**University Admission Prediction**

Using Multi Linear Regression

Developed by: Samarendra, Manideep, Yaswanth, Sarat Chandra

Smart Bridge-Remote Summer Internship Program

1. **INTRODUCTION:**

**1.1 Overview:**

In today’s era we see a lot of students pursuing their education away from their home countries. The main country targeted by these international students is The United States of America. Majority of the international students in the United States of America are from India and China. In the past decade the number of Indian students pursuing post graduate education from the USA has rapidly increased. With the increase in the number of international students studying in the USA, each applicant has to face a tough competition to get admission in their dream university. Generally as the students don’t have much idea about the procedures, requirements and details of the universities in the USA they seek help from the education consultancy firms to help them successfully secure admission in the universities which are best suitable for their profile, for this they have to invest huge amount of money as consultancy fees. Apart from these the education consultancy firms there are few websites and blogs that guide the students on the admission procedures. The drawback of the currently available resources is that they are very limited and also they are not truly dependable taking into consideration of their accuracy and reliability.

The aim of this research is to develop a system using machine learning algorithm, we will name it as University Admission Prediction. It will help the students to identify the chances of their application to an university being accepted. Also it will help them in identifying the universities which are best suitable for their profile and also provide them with the details of those universities. A simple user interface will be developed for the users to access the University Admission Prediction system.

**1.2 Purpose:**

The aim of this project is to make use of pandas, NumPy, matplotlib, & seaborn libraries from python to extract the libraries for machine learning for the University Admission Prediction. It will help the students to identify the chances of their application to an university being accepted. Also it will help them in identifying the universities which are best suitable for their profile.

**2. LITERATURE SURVEY:**

**2.1 Existing problem:**

The Overview of this Project using latest Machine Learning Technology it is very much easy to process the ML algorithms and preprocess the given dataset and train and test the model use the appropriate algorithm which is giving correct prediction. As it is a continuous value we are going to use regression algorithm. In this we will check the correlation between the independent and dependent variables then we will get to know which column will have high and low correlation. We have the dataset of 9 attributes which are used in the process of to get   % of chance of admit.

**2.2 Proposed solution:**

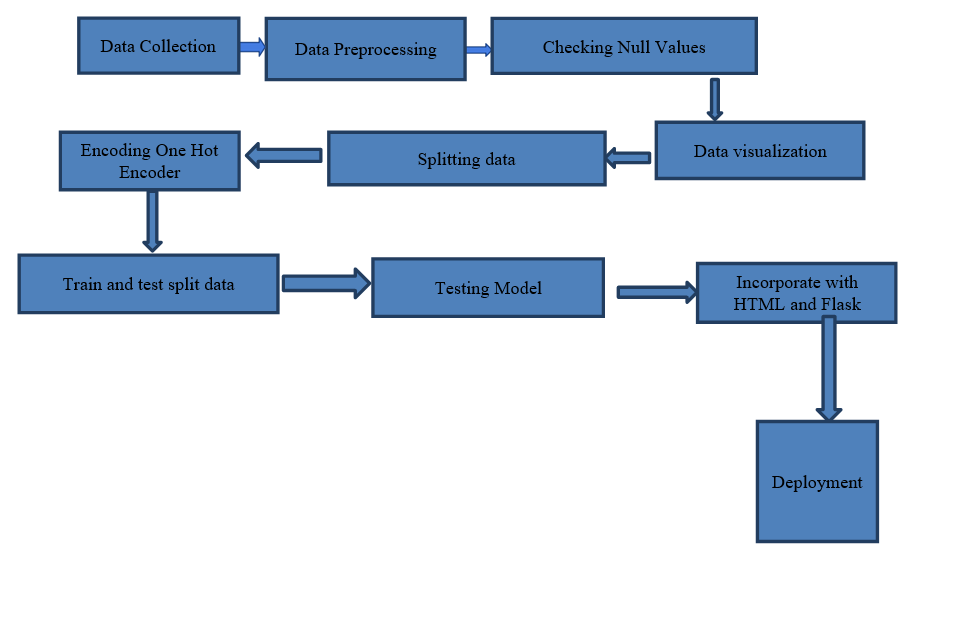
**Machine Learning (Multi Linear regression)**:

Here in this project we are going to use Multi Linear Regression Algorithm (Because here the output is continuous we apply regression). By using this algorithm we are going to predict the chance of admission. And also we have created an UI using the Flask for the chance of admission, this UI will allow the users to predict the loan status

Regression is continuous output based on the independent variables.

