Data Source AND Data Ingestion

Files used:
train.csv -> contains features + SalePrice
test.csv -> contains features only

Data describes residential properties in Ames, Iowa

Key Steps:

Loaded both datasets using Pandas
Standardized column names (lowercase, trimmed)
Added data_source column (train / test)

Data Cleaning & Missing Values

01 Merged train & test into one dataset

02 Removed duplicates

03 Dropped id column (no analytical value)

04 Missing Values

- Replaced invalid values (NA, N/A, empty) ->NaN
- Categorical features where missing = not available
  Filled with "None"
- Numerical features where missing = zero
  Filled with 0
  Examples: GarageArea, PoolArea
- (LotFrontage)->Filled using Neighborhood median

Numerical → median
Categorical → mode

Feature Engineering

Computed using train data only (to avoid data leakage)

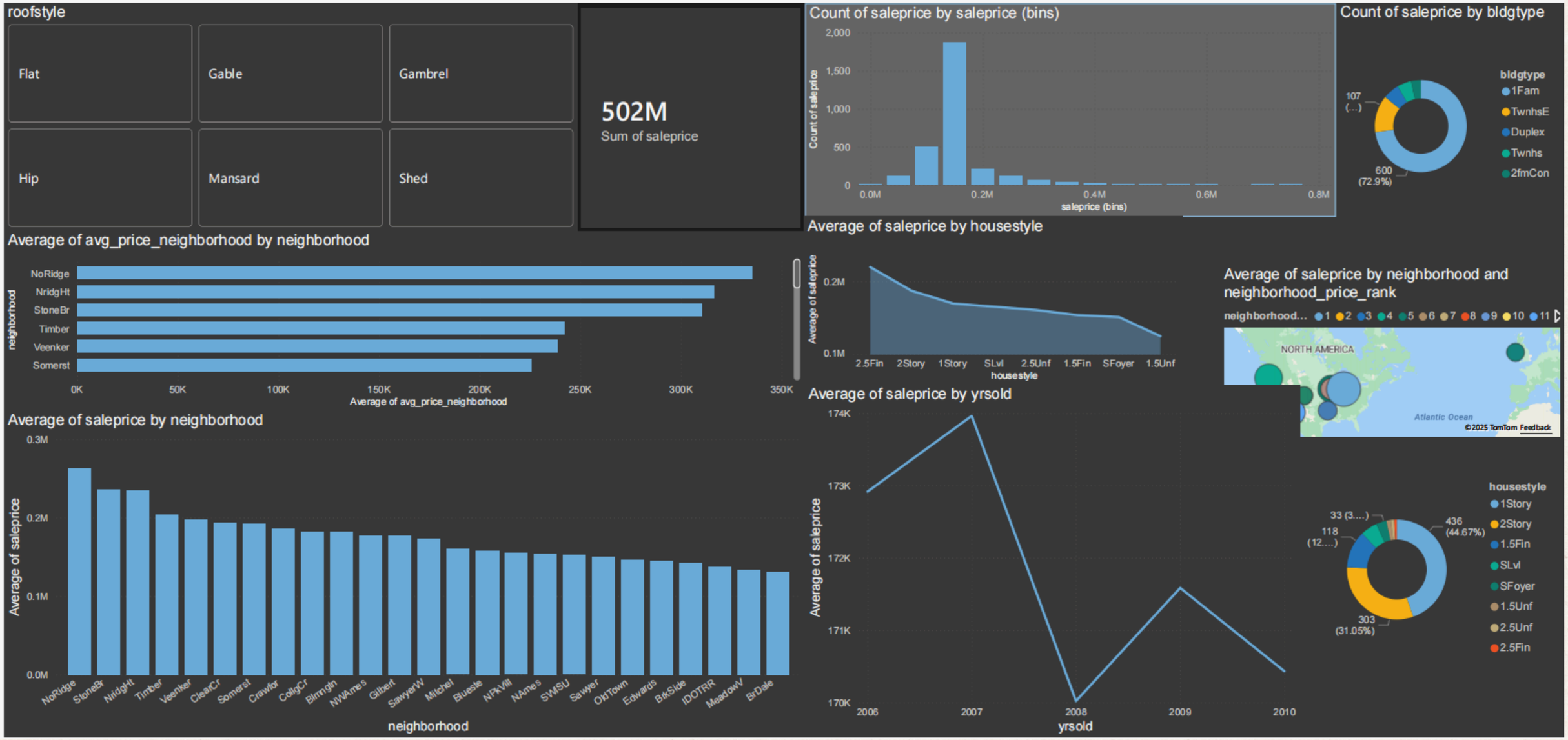
New Features:

