# Fireplace Count Prediction Using KNN and MV Linear Regression

Ram Sunder

On 7th April 2025

# Table of Content



# Understanding Dataset

## Initial Observations from the Dataset

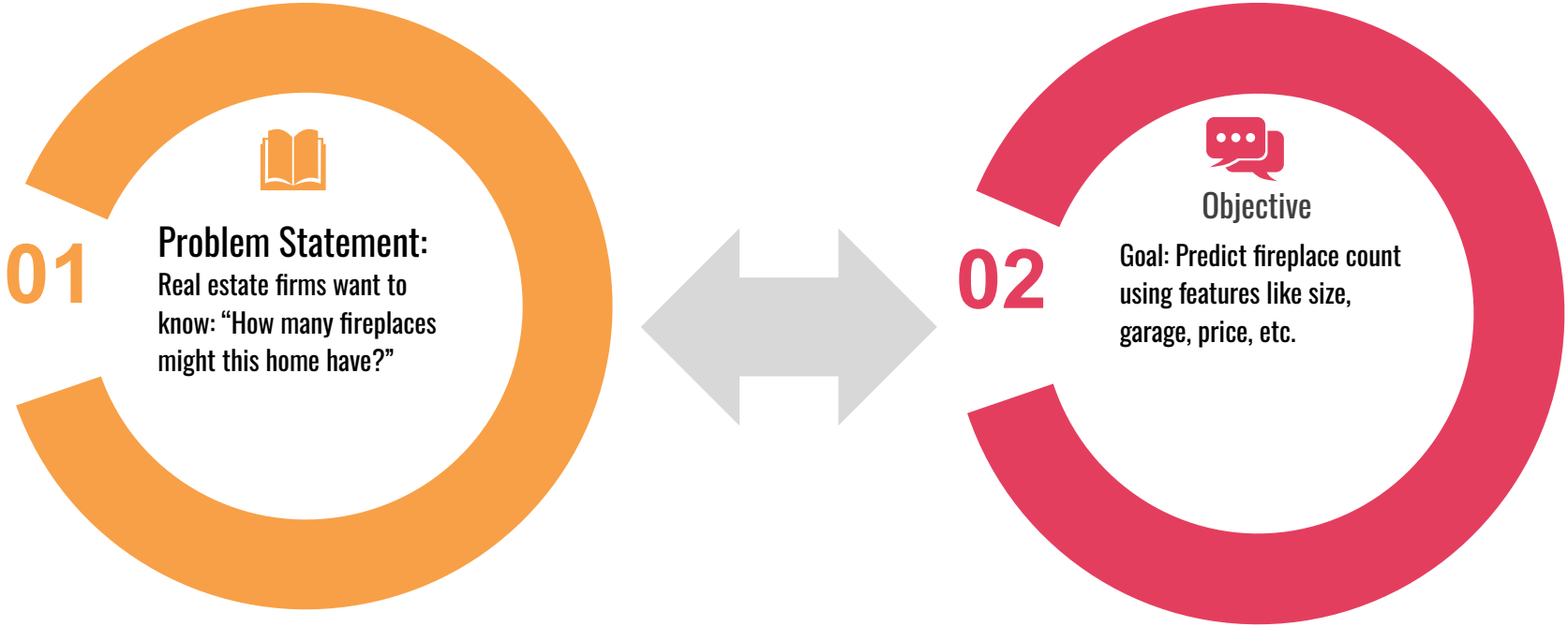
Total Records: 5,000 rows

Columns: 16

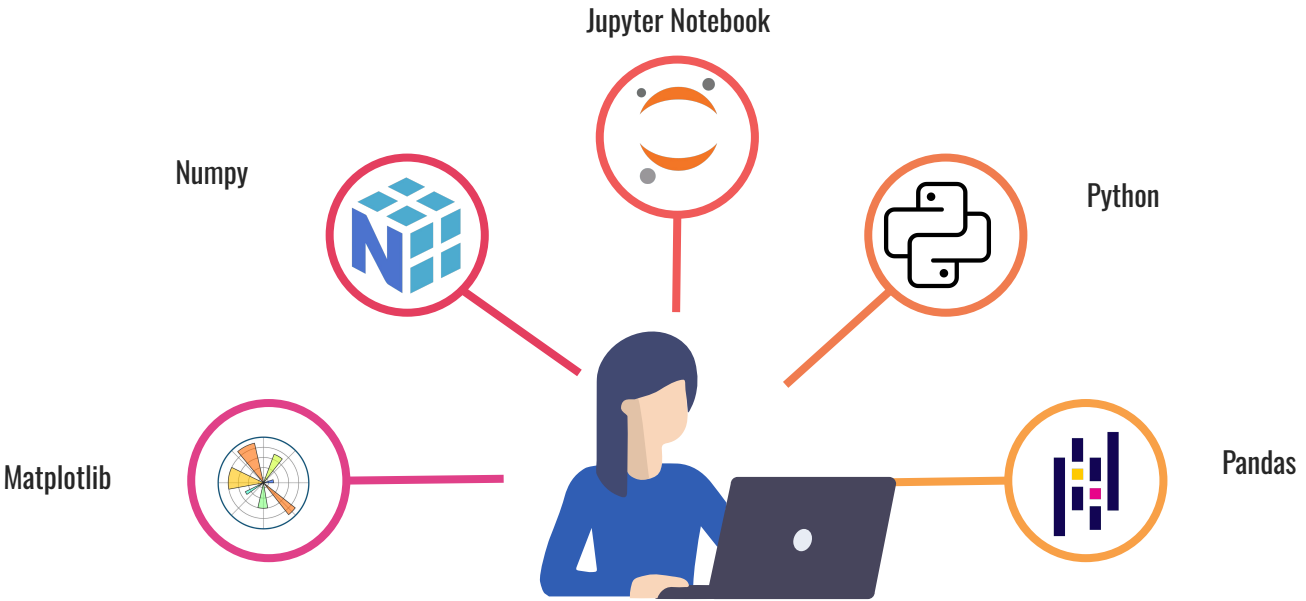
Data Types: Numerical: MLS, sold_price, zipcode, longitude, latitude, lot_acres, taxes, year_built, bedrooms, fireplaces,bathrooms,garage,sqrt_ft
Data Types: Categorical (Objects): kitchen_features, floor_covering, HOA

Missing Values:
lot_acres (10 missing)
fireplaces (25 missing)
sqrt_ft, garage, and HOA have None values (Need conversion to NaN)