1. **THEORETICAL ANALYSIS:**

**3.1 Block diagram:**

****

**3.2 Software designing:**

● Jupyter Notebook Environment

● Spyder Ide

● Machine Learning Algorithms

● Python (pandas, numpy, matplotlib, seaborn, sklearn)

● HTML

● Flask

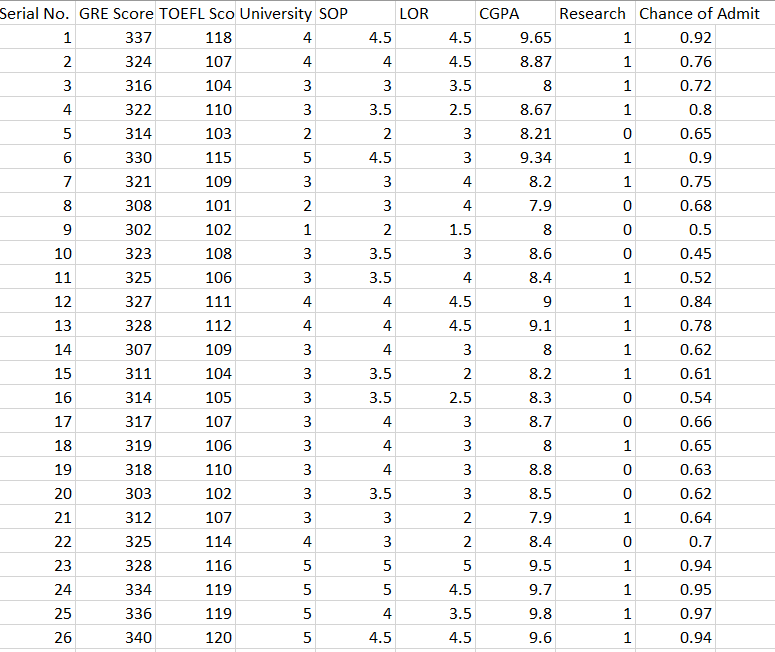
We developed this university admission prediction by using the Python language which is a interpreted and high level programming language and using the Machine Learning algorithms. For coding we used the Jupyter Notebook environment of the Anaconda distributions and the Spyder, it is an integrated scientific programming in the python language. For creating a user interface for the prediction we used the Flask. It is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions, and a scripting language to create a webpage is HTML by creating the templates to use in the functions of the Flask and HTML.

**4. EXPERIMENTAL INVESTIGATIONS:**

**4.1 Source of Data:**

In this paper, the input dataset used in developing this model has been downloaded from kaggle. Kaggle is an online community for descriptive analysis and predictive modelling. It collects variety of research fields dataset from data analytic practitioners. Structure of Dataset The dataset contains 9 columns and 500 samples. The missing values are filled in by means of mode interpolation, and the duplicate or meaningless attributes are deleted, finally we have retained to 9 attributes.

**4.2 Structure of Dataset:**

****

**4.3 Data Pre-Processing:**

Coming to the step of data pre-processing we use three libraries pandas, numpy, matplotlib. Load the given dataset and check for any null values in dataset Later The aim was to assess skewness of each variable and detecting outliers. Moreover, boxplot and threshold adjustment were used to detect outliers and strange numbers. Outliers were observed in Ore-Pulp Flow. The outliers were eliminated by using Z-score method.

**4.5 Correlation Analysis:**

A correlation analysis was employed in this study to examine if explanatory variables share the same linear relationship with the outcome variable in order to detect duplications of variables in the dataset. Among other things, highly correlations between variables were observed in the dataset. The Pearson correlation coefficient r, takes a range of values between +1 to -1. A value of 0 indicates that there of no relationship between the two variables. A value less than zero indicate a negative relationship and a value greater than zero connotes a positive association: that is as one unit of variable increases, so does the value of the other variable.