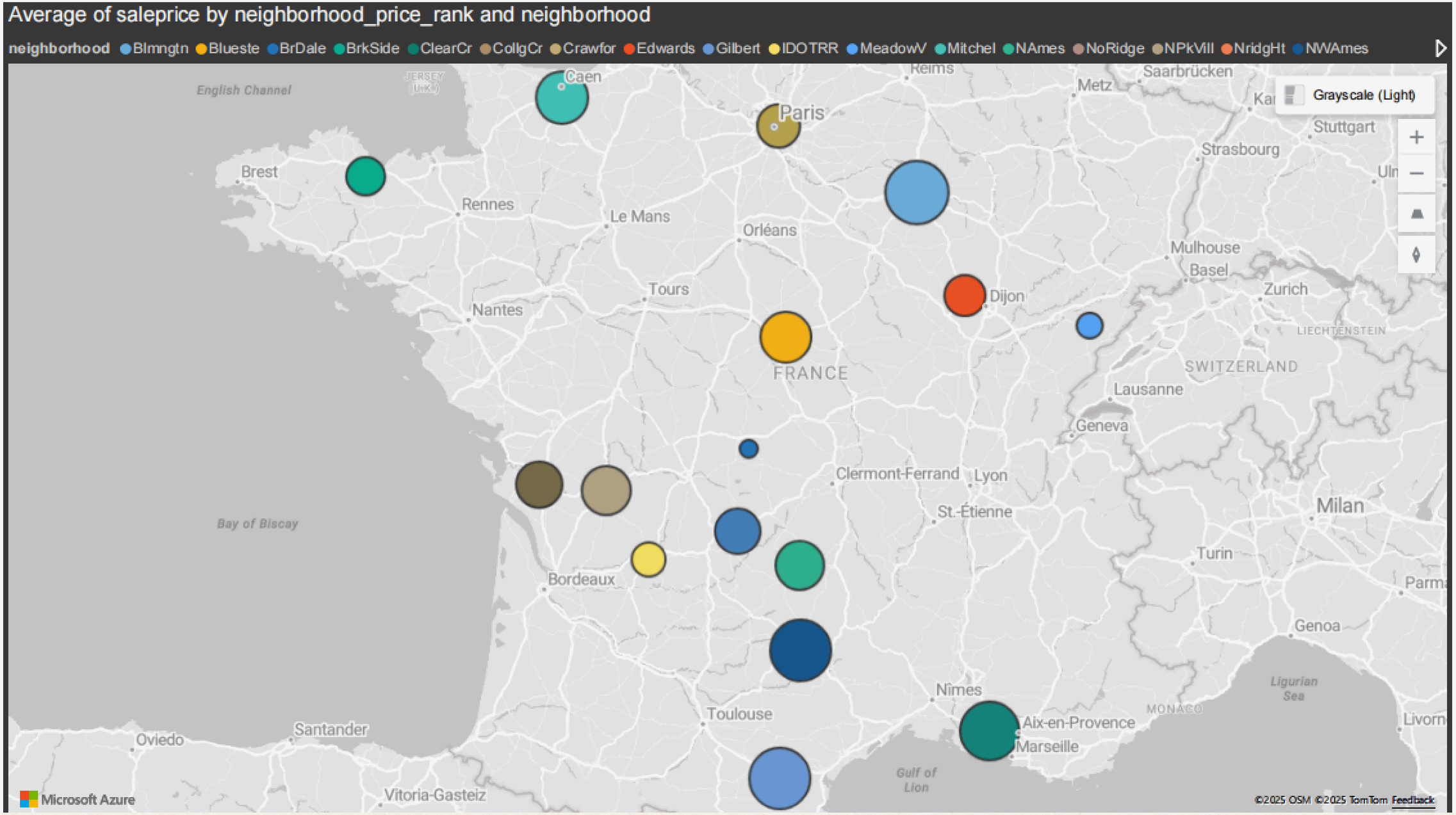
- avg_price_neighborhood
- median_price_neighborhood
- neighborhood_property_count
- neighborhood_price_rank
- relative_price_to_neighborhood

