**Author: Naveen Donthula**

**DATA 606: CAPSTON PROJECT PROPOSAL**

**Real Estate Price Predictor**

**Introduction**

This data science project about the creating a real estate price prediction website step by step. First, we will create a model with sklearn and linear regression using the Bangalore home prices dataset from kaggle.com. The second step would be to create a Python Flask server that serves http requests using the stored model. The third component is a website developed in html, CSS, and JavaScript that allows users to enter information such as home square footage, bedrooms, and so on, and then calls a Python Flask server to receive the anticipated price. Almost all data science principles will be covered during model creation, including data load and cleaning, outlier identification and removal, feature engineering, dimensionality reduction, gridsearchcv for hyperparameter tuning, and k fold cross validation etc. The tools which used in the project are,

**Project Overview**

The aim of this project is to build a real estate price prediction and also creating a website. It involves creating a model using linear regression and sklearn, cleaning and preparing the data, and implementing data science principles like outlier removal and feature engineering. A Python Flask server will serve as the backend to handle HTTP requests and provide predictions. The website, developed using HTML, CSS, and JavaScript, will allow users to input property details and receive predicted prices from the Flask server. The project covers various data science techniques and provides an end-to-end solution for real estate price prediction.

Dataset has 13,320 rows and 9 columns.

This dataset is from Kaggle created by amitabhajoy, is having the data of size, 904 KB.

Dataset is downloaded from here:

**SOURCE**: <https://www.kaggle.com/amitabhajoy/bengaluru-house-price-data>

**Features:**

**Columns**

'area\_type' : Arear in which the house is located

'availability' : The availability date of the house

'location', : Location of the house.

'size', : Number of bedrooms in the house.

'society', : In which society the house is located.

'total\_sqft', : The area of the house.

'bath', : This column about number of bathrooms.

'balcony', : No of balcony does a house have.

'price' : Price of the house in lakhs(INR\_ Indian Rupees)

**Techniques and Models:**

Machine Learning Models : Regression Models.

1) Python

2) Numpy and Pandas for data cleaning

3) Matplotlib for data visualization

4) Sklearn for model building

5) Jupyter notebook, visual studio code and pycharm as IDE

6) Python flask for http server

7) HTML/CSS/Javascript for UI

**Research Interests and outcomes**

The research interests in this project include advanced machine learning models for real estate price prediction, novel feature engineering techniques, deep learning algorithms for market trend analysis, and the impact of external factors on real estate prices.

The expected outcomes of this research are improved accuracy in predicting real estate prices, enhanced understanding of factors influencing prices, automated valuation models for financial institutions, and comprehensive market analysis tools for real estate professionals and policymakers.

These research interests aim to drive informed decision-making, facilitate strategic investments, streamline property appraisal processes, and provide valuable insights into the dynamics of real estate markets.

**Importance of the Issue:**

This project is important and useful for real estate buyers, sellers, agents, investors, financial institutions, researchers, and academics. It provides accurate price predictions, empowering decision-making, facilitating fair transactions, attracting clients, aiding investment choices, streamlining loan approvals, and contributing to real estate market efficiency.

**Questions to be answered:**

1. How accurate are the real estate price predictions provided by the developed model compared to the actual market prices, and what is the overall performance evaluation of the model in terms of mean squared error or R-squared value?
2. How well does the real estate price prediction website handle a high volume of concurrent user requests, and what is the average response time for serving predictions? Additionally, what measures have been implemented to ensure the scalability and robustness of the Flask server?
3. How well does the Flask server handle HTTP requests and serve predictions in a timely manner?
4. How efficiently does the model perform with dimensionality reduction techniques, balancing predictive power and computational complexity?
5. How effective is the hyperparameter tuning using GridSearchCV in improving the model's performance?

**References**

* The Flask documentation offers detailed information on creating a Flask server and handling HTTP requests. <https://flask.palletsprojects.com/en/2.3.x/>
* <https://www.kaggle.com/amitabhajoy/bengaluru-house-price-data>