**4.6 Partition of Data:**

Partition of the Data The dataset was partitioned into two parts for training and testing purpose: 70% of the entire dataset for training the selected models and 30% for testing purpose. Most importantly, the respective training and validation dataset were randomly sampled to circumvent sampling biasness. We predictive model by partitioning the original dataset into a training set to train the model, and a validation/test dataset to assess it.

**4.7 Feature Scaling:**

Feature scaling is a technique to standardize the independent feature present in the data in a fixed range.it it performed to handle highly varying units. If feature scaling is not done the machine Learning Algorithms tends to weigh greater values are larger and smaller values are lower. For Example if and algorithm is not using feature scaling method the algorithm thought that 3000meter are greater than 3km but it not true this might Lead to give wrong predictions. So we use Feature scaling to bring back all the values to same magnitude.

**4.8 Algorithm Implementation:**

**Multi Linear Regression:**

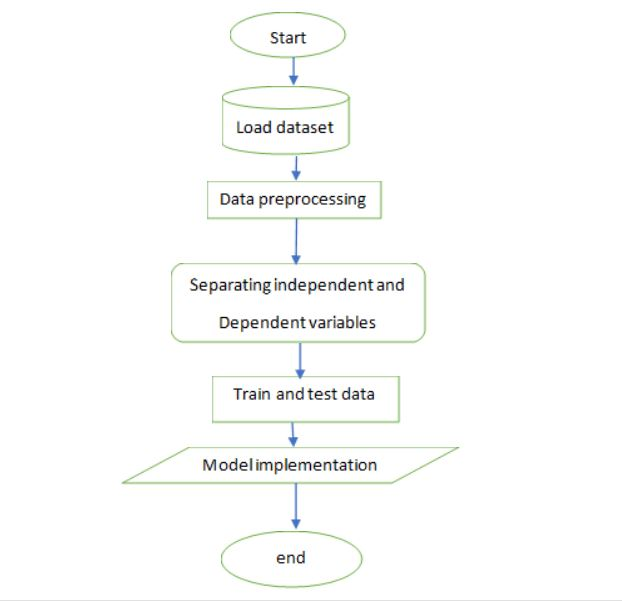
Multiple regression is used to examine the relationship between several independent variables and a dependent variable. While multiple regression models allow you to analyze the relative influences of these independent, or predictor, variables on the dependent, or criterion, variable, these often complex data sets can lead to false conclusions if they aren't analyzed properly.

* **Strengths:** There are two main advantages to analyzing data using a multiple regression model. The first is the ability to determine the relative influence of one or more predictor variables to the criterion value. The second advantage is the ability to identify outliers, or anomalies.
* **Weaknesses:** Any disadvantage of using a multiple regression model usually comes down to the data being used.

**Evaluation metrics:**

Evaluation Metrics. The prediction error is defined as the difference between its actual outcome value and its predicted outcome value. In this study, two metrics were used to compare models: - RMSE and MAE. RMSE (root mean squared error) is calculated as RMSE =.This is computed by taking the differences between the target and the actual algorithm outputs, squaring them and averaging over all classes and internal validation samples.

**5. FLOW CHART:**



**6. RESULT:**

Since we have to predict the percentage of chance of admit which is a continuous value, we have used regression technique to solve this problem. We have applied various regression algorithms as follows:

|  |  |
| --- | --- |
| **Algorithms Used** | **Accuracy** |
| Multi Linear Regression | 87.5 |
| Decision Tree Regressor | 73.12 |
| Random Forest Regressor | 86.61 |

**7. ADVAMTAGES & DISADVANTAGES:**

**Multi Linear Regression:**

**Advantages:**

●The first is the ability to determine the relative influence of one or more predictor variables to the criterion value.

 ●The second advantage is the ability to identify outliers, or anomalies.

 ●It is composed using the HTML and Python for the web usage in real time.