Why Important?

- Enables fair price comparison
- Captures neighborhood market behavior
- Improves analysis AND modeling quality

Visualization




Average of saleprice by neighborhood_price_rank and neighborhood
neighborhood Blmngtn Blueste BrDale BrkSide ClearCr CollgCr Crawfor Edwards Gilbert IDOTRR MeadowV Mitchel NAmes NoRidge NPkVill NridgHt NWAmes
Grayscale (Light)
English Channel
Caen
Paris
Reims
Metz
Saarbrücken
Stuttgart
Strasbourg
Brest
Rennes
Le Mans
Orléans
Mulhouse
Basel
Zurich
Tours
Nantes
Dijon
LIECHTENSTEIN
SWITZERLAND
FRANCE
Lausanne
Geneva
Clermont-Ferrand
Lyon
St.-Étienne
Milan
Turin
Bay of Biscay
Bordeaux
Genoa
Nîmes
Ligurian Sea
Toulouse
Aix-en-Provence
Marseille
MONACO
Oviedo
Santander
Gulf of Lion
Vitoria-Gasteiz
Microsoft Azure
©2025 OSM ©2025 TomTom Feedback

System Overview

- The goal of this mini project is to build a machine learning system that predicts property prices.
- The system supports real estate investment decisions.
- It combines:
  - Price prediction
  - Demand-aware pricing
  - Investment scoring

Models Used:
Linear Regression
Random Forest

Evaluation Metrics:
MAE
RMSE
R^2 Score

- R^2 score was close to zero for both models.
- Error values were high.

- Models failed to learn meaningful relationships.
- The issue was data representation, not model choice.

Gradient Boosting Regressor : Highest R^2 (~0.99) and Lowest error values

Feature Engineering Impact and Feature Importance

Engineered Features:

Total Area

Age of House

Quality Score

Neighborhood-based pricing features

After :

