AUTUMN INTERNSHIP PROJECT REPORT

Analysing house prices and predicting price based on other factors related to it

Data Cleaning and Regression

Notebook - 05

Prabrisha Bharadwaj,

Course – 4 week Autumn Internship Program (Section I)

Institute - Government College of Engineering and Leather Technology

Period of Internship: 25th August 2025 - 19th September 2025

Report submitted to: IDEAS – Institute of Data Engineering, Analytics and Science Foundation, ISI Kolkata

1. Abstract

This project focuses on cleaning and analyzing a housing dataset from India with regression models. The dataset, house_price_india.csv, contains attributes related to property features and their corresponding prices. Such data is pre-processed (handling missing values, removing duplicates, encoding categorical variables, correcting data types etc.) and passed through regression analysis. Regression models are developed to evaluate how features influence house price, and performance has been measured using metrics such as R^2 and Mean Squared Error. The study demonstrates the importance of systematic data cleaning and regression analysis in extracting insights from real estate datasets

2. Introduction

Background and Relevance

In the modern data-centric landscape, organizations increasingly rely on well-prepared data to drive predictive modeling, decision-making, and optimization. The Indian housing market is one of the fastest-growing real estate sectors in the world. Analyzing house price data provides crucial insights for buyers, sellers, and policymakers. Regression models help identify key drivers of price variation, such as location, property size, and amenities.

Technology Involved:

The project leverages Python as the programming language, utilizing libraries such as:

- Pandas: Data preprocessing, manipulation and cleaning
- NumPy: Numerical computations and handling of arrays
- Matplotlib & Seaborn: Visualization of trends via graphs and charts
- Scikit-learn: Regression modeling and evaluation
- Google Colab: Development and execution environment

Procedure/Method:

- **Data Collection:** Already provided dataset has been used
- **Data Preprocessing:** Missing values imputed using three methods — linear interpolation, polynomial interpolation, and KNN imputation.
- **Exploratory Data Analysis:** Conduction of correlation analysis and visualization of relationships using distplots, countplots, and other seaborn and matplotlib graphs/charts to understand feature importance.
- **Model Building:** Threshold 0.6 taken to select features with greatest correlation with the target price for linear regression modeling. The dataset is split into training and testing sets (60-40). Linear regression fittings and predictions made on the test set.

- **Model Evaluation:** Use of Mean Squared Error and R-squared to evaluate model performance.
- Sample Secondary Model Use: Model (Linear Regression) used again on different dataset (Scikit-learn Diabetes Dataset, 80-20 train-test split)
- **Tools Used:** Python libraries including Pandas, NumPy, Seaborn, Matplotlib, Scikit-learn.

Purpose of doing the project:

- The primary objective of this project is to predict house prices based on various factors such as the number of bedrooms, bathrooms, living area, condition of the house, grade of the house, area of the house, and other related features.
- By analyzing different features (e.g., number of bedrooms, lot size, house condition), one can identify which factors have the strongest influence on house prices. This helps real estate agents, investors, and homeowners make more informed decisions.
- Other purposes include a general application of data handling and use of models for prediction, which involves the following objectives:
  - To practice and demonstrate data preprocessing techniques on a real estate dataset.
  - To identify key features influencing house prices, such as the number of bedrooms, living area, and house condition.
  - To build a linear regression model for predicting house prices based on various property attributes.
  - To explore the relationship between house features and their impact on market value.
  - To develop skills in Python-based data analysis and prepare the groundwork for future predictive modeling in real estate or other domains

Topics covered in Training

During the internship, I received training covering fundamental concepts in data science, programming, and machine learning. They include:

• Python Programming Basics: Variables, loops, operators, lists, tuples, strings

• Functions, Classes & Recursion, examples like Fibonacci series, Armstrong numbers, etc.