 ●Multi Linear Regression gave the accurate result of the prediction up to 87.5% which is the algorithm we used for prediction.

●It can work in real time and predict as soon as the necessary details for prediction are given to the model. It can work in real time and predict as soon as the necessary details for prediction are given to the model.

**Disadvantages:**

● Disadvantage of using a multiple regression model usually comes down to the data being used.

● It could not work anywhere like a web-application, if one is using other should be static.

● Needs multiple value for the prediction.

**8. APPLICATIONS:**

* We use Machine Learning Algorithms to analyze the data and helps students in shortlisting universities with their profiles.
* This model allows the students to save the extra amount of time and money that they would spend on education consultants and application fees for the universities.
* Navigating the chance of admit through the site is easy.

**9. CONCLUSION:**

The aim is to construct a robust model that simultaneously estimates the chance of admit. For the project , University Admission Prediction Multi Linear regression provides greater accuracy than the other regression algorithms which are Decision tree and Random forest. Multi Linear regression algorithm provides 87.5% accuracy.

**10. FUTURE SCOPE:**

The Multi Linear Regression algorithm can be applied on other data sets available for student profiles to further investigate its accuracy. An analysis of other machine learning algorithms other than these three can also be done to investigate the power of machine learning algorithms for University Admission Prediction. In further study, we will try to conduct experiments on larger data sets. Also, the other classification algorithms can be evaluated to resolve the problem if they perform better than the current algorithm

**11. BIBILIOGRAPHY:**

Abdul Fatah S; M, A. H. (2012). Hybrid Recommender System for Predicting College Admission, pp. 107–113.

Bibodi, J.,Vadodaria, A., Rawat, A. and Patel, J. (n.d.). Admission Prediction System Using Machine Learning.

Eberle, W., Simpson, E., Talbert, D., Roberts, L. and Pope, A. (n.d.). Using Machine Learning and Predictive Modeling to Assess Admission Policies and Standards.

Jamison, J. (2017). Applying Machine Learning to Predict Davidson College ’ s Admissions Yield, pp. 765–766

Mane, R. V. (2016). Predicting Student Admission decisions byAssociation Rule Mining with Pattern Growth Approach, pp. 202–207.

MasterPortal (2017). MasterPortal. URL: <http://www.mastersportal.eu/countries/82/united-states.html>

Mishra, S. and Sahoo, S. (2016). A Quality Based AutomatedAdmission System for Educational Domain, pp. 221–223.

**11. APPENDIX:**

**HTML:**

<!DOCTYPE html>

<html >

<!--From https://codepen.io/frytyler/pen/EGdtg-->

<head>

<meta charset="UTF-8">

<title>Graduates ML API</title>

<link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>

<link rel="stylesheet" href="{{ url\_for('static', filename='css/graduatescss.css') }}">

<style>

::placeholder

{

color: black;

opacity: 1; /\* Firefox \*/

}

.login

{

top: 20%;

left: 50%;

}

</style>

</head>

<body>

<style type="text/css">

body

{

background-repeat: no-repeat;

background-size: cover;

background-image: url('https://images.unsplash.com/photo-1472289065668-ce650ac443d2?ixlib=rb-1.2.1&ixid=eyJhcHBfaWQiOjEyMDd9&w=1000&q=80')

}

input

{

width: 100%;

margin-bottom: 10px;

background: rgba(0,0,0,0.3);

border: none;

outline: none;

padding: 10px;

font-size: 15px;

color: white;

text-shadow: 1px 1px 1px rgba(0,0,0,0.3);

border: 1px solid rgba(0,0,0,0.3);

border-radius: 20px;

box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);

-webkit-appearance: none;

-webkit-transition: box-shadow .5s ease;

-moz-transition: box-shadow .5s ease;

-moz-appearance: textfield;

-o-transition: box-shadow .5s ease;

-ms-transition: box-shadow .5s ease;

transition: box-shadow .5s ease;

}

input:focus

{

box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px 1px rgba(255,255,255,0.2);

}

label

{

padding: 10px;

font-size: 17px;

color: black;

text-shadow: 1px 1px 1px rgba(0,0,0,0.3);