# Potential Issues to Address



# Technology used in EDA



# Exploratory Data Analysis (EDA)

Overview of EDA: EDA is a crucial step in understanding the dataset and its underlying patterns before building any machine learning models.

## Missing Values

Identify missing values in the dataset.

## Summary Statistics

Analyze mean, median, and standard deviation for numerical variables.

## Data Distribution:

Visualize the distribution of key features (e.g., sold_price, lot_acres, etc.).

Outliers Handling: Mention any outliers that were identified and handled (if applicable).

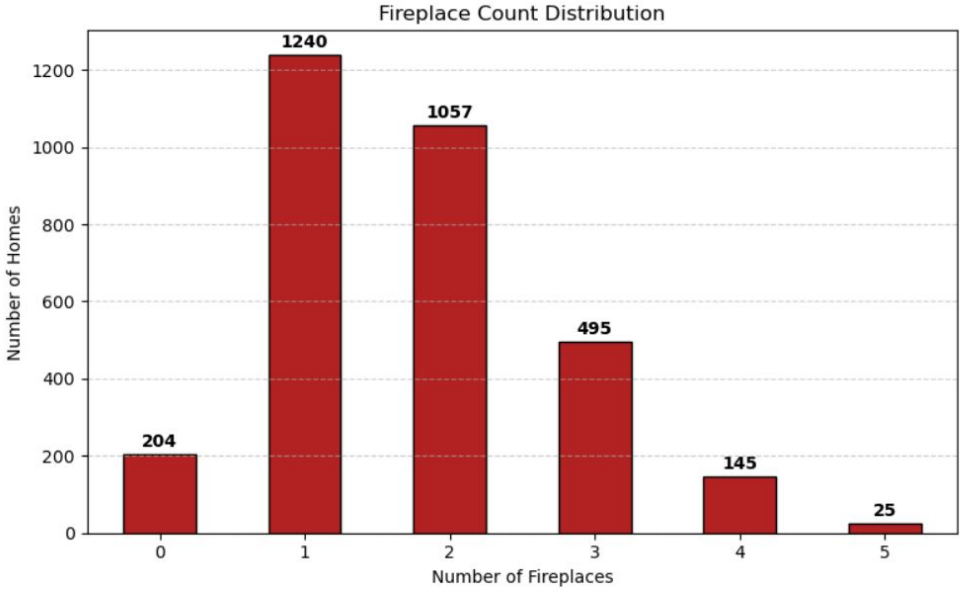
Data Transformation: Explain any transformation, such as creating new features

