Pattern Recognition and Machine Learning Project: Flight Ticket price prediction

Mukul Shingwani (B20AI023)

April 30, 2022

1 Data Preprocessing

The given dataset was read using the *read_excel* function of the *pandas* library.We checked for NaN values and found 1 each in Route and *'Total_Stops'*, since they were less in number so we dropped it.

After this, some of the features were converted to their suitable data types like *'Dep_Time', 'Arrival_Time', 'Date_of_Journey'* were converted from object into DateTime.

2 Data Cleaning

- Only the day and month were of relevance from *'Date_of_Journey'*, so we created two new columns for them and dropped the original one.
- Only the hour and minutes of departure and arrival time column matters to us and not seconds so extracting them from *'Dep_Time'* and *'Arrival_Time'* respectively, storing them in new columns, and dropping the original ones.
- Next, from the *'Duration'* column we separate the duration hours and minutes, store them in new columns, and drop the original column.
- We drop the *'Additional_Info'* column since 8344 entries out of 10682 are having value as *'No Info'*.
- Next, we analyzed the *'Route'* column and found that there are maximum 5 stops (Eg: DEL $\rightarrow$ RPR $\rightarrow$ NAG $\rightarrow$ BOM $\rightarrow$ COK) so breaking each stop/halt destination into a different column and dropping the original *'Route'* column, we are doing this since encoding the complete value in *'Route'* column doesn't make sense, we will have to use too many labels in that case since each will be a unique value

Next, we need to handle the encoding of the dataset so that they can be conveniently passed to various models, but before doing that we did some Data analysis and visualization to get a better understanding and clear picture of the dataset which is given in the next section.

For Handling the categorical data we opted for the following measures:-

- The nominal data features will be one hot encoded like *'Airline'*, *'Source'*, *'Destination'* etc
- The ordinal data features will be label encoded like *'Total_Stops'*

3 Data visualizations and Analysis

The count plots depicting the count of unique values for various features were plotted and shown below.

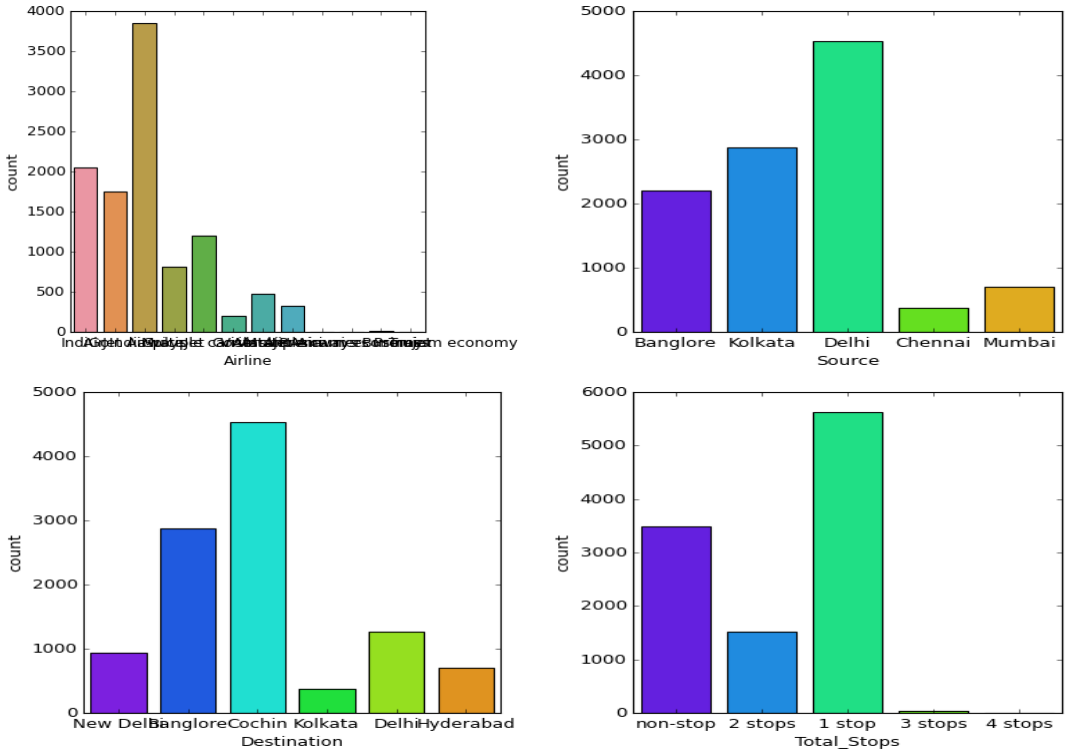


Figure 1: count vs unique values of features

The relative importance of the features for the current dataset is shown below.

