Housing Analysis

How to Provide Accurate Housing Predictions to Everyday People

Project Overview

- Housing availability and pricing has been cause of concern across the United States
- Record high interest rate and home evaluation is making it difficult on both sellers and buyers
- The purpose of this project is to evaluate the key attributes that cause pricing to increase/decrease and provide accurate predictions to stakeholders

Dataset

- The dataset that is being evaluated is from Kaggle, which includes home attributes that are important to buyers/ sellers
- After the first initial analysis, I have had to feature engineer the dataset in order to properly evaluate the model accuracy

	price	area	bedrooms	bathrooms	stories	mainroad	guestroom	basement	hotwaterheating	airconditioning	parking	prefarea	furnishingstatus
0	13300000	7420	4	2	3	yes	no	no	no	yes	2	yes	furnished
1	12250000	8960	4	4	4	yes	no	no	no	yes	3	no	furnished
2	12250000	9960	3	2	2	yes	no	yes	no	no	2	yes	semi-furnished
3	12215000	7500	4	2	2	yes	no	yes	no	yes	3	yes	furnished
4	11410000	7420	4	1	2	yes	yes	yes	no	yes	2	no	furnished

Data Feature Engineering

- Created a separate dataframe with feature engineered attributes to make sure I was not changing the original data
- New features include:
  - Price per square foot
  - Total rooms
  - Multiple stories
  - Fully equipped
  - Log Price

[15]:
```
#Feature engineering original dataset to try to get higher accuracy in model

df_fe = df.copy()

#  Price per square foot
df_fe['price_per_sqft'] = df_fe['price'] / df_fe['area']

#  Total rooms (bedrooms + bathrooms)
df_fe['total_rooms'] = df_fe['bedrooms'] + df_fe['bathrooms']

# Has multiple stories
df_fe['multi_story'] = df_fe['stories'].apply(lambda x: 1 if x > 1 else 0)

# Is fully equipped (hot water heating + AC + parking >= 2)
df_fe['fully_equipped'] = df_fe.apply(
    lambda row: 1 if (row['hotwaterheating'] == 'yes' and
                      row['airconditioning'] == 'yes' and
                      row['parking'] >= 2) else 0, axis=1)

#  Log-transformed price to reduce skew
import numpy as np
df_fe['log_price'] = np.log1p(df_fe['price'])

# Preview the new features
df_fe[['price', 'area', 'price_per_sqft', 'total_rooms', 'multi_story', 'fully_equipped', 'log_price']].head()
```

[15]:

	price	area	price_per_sqft	total_rooms	multi_story	fully_equipped	log_price
0	13300000	7420	1792.452830	6	1	0	16.403275
1	12250000	8960	1367.187500	8	1	0	16.321037
2	12250000	9960	1229.919679	5	1	0	16.321037
3	12215000	7500	1628.666667	6	1	0	16.318175
4	11410000	7420	1537.735849	5	1	0	16.250001

Data Modeling & Evaluation

Two Models were created and evaluated by accuracy: Random Forest Regressor & XGB Regressor

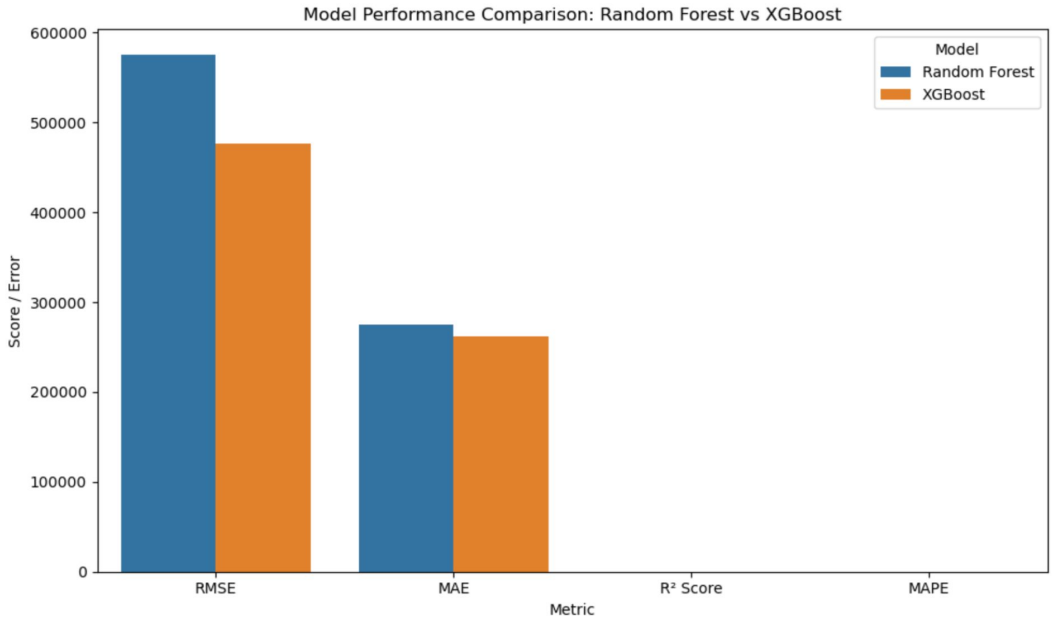
Random Forest Evaluation

```
RMSE: 566,682
MAE: 275,653
R² Score: 0.936
MAPE: 5.22%
```

