**INTRODUCTION**

This project was made because we were intrigued and we wanted to gain hands-on experience with the Machine Learning Project. As a first project, we intended to make it as instructional as possible by tackling each stage of the machine learning process and attempting to comprehend it well. We have picked Bangalore Real Estate Prediction as a method, which is known as a "toy issue," identifying problems that are not of immediate scientific relevance but are helpful to demonstrate and practice. The objective was to forecast the price of a specific apartment based on market pricing while accounting for various "features" that would be established in the following sections.

**TASK DEFINATION**

Real Estate Property is not only a person's primary desire, but it also reflects a person's wealth and prestige in today's society. Real estate investment typically appears to be lucrative since property values do not drop in a choppy fashion. Changes in the value of the real estate will have an impact on many home investors, bankers, policymakers, and others. Real estate investing appears to be a tempting option for investors. As a result, anticipating the important estate price is an essential economic indicator. According to the 2011 census, the Asian country ranks second in the world in terms of the number of households, with a total of 24.67 crores. However, previous recessions have demonstrated that real estate costs cannot be seen. The expenses of significant estate property are linked to the state's economic situation. Regardless, we don't have accurate standardized approaches to live the significant estate property values.

**ALGORITHM DEFINATIONS**

**Linear Regression**

Linear regression is a supervised learning technique. It is responsible for predicting the value of a dependent variable (Y) based on a given independent variable (X). It is the connection between the input (X) and the output (Y). It is one of the most well-known and well-understood machine learning algorithms. Simple linear regression, ordinary least squares, Gradient Descent, and Regularization are the linear regression models.

**Decision Tree Regression**

It is an object that trains a tree-structured model to predict data in the future in order to provide meaningful continuous output. The core principles of decision trees, Maximizing Information Gain, Classification trees, and Regression trees are the processes involved in decision tree regression. The essential notion of decision trees is that they are built via recursive partitioning. Each node can be divided into child nodes, beginning with the root node, which is known as the parent node. These nodes have the potential to become the parent nodes of their resulting offspring nodes. The nodes at the informative features are specified as the maximizing information gain, to establish an objective function that is to optimize the tree learning method.

**Random Forest Regression**

It is an essential learning approach for classification and regression to create a large number of decision trees. Preliminaries of decision trees are common approaches for a variety of machine learning problems. Tree learning is required for serving n off the self-produce for data mining since it is invariant despite scaling and several other changes. The trees are grown very deep in order to learn a high regular pattern. Random forest is a method of averaging several deep decision trees trained on various portions of the same training set. This comes at the price of a slight increase in bias and some interoperability.

**METHODOLOGY**

**Data Science**

The first stage is standard data science work, in which we take a data set named ‘Bengaluru House pricing data' from Kaggle. We will do significant data cleaning on it to guarantee that it provides reliable predictions throughout prediction. This Jupyter notebook, ‘Bangalore-HousePrice-Prediction-Model.ipynb,' is where we do all of our data science work. Because the Jupyter notebook is selfexplanatory, we will only touch on the principles that we have implemented briefly. In terms of data cleansing, our dataset needs a significant amount of effort. In fact, 70% of the notebook is dedicated to data cleaning, in which we eliminate empty rows and remove superfluous columns that will not aid in prediction. The process of obtaining valuable and significant information from a dataset that will contribute the most to a successful prediction is the next stage. The final stage is to deal with outliers. Outliers are abnormalities that do massive damage to data and prediction. There is a lot to comprehend conceptually from the dataset in order to discover and eliminate these outliers. Finally, the original dataset of over 13000 rows and 9 columns is reduced to about 7000 rows and 5 columns.

**Machine Learning**

The resulting data is fed into a machine learning model. To find the optimal procedure and parameters for the model, we will mostly employ K-fold Cross-Validation and the GridSearchCV approach. It turns out that the linear regression model produces the best results for our data, with a score of more than 80%, which is not terrible. Now, we need to export our model as a pickle file (model.pickle), which transforms Python objects into a character stream. Also, in order to interact with the locations(columns) from the frontend, we must export them into a JSON (columns.json) file.