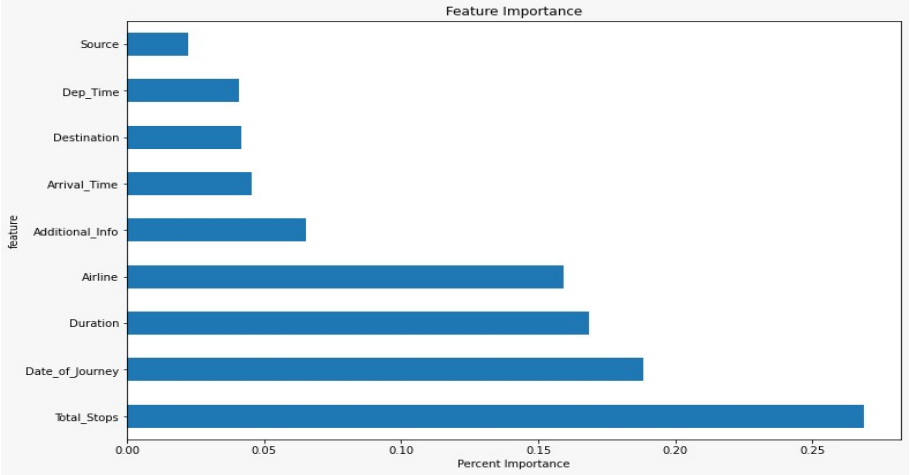


Figure 2: Relative importance of features

The plot below suggests that jet airways business class is the most expensive flight.

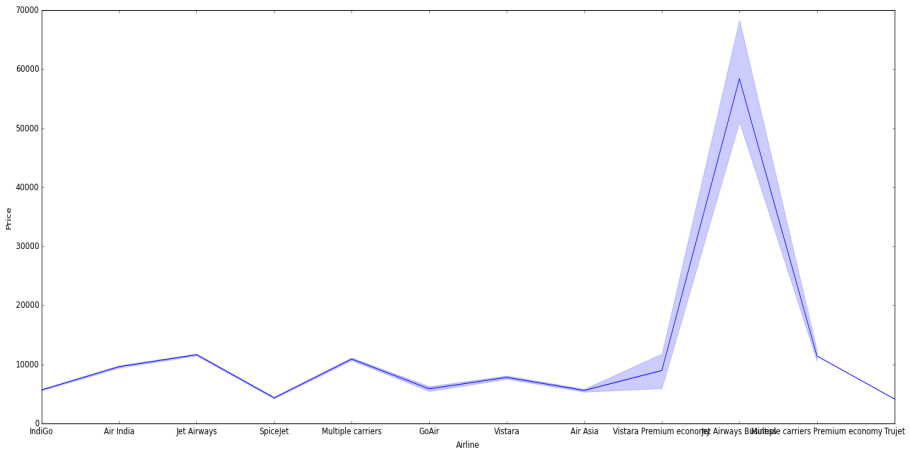


Figure 3: Lineplot for Airline v/s Price

The plot below tells us that almost all of the flights are having similar median except jet airways business class

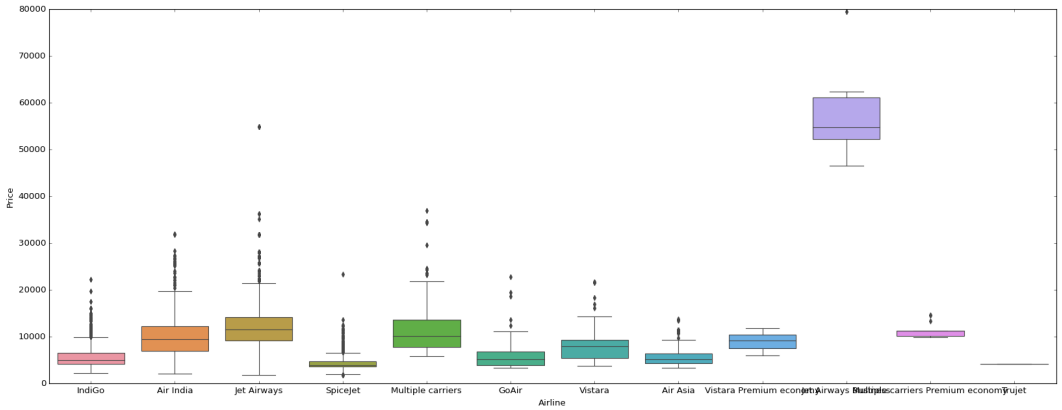


Figure 4: Boxplot for Ailrine v/s Price

The plot below tells us that flights from Delhi are having the highest median price, with maximum outliers present in flights departing from Banglore. Flights from Chennai have the lowest median price.

