UCLA Graduate School Admission Predictions
STA561 Final Project
Github: https://github.com/wma8/CPS561

Jingxian Huang
jh654@duke.edu

Weikun Ma
wm101@duke.edu

Yuan Qu
yq72@duke.edu

Xiangcheng Shen
xs76@duke.edu

Abstract

The University of California, Los Angeles usually known as UCLA is a famous public research university in Los Angeles. They made a dataset contains several parameters which are considered important for application available on Kaggle – https://www.kaggle.com/mohansacharya/graduate-admissions.
In this paper, we try to use several different machine learning techniques — Support Vector Machine(SVM), Logistic Regression, and Neural Network — to predict whether a student is likely to be accepted by UCLA. We pick over 73% of chances in the dataset as "Approved" and lower otherwise.

1 Introduction

When Students apply to graduate school, there are a number of materials that need to be submitted: GRE Scores, TOEFL Scores, Undergraduate School Rating, Statement of Purpose, GPA, Research Experience, etc. Even though a student can deliver as many applications as he/she wants, no one is willing to spend money on school with little probability of admission. Therefore, they better calculate the chances of being admitted before submitting the applications.
As one of the most famous public research universities in the US, UCLA receives thousands of applications each season. One of the Kaggle datasets contains 500 applicants containing their scores, undergraduate School Ratings, GPA, etc along with their chances of admission; since the course focuses on classification, we decided to label the admission result by the chances of admission first. We decide to label the ones with over 73% chances of admission as "admitted" and lower otherwise. Since this project mainly about prediction, we will focus on the comparison between different classification methods instead of analyzing the influences of each feature in the dataset.

2 Data Analysis and Prepeocessing

The data given by the kaggle competition consist following properties[1]:

- GRE Scores (out of 340)
- TOEFL Scores (out of 120)
- University Rating (out of 5)
- Statement of Purpose and Letter of Recommendation Strength (out of 5)
- Undergraduate GPA (out of 10)
- Research Experience (either 0 or 1)
- Chance of Admit (ranging from 0 to 1)

- Admitted (ranging from 0 to 1)

We aim to build a reliable model to predict whether input scores/materials have a high chance of being admitted.

2.1 Distribution of data

First, we check the distribution of different properties and this is shown in figure 1.
Since 'admit or not' is not directly given, but only the 'chance of admission'. We set a threshold of chance of admission as 0.73. Any chance higher than this will be taken as admitted and lower will be not admitted. In this way, we can achieve a balanced admission rate.

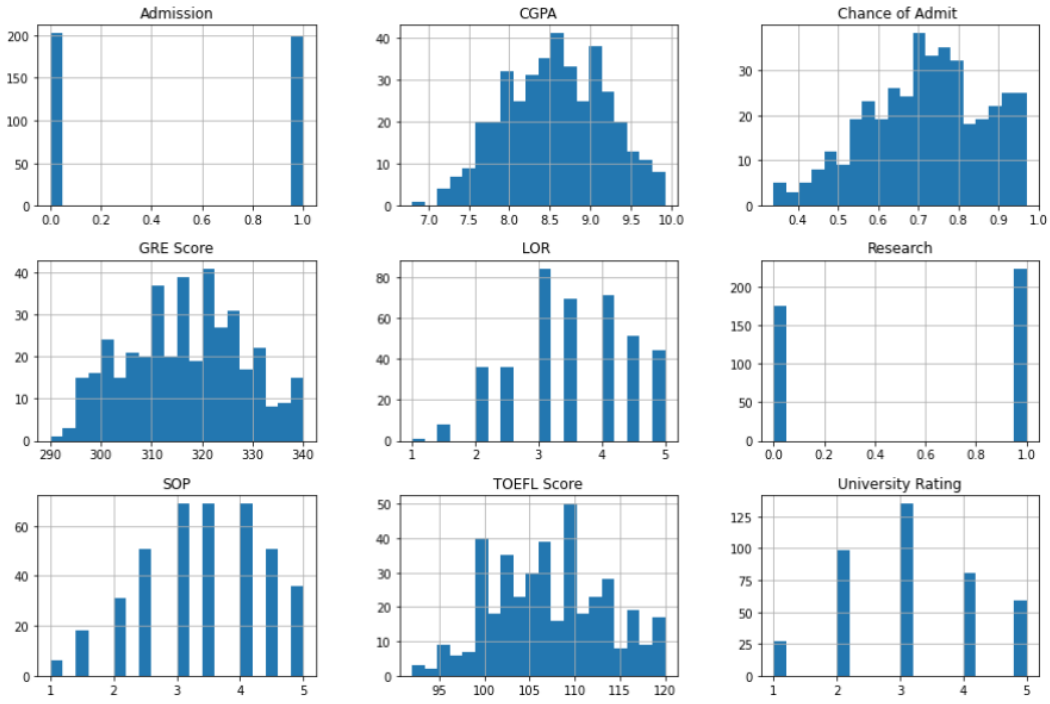

Figure 1: Distribution of different properties.