**Frontend**

The front end is built up of straightforward HTML. To receive an estimated pricing, the user may fill-up the form with the number of square feet, BHK, bathrooms, and location and click the ‘ESTIMATE PRICE' button. We have used Flask Server and configured it in python. It takes the form data entered by the user and executes the function, which employs the prediction model to calculate the projected price in lakhs of rupees (1 lakh = 100000).

**RESULTS**

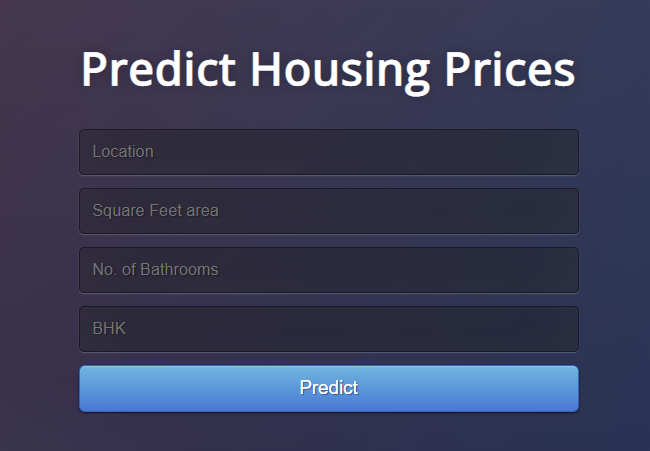
****

Fig.1 User Interface

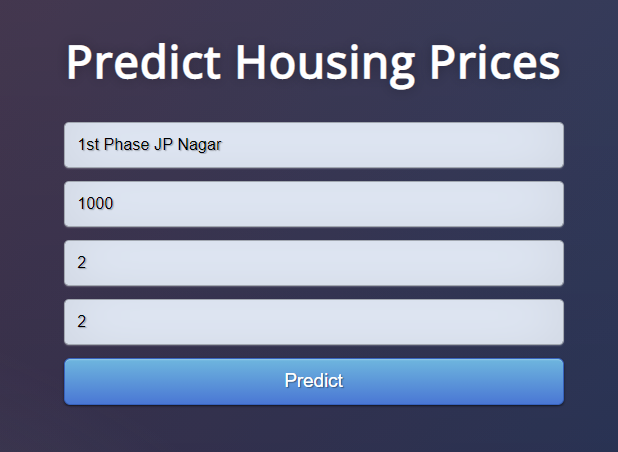


Fig.2 Estimating Price

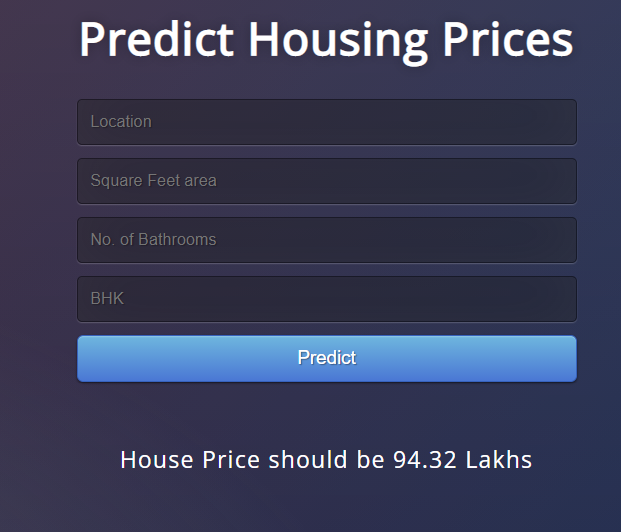


Fig.3 Prediction

**CONCLUSIONS**

With several characteristics, the suggested method predicts the property price in Bangalore. We experimented with different Machine Learning algorithms to get the best model. When compared to all other algorithms, the Lasso Regression Algorithm achieved the lowest loss and the greatest R-squared. Flask was used to create the website. Let's see how our project pans out. Open the HTML web page we generated and run the app.py file in the backend. Input the property's square footage, the number of bedrooms, the number of bathrooms, and the location, then click 'Predict.' We forecasted the cost of what may be someone's ideal home.