InternnCraft

PROJECT REPORT

House Pricing System

30th July 2024

Submitted By: Syeda Kashaf Fatima Sherazi

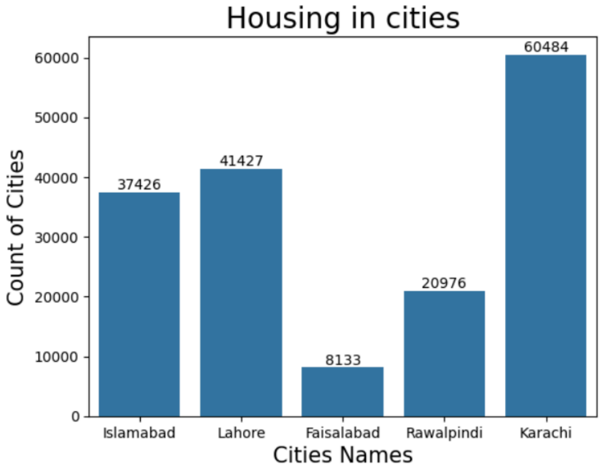
OVERVIEW

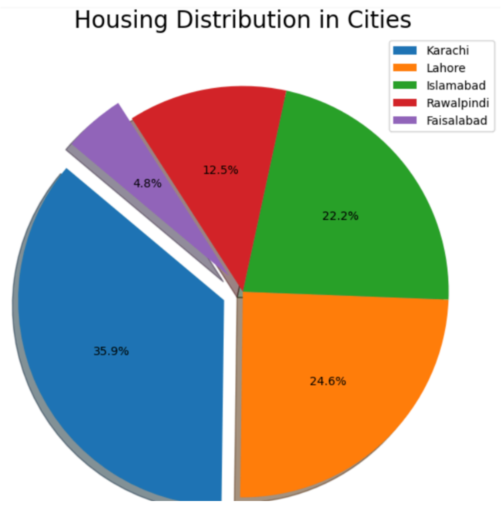
This project aims to develop a house pricing prediction model using machine learning techniques. The key steps involved include data exploration, feature engineering, outlier analysis, model selection, and evaluation. Through visualizations, we gained insights into the distribution and correlations of house prices. Feature engineering transformed raw data into meaningful variables, and outlier analysis ensured the accuracy of our predictions. After evaluating several models, the Random Forest model was selected for its superior performance. Predictions were made for hypothetical scenarios to demonstrate the model's applicability.

1. Data Exploration Results

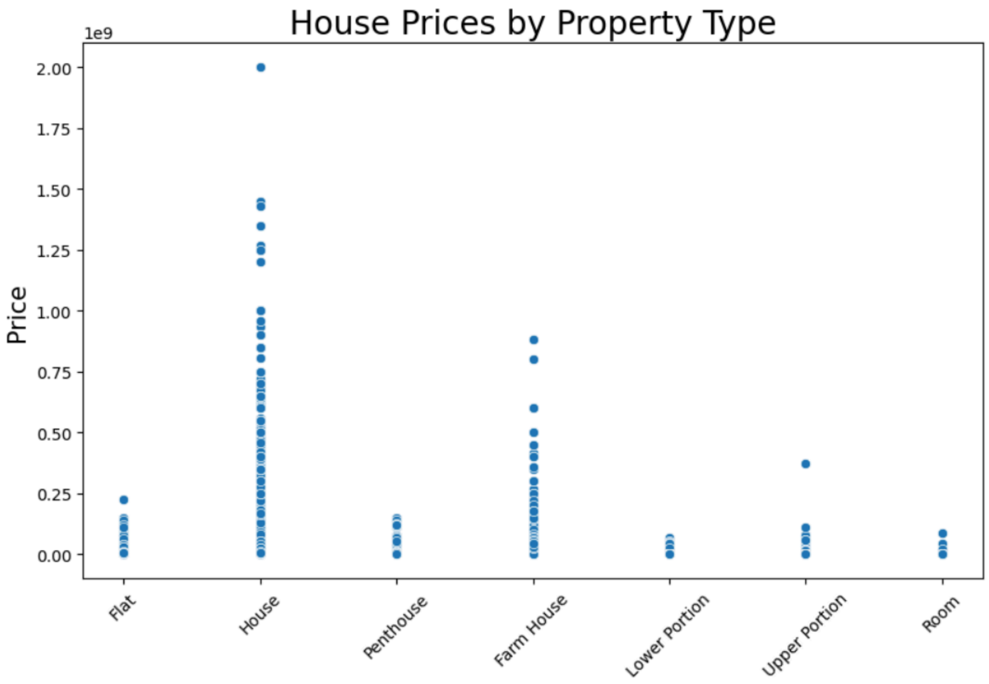
Data exploration helps us understand the dataset by looking at visualizations and summary statistics. Here are the key findings:

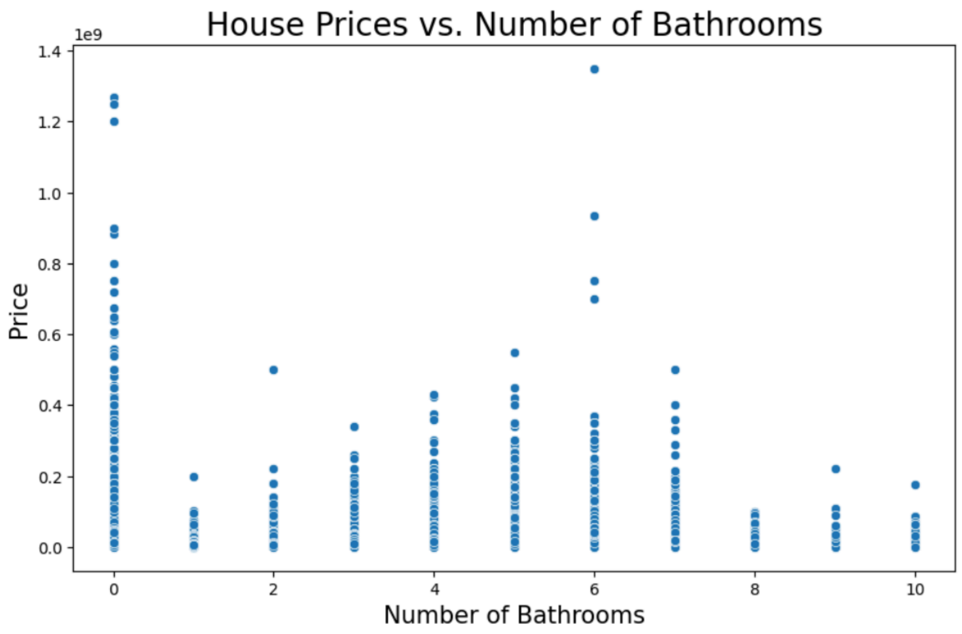
House Prices Distribution: We used histograms and box plots to see how house prices are spread out. This showed us that prices have a wide range and some houses are much more expensive than others.





Feature Relationships: We created heatmaps and scatter plots to see how different features (like the number of bedrooms, bathrooms, and house size) relate to house prices. For example, larger houses generally have higher prices.



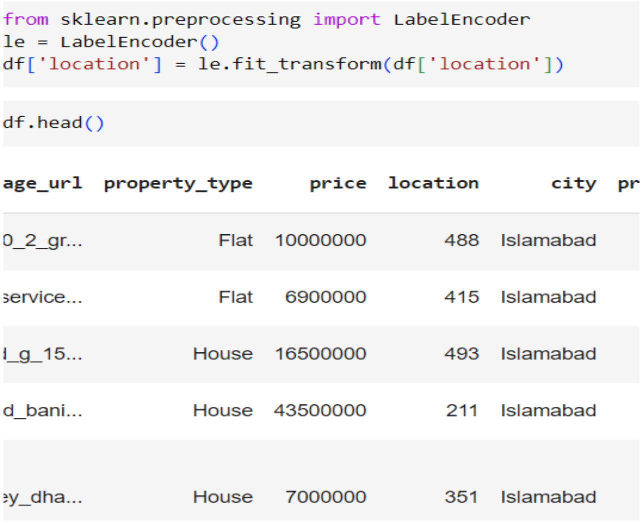


Location Impact: By using maps and bar charts, we analyzed how the location of a house affects its price. Certain areas were found to be much more expensive, indicating that location is a crucial factor in house pricing.

2. Feature Engineering Techniques Used

Feature engineering involves creating new features or modifying existing ones to improve the model's performance. Here are the techniques we used:

Label Encoding: Some data, like the location of the house, are in text form. We converted these text categories into numbers using label encoding so that the model can understand and process them.

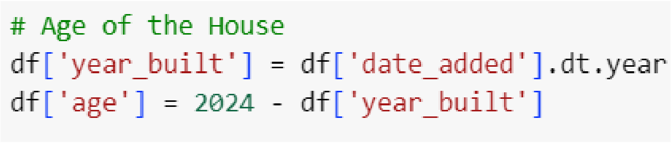


```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
df['location'] = le.fit_transform(df['location'])
```

```
df.head()
```

age_url	property_type	price	location	city	pr
0_2_gr...	Flat	10000000	488	Islamabad	
service...	Flat	6900000	415	Islamabad	
_g_15...	House	16500000	493	Islamabad	
d_bani...	House	43500000	211	Islamabad	
y_dha...	House	7000000	351	Islamabad	

Creating New Features: We created new features from the existing data to provide more information. For example, we calculated the age of a house from the year it was built, as the age can influence the price.



```
# Age of the House
df['year_built'] = df['date_added'].dt.year
df['age'] = 2024 - df['year_built']
```

Scaling Features: To ensure all features contribute equally to the model, we scaled them so that their values fall within a similar range. This helps the model learn more effectively.