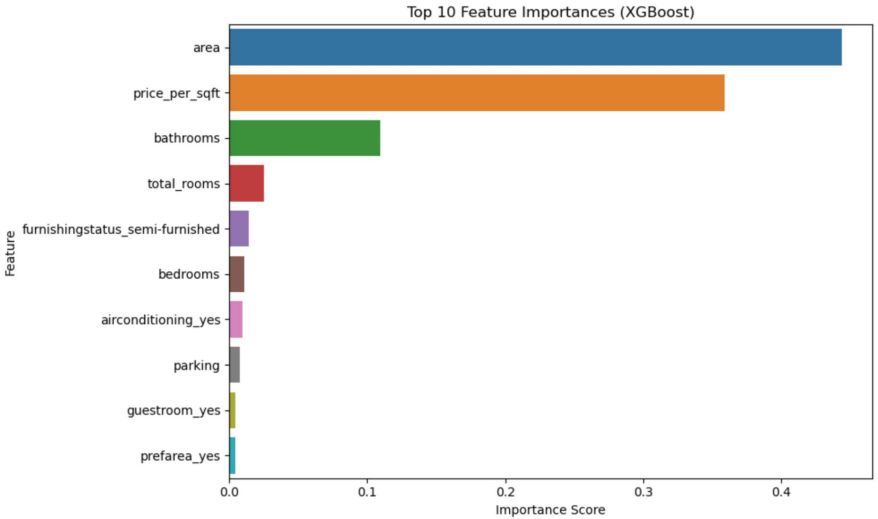
XGB Regressor Evaluation

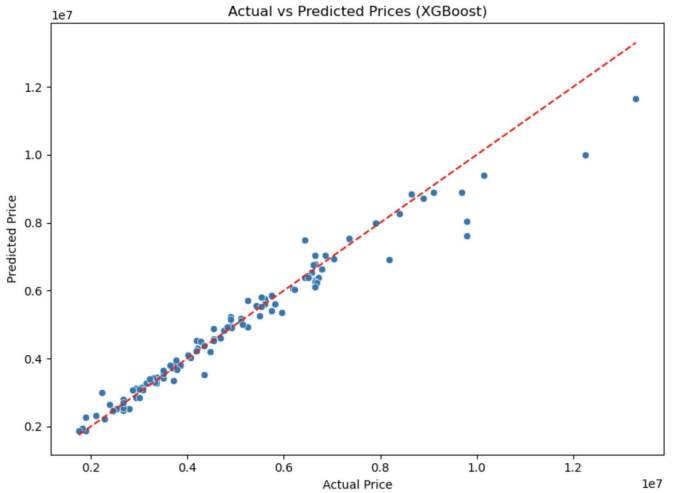
```
XGBoost RMSE: 476,155
XGBoost R² Score: 0.955
XGBoost MAE: 262,256
XGBoost MAPE: 4.83%
```

Model Comparison



Key Visuals: XGB Model





Conclusion

- The XGB was the more accurate model when it comes to predicting housing
- The model was able to provide insight on the top 10 features
- The actual vs predicted linear regression demonstrates accuracy throughout all price points
- This project was successful in able to provide key features and pricing for key stakeholders

Sources

https://www.kaggle.com/datasets/yasserh/housing-prices-dataset

H, M. Y. (2022, January 12). *Housing prices dataset*. Kaggle.

https://www.kaggle.com/datasets/yasserh/housing-prices-dataset