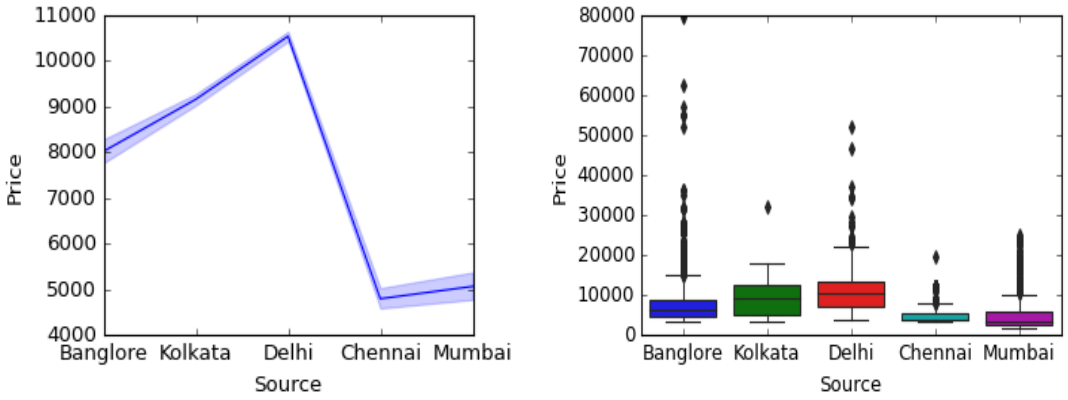


Figure 5: plots for source v/s Price

The plot below tells us that flights going to New Delhi are having the highest median price and also the maximum outliers, Flights going to Kolkata have the lowest median price

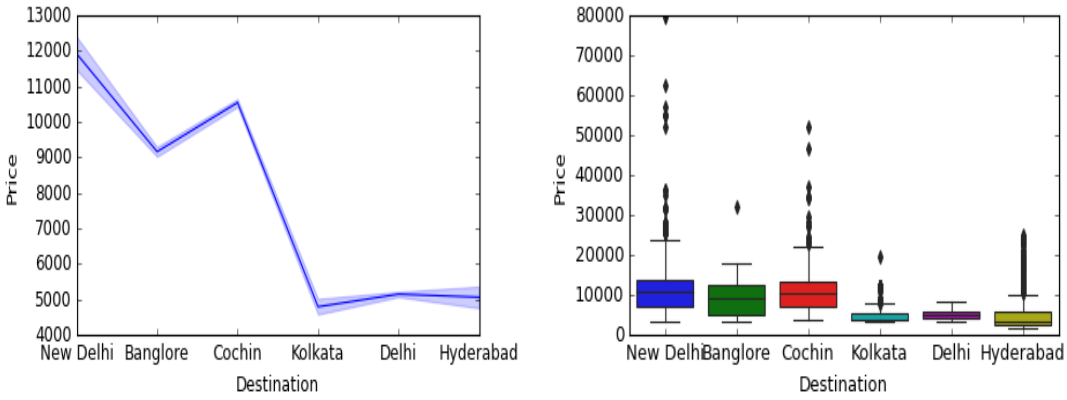


Figure 6: plots for destination v/s Price

The plot below tells us that there is only 1 flight with 4 stops having the highest median price and apart from that flights with 2 or 3 stops have the highest median price with max outliers in flights with 3 stops.

