

PROJECT REPORT



GRADUATE ADMISSION PREDICTION

SUBMITTEDBY:-Dinesh

USN:-3GN18CS026

DEPT:-COMPUTERSCIENCE

UNDERTAKING

I declare that the work presented in this project titled “UNIVERSITY ADMISSION PREDICTION”, submitted to the TIMTS, for the award of the Internship in Data Science, is my original work. I have not plagiarized or submitted the same work for the award of any other Internship. In case this undertaking is found incorrect, I accept that my Project may be unconditionally withdrawn.

December,2021.

DINESH

CERTIFICATE

This is to certify that the work contained in the project titled "UNIVERSITY ADMISSION PREDICTION", by Dinesh Kalse, has been carried out under my supervision and that this work has not been submitted elsewhere for internship.

Times Institute of Management and Technical Studies
Data science and ML
Bengaluru, Karnataka 560103

ACKNOWLEDGEMENT

I take upon this opportunity to acknowledge the many people whose prayers and support meant a lot to me. I am deeply indebted to my mentor **Mr. Sumit Chatterjee** who motivated me along the way.

I would like to thank all my teachers who help me in this project.

I further thank my friends. My heartfelt thanks to parents who supported me a lot. I owe my sincere gratitude towards the almighty God.

Finally, I would like to wind up by paying my heartfelt thanks to TIMTS institute who provided me with this great opportunity

CONTENTS

Topic	Page No.
1) Purpose	6
2) Dataset	7
3) Theory of linear Regression	8
4) Implementation	11
5) Import Libraries	12
6) Conslusion	17
7) References	18

Purpose

To apply for a master's degree is a very expensive and intensive work. With this kernel, students will guess their capacities and they will decide whether to apply for a master's degree or not.

So, basically this set is about the Graduate Admissions data i.e. Given a set of standardized scores like GRE, TOEFL, SOP standard scores, LOR standard scores, what is probability (basically i have done a YES/NO scenario) of gaining admission into a particular school. All those folks who are preparing for MS, might point out this question, from where did you get SOP & LOR scores. These aren't public figures ? I mean yes, it might not be public, but don't you think universities might be grading these applications on some scale of rating so that the scores can be standardized. Hence the SOP, LOR scores.

Dataset

This dataset is created for prediction of graduate admissions and the dataset link is below:

<https://www.kaggle.com/mohansacharya/graduate-admissions>

Features in the dataset:

- GRE Scores (290 to 340)
- TOEFL Scores (92 to 120)
- University Rating (1 to 5)
- Statement of Purpose (1 to 5)
- Letter of Recommendation Strength (1 to 5)
- Undergraduate CGPA (6.8 to 9.92)
- Research Experience (0 or 1)
- Chance of Admit (0.34 to 0.97)

THEORY OF LINEAR REGRESSION

In machine learning, any problem can be taken as classification problem or regression problem. Different from classification, the prediction value is always continuous in a certain range, simply to say, target value is the probability of one event happening. For example, a medicine institution wants to diagnose a patient according to known a set of data of patients health records. If we want to know the patient is ill or not, this is a classification problem. But sometimes the result is not absolute. Before getting ill, the patient wants to know the probability of being ill, which can be considered as regression problem.

Linear regression is the simplest statistic model to imply the relation between variables. Firstly, let us only consider two variables relation, then I will extend it to multi variables linear regression .

A. Simple Linear Regression:-

With simple linear regression when we have a single input, we can use statistics to estimate the coefficients.

This requires that you calculate statistical properties from the data such as means, standard deviations, correlations and covariance. All of the data must be available to traverse and calculate statistics.

The representation is a linear equation that combines a specific set of input values (x) the solution to which is the predicted output for that set of input values (y). As such, both the input values (x) and the output value are numeric.

For example, in a simple regression problem (a single x and a single y), the form of the model would be:

$$y = B_0 + B_1 * x$$

In higher dimensions when we have more than one input (x), the line is called a plane or a hyper-plane. The representation therefore is the form of the equation and the specific values used for the coefficients (e.g. B₀ and B₁ in the above example).