• NumPy: Initializing and manipulating matrices, 2D arrays, and performing mathematical operations

• Pandas: Data Frames, and related operations like filtering, grouping, and merging

• General Introduction to Data Science

• Machine Learning Overview: Supervised and unsupervised learning methods, working of popular AIs

• Regression and classification

• LLM (Large Language Model) Fundamentals & Lab: Introduction to modern AI tools and applications

• Professional Development: Communication skills

3. Project Objective

- **To predict house prices using linear regression**, based on various features such as number of bedrooms, living area, house condition, and lot size. The goal is to build a reliable model that can estimate the price of a house given its characteristics.
- **To illustrate how different property features impact market value**, helping identify which factors (e.g., living area, grade, renovation status) are most strongly associated with house prices.
- **To apply data preprocessing and cleaning techniques** on real-world housing data, including handling missing values, converting data types, and selecting relevant features for modeling.
- **To demonstrate the practical application of linear regression** in solving real estate pricing problems, and show how machine learning can assist in data-driven decision-making in housing markets.
- **To test the hypothesis** that "larger houses with more features (like higher grade, renovation, or better condition) are associated with higher market prices," using regression analysis and correlation metrics.

Note: No sample survey was conducted for this project. The dataset used is assumed to be historical housing sales data, and the analysis is intended for the general home-buying and real estate investment population. In case of conduction of survey, target population would be **Homeowners and recent home buyers** – to understand the features they considered most important when buying or pricing a house.

4. Methodology

Data Collection and Loading

- The dataset used for this project was house_price_india.csv, a real-world housing dataset containing multiple features related to houses such as price, area, number of bedrooms, year built, and other characteristics.

- The dataset was stored on **Google Drive** and imported into **Google Colab** using the pandas library, which facilitated easy data manipulation and exploration.

Data Cleaning and Preprocessing

- Initially, synthetic missing values were inserted to simulate real-world data imperfections: 20% of data points in each column were randomly replaced with NaN values.
- Focused missing values were also introduced in the 'Built Year' column to test various imputation strategies.
- Irrelevant or problematic columns such as Date, Longitude, Postal Code, Renovation Year, Lattitude, and some renovation-related area columns were dropped to reduce noise and improve model accuracy.
- Data types were adjusted where necessary, e.g., converting the "Number of schools nearby" column to integer type for categorical grouping.

Exploratory Data Analysis (EDA)

- Descriptive statistics of numerical columns generated to understand distributions, central tendencies, and spread.

- Visualizations include:

    - Line plots for price distribution across the dataset.

    - Distribution plots (using Seaborn’s distplot) for the total area of houses (combining living area and lot area).

    - Count plots and histograms for categorical features such as the number of nearby schools.

- Correlation analysis performed using a heatmap to identify relationships between features, particularly focusing on those highly correlated with the house area (excluding the basement).

Handling Missing Values

Several imputation methods were explored to handle missing data:

- **Row deletion:** Simply removing rows containing missing values.

- **Mean imputation:** Filling missing values with the column mean.

- **Standard deviation imputation:** Filling missing values with the column standard deviation.

- **Interpolation:** Using linear and polynomial interpolation methods.

- **K-Nearest Neighbors (KNN) imputation:** Applied both on original and scaled data (scaled using MinMaxScaler), with inverse transformation applied post-imputation to return to original scales.

Feature Engineering and Selection

- A new feature total_area was created by summing living area and lot area.
- Features with strong correlation (correlation coefficient > 0.6) with the target variable (Price) were selected for modeling, such as number of bathrooms, living area, house grade, and area excluding basement.
- Pairwise relationships between these features and the price were visualized using Seaborn's pairplot.

Model Building and Evaluation

- The data was split into training and testing sets using **train_test_split** from Scikit-learn:
    - 60%-40% split was experimented with

- A **Linear Regression** model was built using the training data.

- Model predictions on the test set were evaluated using metrics:
    - Mean Squared Error (MSE)
    - R-squared (R^2) score, indicating how well the model explains variance in house prices.

- To demonstrate model adaptability, the same linear regression approach was applied to the **Diabetes dataset** from Scikit-learn as a secondary example of regression modeling and evaluation.

Tools and Libraries Used

- **Google Colab** for code execution and interactive data analysis.

- **Python libraries:**

    - pandas for data handling

    - numpy for numerical operations

    - matplotlib and seaborn for data visualization

- sklearn (Scikit-learn) for preprocessing, imputation, model building, and evaluation.