2.2 Admission Correlation

As we seen in Figure 2 the average GRE Score and TOEFL Score are noticeably higher for admission than no-admission.
Those who did more research or did higher quality of researches have more chance to be admitted.
Surprisingly, those universities having higher ratings tend to have a higher chance of admit.
The frequency of admission depends a great deal on the CGPA. Thus, the CGPA can be a good predictor of the outcome variable.

3 Analysis and Techniques

We apply three different ways to do the classification tasks: support vector machine, logistic regression and neural networks.

Figure 2: The correlation between admission and the properties.

3.1 Support Vector Machine

support-vector machines (SVMs, also support-vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier.
The Support vector machine can use different kernel functions to project the input data into different kernel spaces for better distinctions such as RBF, Polynomial, and Linear. However, by using high dimensional kernel function can bring over-fitting because of its high dimension features. Therefore, as we can see the rbf kernel function have the lowest testing accuracy.

- RBF kernel funciton has accuracy: 84%
- linear kernel function has accuracy: 87%
- poly kernel function has accuracy: 94%

3.2 Logistic Regression

Logistic Regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.). In other words, the logistic regression model predicts P(Y=1) as a function of X.
Here are some important properties of logistic regression [2]:

- Binary logistic regression requires the dependent variable to be binary.
- For a binary regression, the factor level 1 of the dependent variable should represent the desired outcome.
- The independent variables should be independent of each other. That is, the model should have little or no multicollinearity.
- The independent variables are linearly related to the log odds.

Final Accuracy: the testing accuracy is 81%

3.3 Neural Network

Neural network is a form of machine learning and is widely used in image classification by combining Convolutional Neural Network with some fully connected layers [3]. In this project, since the data is already in the form of a one-dimensional feature, we only apply the fully connected layers, or otherwise called multi-layer perceptron. The structure of the network is shown in figure 3:

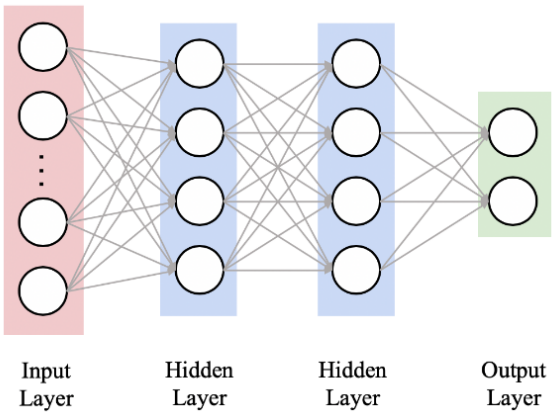

Figure 3: The structure of the model.

- Input layer: let n denotes the batch size, d denotes the feature dimension, the size of the input layer X is $n \times d$.
- Hidden layer 1: for the first hidden layer $H_1 = Relu(Xw_1 + b_1)$ where w_1 is a trainable matrix with dimension of $d \times h_1$ and b_1 is the bias.
- Hidden layer 2: similar as H_1, the second hidden layer $H_2 = Relu(H_1w_2 + b_2)$. w_2 is a trainable matrix with dimension of $h_1 \times h_2$ and b_2 is the bias.
- Output layer: the output layer $O = \sigma(H_2w_3 + b_3)$ where σ is the softmax function and w_3 is a trainable matrix of dimension $h_2 \times c$ with b as the bias. Here we have the output dimension c to be 2 because it is a binary classification so that there are only two classes involved.

We apply cross entropy as the loss function.
The experiment is under Jupyter Notebook with the help of Pytorch. We divide the data into two the training set and test set with the ratio of 4:1. The learning rate is 0.001. The optimizer we choose is Adam [4].
Final Accuracy: the testing accuracy is 81%.

4 Code Links

The link of our code can be found in: https://github.com/wma8/CPS561.

5 Conclusion

The admission prediction model given by 3 different machine algorithms provide us a rough perspective about the admission standards. With only 8 features and 300 training data, the models with high fitting capabilities can easily failed in this model because of their over-fitting nature such as neural network; thus, even though the training accuracy of the neural network is 100%, the testing accuracy is still relatively low. Out of the same reason, the RBF SVM also not capable of reaching

high testing accuracy.
On the contrary, the logistic regression also has limited accuracy because the weight of each parameter is not directly related to the admission, and the tuning parameters of logistic regression can not effectively conclude the relations between the admission and input data.
Therefore, with limited input data, we prefer to use Support Vector Machine with Polynomial kernel function.

References

[1] Graduate admission 2: Predicting admission from important parameters. https://www.kaggle.com/mohansacharya/graduate-admissions. Accessed: 2020-4-23.

[2] What is logistic regression? https://www.statisticssolutions.com/what-is-logistic-regression/. Accessed: 2020-4-25.

[3] Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017.

[4] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.