B. Multiple Linear Regression

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression is to model the [linear relationship](#) between the explanatory (independent) variables and response (dependent) variables. In essence,

multiple regression is the extension of ordinary least-squares (OLS) [regression](#) because it involves more than one explanatory variable.

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

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)

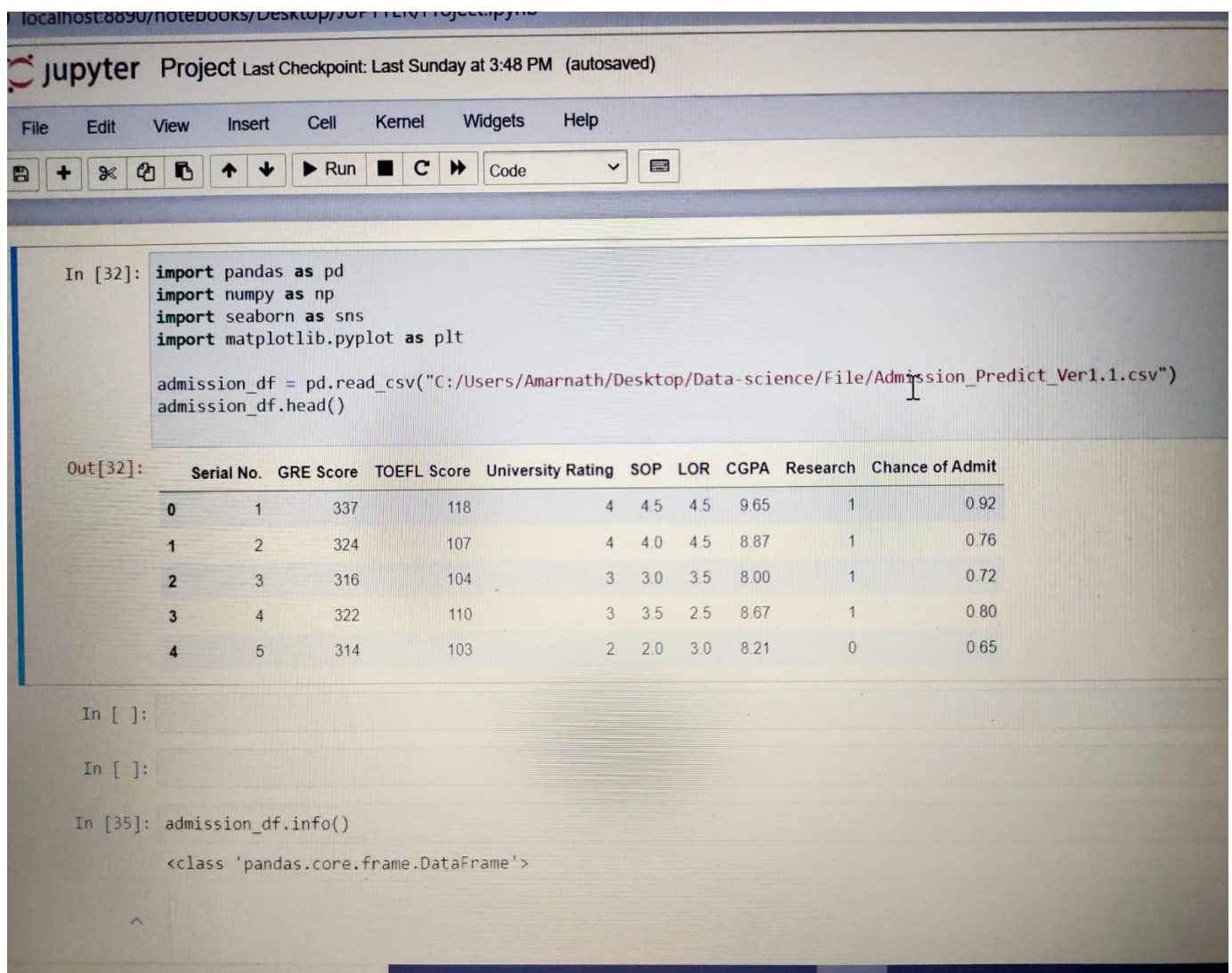
IMPLEMENTATION

This work used Python programming for this project, as it is a high-level programming language and it has vast libraries and Python automates tasks and makes it efficient. Firstly, we need to install Python then we need to import some libraries, they are:

- 1) Numpy: Numpy is used for multi-dimensional arrays, it does element to element operations and it also has different methods for processing arrays.
- 2) Panda: Pandas is one of the highly used python libraries, it provides high performance. It manipulates data and it makes data analysis fast and easy.
- 3) Sklearn: It is most useful library, this library contains lot of efficient tools, it is used to build models like statistical modelling including classification, regression, clustering. After loading required packages, we divide dataset as training and testing as follows, here 80 % of dataset is taken as training and remaining 20 % as to perform test.

IMPORT LIBRARIES and LOAD DATA

First, let's import all the modules, functions and objects we are going to use. We can load the data directly from the UCI Machine Learning repository. We are using pandas to load the data. We will also use pandas next to explore the data both with descriptive statistics and data visualization.



```
In [32]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

admission_df = pd.read_csv("C:/Users/Amarnath/Desktop/Data-science/File/Admission_Predict_Ver1.1.csv")
admission_df.head()
```

Out[32]:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	1	337	118	4	4.5	4.5	9.65	1	0.92
1	2	324	107	4	4.0	4.5	8.87	1	0.76
2	3	316	104	3	3.0	3.5	8.00	1	0.72
3	4	322	110	3	3.5	2.5	8.67	1	0.80
4	5	314	103	2	2.0	3.0	8.21	0	0.65

```
In [ ]:

In [ ]:

In [35]: admission_df.info()

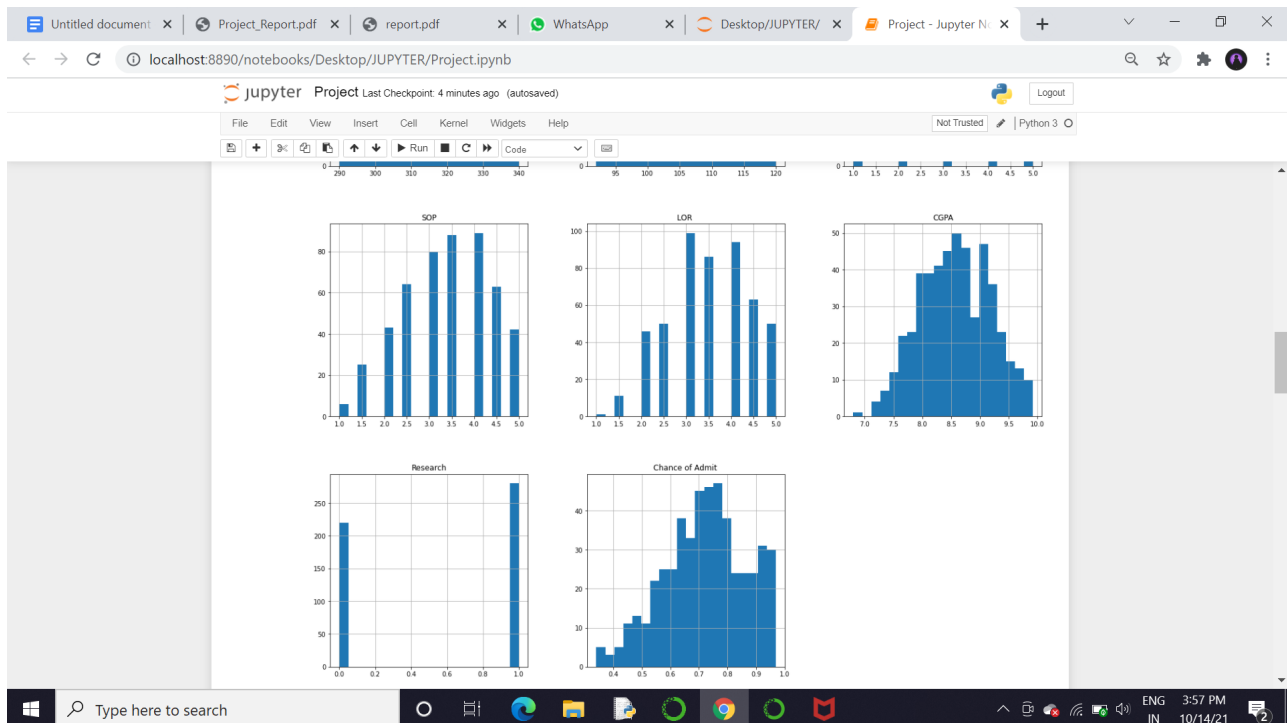
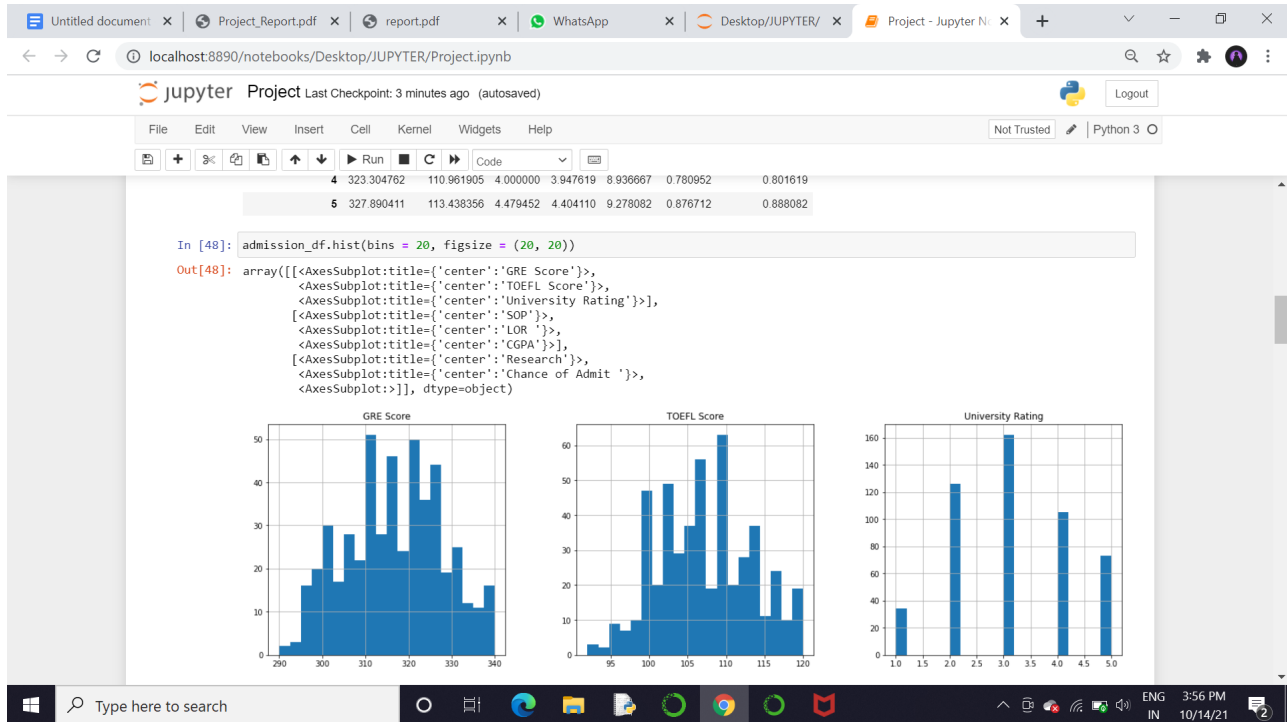
<class 'pandas.core.frame.DataFrame'>
```

[illegible]