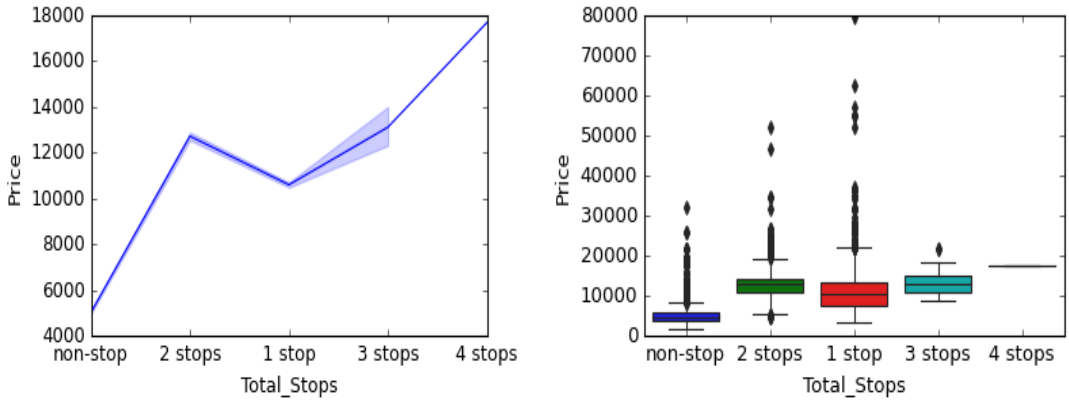


Figure 7: plots for no. of stops v/s Price

Given below are some of the plots which compares the count of various features and shows us the statistics of best airlines, source destinations, routes and some other aspects as well

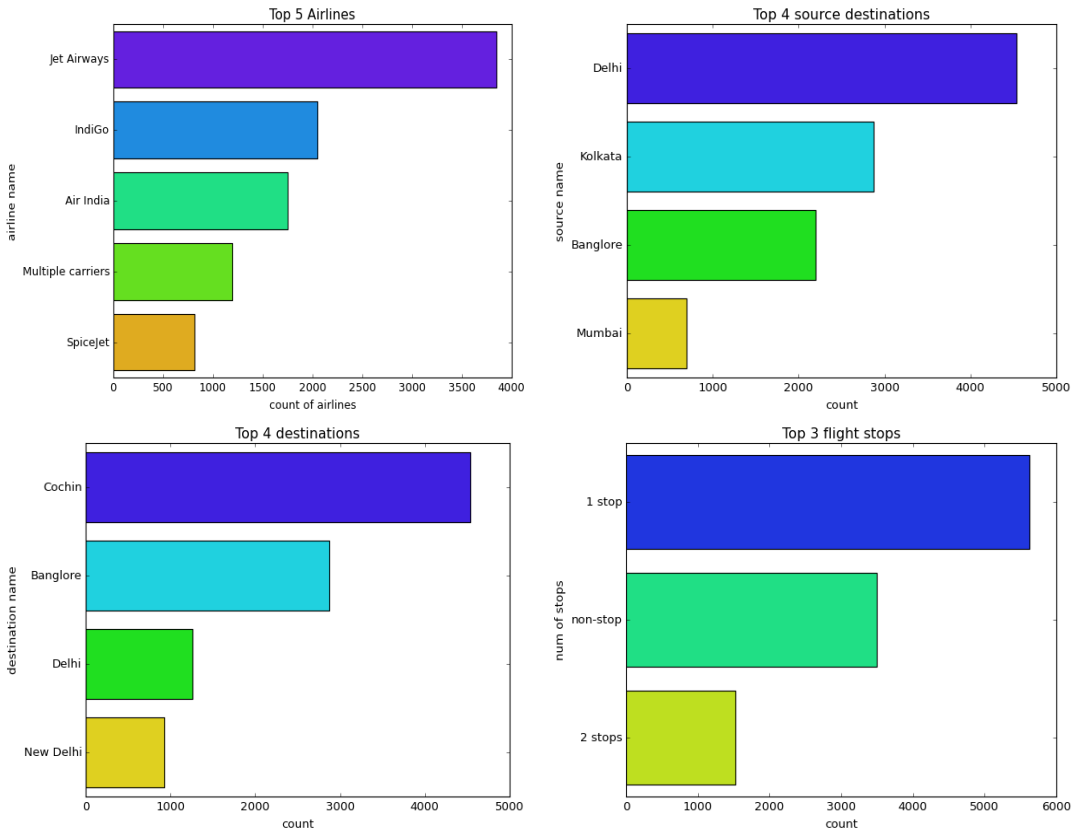


Figure 8: Analysis of features

some key takeaways from the above plots :-

- The top 5 airlines are Jet airways,Indigo, AirIndia, Multiple Carriers and Spicejet.
- The top 4 sources of the flights are Delhi, Kolkata, Banglore and Mumbai.
- The top 4 destinations of the flights are Cochin, Banglore, Delhi and New Delhi