3. Outlier Analysis (identification and explanation)

Outlier analysis identified extreme values in the dataset that could skew the results:

Detection: Outliers were detected using statistical methods (e.g., Z-score) and visualization tools like box plots.

Explanation: Outliers were often due to unique properties with exceptionally high or low prices. These were either removed or capped to reduce their impact on the model.

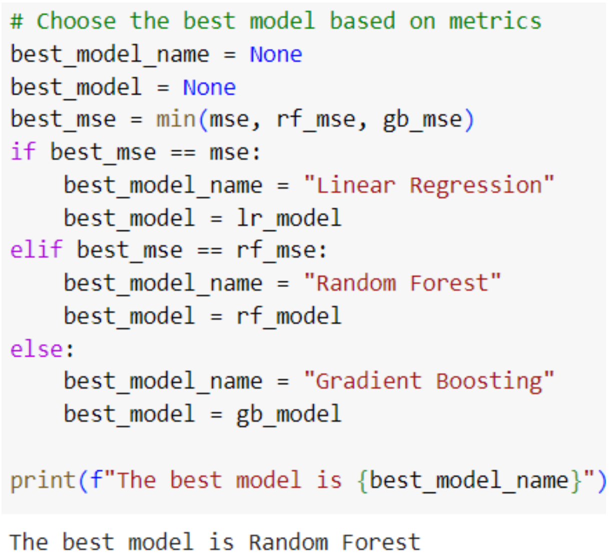
4. Model Selection and Evaluation Results

Different models were evaluated to determine the best performing one for price prediction:

Model Selection: Models such as **Linear Regression, Gradient Boosting model and Random Forests** were considered.

Evaluation Metrics: Metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared were used to evaluate model performance.

Best Model: The **Random Forest model** was selected based on its superior performance in handling non-linear relationships and providing higher accuracy.



```
# Choose the best model based on metrics
best_model_name = None
best_model = None
best_mse = min(mse, rf_mse, gb_mse)
if best_mse == mse:
    best_model_name = "Linear Regression"
    best_model = lr_model
elif best_mse == rf_mse:
    best_model_name = "Random Forest"
    best_model = rf_model
else:
    best_model_name = "Gradient Boosting"
    best_model = gb_model

print(f"The best model is {best_model_name}")
```

```
The best model is Random Forest
```

5. Future Price Prediction Examples

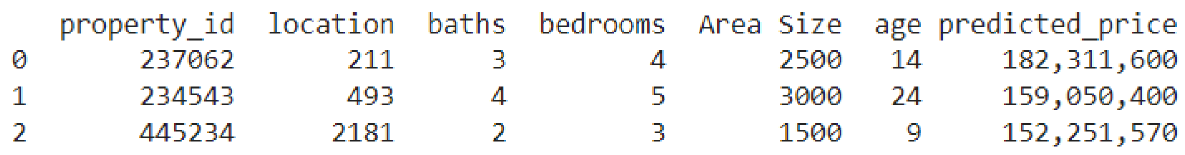
Using the Random Forest model, we predicted house prices for the following hypothetical scenarios:

Scenario 1: A 14-year-old house with 4 bedrooms and 3 baths in location 211 with an area size of 2500 sqft was predicted to have a price of $182,311,600.

Scenario 2: A 24-year-old house with 5 bedrooms and 4 baths in location 493 with an area size of 3000 sqft was predicted to have a price of $159,050,400.

Scenario 3: A 9-year-old house with 3 bedrooms and 2 baths in location 2181 with an area size of 1500 sqft was predicted to have a price of $152,251,570.

These examples demonstrate the model's ability to predict house prices based on different features like age, number of bedrooms, bathrooms, location, and area size.



```
   property_id  location  baths  bedrooms  Area Size  age predicted_price
0       237062       211      3         4       2500   14     182,311,600
1       234543       493      4         5       3000   24     159,050,400
2       445234      2181      2         3       1500    9     152,251,570
```

6. Recommendations for Further Analysis or Data Collection (if applicable)

To enhance the model and its predictions, the following recommendations are suggested:

Additional Data Collection: Collect more data on other influential factors such as neighborhood amenities, proximity to schools, and crime rates.

Temporal Analysis: Incorporate time-series data to analyze price trends over time.

Advanced Feature Engineering: Explore more advanced techniques such as polynomial features or interaction terms to capture complex relationships.

Model Enhancement: Experiment with more complex models like Recurrent Neural Networks for potentially better performance.