Flowchart of Activities

Data Loading → Data Cleaning → Missing Value Insertion → Exploratory Data Analysis → Missing Value Imputation → Feature Engineering → Feature Selection → Model Training → Model Evaluation → Interpretation

5. Data Analysis and Results

1. Descriptive Analysis

- The dataset contains numeric features such as Price, Living Area, Lot Area, Number of Bedrooms, Number of Bathrooms, Built Year, and Number of Schools Nearby.

- Missing values were synthetically introduced (20% to 'Built Year' column) and then handled using various imputation techniques.

- After cleaning, columns like Date, Longitude, Renovation Year, Postal Code, Latitude, and renovation-related areas were dropped.

- **Distribution of Price:**
  The price of houses shows a **right-skewed** distribution with many extreme outliers. Most of the data points are clustered towards the lower range, but a few high-value properties create a long tail. This indicates that while most houses are moderately priced, there are a small number of significantly more expensive properties.

- **Distribution of Total Area (Living + Lot Area):**
  The distribution of **Total Area (Living + Lot Area)** seems to be highly **skewed**, with a peak close to 0 and a long tail extending outwards. This is common when a large number of houses have smaller areas.
- **Number of Schools Nearby:**
  Majority of houses are near 2 schools (~5,979 houses), followed by near 1 school (~4,600 houses) and near 3 schools (~4,041 houses). This shows that most houses are near 1, 2, or 3 schools and are in moderately well-served areas.

2. Missing Value Handling

Method	Missing Values Remaining

Dropping Missing Rows	0
Mean Imputation	0
Standard Deviation Imputation	0
Linear Interpolation	0
Polynomial Interpolation	2
KNN Imputation	0

3. Correlation Analysis

- Heatmap of feature correlations shows:
    - Top three features with strong correlation between **Area of the house (excluding basement)**:
        - living area 0.875793
        - grade of the house 0.758222
        - number of bathrooms 0.684391

4. Regression Model Performance

Train-Test Split	Mean Squared Error (MSE)	R-squared (R^2)
60%-40%	1348708953.4578605	0.053139534298155544

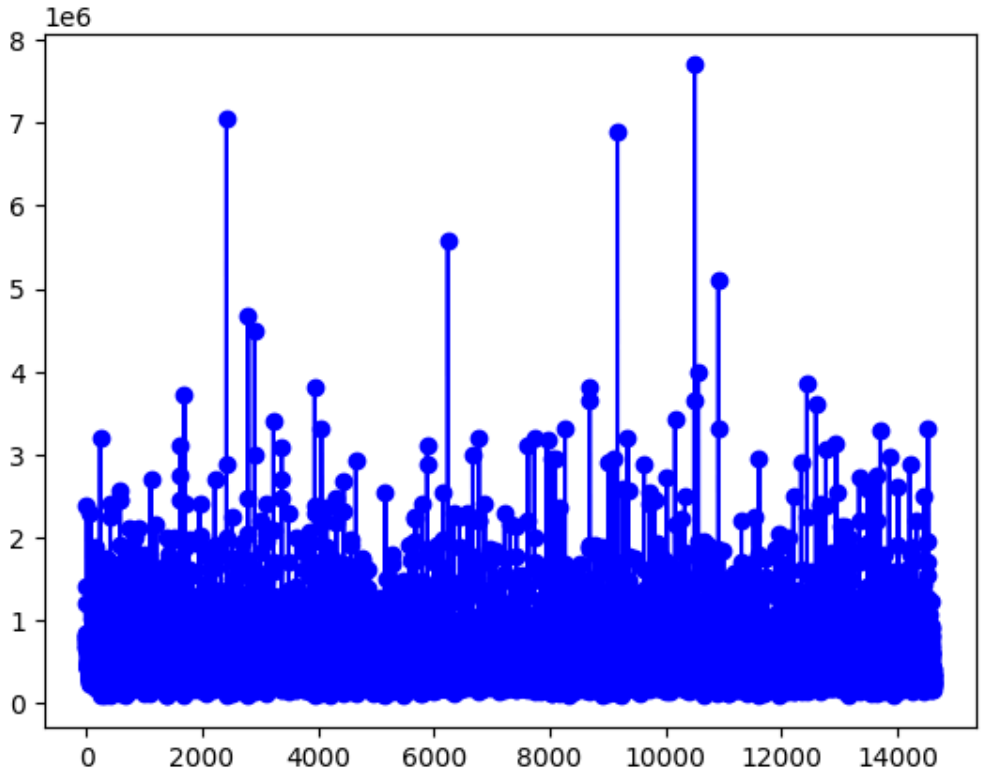
- The linear regression model shows a decent fit (R^2 around 0.053139534298155544), indicating that the selected features explain a significant portion of the variance in house prices.
- The MSE values indicate the average squared error between predicted and actual prices on the test set.