}

select

{

font-size: 15px;

background: rgba(0,0,0,0.3);

}

</style>

<div class="login">

<h1><p style="color:Black;">University Admission Prediction</p></h1>

<!-- Main Input For Receiving Query to our ML -->

<form action="{{ url\_for('y\_predict')}}"method="post">

<!---<p style="color:Black;">Enter your GRE Score(340):<br></p>-->

<input type="number" step="1" min="0" max="340" name="GRE" placeholder="Enter your score in GRE" required="required" /></span>

<!---Enter your TOEFL Score(120):<br>-->

<input type="number" step="1" min="0" max="120" name="TOEFL" placeholder="Enter your score in TOEFL" required="required" />

<!---Enter the Universit Rating (5):<br>-->

<input type="number" step="0.01" min="0" max="5" name="University Rating" placeholder="Enter your University Rating" required="required" />

<!---Enter the SOP points(5):<br>-->

<input type="number" step="0.01" min="0" max="5" name="SOP" placeholder="Enter your SOP points" required="required" />

<!---Enter the LOR points(5):<br>-->

<input type="number" step="0.1" min="0" max="5" name="Letter of Recommendation" placeholder="Enter the Letter of Recomendation points" required="required" />

<!---Enter your CGPA(10):<br>-->

<input type="number" step="0.01" min="0" max="10" name="CGPA" placeholder="Enter your CGPA" required="required" />

<!---Enter your Research status(1/0):<br>-->

<label for="Research">Enter Research:</label>

<select name="Research" required="required">

<option value="">Select here</option>

<option value="0">0</option>

<option value="1">1</option>

</select>

<br>

<br>

<button type="submit" class="btn btn-primary btn-block btn-large">PREDICT</button>

</form>

<br>

{{ prediction\_text }}

</div>

</body>

</html>

**CSS:**

@import url(https://fonts.googleapis.com/css?family=Open+Sans);

.btn { display: inline-block; \*display: inline; \*zoom: 0.1; padding: 4px 10px 4px; margin-bottom: 0; font-size: 13px; line-height: 18px; color: #333333; text-align: center;text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75); vertical-align: middle; background-color: #f5f5f5; background-image: -moz-linear-gradient(top, #ffffff, #e6e6e6); background-image: -ms-linear-gradient(top, #ffffff, #e6e6e6); background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#ffffff), to(#e6e6e6)); background-image: -webkit-linear-gradient(top, #ffffff, #e6e6e6); background-image: -o-linear-gradient(top, #ffffff, #e6e6e6); background-image: linear-gradient(top, #ffffff, #e6e6e6); background-repeat: repeat-x; filter: progid:dximagetransform.microsoft.gradient(startColorstr=#ffffff, endColorstr=#e6e6e6, GradientType=0); border-color: #e6e6e6 #e6e6e6 #e6e6e6; border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25); border: 1px solid #e6e6e6; -webkit-border-radius: 4px; -moz-border-radius: 4px; border-radius: 4px; -webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05); -moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05); box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05); cursor: pointer; \*margin-left: .3em; }

.btn:hover, .btn:active, .btn.active, .btn.disabled, .btn[disabled] { background-color: #e6e6e6; }

.btn-large { padding: 9px 14px; font-size: 15px; line-height: normal; -webkit-border-radius: 5px; -moz-border-radius: 5px; border-radius: 5px; }

.btn:hover { color: #333333; text-decoration: none; background-color: #e6e6e6; background-position: 0 -15px; -webkit-transition: background-position 0.1s linear; -moz-transition: background-position 0.1s linear; -ms-transition: background-position 0.1s linear; -o-transition: background-position 0.1s linear; transition: background-position 0.1s linear; }

.btn-primary, .btn-primary:hover { text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25); color: #ffffff; }

