Predicting House Prices with Supervised Learning

For

AI Project(K24MCA18P)
Session (2024-25)

Submitted by

Ram Prasann Pandey(202410116100161)
Raman Verma (202410116100162)

Ravi Kant Tiwari (202410116100164)
Rohit Kumar (202410116100172)

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of

Mr. Komal Salgotra

Assistant Professor

Submitted to

Department Of Computer Applications

KIET Group of Institutions, Ghaziabad

Uttar Pradesh-201206

(April- 2025)

Table of Contents

1. Introduction

In the current real estate market, house prices are influenced by numerous factors such as location, property size, number of rooms, and market conditions. Accurate prediction of house prices is vital for both buyers and sellers to make informed decisions. With the growth of data and machine learning, predictive modeling offers an efficient way to estimate house prices using historical data and supervised learning techniques.

2. Problem Statement

The real estate industry struggles with volatile markets and inconsistent pricing strategies. Traditional methods of property valuation can be subjective and error-prone. This project aims to build an intelligent, data-driven system that predicts house prices accurately by learning patterns from past data.

3. Objective

- To develop a predictive model using supervised learning techniques.
- To preprocess and analyze housing datasets.
- To evaluate multiple regression algorithms and compare their performances.
- To deploy the best model that can generalize well on unseen data.

.

4. Scope of the Project

- Focused on residential properties.
- Uses publicly available data from Kaggle (Ames Housing dataset).
- Models built using regression-based supervised learning algorithms.
- Does not include commercial real estate or rental predictions.

5. Literature Review

Numerous studies have addressed the house price prediction problem using various machine learning techniques. Linear Regression has traditionally been used for simplicity and interpretability. However, recent research shows that ensemble models like Random Forest and XGBoost outperform linear models in capturing non-linear relationships and interactions between features. This project draws inspiration from these findings and implements a range of algorithms for comparison.

6. Dataset Description

6.1 Source of Dataset

- **Kaggle: House Prices - Advanced Regression Techniques**
- Link: https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques

6.2 Key Features

- `LotArea`: Lot size in square feet
- `OverallQual`: Overall quality of the house
- `YearBuilt`: Year the house was built
- `GrLivArea`: Above ground living area
- `GarageCars`: Garage size (in car capacity)
- `TotalBsmtSF`: Basement size
- `FullBath`, `BedroomAbvGr`, `KitchenQual`: Utilities and features

6.3 Target Variable

- `SalePrice`: The actual sale price of the house (our prediction target)

7. Tools and Technologies Used

- **Programming Language**: Python
- **Libraries**:
  - *Pandas* for data manipulation
  - *NumPy* for numerical computation
  - *Matplotlib/Seaborn* for visualization
  - *Scikit-learn* for machine learning models
  - *XGBoost* for gradient boosting
- **Environment**: Jupyter Notebook / Google Colab

8. Methodology

8.1 Data Collection

The dataset was downloaded from Kaggle and loaded into the Python environment using Pandas.

8.2 Data Preprocessing

- Missing values handled using mean/median imputation and mode for categorical variables.
- Categorical variables converted using one-hot encoding.
- Outliers detected using boxplots and treated.
- Feature scaling using StandardScaler.

8.3 Exploratory Data Analysis (EDA)

- Correlation matrix used to find relationships between features.
- Histograms and boxplots used for distribution analysis.
- Scatter plots to observe trends between variables like `GrLivArea` and `SalePrice`.

8.4 Feature Selection & Engineering

- Removed features with low correlation to `SalePrice`.
- Combined similar features (e.g., `TotalBsmtSF + 1stFlrSF` as `TotalHouseArea`).
- Log transformation on skewed features to improve normality.

8.5 Model Selection

The following models were considered:

- Linear Regression
- Ridge and Lasso Regression
- Decision Tree Regressor
- Random Forest Regressor
- XGBoost Regressor

8.6 Model Training

Data split into training (80%) and testing (20%) sets using `train_test_split`.

Cross-validation used to validate model performance and avoid overfitting.

8.7 Model Evaluation

Performance metrics:

- **Root Mean Squared Error (RMSE)**
- **Mean Absolute Error (MAE)**
- **R^2 Score**

9. Result Analysis

Model	**RMSE**	**MAE**	**R² Score**
Linear Regression	34,520	25,000	0.83
Ridge Regression	32,890	22,780	0.85
Random Forest Regressor	27,450	19,000	0.89
XGBoost Regressor	**24,900**	**17,320**	**0.91**

- **XGBoost** gave the best performance across all evaluation metrics.
- Linear models were easy to interpret but performed worse on complex relationships.

10. Model Comparison

- **Linear Models**: Best for quick baseline models, suffer with non-linearity.
- **Tree-based Models**: Better handle outliers and non-linearities.
- **Ensemble Models (XGBoost)**: Capture complex interactions, low bias and variance.

11. Conclusion

The project demonstrated that machine learning can effectively predict house prices using supervised learning. The XGBoost Regressor outperformed all other models, proving highly accurate in estimating property values. With proper preprocessing, feature engineering, and evaluation, predictive models can be deployed for real-world use cases.

12. Future Work

- Add geospatial features using APIs like Google Maps.
- Create a web interface using Flask or Django.
- Integrate real-time data pipelines.
- Try deep learning models (e.g., Neural Networks).

13. References

- Kaggle Housing Dataset
- Scikit-learn Documentation: https://scikit-learn.org
- XGBoost Documentation: https://xgboost.readthedocs.io
- Python Official Docs

14. Appendix

- Python code snippets used for data preprocessing and modeling.
- Full correlation heatmap
- Visuals and plots (feature importance, residual plots, etc