5. Additional Model on Diabetes Dataset (Comparative Analysis)

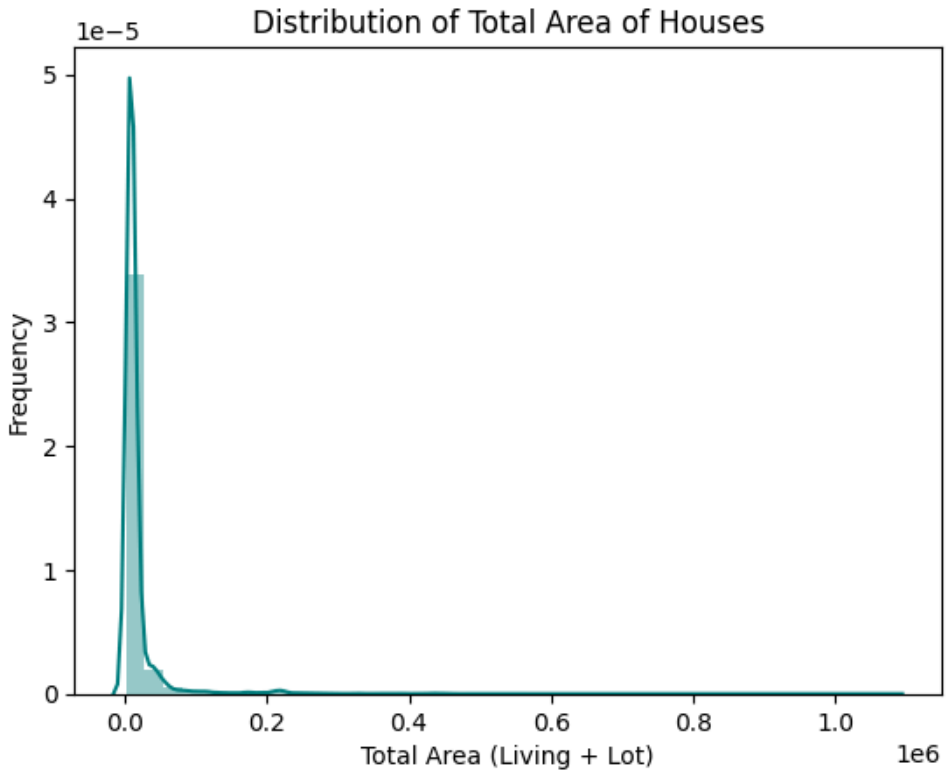
- Applied linear regression on the diabetes dataset with 80%-20% split.
- Model performance:
    - MSE: 2900.193628493482
    - R^2: 0.4526027629719195
    -