# Target Variable Creation:

Target: fireplaces (0 to 5)

Features used:

- sold_price, zipcode, sqrt_ft, lot_acres, year_built, bedrooms, bathrooms, garage
- ~3166 cleaned row



# Implementation Flow

Cleaned data → Removed outliers

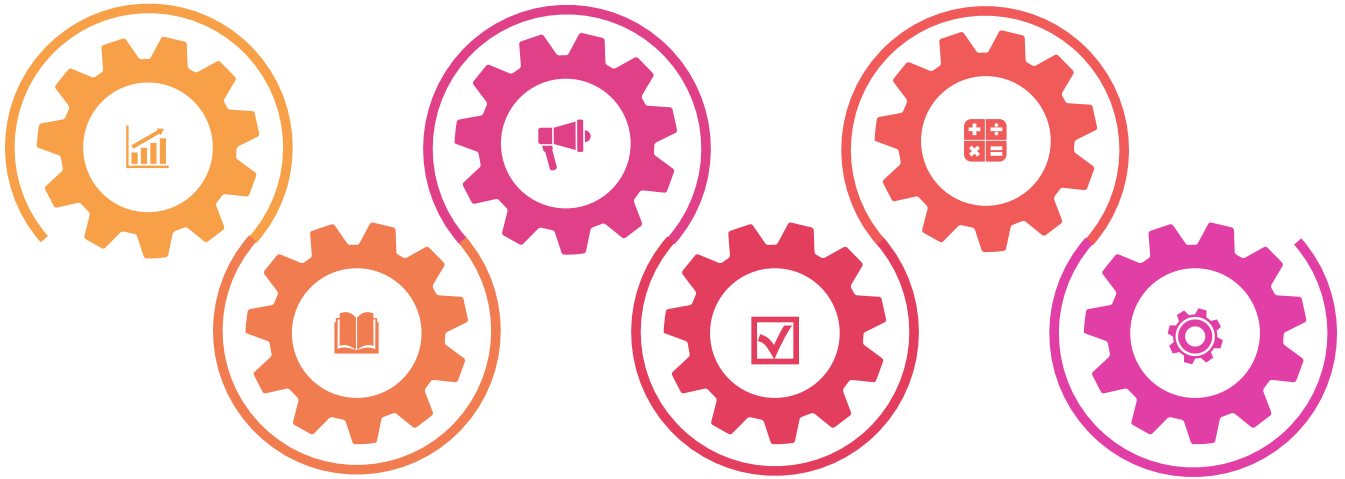
Split into train/test

Standardized features (important for MV Linear)

Trained both models

Evaluated using:
MAE (Mean Absolute Error)
$R^2$ Score (closer to 1 is better)

Visualized results with:
Residuals histogram (bell curve)
Actual vs predicted scatter
MAE/$R^2$ comparison bar chart



# Models Used

## A. KNN Regressor (Custom Implementation)

- Predicts based on similar nearby homes
- No assumption of linearity — great for real-world, uneven data

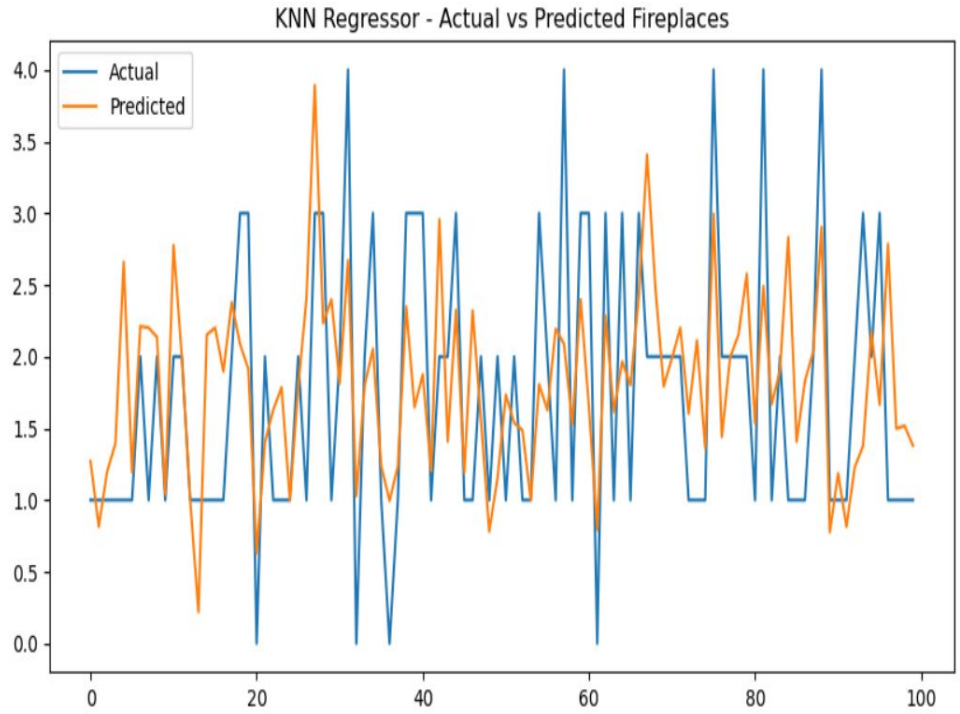
## B. Multivariate Linear Regression (Custom SGD version)

