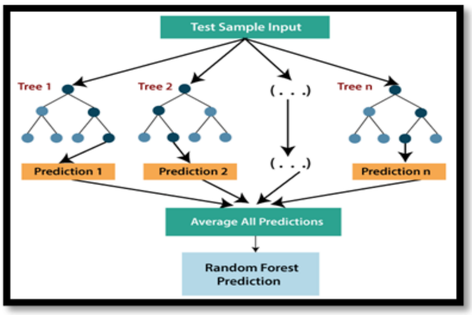


Fig – 5: Random Forest Regression

Support Vector Machine

Support Vector Machine is a supervised learning algorithm that can be used for classification as well as regression. Although it can be used for regression, it is best suited for classification problems. The objective of SVM is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. The dimension of this hyperplane depends on the number of features.

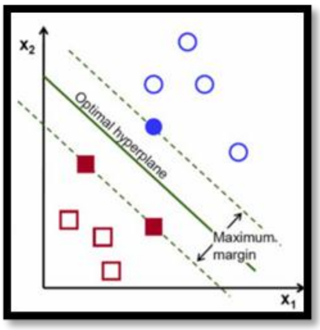


Fig – 6: Support Vector Machine

Here, we've used the same classification approach discussed earlier. This gave us an accuracy of 96% as well. The confusion matrix revealed that 2 out 100 students that were rejected were predicted as admitted and 2 out of 100 of students that were admitted were predicted as rejected.

Logistic Regression

Logistic Regression is a supervised learning technique that is used to predict a categorical dependent variable using a set of independent variables. The dependent variable is a binary value, but instead of giving the exact binary values, it gives the probabilistic values that lies between 0 and 1. Logistic regression and linear regression are quite similar. In logistic regression, instead of fitting a regression line, we fit an S shaped curve or a logistic function. The curve from the logistic function indicates the likelihood of something, such as the chances of a student getting admission in a particular university.

The classification approach discussed earlier is used here as well. This gave us an accuracy of 97%. The confusion matrix revealed that 2 out 100 students that were rejected were predicted as admitted and 1 out of 100 of students that were admitted were predicted as rejected.

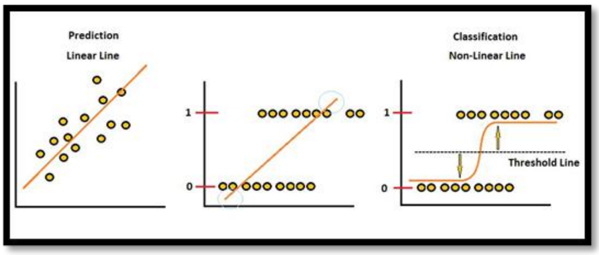


Fig – 7: Logistic Regression

This is the best accuracy that we could achieve. It gave us satisfactory results.

Mapping chance of admit to universities

We initially set out to collect data so as to predict the universities instead of the chance of admit. Due to a lack of enough data, we decided to map the chance of admit predicted in the current dataset to universities. To do so, we collected the cut-off GRE and TOEFL scores of several universities and used this dataset for mapping purposes.

The universities thus obtained are then categorized as 'Dream Universities', 'Safe Universities' and 'Backup Universities'. The students can divide their resources and apply for a combination of these universities.

5. RESULTS

As demonstrated in the previous section. The logistic regression model gave the best results, followed by SVM and RFR, then MLR with PCA.

The central concept of this application is to allow the students to predict the universities that they can get admission into. This allows the student to manage their funds and save money during the university application process, by only applying to those universities that they have a high chance of getting into. A website was created using HTML, CSS, Bootstrap for frontend, and SQLite for database. Flask was used as a web framework.

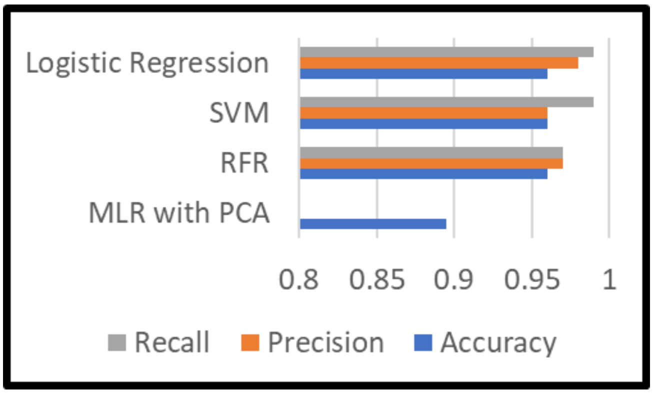
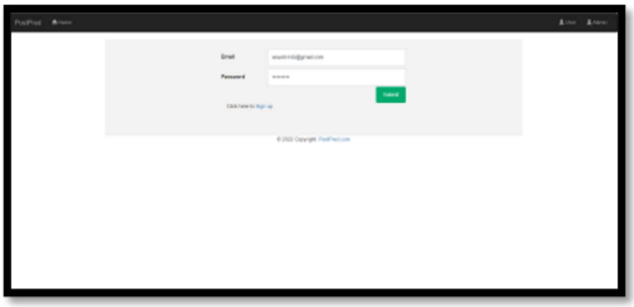


Fig – 8: Comparison of models

Fig – 9: Homepage of Postpred

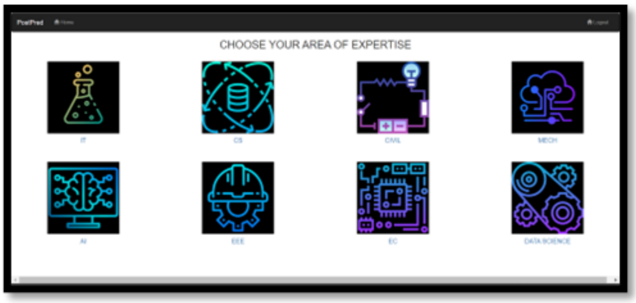
Fig - 9 shows the homepage of our website, PostPred. This is the page that the user encounters once he opens the website.