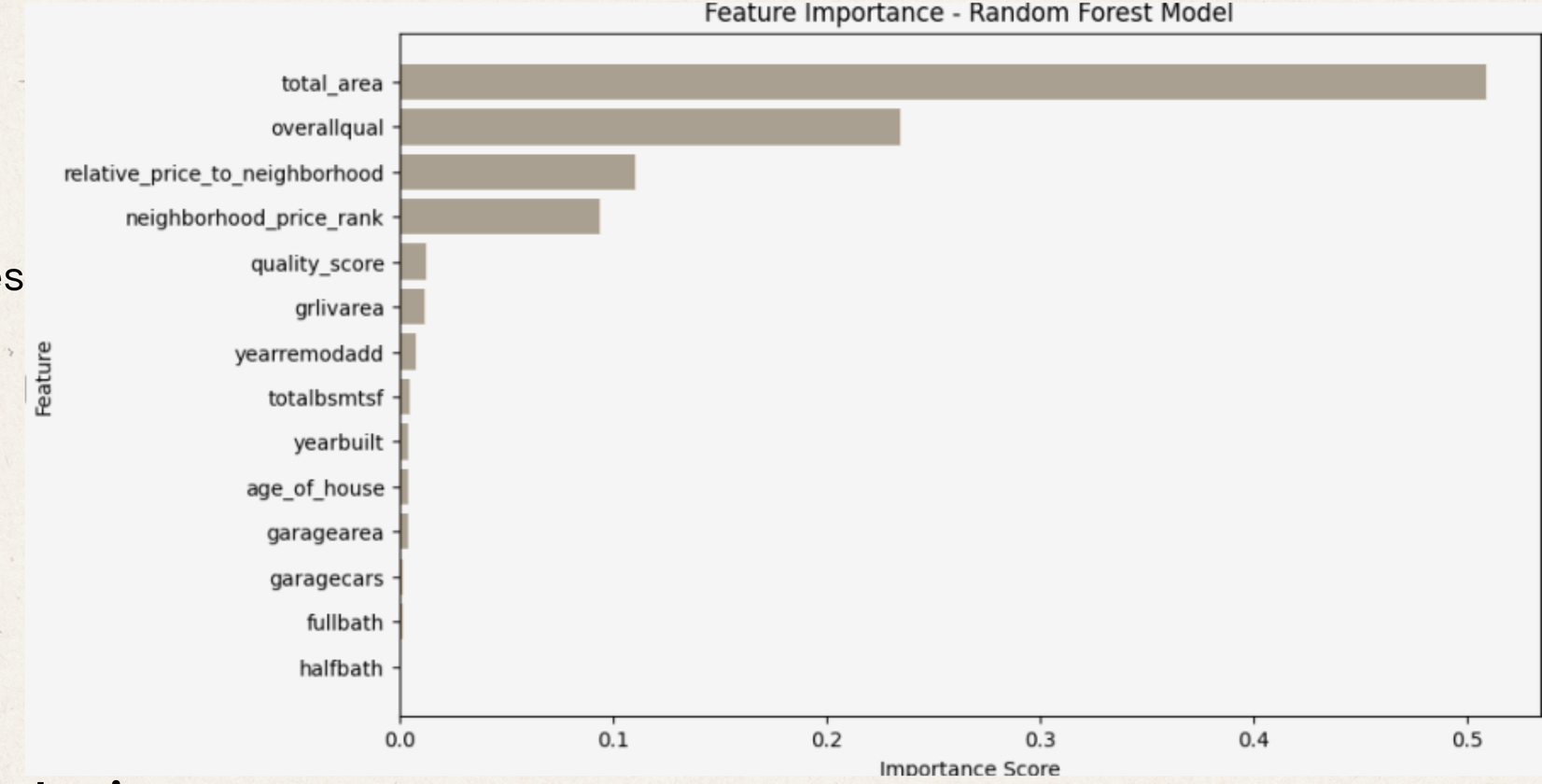
Linear Regression:

$R^2 \approx 0.89$

Random Forest:

$R^2 \approx 0.96$

Much lower MAE and RMSE



Models started capturing real estate behavior.

Demand Index(not applied in the project ,used for scenario testing)

Machine learning predicts a fair base price.
Real estate markets are dynamic, not static.
Prices change based on:

- Market demand
- Competition
- Timing

Dynamic pricing allows the system to adapt to market conditions.

The system uses a Demand Index as a business input.
Pricing adjustment rules:
High demand → Increase price
Low demand → Decrease price
Normal demand → Keep price unchanged
Final price = Predicted Price + Market Adjustment

Base predicted price = $100,000
High demand market
Buyers compete more
Final price = $110,000
Low demand market
Fewer buyers
Final price = $90,000
Stable market
No pressure
Final price = $100,000

Demand Index In this project, it is treated as a business-driven input, not a learned feature.

Investment Scoring

Investment Score ranging from 0 to 100.

Convert predicted prices into actionable investment insights.

Inputs :

Predicted Price – from ML model

Neighborhood Average Price

Property Quality

Property Age

```
Investment Score = 0.4 * (predicted / neighborhood avg)
                 + 0.3 * normalized quality
                 + 0.3 * normalized age
```

Score :

80–100 → Excellent Investment

60–79 → Good Investment

40–59 → Moderate Investment

Below 40 → High Risk

Example:

Property price = $208,500

Neighborhood avg = $197,965

Score = 91 -> Excellent investment