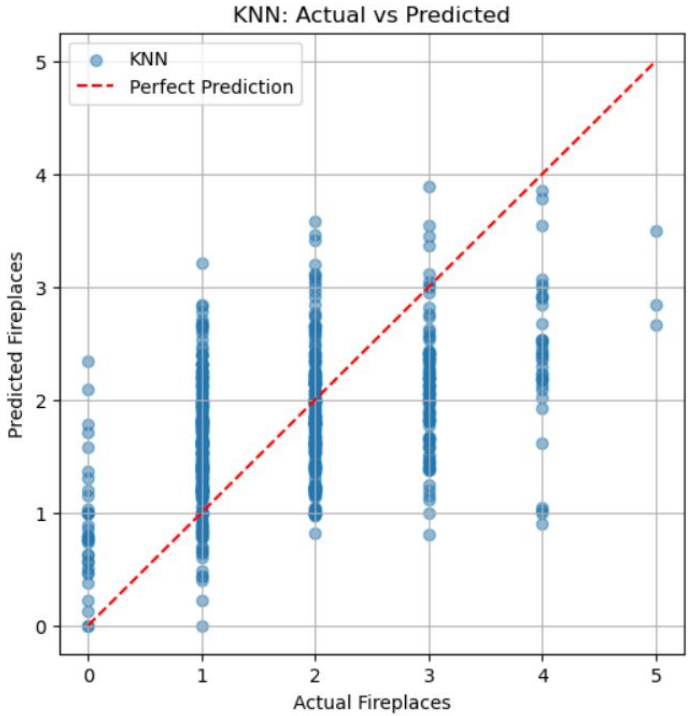
- Assumes a linear relationship between features and fireplaces
- We trained it using gradient descent

# Key Observations

- **KNN Regressor performed better**
  - Lower MAE
  - Positive $R^2$ (closer to 0.5 depending on K)
- **Linear Regression failed**
  - $R^2$ was **negative**
  - Predictions were worse than just guessing the average
- Likely reason: **Fireplace counts are skewed** (mostly 1 or 2) → not ideal for linear models

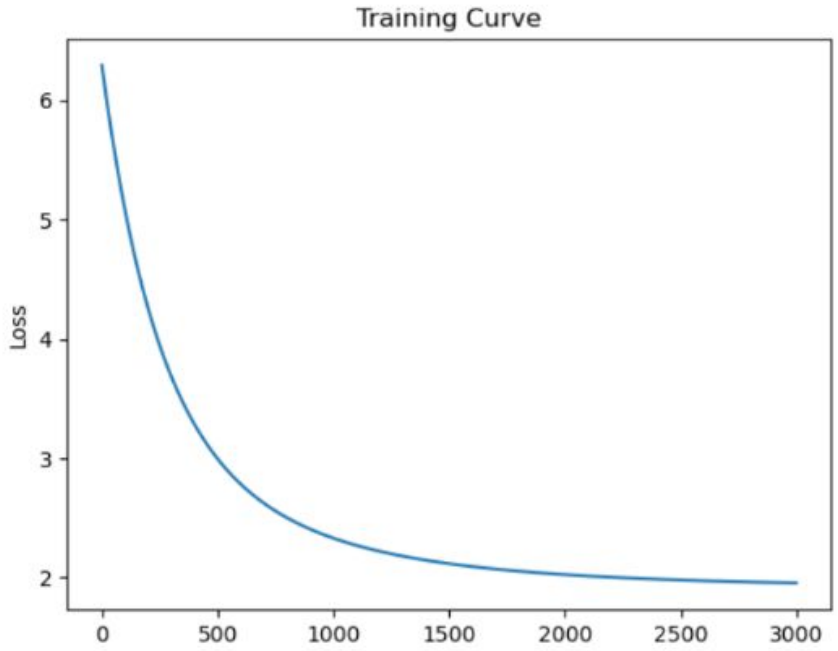
# KNN Regressor - Actual vs Predicted Fireplaces