- The top 4 flight routes are :-
  - 1.DEL → BOM → COK.
  - 2.BLR → DEL
  - 3.CCU → BOM → BLR.
  - 4.CCU → BLR.
- The most number of flights are 1 stop

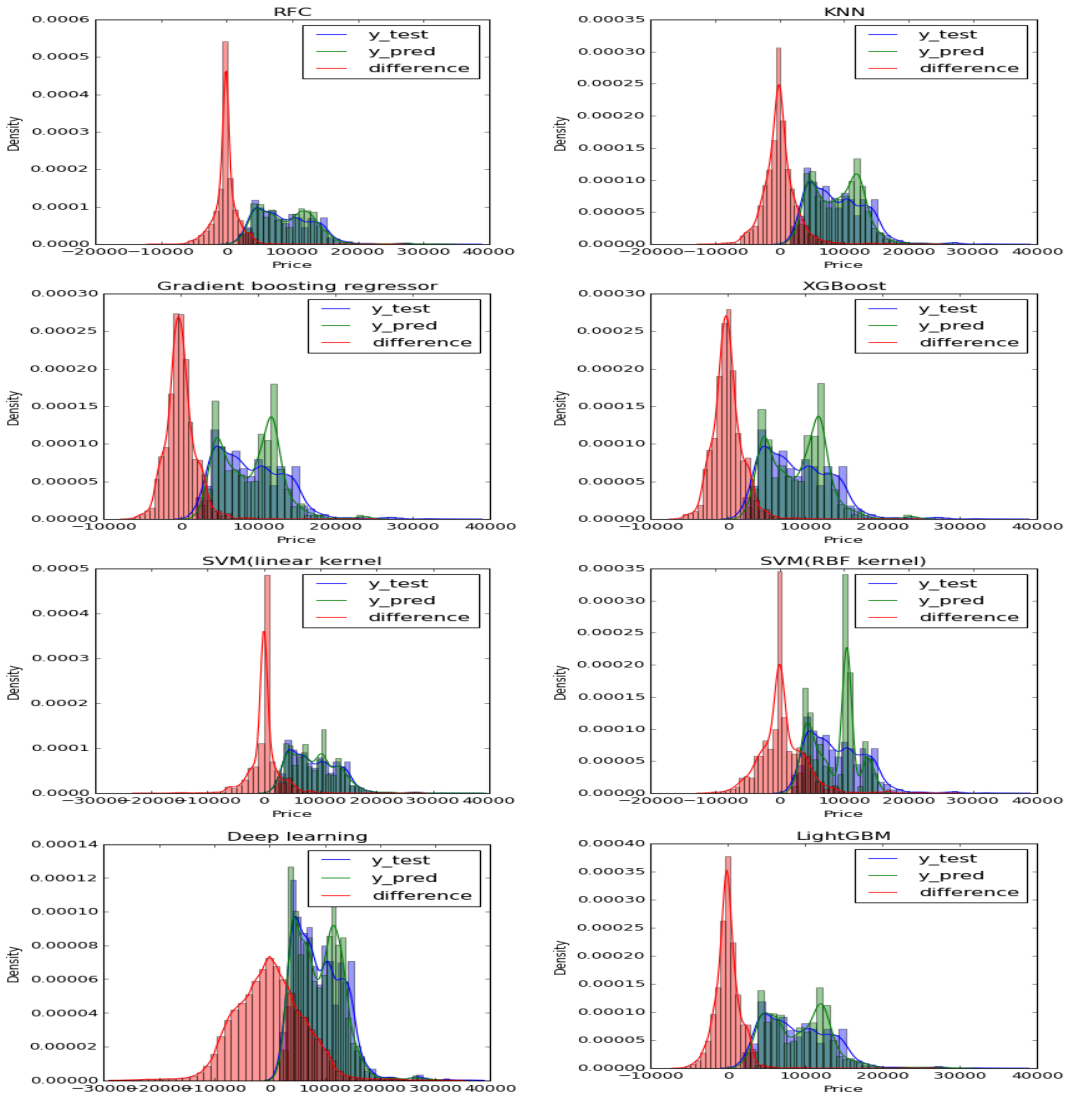
After this, the given dataset was split into training and testing datasets in a 70:30 ratio for applying various models.

