Name:

Muhammad Awais

Section:

BSAI-4C

Roll NO:

114

Subject:

PAI-Lab

Task 02

House Price Prediction Model Report

---

1. Introduction

This report details the development of a machine learning model designed to predict house prices based on various features from the House Price dataset. The process includes data preprocessing, feature selection, model training, and evaluation using machine learning techniques. The objective is to build a predictive model that can accurately estimate house prices based on the provided dataset.

---

2. Data Preprocessing

The preprocessing steps aimed to prepare the dataset for training, ensuring that missing values were handled appropriately and the numerical features were scaled correctly.

- **Dataset Used:** House price dataset (`train.csv`).
- **Missing Value Handling:**
  - Columns with missing values were dropped.
  - For numerical columns with missing values, the missing values were replaced with the mean of the respective column to ensure consistency.
- **Feature Scaling:**

    - **StandardScaler** was applied to normalize the numerical values to have a mean of 0 and a standard deviation of 1. This helps improve the performance of the machine learning model by ensuring all numerical features are on the same scale.
- **Feature Selection:**
    - After reviewing the available features, the following columns were selected as relevant for predicting house prices:
        - `LotArea` (size of the lot)
        - `YearBuilt` (year the house was built)
        - `1stFlrSF` (square footage of the first floor)
        - `2ndFlrSF` (square footage of the second floor)
        - `FullBath` (number of full bathrooms)
        - `BedroomAbvGr` (number of bedrooms above grade)
        - `TotRmsAbvGrd` (total rooms above grade)

---

3. Model Training

The Random Forest Regressor algorithm was chosen for model training because of its robustness, flexibility, and good performance on tabular datasets like house prices.

- **Train-Test Split:**
    - The dataset was split into training and testing sets: 80% of the data was used for training, and 20% was reserved for validation.
    - A random state of 42 was used to ensure reproducibility of results.
- **Algorithm Used:** Random Forest Regressor
    - **Hyperparameters:**
        - `n_estimators=100`: The model uses 100 trees in the forest.
        - `random_state=42`: Ensures that the results are consistent across runs.
- **Training Process:**
    - The model was trained using the `RandomForestRegressor.fit(X_train, y_train)` method. This involved fitting the model to the training data, allowing it to learn the relationship between the features and the target variable (house price).

---

4. Model Evaluation

Once the model was trained, it was evaluated using the test dataset to assess its performance. The following evaluation metrics were used to measure the model's effectiveness:

- **Predictions:**
    - The model was used to predict house prices on the test dataset.
- **Performance Metrics:**
    - **Mean Absolute Error (MAE):** Measures the average absolute error in the model's predictions. A lower MAE indicates better predictive accuracy.
    - **Mean Squared Error (MSE):** Similar to MAE but penalizes larger errors more significantly. A lower MSE indicates better performance.
    - **R^2 Score:** Measures how well the model explains the variance in the target variable. A score closer to 1 indicates that the model fits the data well.

- **Evaluation Code:**

```
python
Copy
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"MAE: {mae}")
print(f"MSE: {mse}")
print(f"R²: {r2}")
```

---

5. Results

Unfortunately, the actual dataset (`train.csv`) was not provided, so the model's performance could not be computed directly. However, if the dataset were available and the model were executed, the following results would be generated:

- **MAE (Mean Absolute Error):** Measures the average error in the model's predictions. A lower value indicates better performance.
- **MSE (Mean Squared Error):** A higher penalty is applied for larger errors, so the model's ability to minimize large mistakes is critical.
- **R^2 Score:** Indicates how well the model fits the data, with values closer to 1 representing better fit.

---

6. Conclusion and Next Steps

- **Conclusion:**
  - The model successfully predicts house prices using machine learning techniques.
  - Random Forest was chosen due to its robust performance and ability to handle tabular data, especially with numerous numerical features.
- **Next Steps for Improvement:**
  - **Feature Engineering:** Additional feature engineering, such as adding interaction terms or polynomial features, could help improve the model's predictive power.
  - **Hyperparameter Tuning:** Performing grid search or random search for hyperparameter optimization could further enhance the Random Forest model's performance.
  - **Testing Alternative Models:** Exploring other machine learning models, such as Gradient Boosting or XGBoost, might yield better results.