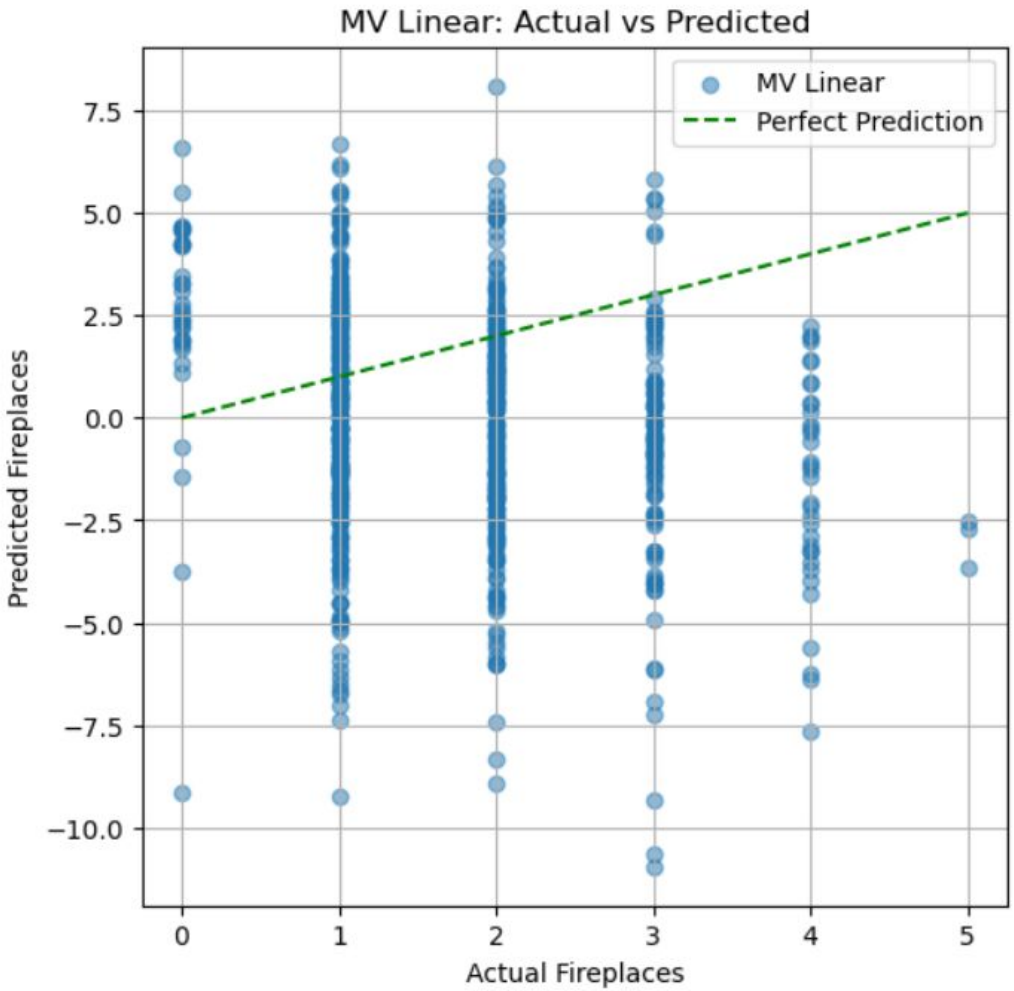


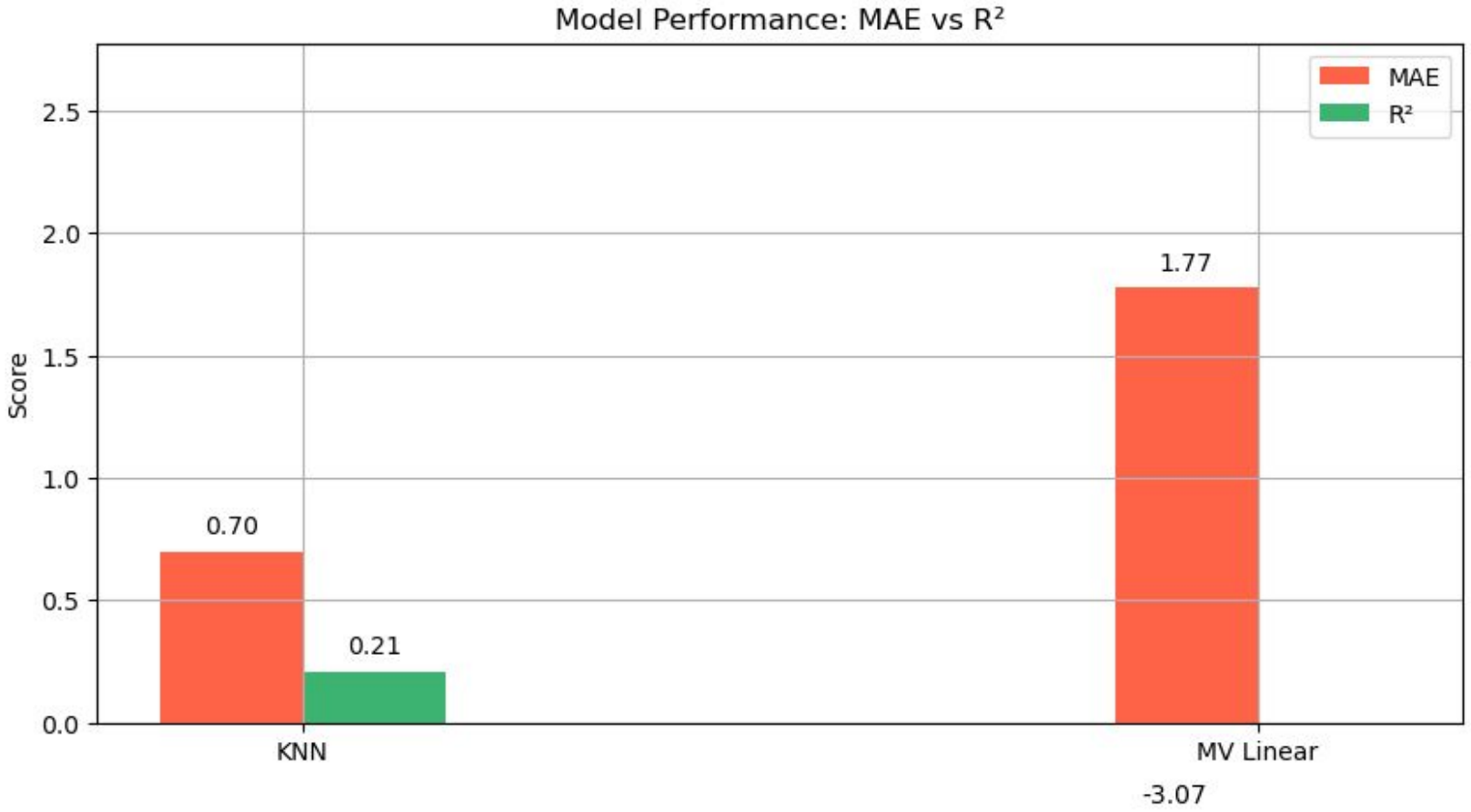



# Train and Predict with MV Linear Regression

```
linreg = MVLinearRegression()
linreg.fit(X_train_scaled, y_train, eta=1e-3, epochs=3000)
y_pred_lin = linreg.predict(X_test_scaled)
```




MV Linear: Actual vs Predicted
MV Linear
Perfect Prediction
Predicted Fireplaces
Actual Fireplaces
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
0
1
2
3
4
5


Model Performance: MAE vs R²
MAE
R²
Score
2.5
2.0
1.5
1.0
0.5
0.0
0.70
0.21
1.77
KNN
MV Linear
-3.07

# Improvements / Future Work

- Try KNN Classifier or Decision Trees (fireplaces = discrete)
- Add more features (e.g., neighborhood quality, presence of chimney)
- Use GridSearch for better K in KNN
- Handle skew with techniques like SMOTE (for classification)
- Try ensemble models like Random Forests for more power

# Final Statement

- Built and compared two custom ML models to predict the number of fireplaces in homes.
- KNN performed better due to the non-linear nature of the data, while MV Linear struggled.

# Thank you!