4 Applying Models

Various models were applied, and their performances were judged based on various evaluation metrics, the statistics for the same are given below in the table (these scores are without any optimization of the hyperparameters)

Models report				
Model	R2 score	MAE	MSE	RMSE
Random Forest regressor	0.8346	1124.28	3254396.39	1803.99
KNN regressor	0.6347	1767.16	7187733.93	2680.99
Gradient boosting regressor	0.7990	1426.48	3953849.26	1988.42
XGB regressor	0.7984	1430.31	3965559.52	1991.37
SVM(linear kernel)	0.5965	1544.04	7937785.03	2817.40
SVM(RBF kernel)	0.3497	2385.67	12794500.67	3576.94
Deep neural networks	0.8051	1291.41	3834047.65	1958.07
Light GBM regressor	0.8546	1166.75	2825290.35	1680.86

The plots of our model's prediction, true price and the difference between the two are plotted for all the models and shown below.



After this, we also tried feature selection techniques using the concept of Sequential Feature Selection (SFFS) and tried to find the features which are most important and it came to our knowledge that when 18 features were used our Random Forest regressor model gave an R2 score of 0.7220 which was way less than the score when we considered all of the features, so we decided to keep

all of the features and not drop any. The plot for the same is given below.

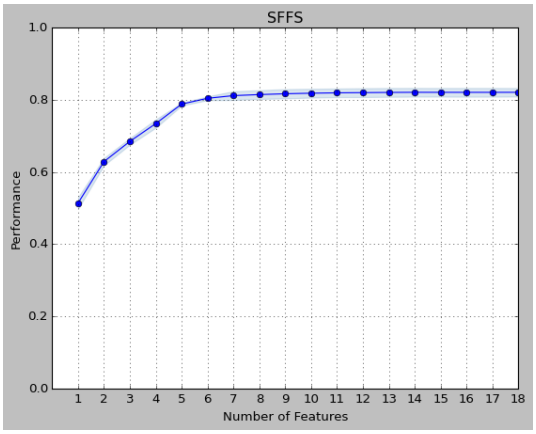


Figure 9: SFFS plot

5 Model Optimization

5.1 Grid Search

We applied the concept of Grid search to find the best value of hyper-parameters like max_depth and n_estimators for various models and saw a significant increase in the R2 scores, the results are given below in the table

Grid search results			
Model	best max depth	best n_estimators	R2 score
Random Forest regressor	15	180	0.8501
Gradient boosting regressor	6	200	0.8609
XGB regressor	6	200	0.8661
Light GBM regressor	8	300	0.8670

5.2 Auto ML

It is the process of automating the tasks of applying machine learning to real-world problems. It considers various models on its own based on the type of task mentioned, in this case, 'regression'. The metric we used to find the best model was the R2 score, it returned the XGBoost regressor as the best model with the best configuration of hyper-parameters, and upon training the model using those values we got an R2 score of 0.8334.

6 Deployment of the Model

Machine learning research usually focuses on optimizing and testing some criteria, but more criteria are needed to deploy in public policy settings. The issue of technical and non-technical deployments has received relatively little attention. However, effective implementation is essential to the true benefits and impact of machine learning models

After Analyzing various models and techniques we decided to go with Multi Layer Perceptron (Deep Learning Technique) as the model which will work for predicting the flight prices based on user input.

6.1 The Keras Model

Keras is a powerful and easy-to-use free open source Python library for developing and evaluating deep learning models.
A keras model consists of mulitple components :-

- The architecture, or configuration, which specifies what layers the model contain, and how they're connected.
- A set of weights values (the "state of the model").
- An optimizer (defined by compiling the model).
- A set of losses and metrics (defined by compiling the model or calling add_loss() or add_metric()).

We basically need to save the architecture / configuration only, typically as a JSON file and the file which contains weights values only which is generally used when training the model.
Saving a Model :-

model = ... Get model
model.save('path/to/location')

Loading a Model :-

from tensorflow import keras
model = keras.models.load_model('path/to/location')

6.2 Web Development

We designed the front end of our Website using HTML, CSS, SCSS and JavaScript and successfully deployed it using github, given below is one of the photo of our website's Interface

Figure 10: Hosted Website Interface

The user here entered the required details for flight price prediction, those details were extracted in the back-end of our website and using them we created a Numpy Array of 29 Features which was then passsed to the predict function of our ML model, the prediction which it returned was then indeed returned back to the HTML page and displayed to the user.

Link of the hosted website : - Predict Flight Price

Github Link of the Project : - View on Github