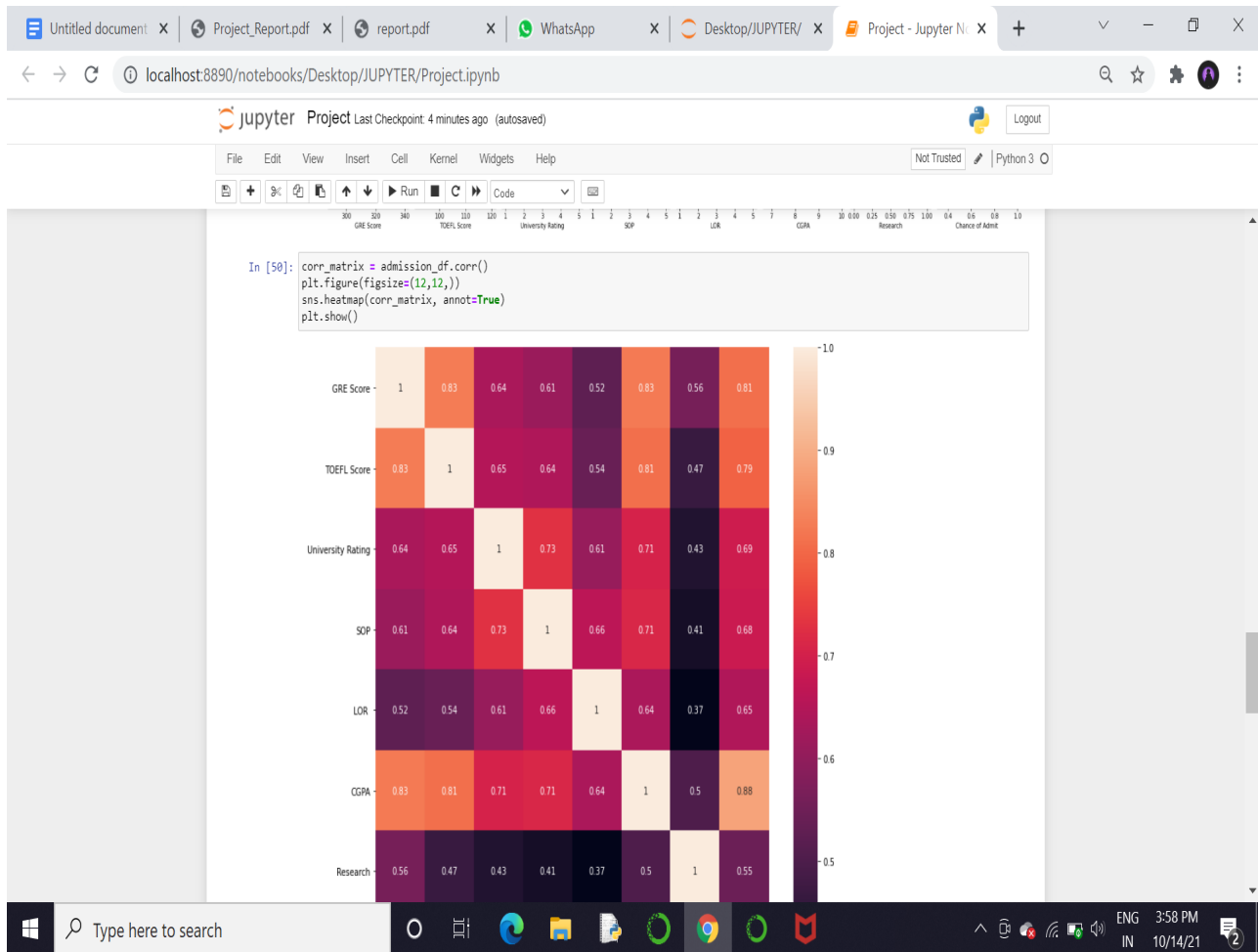
```
In [36]: admission_df.describe()
```

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
count	500 000000	500 000000	500 000000	500 000000	500 00000	500 000000	500 000000	500 00000

