

# Task 2: Model Optimization, Feature Engineering & Performance Comparison

## 1. Introduction

Machine Learning projects typically require more than a single model training step. Effective ML workflows include preprocessing, feature scaling, model comparison, and the selection of the most suitable algorithm based on objective performance metrics.

This task extends the baseline Linear Regression model built in Task-1 by applying feature scaling, experimenting with multiple algorithms, and evaluating their performance on the California Housing dataset.

## 2. Dataset Description

The **California Housing dataset** contains **20,640 records** collected from the 1990 US Census. It includes demographic, geographic, and housing-related features.

### Selected Features

- Median Income
- House Age
- Average Rooms
- Average Bedrooms
- Population
- Average Occupancy
- Latitude
- Longitude

### Target Variable

- Median House Value (HousePrice)

The dataset has **no missing values**, making it suitable for direct model development.

## 3. Data Preprocessing & Feature Scaling

Before model training, the dataset was divided into input variables (**X**) and the target variable (**y**). Since the features have varying numerical scales—such as population in thousands vs. income in small values—**StandardScaler** was used to normalize the input features.

### Why Scaling Was Necessary

- Prevents large-scale features from dominating the model
- Improves model performance and stability
- Ensures gradient-based models behave correctly

- Promotes fair comparison between different algorithms

Scaling was applied **after splitting** to prevent data leakage.  
Models were trained using an **80:20 train–test split**.

## 4. Model Development

Three regression models were trained and evaluated:

### 1. Linear Regression

A baseline model that fits a linear relationship between features and output.

### 2. Ridge Regression ( $\alpha = 1.0$ )

A regularized linear model that reduces overfitting by penalizing large coefficients.

### 3. Decision Tree Regressor ( $\text{max\_depth} = 5$ )

A non-linear model capable of learning complex interactions among features.

All models were trained on scaled input data for consistency.

## 5. Model Evaluation & Comparison

Each model was evaluated using **RMSE** and **R<sup>2</sup> Score**, which measure prediction error and variance explained by the model.



### Model Performance Table

| Model                   | RMSE     | R <sup>2</sup> Score |
|-------------------------|----------|----------------------|
| Linear Regression       | 0.745581 | 0.575788             |
| Ridge Regression        | 0.745554 | 0.575819             |
| Decision Tree Regressor | 0.724234 | 0.599732             |

### Interpretation of Results

- **Decision Tree Regressor** achieved the **lowest RMSE (0.724234)**, meaning it produced the most accurate predictions.
- It also achieved the **highest R<sup>2</sup> score (0.599732)**, indicating it explained more variance in house prices compared to Linear and Ridge models.
- **Linear and Ridge Regression** performed almost identically with very small differences, showing that scaling and regularization provided limited improvement for linear models.

### Visual Analysis

A scatter plot of **Actual vs Predicted values** for the best-performing model (Decision Tree) shows closer alignment with the diagonal reference line, confirming better predictive accuracy.

## 6. Best Model Justification

Based on objective performance metrics:

### Best Model: Decision Tree Regressor

#### Reasons:

- Lowest RMSE among all tested models
- Highest  $R^2$  Score
- Better ability to capture non-linear patterns in the dataset
- Produces visibly tighter clustering around the reference line in the scatter plot

Thus, the Decision Tree model is selected as the **final optimized model for Task-2**.

## 7. Conclusion

This task demonstrated an industry-standard ML workflow involving preprocessing, feature scaling, multi-model training, and systematic performance comparison.

Compared to Task-1, this task resulted in a more accurate predictive system by:

- Standardizing features
- Evaluating multiple models
- Objectively selecting the best-performing algorithm

The **Decision Tree Regressor** outperformed both Linear and Ridge Regression models by achieving the best overall accuracy. This approach showcases the importance of trying multiple algorithms instead of relying on a single model.

Future improvements can include:

- Hyperparameter tuning
- Cross-validation
- Advanced models such as Random Forest or Gradient Boosting
- Feature importance analysis