Visualizations

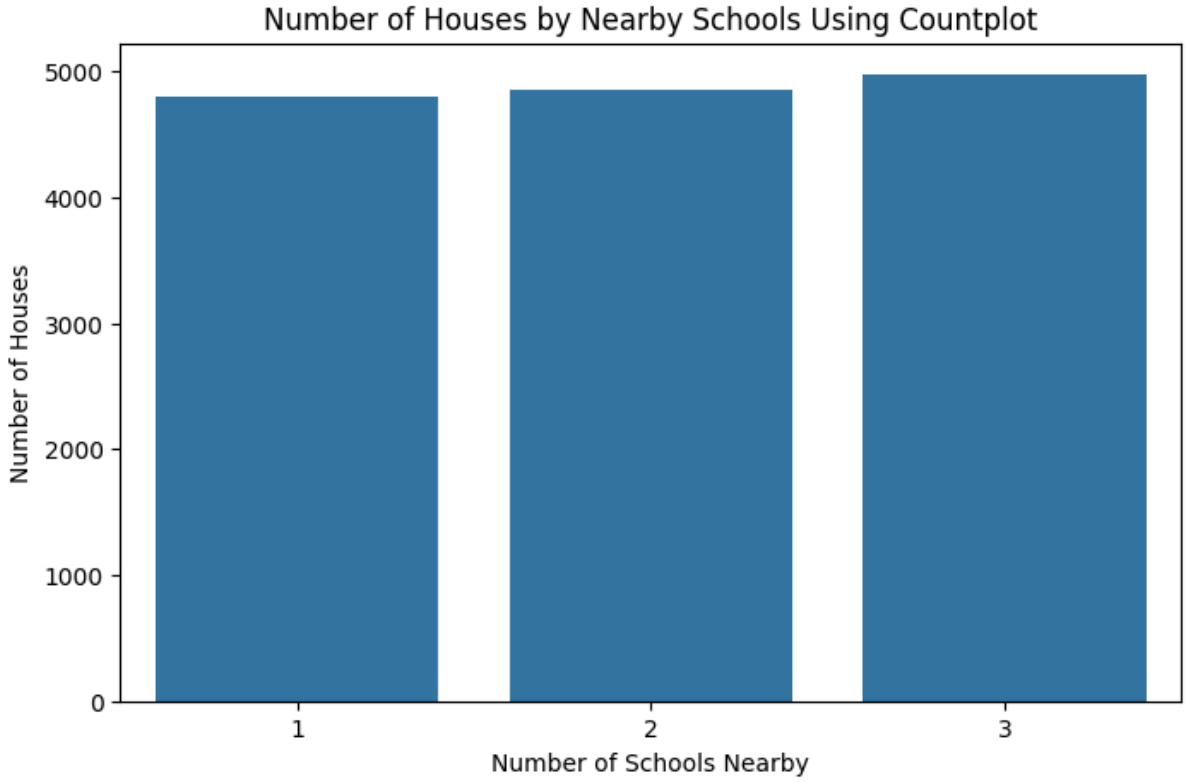
- **Price Distribution Plot**



- **Total Area Distribution Plot**

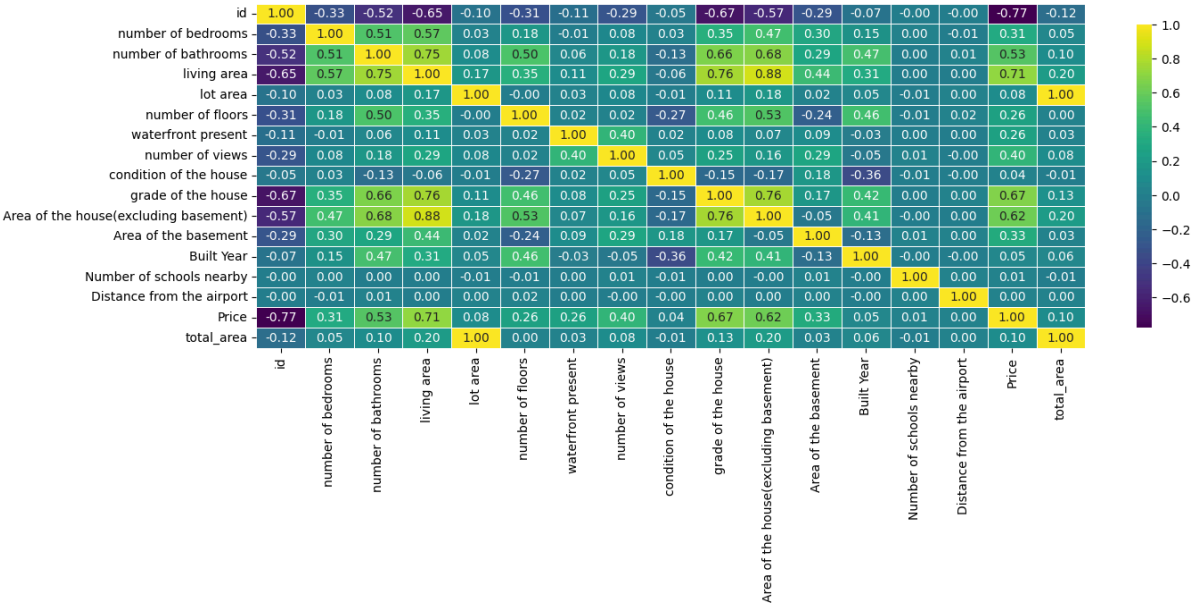


- **Number of Schools Nearby (Countplot and Distplot)**

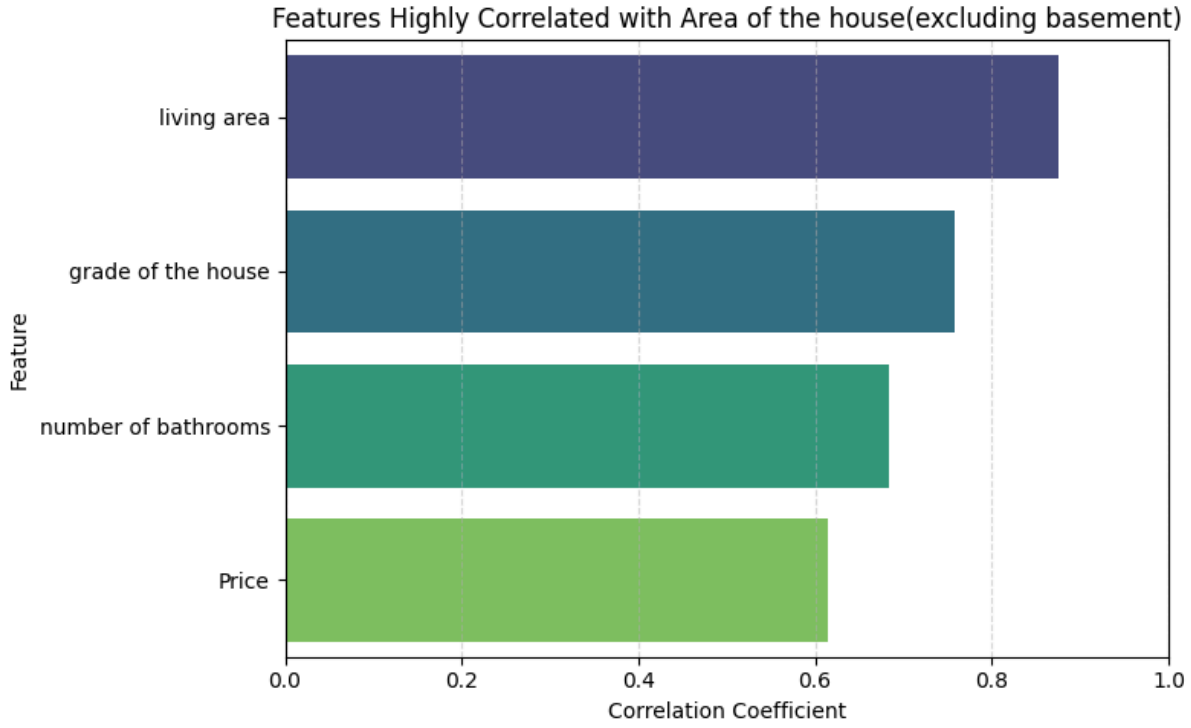


- **Correlation Heatmap**
  (Heatmap showing correlations among features)

- **Pairwise Distribution (Pairplot)**



- **Features highly correlated with 'Area of the house(excluding basement)**



6. Conclusion

This project provided hands-on experience in data cleaning, feature engineering, visualization, and linear regression modeling. While the linear regression model showed limited predictive power on the housing dataset, it performed better on the diabetes dataset. This highlights the importance of model choice and data characteristics. Future work could explore advanced models such as Random Forest, Gradient Boosting, or Neural Networks to improve prediction accuracy. Additional feature engineering and hyperparameter tuning could also enhance performance.

7. APPENDICES

Appendix A: References

a. Pandas Documentation: https://pandas.pydata.org/docs/
b. NumPy Documentation: https://numpy.org/doc/
c. Matplotlib Documentation: https://matplotlib.org/stable/contents.html
d. Seaborn Documentation: https://seaborn.pydata.org/
e. Scikit-learn Documentation: https://scikit-learn.org/stable/
f. Google Colab Documentation: https://colab.research.google.com/

Appendix B: GitHub Link:

Code, Dataset and Sample Video:
https://github.com/esho-he-boisakhh/IDEAS-TIH-Project-AUTUMN-INTERNSHIP-2025