Email [illegible]
Password ******
Click here to Sign up

Fig – 10: User Login

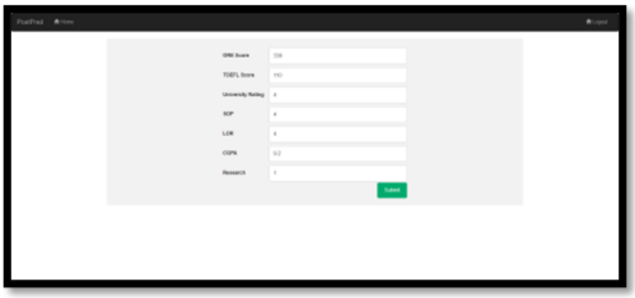
Fig - 10 shows the user login page. The user can enter their email ID and password in order to access their account. If the user doesn't already have an account, he/she can create one using the Sign Up button below.



CHOOSE YOUR AREA OF EXPERTISE

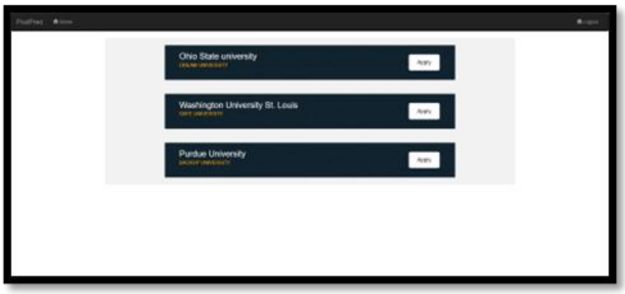
Fig – 11: Area of Expertise

Fig – 11 shows the expertise page. This page lets the user enter his/her area of expertise. Once the selection is done, the user is redirected to a prediction page.



GRE Score 338
TOEFL Score 110
University Rating 4
SOP 4
LOR 4
CGPA 9.2
Research 1

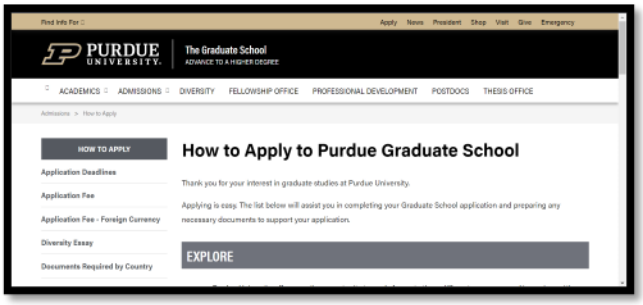
Fig – 12: Prediction Page



Ohio State university
DREAM UNIVERSITY
Washington University St. Louis
SAFE UNIVERSITY
Purdue University
BACKUP UNIVERSITY

Fig – 13: Prediction Results

The page shown in Fig - 13 shows the prediction results based on the previous scores. The results are predicted by the Machine Learning model. The universities are characterized as Dream University, Safe University and Backup University.



How to Apply to Purdue Graduate School

Thank you for your interest in graduate studies at Purdue University.

Applying is easy. The list below will assist you in completing your Graduate School application and preparing any necessary documents to support your application.

Fig – 14: Application Page for Purdue University

Fig – 14 shows the application page of Purdue University. The user is redirected to the application of the respective universities when he/she clicks on the apply button next to the university as shown in Fig – 13.

6. CONCLUSION

The central concept of this application is to allow the students to predict the universities that they can get admission into. This allows the student to manage their funds and save money during the university application process, by only applying to those universities that they have a high chance of getting into.

We've used multiple machine learning models for prediction of the same. The best one i.e., Logistic Regression is used as a pickle file and used in the Flask web framework. A user-friendly interface is thus provided to make the process easier for users from a non-technical background.

REFERENCES

[1] "Prediction fora University Admission Using Machine Learning" by Chitra Apoorva D.A, Malepati Chandu Nath, Peta Rohith, Swaroop S, Bindushree S - Blue Eyes Intelligence Engineering & Sciences Publication. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8, Issue-6 March 2020.

[2] "Machine Learning Basics with the K-Nearest Neighbors Algorithm"- https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm

[3] "Random Forest Regression" - https://www.kaggle.com/dansbecker/random-forests

[4] "Data pre-processing & Machine Learning" https://archieve.ics.uci.edu/ml/index.php

[5] "User Interface Design" - https://pidoco.com/en/help/ux/user-interface-design

[6] "Graduate Admission Prediction Using Machine Learning" by Sara Aljasmi, Ali Bou Nassif, Ismail Shahin, Ashraf Elnagar - ResearchGate Publication, December 2020.

[7] Sujay S "Supervised Machine Learning Modelling & Analysis For Graduate Admission Prediction" Published in International Journal of Trend in Research and Development (IJTRD), ISSN: 2394-9333, Volume-7 | Issue-4 , August 2020.