DATA VISUALIZATION

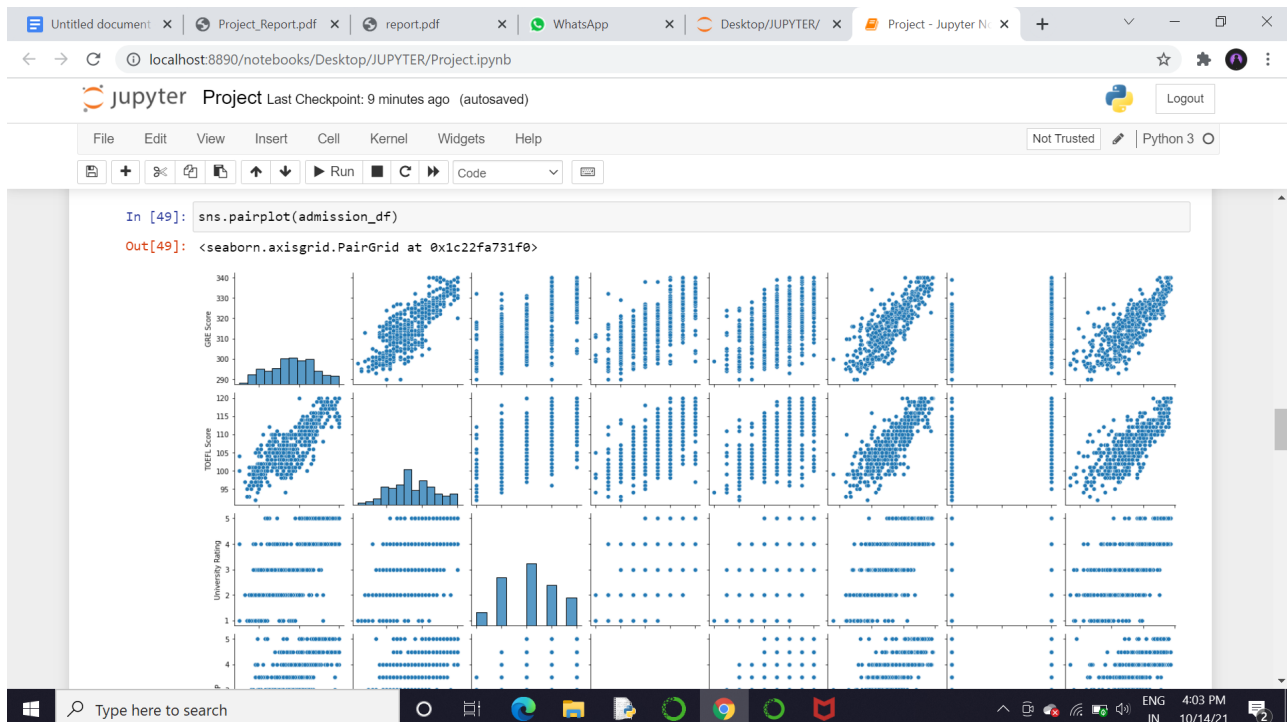


Heat Map:



Logistic Regression:

```
Untitled document x Project_Report.pdf x report.pdf x WhatsApp x Desktop/JUPYTER/ x Project - Jupyter N x +
localhost:8890/notebooks/Desktop/JUPYTER/Project.ipynb
jupyter Project Last Checkpoint: 6 minutes ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3
In [ ]:
In [68]: from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, accuracy_score
In [76]: reg = LinearRegression()
reg.fit(X_train, y_train)
Out[76]: LinearRegression()
In [78]: accuracy_LinearRegression = reg.score(X_test, y_test)
accuracy_LinearRegression
Out[78]: 0.8284436976725955
In [83]: reg.coef_
Out[83]: array([[0.13961951, 0.11126757, 0.06236484, 0.01184049, 0.09939938,
0.54220019, 0.06773376]])
In [84]: reg.intercept_
Out[84]: array([0.00207994])
In [ ]:
```



CONCLUSION

My project mainly includes three parts: data preprocessing, base model selection, modeling and stacking. Because the original data is completed and clean without too many features, most energy I spent on my project is the base model selection and stacking. Selecting base model is important so as to I save more energy in result analysis and have base line to compare with following models.

The Analysis done on the data helps both student and college to choose according to the grades.

Without proper analysis student will not be able to choose college, which affect the decision making to which they apply. That's where Data Analysis help them to select college according to their marks.

REFERENCES

- <https://machinelearningmastery.com/linear-regression-for-machine-learning>
- <https://www.investopedia.com/terms/m/mlr.asp>
- https://github.com/satwik2663/Machine-Learning-Graduate-Student-Admission-Predictor/tree/master/Final%20Project%20_%20Graduate%20Admission%20Predictor
-