.btn-primary.active { color: rgba(255, 255, 255, 0.75); }

.btn-primary { background-color: #4a77d4; background-image: -moz-linear-gradient(top, #6eb6de, #4a77d4); background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4); background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#6eb6de), to(#4a77d4)); background-image: -webkit-linear-gradient(top, #6eb6de, #4a77d4); background-image: -o-linear-gradient(top, #6eb6de, #4a77d4); background-image: linear-gradient(top, #6eb6de, #4a77d4); background-repeat: repeat-x; filter: progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de, endColorstr=#4a77d4, GradientType=0); border: 1px solid #3762bc; text-shadow: 1px 1px 1px rgba(0,0,0,0.4); box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.5); }

.btn-primary:hover, .btn-primary:active, .btn-primary.active, .btn-primary.disabled, .btn-primary[disabled] { filter: none; background-color: #4a77d4; }

.btn-block { width: 100%; display:block; }

\* { -webkit-box-sizing:border-box; -moz-box-sizing:border-box; -ms-box-sizing:border-box; -o-box-sizing:border-box; box-sizing:border-box; }

html { width: 100%; height:100%; overflow:hidden; }

body {

width: 100%;

height:100%;

font-family: 'Open Sans', sans-serif;

background: #092756;

color: #fff;

font-size: 18px;

text-align:center;

letter-spacing:1.2px;

background: -moz-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%),-moz-linear-gradient(top, rgba(57,173,219,.25) 0%, rgba(42,60,87,.4) 100%), -moz-linear-gradient(-45deg, #670d10 0%, #092756 100%);

background: -webkit-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -webkit-linear-gradient(top, rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -webkit-linear-gradient(-45deg, #670d10 0%,#092756 100%);

background: -o-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -o-linear-gradient(top, rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -o-linear-gradient(-45deg, #670d10 0%,#092756 100%);

background: -ms-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -ms-linear-gradient(top, rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -ms-linear-gradient(-45deg, #670d10 0%,#092756 100%);

background: -webkit-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), linear-gradient(to bottom, rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), linear-gradient(135deg, #670d10 0%,#092756 100%);

filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#3E1D6D', endColorstr='#092756',GradientType=1 );

}

.login {

position: absolute;

top: 40%;

left: 50%;

margin: -150px 0 0 -150px;

width:400px;

height:400px;

}

.login h1 { color: #fff; text-shadow: 0 0 10px rgba(0,0,0,0.3); letter-spacing:1px; text-align:center; }

input {

width: 100%;

margin-bottom: 10px;

background: rgba(0,0,0,0.3);

border: none;

outline: none;

padding: 10px;

font-size: 13px;

color: red;

text-shadow: 1px 1px 1px rgba(0,0,0,0.3);

border: 1px solid rgba(0,0,0,0.3);

border-radius: 4px;

box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);

-webkit-appearance: none;

-webkit-transition: box-shadow .5s ease;

-moz-transition: box-shadow .5s ease;

-moz-appearance: textfield;

-o-transition: box-shadow .5s ease;

-ms-transition: box-shadow .5s ease;

transition: box-shadow .5s ease;

}

input:focus { box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px 1px rgba(255,255,255,0.2); }

**APP.PY:**

import numpy as np

from flask import Flask, request, jsonify, render\_template

from joblib import load

app = Flask(\_\_name\_\_)

model= load('proj.save')

trans=load('datatransform')

@app.route('/')

def home():

return render\_template('Graduates.html')

@app.route('/y\_predict',methods=['POST'])

def y\_predict():

'''

For rendering results on HTML GUI

'''

x\_test = [[x for x in request.form.values()]]

print(x\_test)

x\_test=np.delete(x\_test,[2,3],axis=1)

test=trans.transform(x\_test)

print(test)

prediction = model.predict(test)

print(prediction)

output=prediction[0]

return render\_template('Graduates.html', prediction\_text='Chance of Admit {}'.format(output))

'''@app.route('/predict\_api',methods=['POST'])

def predict\_api():

#For direct API calls trought request

data = request.get\_json(force=True)

prediction = model.y\_predict([np.array(list(data.values()))])

output = prediction[0]

return jsonify(output)'''

if \_\_name\_\_ == "\_\_main\_\_":

app.run(debug=True)

**##THE  END##**